electronics

MDPI

Article

A Privacy-Preserving Authentication and Key Agreement Scheme with Deniability for IoT

Yousheng Zhou [1,2,*], Tong Liu [1], Fei Tang [1,2], Feng Wang [3] and Magara Tinashe [1]

1 College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; s160201042@stu.cqupt.edu.cn (T.L.); tangfei@cqupt.edu.cn (F.T.); L201810009@stu.cqupt.edu.cn (M.T.)
2 School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
3 College of Mathematical Sciences, De Zhou University, Shandong 253023, China; flys99@126.com
* Correspondence: zhouys@cqupt.edu.cn

Received: 23 March 2019; Accepted: 15 April 2019; Published: 19 April 2019

Abstract: User authentication for the Internet of Things (IoT) is a vital measure as it consists of numerous unattended connected devices and sensors. For security, only the user authenticated by the gateway node can access the real-time data gathered by sensor nodes. In this article, an efficient privacy-preserving authentication and key agreement scheme for IoT is developed which enables the user, the gateway node and sensor nodes to authenticate with each other. Only the trusted gateway node can determine the real identity of user; however, no other entities can get information about user' identity by just intercepting all exchanged messages during authentication phase. The gateway cannot prove the received messages from the sender to a third party, and thus preserving the privacy of the sender. The correctness of the proposed scheme is proved to be feasible by using BAN logic, and its security is proved under the random oracle model. The execution time of the proposed scheme is evaluated and compared with existing similar schemes, and the results demonstrate that our proposed scheme is more efficient and applicable for IoT applications.

Keywords: IoT; security; authentication; anonymity; deniability

1. Introduction

The Internet of Things (IoT) [1] is an enormous ubiquitous-network which is connecting the objects through various sensor devices and networks. It plays an important role in people's lives and has been widely used in many fields to gather data such as transportation [2], education, smart healthcare [3–5], logistics, etc. In general, the network of IoT is formed by end-users, sensors and base stations (e.g., gateway), in which sensors can collect data of specific areas around them and then end users can access data on demand through the network.

However, the IoT is vulnerable to lots of malicious attacks due to its inherent the computational constraints of the sensors and the openness of wireless channel in IoT environment [1]. It is becoming a principal security concern that how to ensure that only valid end-users can access the critical data. To address this problem effectively, several authentication mechanisms [6–9] have been proposed to guarantee the authenticity of entities as well as the confidentiality of transferred data during communication in IoT. In an IoT environment, there exist three types entities, i.e., users, gateways and sensors. The gateways are specific modes which are served as trusted servers during authentication. Then sensors locate in various application environment to collect data. The user can access data in sensors while he or she has been authenticated by gateway. The basic goal of authentication is to enable gateway nodes, end-users and sensor nodes to authenticate each other. In order to meet functionality

and security requirements, however, designing authentication and key agreement schemes to guarantee secure communication for the Internet of Things is challenging.

User authentication is vital in the IoT environment since it is used to distinguish legitimate users from illegal users. Only legitimate users can be granted with permission to access the data collected by sensor devices. Over the past few years, many user authentication schemes about the IoT environments have been designed. For example, Wong et al. [10] in 2006 put forward a user authentication protocol using symmetric encryption which utilizes hash and XOR operations to lower the computational complexity. Later, Das [11] presented an improved password-based user authentication to enhance the security of Wong et al.'s scheme. [10]. Other scholars [12–14] revealed that Wong et al.'s scheme is short of providing user anonymity and mutual authentication. Due to the merits of identity-based cryptography, some researchers presented novel identity-based authentication schemes [15,16], however, the computational cost in these schemes are expensive because of the adoption of pairing operation. Taking account of many existing construction of classic authentication schemes are based on public key technique, some researchers adopted symmetric cryptography-based means to improve the performance of authentication. Jung et al. [17] proposed a user anonymous authentication scheme based on symmetric encryption, which uses dynamic ID to achieve anonymity. Considering mutual authentication is important in some IoT applications, Xue et al. [18] constructed a user authentication scheme based on temporal-credential where the gateway node issues temporary certificates to the user and sensor nodes to achieve mutual authentication. Jiang et al. [19] pointed out that Xue et al.'s scheme fails to resist privileged-insider attack and then proposed an improved signature-based authentication scheme. Das [20] introduced an enhanced three-factor user authentication scheme based on Jiang et al.'s [19] work using user biometric information.

Since privacy plays a central role in designing authentication and key agreement schemes, and great efforts have been made in privacy-preserving authentication. Fox example, in 2015, Wang et al. [21] presented a new authentication scheme for wireless body area networks(WBANs) using bilinear pairing to achieve anonymity; however, it is vulnerable to the impersonation attack. Li et al. [22] proposed an anonymous authentication scheme using the hash message authentication code (HMAC). However, it is infeasible for the limited IoT environment since the bilinear pairing would bring enormous costs. Porambage et al. [23] presented an ECC-based authentication protocol without bilinear pairing to achieve high efficiency. Some signature-based authentication schemes [24,25] have been investigated besides interactive protocol-based authentication schemes.

The previous work has proposed different methods to ensure security and to meet the functionality requirements. However, most of the existing schemes have weaknesses, such as high computation overhead, being susceptible to some attacks or not providing user privacy-preserving. Furthermore, all these existing schemes fail to deal with deniability and traceability at the same time, which looks contradicts with each other. Deniability is essential for users in IoT environment to preserve her or his privacy, however, traceability is vital to prevent malicious entities to damage the IoT applications. Hence, based on the previous work, we propose an ECC-based privacy-preserving authentication and key agreement scheme for IoT, which aims to provide conditional privacy protection and desirable performance.

This paper presents a privacy-preserving authentication and key agreement scheme with deniability for IoT, which enables user to access IoT sensor securely. More specifically, the scheme meets appropriate security requirements and supports desirable features. The characteristics of our proposal are as follows:

1. **User anonymity**. No entity except the trusted gateway nodes can obtain any information about the identity of the users during the authentication phase.
2. **Deniability**. The gateway node can generate another message that is indistinguishable from the received message from the user, such that when the user request a service via the gateway node, any third party cannot tell whether the message is sent by the user or generated by the gateway node. Therefore, the user can deny that he or she has requested the service.

3. **Unlinkability**. Any external entity except the trusted gateway node cannot determine whether two messages from distinguished authentication sessions are sent by the same entity.
4. **Traceability**. If any dispute or misbehavior occurs during the authentication phase, the trusted gateway node can reveal the identity of the user with the exchanged messages.
5. **High-efficiency**. Due to the adoption of low-cost hash functions and ECC(elliptic curve cryptography) operations, the proposed scheme is more efficient than the existing exponential or bilinear pairing-based authentication schemes.

The remainder of this article is structured as follows. Section 2 provides related preliminaries. The concrete construction of the proposed scheme is described in Section 3. Section 4 presents a rigorous security analysis about the proposed scheme. Section 5 conducts the performance evaluation. Conclusions of the paper are presented in Section 6.

2. Preliminaries

In this section, some basic knowledge including communication model, the random oracle model and elliptic curve discrete logarithm problem are introduced.

2.1. Communication Model

The communication model of our proposed scheme is shown in Figure 1. It includes three kinds of entities: the gateway node GWN, the user U and the sensor node S. A secure communication channel can be established between U and S. Once the user U intends to request a certain service or access the data via GWN, the authentication session is initiated. U first sends an authentication request the message $M1$ to GWN which requests GWN for authentication; after checking the validity of messages from U, GWN sends the message $M2$ to S. When receives the message $M2$ from GWN, S replies the confirmation message about session key establishment with message $M3$ to GWN. Then GWN verifies $M3$, generates and sends the message $M4$ including the message $M3$ to U. At last, after U authenticating GWN and S, U securely establishes a session key with S successfully.

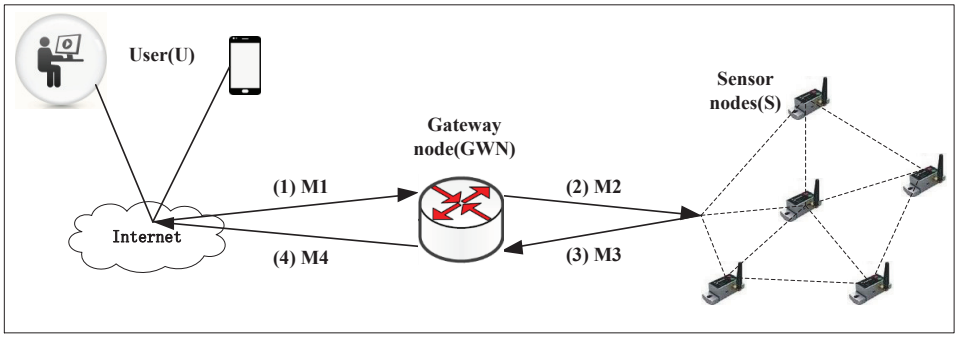

Figure 1. Communication model of the proposed scheme.

2.2. Security Definition

The secrecy of the session key is the central security goal for authentication and key agreement scheme. To formally prove the security, a game-based method is introduced in our paper based on Abdalla et al.'s [26] method. The security model of our proposed scheme is introduced as follows.

Participants. There are three types of participants: users, gateway nodes and sensor nodes. Let $\prod_P^n$ be the instance n of the participants such that $P \in \{U, G, S\}$, where U, G, S represent users, gateway nodes and sensor nodes respectively. Let $\prod_S^j$ represent the j-th instance of S, $\prod_U^i$ denote the i-th instance of U, and $\prod_G^n$ represent the k-th instance of G. Any participant instance is assumed as an oracle.

Partnering. Let *sid* denote the session identification which is unique for each conversation. If the instances $\prod_U^i$ and $\prod_S^j$ are called partners, then the following conditions would be satisfied: (1) A same

sid between $\prod_U^i$ and $\prod_S^j$ is shared; (2) $\prod_U^i$ and $\prod_S^j$ have accepted the conversation; (3) $\prod_U^i$ and $\prod_S^j$ are each other's partners.

Adversary. It is assumed that there exists a probabilistic polynomial-time(PPT) adversary $\mathcal{A}$ that can fully control all the communications by accessing to a series of oracle queries during the execution of the protocol. All the adversary's queries are listed as below:

- $Execute(\prod_U^i, \prod_G^n, \prod_S^j)$: This query issued by the adversary $\mathcal{A}$ simulates the eavesdropping attacks on honest executions among the user instance $\prod_U^i$, trusted gateway instance $\prod_G^n$ and sensor instance $\prod_S^j$. It outputs a transcript of the exchanged messages during the honest execution of the protocol.
- $Send(\prod_P^n, M)$: This query models the active attacks such as impersonation attack and replay attack. Once has received the messages, $\prod_P^n$ returns a corresponding result to $\mathcal{A}$.
- $Corrupt(\prod_P^n)$: This query is issued by the adversary $\mathcal{A}$, it is used to simulate the attack that $\mathcal{A}$ corrupts an entity from $\prod_P^n$. $\mathcal{A}$ can get the private key of a participant with this query.

  Please note that this query does not corrupt the partner's same internal data and ephemeral values of the instance $\prod_P^n$.

- $Reaveal$ $(\prod_P^n)$: The query is designed to simulate known session key attack. If there is a valid session from the instance $\prod_P^n$, returns the shared session key to $\mathcal{A}$. Otherwise, returns null.
- $Test(\prod_P^n)$: This query is used to model the capability of the adversary $\mathcal{A}$ to distinguish between a random number and a real session key SK by flipping an unbiased coin b. If the session key of the instance $\prod_P^n$ has been defined, the session key of $\prod_P^n$ will be responded to $\mathcal{A}$ if $b = 1$ or a random value will be returned if $b = 0$; otherwise, $\perp$ will be responded.
- $H_1(x, v_1)$: As soon as the adversary $\mathcal{A}$ makes H_1 query adaptively on the message x, it returns the existing v_1 if the list L_1 exist a tuple $\{x, v_1\}$, where L_1 initially is an empty set; otherwise, it picks a random value $v_1^{'}$, stores the tuple $\left\{x, v_1^{'}\right\}$ in the list L_1 and returns $v_1^{'}$ to $\mathcal{A}$.
- $H_2(y, v_2)$: Upon receiving the query about y from the adversary $\mathcal{A}$, examines whether the tuple $\{y, v_2\}$ is in L_2, where L_2 initially is an empty set. If so, it responds to the existing v_2 to $\mathcal{A}$; otherwise, it generates a random value $v_2^{'}$, stores the tuple $\left\{y, v_2^{'}\right\}$ in the list L_2 and returns $v_2^{'}$ to $\mathcal{A}$.

The adversary $\mathcal{A}$ could issue any *Test* query to the instances after being provided with the above queries. The output of *Test* query is relevant to the bit b. At last, $\mathcal{A}$ outputs a guessing bit $b^{'}$ about b. $\mathcal{A}$ is successful if $b^{'} = b$. Let $Succ$ represent the event that $\mathcal{A}$ succeeds in the game, the advantage of the adversary $\mathcal{A}$ is defined as follows:

$$Adv_{\mathcal{A}}^{Ind} = \mid 2 \cdot Pr[Succ] - 1 \mid$$

If the advantage $Adv^{ake}(\mathcal{A})$ is negligible, then we conclude that the proposed scheme is secure.

2.3. Elliptic Curve Discrete Logarithm Problem

Let G be a cyclic additive elliptic curve group with the prime order q and P is a generator of G. Suppose that the multiplication and inversion operation in G can be computed efficiently, the two intractable problems in G are defined as follows:

- **Elliptic curve discrete logarithm** (ECDL) problem: Given $P, aP \in G$ for unknown $a \in Z_q^*$, to find a.
- **Elliptic curve computational Diffie-Hellman** (ECCDH) problem: Given P, aP, $bP \in G$ for unknown $a, b \in Z_q^*$, to compute abP.

3. The Proposed Scheme

In this section, we describe the proposed scheme in detail. It consists of four phases: system set up, user registration, sensor node registration and authentication phase. Table 1 summarizes all the notations used in this paper.

Table 1. NOTATIONS.

Symbol	Definition
$E_p(a,b)$	An elliptic curve over a prime finite Z_p defined by the equation $y^2 = x^3 + ax + b \ mod \ p$
G	An elliptic curve group with the order q, where G is constitutive of all points on E and the point at infinity O
P	A generator of the group G
p, q	Two large prime numbers
U	User
S	Sensor node
GWN	Gateway node
ID_U	Identity of the user U
ID_S	Identity of the sensor node S
h, H_1, H_2	Three collision-resistant one-way hash functions, where $h : \{0,1\}^* \rightarrow z_q^*$, $H_1 : \{0,1\}^* \rightarrow z_q^*$,$H_2 : \{0,1\}^* \rightarrow z_q^*$
$P = \left(P^{(x)}, P^{(y)}\right)$	An elliptic curve point in a non-singular elliptic curve $E_p(a,b)$, $P^{(x)}$ and $P^{(y)}$ are x and y coordinates of P respectively
d_{GWN}, Q_{GWN}	The private key and the corresponding public key of GWN respectively
d_U, Q_U	The private key and the corresponding public key of U respectively
d_S, Q_S	The private key and the corresponding public key of S respectively
r	The random number selected by involved entities
t_U, t_{GWN}, t_S	The time stamps of U, GWN, S respectively
Δt	Maximum transmission delay
$\oplus$	The XOR operation
$\|$	Thet concatenation operation

3.1. System Setup Phase

System setup is performed by GWN as follows,

1. GWN chooses a non-singular elliptic curve $E_p(a,b)$ over a prime finite Z_p, where p is a large prime. Let G be an elliptic curve group. Then, GWN chooses a generator P of order q over E_p. GWN selects its private key d_{GWN} and computes the public key $Q_{GWN} = d_{GWN}P$ in accordance with d_{GWN}.
2. GWN selects three collision-resistant one-way hash functions $h, H_1, H_2 : \{0,1\}^* \rightarrow Z_q$.
3. Finally, the system parameters $params = \{E_p(a,b), P, p, q, h, H_1, H_2, Q_{GWN}\}$ is published while the private key d_{GWN} is kept secretly by GWN.

3.2. Registration Phase

A user U registers at the gateway node GWN in line with the requirement, while a regular sensor node S registers at GWN offline. A detailed process of registration process about U and S is highlighted as below.

3.2.1. User Registration Phase

The registration process is between the GWN and U is as follows:

1. U selects an identity ID_U, a private key d_U and then gets the public key $Q_U = d_U P$ according to d_U. Then, U calculates the registration message $MID_U = h\left(ID_U\right)$, and sends it to GWN via a non-public channel.

2. After receiving the registration message from U, GWN calculates $M_U = h\left(MID_U \parallel d_{GWN}\right)$ and returns it to U via a non-public channel.
3. U computes $M_U^* = M_U \oplus h\left(ID_U \parallel d_U\right)$ and deletes M_U.

3.2.2. Sensor Node Registration Phase

S proceeds offline registration with the help of GWN as below:

1. S generates its identity ID_S, private key d_S and computes the corresponding public key $Q_S = d_S P$ and $h(ID_S \parallel d_S)$. Then, S sends $\{ID_S, Q_S, h(ID_S \parallel d_S)\}$ to GWN via a non-public channel.
2. After receiving the message $\{ID_S, Q_S, h(ID_S \parallel d_S)\}$ from S, GWN computes $R_S = \left(h\left(ID_S \parallel d_S\right) + h\left(ID_S \parallel d_{GWN}\right)\right) P$ and sent it to S. GWN publish Q_S and stores $\{ID_S, Q_S, R_S\}$ into its database.
3. Upon receiving R_S from GWN, S stores it into its memory.

3.3. Authentication and Key Agreement Phase

When the user U wants to access the sensor node S, he or she initiates this phase by issuing a request via GWN. This phase enables GWN, U and S to effectively authenticate each other and then establish a session key between U and S. If a session key is negotiated successfully by U and S, then they can exchange private messages with each other via a public channel. A detailed description of the steps of this phase are as follows:

1. U selects a random number $r_U \in z_q^*$, generates the current timestamp t_1 and computes $E_U = r_U P$, $M_U^{'} = M_U^* \oplus h\left(ID_U \parallel d_U\right)$, $N_U = r_U Q_{GWN} = (N_U^{(x)}, N_U^{(y)})$, $AID_U = MID_U \oplus N_U^{(y)}$, $K_U = (r_U + d_U) Q_{GWN}$ and $h_U = H_1(K_U \parallel M_U^{'} \parallel t_1)$. Then, U sends the request message $\{E_U, AID_U, h_U, t_1\}$ via a public channel to GWN.
2. When GWN receives the authentication request message from U at the time $t_1^{'}$, it checks whether the condition $|t_1^{'} - t_1| \leq \Delta t$ holds. If yes, GWN then computes: $N_U^{'} = d_{GWN} E_U = (N_U^{(x)'}, N_U^{(y)'})$. GWN then verifies U by computing the following: $MID_U^{'} = AID_U \oplus N_U^{(y)'}$, $M_U = h(MID_U^{'} \parallel d_{GWN})$, $K_U^{'} = d_{GWN}(Q_U + E_U)$, and $h_U^{'} = H_1\left(K_U^{'} \parallel M_U \parallel t_1\right)$. GWN verifies if the equation $h_U^{'} = h_U$ holds or not. If the verification does not hold, GWN rejects the user's authentication request; else, goes to 3.
3. GWN generates its current timestamp t_2, selects a random number $r_{GWN} \in z_q^*$ and calculates: $E_{GWN} = r_{GWN} P$, $K_{GWN} = (r_{GWN} + d_{GWN}) Q_S$, $M_{GWN} = N_U^{(x)'} \oplus h\left(R_S \parallel K_{GWN} \parallel E_{GWN}\right)$, $h_{GWN} = H_1\left(K_{GWN} \parallel ID_S \parallel t_2\right)$. Then, the gateway node GWN sends the message $\{E_U, E_{GWN}, M_{GWN}, h_{GWN}, t_2, t_1\}$ to S via a public channel.
4. Upon receiving the authentication message from GWN at time $t_2^{'}$, S first checks the validity of the timestamp on the condition $|t_2^{'} - t_2| \leq \Delta t$. If t_2 is invalid, S terminates the session. If it is valid, S then computes: $K_{GWN}^{'} = d_S(E_{GWN} + Q_{GWN})$, $N_U^{(x)''} = M_{GWN} \oplus h\left(R_S \parallel K_{GWN}^{'} \parallel E_{GWN}\right)$, and $h_{GWN}^{'} = H_1\left(K_{GWN}^{'} \parallel ID_S \parallel t_2\right)$. Next, S verifies $h_{GWN}^{'}$. If $h_{GWN}^{'} = h_{GWN}$, the sensor node S accepts GWN and goes to 5; otherwise, it rejects GWN.
5. S generates its current timestamp t_3 and selects a random number $r_S \in z_q^*$, and computes $E_S = r_S P$, $K_S = r_S\left(R_S - h(ID_S \parallel d_S) P\right)$, $h_S = H_1\left(K_S \parallel ID_S \parallel t_3\right)$, $sk_S = r_S(E_U + N_U^{(x)''} P)$ and $Auth_S = H_1(sk_S \parallel t_3)$. S sends the message $\{E_S, t_3, h_S, Auth_S\}$ to GWN via a public channel. Then, S computes the session key $SK = H_2(sk_S \parallel E_S \parallel E_U \parallel t_3 \parallel t_1)$.
6. Upon receiving the replied message from S at time $t_3^{'}$, GWN checks the validity of t_3 on the condition $|t_3^{'} - t_3| \leq \Delta t$. If t_3 is valid, GWN computes $K_S^{'} = h\left(ID_S \parallel d_{GWN}\right) E_S$ and $h_S^{'} = H_1\left(K_S^{'} \parallel ID_S \parallel t_3\right)$. Then, GWN checks whether $h_S^{'} = h_S$. If yes, GWN generates its

current timestamp t_4, computes $Auth_{GWN} = H_1(r_{GWN}Q_U \parallel M_U \parallel t_4)$ and sends the message $\{E_S, E_{GWN}, t_3, t_4, Auth_S, Auth_{GWN}\}$ to U.

7. After receiving the replied message from GWN at time $t_4^{'}$, U checks the validity of $t_4^{'}$ with the condition $|t_4^{'} - t_4| \leq \Delta t$. If it is valid, U computes $Auth_{GWN}^{'} = H_1(d_U E_{GWN} \parallel M_U^{'} \parallel t_4)$ and checks whether $Auth_{GWN}^{'} = Auth_{GWN}$. If yes, U computes $sk_U = (r_U + N_U^{(x)})E_S$, $Auth_S^{'} = H_1(sk_U \parallel t_3)$. Then, U checks whether $Auth_S^{'} = Auth_S$. If yes, U calculates the secret session key $SK = H_2(sk_U \parallel E_S \parallel E_U \parallel t_3 \parallel t_1)$.

The process of authentication and key agreement is visually illustrated in Figure 2.

Figure 2. Authentication and key establishing phase of the proposed scheme.

4. Analysis of Correctness and Security

In this section, the correctness of the proposed scheme is validated using BAN-logic and the security of our scheme is proved under the random oracle model. In addition, some other security features are also discussed in the end.

4.1. Correctness

With the formal validation tool Burrows-Abadi-Needham Logic (BAN-logic) [27], we provide the proof of correctness of the proposed scheme in this section. Let U be the user, S represent the sensor node and GWN denote the gateway node. We demonstrate that a session key can be created successfully after the process of mutual authentication among S and U. Now, the basic notations of BAN-logic are given below:

- $P \mid\equiv X$: P believes X.
- $P \triangleleft X$: P sees X. i.e., P has received messages containing X.
- $P \mid\sim X$: P said X. i.e., P has sent messages containing X.
- $P \mid\Rightarrow X$: P controls X.
- $\#(X)$ or $fresh(X)$: X is a fresh message. X is usually a temporary value.
- (X): The hashed value of X.
- $P \overset{K}{\longleftrightarrow} Q$: K is a shared secret key between P and Q.
- $\langle X \rangle_Y$: X is combined with secret Y.
- (X, Y): X or Y is one part of (X, Y).

Some logic postulates of BAN-logic are described as follows:

- Message-meaning rule(MMR): $\frac{P|\equiv Q \overset{k}{\longleftrightarrow} P, P \triangleleft \{X\}_K}{P|\equiv Q|\sim X}$ or $\frac{P \text{ believes } Q \overset{k}{\longleftrightarrow} P, P \text{ sees } \{X\}_K}{P \text{ believes } Q \text{ said } X}$

  If P believes that K is a shared secret key between P and Q and has received messages containing X, P believes that Q has sent messages containing the message X.
- Nonce-verification rule(NVR): $\frac{P|\equiv \#(X), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$ or $\frac{P \text{ believes } fresh(X), P \text{ believes } Q \text{ said } x}{P \text{ believes } Q \text{ believes } X}$

  If P believes that X is a fresh message and Q has sent messages containing the message X, P believes that Q believes the message X.
- Jurisdiction rule(JR): $\frac{P|\equiv Q \Rightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$ or $\frac{P \text{ believes } Q \text{ controls } X,\ P \text{ believes } Q \text{ believes } X}{P \text{ believes } X}$

  If P believes that Q controls the message X and Q believes the message X, P believes the message X.
- Freshness rule(FR): $\frac{P|\equiv \#(X)}{P|\equiv \#(X,Y)}$ or $\frac{P \text{ believes } fresh(X)}{P \text{ believes } fresh(X,Y)}$

  If P believes that X is a fresh message, P believes (X, Y) is fresh messages.
- Belief rule(BR): $\frac{P|\equiv (X,Y)}{P|\equiv X}$ or $\frac{P \text{ believes } (X,Y)}{P \text{ believes } (X)}$

  If P believes the messages (X, Y), P believes the message X.

Our proposed scheme can realize the establishment of a secret session key SK between U and S, and the following goals can be achieved after the protocol execution.

- *Goal 1*: $U \mid\equiv (U \overset{SK}{\longleftrightarrow} S)$
- *Goal 2*: $S \mid\equiv (U \overset{SK}{\longleftrightarrow} S)$

The exchange of messages during the authentication phase is depicted as follows:

- Message 1: $GWN \rightarrow S$: $\left\langle r_{GWN} P, t_2, \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S \right) \right\rangle_{R_S}$
- Message 2: $GWN \rightarrow S$: $\left\langle r_U P, r_{GWN} P, t_2, t_1, \left(U \overset{N_U^{(x)\prime\prime} = N_U^{(x)}}{\longleftrightarrow} S \right) \right\rangle_{K_{GWN}}$
- Message 3: $GWN \rightarrow U$: $\left\langle r_S P, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right) \right\rangle_{M_U}$

- Message 4: $GWN \rightarrow U$: $\left\langle r_S P, t_3, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN\right), \left(U \overset{sk_U = sk_S}{\longleftrightarrow} S\right)\right\rangle_{r_{GWN} Q_U}$

To proceed the derivation, the initial state assumptions are set as A1–A9:

- A1: $S \mid\equiv \#(t_2)$
- A2: $S \mid\equiv \#(t_1)$
- A3: $U \mid\equiv \#(t_4)$
- A4: $S \mid\equiv GWN \overset{R_S}{\leftrightarrow} S$
- A5: $U \mid\equiv U \overset{M_U}{\leftrightarrow} GWN$
- A6: $S \mid\equiv GWN \mid\Rightarrow \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S\right)$
- A7: $S \mid\equiv GWN \mid\Rightarrow \left(U \overset{N_U^{(x)''} = N_U^{(x)}}{\longleftrightarrow} S\right)$
- A8: $U \mid\equiv GWN \mid\Rightarrow \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN\right)$.
- A9: $U \mid\equiv GWN \mid\Rightarrow \left(U \overset{sk_S}{\leftrightarrow} S\right)$.

U and S intend to share a session key SK to achieve confidential communication. As stated above, the mutual authentication between U and S shows that *Goal* 1 and *Goal* 2 can be achieved in the end. The result is proved as follows:

- From Message 1, we have:

$$S \triangleleft \left\langle r_{GWN} P, t_2, \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S\right)\right\rangle_{R_S} \tag{1}$$

S has received the message $\{r_{GWN} P, t_2, (GWN \overset{K_{GWN}}{\longleftrightarrow} S)\}$ encrypted by R_S.

- According to the message-meaning rule, if the Formula (1) and the state assumption A4 hold at the same time, we can infer that:

$$S \mid\equiv GWN \mid\sim \left\langle r_{GWN} P, t_2, \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S\right)\right\rangle \tag{2}$$

S believes that GWN has sent the messages $\{r_{GWN} P, t_2, (GWN \overset{K_{GWN}}{\longleftrightarrow} S)\}$.

- According to the freshness rule, if the state assumption A1 holds, we then obtain:

$$S \mid\equiv \# \left\langle r_{GWN} P, t_2, \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S\right)\right\rangle \tag{3}$$

S believes the message $\{r_{GWN} P, t_2, (GWN \overset{K_{GWN}}{\longleftrightarrow} S)\}$ are fresh.

- According to the nonce-verification rule, if the Formula (2) and (3) hold at the same time, we can deduce:

$$S \mid\equiv GWN \mid\equiv \left\langle r_{GWN} P, t_2, \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S\right)\right\rangle \tag{4}$$

S believes that GWN believes the message $\{r_{GWN} P, t_2, (GWN \overset{K_{GWN}}{\longleftrightarrow} S)\}$ are real.

- According to the belief rule, if the Formula (4) holds, we can get:

$$S \mid\equiv GWN \mid\equiv \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S\right) \tag{5}$$

S believes that GWN believes K_{GWN} is a shared secret key between GWN and S.

- According to the jurisdiction rule, if the Formula (5) and the state assumption A6 hold at the same time, we can obtain:

$$S \mid\equiv \left(GWN \overset{K_{GWN}}{\longleftrightarrow} S\right) \tag{6}$$

S believes that K_{GWN} is a shared secret key between GWN and S.

- From Message 2, we can have:

$$S \triangleleft \left\langle r_U P, r_{GWN} P, t_2, t_1, \left(U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S\right)\right\rangle_{K_{GWN}} \tag{7}$$

S has received the message $\{r_U P, r_{GWN} P, t_2, t_1, (U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S)\}$ encrypted by K_{GWN}.

- According to the message-meaning rule, if the Formula (6) and (7) hold at the same time, we can infer that:

$$S \mid\equiv GWN \mid\sim \left\langle r_U P, r_{GWN} P, t_2, t_1, \left(U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S\right)\right\rangle \tag{8}$$

S believes that GWN has sent the message $\{r_U P, r_{GWN} P, t_2, t_1, (U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S)\}$.

- According to the freshness rule, if the state assumption A2 holds, we can deduce:

$$S \mid\equiv \#\left\langle r_U P, r_{GWN} P, t_2, t_1, \left(U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S\right)\right\rangle \tag{9}$$

S believes the messages $\{r_U P, r_{GWN} P, t_2, t_1, (U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S)\}$ are fresh.

- According to the nonce-verification rule, if the Formula (8) and (9) hold at the same time, we can get:

$$S \mid\equiv GWN \mid\equiv \left\langle r_U P, r_{GWN} P, t_2, t_1, \left(U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S\right)\right\rangle \tag{10}$$

S believes that GWN believes the message $\{r_U P, r_{GWN} P, t_2, t_1, (U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S)\}$ are real.

- According to the belief rule, if the Formula (10) holds, we can obtain:

$$S \mid\equiv GWN \mid\equiv \left(U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S\right) \tag{11}$$

S believes that GWN believes $N_U^{(x)}$ is a shared secret key between U and S.

- According to the jurisdiction rule, if the Formula (11) and the state assumption A7 hold at the same time, we can have:

$$S \mid\equiv \left(U \overset{N_U^{(x)\prime\prime}=N_U^{(x)}}{\longleftrightarrow} S\right) \tag{12}$$

S believes that $N_U^{(x)}$ is a shared secret key between U and S.

- According to the belief rule, if the Formula (12) holds, the Formula (13) holds, we can infer:

$$S \mid\equiv \left(U \overset{SK}{\leftrightarrow} S\right) \qquad \textit{Goal } 2 \tag{13}$$

S believes that SK is a shared secret key between U and S, which can be seen that *Goal* 2 has been achieved.

- From Message 3, we can get:

$$U \triangleleft \left\langle r_S P, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right) \right\rangle_{M_U} \tag{14}$$

U has received the message $\{r_S P, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN)\}$ encrypted by M_U.

- According to the message-meaning rule, if the Formula (14) and the state assumption A5 hold at the same time, we can deduce:

$$U \mid\equiv GWN \mid\sim \left\langle r_S P, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right) \right\rangle \tag{15}$$

U believes that GWN has sent the message $\{r_S P, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN)\}$.

- According to the freshness rule, if the state assumption A3 holds, we can have:

$$U \mid\equiv \# \left\langle r_S P, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right) \right\rangle \tag{16}$$

U believes the message $\{r_S P, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN)\}$ are fresh.

- According to the nonce-verification rule, if the Formula (15) and (16) hold at the same time, we can obtain:

$$U \mid\equiv GWN \mid\equiv \left\langle r_S P, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right) \right\rangle \tag{17}$$

U believes that GWN believes the message $\{r_S P, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN)\}$ are real.

- According to the belief rule, if the Formula (17) holds, we can infer:

$$U \mid\equiv GWN \mid\equiv \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right) \tag{18}$$

U believes that GWN believes $r_{GWN} Q_U$ is a shared secret key between U and GWN.

- According to the jurisdiction rule, if the Formula (18) and the state assumption A8 hold at the same time, we can deduce:

$$U \mid\equiv \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right) \tag{19}$$

U believes that $r_{GWN} Q_U$ is a shared secret key between U and GWN.

- From Message 4, we can get:

$$U \triangleleft \left\langle r_S P, t_3, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right), \left(U \overset{sk_U = sk_S}{\longleftrightarrow} S \right) \right\rangle_{r_{GWN} Q_U} \tag{20}$$

which means that U has received the message $\{r_S P, t_3, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN), (U \overset{sk_U = sk_S}{\longleftrightarrow} S)\}$ encrypted by $r_{GWN} Q_U$.

- According to the message-meaning rule, if the Formula (19) and (20) and the state assumption A5 hold at the same time, we can deduce:

$$U \mid\equiv GWN \mid\sim \left\langle r_S P, t_3, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right), \left(U \overset{sk_U = sk_S}{\longleftrightarrow} S \right) \right\rangle \tag{21}$$

which means that U believes that GWN has sent the message $\{r_S P, t_3, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN), (U \overset{sk_U = sk_S}{\longleftrightarrow} S)\}$.

- According to the freshness rule, if the state assumption A3 holds, we can have:

$$U \mid\equiv \# \left\langle r_S P, t_3, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right), \left(U \overset{sk_U = sk_S}{\longleftrightarrow} S \right) \right\rangle \tag{22}$$

which means that U believes the message $\{r_S P, t_3, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN), (U \overset{sk_U = sk_S}{\longleftrightarrow} S)\}$ are fresh.

- According to the nonce-verification rule, if the Formula (21) and (22) hold at the same time, we can obtain:

$$U \mid\equiv GWN \mid\equiv \left\langle r_S P, t_3, t_4, \left(U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN \right), \left(U \overset{sk_U = sk_S}{\longleftrightarrow} S \right) \right\rangle \tag{23}$$

U believes that GWN believes the message $\{r_S P, t_3, t_4, (U \overset{r_{GWN} Q_U}{\longleftrightarrow} GWN), (U \overset{sk_U = sk_S}{\longleftrightarrow} S)\}$ are real.

- According to the belief rule, if the Formula (23) holds, we can infer:

$$U \mid\equiv GWN \mid\equiv \left(U \overset{sk_U = sk_S}{\longleftrightarrow} S \right) \tag{24}$$

U believes that GWN believes sk_U is a shared secret key between U and S.

- According to the jurisdiction rule, if the Formula (24) and the state assumption A9 hold at the same time, we can deduce:

$$U \mid\equiv \left(U \overset{sk_U = sk_S}{\longleftrightarrow} S \right) \tag{25}$$

U believes that sk_U is a shared secret key between U and S.

- According to the belief rule, if the Formula (25) holds, we can have:

$$U \mid\equiv \left(U \overset{SK}{\leftrightarrow} S \right) \qquad Goal\ 1 \tag{26}$$

U believes that SK is a shared secret key between U and S.

At this point, it can be seen that *Goal* 1 and *Goal* 2 have been achieved, which means that the proposed scheme is correct and feasible.

4.2. Security

We first demonstrate that our proposed scheme possesses semantic security under the random oracle model.

Theorem 1. *Let $\mathcal{A}$ denote an adversary within a polynomial time t against the proposed protocol under the random oracle model, then we have:*

$$Adv_{\mathcal{A}}^{Ind} \leqslant \frac{q_{H_1}^2}{|H_1|} + \frac{q_{H_2}^2}{|H_2|} + \frac{(q_{exe} + q_{send})^2}{p} + 2Adv_{\mathcal{A}}^{ECCDH}(t)$$

where $Adv_{\mathcal{A}}^{ECCDH}(t)$ is the advantage of $\mathcal{A}$ breaks the ECCDH problem; q_{H_1}, q_{H_2}, q_{exe}, q_{send} represent the number of H_1, H_2, Execute and Send queries respectively; $|H_1|$, $|H_2|$ denote the range space of H_1 and H_2 function respectively.

Proof. Let $Succ_i$ represent the event that $\mathcal{A}$ wins in the game G_i, i.e., $\mathcal{A}$ guesses bit b, where $i = [0, 3]$.

Game G_0: In G_0, a real attack against our proposed scheme from $\mathcal{A}$ is simulated. Firstly, the value of b is selected randomly. According to the above definitions, we obtain:

$$Adv_{\mathcal{A}}^{Ind} = 2 \cdot Pr[Succ_0] - 1 \tag{27}$$

Game G_1: To increase the probability that $\mathcal{A}$ wins game, the query $Execute$ is used to model the eavesdropping attacks. Since its goal is to get some information about SK, $\mathcal{A}$ has to compute sk_U or sk_S according to the definition of the proposed scheme; however, $sk_U = r_S(r_U + N_U^{(x)})P$, where r_U, r_S are unknown. Without corrupting the gateway node GWN to get d_{GWN}, the probability of success would not be increased just by eavesdropping the transmitted messages, which implies that

$$Pr[Succ_1] = Pr[Succ_0] \tag{28}$$

Game G_2: The game is transferred from G_1 is used to simulate active attacks by adding H_1, H_2 and $Send$ oracles in which $\mathcal{A}$ tries to forge messages. By arbitrarily issuing queries to H_1, H_2, $\mathcal{A}$ attempts to capture collisions. The probability of collisions is at most $(\frac{q_{H_1}^2}{|H_1|} + \frac{q_{H_2}^2}{|H_2|})$ according to the birthday paradox. The probability of collisions in the transcripts is at most $\frac{(q_{send}+q_{exe})^2}{p}$. Therefore, we get:

$$|Pr[Succ_1] - Pr[Succ_2]| \leqslant \frac{q_{H_1}^2}{2 \mid H_1 \mid} + \frac{q_{H_2}^2}{2 \mid H_2 \mid} + \frac{(q_{exe} + q_{send})^2}{2p} \tag{29}$$

Game G_3: G_3 models the attack that the the gateway node GWN has been corrupted. By issuing $Corrupt(\prod_P^k)$ oracles, $\mathcal{A}$ can get the long-term key of GWN. According to the definition, the common secret value sk_S or sk_U are the core of the session key SK. Considering the following fact,

$$\begin{aligned} sk_U &= sk_S \\ &= r_S(r_U + N_U^{(x)})P = r_U r_S P + r_S N_U^{(x)} P \\ &= r_U r_S P + (d_{GWN} E_U)^{(x)} E_S \end{aligned}$$

Thus, $\mathcal{A}$ can use the long-term key d_{GWN} to compute partial value from transcripts. The probability of success of $\mathcal{A}$ between G_3 and G_2 would not be greater than the advantage of solving ECCDH problem instance. Let $Adv_{\mathcal{A}}^{ECCDH}$ be the advantage that the adversary $\mathcal{A}$ solves ECCDH problem instance within t in this game. Hence, we get

$$|Pr[Succ_2] - Pr[Succ_3]| \leqslant Adv_{\mathcal{A}}^{ECCDH}(t) \tag{30}$$

To win the game G_3, $\mathcal{A}$ has no choice but guess the bit b, which leads to the following result

$$Pr[Succ_3] = \frac{1}{2} \tag{31}$$

Thus, from (28)–(31), we get

$$\begin{aligned} |Pr[Succ_0] - \frac{1}{2}| &= |Pr[Succ_0] - Pr[Succ_3]| \\ &\leqslant |Pr[Succ_0] - Pr[Succ_1]| + |Pr[Succ_1] - Pr[Succ_2]| \\ &+ |Pr[Succ_2] - Pr[Succ_3]| \\ &\leqslant \frac{q_{H_1}^2}{2 \mid H_1 \mid} + \frac{q_{H_2}^2}{2 \mid H_2 \mid} + \frac{(q_{send} + q_{exe})^2}{2p} + Adv_{\mathcal{A}}^{ECCDH}(t) \end{aligned}$$

From (27), we have $Pr[Succ_0] = Adv_{\mathcal{A}}^{Ind}/2 + 1/2$. Hence,

$$Adv_{\mathcal{A}}^{Ind} \leqslant \frac{q_{H_1}^2}{\mid H_1 \mid} + \frac{q_{H_2}^2}{\mid H_2 \mid} + \frac{(q_{send} + q_{exe})^2}{p} + 2Adv_{\mathcal{A}}^{ECCDH}(t)$$

□

4.3. Deniable Authentication

In our proposed scheme, the polynomial time deniability means that the gateway node as a receiver can simulate the messages sent by the user which are indistinguishable for any third party. The concrete simulation process of GWN is as follows:

1. GWN selects a random number $\overline{r_U} \in z_q^*$, computes $\overline{E_U} = \overline{r_U}P$ and $\overline{N_U} = \overline{r_U}Q_{GWN} = (\overline{N_U^{(x)}}, \overline{N_U^{(y)}})$.
2. GWN chooses a user pseudo-identity $\overline{h(ID_U)}$ and a public key, computes $\overline{AID_U} = \overline{h(ID_U)} \oplus \overline{N_U^{(y)}}$, $\overline{K_U} = d_{GWN}(\overline{E_U} + \overline{Q_U})$ and $\overline{h_U} = H_1(\overline{K_U} \parallel h(\overline{h(ID_U)} \parallel d_{GWN}) \parallel t_1)$.

GWN sends $\overline{E_U}, \overline{AID_U}, \overline{h_U}, t_1$ to the third party. After receiving the message, the third party cannot get any information related to the user by $\overline{AID_U}$. In addition, $\overline{h_U}$ can be calculated by the user or the gateway. Hence, the third party is unable to determine the true source of the message. Therefore, our proposed scheme achieves deniable authentication.

4.4. Anonymity

Since the transmitted authentication messages are carried via a public channel, an outside adversary can easily eavesdrop the communication. However, our proposed scheme can preserve the anonymity of the user. Suppose that an adversary $\mathcal{A}$ intercepts $\{E_U, AID_U, h_U, t_1\}$ during the authentication phase and attempts to reveal some information about the user's identity. $\mathcal{A}$ obtains $N_U = r_U Q_{GWN} = (N_U^{(x)}, N_U^{(y)})$, $AID_U = MID_U \oplus N_U^{(y)}$, which $MID_U = h(ID_U)$. Due to the utilization of random number r_U and one-way hash function, $\mathcal{A}$ cannot calculate N_U and get ID_U. Since the use of the timestamps and random numbers, those intercepted messages by $\mathcal{A}$ are unique and dynamic for each authentication between U, S and GWN. Therefore, the proposed scheme ensures user anonymity.

4.5. Mutual Authentication

With the received request message $\{E_U, AID_U, h_U, t_1\}$ U sent, GWN can compute $N_U^{'} = d_{GWN}E_U = (N_U^{(x)'}, N_U^{(y)'})$ to get the values M_U and K_U and checks the validity of U via the equivalence $h_U = h_U^{'}$. After receiving the message $\{E_U, E_{GWN}, M_{GWN}, h_{GWN}, t_2, t_1\}$ from GWN, the sensor node S could obtain the values K_{GWN} and $N_U^{(x)'}$ and then computes $h_{GWN}^{'} = H_1\left(K_{GWN}^{'} \parallel ID_S \parallel t_2\right)$ to verify the validity of GWN via the equivalence $h_{GWN} = h_{GWN}^{'}$. Once receiving the message $\{E_S, t_3, h_S, Auth_S\}$ from S, GWN computes $K_S^{'}$ and $h_S^{'} = H_1\left(K_S^{'} \parallel ID_S \parallel t_3\right)$ to check the validity of S via the equivalence $h_S^{'} = h_S$. Then, GWN sends message $\{E_S, t_3, t_4, Auth_S, Auth_{GWN}\}$ to U and U computes $sk_U = (r_U + N_U^{(x)})E_S$, $Auth_{GWN}^{'} = H_1(d_U E_{GWN} \parallel M_U^{'} \parallel t_4)$ and $Auth_S^{'} = H_1(sk_U \parallel t_3)$ and checks the validity of GWN and S by the equivalence $Auth_{GWN}^{'} = Auth_{GWN}$ and $Auth_S^{'} = Auth_S$. If the above verification processes are successfully completed, our protocol provides mutual authentication.

4.6. Unlinkability

In our proposed scheme, the real identities or related information of all participants are not sent in plaintext over the insecure network because each transmitted message contains timestamps, random values and one-way hash function values. An outside adversary $\mathcal{A}$ cannot determine whether two or

more authentication messages come from the same participant. Therefore, the transmitted messages cannot be linked by the adversary.

4.7. Traceability

In our proposed scheme, given a disputed message $\{E_U, AID_U, h_U, t_1\}$, only the trusted gateway node(GWN) can reveal the identity of the user. With above message, GWN computes $N_U^{'} = d_{GWN}E_U = (N_U^{(x)'}, N_U^{(y)'})$ and $MID_U^{'} = AID_U \oplus N_U^{(y)'}$ to get the user's identity MID_U. In addition, the tracing process does not need real user to participate because the message $\{E_U, AID_U, h_U, t_1\}$ sent by the user contains sufficient information to derive the user identity. Therefore, our proposed scheme achieves traceability.

4.8. Resistance to Impersonation Attack

Assume an adversary $\mathcal{A}$ intercepts message $\{E_U, AID_U, h_U, t_1\}$ to impersonate a user, where $E_U = r_U P$, $AID_U = MID_U \oplus N_U^{(y)}$, $K_U = (r_U + d_U)Q_{GWN}$, $h_U = H_1(K_U \parallel M_U^{'} \parallel t_1)$. By following the authentication process, the adversary produces a timestamp $t_1^{'}$ and a value $r_U^{'} \in Z_q^*$ randomly to get $E_U^{'}$, $AID_U^{'}$ and $K_U^{'}$. However, $\mathcal{A}$ is unable to successfully compute $h_U^{'}$ because he or she does not has the user's real identity ID_U and private key d_U. Hence, our scheme can resist such attacks according to the above analysis.

4.9. Resistance to Replay Attack

Suppose an adversary $\mathcal{A}$ intercepts all transmitted messages between participants and then attempts to replay some or all of them. In our scheme, however, timestamps and random numbers are integrated into the generation of the messages for U, GWN, S, thus the freshness of messages is well preserved. Therefore, the proposed protocol can resist replay attacks.

4.10. Forward Security

Assume an adversary $\mathcal{A}$ could get the private keys of all participants, i.e., d_U, d_{GWN}, d_S. Even if the adversary $\mathcal{A}$ had obtained the current session key $SK = H_2(sk_U \parallel E_S \parallel E_U \parallel t_3 \parallel t_1)$, he or she cannot derive the previous session key. However, due to $sk_U = sk_S = (r_U + N_U^{(x)})E_S = r_U r_S P + (d_{GWN}E_U)^{(x)}E_S$, where r_U and r_S are chosen randomly by U and S respectively. $\mathcal{A}$ can never obtain the previous session key since the difficulty of the ECCDH problem. So, our proposed scheme achieves forward security.

5. Performance Comparison

In this section, we evaluate the performance of our scheme regarding the computational cost in the authentication phase. Moreover, we present the comparison between the proposed scheme and some existing similar schemes [15,16,21,23–25]. For convenience, we use the symbols in Table 2 to denote the computational cost regarding hash operation, ECC-based operation and bilinear paring operation and the approximate running time required of various operations is presented in Table 2.

Table 2. Approximate running time of operations.

Operation	Description	Computation Time (ms)
T_h	a hash function	3×10^{-3}
T_{bp}	a bilinear pairing	2.14×10^{-1}
T_{pmul}	a ECC-based point multiplication	1.6×10^{-2}
T_{padd}	a ECC-based point addition	6.07×10^{-1}

Please note that we only consider the operations listed in Table 2 since the running time of addition operation and XoR operation is ignorable. To fairly compare the computational time cost of these similar protocols. The experiments use OpenSSL and JPBC cryptographic libraries, and then are programmed with Visual C language.

Table 3 and Figure 3 presents the comparisons among the other protocols [15,16,21,23–25] and ours. Table 4 presents the comparison of security properties between ours and the above protocols. According to the experimental results, it is observed that our scheme costs 3.791 ms, which is better than [15,16,24,25]. We sort the time consumption on the operations as below: $T_h < T_{padd} < T_{pmul} < T_{bp}$. The hash function spends the least time, while the bilinear pairing operation takes the more time. To fully demonstrate the proposed scheme's advantage, we define $\left(T_{[others]} - T_{[ours]}\right) / T_{[others]}$, where $T_{[others]}$ denotes computational cost of the other schemes and $T_{[ours]}$ represents computational cost of ours, as the improved ratio of ours compared with others [15,16,24,25]. Hence, the improved ratios of the proposed scheme compared with [15,16,24,25] are $(7.041 - 3.791)/7.041 \approx 43.44\%$, $(8.705 - 3.791)/8.705 \approx 58.81\%$, $(5.927 - 3.791)/5.927 \approx 32.51\%$ and $(5.215 - 3.791)/5.215 \approx 23.37\%$ respectively.

Table 3. Comparison of computational cost.

Protocol	Computational Cost	Running Time (ms)
Ours	$18T_h + 17T_{pmul} + 4T_{padd}$	≈3.791
[15]	$9T_h + 15T_{pmul} + 3T_{padd} + 9T_{bp}$	≈8.705
[16]	$9T_h + 8T_{pmul} + 2T_{padd} + 6T_{bp}$	≈5.927
[21]	$10T_h + 5T_{pmul} + 2T_{bp}$	≈2.779
[23]	$14T_h + 8T_{pmul} + 3T_{padd}$	≈2.079
[24]	$15T_h + 7T_{pmul} + 9T_{bp}$	≈7.041
[25]	$5T_h + 7T_{pmul} + 6T_{bp}$	≈5.215

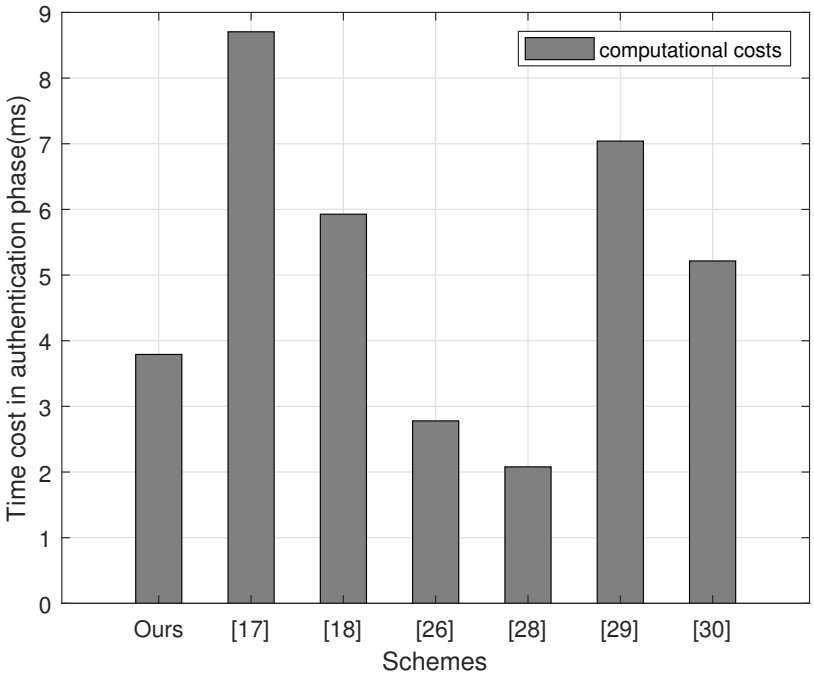

Figure 3. Computational costs of different authentication schemes.

Table 4. The comparison of security features.

Scheme	Ours	[15]	[16]	[21]	[23]	[24]	[25]
Anonymity	Yes	No	No	No	No	No	No
Mutual authentication	Yes	No	Yes	Yes	Yes	Yes	Yes
Session key security	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Perfect forward secrecy	Yes	Yes	Yes	Yes	No	Yes	Yes
Resistance to replay attack	Yes	No	No	Yes	No	No	No
Resistance to impersonation attack	Yes	Yes	Yes	No	Yes	No	No

Compared to Porambage's scheme [23] and Wang's scheme [21], our scheme requires more communication overheads from Table 3 and Figure 3. However, from Table 4 our scheme possesses more desirable security compared with the existing schemes. However, Porambage's scheme cannot protect against the replay attack and provide the user's anonymity. In addition, the user's anonymity can be violated. Wang's scheme [21] is prone to client impersonation attacks. Specifically, an adversary is able to masquerade as a legitimate client to be authenticated by application provider. Therefore, our proposed scheme provides a better secure communication and higher efficiency than the compared existing schemes in IoT.

6. Conclusions

With the evolution of the Internet of Things, its security is currently drawing wide attention. The privacy protection in communication is a major concern for people. In this article, we proposed an anonymous authentication and key agreement protocol with deniability property using elliptic curve. In our proposed scheme, other participants except the trusted gateway node can obtain nothing regarding the real identity of a user. We have demonstrated that our proposed scheme posses more appropriate security features than similar schemes, which are shown in the BAN logic-based proof and random oracle model-based proof. In addition, we have provided informal analysis to further confirm that our scheme can resist various attacks. By experimental evaluation, we demonstrate that the proposed scheme is efficient according to the comparison on computational costs against other similar protocols. In view of the advantages in security and performance, our proposed scheme is more suitable for IoT systems.

From the analysis, the computational overhead of our proposed scheme become relatively low. Therefore, we aim to achieve a better trade-off among security and efficiency in designing authentication protocols for IoT applications in our future work, so as to meet the requirements of low-cost computation and communication of resource-constrained sensors.

Author Contributions: Y.Z. and T.L. conceived and designed the experiments and wrote the paper; F.T. and F.W. designed the experiments; M.T. performed the experiments.

Acknowledgments: This work was supported in part by the Venture and Innovation Support Program for Chongqing Overseas Returnees under Grant CX2018122, and in part by the National Natural Science Foundation of China under Grant 61702067.

Conflicts of Interest: The authors declare that there is no conflict of interest regarding the publication of this paper.

References

1. Sundmaeker, H.; Guillemin, P.; Friess, P. Vision and challenges for realising the Internet of Things. *Clust. Eur. Res. Proj. Internet Things Eur. Commis.* **2010**, *3*, 34–36, doi:10.2759/26127.
2. Lo, N.W.; Tsai, J.L. An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1319–1328, doi:10.1109/TITS.2015.2502322.
3. He, D.; Kumar, N.; Chen, J. Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks. *Multimed. Syst.* **2015**, *21*, 49–60, doi:10.1007/s00530-013-0346-9.

4. Li, X.; Niu, J.; Kumari, S.; Liao, J.; Liang, W.; Khan, M.K. A new authentication protocol for healthcare applications using wireless medical sensor networks with user anonymity. *Secur. Commun. Netw.* **2016**, *9*, 2643–2655, doi:10.1002/sec.1214.
5. Wu, F.; Xu, L.; Kumari, S. An improved and anonymous two-factor authentication protocol for health-care applications with wireless medical sensor networks. *Multimed. Syst.* **2017**, *23*, 195–205, doi:10.1007/s00530-015-0476-3.
6. He, D.; Kumar, N.; Chilamkurti, N. A secure temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. *Int. Symp. Wirel. Pervasive Comput.* **2013**, *36*, 316–323, doi:10.1109/ISWPC.2013.6707446.
7. Castiglione, A.; Santis, A.D.; Castiglione, A.; Palmieri, F. An Efficient and Transparent One-Time Authentication Protocol with Non-interactive Key Scheduling and Update. In Proceedings of the 2014 IEEE 28th International Conference on Advanced Information Networking and Applications, Gwangju, Korea, 25–27 March 2014; pp. 351–358; doi:10.1109/AINA.2014.45.
8. Gupta, A.; Tripathi, M. A lightweight Mutually Authenticated Key-Agreement scheme for Wireless Body Area Networks in Internet of Things Environment. *Radio Freq. Identif. IoT Secur.* **2018**, 804–806, doi:10.1145/3241539.3267775.
9. Li, X.; Niu, J.; Kumari, S.; Wu, F.; Sangaiah, A.K.; Choo, K.-K.R. A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments. *J. Netw. Comput. Appl.* **2018**, *103*, 194–204, doi:10.1016/j.jnca.2017.07.001.
10. Wong, K.H.M.; Zheng, Y.; Cao, J.; Wang, S. A dynamic user authentication scheme for wireless sensor networks. In Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06), Taichung, Taiwan, 5–7 June 2006; Volume 1, p. 8, doi:10.1109/SUTC.2006.1636182.
11. Das, M.L. Two-factor user authentication in wireless sensor networks. *IEEE Trans. Wirel. Commun.* **2009**, *17*, 1086–1090, doi:10.1109/TWC.2008.080128.
12. Khan, M.K.; Alghathbar, K. Cryptanalysis and Security Improvements of 'Two-Factor User Authentication in Wireless Sensor Networks'. *Sensors* **2010**, *10*, 2450–2459, doi:10.3390/s100302450.
13. Chen, T.-H.; Shih, W.-K. A Robust Mutual Authentication Protocol for Wireless Sensor Networks. *ETRI J.* **2010**, *32*, 704–712, doi:10.4218/etrij.10.1510.0134.
14. He, D.; Gao, Y.; Chan, S. An Enhanced Two-factor User Authentication Scheme in Wireless Sensor Networks. *Ad Hoc Wirel. Netw.* **2010**, *10*, 361–371.
15. Holbl, M.; Welzer, T.; Brumen, B. Two proposed identity-based three-party authenticated key agreement protocols from pairings. *Comput. Secur.* **2010**, *29*, 244–252, doi:10.1016/j.cose.2009.08.006.
16. Holbl, M.; Welzer, T.; Brumen, B. An improved two-party identity-based authenticated key agreement protocol using pairings. *J. Comput. Syst. Sci.* **2012**, *78*, 233–271, doi:10.1016/j.jcss.2011.01.002.
17. Jung, J.; Kim, J.; Choi, Y. An Anonymous User Authentication and Key Agreement Scheme Based on a Symmetric Cryptosystem in Wireless Sensor Networks. *Sensors* **2016**, *16*, 1299, doi:10.3390/s16081299.
18. Xue, K.; Ma, C.; Hong, P. A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. *J. Netw. Comput. Appl.* **2013**, *36*, 316–312, doi:10.1016/j.jnca.2012.05.010.
19. Jiang, Q.; Ma, J.; Lu, X. An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks. *Peer-to-Peer Netw. Appl.* **2015**, *8*, 1070–1081, doi:10.1007/s12083-014-0285-z.
20. Das, A.K. A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks. *Peer-to-Peer Netw. Appl.* **2016**, *9*, 223–244, doi:10.1007/s12083-014-0324-9.
21. Wang, C.; Zhang, Y. New Authentication Scheme for Wireless Body Area Networks Using the Bilinear Pairing. *J. Med. Syst.* **2015**, *39*, 136, doi:10.1007/s10916-015-0331-2.
22. Tong, L.; Yuhui, Z.; Ti, Z. Efficient Anonymous Authenticated Key Agreement Scheme for Wireless Body Area Networks. *Secur. Commun. Netw.* **2017**, *2017*, doi:10.1155/2017/4167549.
23. Porambage, P.; Braeken, A.; Schmitt, C. Group Key Establishment for Enabling Secure Multicast Communication in Wireless Sensor Networks Deployed for IoT Applications. *IEEE Access* **2015**, *3*, 1503–1511, doi:10.1109/ACCESS.2015.2474705.
24. Xiong, H.; Qin, Z. Revocable and Scalable Certificateless Remote Authentication Protocol With Anonymity for Wireless Body Area Networks. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 1442–1455, doi:10.1109/TIFS.2015.2414399.

25. Liu, J.; Zhang, Z.; Chen, X.; Kwak, K.S. Certificateless Remote Anonymous Authentication Schemes for WirelessBody Area Networks. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 332–342, doi:10.1109/TPDS.2013.145.
26. Abdalla, M.; Fouque, P.-A.; Pointcheval, D. *Password-Based Authenticated Key Exchange in the Three-Party Setting*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 65–84.
27. Burrows, M.; Abadi, M.; Needham, R.M. A logic of authentication. *R. Soc.* **1989**, *426*, 233–271.