An Efficient Identity-Based Aggregate Signcryption Scheme With Blockchain for IoT-Enabled Maritime Transportation System

Yi Yang, Debiao He, *Member, IEEE*, Pandi Vijayakumar, *Senior Member, IEEE*, Brij B. Gupta, *Senior Member, IEEE*, and Qi Xie

***Abstract*—Maritime transportation has ushered in a period of vigorous development and has been the most critical mode of transport in international trade. Maritime communication is also becoming increasingly complex. This trend needs the IoT-enabled Maritime Transportation System (IMTS) to enhance its ability of privacy-preserving, performance continuously, and add the function of joint management. Although many aggregate signcryption (ASC) schemes have been presented to satisfy this requirement, they still suffer from performance deficiencies or security weaknesses. To address these issues, we design a more practical "Perception-Network-Application" IoT-enabled MTS (PNA-IMTS) network structure. Then, we propose an efficient Identity-based aggregate signcryption scheme with blockchain (B-ID-ASC) for PNA-IMTS based on this network model. The security analysis shows that the B-ID-ASC scheme is capable of providing confidentiality and unforgeability. It also can resist existing known attacks (impersonation attack, modification attack, man-in-the-middle attack). The performance analysis indicates that our B-ID-ASC scheme, having lower computation and communication costs than current ASC schemes, is more applicable for PNA-IMTS.**

***Index Terms*—IMTS, PNA-IMTS, blockchain, B-ID-ASC, confidentiality, unforgeability.**

Manuscript received 10 October 2021; revised 13 February 2022; accepted 25 March 2022. Date of publication 30 March 2022; date of current version 19 August 2022. This work was supported in part by the Major Scientific and Technological Innovation Project of Shandong Province under Grant 2020CXGC010115; in part by the National Natural Science Foundation of China under Grant U21A20466, Grant 62172307, Grant 61972294, and Grant 61932016; in part by the Peng Cheng Laboratory Project under Grant PCL2021A02; in part by the Special Project on Science and Technology Program of Hubei Provience under Grant 2020AEA013; in part by the Natural Science Foundation of Hubei Province under Grant 2020CFA052; and in part by the Wuhan Municipal Science and Technology Project under Grant 2020010601012187. *(Corresponding authors: Debiao He; Pandi Vijayakumar; Brij B. Gupta; Qi Xie.)*

Yi Yang is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China, and also with the Shandong Provincial Key Laboratory of Computer Networks, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250014, China (e-mail: yangyi.ip@qq.com).

Debiao He is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China, also with the Institute of Network System and Security, Peng Cheng Laboratory, Shenzhen 518055, China, and also with the Shanghai Key Laboratory of Privacy-Preserving Computation, MatrixElements Technologies, Shanghai 201204, China (e-mail: hedebiao@163.com).

Pandi Vijayakumar is with the Department of Computer Science and Engineering, University College of Engineering Tindivanam, Tindivanam 604001, India (e-mail: vijibond2000@gmail.com).

Brij B. Gupta is with the Department of Computer Science and Information Engineering, Asia University, Taichung 40704, Taiwan, also with the Department of Management Information Systems, King Abdulaziz University, Jeddah 21589, Saudi Arabia, and also with the School of Digital, Technologies & Arts, Staffordshire University, Stoke-on-Trent ST4 2DE, U.K. (e-mail: gupta.brij@mail.com).

Qi Xie is with the Key Laboratory of Cryptography of Zhejiang Province, Hangzhou Normal University, Hangzhou 311121, China (e-mail: qixie68@126.com).

Digital Object Identifier 10.1109/TGCN.2022.3163596

I. Introduction

RECENTLY, maritime transportation [1]–[3] has gradually become one of the essential import-export-trading means due to its advantages of large transportation volume and low unit price [4]. With the continuous development of marine transportation, the communication, rescue, and regulatory demands are also increasing. Therefore, relevant maritime authorities (including Monitoring Center (MC), Maritime Bureau (MB), Relief Center (RC), etc.) have established the IoT-enabled Maritime Transportation System (IMTS) [5], [6]. The IMTS is responsible for integrating and storing ship navigation data, daily monitoring data, and ship sensor data. Furthermore, this service can help relevant entities to carry out comprehensive maritime management [7]–[10]. Its core technology is to collect the dynamic parameters (ocean velocity, regional vessel density, temperature, etc.) through sensors, and then carry out data storage, calculation and message transmission through a unified platform, and finally unified scheduling. An efficient IMTS not only concerns the security of vessels, but also can greatly save the transportation cost of shipping companies.

A traditional structure of IMTS [11]–[13] is displayed in Fig. 1. There are four layers: detecting devices layer, patrol vessels layer, relay stations layer, and organizations layer [14]:

- *Detecting devices layer:* In this layer, detecting devices (boats, buoys, etc.) are responsible for collecting maritime traffic information (e.g., traffic density, weather condition, rescue condition) and hydrological information (e.g., water quality, water temperature, flow rate). The boats (IoT devices) use wireless sensors to get maritime information and send these information through wireless network. The buoys use specific sensors, being incapable of communicating, to get and store maritime information.
- *Patrol vessels layer:* In this layer, patrol vessels are responsible for getting maritime information from detecting devices and transmitting them to the relay stations via the wireless network. The patrol vessels from MB and RC get maritime information from IoT-enabled boats

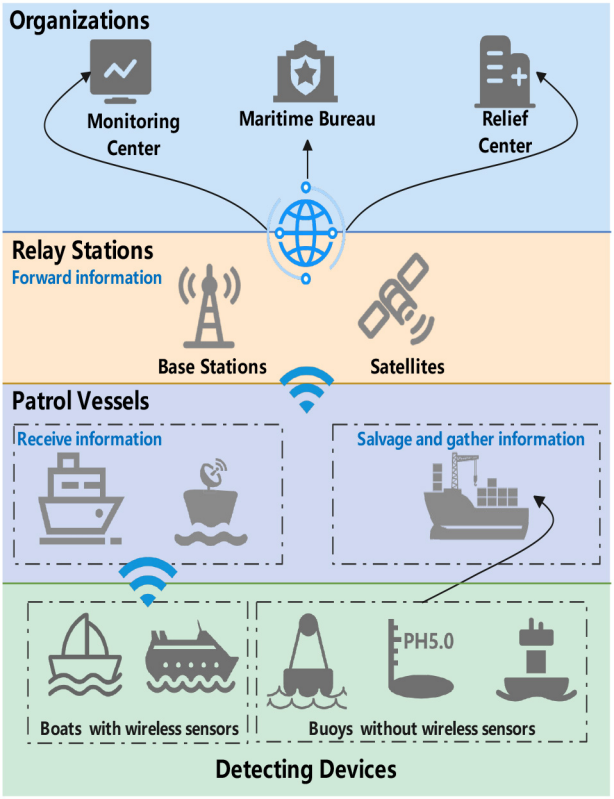

Fig. 1. A Traditional Structure of IoT-enabled Maritime Transportation System.

through the wireless network. The patrol vessels from MC salvage buoys to get navigational information. After gathering navigational information, the patrol vessels from MC reposition buoys.

- *Relay stations layer:* In this layer, relay stations (seaside base stations, satellites, etc.) are responsible for forwarding the information received to the appropriate organizations. When the patrol vessels are at the seaside, choosing seaside base stations as a relay station. When the patrol vessels are far from the seaside, choosing satellites as a relay station.
- *Organizations layer:* In this layer, organizations (MC, MB, RC, etc.) are responsible for doing maritime management according to corresponding maritime information from relay stations.

This structure has two practical downsides: one is that buoys need to be picked up and reset regularly by patrol vessels, and the other is that relay stations need to forward different information to the corresponding organizations. These two downsides make it difficult for these organizations to do joint maritime management.

In addition, with the increasing of marine density and navigational complexity, IMTS's communication security and efficiency problems are becoming more and more prominent [15], [16]. The IMTS often uses digital signature schemes to ensure the non-repudiation of information senders [17]. But there are no effective protection measures for the sensitive data (rescue information, ship locations, etc.). When this data is eavesdropped by criminals, ships in IMTS will risk a hostage situation. Meanwhile, if the IMTS adds encryption schemes to protect sensitive data, there will be low efficiency.

The aggregate signcryption (ASC) schemes can balance the communication security and efficiency [18]. They could generate a signature and encrypt a message in one logical process and support aggregation. So far, many researchers have proposed ASC schemes [19]–[27]. However, when these ASC schemes are applied in IMTS, the following challenges are not addressed:

- *Effectiveness:* Current ASC schemes often use bilinear pairing, which needs large computation cost in IMTS.
- *Validation:* The receiver can not verify the validity of devices. If the device in IMTS is invalid, the collected information may be inaccurate, affecting organizations' management.
- *Computational complexity:* The receiver usually needs to calculate all the values before deciding whether to receive the corresponding plaintext message.

Then, Blockchain comes into view. It is a distributed ledger technology relying on smart contracts and other logical control functions to evolve into a complete storage system. Our goal is introducing blockchain [28]–[30] to resolve these challenges. In IMTS, the patrol vessels and organizations can be nodes of the blockchain to strengthen management capacity because of the following characteristics [31] of blockchain:

- *Decentralized:* It can ensure nodes on the blockchain (the patrol vessels, organizations, etc.) jointly maintain the data in real time.
- *Openness:* It can ensure patrol vessels and organizations in IMTS query data stored on the blockchain and develop related management.
- *Independence:* It can ensure patrol vessels and organizations in IMTS exchange data on the blockchain without any human intervention.

This work is motivated by the privacy-preserving and efficiency challenges of the IMTS and addresses the core problem of "could we build a secure and efficient messages transmission channel in IMTS?"

A. Contributions

Our contributions are mainly described in the following three aspects:

1) We design a more practical "Perception-Network-Application" IMTS (PNA-IMTS) network structure. Based on this structure, we propose an efficient ID-based ASC scheme with blockchain (B-ID-ASC) to ensure IMTS's communication security and efficiency. Moreover, we use the predictive calculation method to reduce the computation cost for IoT devices in our proposed scheme.
2) We introduce blockchain to improve the performance and security of B-ID-ASC scheme for PNA-IMTS. Blockchain plays three specific roles in our proposed scheme. First, verifying the validity of IoT devices ensuring that signcryption generated by invalid devices can not be forwarded. Second, making a preliminary verification on the signcryption, and receivers will start to decrypt the

signcryption only after the primary verification is correct. Third, reducing the communication cost with shortening signcryption size.

3) Through security analysis and performance analysis, it demonstrates that our proposed scheme is more suitable for PNA-IMTS than existing ID-ASC schemes.

B. Organization

Section II discusses some related work on ASC schemes. Section III shows some basic concepts and security model in ID-ASC schemes. Section IV gives the description of our B-ID-ASC scheme. Section V analyzes the security of our B-ID-ASC scheme and Section VI compares our B-ID-ASC scheme with current ID-ASC schemes in performance. Section VII concludes what we have done.

II. Related Work

In public key cryptography, encryption can ensure the confidentiality of information and the digital signature [32] can ensure messages integrity, non-repudiation, authentication. In order to satisfy some applications needing both encryption and digital signature, the signcryption scheme, a vital cryptographic mechanism, was introduced by Zheng [18]. The core of signcryption scheme is a logical process to realize two functions (digital signatures and secure encryption). Its efficiency is higher than the traditional method of sign-then-encrypt. Recently, Identity(ID)-based cryptography [33]–[35] was studied widely because of its advantage of solving the certificate management problem and a lot of ID-based signcryption schemes [19]–[27], [36], [37] had been proposed.

Lee's scheme [36] is the first ID-based signcryption scheme defining the indistinguishability under chosen ciphertext attack (IND-ISC-CCA) security and existential unforgeability under adaptive chosen message attack (EF-ISC-ACMA) security in every ID-based signcryption scheme. Then an ID-based signcryption scheme having strong forward-security was published by Muñiz and Laud [19]. The first ID-based aggregate signcryption scheme, having an advantage on full aggregation, was proposed by Gentry and Ramzan [20]. Then the analysis of current aggregate signature schemes by Selvi *et al.* [21], [22] were presented. It shows that these schemes, having limited efficiency, can not be used well in practical applications. Basing on these analysis, Selvi *et al.* [21], [22] also proposed a novel identity-based aggregate signature schemes. But Selvi *et al.*'s schemes [21], [22] lacked detailed security proof. Then many ID-based aggregate signcryption schemes having complete security analysis and performance analysis [23], [24] were proposed. However, these schemes usually have some weakesses on the performance.

In such a scenario, an efficient ID-based aggregate signcryption scheme without pairing was proposed by Abouelkheir and El-Sherbiny [27]. This scheme used an elliptic curve instead of bilinear pairing to improve efficiency. Unfortunately, although the authors showed detailed security analysis of their scheme, we find Abouelkheir and Elsherbiny's scheme [27] fails to protect the master secret key and ensure confidentiality [38]. Then, many researchers [39], [40] presented multi-receiver signcryption Schemes. However, these schemes are not suitable well for IMTS.

III. Preliminaries

We introduce basic concepts including bilinear pairing, hard problems, definitions and security model in this section.

A. Bilinear Pairing

There is a mapping e on G_1, G_2, G_T, which is $e : G_1 \times G_2 \rightarrow G_T$. The G_1, G_2 are two additive cyclic group and the G_T is a multiplicative cyclic group. If $G_1 = G_2$, the bilinear pairing is symmetrical. Otherwise, it's asymmetric. This mapping e has three features:

- *Bilinearity:* For any $X \in G_1$, $Y \in G_2$ and $x, y \in Z_p$, the equation $e(xX, yY) = e(X, Y)^{xy}$ holds.
- *No Degeneracy:* The equation $e(X, Y) \neq 1_{G_T}$ holds for $X \in G_1$, $Y \in G_2$.
- *Computability:* $Y \in G_2$, $e(X, Y)$ is calculable for any $X \in G_1$.

B. Bilinear Diffie-Hellman Problem (BDHP)

Giving $(P, aP, bP, cP) \in G$, at the same time, unknowing $a, b, c \in Z_q^*$, it is hard to compute $e(P, P)^{abc}$.

C. Modified Diffie-Hellman Problem (MBDHP)

Giving $(P, aP, bP) \in G$, at the same time, unknowing $a, b \in Z_q^*$, it is hard to compute $e(P, P)^{a^2 b}$.

D. Security Model

A normal ID-ASC scheme has three entities: 1) the Private Key Generator (PKG); 2) the sender and 3) receiver. There are five algorithms: **Setup** (system setup algorithm), **KeyExtract** (key generating algorithm), **Signcryption** (signcryption generating algorithm), **Aggregation** (signcryption aggregating algorithm), **Unsigncryption** (unsigncryption algorithm):

1) *$(par, msk) \leftarrow$ Setup(λ):* Inputting a security parameter λ, the PKG generates the system parameter *par*, the master secret key *msk*, the master public key *mpk*. Then the PKG publish *par*, *mpk* and stores *msk* in secret.
2) *$(s_{ID}) \leftarrow$ KeyExtract(msk, ID):* Inputting *msk* and the user's identity *ID*, the PKG generates user's private key (s_{ID}) and sends it to the user in a secure channel.
3) *$(\sigma_A) \leftarrow$ Signcryption(M, s_{ID_A}, ID_B):* Inputting a message *M*, the sender's private key s_{ID_A} and the receiver's identity ID_B, the sender *A* generates the signcryption σ_A and sends it to the receiver *B*.
4) *$(\sigma_{agg}) \leftarrow$ Aggregation($ID_B, \{\sigma_i\}_{i=1,\ldots,n}$):* Inputting a set of $(ID_B, \{\sigma_i\}_{i=1,\ldots,n})$, a member in system outputs the aggregating signcryption σ_{agg}.
5) *$(M/\perp) \leftarrow$ Unsigncryption($\sigma_{agg}, s_{ID_B}, \{ID_i\}_{i=1,\ldots,n}$):* Inputting the signcryption σ, the receiver's private key s_{ID_B} and the senders' identities $\{ID_i\}_{i=1,\ldots,n}$, the receiver *B* unsigncrypt to get $M = \{M_i\}_{i=1,\ldots,n}$ or $\perp$ (indicating that σ is invalid).

In a secure ID-ASC scheme, only the specified receiver can get the real message and the sender cannot deny the act of

sending the signcryption. That means a secure ID-ASC scheme need have the properties confidentiality and unforgeability. There are 2 games to formalize confidentiality (*IND-IBAS-CCA2-Game*) and unforgeability (*EUF-IBAS-CMA-Game*) of the ID-ASC scheme. In these two games, the adversary $\mathcal{A}$ initiates some oracle queries to the challenger $\mathcal{C}$ and acquires the corresponding responses.

The entire processes of *IND-IBAS-CCA2-Game* are as below.

- *Setup:* $\mathcal{C}$ generates master secret key s, public parameters Par. Then $\mathcal{C}$ stores s in secret and sends Par to $\mathcal{A}$.
- *Phase 1:* $\mathcal{C}$ chooses target identities (senders' identities $\{ID_i\}_{i=1,\ldots,n}$, a receivers' identity ID_B) and sends them to $\mathcal{A}$.
- *Phase 2:* $\mathcal{A}$ adaptively makes the oracle queries, but $\mathcal{A}$ can not queries to get ID_B's private key. Then $\mathcal{C}$ gives the corresponding responses. The basic oracle queries are described as follows:
  - *Key Extraction Query:* $\mathcal{A}$ inputs the identity *ID*. Then $\mathcal{A}$ gets its private key $s_{ID} = KeyExtract(ID)$.
  - *Signcryption Query:* $\mathcal{A}$ inputs senders' identity ID_i, a receivers' identity ID_B and message M_i. Then $\mathcal{C}$ computes $s_{ID_i} = KeyExtract(ID_i)$, $\sigma_i = Signcryption(M_i, s_{ID_i}, ID_B)$ and sends σ_i to $\mathcal{A}$.
  - *Aggregate Signcryption Query:* $\mathcal{A}$ inputs senders' identities $\{ID_i\}_{i=1,\ldots,n}$, a receivers' identity ID_B and messages $\{M_i\}_{i=1,\ldots,n}$. Then $\mathcal{C}$ computes: $s_{ID_i} = KeyExtract(ID_i)$ for $i = 1$ *to* n, $\sigma_{agg} = Aggregate(\{M_i, s_{ID_i}\}_{i=1,\ldots,n}, ID_B)$, and sends σ_{agg} to $\mathcal{A}$.
  - *Unsigncryption Query:* $\mathcal{A}$ inputs (σ_{agg}, ID_B). Then $\mathcal{C}$ computes: $s_{ID_B} = KeyExtract(ID_B)$, $Result = Unsigncryption(\sigma_{agg}, s_{ID_B}, ID)$, and sends *Result* to $\mathcal{A}$.
- *Challenge:* $\mathcal{A}$ chooses some challenge messages $\{[(M_0)_i, (M_1)_i, ID_i]_{i=1,\ldots,n}, ID_B\}$ to $\mathcal{C}$. Then $\mathcal{C}$ randomly chooses the $b \in \{0,1\}$ and signcrypts $\{(m_b)_i\}_{i=1,\ldots,n}$ using the private keys of senders and the receiver. At the last, $\mathcal{C}$ returns the aggregate signcryption σ^*_{agg} to $\mathcal{A}$.
- *Phase 3:* $\mathcal{A}$ adaptively makes some oracle queries, but $\mathcal{A}$ cannot initiate the unsigncryption query with σ^*_{agg}.
- *Guess:* $\mathcal{A}$ outputs a series of guess bit $b' \in \{0,1\}$. If b' and c are equal, $\mathcal{A}$ wins the game.
- *Probability Advantage:* $Adv^{\mathsf{IND-CCA2}}_{\mathsf{ID-ASC}}(\mathcal{A}) = |Pr[b' = b] - 1/2|$

The entire processes of *EUF-IBAS-CMA-Game* are as below.

- *Setup:* $\mathcal{C}$ generates master secret key s, public parameters Par. Then $\mathcal{C}$ stores s in secret and sends Par to $\mathcal{A}$.
- *Query:* $\mathcal{A}$ adaptively makes oracle queries, and $\mathcal{C}$ gives the corresponding responses.
- *Forgery:* $\mathcal{A}$ outputs a forged signature σ^*_{agg}.If σ^*_{agg} can be unsigncrypted correctly by any receiver identity, $\mathcal{A}$ wins the game.
- *Probability Advantage:* $Adv^{\mathsf{EUF-CMA}}_{\mathsf{ID-ASC}}(\mathcal{A}) = Pr[\mathsf{Unsigncrypt}(\sigma^*) \neq \perp]$

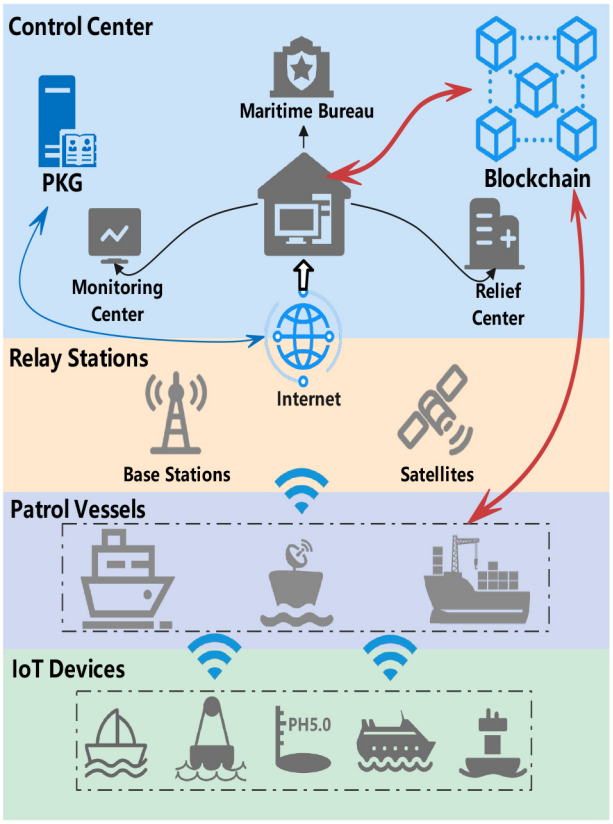

Fig. 2. The Network Model of Our Designed "Perception-Network-Application" IMTS.

Based on above security definitions, if the probability advantage of any polynomial-time (PPT) adversary wining *IND-IBAS-CCA2-Game* and *EUF-IBAS-CMA-Game* is negligible, the ID-ASC scheme is secure. Denoting ϵ is a negligible probability, the following inequalities hold in a secure ID-ASC scheme:

- $Adv^{\mathsf{IND-CCA2}}_{\mathsf{ID-ASC}}(\mathcal{A}) \leq \epsilon$,
- $Adv^{\mathsf{EUF-CMA}}_{\mathsf{ID-ASC}}(\mathcal{A}) \leq \epsilon$.

IV. The B-ID-ASC Scheme for PNA-IMTS

We detailed show the network model of PNA-IMTS and our proposed B-ID-ASC scheme in this section. It can demonstrate the applicability of our proposed B-ID-ASC scheme for PNA-IMTS.

A. Network Model

We introduce the blockchain into IMTS to form an improved PNA-IMTS. The network model is as shown in Fig. 2. There are six roles in PNA-IMTS: Private Key Generator (PKG), Blockchain, Control Center (CC), Relay Station (RS), Patrol Vessel (PV), IoT Device (DE). In Fig. 2, the red lines indicate CC and PV have the ability to access the blockchain. The blue lines among RS, PV and DE indicate they communicate through wireless networks. Comparing with the traditional

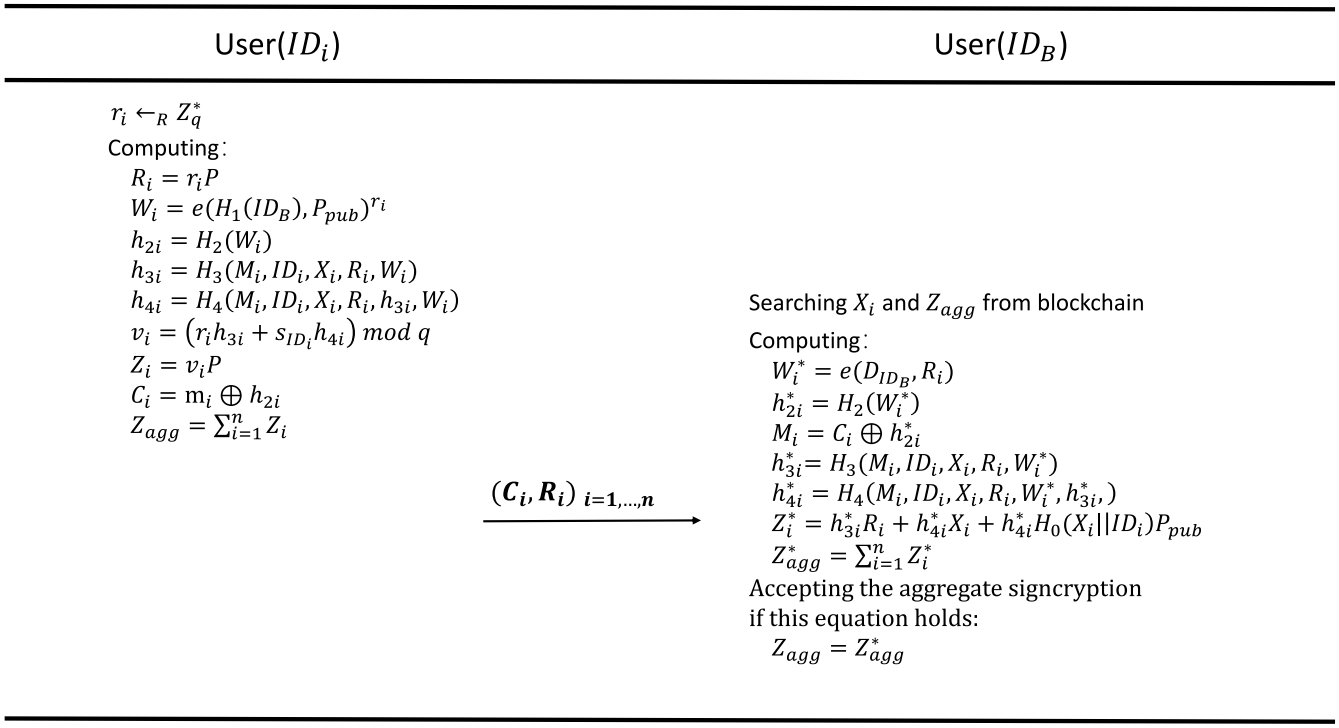

Fig. 3. Generating Signcryption.

IMTS, it not only adds blockchain, but also enriches the functions of each role. Their specific differences are described as follows.

- *PKG:* Generating private keys of signcrypting for DE, CC, and organizations in CC (MC, MB and RC). And PKG is trusted.
- *Blockchain:* Enhancing the security and shortening the signcryption size. There are some public information of DE, part of signcryption making a preliminary verification on the blockchain. Thus, through the blockchain, PV can verify the validity of DE to ensure that signcryption generated by invalid devices can not be forwarded. And CC can make a preliminary verification on the signcryption. CC decrypts the signcryption only after the preliminary verification is correct.
- *CC:* Carrying out the joint management and overall arrangement for MC, MB and RC according to received maritime messages. Usually, CC has great computing and storage power. After the primary verification passed, CC does different operations according to different receivers. If the receiver in signcryption is CC , CC unsigncrypts these signcryption. If the receiver in signcryption is MC, MB or RC, CC sends signcryption to the receiver. This method can not only ensure that CC can carry out normal joint management, but also ensure that MC, MB and RC can get maritime information separately when necessary. And CC is semi-honest.
- *RS:* Forwarding all information from PV to the CC without judging. This method can speed up the safe flow of maritime information. And RS is semi-honest.
- *PV:* Verifying the validity of DE through blockchain and aggregates signcryption from DE. This method can increase efficiency in B-ID-ASC. And RS is semi-honest.
- *DE:* Every DE (boat or buoy) has wireless sensors to collect maritime information and send them to PV through the wireless network. And DE may be malicious.

Our designed PNA-IMT have two advantages. The one is that PNA-IMT can ensure the security and timeliness of the marine information collection and feedback through the full deployment of DE, B-ID-ASC scheme, and blockchain. The other one is that in PNA-IMT, CC obtains information collected by DE, and then sends relevant information and instructions to the corresponding departments (MC, MB, RC, etc.), which is conducive to the unified management of IMT and promotes the cooperation among departments.

B. Our Proposed Scheme

Before introducing our proposed scheme, some preliminaries in our B-ID-ASC scheme are presented in Table I for better understanding.

Then, we describe our B-ID-ASC scheme. It is about the initialization phase, extraction and uploading phase, signcryption phase, aggregation phase, unsigncryption phase. The flow chart of generating signcryption is shown in Fig. 3. It is worth noting that in B-ID-ASC scheme, before being put into use, every DE stores the value of bilinear pairing $W = e(H_1(ID_B), P_{pub})$, where ID_B is CC's identity or corresponding organization's identity. This method can reduce the computation cost when generating signcryption.

1) Initialization: In this phase, PKG generates the system parameters and sends them into each users in PNA-IMTS. The specific operations are as follows.

TABLE I
NOTATIONS AND THEIR DESCRIPTIONS

Notations	Descriptions
p, q	two large prime numbers
E	an elliptic curve $y^2 = x^3 + ax + b \ mod \ p$
P	a random non-zero base point in E
$\mathbb{G}$	an additive group having the order q, all points which are on the elliptic curve E over F_p
$\mathbb{G}_\mathrm{T}$	an multiplicative group having the order q, all points which are on the elliptic curve E over F_p
e	a bilinear pairing, $\mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_\mathrm{T}$
H_0	hash function, $\mathbb{G} \times \{0,1\}^* \rightarrow Z_q^*$
H_1	hash function, $\{0,1\}^* \rightarrow \mathbb{G}$
H_2	hash function, $\mathbb{G} \rightarrow Z_q^*$
H_3	hash function, $\{0,1\}^* \times \{0,1\}^* \times \mathbb{G} \times \mathbb{G} \times \mathbb{G}_\mathrm{T} \rightarrow Z_q^*$
H_4	hash function, $\{0,1\}^* \times \{0,1\}^* \times \mathbb{G} \times \mathbb{G} \times \mathbb{G}_\mathrm{T} \times Z_q^* \rightarrow Z_q^*$
s	master secret key of PKG
P_{pub}	master public key of PKG, $P_{pub} = sP_{pub}$
M_i	the message
σ_i	the sincryption
σ_{agg}	the aggregate sincryption
Z_{agg}	aggregate $\sum_{i=1}^{n} Z_i$

- Choose large prime number p, q and a non-singular elliptic curve E defined over F_p.
- Choose a generator P with order q of the elliptic curve point group $\mathbb{G}$.
- Select hash functions $\{H_0, H_1, H_2, H_3, H_4\}$ as defined in table I.
- Choose a random number $s \in Z_q^*$ as its master secret key, and computes the $P_{pub} = sP$ as its master public key.

Next, PKG sends system parameters:

$$\left\{q, \mathbb{G}, \mathbb{G}_\mathrm{T}, e, P, H_0, H_1, H_2, H_3, H_4, P_{pub}\right\}$$

to all users.

2) Extraction and Uploading: In this phase, PKG generates private keys for users. When the user is DE, PKG pre-load private key into DE's tamper-proof device. When the user is CC or an organization in CC, PKG sends private key to this user through a secure channel. Taking an user's identity ID_i as an example, the PKG executes following operations.

- Randomly choose $x_i \in Z_q^*$,
- Compute:

$$\begin{aligned} X_i &= x_i P, \\ h_{0i} &= H_0(X_i || ID_i), \\ h_{1i} &= H_1(ID_i), \\ s_{ID_i} &= x_i + sh_{0i}, \\ D_{ID_i} &= sh_{1i}. \end{aligned}$$

PKG sets $T_i \in \{0,1\}^*$ as this user's period of validity and sends (ID_i, X_i, T_i) to blockchain through smart contract. Then, PKG sends $(s_{ID_i}, X_i, D_{ID_i})$ to the user. Next, PKG gets the identity of CC (ID_C), MC (ID_{MC}), MB (ID_{MB}), RC (ID_{RC}) and executes following operations.

- Compute:

$$\begin{aligned} h_C &= H_1(ID_C), \\ h_{MC} &= H_1(ID_{MC}), \\ h_{MB} &= H_1(ID_{MB}), \\ h_{RC} &= H_1(ID_{RC}), \end{aligned}$$

- Compute:

$$\begin{aligned} W_C &= e(h_C, P_{pub}), \\ W_{MC} &= e(h_{MC}, P_{pub}), \\ W_{MB} &= e(h_{MB}, P_{pub}), \\ W_{RC} &= e(h_{RC}, P_{pub}). \end{aligned}$$

Next, PKG pre-loads W_C into all DE. After that, PKG pre-loads W_C into DE if it belongs to MC, W_{MB} into DE if it belongs to MB, or W_{RC} into DE if it belongs to RC.

3) Signcryption: In this phase, the DE with an identity ID_i generates signcryption for the message M_i and the $ID_B \in \{ID_C, ID_{MC}, ID_{MB}, ID_{RC}\}$ is a receiver. The specific operations are as follows.

- Randomly choose $r_i \in Z_q^*$.
- Compute:

$$\begin{aligned} R_i &= r_i P, \\ W_i &= (W_B)^{r_i}, \\ h_{2i} &= H_2(W_i), \\ h_{3i} &= H_3(M_i, ID_i, X_i, R_i, W_i), \\ h_{4i} &= H_4(M_i, ID_i, X_i, R_i, W_i, h_{3i}), \\ v_i &= (r_i h_{3i} + s_{ID_i} h_{4i}) \mod q, \\ Z_i &= v_i P, \\ C_i &= M_i \oplus h_{2i}, \\ \sigma_i &= (R_i, Z_i, C_i). \end{aligned}$$

Then, DE (ID_i) sends signcryption σ_i to PV.

4) Aggregation: In this phase, PV aggregates signcryption. First, PV receives signcryption from a set of DE with identities $\{ID_i\}_{i=1,\ldots,n}$ and searches $\{ID_i, T_i\}_{i=1,\ldots,n}$ on blockchain. If they are existing and within the period of validity, PV aggregates this set of signcryption from $\{ID_i\}_{i=1,\ldots,n}$ for ID_B as follows.

- compute:

$$\begin{aligned} Z_{agg} &= \sum_{i=1}^{n} Z_i, \\ \sigma_{agg} &= (\sigma_{agg1}, \sigma_{agg2}) = (\{C_i\}_{i=1,\ldots,n}, \{R_i\}_{i=1,\ldots,n}). \end{aligned}$$

Then PV sends $(\sigma_{agg1}, \sigma_{agg2})$ to RS and sends Z_{agg} to blockchain through the smart contract. Next, RS honestly forwards $(\sigma_{agg1}, \sigma_{agg2})$ to CC.

5) Unsigncryption: In this phase, the receiver $ID_B \in \{ID_C, ID_{MC}, ID_{MB}, ID_{RC}\}$ unsigncrypt to get messages $\{M_i\}_{i=1,\ldots,n}$.

First, CC queries on the blockchain to get Z_{agg} and $\{X_i\}_{i=1,\ldots,n}$ according to $\{ID_i\}_{i=1,\ldots,n}$. If they are on blockchain, CC passes the preliminary verification and executes the following operations. Otherwise, CC rejects these signcryption.

Second, if the receiver ID_B is ID_C, CC decrypts and verifies the aggregate signcryption. If the receiver ID_B is ID_{MC}, ID_{MB} or ID_{RC}, CC stores and sends the aggregate signcryption to MC, MB or RC.

Then ID_B unsigncrypts and verifies the aggregate signcryption as follows.

- Compute:

$$\begin{aligned} W_i^* &= e(D_{ID_B}, R_i), \\ h_{2i}^* &= H_2(W_i^*), \\ h_{3i}^* &= H_3(M_i, ID_i, X_i, R_i, W_i^*), \\ h_{4i}^* &= H_4(M_i, ID_i, X_i, R_i, W_i^*, h_{3i}^*), \\ Z_i^* &= h_{3i}^* R_i + h_{4i}^* X_i + h_{4i}^* h_{0i} P_{pub}, \\ Z_{agg}^* &= \sum_{i=1}^{n} Z_i^*. \end{aligned}$$

- Check the correctness of equation $Z_{agg}^* = Z_{agg}$. If this equation holds, the receiver ID_B will accecpts the signcryption and computes $M_i = C_i \oplus h_{2i}^*$ to get all messages.

6) Correctness: If the phases described above are carried out honestly, the following equations hold.

$$\begin{aligned} W_i &= e\big(H_1(ID_B), P_{pub}\big)^{r_i} \\ &= e(sH_1(ID_B), r_i P) \\ &= e\big(D_{ID_B}, R_i\big), \\ Z_i &= v_i P \\ &= \big(r_i h_{3i} + s_{ID_i} h_{4i}\big) P \\ &= h_{3i} R_i + (x_i + sH_0(X_i || ID_i)) h_{4i}) P \\ &= h_{3i} R_i + h_{4i} X_i + h_{4i} H_0(X_i || ID_i) P_{pub}. \end{aligned}$$

In usigncryption phase, the receiver:

$$ID_B \in \{ID_C, ID_{MC}, ID_{MB}, ID_{RC}\}$$

computes:

$$\begin{aligned} W_i^* &= e\big(D_{ID_B}, R_i\big), \\ Z_i^* &= h_{3i}^* R_i + h_{4i}^* X_i + h_{4i}^* h_{0i} P_{pub}. \end{aligned}$$

Namely, $W_i^* = W_i, Z_i^* = Z_i$. That is, as long as the generation method of signcryption is correct, the receiver can successfully verify the signcryption and get the message. Thus, our proposed B-ID-ASC scheme is correct.

V. Security Analysis

We prove the B-ID-ASC scheme is secure through proving following theorems in this section. We take generating a single signcryption as a pointcut, and extend to prove the security of B-ID-ASC scheme. There are five hash oracle queries ($H_0 - Query$, $H_1 - Query$, $H_2 - Query$, $H_3 - Query$, $H_4 - Query$) with five lists (L_0, L_1, L_2, L_3, L_4) in our security proof.

- $H_0 - Query$: $\mathcal{A}$ inputs (X_i, ID_i). If $(X_i, ID_i, (y_0)_i)$ is not in L_0, $\mathcal{C}$ chooses a random number $(y_0)_i \in Z_q^*$ and stores it in L_0. Otherwise, $\mathcal{C}$ returns $(y_0)_i$ to $\mathcal{A}$.
- $H_1 - Query$: $\mathcal{A}$ inputs (ID_i). If $(ID_i, (y_1)_i)$ is not in L_1, $\mathcal{C}$ chooses a random number $(y_1)_i \in G$ and stores it in L_1. Otherwise, $\mathcal{C}$ returns $(y_1)_i$ to $\mathcal{A}$.
- $H_2 - Query$: $\mathcal{A}$ inputs (W_i). If $(W_i, (y_2)_i)$ is not in L_2, $\mathcal{C}$ chooses a random number $(y_2)_i \in Z_q^*$ and stores it in L_2. Otherwise, $\mathcal{C}$ returns $(y_2)_i$ to $\mathcal{A}$.
- $H_3 - Query$: $\mathcal{A}$ inputs $(M_i, ID_i, X_i, R_i, W_i)$. If $(M_i, ID_i, X_i, R_i, W_i, (y_3)_i)$ is not in L_3, $\mathcal{C}$ chooses a random number $(y_3)_i \in Z_q^*$ and stores it in L_3. Otherwise, $\mathcal{C}$ returns $(y_3)_i$ to $\mathcal{A}$.
- $H_4 - Query$: $\mathcal{A}$ inputs $(M_i, ID_i, X_i, R_i, W_i, (y_3)_i)$. If $(M_i, ID_i, X_i, R_i, W_i, (y_3)_i, (y_4)_i)$ is not in L_4, $\mathcal{C}$ chooses a random number $(y_4)_i \in Z_q^*$ and stores it in L_4. Otherwise, $\mathcal{C}$ returns $(y_4)_i$ to $\mathcal{A}$.

Theorem 1: If a PPT adversary $\mathcal{A}$ can win the IND-IBAS-CCA2-Game in a non-neligible advantage ϵ, the challenger $\mathcal{C}$ can solve the Bilinear Diffie-Hellman Problem (BDHP) in a non-neligible advantage ϵ'.

Proof: Giving a BDHP instance $(P, aP, bP, cP) \in G$, $\mathcal{C}$ solve the BDHP by interaction with $\mathcal{A}$.

- *Setup:* $\mathcal{C}$ generates public parameters and sets $P_{pub} = aP$. Then sending *Par* to $\mathcal{A}$.
- *Phase 1:* $\mathcal{C}$ chooses target identities$\{ID_i\}_{i=1,\ldots,n}$ (senders), ID_B^* (reciever) and $k \in [1, n]$. Then $\mathcal{A}$ takes hash oracle queries. When $ID = ID_B^*$ in $H_1 - Query$, $\mathcal{C}$ randomly chooses $\xi_i \in Z_q^*$ and set $(y_1)_i = \xi_i bP$. Next, $\mathcal{C}$ updates the list L_1.
- *Phase 2:*
  1) Key-extract-query. Inputting ID_i, if ID_i is not the target identity, $\mathcal{C}$ returns $X_i = x_i P$, where x_i is from L_0 . Otherwise, $\mathcal{C}$ returns "failure".
  2) Signcryption-query. Inputting sender's identity ID_i, reciever's identity ID_B and message M_i, $\mathcal{C}$ searches them in L_U and signcrypts the message:
     – If ID_i is the target identity, $\mathcal{C}$ get $X_i, (y_0)_i$ from L_0 and randomly chooses $(y_3)_i, (y_4)_i, z_i \in Z_P^*$ and computes:

$$\begin{aligned} R_i &= (y_3)_i^{-1}(z_i P - (y_4)_i (X_i) + (y_0)_i P_{pub}) \\ W_i &= e(R_i, D_{ID_B}). \end{aligned}$$

     $\mathcal{C}$ get $(y_2)_i$ from L_2 and computes:

$$C_i = (M_i || ID_i) \oplus (y_2)_i.$$

     Then, $\mathcal{C}$ outputs $\sigma_i = (X_i, C_i, R_i, Z_i)$ and store them in the list.
     – Otherwise, $\mathcal{C}$ performs the normal signcryption algorithm to return σ_i.
  3) Aggregate signcryption-query. Inputting senders' identities $\{ID_i\}_{i=1,\ldots,n}$, a receivers' identity ID_B and messages $\{M_i\}_{i=1,\ldots,n}$. For every i, $\mathcal{C}$ does operations in signcryption-query. Then $\mathcal{C}$ computes $Z_{agg} = \sum_{i=1}^{n} Z_i$, and sets $\sigma_{agg} = (\{X_i, C_i, R_i\}_{i=1,\ldots,n}, Z_a gg)$. $\mathcal{C}$ outputs σ_{agg}.
  4) Unsigncryption-query. Inputting the senders' identities $\{ID_i\}_{i=1,\ldots,n}$, the receiver's identity ID_B and aggregate signcryption ciphertext σ_{agg}, $\mathcal{C}$ computes $M_i || ID_i = C_i \oplus (y_2)_i$. Then $\mathcal{C}$ return M_i to $\mathcal{A}$.

  $\mathcal{A}_1$ adaptively take some queries to $\mathcal{C}$ in this phase. $\mathcal{C}$ creates a empty list L_U and rensponds queries as follows.
- *Challenge:* $\mathcal{A}$ sends $\{[(M_0)_i, (M_1)_i, ID_i]_{i=1,\ldots,n}, ID_B\}$ to $\mathcal{C}$. $\mathcal{C}$ checks ID_B. If $ID_B = ID_B^*$, abort. Otherwise, $\mathcal{C}$ does the following operations for each i.

- If $i = k$, $\mathcal{C}$ randomly chooses $(y_3)_i$, $(y_4)_i \in Z_q^*$, $A_i \in G$ and sets:

$$\begin{aligned} R_i &= cP \\ (h_3)_i &= (y_3)_i \\ (h_4)_i &= (y_4)_i \\ (h_2)_i &= A_i \oplus (M_i||ID_i) \\ Z_i &= (h_3)_i R_i + (h_4)_i (X_i) + (h_0)_i P_p ub \end{aligned}$$

  Thus, the $\sigma_i = (X_i, A_i, R_i, Z_i)$
- Otherwise, $\mathcal{C}$ chooses a random number $b \in \{0, 1\}$. Then $\mathcal{C}$ signcrypts $(M_b)_i$ using the sender's private key and receiver's public key to generate the signcryption.

Then $\mathcal{C}$ aggregates all signcryption σ_i^* from $i = 1$ to $i = n$ and returns challenge aggregate signcryption $(\sigma_{agg}^*, Z_{agg}^*) = (\{C_i^*\}_{i=0,\ldots,n}, \{R_i^*\}_{i=0,\ldots,n}, Z_{agg}^*)$ to $\mathcal{A}$

- *Phase 3:* It is the same with the phase 2. But $\mathcal{A}_1$ can not take the unsigncryption query for $(\sigma_{agg}^*, Z_{agg}^*)$.
- *Guess:* $\mathcal{A}$ outputs guess bit $b' \in \{0, 1\}$ for each $i = 1$ to $i = n$.

If $\mathcal{A}$ can win the game in a non-negligible advantage ϵ, $\mathcal{C}$ can get W_k from list L_2, ξ_k from list L_U and outputs $W_k^{-\xi_k}$ as the solution of BDHP:

$$\begin{aligned} & W_k^{-\xi_k} \\ &= e(D_{ID_B}, R_k)^{-\xi_k} \\ &= e(\xi_k abP, cP)^{-\xi_k} \\ &= e(P, P)^{abc} \end{aligned}$$

Thus, $\mathcal{C}$ can solute BDHP with an advantage $\epsilon' = \epsilon$. ■

Theorem 2: If a PPT adversary $\mathcal{A}$ can win the EUF-IBAS-CMA-Game in a non-neligible advantage ϵ, the challenger $\mathcal{C}$ can solve the Bilinear Modified Diffie-Hellman Problem (MBDHP) in a non-neligible advantage ϵ'.

Proof: Giving a BDHP instance $(P, aP, bP) \in G$, $\mathcal{C}$ solve the BDHP by interaction with $\mathcal{A}_2$.

- *Setup:* $\mathcal{C}$ generates public parameters and sets $P_{pub} = aP$. Then sending *Par* to $\mathcal{A}$.
- *Query* $\mathcal{C}$ chooses target identities$\{ID_i\}_{i=1,\ldots,n}$ (senders), ID_B^* (reciever) and $k \in [1, n]$. Then $\mathcal{A}$ takes hash oracle queries. When $ID = ID_B^*$ in $H_1 - Query$, $\mathcal{C}$ randomly chooses $\xi_i \in Z_q^*$ and set $(y_1)_i = \xi_i bP$. Next, $\mathcal{C}$ updates the list L_1.
  1) Key-extract-query. Inputting ID_i, if ID_i is not the target identity, $\mathcal{C}$ returns $X_i = x_i P$, where x_i is from L_0 . Otherwise, $\mathcal{C}$ returns "failure".
  2) Signcryption-query. Inputting sender's identity ID_i, reciever's identity ID_B and message M_i, $\mathcal{C}$ searches them in L_U and signcrypts the message:
     – If ID_i is the target identity, $\mathcal{C}$ get $X_i, (y_0)_i$ from L_0 and randomly chooses$(y_3)_i, (y_4)_i, z_i \in Z_P^*$ and computes:

$$\begin{aligned} R_i &= (y_3)_i^{-1}\big(z_i P - (y_4)_i (X_i) + (y_0)_i P_{pub}\big) \\ W_i &= e\big(R_i, D_{ID_B}\big). \end{aligned}$$

     $\mathcal{C}$ get $(y_2)_i$ from L_2 and computes:

$$C_i = (M_i||ID_i) \oplus (y_2)_i.$$

     Then, $\mathcal{C}$ outputs $\sigma_i = (X_i, C_i, R_i, Z_i)$ and store them in the list.
     – Otherwise, $\mathcal{C}$ performs the normal signcryption algorithm to return σ_i.
  3) Aggregate signcryption-query. Inputting senders' identities $\{ID_i\}_{i=1,\ldots,n}$, a receivers' identity ID_B and messages $\{M_i\}_{i=1,\ldots,n}$. For every *i*, $\mathcal{C}$ does operations in signcryption-query. Then $\mathcal{C}$ computes $Z_{agg} = \sum_{i=1}^{n} Z_i$, and sets $\sigma_{agg} = (\{X_i, C_i, R_i\}_{i=1,\ldots,n}, Z_a gg)$. $\mathcal{C}$ outputs σ_{agg}.
  4) Unsigncryption-query. Inputting the senders' identities $\{ID_i\}_{i=1,\ldots,n}$, the receiver's identity ID_B and aggregate signcryption ciphertext σ_{agg}, $\mathcal{C}$ computes $M_i||ID_i = C_i \oplus (y_2)_i$. Then $\mathcal{C}$ return M_i to $\mathcal{A}$.
- *Forgery:* $\mathcal{C}$ outputs a forged signcryption $\sigma_{agg} = (\{X_i, C_i, R_i\}_{i=1,\ldots,n}, Z_a gg^*)$ on ID_i^*. If $\mathcal{A}$ wins, the equation: $Z_{agg}^* - \sum_{i=k+1}^{n} \sum_{i=1}^{n} (y_4)_i (X_i + (y_0)_i P_{pub}) = \sum_{i=1}^{n} r_i (y_3)_i + \sum_{i=1}^{k} (y_4)_i (X_i + (y_0)_i P_{pub})$ holds.

If $\mathcal{A}$ can win the game in a non-negligible advantage ϵ, $\mathcal{C}$ can outputs $e(D_i, P_{pub})^{-\xi_i}$ as the solution of MBDHP:

$$\begin{aligned} & e(D_i, P_{pub}) \\ &= e(\xi_i abP, aP)^{-\xi_i} \\ &= e(P, P)^{a^2 b} \end{aligned}$$

Thus, $\mathcal{C}$ can solute BDHP with an advantage $\epsilon' = \epsilon$. ■

1) *Confidentiality:* According to Theorem 1, there is no PPT adversary can win the IND-IBAS-CCA2-Game in a non-neligible advantage ϵ if the BDHP is hard. Therefore, only the receiver can get the message M_i by computing $M_i = C_i \oplus h_{2i}$. Thus, our proposed B-ID-ASC scheme provides confidentiality.
2) *Unforgeability:* According to Theorem 2, there is no PPT adversary $\mathcal{A}$ can win the EUF-IBAS-CMA-Game in a non-neligible advantage ϵ if the MBDHP is hard. Therefore, the receiver can check the validity and integrity of signcryption by verifying whether the equation $Z_{agg}^* = Z_{agg}$ holds. Thus, our proposed B-ID-ASC scheme provides unforgeability.
3) *Resisting against current attacks:* Our proposed B-ID-ASC scheme for PNA-IMTS could withstand impersonation attack, modification attack, man-in-the-middle attack.
   – *Impersonation attack:* According to Theorem 1 and Theorem 2, the attacker cannot generate a correct signcryption. CC can detect this attack by computing equation $Z_i = h_{3i} R_i + h_{4i} X_i + h_{4i} h_{0i} P_{pub}$ and checking the correctness of equation $Z_{agg}^* = Z_{agg}$. Thus, our proposed B-ID-ASC scheme can withstand impersonation attack.
   – *Man-in-the-middle attack:* According to above analysis of confidentiality and unforgeability, even if the attacker play a role of man-in-the-middle, he can not

Algorithm 1 Uploading

```
func (t *ABstore)
Set(ctx contractapi.TransactionContextInterface, A string, Aval int)
error
{
  err := ctx.GetStub().PutState(A, []byte(strconv.Itoa(Aval)))
  if err != nil
    {
      return err
    }
  return nil
}
```

TABLE II
COSTS OF THE SMART CONTRACTS

Operations	Time Cost(ms)	TPS(n/10^3ms)
Upload	0.06	1601.57
Query	0.03	2984.18
Delete	0.06	1615.53

get any plain text message. Thus, our proposed B-ID-ASC scheme can withstand the man-in-the-middle attack.

– *Modification attack:* According to Theorem 2, any modification of signcryption could be found by checking the equation $Z_i = h_{3i}R_i + h_{4i}X_i + h_{4i}h_{0i}P_{pub}$. Thus, our proposed B-ID-ASC scheme can withstand the modification attack.

VI. PERFORMANCE ANALYSIS

First, we describe how to implement the blockchain system for B-ID-ASC scheme and count the execution time of uploading, querying and deleting. Second, we compare the performance of our proposed B-ID-ASC shceme with current ID-ASC schemes.

A. Implementation of Blockchain System for B-ID-ASC Scheme

The PKG, PV and CC are nodes of blockchain in B-ID-ASC scheme. They have the ability to upload information to blockchain and have searching function, deleting function. Based on these requirements, we implement the blockchain system (Fabric 2.2.1) on Wuhan University cloud server having Quad-core CPU, 8G memory, 50G SSD. Deploying and invoking smart contracts relying on Go language and the core codes are shown as Algorithm 1-3. The user sends information to blockchain with Algorithm 1. The user searchs corresponding information with Algorithm 2. The user deletes information sended by himself with Algorithm 3.

After realizing functions of blockchain in B-ID-ASC shceme, we provide related statistics. As shown in the table II, the time to execute a transaction for uploading is 0.06*ms*. The time to execute a transaction for querying is 0.03*ms* and the time to execute a transaction for deleting is 0.06*ms*. In addition, we have measured the time of executing 10000 transactions simutaneously for uploading, querying and deleting in Fig.4. It indicates that operations on delegating can proceed quickly when multiple transcations are executed concurrently.

Algorithm 2 Querying

```
func (t *ABstore)
Query(ctx contractapi.TransactionContextInterface, A string) (string,
error)
{
  var err error
  % Getting the state from the ledger.
  Avalbytes, err := ctx.GetStub().
  GetState(A)
  if err != nil
    {
    jsonResp:="{\"Error\":\"Failed to get state for " + A + "\"}"
    return "", errors.New(jsonResp)
    }
  if Avalbytes == nil
    {
    jsonResp := "{\"Error\":\"Nil amount for " + A + "\"}"
    return "", errors.New(jsonResp)
    }
  jsonResp := "{\"Name\":\"" + A + "\",\"Amount\":\"" +
string(Avalbytes) + "\"}"
  fmt.Printf("Query Response: % s\n", jsonResp)
  return string(Avalbytes), nil
}
```

Algorithm 3 Deleting

```
func (t *ABstore)
Delete(ctx contractapi.TransactionContextInterface, A string) error
{
  % Deleting the state in ledger.
  err := ctx.GetStub().DelState(A)
  if err != nil
    {
      return fmt.Errorf("Failed to delete state")
    }
  return nil
}
```

Fig. 4. Execution Time on Blockchain.

For a blockchain system, the performance evaluation is measured by the number of transactions completed per second, also called Transaction Per Second (TPS). The higher TPS is, the better performance of the system will be. But at the same time, the server will be under greater pressure. The table II and Fig.5 describe TPSs for uploading, querying and deleting in our blockchain system. It indicates that our blockchain system can be suitable for PNA-IMTS. Furthermore, system performance can be improved by further optimizing cloud server configuration, which is what we can do in the future.

TABLE III
COMPARISION OF COMPUTATION COSTS (MS) AND COMMUNICATION COSTS (BITS)

Schemes	Signcryption	Unsigncryption	Ciphertext Length
[25]	$4nT_{exe-t} + nT_{bp}$ = 5.138n	$6nT_{bp} + nT_{mi}$ = 8.891n	$4n\|\mathbb{G}\| + n\|\mathbb{G}_T\|$ = 3072n
[26]	$4nT_{exe-t}$ = 3.66n	$2nT_{exe-t} + 2nT_{bp} + 2nT_{mi}$ = 4.832n	$2n\|\mathbb{G}\| + 2n\|\mathbb{G}_T\| + n\|Z_q\|$ = 3097n
[27]	$2nT_{sm} + nT_{pa} + 3nT_{pm} + 3nT_h$ = 0.451n	$nT_{pa} + 4nT_{pm} + 3nT_h$ = 0.594n	$(2n+1)\|\mathbb{G}\| + n\|Z_q\|$ = 1280n + 512
Proposed Approach	$2nT_{sm} + nT_{pa} + 2nT_{pm} + nT_{exp-t} + 3nT_h$ = 2.699n + 0.645	$nT_{sm} + 3nT_{pa} + 3nT_{pm} + nT_{bp} + 3nT_h$ = 2.848n	$n\|\mathbb{G}\| + n\|M\|$ = 768n

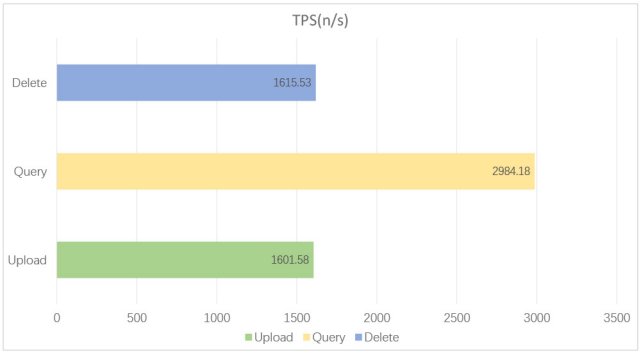

Fig. 5. TPS on Blockchain.

B. Performance of B-ID-ASC Shceme

We analyze the performance of our B-ID-ASC scheme for PNA-IMTS in this section. We carry out a comprehensive analysis through statistics data of computation cost and communication cost. In addition, we compare the performance of our B-ID-ASC scheme for PNA-IMTS with three current efficient ID-ASC schemes [25]–[27].

Initially, we set $\lambda = 128$ as the system secure parameter. Then, we use $e : G \times G \rightarrow G_T$ to reach the security level, where G is an additive group generated by a point P with the order q on the elliptic curve parameters "BN-256". p is a 256-bit prime number. q is a 256-bit prime number. Based on the secure length 128*bits*, we set $|Z_q| = 256bits$, $|G| = 512bits$, $|G_T| = 1024bits$.

We get the execution time of above cryptographic operations using RELIC library, a famous cryptographic library and having been widely used to implement cryptographic operations, to compare computation costs of related ID-ASC schemes. Our hardware platform are a notebook PC having Intel Core i7-8850U 1.80 GHz processor, 16GB RAM and Window 10 (64 bit) operating system. Defining the excution time of corresponding operations and make statistic. The results are as follows:

- T_{sm}: The scalar multiplication operation time in Z_q, 0.001*ms*.
- T_{pa}: The point addition operation time in G, 0.002*ms*.
- T_{pm}: The point multiplication operation time in G, 0.145*ms*.
- T_{mi}: The inversion operation time in Z_q, 0.023.
- T_{bp}: The bilinear pairing operation time, 1.478*ms*.
- T_{exp-t}: The modular exponentiation operation time in G_T, 0.915*ms*.
- T_h: The map to Z_q hash operation time, 0.004*ms*.
- T_H: The map to point hash operation time, 0.645*ms*.

Table III shows the comparison of computation costs and communication costs between our proposed B-ID-ASC scheme with current ID-ASC schemes.

On aspect of computation costs, for n number senders, our proposed B-ID-ASC scheme's computation cost in signcryption is $2nT_{sm} + nT_{pa} + 2nT_{pm} + nT_{exp-t} + 3nT_h = 1.239n$ and in unsigncryption is $nT_{sm} + 3nT_{pa} + 3nT_{pm} + nT_{bp} + 3nT_h = 1.941n$. Li *et al.*'s scheme [25] in signcryption is 5.138n and in unsigncryption is 8.891n. Karati *et al.*'s scheme [26] in signcryption is 3.66n and in unsigncryption is 4.832n. Abouelkheir and El-Sherbiny's scheme [27] in signcryption is 0.451n and in unsigncryption is 0.594n.

According to the above computation costs, it shows that our proposed scheme's computation cost is much lower than Li *et al.* [25] and Karati *et al.* [26] in terms of both signcryption and unsigncryption processes. The computation cost of Abouelkheir and Elsherbiny's scheme [27] is slightly lower than ours, but their scheme has defects on the master secret key protection and confidentiality protection (showing in the appendices)and is not secure enough. In practical application, the scheme performance can not be improved at the expense of security.

On aspect of communication costs, for n number senders, our proposed B-ID-ASC scheme's communication cost is $n|\mathbb{G}| + n|M| = 768n - bit$. Li *et al.*'s scheme [25] is $3072n - bit$. Karati *et al.*'s scheme [26] is $3097 - bit$. And Abouelkheir and El-sherbiny's scheme [27] is $(1280n + 512) - bit$. It is clear that our B-ID-ASC scheme's communication cost is the lowest of all presented ID-ASC schemes.

Due to the introduction of blockchain, the communication costs of our B-ID-ASC scheme are significantly reduced, and its security is greatly increased. Fig. 6 describes how the communication costs of our B-ID-ASC scheme and other current ID-ASC schemes changes in the different number of senders. It is evident that as the number of senders increases, the communication cost advantage of our B-ID-ASC scheme increases.

According to the above analysis, it could be concluded that our B-ID-ASC scheme has advantages in comprehensive performance and is more suitable for PNA-IMTS.

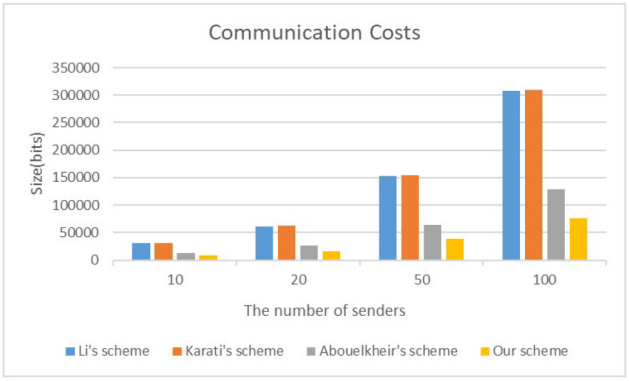

Fig. 6. Communication Costs Comparison.

VII. Conclusion

In order to better joint management for maritime transportation, we design a fusing network model PNA-IMTS and propose an efficient B-ID-ASC scheme with blockchain. We introduce blockchain to improve security and reduce communication costs. Finally, compared with current ID-ASC schemes in security, computation costs, and communication costs, the results show that our scheme is suitable for PNA-IMTS. In the future, we will design a novel and secure ID-ASC scheme without using bilinear pairing. Furthermore, we will study more possibilities of joint management for maritime transportation.

References

[1] M. Christiansen, K. Fagerholt, B. Nygreen, and D. Ronen, "Maritime transportation," in *Handbooks in Operations Research and Management Science*, vol. 14. Amsterdam, The Netherlands: Elsevier, 2007, pp. 189–284.

[2] Ø. Berle, J. B. Rice Jr., and B. E. Asbjørnslett, "Failure modes in the maritime transportation system: A functional approach to throughput vulnerability," *Maritime Policy Manag.*, vol. 38, no. 6, pp. 605–632, 2011.

[3] A. A. Shah, N. A. Bhatti, K. Dev, and B. S. Chowdhry, "MUHAFIZ: IoT-based track recording vehicle for the damage analysis of the railway track," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 9397–9406, Jun. 2021.

[4] I. Stellios, P. Kotzanikolaou, M. Psarakis, C. Alcaraz, and J. Lopez, "A survey of IoT-enabled cyberattacks: Assessing attack paths to critical infrastructures and services," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3453–3495, 4th Quart., 2018.

[5] A. Munusamy *et al.*, "Edge-centric secure service provisioning in IoT-enabled maritime transportation systems," *IEEE Trans. Intell. Transp. Syst.*, early access, Aug. 21, 2021, doi: 10.1109/TITS.2021.3102957.

[6] K. Salah *et al.*, "IoT-enabled shipping container with environmental monitoring and location tracking," in *Proc. IEEE 17th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Jan. 2020, pp. 1–6.

[7] J. R. van Dorp and J. R. W. Merrick, "On a risk management analysis of oil spill risk using maritime transportation system simulation," *Ann. Oper. Res.*, vol. 187, no. 1, pp. 249–277, 2011.

[8] P. K. Selvam, G. Raja, V. Rajagopal, K. Dev, and S. Knorr, "Collision-free path planning for UAVs using efficient artificial potential field algorithm," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Helsinki, Finland, Apr. 2021, pp. 1–5.

[9] Z. Zhou, A. Gaurav, B. B. Gupta, M. D. Lytras, and I. Razzak, "A fine-grained access control and security approach for intelligent vehicular transport in 6G communication system," *IEEE Trans. Intell. Transp. Syst.*, early access, Sep. 2, 2021, doi: 10.1109/TITS.2021.3106825.

[10] H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta, and C.-H. Hsu, "Efficient and secure routing protocol based on artificial intelligence algorithms with UAV-assisted for vehicular ad hoc networks in intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4757–4769, Jul. 2021.

[11] R. S. P. Gaonkar, M. Xie, K. M. Ng, and M. S. Habibullah, "Subjective operational reliability assessment of maritime transportation system," *Expert Syst. Appl.*, vol. 38, no. 11, pp. 13835–13846, 2011.

[12] D. Tsiotas and S. Polyzos, "Analyzing the maritime transportation system in Greece: A complex network approach," *Netw. Spatial Econ.*, vol. 15, no. 4, pp. 981–1010, 2015.

[13] A. Ranjha, G. Kaddoum, and K. Dev, "Facilitating URLLC in UAV-assisted relay systems with multiple-mobile robots for 6G networks: A prospective of agriculture 4.0," *IEEE Trans. Ind. Informat.*, early access, Nov. 30, 2021, doi: 10.1109/TII.2021.3131608.

[14] Z. Xiao, X. Fu, L. Zhang, and R. S. M. Goh, "Traffic pattern mining and forecasting technologies in maritime traffic service networks: A comprehensive survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 1796–1825, May 2020.

[15] S. Fu, X. Yan, D. Zhang, and M. Zhang, "Risk influencing factors analysis of arctic maritime transportation systems: A Chinese perspective," *Maritime Policy Manag.*, vol. 45, no. 4, pp. 439–455, 2018.

[16] S. Fang, Y. Wang, B. Gou, and Y. Xu, "Toward future green maritime transportation: An overview of seaport microgrids and all-electric ships," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 207–219, Jan. 2020.

[17] D. He, S. Zeadally, B. Xu, and X. Huang, "An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Trans. Inf. Forensics Security*, vol. 10, pp. 2681–2691, 2015.

[18] Y. Zheng, "Digital signcryption or how to achieve cost," in *Proc. Annu. Int. Cryptol. Conf. (CRYPTO)*, Aug. 1997, pp. 165–179.

[19] M. G. Muñiz and P. Laud, "Strong forward security in identity-based signcryption," *J. Discrete Math. Sci. Cryptogr.*, vol. 16, nos. 4–5, pp. 235–258, 2013.

[20] C. Gentry and Z. Ramzan, "Identity-based aggregate signatures," in *Proc. Int. Workshop Public Key Cryptogr. (PKC)*, New York, NY, USA, Apr. 2006, pp. 257–273.

[21] S. S. D. Selvi, S. S. Vivek, J. Shriram, S. Kalaivani, and C. P. Rangan, "Security analysis of aggregate signature and batch verification signature schemes," IACR Cryptol. ePrint Arch., Lyon, France, Rep. 2009/290, 2009.

[22] S. S. D. Selvi, S. S. Vivek, J. Shriram, and C. P. Rangan, "Efficient and provably secure identity based aggregate signature schemes with partial and full aggregation," IACR Cryptogr. ePrint Arch., Lyon, France, Rep. 2010/461, 2010.

[23] C. Yuan, W. Chen, and D. Li, "A hierarchical identity-based signcryption scheme in underwater wireless sensor network," in *Wireless Sensor Networks*, vol. 812. Singapore: Springer, 2018, ch. 5, pp. 44–54.

[24] B. Nayak, "A secure ID-based signcryption scheme based on elliptic curve cryptography," *Int. J. Comput. Intell. Stud.*, vol. 6, nos. 2–3, pp. 150–156, 2018.

[25] X. Li, H. Qian, J. Weng, and Y. Yu, "Fully secure identity-based signcryption scheme with shorter signcryptext in the standard model," *Math. Comput. Model.*, vol. 57, nos. 3–4, pp. 503–511, 2013.

[26] A. Karati, S. H. Islam, G. P. Biswas, M. Z. A. Bhuiyan, P. Vijayakumar, and M. Karuppiah, "Provably secure identity-based signcryption scheme for crowdsourced industrial Internet of Things environments," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2904–2914, Aug. 2018.

[27] E. Abouelkheir and S. El-sherbiny, "Pairing free identity based aggregate signcryption scheme," *IET Inf. Security*, vol. 14, no. 6, pp. 625–632, 2020.

[28] N. Radziwill, "Blockchain revolution: How the technology behind bitcoin is changing money, business, and the world," *Qual. Manag. J.*, vol. 25, no. 1, pp. 64–65, 2018.

[29] G. Raja, Y. Manaswini, G. D. Vivekanandan, H. Sampath, K. Dev, and A. K. Bashir, "AI-powered blockchain—A decentralized secure multiparty computation protocol for IoV," in *Proc. IEEE INFOCOM Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Toronto, ON, Canada, Jul. 2020, pp. 865–870.

[30] Mamta, B. B. Gupta, K.-C. Li, V. C. M. Leung, K. E. Psannis, and S. Yamaguchi, "Blockchain-assisted secure fine-grained searchable encryption for a cloud-based healthcare cyber-physical system," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 12, pp. 1877–1890, Dec. 2021.

[31] M. Ali, J. Nelson, R. Shea, and M. J. Freedman, "Blockstack: A global naming and storage system secured by blockchains," in *Proc. USENIX Annu. Techn. Conf. (USENIX ATC)*, Denver, CO, USA, Jun. 2016, pp. 181–194.

[32] R. C. Merkle, "A digital signature based on a conventional encryption function," in *Proc. Conf. Theory Appl. Cryptograph. Techn. (CRYPTO)*, Santa Barbara, CA, USA, Aug. 1987, pp. 369–378.
[33] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. Workshop Theory Appl. Cryptograph. Techn.*, 1984, pp. 47–53.
[34] T. Okamoto and S. Uchiyama, "Security of an identity-based cryptosystem and the related reductions," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn. (EUROCRYPT)*, May 1998, pp. 546–560.
[35] J.-S. Hwu, R.-J. Chen, and Y.-B. Lin, "An efficient identity-based cryptosystem for end-to-end mobile security," *IEEE Trans. Wireless Commun.*, vol. 5, no. 9, pp. 2586–2593, Sep. 2006.
[36] J. Malone-Lee, "Identity-based signcryption," IACR Cryptol. ePrint Arch., Lyon, France, Rep. 2002/98, 2002.
[37] H. Qin, Y. Dai, and Z. Wang, "Identity-based multi-receiver threshold signcryption scheme," *Security Commun. Netw.*, vol. 4, no. 11, pp. 1331–1337, 2011.
[38] S. Ullah, L. Marcenaro, and B. Rinner, "Secure smart cameras by aggregate-signcryption with decryption fairness for multi-receiver IoT applications," *Sensors*, vol. 19, no. 2, p. 327, 2019.
[39] C. Peng, J. Chen, M. S. Obaidat, P. Vijayakumar, and D. He, "Efficient and provably secure multireceiver signcryption scheme for multicast communication in edge computing," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6056–6068, Jul. 2020.
[40] M. Zia and R. Ali, "A multi recipient aggregate signcryption scheme based on elliptic curve," *Wireless Pers. Commun.*, vol. 115, no. 2, pp. 1465–1480, 2020.

Yi Yang received the master's degree from the School of Cyber Science and Engineering, Wuhan University, China, in 2019, where she is currently pursuing the Ph.D. degree. Her research interests are in the areas of cryptographic protocols and secure multiparty computation.

Debiao He (Member, IEEE) received the Ph.D. degree in applied mathematics from the School of Mathematics and Statistics, Wuhan University, Wuhan, China, in 2009, where he is currently a Professor with the School of Cyber Science and Engineering. He has published over 100 research papers in refereed international journals and conferences, such as IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and Usenix Security Symposium. His work has been cited more than 10 000 times at Google Scholar. His main research interests include cryptography and information security, in particular, cryptographic protocols. He is the recipient of the 2018 IEEE SYSEMS JOURNAL Best Paper Award and the 2019 *IET Information Security* Best Paper Award. He is on the Editorial Board of several international journals, such as *Journal of Information Security and Applications*, *Frontiers of Computer Science*, and *Human-Centric Computing and Information Sciences*.

Pandi Vijayakumar (Senior Member, IEEE) received the B.E. degree in computer science and engineering from Madurai Kamaraj University, Madurai, India, in 2002, the M.E. degree in computer science and engineering from the Karunya Institute of Technology, Coimbatore, India, in 2005, and the Ph.D. degree in computer science and engineering from Anna University, Chennai, India, in 2013. He is the Former Dean and currently an Assistant Professor with the Department of Computer Science and Engineering, University College of Engineering Tindivanam, Melpakkam, India, which is a constituent college of Anna University Chennai. He has seventeen years of teaching experience and he has produced four Ph.D. candidates successfully. He has also authored and coauthored more than 100 quality papers in various IEEE transactions/journals, ACM transactions, Elsevier, IET, Springer, Wiley, and IGI Global journals. Till now, he has authored four books for various subjects that belong to the Department of Computer Science and Engineering. He is serving as an Associate Editor in many SCI indexed journals, namely, *International Journal of Communication Systems* (Wiley), *PLOS One*, *International Journal of Semantic Web and Information Systems* (IGI Global), and *Security and Communication Networks* (Wiley|Hindawi). Moreover, he is serving as an Academic Editor for the *International Journal of Organizational and Collective Intelligence* (IGI Global), *International Journal of Software Science and Computational Intelligence* (IGI Global), *International Journal of Cloud Applications and Computing* (IGI Global), *International Journal of Digital Strategy, Governance, and Business Transformation* (IGI Global), and *Security and Privacy* (Wiley). He is also serving as a Technical Committee Member for *Computer Communications* (Elsevier). Recently, he was elevated to the Editor-in-Chief of the *Cyber Security and Applications* (KeAilElsevier). He is also listed in the world's Top 2% Scientists for citation impact during the calendar year 2020 by Stanford University.

Brij B. Gupta (Senior Member, IEEE) received the Ph.D. degree from the Indian Institute of Technology Roorkee, India. In more than 16 years of his professional experience, he published over 400 papers in journals/conferences, including 25 books and eight patents with over 13 900 citations. He is currently working as a Professor with the Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan. His research interests include information security, cyber–physical systems, cloud computing, blockchain technologies, intrusion detection, AI, social media, and networking. He has received numerous national and international awards, including the Canadian Commonwealth Scholarship in 2009, the Faculty Research Fellowship Award in 2017 from the Government of Canada, MeitY, GoI, the IEEE GCCE Outstanding and WIE Paper Awards, and the Best Faculty Award in 2018 and 2019, NIT Kurukshetra, respectively. He is also selected in the 2021 and 2020 Stanford University's ranking of the world's top 2% scientists. He is/was also a visiting/adjunct professor with several universities worldwide. He was selected as a 2021 Distinguished Lecturer in IEEE CTSoc. He is also serving as the Member at Large, Board of Governors, IEEE Consumer Technology Society (2022–2024). He is also leading *International Journal on Semantic Web and Information Systems*, *International Journal of Software Science and Computational Intelligence*, and *International Journal of Cloud Applications and Computing* (IGI Global), as the Editor-in-Chief. Moreover, he is also serving as a lead-editor of a Book Series with CRC and IET Press. He also served as a TPC member in more than 150 international conferences. He is also serving/served as an associate/guest editor of various journals and transactions.

Qi Xie received the Ph.D. degree in applied mathematics from Zhejiang University, China, in 2005. He is a Professor with Hangzhou Normal University, the Director of Key Laboratory of Cryptography of Zhejiang Province. From 2009 to 2010, he was a Visiting Scholar with the Department of Computer Science, University of Birmingham, U.K., and a Visiting Scholar with the Department of Computer Science, City University of Hong Kong in 2012. His research area is applied cryptography, including digital signatures, authentication, and key agreement protocols. He has published over 80 research papers in international journals and conferences, and served as general co-chairs of ISPEC2012 and ACM ASIACCS2013, and a reviewer for over 30 international journals.