Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.049608

ARTICLE

Securing the Internet of Health Things with Certificateless Anonymous Authentication Scheme

Nisreen Innab*

Department of Computer Science and Information Systems, College of Applied Sciences, AlMaarefa University, Riyadh, 13713, Saudi Arabia

*Corresponding Author: Nisreen Innab. Email: ninnab@um.edu.sa

Received: 12 January 2024 Accepted: 06 June 2024 Published: 15 August 2024

ABSTRACT

Internet of Health Things (IoHT) is a subset of Internet of Things (IoT) technology that includes interconnected medical devices and sensors used in medical and healthcare information systems. However, IoHT is susceptible to cybersecurity threats due to its reliance on low-power biomedical devices and the use of open wireless channels for communication. In this article, we intend to address this shortcoming, and as a result, we propose a new scheme called, the certificateless anonymous authentication (CAA) scheme. The proposed scheme is based on hyperelliptic curve cryptography (HECC), an enhanced variant of elliptic curve cryptography (ECC) that employs a smaller key size of 80 bits as compared to 160 bits. The proposed scheme is secure against various attacks in both formal and informal security analyses. The formal study makes use of the Real-or-Random (ROR) model. A thorough comparative study of the proposed scheme is conducted for the security and efficiency of the proposed scheme with the relevant existing schemes. The results demonstrate that the proposed scheme not only ensures high security for health-related data but also increases efficiency. The proposed scheme's computation cost is 2.88 ms, and the communication cost is 1440 bits, which shows its better efficiency compared to its counterpart schemes.

KEYWORDS

Internet of things; internet of health things; security; authentication; hyperelliptic curve cryptography

1 Introduction

The Internet of Health Things (IoHT) is a networked system that incorporates various biomedical devices, including smart wearables, implants, and ingestible electronics. These devices are integrated with appropriate software applications to facilitate the collection, analysis, and dissemination of physiological data through the internet [1,2]. Physiological data often encompasses many health-related indications such as blood pressure, chest sounds, body temperature, respiration rate, electrocardiogram (ECG), patient posture, breathing rate, and other vital parameters [3–7]. In addition, IoHT systems could be used to update environmental factors such as patient care settings, room conditions, laboratory shift timings, treatment durations, and staff-to-patient ratios. The health information system maintains computerized records of patients' environmental conditions and health-related information, accessible to medical professionals anytime the patient enters the hospital.

Privacy and security issues often occur in IoHT systems because biomedical sensors and user-customized devices are frequently involved in internet-based communication. The typical IoHT system architecture is shown in Fig. 1 [8]. To provide secure connectivity in an IoHT system, the primary security challenge is validating the integrity of data transmitted across an unsecured wireless link. The second security concern is ensuring receiver anonymity, which means that only the sender is aware of the identities of the receivers. An intruder can, for instance, intercept communication between biomedical devices and sensors to steal or forge health-related data. This system may also be susceptible to the "greedy behavior attack," one of the most aggressive DoS attacks. It attempts to prevent authorized nodes from accessing the communication channel [9]. Unfortunately, most IoHT devices have low processing and storage capabilities, making it impossible for them to execute complex cryptographic computations to protect against such attacks. The majority of public key cryptosystems mentioned in the literature require a lot of computation, making them unsuitable for IoHT systems.

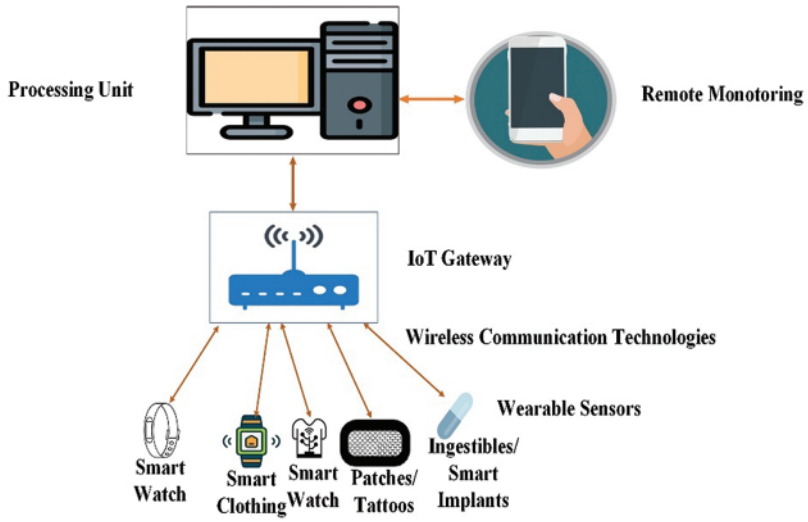

Figure 1: A typical architecture of IoHT system

To secure IoHT systems, authentication mechanisms based on the digital signature system can be implemented [10]. This mechanism uses a shared key to provide not just authentication and privacy but also the other three primary requirements of confidentiality, integrity, and non-repudiation [11]. Two of the most common authentication mechanisms used in public-key cryptosystems are Identity-Based Cryptography (IBC) and Public Key Infrastructure (PKI). It is essential in the PKI setting to have a reliable unforgeable connection between a user's identity and their public keys. Because of this, a Certificate Authority (CA) that gives each link its signature is required. With certificates, the CA limits the public key to serve as the identification of a participant [12]. Problems with certificates expiring, being distributed, and being stored are only a few of the drawbacks of PKI systems. As an alternative, IBC is promoted as a means of cutting down on the cost of managing public keys [13]. The trustworthy Private Key Generator (PKG) has direct knowledge of the private keys of the participants, which is at the cost of private key escrow problems [14,15]. The issue of key escrow in authentication methods can be solved by using a certificateless cryptosystem with a signature strategy. Additionally, there is the problem of receiver anonymity, which means that only the sender knows who the receivers are. Fortunately, this obstacle can be circumvented by using anonymous authentication.

Motivation and Contributions

Authentication schemes are usually built using computing-based cryptographic operations like bilinear pairing, Rivest-Shamir-Adleman (RSA), and ECC, and then evaluated to see how well the proposed scheme works. These operations, on the other hand, have high computation and communication costs. As a result, HECC, an enhanced variant of ECC that employs 80-bit keys, identities, and certificate sizes to give the same level of security as ECC, bilinear pairing, and RSA [16], can be employed. As a result, for resource-constrained devices in IoHT systems, HECC would be a preferable alternative. This work provides a one-of-a-kind IoHT solution called the certificateless anonymous authentication scheme. The scheme is based on the HECC concept and has a small key size. The following are some of the key contributions of the undertaken research work:

1. We propose an efficient anonymous authentication scheme in certificateless settings for IoHT systems.
2. To overcome constraints such as low processing capabilities associated with biomedical devices and sensors, the proposed scheme uses a public-key cryptography method based on the HECC concept.
3. The proposed scheme is secure against various attacks in both formal and informal security analyses. The formal study makes use of the Real-or-Random (ROR) model.
4. Lastly, we show that the proposed scheme has lower costs for both computation and communication than relevant existing schemes.

The following section outlines the organizational structure of the remaining portions of the article. The literature review is further discussed in Section 2. In Section 3, we provide the preliminary information. The CAA plan under consideration is included in Section 4. Section 5 of the document encompasses the security analysis. Section 6 of the document encompasses a comprehensive performance assessment analysis, while Section 7 presents conclusions and future work.

2 Related Work

This section focuses on the security and privacy concerns of the IoHT system while using an authentication scheme. Also, we have given the limitations of the scheme, which appear in this section's literature review in Table 1. Chen et al. [17] proposed a telemedicine information system authentication scheme based on dynamic identity. According to the authors, their scheme overcomes the problem of user anonymity. Jiang et al. [18] pointed out that the Chen et al.'s scheme did not ensure user anonymity or untraceability, which could lead to identity guessing and tracking attacks. They presented an updated strategy for maintaining user anonymity, saying their scheme could survive any attack. On the other hand, Kumari et al. [19] revealed that Jiang et al.'s scheme is vulnerable to several attacks, including password guessing, impersonation, DoS attacks, and even improper login request verification.

Chen et al. [20] proposed an authentication scheme for medical data interchange in the cloud environment to prevent security issues and safeguard patients' health data. Neither patient privacy nor message authentication were ensured by the proposed scheme, as shown by the work proposed by Chiou et al. [21]. Therefore, they improved the cloud-based healthcare system's privacy authentication process. Patients' privacy was not protected in Mohit et al.'s [22] analysis of the method reported in Chiou et al.'s scheme, which they found could be compromised by attacks from stolen mobile devices. To strengthen healthcare security and reduce its burdensome complexity, they designed a lightweight two-factor authentication solution using cloud computing. It was pointed out by Li et al. [23] that the

scheme proposed by Mohit et al. is susceptible to the forged inspection report and cannot guarantee patient anonymity or data confidentiality.

Table 1: Limitations of existing schemes

No.	Scheme	Limitations
1	Chen et al. [17]	• The problem of user anonymity • High computational cost • High communication overheads
2	Jiang et al. [18]	• Not safeguarded from password guessing attack • Not withstand impersonation attack • Not resisting the DoS attack • Facing problems like improper login request verification
3	Chen et al. [20]	• This scheme does not ensure patient privacy • Also, it does not provide the message authentication
4	Chiou et al. [21]	• This scheme does not ensure patient privacy
5	Mohit et al. [22]	• The scheme is susceptible to the forged inspection report • Also, it cannot guarantee patient anonymity or data confidentiality
6	Li et al. [23]	• High computational cost • High communication overheads
7	Saeed et al. [24]	• The scheme is vulnerable to a forgery attack
8	Liao et al. [25]	• High computational cost • High communication overheads
9	He et al. [26]	• High computational cost • High communication overheads
10	Kasyoka et al. [27]	• High computational cost • High communication overheads
11	Liu et al. [28]	• High computational cost • High communication overheads

Saeed et al. [24] proposed an online/offline certificateless signature approach in wireless body area networks based on the internet of things to build a heterogeneous remote anonymous authentication mechanism. However, Liao et al. [25] demonstrated that the approach of Saeed et al. is vulnerable to a forgery attack in which no information other than public system parameters is required. He et al. [26] proposed anonymous authentication for WBAN and shown that their scheme is both secure and efficient when compared to its counterpart schemes. Kasyoka et al. [27] proposed an ECC-based authentication scheme for WBANs. The proposed scheme was both certificateless and pairing-free. The authors of the presented scheme of Kasyoka et al. claimed that their scheme is efficient in terms of communication cost and running time. Liu et al. [28] introduced a novel and efficient anonymous authentication scheme for WBANs that ensures that doctors and patients are legal in a secure manner.

All of the above-mentioned solutions involve the use of cryptographic techniques; these schemes are mostly based on ECC and bilinear pairing, both of which have prohibitively expensive communication and computation costs. However, the proposed scheme is based on the concept of HECC, a more sophisticated form of ECC. HECC uses an 80-bit key size, which is half as large as ECC's key size, but it still offers the same level of security as ECC and bilinear pairing.

3 Preliminaries

This section explains some of the key concepts and materials that will be used in constructing the proposed scheme.

3.1 Hyperelliptic Curve

In 1989, Koblitz introduced a group law for the Jacobian of a Hyperelliptic curve (HE). It is defined by the Jacobian of genus £ and is based on the discrete logarithm Problem.

It typically has $n^{£}$ points, where n is the number of Jacobian components. Under the Decisional Diffie-Hellman assumption, the HE-based authentication and key management scheme is completely safe. HE curves are a particular subclass of elliptic curves in algebra, which can be viewed as a generalization. For a HE, we have:

(HE): $\varnothing^2 + h(\Delta)\varnothing = f(\Delta)$, where $f(\Delta)$ denotes a monic polynomial and his degree defined over $Field_n$ as $2£ + 1$ and $h(\Delta)$ represents a polynomial that is defined over $Field_n$, which may be equal or less from £.

3.2 Hyperelliptic Curve Deffie-Hellman (HEDH) Problem

Suppose the random triple $(\mathcal{D},\ V.\mathcal{D},\ U.\mathcal{D})$ of the HEDH Problem is given, then to compute the value $V.U.\mathcal{D}$ is called HEDH problem, where V, U are belongs to $Field_n$.

3.3 Hyperelliptic Curve Discrete Logarithm (HEDL) Problem

Suppose the random tuple $(\mathcal{D},\ V.\mathcal{D})$ of the HEDL Problem is given, then to compute the value $V.\mathcal{D}$ is called the $HEDL$ problem, where V belongs to $Field_n$. The symbols used in the scheme are illustrated in Table 1.

4 Proposed Scheme

In this section, the network model is provided first, followed by the construction and correctness of the proposed scheme.

4.1 Network Model

Depending on the requirements, the network model of IoHT systems can be implemented in several topologies; one such networking architecture is shown in Fig. 2. Wearable Biomedical Devices (WBD), Network Manager (NMGR)/Service Provider, Application Provider (APDR), and IoT gateway are all included in this network model. Each entity's function is described as follows:

1. APDR: The APDR will assess a patient's health and generate health-related information. Following this, it sends its encrypted identity along with a public number send $(E_{Ids},\ \eta_s)$ over an un-secure channel to NMGR/Service Provider for registration. Upon receiving $(E_{Ids},\ \eta_s)$, NMGR/Service Provider can compute the secret key as SK, recover the user the identity ID_s using decryption method, and send the encrypted version of partial private key (Pr_{Ids}) to APDR. Then APDR first decrypt Pr_{Ids} and set his public key pair as $(\eta_s,\ \xi_s)$ and private key pair as $(\delta_s,\ \chi_s)$.
2. WBD: To register WBD, the user compute and send (E_{Idr}, η_r) to NMGR/Service Provider via un-secure channel. Upon receiving $(E_{Idr},\ \eta_r)$, NMGR/Service Provider can compute the secret key SK, recover ID_r, then compute and Pr_{Ids} by using un-secure network to the identity (ID_r).

Then a user with identity (ID_r), Then a user with identity (ID_r), first recover (χ_r, ξ_r), set his public key pair as (η_r, ξ_r) and private key pair as (δ_r, χ_r). When WBD gets a partial private key from NMGR/Service Provider and a digital signature from APDR with a key management request, it generates its own private and public keys before verifying the digital signature. If the signature verification procedure is successful, WBD will produce a secret key, encrypt health-related data with it, and send it to the APDR.

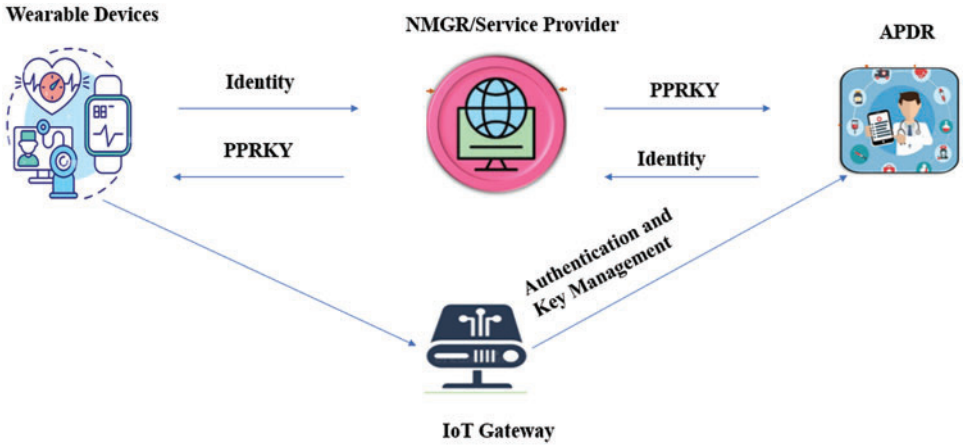

Figure 2: Proposed network model

1. NMGR/Service Provider: When NMGR/Service Provider receives identity from both WBD and APDR, this entity will act as a key generation canter in certificateless cryptography and will be responsible for creating the partial private key for both WBD and APDR.
2. IoT Gateway: The IoT gateway router can be used to connect any things that communicate using wireless technologies.

4.2 Construction of the Proposed Scheme

The proposed scheme has been constructed on the basis of the following phases and all the symbols used in the construction is included in Table 2.

Table 2: Notations used in the proposed scheme

S. No.	Symbol	Descriptions
1	l	Security parameter with the size of 80 bits
2	λ	It is selected by NMGR from the finite field of hyperelliptic curve and set as his secret key
3	$\gamma = \lambda.\mathcal{D}$	This value is calculated by NMGR and then set as his public key
4	Φ	It is set by NMGR and its work is that it is available in a network to each user
5	$Hash_a$, $Hash_b$, $Hash_c$	It is used to perform the function (SHA 256) irreversible hash functions

(Continued)

Table 2 (continued)

S. No.	Symbol	Descriptions
6	HE	A selected hyperelliptic curve for this scheme with genus 2
7	$Field_n$	A finite field of the hyperelliptic curve for with genus 2
8	.	It is used for divisor multiplication in hyperelliptic curve
9	δ_s	The secret value utilized by APDR
10	δ_r	The secret value utilized by WBD
11	ID_s	The identity utilized by APDR
12	ID_r	The identity utilized by WBD
13	r_a	It represent the hash value as $r_a = Hash_a\ (ID_s,\ \eta_s,\ \xi_s)$
14	r_b	It represents the hash value as $r_b = Hash_b\ (ID_s,\ \lambda\eta_s)$
15	$\eta_s,\ \zeta_s$	The public key pair utilized by APDR
16	$\eta_r,\ \zeta_r$	The public key pair utilized by WBD
17	$\delta_s,\ \zeta_s$	The private key pair utilized by APDR
18	$\delta_r,\ \zeta_r$	The private key pair utilized by WBD
19	R_a	It represents the hash value as $R_a = Hash_a\ (ID_r,\ \eta_r,\ \xi_r)$
20	R_b	It represent the hash value as $R_b = Hash_b\ (ID_r,\ \lambda\eta_r)$
21	$E_{\mathcal{G}}$	It is used for encryption functions by using the secret key $\mathcal{G}$
22	$\mathcal{G}$	It is used for the Shared secret key
23	$D_{\mathcal{G}}$	It is used for the decryption function by using the secret key $\mathcal{G}$
24	C	It represents the cipher text
25	M	It represents the plaintext

i. Initialization

Given an 80-bit hyper elliptic curve parameter l for the security of the proposed scheme, NMGR selects λ and computes $\gamma = \lambda.\mathcal{D}$ as a private and public key. It then set $\Phi = \{\gamma,\ \mathcal{D},\ Hash_a,\ Hash_b,\ Hash_c,\ HE,\ Field_n\}$, where γ denotes the public key of NMGR, $\mathcal{D}$ is the divisor over HE, $Hash_a$, $Hash_b$, and $Hash_c$ are one way cryptographic hash functions with size of 512 bits, HE is a selected hyper elliptic curve with genus 2, and $Field_n$ is the finite field over HE order $n = 80$ bits, respectively. Then, NMGR disclosed Φ to the network and keep private λ.

ii. Registration Phase

In this phase, APDR and WBD will be registered using the following steps:

1. Fig. 3 shows the registration process of APDR, in which the user selects δ_s, computes $\eta_s = \delta_s.\mathcal{D}$, compute $SK = \delta_s.\gamma$, encrypt the identity (ID_s) as $E_{Ids} = E_{SK}(ID_s)$, and send $(E_{Ids},\ \eta_s)$ to NMGR via un-secure channel. Upon receiving $(E_{Ids},\ \eta_s)$, NMGR can compute the secret key as $SK = \eta_s.\lambda$, decrypt the identity as $ID_s = D_{SK}(E_{Ids})$, select ζ_s and compute $\xi_s = \zeta_s.\mathcal{D}$ and $r_a = Hash_a\ (ID_s,\ \eta_s,\ \xi_s)$, and then process $\chi_s = \zeta_s. + r_a.\lambda$. Finally, encrypt the tuple $(\chi_s,\ \xi_s)$ as $Pr_{Ids} = E_{SK}(\chi_s,\ \xi_s)$ and by using un-secure network delivered to the identity (ID_s). Then a user with identity (ID_s), first decrypt Pr_{Ids} as $(\chi_s,\ \xi_s) = D_{SK}\ (Pr_{Ids})$, set his public key pair as $(\eta_s,\ \xi_s)$ and private key pair as $(\delta_s,\ \chi_s)$.

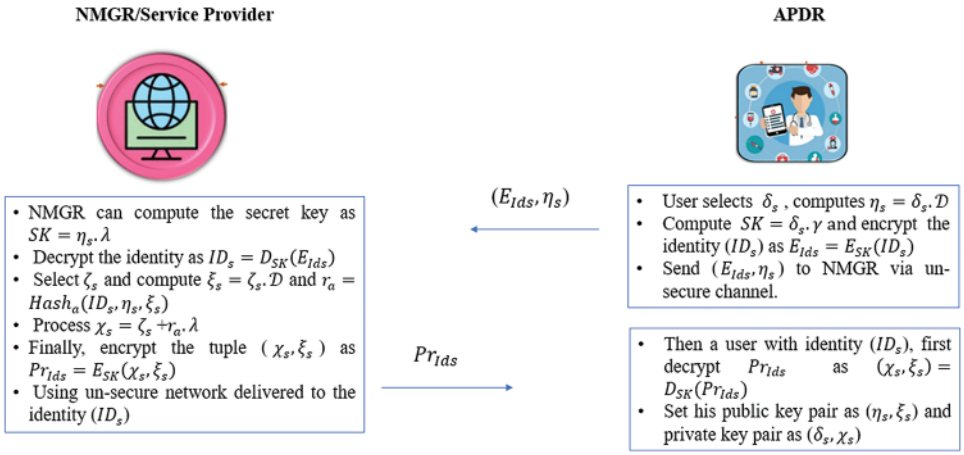

Figure 3: Application provider registration

2. Fig. 4 shows the registration process of WBD, in which the user selects δ_r and computes $\eta_r = \delta_r.\mathcal{D}$ for the identity (ID_r), compute $RK = \delta_r.\gamma$, encrypt the identity (ID_r) as $E_{Idr} = E_{RK}(ID_r)$ and send $(E_{Idr},\ \eta_r)$ to NMGR via secure channel. Upon receiving $(E_{Idr},\ \eta_r)$, NMGR can compute the secret key as $SK = \eta_r.\lambda$ and decrypt the identity as $ID_r = D_{SK}(E_{Idr})$, then select ζ_r, compute $\xi_r = \zeta_r.\mathcal{D}$, $r_b = Hash_a(ID_r,\ \eta_r,\ \xi_r)$, and then process $\chi_r = \zeta_r + r_b.\lambda$. Finally, it encrypts the tuple $(\chi_r,\ \xi_r)$ as $Pr_{Ids} = E_{SK}(\chi_r,\ \xi_r)$ and by using un-secure network delivered to the identity (ID_s). Then a user with identity (ID_r), Then a user with identity (ID_r), first decrypt Pr_{Ids} as $(\chi_r,\ \xi_r) = D_{SK}(Pr_{Ids})$, set his public key pair as $(\eta_r,\ \xi_r)$ and private key pair as $(\delta_r,\ \chi_r)$.

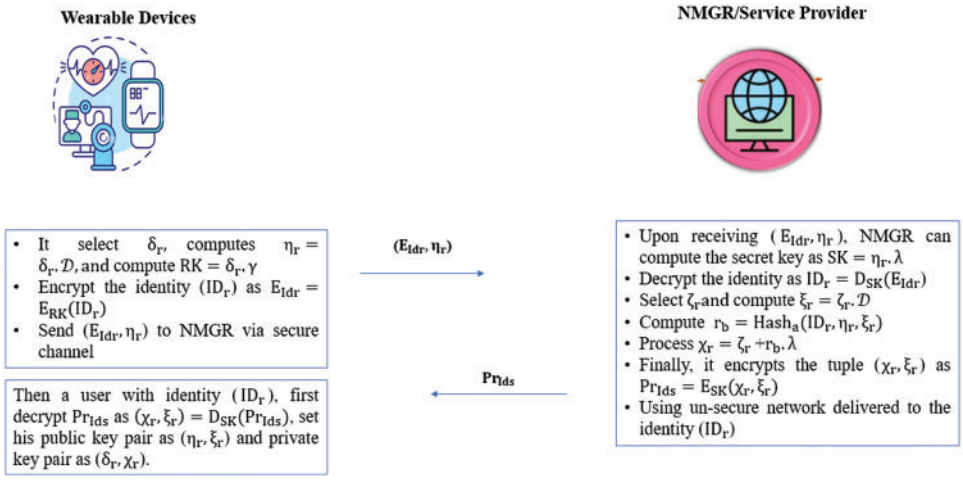

Figure 4: Wearable biomedical devices registration

iii. Authentication and Key Management

Fig. 5 represents the flow of proposed scheme, in which the application provider generates the signature for mutual authentication with WBD by using the following steps:

- The APDR can compute $\omega = \upsilon.\mathcal{D}$, where $\mathcal{D}$ is the divisor of hyper elliptic curve.
- Compute $r_b = Hash_a(ID_r,\ \eta_r,\ \xi_r)$ and $K = \upsilon\ (\eta_r + \xi_r + r_b.\gamma)$
- Calculate $\mathcal{G} = Hash_b(\omega,\ ID_r,\ ID_s,\ \eta_s,\ \xi_s)$ and $\varphi = \delta_s + \upsilon/\mathcal{G} + \delta_s + \chi_s$, then send $(\varphi,\ \omega)$ to WBD.
- Upon reception $(\varphi,\ \omega)$, the receiver compute$K = (\delta_r + \chi_r)\omega$, where δ_r denotes the secret value of WBD.
- Also, compute $\mathcal{G} = Hash_b\ (\omega,\ ID_r,\ ID_s, \eta_s,\ \xi_s)$, $r_a = Hash_a(ID_s,\ \eta_s,\ \xi_s)$, and $\varphi(\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega$, if the equality of $\varphi\ (\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega$ is satisfied, then the mutual authentication can be done between sender and WBD, where $\eta_s,\ \xi_s$ denotes the public key pair of application provider.
- The WBD set the secret key as $K = (\delta_r + \chi_r)\omega$ and done the encryption of a message as $C = Hash_c(K) \oplus (M)$, where $\oplus$ denotes the encryption function which encrypt the message (M) using the secrete key K and finally send C to application provider.
- After reception of (C), the application provider performs decryption function as $M = (Hash_c(K) \oplus C)$, for the recovery of plaintext.

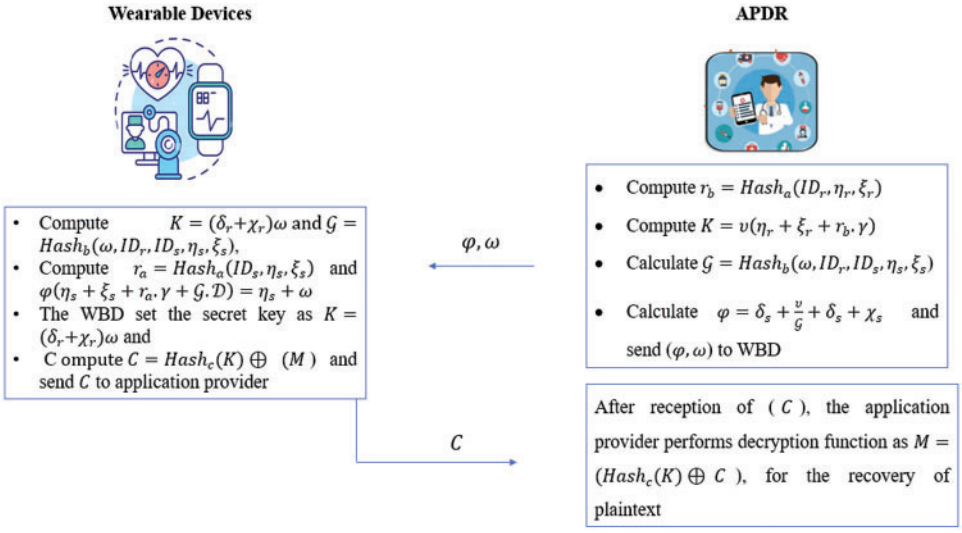

Figure 5: Mutual authentication phase of proposed scheme

4.3 Correctness

As a receiver of $(\varphi,\ \omega)$, the WBD accepts only this pair when the following equation is satisfied:

$$\varphi\ (\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega = \varphi\ (\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \varphi\ (\delta_s.\mathcal{D} + \zeta_s.\mathcal{D} + r_a.\lambda.\mathcal{D} + \mathcal{G}.\mathcal{D})$$

$$= \varphi\ (\delta_s + \zeta_s + r_a.\lambda + \mathcal{G})\ .\mathcal{D} = (\delta_s + \upsilon/\mathcal{G} + \delta_s + \chi_s)\ (\delta_s + \delta_s + \chi_s + \mathcal{G})\ .\mathcal{D}$$

$$= (\delta_s.\mathcal{D} + \upsilon.\mathcal{D}) = \eta_s + \omega$$

Also, WBD can process its secret key as followed:

$$K = (\delta_r + \chi_r)\omega = (\delta_r + \chi_r)\upsilon.\mathcal{D}$$
$$= (\delta_r + \zeta_r + r_b.\lambda)\upsilon.\mathcal{D} = (\delta_r.\mathcal{D} + \zeta_r.\mathcal{D} + r_b.\lambda.\mathcal{D})\upsilon$$
$$= \upsilon(\eta_r + \xi_r + r_b.\gamma) = K.$$

5 Security Analysis

In this section, we provide the provable security based on Random Oracle Model (ROM) and informal security analysis, which are as follows.

5.1 Provable Security Analysis

Here, we are going to prove the confidentiality and unforgeability of our proposed scheme against the following two types of attackers by using the ROM.

Outsider Attacker (ξ_{OUA}): ξ_{OUA} is also known as a key replacement attacker, has the ability to replace user public keys but lacks access to the master key. ξ_{OUA} has access to $Hash_a$ Query, $Hash_b$ Query, $Hash_c$ Query, *Public Key Queries, Partial Private Key Queries, Private Key Queries, Replace Public Key Queries, Sender Query, and Receiver Query*.

Insider Attacker (ξ_{ISA}): The Insider Attacker (ξ_{ISA}), also known as a malevolent NMGR attacker, is one who has access to the master key but is unable to replace the public keys. ξ_{ISA} has access to $Hash_a$ Query, $Hash_b$ Query, $Hash_c$ Query, *Public Key Queries, Private Key Queries, Sender Query, and Receiver Query*.

So, keeping in view the capabilities of the above attackers (ξ_{OUA}, ξ_{ISA}), we have included *Theorems 1* and *2* to prove the confidentiality of our proposed scheme. Then, by using *Theorems 3* and *4*, we have proved that our proposed scheme is unforgeable against (ξ_{OUA}, ξ_{ISA}). The following are the proofs of theorems:

Theorem 1-Confidentiality: Suppose the outsider attacker (ξ_{OUA}), with his advantages OUA_{ξ}, wants to break *IND-CAA-CCA-I* with help of helper called H_{OUA} and his task is to solve HEDH Problem with advantage as followed: $OUA_{\xi}/(Q_{nq} + 1)^2$, where Q_{nq} denotes the number of private key extract query, partial private key extract query, and sender query, respectively.

Proof: Suppose the random triple ($\mathcal{D}$, $V_{OUA}.\mathcal{D}$, $U_{OUA}.\mathcal{D}$) of HEDH Problem is given to H_{OUA} and his task to compute $H = V_{OUA}.U_{OUA}.\mathcal{D}$. The following sub steps represents the processing of *Theorem 1*:

Setup: Given an 80 bit hyper elliptic curve parameter l for the security of the proposed scheme, the H_{OUA} set $\gamma = V_{OUA}.\mathcal{D}$ and $\Phi = \{\gamma, \mathcal{D}, Hash_a, Hash_b, Hash_c, HE, Field_n\}$. Then, ($H_{OUA}$ disclosed Φ to the ξ_{OUA}). Not that, here H_{OUA} has no access to λ.

$Hash_a$ Query: ξ_{OUA} chooses the identity ID_i and sends out $Hash_a$ oracle queries to H_{OUA}. H_{OUA} checks to see if there is a tuple (ID_i, η_i, ξ_i, r_i), in list L_{Hash_a}, if the condition is met, H_{OUA} return with r_i to ξ_{OUA}, else, it select R_i from {0, 1} such that $[R_i = 1] = 1/(Q_{nq} + 1)$, and do the following steps.

- If $R_i = 0$, then it picks r_i randomly, send it to ξ_{OUA}, and update the list L_{Hash_a} with the values r_i and $R_i = 0$.
- If $R_i = 1$, then it set $r_i = P$, send it to ξ_{OUA}, and update the list L_{Hash_a} with the values r_i and $R_i = 1$.

Hash$_b$ Query: ξ_{OUA} chooses the identity ID_i and sends out *Hash*$_b$ oracle queries to H_{OUA}. H_{OUA} checks to see if there is a tuple $(\omega,\ \eta_i,\ \xi_i ID_i,\ \mathcal{G}_i)$, in list L_{Hash_b}, if the condition is met, H_{OUA} return with $\mathcal{G}_i$ to ξ_{OUA}, else, it returns $\mathcal{G}_i$ of its choice and add $(ID_i,\ \mathcal{G}_i)$ into the list L_{Hash_c}.

Hash$_c$ Query: ξ_{OUA} chooses the identity ID_i and sends out *Hash*$_c$ oracle queries to H_{OUA}. H_{OUA} checks to see if there is a tuple (K_i), in list L_{Hash_b}, if the condition is met, H_{OUA} return with K_i to ξ_{OUA}, else, it returns K_i of its choice and add $(ID_i,\ K_i)$ into the list L_{Hash_c}.

Public Key Queries: ξ_{OUA} chooses the identity ID_i and sends out *Public Key* oracle queries to H_{OUA}. H_{OUA} check $(\eta_i,\ \xi_i)$ in the list L_{pKq}, if it is existing, then it delivers $(\eta_i,\ \xi_i)$ to ξ_{OUA}, otherwise the following two conditions should consider:

1. If $R_i = 0$, H_{OUA} compute $\eta_i = \delta_i.\mathcal{D},\ \xi_i = \zeta_i.\mathcal{D}$, and $\chi_i = \zeta_i + Hash_a\ (ID_i,\ \eta_i,\ \xi_i).\ s$ where $\delta_i,\ \zeta_i,\ s$ is the randomly chosen numbers from hyper elliptic curve Jacobian group. Then, H_{OUA} send $(\eta_i,\ \xi_i)$ to ξ_{OUA} and update the list L_{pKq} with $(\eta_i,\ \xi_i)$ and $(\delta_i,\ \chi_i)$.
2. If $R_i = 1$, H_{OUA} compute $\eta_i = \delta_i.\mathcal{D}$ and $\xi_i = \zeta_i.\mathcal{D}$, where $\delta_i,\ \zeta_i$ is the randomly chosen numbers from hyper elliptic curve Jacobian group. Then, H_{OUA} send $(\eta_i,\ \xi_i)$ to ξ_{OUA} and update the list L_{pKq} with $(\eta_i,\ \xi_i)$ and $(\delta_i,\ \zeta_i)$.

Partial Private Key Queries: ξ_{OUA} chooses the identity ID_i and sends out *Partial Private Key* oracle queries to H_{OUA}. Then H_{OUA} check $(\chi_i,\ \xi_i)$ in the list L_{pRKq}, if available, H_{OUA} will send $(\chi_i,\ \xi_i)$ to ξ_{OUA}. Otherwise, the following two conditions should consider:

1. If $R_i = 0$, H_{OUA} call *Public Key Queries*, get (χ_i, ξ_i) and send it to the ξ_{OUA}.
2. If $R_i = 0$, H_{OUA} will not further processed this game and stop the simulation.

Private Key Queries: ξ_{OUA} chooses the identity ID_i and sends out *Private Key* oracle queries to H_{OUA}. Then H_{OUA} check $(\chi_i,\ \xi_i)$ in the list L_{pRiKq}, if available, H_{OUA} will send $(\chi_i,\ \delta_i)$ to ξ_{OUA}. Otherwise, the following two conditions should consider:

1. If $R_i = 0$, H_{OUA} call *Public Key Queries*, get $(\chi_i,\ \delta_i)$ and send it to the ξ_{OUA}.
2. If $R_i = 0$, H_{OUA} will not further processed this game and stop the simulation.

Replace Public Key Queries: ξ_{OUA} choose $(\eta_i^{/},\ \xi_i^{/})$ and send to H_{OUA}. Upon reception $(\eta_i^{/},\ \xi_i^{/})$, H_{OUA} replaces $\eta_i,\ \xi_i$ on $(\eta_i^{/},\ \xi_i^{/})$ and adds into the list L_{RP}.

Sender Query: As a response of this query, H_{OUA} examines of the value R_s by using the following computations:

- If $R_s = 0$, H_{OUA} pick $(\chi_i,\ \delta_i)$ from L_{pRiKq}, if it is not available in L_{pRiKq}, then it call *Private Key Queries, obtain* $(\chi_i,\ \delta_i)$, perform the algorithmic steps as made by APDR, and generate $(\varphi,\ \omega)$. Finally, H_{OUA} will delivered $(\varphi,\ \omega)$ to ξ_{OUA}.
- If $R_s = 1$, H_{OUA} will stop simulation of this process.

Receiver Query: H_{OUA} examines R_r and R_s responds as follows when ξ_{OUA} queries for sender signature and encryption:

- If R_r and $R_s = 0$, H_{OUA} pick $(\chi_i,\ \delta_i)$ from L_{pRiKq}, if it is not available in L_{pRiKq}, then it call *Private Key Queries, obtain* (χ_i, δ_i), perform the alogorithmic steps as made by WBD during signature verifications and APDR decryption of ciphertext, and generate (M). Finally, H_{OUA} will delivered (M) to ξ_{OUA}.
- If $R_s = 1$, H_{OUA} pick the value K_i, compute $M = (Hash_c(K) \oplus C)$, and send M to ξ_{OUA}.
- If $R_r = 1$, H_{OUA} compute $\varphi\ (\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega$, where r_a and $\mathcal{G}$ is selected randomly.

Challenge: ξ_{OUA} *send* (M_{OUA1}, M_{OUA2}) *and* (ID_s^*, ID_r^*) to H_{OUA}, decides on R_r^*, and do the following steps:

- If $R_r^* = 0$, H_{OUA} stop the simulation.
- If $R_r^* = 1$, H_{OUA} compute $\omega^* = U_{OUA}.\mathcal{D}$, obtained K_i^* from L_{Hash_c}, select φ^* randomly, g^* from {0, 1}, compute $C^* = Hash_c(K_i^*) \oplus (M_{g^*})$, and send $(\omega^*,\ \varphi^*,\ C^*)$ to ξ_{OUA}.

Guess: After performing the same number of queries as performed above, ξ_{OUA} outputs with a guess value g' from {0, 1} of g^*. Then, H_{OUA} can perform the following two steps:

- In the event that any steps of the simulation are avoided, H_{OUA} chooses g from {0, 1} at random as its best guess for the solution of $(H = (V_{OUA}.U_{OUA}.\mathcal{D}))$.
- Otherwise, if $g' = g^*$, H_{OUA} return δ_r^* and ζ_r^* from L_{pKq}, further the processing at challenge step will not be aborted and we have $R_r^* = 1$ and $r_r^* = L$. So, H_{OUA} obtains $(\omega^*,\ \varphi^*,\ K_i^*)$ and check the equation: $\dfrac{K^* - (\delta_r^* + \zeta_r^*)\omega^*}{L} = H$, is hold. If yes, H_{OUA} return $g = 1$, otherwise it returns $g = 0$.

Note that, H_{OUA} has no knowledge about $\chi_r^* = \zeta_r^* + r_r.\lambda,\ \gamma = V_{OUA}.\mathcal{D}$, and $\lambda = V_{OUA}$, so λ is not accessible for H_{OUA}. In the challenge phase, $\omega^* = U_{OUA}.\mathcal{D}$; it means that $U_{OUA} = \upsilon$, which is not known to H_{OUA}. In a situation, if $(\omega^*,\ \varphi^*,\ C^*)$ valid then we can get:

$$\frac{K^* - (\delta_r^* + \zeta_r^*)\omega^*}{L} = \frac{U_{OUA}\,(\eta_r^* + \xi_r^* + r_r^*.\gamma) - (\delta_r^* + \zeta_r^*)\omega^*}{L}$$

$$= \frac{U_{OUA}\,(\eta_r^* + \xi_r^* + r_r^*.\gamma) - (\delta_r^* + \zeta_r^*)\omega^*}{L} = \frac{U_{OUA}\,(\eta_r^* + \xi_r^* + r_r^*.\gamma) - (\delta_r^* + \zeta_r^*)U_{OUA}.\mathcal{D}}{L}$$

$$= \frac{U_{OUA}\,(\delta_r^*.\mathcal{D} + \zeta_r^*.\mathcal{D} + r_r^*.V_{OUA}.\mathcal{D}) - (\delta_r^* + \zeta_r^*)U_{OUA}.\mathcal{D}}{L}$$

$$= \frac{U_{OUA}\,(\delta_r^*.\mathcal{D} + \zeta_r^*.\mathcal{D} + L.V_{OUA}.\mathcal{D}) - (\delta_r^* + \zeta_r^*)U_{OUA}.\mathcal{D}}{L}$$

$$= \frac{U_{OUA}.\mathcal{D}\,(\delta_r^* + \zeta_r^* + L.V_{OUA}) - (\delta_r^* + \zeta_r^*)U_{OUA}.\mathcal{D}}{L}$$

$$= \frac{U_{OUA}.\mathcal{D}[(\delta_r^* + \zeta_r^* + L.V_{OUA}) - (\delta_r^* + \zeta_r^*)]}{L}$$

$$= \frac{U_{OUA}.\mathcal{D}[(L.V_{OUA})]}{L} = V_{OUA}.U_{OUA}.\mathcal{D} = H,$$

if it is holds, for the answer of H, H_{OUA} will return $g = 1$, otherwise it returns $g = 0$.

Probability Analysis: In the above game, let $Pr[H_{OUA}\ Sucess]$ denote H_{OUA}'s probability of solving $H = (V_{OUA}.U_{OUA}.\mathcal{D})$, and $Pr[\xi_{OUA}Sucess]$ denote ξ_{OUA}'s probability of doing so. If the process of this game is stop at any stage, then for the answer of $H = (V_{OUA}.U_{OUA}.\mathcal{D})$ and H_{OUA} will select g from {0, 1} as his guess. So, the $Pr\,[H_{OUA}Sucess] = \dfrac{1}{2}$ and $Pr\,[\xi_{OUA}Sucess] = \dfrac{1}{2} + OUA_{\xi}$. We have the following computations: $Pr\,[H_{OUA}Sucess] = Pr\,[H_{OUA}Sucess|Stop]\,Pr\,[Stop] + Pr\left[H_{OUA}Sucess|\overleftarrow{Stop}\right] Pr\left[\overleftarrow{Stop}\right]$

$$= \frac{1}{2} Pr[Stop] + Pr[\xi_{OUA} Sucess] Pr\left[\overleftarrow{Stop}\right] = \frac{1}{2}\left(1 - Pr\left[\overleftarrow{Stop}\right]\right) + \left(\frac{1}{2} + OUA_{\xi}\right) Pr\left[\overleftarrow{Stop}\right]$$
$$= \frac{1}{2} - \left(OUA_{\xi} Pr\left[\overleftarrow{Stop}\right]\right).$$

On the other hand, if each of the independent events listed below occurs, H_{OUA} will not terminate.

E_{OUA1}: During *Private Key Queries and Partial Private Key Queries* $R_i = 0$, E_{OUA2}: During *Sender Queries* $R_s = 0$, and E_{OUA3}: During *Challenge Phase* $R_r^* = 0$.

So, the probability of $E_{OUA3} = 1/(Q_{nq} + 1)$ and $(E_{OUA1}, E_{OUA2}) = 1 - Q_{nq}/(Q_{nq} + 1)$, then we have $(E_{OUA1},\ E_{OUA2},\ E_{OUA3}) = (1/(Q_{nq} + 1)) = 1/(Q_{nq} + 1)^2$.

Finally, we can get: $Pr[H_{OUA} Sucess] >= \frac{1}{2} + OUA_{\xi}/(Q_{nq} + 1)^2$.

Theorem 2-Confidentiality: Suppose the insider attacker (ξ_{ISA}), with his advantages ISA_{ξ}, wants to break *IND-CAA-CCA-I* with help of helper called H_{OUA} and his task is to solve HEDH Problem with advantage as followed: $Pr[H_{OUA} Sucess] >= \frac{1}{2} + ISA_{\xi}/(Q_{nq} + 1)^2$, where Q_{nq} denotes the number of private key extract query, partial private key extract query, and sender query, respectively.

Proof: Suppose the random triple ($\mathcal{D}$, $V_{OUA}.\mathcal{D}$, $U_{OUA}.\mathcal{D}$) of HEDH Problem is given to H_{OUA} and his task to compute $H = V_{OUA}.U_{OUA}.\mathcal{D}$. The following sub steps represents the processing of *Theorem 2:*

Setup: Given an 80-bit hyper elliptic curve parameter ℓ for the security of the proposed scheme, the H_{OUA} set $\gamma = \lambda.\mathcal{D}$ and $\Phi = \{\gamma,\ \mathcal{D},\ Hash_a,\ Hash_b,\ Hash_c,\ HE,\ Field_n\}$. Then, H_{OUA} disclosed Φ to the ξ_{ISA}. Not that, here H_{OUA} has access to λ.

$Hash_a$ Query, $Hash_b$ Query, $Hash_c$ Query: ξ_{ISA} chooses the identity ID_i and sends out $Hash_a$, $Hash_b$, $Hash_c$ oracle queries to H_{OUA}. H_{OUA} will give respond as like in *Theorem 1*.

Public Key Queries: ξ_{ISA} chooses the identity ID_i and sends out *Public Key* oracle queries to H_{OUA}. H_{OUA} check $(\eta_i,\ \xi_i)$ in the list L_{pKq}, if it is existing, then it delivers $(\eta_i,\ \xi_i)$ to ξ_{OUA}, otherwise the following two conditions should consider:

- If $R_i = 0$, H_{OUA} compute $\eta_i = \delta_i.\mathcal{D}$, $\xi_i = \zeta_i.\mathcal{D}$ and $\chi_i = \zeta_i + Hash_a(ID_i,\ \eta_i,\ \xi_i).\lambda$ where $\delta_i,\ \zeta_i$ is the randomly chosen numbers from hyper elliptic curve Jacobian group. Then, H_{OUA} send $(\eta_i,\ \xi_i)$ to ξ_{ISA} and update the list L_{pKq} with $(\eta_i,\ \xi_i)$ and $(\delta_i,\ \chi_i)$.
- If $R_i = 1$, H_{OUA} compute $\eta_i = \delta_i.\mathcal{D}$ and $\xi_i = V_{OUA}.\mathcal{D}$, where δ_i is the randomly chosen numbers from hyper elliptic curve Jacobian group. Then, H_{OUA} send $(\eta_i,\ \xi_i)$ to ξ_{ISA} and update the list L_{pKq} with $(\eta_i,\ \xi_i)$ and $(\delta_i,\ \zeta_i)$.

Private Key Queries: ξ_{ISA} chooses the identity ID_i and sends out *Private Key* oracle queries to H_{OUA}. Then H_{OUA} check $(\chi_i,\ \xi_i)$ in the list L_{pRiKq}, if available, H_{OUA} will send $(\chi_i,\ \delta_i)$ to ξ_{ISA}. Otherwise, the following two conditions should consider:

1. If $R_i = 0$, H_{OUA} call *Public Key Queries*, get $(\chi_i,\ \delta_i)$ and send it to the ξ_{ISA}.
2. If $R_i = 0$, H_{OUA} will not further processed this game and stop the simulation.

Sender Query: As a response of this query, H_{OUA} examines of the value R_s by using the following computations:

- If $R_s = 0$, H_{OUA} pick (χ_i, δ_i) from L_{pRiKq}, if it is not available in L_{pRiKq}, then it call *Private Key Queries, obtain* $(\chi_i,\ \delta_i)$, perform the algorithmic steps as made by APDR, and generate $(\varphi,\ \omega)$. Finally, H_{OUA} will delivered $(\varphi,\ \omega)$ to ξ_{ISA}.
- If $R_s = 1$, H_{OUA} will stop simulation of this process.

Receiver Query: H_{OUA} examines R_r and R_s responds as follows when ξ_{OUA} queries for sender signature and encryption:

- If R_r and $R_s = 0$, H_{OUA} pick $(\chi_i,\ \delta_i)$ from L_{pRiKq}, if it is not available in L_{pRiKq}, then it call *Private Key Queries, obtain* (χ_i, δ_i), perform the alogorithmic steps as made by WBD during signature verifications and APDR decryption of ciphertext, and generate (M). Finally, H_{OUA} will delivered (M) to ξ_{ISA}.
- If $R_s = 1$, H_{OUA} pick the value K_i, compute $M = (Hash_c(K)\ \oplus\ C)$, and send M to ξ_{ISA}.
- If $R_r = 1$, H_{OUA} compute $\varphi\ (\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega$, where r_a and $\mathcal{G}$ is selected randomly.

Challenge: ξ_{ISA} send (M_{OUA1}, M_{OUA2}) *and* (ID_s^*, ID_r^*) to H_{OUA}, decides on R_r^*, and do the following steps:

- If $R_r^* = 0$, H_{OUA} stop the simulation.
- If $R_r^* = 1$, H_{OUA} compute $\omega^* = U_{OUA}.\mathcal{D}$, obtained K_i^* from L_{Hash_c}, select φ^* randomly, g^* from {0, 1}, compute $C^* = Hash_c(K_i^*) \oplus (M_{g^*})$, and send $(\omega^*,\ \varphi^*,\ C^*)$ to ξ_{ISA}.

Guess: After performing the same number of queries as performed above, ξ_{ISA} outputs with a guess value g' from {0, 1} of g^*. Then, H_{OUA} can perform the following two steps:

- In the event that any steps of the simulation are avoided, H_{OUA} chooses g from {0, 1} at random as its best guess for the solution of $(H = (V_{OUA}.U_{OUA}.\mathcal{D}))$.
- Otherwise, if $g' = g^*$ H_{OUA} return δ_r^* from L_{pKq}, further the processing at challenge step will not be aborted and we have $R_r^* = 1$ and $r_r^* = L$. So, H_{OUA} obtains $(\omega^*,\ \varphi^*,\ K_i^*)$ and check the equation: $H = K^* - (\delta_r^* + L.\lambda)\omega^*$, is hold. If yes, H_{OUA} return $g = 1$, otherwise it returns $g = 0$.

Note that, H_{OUA} has no knowledge about $\xi_i^* = V_{OUA}.\mathcal{D}$, and $\zeta_r^* = V_{OUA}$. In the challenge phase, $\omega^* = U_{OUA}.\mathcal{D}$; it means that $U_{OUA} = \upsilon$, which is not known to H_{OUA}. In a situation, if $(\omega^*,\ \varphi^*,\ C^*)$ valid then we can get:

$$K^* - (\delta_r^* + L.\lambda)\,\omega^* = U_{OUA}\,(\eta_r^* + \xi_r^* + r_r^*.\gamma) - (\delta_r^* + L.\lambda)\omega^*$$
$$= U_{OUA}\,(\eta_r^* + \xi_r^* + r_r^*.\gamma) - (\delta_r^* + L.\lambda)\,U_{OUA}.\mathcal{D}$$
$$= U_{OUA}\,(\delta_r^*.\mathcal{D} + \xi_r^* + r_r^*.\gamma) - (\delta_r^* + L.\lambda)\,U_{OUA}.\mathcal{D} = U_{OUA}\,(\delta_r^*.\mathcal{D} + \zeta_r^*.\mathcal{D} + r_r^*.\gamma) - (\delta_r^* + L.\lambda)\,U_{OUA}.\mathcal{D}$$
$$= U_{OUA}\,(\delta_r^*.\mathcal{D} + \zeta_r^*.\mathcal{D} + r_r^*.\lambda.\mathcal{D}) - (\delta_r^* + L.\lambda)\,U_{OUA}.\mathcal{D}$$
$$= U_{OUA}.\mathcal{D}\,(\delta_r^* + \zeta_r^* + r_r^*.\lambda) - (\delta_r^* + L.\lambda)\,U_{OUA}.\mathcal{D} = U_{OUA}.\mathcal{D}\,(\delta_r^* + \zeta_r^* + L.\lambda) - (\delta_r^* + L.\lambda)\,U_{OUA}.\mathcal{D}$$
$$= U_{OUA}.\mathcal{D}\,(\delta_r^* + \zeta_r^* + L.\lambda) - (\delta_r^* + L.\lambda) = U_{OUA}.\mathcal{D}\,(\delta_r^* + V_{OUA} + L.\lambda) - (\delta_r^* + L.\lambda)) = U_{OUA}.V_{OUA}.\mathcal{D} = H,$$

if it is holds, for the answer of H, H_{OUA} will return $g = 1$, otherwise it returns $g = 0$.

Probability Analysis: In the above game, let $Pr[H_{OUA}Sucess]$ denote H_{OUA}'s probability of solving $H = (V_{OUA}.U_{OUA}.\mathcal{D})$, and $Pr[\xi_{ISA}Sucess]$ denote ξ_{ISA}'s probability of doing so. If the process of this game is stop at any stage, then for the answer of $H = (V_{OUA}.U_{OUA}.\mathcal{D})$ and H_{OUA} will select g from {0, 1} as his guess. So, the $Pr\,[H_{OUA}Sucess] = \dfrac{1}{2}$ and $Pr\,[\xi_{ISA}Sucess] = \dfrac{1}{2} + ISA_{\xi}$. We have the following computations. $Pr\,[H_{OUA}Sucess] = Pr\,[H_{OUA}Sucess|Stop]\,Pr\,[Stop] + Pr\left[H_{OUA}Sucess|\overleftarrow{Stop}\right] Pr\left[\overleftarrow{Stop}\right]$

$$= \frac{1}{2} Pr[Stop] + Pr[\xi_{ISA} Sucess] Pr\left[\overleftarrow{Stop}\right]$$
$$= \frac{1}{2}\left(1 - Pr\left[\overleftarrow{Stop}\right]\right) + \left(\frac{1}{2} + ISA_{\xi}\right) Pr\left[\overleftarrow{Stop}\right] = \frac{1}{2} - \left(ISA_{\xi} Pr\left[\overleftarrow{Stop}\right]\right).$$

On the other hand, if each of the independent events listed below occurs, H_{OUA} will not terminate.

E_{ISA1}: During *Private Key Queries and Partial Private Key Queries* $R_i = 0$, E_{ISA2}: During *Sender Queries* $R_s = 0$, and E_{ISA3}: During *Challenge Phase* $R_r^* = 0$.

So, the probability of $E_{ISA3} = 1/(Q_{nq} + 1)$ and $(E_{ISA1}, E_{ISA2}) = 1 - Q_{nq}/(Q_{nq} + 1)$, then we have $(E_{ISA1},\ E_{ISA2},\ E_{ISA3}) = (1/(Q_{nq} + 1))\ (\ 1 - Q_{nq}/(Q_{nq} + 1)) = 1/(Q_{nq} + 1)^2$.

Finally, we can get:

$$Pr[H_{OUA} Sucess] >= \frac{1}{2} + ISA_{\xi}/(Q_{nq} + 1)^2.$$

Theorem 3-Unforgeability: Suppose the outsider attacker (ξ_{OUA}), with his advantages OUA_{ξ}, wants to break the security regarding existentially unforgeable under the adaptive chosen- message attacks (*EF-CAA-CMA-I*) with help of helper called H_{OUA} and his task is to solve HEDLP Problem with advantage as followed: $Pr[H_{OUA} Sucess] >= \frac{1}{2} + OUA_{\xi}/(Q_{nq} + 1)^2$, where Q_{nq} denotes the number of private key extract query, partial private key extract query, and sender query, respectively.

Proof: Suppose the random triple ($\mathcal{D}$, $V_{OUA}.\mathcal{D}$) of HEDLP Problem is given to H_{OUA} and his task to compute $H = V_{OUA}.\mathcal{D}$. The following sub steps represents the processing of *Theorem 3*.

Setup: Given an 80 bit hyper elliptic curve parameter l for the security of the proposed scheme, the H_{OUA} set $\gamma = V_{OUA}.\mathcal{D}$ and $\Phi = \{\gamma,\ \mathcal{D},\ Hash_a,\ Hash_b,\ Hash_c,\ HE,\ Field_n\}$. Then, H_{OUA} disclosed Φ to the ξ_{OUA}). Not that, here H_{OUA} has no access to λ.

Queries: The queries which can ask by ξ_{OUA} is same as asked in *Theorem 1*.

Output: After performing all the queries that are done in *Theorem 1*, ξ_{OUA} *generate* $(\omega^*,\ \varphi^*)$ *from* APDR* to WBD* and C^* from WBD* to APDR* on M^*. So, according to forking lemma, ξ_{OUA} will generate two valid signature that are $(C^*,\ \varphi^*,\ \omega^*)$ and $(C^{**},\ \varphi^{**},\ \omega^{**})$, so, we have the following outputs: $\varphi^*(\mathcal{G} + \delta_s{}^* + \chi_s{}^*) = \delta_s{}^* + \upsilon$ and $\varphi^{**}(\mathcal{G}^{/} + \delta_s{}^* + \chi_s{}^*) = \delta_s{}^* + \upsilon$. So, we have:

$\varphi^*(\mathcal{G} + \delta_s{}^* + \chi_s{}^*) = \varphi^{**}(\mathcal{G}^{/} + \delta_s{}^* + \chi_s{}^*)$, as $R_s{}^* = 1$ and $r_s{}^* = L$, then $\chi_r{}^* = \zeta_r{}^* + r_s{}^*.\lambda$, $\gamma = V_{OUA}.\mathcal{D}$, and $\lambda = V_{OUA}$, so λ is not accessible for H_{OUA}. Further, $\chi_r{}^* = \zeta_r{}^* + L.V_{OUA}$, so we have $\varphi^*(\mathcal{G} + \delta_s{}^* + \zeta_r{}^* + L.V_{OUA}) = \varphi^{**}(\mathcal{G}^{/} + \delta_s{}^* + \zeta_r{}^* + L.V_{OUA})$.

So, in the above equations, V_{OUA} is unknown to H_{OUA} which is solution for HEDLP Problem.

Probability Analysis: In the above game, let $Pr[H_{OUA} Sucess]$ denote H_{OUA}'s probability of solving $H = (V_{OUA}.\mathcal{D})$, and $Pr[\xi_{OUA} Sucess]$ denote ξ_{OUA}'s probability of doing so. So, the $Pr[\xi_{OUA} Sucess] \geq OUA_{\xi}$. We have the following computations. $Pr[H_{OUA} Sucess] = Pr\left[\overleftarrow{Stop} \cap H_{OUA} Sucess\right]$
$= Pr\left[\overleftarrow{Stop}\right].Pr[H_{OUA} Sucess] = Pr\left[\overleftarrow{Stop}\right].OUA_{\xi}.$

On the other hand, if each of the independent events listed below occurs, H_{OUA} will not terminate.

E_{OUA1}: During *Private Key Queries and Partial Private Key Queries* $R_i = 0$, E_{OUA2}: During *Sender Queries* $R_s = 0$, and E_{OUA3}: During *Challenge Phase* $R_r^* = 0$.

So, the probability of $E_{OUA3} = 1/(Q_{nq}+1)$ and $(E_{OUA1}, E_{OUA2}) = 1 - Q_{nq}/(Q_{nq}+1)$, then we have $(E_{OUA1}, E_{OUA2}, E_{OUA3}) = (1/(Q_{nq}+1))\ (1 - Q_{nq}/(Q_{nq}+1)) = 1/(Q_{nq}+1)^2$.

Finally, we can get:

$$Pr[H_{OUA}Sucess] >= \frac{1}{2} + OUA_{\xi}/(Q_{nq}+1)^2.$$

Theorem 4-Unforgeability: Suppose the insider attacker (ξ_{ISA}), with his advantages OUA_{ξ}, wants to break the security regarding existentially unforgeable under the adaptive chosen- message attacks (*EF-CAA-CMA-II*) with help of helper called H_{OUA} and his task is to solve HEDLP Problem with advantage as followed: $Pr[H_{OUA}Sucess] >= \frac{1}{2} + ISA_{\xi}/(Q_{nq}+1)^2$, where Q_{nq} denotes the number of private key extract query, partial private key extract query, and sender query, respectively.

Proof: Suppose the random triple ($\mathcal{D}$, $V_{OUA}.\mathcal{D}$) of HEDLP Problem is given to H_{OUA} and his task to compute $H = V_{OUA}.\mathcal{D}$. The following sub steps represents the processing of *Theorem 4*:

Setup: Given an 80-bit hyper elliptic curve parameter ℓ for the security of the proposed scheme, the H_{OUA} set $\gamma = V_{OUA}.\mathcal{D}$ and $\Phi = \{\gamma,\ \mathcal{D},\ Hash_a,\ Hash_b,\ Hash_c,\ HE,\ Field_n\}$. Then, H_{OUA} disclosed Φ to the ξ_{ISA}. Not that, here H_{OUA} has access to λ.

Queries: The queries which can ask by ξ_{ISA} is same as asked in *Theorem 2*.

Output: After performing all the queries that are done in *Theorem 2*, ξ_{ISA}*generate* (ω^*, φ^*) *from* APDR* to WBD* and C^* from WBD* to APDR* on M^*. So, according to forking lemma, ξ_{OUA} will generate two valid signature that are $(C^*, \varphi^*, \omega^*)$ and $(C^{**}, \varphi^{**}, \omega^{**})$, so, we have the following outputs: $\varphi^*(\mathcal{G} + \delta_s{}^* + \chi_s{}^*) = \delta_s{}^* + \upsilon$ and $\varphi^{**}(\mathcal{G}^{/} + \delta_s{}^* + \chi_s{}^*) = \delta_s{}^* + \upsilon$. So, we have:

$\varphi^*(\mathcal{G} + \delta_s{}^* + \chi_s{}^*) = \varphi^{**}(\mathcal{G}^{/} + \delta_s{}^* + \chi_s{}^*)$, as $R_s{}^* = 1$ and $r_s{}^* = L$, then $\chi_r{}^* = \zeta_r{}^* + r_s{}^*.\lambda$, $\gamma = V_{OUA}.\mathcal{D}$, and $\lambda = V_{OUA}$, so λ is not accessible for H_{OUA}. Further, $\chi_r{}^* = \zeta_r{}^* + L.V_{OUA}$, so we have $\varphi^*(\mathcal{G} + \delta_s{}^* + \zeta_r{}^* + L.V_{OUA}) = \varphi^{**}(\mathcal{G}^{/} + \delta_s{}^* + \zeta_r{}^* + L.V_{OUA})$.

So, in the above equations, V_{OUA} is unknown to H_{OUA} which is solution for HEDLP Problem.

Probability Analysis: In the above game, let $Pr[H_{OUA}Sucess]$ denote H_{OUA}'s probability of solving $H = (V_{OUA}.\mathcal{D})$, and $Pr[\xi_{ISA}Sucess]$ denote ξ_{OUA}'s probability of doing so. So, the $Pr[\xi_{ISA}Sucess] \geq ISA_{\xi}$. We have the following computations. $Pr[H_{OUA}Sucess] = Pr\left[\overleftarrow{Stop} \cap H_{OUA}Sucess\right]$

$= Pr\left[\overleftarrow{Stop}\right].Pr[H_{OUA}Sucess] = Pr\left[\overleftarrow{Stop}\right].ISA_{\xi}$.

On the other hand, if each of the independent events listed below occurs, H_{OUA} will not terminate.

E_{ISA1}: During *Private Key Queries and Partial Private Key Queries* $R_i = 0$, E_{ISA2}: During *Sender Queries* $R_s = 0$, and E_{ISA3}: During *Challenge Phase* $R_r{}^* = 0$.

So, the probability of $E_{ISA3} = 1/(Q_{nq}+1)$ and $(E_{ISA1}, E_{ISA2}) = 1 - Q_{nq}/(Q_{nq}+1)$, then we have $(E_{ISA1}, E_{ISA2}, E_{ISA3}) = (1/(Q_{nq}+1))\ 1 - Q_{nq}/(Q_{nq}+1) = 1/(Q_{nq}+1)^2$.

Finally, we can get:

$$Pr[H_{OUA}Sucess] >= \frac{1}{2} + ISA_{\xi}/(Q_{nq}+1)^2.$$

5.2 Informal Security Analysis

The proposed scheme is based on hash function and hyper elliptic curve discrete logarithm problem. The two main properties of the hash function are irreversibility and collision resistance. Suppose $\mathcal{D}_1 = \chi.\mathcal{D}$, finding '$\chi$ is said to be a hyper elliptic curve discrete logarithm problem.

Mutual Authentication: The APDR generates the digital signature for mutual authentication with WBD as $\varphi = \delta_s + \upsilon/\mathcal{G} + \delta_s + \chi_s$ and sends it together with ω to the WBD. When the WBD gets φ and ω, it checks the equation $\varphi\,(\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega$

For authentication, and if the equality of $\varphi\,(\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega$ holds, which means that mutual authentication between APDR and the WBD is successfully done. The equality test of $\varphi\,(\eta_s + \xi_s + r_a.\gamma + \mathcal{G}.\mathcal{D}) = \eta_s + \omega$ can be seen from Section 3.3.

Signature Unforgeability: In the proposed scheme, the signature generated by the application provider is computed using the formula, $\varphi = \delta_s + \upsilon/\mathcal{G} + \delta_s + \chi_s$. If an attacker attempts to generate a forge one, then he/she must extract δ_s, and υ from $\eta_s = \delta_s.\mathcal{D}$ and $\omega = \upsilon.\mathcal{D}$ that are normally equals to process two-time hyper elliptic curve discrete logarithm problem, which is difficult.

Sender Anonymity: To register APDR, the user selects δ_s, computes $\eta_s = \delta_s.\mathcal{D}$, compute $SK = \delta_s.\gamma$, encrypt the identity (ID_s) as $E_{Ids} = E_{SK}\,(ID_s)$, and send $(E_{Ids},\ \eta_s)$ to NMGR via un-secure channel. Upon receiving $(E_{Ids},\ \eta_s)$, NMGR can compute the secret key as $SK = \eta_s.\lambda$ and decrypt the identity as $ID_s = D_{SK}(E_{Ids})$. From the above discussion it is clear that the attacker has no access to the identity of APDR which play the role of sender. Further, the application provider can compute $\omega = \upsilon.\mathcal{D}$, where and $\mathcal{D}$ denotes the private number and a divisor on hyper elliptic curve. Compute $r_b = Hash_a(ID_r,\ \eta_r,\ \xi_r)$ and $K = \upsilon\,(\eta_r + \xi_r + r_b.\gamma)$. Calculate $\mathcal{G} = Hash_b\ (\omega,\ ID_r,\ ID_s,\ \eta_s,\ \xi_s)$ and $\varphi = \delta_s + \upsilon/\mathcal{G} + \delta_s + \chi_s$, then send $(\varphi,\ \omega)$ to WBD. It is evident from these calculations that the application provider is not providing the identity of any user via an open channel.

Receiver Anonymity: To register WBD, the user selects δ_r and computes $\eta_r = \delta_r.\mathcal{D}$ for the identity (ID_r), compute $RK = \delta_r.\gamma$, encrypt the identity (ID_r) as $E_{Idr} = E_{RK}(ID_r)$ and send $(E_{Idr},\ \eta_r)$ to NMGR via un-secure channel. Upon receiving $(E_{Idr},\ \eta_r)$, NMGR can compute the secret key as $SK = \eta_r.\lambda$ and decrypt the identity as $ID_r = D_{SK}(E_{Idr})$. From the above discussion it is clear that the attacker has no access to the identity of WBD which play the role of receiver. As a result, the proposed scheme is immune to anonymous communication. From the above discussion we can observed that in our proposed scheme ensures anonymous communication, because the attacker cannot get any access to the identity of any participated user.

Integrity: The WBD set the secret key as $K = (\delta_r + \chi_r)\omega$ and done the encryption of a message as $C = Hash_c(K) \oplus (M)$, where $\oplus$ denotes the encryption function which encrypt the message (M) using the secrete key K and finally send C to application provider. The attacker cannot modify M, because it must need the secret K, which is not accessible for attacker according to the proof of *Theorems 1* and *2*. Hence, we can conclude from the discussion our proposed scheme satisfies the integrity property.

6 Comparative Analysis

This section compares the proposed scheme to other relevant schemes in terms of computation cost, communication cost, memory overhead and security functionalities.

6.1 Computational Cost

In this section, we compare the proposed approach with the methods presented by Liao et al. [25–28]. The comparison is conducted based on the computational costs. Table 2 provides a summary of the important conclusions obtained from the comparison of computing costs. The computational time for Elliptic Curve Point Multiplication (EMUL) is 0.97 milliseconds, while bilinear pairing (P) takes 14.90 ms. Pairing-based point multiplications (BPM) need 4.31 ms, while exponentials on bilinear pairing (EXP) take 1.25 ms [29–32]. To evaluate the computational cost of the suggested approach, we used the Multi-precision Integer and Rational Arithmetic C Library (MIRACL) [33]. The library does a substantial quantity of tests, potentially reaching up to 1000, on fundamental cryptographic procedures. The simulations are executed on a computing device equipped with an Intel Core i7-4510U CPU operating at a frequency of 2.0 GHz, 8 GB of random-access memory (RAM), and the Windows 7 operating system [34]. The HEMUL is expected to have a processing time of 0.48 milliseconds due to its key size of 80 bits. The analysis of Tables 3 and 4, and Fig. 6 demonstrates that the suggested design exhibits much higher cost-effectiveness in terms of computing.

Table 3: Comparison based on computation cost

Schemes	Sender	Receiver	Total
Liao et al. [25]	1 EXP + 5 BPM	2 EXP + 4P + 1BPM	3 EXP + 4P + 6BPM
He et al. [26]	5 BPM	2P + 4BPM	2P + 9BPM
Kasyoka et al. [27]	4 EMUL	2 EMUL	6 EMUL
Liu et al. [28]	6 EMUL	8 EMUL	14 EMUL
Proposed	3 HEMUL	3 HEMUL	6 HEMUL

Table 4: Comparison based on computation cost (in ms)

Schemes	Sender	Receiver	Total (in ms)
Liao et al. [25]	$1 * 1.25 + 5 * 4.31 = 22.8$	$2 * 1.25 + 4 * 14.90 + 1 * 4.31 = 66.41$	$3 * 1.25 + 4 * 14.90 + 6 * 4.31 = 89.21$
He et al. [26]	$5 * 4.31 = 21.55$	$2 * 14.90 + 4 * 4.31 = 47.04$	$2 * 14.90 + 9 * 4.31 = 68.59$
Kasyoka et al. [27]	$4 * 0.97 = 3.88$	$2 * 0.97 = 1.94$	$6 * 0.97 = 5.82$
Liu et al. [28]	$6 * 0.97 = 5.82$	$8 * 0.97 = 7.76$	$14 * 0.97 = 13.58$
Proposed	$3 * 0.48 = 1.44$	$3 * 0.48 = 1.44$	$6 * 0.48 = 2.88$

6.2 Communication Cost

In this part, the proposed scheme is compared to existing schemes proposed by Liao et al. [25–28] in terms of communication cost. Tables 5 and 6 show the primary conclusions drawn from the comparison. According to the data shown in Fig. 7, choosing the proposed scheme results in a significant reduction in communication costs.

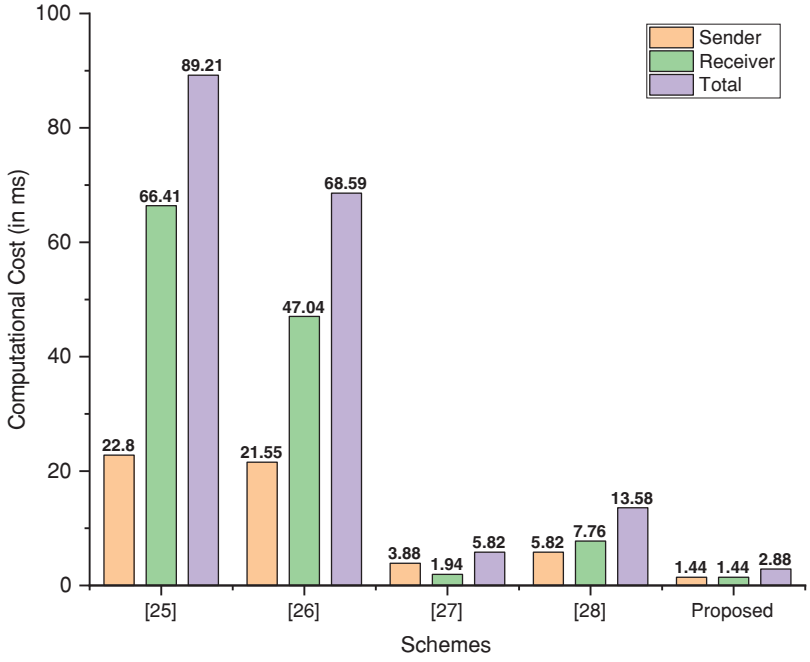

Figure 6: Comparative analysis based on communication cost (in bits) (Liao et al. [25], He et al. [26], Kasyoka et al. [27], Liu et al. [28])

Table 5: Comparison based on communication cost

Schemes	Signature size
Liao et al. [25]	(m) + 6(G)
He et al. [26]	(m) + 3(G) + (H)
Kasyoka et al. [27]	(m) + 2(q) + 2(H)
Liu et al. [28]	(m) + 6(q)
Proposed	(m) + 2(n) + (H)

Note: (m) = message, (H) = hash function, (q) = elliptic curve bits for single parameter, (G) = bilinear pairing bits for single parameter, (n) = hyper elliptic curve bits for single parameter (m) = 1024 bits, (H) = 256 bits, (q) = 160 bits, (G) = 1024 bits, (n) = 80 bits.

Table 6: Comparison based on communication cost (in bits)

Schemes	Signature size	Total (in bits)
Liao et al. [25]	(1024) + 6 ∗ (1024)	7168
He et al. [26]	(1024) + 3 ∗ (1024) + (256)	4352
Kasyoka et al. [27]	(1024) + 2 ∗ (160) + 2 ∗ (256)	1856
Liu et al. [28]	(1024) + 6 ∗ (160)	1984
Proposed	(1024) + 2 ∗ (80) + (256)	1440

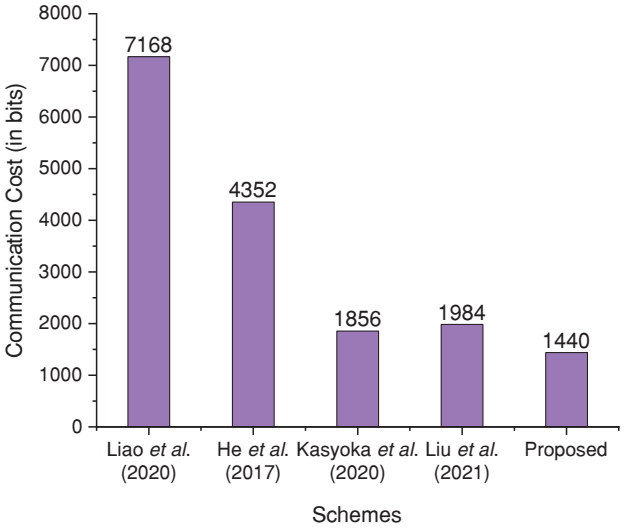

Figure 7: Comparative analysis based on communication cost (in bits) [25–28]

7 Conclusions and Future Works

IoHT systems are used to exchange remote data for a variety of physical activities, including patient monitoring, treatment development, observation and counseling. In IoHT, communication, computing, and interoperability are provided through multiple sensors, actuators, and controllers, resulting in a seamless connection and efficient resource management. Traditional cryptographic techniques, on the other hand, are not practicable for the vast majority of IoHT deployments due to the restrictions of low-power embedded devices. As a result, we proposed the CAA scheme, a security method based on the HECC in this article. The HECC method is effective with small key sizes and can be used in IoHT systems. The security analysis demonstrates the proposed scheme's efficacy in preventing several cyber-attacks. Second, comparing the proposed scheme with existing schemes is carried out in terms of computation and communication costs. The proposed scheme takes 2.88 ms to compute, compared to 89.21, 68.59, 5.82, and 13.58 ms for Liao et al. [25–28], respectively. Similarly, the proposed scheme's communication cost is 1440 bits, compared to 7168, 4352, 1856, and 1984 bits for Liao et al. [25–28], respectively. This comparison reveals that the proposed scheme is more efficient than the existing methods in computation and communication costs. In the future, we are intended to consider Genus 3 hyperelliptic curve, which will further improve the proposed scheme efficiency.

Acknowledgement: Nisreen Innab would like to express sincere gratitude to AlMaarefa University, Riyadh, Saudi Arabia, for supporting this research.

Funding Statement: The author received no specific funding for this study.

Availability of Data and Materials: The data supporting the conclusions of this article are included within the article.

Conflicts of Interest: The author declares that he has no conflicts of interest to report regarding the present study.

References

[1] J. J. P. C. Rodrigues *et al.*, "Enabling technologies for the internet of health things," *IEEE Access*, vol. 6, no. 1, pp. 13129–13141, 2018. doi: 10.1109/ACCESS.2017.2789329.

[2] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K. S. Kwak, "The internet of things for health care: A comprehensive survey," *IEEE Access*, vol. 3, no. 1, pp. 678–708, 2015. doi: 10.1109/ACCESS.2015.2437951.

[3] L. Catarinucci *et al.*, "An IoT-aware architecture for smart healthcare systems," *IEEE Internet Things J.*, vol. 2, no. 6, pp. 515–526, Dec. 2015. doi: 10.1109/JIOT.2015.2417684.

[4] Y. Yin, Y. Zeng, X. Chen, and Y. Fan, "The internet of things in healthcare: An overview," *J. Ind. Inf. Integr.*, vol. 1, no. 1, pp. 3–13, Mar. 2016. doi: 10.1016/j.jii.2016.03.004.

[5] C. Dong, Y. Sun, M. Shafiq, N. Hu, Y. Liu and Z. Tian, "Optimizing mobility-aware task offloading in smart healthcare for internet of medical things through multi-agent reinforcement learning," *IEEE Internet Things J.*, vol. 11, no. 8, pp. 13677–13691, 15 Apr. 2024. doi: 10.1109/JIOT.2023.3338718.

[6] B. Farahani, F. Firouzi, V. Chang, M. Badaroglu, N. Constant and K. Ankodiya, "Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare," *Future Gener. Comput. Syst.*, vol. 78, pp. 659–676, Jan. 2018. doi: 10.1016/j.future.2017.04.036.

[7] F. Firouzi *et al.*, "Internet-of-things and big data for smarter healthcare: From device to architecture, applications and analytics," *Future Gener. Comput. Syst.*, vol. 78, no. 1, pp. 583–586, 2018. doi: 10.1016/j.future.2017.09.016.

[8] X. Lin, R. Lu, X. Shen, Y. Nemoto, and N. Kato, "Sage: A strong privacy preserving scheme against global eavesdropping for eHealth systems," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 4, pp. 365–378, 2009. doi: 10.1109/JSAC.2009.090502.

[9] F. S. Sadek, K. Belkadi, A. Abouaissa, and P. Lorenz, "Identifying misbehaving greedy nodes in IoT networks," *Sensors*, vol. 21, no. 15, pp. 5127, 2021. doi: 10.3390/s21155127.

[10] C. M. Chen, Z. Li, S. A. Chaudhry, and L. Li, "Attacks and solutions for a two-factor authentication protocol for wireless body area networks," *Secur. Commun. Netw.*, vol. 2021, no. 2021, pp. 1–12. doi: 10.1155/2021/3116593.

[11] S. A. Chaudhry, A. Irshad, K. Yahya, N. Kumar, M. Alazab and Y. B. Zekeria, "Rotating behind privacy: An improved lightweight authentication scheme for cloud-based IoT environment," *ACM Trans. Internet Technol.*, vol. 21, no. 3, pp. 1–19, 2021.

[12] K. Kim, J. Ryu, Y. Lee, and D. Won, "An improved lightweight user authentication scheme for the internet of medical things," *Sensors*, vol. 23, no. 3, pp. 1122, 2023. doi: 10.3390/s23031122.

[13] S. Hussain, I. Ullah, H. Khattak, M. A. Khan, C. -M. Chen and S. Kumari, "A lightweight and provable secure identity-based generalized proxy signcryption (IBGPS) scheme for industrial internet of things (IIoT)," *J. Inf. Secur. Appl.*, vol. 58, no. 1, pp. 102625, 2021. doi: 10.1016/j.jisa.2020.102625.

[14] C. M. Chen, Z. Chen, S. Kumari, and M. C. Lin, "LAP-IoHT: A lightweight authentication protocol for the Internet of Health Things," *Sensors*, vol. 22, no. 14, pp. 5401, Jul. 2022. doi: 10.3390/s22145401.

[15] P. Kumar, S. Kumari, V. Sharma, X. Li, A. K. Sangaiah and S. H. Islam, "Secure CLS and CL-AS schemes designed for vanets," *J. Supercomput.*, vol. 75, no. 6, pp. 3076–3098, 2018. doi: 10.1007/s11227-018-2312-y.

[16] I. Ullah, N. U. Amin, M. A. Khan, H. Khattak, and S. Kumari, "An efficient and provable secure certificate-based combined signature, encryption and signcryption scheme for internet of things (IoT) in mobile health (m-health) system," *J. Med. Syst.*, vol. 45, no. 1, pp. 1–20, 2020.

[17] H. M. Chen, J. W. Lo, and C. K. Yeh, "An efficient and secure dynamic id-based authentication scheme for telecare medical information systems," *J. Med. Syst.*, vol. 36, no. 6, pp. 3907–3915, 2012. doi: 10.1007/s10916-012-9862-y.

[18] Q. Jiang, J. Ma, Z. Ma, and G. Li, "A privacy enhanced authentication scheme for telecare medical information systems," *J. Med. Syst.*, vol. 37, no. 1, pp. 9897–9898, 2013. doi: 10.1007/s10916-012-9897-0.

[19] S. Kumari, M. K. Khan, and R. Kumar, "Cryptanalysis and improvement of 'a privacy enhanced scheme for telecare medical information systems'," *J. Med. Syst.*, vol. 37, no. 4, pp. 1–11, 2013. doi: 10.1007/s10916-013-9952-5.

[20] C. L. Chen, T. T. Yang, M. L. Chiang, and T. F. Shih, "A privacy authentication scheme based on cloud for medical environment," *J. Med. Syst.*, vol. 38, no. 11, pp. 143–216, 2014. doi: 10.1007/s10916-014-0143-9.

[21] S. Y. Chiou, Z. Ying, and J. Liu, "Improvement of a privacy authentication scheme based on cloud for medical environment," *J. Med. Syst.*, vol. 40, no. 4, pp. 101, 2016. doi: 10.1007/s10916-016-0453-1.

[22] P. Mohit, R. Amin, A. Karati, G. P. Biswas, and M. K. Khan, "A standard mutual authentication protocol for cloud computing based health care system," *J. Med. Syst.*, vol. 41, no. 4, pp. 35–50, 2017. doi: 10.1007/s10916-017-0699-2.

[23] C. T. Li, D. H. Shih, and C. C. Wang, "Cloud-assisted mutual authentication and privacy preservation protocol for telecare medical information systems," *Comput. Methods Programs Biomed.*, vol. 157, no. 1, pp. 191–203, 2018. doi: 10.1016/j.cmpb.2018.02.002.

[24] M. E. S. Saeed, Q. Liu, G. Tian, B. Gao, and F. Li, "Remote authentication schemes for wireless body area networks based on the internet of things," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4926–4944, 2018. doi: 10.1109/JIOT.2018.2876133.

[25] Y. Liao, Y. Liu, Y. Liang, Y. Wu, and X. Nie, "Revisit of certificateless signature scheme used to remote authentication schemes for wireless body area networks," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2160–2168, 2020. doi: 10.1109/JIOT.2019.2959602.

[26] D. He, S. Zeadally, N. Kumar, and J. H. Lee, "Anonymous authentication for wireless body area networks with provable security," *IEEE Syst. J.*, vol. 11, no. 4, pp. 2590–2601, 2017. doi: 10.1109/JSYST.2016.2544805.

[27] P. Kasyoka, M. Kimwele, and S. M. Angolo, "Certificateless pairing-free authentication scheme for wireless body area network in healthcare management system," *J. Med. Eng. Technol.*, vol. 44, no. 1, pp. 12–19, 2020. doi: 10.1080/03091902.2019.1707890.

[28] Y. Liu, Y. Wang, and Z. Peng, "A novel and efficient anonymous authentication for wbans," *Internet Technol. Lett.*, vol. 1, no. 1, pp. 12–19, 2021.

[29] M. A. Khan *et al.*, "An online-offline certificateless signature scheme for internet of health things," *J. Healthc. Eng.*, vol. 2020, pp. 1–10, 2020. doi: 10.1155/2020/6654063.

[30] I. Ullah, S. Zeadally, N. U. Amin, M. Asghar Khan, and H. Khattak, "Lightweight and provable secure cross-domain access control scheme for internet of things (IoT) based wireless body area networks (WBAN)," *Microprocess. Microsyst.*, vol. 81, no. 1, pp. 103477, 2021. doi: 10.1016/j.micpro.2020.103477.

[31] M. A. Khan *et al.*, "A provable and privacy-preserving authentication scheme for uav-enabled intelligent transportation systems," *IEEE Trans. Ind. Inform.*, vol. 18, no. 5, pp. 3416–3425, 2022. doi: 10.1109/TII.2021.3101651.

[32] M. A. Khan, B. A. Alzahrani, A. Barnawi, A. Al-Barakati, A. Irshad and S. A. Chaudhry, "A resource friendly authentication scheme for space-air–ground-sea integrated maritime communication network," *Ocean Eng.*, vol. 250, no. 110894, Apr. 2022. doi: 10.1016/j.oceaneng.2022.110894.

[33] P. Rastegari, M. Khalili, and A. Sakhaei, "Security analysis and improvement of an access control protocol for wbans," *Int. J. Netw. Security*, vol. 25, no. 2, pp. 285–296, 2023.

[34] S. Yu, A. K. Das, Y. Park, and P. Lorenz, "SLAP-IoD: Secure and lightweight authentication protocol using physical unclonable functions for internet of drones in smart city environments," *IEEE Trans. Vehicular Technol.*, vol. 71, no. 10, pp. 10374–10388, 2022. doi: 10.1109/TVT.2022.3188769.