Journal of Multimedia Information System VOL. 10, NO. 3, September 2023 (pp. 237-248): ISSN 2383-7632 (Online)
https://doi.org/10.33851/JMIS.2023.10.3.237

Secure and Lightweight Authentication Protocol in Internet of Things

Yanlong Yang[1], Mengzhu Lu[1*], Xiaohan Niu[2]

Abstract

The further development of Internet of things (IoT) makes the number of various terminal devices grow rapidly. At the same time, the amount of data collected and transmitted through terminal devices is also increasing. However, in the communication between devices and servers, most of them lack efficient identity authentication and encrypted communication mechanisms suitable for IoT environment. Therefore, in order to secure the communication between these devices and servers, they need to be protected by password technology. This paper proposes a secure communication protocol based on chaotic mapping algorithm, which is used to ensure the bidirectional identity authentication and data encryption between IoT devices and servers. The protocol is proved by Burrows Abadi Needham(BAN) logic, Scyther and informal security analysis that it satisfies the security and can resist various security attacks, and achieve anonymity and non-traceability. Finally, the performance comparison analysis with similar protocols shows that the proposed protocol significantly improves the security and has high efficiency.

Key Words: IoT, Chaotic Mapping Algorithm, Authentication, Security, Protocol.

I. INTRODUCTION

With the progress of IoT, in the near future, all hardware devices in the world will be networked [1]. IoT is a huge network composed of computer equipment and hardware equipment connected to the Internet. Over the years, people's lives have undergone earth shaking changes with the application and development of IoT, specifically in making work and life more convenient [2]. It is widely used in daily life. In terms of home furnishing, smart home can enable individual users to remotely control the working state of home appliances, provide individuals with an intelligent life mode, and help people effectively arrange time and save energy [3]. In medical treatment, people can install node sensors on patients to monitor the indexes of patients' bodies. When the indexes are abnormal, people can automatically call for help; In terms of traffic, communication between vehicles and roadside units can enable vehicles and roadside units to grasp the driving conditions of other vehicles in time, adjust their driving conditions in time, and achieve the purpose of optimizing traffic conditions. Internet of Things realizes the intel-ligence and automation of the system through the inter-connection between sensors and controllers. The appli-cation of related technologies lays a good foundation for the popularization of IoT. In addition to the above functions, IoT has been widely used in agriculture, medical care, energy, Internet of Vehicles and other fields, playing an indispensable role in urban intelligence. In the common Internet of Things, there are generally two types of entities. They are sensor nodes and server nodes. Sensor nodes collect data and execute the received instructions. Server node uploads the data and sends instructions, and performs a series of operations. Whether users are in the system deployment range or far away, they can control by sending instructions to nodes after completing identity authen-tication [4].

The Internet of Things technology is developing rapidly, but its security issues have also attracted widespread attention. So far, there is no unified industry standard for the Internet of Things industry, and a large number of IoT devices and systems in the market have security risks [5]. Although the Internet of Things has brought great convenience to our lives, once its security vulnerabilities are exploited by criminals, the losses to individuals, in-dustries, and even the entire society are enormous. The Internet of Things has its own unique architecture, and what sets it apart from the general Internet is the perception layer. Combining the unique security threats of the Internet of Things, it can be seen that the biggest issue with Internet of Things security is also the perception layer security. The IoT perception layer is generally used to collect data required by servers to achieve fixed functions and transmit it

Manuscript received May 29, 2023; Revised July 02, 2023; Accepted July 09, 2023. (ID No. JMIS- 23M-05-023)
Corresponding Author (*): Mengzhu Lu, +86-13514257803, lu20232023@126.com
[1]Department, Handan Polytechnic College, Handan, China, yang20230529@126.com, lu20232023@126.com
[2]Department, Northwest University, Xian, China, niuxiaohan@stumail.nwu.edu.cn

after simple processing. The perception layer mainly consists of various terminal devices, such as sensors, gateways, RFID readers, intelligent terminal devices, etc. The first step in realizing the functionality of the Internet of Things network is to use these terminal devices to collect data, so the perception layer is also the foundation of the Internet of Things [6]. Without the work of these devices, the Internet of Things cannot be discussed. However, in the general Internet of Things, the number of terminal devices is huge, and in practical applications, it does not require too much manpower and resources to supervise them. Therefore, from a physical level, its security management is weak. At the communication level, these devices are also very susceptible to various attacks such as impersonation, interception, Distributed Denial of Service (DDoS), replay, etc. Due to the lack of identity verification on terminal devices, the vast majority of IoT devices operate in a white box environment, resulting in extremely fragile security around the IoT [7]. Attackers use these issues to continuously launch attacks on IoT devices, and once successful, it will cause huge losses to businesses and even society. Therefore, solving the authentication and communication security issues of terminal devices is a key step in achieving IoT security [8].

While IoT facilitates People's Daily life, its security has not been paid enough attention to for a long time. Now, after being threatened by all aspects and suffering heavy losses, people pay more and more attention to the security of IoT [9]. Compared with traditional Internet, the security of IoT has certain similarities. The protection elements of the Internet of things are still usability and confidentiality. As the increasing scale of terminal access devices and data, some criminals launch malicious attacks by finding, utilizing or controlling Internet of things devices with security vulnerabilities [10]. Therefore, security issues are not only related to the interests of users, but also have a great impact on economic development, social stability and national security [11]. Here, we need to use cryptographic theory and technology to build a scheme to provide a comprehensive security mechanism. However, in the environment of IoT, a large number of devices are con-strained [12]. Due to their own limitations, their computing power and storage capacity are relatively limited, and they are very vulnerable to enemy attacks [13]. In order to provide appropriate security protection for these restricted devices, we need to design appropriate security schemes to encrypt their privacy and provide verification for transmitted information based on the authentication of both sides of the communication, so as to resist various types of attacks. Therefore, we propose a secure authentication protocol for IoT, which is based on chaotic mapping algorithm. In the interaction, the communication entities need to authenticate and negotiate the key, through which they can communicate securely. The main contributions are arranged as follows:

1) In view of the problem that the authentication process of Internet of things devices divulges users' privacy, a lightweight device identity authentication protocol is proposed by using chaotic mapping algorithm technology. This scheme provides device anonymity protection and secure negotiation of session key to ensure the subsequent secure communication.
2) It is proved by BAN logic Scyther and informal security analysis that the protocol satisfies anonymity, and realizes mutual authentication and secure session key negotiation.
3) Through the experiment, the protocol can not only ensure the security of identity authentication, but also has less overhead compared with other protocols.

The rest of this paper is structured as follows. Sections 2 and 3 mainly focus on relevant research and preliminaries. Section 4 is about the details of the protocol we propose. Section 5 mainly verifies the security of the protocol. Section 6 mainly analyzes the protocol. Section 7 summarizes the whole paper.

II. LITERATURE REVIEW

Recently, some researchers have conducted extensive research on user identity authentication schemes in the Internet of things.

Sahingoz et al. [13] proposed a multi-level dynamic key agreement protocol, which was based on asymmetric key negotiation and ECC password. In the protocol, each device negotiates with adjacent nodes to complete the verification and signature of data, which increased the amount of calculation, leaded to serious energy loss of nodes, and shortened
the service life of nodes. Liao and Hsiao proposed a security authentication scheme based on elliptic curve [15], which integrated the identity authentication server, but Peeters and Hermans [16] pointed out that it was vulnerable to server simulation attacks. Kalra and Sood [17] pointed out in their paper that the embedded devices in IoT and cloud server cannot support complex encryption algorithms and had limited storage capacity. To solve the above problems, using elliptic curve cryptography, they designed a scheme for two-way authentication and key negotiation between Internet of things devices and cloud server. This scheme used cookies to realize the authentication of identity legitimacy of intelligent devices. However, Chang et al. [18] proved that Kalra's scheme had two security defects, that is, it did not realize the mutual authentication, and the key was fuzzy. They improved the original scheme and

proposed a more complete and secure improvement scheme. The server verified the legitimacy of the server to the device by generating secret values for the device in the device registration stage, Using elliptic curve to realize key negotiation solved the problem of session key ambiguity. Liao and Wang [19] proposed a shared key protocol. Because there are many servers that provide services for users, identity authentication schemes needed authentication protocols applied to multi server environments. Vaidya et al. proposed a device authentication mechanism in smart home environment, which can meet known security features such as forward security and key leakage attacks, but the security of this mechanism had not been demonstrated in the paper [20]. Li designed a secure authentication protocol in the Internet of things environment, which used packet monitoring and a control record to simplify the authenticat ion process, and can establish session keys between communication entities [21]. However, the protocol cannot resist simulation attacks, and realize the anonymity of user identity. Han et al. proposed a secure authentication protocol in smart home system, which realized two-way authentication between smart devices and gateways [22]. However, for devices with limited resources, the protocol required large overhead. Fabian et al. designed a distributed hash table to realize the anonymity of intelligent devices and prevent privacy disclosure [23]. Chaudhry et al. [24] proposed an authentication protocol based on ECC. The protocol completed the identity authentication process through exchanging two messages. Abbasinezhad-Mood and Nikooghadam [25] proposed an anonymous self-authenticated key distribution protocol based on elliptic curve cryptography. Kim et al. proposed a dynamic and effective authentication scheme in the Internet of things environment, which realized the dynamic selection of authentication policy and reduced the computational cost of the protocol [26]. However, the protocol still cannot realize the anonymity of user identity and the non connection between users. Medaglia and Serbanati [28] listed and explained the problems faced by user privacy protection and encrypted communication in the Internet of Things, and gave improvement suggestions for the listed problems from the perspective of short term and long term respectively. The technologies in the short-term plan were mostly existing or under study, while those in the long-term plan remained in the conceptual stage temporarily. Fernandes et al. [29] directly conducted a series of tests and analyses on SmartThings, Samsung's smart home application system. In addition, other experts have analyzed and summarized the security environment of different specific application environments [30]. Vaidya et al. [31] proposed a password-based remote user authentication protocol, but their protocol did not provide two-way authentication between entities, and the scheme was vulnerable to smart card theft attacks. Wazid et al. [32] proposed a multi-factor user remote authentication and key negotiation protocol, which introduced a tool called biological key. Users needed to provide passwords and personal biological templates at the same time to complete identity authentication, and regarded the gateway node as a secure device that could not be stolen, which was somewhat different from the actual application. Kumar et al. [33] designed a secure key protocol suitable for smart home. This protocol can provide mutual authentication between two communication parties and establish temporary communication keys. However, this protocol only gave a general framework.

III. PRELIMINARIES

3.1. System Model

As shown in Fig. 1, the protocol authentication system model is mainly composed of the following entities: Trusted third parties (TTP), Server and IoT devices (IoTD). TTP has enough computing and interaction capabilities to complete any complex task. In this agreement, TTP mainly generates the public parameters, the keys of IoTD and Server, and It can be trusted. Server can verify the information sent by IoTD and negotiate a session key. IoTD is the device that can provide some services, which can be various types of sensors, and need to reach mutual authentication with the server.

3.2. Threat Model

This paper will adopt the universally considered threat model in the IoT security threat model [11]. Like most authentication schemes, this paper first considers the Dolev-Yao threat model [27]. In this threat model, the participants in the key agreement scheme communicate on

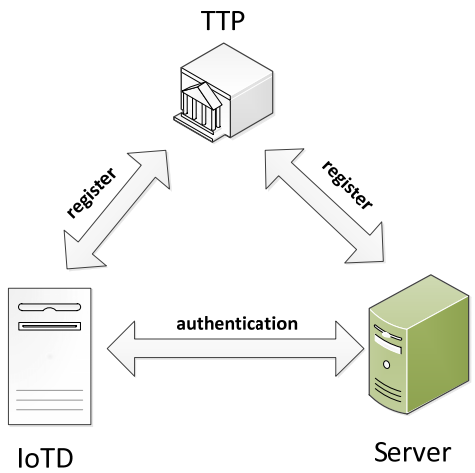

Fig. 1. System model.

insecure channels. Opponents can eavesdrop on all messages in communication and modify or delete them after interception. Opponents can send modified or directly forged information to participants in order to obtain favorable returns. Opponents can shape themselves as legitimate participants in key negotiation and actively attempt to participate in session key negotiation.

3.3. Security Requirements

Due to its own characteristics, the Internet of things faces a variety of attacks, so the design of Internet of things authentication scheme faces many challenges. Generally, a relatively perfect IoT authentication mechanism should have these attributes [8].

1) Mutual authentication. The authentication scheme must meet the requirements that two entities in the communication can mutually verify the legitimacy of each other's identities.
2) User anonymity. Because communication in an open environment, the attacker may track the activities of legitimate personnel to steal the identity of the device. Therefore, the designed authentication scheme of the Internet of things should protect the secret identity information of the device and provide anonymity
3) Forward security. An attacker may acquire the long-term key and use it to try to retrieve key. Therefore, it is necessary to ensure that even if the key is leaked, it will not lead to the session key leakage of historical communication.
4) Anti-attack. Anti-attack requires that the protocol can resist common network attacks such as replay attack, counterfeiting attack, etc.
5) Availability. A reasonable security authentication scheme should have the characteristics of energy saving. The scheme design should avoid complex computing as much as possible, reduce computing overhead and communication consumption, so that it can be truly applied in the Internet of things with limited resources.
6) Security key update: The session key is used to ensure data security and needs to be updated after each data transmission session. In addition, it is also necessary to properly manage the security keys to achieve key synchronization between both parties.
7) Data confidentiality and integrity: The transmitted data needs confidentiality protection and integrity protection.

3.4 Hash Function

Hash function is a class function that can map data of different lengths to fixed length data. The output fixed length data is called hash value. Because hash function is sensitive, minor changes to the original data will lead to huge changes in its hash value. It can be regarded as the only fingerprint of the original data, and it has an irreversible nature. Therefore, hash function is usually used to ensure the integrity of data and conduct authentication. A good hash function should have the following characteristics.

1) Fixed length. This is the basic property of the hash function, that is, it can convert an input of a certain length into a hash of a fixed length. In most cases, the input length is greater than the output length.
2) Pseudorandomness. If the hash function does not have good pseudo randomness, the probability of collision will increase, which will lead to some hash values being more likely to appear than other hash values, and the cost of handling collisions will increase. Therefore, a good hash function should have good pseudorandomness.
3) Easy to calculate. This is easy to ignore, but only when the hash function is fast enough can the efficiency of encryption and other operations be guaranteed.
4) Collision resistance. The most ideal hash function is completely collision free, that is, any hash value has only one input corresponding to it. But in general, considering the actual application scenarios, we only need a hash function with minimal collision probability.

Generally, the algorithms corresponding to hash function are public. This property also makes hash function popular in many fields, but attacks against different hash function also occur. The current attack against the hash function is mainly a hash collision attack, which is essentially a denial of service attack. This attack is aimed at constructing data according to the characteristics of the hash function, so that through the calculation of the hash function, all data have the same hash value. When these data are saved in the hash table, the hash table will become a single linked table due to a large number of collisions, Thus, the time complexity of various operations on the hash table is greatly increased to achieve the goal of preventing the system from responding quickly.

3.5. Chebyshev Chaotic Mapping

Given an integer n and a real number x with a value in the range of $[-1,1]$, we define the Chebyshev polynomial as a mapping such as T_n: $[-1, 1]\rightarrow[-1, 1]$, where: $T_n(x) = \cos(narccos(x))$. The polynomial recursive relationship between different orders is defined as: $T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), T_0(x) = 1, T_1(x) = x, n \geq 2$. In addition, it also has the properties [11].

1) Chaos attribute. When n > 1, the Chebyshev polynomial map T_n: [−1, 1]→[−1, 1] has an invariant function density for the positive Lyapunov exponent $\lambda = \ln n > 0$.
2) Semigroup attribute. For two positive integers (*r*,*s*) and $x \in [-1,1]$, there is: $T_r(T_s(x)) = T_s(T_r(x))$.

Zhang extended this property to the real number field $(-\infty, +\infty)$: $T_n(x)=(2xT_{n-1}(x)-T_{n-2}(x)) \ mod \ p$ where when $n \geq 2, x \in [-1,1]$, p is a large prime number, the Chebyshev polynomial still satisfies this property $T_r(T_s(x)) \equiv T_{rs}(x) \equiv T_s(T_r(x)) \ mod \ p$.

IV. THE PROPOSED SCHEME

The design details of the authentication protocol will be described, and the symbols and their meanings used will be introduced. Here, The specific symbols and their meanings are shown in Table 1, and the specific protocol flow is shown in Fig. 2.

4.1. System Setup

TTP selects two integers randomly x and k and P is a large prime number selected by TTP. TTP calculates $T_K = T_k(x) \ mod \ P$, where k is the secret private key. TTP selects two secure hash functions: $H_1 : \{0,1\}^* \times \{0,1\}^* \rightarrow Z^*_q$, $H_2 : \{0,1\}^* \rightarrow Z^*_q$. Finally, TTP releases the system parameters $\{x, T_k(x), P, \ H_1, H_2\}$.

4.2. Registration

4.2.1. Server Registration

First, the server S_j randomly selects a secret value $r_j \in Z^*_q$ and the real identity ID_j. Then it sends the message $\{ ID_j, r_j \}$ to TTP securely. When the TTP gets this information, it needs to be verified whether the identity has been registered. If you have already registered, refuse to register, otherwise continue. Then TTP calculates the $TID_j = H_1(ID_j, r_j, k)$, stores $\{ID_j, TID_j, r_j\}$ in the database, and sends the message $\{ TID_j \}$ to S_j. S_j stores the $\{ID_j, TID_j, r_j\}$ to the database.

Table 1. BAN logic rules.

Rule	Description
P ◁ X	P receives a message containing X
P|~X	P sends a message containing X
P | ≡ X	P believes X
P $\overset{K}{\leftrightarrow}$ Q	P and Q share secret K
< X >$_Y$	X contains the secret Y
P ⇒ X	P has the right to decide whether X is right or not
Message meaning rule	$\frac{P \mid\equiv P \overset{K}{\leftrightarrow} Q, P \lhd <X>_Y}{P \mid\equiv Q \mid\sim X}$
Belief rule	$\frac{P \mid\equiv X, P \mid\equiv Y}{P \mid\equiv (X, Y)}$
Nonce verification rule	$\frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$
Arbitration rule	$\frac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\sim X}{P \mid\equiv X}$

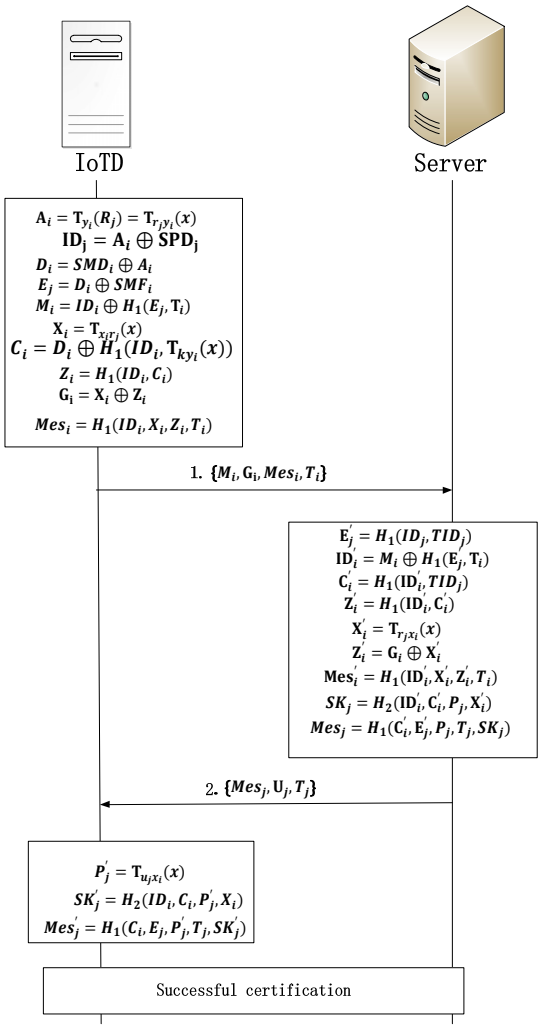

Fig. 2. Certification process.

4.2.2. IoTD Registration

1) IoTD$_i$ first selects the identity information ID_i and randomly selects $y_i \in Z^*_q$, calculates $Y_i = T_{y_i}(x)$, and sends $\{ID_i, Y_i\}$ to TTP securely.

2) When getting the message, TTP will query whether the identity exists. If this identity already exists, it indicates that it has been registered, and the request is rejected, otherwise continue. TTP calculates $R_j = T_{r_j}(x), C_i = H_1(ID_i, TID_j, r_j)$, E_j=H_1(ID_j, TID_j, r_j), D_i=$C_i \oplus H_1$(ID_i, E_j, T_{rjyi}(x)). Then TTP sends $\{ID_j, D_i, E_j, R_j\}$ to IoTD$_i$ through secure channel.

3) IoTD_i calculates $\mathrm{SPD_i} = ID_j \oplus H_1(ID_i, \mathrm{T}_{r_j y_i}(x))$, $SMD_i = D_i \oplus H_1(ID_i, ID_j, y_i)$, $SMF_i = D_i \oplus E_j$, and finally stores the value $\{\mathrm{SPD_i}, SMD_i, SMF_i, R_j\}$ into memory.

4.3. Authentication Phase

When IoTD_i and S_j communicate, IoTD_i and S_j need to authenticate and establish a session key.

1) IoTD_i first calculates $\mathrm{A}_i = \mathrm{T}_{y_i}(R_j) = \mathrm{T}_{r_j y_i}(x)$, then $\mathrm{ID_j} = \mathrm{H}_1(ID_i, \mathrm{A}_i) \oplus \mathrm{SPD_I}$, $D_i = SMD_i \oplus H_1(ID_i, ID_j, y_i)$, $E_j = D_i \oplus SMF_i$, $C_i = D_i \oplus H_1(ID_i, E_j, \mathrm{A}_i)$. Next, IoTD_i generates the current timestamp T_i and calculates $M_i = ID_i \oplus H_1(E_j, \mathrm{T}_i)$. IoTD_i selects a new secret value $x_i \in Z_q^*$ and calculates $\mathrm{X}_i = \mathrm{T}_{x_i r_j}(x)$ $Z_i = H_1(ID_i, C_i)$, $\mathrm{G_i} = \mathrm{X}_i \oplus \mathrm{Z}_i$, $Mes_i = H_1(ID_i, X_i, Z_i, T_i)$. Finally, IoTD_i sends message $\{M_i, \mathrm{G_i}, Mes_i, T_i\}$ to S_j.

2) When S_j receives the information, S_j checks the correctness of the timestamp T_i. If it is not legal, it refuses to authenticate. Otherwise, it calculates $\mathrm{E}'_j = H_1(ID_j, TID_j, r_j)$, $\mathrm{ID}'_i = M_i \oplus H_1(\mathrm{E}'_j, \mathrm{T}_i)$, $\mathrm{C}'_i = H_1(\mathrm{ID}'_i, TID_j, r_j)\mathrm{Z}'_i = H_1(\mathrm{ID}'_i, \mathrm{C}'_i)$, $\mathrm{X}'_i = \mathrm{T}_{r_j x_i}(x)$, $\mathrm{Z}'_i = \mathrm{G}_i \oplus \mathrm{X}'_i$, $\mathrm{Mes}'_i = H_1(\mathrm{ID}'_i, \mathrm{X}'_i, \mathrm{Z}'_i, T_i)$, and then checks whether the values of Mes'_i and Mes_i. If their values are the same, IoTD_i is authenticated. Then, S_j generates a secret random number $u_j \in Z_q^*$ and timestamp T_j, calculates $\mathrm{U}_j = \mathrm{T}_{u_j}(x), P_j = \mathrm{T}_{u_j x_i}(x)$ and session key $SK_j = H_2(\mathrm{ID}'_i, \mathrm{C}'_i, P_j, \mathrm{X}'_i)$, $Mes_j = H_1(\mathrm{C}'_i, \mathrm{E}'_j, P_j, T_j, SK_j)$, and then sends the message $\{Mes_j, \mathrm{U}_j, T_j\}$ to IoTD_i.

3) When IoTD_i receives the information from S_j, IoTD_i checks the correctness of the timestamp T_j. If it is not, it will refuse authentication. Otherwise, IoTD_i calculates $P'_j = \mathrm{T}_{u_j x_i}(x)$, $SK'_j = H_2(ID_i, C_i, P'_j, X_i)$, $Mes'_j = H_1(C_i, E_j, P'_j, T_j, SK'_j)$ and verifies whether the messages Mes'_j and Mes_j are the same. If two values are the same, IoTD_i verifies S_j and the generated session key are also equal. At this time, IoTD_i and S_j can communicate through the session key.

V. SECURITY EVALUATION

5.1. Security Proof

BAN logic is a modal logic based on agent knowledge and belief reasoning [11]. In this logic, a series of artificial symbols are specified to form a formal marking method to describe protocol messages, protocol assumptions, inference rules and subject beliefs. BAN logic, under the action of axioms and inference rules, derives the subject's belief from protocol assumptions and protocol messages to judge whether the protocol meets the set goal. This section describes the use of BAN logic to verify the security.The logic rules are shown in Table 1.

The following describes the specific steps:

1) Idealized form of protocol

$\mathrm{M}_1: \mathrm{IoTD}_i \rightarrow \mathrm{S}_j: \{M_i, \mathrm{G_i}, Mes_i, T_i\}$

$\mathrm{M}_2: \mathrm{S}_j \rightarrow \mathrm{IoTD}_i: \{Mes_j, \mathrm{U}_j, T_j\}$

2) Protocol goal

$G_1: \mathrm{IoTD}_i \mid \equiv \mathrm{IoTD}_i \overset{SK'_j}{\leftrightarrow} \mathrm{S}_j$

$G_2: \mathrm{S}_j \mid \equiv \mathrm{S}_j \overset{SK_j}{\leftrightarrow} \mathrm{IoTD}_i$

$G_3: \mathrm{IoTD}_i \mid \equiv \mathrm{S}_j \mid \equiv \mathrm{S}_j \overset{SK_j}{\leftrightarrow} \mathrm{IoTD}_i$

$G_4: \mathrm{S}_j \mid \equiv \mathrm{IoTD}_i \mid \equiv \mathrm{IoTD}_i \overset{SK'_j}{\leftrightarrow} \mathrm{S}_j$

3) Protocol initialization assumptions

$A_1: \mathrm{S}_j \mid \equiv \mathrm{S}_j \overset{A_i}{\leftrightarrow} \mathrm{IoTD}_i$

$A_2: \mathrm{S}_j \mid \equiv \#(C_i)$

$A_3: \mathrm{S}_j \mid \equiv \mathrm{IoTD}_i \Rightarrow \{M_i, G_i, Mes_i, T_i\}$

$A_4: \mathrm{S}_j \mid \equiv \mathrm{S}_j \overset{P_j}{\leftrightarrow} \mathrm{IoTD}_i$

$A_5: \mathrm{S}_j \mid \equiv \#(P_j)$

$A_6: \mathrm{S}_j \mid \equiv \mathrm{IoTD}_i \Rightarrow \mathrm{S}_j \overset{SK_j}{\leftrightarrow} \mathrm{IoTD}_i$

$A_7: \mathrm{IoTD}_i \mid \equiv \mathrm{S}_j \overset{P_j}{\leftrightarrow} \mathrm{IoTD}_i$

$A_8: \mathrm{IoTD}_i \mid \equiv \#(X'_i)$

$A_9: \mathrm{IoTD}_i \mid \equiv \#(C'_i)$

$A_{10}: \mathrm{IoTD}_i \mid \equiv \mathrm{S}_j \Rightarrow \mathrm{S}_j \overset{SK'_j}{\leftrightarrow} \mathrm{IoTD}_i$

4) Proof of protocol

From M_1, we can get:

$R_1: S_j \lhd < M_i, G_i, Mes_i, T_i >$

From $\mathrm{R}_1, \mathrm{A}_1$ and message meaning rules, we get:

$R_2: S_j \mid \equiv IoTD_i \mid \sim < M_i, G_i, Mes_i, T_i >$

From $\mathrm{R}_2, \mathrm{A}_2$ and nonce number verification rules, we get:

$R_3: S_j \mid \equiv IoTD_i \mid \equiv \{M_i, G_i, Mes_i, T_i\}$

From $\mathrm{R}_3, \mathrm{A}_3$ and jurisdiction rules, we get:

$R_4: S_j \mid \equiv \{M_i, G_i, Mes_i, T_i\}$

From R_4, $\mathrm{R}_2, \mathrm{A}_4, \mathrm{A}_5$ and $\mathrm{SK_j} = \mathrm{H_2}(\mathrm{ID}'_\mathrm{i}, \mathrm{C}'_\mathrm{i}, \mathrm{P_j}, \mathrm{X}'_\mathrm{i})$, we get:

$R_5: S_j \mid \equiv IoTD_i \mid \equiv S_j \overset{SK_j}{\leftrightarrow} IoTD_i$

According to $\mathrm{R}_5, \mathrm{A}_6$ and jurisdiction rules, we get:

$R_6: S_j \mid \equiv S_j \overset{SK_j}{\longleftrightarrow} IoTD_i$

From the message M_2, we can get:

$R_7: IoTD_i \lhd < Mes_j, U_j, T_j >$

Through R_7, A_5 and message freshness rules, we get:

$R_8: IoTD_i \mid \equiv \# (Mes_j, U_j, T_j)$

From R_7, A_7 and message meaning rule, we get:

$R_9: IoTD_i \mid \equiv S_j \mid\sim \{ Mes_j, U_j, T_j\}$

From R_8 and random number verification rules, we get:

$R_{10}: IoTD_i \mid \equiv S_j \mid \equiv \{ Mes_j, U_j, T_j\}$

Through R_{10}, A_7, A_8, A_9 and $SK_j' = H_2(ID_i, C_i, P_j', X_i)$, we can get

$R_{11}: IoTD_i \mid \equiv S_j \mid \equiv S_j \overset{SK_j'}{\longleftrightarrow} IoTD_i$

According to R_{11}, A_{10} and jurisdiction rules, we get:

$R_{12}: IoTD_i \mid \equiv S_j \overset{SK_j'}{\longleftrightarrow} IoTD_i$

Through R_5, R_6, $R_{11}$and R_{12}, we can see that our protocol has reached the goals.

5.2. Scyther Verification

Scyther is widely used to verify protocol security [27]. The formal verification tool Scyther is an effective tool developed by C. Cremers based on model improvement algorithm and widely used to verify security protocols. The Scyther tool can effectively verify and analyze security protocols by characterizing protocols and generating limited representations of all possible protocol behaviors, and by detecting unlimited number of sessions, random numbers, and potential attacks and vulnerabilities of specific statements. At the same time, the tool can give explicit termination to protocols of unlimited sessions and infinite state sets. Because Scyther tool has a clear description, it can effectively help the protocol to conduct attack search. Scyther assumes that all encryption functions are perfect, that is, plaintext messages cannot be decrypted without knowing the key. The tool transforms the unlimited behavior space of the protocol into the limited output of the protocol.

In this paper, the formal analysis tool Scyther is used to formally analyze the proposed protocol. In the protocol, there are mainly two roles IoTD and Server. Scyther models the protocol and analyzes it using security statements. And the article explains different verification levels through the statements, including alive, weakagree, niagree, and nisynch. Alive and weakagree can resist MitM attack. Niagree and nisynch can resist replay attack and ensure forward and backward security.

The operation results after scyther analysis are shown in Fig. 3. From the analysis results, it can be concluded that This protocol satisfies the security requirements in IoT.

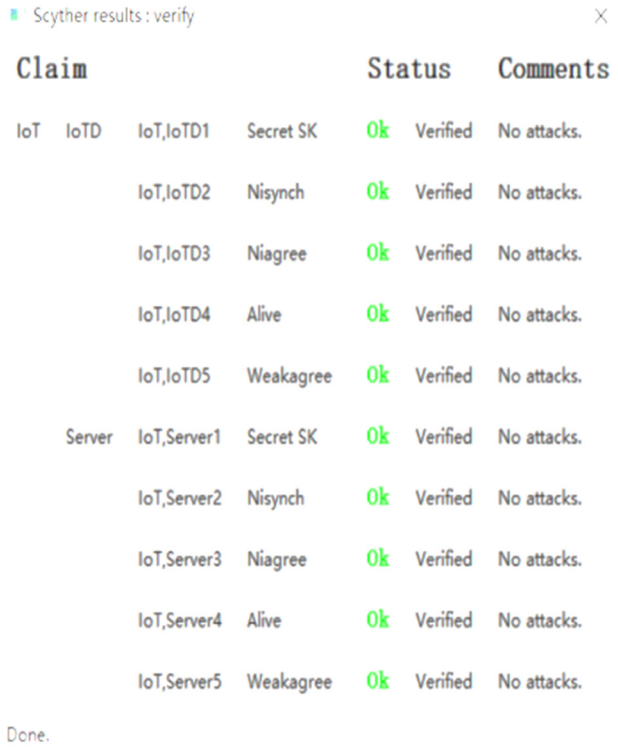

Fig. 3. Scyther result.

5.3. Security Analysis

Here, we analyze the security attributes.

1) Mutual Authentication. The scheme can realize mutual authentication between S_j and $IoTD_i$. $IoTD_i$ generates secret values $Mes_i = H_1(ID_i, X_i, Z_i, T_i)$ and sends them to S_j, which verifies their correctness and authenticates $IoTD_i$. S_j generates $Mes_j = H_1(C_i', E_j', P_j, T_j, SK_j)$ through secret value and sends it to $IoTD_i$. $IoTD_i$ verifies its correctness and authenticates S_j.
2) Anonymity. The scheme provides privacy identity protection of $IoTD_i$ identity. In our scheme, $IoTD_i$ generates a temporary identity $M_i = ID_i \oplus H_1(E_j, T_i)$. Because the timestamp generated by each authentication is different, the temporary identity of the device is different every time. And it is encrypted. Only the legal S_j can decrypt the real identity of $IoTD_i$.
3) Session Key Agreement. In the scheme, $IoTD_i$ and $S_j$negotiate the session key $SK_j = H_2(ID_i', C_i', P_j, X_i')$ through the chaotic mapping algorithm
4) Resist replay attack. The transmitted message $\{M_i, G_i, Mes_i, T_i\}$ and $\{Mes_j, U_j, T_j\}$ in the scheme contain a new timestamp and random number. If the receiver receives the corresponding message, it first verifies it to ensure that its message is fresh. Thus, the attack can be prevented.
5) Resistance to MitM attack. Both $IoTD_i$ and S_j generate corresponding messages with message very-

fication information, so IoTD_i and S_j will check the legal of the corresponding information when they receive them, which can effectively resist the attack.

6) Resistance to counterfeiting attacks. IoTD_i and S_j in the scheme are the effective information generated by their respective secret values, so the attacker cannot construct a legitimate message if he does not know the corresponding secret values, so the scheme can effectively resist counterfeiting attacks.
7) Forward security. IoTD_i and S_j in the scheme calculate $P_j = \mathrm{T}_{u_j x_i}(x)$ and $P_j' = \mathrm{T}_{u_j x_i}(x)$ to generate SK. And even if SK is leaked, SK negotiated in the next authentication process can be deduced. The forward security of the key is guaranteed.
8) Security key update. IoTD generates a new secret value x_i, generates a secret value $P_j' = \mathrm{T}_{u_j x_i}(x)$ through a chaotic mapping algorithm, and then generates a session key $SK_j' = H_2(ID_i, C_i, P_j', X_i)$. Server generates a new secret value u_j, generates a secret value $P_j = \mathrm{T}_{u_j x_i}(x)$ through the chaotic mapping algorithm, and then generates $SK_j = H_2(\mathrm{ID}_i', \mathrm{C}_i', P_j, \mathrm{X}_i')$, so the session key is secure. Then, when authentication is performed again, IoTD and Server need to regenerate new secret values and generate new secret values through chaotic mapping. Therefore, the session key they generate is different each time, which ensures the update of the session key.
9) Data integrity protection: An attacker can destroy the integrity of message data by intercepting, modifying, or inserting data exchanged by users. In the protocol, we can check the integrity of messages because the messages exchanged use the message authentication codes $\mathrm{Mes}_i' = H_1(\mathrm{ID}_i', \mathrm{X}_i', \mathrm{Z}_i', T_i)$ and $Mes_j' = H_1(C_j, E_j, P_j', T_j, SK_j')$.

VI. PERFORMANCE ANALYSIS

The primary performance metric we consider here is computing and communication costs to evaluate the associated performance of the solution.

6.1. Computation Overhead

In protocol authentication, computing efficiency is an important indicator of protocol communication performance. If the execution time of the protocol is short, it indicates that the protocol has good communication performance. In this section, two anonymous authentication protocols [23] and [24] are selected to compare with the proposed protocol in terms of implementation efficiency. First, we test the computing time of ECC point multiplication algorithm, ECC point addition algorithm, hash function and chaotic mapping algorithm respectively. Here, the results are T_{ECC}=2.46 ms, T_{ECA}=0.129 ms, T_H=0.04 ms and $T_C = 0.64$ ms. Since the initialization phase and registration phase of the protocol are executed only once, the operation efficiency of the authentication process is the main factor affecting the implementation efficiency of the protocol, so only the computation overhead of authentication process is discussed. In order to verify the calculated cost, we compare them with the scheme, as shown in Table 2 and Fig. 4. From the table and figure, we can see that our scheme has obvious advantages.

Table 2. Computation overhead.

Protocol	Computation overhead	Time
[23]	$8T_H + 9T_{ECC} + 2T_{ECA}$	22.718 ms
[24]	$9T_H + 9T_{ECC} + 2T_{ECA}$	22.758 ms
Our scheme	$13T_H + 5T_C$	3.72 ms

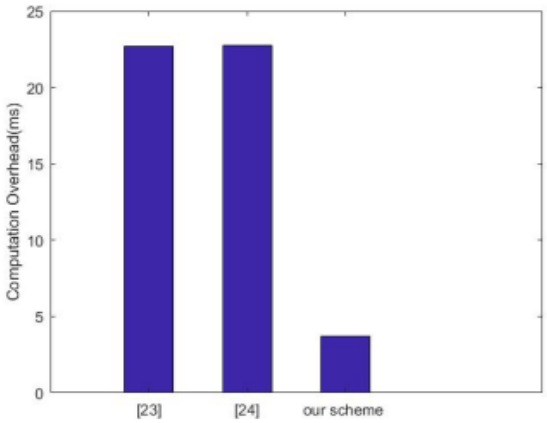

Fig. 4. Computation overhead.

6.2. Communication Overhead

Here we analyze the communication cost of our solution in the authentication stage. We assume that the size of ECC key is 256 bits, the hash function value is 128 bits, the identity information is 64 bits, the timestamp is 32 bits, and the size of chaotic mapping algorithm is 128 bits. In our protocol, IoTD and Server exchange two messages, namely $\{M_i, \mathrm{G_i}, Mes_i, T_i\}$ and $\{Mes_j, \mathrm{U}_j, T_j\}$. Through calculation, the communication overhead of $\{M_i, \mathrm{G_i}, Mes_i, T_i\}$ is 128+128+128+32=316 bits, and the communication overhead of $\{Mes_j, \mathrm{U}_j, T_j\}$ is 128+128+32=288 bits. Therefore, the total communication overhead is 960 bits. The communication overhead of similar schemes [23] and [24] is 1,152 bits and 896 bits respectively. Here we calculate the communication overhead, as shown in Table 3, Fig. 5 and Fig. 6. From the table and figure, we can see that our scheme has obvious advantages.

Table 3. Communication overhead.

Protocol	Communication overhead
[23]	1,152 bits
[24]	896 bits
Our scheme	604 bits

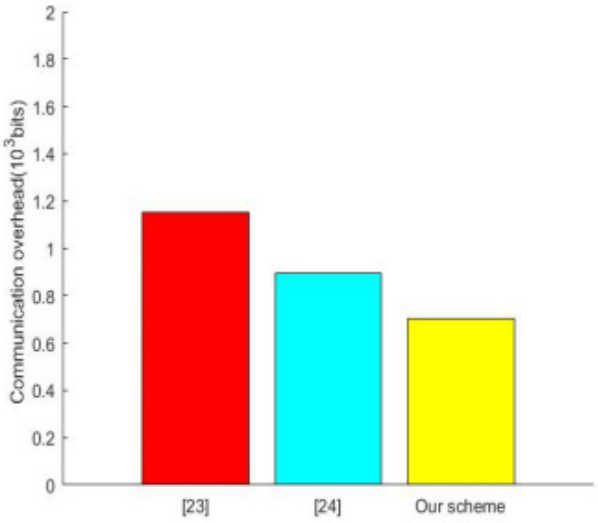

Fig. 5. Communication overhead for number of IoTD.

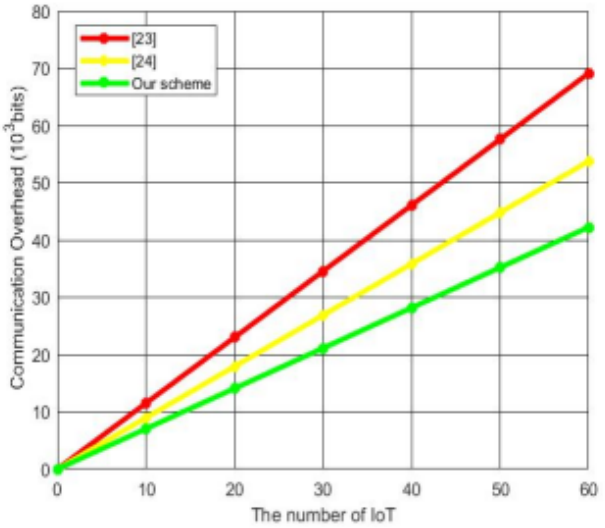

Fig. 6. Communication overhead.

7. CONCLUSION

IoT is widely used in people's lives, and has been widely used in industrial monitoring, telemedicine, green agriculture, intelligent home and other fields. However, the security threats faced by the IoT have become increasingly prominent. IoT is built on the Internet. The security threats encountered by the IoT include not only the security threats existing in the Internet, but also the specific security threats suffered due to the characteristics of IoT. The access mode of IoT mainly depends on networks and has heterogeneity. The complexity of network composition makes it more vulnerable to network attacks. Therefore, Internet of things communication environment is exposed to various security threats, which may undermine their security. Aiming at the communication security and information integrity of IoT, this paper proposes a security authentication protocol of the Internet of things based on chaotic mapping algorithm. This protocol enables Internet of things devices and server to negotiate the key and communicate securely after mutual authentication. The protocol is proved by using BAN logic, Scyther and informal security analysis, and it can resist various security attacks. Finally, our protocol has less overhead.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks,* vol. 54, no. 15, pp. 2787-2805, 2010.

[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645-1660, 2013.

[3] X. Du, S. Tang, Z. Lu, J. Wet, K. Gai, and P. C. K. Hung, "Scientific workflows in IoT environments: A data placement strategy based on heterogeneous edge-cloud computing," *ACM Transactions on Management Information Systems (TMIS),* vol. 13, no. 4, pp. 1-26, 2022.

[4] E. Anthi, L. Williams, M. Słowińska, G. Theodorakopoulos, and P. Burnap, "A supervised intrusion detection system for smart home IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 9042-9053, 2019.

[5] A. Camero and E. Alba, "Smart city and information technology: A review," *Cities*, vol. 93, pp. 84-94, 2019.

[6] J. H. Nord, A. Koohang, and J. Paliszkiewicz "The internet of things: Review and theoretical framework," *Expert Systems with Applications,* vol. 133, pp. 97-108, 2019.

[7] P. P. Ray, "A survey on internet of things architectures," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 3, pp. 291-319, 2018.

[8] C. M. Medaglia and A. Serbanati, "An overview of privacy and security issues in the internet of things," in *The Internet of Things: 20th Tyrrhenian Workshop on Digital Communications*, New York, NY: Springer, 2010, pp. 389-395.

[9] R. H. Weber, "Internet of things-new security and privacy challenges," *Computer Law & Security Review,* vol. 26, no. 1, pp. 23-30, 2010.

[10] D. E. Kouicem, A. Bouabdallah, and H. Lakhlef, "Internet of things security: A top-down survey," *Computer Networks,* vol. 141, pp. 199-221, 2018.

[11] J. Miao, Z. Wang, X. Xue, M. Wang, J. Lv, and M. Li, "Lightweight and secure D2D group communication for wireless IoT," *Frontiers in Physics*, vol. 11, pp. 433, 2023.

[12] A. Čolaković and M. Hadžialić, "Internet of things (IoT): A review of enabling technologies, challenges, and open research issues," *Computer Networks*, vol. 144, pp. 17-39, 2018.

[13] O. K. Sahingoz, "Large scale wireless sensor networks with multi-level dynamic key management scheme," *Journal of Systems Architecture*, vol. 59, no. 9, pp. 801-807, 2013.

[14] J. Miao, Z. Wang, X. Miao, and L. Xing, "A secure and efficient lightweight vehicle group Authentication protocol in 5G networks," *Wireless Communications and Mobile Computing 2021*, pp. 1-12, 2021.

[15] Y. P. Liao and C. M. Hsiao, "A secure ECC-based RFID authentication scheme integrated with ID-verifier transfer protocol," *Ad Hoc Networks*, vol. 18, pp. 133-146, 2014.

[16] R. Peeters and J. Hermans, "Attack on liao and hsiao's secure ECC-based RFID authentication scheme integrated with ID-verifier transfer protocol," *Cryptology ePrint Archive*, vol. 399, 2013.

[17] S. Kalra and S. K. Sood, "Secure authentication scheme for IoT and cloud servers," *Pervasive and Mobile Computing*, vol. 24, pp, 210-223, 2015.

[18] C. C. Chang, H. L. Wu, and C. Y. Sun, "Notes on secure authentication scheme for IoT and cloud servers," *Pervasive and Mobile Computing*, vol. 38, pp. 275-278, 2017.

[19] 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.

[20] B. Vaidya, D. Makrakis, and H. T. Mouftah, "Device authentication mechanism for smart energy home area networks," in *Proceeding of the 2011 IEEE International Conference on Consumer Electronics (ICCE),* Jan. 2011, pp. 787-788.

[21] Y. Li, "Design of a key establishment protocol for smart home energy management system, in *Proceeding of the 2013 Fifth International Conference on Computational Intelligence, Communication Sy-stems and Networks,* Jun. 2013, pp. 88-93.

[22] K. Han, J. Kim, and T. Shon, "A novel secure key paring protocol for RF4CE ubiquitous smart home systems," *Personal and Ubiquitous Computing*, vol. 17, no. 5, pp. 945-949, 2013.

[23] B. Fabian and T. Feldhaus, "Privacy-preserving data infrastructure for smart home appliances based on the octopus DHT," *Computers in Industry*, vol. 65, no. 8, pp. 1147-1160, 2014.

[24] S. A. Chaudhry, H. Alhakami, A. Baz, and F. AI-Turjman, "Securing demand response management: A certificate based access control in smart grid edge computing infrastructure," *IEEE Access*, vol. 8, pp. 101235-101243, 2020.

[25] Abbasinezhad-Mood and Nikooghadam "An anonymous ECC-based self-certified key distribution scheme for the smart grid," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 7996-8004, 2018.

[26] Y. Kim, S. Yoo, and C. Yoo, "Daot: Dynamic and energy-aware authentication for smart home appliances in internet of things," *IEEE International Conference on Consumer Electronics*, vol. 12, no. 10, pp. 196-197, 2015.

[27] J. Miao, Z. Wang, X. Ning, N. Xiao, W. Cai, and R. Liu, "Practical and secure multifactor authentication protocol for autonomous vehicles in 5G," *Software: Practice and Experience*, 2022.

[28] C. M. Medaglia and A. Serbanati, "An overview of privacy and security issues in the internet of things," in in *Proceeding of the Internet of Things: 20th Tyrrhenian Workshop on Digital Communications,* 2010, pp. 389-395.

[29] E. Fernandes, J. Jung, and A. Prakash, "Security analysis of emerging smart home applications, *IEEE Symposium on Security and Privacy (SP)*," 2016, pp. 636-654.

[30] P. McDaniel and S. McLaughlin, "Security and privacy challenges in the smart grid," *IEEE Security & Privacy*, vol. 7, no. 3, pp. 75-77, 2009.

[31] B. Vaidya, J. H. Park, S. S. Yeo, and J. J. P. C. Rodrigues, "Robust one-time password authentication scheme using smart card for home network environment," *Computer Communications*, vol. 34, no. 3, pp. 326-336, 2011.

[32] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 2, pp. 391-406, 2017.

[33] P. Kumar, A. Gurtov, J. Iinatti, M. Ylianttila, and M. Sain, "Lightweight and secure session-key establishment scheme in smart home environments," *IEEE Sensors Journal*, vol. 16, no. 1, pp. 254-264, 2016.

Journal of Multimedia Information System VOL. 10, NO. 3, September 2023 (pp. 237-248): ISSN 2383-7632 (Online)
https://doi.org/10.33851/JMIS.2023.10.3.237

AUTHORS

Yanlong Yang received his Master Degree from Northeastern University in 2017. He is currently a senior lecture at Handan Polytechnic College. His research interests include IoT, image processing, information security, etc.

Mengzhu Lu received her Master Degree from Northeastern University in 2017. She is currently a senior lecture at Handan Polytechnic College. Her research interests include IoT, image processing, information security, etc.

Xiaohan Niu is pursuing her Bachelor Degree at Northwest University. Her research interests include image processing and network security.