Hindawi
International Journal of Intelligent Systems
Volume 2023, Article ID 1814053, 14 pages
https://doi.org/10.1155/2023/1814053

Research Article

Secure Marine Environment Communication: A Multiobject Authentication Protocol Based on Secret Sharing

Jun Ye, Xinhui Cao, and Zhen Guo

School of Cyberspace Security, Hainan University, Key Laboratory of Internet Information Retrieval of Hainan Province, Haikou, Hainan, China

Correspondence should be addressed to Jun Ye; yejun@hainanu.edu.cn

Received 6 September 2022; Revised 11 November 2022; Accepted 2 February 2023; Published 22 February 2023

Academic Editor: Vasudevan Rajamohan

In the field of marine communication, with the rapid development of the Internet of Things, big data, blockchain, and other new-generation information technology, marine information security is gradually being attached to more and more coastal countries, and marine information technology has ushered in epoch-making development opportunities. However, marine communication networks are open, and marine equipment is vulnerable to attacks on communication systems. On the one hand, a malicious adversary can intercept and analyze information, obtain relevant data with high probability, and even obtain the identity information of the equipment. On the other hand, attackers can also maliciously inject false information. At the same time, it is difficult for the existing marine information security technology to guarantee the confidentiality and authenticity of information simultaneously, which will bring substantial potential hidden dangers to the development of the ocean. We propose a multitarget authentication and key exchange protocol for secure communication. Our protocol assigns complex cryptographic operations to servers, thus balancing the system's security and efficiency. The secret is divided into multiple subsecrets using the secret sharing feature, and the subsecret information is used to manage the identity credentials of resource-constrained devices. Then, after successful authentication, the server reassigns the subsecret information of the device to achieve the dynamic generation of identity credentials. Meanwhile, a multitarget authentication scheme is proposed based on recovering secrets. In addition, device extension measures are provided to replace or increase devices. Finally, well-established cryptographic assumptions are used to prove the protocol's security. Simulation results verify the effectiveness of the protocol in multiobjective authentication.

1. Introduction

With the continuous exploration and development of the ocean and the strengthening of ocean management and control by all countries, the development of the ocean will become a hot topic in the new era. The key to ocean development is information management, of which ocean information security is one of the most critical factors. The ocean is a complex system consisting of various elements, such as the marine environment, marine equipment, and human activities. Marine networks generally comprise sensors, satellites, offshore fixed platforms, and onshore platforms, forming a complex marine network environment that brings hidden dangers to the security and privacy of marine information. On the other hand, traditional network security cannot be directly used in the complex network environment of the ocean due to the limitation of the computing and storage capacity of marine devices.

First of all, because of being limited by the complex network environment of the ocean, the way of marine communication is simple, generally using fixed platforms at sea (such as fixed signal towers and fixed buoys), movable platforms (such as ships, submarines, and seaplanes) or other methods. For deep-sea data acquisition, satellite and wireless communication are usually used for data interaction due to environmental conditions. However, wireless networks are open networks that are vulnerable to adversary security attacks such as "eavesdropping," "tampering," "forgery," and "replay" during information transmission. Second, the influence of external factors can also bring trouble to the exploration and collection of marine data. To ensure the confidentiality of data, the reliability and authenticity of data sources, and the correctness of data contents, it is usually necessary to ensure the verifiability of data while encrypting data transmission. In addition, due to the complex marine network environment and fragile network stability, there will be a lot of interference during data transmission, which primarily affects the secure transmission of data, and the overhead used for interdevice communication is greater than that of traditional wired networks. The equipment used for data collection can also be dangerous when exposed for a long time. Finally, the data collected cannot be processed perfectly due to the limited functionality of the data collection devices. Also, there is no verification method for data authenticity set up at the data receiving end, or the verification method is too simple and fixed, which makes it difficult to identify false data injected by malicious adversaries.

Figure 1 shows the marine communication architecture studied in this paper. The architecture is divided into the ocean and land parts, where the solid line represents the transmission path. There are two entities in the ocean part, buoys and relay nodes, where the buoy is the primary data collection device. Relay nodes are set up because it is difficult to transmit data to the designated server at one time when collecting data in distant seas. The relay device plays the role of data storage rather than data processing in the ocean. This specific study is not covered in this paper. The land-based part has only one entity, the shore-based platform, whose primary purpose is to verify the data sources (authentication devices) and to compute session secrets. The buoy will communicate with the shore-based platform to exchange information before it is put into production in the ocean part. The primary industrial significance of the scheme in this paper is divided into the following three aspects of the whole system: (1) encryption of the data collected by the buoy to ensure the confidentiality of the data; (2) the shore-based platform verifies the data source to ensure the legitimacy of the data; (3) multitarget authentication is used to reduce unnecessary waiting time, thus ensuring the system's efficiency.

Currently, marine information security is mainly implemented through cryptography. Authentication and digital signature schemes are utilized primarily to verify data authenticity. Most authentication protocols are divided into three phases: identification, authentication, and authorization. Before authentication, devices are registered on the network and authenticated during the login process. Multiple communications occur between devices during these processes, so data privacy must be considered. To address these issues, different protocols choose different mechanisms to authenticate users. For example, RFID, biometrics, or alphanumeric passwords to authenticate users. In addition, data authenticity assurance can be achieved by data legitimacy verification. However, the existing data legitimacy verification methods do not consider the errors that occur during data transmission and the limited resources of the devices. In terms of data confidentiality, asymmetric encryption algorithms and symmetric encryption algorithms are mainly used. Among them, the asymmetric encryption algorithm has higher security. Still, the algorithm has a high overhead, which is not suitable for the application scenario of ocean buoys that process large amounts of data and require a high frequency of encryption and decryption. Symmetric encryption algorithm has higher encryption efficiency but uses the same key in encryption and decryption, so the algorithm's security depends on the key's security. It is not suitable for communication in marine environments with high-security requirements.

1.1. System Model.

In this system model, we follow two submodels (authentication and threat models) to design our proposed scheme.

1.1.1. Authentication Model.

The two communicating parties of the system are the ocean data acquisition device and the ocean data acquisition server. The ocean data acquisition device has a unique identifier and can perform the encryption algorithms mentioned in this paper. The ocean data acquisition server is an information acquisition platform. It can be a shore-based platform or an ocean mobile server platform. The data placed on it is trustworthy and secure. See Figure 2 for details.

1.1.2. Threat Model.

Because data exchange takes place in an open network environment, we assume that the adversary can control the communication channel somehow. The server in this paper is a trusted data collection server that does not disclose any relevant data to third parties. The adversary can simulate a client sending a message to the server for verification. Sometimes to fully emulate, the adversary may obtain some identity information to communicate with the server. The adversary can also mimic the server. The adversary can pass a fake session key to the client to obtain the client's accurate information during the key exchange.

1.2. Our Research Contributions.

A multitarget authentication based on secret sharing and a key exchange protocol for secure communication are proposed in this paper that can alleviate the security problem. We use an end-to-end authentication transmission model. The device is responsible for the intelligent collection of data and the encrypted transmission of data. The server is responsible for authenticating the device and decrypting the data, and after authentication, a secure channel is established between the two, and data is exchanged. A physical unclonable function (PUF) is proposed here to generate the server's identity credentials. The most important feature of this protocol is that it performs multitarget authentication while recovering the secret, thus saving computational resources and improving communication efficiency. Also, the session key is computed after successful authentication. Moreover, the operation of this protocol is asymmetric complex

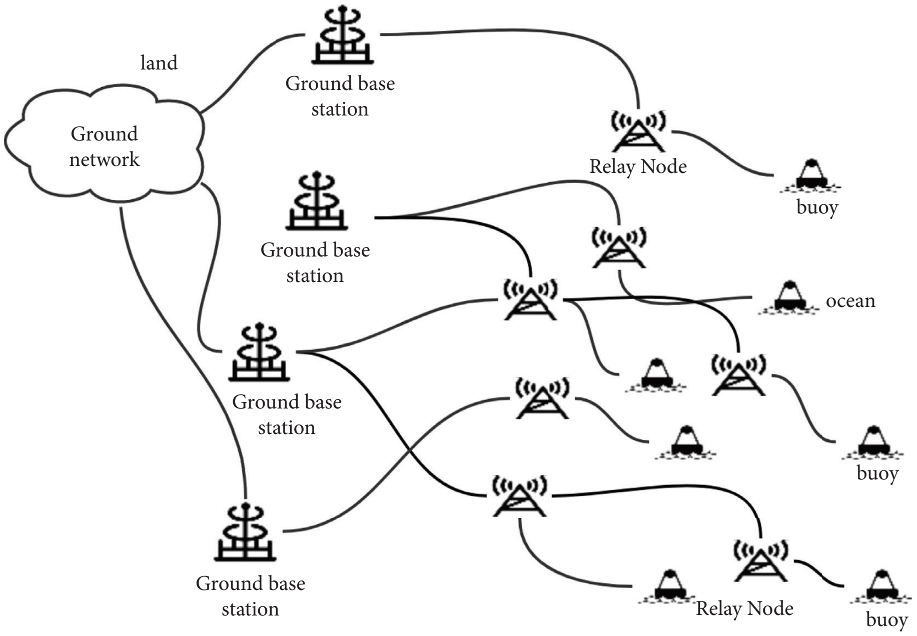

Figure 1: Marine communication environment.

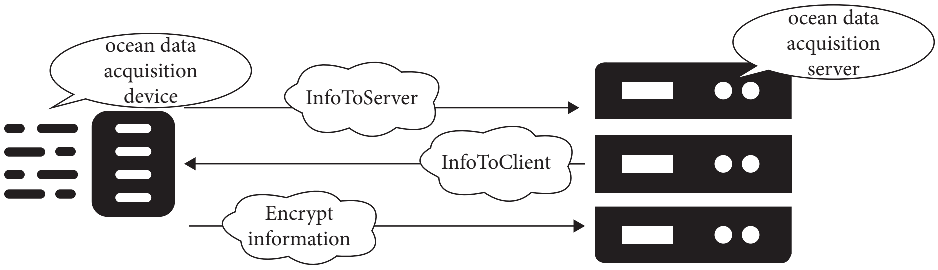

Figure 2: System model.

computational steps are assigned to the server, and the device requires less computation. The following are the main contributions of our protocol:

(1) We propose an efficient, secure, multiobject authentication key exchange protocol. Each communication exchange of data ensures that the receiver authenticates the sender, thus guaranteeing the data's confidentiality, authenticity, and legitimacy.

(2) We draw on the secret sharing approach to perform multitarget bulk authentication processing, thereby reducing the server overhead and ensuring communication efficiency.

(3) We use well-established cryptographic assumptions to prove the security of the protocol.

The structure of this paper is arranged as follows: first, related research will be introduced in the second section. Then, in the third section, related knowledge designed in this paper will be presented. After that the proposed scheme will be explained in the fourth section. In the fifth section, a security analysis of the scheme will be conducted. The experimental analysis and comparison will be carried out in the sixth section. Finally, section seven summarizes our scheme.

2. Related Work

The origin of secret sharing dates back to 1979 by Shamir [1] and Blakley [2]. lt is an information security technique used in cryptography for data privacy protection and requires the cooperation of multiple participants to recover a secret. Secret sharing divides the entire secret into multiple unrelated subsecrets, each of which is kept by a different participant. *t* is not feasible for a single participant to recover the entire secret information. It works when the participants cooperate or when the number of participants meets the relevant threshold. The most attractive advantage of secret sharing is that if a specific range of participants makes an error or defect, the secret can still be fully recovered by the remaining participants, which can effectively prevent attacks by external enemies and betrayal by internal users, facilitating data privacy protection.

At present, secret sharing has been widely used in various fields to provide security services for real-world applications and guarantee data privacy. In 2002, Wu et al. [3] applied secret-sharing technology to ensure the privacy of stored secret images without being easily leaked. While Li and Hwang [4] first proposed the use of threshold-based secret-sharing techniques to provide secure authentication. Al-Ghamdi et al. [5] and others found specific problems in the sharing process of the secret sharing scheme based on the original multimedia count. By studying the limitation of the sharing number and according to the corresponding scope, they try to eliminate the effect that the number of zero-bit boundaries in the original target key has on system performance. It overcomes the shortcoming of bit similarity between the shared key and the target key and improves security. Makino et al. [6] discussed a previous verifiable secret-sharing scheme [7] and pointed out the necessity of multiparty computation to protect the sharing from malicious actors. Then, they studied a new secret-sharing scheme that can be verified without the participation of multiple parties, and the security of verification is proved. Harn et al. [8] found that in secret sharing if the outside world can obtain all the information exchanged during the interaction, the adversary can still recover the secret. They propose a confidential SSR scheme to solve this problem. Even if the external adversary obtains the vast majority of exchange information, the secret will be prevented from leaking, and the security of this extended scheme does not depend on any computational assumptions. Ma et al. [9] considered the cheating problem of internal participants in secret information and designed a secret information-sharing scheme with cheating identification and detection. When the participants restore the secret, the scheme can identify cheaters. Goyal et al. [10] designed a secret-sharing scheme for traceability. Once a participant cheats, all his actions can be traced back to the source.

Identity authentication is an authentication technology in which participants distrust each other but must communicate and identify each other. We need to ensure the anonymity of both parties' identities and the privacy of data transmission during the identity authentication process. Zhang et al. [11] designed a biometric-based authentication protocol in a multiserver environment, utilizing PUF and revocable biometrics for authentication. The authentication identity information of this protocol changes dynamically, but the authentication interaction is so frequent that it is not suitable for the complex marine environment. Based on the Industrial Internet of Things, Srinivas et al. [12] proposed a method for biometric authentication and identification using fuzzy extraction technology. Although this scheme is lightweight, considering that the device does not have corresponding biometric characteristics, this scheme is not suitable for identifying hardware devices. Abbasinezhad-Mood et al. [13] enhanced and improved the anonymous Dl-DAKA protocol proposed by Shu et al. and effectively solved the security weaknesses of some schemes. However, the authentication overhead of this protocol is relatively large, and it is challenging to implement under limited conditions. Wei L et al. [14] designed a privacy-preserving aggregate authentication scheme for vehicle safety warnings, which uses fog nodes to collect relevant data and authenticate. This method requires the help of other network resources. It is not easy to achieve communication in marine networks.

Many studies also propose key agreement schemes to ensure the confidentiality of data transmission. Boyapally et al. [15] proposed an operationally asymmetric mutual authentication and key-exchange protocol for secure communication. The protocol considers security and efficiency issues and strikes a good balance in natural environments. However, the communication overhead of this scheme is relatively large, and it is not easy to realize in the complex marine environment. Meshram et al. [16] proposed a biometric-based authentication scheme that uses convolution-Chebyshev chaotic mapping and key agreement for authentication. This solution requires relatively large device resources and is unfeasible to implement in resource-constrained devices. Saxena et al. [17] designed a batch authentication and key agreement protocol for short message service, which mainly transmits information from one user to multiple users over insecure communication channels. However, the cost of this protocol is very high, which is impracticable to implement under the limited conditions of marine equipment. Poh et al. [18] proposed a privacy-preserving scheme for intelligent home authentication and data security storage, which consists of a key-exchange protocol and an efficient searchable encryption protocol. Alawatugoda and Okamoto [19] proposed an authenticated key-exchange protocol structure based on the Diffie–Hellman. The instantiate is achieved by combining public key encryption and pseudorandom functions. The protocol has also been proven to be secure. Based on the research findings of existing methods, Kilisters and Rausch [20] proposed a new key-exchange plan, which allows most protocols to perform key exchange in a relatively desired way. Based on this, an experimental analysis was used to demonstrate the concept.

3. Preliminaries

3.1. Identity-Based Encryption. Identity-based encryption is a type of encryption with only one public key, the master public key mpk. For situations where the number of users may be exponential, the size of the master public key mpk is generally small relative to the number of users. Based on this, the master public key mpk is a compressed representation of the exponent of the user's public key. However, the user does not generate the identity key himself but obtains the identity key from the key authority holding the master key msk. To send a message to a user with an identity, only the master public key mpk and the identity is required. Users with the same credentials can decrypt the such ciphertext.

An identity-based encryption algorithm consists of the following steps:

IBE. SetUp (1^{λ}): entering a relevant parameter λ, and then you can get the master secret key msk and the master public key mpk

IBE.KeyGen (msk, id): getting a decryption key sk_{id} using the msk and identity id as input

IBE.Encrypt (mpk, id, M): Using mpk, identity id, and plaintext information M to be encrypted as input and then output ciphertext C

IBE.Decrypt (C, sk_{id}}: using the information C to be decrypted and the decryption key sk_{id} as input to get the plaintext information M

3.2. Symmetric Key Encryption. Symmetric key encryption SKE consists of the following steps:

SKE.KeyGen (1^{λ}): getting a secret key (sk) using security parameters as input

SKE.Encrypt (sk, M): using sk and plaintext information M to obtain ciphertext information C

SKE.Decrypt (C, sk): using sk and ciphertext information C to obtain plaintext information M

Although SKE is symmetric encryption, we use the device identity information as part of the encryption key in the initialization phase, which is equivalent to simultaneous authentication of the device as decryption. And we still use the subsecret information as part of another encryption key. The subsecret information will be dynamically adjusted after each transmission, equivalent to the key being dynamically generated. We consider using AES in counter mode to meet the need for security.

3.3. Secret Sharing. Secret sharing means a secret is computationally dispersed into multiple subsecrets, and different members keep the corresponding subsecrets. We choose a secret S and compute the subsecret information as $(a_1, a_2, \ldots, a_{T-1})$ in our proposed scheme, coefficients as nonconstant terms of $(T-1)$, degree polynomial $F(x) =$, and the polynomial $F(x) =$ are as follows:

$$F(x) = S + a_1 x + a_2 x^2 + \cdots + a_{T-1} x^{T-1}. \tag{1}$$

S is a secret that is placed in the polynomial as a constant term. Once we know the T correlated subsecrets, we can use Lagrangian interpolation to recover the coefficients of $F(x)$ and finally, find the secret. However, if the number of subsecrets we know is less than T, the secret S cannot be solved. The Lagrange interpolation formula is as follows:

$$L(x) := \sum_{j=0}^{t-1} y_j \ell_j(x). \tag{2}$$

$l_j(x)$ as a basic polynomial, its expression can be expressed as follows:

$$\ell_j(x) := \prod_{i=0, i\neq j}^{t-1} \frac{x - x_i}{x_j - x_i}. \tag{3}$$

We use the secret sharing technology to generate the client's identity credentials in our proposed scheme, and we can perform multitarget authentication based on restoring the secret S. We will describe the application of this authentication technology in the fourth subsection.

3.4. Security Definitions for Multiobject Authentication Key-Exchange Protocol

3.4.1. Design Goals. Our scheme can meet the following conditions.

(i) Known session keys: the security of the session key can be guaranteed even if the adversary has obtained the previous session key by other means

(ii) Forward secrecy: if the identity credentials of one or more devices are compromised, the security of the previous session keys is not affected

(iii) Known temporary key: if only the information of the temporary key is exposed during the key exchange, the session key will not be affected

(iv) Message authentication: the server can verify the integrity of the received message, and can check the legitimacy of the sender of the message.

(v) Replay attacks: even if an adversary obtains some transmission, the server will recognize that information before deciding whether to respond.

(vi) Resistance of modification attack: once a message has been tampered with during transmission, the message recipient can refuse to respond to the message.

(vii)Traceability: the identity information of the message sender is bound to the transmitted message. In case of a problem with a received message, the server can immediately determine the source of the error.

(viii) User Anonymity: even if all messages are intercepted during transmission, no adversary except the server can know the true identity of the source of the message.

(ix) Revocability: any credentials used by the sender of the message will be revoked during the next authentication process.

Next, we will give a clear description of the design goals of the appeal.

3.4.2. Oracle Queries. Adversary B can conduct the following oracle queries during the experiment:

(i) Key-Reveal (nonce): for a session whose communication identifier is nonce, adversary B can obtain the same session key

(ii) Credentials-Reveal: adversary B can obtain identity credentials, which are considered risky once leaked.

(iii) Ephemeral-Key-Reveal (nonce): for a session whose communication identifier is nonce, adversary B can obtain any of its ephemeral session keys

(iv) Test (nonce): for a session with a communication identifier of the nonce, adversary B can obtain the corresponding real session key, but this situation is generally limited to one such result

3.4.3. Fresh Session. If the following conditions are met, the session is considered to be fresh:

(i) For this session, adversary B cannot perform a Key-Reveal query
(ii) Adversary B can perform Ephemeral-Key-Reveal operations, but it cannot perform Credential-Reveal operations
(iii) Adversary B can perform Credential-Reveal operations, but it cannot perform Ephemeral-Key-Reveal operations

Suppose adversary B executes a test query about the oracle query and wants to obtain the information. When it obtains the session key this time, adversary B splits the data and can obtain the encryption key K and also knows to obtain the subsecret information. At this point, the adversary wants to crack the whole system and obtain the encrypted information of all communications, but according to the following theory proposed in our above scenario:

(i) The number of subsecrets used to calculate the session secret information is two, and the subsecret information is replaced after each authentication
(ii) The identity credentials calculated by the client are also dynamically adjusted according to the subsecret

Knowing that this behavior is not desirable.

4. Multiobject Authentication Key-Exchange Protocol

This chapter details the secret sharing-based multitarget authentication protocol and key agreement scheme. This protocol is designed to solve the problem of secure transmission of ocean data in an open network environment. Due to the environment and conditions, data acquisition and reception consist of two devices: a data acquisition device and a data reception server. The former is randomly dropped into the ocean environment for data collection, while the latter is deployed upon request. The data acquisition device is resource-constrained, while the data reception server has a wide signal reception range, high computing power, and ample storage capacity. All data acquisition devices have a unique identification ID (e.g., device fingerprint, and serial number). The protocol is divided into two phases. The first phase registers the information of the device in the trusted environment. Then, the data-receiving server and the data-collection device exchange information. Entering the second phase, the two parties communicate and transmit data, followed by authentication. Finally, key negotiation is performed.

4.1. Subsecret Allocation Strategy and Device Expansion and Retirement

4.1.1. Secret Shard Allocation Strategy. Depending on the specific implementation requirements of the project, the size of the subsecret N can be specified. If the current number of devices is D, the number of allocated secret fragments is defined as D. When setting the threshold T the number of remaining subsecrets can be defined as follows: the number of fragments stored in the server for calculating the recovery secret S is defined as $T-1$, and the remaining subsecrets are defined as idle subsecrets, i.e., $N-D-T+1$. And, the number of idle subsecrets $N-D-T+1$ is used for later device expansion and replacement. The above settings can effectively ensure the dynamic and random nature of the keys used for data encryption and transmission.

4.1.2. Equipment Expansion. According to the subsecret allocation strategy, the idle subsecret is $N-D-T+1$. Whenever a new device is added, a subsecret will be randomly divided from the idle subsecret to the newly added device. The server will maintain a list of data that defines the association properties between the subsecret information s_i in the distributed device and the device *id*.

4.1.3. Equipment Retirement. Whenever identity authentication is performed, the server will obtain the relevant data of the client through decryption and then use the data to compare the associated attributes. If the match is successful, it indicates the correctness of the data. Finally, authentication is performed by combining the steps of restoring secret S. After authentication, a subsecret from the idle subsecret list will be allocated again and passed to the client to calculate the identity credentials for the next authentication. The idle secret shard list replaces s_i' with s_i. Once the match fails, the id and its associated secret shard will be eliminated (here is not to say that the device is given up if the match fails but the pairing relationship because the random shard of the device will change in the next authentication, this kind of the space of the pairing relationship is large enough for the device to communicate), to ensure that once the information of a certain device is leaked, it prevents the adversary from using the information to wirelessly inject useless data.

4.2. Proposed Protocol $\alpha+\beta=x$ (1). We will describe the information registration, multitarget batch authentication, and key calculation steps in detail.

4.2.1. Setup Phase. IBE is an anonymous, secure, and indistinguishable identity encryption scheme in this paper, while SKE is a PCPA secure symmetric key encryption scheme. Then, a polynomial $F(x)$ is constructed according to the number of initialized production equipment to hide the secret S and calculate the subsecret s_i

4.2.2. Enrollment Phase

(1) Subsecret Generation. Regarding the size of the secret S, we conducted a comparative analysis in the simulation experiments in Section 7. In our simulation experiments, we concluded that the number of bits of the secret S has little to do with the time required for authentication in our scheme and only affects the complexity of our calculation of the

session key. The steps of subsecret generation are generally as follows: select a secret S first, then calculate and obtain N subsecrets in secret sharing.

(2) Client Identity Credentials. Whenever a new client registers to join, it will interact with the server, and the server will randomly divide a subsecret s_i from the idle subsecret list and send it to the client. Meanwhile, the server will maintain a list for managing the current device info. At the same time, the server's identifier $H(R)$ will also be transmitted to the client. Then, we calculate the following results as identity credentials:

$$M = \mathrm{id}\|s_i\, sk_{id} = H(M), \tag{4}$$

id is the unique identifier of the device.

(3) Server Identity Credentials Generation. For a given PUF, a specific input $c \in Z$, called a challenge, will produce an output response $R = \mathrm{PUF}(c)$ that is unique to the specific PUF and therefore unclonable. Then, we set the following result as the identity credentials:

$$sk_s = \mathrm{IBE.KeyGen}(\mathrm{msk}, H(R)). \tag{5}$$

The relevant data generation and interaction at this stage are shown in Figure 3.

(4) Authentication. Authentication between the client and server with a unique ID is as follows:

(1) After the client collects certain data, it broadcasts information to make an identity authentication request, and the server generates the following data:

$$\alpha_1 = H(M\|\mathrm{nonce})\,\beta_1 = \mathrm{IBE.Encrypt}(\mathrm{mpk}, H(R), M). \tag{6}$$

The data transmitted this time is $\mathrm{InfoToServer} = (\beta_1, \alpha_1, \mathrm{nonce})$.

(2) The server obtains information by receiving the signal and when it receives the signal (the server can carry multiple signals reception concurrently). The server uses its identity credentials to decrypt the signal data $M' = \mathrm{IBE.Decrypt}(sk_s, \beta_1)$. Verify whether the data has changed during transmission by calculating $\alpha_1' = H(M'\|\mathrm{nonce})$ and comparing its values with α_1 and α_1'. id and s_i can be generated by data splitting. Use the initialized remaining subsecret s_L (A collection of remaining subsecret) to solve the secret S', compare S' and S and query whether the id and s_i in M' exist in the related list maintained by the server at this time. If the appeal conditions are met, the identity of the data source device is verified. Solving the secret S is the key to multiobjective authentication in this article. Once multiple information sources are received, we can solve the minimum threshold T of secret S through the s_i of each information source and cooperate with the subsecret in the server. This can effectively improve the efficiency of verification and avoid the waiting time for server authentication. When considering multi-objective verification, once there is a problem with the S' solved this time. We propose to use a binary method to detect data sources in batches and return the data after successful verification. We have a one-time pass in the best case, equivalent to the time complexity of a single verification $O(n)$. In the worst case, the time complexity is $O(\log_2 n)$. The waiting time caused by our processing of multiple targets is effectively reduced by employing this method.

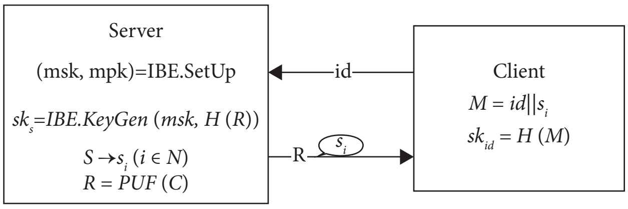

Figure 3: Enrollment phase.

Here, we will introduce the pseudocode of the authentication algorithm used in our experiment and explain the related symbol. $\mathrm{secret}_{\mathrm{cal}}$ represents the secret of this calculation, and secret represents the initial secret. List(left, right) means that all target data are put into a set, and the number of this set is (right, left). Figure 4 is a part of the authentication algorithm used in this paper and expressed in pseudocode. It can be seen from the experiment in Section 7 that algorithm I consume a significant time deviation for authentication at different positions in the set, which is convenient for use when there are few targets. Algorithm II does not significantly impact the time consumed by authentication at various locations in the set, which is suitable for use when there are many targets.

4.2.3. Session Key Calculation

(1) After the authentication in the previous step, the server starts to select a random subsecret s_i' among the remaining subsecret, and then calculates

$$\begin{aligned} g &= H\left(sk_{id}\|sk_s\right), \\ \beta_2 &= \mathrm{SKE.Encrypt}\left(sk_{id}, s_i'\right), \\ g_{s_i} &= H\left(sk_{id}\|sk_s\right)^{s_i}, \\ \alpha_2 &= H\left(s_i'\|(\mathrm{nonce}+1)\right). \end{aligned} \tag{7}$$

Return $\mathrm{infoToClient} = (\beta_2, g_{s_i}, \alpha_2)$ to the client. The servers' remaining subsecret are updated, replacing s_i with s_i' and redistributing remaining subsecret and free subsecret. Then, we calculate the session key as follows:

$$K = H\left(g_{s_i^{s_i'}}\|(\mathrm{nonce}+1)\right). \tag{8}$$

(2) The client uses its own id and the subsecret s_i to calculate $sk_{id}' = H(id\|s_i)$, decrypt and calculate $M_1 = \mathrm{SKE.Decrypt}(sk_{id}', \beta_2)$, and then compare α_2 and $H(M_1\|(\mathrm{nonce}+1))$ is used for data verification and

Algorithm I

```
input: List (left, right)
process:
        if (secret_cal==secret)
           output true
        else
            if right-left==0
               output error data
            else
                mid= (left+right)/2
                Auth (List (mid, mid))
               if left==mid
                else
                  Auth (List (left, mid-1))
               Auth (List (Mid+1, right))
```

Algorithm II

```
input: List (left , right)
process:
 if (secret_cal==secret)
    output true
 else
    if right-left==0
       output error data
    else
       mid= (left+right)/2
        if (left==mid)
          Auth (List (mid, mid))
       else
          Auth (List (left, mid))
     Auth (List (mid+1, right))
```

Figure 4: Multitarget authentication detection algorithm.

comparison. If the two data are equal, it means that there is no problem with the transmission at this time. Then, calculating the session key $K = H(g_{s_i}^{s_i'} \| (\text{nonce} + 1))$, and replace s_i' with s_i. The parameters s_i' and s_i used in the calculation of the session key K are different in each calculation process, and the space combination is large enough.

The specific authentication and key calculation process are as shown in Figure 5.

5. Security Analysis

Theorem 1. *By definition, our protocol is secure under the following conditions: (1) both the SKE and IBE algorithms used for encryption are CPA-secure; (2) the Diffie–Hellman assumption holds in group G; (3) the number of subsecrets is large enough that a random set of subsecret will not be repeated when computing the session key K; (4) the hash function used in this article is random oracles.*

Proof. Assuming that adversary B has passed the authentication key exchange protocol, adversary B can only distinguish the real session key $K = H(g_{s_i}^{s_i'} \| (\text{nonce} + 1))$ of the new session in the following ways:

(i) Key-replication attack: adversary B can force a second session to be constructed using the same session key K as the current session

(ii) Forging attack: during the communication process, the adversary forged data to generate data similar to this communication

Firstly, we analyze adversary B forcing the use of the same session key K as the current target. Because adversary B is limited by definition, it cannot generate multiple sessions with the same nonce value because this attack is equivalent to forcing a collision on the hash function H. On a probabilistic polynomial time algorithm, the probability that adversary B can generate such a hash collision in many queries of Q is expressed as follows:

$$\rho = 1 - \prod_{j=1}^{Q} \left(1 - \Pr\left[H(q_j) = H\left(g^{s_i, s_j'} \mid \text{nonce} + 1 \right) \middle| q_j \neq \left(g^{s_i s_i'} \| \text{nonce} + 1 \right) \right] \right) = 1 - \left(1 - 2^{-\lambda} \right)^{Q}. \tag{9}$$

Deriving from the formula, we can know that this probability is negligible. So adversary B can only perform the forging attack. Here we only consider forgery attacks in two cases. Usually, we need to consider the nature of query B makes, and its forgery attack is negligible under any conditions.

Suppose B performs the Ephemeral-Key-Reveal operation. B can also obtain some parameters currently used for key calculation. To be forged entirely, it needs to know (with a small probability) the subsecret in the interaction between the two and the corresponding identity credentials. However, the server's identity credentials are stored in the device, the adversary cannot obtain this data by definition, and the subsecret is dynamically generated each time (theoretically, the probability of repetition is small). The only information currently available to B is the nonce and the ciphertext encrypted by the corresponding SKE and IBE schemes. Even if the adversary can obtain the identity information of the device with a small probability, since the identity credentials are dynamically adjusted, we know that the adversary cannot recover the corresponding identity credentials to determine the authenticity of the authentication. Thus, we can ensure the feasibility of the next subsecret replacement.

Case 1. Without knowledge of the identity credentials sk_{id}, the probability that adversary B wants to reply s_i' from ciphertext M' is negligible. When the adversary knows the subsecret s_i of a certain communication, it may carry out a data forgery attack by sending many false data attacks, which increases the amount of our computation. However, according to the definition of our scheme, we know that after each calculation of the session key K, the subsecret will be temporarily replaced. Once a large number of unified subsecret attacks occur, we can determine a problem with the data source and directly abandon using this subsecret in the next calculation.

Case 2. Without identity credentials sk_s, adversary B has a negligible probability of trying to reply M from the ciphertext β_1. When the adversary obtains the current subsecret of some device, it can use the client's identity credentials sk_{id}. Nevertheless, according to our definition,

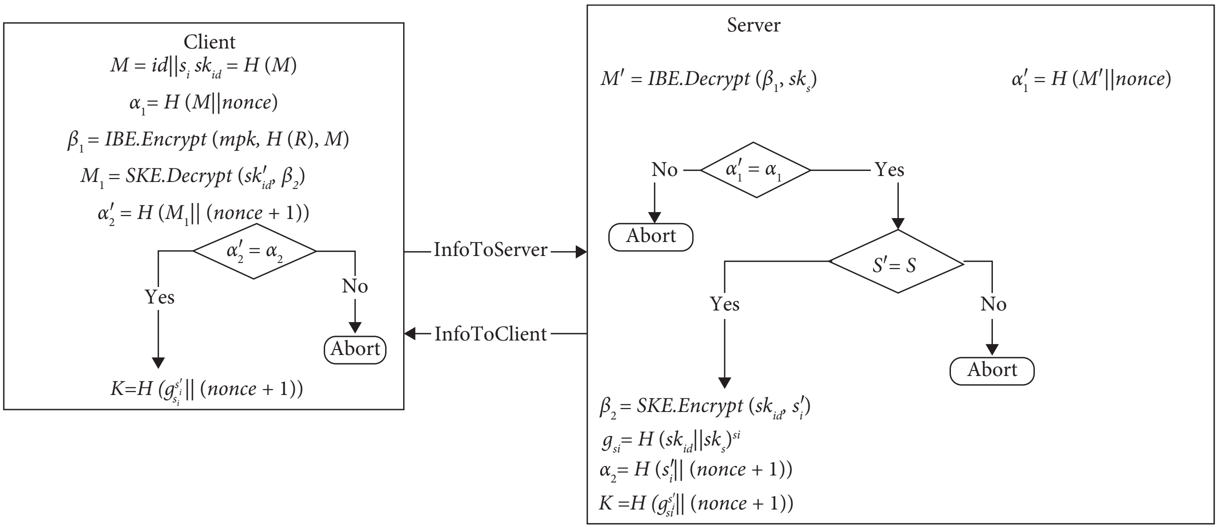
Client
$M = id||s_i\ sk_{id} = H(M)$
$\alpha_1 = H(M||nonce)$
$\beta_1 = IBE.Encrypt(mpk, H(R), M)$
$M_1 = SKE.Decrypt(sk'_{id}, \beta_2)$
$\alpha'_2 = H(M_1||(nonce+1))$
$\alpha'_2 = \alpha_2$
Yes
No
Abort
$K=H(g_{s_i}^{s'_i}||(nonce+1))$
InfoToServer
InfoToClient
Server
$M' = IBE.Decrypt(\beta_1, sk_s)$
$\alpha'_1 = H(M'||nonce)$
No
$\alpha'_1 = \alpha_1$
Yes
Abort
$S' = S$
Yes
No
Abort
$\beta_2 = SKE.Encrypt(sk_{id}, s'_i)$
$g_{si} = H(sk_{id}||sk_s)^{si}$
$\alpha_2 = H(s'_i||(nonce+1))$
$K = H(g_{si}^{s'_i}||(nonce+1))$

Figure 5: Authentication and key calculation.

the adversary cannot know the server's identity credentials sk_s, even if the attacker forges data to verify, it does not have the server's identity credentials, so the server does not verify the source of information.

For the target session, adversary B can only Credential-Reveal queries in this situation. At the same time, suppose the B gets both sk_s and sk_{id}. This results in that B can be calculated β_1, β_2 and $g = H(sk_{id}\|sk_s)$ to improve the attack, it needs to distinguish $(g^{s_i}, g_{s_i}^{s'_i}, K)$ from $(g^{s_i}, g_{s_i}^{s'_i}, K = H(g^z\|(\text{nonce}+1)))$ (z is the multiplication of two random subsecret) without querying for either s_i or s'_i during the fresh session. Let this experiment be $\text{AuthK}_{\mathscr{B},\sqcap}^{\text{eav}}$ when adversary B can successfully discriminate. If there is such an adversary, then we can construct a simulator P, which can solve the DDH problem. Simulator P is given $(G, q, g, g^{s_i}, g^{s'_i}, g^{\omega})$ as input for a target session, where ω is either $s_i\ s'_i$ or, whose goal is to determine which is the case. Adversary B can make any oracle queries to simulator P up to the target session, and will only make Credential-Reveal queries to the target session. The simulator sends the $K = H(H(sk_{id}\|sk_s)^{s_{ix}s_{iy}}\|(\text{nonce}+1))$, (sk_s, sk_{id}), and (s_{ix}, s'_{iy}) session with correlation identifier nonce'. The adversary computes α_1 and α_2. $K = H(g^{\omega}\|(\text{nonce}+1))$ as the key for the fresh session and $(g^{s_i}, g^{s'_i})$ that has been exchanged openly (public). Suppose B can distinguish $g^{s_i s'_i}$ from g^z, we can derive a function negl that can be ignored.

$$\Pr\left[\text{Auth}K_{\text{B},\Pi}^{eav} = 1\right] \leq \frac{1}{2} + \text{negl}(\lambda). \tag{10}$$

We know that the modified authenticated key-exchange protocol randomly selects a z to use to generate the session key $= H(H(sk_{id}\|sk_s))^z$. Although the protocol is not an actual key generation scheme, the experiment $\text{AuthK}_{\mathscr{B},\sqcap}^{eav}$ is still defined. The space created by the secret pairing of two random subsets is large enough, so we can define z is chosen uniformly over Z_q, g^z is a uniform group element in G. It follows that

$$\Pr\left[\text{Auth}K_{B,\Pi}^{eav} = 1\right] = \frac{1}{2}. \tag{11}$$

From the behavior of P, we can draw the following two cases:

Case 3. Once the input of P is randomly selected $s_i, s'_i, z \in Z_q$ is generated, then, when P runs as a subroutine, B views are distributed the same as B'_p views in the experiment $\text{Auth}K_{\mathscr{A},\Pi}^{eav}$. In this way, P can solve the difficult problem of DDH when B outputs 1, we can get that

$$\Pr\left[D(G, q, g, g^{s_i}, g^{s_i}, g^z) = 1\right] = \Pr\left[\text{Auth}K_{\mathscr{A},\Pi}^{eav} = 1\right] = \frac{1}{2}. \tag{12}$$

Case 4. Once the input of P is randomly selected $s_i, s'_i, z \in Z_q$ is generated and calculating $g^{s_i s'_i}$, then, when P runs as a subroutine, B's views are distributed the same as B'_p views in the experiment $\text{AuthK}_{\mathscr{B},\sqcap}^{eav}$. Defining the following equation we get

$$\epsilon(\lambda) = \Pr\left[\text{Auth}K\mathscr{B}_{B,\sqcap}^{eav} = 1\right]^{q(n)c_2}. \tag{13}$$

It can be known that $q(n)$ is the polynomial number of the query issued by B to oracle. In this way, P can solve the difficult problem of DDH when B outputs 1, we can get that

$$\Pr\left[S\left(G, q, g, g^{s_i}, g^{s'_i}, g^{s_i s'_i}\right) = 1\right] = \Pr\left[\text{Auth}K_{B,\Pi}^{eav} = 1\right]^{q(n)C_2}. \tag{14}$$

Because the DDH problem is difficult with respect to G, we can obtain a negligible function negl such that

$$\left(\mathrm{Negl}\left[(\lambda) \geq \left|\Pr\left[S\left(G, q, g, g^{s_i}, g^{s_i^{\prime}}, g^{z}\right) = 1\right] - \Pr\left[S\left(G, q, g, g^{s_1}, g^{s_i^{\prime}}, g^{s_i s_i^{\prime}}\right) = 1\right]\right|\right] = \left|\frac{1}{2} - \epsilon(\lambda)\right|. \tag{15}$$

This implies $\varepsilon(\lambda) \leq (1/2) + \mathrm{negl}(\lambda)$ implying that the authenticated key exchange protocol used in this paper is secure under the assumption of DDH.

5.1. Informal Security Analysis.

5.1. Informal Security Analysis. Our communication channels are subject to adversary attacks in the marine network environment. We will describe some informal security analysis here to demonstrate that our proposed scheme can address data confidentiality, authenticity, and legitimacy.

5.1.1. User Anonymity. We use the feature of secret sharing to fragment the secret and then combine the subsecret and the unique *id* of the device to generate the identity certificate we need. After each authentication, the old identity certificate will be revoked, and the device will obtain a new subsecret used to calculate the new identity credentials for the next authentication. We can achieve device anonymity by ditching the random number mechanism by dynamically changing the subsecret.

5.1.2. Man-in-the-Middle Attack. The adversary can intercept some information through the communication between the two parties and make the two parties exchange their keys separately, resulting in a man-in-the-middle attack. However, the essence of this vulnerability is that the two parties do not authenticate before negotiating the key. Through the analysis of our protocol, we believe that our protocol does not have this problem. We use the identity information of both parties to calculate the corresponding identity credentials and then calculate g. If an adversary wants to calculate g, he must know the identity credentials, but in general, the adversary will not know the identity credentials between the two.

5.1.3. Replay Attacks. We utilize secret sharing techniques and a credential revocation strategy to ensure that our proposed scheme is resistant to replay attacks. Our scheme uses the client's unique *id* and random subsecret as the identity credentials sk_{id} and uses random numbers for authentication in an ocean network environment. After each successful authentication, the current identity credentials will be revoked, and the client will recalculate the identity credentials next time.

6. Experiment Analysis and Comparative Analysis

First of all, we will describe the software implementation of the protocol we proposed in detail. Then, we will conduct a comparative analysis with the scheme [15]. Next, we will analyze the efficiency and effectiveness of the scheme through the simulation results. Finally, we will compare security and functionality with other authentication protocols.

6.1. Software Implementation Details of the Protocol. The marine data acquisition equipment and the marine data acquisition server use the same configuration in the simulation experiment. The specific configuration is a PC with a single-core Intel i7-7700HQ @ 2.80 GHz CPU, 8 Gb RAM, and 400 Gb storage. In this experiment, the IBE scheme of Boneh and Franklin is implemented using many APIs provided by the Java Pairing-Based Cryptography Library (using java to encapsulate the public pairing-based cryptography library). The SKE scheme in this paper is instantiated with AES-256, the hash function is instantiated with SHA-256, and the rich mathematical calculation API in java is used to realize the threshold secret sharing and subsecret allocation strategy.

6.2. Program Comparison Analysis. By comparing the protocol proposed in this paper with the protocol proposed in the literature [15]. In terms of identity authentication, the protocol adopted in this paper can realize multitarget authentication, thereby effectively reducing the authentication waiting delay. At the same time, an algorithm is provided in this paper to detect erroneous data sources. While the literature [15] can only be single-authenticated. A delayed two-way authentication method is used in this paper to ensure that both sides of the device are authenticated during the authentication process. This authentication method effectively reduces the number of communications and improves the overall transmission efficiency. The literature [15] achieves two-way authentication through multiple communications, which consumes many communication resources. In terms of encryption processing, the identity credentials used in this paper are dynamically generated to ensure that the encrypted content is more secure. In contrast, the identity credentials used in [15] are fixed during initialization. By comparison, the protocol proposed in this paper effectively reduces the number of communications and is more efficient in multiobjective authentication. Time complexity will not exceed $O(\log_2 n)$ in the worst case. The session key can be calculated with only one key exchange after successful authentication.

6.3. Comparison of Experimental Results. Research on the influence of threshold secret sharing-related parameters on authentication efficiency. The number of bits of the secret S has little correlation with the time required for authentication in this experiment. Due to the limited computing power of the experimental equipment, once the number of digits of the secret S exceeds 78, the following situation will occur when solving the secret S. There is no problem with the data and the program, but the secret S still cannot be

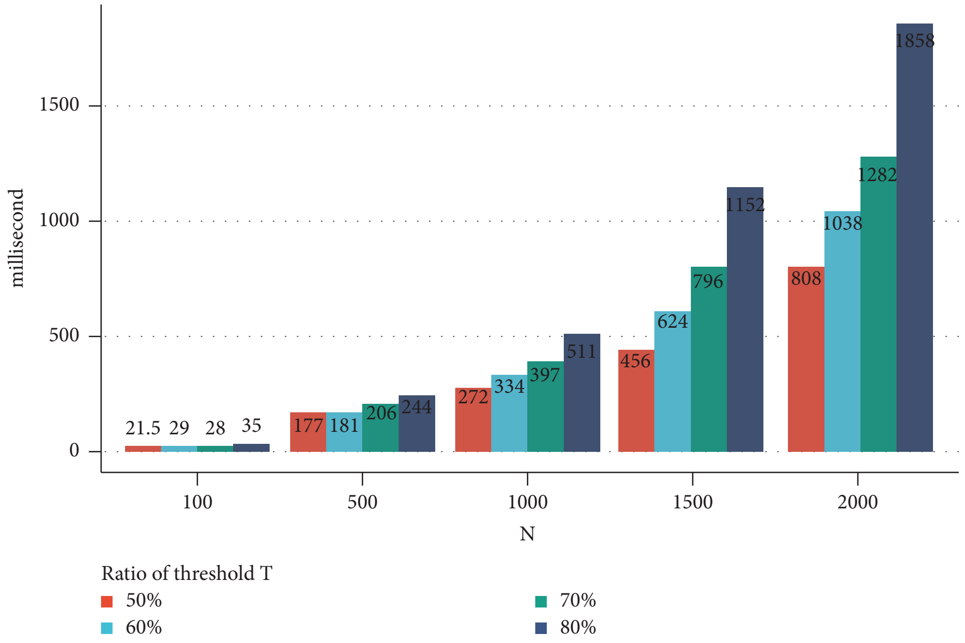

Figure 6: Relationship between authentication time and the number of secret copies and threshold ratio.

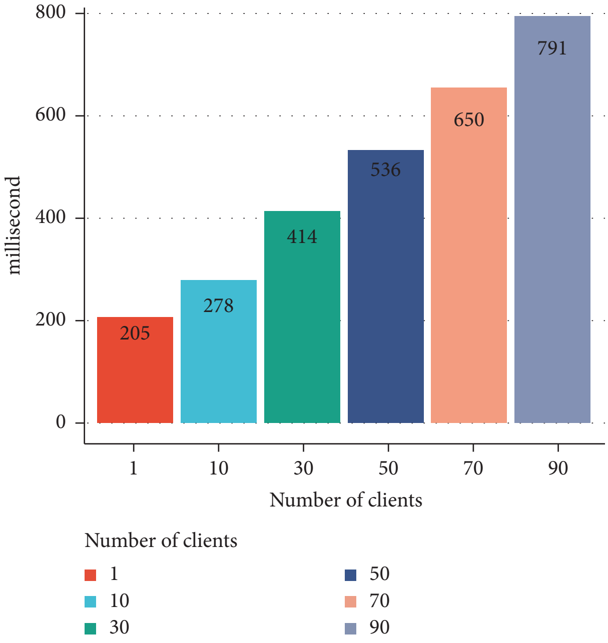

Figure 7: Comparison diagram of time required for multiobjective authentication.

computed correctly. The number of bits of the secret S is fixed to 32 in the following experiments. It can be seen from the experimental results in Figure 6 that when the value of N is not higher than 500, the time required to restore the secret S has not changed much as the proportion of the threshold T required for decryption becomes higher and higher. However, when the value of N is above 1000, as the proportion of the threshold T required for decryption becomes higher and higher, the time difference required to restore the secret S is larger. When fixing the proportion of the threshold T, the time required to restore the secret S grows multiplicatively as the value N becomes larger and larger.

Next, we set the number of bits in the secret S to 32; the initial secret *N* value is 500, and the threshold T ratio is 70\%. As seen from the experimental results in Figure 7, as the number of authentication targets increases, the time required for authentication increases very slowly. When authenticating one target, the time required is 205 milliseconds, and when authenticating 30 targets simultaneously, the time required is only twice that of single-

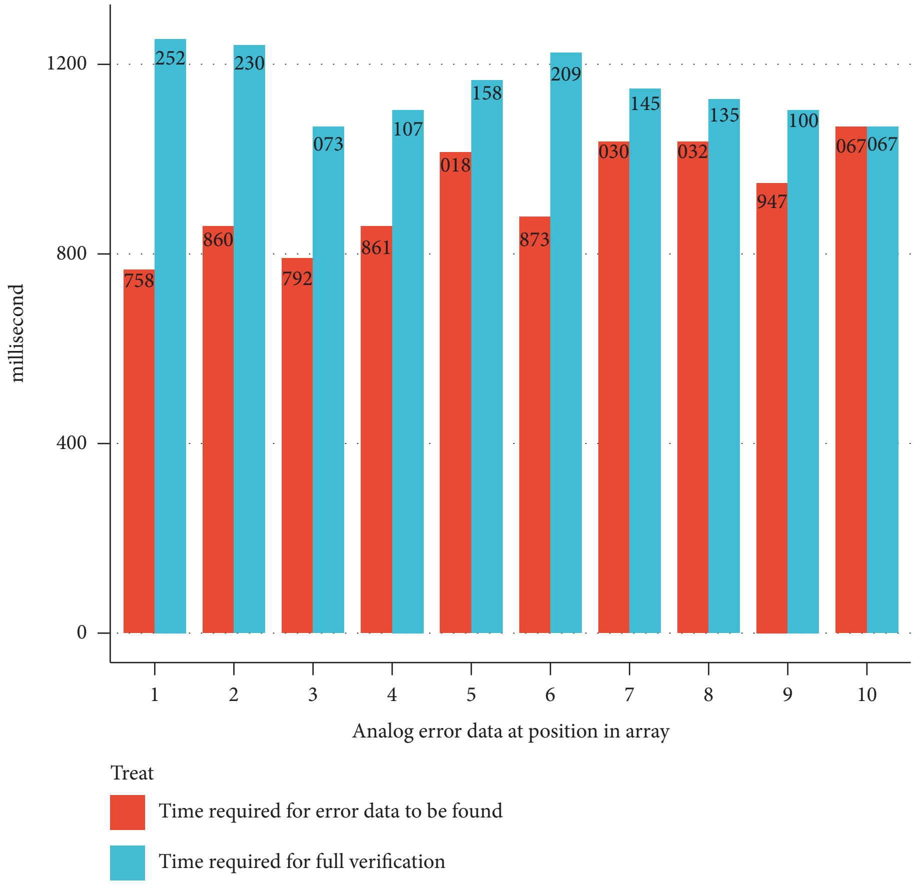

Figure 8: Experimental data of algorithm I.

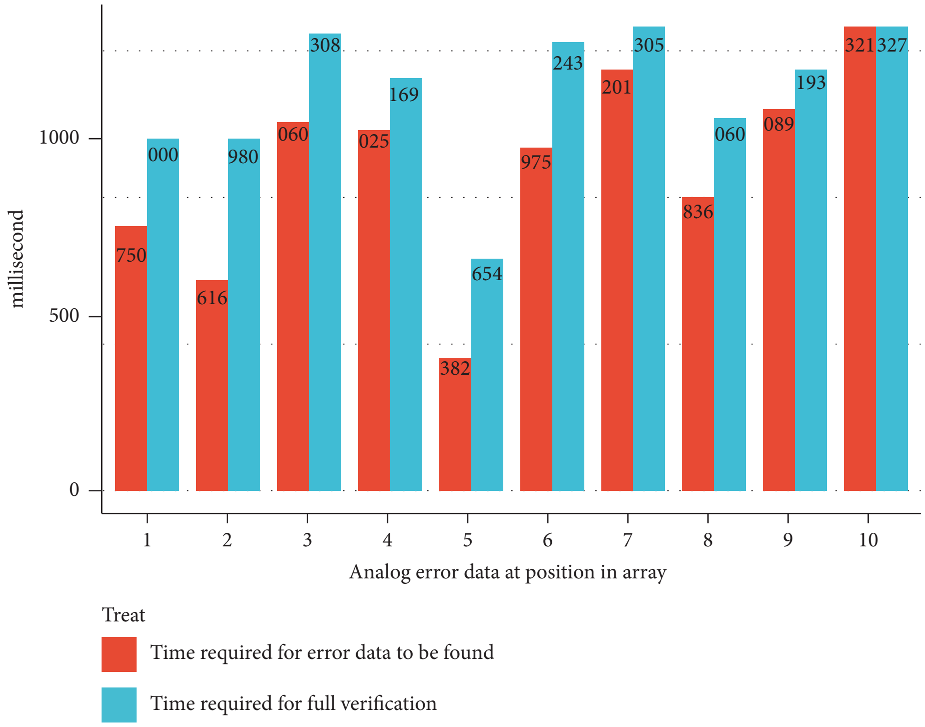

Figure 9: Experimental data of algorithm II.

Table 1: Comparison of security and functionality features.

Protocol	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}
[11]	√	√	√	√	×	√	√	√	√	√	√
[15]	×	√	√	×	√	√	√	×	√	√	√
[21]	√	×	√	√	×	×	×	×	×	×	×
[22]	×	√	√	√	×	×	√	×	×	×	×
[12]	×	√	√	×	×	√	√	√	√	√	√
[23]	√	√	√	×	×	×	×	×	×	×	×
[14]	×	√	√	×	×	×	√	√	×	√	√
[24]	√	×	√	√	×	×	×	×	×	×	×
[13]	×	√	√	√	√	√	√	√	√	×	×
[25]	√	×	√	√	×	×	×	×	√	×	×
[26]	√	√	×	√	×	×	√	√	×	×	×
Ours	√	√	√	√	√	√	√	√	√	√	√

target authentication. As the number of authentication targets increases, authentication efficiency increases.

Finally, we analyze the error data source detection algorithm mentioned in the paper through experiments. In our experiments, data storage is done sequentially and the storage structure is an array. Figures 8 and 9 show that when the fixed data size is ten and the first detection algorithm is used, the time required to detect the source of the wrong data has little to do with the position in the array. However, using the second detection algorithm, the source of the erroneous data in the intermediate position can be detected relatively quickly. Of course, the detection algorithm in this paper may not be perfect, and we will gradually adjust and improve the detection efficiency in the follow-up work.

6.4. Safety and Function Comparison. We will make a comparative analysis from the following aspects:

(i) A_1: one-way authentication to improve efficiency

(ii) A_2: anonymous protection

(iii) A_3: strong replay attack

(iv) A_4: effectively defend against DDoS attacks

(v) A_5: confidentiality before providing

(vi) A_6: resist key leakage simulation attack

(vii) A_7: formal safety certificate

(viii) A_8: formal security verification

(ix) A_9: shared session key

(x) A_{10}: man in the middle attack

(xi) A_{11}: simulated attack

The specific results are shown in Table 1.

7. Conclusion

The communication security of marine devices is studied in the marine open network environment. First, identity-based encryption and symmetric key encryption are used to ensure the confidentiality of data. Then, we improve identity-based encryption considering the Practical application scenarios. A secret sharing-based multitarget authentication and key exchange protocol are proposed for identity authentication and computing session keys. The subsecret information is embedded in the identity credentials of the device, and the server obtains the subsecret information of the device while decrypting the information. After that the server uses subsecret information from multiple devices to recover the secret, thus enabling multitarget authentication. Next, the device's subsecret information is dynamically adjusted after successful authentication. The session key is calculated using the subsecret information. A subsecret allocation policy is designed to achieve dynamic adjustment of subsecret information and single-target authentication. Also, complex cryptographic computations are assigned to the server, considering that the client is a resource-constrained device. The protocol is shown to protect the privacy and resist impersonation, man-in-the-middle, and replay attacks.

Due to the complexity of the marine network environment, there are some shortcomings in this paper. We will further improve the scheme in the next work. The system is divided into two different communication ways in the marine communication architecture (Figure 1): (1) communication between devices with weak performance and devices with strong performance and (2) communication between devices with the same performance. Among them, the communication from weak-performance devices to strong-performance devices is divided into two different communication ways: (1) communication between relay devices and buoy devices and (2) communication between shore-based servers and relay devices. Because it is difficult for all buoy devices to transmit data synchronously, the relay device temporarily stores data. The relay devices are then transmitted uniformly to the shore-based server for authentication and decryption. Based on this idea, multitarget authentication is proposed. Next, we will study a lightweight authentication algorithm to solve the authentication problem between the relay device and the buoy device. Then, we will consider solving the transmission problem between the relay device and the relay device. The techniques mentioned above implement the whole systemic.

Data Availability

The experiment data used to support the findings of this study are currently under embargo while the research findings are commercialized. Requests for data, 12 months after the publication of this article, will be considered by the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study is supported by the National Natural Science Foundation of China (no. 62162020).

References

[1] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612-613, 1979.

[2] G. R. Blakley, “Safeguarding cryptographic keys,” *Managing requirements knowledge*, Vol. 313, IEEE, , New York, NY, USA, 1899.

[3] Y. S. Wu, C. C. Thien, and J. C. Lin, “Sharing and hiding secret images with size constraint,” *Pattern Recognition*, vol. 37, no. 7, pp. 1377–1385, 2004.

[4] C. T. Li and M. S. Hwang, “An online biometrics-based secret sharing scheme for multiparty cryptosystem using smart cards,” *Network*, vol. 3, no. 4, p. 5, 2010.

[5] M. Al-Ghamdi, M. Al-Ghamdi, and A. Gutub, “Security enhancement of shares generation process for multimedia counting-based secret-sharing technique,” *Multimedia Tools and Applications*, vol. 78, no. 12, pp. 16283–16310, 2019.

[6] T. Makino, Y. Kamidoi, and S. Wakabayashi, “A verifiable secret sharing scheme without using multi-party computations,” in *Proceedings of the 2020 IEEE44th Annual Computers, Software, and Applications Conference(COMPSAC)*, pp. 845–850, IEEE, Madrid, Spain, July 2020.

[7] C. Tang and Z. Yao, “A new(tn)-threshold secret sharing scheme,” in *Proceedings of the 2008 International Conference on Advanced Computer Theory and Engineering*, pp. 920–924, lEEE, Washington, DC, USA, May 2008.

[8] L. Harn, Z. Xia, C. Hsu, and Y. Liu, “Secret sharing with secure secret reconstruction,” *Information Sciences*, vol. 519, pp. 1–8, 2020.

[9] Z. Ma, Y. Ma, X. Huang, M. Zhang, and Y. Liu, “Applying cheating identifable secret sharing scheme in multimedia security,” *EURASIP Journal on lmage and Video Processing*, vol. 2020, no. 1, pp. 1–10, 2020.

[10] V. Goyal, Y. Song, and A. Srinivasan, “Traceable secret sharing and applications,” in *Proceedings of the Annual International Cryptology Conference*, pp. 718–747, Springer, Santa Barbara, CA, USA, August 2021.

[11] H. Zhang, W. Bian, B. Jie, D. Xu, and J. Zhao, “A complete user authentication and key agreement scheme using cancelable biometrics and PUF in multi-server environment,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 5413–5428, 2021.

[12] J. Srinivas, A. K. Das, M. Wazid, and N. Kumar, “Anonymous lightweight chaotic map-based authenticated key agreement protocol for industrial Internet of Things,” *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 6, pp. 1133–1146, 2020.

[13] D. Abbasinezhad-Mood, S. M. Mazinani, M. Nikooghadam, and A. Ostad Sharif, “Efficient provably-secure dynamic ID-based authenticated key agreement scheme with enhanced security provision,” *IEEE Transactions on Dependable and Secure Computing*, vol. 1, Article ID 3024654, 2020.

[14] L. Wei, J. Cui, Y. Xu, J. Cheng, and H. Zhong, “Secure and lightweight conditional privacy-preserving authentication for securing traffic emergency messages in VANETs,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1681–1695, 2021.

[15] H. Boyapally, P. Mathew, S. Patranabis, U. Chatterjee, U. Agarwal, and M. Maheshwari, “Safe is the new smart: PUF-based authentication for load modification-resistant smart meters,” *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 1, pp. 663–680, 2020.

[16] C. Meshram, R. W. lbrahim, S. G. Meshram, A. L. Imoize, S. S. Jamal, and S. K. Barve, “An efficient remote user authentication with key agreement procedure based on convolution-Chebyshev chaotic maps using biometric,” *The Journal of Supercomputing*, vol. 20, pp. 1–23, 2022.

[17] N. Saxena, H. Shen, N. Komninos, K. K. R. Choo, and N. S. Chaudhari, “BVPSMS:A batch verification protocol for end-to-end secure SMS for mobile users,” *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 3, pp. 1–565, 2018.

[18] G. S. Poh, P. Gope, and J. Ning, “PriHome: privacy-preserving authenticated communication in smart home environment,” *lEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1095–1107, 2019.

[19] J. Alawatugoda and T. Okamoto, “Standard model leakage-resilient authenticated key exchange using inner-product extractors,” *Designs, Codes and Cryptography*, vol. 90, no. 4, pp. 1059–1079, 2022.

[20] R. Klisters and D. Rausch, “A framework for universally composable Diffe-Hellman key exchange,” in *Proceedings of the 2017 IEEE Symposium on Security and Privacy(SP)*, pp. 881–900, IEEE, San Jose, CA, USA, June 2017.

[21] M. L. Das, A. Saxena, and V. P. Gulati, “A dynamic ID-based remote user authentication scheme,” *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 629–631, 2004.

[22] D. Mishra, “Efficient and secure two-factor dynamic ID-based password authentication scheme with provable security,” *Cryptologia*, vol. 42, no. 2, pp. 146–175, 2018.

[23] Y. P. Liao and S. S. Wang, “A secure dynamic ID based remote user authentication scheme for multi-server environment,” *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 24–29, 2009.

[24] M. K. Khan, S. K. Kim, and K. Alghathbar, “Cryptanalysis and security enhancement of a ‘more efficient & secure dynamic ID-based remote user authentication scheme,” *Computer Communications*, vol. 34, no. 3, pp. 305–309, 2011.

[25] F. Wen and X. Li, “An improved dynamic ID-based remote user authentication with key agreement scheme,” *Computers & Electrical Engineering*, vol. 38, no. 2, pp. 381–387, 2012.

[26] S. Jangirala, S. Mukhopadhyay, and A. K. Das, “A multi-server environment with secure and efficient remote user authentication scheme based on dynamic ID using smart cards,” *Wireless Personal Communications*, vol. 95, no. 3, pp. 2735–2767, 2017.