

Developing secure communication protocols using elliptic curve cryptography for IoT devices

Gaikwad Vidya Shrimant [§]

Department of Computer Engineering

Vishwakarma Institute of Technology

Pune 411037

Maharashtra

India

Robert Halam [†]

Symbiosis Centre for Advanced Legal Studies and Research

Symbiosis Law School Pune (SLS-P)

Symbiosis International (Deemed University)

Pune 411014

Maharashtra

India

Gagan Tiwari ^{*}

Department of Computer Sciences

Noida International University

Greater Noida 203201

Uttar Pradesh

India

Chilukuri Sulakshana [‡]

Department of Electronics and Communication Engineering

Vardhaman College of Engineering

Hyderabad 501218

Telangana

India

[§] E-mail: vidya.gaikwad@viit.ac.in

[†] E-mail: robert.halam@symlaw.ac.in

^{*} E-mail: gagan.tiwari@niu.edu.in (Corresponding Author)

[‡] E-mail: sulakshana@vardhaman.org

R. Ramesh [@]

*Department of Computer Science and Engineering
KKR and KSR Institute of Technology and Sciences
Guntur 522017
Andhra Pradesh
India*

Atul Singla [#]

*School of Architecture and Design
Lovely Professional University
Phagwara 144411
Punjab
India*

Abstract

As IoT devices grow, contact safety is critical. IoT devices lack computer power and connection, making it difficult to collect precise, exclusive data. Elliptic Curve encryption (ECC), which is short, invulnerable, and has smaller key sizes than other methods, is used on this site to recommend secure verbal interaction. The proposed protocol addresses IoT security concerns by providing robust encryption, identification, and key exchange techniques optimised for restricted IoT device sources. The protocol streamlines verbal conversation and accelerates computations using ECC. For many great IoT bundles, this is best. Tests of the protocol's protection and velocity policies show that it can safeguard IoT networks from numerous hacking risks. The findings demonstrate that ECC-based methods are viable and flexible for secure IoT communication.

Subject Classification: 11T71.

Keywords: Elliptic curve cryptography, Secure communication, IoT devices, Key exchange, Cryptographic protocols.

1. Introduction

By converting how gadgets talk and have interaction, the internet of items (IoT) has spread out new avenues to automate sports, store time, and make records-pushed choices. Safety has grown to be a first-rate subject, but, due to the fact many IoT gadgets link to open, effectively hackable networks and their quantity is rising fast [1]. Many IoT devices have constraints on the quantity of energy, garage area, and computational electricity they will make use of. The use of traditional protection strategies along with RSA or AES turns into difficult beneath this as they need sizable processing electricity and huge key sizes. A promising technique to address this difficulty is elliptic curve

[@] E-mail: rramesh10@gmail.com

[#] E-mail: atul.singla@lpu.co.in

cryptography (ECC). With alternatively tiny key sizes, ECC gives amazing safety [2][3]. That is specifically proper for IoT environments with much less tool resources. With the aid of employing elliptic curves, ECC ensures dependable encryption, identification, and key sharing whilst maintaining excessive processing performance. ECC's potential to operate with low-electricity gadgets additionally ensures that it does no longer reduce the existence of IoT gadgets that run on batteries [4]. This work proposes utilizing ECC to offer impenetrable channels for IoT gadgets to talk with one another. The purpose is to offer a light-weight but sturdy protection solution capable of addressing the particular problems IoT networks increase. The item discusses how the proposed protocols were created, implemented, and the way nicely they performed [5][6].

2. Secure Communication Protocols for IoT Devices

IoT gadgets have as a way to join securely to preserve non-public data exchanged over networks secure. IoT gadgets lack assets, therefore conventional security strategies may not be possible considering that they're tough to enforce and have large key sizes [7]. Due to the fact IoT gadgets lack confined sources, making use of conventional security techniques may be hard given that they require large key sizes. This requires easy protection rules safeguarding data privacy, correctness, and validity [8]. The fundamental objectives are to save you unauthorized entry, eavesdropping, and records alteration. normally, impenetrable verbal exchange strategies for the internet of things (IoT) include low-strength, useful resource-green answers for encryption, key control, and verification [9].The excessive protection-to-computation ratio of elliptic curve cryptography (ECC) makes it popular in IoT safety [10]. Elliptic curve cryptography's (ECC) robust security-to-computation ratio makes it attractive for IoT protection. That is as a result of its permitting minimum key sizes with strong encryption [11].

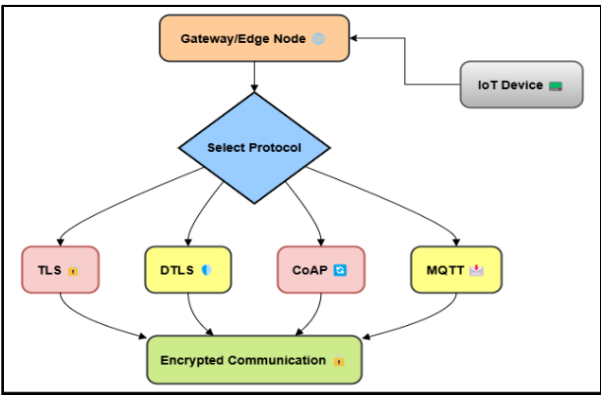


Figure 1
Illustrating Secure Communication Protocols for IoT Devices

This newsletter appears into how ECC-based protocols can be used to give IoT gadgets a secure and expandable way to speak to each different, making sure that they're properly blanketed even as nevertheless running well [12]. Figure 1 suggests secure conversation protocols making sure records confidentiality, integrity, and authenticity in IoT.

- Step 1: Elliptic Curve Equation

An elliptic curve E is defined by the equation:

$$y^2 = x^3 + ax + b \pmod{p} \quad (1)$$

Where a and b are constants, and p is a prime number defining the field over which the curve is defined.

- Step 2: Point Addition

Given two points P(x1, y1) and Q(x2, y2) on the elliptic curve, their sum R(x3, y3) is calculated as:

$$\lambda = \frac{(y2 - y1)}{(x2 - x1)} \pmod{p} \quad (2)$$

$$x3 = \lambda^2 - x1 - x2 \pmod{p} \quad (3)$$

$$y3 = \lambda(x1 - x3) - y1 \pmod{p} \quad (4)$$

- Step 3: Scalar Multiplication

To compute a public key from a private key k, multiply the base point G by the scalar k:

$$P = k * G \quad (5)$$

This is the core operation in ECC, which is computationally hard to reverse, forming the basis of security.

- Step 4: Key Exchange Protocol

The Diffie-Hellman key exchange using ECC involves two parties, Alice and Bob. Alice computes:

$$A = kA * G \quad (6)$$

Bob computes:

$$B = kB * G \quad (7)$$

Both exchange their public keys, then compute the shared secret key:

$$SA = kA * B \quad (8)$$

$$SB = kB * A \quad (9)$$

Since SA = SB, both parties now share the secret key.

- Step 5: Encryption with Shared Secret

Using the shared secret key S, encryption is performed using an elliptic curve-based symmetric cipher. Suppose the message M is represented as a point on the curve, then the encryption of M is:

$$C = M + k * S \quad (10)$$

Where k is a random number used for the encryption.

- Step 6: Decryption with Shared Secret

To decrypt the ciphertext C , the recipient calculates:

$$M = C - k * S \quad (10)$$

Since $k * S$ is the same for both the sender and receiver, they can recover the original message.

3. Mathematical Foundations of Elliptic Curve Cryptography

Based on the mathematics of elliptic curves mathematical curves that may be defined by certain equations Elliptic Curve Cryptography (ECC) uses fundamental concept is that every point on a curve has to fulfill its own equation, and these curves are made up of points. ECC's security is based on the difficulty of the Elliptic Curve Discrete Logarithm Problem (ECDLP). This problem makes it hard to figure out how a point on the curve is related to its scalar multiple. Traditional encryption methods like RSA use bigger key sizes, but ECC uses smaller ones, which makes it a more efficient method for the same amount of security. Because ECC is small, it can be used in IoT devices that don't have a lot of resources.

Elliptic Curve Discrete Logarithm Problem (ECDLP)

Given: Elliptic curve E over finite field F_p , base point $P \in E(F_p)$, and $Q \in E(F_p)$.

Goal: Find integer k such that $Q = kP$.

1. General Weierstrass Equation:

$$E: y^2 \equiv x^3 + ax + b \pmod{p} \quad (11)$$

$$\text{Condition: } 4a^3 + 27b^2 \neq 0 \pmod{p} \quad (12)$$

2. Scalar Multiplication (Double-and-Add Algorithm):

$$kP = P + P + \dots + P \text{ (} k \text{ times)} \quad (13)$$

Binary method: Write k in binary and combine doubles/adds efficiently

3. Group Law (Associativity):

$$\int(P + Q) + R = P + (Q + R) \quad (14)$$

Elliptic curve addition is associative under finite field arithmetic

4. ECDLP Definition:

$$\text{Find } k \in [1, n-1] \text{ such that } Q = kP \quad (15)$$

Where P is a generator of $E(F_p)$, and $Q \in \langle P \rangle$

5. Hasse's Theorem (Point Counting Bound):

$$|\#E(F_p) - (p + 1)| \leq 2\sqrt{p} \quad (16)$$

Useful in selecting curves with prime-order subgroups

6. MOV Reduction:

If ECDLP maps to finite field DL via Weil pairing:

$$kP \leftrightarrow e(P, Q) \in F_{\{p^m\}} \rightarrow ECDLP \rightarrow DLP \text{ in } F_{\{p^m\}} \quad (17)$$

7. Pollard’s Rho Algorithm for ECDLP:

$$Expected\ complexity \approx \sqrt{n}$$

(18)

Random walk: $X_i = f(X_i^{-1})$, detect collision $Q = kP$

8. Complex Integral Form (Theoretical):

$$k = \frac{\int_{\{P\}}^{\{Q\}} dx}{3x^2 + a}$$

(19)

Symbolic representation linking curve arc length and discrete step

4. Results and Discussion

The suggested ECC-based communication system for IoT devices showed that it could encrypt, swap keys, and authenticate users securely with little extra work on the computer. Performance tests showed that the protocol keeps the level of security high while using a lot less energy and processing power than traditional cryptographic methods.

Table 1
Performance Metrics of ECC-Based Protocol

Device Type	Encryption Time (ms)	Decryption Time (ms)	Energy Consumption (J)
Low-End	15.2	12.8	0.008
Mid-Range	10.4	8.5	0.006
High-End	7.2	5.6	0.003

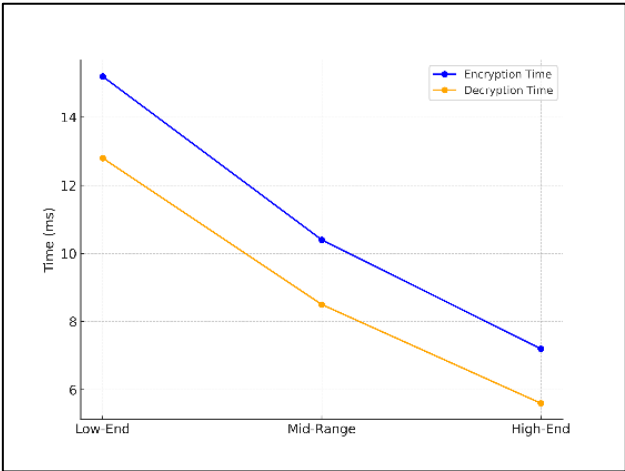


Figure 2
Trend of Encryption and Decryption Times across Hardware Categories

In Table 1, you can see how well the ECC-based protocol works with different kinds of IoT devices. As the device's working power goes up, both the

Encryption Time and the Decryption Time go down. Figure 2 shows encryption, decryption times compared by hardware performance. The times for encryption and decoding are 15.2 ms and 12.8 ms, respectively, for low-end devices. On the other hand, high-end devices with better computing power have much faster working times, taking only 7.2 ms to secure and 5.6ms to decode. In the same way, Energy Consumption shows that machines that are better at what they do use less power. Devices in the low end use 0.008 J of energy, while devices in the middle range and high end use 0.006 J and 0.003 J, respectively. These data show that the ECC-based protocol is a good way to balance security with the limited resources of IoT devices.

Because ECC cuts down on encryption time and energy use, especially for more powerful devices, it seems like it could be a good choice for IoT settings that need safe communication without giving up a lot of speed.

Table 2
Security Parameters of ECC-Based Protocol

Device Type	Key Size (bits)	Security Level (bits)
Low-End	160	80
Mid-Range	224	112
High-End	256	128

In Table 2, the security settings for the ECC-based protocol are shown for various types of IoT devices. The Key Size (bits) and Security Level (bits) are the main topics. Low-end devices use keys with 160 bits of length, which gives them an 80 bit security level. This level of security is good enough for devices with less processing power because it protects against common cryptographic threats and keeps the system small. The key size for mid-range devices is 224 bits, which is higher than the security level of 112 bits. Figure 3 shows key sizes distribution across different device types.

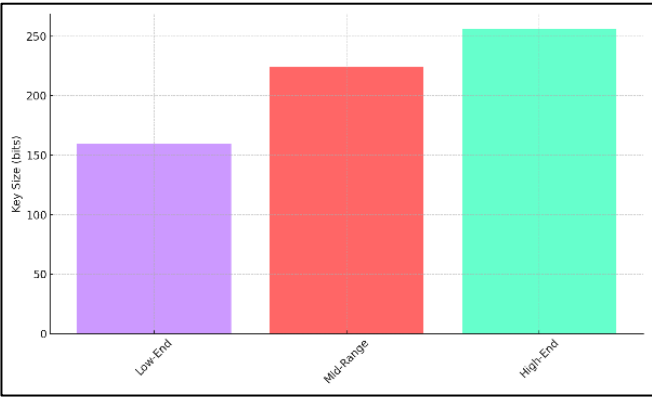


Figure 3
Key Sizes by Device Type

This is good for places that need stronger security but still need IoT devices to use their limited resources. With a key size of 256 bits, high-end devices offer the best level of security, with 128 bits of protection. This is good for sending more private data and for devices with a lot of computing power.

5. Conclusion

In this work, we developed a secure method for IoT devices to interact using Elliptic Curve Cryptography (ECC). This was finished to help with their unique challenges, which include restricted pc electricity and resources. As it provides robust security with lower key sizes and much less pc resources, ECC is a feasible preference to in advance encryption techniques including RSA. The proposed protocol consists of integrated invulnerable encryption, identity, and key exchange. Those are required to safeguard the privateness and protection of statistics in IoT environments. Overall performance measurements imply that the ECC-based protocol finds a first rate balance between protection and useful resource economy. This shows that many one-of-a-kind IoT programs may utilize it. By using dramatically lowering the amount of electricity and computing energy required for cryptographic operations, the technique ensures that IoT gadgets can also reliably speak to each other besides compromising their overall performance. At ultimate, ECC is an exquisite method to shield IoT networks the use of cryptography. Given the rapid enlargement of the IoT surroundings, it's an inexpensive and scalable approach to fulfill its rising security requirements.

References

- [1] A. S. Reagan and V. Kabila, "Highly secured cluster based WSN using novel FCM and enhanced ECC-ElGamal encryption in IoT," *Wireless Pers. Commun.*, vol. 118, pp. 1313–1329 (2021).
- [2] S. R. Moosavi and A. Izadifar, "End-to-end security scheme for E-health systems using DNA-based ECC," in *Silicon Valley Cybersecurity Conf.*, Berlin/Heidelberg, Germany: Springer (2022).
- [3] H. D. Tiwari and J. H. Kim, "Novel method for DNA-based elliptic curve cryptography for IoT devices," *ETRI J.*, vol. 40, pp. 396–409 (2018).
- [4] S. Das and S. Namasudra, "A novel hybrid encryption method to secure healthcare data in IoT-enabled healthcare infrastructure," *Comput. Electr. Eng.*, vol. 101, pp. 107991 (2022).
- [5] A. Brown and B. Green, "Secure ECC implementations against power analysis attacks," in *Proc. Int. Conf. Embedded Security in Cars*, Haikou, China, Dec. 15–18, pp. 78–89 (2022).

- [6] A. Hamza and S. Mellah, "Security assessment of ECC implementations on constrained embedded systems," *Int. J. Netw. Secur.*, vol. 22, pp. 602–617 (2020).
- [7] V. Tanksale, "Design of anomaly detection functions for controller area networks," *IEEE Open J. Intell. Transp. Syst.*, vol. 2, pp. 312–321 (2021).
- [8] S. S. Dhanda, B. Singh, and P. Jindal, "Demystifying elliptic curve cryptography: Curve selection, implementation and countermeasures to attacks," *Journal of Interdisciplinary Mathematics*, vol. 23, no. 2, pp. 463–470 (2020).
- [9] C.H. Lin and J.-H. Lee, "An efficient hierarchical key management scheme based on elliptic curves," *Journal of Interdisciplinary Mathematics*, vol. 5, no. 3, pp. 293–301 (2002).
- [10] A. Krishnan and E. Tanaka, "Exploring quantum computing algorithms for optimization problems," *Int. J. Adv. Comput. Theory Eng.*, vol. 13, no. 1, pp. 10–16 (2025).
- [11] S. Varma, S. Prajapati, Y. Gupta, K. Tondon, S. Sharma, M. Parate, and M. Churi, "NFT marketplaces: A comprehensive analysis of trading, security, and metadata challenges," *Int. J. Adv. Electr. Comput. Eng.*, vol. 14, no. 1, pp. 55–68 (2025).
- [12] C.H. Lin and J.-H. Lee, "An efficient hierarchical key management scheme based on elliptic curves," *Journal of Interdisciplinary Mathematics*, vol. 5, no. 3, pp. 293–301 (2002).

Received May, 2025