A Review on Elliptic Curve Cryptography

Maiya Al Saadi
Information Technology Department
Modern College of Business and Science
Muscat, Oman
Maiya_alsaadi@mcbs.edu.om

Basant Kumar
Deptt of Comp Sc
Modern College of Business and Science
Muscat, Sultanate of Oman
basant@mcbs.edu.om

Abstract

Cryptography has many algorithms that are used to secure the communication and make the data exchange between the sender and receiver safe and protected from the attackers. The proposed algorithm will play a vital role to address limitations in Elliptic Curve Cryptography to tackle all issues pertaining to exchange keys. This paper also illustrates a review of ECC encompassed with ECC Algorithm Process, Basic Protocols of ECC Algorithm, Various ECC Implementation and ECC Application. This paper also signifies various graphical representations of cryptographic process on EC over infinite and finite field.

Keywords- *ECC, public key, private key, secret key, ECDSA, ECDH*

I. INTRODUCTION

Security is one of the main concerns in each communication nowadays. Thus, cryptography is used to protect the communication between the communicated senders and receivers. For many centuries, cryptography has been used [1] where It is defined as technique, method or several formulas which provide security to protect data or network to achieve Integrity , Confidentiality, , Non-repudiation, Authentication, Access Control and Availability [2][3] by transforming the message to unreadable form[4]. Since the beginning of public key Cryptography in 1976 by Martin Hellman and Whit Diffie the importance of cryptography was really apparent and the problem of discrete logarithm there were a seriously and well-studied [5]. Cryptography consist of three main categories which are Secret Key, Public Key and Hash Functions. In symmetric or also called private key cryptography, one key is used for both encryption and decryption. But in asymmetric cryptography or public key cryptography, more than one key are used which are public key that cipher or decrypt the transmitted data and the privet key which utilized by the key owner to decrypt the message[6]. The public key cryptography named as public because it made the encryption of the message publicly available[7]. Elliptic Curve Cryptography or called (ECC) is considered as a public key cryptography or it also called asymmetric key cryptography[2]. This paper will present a review of ECC.

II. OVERVIEW OF ECC

Elliptic Curve Cryptography is deemed as a public or asymmetric key cryptography which is proposed in 1985 by both Victor Miller and Neil Koblitz that purely based on mathematical operations[3] [4][8]. The distinction of ECC is that it provide its user low key size and difficult exponential time challenge to the attacker to breach the system or the communication[9]. Since 160 bits of elliptic curve cryptography is equal to 1024 bit of RSA and ratio of 1:7, ECC provides equal security compared with RSA but with less key size [9][10]. Consequently, ECC have low key size, high security and faster cryptographic operations which it provides a compact chips and software [9].In ECC, the cryptosystem requires three algorithm which are: key generation, encryption algorithm and decryption algorithm and that checked through the domain parameters of the elliptic curve [11]. There two keys that should be generated in ECC which are the public key which used the message sender to encode the message and transfer it to ciphertext and the private key which kept with its owner that used to decode the cipher text and convert it to a readable format [8]. ECC is purely founded on the Elliptic Curve Discrete Logarithm problem over finite field [1][2][12] or integer factorization [8]. Form its name elliptic curve, ECC is relay on the general equation:

$y^2= x^3 + ax + b$[3][13][14]

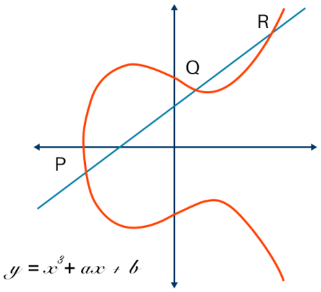

Figure 1Elliptic Curve illustrates the operation P+Q=R [15]

III. ELLIPTIC CURVE CRYPTOGRAPHY OVER FINITE FIELD MATHEMATICS

By using the coordinate points of EC, cryptographic process on EC over finite field is done. The given formula y2 = x3 + ax + b mod p illustrates the Elliptic curve over finite field equation [16][17]. Specific formulas are located for operation with the points which are point adding, point subtraction, point doubling and point multiplication.

A. EC point adding

The two point P(x1,y1) and Q(x2,y2) are distinct and to perform the calculation, the two points are added to generate the point R(x3,y3). Figure 2 shows EC point adding graphical representation

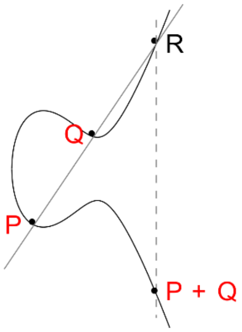

Figure 2 Elliptic Curve Point addition[18]

B. EC point subtration

Point subtraction is performed by getting the reflect coordinate of the subtracted point x and adding the generated coordinate point and the other coordinate [17].

C. EC point Doubling

Doubling the point is done to add two same points that have same coordinate value. Figure 3 shows EC point doubling graphical representation[16] [17].

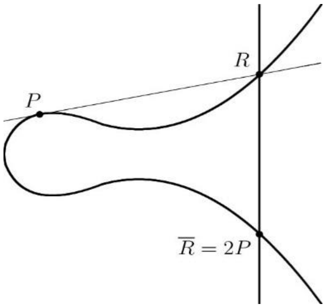

Figure 3Elliptic Curve doubling addition[19]

D. EC point multiplication

In point multiplication, when p is any point in EC, the multiplication is defined by repeating the addition of the same point so that nP= P+P+P...n times[16] [17]

E. Point at infinity

The intersection of the points is at the infinity is represented as O and that happened when the points of x_1 is equal to x_2 and y_1 is equal to y_2 so it equal to 0 or x_1 is equal to x_2 and y_1 is equal to -y_2 [16].

IV. ECC ALGORITHM PROCESS

To apply the ECC algorithm, certain steps should be followed which are

Step 1: the curve is used as the general formula: y^2= x^3 + ax + b where the values a and b are the curve parameters [20][21].

Step 2: two prim numbers should be selected [21]

Step 3: calculate adding and doubling points on the curve[21].

Step 4: take a generating point other than the previous points where its sequence should be large[21].

Step 5: choose a less a prime number than the generating point in order and treat it as a private number for each entity [21].

Step 6: by multiplying the generating number with the secret number, the public key can be generated using the entity. This step will generate the point [21].

As an example of the above process the, below explanation will discuss the key generation, secret key calculation and key exchange. There are two global elliptic curve elements need to be understood which are Eq(a ,b) where q is an integer or a prime number of form 2^m and G point (Generator) whose order is large value n[14].

F. Generation of Key

The first step to start the encryption is generating keys by selecting numbers and multiply it with the generator[14]

- Alice Key generation

Step 1: Select a secret number n_A where $n_A<n$

Step 2: compute the public key $P_A= n_AG$

- Bob Key generation process

Step 1: Select a secret key $n_{\mathbf{B}}$ where $n_{\mathbf{B}} <n$

Step 2: compute the public key $P_{\mathbf{B}} = n_{\mathbf{B}} G$[14]

G. Secret Key Calculation of both Alice and Bob

The calculation of the secret keys can be calculated by:

Alice secret key: $SK_A=P_B\ n_A$

Bob secret key: $SK_B=P_A\ n_B$ [14]

H. Encryption of the message by Alice using public key of Bob

Step 1: the message is choosing by Alice (P_m) and a random positive number K (K should be integer)

Step 2: Calculating the Ciphertext where $C_m= \{KG, P_m + KP_B\}$

I. Decryption of the received ciphertext by Bob using his own Private key

Step 1: Calculating the plaintext where $P_m = P_m +K(n_BG)-n_B(KG)$

J. Key Exchange

Below are the basic steps used by the communicated parties to exchange the key:

Step 1: Alice select a random number d_A

Step 2: Calculate the public key of Alice using the formula:

$P_A(x,y) = d_{A.}G(x,y) = (d_AGx, d_A Gy)$

Step 3: Repeat the above steps (1, 2) to calculate Bob's Public key using the private key (d_B)

Step 4: Alice Secret key calculation

$SK_A(x,y) = d_A . P_B$

Step 5: Calculate the Secret Key of Bob using the operation $SK_B (x,y) = d_B . P_A$

Using the above method, both parties will get the matching secret key [14].

V.BASIC PROTOCOLS OF ECC ALGORITHM

In the key exchange and digital Signature, some general ECC algorithms can be used.

A. Elliptic Curve Digital Signature Algorithm (ECDSA)

Usually, DS is applied to authenticate the machines, or the message is sent from the device so that the message is sent with sign signature to ensure that the message is not tampered by anyone[13][22]. ECDSA is under ANSI and IEEE standards committees[5]. It differs from DSA that operates on EC groups. In order to send a sign message, both communicated parties should agree on the EC domain parameters. As an example, if the parties are Alice and Bob, the sender Alice have a key pair composed of a dA as a private key (random number) and a QA as a public key where QA = G. dA where G is the generated point define on elliptic curve domain parameters [13][22].

B. Elliptic Curve Diffie Hellman Key Exchange (ECDH)

This protocol is used as an agreement to allow the communicated group to initiate a shared secret key which can be applied in the private key algorithm. To start, first the parties exchange some of the public details between them and by using their own private key and the public information, both parties can compute the shared secret key. Apart from those to parties, any third party will face difficulty to compute the shared secret key from the obtainable public information. In order to create a shared secret key between Alice and Bob using this protocol, both parties should have no conflict on EC domain parameters. Both Alice and Bob have a key consisting of a private key d (d_A, d_B) and a public key Q_A= G.d_A for Alice and Q_B= G.d_B for Bob. Thus, the private, public keys of Alice is (d_A, Q_A) and the private, public keys of Bob is (d_B, Q_B). Then, Alice calculate E=(xE,yE)=d_A.Q_B where x and y are domain parameters of EC. On the other hand, Bob calculated F=(xF,yF)=d_B.Q_A. Hence d_AQ_B= d_Ad_BG=d_Bd_AG=d_BQ_A. therefor. E is equal to F and xE =xF. Thus, the shared secret key is xE. Using the above calculations, in practice, it is not possible to discover the private key of Alice and Bob (d_A and d_B) from the public keys E and F. So, it is impossible to obtain the shared secret key by third person[13][22].

VI. VARIOUS ECC IMPLEMENTATION

In the digital world, small size devices play a major nowadays. However, those devices contain limited memory, but they required security as well and that what makes a challenge where the security required appropriate specification to be achieved. Because of the low-key size of ECC, the less operation and small encryption and decryption parameters, ECC considered as the most appropriate cryptography for limited memory devices like Smartphones, palmtop, smartcards. etc. using scalar multiplication, point addition and point doubling[14]. As an example, in Radio Frequency Identification (RFID), ECC was implements and it proved that it provides a genuine security for the commination and data access tagged memory. Besides, it minimizes the required storage for the key and the backend system and that done by storing the private key only. Consequently, it reduces the commutation of the tag[14]. Moreover. ECC is an attractive public key cryptography to be implemented in mobile or wireless environments[23].

VII. ECC APPLICATION

In real world, ECC is applied in different areas and technologies. Below are some of them.

A. Bitcoin

The cryptography of Bitcoin requires the payment to be transfer from one peer to other peer directly without passing through any financial organization. The public block chain in Bitcoin is a group of daily collected transection in Bitcoin. Every block in this group have a hash of SHA-256 of the prior block, so the blocks chaining is starting from the origin block. The user account in Bitcoin is applied as ECDSA private secret key so that the conveying of the ownership of the Bitcoin between the parties is checked by connecting a DS using the private key of sender of the prior hash operation and the public key details of receiver. Thus, the verification of the signature could be applied by using sender's public key from the prior transaction[24].

B. Secure Shell (SSH)

ECC can be applied in SSH protocol in different locations. The authentication between the server and client can be done using host key which help the self-authenticity of the server to the client. At the key exchange time, the server key is sent to the client by the server so that the client checked if the fingerprint resembles the saved value. Then, the server performs a self-authenticity by signing a duplicate of the exchange key and ECDSA can be the key. finally, for client authentication, ECDSA can be used by the clients as public keys [24].

C. Austrian e-ID

As physical security, one of the widely deployed method to grant access to users is a physical smart cards which used to authenticate users. In order to perform the cryptographic computations; the smarts cards are embedded with cryptographic hardware modules so that it contains a private key used for encryption and signatures. Because of the small key size and less computational complexity of ECC, ECC is considered as attractive option to be implemented in this kind of application. To provide legally binding digital signatures, ECDSA public key cab be used[24].

D. Transport Layer Security (TLS)

EC can apply in different positions in the protocol. The ECDH key exchange is used in all cipher suites set in RFC. The certificates of TLS contain a public key used by the server to authenticate itself and ECDSA can be this public key. TLS added ECC in the client and server hello messages and through an additional set of cipher suites and. The suites point a support for an election of key exchange, encryption, authentication algorithms of the message and identity varication. Instead of sending the entire choices of curves or cipher suites that are used by the sever, list of supported elliptic curves and cipher suites are sent by the client, and the server have to option which are replying with a one cipher suite from the sent list otherwise the connection is terminated if it does not boost any cipher suites. In case ECC is needed by the suite, only one curve type with a key or signature is included in the server. This complicate identifying the curves that the server supports, a client must use various TLS connections to propose a varying set of curves to learn ordered preference and server's support[24].

VIII. CONCLUSIONS

Elliptic Curve Cryptography is one of the vastly used asymmetric or a public key cryptography where there are three main algorithms used in ECC which are: key generation, encryption and decryption. Because of its small size and complicated algorithm, it increases the level of security so that it makes the communication almost impossible to breach by any third parties or attackers.

REFERENCES

[1] V. S. Abraham, “E lli p t ic C ur ve C r y p t og r a ph y,” vol. 9, no. 20, pp. 1–8, 2008.

[2] R. Bhanot and R. Hans, “A review and comparative analysis of various encryption algorithms,” *Int. J. Secur. its Appl.*, vol. 9, no. 4, pp. 289–306, 2015.

[3] A. K. Singh, “A Review of Elliptic Curve based Signcryption Schemes,” *Int. J. Comput. Appl.*, vol. 102, no. 6, p. 8887, 2014.

[4] K. Vasundhara, Y. V S Sai Pragathi, and Y. Sai Krishna Vaideek, “A Comparative Study of RSA and ECC,” *Int. Journal of Engineering Research and Application www.ijera.com*, vol. 8, no. 1. pp. 49–52, 2018.

[5] A. J. M. Silverman, Joseph H.Aleksandar Jurisic, “Elliptic curves and cryptography.” pp. 91–112, 2005.

[6] G. C. Kessler, “An Overview of Cryptography (Updated Version,” vol. 1998, no. January, pp. 1–65, 2019.

[7] J. L. Vagle, “A Gentle Introduction to Elliptic Curve Cryptography,” *BBN Technol.*, 2000.

[8] Sujith Narayan, “A REVIEW ON ELLIPTIC CURVE,” *Int. J. Emerg. Technol. Innov. Eng.*, vol. 4, no. 12, pp. 132–138, 2018.

[9] T. N. Shankar and G. Sahoo, “Cryptography with elliptic curves,” vol. 2, no. 1, pp. 38–42, 2009.

[10] M. Bafandehkar, S. M. Yasin, R. Mahmod, and Z. M. Hanapi, “Comparison of ECC and RSA algorithm in resource constrained devices,” *2013 Int. Conf. IT Converg. Secur. ICITCS 2013*, pp. 0–2, 2013.

[11] S. A. Mailov Arif , Abbasov Habib , Isayev Rufat, “Study and Implementation of Elliptic Curve Encryption Algorithm for Azerbaijan E-ID,” pp. 3708–3713, 2015.

[12] V. S. Kumari Archana, “Comparative Analysis of RSA and ECC,” *Int. J. Innov. Res. Comput. Commun. Eng. (An ISO Certif. Organ.*, vol. 3, no. 7, pp. 1–5, 2015.

[13] A. MS, “Elliptic curve cryptography,” *Pract. Cryptogr. Algorithms Implementations Using C++*, pp. 147–181, 2014.

[14] D. Mahto and D. K. Yadav, “Performance Analysis of RSA and Elliptic Curve Cryptography,” *International Journal of Network Security*, vol. 20, no. 4. pp. 625–635, 2018.

[15] “What is Elliptic Curve Cryptography_ Definition & FAQs _ Avi Networks.” .

[16] S. Vasundhara, “The Advantages of Elliptic Curve Cryptography for Security,” *Glob. J. Pure Appl. Math.*, vol. 13, no. 9, pp. 4995–5011, 2017.

[17] L. D. Singh and K. M. Singh, “Image Encryption using Elliptic Curve Cryptography,” *Procedia Comput. Sci.*, vol. 54, pp. 472–481, 2015.

[18] “Elliptic Curves mainpage _ Mathematical Institute.” .

[19] A. Chmielowiec, “Elliptic curve cryptography in small devices,” no. December, 2000.

[20] V. S. A. Vivek Kapoor, Ramesh Singh, “Elliptic Curve Cryptography,” vol. 9, no. 20, pp. 1–8, 2008.

[21] V. B. Kute, P. R. Paradhi, and G. R. Bamnote, “A software comparison of RSA and ECC,” *Int. J. Comput. Sci. Appl.*, vol. 2, no. 1, pp. 61–65, 2009.

[22] R. Afreen and S. C. Mehrotra, “A review on elliptic curve cryptography for embedded systems,” *Int. J. Comput. Sci. Inf. Technol.*, vol. 3, no. 3, pp. 84–103, 2011.

[23] N. Muyinda, “Elliptic curve cryptography,” *African Inst. Math. Sci.*, no. August, pp. 147–181, 2014.

[24] J. W. Bos, J. A. Halderman, N. Heninger, J. Moore, M. Naehrig, and E. Wustrow, “Elliptic curve cryptography in practice,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8437, pp. 157–175, 2014.