electronics

MDPI

Article

A Noval and Efficient ECC-Based Authenticated Key Agreement Scheme for Smart Metering in the Smart Grid

Cong Wang [1], Su Li [1], Maode Ma [2,*], Xin Tong [1], Yiying Zhang [1] and Bo Zhang [3]

[1] College of Artificial Intelligence, Tianjin University of Science and Technology, Tianjin 300453, China
[2] College of Engineering, Qatar University, Doha 2713, Qatar
[3] State Grid Smart Grid Research Institute Co., Ltd., Nanjing 211167, China
* Correspondence: mamaode@qu.edu.qa

Abstract: With the gradual maturity of the smart grid (SG), security challenges have become one of the important issues that needs to be addressed urgently. In SG, the identity authentication and key agreement protocol between a smart meter (SM) and an aggregator (AG) is a prerequisite for both parties to establish a secure communication. Some of the existing solutions require high communication cost, some have key escrow problems and security defects. Elliptic curve cryptosystem (ECC) holds the feature of low-key requirement and high security to make it more suitable for the security solutions to the communications in SG. In this paper, we propose a mutual anonymous authentication with an ECC-based key agreement scheme to secure the communications in SG. In addition, we compare our scheme with other existing schemes by the number of encryption operations, the computation delay, and the communication cost. The results indicate that our scheme is more efficient without the loss of safety properties.

Keywords: smart grid; ECC; key management; forward secrecy

Citation: Wang, C.; Li, S.; Ma, M.; Tong, X.; Zhang, Y.; Zhang, B. A Noval and Efficient ECC-Based Authenticated Key Agreement Scheme for Smart Metering in the Smart Grid. *Electronics* **2022**, *11*, 3398. https://doi.org/10.3390/electronics11203398

Academic Editor: Ahmed Al Durra

Received: 15 September 2022
Accepted: 16 October 2022
Published: 20 October 2022

1. Introduction

The traditional power grid can no longer meet people's demand for electricity, and it relies on fossil fuels, so it has an impact on the environment and energy. As a result, the smart grid (SG) was proposed at the beginning of the 21st century and developed rapidly. It uses advanced and modern technology to transmit the power efficiently and reliably, to control more effectively the cost and manage the power resources [1]. SG as the next generation of the power grid has received much attention for its efficiency, reliability, and sustainability.

SG can promote the rapid development of a country and provide convenience for people's lives, having cost-effective and reliable characteristics. However, in SG, there could be many security risks between an aggregator (AG) and a smart meter (SM), if the communication between them has not been well protected. Security issues such as software vulnerabilities have always been the key issues with its development. With the vulnerabilities, hackers can easily launch attacks to cause power supply failures, power grid overload, and energy theft. In addition, the uncoordinated planning, design, and development speed can also bring more serious security challenges [2]. Therefore, in SG, the identity authentication and key agreement protocol between an AG and a SM is a prerequisite for both parties to establish a secure communication.

1.1. Related Works

Mutual authentication between an SM and an AG is the first critical step in the design of security countermeasures. Over the last few years, various researchers have made many efforts in the design of authenticated key distribution schemes in SG [3,4]. By Gope's scheme [5], before communication, an AG has to check the validity of the

SMs, which increases the computational complexity linearly with the number of *SMs* [6]. Kumar et al. proposed a lightweight authentication and key agreement in intelligent energy networks in [7]. But Kumar's scheme is unable to resist a brief leak of secrets and suffers from time synchronization attacks. Odelu et al. proposed a provably secure authenticated key agreement scheme for SG in [8], but Odelu's scheme is vulnerable to impersonation attack and traceability attack. Braeken et al. proposed a provably secure key agreement model for intelligent metering communications in [9], but Braeken's scheme cannot deal effectively with malicious internal attackers [10]. Xiang and Zhang have presented a situation-aware protocol for device authentication in SG using a hash function and synchronous encryption for authentication, but without supporting perfect forward secrecy and anonymity of *SMs* [11]. Physical unclonable functions are an original function against physical attacks [12,13], but they are extremely vulnerable to modeling attacks.

Compared to other cryptographic functions, the elliptic curve cryptosystem (ECC) can reduce computational work effectively, and can also be used in the design of security schemes for SG [14–22]. Dariush et al. proposed a lightweight authentication scheme for a security-enhanced ECC in [14], which can improve its security performance and provide perfect positive confidentiality but fails to protect the anonymity of the *SMs*. Srinivas's scheme cannot defend the man-in the middle attacks and simulated attacks in [15]. In addition, it does not render the anonymity feature of *SMs*. Chaudhry in [16] analyzed the password-based anonymous lightweight key agreement protocol proposed by Khan et al. in [17] and showed that Khan's scheme has a false login and authentication stage according to the ECC operation. Thus, Chaudhry proposed an improved scheme, which is not free from the public key infrastructure challenges with a high computation cost in [16]. By Khan's new scheme in [18], the communication among the user, trusted third party and server creates some security issues as a result of the introduction of a third party. To alleviate such problems, Sureshkumar et al. proposed a modified mutual authentication and key protocol mechanism using the ECC for SG in [19], but which requires a separate security mechanism to organize and initialize the real-time information. In order to solve the problems of *SMs* being vulnerable to tracking attacks, Baghestani et al. in [20] and Chaudhry et al. in [21] proposed a new mutual authentication key agreement scheme based on ECC for SG. However, both the schemes are weak against key compromise impersonation attack. The authentication scheme for SG proposed by Jalili et al. can ensure the anonymity of *SM* in [22], but it is vulnerable to de-synchronization attacks and has a large communication cost because the transmitted parameters need to be updated in the authentication stage.

Some other schemes in [23–26] can cause a significant drain on communication and computation resources due to their high computation complexity. Wang et al. introduced a blockchain-based authentication and key agreement protocol (BAKA) for edge-computing-based SG in [24]. The adoption of blockchain technology can solve the issue of undeniability and transparency of data but it will greatly increase the time delay for message chaining up and retrieval. Qi et al. proposed a scheme named as two-pass privacy preserving authenticated key agreement (TPPA) based on the elliptic curve Qu-Vanstone implicit certificates with a trusted third-party participation in SG in [25]. However, under the Canetti and Krawczyk (CK) adversary model [27], it does not ensure session key security, and suffers from denial of service attacks. Xiang et al. in [26] proposed a secure privacy-preservation authentication key agreement scheme (SPAK) for SG communications without providing anonymity of the *SMs*. Thus, it could not provide a better privacy-preservation and efficient authentication process.

1.2. Motivation

Due to the requirements for high reliability and security for the communications in SG, each *SM* needs to be authenticated by an *AG* before entering the SG. The main design goal in SG is to provide robust quality data transmission to meet the requirements of QoS, such as reliability, throughput, latency, and security [28]. In fact, power information generated by

SMs is often transmitted over public insecure channels, which definitely gives the attackers an opportunity to break into the SG system. Secure identity mutual authentication is a critical first step in deterring attackers. Although there are many researchers working on authentication schemes, they all have their own problems, such as not providing anonymity of *SMs* in [14,15,26], holding a key hosting problem in [20,21,24,26], being vulnerable to active attacks in [15,25], without the ability to resist the man-in-the-middle attacks in [15,24], lack of providing session key security in [25], requiring a high computation cost in [23–26]. Therefore, to provide thoroughly the desired security functions for the communications in SG, particularly with the ability against key compromise attacks, we put forward an efficient and anonymity self-authentication scheme between the *AG* and the *SM* based on the ECC in SG. The scheme adopts less bilinear mapping and multiplication operations to reduce the computational costs without sacrificing security function.

1.3. Contributions

To solve the above existing problems, we present our contributions as follows:

(1) We propose an ECC-based authentication and key agreement scheme (EAKA) for SG. The *SM* and *AG* are registered with a trusted third party and conduct two-way identity authentication to provide anonymity protection for the *SM*.
(2) The proposed EAKA scheme can achieve a strong voucher privacy evaluated by the CK adversary model. It is testified to be safe under the random oracle model. Theoretical safety analysis indicates that the proposed EAKA scheme can oppose some classic attacks such as replay attacks and MITM attacks.
(3) The proposed EAKA scheme has more advantages on network performance. According to the number of encryption operations, the computation delay, and communication cost, we compare the proposed EAKA scheme with other schemes to demonstrate that it is effective in terms of security and computational cost in the authentication process.

1.4. Paper Organization

The rest of the paper is arranged as follows:

In the second part, we present the mathematical background and the encryption concepts involved in the proposed scheme. In addition, we also introduce the system model and the threat model. In the third part, we describe the process of identity authentication of the proposed EAKA scheme in detail. In the fourth part, we perform a safety analysis of the proposed EAKA scheme using the CK adversary model and qualitative safety analysis. In the fifth part, we evaluate the performance of the proposed EAKA scheme through simulation experiments. Finally, we summarize the paper and propose the future work.

2. Preliminaries

In this section, we first describe the relevant content about the cryptography of the ECC. We then introduce the system model of the communication networks in SG. Finally, we discuss the threat model used in the security analysis.

2.1. ECC

As shown in Equation (1), the elliptic curve group $E_p(a, b)$ is defined over the prime finite field F_p by the nonsingular elliptic curve equation E, where p is a prime number:

$$y^2 = x^3 + ax + b(mod\ p), a, b \in F_p, \Delta = 4a^3 + 27b^2 mod\ p \neq 0 \quad (1)$$

Lemma 1. *(Elliptic Curves Discrete Log Problem (ECDLP))* [29,30]*: Given the discrete log problem of fixed points* $G \in E_p(a, b)$ *and* $P = KG \in E_p(a, b)$*, it is very hard to calculate* $k \in Z_q^*$.

Lemma 2. *(Elliptic Curve Diffie-Helman problem (ECDHP)) [29,30]: The security of the ECDHP key exchange system works based on the security of the ECDLP. Given* $G, xG, yG \in E_p(a,b)$*, it is very difficult to calculate* $xyG \in E_p(a,b)$*.*

We choose a random number $m \in Z_q^*$, the random number on the elliptic curve which meets its scalar point multiplication is defined as $mG = G + G + \ldots + G(m\ times)$. Let G_1 and G_2 be a cycle group of prime order q, where G_1 is an additive group of cycles and G_2 is a multiplicative cycle group. The map $e : G_1 \times G_1 \to G_2$ is proved to be an admissible bilinear map if it meets the following conditions.

(1) Bilinearity: $e(aP, bQ) = e(abP, Q) = e(P, abQ) = e(P, Q)^{ab}$, for all $a, b \in Z_q^*$, $p, Q \in G_1$.
(2) Non-degeneracy: There exists $P, Q \in G_1$, so that $e(P, Q) \neq 1$, where 1 is the multiplication unit of G_2.
(3) Computability: For all $P, Q \in G_1$, $e(P, Q)$ can be efficiently computed.

2.2. System Model

As demonstrated in Figure 1, a communication network in SG consists of three traditional networks including a Home Area Network, a Local Area Network, and a Wide Area Network [31]. Based on the aspects of the public utilities, the Home Area Network is a group of household appliances, entertainment systems, lighting systems, energy storage, and power generation. In the Home Area Network, the *SM* is a home gateway that can gather energy depletion readings, which then transmits the collected readings to the service provider through the *AG* and performs the control command obtained from the service provider. The Local Area Network supports communication between the *SMs* and the *AGs*. Data concentrators and *AGs* can be concentrated in the surrounding residential areas. We set up a wireless mesh network between metering gateways and the *SMs*, through which the *AGs* can periodically collect all the required data, and then transmit them to the utilities via fixed-line communication. The Local Area Network usually communicates through the powerline communication.

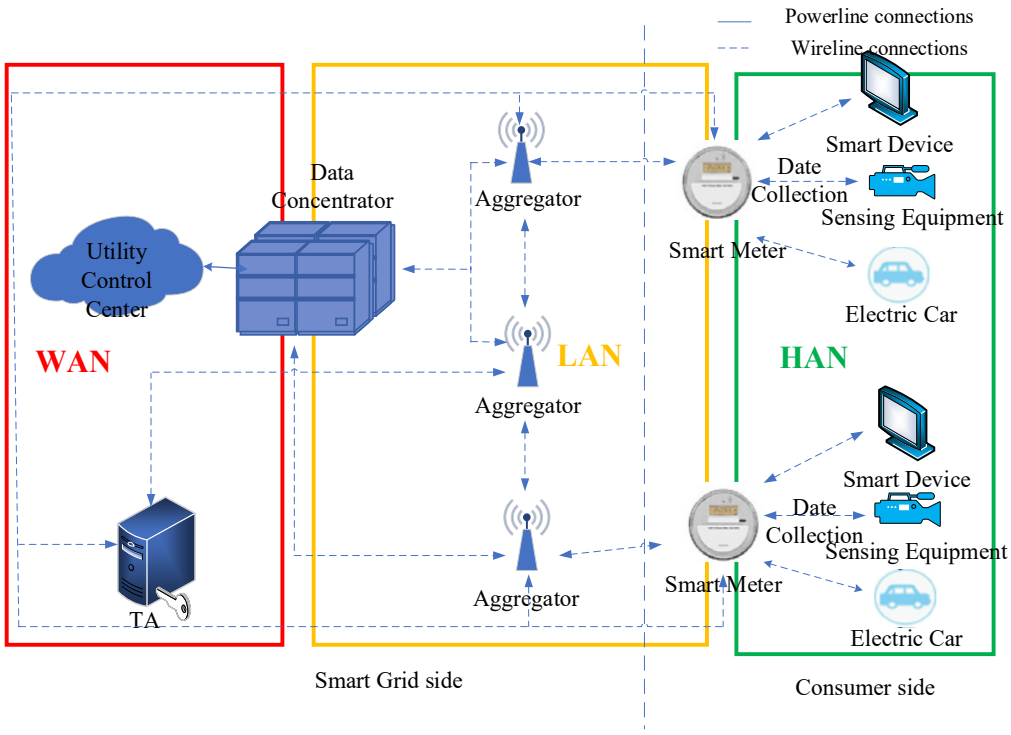

Figure 1. Architecture of Communication Networks in SG.

Wireless mesh networks have been widely used, where each SM collects its information and becomes a router for other SMs to send consumption usage information to the data concentrator. The Wide Area Network provides connections between multiple data concentrators and the utility control centers, which is called the advanced network. In addition, the Wide Area Network can transmit and receive large amounts of smart metering infrastructure data, control commands, and signals, so it is also considered as a core network. The AG and the SM should be mutually authenticated to obtain a session key agreement. Before the authentication, AGs and SMs should register with the registration authority (RA) located near the utility service provider in the Wide Area Network. If the two-way communication between the AG and the SM is exposed to the public, attackers may launch malicious attacks to threaten the security of communications. If sensitive data is leaked, customer privacy will be compromised. In addition, the delay of real-time communication also affects the efficiency of the communication. Therefore, a more secure and strict authentication scheme should be adopted to protect the privacy of users.

2.3. Threat Model

This paper employs the widely-accepted and well-known Canetti and Krawczyk (CK) adversary model [27]. By the CK adversary model, a probabilistic polynomial-time adversary A can control the communication channel to achieve the function of listening, modification, and free interception. In addition, the secret information can be obtained by attackers and the session key also can be further damaged to create security threats during the communication process. A can launch the following query to interact with the protocol participant x, where x represents the SM_i or AG_j in this paper.

$Execute(SM_i, AG_j)$: A can only initiate a passive attack which is to eavesdrop information on the communication channel and will return the messages that participants exchange while executing this query.

$Send(x, m)$: Send query is defined on the basis of modification attacks, replay attacks, simulation attacks, etc. A can use this query to send a message m to x and and will receive a response message by x.

$h_n(m)$: By this query, A performs a hash query on the message m and receives a random number r_n as the hash value of m.

$Test(x)$: When obtaining a $Test(x)$ query, x returns its session key or the same random value of the participating session key. An unbiased coin $a \in \{0.1\}$ is flipped, if a is 1, the realistic session key is returned. Otherwise, x returns a arbitrary value with the same bit length of session key.

$Corrupt(x)$: By this query, A can obtain the static privacy of x to capture the concept of forward secrecy.

$ESReveal(x)$: With this query, A can get the brief secret held by x.

$SKReveal(x)$: A can get the session key of x through this query.

$Expire(x)$: In this query, the completed session key held by x is removed.

There are the following definitions found in terms for this model:

Definition 1. *If SM_i and AG_j in the receiving state can authenticate each other and establish a session key, they can be called partners.*

Definition 2. *If the $SKReveal(x)$ and $Corrupt(x)$ queries are made before the $Expire(x)$ query, the session s would be locally exposed. Conversely, if the session is not disclosed, it can be considered as having freshness.*

Definition 3. *The security of authenticated key agreement (AKA) is modeled by the game $Game^{AKA}(x, A)$ in which A can send out many queries to x. The purpose of A is to correctly guess the hidden bit $a \in \{0, 1\}$ through the $Test(x)$ query. It is assumed that Succ indicates the event where A wins and $Pr(Succ)$ indicates the probability of A wining the game $Game^{AKA}(x, A)$. Therefore, as shown in Equation (2), the advantage of the disruption AKA is defined as:*

$$Adv_{AKA}(A) = \left|2Pr\left[a' = a\right] - 1\right| = \left|2Pr[Succ(A)] - 1\right| \quad (2)$$

If there exists $\varepsilon > 0$ satisfying $Adv_{AKA}(A) < \varepsilon$, then we argue that our scheme is safe for the CK adversarial model.

3. The Proposed Eaka Scheme

As depicted in Figure 2, we describe the proposed EAKA scheme in details, the whole process of which includes three stages, respectively "system initialization", "registration", and "authentication and key agreement". Table 1 presents the notations applied in this paper.

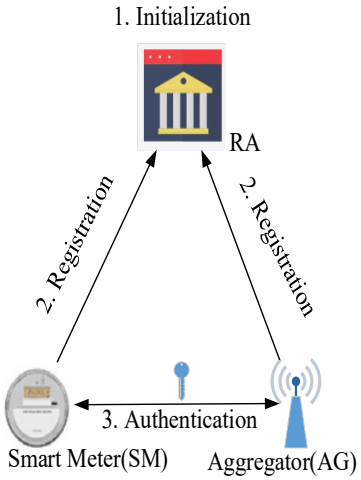

Figure 2. General overview of the proposed scheme.

Table 1. Notations.

Notations	Description
SM_i	The ith smart meter
AG_j	The jth service provider
(k, P_{pub})	Private and public keys of RA
RA	Trusted authority
id_i, id_j	The identity of SM_i and AG_j
(s_i, P_i)	Private and public keys of SM_i
$\left(s_j, P_j\right)$	Private and public keys of AG_j
P	The base point
SK	Shared session key
$\|\|$	The concatenation operator
$\oplus$	The exclusive-or operator

3.1. System Initialization

In this stage, RA selects and publishes the system parameters. The steps of this phase are as follows.

(1) RA chooses a large prime p on the non-singular elliptic curve $E_p(a, b)$, and a point $P \in E_p(a, b)$ as the base point or generator, RA also chooses a cyclic additive group G_1 and a multiplicative group $G_1 \times G_1 \rightarrow G_2$, then it calculates a bilinear mapping $e : G_1 \times G_1 \rightarrow G_2$, $g = e(P, P)$.

(2) RA randomly selects four one-way hash functions (h_0, h_1, h_2, h_3). RA selects a random number $k \in Z_q^*$ as its private key, then computes its own public key as $P_{pub} = kP$.

(3) RA publishes the system parameters $\left\{E_p(a, b), P, P_{pub}, h_0, h_1, h_2, h_3\right\}$.

3.2. Registration

After completing the registration stage, SM_i and AG_j calculate their private keys separately by the returned values from the RA.

(1) SM_i first sends its own id to RA via a secure channel.
(2) After obtaining the registration information, RA computes $D_i = k + h_0(id_i)$ and sends $\{D_i\}$ to SM_i through a safe channel.
(3) After getting $\{D_i\}$, SM_i computes its private key s_i as $s_i = \frac{1}{D_i}$ and public key P_i as $P_i = s_i P$.

Similarly, the same process is performed for the registration of AG_j. RA computes $D_j = k + h_0(id_j)$, and sends $\{D_j\}$ to AG_j through a secure channel. AG_j also gets its private key s_j and public key P_j as $s_j = \frac{1}{D_j}$, $P_j = s_j P$. Figure 3 and Algorithm 1, Figure 4 and Algorithm 2, illustrate the SM_i and AG_j registration processes respectively.

Algorithm 1. AG_j registration

Input: id_j;
Output: s_j, P_j;
1: $D_j = k + h_0(id_j)$;
2: **return** $AG_j \leftarrow (D_j)$;

Algorithm 2. SM_i registration

Input: id_i;
Output: s_i, P_i;
1: $D_i = k + h_0(id_i)$;
2: **return** $SM_i \leftarrow (D_i)$;

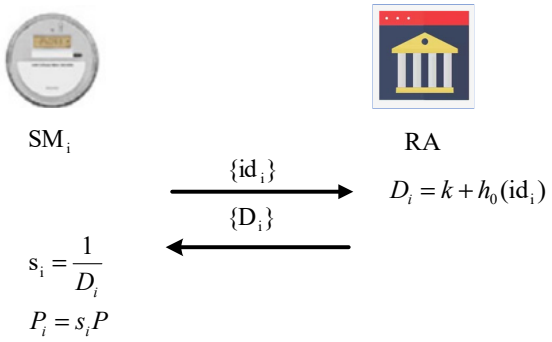

Figure 3. The process of SM_i registration.

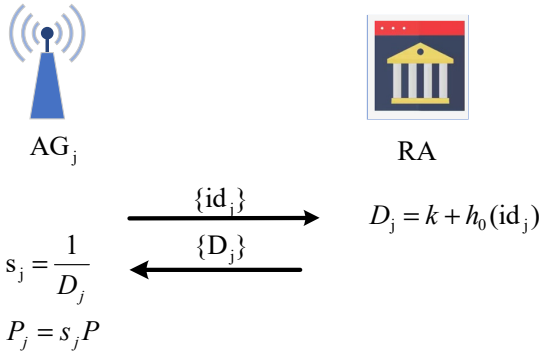

Figure 4. The process of AG_j registration.

3.3. Authentication and Key Agreement

As demonstrated in Figure 5, Algorithm 3 and Algorithm 4, AG_j and SM_i authenticate each other and generate a session key, and the two parties then communicate through the session key.

(1) At first, SM_i generates a random number $x_i \in Z_q^*$, and then computes $X_i = x_i P$, $g_1 = g^{x_i}$, $R_1 = x_i\left(P_{pub} + h_0(id_j)P\right)$, $M = (id_i||X_i) \oplus h_1(g_1\left|\right|t_i)$, where t_i is its current timestamp. Then SM_i sends $\{M, R_1, t_i\}$ to AG_j.

(2) When obtaining $\{M, R_1, t_i\}$ at the time t_i^*, AG_j verifies the freshness of t_i by checking $t_i^* - t_i ? < \Delta t$. If not, AG_j terminates the session. Otherwise, AG_j proceeds to compute $g_1 = e(R_1, P_j)$ shown in Equation (3). The value t_i and the resulting g_1 are hashed to calculate $id_i||X_i = M \oplus h_1(g_1||t_i)$. Then, AG_j checks whether $X_i D_j$ and R_1 are the same to verify the authenticity of SM_i. If it is false, AG_j breaks this procedure. Otherwise, AG_j generates a random number $x_j \in Z_q^*$ and computes $R_2 = x_j\left(P_{pub} + h_0(id_i)P\right)$, $C_1 = x_j R_1 s_j$. Then AG_j can get $V_{ij} = h_2(id_j||C_1||t_j)$ and the session key $SK = h_3(id_i\left|\right|id_j\left|\right|C_1)$. Finally, AG_j replies $\{V_{ij}, R_2, t_j\}$ to SM_i, where t_j is its current timestamp.

(3) When obtaining $\{V_{ij}, R_2, t_j\}$ at the time t_j^*, SM_i verifies the freshness of t_j by checking $t_j^* - t_j ? < \Delta t$. If not, SM_i terminates the session. Otherwise, SM_i proceeds to compute $C_2 = x_i R_2 s_i$ shown in Equations (4) and (5), where s_i is SM_i's private key. Then SM_i checks whether $h_2(id_j||C_2||t_j)$ and V_{ij} are the same to verify the authenticity of AG_j. If false, SM_i breaks this procedure. Otherwise SM_i generates the session key $SK = h_3(id_i\left|\right|id_j\left|\right|C_2)$. SM_i and AG_j complete the verification for both parties to obtain a common session key agreement.

$$\begin{aligned} g_1 = e(R_i, P_j) &= e\left(x_i\left(P_{pub} + h_0(id_j)P\right), s_j P\right) = e\left(x_i(k + h_0(id_j))P, \tfrac{1}{k+h_0(id_j)}P\right) \\ &= e(P,P)^{x_i(k+h_0(id_j))\cdot\frac{1}{k+h_0(id_j)}} = g^{x_i} \end{aligned} \tag{3}$$

$$C_1 = x_j R_1 s_j = x_j x_i\left(P_{pub} + h_0(id_j)P\right)s_j = x_i x_j(k + h_0(id_j))P\frac{1}{k + h_0(id_j)} = x_i x_j P \tag{4}$$

$$C_2 = x_j R_2 s_i = x_j x_i\left(P_{pub} + h_0(id_i)P\right)s_i = x_i x_j(k + h_0(id_i))P\frac{1}{k + h_0(id_i)} = x_i x_j P = C_1 \tag{5}$$

Algorithm 3. SM_i authenticates AG_j

Input: V_{ij}, R_2, t_j;
Output: accept or reject;
1: $x_i \in Z_q^*$, t_i; //generate a random number and a timestamp
2: $X_i = x_i P$;
3: $g_1 = g^{x_i}$;
4: $R_1 = x_i\left(P_{pub} + h\left(id_j\right)P\right)$;
5: $M = (id_i||X_i) \oplus H_1(g_1||t_i)$;
6: t_j^*; //generate a timestamp
7: **if**($t_j^* - t_j < \Delta t$) then
8: $C_2 = x_i R_2 s_i$;
9: **if**($V_{ij} = h_2(id_j||C_2||id_j)$ then
10: accept;
11: $SK = h_3\left(id_i\left|\right|id_j\left|\right|C_2\right)$;
12: **else** reject;
12: **else** reject;
13: **end if**

Algorithm 4. AG_j authenticates SM_i

Input: M, R_1, t_i;
Output: accept or reject;
1: t_i^*; //generate a timestamp
2: **if**$(t_j^* - t_j < \Delta t)$ then
3: $g_1 = e\left(R_1, P_j\right)$;
4: $id_i||X_i = M \oplus h_1(g_1||t_i)$;
5: **if**$(X_i D_j = R_1)$ then
6: accept;
7: $x_j \in Z_q^*$, t_j; //generate a random number and a timestamp
8: $R_2 = x_j\left(P_{pub} + h_0(id_i)P\right)$;
9: $C_1 = x_j R_1 s_j$;
10: $V_{ij} = h_2\left(id_j||C_1||t_j\right)$;
11: $SK = h_3\left(id_i\left|\right|id_j\left|\right|C_1\right)$;
12: **else** reject;
13: **else** reject;
14: **end if**

SM_i

AG_j

Select a random number $x_i \in z_q^*$

$X_i = x_i P$

$g_1 = g^{x_i}$

$R_1 = x_i(P_{pub} + h_0(id_j)P)$

$M = (id_i \| X_i) \oplus h_1(g_1 \| t_i)$

$\{M, R_1, t_i\}$ →

Check $t_i^* - t_i ? < \Delta t$, if true, proceed

$g_1 = e(R_1, P_j)$

$id_i \| X_i = M \oplus h_1(g_1 \| t_i)$

Check $X_i D_j ? = R_1$, if true, proceed

Select a random number $x_j \in z_q^*$

$R_2 = x_j(P_{pub} + h_0(id_i)P)$

$C_1 = x_j R_1 s_j$

$V_{ij} = h_2(id_j \| C_1 \| t_j)$

$SK = h_3(id_i \| id_j \| C_1)$

← $\{V_{ij}, R_2, t_j\}$

Check $t_j^* - t_j ? < \Delta t$, if true, proceed

$C_2 = x_i R_2 s_i$

Check $V_{ij} ? = h_2(id_j \| C_2 \| t_j)$

$SK = h_3(id_i \| id_j \| C_2)$

Figure 5. The authentication and key agreement phase.

4. Security Analysis

In this section, we evaluate our proposed EAKA scheme under the CK adversary model. By this model, an attacker A can perform a series of operations to achieve the effect of controlling communication. Besides, A can also interact with SM_i or AG_j.

4.1. Formal Evaluation by Random Oracle Model

By the random oracle model, all entities can interact with each other. Furthermore, they can also make oracle queries defined in Section II-C, whose questions are answered by a function uniformly selected among all possible functions. If any adversary has only a negligible probability of success with given abilities, the scheme is described as an ideal system.

Theorem 1. *The model supposes that A can fight the semantic security of the protocol and issue Execute query q_e, Send query q_s, and Hash query q_h. As shown in (3), the advantages of A are defined in Equation (6):*

$$Adv_{AKA}(A) \leq \frac{(q_s+q_e)^2}{p} + \frac{q_h^2+2q_s}{2^l} + 2q_h max\left\{Adv_{AKA}^{ECDHP}(A), Adv_{AKA}^{ECDLP}(A)\right\} \quad (6)$$

where l is the length of hash value.

Proof. The stochastic model defines the game sequence G_i (i = 0, 1, 2, 3, 4) to prove the semantic security of the protocol. G_0 indicates a real attack, while G_4 represents a game where A lacks the superiority. s_i represents the incident where A speculates the correct random number a in the Test query. □

Game G_0: This game is a simulation of a real attack by A under a random model. We can obtain Equation (7):

$$Adv_{AKA}(A) = |2Pr[s_0] - 1| \quad (7)$$

Game G_1: In this game, the query simulates a real attack. The simulation of the game which stores the results in the corresponding list is basically the same as the actual situation. If the result of the query is in the list, then we return it directly. If not, we output an arbitrary value of the same length as a result of the query and add it to the list. Thus, we have Equation (8):

$$Pr[s_1] = Pr[s_0] \quad (8)$$

Game G_2: The game G_2 is identical to the previous game simulation but G_2 is terminated if the value of the query conflicts with the list. Therefore, we can derive from the birthday paradox, the probability of hash collision is at most $q_h^2/2^{l+1}$, and the collision probability of transcription to the list is at most $(q_s+q_e)^2/2p$. So, we can obtain Equation (9):

$$|Pr[s_2] - Pr[s_1]| \leq \frac{q_h^2}{2^{l+1}} + \frac{(q_s+q_e)^2}{2p} \quad (9)$$

Game G_3: If A does not use the hash query, but A can also accurately speculate the validation value R_1 and V_{ij}, then the game will be suspended. Therefore, we can obtain Equation (10):

$$|Pr[s_3] - Pr[s_2]| \leq \frac{q_s}{2^l} \quad (10)$$

Game G_4: By this game, an analysis on the security of the session key can be made. A queries to $C_2 = x_i R_2 s_i$ to the random oracle h_3 in the test-session but it cannot implement the session key unless showing one of (x_i, s_i) or (x_j, s_j) to A. Therefore, A uses the query to calculate the session key with the following four scenarios:

(1) Corrupt (SM_i) and Corrupt (AG_j) By this, A can obtain the static private keys s_i of SM_i and s_j of AG_j.
(2) Corrupt (SM_i) and ESReveal (AG_j) By this, A can get the static private key s_i of SM_i and brief secret x_j of AG_j.
(3) ESReveal (SM_i) and Corrupt (AG_j) By this, A can obtain the brief secret x_i of SM_i and static private key s_j of AG_j.
(4) ESReveal (SM_i) and ESReveal (AG_j) By this, A can obtain the brief secret x_i of SM_i and brief secret x_j of AG_j.

According to $C_2 = x_i R_2 s_i$, the condition for A to obtain the session key is that A must know both the brief secret x_i and static private key s_i of SM_i. However, for the above four cases, the session key SK cannot be calculated by A without obtaining h_4 or solving

the ECDHP and ECDLP assumptions. If the ECDHP and ECDLP assumptions hold, the distinction between G_3 and G_4 is negligible [29]. So, we can get Equation (11):

$$|Pr[s_4] - Pr[s_3]| \leq q_h max\left\{Adv_{AKA}^{ECDHP}(A), Adv_{AKA}^{ECDLP}(A)\right\} \tag{11}$$

Otherwise, in the game G_4, the guess bit α is random and independent which is in the Test query. And there is no advantage to distinguish between real sessions and random keys for a query with incorrect input. Therefore, we have Equation (12):

$$Pr[s_4] = \frac{1}{2} \tag{12}$$

Finally, combining the above Equations (7)–(12), we can conclude that Equation (6) holds. So here exists $\varepsilon = \frac{(q_s+q_e)^2}{p} + \frac{q_h^2+2q_s}{2^l} + 2q_h max\{Adv_{AKA}^{ECDHP}(A), Adv_{AKA}^{ECDLP}(A)\} > 0$ satisfying $Adv_{AKA}(A) < \varepsilon$, then we conclude that our proposed EAKA scheme is safe for the CK adversarial model.

4.2. Informal Security Analysis

In this subsection, we qualitatively analyze the security features of the EAKA scheme as follows:

(1) Mutual authentication: First, SM_i can authenticate AG_j by checking whether $h_3(id_i||WT_i||C_2) = V_{ij}$ holds. With SM_i's private key s_i and random number x_i, SM_i can compute $C_2 = x_i R_2 s_i$ to verify the identity of AG_j. AG_j also confirms the identity of SM_i by verifying whether $R_1 = X_i D_j$ holds, where D_j is sent by RA. Both SM_i and AG_j have completed mutual certification.

(2) Key agreement: As we can see in Figure 5, after successful mutual authentication, both participants can get the same session key $SK = h_3(id_i||id_j||C_1) = h_3(id_i||id_j||C_2) = h_3(id_i||id_j||x_i x_j P)$, Under the premise that the assumptions of the ECDH problem [30] are established, the session key cannot be obtained by A.

(3) SM_i's identity anonymity: In the authentication phase, the id_i which is encrypted by $M = (id_i||X_i) \oplus h_1(g_1||t_i)$ is sent to AG_j on the open channel. The random number x_i is randomly generated and constantly changed in each session, M is dynamic so that it is different in each session. Therefore, SM_i can maintain the anonymity of the identity.

(4) Perfect forward confidentiality: According to $C_1 = x_j R_1 s_j$ and $SK = h_3(id_i||id_j||C_1)$, A needs to know the random number x_j and private key s_j of AG_j to get the session key. However, it is very hard for A to get x_i or x_j which are generated by SM_i or AG_j and guess the session key. Even if the static private key s_i and s_j of SM_i and AG_j can be obtained by A, x_i and x_j are different values generated randomly in each session, so any previous established session keys cannot be derived by A. Due to the difficulty of the ECDHP and ECDLP assumptions, the session key cannot be cracked without knowing the random number. Therefore, our proposed EAKA scheme provides perfect forward confidentiality.

(5) Man-in-the-middle attack: By this type of attack, A tries to establish a connection with AG_j and SM_i individually to make AG_j and SM_i mistakenly believe that both parties are connected. If A wants to establish connections with SM_i, it needs to obtain the random number x_j and private key s_j of AG_j to get the session key, but A cannot access the private key s_j. By the partial session key C_1 or C_2, A is unable to calculate the session key. Therefore, the analysis shows that our proposed EAKA scheme is able to resist man-in-the-middle attack.

(6) Replay attack: Replay attack is launched for spoofing hosts by sending previous data. A timestamp and random number mechanism is introduced in our scheme to cope with replay attack. SM_i generates a timestamp t_i, then AG_j verifies the freshness of the timestamp. If $t_i^* - t_i > \Delta t$, AG_j will discard this replay elimination. The replay

attack fails, even if t_i^* is modified by the adversary A because the original timestamp is embedded in $h_1(g_1||t_i)$, and the integrity of the time can be ensured by h_1. Similarly, SM_i verifies the freshness of the timestamp t_j which is sent by AG_j. SM_i and AG_j validate the timestamps sent by each other to ensure the freshness of the information in each data interaction. Therefore, our proposed EAKA scheme is considered feasible and effectively effective against replay attack.

(7) Key leakage attack: Even if A can obtain respectively the private key s_i of SM_i and private key s_j of AG_j during the communication, A cannot succeed in getting the session key. According to formula $C_1 = x_j R_1 s_j$ or $C_2 = x_i R_2 s_i$, A should know x_i or x_j to get the shared key besides the private key. Therefore, our proposed EAKA scheme can defend key leakage attack.

4.3. Comparison of Security Features

According to the security features, we compare our proposed EAKA scheme with the recently proposed solutions including the BAKA in [24], the TPPA in [25], and the SPAK in [26] schemes. As depicted in Table 2, the TPPA scheme lacks the session key security under the CK adversary model and it is unable to resist DoS attacks. In addition, the BAKA scheme has high computational and communication cost. The SPAK scheme cannot provide strong SM anonymity. Since the private key of our scheme is computed by itself in the registration phase, there is no key escrow issue. Therefore, our proposed EAKA scheme has better security properties with low computation and communication cost.

Table 2. Feature-based comparison with the related schemes.

Security Attributes	EAKA	BAKA	TPPA	SPAK
Authentication and key agreement	√	√	√	√
Providing perfect forward secrecy	√	√	√	√
Session key security under CK adversary model	√	√	×	√
DoS attack	√	√	×	√
No key escrow issue	√	×	√	×
Low computation and communication cost	√	×	√	√
Replay attack resistance	√	√	√	√
Free from the PKI challenges	√	√	√	√
Prividing strong SM anonymity	√	√	√	×

5. Performance Analysis

In this section, we evaluate the performance of the proposed EAKA scheme in terms of number of the number of cryptographic operations, computation delay, and communication cost. In addition, we also compare our EAKA scheme with other existing related solutions including the BAKA, the TPPA, and the SPAK schemes.

5.1. Number of Cryptographic Operations

We divide the cryptographic operations into five categories. PAD, HAS, EXP, BPA, and MUL representing point addition, hash operation, modular exponent, bilinear pairing, and scalar multiplication, respectively. The hardware of an SM as a user will use a RASPBERRY PI 3B+ with 1 GB LPDDR2 SDRAM and a BCM2837B0 system on chip with 1.4 GHZ frequency. The hardware of an AG as a server will use a computer with 4 GB RAM and an INTEL(R) CELERON (R) J1900 CPU. We adopt the encryption algorithm in [32–34] to simulate the time required to perform each of the cryptographic operations, which are shown in Table 3. From Table 3, the BPA operation spends the most time for users and the MUL operation also consumes a lot of time. So, in our scheme, high time-consuming encryption elements should be avoided as much as possible to reduce the total authentication latency without sacrificing the security features. From Table 4, each scheme uses different numbers of the encryption operations. We compare our scheme with the other four and find that they use the cryptographic operations more to increase their authentication delays. However,

our scheme avoids the number of password operations, which is the most time-consuming operation, to reduce the authentication delays.

Table 3. Execution time of basic operations.

Operation	Description	Google Nexus (ms)	Alibaba Cloud (ms)
T_{BPA}	The execution time of a bilinear pairing	48.66	5.275
T_{EXP}	The execution time of a modular exponentiation	3.328	0.339
T_{MUL}	The execution time of a scalar multiplication	19.919	1.97
T_{HAS}	The execution time of a general hash function	0.089	0.009
T_{PAD}	The execution time of a point addition	0.118	0.012

Table 4. Comparison of number of cryptography operations.

	EAKA		BAKA		TPPA		SPAK	
	User	Server	User	Server	User	Server	User	Server
BPA	0	1	0	0	0	0	1	1
MUL	3	3	6	6	4	4	2	2
PAD	1	1	2	2	3	4	3	3
EXP	1	0	0	0	2	2	1	1
HAS	4	4	4	4	4	6	6	6

5.2. Computation Delay

According to the data in Tables 3 and 4, we can simply calculate the time spent by each scheme for its cryptographic operations. The total number of cryptographic operations of the BAKA scheme is 24 with the time taken as 131.986 ms shown in Equation (14). The total number of password operations of the TPPA scheme is 29 with the time consumed as 95.702 ms shown in Equation (15). The total number of cryptographic operations of the SPAK scheme is 26 with the time consumed as 102.358 ms, shown in Equation (16). The total number of cryptographic operations performed by the proposed EAKA scheme is 18 with the time consumed as 74.792 ms shown in (13), which is 43% lower than that of the BAKA scheme, 22% lower than that of the TPPA scheme, and 27% lower than that of the SPAK scheme. The three schemes use more encryption components, which may increase their authentication delays.

$$T_{EAKA} = 3T_{MUL}^{SM} + 3T_{MUL}^{AG} + 4T_{HAS}^{SM} + 4T_{HAS}^{AG} + T_{EXP}^{SM} + T_{BPA}^{AG} + T_{PAD}^{SM} + T_{PAD}^{AG} = 74.792\ ms \quad (13)$$

$$T_{BAKA} = 6T_{MUL}^{SM} + 6T_{MUL}^{AG} + 4T_{HAS}^{SM} + 4T_{HAS}^{AG} + 2T_{PAD}^{SM} + 2T_{PAD}^{AG} = 131.986\ ms \quad (14)$$

$$T_{TPPA} = 4T_{MUL}^{ES} + 4T_{MUL}^{SM} + 4T_{PAD}^{ES} + 3T_{PAD}^{SM} + 6T_{HAS}^{ES} + 4T_{HAS}^{SM} + 2T_{EXP}^{ES} + 2T_{EXP}^{SM} = 95.702\ (ms) \quad (15)$$

$$T_{SPAK} = 2T_{MUL}^{ES} + 2T_{MUL}^{SM} + 3T_{PAD}^{ES} + 3T_{PAD}^{SM} + 6T_{HAS}^{ES} + 6T_{HAS}^{SM} + T_{EXP}^{ES} + T_{EXP}^{SM} + T_{BPA}^{ES} + T_{BPA}^{SM} = 102.358\ (ms) \quad (16)$$

In addition, Figure 6 details the computation delay of the SM and the AG by each scheme. On the SM side, the authentication delay by the BAKA scheme is the largest reaching 120.106 ms. The delay of the TPPA scheme and the delay of the SPAK scheme is 87.042 ms and 92.714 ms, respectively, while the delay of the EAKA scheme is minimal. On the AG side, the delay by the BAKA scheme is the maximum as 11.88 ms, and the delay by the TPP scheme and the delay by the SPAK schemes is 8.66 ms and 9.644 ms, respectively. The delay by our scheme is slightly higher at 11.233 ms. The result shows that our scheme meets the low time-consuming requirements of the SG.

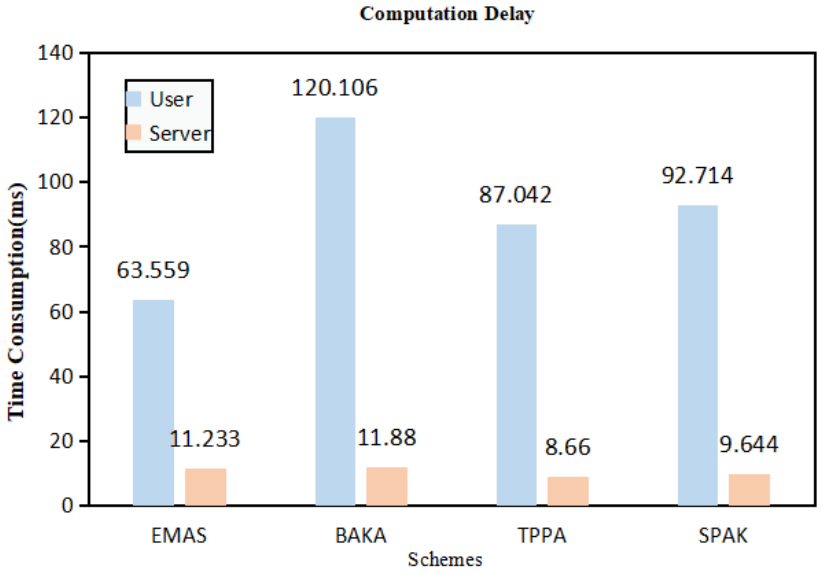

Figure 6. Computation delay of each scheme in authentication phase.

C++ encoding is used to simulate the attack environment for security validation. Since there are some new types of malicious attacks which are unpredictable, and the authentication process of the four schemes may be interrupted by those unknown attacks, we simulate the attack environment as constantly changing the proportion of unknown attacks. Those attacks that can be resisted through security analysis are known as known attacks. The emergence of some new malicious attacks is unpredictable, and all these potential attacks are called unknown attacks. Assume the unknown attacks can interrupt the authentication process of these four methods. The computation delay is fixed for each type of the scheme under a known attack, while it could be uncertain under an unknown attack. We compare the computation delay in different scenarios and perform a total of 10,000 validation procedures for these four schemes by constantly varying the proportion of attack types to analyze the performance and validation delays of the different schemes.

As depicted in Figure 7, we perform specific simulations for each scheme, as the relationship between the different ratios of unknown attacks and the average computation delay by each scheme. The abscissa indicates the ratio of unknown attacks and the ordinate denotes the authentication time consumption by each of the four schemes. The ratio of unknown attacks increases from 0.1 to 0.9. Figure 7 shows that when the ratios of unknown attacks keep getting larger, the average authentication delay of each scheme also keeps increasing. The proposed EAKA scheme always has the lowest computation delay when the unknown attacks ratio is increasing. When the ratio of unknown attacks is the same, the computation delay of the proposed EAKA scheme is the lowest. Therefore, the proposed EAKA scheme has efficiency advantages even under different unknown attacks.

5.3. Communication Cost

The communication cost is also one of the important measures to evaluate the quality of a solution. We assume that the identity and the bit length of EXP are 64 b, the time stamp is 32 b, 160 b for random number verification and hash function, the operation for each point on the elliptic curve is 161 b, and an element in the multiplication group is 512 b. The statistical results of the proposed EAKA, the BAKA, the TPPA, and the SPAK schemes are respectively 706 b, 1027 b, 966 b, and 962 b which is depicted in Figure 8. The proposed EAKA scheme has a 31% lower communication cost than the BAKA scheme and 27% lower than the TPPA scheme and the SPAK scheme. Thus, the results indicate that the EAKA scheme has certain superiorities in terms of communication cost.

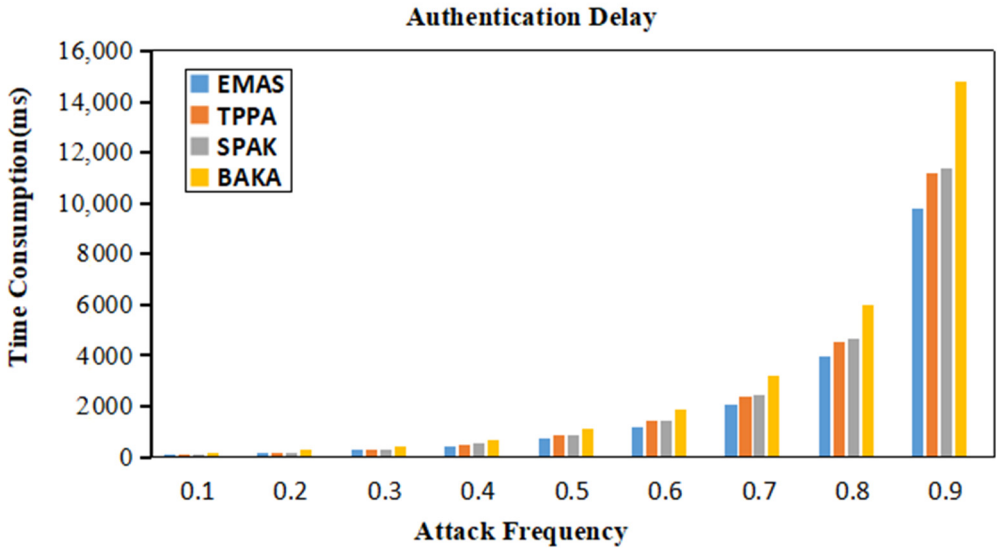

Figure 7. Authentication delay of each scheme under different attack frequency.

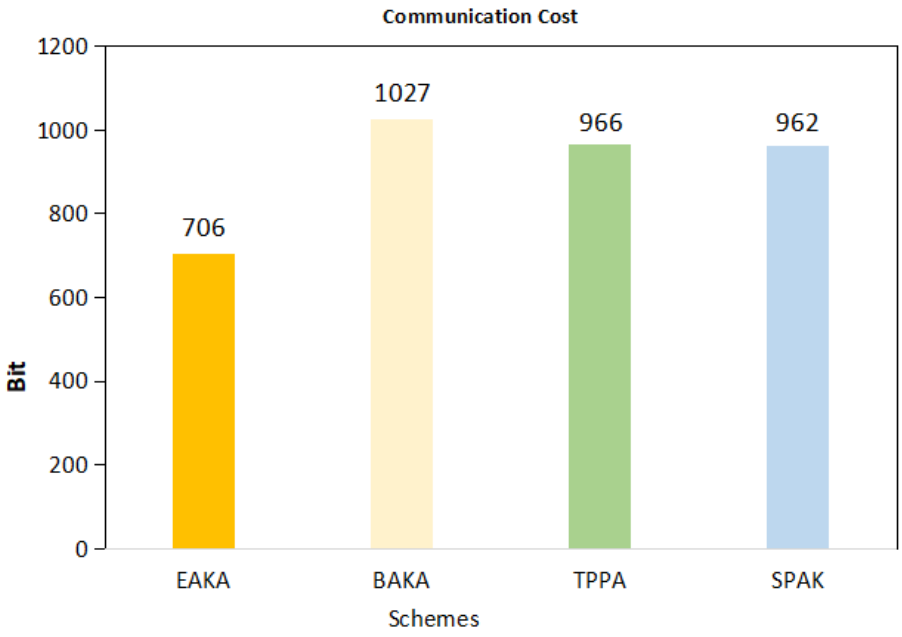

Figure 8. Communication cost of each scheme in authentication phase.

6. Conclusions

In this paper, we proposed a two-way anonymous authentication scheme based on ECC for the communications for smart metering in SG. The scheme adopts a self-authentication method to resist simulated attacks and provides the maximum protection in the authentication process. For the safety of the proposed EAKA scheme, we conducted a qualitative analysis. The proposed EAKA scheme can provide session key agreement, perfect forward secrecy, and privacy protection of *SM*. In addition, we evaluated the performance of the EAKA scheme by comparing it to other existing solutions to conclude that the proposed EAKA scheme cannot only incur a low computation delay but also can realize all the security functions provided by other schemes. In future research, we will introduce a trust-based weighted assessment pseudonymous to realize the secure storage control of distributed trust data, and we will design a comprehensive trust model approach to better study this part of identity authentication.

Author Contributions: Conceptualization, C.W.; Methodology and writing, S.L.; Formal analysis, M.M.; Data curation, Y.Z.; Investigation, X.T.; Supervision, B.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Orr, F.; Nafees, M.N.; Saxena, N.; Choi, B.J. Securing Publisher-Subscriber Smart Grid Infrastructure. *Electronics* **2021**, *10*, 2355. [CrossRef]
2. Shokry, M.; Awad, A.I.; Abd-Ellah, M.K.; Khalaf, A.A.M. Systematic survey of advanced metering infrastructure security: Vulnerabilities, attacks, countermeasures, and future vision. *Future Gener. Comput. Syst.* **2022**, *136*, 358–377. [CrossRef]
3. Barreto, R.; Faria, P.; Vale, Z. Electric Mobility. An Overview of the Main Aspects Related to the Smart Grid. *Electronics* **2022**, *11*, 1311. [CrossRef]
4. Je, S.M.; Woo, H.; Choi, J.; Jung, S.H.; Huh, J.H. A Research Trend on Anonymous Signature and Authentication Methods for Privacy Invasion Preventability on Smart Grid and Power Plant Environments. *Energies* **2022**, *15*, 4363. [CrossRef]
5. Gope, P.; Sikdar, B. Lightweight and Privacy-Friendly Spatial Data Aggregation for Secure Power Supply and Demand Management in Smart Grids. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 1554–1566. [CrossRef]
6. Zhang, H.; Wang, J.; Ding, Y. Blockchain-based decentralized and secure keyless signature scheme for smart grid. *Energy* **2019**, *180*, 955–967. [CrossRef]
7. Kumar, P.; Gurtov, A.; Sain, M.; Martin, A.; Ha, P. Lightweight Authentication and Key Agreement for Smart Metering in Smart Energy Networks. *IEEE Trans. Smart Grid* **2018**, *10*, 4349–4359. [CrossRef]
8. Odelu, V.; Das, A.K.; Wazid, M.; Conti, M. Provably Secure Authenticated Key Agreement Scheme for Smart Grid. *IEEE Trans. Smart Grid* **2018**, *9*, 1900–1910. [CrossRef]
9. Braeken, A.; Kumar, P.; Martin, A. Efficient and Provably Secure Key Agreement for Modern Smart Metering Communications. *Energies* **2018**, *11*, 2662. [CrossRef]
10. Xu, G.; Li, X.; Jiao, L.; Wang, W.; Liu, A.; Su, C.; Zheng, X.; Liu, S.; Cheng, X. BAGKD: A Batch Authentication and Group Key Distribution Protocol for VANETs. *IEEE Commun. Mag.* **2020**, *58*, 35–41. [CrossRef]
11. Xiang, A.; Zheng, J. A situation-aware scheme for efficient device authentication in smart grid-enabled home area networks. *Electronics* **2020**, *9*, 989. [CrossRef]
12. Kaveh, M.; Martín, D.; Mosavi, M.R. A Lightweight Authentication Scheme for V2G Communications: A PUF-Based Approach Ensuring Cyber/Physical Security and Identity/Location Privacy. *Electronics* **2020**, *9*, 1479. [CrossRef]
13. Mall, P.; Amin, R.; Das, A.K.; Leung, M.T.; Choo, K.K.R. PUF-Based Authentication and Key Agreement Protocols for IoT, WSNs, and Smart Grids: A Comprehensive Survey. *IEEE Internet Things J.* **2022**, *9*, 8205–8228. [CrossRef]
14. Dariush, A.M.; Morteza, N. Design and hardware implementation of a security-enhanced elliptic curve cryptography based lightweight authentication scheme for smart grid communications. *Future Gener. Comput. Syst.* **2018**, *84*, 47–57.
15. Srinivas, J.; Das, A.K.; Li, X.; Khan, M.K.; Jo, M. Designing Anonymous Signature-Based Authenticated Key Exchange Scheme for IoT-Enabled Smart Grid Systems. *IEEE Trans. Ind. Inform.* **2021**, *17*, 4425–4436. [CrossRef]
16. Chaudhry, S.A. Correcting PALK: Password-based anonymous lightweight key agreement framework for smart grid. *Int. J. Electr. Power Energy Syst.* **2021**, *125*, 106529. [CrossRef]
17. Khan, A.A.; Kumar, V.; Ahmad, M.; Rana, S.; Mishra, D. PALK: Password-based anonymous lightweight key agreement framework for smart grid. *Int. J. Electr. Power Energy Syst.* **2021**, *121*, 106121. [CrossRef]
18. Khan, A.A.; Kumar, V.; Ahmad, M.; Rana, S. LAKAF: Lightweight authentication and key agreement framework for smart grid network. *J. Syst. Archit.* **2021**, *116*, 102053. [CrossRef]
19. Sureshkumar, V.; Anandhi, S.; Amin, R.; Selvarajan, N.; Madhumathi, R. Design of Robust Mutual Authentication and Key Establishment Security Protocol for Cloud-Enabled Smart Grid Communication. *IEEE Syst. J.* **2021**, *15*, 3565–3572. [CrossRef]
20. Baghestani, S.H.; Moazami, F.; Tahavori, M. Lightweight Authenticated Key Agreement for Smart Metering in Smart Grid. *IEEE Syst. J.* **2022**, *16*, 4983–4991. [CrossRef]
21. Chaudhry, S.A.; Nebhan, J.; Yahya, K.; Al-Turjman, F. A privacy enhanced authentication scheme for securing smart grid infrastructure. *IEEE Trans. Ind. Inform.* **2021**, *18*, 5000–5006. [CrossRef]
22. Taqi, S.A.M.; Jalili, S. LSPA-SGs: A lightweight and secure protocol for authentication and key agreement based Elliptic Curve Cryptography in smart grids. *Energy Rep.* **2022**, *8*, 153–164. [CrossRef]
23. Li, K.; Shi, R.; Wu, M.; Li, Y.; Zhang, X. A novel privacy-preserving multi-level aggregate signcryption and query scheme for Smart Grid via mobile fog computing. *J. Inf. Secur. Appl.* **2022**, *67*, 103214. [CrossRef]

24. Wang, J.; Wu, L.; Choo, K.-K.R.; He, D. Blockchain-Based Anonymous Authentication with Key Management for Smart Grid Edge Computing Infrastructure. *IEEE Trans. Ind. Inform.* **2020**, *16*, 1984–1992. [CrossRef]
25. Qi, M.; Chen, J. Two-Pass Privacy Preserving Authenticated Key Agreement Scheme for Smart Grid. *IEEE Syst. J.* **2021**, *15*, 3201–3207. [CrossRef]
26. Xiang, X.Y.; Cao, J. An efficient authenticated key agreement scheme supporting privacy-preservation for smart grid communication. *Electr. Power Syst. Res.* **2022**, *203*, 107630. [CrossRef]
27. Canetti, R.; Krawczyk, H. Analysis of key-exchange protocols and their use for building secure channels. *Theory Appl. Cryptogr. Tech.* **2001**, *2045*, 453–474.
28. Liberati, F.; Garone, E.; di Giorgio, A. Review of Cyber-Physical Attacks in Smart Grids: A System-Theoretic Perspective. *Electronics* **2021**, *10*, 1153. [CrossRef]
29. Liu, X.X.; Ma, W.P.; Cao, H. NPMA: A Novel Privacy-Preserving Mutual Authentication in TMIS for Mobile Edge-Cloud Architecture. *J. Med. Syst.* **2019**, *43*, 318. [CrossRef]
30. Chande, M.K.; Lee, C.C.; Li, C.T. Cryptanalysis and improvement of a ECDLP based proxy blind signature scheme. *J. Discret. Math. Sci. Cryptogr.* **2018**, *21*, 23–34. [CrossRef]
31. Kumar, P.; Lin, Y.; Bai, G.; Paverd, A.; Dong, J.S.; Martin, A. Smart Grid Metering Networks: A Survey on Security, Privacy and Open Research Issues. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2886–2927. [CrossRef]
32. Arduinolibs: Cryptographic Library. Available online: http://rweather.github.io/arduinolibs/crypto.html (accessed on 2 October 2017).
33. OpenSSL, Cryptography and SSL/TLS Toolkit. Available online: http://www.openssl.org (accessed on 1 April 2017).
34. Wang, C.; Zhang, Y.; Chen, X.; Liang, K.; Wang, Z. SDN-Based Handover Authentication Scheme for Mobile Edge Computing in Cyber-Physical Systems. *IEEE Internet Things J.* **2019**, *6*, 8692–8701. [CrossRef]