Int. J. Nonlinear Anal. Appl. 16 (2025) 4, 93–102
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2023.32465.4831

Lightweight secure message transfer protocol based on ECC (Nist P-256) Internet of Things equipped with satellite communications

Mahdi Baghaei Jezehei, Seyed Ahmad Olamaei*, Ali Broumandnia

Department of Electrical Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran

(Communicated by Seyed Hossein Siadati)

Abstract

With the expansion of Internet of Things (IoT) services and the use of satellite communications, according to the regional or continental extent of these services, the need for lightweight encryption has increased. In satellite communications, security cannot be fully implemented given the long transmission distances, rendering heavy encryption algorithms, such as RSA, unreliable. ECC using mathematical solutions and elliptic curve discrete logarithm problems (ECDLP) can be considered a lightweight algorithm in telecommunications. Here, we propose a new strategy for secure IOT data communication between a satellite link and a terrestrial link that uses the principles of ECC elliptic curve cryptography and the NIST P-256 standard for key agreement and encryption for transmitting messages over the satellite communication platform.

Keywords: elliptic curve cryptography, internet of things, cryptographic protocols, satellite and information security, satellite communications
2020 MSC: 94A60, 68M11

1 Introduction

Satellite Internet-of-Things (S-IOT) communications are growing as an important part of Internet-of-Things (IOT) services. This technology can be used in many space applications such as drone control, industrial control, medical cases, etc. During the provision of S-IOT services, security is fragile owing to the long transmission distance between the source and destination. In addition, authentication mechanisms must be selected considering the low capacity of IoT devices, aiming to minimize power consumption, processing burden, and delays when sending and receiving messages. There is a need for encryption to ensure data safety, so the cryptographic algorithm used in satellite communication must be complex, low-power, and overall lightweight. Attacks on the Internet of Things equipped with satellite communication are conceivable [15]. Usually, asymmetric algorithms such as the RSA algorithm (which is based on an integer factorization problem) and DSA (which is based on a discrete logarithm problem) are used for data transmission in terrestrial communication. As a result, to establish relative security in a terrestrial connection, the key

*Corresponding author
Email addresses: baghaee90@gmail.com (Mahdi Baghaei Jezehei), sa_olamaee@azad.ac.iIr (Seyed Ahmad Olamaei), broumandnia@gmail.com (Ali Broumandnia)
Digital Signature Algorithm

Received: October 2023 *Accepted:* December 2023

size for both RSA and DSA algorithms is recommended to be at least 2048 bits. Consequently, systems implementing these algorithms use long keys and perform several calculations. Due to limited resources in S-IOT devices, there is a need for a lightweight encryption algorithm [16]. Elliptic curve cryptography (ECC) provides a lightweight port function based on the elliptic curve discrete logarithm problem (ECDLP). The key size in the ECC algorithm is substantially smaller than that in other encryption algorithms such as RSA. Elliptic curve cryptography is a public key encryption method that uses smaller keys for encryption than other encryption techniques that use relatively larger keys. As a result, the keys used for ECC are much smaller compared to the keys used by the alternatives. ECDSA is a popular method used in many applications for authorization and user identification [2], but the proper exploitation of the ECDSA standard in satellite communication requires changes and improvements. to avoid the possibility of revealing the private key when two communication links are connected.

Here, using a random selection of integers, using the NIST P-256 standard, and improving the efficiency of the ECC encryption algorithm in satellite communications, the proposed algorithm can create higher reliability for authentication. Whenever the random integer key is reused, it resists MITM attacks [10, 12, 5].

2 LEO Orbit Communication Satellites

Satellites are moving around the earth in a closed path, which is called an orbit. Generally, satellites are placed on four types of orbits that depend on the type of satellite application:

- LEO Low Earth Orbit
- The Polar orbit POLAR
- GEO Earth Station Orbit
- Elliptical orbit

In this article, to improve security in satellite communications, we use the LEO satellite circuit, which orbits at heights of 300–1200 km from west to east, following the Earth's rotation. The time for one revolution around the earth in these orbits is about 90. These orbits are located at a relatively low altitude, as a result, relatively heavy objects can be placed in those orbits with a simple launcher system. These orbits are usually used for observation, satellite communication, and military satellite activities. Due to the close distance of these types of satellites from the earth's surface, the movement speed of these satellites is much higher than the speed of the earth's rotation around itself; sometimes their speed reaches 27,000 kilometres per hour. Advantages of Using Leo Orbits: Satellites require lower energy when deployed to LEO orbits than to other orbits. Among its other advantages is the provision of high data bandwidth and low communication delay. The LEO orbits many communication services, such as the Iridium phone system. Some communication satellites use GEO geocentric orbit geographic station orbits, which move at a speed equal to the speed of the Earth and are always in the same area. And they have a higher delay.

The proposed satellite communication networks use LEO low earth orbit constellations. The Satellites in GEO orbit have a high propagation delay, but a few satellites are enough to communicate around the world. The Satellites in LEO orbit have less propagation delay due to their lower altitude, but many satellites are needed to provide global service. GSO satellites in geostationary orbit have a propagation delay of about 500 milliseconds, and satellites in LEO orbit have a delay of 50 milliseconds. The specifications of satellites deployed to three main orbits are listed in Table 1 [4, 7, 19].

Table 1: Characteristics of spatial layers

Circuit type	The altitude of the orbit (km)	The number of satellites required to cover the entire earth	Timer (milliseconds)
Geostationary (GEO)	36000	3	500
Middle Earth Orbit (MEO)	5000 to 20000	6	80
Low Earth Orbit (LEO)	300 to 1200	100	50

Elliptic Curve Digital Signature Algorithm
Man-In-The-Middle

3 Problem Statement

In general, encryption in satellite communication requires the use of lightweight encryption due to limitations such as long distances and high latency. An elliptic curve enables lightweight algorithms with a shorter key length than other asymmetric algorithms, such as RSA, while guaranteeing a higher security level. This algorithm emphasizes secure management services and advanced authentication. Advanced behavior allows them to generate unique sequences that are in no way inferior to modern encryption programs. In S-IOT communication, several security-related prerequisites need to be addressed. A set of measures to secure the message, such as authentication, confidentiality, integrity, and data availability. Data encryption and decryption using the proposed algorithm in elliptic curve encryption with the key exchange between the sender and the receiver can play an essential role in improving the confidentiality of the message and secure transmission of information exchanged in satellite communications.

4 Mathematical Background and Assumptions: Elliptic Curve Encryption (ECC)

Encryption is the transformation of a simple message into an encrypted form to make it impenetrable and undetectable to intruders. In 1985, Miller [8] independently described Victor S. Elliptic curve cryptography, ECC. Elliptic curve asymmetric encryption is a method for public key encryption between the sender and the receiver of the message.

ECC with a smaller key compared to other asymmetric cryptography such as RSA can provide equivalent security with a smaller key. ECC has many advantages. The elliptic curve cryptographic algorithm is widely used in telecommunication equipment such as satellite communication due to the short key length and less processing. Algorithms such as RSA due to the larger key length have a lower speed and cause delays in communication. In the elliptic curve algorithm, there are different curves, each of which creates a different level of security. In this article, using the curve of equation (4.1) and the P-256 standard, we propose a solution to improve security and reduce delays in satellite communications.

$$y^2 = x^3 + ax + b \tag{4.1}$$

where a and b are constants.

$$4a^3 + 27b^2 \neq 0.$$

Calculations in elliptic curve cryptography are for finite fields or Galva fields. Public key cryptography is based on not solving specific mathematical problems.

$$y2 = \{x^3 + ax + b\} \bmod \{p\}.$$

ECC requires a small key that reduces storage and transmission requirements. By using an elliptic curve, stronger security with a small key and reduced delay compared with asymmetric cryptography algorithms (for example, RSA) can be achieved [18, 17].

5 Secure Key Exchange

Key exchange is a solution for encryption between two links, using which the encryption keys are exchanged between the two parties. As a result, the sender and receiver of the message can use an encryption algorithm.

The nature of the equipment they need depends on the encryption technique they may use. If they use the same code, they both need a copy of the same codebook. If they use a password, they need the appropriate keys. If the cipher is a symmetric key cipher, each pair requires a copy of the same key. If it is a key asymmetric encryption with the public/private key attribute, both require another public key.

The key cannot be sent via normal methods because the files sent between the two parties may end up in the wrong hands and thus be decrypted. Therefore, an alternative method should be easy to use, safe, and highly scalable. It should also be designed for fast, connected, but highly insecure Internet highways. Otherwise, it would not be suitable for commercial use, as sensitive and high-volume transactions are often made daily or hourly over very large intervals. There are different ways to send and receive keys, which can be mentioned below.

- Key exchange with SSL (Secure Sockets Layer)
- Diffie-Hellman key exchange
- key exchange QKD (Quantum Key Distribution)

Each of the above methods has advantages and disadvantages that can have a significant impact on creating security. The point that is important here regarding the S-IOT connection, considering the mentioned limitations, one should be careful in choosing the key that, in addition to solving the concerns in the satellite link, also has the minimum delay in sending and receiving information [1, 13, 3, 6].

6 Analysis of Standard Elliptic Curve ECC (NIST P-256)

As mentioned, there are different curve standards in the elliptic curve encryption algorithm, some of them are mentioned below. In this article, the NIST P-256 standard is considered

1. Secp256r1
2. M-511 Curve
3. Brain pool P256t1 Curve
4. Nist P-224 Curve
5. Secp256k1
6. Nist P-256 Curve
7. M221 Curve
8. Curve 25519 Curve
9. Bn (2,254) Curve
10. Nist P-384 Curve

Each selected curve has different field sizes which are nothing but ECC key sizes. The selected curves are analyzed by performing two algorithms used in ECC, namely ECDH, and ECDSA.

A. Standard Nist P-256

NIST P-256 is one of the elliptic cryptographic curves recognized by the Institute of Standards (NIST). It is also known as secp256r1 or prime256v1. The curve is defined on the finite field of the first order $p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$ and its equation is $y^2 = x^3 - 3x + b$ where b is the y coordinate of the base point of the G base point with NIST coordinates P256 is widely used in various cryptographic protocols and applications such as SSL/TLS, SSH. With a key size of 256 bits and relatively fast cryptographic operations, it offers a good balance between security and performance. The initial curve of the standard used in this article is NIST P-256 whose equation is shown below Standard Nist P-256:

1. $y^2 = x^3 - 3x + 41058363725152142129326129780047268409114441015993725554835256314039467401291$
2. modulo $p = 2^{256} - 2^{224} - 2^{192} + 2^{96} - 1$
3. $EP(a, b) = EP(-3, 41058363725152142129326129780047268409114441015993725554835256314039467401291)$
   With the generating point:
4. $G = (G_X, G_Y)$
   $G_X = 48439561293906451759052585252797914202762949526041747995844080717082404635286$
   $G_Y = 36134250956749795798585127919587881956611106672985015071877198253568414405109$

B. Comparison of standard ECC(P-256) compared to RSA

P-256 encryption has the following advantages over RSA encryption [2]:

1. Short key: The P-256 standard, as it is known, has a key length of 256 bits and can provide equal security to other asymmetric algorithms that have a larger key length.
2. Faster cryptographic operations: ECC algorithms such as P-256 can perform cryptographic operations such as encryption, decryption, and signing faster than RSA for the same level of security. This is because the math operations used in ECC are simpler and faster than those used in RSA.

3. Lower energy consumption: The smaller key size and faster encryption operation of P-256 results in lower energy consumption in devices that use it. This makes the P-256 a good choice for applications that require low power consumption, such as mobile devices and IOT devices [11].
4. Resistance to certain types of attacks: ECC algorithms such as P-256 are resistant to certain types of attacks, such as attacks based on number field sifting algorithms that can be used to break RSA. This makes P-256 a good choice for applications that need protection against these types of attacks [14, 9].

In Table 2, a comparison has been made between two asymmetric algorithms:

Table 2: Application comparison of RSA and ECC

ECC(P-256)	RSA
ECC works based on different curves and different standards	The working method is using factorization
ECC Processing is time-consuming	RSA can run faster than ECC thanks to its simplicity.
It has a higher level of security and is expanding	It has a lower level of security and is becoming obsolete
Short key and high-security level.	RSA has a larger key, lower security level

7 Security analysis Suggested Work

We propose a security algorithm based on ECC for IoT data transmission through satellites. Security at the beginning of the connection is provided by using the Diffie-Hellman key exchange method and smart registration and authentication. With the NIST P-256 standard, three main parameters are added to the data:

- Hidden values of EC parameters
- Generating point in two message sender and receiver nodes
- The hash value of the IP address of the node

At the beginning of the work, the key is transmitted using the Diffie-Hellman key. The public key is generated using the transmitted values and the private key is generated and shared by the sender and receiver of the message. According to the authentication that is done, the data is transferred to the relevant channel, and the enemy's intrusion and access to the data are prevented. Due to the limitations that exist in satellite communications and the need for the protocol used not to have a complex structure. The proposed protocol mentioned here has less overhead and thus can be considered as a lightweight security protocol with minimum key and low processing that provides minimum latency for data communication over satellite networks.

The working steps of the proposed protocol are as follows:

A. The initiation or preparation stage

B. Initial parameters agreement stage

C. key exchange step

D. elliptic curve encoding step

A) **The Start Stage (Preparation)**

In the beginning phase, we will first introduce the parameters used in this article. The link or data sender reference (SA) and data receiver reference (RA) and transmission-related parameters (CAt) and the two main variables that exist in EC the S-IOT network:

1) (a, b): The parameter is fixed.

2) point 'G': The generating point at the sender and receiver nodes.

The key exchange parameters include the public key in the sender part of the message, known as Pub_{SA}, and the public key in the receiver part of the message, known as Pub_{RA}. Also, a private key is generated through the sender and receiver, namely Pub_{SA} and Pub_{RA}, which includes $E.Key_{SA}$ and $E.Key_{RA}$ is obtained through the following calculations:

(a) To calculate the public key, we use two private keys r and s and multiply at the generating point G:

$$Pub_{SA} = s \diamond G$$

$$Pub_{RA} = r \diamond G.$$

(b) The public key transmitted between the sender and receiver of the message (SA, RA) is transmitted through the public channel between two nodes, and in case of enemy intrusion, the private key values cannot be identified.

(c) When public keys (Pub_{SA}, Pub_{RA}) are exchanged between sender and receiver. (SA, RA) calculate the shared secret key $E.Key_{SA}$ and $E.Key_{RA}$ respectively.

$$E.Key_{SA} = s \diamond Pub_{RA}$$

$$E.Key_{RA} = r \diamond Pub_{SA}.$$

(d) When transmitting a message, it is added to the public keys of the IP address in the CA_t for authentication. The central hub stores the IP hash values received from the registered nodes in a tabular format to verify the authenticity of the particular node in the future data transmission phase. Therefore, when transferring SA and RA public keys, $< Pub_{SA}||$ should be used respectively $ID >$ and $ID < Pub_{RA}||$ to be $ID >$ be sent to SA and RA.

Table 3: Description of parameters in the proposed protocol

Row	Abbreviation	Description
1	$\diamond$	Scalar multiplication operation in ECC
2	$s,\ r$	Sender and receiver secret key
3	Pub_{SA}	The public key of the sending node at the login stage
4	Pub_{RA}	The public key of the receiving node at the login stage
5	$G = (g_1, g_2)$	conversion point
6	a, b	Selected elliptic curve parameters
7	P_{SA}	The public key of the node sending the data
8	P_{RA}	The public key of the node receiving the data
9	P_m	A message to be sent via satellite communications
10	K	Unique hidden value
11	C_1, C_2	Cipher text
12	$E.Key_{SA}$	Shared encryption keys on the sender side
13	$E.Key_{RA}$	Shared encryption keys on the receiver side
14	ID	The corresponding IP address of the SA
15	$h(ID)$	The hash value of the corresponding IP address of the sender
16	$\oplus$	XOR operation

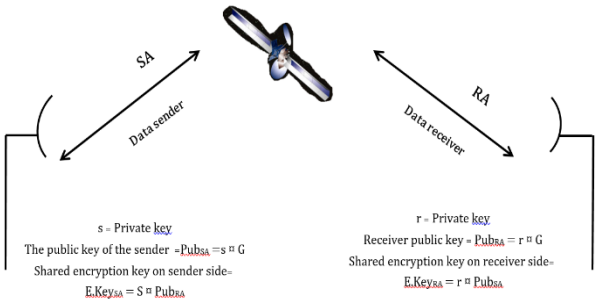


Figure 1: The start phase of the proposed protocol

B) **Initial Parameters Agreement Stage**

To the security of the Internet of Things equipped with S-IOT satellite communications, every time the message is transmitted, SA and RA must be transferred on an elliptic curve with a new generating point, which increases security, and these points and parameters must be transmitted over the public channel secretly.

The Elliptic curve algorithm has variables $EC(a, b)$ and generating point $G = (g_1, g_2)$. In the initial parameter agreement stage, power mathematical operations are used to calculate the numerical values of the variables. Two

main parameters EC and generating point G are verified separately and step by step here. As a result, according to the calculation of EC and the generating point G, the values (a, b) are transmitted secretly through the public channel between the sender and the receiver. The above values EC and G along with the IP hash values are shared in the public channel.

The above values EC and G along with the IP hash values are shared in the public channel. If these values are present in the checklist, the received data is confirmed. If the received data is not according to the checklist, it rejects it and reports it to the main hub. The steps to do the above work are as follows:

Initial stage:

SA Data sender authority	Available parameters include $a,\ b,\ G$	RA Data receiving reference
s = Private key Receiver public key: $Pub_{SA} = s \diamond G$ Shared encryption key on the sender side: $E.Key_{SA} = S \diamond Pub_{RA}$	$Pub_{SA}\|ID$ $\longrightarrow$ Public channel $Pub_{RA}\|\|ID$ $\longleftarrow$	r=Private key Receiver public key: $Pub_{RA} = r \diamond G$ Shared encryption key on the receiver side: $E.Key_{RA} = r \diamond Pub_{SA}$

Agreement stage of initial parameters:

Power in terms of values a, b		
$X = (x_k)\ a \bmod P$	$X, Y\|\|h(ID)$ $\longrightarrow$	$a = \log(X - x_k)$
$Y = (y_k)\ b \bmod p$	Public channel	$b = \log(Y - y_k)$

Key exchange step:

s = Private key		r = Private key
The public key of the sender node $PSA = s \diamond G$ $Pm = E(M)$ $C_1 = E(M) \oplus K \diamond\ PRA$ $C_2 = K \diamond G$	$P_{SA}\|\|h(ID)$ $\longrightarrow$ $P_{RA}\|\|h(ID)$ $\longleftarrow$ Public channel C_1, C_2	The public key of the receiving node $P_{RA} = r \diamond G$ Decoding (C_1, C_2) to find Pm

C) **Key Exchange Step**

Before transferring data between the transmitter and receiver in satellite communications, the public key is shared between the communication nodes. Then, the original data are transferred using secret key encryption. The two parameters $EC(a, b)$ and the generator point G agreed according to the NIST P-256 standard form a new elliptic curve, based on which two public keys $\lfloor_{SA}$ and $\lfloor_{RA}$ are calculated.

$$P_{SA}||h(ID), \quad P_{RA}||h(ID)$$

D) **Elliptic Curve Encoding Step**

In most cases, information must be transmitted through hubs in the S-IOT network, which consist of signal messages and data messages. After encoding, the message to be sent to the receiver is XORed with the values of the public key to create cipher text. The encrypted text (C_1, C_2) is formed and transmitted through the channel between the sender and the receiver. After receiving the message, the values (C_1, C_2) are decoded by the receiver.

(a) **Security Analysis of The Proposed Protocol**

The starting stage (preparation) is the calculation of the secretkey between the sender and the receiver (SA, RA), i.e. $(E.Key_{SA}, E.Key_{RA})$:

$E.Key_{SA} = s * Pub_{RA}$
$E.Key_{SA} = s * rG$
$E.Key_{SA} = (x, y)$
$E.Key_{RA} = r\ Pub_{SA}$
$E.Key_{RA} = r * s\ G$
$E.Key_{RA} = (x, y)$

As we said, the public key is equal to:
$Pub_{RA} = r \diamond G$
$Pub_{SA} = s \diamond G$.

The secret key shared between the sender and receiver of the message is not accessible on the public channel. The above secret key is for transferring information in the S-IOT network. So, sharing (Pub_{SA}, Pub_{RA}) in the network is the start or initial preparation of the network. In the agreement stage of the initial parameters, if $EC(a, b)$ and the generating point $G(g_1, g_2)$ are the parameters of the elliptic curve algorithm, the data will be transmitted with greater security. Care should be taken to transfer (SA, RA) secretly. So:

1) SA X and Y is:
   $X = (x_k)\ a \bmod P,\ Y = (y_k)\ b \bmod$.
2) a and b is:
   $a = \log(X - x_k),\ b = \log(Y - y_k)$.

Proof . $X = (x_k)a \bmod P$
$Log\ X = a.\log(x_k) \bmod P$
$Log\ X/\log(x_k) = a.\bmod P$
$a = \log(X - x_k)$
$Y = (y_k)b \bmod P$
$Log\ Y = b.\log(y_k) \bmod P$
$Log\ Y/\log(y_k) = b.\bmod P$
$b = \log(Y - y_k)$
In the encryption stage and the key exchange stage, the original message (M) is encrypted as cipher text (C_1, C_2).
$P_m = E(M),\ \{C_1 = E(M) \oplus KP_{RA}$ and $C_2 = KG\}$
Proof to find Pm:
$P_m = C_1 \oplus (C_2 r)$
$P_m = E(M) \oplus KP_{RA} \oplus (C_2 r)$
$P_m = E(M) \oplus KrG \oplus (KGr)$
$P_m = E(M)$ decoded at the other end.

After sending, the main message (M) is received by the receiver and then decoded. Therefore, there is no possibility of enemy penetration and access to the message. Considering that the elliptic curve algorithm is used here, the above message is sent with minimum key and minimum delay. □

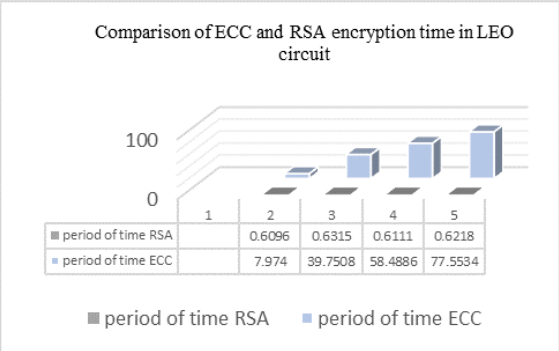
(b) **Comparison of The Proposed Protocol ECC (NIST P-256) With RSA Cipher Algorithm**

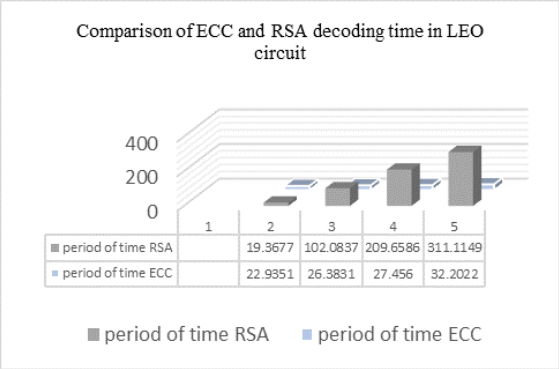
Here we implement RSA and the proposed ECC algorithm with the NIST P-256 standard for information confidentiality with 256-bit data input and random private keys in the LEO circuit. The efficiency of the proposed algorithm compared to RSA is shown in Table IV and Graphs 1, 2, and 3. According to research, RSA asymmetric encryption has a very high performance in encryption and is faster, but it has more latency and is slow in decryption. While the proposed algorithm has poor encryption performance and is slow, it decrypts faster than the RSA asymmetric algorithm. In general, the proposed algorithm is more efficient and safer than RSA, considering that in satellite communications, due to the long distance, we must apply the minimum time required for security. The algorithm, which is designed based on the elliptic curve, has a higher security level and less delay in sending than other asymmetric algorithms. And it is received.

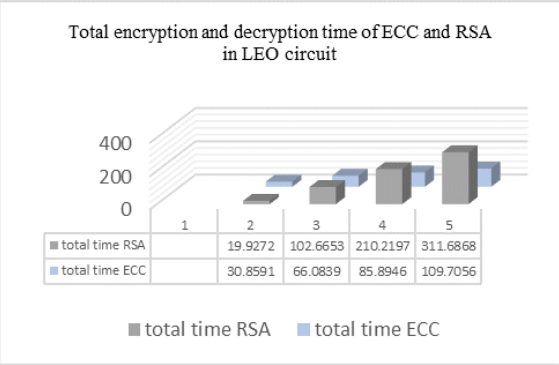
Table 4: 256 bits - Encoding, Decoding, and total time (in seconds)

Entrance: 256 bits with LEO satellite orbit delay						
Security Bit level	Encryption		Decoding		Total time	
	time ECC	time RSA	time ECC	time RSA	total time ECC	total time RSA
80	7.9240	0.5596	22.8851	19.3177	30.8091	19.8772
112	39.7008	0.5815	26.3331	102.0337	66.0339	102.6153
128	58.4386	0.5611	27.4060	209.6086	85.8446	210.1697
144	77.5034	0.5718	32.1522	311.0649	109.6556	311.6368

The graph related to the values presented in Table 4 is given below







8 Conclusion and future work

Security is essential in satellite communications. In this research, an encryption method with an asymmetric algorithm was proposed. To reduce the problems of key distribution and management and ensure a message's confidentiality and integrity, asymmetric key encryption with NIST P-256 standard with Diffie-Hellman key exchange mechanism is proposed. This article also presented a comparative analysis between RSA and ECC. This test was performed to find the time interval during encryption, and decryption of the message on the 256-bit input event with random keys based on the NIST P-256 standard. Based on this experiment, it was found that ECC is better than RSA in terms of operational efficiency and security with fewer parameters. In this project, here, we proposed a lightweight communication algorithm for the S-IOT network-based node that uses the NIST P-256 elliptic curve encryption standard, which minimizes overhead and increases security. The investigation found that this method has higher security and less delay than other methods. The proposed scheme has a series of features such as mutual authentication between satellite nodes and secret key exchange. Also, the above design is a lightweight algorithm that has minimal calculations, thus reducing the communication cost. The method studied in this article and the obtained data show that the above method performs well in an S-IOT network.

References

[1] A. Abdellahi, F. N. Mohamedade, and G. Bamba, *The effectiveness of a hybrid diffie-hellman-RSA-AES model*, Int. Conf. Comput. Commun. Inf. (ICCCI) Pub., 2022.

[2] Y. Chen, M. Zhang, X. Li, T. Che, R. Jin, J. Guo, W. Yang, B. An, and X. Nie, *Satellite-enabled internet of remote things network transmits field data from the most remote areas of the Tibetan plateau*, Sensors **22** (2022), no. 10, 3713.

[3] J. Díaz, A.V. Ferrari, and S.J.L. Fenner, *On-the-fly diffie-hellman for IoT*, Int. Conf. Chilean Comput. Sci. Soc. (SCCC), 2019.

[4] J. A. Fraire, O. Iova, and F. Valois, *Space-terrestrial integrated internet of things: Challenges and opportunities*, IEEE Commun. Mag. **60** (2022), no. 12, 64–70.

[5] A. Goulart, A. Chennamaneni, D. Torre, B. Hur, and F.Y. Al-Aboosi, *On wide-area IoT networks, lightweight security and their applications*, Practic. Rev. Electron. **11** (2022), no. 4, 1762.

[6] N. Li, *Research on Diffie-Hellman key exchange protocol*, Int. Conf. Comput. Engin. Technol., 2010.

[7] B. Li, Z. Fei, C. Zhou, and Y. Zhang, *Physical layer security in space information networks: A survey*, IEEE Internet Things J. **7** (2019), no. 1, 33–52. DOI: 10.1109/jiot.2019.2943900.

[8] V. Miller, *Use of elliptic curves in cryptography*, Adv. Cryptol. Conf. CRYPTO '85, Santa Barbara, 1985, pp. 417–426.

[9] M. Mingxuan, *Comparison between RSA and ECC*, 2nd Int. Seminar Artific. Intell. Network. Inf. Technol. (AINIT), 2021.

[10] M.G. Schraml, R.T. Schwarz, and A. Knopp, *Multiuser mimo concept for physical layer security in multibeam satellite systems*, IEEE Trans. Inf. Forensics Secur. **16** (2021), no. 4, 1670–1680.

[11] J.R. Shaikh, M. Nenova, G. Iliev, and Z. Valkova-Jarvis, *Analysis of standard elliptic curves for the implementation of elliptic curve cryptography in resource-constrained e-commerce applications*, IEEE Int. Conf. Microwaves Antennas Commun. Electronic Syst. (COMCAS), 2017.

[12] P. Tedeschi, S. Sciancalepore, and R.D. Pietro, *Satellite-based communications security: a survey of threats, solutions, and research challenges*, Comput. Networks **216** (2022), no. 4, 109246.

[13] M. Turkanović, B. Brumen, and M. Hölbl, *A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion*, Ad Hoc Networks **20** (2014), no 2, 96–112.

[14] E. Vidhya, S. Sivabalan, and R. Rathipriya, *Hybrid Key Generation for RSA and ECC*, Int. Conf. Commun. Electron. Syst. (ICCES), 2019.

[15] M. Wazid, A. Das, N. Kumar, V. Odelu, G. Reddy, K. Park, and Y. Park, *Design of lightweight authentication and key agreement protocol for vehicular ad hoc networks*, IEEE Access **5** (2017), no. 2, 14966–14980.

[16] K. Xue, C. Ma, P. Hong, and R. Ding, *A temporal credential-based mutual authentication and key agreement scheme for wireless sensor networks*, J. Network Comput. Appl. **36** (2013), no. 1, 316–323.

[17] H.L. Yeh, T.H. Chen, P.C. Liu, T.H. Kim, and H.W. Wei, *A secured authentication protocol for wireless sensor networks using elliptic curves cryptography*, Sensors **11** (2011), no. 5, 4767–4779.

[18] Y. Yan, *The overview of elliptic curve cryptography (ECC)*, J. Phys.: Conf. Ser. **2386** (2022), no 14, 012019.

[19] Y. Zhang, Y. Wang, Y. Hu, Z. Lin, Y. Zhai, L. Wang, Q. Zhao, K. Wen, and L. Kang, *Security performance analysis of LEO satellite constellation networks under DDoS Attack*, Sensors **22** (2022), no. 19, 7286.