Computers, Materials & Continua
DOI:10.32604/cmc.2022.019800
Article

Tech Science Press

Hyper Elliptic Curve Based Certificateless Signcryption Scheme for Secure IIoT Communications

Usman Ali[1,2], Mohd Yamani Idna Idris[1,3,*], Jaroslav Frnda[4], Mohamad Nizam Bin Ayub[1], Roobaea Alroobaea[5], Fahad Almansour[6], Nura Modi Shagari[1], Insaf Ullah[7] and Ihsan Ali[1]

[1]Department of Computer System and Technology, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, 50603, Malaysia
[2]Department of Computer Science, University of Swat, Saidu Sharif, 19130, Pakistan
[3]Center for Research in Mobile Cloud Computing, University of Malaya, Kuala Lumpur, 50603, Malaysia
[4]Department of Quantitative Methods and Economic Informatics, Faculty of Operation and Economics of Transport and Communications, University of Zilina, 010 26 Zilina, Slovakia
[5]Department of Computer Science, College of Computers and Information Technology, Taif University, Taif, 21944, Saudi Arabia
[6]Department of Computer Science, College of Sciences and Arts in Rass, Qassim University, Buraydah, 51452, Saudi Arabia
[7]Department of Computer Science, Hamdard Institute of Engineering and Technology, Islamabad, 44000, Pakistan
[*]Corresponding Author: Mohd Yamani Idna Idris. Email: yamani@um.edu.my
Received: 26 April 2021; Accepted: 30 June 2021

Abstract: Industrial internet of things (IIoT) is the usage of internet of things (IoT) devices and applications for the purpose of sensing, processing and communicating real-time events in the industrial system to reduce the unnecessary operational cost and enhance manufacturing and other industrial-related processes to attain more profits. However, such IoT based smart industries need internet connectivity and interoperability which makes them susceptible to numerous cyber-attacks due to the scarcity of computational resources of IoT devices and communication over insecure wireless channels. Therefore, this necessitates the design of an efficient security mechanism for IIoT environment. In this paper, we propose a hyperelliptic curve cryptography (HECC) based IIoT Certificateless Signcryption (IIoT-CS) scheme, with the aim of improving security while lowering computational and communication overhead in IIoT environment. HECC with 80-bit smaller key and parameters sizes offers similar security as elliptic curve cryptography (ECC) with 160-bit long key and parameters sizes. We assessed the IIoT-CS scheme security by applying formal and informal security evaluation techniques. We used Real or Random (RoR) model and the widely used automated validation of internet security protocols and applications (AVISPA) simulation tool for formal security analysis and proved that the IIoT-CS scheme provides resistance to various attacks. Our proposed IIoT-CS scheme is relatively less expensive compared to the current state-of-the-art in terms of computational cost and communication overhead. Furthermore, the IIoT-CS scheme is 31.25% and

51.31% more efficient in computational cost and communication overhead, respectively, compared to the most recent protocol.

Keywords: IoT security; authentication protocols; hyperelliptic curve cryptography; certificateless public key cryptography

1 Introduction

The Internet of Things (IoT) is a rapidly evolving infrastructure which allows traditional systems to connect with one another by incorporating new devices such as sensors, actuators, and other smart devices. The integration of IoT and wireless sensor networks (WSN) has boosted the usage of IoT in our everyday lives, such as health tracking, smart houses, smart cities, and smart transportation [1]. The widespread use of IoT can also be seen in an industrial environment known as Industrial IoT (IIoT) or Industry 4.0, including industrial automation, aviation, smart retail, smart farming, and power systems [2–4]. The IIoT refers to the use of well-connected IoT devices for collecting and communicating real-time events in industrial systems to reduce human effort and operational costs and to enhance manufacturing and industrial processes. However, these interconnected smart devices and networks have been used to enable a variety of cyber-attacks due to the inadequate computational resources and communication over insecure wireless channels. Therefore, this necessitates the design of an efficient and secure mechanisms for IIoT environment. The limited battery life of smart devices is one of the main obstacles in the design of security solutions for IIoT applications. As a result, a current research focus is on developing a secure and efficient solutions for resource-constrained IoT devices. The security requirements for IIoT data, such as confidentiality, integrity, authenticity, and non-repudiation must always be ensured due to the resource-constrained IoT devices and communications over an insecure network. A signature-then-encryption mechanism is one solution to ensure such security requirements, however, this approach is not appropriate for low computing IoT devices as it produces the message's signature and encryption in two separate steps. To enhance the performance, Zheng [5] introduced Signcryption techniques, which incorporates signature and encryption in a single logical step. However, Zheng approach is based on public key cryptography (PKC). In PKC based schemes, the public key of a participating entity contains a random number belonging to some group that does not offer authenticity to the participating entity as the group elements provide no identity to the participating entities [6]. To address the flaws in PKC based schemes, the notion of public key infrastructure (PKI) was introduced in which a certificate authority (CA) is used that binds the public key with certificates [7]. However, this mechanism suffers from certificate storage, distribution, and manufacturing difficulties [8]. To overcome these shortcomings, the idea of identity-based cryptography (IBC) was suggested in [9]. IBC enables the participating entities to produce public keys directly from their identities, such as e-mail and phone numbers, without the need for CA, and the private key for each participating entity is generated by the trusted server which acts as the key generation center (KGC). The principle Signcryption was implemented to merge the features of signature and encryption into a single step [10]. However, IBC based schemes suffer from the key escrow problem in which the KGC has the complete knowledge of the private keys of all participants. To address this problem, the idea of Certificateless Public Key Infrastructure (CPKI) was suggested in [11]. In CPKI, a participant's private key is made up of two parts: one part is the private key provided by the KGC, and the second part is a secret value generated by the participant itself. The concept of Certificateless Signcryption (CS) was introduced, in which the principle of Signcryption was implemented to merge the features of signature and encryption into a single step [12].

Normally, the above-mentioned Signcryption schemes' security and efficiency depend on some computationally difficult problems, for instance, RSA, bilinear pairing (BP), and elliptic curve cryptography (ECC). The RSA [13,14] scheme is not appropriate for resource constraint devices because it contains large factorization and uses a 1024-bit large key size [15]. Furthermore, BP is 14.31 times worse than RSA [16]. ECC has been introduced to address the shortcomings of RSA and BP [17]. In comparison to BP and RSA, ECC uses less parameter size, public key, and private key sizes. Furthermore, the efficiency and security of the ECC is based on 160-bit key size [18]. However, ECC based schemes are still inefficient for resource constraint IoT devices. To enhance the efficiency of ECC based schemes, the idea of hyper elliptic curve cryptography (HECC) was introduced [19]. The HECC offers the similar level of security as ECC by utilizing 80-bit small key sizes [20–22]. Thus, HECC is considered a better choice for resource constraint IoT devices. In this paper, we proposed HECC based IIoT certificateless Signcryption (IIoT-CS) scheme for secure communication in IIoT environment.

1.1 Motivation and Contributions

Recently, Garg et al. [23] Proposed authentication scheme for IIoT environment. We found that their scheme is based on a hierarchical approach in which two participating IoT nodes cannot perform mutual authentication directly without an active server. In their scheme, the intended IoT nodes need to perform an authentication process with the server before they start communication, which increases the communication overhead for each IoT node. Furthermore, the efficiency and security of their scheme is based on ECC which suffer from high computational overhead due to the large parameters and key sizes compared to HECC. Their scheme's verification is not proved using formal security verification tools such as RoR. To address these shortcomings, we propose HECC based IIoT-CS scheme for secure IIoT communications. As we mentioned in the introduction, the HECC offers a similar level of security as ECC, RSA, and bilinear pairing by using a smaller key size, which reduces the computational and communication overhead. We verified the security of IIoT-CS scheme using the RoR model and AVISPA simulation tool. We also performed the informal security analysis of the proposed scheme. Furthermore, the results proved the efficiency of IIoT-CS scheme.

1.2 Outline of the Paper

The remainder of the paper is presented as follows. Section 2 contains related work; Section 3 shows the system model and threat models; Section 4 presents the proposed scheme; Section 5 demonstrates the proof of correctness; Section 6 presents the security evaluation; Section 7 presents the comparative analysis; Section 8 discusses the conclusion and finally Section 9 shows future work.

2 Related Work

Information security is important to protect critical information in modern communication systems where the communication is held through an insecure public networks. The research community is also echoing the significance of such a topic [24–26]. Hassija et al. [27] addressing the evolving security issues in IoT environments, emphasizing the significance of maintaining secure communication among IoT nodes. To safeguard sensitive data/information, it must be concealed from unauthorized access (confidentiality), identify who sent the message (authentication), be protected from alteration (integrity), and be available to a legitimate user [28]. Therefore, encryption techniques are used to ensure confidentiality, whereas digital signatures are applied to guarantee integrity and authenticity. In the conventional encryption mechanism known as

signature-then-encryption in which the sender has to first sign and then encrypt the data. However, this approach has some drawbacks, such as requiring more machine cycles and energy, which reduces the performance. To address these shortcomings, the concept of Signcryption was introduced in [5]. However, this approach is based on PKC in which the public key of a participating entity contains a random number belonging to some group that does not offer authenticity to the participating entity as the group elements provide no identity to the participating entities [6]. To address Signcryption flaws in [5], IBS scheme were suggested in [10]. However, it turned out that IBS scheme suffer from the key escrow problem. To address this issue, CS scheme was introduced in [12]. Following this scheme, another CS scheme based on random oracle model (ROM) was proposed [29]. Wahid et al. [30] proposed EC-based CS efficient scheme. Zhou et al. [31] proposed a new SM based CS scheme. They used the modified decisional bilinear Diffie Hellman problem and square computational Diffie Hellman problem to prove their scheme's security requirements. Rastegari et al. [32] proposed SM based CS scheme. Yu et al. [33] proposed a new CS scheme and demonstrated their scheme's security by using ROM. Lin et al. [34] presented the cryptanalysis of the of scheme in [33] and found that since the requirements of confidentiality and unforgeability are not fulfilled, their scheme may be completely thwarted. Zhou [35] suggested a new BP based CS scheme using SM for security proof.

3 System and Threat Model

This section shows the details of the system model and threat model considered for the proposed IIoT-CS scheme.

3.1 System Model

Primarily, an IIoT environment consists of multiple IoT domains made up of IoT devices called nodes such as sensors, actuators, and other devices as shown in Fig. 1. These IoT devices routinely collect information and transmit it to other devices in the network. The focus of this article is to design an authentication schemes to secure the communication among IoT nodes. The IoT nodes have minimal computing resources, while the KGC is a trusted server which has ample resources. We further assume that certain cryptographic elements are preloaded into the memory of all participating nodes and the nodes have to transmit their public keys and identities to KGC and other nodes to which they want to communicate.

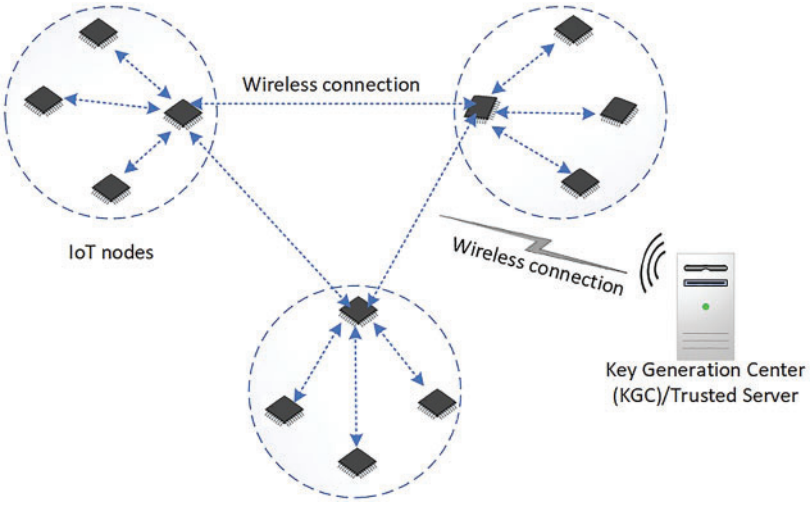

Figure 1: System model of the proposed scheme

3.2 Threat Model

In the proposed scheme, we considered a powerful threat model called Dolev-Yao (DY) threat model [36], which allows an adversary to execute passive and active attacks. According to DY threat model, the adversary has access to the communication network and can listen in to all communications between participating entities. Furthermore, the adversary has complete knowledge of all public parameters of participants in the system, however the adversary has no access to the participant's private data. Furthermore, the adversary can impersonate any device in the system by replaying messages previously eavesdropped from the communication channel.

4 Proposed Scheme

The proposed IIoT-CS scheme is based on HEC certificateless Signcryption and involves two phases, namely: pre-deployment phase and authentication phase, as shown in Fig. 2. The notations used in the proposed IIoT-CS scheme are shown in Tab. 1.

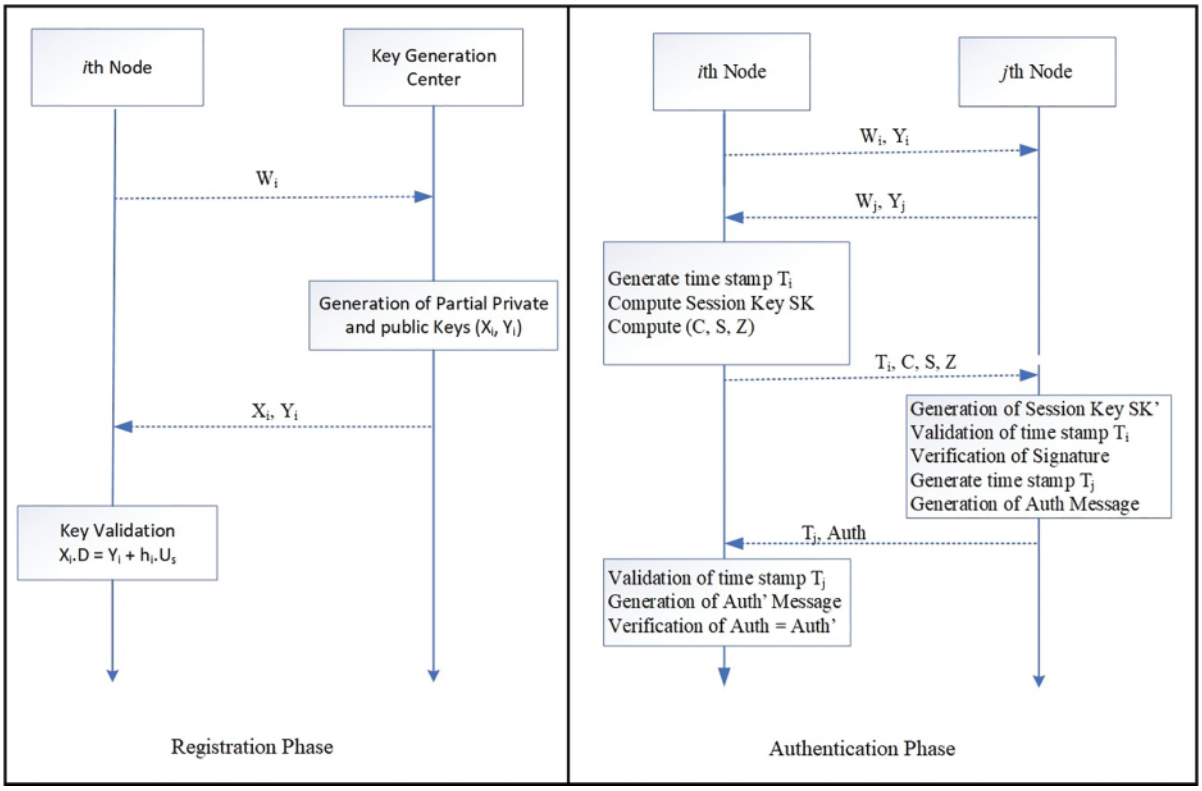

Figure 2: Flow of interaction in the proposed IIoT-CS scheme

4.1 Pre-Deployment Phase

The predeployment phase is performed by the system administrator before the effective deployment of the system. In this phase, the IoT nodes are equipped with the basic cryptographic parameters necessary to establish secret session keys. This process is divided into two parts, namely, the system initialization stage and the registration stage.

Table 1: Notations used in the proposed IIoT-CS scheme

Notation	Description
E	HEC
D	Divisor of HEC
F_q	Finite prime field of size $q = 2^{80}$
H	Hash function
(x, y)	HEC parameters
(U_s, V_s)	KGC public and private key pair
ith-node	Sender IoT node
jth-node	Receiver IoT node
ID_i, ID_j	Identity of sender node and receiver node
(V_i, U_i)	First part of private and public key of *ith-node*
(X_i, Y_i)	Second part of private and public key of *ith-node*
(V_j, U_j)	First part of private and public key of *jth-node*
(X_j, Y_j)	Second part of private and public key of *jth-node*
T_i, T_j	Time stamp produced by the *ith-node* and *jth-node*
n_i	Nonce
m, C	Plaintext and cipher text
SK	Session Key established between *ith-node* and *jth-node*
$E_{SK}()$, $D_{SK}()$	Encryption and decryption algorithms
S	Digital signature

4.1.1 System Initialization Phase

This process is carried out by the KGC, during which the following cryptographic information are initialized and made public.

i) The hyperelliptic curve E/F_q over a prime finite field F_q.
ii) The algebraic closure f^* of F_q.
iii) The Divisor group D of the curve E.
iv) Hashing function H: $\{0, 1\} \rightarrow Z_q^*$, where Z_q^*, $= \{1, 2, \ldots, q - 1\}$

In addition, the KGC generates its master private key $V_s \in_R Z_q^*$ and master public key $U_s = V_s.D$. Finally, it makes the public parameters $params = \{F_q, f^*, q, x, y, D, U_s, H\}$, publicly available to all participants.

4.1.2 Registration Phase

During the registration stage, the system's IoT nodes communicate with the KGC across a secure network in order to obtain dedicated cryptographic components. During the registration stage, the IoT nodes participating in the system communicate with the KGC through a secure communication channel to register their self and receive dedicated cryptographic information from the KGC. The flow of interaction of IoT nodes with the KGC is described below and shown in Fig. 2.

Step 1: The intended IoT node (say *ith-node*), that requires to be registered with the KGC, generates its identity ID_i and private key as $V_i \in_R Z_q^*$. Next, the node computes the first part of

its public key as $U_i = V_i.D$. The node then, computes a string $W_i = (ID_i \,||\, U_i)$, and transmits it to the KGC using a secure channel.

Step 2: Upon receiving $\{W_i\}$, the KGC performs the following operations to compute the corresponding second part of the private and public keys on behalf of *ith-node*.

i) The KGC selects a random value $\mathrm{r}_i \in_R Z_q^*$, compute $\mathrm{Y}_i = \mathrm{r}_i.\mathrm{D}$ and sets it as the second part of the public key of the *ith-node*.
ii) The KGC computes $\mathrm{h}_i = H(\mathrm{W}_i \,||\, \mathrm{Y}_i)$ and $\mathrm{X}_i = ((\mathrm{r}_i + \mathrm{h}_i.\mathrm{V}_s) \mod \mathrm{q})$ and sets X_i as the second part of the private key of the *ith-node*. The KGC delivers X_i and Y_i to the *ith-node* using a secure channel.

Step 3: Upon receiving the second part of its private and public keys from KGC, the *ith-node* can verify the authenticity of these keys by using the equation $\mathrm{X}_i.D = \mathrm{Y}_i + \mathrm{h}_i.\mathrm{U}_s$. If this equation is validated, then the keys could be deemed valid and correctly generated by the KGC. Thus, the *ith-node* can set its full private key as $(\mathrm{V}_i,\ \mathrm{X}_i)$ and full public key as $(\mathrm{U}_i,\ \mathrm{Y}_i)$.

4.2 Authentication Phase

The authentication process is initiated by an IoT node (say *ith-node*) with the intention of communicating with the other IoT nodes (say *jth-node*) as depicted in Fig. 2. As described in the predeployment phase, each IoT node is preloaded with certain cryptographic information. Furthermore, to begin the authentication process, the *ith-node* generate a message $M1 = \langle \mathrm{W}_i,\ \mathrm{Y}_i \rangle$ and transmit it to the *jth-node*. On receiving $M1$ the *jth-node* replies with a new message $M2 = \langle \mathrm{W}_j,\ \mathrm{Y}_j \rangle$. On receiving $M2$ from the *jth-node*, the *ith-node* generates a fresh session key, ciphertext, and signature by using the certificateless Signcryption operation as described below.

i) Generate a timestamp T_i, select a fresh nonce $\mathrm{n}_i \in \{1, 2, 3, \ldots, q-1\}$ and a random secret value $b \in \{1, 2, 3, \ldots, q-1\}$ and compute $Z = b.D$.
ii) Compute $\alpha = Y_j + U_S.H(W_j \,||\, Y_j)$
iii) Compute a secret session key $SK = b(U_j + \alpha)$
iv) Compute cipher text $\mathrm{C} = \mathrm{E}_{SK}(\mathrm{ID}_i, m, \mathrm{n}_i)$, where m is plaintext.
v) Computes the digital signature $S = (X_i + H(ID_i \,||\, m \,||\, n_i)(V_i + b))\ mod\ q$
vi) The *ith-node* sends $M3 = \langle T_i, C, S, Z \rangle$ to the *jth-node* using insecure channel.

On receiving M3, the *jth-node* check the validity of T_i and if it is found to be valid, then proceed with the authentication procedure, otherwise terminate the session. The *jth-node* validates the digital signature and decrypt the ciphertext by using certificateless Un-Signcryption operation as described below.

i) Computes the secret session key $SK' = Z(V_j + X_j)$
ii) Perform decryption operation $\mathrm{D}_{\mathrm{SK}'}(\mathrm{C}) = (\mathrm{ID_i}, \mathrm{m}, \mathrm{n_i})$
iii) compute $\beta = Y_i + U_S.H(W_i \,||\, Y_i)$
iv) if $\mathrm{S.D} = \beta + \mathrm{H}(\mathrm{ID_i} \,||\, m \,||\, \mathrm{n_i}).(\mathrm{Z} + \mathrm{U_i})$ is hold, then *ith-node* is authenticated successfully.

The *jth-node* Compute $K_{ij} = V_j.U_i$ and compute the message $Auth = H(W_i \oplus W_j \oplus n_i \oplus K_{ij})$.

Finally, the *jth-node* generate time stamp T_j and send the message $M4 = \langle T_j,\ Auth \rangle$ to the *ith-node*. The *ith-node* after receiving M4 from the *jth-node*, first validate T_j and if it is found to be valid, then proceed with the authentication procedure, otherwise terminate the session.

The *ith-node* compute $K_{ji} = V_i.U_j$ and $Auth' = H(\mathrm{W}_i \oplus \mathrm{W}_j \oplus \mathrm{n}_i \oplus \mathrm{K}_{ji})$.

If $Auth = Auth'$, then the *jth-node* is authenticated successfully.

5 Proof of Correctness

This section presents the proof of the correctness of the secret session key and signature verification.

5.1 Proof of Secret Session Key SK′ = SK

$SK' = Z(V_j + X_j)$, *where* $Z = b.D$ *and* $X_j = r_j + V_s.H(W_j \,||\, Y_j)$

$\Rightarrow b.D(V_j + r_j + V_s.H(W_j \,||\, Y_j))$

$\Rightarrow b.\mathrm{V}_j.D + b.\mathrm{r}_j.D + b.\mathrm{V}_s.D.H(\mathrm{W}_j \,||\, \mathrm{Y}_j)$

$\Rightarrow b(\mathrm{V}_j.D + \mathrm{r}_j.D + \mathrm{V}_s.D.H(\mathrm{W}_j \,||\, \mathrm{Y}_j))$, where $U_j = V_j.D, \mathrm{Y}_j = \mathrm{r}_j.D$, and $U_s = V_s.D$

$\Rightarrow b(\mathrm{U}_j + \mathrm{Y}_j + \mathrm{U}_s.H(\mathrm{W}_j \,||\, \mathrm{Y}_j))$, where $\alpha = Y_j + U_s.H(W_j \,||\, Y_j)$

$\Rightarrow b(U_j + \alpha) = SK$ hence proof of correctness is verified.

5.2 Proof of Signature Verification

$\beta + (Z + U_i)H(ID_i \,||\, m \,||\, n_i) = S.D$

$\beta + (\mathrm{Z} + U_i).H(ID_i \,||\, m \,||\, n_i)$, where $\beta = Y_i + U_s.H(W_i \,||\, Y_i)$

$\Rightarrow Y_i + U_s.H(W_i \,||\, Y_i) + (\mathrm{Z} + U_i).H(ID_i \,||\, m \,||\, n_i)$

$\Rightarrow Y_i + U_s.H(W_i \,||\, Y_i) + Z.H(ID_i \,||\, m \,||\, n_i) + U_i.H(ID_i \,||\, m \,||\, n_i)$, where $\mathrm{Y}_i = \mathrm{r}_i.D, \mathrm{U}_s = \mathrm{V}_s.D, Z = b.D$ and $\mathrm{U}_i = \mathrm{V}_i.D$

$\Rightarrow \mathrm{r}_i.D + \mathrm{V}_s.D.H(W_i \,||\, Y_i) + b.D.H(ID_i \,||\, m \,||\, n_i) + \mathrm{V}_i.D.H(ID_i \,||\, m \,||\, n_i)$

$\Rightarrow (\mathrm{r}_i + \mathrm{V}_s.H(W_i \,||\, Y_i) + (b + \mathrm{V}_i).H(ID_i \,||\, m \,||\, n_i))\mathrm{D}$, where $\mathrm{X}_i = \mathrm{r}_i + \mathrm{V}_s.H(W_i \,||\, Y_i)$

$\Rightarrow (\mathrm{X}_i + (b + \mathrm{V}_i).H(ID_i \,||\, m \,||\, n_i))\mathrm{D}$, where $S = \mathrm{X}_i + (b + \mathrm{V}_i).H(ID_i \,||\, m \,||\, n_i)$

$\Rightarrow S.\mathrm{D}$, hence correctness of digital signature is verified.

6 Security Evaluation

We conducted both formal and informal security assessments to illustrate the potential of the IIoT-CS scheme against various attacks. The two computational problems that are useful in performing the formal security analysis are described below.

Definition 1: Collision-Resistant One-Way Hash Function $(\mathrm{H}(.): \{0,1\}^* \rightarrow \{0,1\}^n)$

It is a "deterministic mathematical function that accepts a variable-length input string and produces a n-bit fixed-length output string".

Definition 2: (Hyper Elliptic Curve Discrete Logarithm Problem (HECDLP))

According to HECDLP, it is infeasible for an attacker to extract a value j from the relation $L = j.D$, whereas $\mathrm{j} \in Z_q^*$ is the random number from $Z_q^* = \{1, 2, \ldots, \mathrm{q} - 1\}$.

6.1 Formal Security Analysis Using RoR Model

We used ROR model [37] in which an adversary simulates real attacks to target the communication between IoT nodes. In the proposed IIoT-CS scheme, an adversary is represented by *Ad* and the participating nodes are represented by *ith-node* and *jth-node*. Further, we assume the instances of *ith-node* and *jth-node* are represented by $\Phi = \{\Phi i \text{ and } \Phi j\}$. *Ad* initiates the following queries to interact with Φ.

i) *Execute query: Ad* eavesdrops on the communication channel and intercepts all communication between Φ.
ii) *Send query: Ad* transmits a message to Φ and obtains a reply from it consequently.
iii) *Reveal query: Ad* attempts to recover the session key between Φi and Φj.
iv) *Test query: Ad* requests Φ for session key and it responds with a random bit c.

Moreover, H(.) is modeled as a random oracle which is available to all participants and adversary *Ad*. In the proposed IIoT-CS scheme, we demonstrated the existence of session key security (semantic security) by using Theorem 1 as stated below.

Theorem 1: Assume *Ad* runs in a polynomial time *pt* and attempts to break the session key security between Φi and node Φj then *Ad*'s advantage in breaching the session key security can be written as follows:

$$Adv_{Ad}^{\text{IIoT-CS}}(pt) \leq \frac{q_h^2}{|Hash|} + 2.Adv_{Ad}^{HECDLP}(pt) \tag{1}$$

where the variables $|Hash|, q_h^2$, and $Adv_{Ad}^{HECDLP}(pt)$ represent the range space of H(.), the number of hash queries, and the non-negligible winning advantage of breaking HECDLP respectively.

Proof of Theorem 1: To prove Theorem 1, we used three Games *Gi (i = 1, 2, 3)*. Within each game *Gi*, *Ad* attempts to guess the bit *c* by applying the test query. If $wins_{Ad}^{G_i}$, is an event where *Ad* accurately guesses *c*, so *Ad*'s advantage is as follows:

$$Adv_{Ad,G_i}^{\text{IIoT-CS}}(pt) = \Pr[wins_{Ad}^{G_i}] \tag{2}$$

Game G1: This game is similar like the real scheme that runs in RoR model. We obtain the following result in this game.

$$Adv_{Ad}^{IIoT-CS}(pt) = |2Adv_{Ad,G_1}^{IIoT-CS} - 1| \tag{3}$$

Game G2: In *G2*, Ad intercepts all messages exchanged between Φi and Φj, these messages are $m1 = \{Wi, Yi\}$, $m2 = \{Wj, Yj\}$, $m3 = \{C, R, S, Z\}$ and $m4 = \{Auth\}$. Next, *Ad* employs the Execute query to retrieve the session key, then employs the Reveal and Test queries to examine if the obtained session key is original or randomly generated. In the proposed IIoT-CS scheme, the session key can be produced as $SK = b(U_j + \alpha) = SK' = Z(V_j + X_j)$. To obtain this key correctly, *Ad* needs the secret values b, V_j and X_j. It implies that just eavesdropping of m1, m2, m3, and m4 would not improve *Ad*'s winning probability. Hence, G1 and G2 are indistinguishable as shown in the following equation.

$$Adv_{Ad,G_2}^{IIoT-CS} = Adv_{Ad,G_1}^{IIoT-CS} \tag{4}$$

Game 3: This game makes use of the Send and Hash queries. In *G2*, we know that eavesdropping on *m1, m2, m3*, and *m4* between Φi and Φj, would not result in hash collision as these

messages are safeguarded by HECDLP and hash function. HECDLP protects the variables b, V_s, V_i, and V_j used within Z, U_s, U_i and U_j respectively, while the hash function protects the variable S and the encryption algorithm protects the variables C, and Auth. Moreover, *G2* and *G3* are indistinguishable except G3 solves HECDLP and performs the Hash and Send queries. The advantage of solving HECDLP by A is $Adv_A^{HECDLP}(pt)$, and, as per the birthday paradox, using such a hash oracle query has a probability $\frac{q_h^2}{2|hash|}$. Overall, the following result is obtained.

$$\left|Adv_{Ad,G_2}^{IIoT-CS} - Adv_{Ad,G_3}^{IIoT-CS}\right| \leq \frac{q_h^2}{2|hash|} + Adv_{Ad}^{HECDLP}(pt) \tag{5}$$

Now Ad executes all queries and guessing the bit c, the following result is obtained

$$Adv_{Ad,G_3}^{IIoT-CS} = \frac{1}{2} \tag{6}$$

From Eqs. (3) and (4), we obtain the following result.

$$\frac{1}{2}.Adv_{Ad}^{IIoT-CS}(pt) = \left|Adv_{Ad,G_1}^{IIoT-CS} - \frac{1}{2}\right| = \left|Adv_{Ad,G_2}^{IIoT-CSS} - \frac{1}{2}\right| \tag{7}$$

From Eqs. (6) and (7), we obtain the following result.

$$\frac{1}{2}.Adv_{Ad}^{IIoT-CS}(pt) = \left|Adv_{Ad,G_2}^{IIoT-CS} - Adv_{Ad,G_3}^{IIoT-CS}\right| \tag{8}$$

Similarly, from Eqs. (5) and (8), we obtain the following result.

$$\frac{1}{2}.Adv_{Ad}^{IIoT-CS}(pt) \leq \frac{q_h^2}{2|hash|} + Adv_{Ad}^{HECDLP}(pt) \tag{9}$$

Now multiplying Eq. (9) by "2" we obtain the following result.

$Adv_{Ad}^{\mathrm{IIoT-CS}}(pt) \leq \frac{q_h^2}{|hash|} + 2.Adv_{Ad}^{HECDLP}(pt)$, hence, Theorem 1 is proved.

6.2 Formal Security Verification Using AVISPA

We used AVISPA tool [38] to verify the proposed IIoT-CS scheme security towards known attacks. AVISPA gives the results by using the keywords SAFE, or UNSAFE, which denotes whether the protocol is secure or not secure against various attacks. We applied two backends of AVISPA simulation tool, namely: OFMC and CL-ATSe to verify the security of our scheme. The result show that the IIoT-CS scheme is secure against various attacks under the DY threat model as shown in Fig. 3.

6.3 Informal Security Analysis

The following assumptions were taken into account for the informal security analysis. The secret values (b, V_s, V_i *and* V_j) are only known to the corresponding participating entity (KGC and IoT nodes) and the adversary has no knowledge about it. The encryption algorithm (E_{SK}) is secure enough that an attacker cannot not decrypt C and $\{Auth\}$.

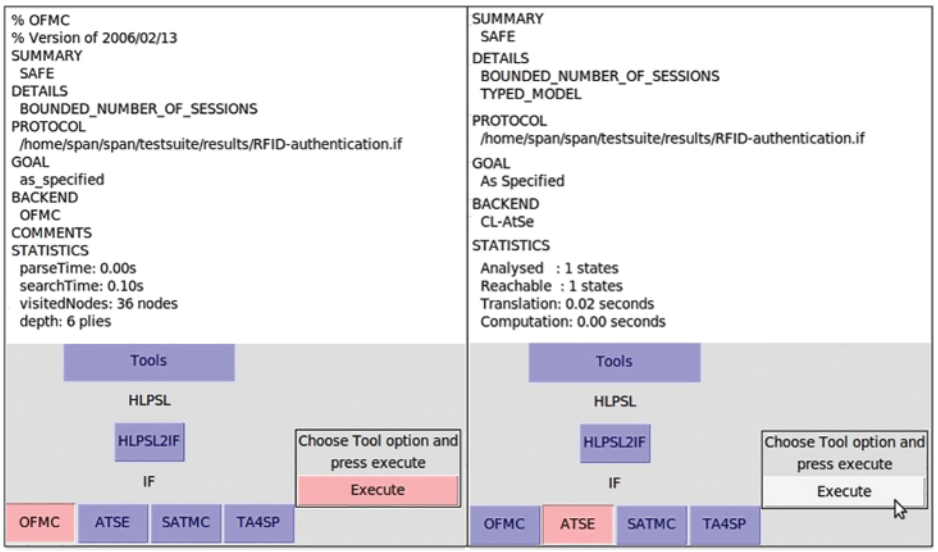

Figure 3: AVISPA simulation results for the proposed IIoT-CS scheme

6.3.1 Confidentiality

Confidentiality refers to the assurance that private information will be kept secret during transmission. In the start, the *ith-node* and *jth-node* share their public keys and identities in the form of plain text with each other because they are not required to be kept secret. The *ith-node*, then transmit the message $\{T_i, C, S, Z\}$ to the *jth-node*. The time stamp T_i which discloses no information. The adversary cannot interpret the ciphertext C as it requires the secret session key SK which depends on the private random number b. According to HECDLP, an adversary is unable to compute b given Z and D. Similarly, Ad is unable to extract any knowledge from S because it depends on the private values (V_i and b) of *ith-node*. The messages $\{T_j, Auth\}$ sent by the *jth-node* to the *ith-node* also reveals no information. T_j is the time stamp and $Auth$ is a hash message in which an adversary cannot extract any information. As a result, the existing protocol successfully provides confidentiality features.

6.3.2 Authentication

To ensure secure communication between IoT nodes, they must authenticate each other at the start within each session and vice versa.

ith-node authentication: The *jth-node* calculates the session key SK after obtaining the message $\{C, S, Z\}$ from *ith-node*. The *jth-node* verify the signature $S = \mathrm{X}_i + (b + \mathrm{V}_i).H(ID_i \,||\, m \,||\, n_i)$of the *ith-node* by using the equation $S.D = \beta + (\mathrm{Z} + U_i)H(ID_i \,||\, m \,||\, n_i)$. If this equation hold then the *ith-node* is successfully authenticated by the *jth-node*. Suppose an adversary imitates to be a legitimate node, in that scenario, it would need to generate a valid S. However, S is based on the private values of *ith-node* which are only known to the *ith-node* so any adversary would not be able to produce the right value of S.

jth-node authentication: After receiving $\{Auth\}$ from the *jth-node*, the *ith-node* computes$\{Auth'\}$. The *ith-node* check if $Auth = Auth'$, then *jth-node* is successfully authenticated by the *ith-node*. If an adversary pretends itself as a legitimate node, it must send the right message $\{Auth\}$. However,

{*Auth*} is hashed message which is based on private key of *jth-node*, making it difficult for an adversary to transmit the right message {*Auth*}.

6.3.3 Non-Repudiation

The value of S transferred to the *jth-node* by the *ith-node* is based on the private key of *ith-node*. Similarly, the message {Auth} sent by the *jth-node* to the *ith-node* is based on the private key of the *jth-node*. If the *jth-node* verified *ith-node* signature i.e., if $S.D = \beta + (\mathrm{Z} + U_i)H(ID_i \,||\, m \,||\, n_i)$ is hold, the *ith-node* will not deny that it sent the message to the *jth-node*, and if $Auth = Auth'$, the *jth-node* will not deny that it delivered the message to the *ith-node*.

6.3.4 Integrity

The proposed scheme can verify that whether a cipher text C was changed or not during the communication, by using the equation $S.D = \beta + (\mathrm{Z} + U_i)H(ID_i \,||\, m \,||\, n_i)$. If an adversary modifies C, then this equation will not hold, otherwise this equation will hold. Similarly, if an adversary modifies the message {*Auth*}, it can be quickly detected because it would not be the same as {*Auth*′}. In both cases, the authentication would not succeed, and the session would be terminated. Thus, integrity is ensured in the proposed scheme.

6.3.5 Unforgeability

In the proposed IIoT-CS scheme, if *Ad* tries to produce a legitimate signature, then *Ad* must compute the equation $S = \mathrm{X}_i + (b + \mathrm{V}_i).H(ID_i \,||\, m \,||\, n_i)$. For this, *Ad* would need the private key pair (V_i, X_i) of the *ith-node*. To compute the private keys, *Ad* must solve HECDP which is infeasible. Hence, the proposed IIoT-CS schemes provides security against unforgeability.

6.3.6 Forward Secrecy

In the proposed IIoT-CS scheme, the secret session key is renewed after every session completion process. The secret session key depends on the private values b, V_{j} and X_j of participating nodes, and it is infeasible for an adversary to find these private values due to HECDLP. Thus, the adversary *Ad* is not able to read and use the previous messages later. Hence, the proposed scheme ensures forward secrecy.

6.3.7 Security from Replay Attack

An adversary can obtain the previous messages {W_i, Y_i}, {W_j, Y_j}, {T_i, C, S, Z}, and {T_j, *Auth*} eavesdropping on the communication channel between *ith-node* and *jth-node*. The adversary replays such messages to produce an invalid effect. In the proposed IIoT-CS scheme, the value of C depends on fresh nonce ni, the value S depends on the fresh private random numbers b and V_i, the value of Z depends on b, and the value of *Auth* depends on ni and private key V_j. This means that for every session the values of C, S, Z, and *Auth* are updated. Therefore, the adversary in the next communication session is incapable to utilize the past messages. Thus, the proposed IIoT-CS scheme ensures security against replay attack.

6.3.8 Security from Eavesdropping Attacks

In the proposed IIoT-CS scheme, the messages are transmitted in plain text, hashed and cipher text format. The plain text messages contain no confidential information and provide no advantage to the adversary. Furthermore, all messages containing confidential information are always protected by using HECDLP, one-way hash function and encryption algorithm, rendering the retrieval of the confidential information computationally infeasible for an adversary. Therefore, the proposed IIoT-CS scheme prevents eavesdropping attacks.

6.3.9 Security from Denial of Service (DoS) Attack

In the proposed IIoT-CS scheme, the participating nodes first check the validity of the received timestamps. If the timestamps are not valid, then the messages are rejected. Furthermore, the information transmitted are complemented by an integrity checks in the form of signature and the encrypted message always contain the latest timestamp. Thus, the proposed scheme can identify incorrect messages and avoid DoS attacks by essentially terminating the session.

6.3.10 Security Against Impersonation Attack

In node impersonation attack, an adversary mimics the behavior of legitimate IoT nodes by eavesdropping on the communication channel. In the proposed IIoT-CS scheme, if the *Ad* mimics the behavior of a valid sender node (*ith-node*). In doing so, *Ad* produces a message $\{W_a, Y_a\}$ and sends it to a valid receiver node (*jth-node*). The *jth-node* replies the adversary with a message $\{W_j, Y_j\}$. The adversary A, when receiving $\{W_j, Y_j\}$, generate the message $\{C', S', Z'\}$ and send it to the *jth-node*. As the adversary is incapable to compute the private keys of a valid sender node, the message $\{C', S', Z'\}$ transmitted by the adversary is incorrect. The *jth-node*, upon obtaining this inaccurate message $\{C', S', Z'\}$, decrypt C' to validate the signature, but since $S'.D \neq \beta + (Z' + U_i).H(ID_i \,||\, m \,||\, n_i)$, thus the authentication fails. Furthermore, the adversary *Ad* is unable to mimics the behavior of the valid receiver (*jth-node*) because it is not feasible for *Ad* to compute the private key V_j of *jth-node*, and thus is unable to correctly produce the message $\{Auth\}$, as a result the nodes finish the session. Thus, the proposed scheme ensures security against impersonation attack.

6.3.11 Security from Man in the Middle (Mitm) Attack

In MitM attack, an adversary attempts to modify the messages from *ith-node* to the *jth-node* and vice versa. The adversary pretends itself as a valid participating entity and passes the updated messages to either node. The proposed scheme performs the mutual authentication using the messages $\{C, S, Z\}$ and $\{Auth\}$. *Ad* can only spoof a valid participant if it can produce any of these messages correctly. However, according to HECDLP the retrieval of the private key is computationally not feasible. Thus, the proposed scheme can easily withstand MitM attacks.

6.3.12 Security from Key Compromise Attack

The private key V_j and secret value b are used to obtain the secret session key SK, the adversary is incapable to get the private values due to HECDLP, as a result the adversary can't generate the secret session key and hence, the proposed IIoT-CS scheme can ensure security against key compromise attack.

7 Comparative Analysis

This section presents the comparative analysis of computational cost, communication overhead and security features.

7.1 Computational Cost

The computational overhead depends on the execution time of different cryptographic operations involved in an authentication scheme. Garg et al. [23] show that the time required to execute elliptic curve scalar multiplication (ECSM) and hash-to-point (HtP) operations is 0.986 and 14.293 ms, respectively, using MIRACL [39]. The execution time of Hyperelliptic Curve Divisor Multiplication (HECDM) is considered as 0.48 ms [40]. The time consumption of cryptographic operations is very small compared to the time consumption of ECSM and HECDM and therefore

can be ignored. In the proposed scheme, each sender node (*ith-node*) and the receiver node (*jth-node*) performs 3 HECDM operations. Therefore, the time consumed by the sender and receiver node together is $6 \times 0.48 = 2.88$ ms. The KGC performs 3 HECDM operations for at least 2 IoT nodes in the system to authenticate each other. Therefore, the time consumed by the KGC is $3 \times 0.48 = 1.44$ ms. The total time consumed by the KGC and nodes to for mutual authentication is 2.88 ms + 1.44 ms = 4.32 ms. The comparison of the computational cost of IIoT-CS scheme with the existing schemes [15,23,41] is shown in Tab. 2 and Fig. 4a. It is clear from the results that IIoT-CS scheme is less expensive in computational cost as compared to the existing schemes.

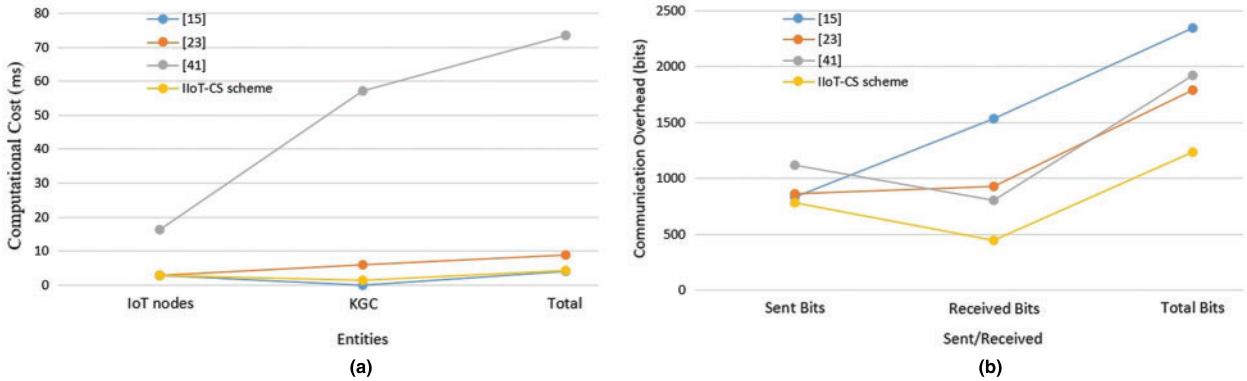

Figure 4: Comparative analysis of (a) computational cost and (b) communication overhead

Table 2: Computational cost analysis

Schemes	IoT nodes	KGC	Total
[15]	3 ECSM = 2.958 ms	1 ECSM = 0.086 ms	4 ECSM = 3.944 ms
[23]	3 ECSM = 2.958 ms	6 ECSM = 5.916 ms	9 ECSM = 8.874 ms
[41]	2 ECSM + 1HtP = 16.265 ms	4 HtP = 57.172 ms	2 ECSM + 5HtP = 73.437 ms
Ours	6 HCDM = 2.88 ms	3 HCDM = 1.44 ms	9 HCDM = 4.32 ms

7.2 Communication Overhead

Communication overhead can be determined from the number of bits sent and received by the participating IoT nodes in the authentication phase. We assumed SHA-256 as our hash function, which generates 256-bits output and 128-bit AES as our encryption algorithm which generates 128-bits ciphertext. In the proposed IIoT-CS scheme, an IoT node is required to send two messages $\{W_i, Y_i\}$ and $\{T_i, C, S, Z\}$ and receive two messages $\{W_j, Y_j\}$ and $\{T_j, Auth\}$. The communication overhead of an IoT node to send the message $\{W_i, Y_i\}$ and $\{T_i, C, S, Z\}$ is 160 + 80 + 80 + 128 + 256 + 80 = 784 bits. Whereas the communication overhead of an IoT node to receive the messages$\{W_j, Y_j\}$ and $\{T_j, Auth\}$ is 160 + 80 + 80 + 128 = 448 bits. The overall communication overhead of an IoT node is 784 + 448 = 1232 bits. The comparison of the communication overhead of IIoT-CS scheme with the existing schemes [15,23,41] is shown in Tab. 3 and Fig. 4b. It is clear from the results that IIoT-CS scheme incurs the lowest communication overhead as compared to the existing schemes.

Table 3: Communication overhead analysis

Schemes	Sent (bits)	Received (bits)	Total (bits)
[15]	832	1536	2344
[23]	864	928	1792
[41]	1120	800	1920
Ours	784	448	1232

7.3 Comparison of Security Attributes

We compare the proposed scheme's security functionality with existing state-of-the-art [15,23, 41]. The proposed scheme offers mutual authentication, non-repudiation, unforgeability, forward secrecy, resist, replay, eavesdropping, DoS, impersonation, MitM, and key compromise attacks as shown in the Tab. 4. It is obvious that the proposed IIoT-CS scheme is by far the most secure scheme amongst the existing protocols.

Table 4: Comparison of the security features

Protocols	Security features									
	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
[15]	Y	–	N	N	Y	–	Y	Y	Y	–
[23]	Y	–	N	Y	Y	Y	Y	Y	Y	–
[40]	Y	–	N	–	N	–	N	N	N	–
Proposed IIoT-CS scheme	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Notes: F1: Supports Mutual Authentication; F2: Supports Non-Repudiation; F3: Prevent Unforgeability; F4: Support Forward Secrecy; F5: Prevent Reply attack; F6: Prevent Eavesdropping attack; F7: Prevent DoS Attack; F8: Prevent Impersonation Attack; F9: Prevent MitM Attack; F10: Prevent Key Compromise Attack; Y: Yes; N: No; –: Not available.

8 Conclusion

In this study, we used HEC based CS scheme in the developing of an efficient and secure authentication mechanism for IIoT environment. The proposed scheme uses 80-bit HEC rather than 160-bit ECC for security and performance. We apply both formal and informal security analysis to evaluate the proposed scheme's security. We performed the formal security analysis by using AVISPA tool and RoR model, which affirms the security of the proposed scheme. It has been shown in the analysis that the proposed scheme offers confidentiality, mutual authentication, integrity, and non-repudiation and is also robust to a range of security attacks such as replay, eavesdropping, impersonation, MitM, DoS, and key compromise attacks etc. Our proposed scheme is relatively less expensive compared to the current state-of-the-art. Our proposed scheme is 31.25% and 51.31% more efficient in computational cost and communication overhead, respectively, compared to the most recent protocol. Thus, our proposed scheme is a viable option for IoT devices with inadequate resources.

9 Future Work

We want to incorporate and evaluate the proposed IIoT-CS scheme in a real-world IIoT environment in the future. This will make more improvements to the proposed scheme and will encourage us to evaluate its security and efficiency more accurately.

Funding Statement: This work is supported by the University of Malaya IIRG Grant (IIRG008A-19IISSN), Ministry of Education FRGS Grant (FP055-2019A). This work was also supported by Grant System of University of Zilina No. 1/2020. (Project No. 7962) and partially supported by the Slovak Grant Agency for Science (VEGA) under Grant Number 1/0157/21. The authors are grateful to the Taif University Researchers Supporting Project (Number TURSP-2020/36), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. Shen, J. Ma, X. Liu, F. Wei and M. Miao, "A secure and efficient id-based aggregate signature scheme for wireless sensor networks," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 546–554, 2016.

[2] C. B. Mwakwata, H. Malik, M. M. Alam, Y. L. Moullec, S. Parand *et al.*, "Narrowband internet of things (NB-ioT): From physical (PHY) and media access control (MAC) layers perspectives," *Sensors*, vol. 19, no. 11, pp. 2613, 2019.

[3] K. Kaur, S. Garg, G. Kaddoum, E. Bou-Harb and K. R. Choo, "A big data-enabled consolidated framework for energy efficient software defined data centers in IoT setups," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2687–2697, 2019.

[4] K. Kaur, S. Garg, G. Kaddoum, S. H. Ahmed and M. Atiquzzaman, "KEIDS: Kubernetes based energy and interference driven scheduler for industrial IoT in edge-cloud ecosystem," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4228–4237, 2019.

[5] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption)? cost (signature) + cost (encryption)," in *Advances in Cryptology-CRYPTO*, vol. 97. Springer: Berlin/Heidelberg, Germany, pp. 165–179, 1997.

[6] Z. Chen, S. Chen, H. Xu and B. Hu, "A security scheme of 5G ultradense network based on the implicit certificate," *Wireless Communications and Mobile Computing*, vol. 2018, 11 pages, 2018. https://doi.org/10.1155/2018/8562904.

[7] N. C. Kumar, A. Basit, P. Singh and V. C. Venkaiah, "Lightweight cryptography for distributed PKI based MANETS," arXiv preprint arXiv: 1804.06313, 2018.

[8] S. Ullah, L. Marcenaro and B. Rinner, "Secure smart cameras by aggregate-signcryption with decryption fairness for multi-receiver IoT applications," *Sensors*, vol. 19, no. 2, pp. 327, 2019.

[9] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the Theory and Application of Cryptographic Techniques*, Springer: Berlin, pp. 47–53, 1984.

[10] J. Malone-Lee, "Identity-based signcryption," *International Association for Cryptologic Research (IACR)*, pp. 98, 2002.

[11] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Proc. of the Int. Conf. on the Theory and Application of Cryptology and Information Security*, Berlin, Germany, pp. 452–473, 2003.

[12] M. Barbosa and P. Farshim, "Certificateless signcryption," in *Proc. of the ACM Symp. on Information, Computer and Communications Security*, Tokyo, Japan, pp. 369–372, 2008.

[13] M. Suárez-Albela, P. Fraga-Lamas and T. M. Fernández-Caramés, "A practical evaluation on RSA and ECC-based cipher suites for IoT high-security energy-efficient fog and mist computing devices," *Sensors*, vol. 18, no. 11, pp. 3868, 2018.

[14] M. Yu, J. Zhang, J. Wang, J. Gao, T. Xu *et al.,* "Internet of things security and privacy-preserving method through nodes differentiation, concrete cluster centers, multi-signature, and blockchain," *International Journal of Distributed Sensor Networks*, vol. 14, no. 12, pp. 1–15, 2018.
[15] A. Braeken, "PUF based authentication protocol for IoT," *Symmetry*, vol. 10, no. 8, pp. 352, 2018.
[16] C. Zhou, Z. Zhao, W. Zhou and Y. Mei, "Certificateless key-insulated generalized signcryption scheme without bilinear pairings," *Security and Communication Networks*, vol. 2017, 17 pages, 2017. https://doi.org/10.1155/2017/8405879.
[17] S. Kumari, M. Karuppiah, A. K. Das, X. Li, F. Wu *et al.*, "A secure authentication scheme based on elliptic curve cryptography for IoT and cloud servers," *Journal of Supercomputing*, vol. 74, no. 12, pp. 6428–6453, 2018.
[18] A. Omala, A. Mbandu, K. Mutiria, C. Jin and F. Li, "Provably secure heterogeneous access control scheme for wireless body area network," *Journal of Medical Systems*, vol. 42, no. 6, pp. 1–14, 2018.
[19] C. Tamizhselvan and V. Vijayalakshmi, "An energy efficient secure distributed naming service for IoT," *International Journal of Advanced Studies of Scientific Research*, vol. 3, no. 8, 5 pages, 2018.
[20] V. Naresh, R. Sivaranjani and N. V. E. S. Murthy, "Provable secure lightweight hyper elliptic curve-based communication system for wireless sensor networks," *International Journal of Communication Systems*, vol. 31, no. 15 pp. 3763, 2018.
[21] A. Rahman, I. Ullah, M. Naeem, R. Anwar, H. Khattak *et al.,* "A lightweight multi-message and multi-receiver heterogeneous hybrid signcryption scheme based on hyper elliptic curve," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 5, pp. 160–167, 2018.
[22] U. Ali, M. Y. I. B. Idris, M. N. B. Ayub, I. Ullah, I. Ali *et al.,* "RFID authentication scheme based on hyperelliptic curve signcryption," *IEEE Access*, vol. 9, pp. 49942–49959, 2021.
[23] S. Garg, K. Kaur, G. Kaddoum and K. K. Choo, "Toward secure and provable authentication for internet of things: Realizing industry 4.0," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4598–4606, 2019.
[24] S. Garg, K. Kaur, N. Kumar, and J. J. Rodrigues, "Hybrid deep-learningbased anomaly detection scheme for suspicious flow detection in SDN: A social multimedia perspective," *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 566–578, 2019.
[25] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya *et al.*, "A hybrid deep learning-based model for anomaly detection in cloud datacenter networks," *EEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 924–935, 2019.
[26] K. Seyhan, T. N. Nguyen, S. Akleylek, K. Cengiz and S. H. Islam, "Bi-gISIS KE: Modified key exchange protocol with reusable keys for IoT security," *Journal of Information Security and Applications*, vol. 58, pp. 102788, 2021.
[27] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal *et al.*, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019.
[28] A. Mehmood, I. Noor-Ul-Amin and A. I. Umar, "Public verifiable generalized authenticated encryption based on hyper elliptic curve," *Journal of Applied Environmental and Biological Sciences*, vol. 7, no. 12, pp. 69–73, 2017.
[29] W. Shi, N. Kumar, P. Gong and Z. Zhang, "Cryptanalysis and improvement of a certificateless signcryption scheme without bilinear pairing," *Frontiers of Computer Science*, vol. 8, no. 4, pp. 656–666, 2014.
[30] A. Wahid and M. Mambo, "Implementation of certificateless signcryption based on elliptic curve using javascript," *International Journal of Computing and Informatics*, vol. 1, no. 3, pp. 90–100, 2016.
[31] C. Zhou, G. Gao and Z. Cui, "Certificateless signcryption in the standard model," *Wireless Personal Communications*, vol. 92, no. 2, pp. 495–513, 2017.
[32] P. Rastegari and M. Berenjkoub, "An efficient certificateless signcryption scheme in the standard model," *ISC International Journal of Information Security (ISeCure)*, vol. 9, no. 1, pp. 3–16, 2017.
[33] H. Yu and B. Yang, "Pairing-free and secure certificateless signcryption scheme," *Computer Journal*, vol. 60, no. 8, pp. 1187–1196, 2017.

[34] X. J. Lin, L. Sun, H. Qu and D. Liu, "Cryptanalysis of a pairing-free certificateless signcryption scheme," *Computer Journal*, vol. 61, no. 4, pp. 539–544, 2018.

[35] C. Zhou, "Certificateless signcryption scheme without random oracles," *Chinese Journal of Electronics*, vol. 27, no. 5, pp. 1002–1008, 2018.

[36] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[37] M. Abdalla, P. A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," In: S. Vaudenay (eds.), *Public Key Cryptography - Lecture Notes in Computer Science*, vol. 3386, Berlin, Heidelberg: Springer, pp. 65–84, 2005.

[38] T. Genet, "SPAN, the security protocol animator for AVISPA," Accessed: April. 2021. [Online]. Available: http://people.irisa.fr/Thomas.Genet/span/.

[39] MIRACL Cryptographic SDK. Accessed: April. 2021. [Online]. Available: https://github.com/miracl/MIRACL.

[40] M. A. Khan, I. Ullah, S. Nisar, F. Noor, I. M. Qureshi *et al.,* "An efficient and provably secure certificateless key-encapsulated signcryption scheme for flying ad-hoc network," *IEEE Access*, vol. 8, pp. 36807–36828, 2020.

[41] U. Chatterjee, R. S. Chakraborty, and D. Mukhopadhyay, "A PUFbased secure communication protocol for IoT," *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 3, pp. 67, 2017.