Research Article

M. Sruthi and Rajkumar Rajasekaran*

Hybrid lightweight Signcryption scheme for IoT

https://doi.org/10.1515/comp-2020-0105
received September 30, 2019; accepted March 27, 2020

Abstract: The information transmitted in IoT is susceptible to affect the user's privacy, and hence the information ought to be transmitted securely. The conventional method to assure integrity, confidentiality, and non-repudiation is to first sign the message and then encrypt it. Signcryption is a technique where the signature and the encryption are performed in a single round. The current Signcryption system uses traditional cryptographic approaches that are overloaded for IoT, as it consists of resource-constrained devices and uses the weak session key to encrypt the data. We propose a hybrid Signcryption scheme that employs PRESENT, a lightweight block cipher algorithm to encrypt the data, and the session key is encrypted by ECC. The time taken to signcrypt the proposed Signcryption is better when compared to current Signcryption techniques, as it deploys lightweight cryptography techniques that are devoted to resource-constrained devices.

Keywords: IoT, security, lightweight cryptography, hybrid cryptography, Signcryption

1 Introduction

IoT is a widely trending pervasive paradigm that interconnects heterogeneous devices like RFID tags, smart meters, various sensors, and physical things with the internet. By connecting things to internet, they are accessible dynamically from anywhere at any time. IoT has various applications, such as healthcare, home automation, smart city, agriculture, industry, etc. The objects connected to the internet facilitate the user to monitor and actuate in the environment dynamically irrespective of their geographical location. The data transmitted by the things could be data about an individual or about an environment. The IoT network carries various private sensitive data that must be transported in a secured manner. Various security threats are present in the IoT environment. Confidentiality, integrity, and availability (CIA) triad is to be ensured for secured communication. Confidentiality assures that the message is known only to the sender/receiver and not to any third person. Non-repudiation provides the message authorship. Integrity assures that the message is not modified by any other unauthorized user. The security mechanisms used for ensuring confidentiality and integrity are by encryption and digital signature, respectively.

In traditional methods, the message is digitally signed followed by encryption, which is time-consuming. Providing the security solution at low cost is the need in the IoT environment, as the devices in IoT are resource constrained.

1.1 Research objective

The objective of this research is to ensure the non-repudiation, integrity, and confidentiality at low computation cost, to propose security solutions that lower the overhead to the resource-constrained IoT environment, and to take advantage of both symmetric and asymmetric cryptographic approaches by combining both the approaches.

1.2 Our contribution

Signcryption technique performs digital signature and encryption in parallel. Signcryption ensures confidentiality, integrity, and non-repudiation in a single step. The symmetric approach is less time consuming, but the security of the symmetric key is taken into consideration. The asymmetric approach is much stronger and a little time consuming than the symmetric approach. By combining both the symmetric and asymmetric cryptographic techniques, a hybrid Signcryption system is being

* **Corresponding author: Rajkumar Rajasekaran,** Department of Data Analytics, Vellore institute of Technology, Vellore, India, e-mail: vitrajkumar@gmail.com
M. Sruthi: School of Computer Science and Engineering, Vellore institute of Technology, Vellore, India, e-mail: sruthi.m0611@gmail.com

proposed in this article. Hybrid encryption combines the advantages of both the symmetric and asymmetric algorithms. Lightweight cryptography [1] is a new cryptographic technique specific for the resource-constrained devices with tailored S boxes and P boxes, and the number of gate equivalents (GEs) is very low when compared to the traditional cryptographic techniques. Lightweight cryptography can be employed in IoT, as the devices in IoT are resource constrained. Various lightweight ciphers and lightweight hashing algorithms [1] are discussed. PRESENT is an ultra-lightweight block cipher [2], which is 23 times smaller than the traditional AES cryptographic algorithm. Lightweight encryption algorithm PRESENT-128 has a GE (Gate Equivalent) 1886. Based on the throughput, the PRESENT and CLEFIA are the best, but the CLEFIA has many GEs when compared to the PRESENT. PRESENT algorithm consumes less power when compared to the algorithms like CLEFIA, DESL, etc. and consumes more GE (Gate equivalent) than PRESENT. So, the PRESENT cipher can be considered as an ultra-lightweight cipher in terms of power consumption, GE, and throughput. A hybrid Signcryption technique proposed uses "PRESENT," a lightweight cryptographic block cipher, to reduce the overload. ECC is better than RSA in key size, latency, memory, and signature key generation (KG) [3–7]. So, a hybrid Signcryption system proposed consists of PRESENT and ECC algorithms that ensure the confidentiality, integrity, and non-repudiation using the Signcryption technique at a low cost by deploying the lightweight symmetric block cipher and is much secured by using the hybrid technique.

The article is organized as follows: Section 2 discusses about the existing Signcryption systems, Section 2.1 gives a brief of Signcryption process, Section 2.2 discusses about the disadvantage of the existing system, Section 2.3 discusses about the proposed hybrid Signcryption scheme, Section 3 discusses about the experiments carried out to implement the proposed system, Section 4 discusses about the results and discussion, and Section 5 discusses the conclusion.

2 Literature survey

In IoT, the data sent from devices travel through many hops of network and reach the receiver. Many devices, networks, and services are unified to provide a smart environment in IoT. This makes the devices in IoT vulnerable to attacks, and attackers may modify data generated by the devices with a malevolent intention [8]. For a secured IoT environment, confidentiality, data integrity, authentication, and authorization of users are the basic essential things. Security is provided by implementing cryptographic services to safeguard the IoT environment from the attacks [9]. The security services required for the IoT are mostly based on traditional cryptographic techniques. Energy saving is an important factor in IoT as it is a constrained environment [9]. Some of the security measures considering the constrain in IoT include online/offline security protocols, low power security protocols, and hybridization techniques [9]. In online/offline security practices, a part of the security computation is done offline, and thus only a part of the execution is done online, and by doing so, the power consumption is reduced. The usage of lightweight cryptography protocols for security requirements will reduce the overload in constraint environment. The lightweight cryptography is lightweight in terms of both software with small S box and P box, and hardware with less amount of gate equivalence required. Hybridization can be a solution for minimizing energy by combining any two techniques or protocols [9]. Low power security protocols can be an effective security solution for a resource-constrained environment like IoT. Considering the low power security solutions can reduce the overhead in IoT, this can be achieved by substituting the traditional cryptographic protocols with the lightweight cryptographic protocols. Similar to traditional cryptographic algorithms, lightweight cryptographic algorithms also have equivalent protocols. It has a variety of block ciphers, stream ciphers, and lightweight hash functions. Lightweight cryptography provides effective security with very few heavy operations [9].

The traditional method to secure the message is to first sign and then encrypt. This is a time-consuming process as the message digest must be signed first and then the message is encrypted to cipher text. The Signcryption scheme, as first proposed in ref. [10], enables the user to do the signature and encryption concurrently. Signcryption ensures the authenticity and confidentiality in a single process by consuming less time and is applicable to IoT comprising resource-constrained devices. The ElGamal-based short Signcryption scheme has been proposed in ref. [10] and then compared with the RSA algorithm based on signature then encryption. In ref. [11] is proposed an online and offline Signcryption scheme based on identity-based encryption (IDE). The main advantage of the system in ref. [11] is that the system is based on IDE but is able to send messages to a public key infrastructure (PKI) based system. And the proposed method [11] can connect the WSN to IoT. Most of the operations in ref. [11] are made offline, which make the

system to be powered on continuously and not suitable for the low powered devices. In ref. [12], have proposed a short ECC based Signcryption scheme and proved the performance value by comparing a traditional RSA, ECC scheme. Also, the mathematical proofs for the correctness of the short ECC Signcryption, confidentiality, unforgeability, and non-repudiation of the proposed short ECC-based Signcryption scheme are given. Thus, a short ECC-based system reduces the computational time by 89% when compared to the RSA-based Signcryption system. The PKI-based or IDE-based online/offline Signcryption scheme has a drawback of certificate management. So, in ref. [13] they proposed a certificate-less online or offline Signcryption scheme. In ref. [14] is a cryptanalysis of COOSC (Certificateless Online/Offline Signcryption) scheme that is proposed in ref. [13]. Explains about the COSSC method with mathematical proofs and considers linearity, non-degeneracy, computability parameters and mathematically proof that [13] is not secured as it is proposed and concludes COSSC compromise the private key. Although the COOSC model presented in ref. [13] is much more efficient in terms of computational time, it compromises the security of the secret key as said in ref. [14] cryptanalysis. Various privacy-preserving techniques, such as homomorphic encryption, attribute-based signature, ring signature, and group signature techniques, which have been proposed in previous studies are compared [15]. Some of the existing techniques are implemented in the following three platforms: chips, mobile, and PC [15]; and finally concludes that short signature techniques are suitable for IoT devices. Korean certification-based signature standard using the ECC signature system has been proposed in ref. [16]. The proposed ECC-based Signcryption scheme [16] is stronger to insider/outsider attack, ensures confidentiality, integrity, non-repudiation, and it is certificate less. In ref. [17] is cryptanalysis of an ECC-based Signcryption system in ref. [18]. In ref. [18] defines the three stages in signcryption with mathematical notation and concluded with mathematical proof that in the ref. [18] has many security flaws because of weak session key. The devices in IoT are vulnerable to the device capture attack. Unauthorized users may have access to the secret key used in Signcryption system. So, an obfuscate aggregate Signcryption scheme has been proposed in ref. [19] such that the obfuscator is designed with less computational cost. The data are converted to be unintelligible and then aggregated. By this technique, the data are safe even if an unauthorized user gains access to the secret key. The heterogeneous offline/online Signcryption scheme [20] that allows a user in identity-based encryption can message a user in a public key based environment. The IDE-based system [11] proves that it is efficient than the heterogeneous system in terms of computation storage requirements and time taken to signcrypt and unsigncrypt [20].

2.1 Basic Signcryption process

The Signcryption algorithm has three general functions as follows: KG, Signcryption, and unsigncrytpion. The general process of Signcryption is shown in Figure 1.

2.1.1 KG algorithm

In any PKI-based Signcryption system, the first step is to generate the public/private key pair.

$$\mathrm{SK}, \mathrm{PK} \leftarrow \mathrm{KG(ID)}.$$

Initially, each user registers with the third-party key generator who is responsible for the generation of a pair key–private key and the public key of users. Here, "SK" is the secret key and "PK" is the public key returned to the sender by the KG algorithm.

2.1.2 Signcryption S (S_{SK}, R_{PK}, M)

The sensor data to be transmitted are signcrypted. To signcrypt: the message M, sender secret key S_{SK}, and the receiver's public key R_{PK} are given as an input.

$$\mathrm{S}(S_{\mathrm{SK}}, R_{\mathrm{PK}}, \mathrm{M}) \rightarrow \mathrm{SC} \ (\textit{signcrypt the message } \mathrm{M})$$

The signature and cipher text (SC) in a single step is being generated by computing the encryption and signature generation in parallel.

2.1.3 Unsigncrytpion US (R_{SK}, S_{PK}, SC)

$$\mathrm{SC}(S_{\mathrm{SK}}, R_{\mathrm{PK}}, \mathrm{M}) \rightarrow \mathrm{US}(R_{\mathrm{SK}}, S_{\mathrm{PK}}, \mathrm{SC})$$

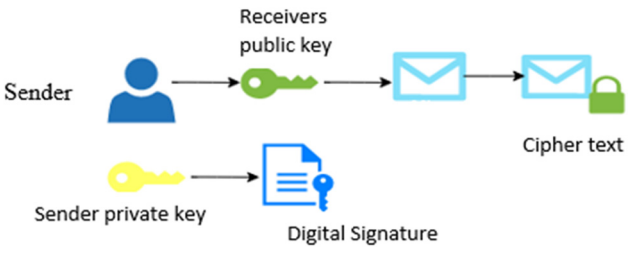

Figure 1: General Signcryption process.

The received signcrypted message SC (S_{SK}, R_{PK}, M) is unsigncrypted US ($R_{SK,}$ S_{PK}, SC) by applying the receiver R_{SK} and the sender public key S_{PK}, on the signcrypted message SC and the message is verified by applying the S_{PK} on the message, and the decryption also takes place in parallel.

The message can be verified by the signature verification process to know it is to the corresponding receiver, as there are chances for the third party to remove the signature and append their signature.

2.2 Research findings

In the heterogeneous online/offline Signcryption scheme [9], the largest part of the Signcryption process is done in offline mode that exhausts the power resource of the resource-constrained device. The cryptanalysis of the COSSC method in ref. [13] is done in ref. [14] and has identified that the COSSC model compromises the private key. The short Signcryption scheme based on the ECC is being proposed in ref. [12] that is suitable for resource-constrained devices.

The cryptanalysis of the ECC-based Signcryption scheme in ref. [18] proved that the session key is weak [17]. Thus, the session key is important to secure the message. The proposed Signcryption method uses a hybrid technique to secure the session key and perform Signcryption effectively.

2.2.1 Security notations

Game EUF-CMA:

Initialization: Given a security parameter K, the challenger C runs a setup algorithm and generates the private key PR_K and the corresponding public key P_K. The parameter K and the public key P_K are sent to the adversary A.

Attack: The adversary issues a query to the challenger C to unsigncrypt the C, sender public key PK_S, and the receiver public key PK_R. The challenger C performs unsigncryption and then sends the output to adversary A.

When the adversary A issues query to the challenger C to signcrypt M message with sender public key PK_S and the receiver public key PK_R, the challenger C performs Signcryption and then sends the output to the adversary A.

Forgery: The adversary A forges the output σ* and the message M* from the sender S to the receiver R such that it passes the verification test.

ADV $A^{EUF\text{-}CMA}$

= PROBABILITY [unsigncryption (σ*, PR_S, PK_R) = M*].

2.3 Proposed hybrid Signcryption system

PRESENT is an SPN lightweight block cipher with 31 rounds. The input of a 64-bit message is taken and converted into a 64-bit cipher text by applying the key of 80/128 bit. There are two variants of PRESENT based on key size as follows: 80-bit PRESENT 80 and 128-bit PRESENT 128. The S box of PRESENT is of 4 bytes. Thus, each round has added round key, S box function, and P box function. In the existing Signcryption scheme, the problem is weak session key, or the secret key used to encrypt the message is weak.

To overcome these issues, we propose a hybrid system (Figure 2) that involves asymmetric encryption to secure the session key. There are three phases in the proposed system, which are as follows: setup phase, Signcryption phase, and unsigncrytpion phase.

2.3.1 Setup phase

The key distribution center (KDC) is a central third party that provides key to the registered user. The sender S and the receiver R registered in KDC have their own pair of public and private keys. Let us assume that before communication the sender S and the receiver R prove their identity and get the public key of each other from the KDC.

Let G be the additive group of order q and p be the generator of the G. Then, given a K security parameter, let E be the elliptic curve over finite field Z.

P = point on the curve; N = greatest the prime number; select x = [0, N–1]

Public key PK = x * p,

Private Key PRK = x,

where PK_S, PRK_S is the public key/private key of the sender and PK_R, PRK_R is the public key/private key of the receiver.

2.3.2 Signcryption phase

2.3.2.1 Signature generation phase

To generate a digital signature, the MAC code of the message m is computed. After the computation of the MAC code, it is digitally signed with the sender's private key.

Thus, the sender in the future cannot deny the message authorship, and the non-repudiation is ensured.

2.3.2.2 Encryption phase

The message M of 64 bits is encrypted using the ultra-lightweight symmetric block cipher PRESENT by the sender secret session key of 80 bits. Thus, a cipher text of 64 bits is created.

$$M \rightarrow CT(M, S_K).$$

The session secret key is to be transmitted securely to the receiver. Thus, to keep it secure, a hybrid approach is proposed that employs the ECC algorithm to encrypt the session key that is used to encrypt the message. So, the secret session key of 80 bits is encrypted by the receiver's public key using the ECC algorithm.

$$S_K \rightarrow CK(R_{PK}, S_K).$$

The cipher text and the cipher key are being sent to the receiver along with the signature.

$$CT + CK \rightarrow R.$$

Cipher text and the cipher key are encapsulated and sent to the sender. Signature generation and encryption processes are done in parallel to minimize the time taken in serial processing.

Choose two numbers k ∈ Z_q and t ∈ Z_q,

Compute b = t mod q,

Compute k2 = k PRKR,

M = (m1, m2, m3....mi),

where mi is the message block of size 64 bits and SK is the symmetric key of the 80-bit PRESENT algorithm.

Ci = (SK,mi,{E(SK) → SK,PKr})

Compute c = m ⊕ H2 (PRKS, K2)

Compute h = H3 (m, PRKS, K2, b, c)

Compute w = (h + k) – 1 t – 1 mod q.

Thus, the signcrypted message σ = (c, PRKS, b, v, w) and the m message can be recovered by receiver PRKR → C (SK) → m.

2.3.3 Unsigncrytpion phase

In the receiving end, the cipher key, the cipher text, and a signature are being verified. The cipher key is decrypted using the receiver private key, and then the key is applied on the cipher text to decrypt the message. Using the sender public key, the message's digital signature is verified.

Decrypting the message and signature verification are done in parallel to unsigncrypt the message.

The proposed system is more efficient in terms of time taken and at the same time provides confidentiality, integrity, and non-repudiation in a single step.

Given a signcrypted σ = (c, PRK_S, b, v, w), where v and w are used to verify the signature of the message m. If it is valid, then the signature is valid for message m or else the message is ignored.

Compute the message by m = c ⊕ H2 (PRK_S, K2)

Compute m = h = H3 (m, PRK_S, K2, b, c).

Proposed lightweight hybrid Signcryption scheme

Step 1: D(d1, d2, d3) → M

where d1, d2, and d3 are the various data from the device D, and M is the message to be encrypted.

Step 2: M → CT(M, S_K)

The message M is encrypted to a cipher text CT by applying the session secret key of PRESENT algorithm S_K on the message M.

Step 3: SK → CK(R_{PK}, S_K)

The secret key S_K is encrypted into a cipher key CK using the receiver R public key PK on the secret key S_K.

Step 4: S → CT + CK → R

The sender will encapsulate the cipher text CT and cipher key CK to the receiver R.

Step 5: S_{PRK} → SC(M) → R

The sender S private key PRK is used to sign the message and send it to the receiver R.

Steps 4 and 5 are computed in parallel. Sender ensures the non-repudiation by signing the message M using the private key.

Step 6: R → CT + CK

The receiver R will receive the cipher text CT and the cipher key CK.

Step 7: R → S_K(CK, R_{PRK})

The receiver R computes the secret key S_K by applying the private key $_{PK}$ on the cipher key CK.

Step 8: R → US(S_{PK}, M)

The receiver verifies the authenticity of the message using sender S public key $_{PK}$ on the message M.

Step 9: R → M (CT, S_K)

Steps 8 and 9 are computed in parallel. The receiver R decrypts the message M by applying the secret session key S_K on the cipher text CT.

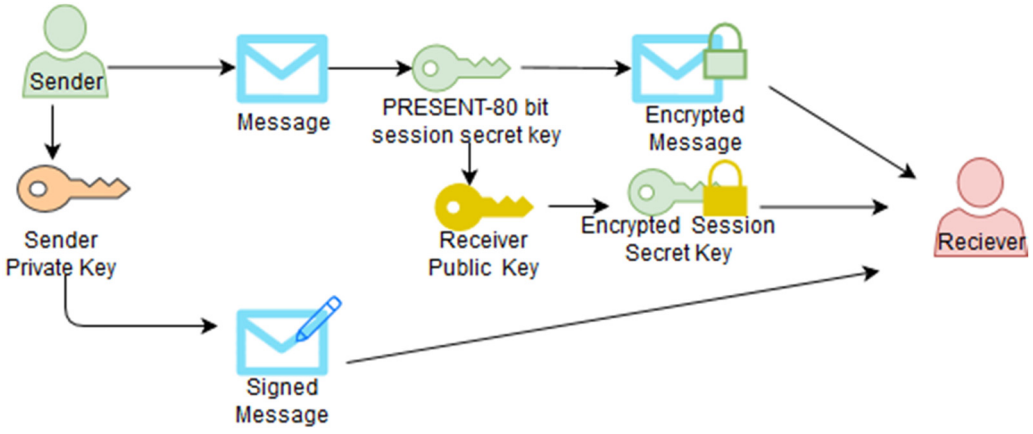

Figure 2: Hybrid lightweight Signcryption scheme.

2.3.3.1 Security analysis

The attacks in Signcryption scheme are mainly based on the signature forgery attack. The attacker signs the message m with a valid signature σ, where the signature σ is not signed by the actual sender.

The adversary A has a probability of breaking the EUF-CMA $\varepsilon \geq 10(q_s + 1)(q_s + qh2)/2^K$. Let the expected time t is less than $t' \leq 120{,}686\, q_s\, t/\varepsilon$, where Hi = (1,2,3) hash queries and q_s are the Signcryption queries.

Proof:

Initialization: Let the challenger C runs a setup algorithm to choose a $\mathcal{X} \in \mathbb{Z}_p$ and compute the corresponding public key $Ppub = \mathcal{X}^P$ and then randomly computes $K_P = U \in \mathbb{Z}_p$, the receiver private key and corresponding public key $K_{SR} = U^P$, and then sends (P, P_{PUB}, q, H1, H2), receiver public key K_P to the adversary A.

Attack:

Let the challenger C simulates hash functions H1, H2, and H3 with three lists L1, L2, and L3.

Hash H1 query: The Adversary A makes a hash query H1; the challenger checks for the existence in list L1 form of (PRK_S, h_{1i}) and returns h1 or otherwise will generate a random h1 $\in \mathbb{Z}_q$ and then inserts (PRK_S, h_{1i}) in list L1 and returns h1 to adversary A.

Hash H2 query: The adversary A makes a hash query H2; the challenger checks for the existence in list L2 in the form of (PRK_S, k2, h_{2i}) and returns h2 or otherwise will generate a random h2 $\in \mathbb{Z}_q$ and then inserts (PRK_S, k2, h_{2i}) in list L2 and returns h2 to the adversary A.

Hash H3 query: The adversary A makes a hash query H3; the challenger checks for the existence in list L3 in the form of (m, PRK_S, k2, d, c, h_{3i}) and returns h3 or otherwise will generate a random h3 $\in \mathbb{Z}_q$ and then inserts (m, PRK_S, k2, d, c, h_{3i}) in list L3 and returns h3 to the adversary A.

Signcryption query:

When adversary A issues a request (m, PK_A, PK_C), the challenger C computes the following:

Chooses a secret key PR_A from the list L_K corresponding to the matching entry of PK_A;

Computes c = m ⊕ H2 (PRK_S, k2);

Computes h = m ⊕ H3 (m, PRK_S, k2, d, c).

The proposed hybrid Signcryption system is strong against SUF-CMA, EUF-CMA attacks, in specific sEUF-CMA, but the UUF-CMA attacks are possible to proposed hybrid lightweight Signcryption system.

Table 1: Performance comparison of existing schemes with proposed hybrid Signcryption scheme

Schemes	Computational cost		Security	
	Signcrypt	Unsigncrypt	Unforgeability	Insiders' confidentiality
Ref. [20]	2M	2M + 2P	Yes	Yes
Ref. [11]	2M	4M	Yes	Yes
Proposed hybrid scheme	1M	1M + 1P	Yes	Yes

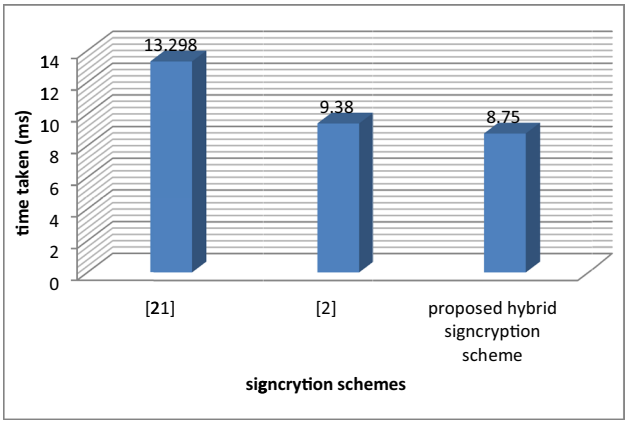

Figure 3: Time taken to signcrypt.

3 Experiment

Time taken is an important consideration in the constrained environment like IoT as the aim is to provide maximum security with a minimum amount of time when compared to the existing system. Ultimately, the proposed system uses a hybrid system using a lightweight algorithm that reduces the overhead when compared to the prevailing Signcryption scheme. The hybrid lightweight cryptosystem-based Signcryption system is implemented using the Raspberry Pi3 B model. Raspberry Pi is a minicomputer with low specifications as in a computer. It has a GPU VideoCore-IV. It has Broadcom BCM2837 64-bit quad core processor with the speed of 1.2 GHz. It has 1 GB of low power DDRAM-LPDDRAM2. The Raspbian OS is flashed through microSD card. It has various connectivity options, such as Wi-Fi and Bluetooth low energy with 40 GPIO pins. The codes are written in the Python language.

4 Results and discussion

Security in IoT is an important concern, and the privacy of the user need to be ensured. Security can be ensured by various cryptographic techniques. Lightweight cryptographic techniques can be used, which are dedicatedly designed for resource-constrained devices. Thus, the proposed system lowers the overhead in the existing system by using a lightweight technique in hybrid methodology. The system is implemented in Raspberry Pi, and the results are shown as a graphical figure. The sensor data reading is being written in SD card and then processed using the hybrid technique. The PRESENT algorithm is a block cipher that takes 64-bit input data. And the 80-bit key is encrypted by the ECC algorithm. Obtained results shows the proposed system takes 8.75 ms to signcrypt the message which is less time when compared to the IDE-based Signcryption system in ref. [11].

In Table 1, we compare the computational cost of the proposed scheme with that of ref. [11,20]. In G1, let M indicate the point multiplication operation. In G2, let E indicate the exponentiation operation. Thus, the computational cost for the proposed system is better when compared to the computational cost of ref. [11,20]. In unforgeability, the system is strong against EUF-CMA attacks. Until the key remains secure, i.e., not known to others, the message will not be read by others, except for the sender and the receiver (INT-CCA). The system is secure against insider security.

5 Conclusion

Signcryption process ensures the confidentiality, integrity, and non-repudiation in a single step by parallelizing the signature generation and encryption. Signcryption scheme is less time consuming than the traditional signature and encrypt and applicable to the IoT environment comprises of resource-constrained device, where time taken, is an important factor. Existing Signcryption scheme uses a traditional cryptographic algorithm, which is overloaded for a resource-constrained environment. The session key used to encrypt is weak in the existing system. To overcome the drawbacks, the proposed hybrid system uses the lightweight block cipher, which is dedicated to resource-constrained environment and uses a hybrid approach to keep the session key secured. The proposed system takes 8.75 ms to signcrypt, whereas the system in ref. [11,20] IDE-based Signcryption takes 9.38 and 13.298 ms as shown in Figure 3. Thus, the proposed Signcryption system is efficient than the existing system in terms of time taken.

Conflict of interest: Authors state no conflict of interest.

References

[1] W. J. Buchanan, S. Li, and R. Asif, "Lightweight cryptography methods," *J. Cyber Sec.Technol*, vol. 1, no. 3–4, pp. 187–201, 2017.

[2] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. Robshaw, et al., "PRESENT: An ultra-lightweight block cipher," *International Workshop on Cryptographic Hardware and Embedded Systems*, Berlin, Heidelberg: Springer, 2007, September, pp. 450–466.

[3] A. V. Mota, S. Azam, B. Shanmugam, K. C. Yeo, and K. Kannoorpatti, "Comparative analysis of different techniques of encryption for secured data transmission," *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering* (*ICPCSI*), Chennai, India: IEEE, 2017, September, pp. 231–237.

[4] M. Bafandehkar, S. M. Yasin, R. Mahmod, and Z. M. Hanapi, "Comparison of ECC and RSA algorithm in resource constrained devices," *IT Convergence and Security* (*ICITCS*), *2013 International Conference on*, Macao: IEEE, 2013, December, pp. 1–3.

[5] C. Sharma, "Performance analysis of ECC and RSA for securing CoAP-based remote health monitoring system," *Ambient Communications and Computer Systems*, Singapore: Springer, 2018, pp. 615–628.

[6] D. Mahto and D. K. Yadav, "RSA and ECC: A comparative analysis," *Int. J. Appl. Eng. Res.*, vol. 12, no. 19. pp. 9053–9061, 2017.

[7] G. V. S. Raju and R. Akbani, "Elliptic curve cryptosystem and its applications," *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, vol. 2, Washington, DC, USA: IEEE, 2003, October, pp. 1540–1543.

[8] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Future Gene. Comput. Syst.*, vol. 82, pp. 395–411, 2018.

[9] H. Hellaoui, M. Koudil, and A. Bouabdallah, "Energy-efficient mechanisms in security of the internet of things: A survey," *Comput. Netw.*, vol. 127, pp. 173–189, 2017.

[10] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) $\ll$ cost (signature) + cost (encryption)," *Adv. Crypto – Crypto'97*, pp. 165–179, 1997.

[11] P. Y. Ting, J. L. Tsai, and T. S. Wu, "Signcryption method suitable for low-power IoT devices in a wireless sensor network," *IEEE Syst. J.*, vol. 12, pp. 2385–2394, 2018.

[12] X. Zhou, Z. Jin, Y. Fu, H. Zhou, and L. Qin, "Short signcryption scheme for the Internet of Things," *Informatica*, vol. 35, no. 4, pp. 521–530, 2011.

[13] M. Luo, M. Tu, and J. Xu, "A security communication model based on certificateless online/offline signcryption for Internet of Things," *Sec. Commun. Netw.*, vol. 7, no. 10. pp. 1560–1569, 2014.

[14] W. Shi, N. Kumar, P. Gong, N. Chilamkurti, and H. Chang, "On the security of a certificateless online/offline signcryption for Internet of Things," *Peer Peer Netw. Appl.*, vol. 8, no. 5. pp. 881–885, 2015.

[15] L. Malina, J. Hajny, R. Fujdiak, and J. Hosek, "On perspective of security and privacy-preserving solutions in the internet of things," *Comput. Netw.*, vol. 102, pp. 83–95, 2016.

[16] K. T. Nguyen, N. Oualha, and M. Laurent, "Lightweight certificateless and provably-secure signcryptosystem for the internet of things," In: *Trustcom/BigDataSE/ISPA, 2015 IEEE*, vol. 1, Helsinki, Finland: IEEE, 2015, August, pp. 467–474.

[17] M. Toorani and A. A. Beheshti, "Cryptanalysis of an elliptic curve-based signcryption scheme. arXiv preprint arXiv:1004.3521," 2010.

[18] Y. Han, X. Yang, and Y. Hu, "Signcryption based on elliptic curve and its multi-party schemes," In: *Proceedings of the 3rd International Conference on Information Security*, Shanghai, China: ACM, 2004, November, pp. 216–217.

[19] Y. Shi, J. Han, X. Wang, J. Gao, and H. Fan, "An obfuscatable aggregatable signcryption scheme for unattended devices in IoT systems," *IEEE Inter. Things J.*, vol. 4, pp. 1067–1081, 2017.

[20] F. Li and P. Xiong, "Practical secure communication for integrating wireless sensor networks into the internet of things," *IEEE Sensors J.*, vol. 13, no. 10. pp. 3677–3684, 2013.