Hindawi
Security and Communication Networks
Volume 2022, Article ID 9718580, 9 pages
https://doi.org/10.1155/2022/9718580

Research Article

An Efficient Certificate-Based Aggregate Signature Scheme for Internet of Drones

Muhammad Asghar Khan,[1] **Insaf Ullah**,[1] **Mohammed H. Alsharif**,[2] **Abdulaziz H. Alghtani**,[3] **Ayman A. Aly**,[3] **and Chien-Ming Chen**[4]

[1]*Hamdard Institute of Engineering & Technology, Islamabad 44000, Pakistan*
[2]*Department of Electrical Engineering, College of Electronics and Information Engineering, Sejong University, Seoul 05006, Republic of Korea*
[3]*Department of Mechanical Engineering, Taif University, Taif 21944, Saudi Arabia*
[4]*College of Computer Science and Technology, Shandong University of Science and Technology, Qingdao, Shandong, China*

Correspondence should be addressed to Chien-Ming Chen; chienmingchen@ieee.org

Received 11 November 2021; Accepted 22 December 2021; Published 15 January 2022

Academic Editor: Irshad Azeem

Internet of drones (IoD) is a network of small drones that leverages IoT infrastructure to deliver real-time data communication services to users. On the one hand, IoD is an excellent choice for a number of military and civilian applications owing to key characteristics like agility, low cost, and ease of deployment; on the other hand, small drones are rarely designed with security and privacy concerns in mind. Intruders can exploit this vulnerability to compromise the security and privacy of IoD networks and harm the information exchange operation. An aggregate signature scheme is the best solution for resolving security and privacy concerns since multiple drones are connected in IoD networks to gather data from a certain zone. However, most aggregate signature schemes proposed in the past for this purpose are either identity-based or relied on certificateless cryptographic methods. Using these methods, a central authority known as a trusted authority (TA) is responsible for generating and distributing secret keys of every user. However, the key escrow problem is formulated as knowing the secret key generated by the TA. These methods are hampered by key distribution issues, which restrict their applicability in a variety of situations. To address these concerns, this paper presents a certificate-based aggregate signature (CBS-AS) scheme based on hyperelliptic curve cryptography (HECC). The proposed scheme has been shown to be both efficient in terms of computation cost and unforgeable while testing its toughness through formal security analysis.

1. Introduction

Drones have recently gained a lot of attention for their wide range of applications in areas including surveillance, agriculture, healthcare, traffic management, inspections, and public safety [1, 2]. Likewise, multiple small drones can be connected to accomplish given tasks more efficiently than a single large drone [3]. Therefore, a new clan of networks known as the Internet of drones (IoD) has evolved as a result of advancing from a single drone to multiple drones connected via the Internet. This network has all of the technological resources that needs to perform the assigned task autonomously, including a communication module for transmitting and receiving data, sensors for gathering data, memory for storing sensor data, and processors for computation [4]. However, drones in IoD network typically have limited storage, energy, and computing capacities, making it difficult for them to perform computationally complex operations [5, 6].

IoD networks are typically deployed for applications that require users to retrieve real-time data from drones. There is a high chance that a malicious actor may conceivably control some drones or carry out impersonation attacks due to the multiple wireless connections among drones. Additionally, security and privacy concerns are rarely considered when small drones are designed [7]. Intruders who intend to violate the security and privacy measures of the IoD network have several options to carry out their malicious intent. They can, for example, transmit a large number of reservation requests, eavesdrop on the control messages, and/or forge information exchange [8]. A lightweight cryptographic scheme to offer data confidentiality, as well as a digital signature scheme to assure the integrity of data generated by a drone in an IoD environment, is required to solve this problem. Similarly, in an IoD network, where multiple drones are often connected to gather data from a designated zone, the notion of aggregation is essential for improving data distribution efficiency. The aggregate signature [9] is a sort of digital signature that allows several messages from different users to be compressed into a single signature. Instead of verifying all of the individual signatures, the verifier simply needs to examine the aggregate signature, resulting in a considerable decrease in the overall length of signatures. As a result, the load of network transmission can be minimized, and the efficiency of validating multiple signatures can be improved when employing the aggregate signature scheme.

Most of the existing aggregate signature schemes generate aggregate signatures using either pairing operations or ECC. These methods are inefficient since they require heavy computations and are not suitable for devices with limited resources. Moreover, a Public Key Infrastructure (PKI) encryption mechanism was utilized in an early digital signature scheme. Following that, identity-based cryptography (IBC), identity-based signatures (IBS), identity-based aggregate signatures (IBAS), and certificateless cryptography (CLC) were used to create digital signature and aggregate signature schemes. Both the IBC and CLC approaches, however, have issues with key escrow and/or key distribution [10–12]. certificate-based signatures (CBS) and certificate-based aggregate signatures (CB-AS) have been offered as solutions to overcome these issues, and research is underway to guarantee that they can fulfil a number of security requirements, including data integrity, nonrepudiation, and resistance to signature forgery [13].

To address the abovementioned issues, this article proposes a CB-AS scheme for IoD networks. The proposed scheme is efficient because it employs the concept of HECC. The HECC provides the same level of security as bilinear pairing (BP) and elliptic curve cryptography (ECC) with a small key size. The key contributions of the proposed scheme are summarized as follows:

(i) Firstly, the primary contribution of this research work is to design an aggregate signature scheme for an IoD network, in which a drone (aggregator drone) in a cluster will aggregate individual signatures of member drones and verify the validity of aggregated data.

(ii) Secondly, based on the notion of hyperelliptic curve cryptography (HECC) in a certificate-based setting, the proposed scheme is proved to be existentially unforgeable under adaptive chosen message.

(iii) Finally, the proposed scheme is compared to relevant existing schemes, and the comparison analysis reveals that our scheme is more efficient in terms of computation and communication costs.

The rest of this paper is laid out as follows. We provide related work in Section 2. Preliminaries are provided in Section 3. The system model and proposed CB-AS scheme is presented in Section 4. We evaluate provable security analysis in Section 5 before evaluating performance in terms of computation and communication costs in Section 6. Finally, in Section 7, we make a conclusion.

2. Related Work

Aggregate signatures, which are based on public key cryptography (PKC) methods, are commonly used for aggregate authentication of information exchange. In this approach, the senders sign the message using their own private keys, and then the aggregator, who is chosen by the senders, uses aggregation algorithms to compress all of the individual signatures into a fixed-length short signature. The validity of the short signature is the same as the validity of all individual signatures utilized to create the aggregate signature. Any verifier may only establish whether or not all individual signatures from the given users are legitimate by examining the aggregate signature. As a result, aggregate signature is more beneficial for IoD networks, increasing data verification and transmission efficiency.

Liu et al. [14] introduced the first CBC aggregate signature scheme, in which signers use sequential aggregation to create an AS from a prior aggregated signature. As a result, aggregation is performed by each signer. However, in practice, this approach has limited use. It is also pairing-based, which makes it inappropriate for IoD systems. Wang et al. [15] proposed a provably secure aggregate authentication scheme for a UAV cluster network. The scheme is based on an ID-based encryption method, which is prone to key escrow issues. Moreover, the proposed scheme is based on elliptic curve cryptography, which is not well suited to IoD networks. Li et al. [16] proposed an authentication framework for UAVCN based on identity-based aggregate signature method. According to security analysis, the authors claimed that their scheme is unforgeable for (attested) authentication requests and (aggregate) responses. The scheme, however, has a large computational cost. Li et al. [17] presented a certificateless pairing-free authentication system for UAV networks. The authentication mechanism of the proposed scheme is based on the notion of elliptic curve cryptography and uses an aggregator signature. Kar et al. [18] proposed an efficient and low-cost certificateless aggregate signature scheme for wireless sensor networks. Security toughness of the proposed scheme is tested under random oracle model. Both of the schemes presented in [17, 18] were, however, based on ECC cryptography, which has a marginally higher computational cost than HECC.

Verma et al. [19] proposed a pairing-free CBC-AS solution for healthcare monitoring that is devoid of key distribution and certificate management issues. The number of signers, on the other hand, determines the size of the aggregated signature. As a result of the variability in AS duration, the solution is impractical for resource-constrained IoD networks. Very recently, Verma et al. [20] presented another certificate-based efficient signature scheme with compact aggregation. The proposed CB-CAS scheme is the shortest since it uses compact aggregation. However, it may not meet the requirements of distributed ledger systems (DLSs). The reason for this is that with DLSs, several signers sign a single message. As a result, a multi-signature method is needed. Furthermore, the proposed scheme is based on the concept of ECC, which is incompatible with IoD networks. Our scheme, on the other hand, is based on HECC, a more advanced variant of ECC that offers the same level of security as ECC but with a smaller key size, lowering computation, and communication costs.

3. Preliminaries

Firstly, we will go over some basics regarding HEC, which is an advanced version of EC that only require 80 bits of parameter and key size. The advantage of the hyper elliptic curve is that it provides the same level of security robustness as the elliptic curve. Secondly, we explain the hyperelliptic curve discrete logarithms problem, which is as follows: suppose $\pi = \gamma.\mathscr{D}$; then the task of the attacker is to extract the unknown γ from π that is called hyperelliptic curve discrete logarithm. Thirdly, we present two sorts of adversaries: Type 1 and Type 2 adversaries. Type 1 is an external attacker whose objective is to forge the signature; it also lacks access to the CA's secret key. Type 2 is a malicious CA whose mission is to forge signatures. It also has access to the CA's secret key and will be unable to perform public key replacement and certificate queries. Finally, we evaluate the open channel for our proposed scheme, in which these two attackers could perform the forging procedure against it.

4. System Model and Proposed CB-AS Scheme

This section illustrates the overall concept and syntax of the proposed CB-AS scheme for IoD networks.

4.1. System Model. The proposed CB-AS system model [17] is depicted in Figure 1. Member drones (M-Drones), aggregator drones (AGT-Drones), certificate authority (CA), and base station (BS) are the four categories of entities in the proposed system. The M-Drones are in charge of monitoring a certain zone, and the AGT-Drone serves as a cluster head for a group of M-Drones that are directly attached to it. The CA is in charge of the setup and certificate generation. The BS, on the other hand, does mutual authentication before to assigning tasks to both types of drones (AGT-Drone and M-Drones). The authentication process is started by BS, which allows the aggregator drone to validate, attest, and disseminate authentication requests to its M-Drones. AGT-Drone serves as a bridge between BS and M-Drones, providing computing and communication capabilities to control its M-Drone in the cluster. The AGT-Drone in the cluster is used to communicate between the BS and the M-Drones. Each M-Drone may check its real source and the attested request before responding to authentication request of BS. AGT-Drone can validate the responses of M-Drone in the same cluster in batch. The notions used in the proposed scheme are illustrated in Table 1.

4.2. Proposed CB-AS Scheme. The phases of the proposed CB-AS scheme [19] are listed as follows:

(i) Setup: given μ^k is a security parameter, this phase enables the certifiers to publish a param $\mathscr{F} = \{\mathscr{D}, F^n, \hbar\ell, \hbar_0, \hbar_1, \hbar_2, \Theta\}$, where $\mathscr{D}$ is the divisor, F^n represents a finite field, $\hbar\ell$ is used for hyper elliptic curve, $(\hbar_0, \hbar_1, \hbar_2)$ are the three irreversible cryptographic hash functions, and $\Theta = \eta.\mathscr{D}$ means the public key of certifiers. Further, certifiers set η is his private key.

(ii) Key generation: each user $(\mho_i)$ with ID_i compute $\sigma_i = \varphi_i.\mathscr{D}$, where φ_i is private key selected by user randomly from $\hbar\ell$ group.

(iii) Certificate generation: for each user $(\mho_i)$ with ID_i, certifiers select χ_i randomly from $\hbar\ell$ group and compute $\omega_i = \chi_i.\mathscr{D}$, $\partial_i = \chi_i + \eta\hbar_0(ID_i, \omega_i, \sigma_i)$ and set $\mathfrak{S}_i = (\omega_i, \partial_i)$ as a certificate. When user wants verification of $\mathfrak{S}_i = (\omega_i, \partial_i)$, then he/she use the following equation: $\partial_i.\mathscr{D} = \omega_i + \Theta\hbar_0(ID_i, \omega_i, \sigma_i)$.

(iv) Certificate-based signature generation: for a signature generator with ID_i, compute $\alpha_i = \ell_i.\mathscr{D}$, where ℓ_i is selected randomly from $\hbar\ell$ group, $r_i = \hbar_1(\omega_i, \sigma_i, m_i, \alpha_i)$, $R_i = \hbar_2(ID_i, \omega_i, \sigma_i, m_i, \alpha_i)$, compute $\zeta_i = \partial_i + \varphi_i r_i + \ell_i R_i$, and set $\psi_i = (\alpha_i, \omega_i, \zeta_i)$ as a signature.

(v) Certificate-based signature verifications: a verifier can do the following computational steps: it computes $r_i = \hbar_1(\omega_i, \sigma_i, m_i, \alpha_i)$, $R_i = \hbar_2(ID_i, \omega_i, \sigma_i, m_i, \alpha_i)$, $\ell_i = \hbar_0(ID_i, \omega_i, \sigma)_i$ and checks if $\zeta_i.\mathscr{D} = \omega_i + \ell_i.\Theta + r_i.\sigma_i + R_i.\alpha_i$ equals and then accepts the signature.

(vi) Certificate-based signature aggregations: after reception of $\psi_i = (\alpha_i, \omega_i, \zeta_i)$, an aggregator can make $\zeta = \sum_{i=0}^{n} \zeta_i$; it means that ζ_i is the aggregated signature on m_i.

(vii) Certificate-based signature aggregations verifications: a verifier can do the following computational steps: it computes $\sum_{i=0}^{n} r_i = \hbar_1(\omega_i, \sigma_i, m_i, \alpha_i)$, $\sum_{i=0}^{n} R_i = \hbar_2(ID_i, \omega_i, \sigma_i, m_i, \alpha_i)$, $\sum_{i=0}^{n} \ell_i = \hbar_0(ID_i, \omega_i, \sigma_i)$ and checks if $\zeta.\mathscr{D} = \sum_{i=0}^{n} \omega_i + (\sum_{i=0}^{n} \ell_i).\Theta + +(\sum_{i=0}^{n} r_i).\sigma_i + (\sum_{i=0}^{n} R_i).\alpha_i$ equals and then accepts ζ.

4.3. Correctness. A verifier can do the following computational steps for verification of $\psi_i = (\alpha_i, \omega_i, \zeta_i)$:

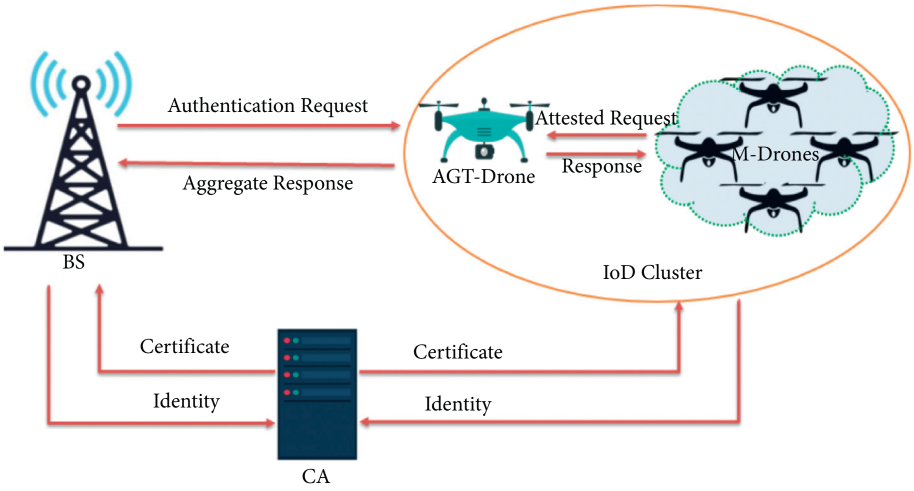

Figure 1: The proposed CB-AS scheme model for IoD.

Table 1: Notions used in the proposed scheme.

No.	Symbol	Purpose
1	μ^k	A security parameter on HEC with 80 bit size
2	$\mathcal{F}$	A published set of defined parameter in IoD network
3	$h\ell$	Symbol used to represent HEC
4	F^n	Symbol used to represent a finite field of n order on HEC
5	$\mathcal{D}$	Symbol used to represent a divisor on HEC
6	Θ	Symbol used to represent the public key of certifiers
7	η	Symbol used to represent the private key of certifiers
8	$\hbar_0, \hbar_1, \hbar_2$	Symbols used to represent the irreversible cryptographic hash functions, that is, SHA 256
9	$\eth_i$	Symbol used to represent each user with ID_i
10	σ_i	Symbol used to represent the public key of each user with ID_i
11	φ_i	Symbol used to represent the private key of each user with ID_i
12	$\mathfrak{S}_i$	Symbol used to represent the certificate of each user with ID_i
13	ζ_i	Symbol used to represent the normal certificate based signature
14	ζ	Symbol used to represent the aggregated certificate based signature
15	A_{AKR1}	Symbol used to represent Type1 attacker
16	A_{AKR2}	Symbol used to represent Type2 attacker
17	Φ	Symbol used to represent maximum number of queries
18	$\Phi_{\hbar_0}$	Symbol used to represent queries request for $\hbar_0$-oracle.
19	e	Symbol used to represent a helper, which help to solve hyper elliptic curve discrete logarithm problem for A_{AKR1} and A_{AKR2}

$$\begin{aligned}
\zeta_i.\mathcal{D} &= \omega_i + \ell_i \cdot \Theta + r_i \cdot \sigma_i + R_i \cdot \alpha_i, \\
\zeta_i.\mathcal{D} &= \left(\partial_i + \varphi_i r_i + \alpha_i R_i\right) \cdot \mathcal{D} \\
&= \left(\chi_i + \eta \hbar_0\left(ID_i, \omega_i, \sigma_i\right) + \varphi_i r_i + \alpha_i R_i\right) \cdot \mathcal{D} \\
&= \left(\chi_i \cdot \mathcal{D} + \eta \cdot \mathcal{D} \hbar_0\left(ID_i, \omega_i, \sigma_i\right)\right. \\
&\quad \left. + \varphi_i \cdot \mathcal{D} r_i + \ell_i \cdot \mathcal{D}\, R_i\right) \\
&= \left(\omega_i + \Theta \cdot \ell_i + \sigma_i \cdot r_i + \alpha_i \cdot R_i\right) \\
&= \omega_i + \ell_i \cdot \Theta + r_i \cdot \sigma_i + R_i \cdot \alpha_i.
\end{aligned} \tag{1}$$

Hence, it is proved.

Also, a verifier can do the following computational steps for verification of ζ:

$$\begin{aligned}
\zeta.\mathcal{D} &= \sum_{i=0}^{n} \omega_i + \left(\sum_{i=0}^{n} \ell_i\right) \cdot \Theta + + \left(\sum_{i=0}^{n} r_i\right) \cdot \sigma_i + \left(\sum_{i=0}^{n} R_i\right) \cdot \alpha_i, \\
\zeta.\mathcal{D} &= \left(\sum_{i=0}^{n} \partial_i + \varphi_i \sum_{i=0}^{n} r_i + \ell_i \sum_{i=0}^{n} R_i\right) \cdot \mathcal{D} \\
&= \left(\sum_{i=0}^{n} \chi_i + \eta \sum_{i=0}^{n} \ell_i + \varphi_i \sum_{i=0}^{n} r_i + \ell_i \sum_{i=0}^{n} R_i\right) \mathcal{D} \\
&= \left(\sum_{i=0}^{n} \chi_i \cdot \mathcal{D} + \eta \cdot \mathcal{D} \sum_{i=0}^{n} \ell_i + \varphi_i \cdot \mathcal{D} \sum_{i=0}^{n} r_i + \ell_i \cdot \mathcal{D} \sum_{i=0}^{n} R_i\right) \\
&= \sum_{i=0}^{n} \omega_i + \left(\sum_{i=0}^{n} \ell_i\right) \cdot \Theta + + \left(\sum_{i=0}^{n} r_i\right) \cdot \sigma_i + \left(\sum_{i=0}^{n} R_i\right) \cdot \alpha_i.
\end{aligned} \tag{2}$$

Hence, it is proved.

5. Provable Security Analysis

In this section, we intend to prove that the proposed scheme is unforgeable under the attack of both Type 1 and Type 2 adversaries. For this purpose, we perform the following four games [19].

In Game 1, we evaluate the unforgeability of our proposed CB-AS scheme against Type 1 attacker (A_{AKR1}). A_{AKR1} is the outsider attacker; its work is to forge the proposed scheme signature and solve hyperelliptic curve discrete logarithm problem (HECDLP) with the help of another entity e by using the advantage of $\text{Adve}^{\text{HECDLP}} = 1/\Phi + (1 - 1/\Phi)^{\Phi}\xi$. Note Φ represents maximum number of queries.

Proof. When e received $\pi = \gamma.\mathscr{D}$, then his task is to extract the unknown γ from π. Further, it can do the following Oracles:

(i) Setup (.)-Oracle: e set a param $\mathscr{F} = \{\mathscr{D}, F^n, \hbar\ell, \hbar_0, \hbar_1, \hbar_2, \Theta\}$, and $\Theta = \pi$. Then, e gives $\mathscr{F}$ to A_{AKR1}. Further, e choose ν is an index sustaining $1 \le \nu \le \Phi_{\hbar_0}$, where $\Phi_{\hbar_0}$ is a number of query request for $\hbar_0$-Oracle.

(ii) Key Generation (.)-Oracle: A_{AKR1} ask for this query, e combs in L_{Key} for $(ID_i, \varphi_i, \sigma_i)$, if it is exist, then it gives σ_i to A_{AKR1}. Otherwise, e compute $\sigma_i = \varphi_i.\mathscr{D}$, where φ_i is private key selected by user randomly from $\hbar\ell$ group and gives it to A_{AKR1}, further it updates the list L_{Key} with $(ID_i, \varphi_i, \sigma_i)$.

(iii) $\hbar_0$ (.)-Oracle: A_{AKR1} ask for this query, e combs in $L_{\hbar_0}$ for $(ID_i, \omega_i, \sigma_i, \ell_i)$, if it is exist, then it gives ℓ_i to A_{AKR1}. Otherwise, e select ℓ_i at random and gives it to A_{AKR1}, further it updates the list $L_{\hbar_0}$ with $(ID_i, \omega_i, \sigma_i, \ell_i)$.

(iv) $\hbar_1$ (.)-Oracle: A_{AKR1} ask for this query, e combs in $L_{\hbar_1}$ for $(\omega_i, \sigma_i, m_i, \alpha_i, r)_i$, if it is exist, then it gives ℓ_i to A_{AKR1}. Otherwise, e select r_i at random and gives it to A_{AKR1}, further it updates the list $L_{\hbar_1}$ with $(\omega_i, \sigma_i, m_i, \alpha_i, r_i)$.

(v) $\hbar_2$ (.)-Oracle: A_{AKR1} ask for this query, e combs in $L_{\hbar_1}$ for $(ID_i, \omega_i, \sigma_i, m_i, \alpha_i, R_i)$, if it is exist, then it gives ℓ_i to A_{AKR1}. Otherwise, e select R_i at random and gives it to A_{AKR1}, further it updates the list $L_{\hbar_2}$ with $(ID_i, \omega_i, \sigma_i, m_i, \alpha_i, R_i)$.

(vi) Public Key Replacement (.)-Oracle: A_{AKR1} ask for this query with $(ID_i, \varphi_i, \sigma_i^l)$, e combs in L_{Key} for $(ID_i, \varphi_i, \sigma_i^l)$, if it is exist, then it change the triple by $(ID_i, \perp, \sigma_i^l)$.

(vii) Corruption (.)-Oracle: A_{AKR1} ask for this query, e combs in L_{Key} for $(I\ D, \varphi, \sigma)$, if it is exist, then it gives σ to A_{AKR1}. Otherwise, e compute $\sigma = \varphi.\mathscr{D}$, where φ is private key selected by user randomly from $\hbar\ell$ group and gives it to A_{AKR1}, further it updates the list L_{Key} with $(I\ D, \varphi, \sigma)$.

(viii) Certificate Generation (.)-Oracle: A_{AKR1} ask for this query, if $i = \nu$, then e abort further processing, otherwise e check the list L_C for certificate, if it is exists, e sends it to A_{AKR1}. If it is not exists, then it pick ∂_i and r_i, then compute $\omega_i = \partial_i.\mathscr{D} - \ell_i.\ \pi$. At the end of this process, e give (∂_i, ω_i) to A_{AKR1} and update the list L_C accordingly.

(ix) Certificate Based Signature Generation (.)-Oracle: A_{AKR1} ask for this query, if $i = \nu$, then e set $r_\nu = \hbar_1(\omega_\nu, \sigma_\nu, m_\nu, \alpha_\nu)$, $R_\nu = \hbar_2(ID_\nu, \omega_\nu, \sigma_\nu, m_\nu, \alpha_\nu)$, and $\ell_\nu = \hbar_0(ID_\nu, \omega_\nu, \sigma_\nu)$, then compute $\omega_\nu = \partial_\nu.\mathscr{D} - \ell_\nu.\ \pi, \alpha_\nu = (R_\nu - \partial_\nu.)\mathscr{D} - r_\nu\sigma_\nu$. It also pick ζ_ν randomly and delivers $(\alpha_\nu, \omega_\nu, \zeta_\nu)$ to A_{AKR1}.

Eventually, A_{AKR1} returns a forge signature $\psi^* = (\alpha^*, \omega^*, \zeta^*)$ on m^*. Though, by using the concept of forking lemma, e returns two signatures that are $(\alpha_1^*, \omega_1^*, \zeta_1^*)$ and $(\alpha_1^*, \omega_1^*, \zeta_2^*)$. Thus, $\zeta_1^*.\mathscr{D} = \omega_1^* + \ell_1.\pi + r.\sigma_i + R.\alpha_1^*$ and $\zeta_2^*.\mathscr{D} = \omega_1^* + \ell_2.\pi + r.\sigma_i + R.\alpha_1^*$. So, $(\zeta_1^* - \zeta_2^*).\mathscr{D} = (\ell_1 - \ell_2).\gamma.\mathscr{D} = \gamma = \zeta_1^* - \zeta_2^*/\ell_1 - \ell_2$ will be the solution of HECDLP.

In the probability analysis, taking into account the above game, we have the probability of the following events.

(i) Event 1: e has not any intentions to stop this game and its probability as $P(\text{Event } 1) \ge (1 - 1/\Phi)^{\Phi}$

(ii) Event 2: A_{AKR1} has the capacity to stop this game and its probability as $P(\text{Event } 2) \ge \xi$

(iii) Event 3: it can don the forgery for target identity and its probability as $P(\text{Event } 1) \ge 1/\Phi$

So, $P(\text{Event } 1) P(\text{Event } 2)\, P(\text{Event } 3) = 1/\Phi + (1 - 1/\Phi)^{\Phi}\xi$

In Game 2, we test the property of unforgeability of our proposed CB-AS scheme against Type 1 attacker (A_{AKR1}).A_{AKR1}struggles to forge the proposed scheme signature and solve HECDLP with the help of another entity e by using the advantage of $\text{Adve}^{\text{HECDLP}} = 1/\Phi + (1 - 1/\Phi)^{\Phi}\xi$. Note that Φ represents maximum number of queries. □

Proof. When e received $\pi = \gamma.\mathscr{D}$, then his task is to extract the unknown γ from π. Further, it can do the following Oracles:

(i) Setup (.)-Oracle: e set a param as Game 1, and set $\Theta = \pi$. Then, A_{AKR1} ask for the queries same as Game 1.

Finally, by using the concept of forking lemma, e returns two signatures that are $(\alpha_1^*, \omega_1^*, \zeta^*)$ and $(\alpha_1^*, \omega_1^*, \zeta^{**})$. Thus, $\gamma = \zeta^* - \zeta^{**}/\ell_1 - \ell_2$ will be the solution of HECDLP.

In the probability analysis, taking into account the above game, we have the probability of the following events.

(i) Event 1: e has not any intentions to stop this game and its probability as $P(\text{Event } 1) \ge (1 - 1/\Phi)^{\Phi}$

(ii) Event 2: A_{AKR1} has the capacity to stop this game and its probability as $P(\text{Event } 2) \ge \xi$

(iii) Event 3: it can don the forgery for target identity and its probability as $P(\text{Event 1}) \geq 1/\Phi$

So, $P(\text{Event 1})P(\text{Event 2})P(\text{Event 3}) = 1/\Phi +$ $(1 - 1/\Phi)^{\Phi}\xi$.

In Game 3, we are explaining the unforgeability of our proposed CB-AS scheme against Type 2 attacker $(A_{AKR2}).A_{AKR2}$ is the malicious certifiers attacker; its work is to forged the proposed scheme signature and solve hyperelliptic curve discrete logarithm problem (HECDLP) with the help of another entity e by using the advantage of $\text{Adve}^{\text{HECDLP}} = 1/\Phi + (1 - 1/\Phi)^{\Phi}\xi$. Note that Φ represents maximum number of queries. □

Proof. When e received $\pi = \gamma.\mathscr{D}$, then his task is to extract the unknown γ from π. Further, it can do the following oracles.

(i) Setup (.)-Oracle: e set a param $\mathscr{F} = \{\mathscr{D}, F^{n}, \hbar\ell, \hbar_0, \hbar_1, \hbar_2, \Theta\}$, and $\Theta = \eta.\mathscr{D} = \pi$. Then, e gives $\mathscr{F}$ and η to A_{AKR2}. Further, e choose ν is an index sustaining $1 \leq \nu \leq \ \Phi_{\hbar_0}$, where $\Phi_{\hbar_0}$ is a number of query requests for $\hbar_0$-Oracle.

Then, A_{AKR2} ask for the same queries as Game 1 neglecting the public key replacement (.)-Oracle and certificate generation (.)-Oracle.

Finally, A_{AKR2} returns a forged signature $\psi^* = (\alpha^*, \omega^*, \zeta^*)$ on m^*, though, by using the concept of forking lemma**,** e returns two signatures that are $(\alpha_1^*, \omega_1^*, \zeta_1^*)$ and $(\alpha_1^*, \omega_1^*, \zeta_2^*)$. Thus, $\zeta_1^*.\mathscr{D} = \omega_1^* + \ell_1.\pi + r.\sigma_i + R.\alpha_1^*$ and $\zeta_2^*.\mathscr{D} = \omega_1^* + \ell_2.\pi + r.\sigma_i + R.\alpha_1^*$. So, $(\zeta_1^* - \zeta_2^*).\mathscr{D} = (\ell_1 - \ell_2).\gamma.\mathscr{D} = \gamma = \zeta_1^* - \zeta_2^*/\ell_1 - \ell_2$ will be the solution of HECDLP.

In the probability analysis, taking into account the above game, we have the probability of the following events.

(i) Event 1: e has not any intentions to stop this game and its probability as $P(\text{Event 1}) \geq (1 - 1/\Phi)^{\Phi}$

(ii) Event 2: A_{AKR2} has the capacity to stop this game and its probability as $P(\text{Event 2}) \geq \xi$

(iii) Event 3: it can don the forgery for target identity and its probability as $P(\text{Event 1}) \geq 1/\Phi$

So, $P(\text{Event 1})\,P(\text{Event 2})P(\text{Event 3}) = 1/\Phi +$ $(1 - 1/\Phi)^{\Phi}\xi$**.**

In Game 4, we intend to prove the unforgeability of our proposed CB-AS scheme against Type 2 attacker $(A_{AKR2}).A_{AKR2}$ is struggles to forge the proposed scheme signature and solve HECDLP with the help of another entity e by using the advantage of $\text{Adve}^{\text{HECDLP}} = 1/\Phi + (1 - 1/\Phi)^{\Phi}\xi$. Note that Φ represents maximum number of queries. □

Proof. When e received $\pi = \gamma.\mathscr{D}$, then his task is to extract the unknown γ from π. Further, it can do the following oracles.

(i) Setup (.)-Oracle: e set a param as Game 3, and set $\Theta = \pi$. Then, A_{AKR2} ask for the same queries as Game 3

Finally, by using the concept of forking lemma**,** e returns two signatures that are $(\alpha_1^*, \omega_1^*, \zeta^*)$ and $(\alpha_1^*, \omega_1^*, \zeta^{**})$. Thus, $\gamma = \zeta^* - \zeta^{**}/\ell_1 - \ell_2$ will be the solution of HECDLP.

In the probability analysis, taking into account the above game, we have the probability of the following events.

(i) Event 1: e has not any intentions to stop this game and its probability as $P(\text{Event 1}) \geq (1 - 1/\Phi)^{\Phi}$

(ii) Event 2: A_{AKR2} has the capacity to stop this game and its probability as $P(\text{Event 2}) \geq \xi$

(iii) Event 3: it can don the forgery for target identity and its probability as $P(\text{Event 1}) \geq 1/\Phi$

So, $P(\text{Event 1})(\text{Event 2})P(\text{Event 3}) = 1/\Phi +$ $(1 - 1/\Phi^{\Phi}\xi)$. □

6. Performance Evaluation

In this section, we evaluate performance evaluation of the proposed scheme in terms of computation and communication costs.

6.1. Computational Cost. Suppose HDML, PM, PML, and P denote hyperelliptic curve divisor multiplication, multiplication operation on pairing, point multiplication on elliptic curve, and pairing operations, respectively. We picked the consuming time for PM, PML, and P as 4.31, 0.97, and 14.90 milliseconds (ms) from [23]; they did this experiment through the computer system with specifications of Intel Core i7-4510U Central Processing unit, 2.0 Gigahertz, Eight Giga Byte Random Access Memory, MIRACL, and Windows 7 Home Basic 64-bit OS. We then further picked the consuming cost for HDML from [21, 22] that is 0.48 ms. On the basis of these findings, we compared our scheme with similar published schemes that are of Wang et al. [15], Li et al. [16], and Li et al. [17]. The major findings obtained from the comparison are mentioned in Table 2 and depicted in Figure 2, which are as follows: Wang et al. [15] consumes $2P + 5\text{PM} = 2 * 14.90 + 5 * 4.31 = 51.35$ms; Li et al. [16] consumes $6P + 7\text{PM} = 6 * 14.90 + 7 * 4.31 = 119.57$ms; and Li et al. [17] consumes $16\text{PML} = 16 * 0.97 = 15.52$ms, and they proposed scheme consumes $7\text{HDML} = 7 * 0.48 = 3.36$ms, respectively. Hence, from the above calculation, it is obvious that the proposed scheme requires less running time from the schemes proposed by Wang et al. [15], Li et al. [16], and Li et al. [17].

6.2. Communication Cost. Suppose $|m|$, $|G|$, $|q|$, and $|n|$ denote the size of message, size of group parameter of bilinear pairing, parameter size of elliptic curve, and parameter size of hyperelliptic curve, respectively. We picked the utilized size in bits for $|m|$, $|G|$, $|q|$, and $|n|$ as 1024, 1024, 160, and 80 [21, 22]. On the basis of this data, we compared the proposed scheme with similar published schemes presented by Wang et al. [15], Li et al. [16], and Li et al. [17], which are presented in Table 3. Then, in the last column of Table 3, by using the above-consuming bits for $|m|$, $|G|$, $|q|$, and $|n|$, we have calculated the total communication cost of proposed

Table 2: Comparison of the proposed scheme with the existing schemes in terms of computation cost.

Schemes	Sender	Receiver	Total (ms)
Wang et al. [15]	2PM = 8.62	$2P + 3PM = 42.73$	51.35
Li et al. [16]	2P4PM = 47.04	$4P + 3PM = 72.53$	119.57
Li et al. [17]	9PML = 8.73	7PML = 6.79	15.52
Proposed scheme	3HDML = 1.44	HDML = 1.924	3.36

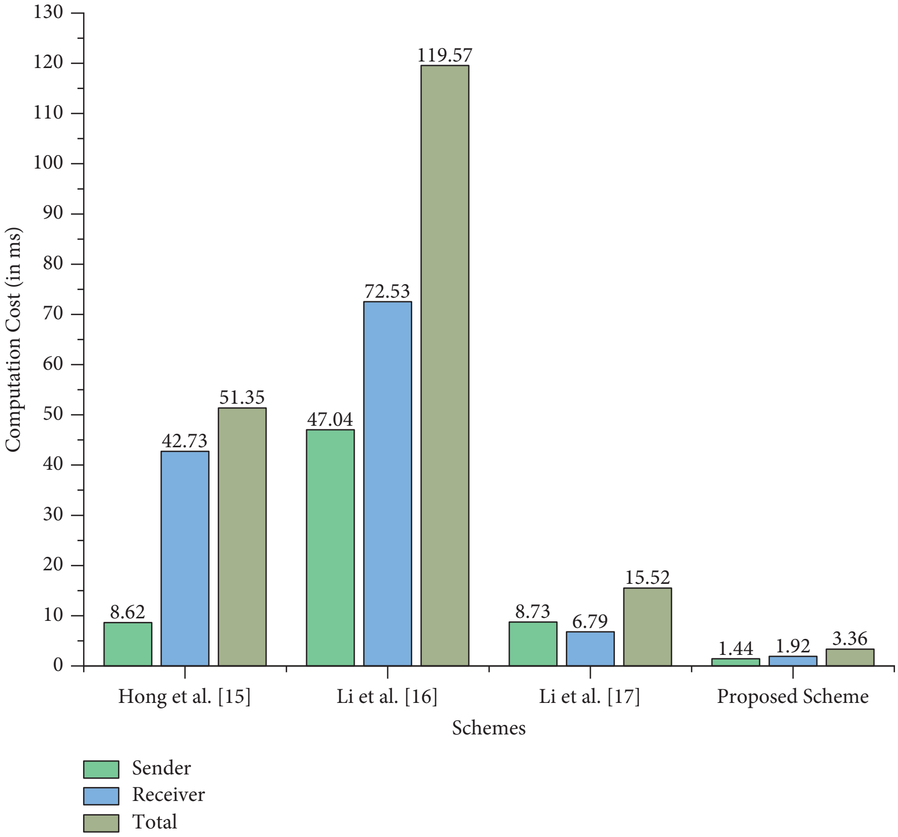

Figure 2: Comparison of computation cost (in ms).

Table 3: Comparison of the proposed scheme with the existing schemes in terms of communication cost.

Schemes	Communication cost	Communication cost in bits
Wang et al. [15]	$\lvert m\rvert + 2\lvert G\rvert$	$\lvert 1024\rvert + 2 * \lvert 1024\rvert = 3072$
Li et al. [16]	$3\lvert m\rvert + 8\lvert G\rvert$	$3 * \lvert 1024\rvert + 8 * \lvert 1024\rvert = 11264$
Li et al. [17]	$2\lvert m\rvert + 7\lvert q\rvert$	$2 * \lvert 1024\rvert + 7 * \lvert 160\rvert = 3168$
Proposed scheme	$\lvert m\rvert + 3\lvert n\rvert$	$\lvert 1024\rvert + 3 * \lvert 80\rvert = 1264$

scheme and those that are presented by Wang et al. [15], Li et al. [16], and Li et al. [17], and the results are described in Table 3 and illustrated in Figure 3, respectively. The results show that the proposed scheme requires less amount of bits during communication.

7. Conclusion

IoD networks are equipped with cutting-edge technologies that can be used for a wide range of civilian and commercial applications. It does, however, have a lot of drawbacks, the most significant of which being security and privacy issues. In this article, we proposed a CB-AS scheme to address the security and privacy concerns of IoD networks. Unfortunately, existing CB-AS construction models rely on pairing and elliptic curve-based operations, which are computationally costly for small drones. As a result, in this paper, we provided a new construction model of CB-AS scheme, which is based on the HECC, an enhanced variant of the elliptic curve with a smaller parameter and key size (80 bits). A security analysis demonstrates that the proposed scheme provides substantial protection against malicious entity from forging the authentication request and responses of others. When compared to relevant schemes, it was found that the

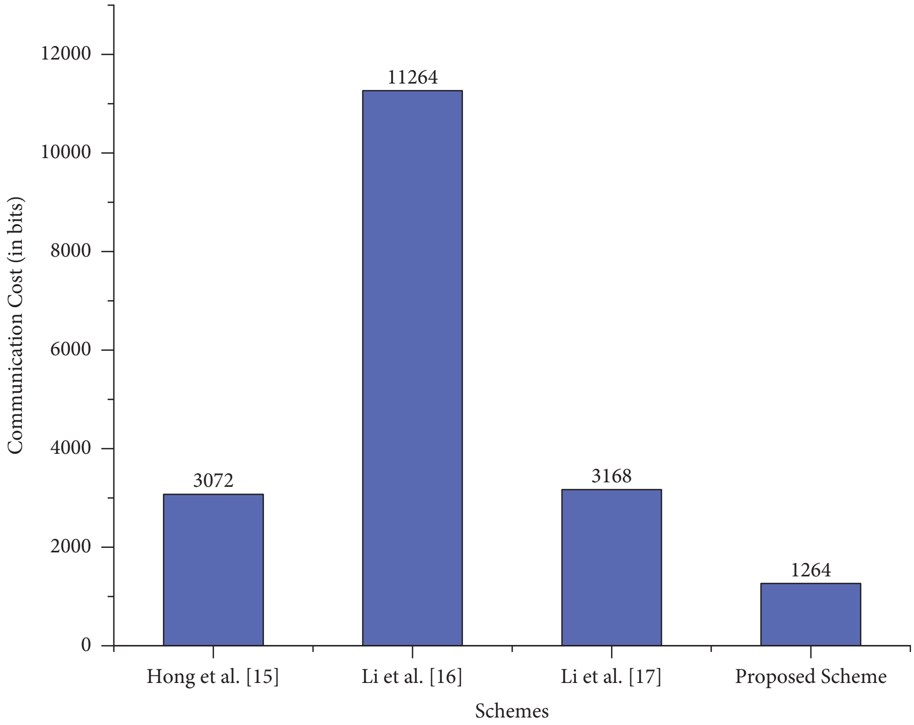

Figure 3: Comparison of communication cost (in bits).

proposed scheme has the lowest computation and communication costs, with 3.36 milliseconds and 1264 bits, respectively, indicating that the proposed scheme is efficient in both computation and communication costs.

Data Availability

All data generated or analyzed during this study are included in this published article.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was also supported by Taif University, Saudi Arabia (TURSP-2020/349).

References

[1] F. Noor, M. A. Khan, A. Al-Zahrani, I. Ullah, and K. A. Al-Dhlan, “A review on communications perspective of flying ad-hoc networks: key enabling wireless technologies, applications, challenges and open research topics,” *Drones*, vol. 4, no. 4, p. 65, 2020.

[2] S. Shakoor, Z. Kaleem, M. I. Baig, O. Chughtai, T. Q. Duong, and L. D. Nguyen, “Role of UAVs in public safety communications: energy efficiency perspective,” *IEEE Access*, vol. 7, pp. 140665–140679, 2019.

[3] V. Sharma, “Advances in drone communications, state-of-the-art and architectures,” *Drones*, vol. 3, no. 1, p. 21, 2019.

[4] E. Yanmaz, S. Yahyanejad, B. Rinner, H. Hellwagner, and C. Bettstetter, “Drone networks: communications, coordination, and sensing,” *Ad Hoc Networks*, vol. 68, pp. 1–15, 2018.

[5] S. Zhang, H. Zhang, and L. Song, “Beyond D2D: full dimension UAV-to-everything communications in 6G,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6592–6602, 2020.

[6] M. A. Khan, H. Shah, S. U. Rehman et al., “Securing Internet of drones with identity-based proxy signcryption,” *IEEE Access*, vol. 9, pp. 89133–89142, 2021.

[7] C. Lin, D. He, N. Kumar, K.-K. R. Choo, A. Vinel, and X. Huang, “Security and privacy for the Internet of drones: challenges and solutions,” *IEEE Communications Magazine*, vol. 56, no. 1, pp. 64–69, 2018.

[8] S. Hussain, K. Mahmood, M. K. Khan, C. M. Chen, B. A. Alzahrani, and S. A. Chaudhry, “Designing secure and lightweight user access to drone for smart city surveillance,” *Computer Standards & Interfaces*, vol. 80, Article ID 103566, 2021.

[9] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, “Aggregate and verifiably encrypted signatures from bilinear maps,” in *Proceedings of the EUROCRYPT’03, LNCS 2656*, pp. 416–432, Warsaw, Poland, May 2003.

[10] W. Diffie and M. Hellman, “New directions in cryptography,” *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[11] A. Shamir, “Identity-based cryptosystems and signature schemes,” in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques*, pp. 47–53, Paris, France, April 1984.

[12] S. S. Al-Riyami and K. G. Paterson, “Certificateless public key cryptography,” in *Proceedings of the International Conference On the Theory and Application of Cryptology and Information Security*, pp. 452–473, Springer, Taipei, Taiwan, November 2003.

[13] Y. W. Hwang and I. Y. Lee, "A lightweight certificate-based aggregate signature scheme providing key insulation," *Computers, Materials & Continua*, vol. 69, no. 2, pp. 1747–1764, 2021.

[14] J. K. Liu, J. Baek, and J. Zhou, "Certificate-based sequential aggregate signature," in *Proceedings of the 2nd ACM Conference on Wireless Network Security*, pp. 21–28, ACM, Zurich, Switzerland, March 2009.

[15] H. Wang, J. Li, C. Lai, and Z. Wang, "A provably secure aggregate authentication scheme for unmanned aerial vehicle cluster networks," *Peer-to-Peer Networking and Applications*, vol. 13, no. 1, pp. 53–63, 2020.

[16] J. Li, M. Zhao, Y. Ding, D. Y. W. Liu, Y. Wang, and H. Liang, "An aggregate authentication framework for unmanned aerial vehicle cluster network," in *Proceedings of the 2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pp. 1249–1256, Xiamen, China, December 2020.

[17] J. Li, Y. Wang, Y. Ding, W. Wu, C. Li, and H. Wang, "A certificateless pairing-free authentication scheme for unmanned aerial vehicle networks," *Security and Communication Networks*, vol. 2021, Article ID 9463606, 10 pages, 2021.

[18] A. Kar, X. Liu, and F. Li, "An efficient and low-cost certificateless aggregate signature scheme for wireless sensor networks," *Journal of Information Security and Applications*, vol. 61, 2021.

[19] G. K. Verma, B. B. Singh, N. Kumar, O. Kaiwartya, and M. S. Obaidat, "PFCBAS: pairing free and provable certificate-based aggregate signature scheme for e-healthcare monitoring system," *IEEE Systems*, vol. 14, 2019.

[20] G. K. Verma, B. B. Singh, N. Kumar, and V. Chamola, "CB-CAS: certificate-based efficient signature scheme with compact aggregation for industrial Internet of things environment," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2563–2572, 2020.

[21] M. Asghar Khan, I. Ullah, A. Alkhalifah et al., "A provable and privacy-preserving authentication scheme for UAV-enabled intelligent transportation systems," *IEEE Transactions on Industrial Informatics*, p. 1, 2021.

[22] I. Ullah, M. A. Khan, F. Khan et al., "An efficient and secure multi-message and multi-receiver signcryption scheme for edge enabled Internet of vehicles," *IEEE Internet of Things Journal*, p. 1, 2021.

[23] C. Zhou, "An improved lightweight certificateless generalized signcryption scheme for mobile-health system," *International Journal of Distributed Sensor Networks*, vol. 15, no. 1, 2019.