2021 IEEE Symposium on Computers and Communications (ISCC) | 978-1-6654-2744-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISCC53001.2021.9631436

A Lightweight Certificateless Non-interactive Authentication and Key Exchange Protocol for IoT Environments

Menghan Pan
School of Software Engineering
East China Normal University
Shanghai, China
51194501098@stu.ecnu.edu.cn

Daojing He
School of Software Engineering
East China Normal University
Shanghai, China
djhe@sei.ecnu.edu.cn

Xuru Li
School of Software Engineering
East China Normal University
Shanghai, China
lixuru217@gmail.com

Sammy Chan
Department of Electrical Engineering
City University of Hong Kong
Hong Kong, China
eeschan@cityu.edu.hk

Emmanouil Panaousis
Dept. Computing and Mathematical Sciences
Univ. of Greenwich
London, United Kingdom
e.panaousis@greenwich.ac.uk

Yun Gao
School of Software Engineering
East China Normal University
Shanghai, China
51194501121@stu.ecnu.edu.cn

***Abstract*—In order to protect user privacy and provide better access control in Internet of Things (IoT) environments, designing an appropriate two-party authentication and key exchange protocol is a prominent challenge. In this paper, we propose a lightweight certificateless non-interactive authentication and key exchange (CNAKE) protocol for mutual authentication between remote users and smart devices. Based on elliptic curves, our lightweight protocol provides high security performance, realizes non-interactive authentication between the two entities, and effectively reduces communication overhead. Under the random oracle model, the proposed protocol is provably secure based on the Computational Diffie-Hellman and Bilinear Diffie-Hellman hardness assumption. Finally, through a series of experiments and comprehensive performance analysis, we demonstrate that our scheme is fast and secure.**

***Index Terms*—identity authentication, key exchange, internet of things, provable security, bilinear pairing**

I. Introduction

The rapid development of electronic services makes Internet of Things (IoT) devices widely deployed in real-world applications. According to the GSMA's prediction [1], the number of global IoT devices will reach 25.2 billion in 2025, and the market will expand to 4 times of the current size. A large number of IoT devices are interwoven to build smart homes and smart cities, bringing great convenience to modern society.

A general network model of IoT is shown in Fig. 1. IoT terminal devices, such as the ubiquitous smart air conditioners and pet trackers, are usually responsible for information acquisition and preprocessing. After the information is captured, it is transmitted to the cloud servers through the trusted gateways. When remote users use mobile terminals (e.g. mobile phones, smart watches) to control these devices, they need to access the Internet and perform identity verification with smart devices.

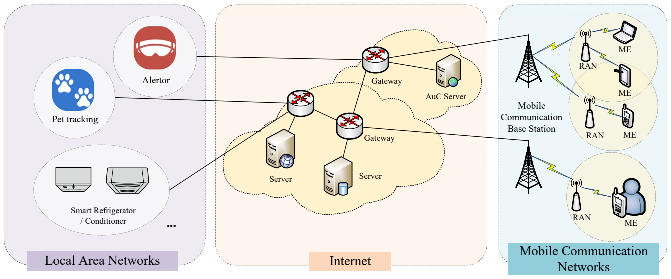

Fig. 1. Network model of internet of things.

IoT devices usually have the characteristics of simple CPU structures, low computation and communication capabilities. To resolve these problems, many authentication and key exchange (AKE) protocols for the IoT environment have been proposed. Traditional solutions either rely on Public Key Infrastructure (PKI) [2] or on a trusted third-party center [3]. The former needs to consume a lot of computation resources thus not suitable for resource-constrained environments, while the latter relies on the third-party center which poses security risks.

In response of the above challenges, based on the certificateless encryption system, we propose a lightweight non-interactive authentication and key exchange protocol. The security of the protocol is based on bilinear Diffie-Hellman and computational Diffie-Hellman hardness assumption [4]. The main contributions of this paper are as follows:

1) First, we propose a lightweight protocol to deal with the possible problems which may be faced in the IoT authentication environment. In response to the resource-constrained envi-

978-1-6654-2744-9/21/$31.00 ©2021 IEEE

ronments, we avoid using expensive cryptographic operations, choose elliptic curve encryption primitive to design protocol.

2) Second, we reduce the reliance on third-party server for the authentication protocol. Our protocol uses the certificateless encryption system, combining the advantages of traditional PKI construction and Identity-Based Cryptography, which can resist the attacks by malicious server. Furthermore, we provide a secure session key for authenticated users. Even if partial private key is leaked, the adversary cannot forge the session key to communicate with the authenticated party.

3) Finally, we reduce the communication load of the protocol. Based on the principle of zero-knowledge proof, we adopt a non-interactive authentication method to reduce the communication rounds. Compared with existing protocols, our protocol reduces communication overhead by 10% to 30%.

The rest of this paper is organized as follows. Section II summarizes the current related research work. Section III introduces background. Section IV gives the security model. Section V describes in detail the protocol proposed in this paper. Section VI presents the security analysis of the proposed protocol. The performance analysis is carried out in Section VII. Finally, the conclusion is given.

II. Related Work

A significant feature of the IoT environment is the limited resources. In recent years, many researchers have claimed that they proposed authentication and key exchange protocols suitable for the IoT environment, but those protocols are still not lightweight enough or weak in security.

Srinivas *et al.* [5] proposed an authentication and key exchange scheme for the industrial IoT scenario, which is based on chaotic mapping operations. Melk *et al.* [6] proposed an authentication protocol mainly relying on hash functions and XOR operations to ensure security. But these protocols are not secure enough due to their simple operations. Amin *et al.* [7] designed an authentication protocol with good performance. However, Arasteh *et al.* [8] pointed out that the protocol is vulnerable to replay and denial of service (DoS) attacks.

Kalra *et al.* [9] introduced the Elliptic Curve Cryptography (ECC) into the mutual authentication for IoT. Subsequently, a series of IoT authentication and key exchange protocols based on elliptic curve were proposed. Hsu *et al.* [10] proposed a symmetric key exchange protocol based on ECC, but the security of the protocol is based on the complete credibility of third-party server. Gupta *et al.* [11] implemented an authentication and key exchange protocol based on bilinear pairing, which reduces the overhead of public key certificates, however, it cannot resist malicious server attack. Although compared with previous protocols, the above protocols reduce some computation and communication costs, most of them rely too much on the security of the third-party server.

Besides, the high communication load is also a problem with current protocols. Nikravan *et al.* [12] designed a multi-factor authentication protocol for the IoT to achieve three-party identity authentication of IoT nodes, gateways and users, but the computation and communication load of the protocol are too high. Majumder *et al.* [13] proposed a lightweight authentication and key exchange protocol based on elliptic curve for smart devices and servers. Ma *et al.* [14] proposed a key agreement protocol that does not require bilinear pairing, which avoids costly pairing operations. However, due to the communication rounds, these protocols bring excessive communication load. The protocol proposed by Mandal *et al.* [15] has the same problem.

In summary, among the current authentication and key exchange protocols designed for IoT environment, some protocols reduce computation overhead by sacrificing their security, the others rely too much on the security of third-party entities, which brings great harm to authentication security. In addition, excessive communication load is also a common problem with these protocols. Therefore, how to design a secure and lightweight protocol while reducing dependence on third-party entities is the challenge addressed in this paper.

III. Background

A. System Model

The system model of the authentication scheme mainly consists of three participants, namely, KGC, user and smart device (Dev).

- KGC: The key generation center exists as a third-party trusted entity under the system model. It generates partial key containing identity information for registered users and smart devices. It also generates system parameters.
- User: The mobile entity can be any smart terminals that users use to access IoT services, such as mobile phones and smart watches. After sending an access request, it needs to prove the legal identity to involved devices.
- Dev: Smart devices, including smart homes, environmental monitoring devices and IoT devices composed of sensors and actuators. They are responsible for collecting real-world data. They participate in the identity authentication process and cooperate with users to generate authentication messages.

B. Security Goals

A secure authentication and key exchange protocol should have the following security goals:

- Mutual Authentication: During the communication process, the identity information of both parties must be confirmed. That is, mobile users must be able to successfully verify the true identity of IoT nodes, and vice versa.
- Session Key Agreement: To ensure the security during communication, the entities involved in the authentication process negotiate to generate the session key.
- User Anonymity: Protect the user's private information and prevent adversary from obtaining user's identity information.
- Perfect Forward Secrecy: Forward secrecy can ensure that the leakage of both long-term keys will not lead to leakage of past session keys. That is, the security of the content transmitted in the previous session is guaranteed.

- Resistance of Various Attacks: In IoT environment, smart devices are vulnerable to various security threats such as man-in-the-middle attack, replay attack, known-key attack, impersonation attack, unknown key-share attack, etc. A secure authentication protocol should be able to resist against attacks.

IV. Security Model

Our basic security model refers to the description provided in [16], and modifies the traditional eCK model to make it more suitable for certificateless cryptosystem.

The security model of the protocol is defined by the game between challenger CH and adversary A_d. According to adversary's attack capabilities, the adversary is defined as $Type\ I$ adversary A_I if it can replace the user's public key but cannot obtain the system master key, the adversary is defined as $Type\ II$ adversary A_{II} if it can obtain the system master key, but cannot replace the user's public key. During the game, adversary A_d makes a set of queries as follows.

- EstablishPart(ID_i): A_d requests CH to register a legal identity for any participant whose identity is ID_i. After the query, the adversary A_d will get the partial public and private key pair of ID_i.
- MasterKeyReveal: A_d can request the system master key of KGC.
- EphemeralKeyValueReveal(ID_i): A_d can request the ephemeral key value r_i of the ID_i which is selected by KGC.
- PartialStaticKeyReveal(ID_i): A_d can request the partial private key of the participant whose identity is ID_i.
- StaticKeyReveal(ID_i): A_d can request the private key of the participant whose identity is ID_i.
- PublicKeyReveal(ID_i): A_d can request the public key of the participant whose identity is ID_i.
- PublicKeyReplacement(ID_i): A_d can choose a new public key of ID_i. CH records these replacements for subsequent calculation.
- Send($\prod_{i,j}^{s}$,m): A_d sends the message m to the session $\prod_{i,j}^{s}$, which ID_i is an initiator and ID_j is receiver. CH responses the adversary A_d according to the protocol specification.
- SessionKeyReveal($\prod_{i,j}^{s}$): A_d asks a particular oracle to reveal the session key that $\prod_{i,j}^{s}$ holds.
- Test($\prod_{i,j}^{s}$): A_d may choose one of the fresh oracles to ask a Test query. To answer the query, the oracle random selects one bit of data $b \in \{0, 1\}$, returns the session key held by $\prod_{i,j}^{s}$ if $b = 0$, or returns a random sample from the distribution of the session key. The adversary A_d is only allowed to conduct this query once.

When all the queries are completed, the adversary A_d outputs b' to be used as a guess for b, game ends.

Definition 1. (AKE security). Set k as the system security parameter, and the *advantage* of adversary A_d to win the game is defined as:

$$Advantage_{A_d}(k) = |Pr[Asuccess] - \tfrac{1}{2}|$$

where $Pr[Asuccess]$ is the probability that the adversary queries test oracle to a fresh instance $\prod_{i,j}^{s}$, outputs the b' such that $b' = b$. b is used by the test oracle.

We say that an authentication and key exchange protocol is secure under the eCK model if it meets the following conditions: For any adversary A_d, the advantage of winning the above game, in probabilistic polynomial time, $Advantage_{A_d}(k)$ can be ignored.

V. Proposed Protocol

In this section, we describe the proposed lightweight certificateless non-interactive authentication protocol. The proposed protocol includes five stages. The symbols used in the protocol are listed in Table I and the protocol interaction process is shown in Fig. 2.

TABLE I
Notations of the Proposed Protocol

Notation	Description
s	System master key
P_{pub}	System public key
r_i	Ephemeral key value of user i
ID_i	Identity information of user i
Q_i	Identity hash of user i
sk_i	Partial private key of user i
x_i	Secret value of user i
R_i, P_i	Partial public key of user i
TS_i	Timestamp
SK_i	Private key of user i
PK_i	Public key of user i
t_i	Ephemeral session secret value of user i
T_i, S_i, W_i	Authentication parameter of user i
K_{i1}, K'_{i1}	Authentication certificate
K_{i2}, K_{i3}	Session key parameter of user i
K_{ij}	Session key

A. System initialization phase

In this phase, KGC chooses a prime integers q and a non-singular elliptic curve $E_p(a, b)$ defined by the formula $y^2 = x^3 + ax + b \bmod p$, where $b \in F_p$. Then, KGC selects an elliptic curve addition cyclic group G_1 with q as the order, a multiplicative cyclic group G_2, and selects P as the generator of the addition group G_1. Assuming that there is a bilinear mapping $e :\ G_1 \times G_1 \rightarrow G_2$, define two hash functions $H_1 : \{0, 1\}^* \rightarrow Z_q^*$, $H_2 : Z_q^{*2} \times G_2^3 \times G_1 \rightarrow \{0, 1\}^n$

1) KGC selects the random number $s \in Z_q^*$ as the system master key, and calculates $P_{pub} = sP$ as the system public key.

2) KGC opens the system parameters $Param = \{G_1, G_2, e, q, P, P_{pub}, H_1, H_2\}$, and keeps s confidential.

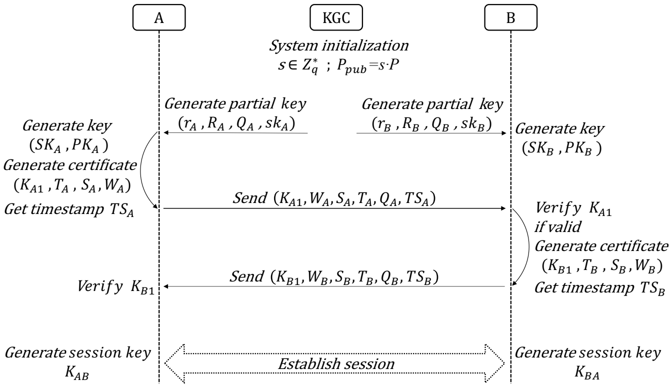

Fig. 2. Flow of the proposed protocol.

B. Key generation phase

In this phase, the aim is to generate the key information of both parties. We will give a simple notation to both parties of the authentication protocol, that is, refer the mobile entity ($User$) as A, and the smart device (Dev) as B.

1) A sends a registration request to the system (KGC). The content of the request should include the entity's own identity information such as ID_A.

2) KGC selects ephemeral key value $r_A \in Z_q^*$ and calculates $R_A = r_A P$, $Q_A = H_1(ID_A)$.

3) KGC calculates partial private key $sk_A = r_A + sQ_A$. Subsequently, the message (r_A, R_A, Q_A, sk_A) is sent to A. Among them, R_A is the partial public key of A.

4) After receiving the message, A selects the corresponding secret value $x_A \in Z_q^*$, and calculates complete private key $SK_A = x_A sk_A$, complete public key $PK_A = (R_A, P_A)$, $P_A = x_A P$.

The same is true for B.

C. Certificate generation phase

This phase mainly generates authentication certificate for two-party entity authentication.

1) A randomly selects ephemeral session secret value $t_A \in Z_q^*$, and then calculates $T_A = t_A P$, $S_A = \frac{t_A}{SK_A}$, $W_A = R_A + H_1(ID_A) \cdot P_{pub}$.

2) A generates authentication certificate: $K_{A1} = e(W_A, \frac{1}{sk_A} T_A) = e(P,P)^{t_A}$.

3) A gets current time TS_A and sends the message $(K_{A1}, W_A, S_A, T_A, Q_A, TS_A)$ to B as a voucher. Public PK_A.

D. Mutual authentication phase

In this phase, the main function is to conduct mutual authentication.

1) B first checks the timestamp information, the session which not fresh will be rejected. Then B performs as follows.

2) B calculates $K'_{A1} = e(S_A W_A, P_A)$, verifies whether K'_{A1} and K_{A1} are equal. If they are not equal, the authentication fails and B rejects the message request.

3) If they are equal, B randomly selects ephemeral session secret value $t_B \in Z_q^*$, calculates $T_B = t_B P$, $S_B = \frac{t_B}{SK_B}$, $W_B = R_B + H_1(ID_B) \cdot P_{pub}$. And then, B generates authentication certificate: $K_{B1} = e(W_B, \frac{1}{sk_B} T_B) = e(P,P)^{t_B}$.

4) B gets current time TS_B and sends the message $(K_{B1}, W_B, S_B, T_B, Q_B, TS_B)$ to A as a voucher. Public PK_B.

5) A checks the timestamp and calculates $K'_{B1} = e(S_B W_B, P_B)$, then verifies whether K'_{B1} and K_{B1} are equal. If they are equal, mutual authentication is completed.

E. Key exchange phase

After two-way authentication, the mobile user and the smart device establish session key through the disclosed information to ensure the security of subsequent communication.

1) A calculates $K_{A2} = e((t_A + sk_A)P_{pub}, T_B + R_B + Q_B P_{pub})$, $K_{A3} = r_A R_B$. Let $K_{AB} = H_2(Q_A, Q_B, K_{A1}, K_{B1}, K_{A2}, K_{A3})$.

2) B calculates $K_{B2} = e((t_B + sk_B)P_{pub}, T_A + R_A + Q_A P_{pub})$, $K_{B3} = r_B R_A$. Let $K_{BA} = H_2(Q_B, Q_A, K_{B1}, K_{A1}, K_{B2}, K_{B3})$.

So we have the same session key: $K_{AB} = K_{BA}$.

VI. Security Analysis

As mentioned earlier, the security of the proposed protocol depends on the solution of the BDH and CDH hardness assumptions. Formal and informal security analysis will be given to prove that our proposed protocol is secure.

A. Provable Security Analysis

Lemma 1. Assuming that the BDH hardness assumption cannot be solved, under the random oracle model, the probability of the adversary A_I winning the game is negligible.

Proof. Assuming that there is a *Type I* adversary A_I who can win the game with a non-negligible advantage $Advantage_{A_I}(k)$ in polynomial time. Then, there must be a hardness assumption adversary CH, which pretends to be an algorithm challenger, and solves the BDH assumption through interaction with the adversary A_I. That is, given CH a BDH assumption instance (P, aP, bP, cP), the goal is to calculate $e(P,P)^{abc}$. The specific process is as follows.

CH constructs system parameters $Param = \{G_1, G_2, e, q, P, P_{pub}, H_1, H_2\}$. Among them, $P_{pub} = cP$. CH randomly selects $ID_I, ID_J \in \{0,1\}^*$, sends the system parameters and ID_I, ID_J to adversary A_I. The game is initialized.

$H_1 query$: CH maintains an initial empty list L_1 : (ID_i, Q_i). After receiving the $H_1 query$, CH randomly selects $Q_i \in Z_p^*$ and returns it to A_I. CH adds (ID_i, Q_i) to L_1.

$H_2 query$: CH maintains an initial empty list L_2 : $(Q_i, Q_j, K_{i1}^s, K_{i2}^s, K_{i3}^s, K_{i4}^s, h_i)$. After receiving the $H_2 query$, CH executes as follows.

- If there is a $(\prod_{i,j}^s, m)$ record in the list L_S, and $ID_i \neq ID_I$. CH selects in the list L_C, L_S, calculates $K_{i1}^s = e(W_i^s, \frac{1}{sk_i} T_i^s)$, $K_{i2}^s = e((t_i^s + sk_i)P_{pub}, T_j^s + R_j + Q_j P_{pub})$. Set $K_{i3}^s = r_i R_j$, where r_i is the ephemeral key value of ID_i, R_j is the partial public key of the communication responder of the $(\prod_{i,j}^s)$'s matching session $(\prod_{j,i}^s)$.

$K_{i4}^s = K_{j1}^s$, where K_{j1}^s is the authentication certificate of the session $(\prod_{j,i}^s)$. CH chooses a random $h_i \in \{0,1\}^n$ and returns as the answer.

- If there is a $(\prod_{i,j}^s, m)$ record in the list L_S, and $ID_i = ID_I$. CH selects in the list L_C, L_S, calculates $K_{i1}^s = e(W_i^s, \frac{1}{sk_i}T_i^s)$, $K_{i2}^s = e((t_i^s + sk_i)P_{pub}, T_j^s + R_j + Q_j P_{pub})$, $K_{i3}^s = r_i R_j$, $K_{i4}^s = K_{j1}^s$. Set $h_i = SK_{ij}^s$ and return h_i as the answer.
- Else if there is no such $(\prod_{i,j}^s, m)$ record in the list L_S, CH chooses a random $h_i \in \{0,1\}^n$ return as answer.
- CH adds $(Q_i, Q_j, K_{i1}^s K_{i2}^s, K_{i3}^s, K_{i4}^s, h_i)$ in list L_2.

$EstablishPart(ID_i)$: CH maintains an initial empty list $L_C : (ID_i,\ sk_i,\ r_i,\ R_i)$. After receiving this query, CH first performs an $H_1 query$ to obtain (ID_i, Q_i). Then CH judges whether ID_i is equal to ID_J. If $ID_i \neq ID_J$, CH chooses a random $r_i \in Z_p^*$ and calculates the partial public key $R_i = r_i P$ and the partial private key $sk_i = r_i + cQ_i$. If $ID_i = ID_J$, CH sets partial public key $R_i = bP$, and calculates the partial private key $sk_i = b + cQ_i$. Then CH adds (ID_i, sk_i, r_i, R_i) to the list L_C.

$EphemeralKeyValueReveal(ID_i)$: On receiving this query. If $ID_i \neq ID_I$ and $ID_i \neq ID_J$, CH searches L_C and returns r_i indexed by ID_i as the answer. Else CH aborts.

$PublicKeyReveal(ID_i)$: CH maintains an initial empty list $L_U : (ID_i, x_i)$. On receiving the $PublicKeyReveal(ID_i)$ query, CH first searches the list L_C to obtain the (ID_i, sk_i, r_i, R_i) indexed by ID_i. Then CH chooses a random $x_i \in Z_p^*$, calculates $P_i = x_i P$ and returns (R_i, P_i) as the answer. Lastly, CH adds (ID_i, x_i) to the list L_U.

$PartialStaticKeyReveal(ID_i)$: On receiving this query, CH searches the list L_C to obtain the sk_i indexed by ID_i, returns sk_i as the answer.

$StaticKeyReveal(ID_i)$: On receiving this query, if $ID_i = ID_I$, CH aborts. Otherwise, CH searches L_U to obtain x_i, searches L_C to obtain the sk_i, and then calculates $SK_i = x_i \cdot sk_i$, returns SK_i as the answer.

$PublicKeyReplacement(ID_i)$: On receiving this query, if $ID_i = ID_J$, CH fails. Otherwise, CH searches the list L_C to obtain the (ID_i, sk_i, r_i, R_i) indexed by ID_i. Then, it updates R_i to $R_i{}'$ and sets $r_i = null$.

$Send(\prod_{i,j}^s, m)$: Assuming that ID_i is the communication initiator and ID_j is the responder of the matching session. CH maintains an initial empty list L_S : $(\prod_{i,j}^s$, K_{i1}^s, K_{j1}^s, W_i^s, S_i^s, T_i^s, Q_i, W_j^s, S_j^s, T_j^s, Q_j, TS_i^s, TS_j^s, $SK_{ij}^s)$, where $m \leftarrow (K_{i1}^s, W_i^s, S_i^s, T_i^s, Q_i, TS_i^s)$ is the sending message. On receiving the $Send(\prod_{i,j}^s, m)$ query, CH executes as follows.

- If $ID_i = ID_I$ and $ID_j = ID_J$, CH selects in the list L_1, L_C, L_U and sets $t_i^s = null$, calculates $T_i^s = aP$, $W_i^s = R_i + Q_i \cdot P_{pub}$, $S_i^s = \frac{t_i^s}{x_i sk_i}$. CH judges whether K_{i1}^s is equal to $e(R_i + cPQ_i, \frac{1}{sk_i} \cdot aP)$. If they are equal, sets $SK_{ij}^s = null$, otherwise chooses a random $SK_{ij}^s \in \{0,1\}^n$.
- Else CH selects in the list L_1, L_C, L_U and chooses a random $t_i^s \in Z_p^*$, calculates $K_{i1}^s = e(W_i^s, \frac{1}{sk_i}T_i^s)$, $T_i^s = t_i^s P$, $W_i^s = R_i + Q_i \cdot P_{pub}$, $S_i^s = \frac{t_i^s}{x_i sk_i}$. Set $SK_{ij}^s = h_i$, which restored in the list L_2 .
- CH returns $(K_{i1}^s, W_i^s, S_i^s, T_i^s, Q_i, TS_i^s)$ as the answer and adds $(\prod_{i,j}^s$, K_{i1}^s, K_{j1}^s, W_i^s, S_i^s, T_i^s, Q_i, W_j^s, S_j^s, T_j^s, Q_j, TS_i^s, TS_j^s, $SK_{ij}^s)$ to list L_S in both case.

$SessionKeyReveal(\prod_{i,j}^s)$: On receiving this query, if $ID_i = ID_I$ and $ID_j = ID_J$, CH aborts. Otherwise CH searches the list L_S, and returns SK_{ij}^s as the answer.

$Test(\prod_{i,j}^s)$: If $\prod_{i,j}^s$ is not the target Test session, CH fails. Else if $\prod_{i,j}^s$ is not fresh, CH aborts. Otherwise CH chooses a random $\omega \in \{0,1\}^n$ and outputs as the answer to A_I.

CH queries whether there is such a situation that $K_{i3} = r_i R_j$, if no such a record exists, CH fails. Otherwise, since

$$\begin{aligned} K_{i2}^s &= e((t_i^s + sk_i)P_{pub}, T_j^s + R_j + Q_j P_{pub}) \\ &= e(t_i^s P_{pub} + sk_i P_{pub}, (T_j^s + Q_j P_{pub}) + R_j) \\ &= e(acP + sk_i P_{pub}, (T_j^s + Q_j P_{pub}) + bP) \\ &= e(acP, bP) \cdot u = e(P,P)^{abc} \cdot u \end{aligned} \tag{1}$$

So CH can solve BDH assumption as the result $K_{i2}^s \cdot u^{-1}$, where K_{i2}^s can be selected in the list L_2, and u is the bilinear operation set which can be calculated with the records in L_S.

Probability analysis: In order to analyze the probability, we define the following events:

- $Ev1$: CH did not fail due to public key replacement.
- $Ev2$: A chooses the target session as test session.
- $Ev3$: A picks the correct tuple from L_2.

Assume that the adversary A_I can perform $H_2 query$ for at most q_{h2} times, create up q_e participating entities, and each entity can be involved in at most q_s sessions. We have

$$\begin{aligned} &Pr[Ev1 \wedge Ev2 \wedge Ev3] \\ &= Pr[Ev1] \cdot Pr[Ev2|Ev1] \cdot Pr[Ev3|Ev1 \wedge Ev2] \\ &= \frac{q_e - 1}{q_e} \cdot \frac{1}{q_e(q_e-1)q_s} \cdot \frac{1}{q_{h2}} = \frac{1}{q_e^2 q_s q_{h2}} \end{aligned} \tag{2}$$

Therefore, if adversary A_I can win the game with the advantage of $Advantage_{A_I}(k)$, there must be a challenger CH who can solve the BDH assumption with the advantage of $Pr[CHwins|A_I] \geq \frac{1}{q_e^2 q_s q_{h2}} Advantage_{A_I}(k)$, which contradicts the inexplicability of the BDH assumption. Therefore, our scheme is secure under the attack of adversary A_I. □

Lemma 2. Assuming that the CDH hardness assumption cannot be solved, under the random oracle model, the probability of the adversary A_{II} winning the game is negligible.

Proof. Assuming that there is a $Type\ II$ adversary A_{II} who can win the game with a non-negligible advantage $Advantage_{A_{II}}(k)$ in polynomial time. Then, given CH a CDH assumption instance (P, aP, bP), the goal is to calculate abP. The proof of this lemma is similar to $Lemma\ 1$, CH simulates all queries, but does not simulate the values of a and b. If the adversary A_{II} cracks the protocol, it can obtain the solution of the CDH assumption $K_{i3}^s = abP$ under the situation that only $R_i = aP$, $R_j = bP$ is known, and a, b are not leaked. □

B. Informal Security Analysis

In this part, we show that the proposed protocol can achieve the security goals described in Section III.

- Mutual Authentication: In authentication process, two parties verify the certificates $K'_{i1} = K_{i1}$ and $K'_{j1} = K_{j1}$, which provided by each other. It is difficult for adversary to forge valid authentication certificate because it does not know the user's private key.
- Session Key Agreement: Using bilinear pairing and Diffie-Hellman key agreement principle, after authentication, both parties can successfully calculate an equal session key K_{ij}. In addition, $Lemma$ 1 and $Lemma$ 2 prove that the protocol is secure.
- User Anonymity: According to the protocol description, the user's real identity information ID_i is masked by $Q_i = H_1(ID_i)$. If adversary wants to get the ID_i, it must destroy the irreversibility of hash operation.
- Perfect Forward Secrecy: Assume that the adversary has obtained partial private keys sk_i and sk_j, and intercepted the messages of both parties. The adversary must calculate the session key K_{ij}^s to get the message content. But it is difficult to calculate K_{i2}^s in the session key because adversary does not know ephemeral session secret value t_i^s of the previous session. Therefore, our protocol provides perfect forward secrecy.
- Resist Man-in-the-Middle Attack: According to the above proof, the protocol supports two-part identity authentication, and both sides of the communication cannot cheat each other. Therefore, the protocol can resist man-in-the-middle attack.
- Resist Replay Attack: The timestamp information is included in the communication messages between the authenticated parties. The receiver uses the timestamp to verify the freshness of the messages. Therefore, the protocol can resist replay attack.
- Resist Known-Key Attack: According to the protocol description, the ephemeral session secret value is selected randomly in each session, the leakage of one session key value has no impact on the security of other session keys. Therefore, the protocol can resist known-key attack.
- Resist Malicious Server Attack: Assume that the adversary corrupts the KGC node, it can only obtain partial private key, but cannot obtain the secret value chosen by the user. Therefore, the protocol can ensure that key is not leaked when there is a malicious server in the network.
- Resist impersonation attack: Assume that the adversary has intercepted the message of $\prod_{i,j}^{s}$. In order to impersonate the ID_j, adversary should forge a message and calculate the session key K_{ji}. However, it is difficult for adversary to calculate certificate S_j in message because it does not know the private key of ID_j. Also, $Lemma$ 2 proves that adversary calculates K_{j3} in K_{ji} is difficult.
- Resist unknown key-share attack: According to the protocol description, ID_i and ID_j use K_{j1} and K_{i1} to calculate the session key K_{ij}, which can be authenticated by verifying K'_{j1} and K'_{i1}. Therefore, the protocol can resist unknown key-share attack.

VII. Performance Analysis

In this section, we compare the security and performance of the proposed CNAKE protocol with existing protocols. In addition, we carry out a rigorous experimental evaluation of the protocol in a resource-constrained environment.

A. Comparison on Functionality and Security

In Table II, we show the security comparison between the proposed protocol and other protocols. It can be seen that our protocol provides more comprehensive security.

TABLE II
Functionality and security comparison

Protocol	A1	A2	A3	A4	A5	A6	A7	A8
[12]	✓	✓	✓	✓	✓	✓	–	–
[14]	✓	✓	✓	✓	✓	–	–	–
[15]	✓	✓	✓	✓	✓	✓	✓	–
CNAKE	✓	✓	✓	✓	✓	✓	✓	✓

A1: User anonymity, A2: Perfect forward secrecy, A3: Resist man-in-the-middle attack, A4: Resist replay attack, A5: Resist known-key attack, A6: Resist malicious server attack, A7: Resist impersonation attack, A8: Resist unknown key-share attack.

B. Comparison on Computation and Communication

Table III shows the computation comparison of our protocol and existing protocols. We use Raspberry Pi to simulate smart devices in the IoT environment, implement based on the commonly PBC Library in C++ language, and the processor configuration is 1.5 GHz with 4 GB of memory, the operating system is Ubuntu-18.04.3-desktop-amd64. Since the execution time of elliptic curve scalar addition operation and hash operation is extremely short under the experimental environment, they will be ignored in the computation cost analysis. The execution time of the elliptic curve scalar multiplication operation T_{sm} = 2.723ms, bilinear pairing operation T_{bm} = 3.335ms, a fuzzy extractor function T_{fe} = 2.727ms. The experimental results are given in Fig. 3.

For the analysis of communication overhead, in the specific implementation process, the element length of entity identity, random number, hash value and other elements in Z_r is 160 bits, the length of point coordinate on elliptic curve in G_1 is 128 bits, the length of bilinear pairing in G_T is 128 bits, and the length of timestamp T is 32 bits. In Table III, we show the communication cost of our protocol and existing protocols. It can be seen that the communication cost of the proposed protocol is much lower than comparison schemes. In addition, the rounds of communication is greatly reduced.

Compared with the protocol proposed by Nikravan $et\ al.$ [12], our protocol has significant improvement in computation

TABLE III
PERFORMANCE COMPARISON

Protocols	**Computation Cost**	**Communication Cost**		
		Total	Cost	No. of Messages
CNAKE	15 T_{sm} + 6 T_{bm}	4$\|Z_r\|$ + 8$\|G_1\|$ + 2$\|G_T\|$ +2$\|T\|$	1984 bits	2
Nikravan *et al.* [12]	20 T_{sm} + 6 T_{bm}	12$\|Z_r\|$ + 2$\|G_1\|$ + 3$\|T\|$	2272 bits	3
AKA [14]	18 T_{sm}	9$\|Z_r\|$ + 10$\|G_1\|$ + 5$\|T\|$	2880 bits	4
CSUAC-IoT [15]	19 T_{sm} + T_{fe}	10$\|Z_r\|$ + 4$\|G_1\|$ + 3$\|T\|$	2208 bits	3

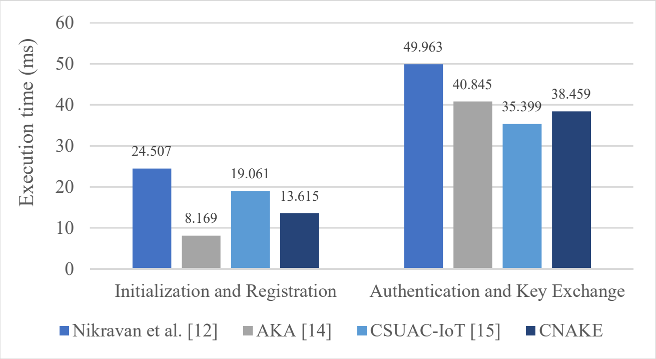

Fig. 3. Execution time of different protocols.

and communication overhead. Compared with AKA [14], our protocol reduces the communication cost obviously. Also, compared with CSUAC-IoT [15], although the computation cost is similar, our protocol effectively reduces the communication overhead through non-interactive authentication. Therefore, the proposed CNAKE protocol has good performance.

VIII. CONCLUSION

With the advent of the Internet of Things, more and more smart devices will enter people's lives, and the security authentication between users and these devices will also be a topic worthy of continuous attention. In response to this problem, we have proposed a lightweight certificateless non-interactive authentication and key exchange protocol based on elliptic curve and bilinear pairing, which is used for identity security deployment in resource-constrained environments of the Internet of Things. The solution adopts the idea of zero-knowledge proof to realize the non-interactive verification between users and devices, which greatly reduces the overload of communication. We have carried out strict security analysis and experimental evaluation, which proves that the protocol meets the security attributes required by the current industry and has better efficiency performance.

ACKNOWLEDGMENT

This research is supported by the National Key R&D Program of China (2017YFB0802805 and 2017YFB0801701), the National Natural Science Foundation of China (Grants: U1936120, U1636216), the Fundamental Research Funds for the Central Universities, and the Basic Research Program of State Grid Shanghai Municipal Electric Power Company (52094019007F). Daojing He is the corresponding author of this paper.

REFERENCES

[1] GSMA Homepage. [Online]. Available: https://www.gsma.com/globalmobiletrends/.

[2] H. Pranata, R. Athauda, and G. Skinner, "Securing and governing access in ad-hoc networks of internet of things," in *Proceedings of the IASTED International Conference on Engineering and Applied Science*, EAS, 2012, pp. 84-90.

[3] H. Tschofenig, and T. Fossati, "Transport layer security (tls)/datagram transport layer security (dtls) profiles for the internet of things," in *RFC 7925, Internet Engineering Task Force*, 2016.

[4] D. Cash, E. Kiltz, and V. Shoup, "The twin Diffie-Hellman problem and applications," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Berlin, Heidelberg, 2008, pp. 127-145.

[5] J. Srinivas, and A. Das, "Anonymous lightweight chaotic map-based authenticated key agreement protocol for industrial internet of things," *IEEE Transactions on Dependable and Secure Computing*, 2018.

[6] R. Melki, H. Noura, and A. Chehab, "Lightweight multi-factor mutual authentication protocol for IoT devices," *International Journal of Information Security*, pp. 1-16, 2019.

[7] R. Amin, and S. Islam, "Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks," *Computer Networks*, vol. 101, pp. 42-62, 2016.

[8] S. Arasteh, S. Aghili, and H. Mala. "A new lightweight authentication and key agreement protocol for internet of things," in *2016 13th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC)*, IEEE, 2016, pp. 52-59.

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

[10] C. Hsu, and T. Chuang, "A dynamic identity end-to-end authentication key exchange protocol for iot environments," in *2017 Twelfth International Conference on Digital Information Management (ICDIM)*, IEEE, 2017, pp. 133-138.

[11] D. Gupta, and S. Islam, "A provably secure and lightweight Identity-Based two-party authenticated key agreement protocol for IIoT environments," *IEEE Systems Journal*, 2020.

[12] M. Nikravan, and A. Reza, "A multi-factor user authentication and key agreement protocol based on bilinear pairing for the internet of things," *Wireless Personal Communications*, vol. 111, no. 1, pp. 463-494, 2020.

[13] S. Majumder, and S. Ray, "ECC-CoAP: Elliptic Curve Cryptography based constraint application protocol for internet of things," *Wireless Personal Communications*, pp. 1-30, 2020.

[14] M. Ma, and D. He, "An efficient and provably secure authenticated key agreement protocol for fog-based vehicular ad-hoc networks," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8065-8075, 2019.

[15] S. Mandal, B. Bera, and AK. Sutrala, "Certificateless-Signcryption-Based three-factor user access control scheme for IoT environment," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3184-3197, 2020.

[16] M. Gervais, L. Sun, K. Wang, and F. Li, "Certificateless authenticated key agreement for decentralized WBANs," in *International Conference on Frontiers in Cyber Security*, Springer, Singapore, 2019, pp. 268-290.