A Secure, Lightweight, and Anonymous User Authentication Protocol for IoT Environments

Seunghwan Son 1, Yohan Park 2,* and Youngho Park 1,3, *

1 School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, Korea; sonshawn@knu.ac.kr
2 School of Computer Engineering, Keimyung University, Daegu 42601, Korea
3 School of Electronics Engineering, Kyungpook National University, Daegu 41566, Korea
* Correspondence: yhpark@kmu.ac.kr (Y.P.); parkyh@knu.ac.kr (Y.P.)

Abstract: The Internet of Things (IoT) is being applied to various environments such as telecare systems, smart homes, and intelligent transportation systems. The information generated from IoT devices is stored at remote servers, and external users authenticate to the server for requesting access to the stored data. In IoT environments, the authentication process is required to be conducted efficiently, and should be secure against various attacks and ensure user anonymity and untraceability to ensure sustainability of the network. However, many existing protocols proposed in IoT environments do not meet these requirements. Recently, Rajaram et al. proposed a paring-based user authentication scheme. We found that the Rajaram et al. scheme is vulnerable to various attacks such as offline password guessing, impersonation, privileged insider, and known session-specific temporary information attacks. Additionally, as their scheme uses bilinear pairing, it requires high computation and communication costs. In this study, we propose a novel authentication scheme that resolves these security problems. The proposed scheme uses only hash and exclusive-or operations to be applicable in IoT environments. We analyze the proposed protocol using informal analysis and formal analysis methods such as the BAN logic, real-or-random (ROR) model, and the AVISPA simulation, and we show that the proposed protocol has better security and performance compared with existing authentication protocols. Consequently, the proposed protocol is sustainable and suitable for real IoT environments.

Keywords: mutual authentication; key agreement; lightweight; anonymity; IoT environment; BAN logic; ROR model; AVISPA simulation

1. Introduction

The Internet of Things (IoT) has become an essential technology in business and industry that is being applied to various environments [1–8] including telecare systems, smart grids, intelligent transportation systems, and global roaming systems to make human lives more prosperous. For example, in IoT-based telecare environments (see Figure 1), medical devices and sensors check the patient’s pulse, blood pressure, and body temperature in real time, and transmit the information to a remote server. Thereafter, users such as doctors and researchers authenticate the server using mobile devices (i.e., smart phones) and access the information for diagnosis or research. IoT-based telecare systems can provide convenience to patients and contribute to the progress of healthcare. In addition, the IoT can be applied to other environments for increasing business productivity and industrial efficiency.

However, despite these advantages, several challenges must be addressed. Communications in IoT environments are performed on wireless channels, which are prone to attacks by adversaries. They can obtain transmitted messages and can attempt various attacks including replay, man-in-the-middle, and impersonation attacks [9–11] and also attempt to trace a user to obtain sensitive information. Additionally, to ensure real-time
communication of mobile devices with limited computing power [12], the efficiency of the authentication protocol should be considered. Therefore, a secure and efficient authentication protocol is necessary for sustainable communications in IoT environments. However, recently proposed authentication schemes in IoT environments have several security vulnerabilities, and require high amounts of computation using elliptic curve cryptosystem (ECC) [13] scalar multiplication and bilinear paring operations [14]. These shortcomings can cause problems with the sustainability of the network.

Rajaram et al. [15] proposed a user authentication scheme using bilinear pairing in 2019. They claimed that their scheme is secure against possible attacks and provides various security features. However, we analyze that their scheme has several security vulnerabilities to be applied in wireless networks. Their scheme cannot guarantee user anonymity and requires a high amount of computational cost because it uses bilinear pairing. Therefore, we propose an improved authentication protocol that can resolve these issues.

Figure 1. IoT-based telecare environment.

1.1. Motivation

In IoT environments, each message is transmitted through public channels. The messages can contain personal sensitive information, and if the information is leaked to an adversary, it can cause serious privacy threats. Additionally, IoT devices have limited computing resources, a high amount of computation cost can cause delays. Therefore, a secure and efficient authentication protocol is essential for sustainable IoT environments. In 2019, Rajaram et al. proposed a pairing-based user authentication scheme. However, Rajaram et al. scheme does not have resistance to off-line guessing, impersonation, privileged insider, and known session-specific temporary information attacks. In addition to Rajaram et al. scheme, we note the limitations and weaknesses of existing researches (see Section 2). Several schemes used elliptic curve multiplication and bilinear pairing which require a high amount of computation, making it unsuitable for IoT environments. Additionally, most schemes do not have resistance to various attacks including impersonation, off-line guessing, and privileged insider attacks, and cannot guarantee security features such as mutual authentication, user anonymity, and user untraceability. Because of these weaknesses, existing schemes are not sustainable in IoT environments, and it motivated us to design a new authentication protocol that overcomes the drawbacks of existing researches and ensures both security and efficiency.

1.2. Main Contributions

The main contributions of this paper are as follows.

- We analyze that Rajaram et al. scheme [15] is vulnerable to offline password guessing, impersonation, privileged insider, and known session-specific temporary information attacks, and does not provide user anonymity and untraceability.
- We propose a secure, lightweight, and anonymous authentication protocol, which uses only hash and exclusive-or operations, making it suitable for IoT environments.
• We provide informal analyses of the proposed protocol and perform a security analysis under the real-or-random (RoR) model to prove the session key security.
• We prove correctness of the proposed protocol using the BAN logic, and show that the proposed protocol is secure against replay and MiTM attacks using automated validation of internet security protocols and application (AVISPA) simulation tool.
• We provide comparison results between the proposed protocol with existing protocols proposed in IoT environments. It shows that our protocol has lower computation and communication costs than the existing protocols, and that the proposed protocol is more suitable to real IoT environments.

2. Related Works

Over the past few years, many authentication protocols have been proposed for IoT environments. In 2018, Chen et al. [16] proposed a lightweight and anonymous authentication scheme for IoT environments. They used ECC for authentication and analyzed the scheme using the BAN logic. They also simulated computation and communication costs using C++. Dhillon and Kalra [17] proposed a three-factor user authentication scheme for IoT-based healthcare systems. They handled authentication between medical professionals and a cloud server. Thakare and Kim [18] indicated security and efficiency of the existing protocols and proposed an ECC-based authentication scheme for IoT environments. They formally analyzed the scheme using both AVISPA and ProVerif tools to prove security and correctness. However, these schemes [16–18] suffer from high computation cost, as they used an elliptic curve cryptosystem.

Some studies proposed lightweight authentication protocols using only hash and exclusive-or operations. In 2014, Kumari et al. [19] proposed a two-factor remote user authentication scheme for distributed systems. They claimed that their scheme has various security characteristics including resistance to smart card stolen attack and user impersonation attack. However, Kaul and Awasthi [20] indicated that the Kumari et al. scheme is vulnerable to smart card loss attacks. They proposed an enhanced authentication protocol and formally analyzed it using the AVISPA simulation tool. Kang et al. [21] showed that Kaul and Awasthi’s scheme cannot guarantee user anonymity, and is not secure against offline password guessing and desynchronization attacks. They proposed a biometric-based key agreement scheme. However, they do not consider known session-specific temporary information attacks. Rana et al. [22] also asserted that Kaul and Awasthi’s scheme cannot resist user impersonation attacks using a stolen smart card, and proposed a lightweight authentication scheme, which is suitable for IoT infrastructures. However, their scheme does not consider known session-specific temporary information attacks and cannot ensure user untraceability.

In 2019, Rajaram et al. [15] proposed a bilinear pairing based user authentication scheme. They claimed that their scheme is secure against off-line guessing, privileged insider, and impersonation attacks and can ensure mutual authentication. However, we observe that Rajaram et al. scheme cannot defend against mentioned attacks, and is vulnerable to known session-specific temporary information attack and suffers from lack of user anonymity. Their scheme also requires high computation cost as it uses bilinear pairing operation. In this paper, we propose a secure, lightweight, and anonymous user authentication scheme that resolves the above mentioned issues and is suitable for IoT environments.

3. Review of Rajaram et al.’s Scheme

Rajaram et al.’s scheme consists of initialization, user registration, user login, user authentication, and password update phase. Table 1 represents the notations of Rajaram et al.’s scheme and the detailed description of each phase is as follows.
The registration phase is performed via a secure channel. Figure 2 shows the registration phase of Rajaram et al. scheme.

### Notations of the Rajaram et al. scheme.

| Notation | Description |
|----------|-------------|
| $U_x$ | User/client $X$ |
| $ID_x, PW_x$ | Identity and password of $U_x$ |
| $SC_x$ | Smart card of $U_x$ |
| $RS$ | The remote server |
| $G_1$ | Cyclic additive group |
| $G_2$ | Cyclic multiplicative group |
| $RP$ | Generator of $G_1$ |
| $q$ | Large prime order of the cyclic groups |
| $h_{f_1} : \{0, 1\}^* \to Z_q$ | Cryptographic hash function |
| $e : G_1 \times G_1 \to G_2$ | Bilinear mapping |
| $sK$ | Secret key of $RS$ |
| $P_{pub} = sK \cdot RP$ | Public key of $RS$ |
| $y_1, y_2$ | Session random numbers |
| $ssK$ | Session key between $U_x$ and $RS$ |
| $*$ | Multiplication operation |

### 3.1. Initialization Phase

$RS$ generates a large prime number $q$, chooses an additive cyclic group $G_1$ and a multiplicative cyclic group $G_2$ with order $q$, and selects a cryptographic hash function $h_{f_1} : \{0, 1\}^* \to Z_q$. Thereafter, $RS$ chooses a generator $RP \in G_1$ and calculates $e(RP, RP) = a$, and selects a secret key $sK$ and the public key of $RS$ is $P_{pub} = sK \cdot RP$. $RS$ keeps $sK$ securely and publishes $(q, G_1, G_2, RP, a, h(\cdot), P_{pub})$.

### 3.2. User Registration Phase

$U_x$ registers to $RS$ using $ID_x$ and $PW_x$. Firstly, $U_x$ chooses a $y \in Z_q$ and computes $PWD_x = h_{f_1}(PW_x || y)$. Thereafter, $U_x$ sends $(ID_x, PWD_x)$ to $RS$. After $RS$ receives the message, $RS$ computes $R_x = h_{f_1}(sK, ID_x) \oplus PWD_x$ and $REG_{ID_x} = h_{f_1}(sK, ID_x) \cdot PWD_x \cdot RP$. After that, $RS$ stores $(R_x, REG_{ID_x}, h_{f_1}(\cdot), RP, P_{pub})$ in $SC_x$. Thereafter, $RS$ issues $SC_x$ to $U_x$, and $U_x$ stores $y$ in $SC_x$. Subsequently, $(R_x, REG_{ID_x}, h_{f_1}(\cdot), RP, P_{pub}, y)$ is stored in $SC_x$. The registration phase is performed via a secure channel. Figure 2 shows the registration phase of Rajaram et al. scheme.

### 3.3. User Login Phase

A registered $U_x$ can be offered services form $RS$ by inserting $SC_x$ to the card reader. After $U_x$ inputs $ID_x^*$ and $PW_x^*$ to $SC_x$, $SC_x$ computes $PWD_x^* = h_{f_1}(PW_x^* || y)$, $L_0 = R_x \oplus PWD_x^*$,
and \( REG_{ID_{t}} = L_{0} \cdot PWD_{X}^{*} \cdot RP \). Thereafter, \( SC_{x} \) checks whether \( REG_{ID_{t}}^{*} \stackrel{?}{=} REG_{ID_{t}} \). If they are equal, \( SC_{x} \) chooses \( y_{1} \) and computes \( L_{1} = h_{f_{1}}(y_{1}||T_{1}) \), \( L_{2} = h_{f_{1}}(PWD_{x}||L_{1}) \), \( L_{3} = (L_{0} + L_{2}) \cdot P_{pub} \), and \( L_{4} = L_{1} \oplus PWD_{x}^{*} \). \( T_{1} \) is a current timestamp. Thereafter, \( U_{x} \) sends \((ID_{x}, R_{x}, L_{3}, L_{4}, T_{1})\) to \( RS \) through a wireless channel. Figure 3 shows the login phase of Rajaram et al. scheme.

| \( U_{x} \) | \( RS \) |
|----------------|----------------|
| inputs \( ID_{x}, PWD_{x}^{*} \) in \( SC_{x} \) | \( SC_{x} \) computes \( PWD_{x}^{*} = h_{f_{1}}(PWD_{x}||y) \) |
| \( REG_{ID_{x}} = L_{0} \cdot PWD_{x}^{*} \cdot RP \) | checks \( REG_{ID_{x}} \stackrel{?}{=} REG_{ID_{x}} \) |
| generates \( y_{1} \in Z_{q}^{*} \) and timestamp \( T_{1} \) | computes \( L_{1} = h_{f_{1}}(y_{1}||T_{1}) \), \( L_{2} = h_{f_{1}}(PWD_{x}||L_{1}) \), \( L_{3} = (L_{0} + L_{2}) \cdot P_{pub} \), \( L_{4} = L_{1} \oplus PWD_{x}^{*} \), \((ID_{x}, R_{x}, L_{3}, L_{4}, T_{1})\) |

**Figure 3.** The login phase of the Rajaram et al. scheme.

### 3.4. User Authentication Phase

After receiving the message from \( U_{x} \), \( RS \) verifies the validity of \( ID_{x} \) and checks whether \( |T_{1} - T_{1}^{*}| \leq \Delta T \), where \( T_{1}^{*} \) is a timestamp when \( RS \) received the message and \( \Delta T \) is a tolerance. Thereafter, \( RS \) computes \( A_{0} = h_{f_{1}}(sK||ID_{x}) \), \( PWD_{x} = R_{x} \oplus A_{0} \), \( L_{1} = PWD_{x} \oplus L_{4} \), and \( L_{2} = h_{f_{1}}(PWD_{x}||L_{1}) \). Subsequently, \( RS \) checks whether \( e(L_{3}, RP) \stackrel{?}{=} \alpha^{sK \cdot (A_{0} + L_{2})} \). If they are equal, \( RS \) chooses \( y_{2} \in Z_{q}^{*} \) and computes \( A_{1} = h_{f_{1}}(y_{2}||T_{2}), A_{2} = h_{f_{1}}(A_{0}||A_{1}), A_{3} = A_{2} \cdot PWD_{x} \cdot RP, y_{3} = L_{1} \oplus A_{1} \), and session key \( ssK = h_{f_{1}}(A_{2}||L_{1}||A_{1}||L_{2}) \). Then, \( RS \) sends \((A_{3}, y_{3}, T_{2})\) to \( U_{x} \). After \( U_{x} \) receives the message, \( U_{x} \) checks whether \( |T_{2} - T_{2}^{*}| \leq \Delta T \) and computes \( A_{1}^{*} = y_{3} \oplus L_{1}, A_{2}^{*} = h_{f_{1}}(L_{0}||A_{1}^{*}) \), and \( A_{3}^{*} = A_{2}^{*} \cdot PWD_{x} \cdot RP \). If \( A_{3} = A_{3}^{*} \), \( U_{x} \) computes session key \( ssK = h_{f_{1}}(A_{2}||L_{1}||A_{1}||L_{2}) \). Figure 4 shows the authentication phase of Rajaram et al. scheme.

### 3.5. Password Change Phase

The password change phase of the Rajaram et al. scheme can be conducted through a new registration (see Section 3.2) or during the authentication phase (see Sections 3.3 and 3.4).
checks format of IDx and T1
computes
A0 = hf1(sK||IDx)
PWDx = Rx ⊕ A0
derives
L1 = hf1(y1||T1) by calculating PWDx ⊕ L4
computes
L2 = hf1(PWDx||L1)
checks
e(L3, RP) = asK(α + L3)
chooses y2 ∈ Z∗q
computes
A1 = hf1(y2||T2)
A2 = hf1(A0||A1)
A3 = A2 · PWDx · RP
y3 = L1 ⊕ A1
ssK = hf1(A2||L1)||A1||L2)

check T2
derives
A∗1 = y3 ⊕ L1 = hf1(y2||T2)
computes
A∗2 = hf1(L0||A∗1) A∗3 = A∗2 · PWDx · RP
checks
A3 = A∗3
computes
ssK = hf1(A2||L1||A1||L2)

Figure 4. The authentication phase of the Rajaram et al. scheme.

4. Cryptanalysis of the Rajaram et al. Scheme

Rajaram et al. asserted that their scheme has resistance to impersonation and privileged insider attacks. However, we observed that their scheme cannot prevent impersonation and privileged insider attacks. We also analyzed the Rajaram et al. scheme with the CK model and found that it is vulnerable to known session-specific temporary information attacks.

4.1. Adversary Model

To analyze the security of the proposed protocol, we adopt the Dolev–Yao (DY) adversary model [23]. The DY model is an widely-accepted adversary model, which is applied to various authentication protocols [24–27]. An adversary A has the following assumptions in the DY model.

• A has full control over transmitted messages on wireless channels. A can eavesdrop, delete, inject, and modify messages.
• A can steal a smart card of a legitimate user and can extract the stored value using power analysis [28,29].
• A can use the values obtained from the previous assumptions to attempt active attacks such as off-line guessing, impersonation, and session key disclosure.

We also apply the Canetti and Krawczyk (CK) adversary model [30] to further analyze the proposed protocol. The CK model has a stronger assumption compared to the DY model, and is also widely used in the analysis of an authentication protocol [31–33]. In the CK model, A can compromise session states and hijack random values generated in each session.
A without being masked and encrypted. Therefore, A can attempt off-line password guessing attack as follows.

- A guesses password $PW_x^A$ from a password dictionary, and then computes $PVD_x^A = hf_1(PW_x^A||y)$. 
- Thereafter, A computes $L_0^A = R_x \oplus PWD_x^A$ and $REG_{ID_x}^A = L_0^A \cdot PWD_x^A \cdot RP$, and checks whether $REG_{ID_x}^A = REG_{ID_x}$. 
- If it satisfies, A succeeds in guessing the correct password. If not, A performs the process again from the beginning.

### 4.3. Impersonation Attack

Using the guessed password as described in Section 4.2, A can impersonate as $U_x$.

- A generates a timestamp $T_A$ and $y_1^A \in Z_3^*$. 
- A computes $L_1^A = h f_1(y_1^A||T_A)$, $L_2^A = h f_1(PWD_x||L_1^A)$, $L_3^A = (L_0^A + L_2^A) \cdot P_{pub}$, and $L_4^A = PWD_x \oplus L_1^A$. 
- A sends $(ID_x||L_1^A||L_3^A||L_4^A||T_A)$ to RS. After RS receives the message, RS regards the message as sent by $U_x$, and sends a response message.

A succeeds in impersonating as $U_X$. Therefore, Rajaram et al.’s scheme cannot defend against the impersonation attack.

### 4.4. Privileged Insider Attack

If A is an privileged insider, A can obtain the registration request message $(ID_x, PWD_x)$ of $U_x$ and can attempt to discover the session key using the information. To calculate $ssK$, A can use $R_x, L_3, L_4,$ and $y_3$, which are obtained from open channels. A computes $L_0 = R_x \oplus PWD_x, L_1 = PWD_x \oplus L_4,$ and $L_2 = h f_1(PWD_x||L_1)$.

Thereafter, A computes $A_1 = y_3 \oplus L_1$ and $A_2 = h f_1(L_0||A_1)$. Finally, A can calculate $ssK = h f_1(A_2||L_1||A_1||L_2)$, and therefore, Rajaram et al.’s scheme cannot resist a privileged insider attack.

### 4.5. Known Session-Specific Temporary Information Attack

Under the CK model, A can comprise a session and obtain session random numbers $y_1$ and $y_2$. Then, A can disclose the session key $ssK$ using the random numbers. A can obtain the messages $(ID_x, R_x, L_3, L_4, T_1)$ and $(A_3, y_3, T_2)$, which are transmitted during the authentication phase. Thereafter, A can compute $L_1 = h f_1(y_1, T_1)$, $PWD_x = L_4 \oplus L_1$, $L_2 = h f_1(PWD_x, L_1)$, $A_1 = y_3 \oplus L_1$, $A_2 = h f_1(R_x \oplus PWD_x||A_1)$. A can calculate $A_3, A_2, L_1$, and $L_2$, and obtain $ssK = h f_1(A_2, L_1, A_1, L_2)$. Therefore, Rajaram et al. scheme is vulnerable to known session-specific temporary information attacks.

### 4.6. User Anonymity and Untraceability

In Rajaram et al.’s scheme, user identity $ID_x$ is transmitted on a wireless channel without being masked and encrypted. Therefore, A can obtain $ID_x$ from transmitted messages to trace $U_x$. Rajaram et al.’s scheme cannot support privacy-preservation and can raise serious security issues in several IoT environments, such as healthcare systems and smart home.

### 5. Proposed Scheme

The proposed scheme includes an initialization, user registration, user login, user authentication, and password update phase. Table 2 describes the notations of the proposed scheme.
5.1. Initialization
RS chooses a large prime \( q \), generates a secret key \( s \in Z_q^* \), and selects a cryptographic hash function \( h(\cdot) : 0, 1^* \rightarrow Z_q \). RS publishes \( (q, h(\cdot)) \) and keeps \( s \) securely.

5.2. User Registration Phase
Before the authentication, \( U_X \) registers to RS. \( U_X \) chooses \( ID_X \) and \( PW_X \), and generates \( r \in Z_q \). \( U_X \) computes \( PWD_X = h(PW_X^r) \), and then sends \( (ID_X, PWD_X) \) to RS. RS checks whether \( ID_X \) is already registered after receiving the message. If not, RS chooses \( t \in Z_q^* \), and computes \( TID_X = h(ID_X^t) \) and \( PID_X = h(TID_X^s) \). Thereafter, RS stores \( (TID_X, h(ID_X)||PWD_X) \) in secure memory. RS generates a fuzzy extractor \( l \in [2^4, 2^8] \) and stores \( (TID_X, PID_X, h(\cdot), l) \) in smart card \( SC_X \). RS transmits \( SC_X \) to \( U_X \). Subsequently, \( U_X \) computes \( A_X = r \oplus h(ID_X)||PW_X \), \( B_X = TID_X \oplus h(ID_X)||PW_X||r \), \( C_X = PID_X \oplus h(TID_X||r) \), and \( Auth_X = h(TID_X||PID_X) \mod l \). \( U_X \) replaces \( PID_X \) to \( (A_X,B_X,C_X,Auth_X) \) in \( SC_X \). The proposed user registration phase is shown in Figure 5.

**Table 2. Notations of the proposed scheme.**

| Notation      | Description                                      |
|---------------|--------------------------------------------------|
| \( U_X \)    | User/client X                                   |
| \( ID_X, PW_X \) | Identity and password of \( U_X \)              |
| \( SC_X \)   | Smart card of \( U_X \)                         |
| \( TID_X \)  | Temporary identity of \( U_X \)                 |
| \( PID_X \)  | Secret temporary identity of \( U_X \)          |
| RS           | The remote server                                |
| \( h : \{0, 1\}^* \rightarrow Z_q \) | Cryptographic hash function                     |
| \( a_s, b_s \) | Session random numbers                           |
| \( s \)      | Secret key of RS                                 |
| \( SK \)     | Session key between RS and \( U_X \)            |

**Figure 5. Proposed registration phase.**
5.3. User Login Phase

$U_X$ inputs $ID_X'$ and $PW_X'$ in $SC_X$. Thereafter, $SC_X$ computes $r' = A_X \oplus h(ID_X'||PW_X')$, $TID_X' = B_X \oplus h(ID_X'||PW_X'||r')$, and $PID_X' = C_X \oplus h(TID_X'||r')$, and checks $Auth_X = h(TID_X'||PID_X') \mod l$. If they are equal, $SC_X$ generates $a_X \in Z_q^{*}$ and current timestamp $T_1$, and computes $M_1 = h(PID_X||h(ID_X||PWD_X)) \oplus a_X$ and $M_2 = h(TID_X||PID_X||a_X||T_1)$. Thereafter, $SC_X$ sends $(TID_X,M_1,M_2,T_1)$ to $RS$. The proposed login phase is shown in Figure 6.

| $U_X$ | RS |
|-------|-----|
| inputs $ID_X', PW_X'$ in $SC_X$ | |
| $SC_X$ computes $r' = A_X \oplus h(ID_X'||PW_X')$ | |
| $TID_X' = B_X \oplus h(ID_X'||PW_X'||r')$ | |
| $PID_X' = C_X \oplus h(TID_X'||r')$ | |
| checks $Auth_X = h(TID_X'||PID_X') \mod l$ | |
| generates $a_X \in Z_q^{*}$ and timestamp $T_1$ | |
| computes $M_1 = h(PID_X||h(ID_X||PWD_X)) \oplus a_X$ | |
| $M_2 = h(TID_X||PID_X||a_X||T_1)$ | |
| $(TID_X,M_1,M_2,T_1)$ | |

Figure 6. Proposed login phase.

5.4. User Authentication Phase

After $RS$ receives the authentication request message, $RS$ checks whether $|T_1 - T_1'| \leq \Delta T$. Subsequently, $RS$ retrieves $h(ID_X||PWD_X)$ using $TID_X'$, computes $PID_X = h(TID_X||s)$ and $a_X = h(PID_X||h(ID_X||PWD_X)) \oplus M_1$, and checks $M_2 = h(TID_X||PID_X||a_X||T_1)$. If it is valid, $RS$ generates $b_X \in Z_q^{*}$ and current timestamp $T_2$. Thereafter, $RS$ generates $TID_X^{new} = TID_X \oplus b_X$, $PID_X^{new} = h(TID_X^{new}||s)$, $M_3 = h(PID_X||h(ID_X||PWD_X)) \oplus b_X$, $M_4 = PID_X^{new} \oplus h(TID_X^{new}||h(ID_X||PWD_X)||b_X)$, $SK = h(PID_X||a_X||b_X)$, and $M_5 = h(SK||PID_X^{new}||T_2)$, where $TID_X^{new}$ and $PID_X^{new}$ are updated temporary identity and secret identity of $U_X$, respectively. $RS$ sends $(M_3,M_4,M_5,T_2)$ to $U_X$. After reception of the message, $U_X$ checks whether $|T_2 - T_2'| \leq \Delta T$, computes $b_X = h(PID_X||h(ID_X||PWD_X)) \oplus M_3$, $TID_X^{new} = TID_X \oplus b_X$, $PID_X^{new} = M_4 \oplus h(TID_X^{new}||h(ID_X||PWD_X)||b_X)$, and $SK = h(PID_X||a_X||b_X)$, and checks whether $M_5 = h(SK||PID_X^{new}||T_2)$. If they are equal, the session key is established. Thereafter, $U_X$ computes $B_{new} = TID_X^{new} \oplus h(ID_X'||PW_X'||r')$, $C_{new} = PID_X^{new} \oplus h(TID_X^{new}||r')$, and $Auth_{new} = h(TID_X^{new}||PID_X^{new}) \mod l$. Subsequently, $U_X$ updates $(B_X,C_X,Auth_X)$ to $(B_{new},C_{new},Auth_{new})$ in $SC_X$. The proposed mutual authentication phase is shown in Figure 7.
6.1.2 Off-Line Guessing Attack

We informally describe that the proposed protocol is secure against the following attacks. After the authentication phase, UX generates a new password PWXnew and a random number rnew, computes PW X new = h(PWX new | rnew), and sends a password change request message to RS including (IDX, PW X new). Thereafter, RS updates h(IDX||PW X new) to h(IDX||PW X new) and the password update is completed.

6. Security Analysis

We analyze the security of the proposed protocol using informal analysis and formal analysis such as the BAN logic, RoR model, and AVISPA.

6.1 Informal Analysis

We informally describe that the proposed protocol is secure against the following attacks.

6.1.1 Replay and MITM Attacks

In the proposed protocol, each message transmitted during the authentication contains a timestamp and random number. Timestamps T1 and T2, and random numbers aX and bX are included in message hash values M2 and M3. An adversary cannot forge these message hash values, and therefore, the proposed protocol is secure against replay and MITM attacks.

6.1.2 Off-Line Guessing Attack

By assumption of the adversary model, an adversary A can attempt off-line guessing attack using transmitted messages (TIDX, M1, M2, T1) and (M3, M4, M5, T2), and the extracted values (AX, BX, CX, AuthX, h(·, l)) from SCX. Using a password dictionary, A can guess PWX and calculate

\[ r^* = A_X \oplus h(ID_X||PW_X) \]

\[ TID_X = B_X \oplus h(ID_X||PW_X^r) \]

**Figure 7.** Proposed authentication phase.

5.5 Password Change Phase

After the authentication phase, UX generates a new password PWXnew and a random number rnew, computes PWXnew = h(PWXnew | rnew), and sends a password change request message to RS including (IDX, PWXnew). Thereafter, RS updates h(IDX||PWXnew) to h(IDX||PWXnew) and the password update is completed.

| UX | RS |
|---|---|
| (TIDX, M1, M2, T1) |  |
| \( h(ID_X||PW_D_X) \) using TIDX | computes PIDX = h(TIDX|s) |
| checks T1 | retrieves h(IDX||PW_D_X) using TIDX |
| checks M2 = h(TIDX||PIDX|aX|T1) | generates bX ∈ Z∗ and timestamp T2 |
| computes | computes |
| TIDnew = TIDX \oplus bX | bX = h(PIDX||h(IDX||PW_D_X)) \oplus M3 |
| PIDnew = h(TIDnew|s) | M3 = h(PIDX||h(IDX||PW_D_X)) \oplus bX |
| M4 = PIDnew \oplus h(TIDnew||h(IDX||PW_D_X)||bX) | SK = h(PIDX|aX|bX) |
| SK = h(PIDX|aX|bX) | checks M5 = h(SK||PIDXnew) |
| checks M5 = h(SK||PIDXnew) | computes bnew = TIDnew \oplus h(IDX||PW||r) |
| computes | Cnew = PIDnew \oplus h(TIDnew||r) |
| Authnew = h(TIDnew||PIDnew) mod l | updates \( (BX, CX, Auth_X) \) to \( (B_{new}, C_{new}, Auth_{new}) \) |
and \( P ID^X = C_X \oplus h(T ID^X || r) \), and can check whether \( Auth^X = h( T ID^X || P ID^X ) \mod l \). However, \( A \) cannot know that the guessed \( PW^*_X \) is legitimate because \( Auth^X \) is masked using fuzzy verifier \( l \). The probability that \( PW^*_X \) is a legitimate password is \( \frac{2^n}{|\text{hash}^2|} \approx \frac{1}{10^7} \), which is negligible.

6.1.3. Impersonation Attack

\( A \) can attempt to impersonate as \( U_X \) and send the authentication request message to \( RS \). \( A \) can obtain \( T ID^X \) from the transmitted message. However, referring to Section 6.1.2, \( A \) cannot generate a valid authentication request message because \( A \) cannot guess the correct \( PW^*_X \) in polynomial time and \( A \) cannot calculate the valid \( M_1 \) and \( M_2 \). Therefore, the proposed protocol can defend against the impersonation attack.

6.1.4. Session Key Disclosure Attack

\( A \) can directly attempt to compute \( SK \) using transmitted messages and extracted values of \( SC_X \). \( A \) has to obtain \( P ID^X, a_X \), and \( b_X \) to calculate \( SK = h(P ID^X || a_X || b_X) \). However, \( A \) cannot acquire any of these values because they are masked using \( s, ID^X, \) and \( P W D^X \), which are secret values. The proposed protocol has resistance to session key disclosure attacks.

6.1.5. Perfect Forward Secrecy

If the long-term secret key \( s \) is compromised to \( A \), then \( A \) can compute \( P ID^X = h(T ID^X || s) \), where \( T ID^X \) is obtained from a public channel. However, \( RS \) cannot calculate \( a_X \) and \( b_X \) without obtaining \( h(ID^X || P W D^X) \), which is stored in the secure memory of \( RS \). Therefore, \( A \) cannot calculate \( SK \) and the proposed protocol supports perfect forward secrecy.

6.1.6. Privileged Insider Attack

As we described in Section 4.4, \( A \) can obtain the message transmitted during the registration phase, and can use it to calculate \( SK \). \( A \) can calculate \( h(ID^X || P W D^X) \) using \( (ID^X, P W D^X) \), and can obtain messages transmitted on public channels. In this scenario, \( A \) still cannot calculate the session key without knowing \( P ID^X \), which is masked with secret key \( s \). \( A \) also cannot calculate \( a_X \) and \( b_X \). Therefore, the proposed protocol is secure against privileged insider attacks.

6.1.7. Stolen Verifier Attack

When the verification table stored in \( RS \) can be compromised, then \( A \) can obtain the list of \( h(ID || P W D) \). \( A \) can retrieve the corresponding \( h(ID^X || P W D^X) \) from \( T ID^X \). However, \( A \) cannot calculate \( P ID^X \) or obtain \( a_s \) and \( b_s \), which are necessary to compute \( SK \). The proposed protocol has resistance to stolen verifier attacks.

6.1.8. Known Session-Specific Temporary Information Attack

\( A \) can hijack the session and obtain random numbers \( a_x \) and \( b_x \), which are generated during the authentication. \( A \) cannot calculate \( P ID^X \) without knowing \( RS \)'s secret key \( s \). Additionally, \( A \) cannot gain any information for a future authentication request message of \( U_X \). Even if session random numbers are leaked to \( A \), the proposed protocol is secure.

6.1.9. User Anonymity and Untraceability

In the proposed protocol, \( ID^X \) is not transmitted in public channels. Instead, \( U_X \) uses temporal identity \( T ID^X \) for authentication. \( T ID^X \) is updated to \( T ID^X_{\text{new}} \) in every session, and \( T ID^X_{\text{new}} \) is veiled until \( U_X \) request a new authentication to \( RS \). Therefore, user anonymity is ensured in the proposed protocol. Additionally, \( A \) cannot find the connection between \( T ID^X \) and \( T ID^X_{\text{new}} \) to trace \( U_X \).
6.2. Formal Analysis Using the Ban Logic

The BAN logic [34] is a widely-accepted [35–37] analysis method to verify the correctness of an authentication protocol. The BAN logic is simple but robust validation logic to prove the mutual authentication of an authentication protocol. In the BAN logic analysis, we set the idealized form of the transmitted messages and assumptions of the BAN logic proof. Thereafter, we perform the proof based on the BAN logic rules. We have presented notations of the BAN logic in Table 3, then describe the assumptions and idealized forms, and conduct the BAN logic proof in turn.

Table 3. BAN logic notations.

| Notation | Description |
|----------|-------------|
| \( \pi_1, \pi_2 \) | Two principals |
| \( \sigma_1, \sigma_2 \) | Two statements |
| \( \pi_1 \equiv \sigma_1 \) | \( \pi_1 \) believes \( \sigma_1 \) |
| \( \pi_1 \sim \sigma_1 \) | \( \pi_1 \) once said \( \sigma_1 \) |
| \( \pi_1 \Rightarrow \sigma_1 \) | \( \pi_1 \) controls \( \sigma_1 \) |
| \( \pi_1 \leftarrow \mu_1 \) | \( \pi_1 \) receives \( \sigma_1 \) |
| \( \#(\sigma_1) \) | \( \sigma_1 \) is fresh |
| \( \sigma_1 \) | \( \pi_1 \) is encrypted with \( K \) |
| \( \pi_1 \leftrightarrow \pi_2 \) | \( \pi_1 \) and \( \pi_2 \) have shared key \( K \) |
| \( SK \) | The session key |

6.2.1. BAN Logic Rules

The basic rules of the BAN logic are as followings:

1. Message meaning rule (MMR):

   \[
   \pi_1 \equiv \pi_1 
   \pi_1 \sim \sigma_1 
   \pi_1 \leftrightarrow \pi_2 
   \pi_1 \leftrightarrow \pi_2 \rightarrow ^{K} \pi_2, \quad \rho_1 \leftarrow (\sigma_1)^{K}
   \]

2. Nonce verification rule (NVR):

   \[
   \pi_1 \equiv \#(\sigma_1), \quad \pi_1 \equiv \pi_2 \sim \sigma_1
   \]

3. Jurisdiction rule (JR):

   \[
   \pi_1 \equiv \pi_2 \Rightarrow \sigma_1, \quad \pi_1 \equiv \pi_2 \equiv \sigma_1
   \]

4. Belief rule (BR):

   \[
   \pi_1 \equiv (\sigma_1, \sigma_2)
   \]

5. Freshness rule (FR):

   \[
   \pi_1 \equiv \#(\sigma_1)
   \]
6.2.2. Goals

The goals are to show that \( UX \) and \( RS \) believe that they agreed on the same session key.

Goal 1: \( UX| \equiv UX \xleftarrow{SK} RS \)

Goal 2: \( UX| \equiv RS| \equiv UX \xleftarrow{SK} RS \)

Goal 3: \( RS| \equiv UX \xleftarrow{SK} RS \)

Goal 4: \( RS| \equiv UX| \equiv UX \xleftarrow{SK} RS \)

6.2.3. Idealized Forms

The idealized forms of the messages exchanged during the authentication can be described as follows.

\( \text{Msg}_1 \): \( UX \rightarrow RS : (TID_X, a_X, T_1)_{PID_X} \)

\( \text{Msg}_2 \): \( RS \rightarrow UX : (PID_{new_X}, a_X, b_X, T_2)_{PID_X} \)

6.2.4. Assumptions

The basic assumptions for the BAN logic proof are as follows:

A₁: \( RS| \equiv \#(T_1) \)

A₂: \( UX| \equiv \#(T_2) \)

A₃: \( UX| \equiv RS \Rightarrow (UX \xleftarrow{SK} RS) \)

A₄: \( RS| \equiv UX \Rightarrow (UX \xleftarrow{SK} RS) \)

A₅: \( UX| \equiv UX \xleftarrow{PID_X} RS \)

A₆: \( RS| \equiv UX \xleftarrow{PID_X} RS \)

6.2.5. Ban Logic Proof

We perform the BAN logic proof of the proposed protocol as below:

Step 1: \( RS \) receives \( \text{Msg}_1 \).

\( S_1 : RS \triangleleft (TID_X, a_X, T_1)_{PID_X} \)

Step 2: Applying \( S_1 \) and \( A_6 \) to the MMR, we can obtain \( S_2 \).

\( S_2 : RS| \equiv UX| \sim (TID_X, a_X, T_1) \)

Step 3: Applying \( S_2 \) and \( A_1 \) to the FR, we can obtain \( S_3 \).

\( S_3 : RS| \equiv \#(TID_X, a_X, T_1) \)

Step 4: Applying \( S_2 \) and \( S_3 \) to the NVR, we can obtain \( S_4 \).

\( S_4 : RS| \equiv UX| \equiv (TID_X, a_X, T_1) \)

Step 5: We can obtain \( S_5 \) applying \( S_4 \) to the BR.

\( S_5 : RS| \equiv UX| \equiv (a_X) \)

Step 6: \( UX \) receives \( \text{Msg}_2 \).

\( S_8 : UX \triangleleft (PID_{new_X}, a_X, b_X, T_2)_{PID_X} \)
Step 7: Applying $S_6$ and $A_5$ to the MMR, we can obtain $S_7$.

$$S_7 : U_X | RS | \equiv (PID_X^w, a_X, b_X, T_2)$$

Step 8: Applying $S_7$ and $A_2$ to the FR, we can obtain $S_8$.

$$S_8 : U_X | RS | \equiv (PID_X^w, a_X, b_X, T_2)$$

Step 9: Applying $S_7$ and $S_8$ to the NVR, we can obtain $S_9$.

$$S_9 : U_X | RS | \equiv (PID_X^w, a_X, b_X, T_2)$$

Step 10: We can obtain $S_{10}$ applying $S_9$ to the BR.

$$S_{10} : U_X | RS | \equiv (PID_X^w, a_X, b_X)$$

Step 11: From $S_5$ and $S_{10}$, $RS$ and $U_X$ can compute session key $SK = h(PID_X || a_X || b_X)$.

$$S_{11} : U_X | RS | \equiv (U_X \leftarrow \rightarrow RS) \quad \text{(Goal 2)}$$

$$S_{12} : RS | U_X | \equiv (U_X \leftarrow \rightarrow RS) \quad \text{(Goal 4)}$$

Step 12: The JR can be applied to $S_{11}$ and $S_{12}$ using $A_3$ and $A_4$, respectively.

$$S_{11} : U_X | \equiv (U_X \leftarrow \rightarrow RS) \quad \text{(Goal 1)}$$

$$S_{12} : RS | \equiv (U_X \leftarrow \rightarrow RS) \quad \text{(Goal 3)}$$

Finally, the user and server are mutually authenticated each other.

6.3. Formal Analysis Using the Ror Model

The RoR model [38] is a formal analysis method of an authentication protocol [39–42]. The RoR model is used to verify the semantic security of session key $SK$ of an authentication protocol probabilistically. Under the RoR model, an adversary $A$ can attempt passive and active attacks represented by queries to find $SK$. Before conducting the RoR model based analysis, we explain notations and queries to be used. We denote $p^t$ as a participant with the $t^{th}$ instance. Thereafter, $p_{U_X}^t$ and $p_{RS}^t$ are participants $U_X$ and $RS$ with $t^{th}$ and $t^{th}$ instances, respectively. Additionally, the queries executed by $A$ are presented in Table 4.

| Query         | Description                                                                 |
|---------------|-----------------------------------------------------------------------------|
| Execute($p_{U_X}^t$, $p_{RS}^t$) | $A$ can perform eavesdropping attack using this query. $A$ can obtain messages transmitted during the authentication between $p_{U_X}^t$ and $p_{RS}^t$ |
| CorruptSC($p_{U_X}^t$) | $A$ can steal the smart card of $p_{U_X}^t$ and extract the stored data using this query. |
| Send($p^t$, $M$) | Using this query, $A$ sends message $M$ to participant $p^t$ to receive the response of sent messages. |
| Test($p^t$) | $A$ executes this query at the end of each game. When this query is executed, an unbiased coin $c$ is flipped. The heads represents 1 and the tails represents 0. After $c$ is flipped, $p^t$ returns $SK$ when $c = 1$ and a random number when $c = 0$. Otherwise, $p^t$ returns NULL. If $A$ can distinguish between $SK$ and a random value, $A$ wins the game. |

A can get the output value of the hash function for the input by performing a Hash query. When $Adv(A)$ is the probability of the advantage for $A$ to break the session key, we prove the following equation:
\[
Adv(A) \leq \frac{q_h^2}{|\text{Hash}|} + \frac{2q_{\text{send}}}{|D_{id}D_{pw}|},
\]

where \(q_h\) represents the number of hash queries performed by \(A\), \(|\text{Hash}|\) represents the range space of the cryptographic hash function, \(q_{\text{send}}\) represents the number of \(\text{Send}\) queries performed by \(A\), and \(|D|\) represents the size of the password dictionary.

**Proof.** \(A\) plays four games \(G_i (i = 0, 1, 2, 3)\) under the RoR model. We denote \(Adv^{G_i}(A)\) is a advantage of \(A\) to compromise \(SK\) after playing \(G_i\).

- \(G_0\): \(A\) selects the bit \(c\) at the start of the game. In the first game, \(A\) has no information about \(SK\) and no queries to perform. By the definition of semantic security, we can induce the following equation.

\[
Adv(A) = |2Adv_{G_0}(A) - 1|.
\]

- \(G_1\): \(A\) attempts an eavesdropping attack in this game. First, \(A\) uses the \(\text{Execute}\) query. Thereafter, \(A\) performs a \(\text{Test}\) query to receive the return value. Finally, \(A\) guesses whether the value is \(SK\) or a random number. For \(A\) to win the game, \(A\) should be able to calculate \(SK\), which is computed by \(PID_X, a_X\), and \(b_X\). These values cannot be obtained using eavesdropping attacks. There is no advantage for \(A\) to be gained by the \(\text{Execute}\) query. Therefore, the advantage of \(A\) at the end of \(G_1\) is equal to that at the end of \(G_0\).

\[
Adv_{G_0}(A) = Adv_{G_1}(A).
\]

- \(G_2\): In this game, \(A\) uses the \(\text{Send}\) and \(\text{Hash}\) queries to compromise \(SK\). From the transmitted messages, \(A\) obtains \((TID_X, M_1, M_2, T_1)\) and \((M_3, M_4, M_5, T_2)\). However, the only way for \(A\) to win the game is to find a hash collision because there is no information about \(SK\) from these values. The advantage function at the end of \(G_2\) is as follows by the birthday paradox.

\[
|Adv_{G_2}(A) - Adv_{G_1}(A)| \leq \frac{q_h^2}{2|\text{Hash}|}.
\]

- \(G_3\): In the final game, \(A\) performs a stolen smart card attack, which is represented as the \(\text{CorruptSC}\) query, then \(A\) can extract \((A_X, B_X, C_X, Auth_X)\) from \(SC_X\). To win the game, \(A\) should succeed in finding the identity \(ID_X\) and password \(PW_X\) simultaneously to login to \(SC_X\). Because \(A\) cannot obtain any information about \(ID_X\) and \(PW_X\), \(A\) should guess the values from identity dictionary \(D_{id}\) and password dictionary \(D_{pw}\). Thereafter, the following equation can be induced.

\[
|Adv_{G_3}(A) - Adv_{G_2}(A)| \leq \frac{q_{send}}{|D_{id}D_{pw}|}.
\]

After all the games are over, \(A\) should guess the correct bit \(c\) from the \(\text{Test}\) query. It follows that

\[
Adv_{G_3}(A) = \frac{1}{2}.
\]

From Equations (2) and (3), we can obtain the following equation:

\[
\frac{1}{2} Adv(A) = |Adv_{G_1}(A) - \frac{1}{2} | = |Adv_{G_1}(A) - Adv_{G_3}(A)|.
\]
We can apply the triangular inequality to Equation (7).

\[ |Adv_{G_3}(A) - Adv_{G_1}(A)| \leq |Adv_{G_3}(A) - Adv_{G_2}(A)| + |Adv_{G_2}(A) - Adv_{G_1}(A)| \]

\[ \leq \frac{q^2_h}{|\text{Hash}|} + \frac{2q_{\text{send}}}{|D_{id}D_{pw}|}. \]  

(8)

We can obtain the following equation using Equations (7) and (8).

\[ Adv(A) \leq \frac{q^2_h}{|\text{Hash}|} + \frac{2q_{\text{send}}}{|D_{id}D_{pw}|}, \]  

(9)

which is the same as Equation (1). Finally, we prove the semantic security of the proposed protocol using the RoR model.

6.4. Formal Analysis Using Avispa Simulation

The AVISPA tool [43] can verify that the authentication protocol is secure against replay and MITM attacks by simulating the implementation of the authentication protocol as code. AVISPA is used in many authentication protocols as a formal verification tool [44–46]. It is implemented using the “High-Level Protocol specification Language” (HLPSL) and uses four back-ends: “On-the-fly Model-Checker (OFMC)”, “SAT-based Model-Checker (SATMC)”, “Constraint Logic Based Attack Searcher (CL-AtSe)”, and “Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP)”. The HLPSL2IF translator converts the intermediate format (IF) of the code implemented using HLPSL into the output format (OF). The OF shows the simulation results which include “SUMMARY”, “DETAILS”, “PROTOCOL”, “GOAL”, “BACKEND”, and “STATISTICS”. The SUMMARY shows whether the protocol is safe, the BACKEND gives the name of the back-ends, and STATISTICS shows the time of the trace of the attack. The left side of Figure 8 represents the role of user implemented in AVISPA tool. The remote server is implemented similar to the user. Furthermore, the right side of Figure 8 shows the session, environments, and goals of the proposed protocol.

![Figure 8. Role of user and session, environment, and goal.](image)

Generally, the AVISPA simulation is executed using the OFMC and CL-AtSe models, which support the exclusive-or operation. If the summaries under the two back-end models are safe, we can say that the protocol is secure against replay and MITM attacks. The simulation results are shown in Figure 9. The simulation summaries of the proposed protocol are safe under the OFMC and CL-AtSe back-ends. This means that the back-ends cannot search for an attack. Therefore, we verify that the proposed protocol has resistance to the replay and MITM attacks.
7. Performance Analysis

In this section, we compare the computation and communication cost of the proposed protocols with related protocols [15–22].

7.1. Computation Cost

Referring to the classification in [12], user’s mobile device is class 15, and the remote server is class 19. Considering the different computing powers of user’s mobile device and the remote server, we conducted the experiment in both Raspberry PI (class 14) and desktop environments (class 17). The desktop is quad-core i7-4790 CPU with 16GB RAM and the operation system is the Linux Ubuntu 20.04, and Raspberry PI 3B is a quad-core system with 1GB RAM. We used the MIRACL Cryptography library to implement the operations used for the authentication protocols [47]. The executed operations and execution time are as presented in Table 5. As shown in Table 5, the operations on Raspberry Pi take 2-3 times longer than the operations on desktop.

Table 5. Execution time of each operation.

| Notation | Description                                | Raspberry PI | Desktop |
|----------|--------------------------------------------|--------------|---------|
| $T_{bp}$ | Bilinear pairing operation                 | 46.130 ms    | 13.440 ms |
| $T_{exp}$ | Modular exponentiation operation           | 5.119 ms     | 1.864 ms |
| $T_{bp}^{\text{mul}}$ | Scalar multiplication in pairing-based group | 5.611 ms     | 2.521 ms |
| $T_{bp}^{\text{add}}$ | Point addition in pairing-based group      | 0.043 ms     | 0.018 ms |
| $T_{ec}^{\text{mul}}$ | Scalar multiplication in elliptic curve group | 2.579 ms     | 1.489 ms |
| $T_{ec}^{\text{add}}$ | Point addition in elliptic curve group     | 0.019 ms     | 0.008 ms |
| $T_h$    | SHA-256 hash operation                     | 0.025 ms     | 0.004 ms |

According to the results shown in Table 5, we calculate the total computation cost of the proposed protocol and other related protocols [15–22] as shown in Table 6. Some existing lightweight schemes are more efficient than the proposed scheme. However, the existing lightweight schemes is vulnerable to a variety of attacks. The proposed scheme has considerably higher efficiency than the public key-based protocols and also can provide better security.
Table 6. Computation cost comparison.

| Scheme | User Overhead | Server Overhead |
|--------|---------------|-----------------|
| [15]   | \(3T_{bp} + 7T_h \approx 17.008\) ms | \(T_{bp} + T_{mul} + 6T_h \approx 15.985\) ms |
| [16]   | \(T_{ecc} + 5T_h \approx 2.704\) ms | \(T_{ecc} + 4T_h \approx 1.505\) ms |
| [17]   | \(2T_{mul} + 3T_h \approx 5.233\) ms | \(T_{mul} + T_h \approx 3.002\) ms |
| [18]   | \(2T_{mul} + 8T_h \approx 5.358\) ms | \(T_{mul} + T_h \approx 1.493\) ms |
| [19]   | \(11T_h \approx 0.275\) ms | \(5T_h \approx 0.02\) ms |
| [20]   | \(10T_h \approx 0.25\) ms | \(5T_h \approx 0.02\) ms |
| [21]   | \(8T_h \approx 0.2\) ms | \(4T_h \approx 0.016\) ms |
| [22]   | \(20T_h \approx 0.5\) ms | \(5T_h \approx 0.032\) ms |
| Proposed | \(14T_h \approx 0.35\) ms | \(8T_h \approx 0.032\) ms |

7.2. Communication Cost

We calculate the total bits transmitted during the authentication to compare the communication costs. We assumed that a timestamp is 32 bits, identity and password are 128 bits, SHA-256 hash output is 256 bits, a random number is 256 bits, an elliptic curve point is 320 bits, and an element of a pairing-based group is 1024 bits. The total communication cost of the existing and proposed schemes are as presented in Table 7.

Table 7. Communication cost comparison.

| Scheme | Total Communication Cost | Number of Messages |
|--------|--------------------------|--------------------|
| [15]   | 1824 bits                | 2                  |
| [16]   | 1696 bits                | 2                  |
| [17]   | 1920 bits                | 3                  |
| [18]   | 1804 bits                | 2                  |
| [19]   | 1344 bits                | 2                  |
| [20]   | 960 bits                 | 2                  |
| [21]   | 1536 bits                | 2                  |
| [22]   | 1216 bits                | 2                  |
| Proposed | 1600 bits               | 2                  |

Similar to the results in Table 6, the total communication cost of the proposed scheme is slightly higher than those of the existing lightweight protocols [19–22]. However, the proposed scheme generates lower communication cost than those in [15–18]. Furthermore, as described in Section 7.3, our protocol is more secure than existing schemes.

7.3. Security Features

We analyzed security characteristics of the proposed protocol with related protocols. This includes: A1: “Resistance to replay attack”, A2: “Resistance to privileged insider attack”, A3: “Resistance to known session specific temporary information attack”, A4: “Resistance to off-line guessing attack”, A5: “Resistance to impersonation attack”, A6: “Resistance to session key disclosure attack”, A7: “Support of perfect forward secrecy”, A8: “Support of user anonymity”, A9: “Support of user untraceability”, A10: “Formal proof using the BAN logic” A11: “Formal proof using the RoR model”, and A12: “Formal proof using the AVISPA simulation”. Table 8 shows that the proposed protocol has better security than other protocols.
Table 8. Security features comparison.

| Security Features | [15] | [16] | [17] | [18] | [19] | [20] | [21] | [22] | Proposed |
|-------------------|------|------|------|------|------|------|------|------|----------|
| A1                | O    | O    | O    | O    | O    | O    | O    | O    | O        |
| A2                | X    | -    | O    | O    | O    | O    | O    | O    | O        |
| A3                | X    | O    | O    | -    | -    | -    | -    | -    | O        |
| A4                | X    | O    | O    | O    | X    | O    | X    | O    | O        |
| A5                | X    | O    | O    | X    | X    | X    | O    | O    | O        |
| A6                | O    | O    | O    | O    | O    | X    | X    | O    | O        |
| A7                | -    | O    | O    | O    | -    | -    | -    | -    | O        |
| A8                | X    | O    | X    | X    | X    | X    | -    | -    | O        |
| A9                | X    | O    | X    | X    | X    | X    | X    | X    | O        |
| A10               | O    | X    | X    | X    | X    | X    | X    | X    | O        |
| A11               | X    | O    | X    | X    | X    | X    | X    | O    | O        |
| A12               | X    | X    | O    | X    | X    | O    | X    | O    | O        |

- Not considered. X: Insecure. O: Secure.

8. Conclusions

In this paper, we designed a secure, lightweight, and anonymous authentication protocol suitable for IoT environments. The proposed protocol resolved security flaws of the Rajaram et al. scheme. It uses only hash and exclusive-or operations during the authentication. Therefore, it is considerably more efficient than Rajaram et al. scheme as well as other existing authentication schemes proposed for IoT environments. Additionally, the proposed protocol can defeat various attacks and ensure more security features than the existing protocols. We analyzed the proposed protocol using BAN logic analysis to prove the correctness, the RoR model to formally verify the session key security, and the AVISPA simulation tool to show the resistance against replay and man-in-the-middle (MITM) attacks. The proposed protocol is sustainable because it has a high security level and low computation cost, and it can contribute to increase energy efficiency and reduce economic costs. Therefore, the proposed protocol can be applied to various IoT environments. In the future work, we plan to implement the proposed protocol in practice.

Author Contributions: Conceptualization, S.S.; methodology, S.S. and Y.P. (Yohan Park); software, S.S.; validation, Y.P. (Yohan Park) and Y.P. (Youngho Park); formal analysis, S.S. and Y.P. (Yohan Park); writing—original draft preparation, S.S.; writing—review and editing, Y.P. (Yohan Park) and Y.P. (Youngho Park); supervision, Y.P. (Youngho Park). All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2021R1I1A3059551).

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

References

1. Chen, C.M.; Xiang, B.; Liu, Y.; Wang, K.H. A secure authentication protocol for internet of vehicles. *IEEE Access* 2019, 7, 12047–12057. [CrossRef]
2. Bagga, P.; Das, A.K.; Wazid, M.; Rodrigues, J.J.P.C.; Kim, K.R.C.; Park, Y. On the design of mutual authentication and key agreement protocol in internet of vehicles-enabled intelligent transportation system. *IEEE Trans. Veh. Technol.* 2021, 70, 1736–1751. [CrossRef]
3. Rathee, G.; Ahmad, F.; Sandhu, R.; Kerrache, C.A.; Azad, M.A. On the design and implementation of a secure blockchain-based hybrid framework for industrial Internet-of-Things. *Inf. Process. Manag.* 2021, 58, 102526. [CrossRef]
4. Nikooghadam, M.; Amintooosi, H.; Islam, S.H.; Moghadam, M.F. A provably secure and lightweight authentication scheme for Internet of Drones for smart city surveillance. *J. Syst. Archit.* 2021, 115, 101955. [CrossRef]
5. Barka, E.; Dahmane, S.; Kerrache, C.A.; Khayat, M.; Sallabi, F. STHM: A secured and trusted healthcare monitoring architecture using SDN and blockchain. *Electronics* 2021, 10, 1787. [CrossRef]
6. Wazid, M.; Das, A.K.; Hussain, R.; Succi, G.; Rodrigues, J.J. Authentication in cloud-driven IoT based big data environment: Survey and outlook. *J. Syst. Archit.* 2019, 97, 185–196. [CrossRef]
7. Mahmood, K.; Akram, W.; Shafiq, A.; Altaf, I.; Lodhi, M.A.; Islam, S.H. An enhanced and provably secure multi-factor authentication scheme for Internet-of-Multimedia-Things environments. *Comput. Elect. Eng.* 2020, 88, 106888. [CrossRef]
8. Belghazi, Z.; Benamar, N.; Addaim, A.; Kerrache, C.A. Secure WiFi-direct using key exchange for IoT device-to-device communications in a smart environment. *Future Internet* **2019**, *11*, 251. [CrossRef]

9. Banerjee, S.; Das, A.K.; Chattopadhyay, S.; Jamal, S.S.; Rodrigues, J.J.; Park, Y. Lightweight failover authentication mechanism for IoT-based fog computing environment. *Electronics* **2021**, *10*, 1417. [CrossRef]

10. Oh, J.; Yu, S.; Lee, J.; Son, S.; Kim, M.; Park, Y. A secure and lightweight authentication protocol for IoT-based smart homes. *Sensors* **2021**, *21*, 1488. [CrossRef]

11. Das, A.K.; Wazid, M.; Yannam, A.R.; Rodrigues, J.J.; Park, Y. Provably secure ECC-based device access control and key agreement protocol for IoT environment. *IEEE Access* **2019**, *7*, 55382–55397. [CrossRef]

12. Terminology for Constrained-Node Networks. Available online: https://datatracker.ietf.org/doc/draft-bormann-lwig-7228bis/06/ (accessed on 17 August 2020).

13. Miller, V.S. Use of elliptic curves in cryptography. In Proceedings of the Conference on the Theory and Application of Cryptographic Techniques, Linz, Austria, 9–11 April 1985; pp. 417–426.

14. Boneh, D.; Franklin, M. Identity-based encryption from the Weil pairing. In *Advances in Cryptology*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 213–229.

15. Rajaram, S.; Maitra, T.; Vollala, S.; Ramasubramanian, N.; Amin, R. eUASBP: Enhanced user authentication scheme based on bilinear pairing. *J. Ambient Intell. Humaniz. Comput.* **2019**, *11*, 2827–2840. [CrossRef]

16. Chen, Y.; Martínez, J.F.; Castillejo, P.; López, L. A lightweight anonymous client–server authentication scheme for the Internet of things scenario: LAuth. *Sensors* **2018**, *18*, 3695. [CrossRef]

17. Thakare, A.; Kim, Y.G. Secure and efficient authentication scheme in IoT environments. *Appl. Sci.* **2021**, *11*, 1260. [CrossRef]

18. Dhillon, P.K.; Kalra, S. Multi-factor user authentication scheme for IoT-based healthcare services. *J. Reliab. Intell. Environ.* **2018**, *4*, 141–160. [CrossRef]

19. Kumari, S.; Khan, M.K.; Li, X. An improved remote user authentication scheme with key agreement. *Comput. Elect. Eng.* **2014**, *40*, 1997–2012. [CrossRef]

20. Kaul, S.D.; Awasthi, A.K. Security enhancement of an improved remote user authentication scheme with key agreement. *Wirel. Pers. Commun.* **2016**, *89*, 621–637. [CrossRef]

21. Kang, D.; Jung, J.; Kim, H.; Lee, Y.; Won, D. Efficient and secure biometric-based user authenticated key agreement scheme with anonymity. *Secur. Commun. Netw.* **2018**, *2018*, 9046064. [CrossRef]

22. Rana, M.; Wazid, M.; Yannam, A.R.; Rodrigues, J.J.; Park, Y. AKM-IoV: Authenticated key management protocol in fog computing-based Internet of vehicles deployment. *IEEE Internet Things J.* **2019**, *6*, 8804–8817. [CrossRef]

23. Ali, Z.; Ghani, A.; Khan, I.; Chaudhry, S.A.; Islam, S.H.; Giri, D. A robust authentication and access control protocol for securing wireless healthcare sensor networks. *J. Inform. Secur. Appl.* **2020**, *52*, 102502. [CrossRef]

24. Xiong, H.; Wu, Y.; Jin, C.; Kumari, S. Efficient and privacy-preserving authentication protocol for heterogeneous systems in IloT. *IEEE Internet Things J.* **2020**, *7*, 11713–11724. [CrossRef]

25. Lee, J.; Kim, G.; Das, A.K.; Park, Y. Secure and efficient honey list-based authentication protocol for vehicular ad hoc networks. *IEEE Trans. Netw. Sci. Eng.* **2021**. [CrossRef]

26. Kocher, P.; Jaffe, J.; Jun, B. Differential power analysis. In *Advances in Cryptology*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 388–397.

27. Messerges, T.S.; Dabbish, E.A.; Sloan, R.H. Examining smart-card security under the threat of power analysis attacks. *IEEE Trans. Comput.* **2002**, *51*, 541–552. [CrossRef]

28. Canetti, R.; Krawczyk, H. Analysis of key-exchange protocols and their use for building secure channels. In *Advances in Cryptology*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 453–474.

29. Foo, M.P.; Yuen, C.S.; Cheung, R.K.; Lo, K.L. A lightweight and secure biometric-based authentication scheme for wireless healthcare sensor networks. *Future Internet* **2021**, *13*, 9241.
38. Abdalla, M.; Fouque, P.A.; Pointcheval, D. Password-based authenticated key exchange in the three-party setting. In Proceedings of the 8th International Workshop on Theory and Practice in Public Key Cryptography (PKC’05), Les Diablerets, Switzerland, 23–26 January 2005; pp. 65–84.

39. Kwon, D.; Yu, S.; Lee, J.; Son, S.; Park, Y. WSN-SLAP: Secure and lightweight mutual authentication protocol for wireless sensor networks. Sensors 2021, 21, 936. [CrossRef] [PubMed]

40. Das, A.K.; Bera, B.; Wazid, M.; Jamal, S.S.; Park, Y. iGCACS-IoD: An Improved Certificate-Enabled Generic Access Control Scheme for Internet of Drones Deployment. IEEE Access 2021, 9, 87024. [CrossRef]

41. Lee, J.; Yu, S.; Kim, M.; Park, Y.; Lee, S.; Chung, B. Secure key agreement and authentication protocol for message confirmation in vehicular cloud computing. Appl. Sci. 2020, 10, 6268. [CrossRef]

42. Ayub, M.F.; Shamshad, S.; Mahmood, K.; Islam, S.H.; Parizi, R.M.; Choo, K.K.R. A provably secure two-factor authentication scheme for USB storage devices. IEEE Trans. Consum. Elect. 2020, 66, 396–405. [CrossRef]

43. AVISPA. Automated Validation of Internet Security Protocols and Applications. Available online: http://www.avispa-project.org/ (accessed on 17 August 2021).

44. Yu, S.; Lee, J.; Park, K.; Das, A.K.; Park, Y. IoV-SMAP: Secure and efficient message authentication protocol for IoV in smart city environment. IEEE Access 2020, 8, 167875–167886. [CrossRef]

45. Banerjee, S.; Odelu, V.; Das, A.K.; Chattopadhyay, S.; Park, Y. An efficient, anonymous and robust authentication scheme for smart home environments. Sensors 2020, 20, 1215. [CrossRef]

46. Kim, M.; Lee, J.; Park, K.; Park, Y.; Park, K.; Park, Y. Design of secure decentralized car-sharing system using blockchain. IEEE Access 2021, 9, 54796–54810. [CrossRef]

47. MIRACL Cryptographic SDK: Multiprecision Integer and Rational Arithmetic Cryptographic Library. Available online: https://github.com/miracl/MIRACL (accessed on 17 August 2021).