IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 May 2024, accepted 31 May 2024, date of publication 4 June 2024, date of current version 20 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3409359

RESEARCH ARTICLE

A Certificate-Based Ring Signcryption Scheme for Securing UAV-Enabled Private Edge Computing Systems

MUHAMMAD ASGHAR KHAN[1], (Senior Member, IEEE), INSAF ULLAH[2], NEERAJ KUMAR[3], (Senior Member, IEEE), FATEMEH AFGHAH[4], (Senior Member, IEEE), GORDANA BARB[5], FAZAL NOOR[6], AND SAAD ALQAHTANY[6]

[1]Department of Electrical Engineering, Prince Mohammad bin Fahd University, Al Khobar 31952, Saudi Arabia
[2]Institute for Analytics and Data Science, University of Essex, CO4 3SQ Colchester, U.K.
[3]Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology (Deemed to be University), Patiala 147004, India
[4]Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634, USA
[5]Department of Communications, Politehnica University of Timişoara, 300006 Timişoara, Romania
[6]College of Computer and Information Systems, Islamic University of Madinah, Madinah 42351, Saudi Arabia

Corresponding author: Gordana Barb (gordana.barb@upt.ro)

ABSTRACT The evolving paradigm of private edge computing seamlessly incorporates the more extensive functionalities of cloud computing with localized processing. This paradigm eliminates the requirement for unmanned aerial vehicles (UAVs) to transmit large volumes of data to a centralized cloud, thereby reducing response times. UAVs' dynamic nature and dependency on unsecured and publicly accessible wireless channels make secure communication between a private edge cloud and a UAV difficult. Therefore, private edge computing-enabled UAV networks require additional security measures to protect the network and users' data. This research article introduces a certificate-based ring signcryption scheme that mitigates security concerns by utilizing the concept of hyperelliptic curve cryptography (HECC). By combining digital signature and encryption into a single operation, the proposed method takes advantage of the most advantageous characteristic of HECC (the ability to use a short key, such as 80 bits) while maintaining the same level of security as RSA and ECC. The security properties of the proposed scheme are validated by implementing a formal security evaluation method known as the random oracle model (ROM), in addition to informal security analysis. Furthermore, the computation and communication costs of the proposed scheme are evaluated and compared to those of relevant existing schemes. The performance and security analysis demonstrate that the proposed scheme enhances efficiency and security.

INDEX TERMS Cloud computing security, private edge computing, unmanned aerial vehicles, cryptography, hyperelliptic curve cryptography, ring signcryption, computational efficiency.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) are expected to find widespread adaptation in diverse civilian, commercial, and military applications in the coming years due to their manoeuvrability in three-dimensional (3D) space, simplicity in control mechanisms and high precision in positioning [1], [2], [3]. In addition, the ability to perform beyond line-of-sight (BLOS) operations is an extra benefit, which adds to their rising popularity. To meet the increasing demand for UAVrelated services in diverse applications, wireless networks beyond 5G (B5G) have additionally encouraged the development of several supporting technologies [4], [5], [6], [7]. These technologies include multi-access edge computing (MEC), software-defined networking (SDN), network function virtualization (NFV), and network slicing (NS) [8]. The UAVs will benefit from these technologies in a way that NFV will facilitate scalability and the

The associate editor coordinating the review of this manuscript and approving it for publication was Cong Pu.

rapid deployment of new services by decoupling network functions from the underlying hardware. On the other hand, SDN enables the automated and programmed configuration and monitoring of UAV networks, thereby contributing to the holistic and global management of the entire infrastructure. NS enhances service customization and resource segregation by building several logical UAV networks on the same physical infrastructure. Through MEC, resource-restricted UAVs can access cloud computing services to execute various computing, storage and processing-related functions [9], [10].

A new concept has emerged under MEC, which proposes using private edge cloud systems as a potential alternative to publicly accessible edge computing solutions [11]. The primary goal of these systems is to more comprehensively address latency, security, and privacy issues while optimizing bandwidth utilization and improving the performance of resource-constrained devices. A private edge cloud system will facilitate UAVs to store, retrieve, and compute information locally while performing operations, as shown in Fig.1. This facility will reduce the full-time dependence on centralized cloud servers for frequent data transmission, thus opening up the possibility of better response time in various applications. However, this raises security concerns since UAVs typically rely on wireless communication channels to transfer data to and from private edge cloud systems. The importance of secure key management and effective encryption and digital signature mechanisms is thus evident from the fact that these security methods address threats that compromise confidentiality and data integrity. Moreover, authentication and authorization concerns may also arise, emphasizing the importance of implementing identity verification and strict access control mechanisms. Ensuring device security includes implementing adequate security measures, adopting hardening procedures, and enabling regular updates. To proactively address potential security threats, it is crucial to prevent unauthorized access, eavesdropping and potential intrusion [12]. Similarly, high scalability, device diversity and mobility must be considered when designing security schemes for private edge computing systems operated by UAVs [13]. Additionally, in many UAV-assisted communication scenarios, the trajectory plays a crucial role, which is pivotal in ensuring the security and efficiency of UAV operations [14]. Furthermore, in mission planning for UAVs, greater emphasis should be placed on addressing the task assignment problem, as this directly impacts security considerations and operational effectiveness [15].

The motivation of this article is summed up as follows. First, UAVs are configured as smart devices with limited computing resources. Second, UAVs collect real-time data and transmit it to private edge computing over insecure channels (public channels). Consequently, there is the possibility of potential threats. Confidentiality and authentication are two essential features of every security protocol [16]. Encryption and digital signatures offer answers for confidentiality and authenticity, respectively [17]. For devices with low resources, such as UAVs, when both features are required simultaneously and in a single logical step, signcryption is preferable. As an extra benefit, signcryption can be employed in ring settings, a method referred to as ring signcryption that can offer favorable security properties such as anonymity, spontaneity, flexibility, and equal membership.

In the literature, the security and efficiency of ring signcryption schemes are typically tested with computationally challenging problems, such as RivestShamir-Adleman (RSA), bilinear pairing (BP), and elliptic curve cryptography (ECC) [18], [19]. However, the high computational and communication costs, complexity, and large-scale mathematical operations associated with the methods above make their implementation on UAVs impractical under normal conditions. As a result, we introduce a novel architecture by integrating a private edge computing capability to a UAV network by deploying a ringsigncryption scheme. Hyperelliptic curve cryptography (HECC) and certificate-based cryptography concepts are used to construct the proposed scheme. With security equivalent to RSA, BP, and ECC, HECC uses a key size of 80 bits. The main contributions made by the proposed research are summarized below:

- We present a certificate-based ring signcryption for UAVenabled private edge computing systems utilizing HECC to provide security equivalent to RSA, BP, and ECC with a key size of only 80 bits.
- The proposed scheme performs encryption and digital signature in a one-step operation to anonymously signcrypt data.
- The security analysis of the proposed scheme is performed using the random oracle model (ROM), a formal security tool, and an informal security analysis against several known and unknown attacks to verify the security robustness of the proposed scheme.
- Finally, the proposed scheme's effectiveness is assessed by a thorough comparative analysis. The results demonstrate that the proposed scheme outperforms comparable schemes in terms of computation and communication costs.

The rest of the article is organized as follows: Section II discusses related work. Section III covers preliminaries, which detail the hyperelliptic curve, network model and structure of the proposed scheme. A security analysis of the proposed scheme is presented in Section VI. In Section V, we provide the performance analysis. Finally, Section VI comprises the conclusion.

II. RELATED WORK

During communication, malicious attackers can get the identity and location information of UAV-enabled private edge computing. Several privacy-preserving ring signcryption schemes have been proposed in recent years to address security and privacy problems. Ring signcryption combines

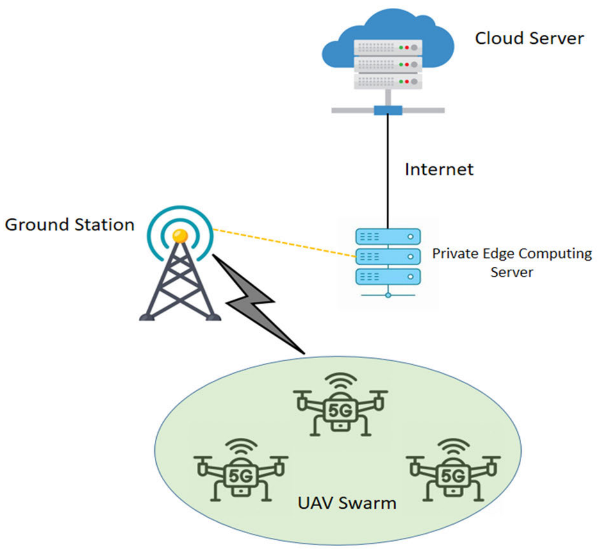

FIGURE 1. An overview of the main entities within the UAV network and a potential cyber-attack scenario.

ring signature with encryption in a single logical step that requires minimal computation and communication costs. Moreover, ring signcryption concurrently accomplishes confidentiality, message authentication, and the complete anonymity of the signcryptor. Guo and Deng [20] developed and validated a certificateless ring signcryption (CLRSC) secure approach under the ROM. The proposed method requires just one BP operation for signcryption but three BP operations to decrypt the message. On the other hand, BP is a mathematically challenging task for a UAV to do ring signcryption. The proposed method has significant consequences for computation and communication costs.

Cai et al. [21] presented a unique solution for conditional privacy protection based on ring signcryption that combines identity-based cryptosystems with ring signatures to offer conditional privacy. However, the approach provided by Cai et al. [21] only allows for one-to-one communication. Moreover, the proposed approach was based on bilinear mapping involving computationally intensive processes. Likewise, Lai et al. [22] proposed a method based on certificateless ring signcryption that permits anonymous authentication and secure communication. If a disagreement arises, the system may also provide a tracking function for vehicles of concern. Gupta and Kumar [23] integrated a ring signature scheme with a signcryption method to provide the anonymity feature for the signcryption scheme, and they addressed the security characteristics. The proposed method is based on the ECC operation, which is somewhat more costly than HECC and uses a key size twice as large as HECC to conduct ring signature and signcryption.

Cui et al. [24] proposed a conditional privacy protection scheme for VANETs based on blockchain and ring signcryption. The scheme offers optional privacy protection for vehicle identification and location. In addition, the scheme employs blockchain technology to eliminate the single point of failure. It immediately distributes the public keys of vehicles, contributing to constructing a ring list. However, the scheme has a high computation cost, and autonomous vehicle tracking of trusted vehicles is not considered. Guo et al. [25] proposed a similar method termed ring signcryption scheme with a conditional privacy-preserving approach based on ECC. The authors also added a tracking mark in the safety message, allowing the trusted party to distinguish malicious vehicles from the member list of the ring. A security analysis using elliptic curve discrete logarithm and elliptic curve computational Diffie-hellman assumptions in the ROM validated the security of this solution.

The scheme we intend to present in this article differs from existing ones. Our certificate-based ring signcryption scheme is based on the concept of hyperelliptic curve cryptography (HECC) for UAV-enabled private edge computing environments, which combines encryption and digital signature in a single step and takes use of HECC's smaller key size for higher security than RSA, BP, and ECC. The HECC is the best option for drones, which are typically resourceconstrained. Furthermore, to our knowledge, HECC has never been employed with ring signcryption in the literature. Tab.1 provides a summary of the existing work and the proposed scheme.

III. PRELIMINARIES

This section details the hyperelliptic curve, network model and structure of the proposed scheme to describe its functioning and implementation. Tab.2 provides details of the notions used in the proposed scheme's algorithm.

A. HYPERELLIPTIC CURVE

Suppose F^q represents a finite field in which the hyperelliptic curve (HEC) is defined over it and q is the order of that field, so we can define HEC by using the following equation HEC : $\lambda^2 + h(\alpha)\lambda = f(\alpha)$, where $h(\alpha) \in F^q(\alpha)$ with the degree of polynomial of δ, $h(\alpha) \in F^q(\alpha)$ utilized the degree like $2\delta + 1$ and indicates that it is a monic polynomial. Here, the main objective is to derive a Jacobian group ($JCB\,(F^q)$) and the devisor $\mathcal{D}$ with a value of 80 bits must be the generator of $JCB\,(F^q)$.

- Hyperelliptic curve discrete logarithm problem
  Here, we consider ($Z = \mathcal{J}.\mathcal{D}$) to be the instance of the hyperelliptic curve and the goal of the challenger $\left(C^A\right)$ is to find $\mathcal{J}$ from Z with the help of F^A with the advantage of ω^{C^A} which would be called hyperelliptic curve discrete logarithm problem for C^A.
- Hyperelliptic diffie-hellman problem
  Here, we consider ($Z = Y.\mathcal{J}.\mathcal{D}$) to be the instance of hyperelliptic curve discrete logarithm problem and the goal of the challenger $\left(C^B\right)$ is to find $\mathcal{J}$ and Y from Z with the help of A^B with the advantage of ω^{C^B} which would be called hyperelliptic curve diffie-hellman problem for C^A.

TABLE 1. Summary of existing work.

Work	Description	Strengths	Weaknesses
Guo and Deng [20]	Proposed a certificateless ring signcryption scheme from pairings.	The proposed scheme requires only one BP operation for signcryption and is proven to be secure under the ROM.	Performing BP is a mathematically complex task for a UAV when executing ring signcryption operation.
Cai *et al.* [21]	Presented a unique solution for conditional privacy protection based on ring signcryption.	The proposed scheme combines identity-based cryptosystems with ring signatures to offer conditional privacy.	Allowing for one-to-one communication. Moreover, the proposed approach is based on bilinear mapping operation that involves computationally intensive processes.
Lai *et al*. [22]	Proposed a method based on certificateless ring signcryption that permits anonymous authentication and secure communication.	The proposed scheme provides a tracking function for vehicles of concern in case a disagreement arises.	Incurs high computation and communication costs, making it impractical for resource-constrained UAV systems.
Gupta and Kumar [23]	Proposed an integrated method of a ring signature scheme with a signcryption.	The proposed scheme provides the anonymity feature for the signcryption scheme and addresses most of the security characteristics.	Uses a key size twice as large as HECC to conduct ring signature and signcryption .
Cui *et al.* [24]	Proposed a conditional privacy protection scheme based on blockchain and ring signcryption.	The proposed scheme offers optional privacy protection for vehicle identification and location. In addition, the scheme employs blockchain technology to eliminate the single point of failure.	Incurs high computation and communication costs. Moreover, the autonomous vehicle tracking of trusted vehicles is not considered.
Guo *et al.* [25]	Proposed a method termed ring signcryption scheme with a conditional privacy-preserving approach based on ECC.	The proposed scheme provides a tracking mark in the safety message, allowing the trusted party to distinguish malicious vehicles from the member list of the ring.	Incurs high computation and communication costs, making it impractical for resource-constrained UAV systems.
Our Scheme	We proposed a certificate-based ring signcryption scheme, based on the concept of HECC for UAV-enabled private edge computing environments.	The proposed scheme combines encryption and digital signature in a single step and takes use of HECC's smaller key size for higher security and efficiency than RSA, BP, and ECC.	Fails to address vulnerabilities posed by machine learning-based and quantum attacks.

TABLE 2. Notation guide.

S.No	Symbol	Description
1	TA	used to signify the trusted authority
2	χ	identifies a received hyperelliptic curve security parameter
3	F^q	identifies a finite field of hyperelliptic curve with order q
4	q	identifies several hyperelliptic curves and their values as q = 80 bits
5	MPB	master public key of TA
6	σ	secret key of TA
7	$\mathcal{D}$	devisor on the hyperelliptic curve with a value of $\mathcal{D}$ = 80 bits
8	HEC	used to signify a genus 2 hyperelliptic curve
9	H_A, H_B, H_C, H_D	identifies four irreversible, collision resistant, and one-way hash function from SHA family
10	ID_{Act}	each actor identity
11	A_i	each joining actor
12	C_i	each actor certificate
13	C_r	receiver actor certificate
14	C_{SAct}	sender actor certificate, which belongs to a sending group
15	$D_{\text{Ʀ}}$	encryption of plain text through a secret key Ʀ
16	ID_{SAct}	sender actor identity which belongs to a sender group
17	ID_r	receiver actor identity
18	$CIPR$	ciphertext
19	$D_{\text{Ʀ}}$	decryption of encrypted text through a secret key Ʀ
20	Ʀ	secret key, which is used for decryption and encryption encrypted and plain text
21	M	plain text

B. NETWORK MODEL

The network model of the proposed scheme comprises the entities, which include an unmanned aerial vehicle (UAV), a ground station (GS), private edge computing (PEC), and cloud computing as illustrated in Fig.1. UAVs, the primary entity of the proposed scheme, have the potential to play a key role in performing a variety of tasks, including parcel delivery, surveillance, etc. The UAVs have essential equipment such as a camera, inertial measurement unit (IMU), sensors, global positioning system (GPS) unit, and flight controller. A wireless connection is established between the UAV and the PEC server via the ground station, which delegates and plans computational duties to the PEC to facilitate rapid processing and local data storage. A certificate request is initiated as the initial step, functioning as the trusted authority (TA). The TA will issue a certificate for the designated UAV and transmit it via an open channel upon receipt of such a request. Before transmitting a message from TA's certificate to PEC, UAV encrypts it with a ring signcryption using the services of GS. When a receiver UAV eventually needs to decrypt the ring-signcrypted message, it will transmit a certificate request to the PEC, which functions as the TA. The TA will then generate a certificate for the requesting UAV and send it to it over an open channel. UAV first produces TA's private key when it receives a certificate and then verifies the signcrypted text.

C. CONSTRUCTION OF THE PROPOSED SCHEME

This section explains the certificate-based ring signcryption scheme for UAV-enabled private edge computing systems. The key notations and definitions used in the proposed scheme are listed in Tab.1. The proposed scheme consists of five phases: setup, key and certificate generation, certificatebased ring signcryption, and certificate-based ring unsigncryption. The following are descriptions of each phase.

1. *Setup:* When a trusted authority (TA) receives the hyperelliptic curve security parameter χ, it executes the setup algorithm to make the secret key and public parameters set that are followed: it chooses σ randomly from Fq and computes. $\mathcal{D}$. TA sets MPB as his/her master public key and σ as a secret key. Then, TA chooses four irreversible, collision-resistant, and one-way hash functions (H_A, H_B, H_C, H_D) from the SHA family. Finally, TA set $\lambda = \{MPB, H\ EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$ as a public parameter set.
2. *Key and certificate generation:* Given the actor's A_i identity ID_{Act}, TA and A_i together execute this phase for making the private key, public key, and certificate. For this process, A_i first, choose $\mathcal{G}_i$ randomly from F^q and compute $\varphi_i = \mathcal{G}_i \cdot \mathcal{D}$, then send (φ_i, ID_{Act}) to TA. When TA receives (φ_i, ID_{Act}), it computes $C_i = \varphi_i + \ell_i.\mathcal{D}$, where ℓ_i is picked randomly by TA from F^q, $E_i = H_A(\sigma.\varphi_i) + \ell_i + \sigma.H_B(C_i, ID_{Act}, MPB)$, and send C_i and E_i to A_i by using an open network. When (C_i, E_i) received to A_i, it computes $\alpha_i = E_i - H_A(\mathcal{G}_i.\text{ MPB})$, $\beta_i = \alpha_i + \mathcal{G}_i$, and compare $\beta_i \cdot \mathcal{D} = C_i + H_B(C_i, ID_{Act}, MPB).MPB$ if this equation is satisfied, then it accepts the certificate.
3. *Certificate-based ring signcryption:* Suppose M is a message to be delivered, ID_{SAct} belongs to $\mathcal{J} = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots., ID_{SActn}\}$, C_{SAct} belongs to $\mathcal{Q} = \{C_{SAct1}, C_{SAct2}, C_{SAct3}, \ldots, C_{SActn}\}$, and $\lambda = \{MPB, H\ EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$ will be taken as input, and then the following computation can be made in the proposed algorithm:
   - The sender with ID_{SAct} and C_{SAct} can choose $\hbar$ randomly from F^q and compute $\mathcal{E} = \hbar.\mathcal{D}$.
   - Compute $\Omega = \hbar.(C_r + H_B(C_r, ID_r, MPB).MPB)$ and $\text{Ʀ} = H_C(\Omega)$.
   - Compute $CIPR = E_{\text{Ʀ}}(\text{M})$ and $S = \hbar + \beta_{SAct}.v$ where $v = H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, CIPR)$, β_{SAct} belongs to $PRTS = \{\beta_{\text{SAct1}}, \beta_{SAct2}, \beta_{SAct3}, \ldots, \beta_{SActn}\}$, which is the private key of one of the senders from the group.
   - Finally, it sends $(S, CIPR, \mathcal{E})$ to the receiver.
4. *Certificate-based ring unsigncryption:* When $(S, C\ IPR, \mathcal{E})$ is received, the receiver does the following for unsigncryption executions.

- Compute $\Omega = \beta_r . \mathcal{E}$ and $\text{Ʀ} = H_C(\Omega)$
- Compute $M = D_{\text{Ʀ}}(CIPR)$ and do for testing the equality of the following equation:
  $S.\mathcal{D} = \mathcal{E} + C_{SAct}.H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, \text{CIPR}) + (C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR).H_B(C_{SAct}, ID_{SAct}, MPB)$ $.MPB$ if it is held, then the receiver accepts the triple of ring signcryption.

OR

- It sets $H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) = r$ and $H_B(C_{SAct}, ID_{SAct}, MPB) = z$.
  It computes $S.\mathcal{D} = \mathcal{E} +$ r. $C_{SAct} +$ r.z. MPB, if it is held, the receiver accepts the triple ring signcryption.

5. *New device/actor adding phase:* Given the new actor's A_{newi} identity ID_{newi}, TA and A_{newi} together execute this phase for making the private key, public key, and certificate. For this process, A_{newi} first, chooses $\mathcal{G}_{newi}$ randomly from F^q and then computes $\varphi_{newi} = \mathcal{G}_{newi} \cdot \mathcal{D}$, sends $(\varphi_{newi}, ID_{newi})$ to TA. When TA receives $(\varphi_{newi}, ID_{newi})$, it first computes $C_{newi} = \varphi_{newi} + \ell_{newi}.\mathcal{D}$, where ℓ_{newi} is picked randomly by TA from F^q, $E_{newi} = H_A(\sigma.\varphi_{newi}) + \ell_{newi} + \sigma.H_B(C_{newi}, ID_{newi}, MPB)$, and send C_{newi} and E_{newi} to A_{newi} by using an open network. When (C_{newi}, E_{newi}) received to A_{newi}, it computes $\alpha_{newi} = E_{newi} - H_A(\mathcal{G}_{newi}.MPB)$, $\beta_{newi} = \alpha_{newi} + \mathcal{G}_{newi}$, and compare $\beta_{newi} \cdot \mathcal{D} = C_{newi} + H_B(C_{newi}, ID_{newi}, MPB).MPB$ if this equation is satisfied, then it accepts the certificate.

D. CORRECTNESS

When $(S, C\ IPR, \mathcal{E})$ is received, the receiver does unsigncryption and verifies the equality of the equation by performing the following operation:

$$\begin{aligned} S \cdot \mathcal{D} &= \mathcal{E} + \mathrm{r} \cdot C_{SAct} + r \cdot z \cdot MPB \\ &= (\text{ƙ} + \beta_{SAct} \cdot H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR)) \cdot \mathcal{D} \\ &= (\text{ƙ} \cdot \mathcal{D} + \beta_{SAct} \cdot H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) \cdot \mathcal{D}) \\ &= (\mathcal{E} + \beta_{SAct} \cdot H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) \cdot \mathcal{D}) \\ &= (\varepsilon + \beta_{SAct} \cdot \mathcal{D} \cdot H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR)) \\ &= (\varepsilon + (C_{SAct} \\ &\quad + H_B(C_{SAct}, ID_{SAct}, MPB) \cdot MPB) \\ &\quad \cdot H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR)) \\ &= (\mathcal{E} + (C_{SAct} \cdot H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) \\ &\quad + H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) \\ &\quad \cdot H_B(C_{SAct}, ID_{SAct}, MPB) \cdot MPB)) \\ &= (\mathcal{E} + (C_{SAct} \cdot \mathrm{r} + z \cdot MPB)) \end{aligned}$$

Also, compute $\Omega = \beta_r . \mathcal{E}$ as $\beta_r . \mathcal{E} = (\alpha_r + \mathcal{G}_r) . \mathcal{E} = (E_i - H_A(\sigma.\varphi_i) + \mathcal{G}_r)\mathcal{E} = (E_i - H_A(\sigma.\varphi_i) + \mathcal{G}_r)\text{ƙ}.\mathcal{D} = (E_i \cdot \mathcal{D} - H_A(\sigma \cdot \varphi_i) \cdot \mathcal{D} + \mathcal{G}_r \cdot \mathcal{D})\text{ƙ}$.

IV. SECURITY ANALYSIS

This section presents formal and informal security analyses, details of which are provided in the following subsections.

A. FORMAL SECURITY ANALYSIS

We consider the following theorems: unforgeability against type 1 forger (F^A), unforgeability against type 2 forger (F^B), confidentiality against type 1 adversary (A^A), and confidentiality against type 2 adversary (A^B), respectively. F^A and A^A can act as an outside adversary and forger, respectively, with the ability to change the user's public key without having access to the master secret key. Next, F^B and A^B can act as an insider forger and an enemy, respectively, with the ability to steal the master secret key but not the user's public key. With the help of the following theorems, we may thus perform security proofs.

Theorem 1 (Confidentiality Against A^A): Here, we consider $(Z = Y.\mathcal{J}.\mathcal{D})$ is the instance of hyperelliptic curve discrete logarithm problem and the goal of the challenger (C^A) is to find $\mathcal{J}$ and Y from Z with the help of A^A with the advantage of ω^{C^A}. We also consider ω^{A^A} of type 1 adversary (A^A) advantages. To prove this theorem, the following query will correspond between CA and the setup phase. The possible probability is as follows:

$$\omega^{C^A} = \frac{1}{\text{Query}_{H_B}}\left(1 - \frac{1}{\text{Query}_{H_B}}\right)^{\text{Query}_{P_{KQ}} + \text{Query}_{C_{RQ}} + \text{Query}_{U_{signQ}}} \left(1 - \frac{\text{Query}_{H_B}}{\text{Query}}\right)\text{Query}_{S_{ignQ}}\,\omega^{A^A}$$

Setup: On the response of $PB = \sigma \cdot \mathcal{D}$ and $\lambda = \{MPB, H\ EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$, A^A send the target identity $(ID_{Acti}{}^*)$ to C^A.

Query$_{H_A}$: Given γ_{Acti}, C^A check-in L_{H_A}, if L_{H_A} contains $(\gamma_{Acti}, \rho_{Acti})$ then it will send ρ_{Acti} to A^A, otherwise, it picks ρ_{Acti} randomly, including $(\gamma_{Acti}, \rho_{Acti})$ into L_{H_A}, and send ρ_{Acti} to A^A.

Query$_{H_B}$: Given $(C_{Acti}, ID_{Acti}, MPB)$, C^A checks in L_{H_B}, if L_{H_B} contains $(C_{Acti}, ID_{Acti}, MPB, \pi_{Acti})$ then it will send π_{Acti} to A^A, otherwise, it picks π_{Acti} randomly, include $(C_{Acti}, ID_{Acti}, MPB, \pi_{Acti})$ into L_{H_B}, and send π_{Acti} to A^A.

Query$_{H_C}$: Given (Ʀ_{Acti}), C^A check in L_{H_C}, if L_{H_C} contain $(\text{Ʀ}_{Acti}, \Delta_{Acti})$ then it will send Δ_{Acti} to A^A, otherwise, it picks Δ_{Acti} randomly, including $(\text{Ʀ}_{Acti}, \Delta_{Acti})$ into L_{H_C}, and send Δ_{Acti} to A^A.

Query$_{H_D}$: Given $(C_{SActi}, ID_{SActi}, \mathcal{E}, C\ IPR)$, C^A check in L_{H_D}, if L_{H_D} contain $(C_{SActi}, ID_{SActi}, \mathcal{E}, C\ IPR, \partial_{Acti})$ then it will send ∂_{Acti} to A^A, otherwise, it picks ∂_{Acti} randomly, including $(C_{SActi}, ID_{SActi}, \mathcal{E}, C\ IPR)$ into L_{H_D}, and send ∂_{Acti} to A^A.

Query$_{C_{UQ}}$: In the create user query $(Query_{C_{UQ}})$, when C^A received a request from A^A, it can check if $ID_{Acti}{}^* = ID_{Acti}$, when it is satisfied, then C^A choose $\mathcal{G}_{Acti}$ and ℓ_{Acti} randomly. C^A compute $\varphi_{Acti} = \mathcal{G}_{Acti} \cdot \mathcal{D}, \Gamma_{Acti} = \ell_{Acti} \cdot \mathcal{D}, C_{Acti} = \Gamma_{Acti} + \varphi_{Acti}$, and include $(\Gamma_{Acti}, \varphi_{Acti}, \mathcal{G}_{Acti}, C_{Acti}, \ell_{Acti}, \perp)$ into $L_{C_{UQ}}$ and send ℓ_{Acti} to A^A. Otherwise, C^A choose $\mathcal{G}_{Acti}$ and α_{Acti} randomly, perform Query$_{H_C}$ to get Δ_{Acti} and compute $\varphi_{Acti} = \mathcal{G}_{Acti}.\mathcal{D}, \Gamma_{Acti} = \alpha_{Acti} \cdot \mathcal{D} - \Delta_{Acti} \cdot MPB, C_{Acti} = \Gamma_{Acti} + \varphi_{Acti}$, and $\beta_{Acti} = \alpha_{Acti} + \mathcal{G}_{Acti}$.

Then, C^A include $(\Gamma_{Acti}, \varphi_{Acti}, \mathcal{G}_{Acti}, C_{Acti}, \ell_{Acti}, \beta_{Acti})$ into $L_{C_{UQ}}$ and send β_{Acti} to A^A.

$Query_{P_{UQ}}$: In the private value query ($Query_{P_{UQ}}$), when C^A received a request from A^A, it can check if $\mathcal{G}_{Acti}$ is existed in $L_{P_{UQ}}$, then C^A send $\mathcal{G}_{Acti}$ to C^A. Otherwise, C^A performs a create user query ($Query_{C_{UQ}}$) and sends $\mathcal{G}_{Acti}$ to A^A.

$Query_{P_{KQ}}$: In the private key query $\left(Query_{P_{KQ}}\right)$, when C^A received a request from AA, it can check if $IDActi* = IDActi$ when it is satisfied then CA will quite, otherwise CA can check if $\beta Acti$ exists in $L_{P_{KQ}}$, then C^A send β_{Acti} to C^A. Else, C^A performs a create user query ($Query_{C_{UQ}}$) and sends β_{Acti} to A^A.

$Query_{C_{RQ}}$: In the certificate generation query ($Query_{C_{RQ}}$), when C^A received a request from AA, it can check if CActi exists in $L_{P_{UQ}}$, then C^A send C_{Acti} to C^A. Otherwise, C^A performs a create user query ($Query_{C_{UQ}}$) and send C_{Acti} to A^A.

$Query_{R_{PBQ}}$: In the replaced public key query ($Query_{C_{RQ}}$), given ID_{Acti} and $\varphi^{/}_{Acti}$, it can check if $ID^*_{Acti} = ID_{Acti}$ when it is satisfied then C^A will quite, otherwise C^A can replace φ_{Acti} on $\varphi^{/}_{Acti}$ and C_{Acti} on $C_{Acti}{}^{/}$.

$Query_{S_{ignQ}}$: Suppose M be a message to be delivered, ID_{Acti} belongs to $\mathcal{J} = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$, C_{SActi} belongs to $Q = \{C_{SAct1}, C_{SAct2}, C_{SAct3}, \ldots, C_{SActn}\}$, and $\lambda = \{MPB, HEC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$ will be taken is input, if $ID^*_{Acti} = ID_{Acti}$. If yes, C^A perform $Query_{C_{RQ}}$ on ID_{Acti} to get C_{Acti} and execute $Query_{H_B}$ on $(C_{Acti}, ID_{Acti}, MPB)$ to get π_{Acti}. Further, C^A choose ∂_{Acti} and S_{Acti} randomly from F^q and compute $\mathcal{E}_{Acti} = S_{Acti}.\mathcal{D} - \partial_{Acti}.C_{Acti} - \pi_{Acti}.\partial_{Acti}\mathcal{I}.\mathcal{D}$. It also computes $\Omega_{Acti} = \beta_{Acti}.E_{Acti}$ on ID_{Acti} and compute $CIPR = E_{\Delta_{Acti}}(\text{M})$. Then, send $(S_{Acti}, CIPR, E_{Acti})$ to A^A.

$Query_{U_{signQ}}$: Suppose $(S_{Acti}, CIPR, \mathcal{E}_{Acti})$, one of the sender's identity ID_{SAct} belongs to $\mathcal{J} = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$, and receiver identity ID_{RAct}, if $ID_{RAct} = ID^*_{Acti}$. If yes, C^A quit; otherwise, it performs the typical un-Signcryption algorithm to get M and dispatch it to A^A.

Challenge: A^A can choose and transmit two same nature and different size plaintext (M^{1*}, M^{2*}) to C^A. So, C^A can execute $Query_{C_{UQ}}$ on ID_{RAct} and ID^*_{Acti}, *Private Value Query* ($Query_{P_{UQ}}$) on ID_{RAct} and ID^*_{Acti}, $Query_{C_{RQ}}$ on ID_{RAct} and ID^*_{Acti}, and get the value of hash queries. According to the processed queries, choose $\rho \in \{0, 1\}$ randomly and perform the execution process to make the signcrypted tuple $(S^*_{Acti}, CIPR^*, \mathcal{E}^*_{Acti})$ and transmit it to A^A.

Guess: A^A guesses $\rho^A \in \{0, 1\}$. If $\rho^A = \rho$, C^A wins, other he will fail.

Probability analysis: Suppose A^A performs all *Hash Queries* $Query_{H_i}(i = A, B, C, D)$, *Create User Query* ($Query_{C_{UQ}}$), *Private Value Query* ($Query_{P_{UQ}}$), *Private Key Query* ($Query_{P_{KQ}}$), *Certificate Generation Query* ($Query_{C_{RQ}}$), *Replaced Public Key Query* ($Query_{C_{RQ}}$), *Signcryption Query* ($Query_{S_{ignQ}}$), *and Un-Signcryption Query* ($Query_{U_{signQ}}$). Ultimately, we can ensure the availability of the following three events.

1. In the event 1 (E_1), C^A never quite in $Query_{P_{KQ}}$, $Query_{C_{RQ}}$, $Query_{S_{ignQ}}$, and $Query_{U_{signQ}}$
2. In the event 2 (E_2), C^A generates a valid signature or cipher text
3. In the event 3 (E_2), $ID^*_{Acti} = ID^*_{SAct}$

The probability for event 1 (E_1) is defined as $Pr(E_1) = (1 - {}^{1}/_{Query_{H_B}})^{Query_{P_{KQ}}+Query_{C_{RQ}}+Query_{U_{signQ}}}$ $(1 - {}^{Query_{H_B}}/_{Query})Query_{S_{ignQ}}$, the probability for event 1 and event 2 as $Pr(E_1|E_2) \geq \omega^{A^A}$, the probability for event 1, event 2, and event 3 as $Pr(E_1|E_2 \wedge E_3) \geq {}^{1}/_{Query_{H_B}}$. The combined probability of the above events is as follows:

$$\omega^{C^A} = \frac{1}{\text{Query}_{H_B}}\left(1 - \frac{1}{\text{Query}_{H_B}}\right)^{\text{Query}_{P_{KQ}}+\text{Query}_{C_{RQ}}+\text{Query}_{U_{signQ}}} \left(1 - \frac{\text{Query}_{H_B}}{\text{Query}}\right)\text{Query}_{S_{ignQ}}\omega^{A^A}$$

Notice that under the above theorem, the adversary and challenger must discover a solution for hyperelliptic curve discrete logarithm to decrypt an encrypted message. If we examine the proposed scheme, we see that the message is encrypted using a secret key $\mathrm{R} = H_C(\Omega)$, where $\Omega = \hbar.(C_r + H_B(C_r, ID_r, MPB).MPB)$ in which the attacker failed to get R. This is because finding the value $\hbar$ from Ω is impractical and will result in a hyperelliptic curve discrete logarithm.

Theorem 2 (Confidentiality Against A^B): Here, we consider $(Z = Y.\mathcal{J}.\mathcal{D})$ is the instance of hyperelliptic curve discrete logarithm problem and the goal of the challenger $\left(C^B\right)$ is to find $\mathcal{J}$ and Y from Z with the help of A^B with the advantage of ω^{C^B}. We also consider ω^{A^B} of type 2 adversary (A^B) advantages. So, for the proof of this theorem, the following query will correspond to C^B and A^B after the setup phase. The combined probability of the above events is as follows:

$$\omega^{C^B} = \frac{1}{\text{Query}_{H_B}}\left(1 - \frac{1}{\text{Query}_{H_B}}\right)^{\text{Query}_{P_{KQ}}+\text{Query}_{C_{RQ}}+\text{Query}_{U_{signQ}}} \left(1 - \frac{\text{Query}_{H_B}}{\text{Query}}\right)\text{Query}_{S_{ignQ}}\omega^{A^B}.$$

Setup: On the response of $PB = \sigma \cdot \mathcal{D}, \sigma$ and $\lambda = \{MPB, H\,EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$, A^B send the target identity $(ID_{Acti}{}^*)$ to C^B.

Queries: The queries such as Hash *Queries* $Query_{H_i}(i = A, B, C, D)$, *Create User Query* ($Query_{C_{UQ}}$), *Private Value Query* ($Query_{P_{UQ}}$), *Private Key Query* ($Query_{P_{KQ}}$), *Certificate Generation Query* ($Query_{C_{RQ}}$), *and Signcryption Query* ($Query_{S_{ignQ}}$) that will perform in this *Theorem* as same as used in *Theorem 1*.

$Query_{U_{signQ}}$: Suppose $(S_{Acti}, CIPR, E_{Acti})$, one of the sender's identity ID_{SAct} belongs to $J = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$, and receiver identity ID_{RAct}, if $ID_{RAct} = ID^*_{Acti}$. If yes, C^B quiet; otherwise, it performs the normal un-signcryption algorithm to get M and dispatch it to A^B.

Challenge: A^B can choose and transmit two same nature and different size plaintext (M^{1*}, M^{2*}) to C^B. So, C^B can execute $Query_{C_{UQ}}$ on ID_{RAct} and ID^*_{Acti}, *Private Value Query* $(Query_{P_{UQ}})$ on ID_{RAct} and ID^*_{Acti}, $Query_{C_{RQ}}$ on ID_{RAct} and ID^*_{Acti}, and get the value of hash queries. According to the processed queries, choose $\rho \in \{0, 1\}$ randomly and perform the execution process to make the signcrypted tuple $(S^*_{Acti}, CIPR^*, \mathcal{E}^*_{Acti})$ and transmit it to A^B.

Guess: A^B guesses $\rho^B \in \{0, 1\}$. If $\rho^B = \rho$, C^B wins, other he will fail.

Probability Analysis: SupposeA^B performs all *Hash Queries* $Query_{H_i}(i = A, B, C, D)$, *Create User Query* $(Query_{C_{UQ}})$, *Private Value Query* $(Query_{P_{UQ}})$, *Private Key Query* $(Query_{P_{KQ}})$, *Certificate Generation Query* $(Query_{C_{RQ}})$, *Signcryption Query* $(Query_{S_{signQ}})$, *and Un-Signcryption Query* $(Query_{U_{signQ}})$. Ultimately, we can ensure the availability of the following three events.

1. In the event 1 (E_1), C^A never quite in $Query_{P_{KQ}}$, $Query_{S_{signQ}}$, and $Query_{U_{signQ}}$
2. In the event 2 (E_2), C^A generates a valid signature or cipher text
3. In the event 3 (E_2), $ID^*_{Acti} = ID^*_{SAct}$

The probability for event 1 (E_1) is defined as $Pr(E_1) = (1 - {}^1/_{Query_{H_B}})^{Query_{P_{KQ}}+Query_{U_{signQ}}}$ $(1 - {}^{Query_{H_B}}/_{Query})Query_{S_{signQ}}$, the probability for event 1 and event 2 as $Pr(E_1|E_2) \geq \omega^{A^B}$, the probability for event 1, event 2, and event 3 as $Pr(E_1|E_2 \wedge E_3) \geq {}^1/_{Query_{H_B}}$. The combined probability of the above events is as follows:

$$\omega^{C^B} = \frac{1}{\text{Query}_{H_B}}\left(1 - \frac{1}{\text{Query}_{H_B}}\right)^{\text{Query}_{P_{KQ}}+\text{Query}_{C_{RQ}}+\text{Query}_{U_{signQ}}} \left(1 - \frac{\text{Query}_{H_B}}{\text{Query}}\right)\text{Query}_{S_{signQ}}\omega^{A^B}$$

To decrypt an encrypted message, an adversary or a challenger must solve a hyperelliptic curve discrete logarithm, as shown in the theorem above. If we examine the proposed scheme, we will see that the message has been decrypted using a secret key $Ʀ = H_C(\Omega)$, where $\Omega = \beta_r. \mathcal{E}$ in which the attacker failed to get Ʀ, because finding the value β_r from Ω is infeasible and will result in hyperelliptic curve discrete logarithm.

Theorem 3 (Unforgeability Against $\boldsymbol{F}^A$): Here, we consider $(Z = \mathcal{J}.\mathcal{D})$ is the instance of hyperelliptic curve discrete logarithm problem and the goal of the challenger (C^A) is to find $\mathcal{J}$ from Z with the help of F^A with the advantage of ω^{C^A}. We also consider ω^{F^A} of type 1 forger (F^A) advantages. So, to prove this theorem, the following query will correspond between C^A and F^A after the setup phase.

Setup: On the response of $PB = \sigma \cdot \mathcal{D}$ and $\lambda = \{MPB, H\ EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$, F^A send the target identity $(ID_{Acti}{}^*)$ to C^A.

$Query_{H_A}$: Given γ_{Acti}, C^A check in L_{H_A}, if L_{H_A} contain ($\gamma_{Acti}, \rho_{Acti}$) then it will send ρ_{Ai} to F^A, otherwise, it pick ρ_{Acti} randomly, include $(\gamma_{Acti}, \rho_{Acti})$ into L_{H_A}, and send ρ_{Acti} to F^A.

$Query_{H_B}$: Given $(C_{Acti}, ID_{Acti}, MPB)$, C^A check in L_{H_B}, if L_{H_B} contain $(C_{Acti}, ID_{Acti}, MPB, \pi_{Acti})$ then it will send π_{Acti} to F^A, otherwise, it pick π_{Acti} randomly, include $(C_{Acti}, ID_{Acti}, MPB, \pi_{Acti})$ into L_{H_B}, and send π_{Acti} to F^A.

$Query_{H_C}$: Given $(Ʀ_{Acti})$, C^A check in L_{H_C}, if L_{H_C} contain $(Ʀ_{Acti}, \Delta_{Acti})$ then it will send Δ_{Acti} to F^A, otherwise, it picks Δ_{Acti} randomly, include $(Ʀ_{Acti}, \Delta_{Acti})$ into L_{H_C}, and send Δ_{Acti} to F^A.

$Query_{H_D}$: Given $(C_{SActi}, ID_{SActi}, \mathcal{E}, C\ IPR)$, C^A check in L_{H_D}, if L_{H_D} contain $(C_{SActi}, ID_{SActi}, \mathcal{E}, C\ IPR, \partial_{Acti})$ then it will send ∂_{Acti} to F^A, otherwise, it picks ∂_{Acti} randomly, include $(C_{SActi}, ID_{SActi}, \mathcal{E}, C\ IPR)$ into L_{H_D}, and send ∂_{Acti} to F^A.

$Query_{C_{UQ}}$: In the create user query $(Query_{C_{UQ}})$, when C^A received a request from F^A, it can check if $ID_{Acti}{}^* = ID_{Acti}$ when it is satisfied, then C^A choose $\mathcal{G}_{Acti}$ and ℓ_{Acti} randomly. C^A compute $\varphi_{Acti} = \mathcal{G}_{Acti}.\mathcal{D}$, $\Gamma_{Acti} = \ell_{Acti}.\mathcal{D}$, $C_{Acti} = \Gamma_{Acti} + \varphi_{Acti}$, and include $(\Gamma_{Acti}, \varphi_{Acti}, \mathcal{G}_{Acti}, C_{Acti}, \ell_{Acti}, \perp)$ into $L_{C_{UQ}}$ and send ℓ_{Acti} to F^A. Otherwise, C^A choose $\mathcal{G}_{Acti}$ and α_{Acti} randomly, perform $Query_{H_C}$ to get Δ_{Acti} and compute $\varphi_{Acti} = \mathcal{G}_{Acti}.\mathcal{D}$, $\Gamma_{Acti} = \alpha_{Acti} \cdot \mathcal{D} - \Delta_{Acti}.MPB$, $C_{Acti} = \Gamma_{Acti} + \varphi_{Acti}$, and $\beta_{Acti} = \alpha_{Acti} + \mathcal{G}_{Acti}$. Then, C^A include $(\Gamma_{Acti}, \varphi_{Acti}, \mathcal{G}_{Acti}, C_{Acti}, \ell_{Acti}, \beta_{Acti})$ into $L_{C_{UQ}}$ and send β_{Acti} to F^A.

$Query\ _{P_{UQ}}$: In the private value query $(Query_{P_{UQ}})$, when C^A received a request from F^A, it can check if $\mathcal{G}_{Acti}$ exists in $L_{P_{UQ}}$, then C^A send $\mathcal{G}_{Acti}$ to C^A. Otherwise, C^A performs a create user query $(Query_{C_{UQ}})$ and sends $\mathcal{G}_{Acti}$ to F^A.

$Query\ _{P_{KQ}}$: In the private key query $(Query_{P_{KQ}})$, when C^A received a request from F^A, it can check if $ID_{Acti}{}^* = ID_{Acti}$ when it is satisfied then C^A will quite, otherwise C^A can check if β_{Acti} exists in $L_{P_{KQ}}$, then C^A send β_{Acti} to C^A. Else, C^A performs a create user query $(Query_{C_{UQ}})$ and sends β_{Acti} to F^A.

$Query_{C_{RQ}}$: In the certificate generation query $(Query_{C_{RQ}})$, when C^A received a request from F^A, it can check if C_{Acti} exists in $L_{P_{UQ}}$, then C^A send C_{Acti} to C^A. Otherwise, C^A performs a create user query $(Query_{C_{UQ}})$ and sends C_{Acti} to F^A.

$Query_{R_{PBQ}}$: In the replaced public key query $(Query_{C_{RQ}})$, given ID_{Acti} and $\varphi_{Acti}{}'$, it can check if $ID_{Acti}{}^* = ID_{Acti}$ when it is satisfied then C^A will quite, otherwise C^A can replace φ_{Acti} on $\varphi_{Acti}/$ and C_{Acti} on C_{Acti} /.

$Query_{S_{\text{ignQ}}}$: Suppose M be a message to be delivered, ID_{Acti} belongs to $\mathcal{J} = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$, C_{SActi} belongs to $\mathcal{Q} = \{C_{SAct11}, C_{SAct2}, C_{SAct3}, \ldots, C_{SActn}\}$, and $\lambda = \{MPB, H\ EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$ will be taken is input, if $ID_{Acti}{}^* = ID_{Acti}$. If yes, C^A perform $Query_{C_{RQ}}$ on ID_{Acti} to get C_{Acti} and execute $Query_{H_B}$ on $(C_{Acti}, ID_{Acti}, MPB)$ to get π_{Acti}. Further, C^A choose ∂_{Acti} and S_{Acti} randomly from F^q and compute $\varepsilon_{Acti} = S_{Acti} \cdot \mathcal{D} - \partial_{Acti}.C_{Acti} - \pi_{Acti}.\partial_{Acti}\mathcal{J}$. D. It also computes $\Omega_{Acti} = \beta_{Acti} \cdot \mathcal{E}_{Acti}$ on ID_{Acti} and compute $CIPR = E_{\Delta_{Acti}}(\text{M})$. Then, send $(S_{Acti}, C\ IPR, \mathcal{E}_{Acti})$ to F^A.

$Query_{U_{signQ}}$: Suppose $(S_{Acti}, C\ IPR, \mathcal{E}_{Acti})$, one of the sender identity ID_{SAct} belongs to $\mathcal{J} = \{ID_{SAct11}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$, and receiver identity ID_{RAct}, if $ID_{RAct} = ID_{Acti}{}^{*}$. If yes, C^A perform $Query_{C_{RQ}}$ on ID_{Acti} to get C_{Acti} and execute $Query_{H_B}$ on $(C_{Acti}, ID_{Acti}, MPB)$ to get π_{Acti}. Further, C^A choose ∂_{Acti} randomly from F^q and verify $S_{Acti} \cdot \mathcal{D} = \mathcal{E}_{Acti} + \partial_{Acti}.C_{Acti} - \pi_{Acti}.\partial_{Acti}MPB$ if holds. If it is holds, C^A access Δ_{Acti} from, recover the plaintext, and send Δ_{Acti} to F^A.

Forgery: on a message M* under one of the sender's identities ID_{SAct} belongs to $\mathcal{J} = \{ID_{SAct11}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$ and receiver identity ID_{RAct}, F^A computes a forged signature $(S_{Acti}, {}^{*}, C\ IPR^{*}, \varepsilon_{Acti}{}^{*})$. According to forking lemma, C^A can also compute the genuine signature $(S_{Acti}, {}^{**}CIPR^{*}, \mathcal{E}_{Acti}{}^{*})$. At the end, the solution obtained by C^A as followed: $\mathcal{J} = \frac{(S_{Acti}{}^{*} - S_{Acti}{}^{**})(\pi_{Acti}{}^{*} - \pi_{Acti}{}^{**})}{\partial_{Acti}{}^{*}}$.

Probability Analysis: Suppose F^A performs all *Hash Queries* $Query_{H_i}(i = A, B, C, D)$, *Create User Query* $(Query_{C_{UQ}})$, *Private Value Query* $(Query_{P_{UQ}})$, *Private Key Query* $(Query_{P_{KQ}})$, *Certificate Generation Query* $(Query_{C_{RQ}})$, *Replaced Public Key Query* $(Query_{C_{RQ}})$, *Signcryption Query* $(Query_{S_{ignQ}})$*, and Un-Signcryption Query* $(Query_{U_{signQ}})$. Ultimately, we can ensure the availability of the following three events.

1. In the event 1 (E_1), C^A never quite in $Query_{P_{KQ}}$, $Query_{C_{RQ}}$, $Query_{S_{ignQ}}$, and $Query_{U_{signQ}}$
2. In the event 2 (E_2), C^A generates a valid signature or cipher text
3. In the event 3 (E_2), $ID^{*}_{Acti} = ID^{*}_{SAct}$

The probability for event 1 (E_1) is define as $Pr(E_1) = (1 - {}^{1}\!/\!_{Query_{H_B}})^{\text{in}Query_{P_{KQ}} + Query_{C_{RQ}} + Query_{U_{signQ}}}$ $(1 - Query_{H_B}/Query) Query_{S_{ignQ}}$, probability for event 1 and event 2 as $Pr(E_1|E_2) \geq \omega^{F^A}$, probability for event 1, event 2, and event 3 as $Pr(E_1|E_2 \wedge E_3) \geq {}^{1}\!/\!_{Query_{H_B}}$. So, the combined probability from the above events as:

$$\omega^{C^A} = \frac{1}{\text{Query}_{H_B}} \left(1 - \frac{1}{\text{Query}_{H_B}}\right)^{\text{Query}_{P_{KQ}} + \text{Query}_{C_{RQ}} + \text{Query}_{U_{signQ}}} \left(1 - \frac{\text{Query}_{H_B}}{\text{Query}}\right) \text{Query}_{S_{ignQ}} \omega^{F^A}$$

As mentioned in the above theorem, an adversary or a challenger must first find the solution of a hyperelliptic curve discrete logarithm to forge a signature. The signature, if we examine the proposed scheme, has been generated as $S = \hbar + \beta_{SAct}.H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, CIPR)$ in which the attacker failed to generate S. This is because finding the values of $\hbar$, β_{SAct} is impossible since it would require solving twotimes hyperelliptic curve discrete logarithm, which is a very challenging task.

Theorem 4 (Unforgeability Against F^B): Suppose $(Z = \mathcal{J}.\mathcal{D})$ is a hyperelliptic curve discrete logarithm problem and the objective of the challenger (C^B) is to compute $\mathcal{J}$ from Z using F^B with the advantage of (C^B). We also analyze the (F^B) advantages of type 1 forger F^B. After the setup step, to prove this theorem, C^B and F^B will exchange the following query.

Setup: On the response of $PB = \sigma \cdot \mathcal{D}, \sigma$,and $\lambda = \{MPB, H\ EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$, F^B sends the target identity $(ID_{Acti}{}^{*})$ to C^B.

Hash Queries: All the hash queries performed in this Theorem are the same as Theorem 3.

$Query_{C_{UQ}}$: When C^A receives a request from F^B, it can check if $ID_{Acti}{}^{*} = ID_{Acti}$. If this condition is satisfied, C^B will then select $\mathcal{G}_{Acti}$ and ℓ_{Acti} at random. Then, C^B $(\Gamma_{Acti}, \varphi_{Acti}, \mathcal{G}_{Acti}, C_{Acti}, \ell_{Acti}, \perp)$ into $L_{C_{UQ}}$ and send ℓ_{Acti} to $F^B.C^B$ computes $\varphi_{Acti} = \mathcal{G}_{Acti} \cdot \mathcal{D}, \Gamma_{Acti} = \ell_{Acti} \cdot \mathcal{D}, C_{Acti} = \Gamma_{Acti} + \varphi_{Acti}$. If not, C^B will randomly select $\mathcal{G}_{Acti}$ and α_{Acti}, run Query $Query_{H_C}$ to obtain Δ_{Acti}, and then compute $\varphi_{Acti} = \mathcal{G}_{Acti} \cdot \mathcal{D}, \Gamma_{Acti} = \alpha_{Acti} \cdot \mathcal{D} - \Delta_{Acti}.MPB, C_{Acti} = \Gamma_{Acti} + \varphi_{Acti}$, and $\beta_{Acti} = \alpha_{Acti} + \mathcal{G}_{Acti}$. After that, C^B adds $(\Gamma_{Acti}, \varphi_{Acti}, \mathcal{G}_{Acti}, C_{Acti}, \ell_{Acti}, \beta_{Acti})$ to $L_{C_{UQ}}$ and send β_{Acti} to F^B.

$Query_{P_{UQ}}$: In the private value query $(Query_{P_{UQ}})$, when C^B received request from F^B, it can check if $\mathcal{G}_{Acti}$ is exist in $L_{P_{UQ}}$, then C^B send $\mathcal{G}_{Acti}$ to C^A. Otherwise, C^B perform a create user query ($Query_{C_{UQ}}$) and send $\mathcal{G}_{Acti}$ to F^B.

$Query_{P_{KQ}}$: This query will perform in this Theorem is the same as *Theorem 1*.

$Query_{C_{RQ}}$: This query that will performed in this *Theorem* is same as *Theorem 1*.

$Query_{S_{ignQ}}$: Suppose M be a message to be delivered, ID_{Acti} belongs to $\mathcal{J} = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$, C_{SActi} belongs to $\mathcal{Q} = \{C_{SAct11}, C_{SAct2}, C_{SAct3}, \ldots, C_{SActn}\}$, and $\lambda = \{MPB, H\ EC, \mathcal{D}, F^q, H_A, H_B, H_C, H_D\}$ will be taken is an input, if $ID_{Acti}{}^{*} = ID_{Acti}$. If yes, C^A perform $Query_{C_{RQ}}$ on ID_{Acti} to get C_{Acti} and execute $Query_{H_B}$ on $(C_{Acti}, ID_{Acti}, MPB)$ to get π_{Acti}. Further, C^A choose ∂_{Acti} and S_{Acti} randomly from F^q and compute $\mathcal{E}_{Acti} = S_{Acti} \cdot \mathcal{D} - \partial_{Acti}.C_{Acti} - \pi_{Acti} \cdot \partial_{Acti}\ \mathcal{J}.\ \mathcal{D}$. It also computes $\Omega_{Acti} = \beta_{Acti}.\ \mathcal{E}_{Acti}$ on ID_{Acti} and compute $CIPR = E_{\Delta_{Acti}}(M)$. Then, send $(S_{Acti}, C\ IPR, \mathcal{E}_{Acti})$ to F^A.

$Query_{U_{signQ}}$: Suppose $(S_{Acti}, CIPR, \varepsilon_{Acti})$, one of the sender identity ID_{SAct} belongs to $\mathcal{J} = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$, and receiver identity ID_{RAct}, if $ID_{RAct} = ID_{Acti}{}^{*}$. If yes, C^A perform $Query_{C_{RQ}}$ on ID_{Acti} to get C_{Acti} and execute $Query_{H_B}$ on $(C_{Acti}, I\ D_{Acti}, MPB)$ to get π_{Acti}. Further, C^A choose ∂_{Acti} randomly from F^q and verify $S_{Acti} \cdot \mathcal{D} = \mathcal{E}_{Acti} + \partial_{Acti}.C_{Acti} - \pi_{Acti}.\partial_{Acti}MPB$ if holds. If it is holds, C^A access Δ_{Acti} from, recover the plaintext, and send Δ_{Acti} to F^A.

Forgery: On a message M* under one of the sender identity ID_{SAct} belongs to $\mathcal{J} = \{ID_{SAct1}, ID_{SAct2}, ID_{SAct3}, \ldots, ID_{SActn}\}$ and receiver identity ID_{RAct}, F^A compute a forged signature $(S_{Acti}{}^{*}, C\ IPR^{*}, \mathcal{E}_{Acti}{}^{*})$. According to the forking lemma, C^A can also compute the genuine signature $(S_{Acti}{}^{**}\ CIPR\ {}^{*}, \varepsilon_{Acti}{}^{*})$. At the end, the solution obtained by C^A as followed: $\mathcal{J} = \frac{(S_{Acti}{}^{*} - S_{Acti}{}^{**})(\pi_{Acti}{}^{*} - \pi_{Acti}{}^{**})}{\partial_{Acti}{}^{*}}$.

Probability Analysis: Suppose F^A performs all *Hash Queries* $Query_{H_i}(i = A, B, C, D)$, *Create User Query*

$(Query_{C_{UQ}})$, *Private Value Query* $(Query_{P_{UQ}})$, *Private Key Query* $(Query_{P_{KQ}})$, *Certificate Generation Query* $(Query_{C_{RQ}})$, *Replaced Public Key Query* $(Query_{C_{RQ}})$, *Signcryption Query* $(Query_{S_{ignQ}})$, *and Un-Signcryption Query* $(Query_{U_{signQ}})$. At the end, we can ensure the availability of the following three events.

1. In the event 1 (E_1), C^A never quite in $Query_{P_{KQ}}$, $Query_{C_{RQ}}$, $Query_{S_{ignQ}}$, and $Query_{U_{signQ}}$
2. In the event 2 (E_2), C^A generates a valid signature or cipher text
3. In the event 3 (E_2), $ID^*_{Acti} = ID^*_{SAct}$

The probability for event 1 (E_1) is defined as $\Pr(E_1) = \left(1 - {}^{1}\!/_{Query_{H_B}}\right)^{\text{in}Query_{P_{KQ}}+Query_{C_{RQ}}+Query_{U_{signQ}}}$ $(1 - {}^{Query_{H_B}}\!/_{Query})Query_{S_{ignQ}}$, probability for event 1 and event 2 as $\Pr(E_1 \mid E_2) \geq \omega^{F^A}$, probability for event 1, event 2, and event 3 as $\Pr(E_1 \mid E_2 \wedge E_3) \geq {}^{1}\!/_{Query_{H_B}}$. The combined probability of the above events is as follows:

$$\omega^{C^A} = \frac{1}{\text{Query}_{H_B}}\left(1 - \frac{1}{\text{Query}_{H_B}}\right)^{\text{Query}_{P_{KQ}}+\text{Query}_{C_{RQ}}+\text{Query}_{U_{signQ}}} \left(1 - \frac{\text{Query}_{H_B}}{\text{Query}}\right)\text{Query}_{S_{ignQ}}\omega^{F^A}$$

Notice that in the preceding theorem, the adversary and the challenger must seek a solution for the discrete logarithm hyperelliptic curve to generate a forgery signature. If we examine the proposed method, we have already generated the signature as $S = \hbar + \beta_{SAct}.H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR)$, in which the attacker failed to generate S because finding the value $\hbar$, β_{SAct} is infeasible and it will result in solving two times hyperelliptic curve discrete logarithm.

B. INFORMAL SECURITY ANALYSIS

The formal security analysis verifies that the proposed scheme is robust against the security criteria of confidentiality and unforgeability and provides the additional security capabilities of message integrity, authentication, non-repudiation, and forward secrecy. These considerations are detailed in further depth below.

Integrity: To check integrity, on the sender side, we calculated $v = H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR)$, and on the receiver side, we computed $H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) = r$. If the receiver wishes to determine if the message has been changed, he/she will compare $v = H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) = H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) = r$; if this condition is fulfilled, the proposed scheme will ensure the message's integrity.

Authentication: We computed the signature as $S = \hbar + \beta_{SAct}.\ v$, and sent it to the receiver to verify authentication under the proposed scheme. The receiver can verify the signature as $S.\mathcal{D} = \mathcal{E} + C_{SAct}.\ H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) + (C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR).H_B(C_{SAct}, ID_{SAct}, MPB).$ MPB. If it holds, the receiver will accept the signature and the authentication will be confirmed. The alternate method of authentication sets $H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) = r$ and $H_B(C_{SAct}, ID_{SAct}, MPB) = z$. It computes $S.\mathcal{D} = \mathcal{E}+$ r. $C_{SAct} + r.z.MPB$. If it holds, the receiver will accept the signature, and authentication will be confirmed.

Non-repudiation: We calculated the signature as $S = \hbar + \beta_{SAct}.v$, where β_{SAct} is the private key of sender and sent to the receiver. The receiver can verify the signature as $S.\mathcal{D} = \mathcal{E} + C_{SAct} \cdot H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) + (C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR).H_B(C_{SAct}, ID_{SAct}, MPB).MPB$. If it holds, the receiver will accept the signature and authentication will be confirmed. The alternate way is that it sets $H_D(C_{SAct}, ID_{SAct}, \mathcal{E}, C\ IPR) = r$ and $H_B(C_{SAct}, ID_{SAct}, MPB) = z$. It computes $S.\mathcal{D} = \mathcal{E}+$ r. $C_{SAct} + r.z.M\ PB$.If it holds then the receiver accepts signature and authentication will be confirmed. In the above two verification equations, we have included C_{SAct} as the public key of the sender; thus, if the sender denies sending the signature, we can simply prove that the sender sent the signature since the public key is directly related to the sender's private key. Thus, according to the proposed scheme, the sender cannot refute his delivered signature.

Forward secrecy: Even if the proposed scheme's private key is compromised, the ciphertext will remain secure since encryption and decryption were performed using the secret key. If the attacker wishes to compromise the security of the cipher text, they must satisfy the two requirements outlined below.

1. We encrypted the message through the secrete key as $\natural = H_C(\Omega)$, where $\Omega = \hbar.(C_r + H_B(C_r, ID_r, MPB).MPB)$ and the attacker fails to get $\natural$ because finding the value $\hbar$ from Ω is infeasible, leading to solving the hyperelliptic curve discrete logarithm, which is very challenging.
2. We decrypted the message through the secrete key as $\natural = H_C(\Omega)$, where $\Omega = \beta_r.\ \mathcal{E}$, and the attacker failed to get $\natural$ because finding the value β_r from Ω is infeasible, leading to the hyperelliptic curve discrete logarithm.

C. SECURITY VERIFICATION USING THE AVISPA TOOL

This section performs the simulation of the proposed using the AVISPA tool [26], a formal security verification method to determine the cryptographic scheme resilience against replay and man-in-the-middle (MitM) attacks. The security characteristics are simulated using an expressive and modular formal language in the AVISPA with the assistance of the high-level protocol specification language (HLPSL). We used a Haier Win8.1 PC workstation with an Intel (R) Core (TM) i3-4010U CPU @ 1.70-GHz and a 64-bit operating system to execute the simulations of the proposed certificate-based ring signcryption scheme. Moreover, Oracle VM Virtual Box (version: 5.2.0.118431) and SPAN (version: SPAN-Ubuntu-10.10-light 1) make up the software portion of the simulation setup. The OFMC and CL-AtSe are executed at the back ends for the vulnerability tests. We have not considered the results of SATMC and TA4SP are not included in the simulation results due to the bitwise XOR operations, which are incompatible with SATMC and TA4SP. The

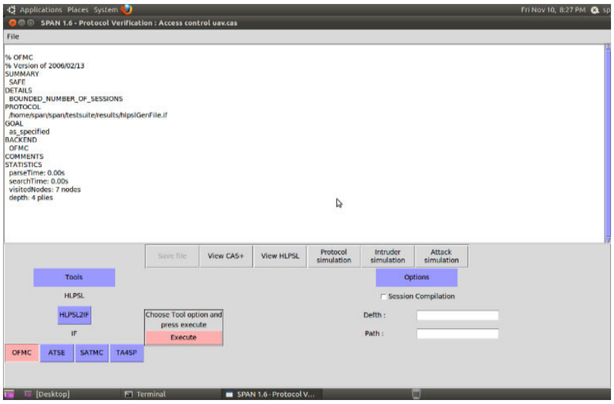

FIGURE 2. Simulation results for OFMC.

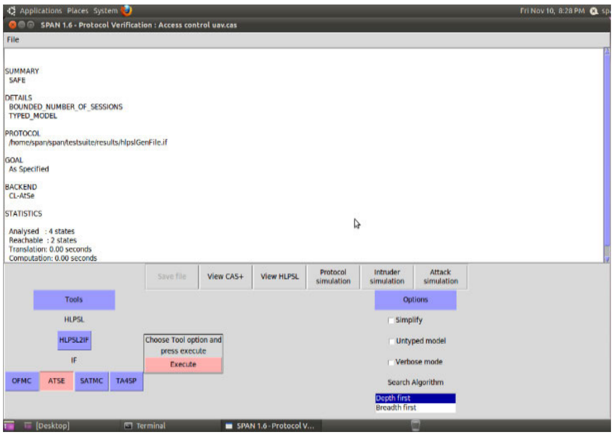

FIGURE 3. Simulation results for ATSE.

proposed scheme is also simulated using the well-known web tool known as specific protocol animator (SPAN) The findings collected from OFMC and AtSe as illustrated in Fig. 2 and 3 authenticate the effectiveness against replay and MitM attacks.

V. PERFORMANCE ANALYSIS

This analysis examines the proposed scheme's efficiency based on its computation and communication costs. It does this by comparing it to other comparable schemes.

A. COMPUTATION COST

Based on the major operations such as elliptic curve scalar addition, hyperelliptic curve scalar addition, elliptic curve scalar multiplication, hyperelliptic curve devisor multiplication, modular exponentiation, pairing multiplication operation, and bilinear pairing, the proposed scheme is compared to those proposed by Guo and Deng [20], Cai et al. [21], Gupta and Kumar [23],Cui et al. [24], and Guo et al. [25]. In Tab. 3, the symbols $\mathcal{ESA}$, $H\mathcal{ESA}$, $\mathcal{ESM}$, $H\mathcal{EDM}$, $M\mathcal{EN}$, PMO, and BP represent the time needed for elliptic curve scalar addition, hyperelliptic curve scalar addition, elliptic curve scalar multiplication, hyperelliptic curve devisor multiplication, modular exponentiation, pairing multiplication operation, and bilinear pairing. The computation cost is mainly determined by the amount of computation involved for the signcryption algorithm and decryption verification calculation.

TABLE 3. Major operation costs.

Symbol	Operation	Running time (ms)
$\mathcal{ESA}$	Time required for a single elliptic curve scalar addition	0.002
$H\mathcal{ESA}$	Time required for a single hyper elliptic curve scalar addition	0.001
$\mathcal{ESM}$	Time required for a single elliptic curve scalar multiplication	0.341
$H\mathcal{EDM}$	Time required for a single hyper elliptic curve devisor multiplication	0.1705
$M\mathcal{EN}$	Time required for a single modular exponentiation	1.915
PMO	Time required for a single pairing multiplication	0.788
BP	Time required for single bilinear pairing	4.669

TABLE 4. Computation costs.

Schemes	Ring Signcryption	Ring Unsigncryption	Total
[20]	$10PMO + 1BP$	$4PMO + 2BP$	$14PMO + 3BP$
[21]	$7PMO + 1BP$	$5PMO + 3BP$	$12PMO + 4BP$
[23]	$6\mathcal{ESM} + 1\mathcal{ESA}$	$9\mathcal{ESM} + 3\mathcal{ESA}$	$15\mathcal{ESM} + 4\mathcal{ESA}$
[24]	$6PMO + 1M\mathcal{EN} + 1BP$	$6BP + 3PMO$	$9PMO + 1M\mathcal{EN} + 7BP$
[25]	$14\mathcal{ESM} + 10\mathcal{ESA}$	$9\mathcal{ESM} + 9\mathcal{ESA}$	$23\mathcal{ESM} + 19\mathcal{ESA}$
Ours	$3H\mathcal{EDM} + 2H\mathcal{ESA}$	$5H\mathcal{EDM} + 2H\mathcal{ESA}$	$8H\mathcal{EDM} + 4H\mathcal{ESA}$

According to Ref. [27], the time required for these operations is listed in Tab.4 is considered. For this experiment, the following execution environment was utilized: CPU: Intel Core i7-6700 @ 3.40GHz; RAM: 8GB; OS: Ubuntu 16.04; MIRACL library. As is well-known, HECC needs half the key size of ECC to provide the same degree of security. Tab. 3 compares the computation performance of the proposed scheme to that of the schemes proposed by Guo and Deng [20], Cai et al. [21], Gupta and Kumar [23], Cui et al. [24], and Guo et al. [25] based on the key operations. As shown in Tab. 5 and illustrated in Fig.4, the proposed scheme is more efficient than its counterpart in terms of computation costs measured in milliseconds, supporting the scheme's feasibility in the UAV-enabled private edge computing environment.

TABLE 5. Computation costs (in ms).

Schemes	Ring Signcryption	Ring Unsigncryption	Total (ms)
[20]	12.55	12.49	25.04
[21]	10.18	17.95	28.12
[23]	2.048	3.075	5.123
[24]	11.312	30.348	41.66
[25]	4.794	3.087	7.88
Ours	0.5135	0.8545	1.368

TABLE 6. Communication cost.

Schemes	[20]	[21]	[23]	[24]	[25]	Ours
Comm. cost	$\|M\| + 6\|G\|$	$2\|M\| + 5\|G\|$	$\|M\| + 4\|Q\|$	$\|M\| + 8\|G$	$\|M\| + 7\|Q\|$	$\|M\| + 2\|N\|$

TABLE 7. Communication cost (in bits).

Schemes	[20]	[21]	[23]	[24]	[25]	Ours
Comm. cost	7168	7168	1664	9216	2144	1184

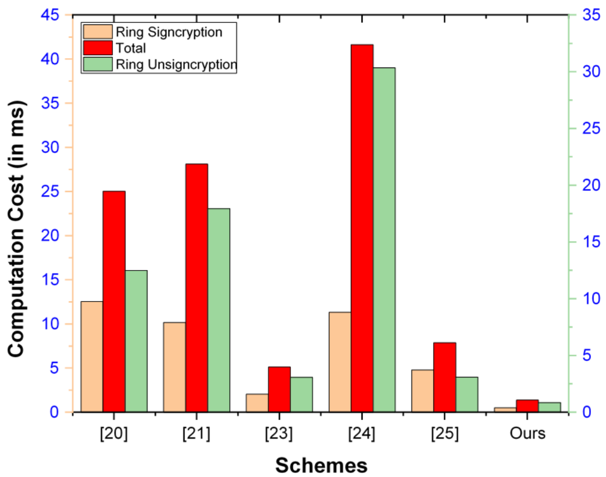

FIGURE 4. Comparative analysis based on computation cost (in ms).

B. COMMUNICATION COST

Communication costs refer to the number of bits that must be transferred in addition to the cipher text or message during the transmission session. Extra bits are often counted as elliptic curve parameter size, hyperelliptic curve parameter size, and bilinear pairing parameter size when calculating the communication cost. Tab.5 provides a comparison of the communication cost between the schemes proposed by Guo and Deng [20], Cai et al. [21], Gupta and Kumar [23], Cui et al. [24], and Guo et al. [25] based on the main operations. Communication costs equal the number of extra bits In Tab.6, The symbols |M|, |N|, |G|, and |Q| stand for the size of the message/cipher text, the size of the hyperelliptic curve parameter, the size of the bilinear pairing parameter, and elliptic curve parameter, and they use 1024 bits, 80 bits, 1024 bits, and 180 bits, respectively.

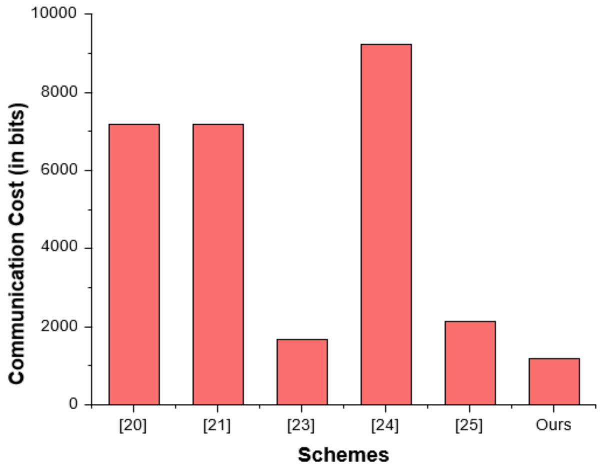

FIGURE 5. Comparative analysis based on communication cost (in bits).

Tab.7 and Fig. 5 compare communication costs in bits, which reveals that the proposed scheme has lower communication costs.

VI. CONCLUSION

UAV-enabled private edge computing systems involve the integration of UAVs into a private edge computing infrastructure. In this system, UAVs are outfitted with a variety of data-collecting sensors and devices within this system; the data is processed locally on an edge computing server. The open wireless channel, nevertheless, renders these systems susceptible to security threats. Threats to UAV-enabled private edge computing systems's security and privacy can be categorized as either violation threats, deliberate threats, or accidental threats. Similarly, while designing security measures for these systems, high scalability, device diversity, and mobility must be considered. Keeping these vulnerabilities in mind, ring signcryption, which provides advantageous characteristics such as anonymity, spontaneity, flexibility, and equal membership, is the most appropriate cryptographic technique. In this article, we proposed a certificate-based ring signcryption method based on HECC that combines encryption and digital signature in a single step and uses the lower key size of HECC to provide more security than RSA, BP, and ECC. The computation and communication costs of the proposed scheme is 1.368(in ms) and 1184 (in bits) respectively, which is significantly less than by the relevant existing schemes. All of these outcomes indicate the practicality of the proposed scheme. In the future work, integrating blockchain or federated learning to the proposed scheme can further enhance the security of UAV-enabled private edge computing systems.

REFERENCES

[1] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif, "Towards the unmanned aerial vehicles (UAVs): A comprehensive review," *Drones*, vol. 6, no. 6, p. 147, Jun. 2022.

[2] J. Wang, K. A. Alattas, Y. Bouteraa, O. Mofid, and S. Mobayen, "Adaptive finite-time backstepping control tracker for quadrotor UAV with model uncertainty and external disturbance," *Aerosp. Sci. Technol.*, vol. 133, Feb. 2023, Art. no. 108088.

[3] A. Najafi, M. T. Vu, S. Mobayen, J. H. Asad, and A. Fekih, "Adaptive barrier fast terminal sliding mode actuator fault tolerant control approach for quadrotor UAVs," *Mathematics*, vol. 10, no. 16, p. 3009, Aug. 2022.

[4] V. W. Wong, R. Schober, D. W. K. Ng, and L.-C. Wang, *Key Technologies for 5G Wireless Systems*. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[5] Q.-V. Pham, F. Fang, V. N. Ha, Md. J. Piran, M. Le, L. B. Le, W.-J. Hwang, and Z. Ding, "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art," *IEEE Access*, vol. 8, pp. 116974–117017, 2020.

[6] I. Alam, K. Sharif, F. Li, Z. Latif, M. M. Karim, S. Biswas, B. Nour, and Y. Wang, "A survey of network virtualization techniques for Internet of Things using SDN and NFV," *ACM Comput. Surv.*, vol. 53, no. 2, pp. 1–40, Mar. 2021.

[7] C. De Alwis, A. Kalla, Q.-V. Pham, P. Kumar, K. Dev, W.-J. Hwang, and M. Liyanage, "Survey on 6G frontiers: Trends, applications, requirements, technologies and future research," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 836–886, 2021.

[8] A. Filali, A. Abouaomar, S. Cherkaoui, A. Kobbane, and M. Guizani, "Multi-access edge computing: A survey," *IEEE Access*, vol. 8, pp. 197017–197046, 2020.

[9] B. Liang, M. A. Gregory, and S. Li, "Multi-access edge computing fundamentals, services, enablers and challenges: A complete survey," *J. Netw. Comput. Appl.*, vol. 199, Mar. 2022, Art. no. 103308.

[10] P. Fondo-Ferreiro, A. Estévez-Caldas, R. Pérez-Vaz, F. Gil-Castiñeira, F. J. González-Castaño, S. Rodríguez-García, X. R. Sousa-Vázquez, D. López, and C. Guerrero, "Seamless multi-access edge computing application handover experiments," in *Proc. IEEE 22nd Int. Conf. High Perform. Switching Routing (HPSR)*, Jun. 2021, pp. 1–6.

[11] S. C. Shah, "Private mobile edge cloud for 5G network applications," *Internet Technol. Lett.*, vol. 2, no. 5, p. e124, Sep. 2019.

[12] B. Ali, M. A. Gregory, and S. Li, "Multi-access edge computing architecture, data security and privacy: A review," *IEEE Access*, vol. 9, pp. 18706–18721, 2021.

[13] M. Yahuza, M. Y. I. Idris, A. W. A. Wahab, T. Nandy, I. B. Ahmedy, and R. Ramli, "An edge assisted secure lightweight authentication technique for safe communication on the Internet of Drones network," *IEEE Access*, vol. 9, pp. 31420–31440, 2021.

[14] X. Wang, Y. Wang, X. Su, L. Wang, C. Lu, H. Peng, and J. Liu, "Deep reinforcement learning-based air combat maneuver decision-making: Literature review, implementation tutorial and future direction," *Artif. Intell. Rev.*, vol. 57, no. 1, Dec. 2023.

[15] X. Gao, L. Wang, X. Yu, X. Su, Y. Ding, C. Lu, H. Peng, and X. Wang, "Conditional probability based multi-objective cooperative task assignment for heterogeneous UAVs," *Eng. Appl. Artif. Intell.*, vol. 123, Aug. 2023, Art. no. 106404.

[16] M. Wazid, A. K. Das, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues, "Design and analysis of secure lightweight remote user authentication and key agreement scheme in Internet of Drones deployment," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3572–3584, Apr. 2019.

[17] 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, Mar. 2021, Art. no. 103477.

[18] M. Tanveer, A. U. Khan, H. Shah, S. A. Chaudhry, and A. Naushad, "PASKE-IoD: Privacy-protecting authenticated key establishment for Internet of Drones," *IEEE Access*, vol. 9, pp. 145683–145698, 2021.

[19] 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, Apr. 2022, Art. no. 110894.

[20] H. Guo and L. Deng, "Certificateless ring signcryption scheme from pairings," *Int. J. Netw. Secur.*, vol. 22, no. 1, pp. 102–111, Jan. 2020.

[21] Y. Cai, H. Zhang, and Y. Fang, "A conditional privacy protection scheme based on ring signcryption for vehicular ad hoc networks," *IEEE Internet Things J.*, vol. 8, no. 1, pp. 647–656, Jan. 2021.

[22] C. Lai, G. Li, and D. Zheng, "SPSC: A secure and privacy-preserving autonomous platoon setup and communication scheme," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 9, p. e3982, Sep. 2021.

[23] P. Gupta and M. Kumar, "A verifiable ring signature scheme of anonymous signcryption using ECC," *Int. J. Math. Sci. Comput.*, vol. 7, no. 2, pp. 24–30, Jun. 2021.

[24] N. C. Nan Cui and H. M. Nan Cui, "Conditional privacy protection scheme based on blockchain and ring signcryption in VANETs," *J. Comput.*, vol. 33, no. 2, pp. 177–188, Apr. 2022.

[25] R. Guo, L. Xu, X. Li, Y. Zhang, and X. Li, "An efficient certificateless ring signcryption scheme with conditional privacy-preserving in VANETs," *J. Syst. Archit.*, vol. 129, Aug. 2022, Art. no. 102633.

[26] AVISPA. *Automated Validation of Internet Security-Sensitive Protocols and Applications*. Accessed: May 2024. [Online]. Available: http://www.avispa-project.org/

[27] B. Cui, L. Wei, and W. He, "A new certificateless signcryption scheme for securing Internet of Vehicles," Tech. Rep., Jan. 2022, doi: 10.21203/rs.3.rs-1272183/v1.

MUHAMMAD ASGHAR KHAN (Senior Member, IEEE) received the Ph.D. degree in electronic engineering from the School of Engineering and Applied Sciences, Isra University, Islamabad, Pakistan. He is currently an Associate Professor and heads the Electrical Engineering Department, Hamdard University, Islamabad. He is also a Research Fellow with the Smart Systems Engineering Laboratory, Prince Sultan University, Riyadh, Saudi Arabia. With an extensive publication history, he has authored or coauthored over 100 technical and review articles, prominently featured in reputable journals, such as IEEE Transactions on Vehicular Technology, IEEE Transactions on Industrial Informatics, and IEEE Internet of Things Journal. His research interests include drones/UAVs, specifically focusing on networks, platforms, security, and applications and services. He has presented his research findings at numerous national and international conferences. Actively engaged in academia, he contributes as a reviewer for distinguished journals published by IEEE, Elsevier, and Springer. He has also served as the guest editor for various international journals. In recognition of his outstanding scholarly contributions, Stanford University acknowledged him as one of the top 2% of highly cited scientists globally, in 2023.

INSAF ULLAH is a Research Fellow (Grade eight position with a path to permanency to Grade nine Lecturer) at the Institute for Analytics and Data Science, University of Essex, Colchester, U.K. He secured an endorsement for Global Talent from the Royal Academy of Engineering, U.K., for his role. Prior to joining the University of Essex, he was an Assistant Professor at the Department of Computing, Hamdard University, Islamabad Campus, Pakistan. He has authored and co-authored over 60 articles in leading journals, such as the IEEE Transactions on Vehicular Technology, IEEE Transactions on Industrial Informatics, IEEE Internet of Things Journal, and IEEE Transactions on Network and Service Management. With more than 1600 Google Scholar Citations, he has performed reviewer and guest editor roles with several reputed IEEE, Springer, ACM, and *Elsevier journals*. He has also engaged with conferences and workshops to organize committees and presenters. His research focuses on network security, intelligent transportation, applied cryptography, IoT, IoV, IoD, WBAN, and IIoT.

NEERAJ KUMAR (Senior Member, IEEE) is currently a Full Professor with the Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology (Deemed to be University), Patiala, India. He has published more than 400 technical research papers in top-cited journals and conferences, which are cited more than 29 410 times from well-known researchers across the globe with a current H-index of 93. He has guided many research scholars leading to Ph.D. and M.E./M.Tech. His research was supported by funding from various competitive agencies across the globe. His broad research areas are green computing and network management, the IoT, big data analytics, deep learning, and cyber-security. He has also edited/authored ten books with international/national publishers like IET, Springer, Elsevier, and CRC, such as *Security and Privacy of Electronic Healthcare Records: Concepts, Paradigms and Solutions, Machine Learning in Cognitive IoT* (CRC Press), *Blockchain, Big Data and Machine Learning* (CRC Press), *Blockchain Technologies Across Industrial Vertical* (Elsevier), *Multimedia Big Data Computing for IoT Applications: Concepts, Paradigms and Solutions* by the Proceedings of First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019), and *Probabilistic Data Structures for Blockchain-Based Internet of Things Applications* (CRC Press). One of the edited textbook titled *Multimedia Big Data Computing for IoT Applications: Concepts*, Paradigms and Solutions published by Springer in 2019 has 3.5 million downloads till June 2020. It attracts the attention of researchers across the globe. He is a Highly-Cited Researcher from WoS, from 2019 to 2021.

FATEMEH AFGHAH (Senior Member, IEEE) received the B.Sc. and M.Sc. (Hons.) degrees in electrical engineering from the Khajeh Nasir Toosi University of Technology (KNTU), Tehran, in 2005 and 2008, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Maine, in 2013. She was an affiliated faculty with the Partnership for Native American Cancer Prevention between Arizona Cancer Center and Northern Arizona University (NAU). She was a Visiting Graduate Student with the ECE Department, University of Maryland, College Park, MD, USA, from 2012 to 2013. She is currently a tenured Associate Professor with the Department of Electrical and Computer Engineering, Clemson University. Before joining NAU, she was an Associate Professor (with tenure) with the School of Informatics, Computing and Cyber Systems, NAU. She is also the Director of the Intelligent Systems and Wireless Networking (IS-WiN) Laboratory. She was a recipient of the NSF CAREER Award, in 2020, the AFOSR Young Investigator Award, in 2019, the NAU's Most Promising New Scholar Award, in 2020, the NSF CISE Research Initiative Initiation (CRII) Award, in 2017, and the AFRL Visiting Research Faculty Award, in 2016 and 2017. She was the Chair and an Organizer of the IEEE Communications and Signal Processing Chapter at the IEEE Central North Carolina Section. She serves as an Associate Editor for *Computer Networks* (Elsevier), *Journal of Network and Computer Applications* (Elsevier), *Ad Hoc Journal* (Elsevier), *ACM Transactions on Computing for Healthcare*, *Neural Processing Letters* (Springer), *IET Wireless Sensor Systems*, and *Frontiers Aerial and Space Networks*. She was a Representative of IEEE Regions R1–6 on the Membership Board Standing Committee for the IEEE Signal Processing Society, from 2016 to 2018.

GORDANA BARB received the M.Sc. degree in electronics and telecommunications from the University of Aveiro, in 2017, and the Ph.D. degree in electronics and telecommunications from the Politehnica University of Timisoara, in 2021. She is currently a University Lecturer with the Communications Department, Politehnica University of Timiçoara, and an RF Engineer. Her research interests include 5G communication systems, cybersecurity, UAVs, massive MIMO, V2X, and millimeter-wave communications. She has contributed to various international journals with her research publications and has also presented her work in international conference proceedings.

FAZAL NOOR received the B.Eng. and M.Eng. degrees in electrical and computer engineering from Concordia University, Montreal, QC, Canada, in 1984 and 1986, respectively, and the Ph.D. degree in engineering from McGill University, Montreal, QC, Canada, in 1993. He is currently a Full Professor with the Faculty of Computer and Information Systems, Islamic University of Madinah, Saudi Arabia. He has authored or coauthored numerous papers in various reputable international journals and conferences. His research interests include AI, FANETS, neural networks, embedded systems, signal processing, security, the IoT, optimization algorithms, and parallel and distributed computing. He is a fellow of IAER. He was a recipient of the Best Faculty Award, in 2007. He held the position of the Vice Dean of Graduate Studies and Scientific Research with FCIS and a Program Coordinator for the Master of Computer Science Program. He has been a TPC member of many conferences. He is also a QA Evaluator of the Computer Engineering Program. He is a reviewer of IEEE, Elsevier, Springer, and various other journals.

SAAD ALQAHTANY received the B.Sc. degree in computer science from King Saud University, Riyadh, Saudi Arabia, in 2002, the master's degree in digital forensics from Bradford University, U.K., in 2010, and the Ph.D. degree in cyber security and digital forensics from Plymouth University, U.K., in 2017. He worked for the Interior Ministry of Saudi Arabia for over 19 years. He has held many positions related to cybersecurity and digital investigation. He was an Experienced Head of the Cyber Crime and Digital Forensics Unit with a demonstrated history of working in the computer and network security industry. He has published several papers and journals in local and international conferences. His prominent companies in Islamabad. His contributions to the field are evident through the publication of around 25 papers in various journals and conferences. His professional interests focus on digital forensics, cyber security, cybercrime investigations, IoT forensics, and cloud forensics. His current interest includes enabling cyberinfrastructure for digital investigation and fighting cyber criminals forensically and proactively.