.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

SMASG: Secure Mobile Authentication Scheme for Global Mobility Network

JIHYEON RYU[1], HAKJUN LEE[2], YOUNGSOOK LEE[2], DONGHO WON[3]

[1]Department of Software, Sungkyunkwan University, 2066 Seoburo, Suwon, Gyeonggido, 16419, Korea (e-mail: jhryu@security.re.kr)
[2]Department of ITSoftwareSecurity, Howon University, 64 Impimyeon, Gunsan, Jeollabukdo, Korea (e-mail: hjlee@security.re.kr, ysooklee@howon.ac.kr)
[3]Department of Electrical and Computer Engineering, Sungkyunkwan University, 2066 Seoburo, Suwon, Gyeonggido, 16419, Korea (e-mail: dhwon@security.re.kr)

Corresponding author: Dongho Won (e-mail: dhwon@security.re.kr).

This work was supported by an Institute of Information & Communications Technology Planning Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2021-0-00558, Development of National Statistical Analysis System using Homomorphic Encryption Technology)

ABSTRACT The rapid growth of the Internet of Things (IoT) has enabled prompt services over mobile devices. The Global Mobility Network (GLOMONET) is an important global network that allows mobile users to access the Internet anywhere. Although implementing a secure mechanism in GLOMONET is a difficult and complex task due to the computational and processing limitations of most mobile devices, an authentication system is vital for secure communications among such mobile devices. In 2021, Rahmani et al. proposed an authentication method, called the advanced mobile authentication protocol for GLOMONET (AMAPG). However, we found three serious vulnerabilities in AMAPG. First, the scheme contains large amounts of information on the smart card of the mobile phone. Therefore, they are vulnerable to attacks that steal critical information. Second, it is susceptible to password-guessing attacks. Third, the scheme cannot guarantee the security of future messages because attackers can steal the session key. In this study, we discuss the weaknesses of AMAPG and propose a new three-factor authentication scheme called the secure mobile authentication scheme for GLOMONET (SMASG). We performed informal and formal security analyses using ProVerif and BAN Logic on SMASG. In addition, we analyzed and compared its performance with that of the latest GLOMONET-based authentication schemes. Our scheme saves an average of 93% time in user login and authentication phase.

INDEX TERMS Authentication GLOMONET IoT

I. INTRODUCTION

Advancements in the Internet of Things (IoT) have facilitated global access to networks through mobile devices. Thus, people can operate these devices from any location. Furthermore, the automated exchange of information among devices and information available over a network helps connected users obtain the desired information [1], [2].

A global mobility network (GLOMONET) [3]–[11] provides security to mobile users accessing the network from anywhere. Global roaming services enable legitimate mobile users to use ubiquitous services. However, with the rapid development of this environment, numerous security issues such as user privacy have risen [11]–[14]. Therefore, anonymous mutual authentication in GLOMONET is important. For this purpose, cryptographers worldwide are developing computationally complex processes based on symmetric/asymmetric encryption/decryption or using modular operations to design authentication protocols [15]–[19]. These protocols must handle various security issues such as forgery attacks, known as session-key attacks, reverse and forward secrecy, and smart card loss issues.

In GLOMONET, authentication is generally divided into three categories, authentication for: (1) mobile users (MU), (2) home agents (HA), and (3) foreign agents (FA) . In the registration stage, MU registers with HA and is issued a smart card. In the subsequent authentication step, MU enters the login process with its information and the smart card to request a session key. FA receives information from MU, requests authentication from HA including its information, and receives a message from HA. It then generates a session key and sends a message to MU. Then, MU generates a session key using the received message (Figure 1).

In 1998, Horn and Preneel [3] first proposed a mobile pay authentication method. Since then, several studies have been

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

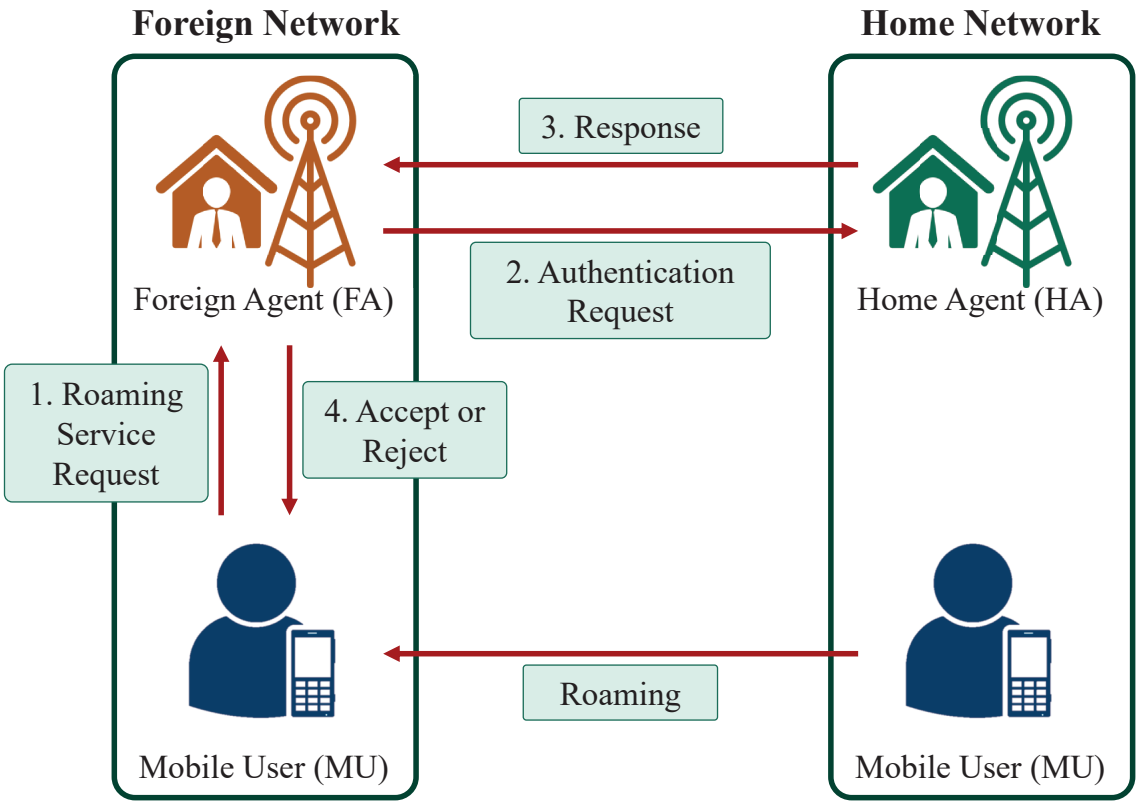

FIGURE 1. Mobile User Authentication Scenarios for GLOMONET.

conducted on mobile pay-related authentication. In 2004, Zhu and Ma [4] first proposed a different GLOMONET authentication method for mobile users, foreign agents, and home agents. However, their proposed method did not satisfy perfect backward secrecy, mutual authentication, or protect against a forgery attack [5]. Lee-Whang-Liao [5] proposed a novel authentication method to address these problems. Chang-Chi-Liu [6] found that Lee-Hwang-Liao's scheme had a weakness in time synchronization and proposed a new scheme; however, the new scheme faced user anonymity and confidentiality challenges [7]. Zhou and Xu [7] introduced a wireless authentication protocol to address these problems. Unfortunately, Gope and Hwang [8] observed that their scheme was also insecure owing to unsuccessful key agreements, replay attacks, and insider attacks; they then proposed a novel scheme to address these vulnerabilities. Xu et al. proposed mutual authentication and key agreement (MAKA) in 2018 [9] as a new method to prevent the storage consumption, computational burden, and replay attack problems faced by the scheme designed by Gope and Hwang [8]. However, in 2020, Shashidhara et al. [10] analyzed and identified problems such as untraceability, impersonation attacks, denial of service attacks, privileged-insider attacks, clock synchronization, and wrong password detection in this scheme. They presented an efficient protocol to address problems, such as the rapid detection of incorrect passwords. However, in the scheme proposed by Shashidhara et al. [10], Rahmani et al. [11] in 2021 discovered problems such as user impersonation, traceability, forward secrecy contradiction, and stolen smart card attacks; they proposed a new scheme, an advanced mobile authentication protocol for GLOMONET (AMAPG), to resolve these schemes [11].

However, AMAPG [11] has three critical vulnerabilities. First, the scheme stores the information on the smart card of mobile phones. Therefore, it is susceptible to attacks that steal critical information. Second, the scheme can be exposed to password-guessing incidents. Third, their protocol cannot guarantee the security of future messages, as attackers can steal the session key. In the following sections, we explain the weaknesses of AMAPG and propose a new secure mobile authentication scheme for GLOMONET (SMASG) that compensates for these weaknesses. The contributions of this study can be summarized as follows:

- We summarize the security properties required for GLOMONET. The following aspects must be satisfied: user anonymity, low communication cost, computational complexity, single registration, user-friendliness, no password table, security.
- However, the recently proposed AMAPG scheme allows password-guessing attacks. In addition, the AMAPG has a fatal problem in that the session key can be calculated by an external attacker. To solve this problem, we used the user's biometric information for authentication. Biometrics are included in the authentication phase,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

and our new SMASG method achieves robust security. Our scheme presents a three-factor method, including biometric authentication, in line with the recent mobile authentication trends. The user's biometric information is randomized using a fuzzy extractor and is used for user authentication.
- We conducted security and performance analyses of SMASG and compared its safety and performance with the latest GLOMONET schemes.

The remainder of this paper is organized as follows. Section II provides a preliminary overview of the basic elements used in this study and describes the threat model and assumptions. Section III provides a review of AMAPG, and Section IV analyzes its security vulnerabilities. Section V proposes a novel three-step authentication scheme called SMASG that compensates for the weaknesses of AMAPG. Sections VI and VII present the security and performance analysis results, respectively. Section VIII discusses the performance, and Section IX concludes the paper.

II. PRELIMINARIES

This section introduces the fuzzy extractor, hash function, and threat model used in the study.

A. FUZZY EXTRACTOR

The fuzzy extractor receives the user's biometric information and can use the error tolerance to obtain a unique string. This error tolerance can distinguish biometric information from the same individual even if the biometric information is not exactly the same. This character string is easy to use because it allows an error range for recognizing the biometric information. A fuzzy extractor uses two operators [20]–[25].

$$GEN\ (B) \rightarrow \langle P, R \rangle \tag{1}$$

$$REP\ (B^*, P) = R \tag{2}$$

GEN and REP are probabilistic and deterministic reproduction functions, respectively. Gen returns a factored-out string $P \in \{0,1\}^k$ for input biometrics B and a coadjutant string $R \in \{0,1\}^*$. Rep is a function that restores R to P, and any vector B^* close to B.

B. THREAT MODEL

Based on previous studies [27]–[29], this study establishes a threat model with the following assumptions:

- An attacker can steal the user's smart card and identity.
- Attackers can eavesdrop on messages shared on public channels. In other words, attackers can eavesdrop on the interactions between mobile users (MU) and the foreign agents (FA) and between foreign and home agents (HA).
- An attacker can discover information on a smart card through a side-channel attack.

C. SECURITY PROPERTY IN GLOMONET

For GLOMONET, mobile device-specific network communication must be applied. The requirements of the user authentication scheme for GLOMONET are as follows:

- **User anonymity**: When an unauthorized attacker eavesdrops on a message, they can track the real-time location of the users from their identities. Hence, GLOMONET requires a protocol that renders its user anonymous.
- **Computational efficiency**: The usable space of a mobile device is limited; thus, if the protocol occupies a large amount of space, its usefulness decreases. Therefore, an authentication scheme should consider the computational efficiency of the device to which it is to be applied.
- **One-time registration**: Mobile users must register only once with their home agents to access the global network.
- **User friendliness**: The registration, login, and authentication phases of the scheme should be easy to use and understand.
- **No password table**: The foreign or home agents should not have a password table for mobile users.
- **No time synchronization**: User authentication schemes should avoid serious time-synchronization problems.
- **Security**: The authentication method should be able to defend the system against real-world security attacks (privileged-insider, replay, traceability, etc., attacks).

III. REVIEW OF RAHMANI ET AL.'S ADVANCED MOBILE AUTHENTICATION PROTOCOL FOR GLOMONET (AMAPG)

This section describes the AMAPG target scheme. This scheme consists of three phases: registration, login and authentication, and password change. The notations used in these phases are listed in Table 1.

TABLE 1. Notations.

Notations	Description
MU	Mobile user
HA	Home agent of a mobile user
FA	Foreign agent of the network
MU_{id}	MU's identity
MU_{psw}	MU's password
HA_{id}	HA's identity
FA_{id}	FA's identity
SK	Session key of MU and FA
SK_{HA}	Secret key of HA
SK_{FA}	Secret key of FA, $SK_{FA} = h\ (FA_{id} \parallel SK_{HA})$
SC	Smart card or smart device
DB	HA's database
MU_r	Random number in MU's smart card
$h\ (\cdot)$	One-way hash function
$E\ (F_p)$	Group of points on a finite field F_p elliptic curve
$\oplus$	XOR operation
$\parallel$	Concatenation operation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

A. REGISTRATION PHASE

In the registration step, a smart card is created when the user enters an identity and password; the card stores the user's information with the home agent. The details of the registration phase for AMAPG are as follows:

1) MU calculates the secret information $RID = h\ (MU_{id} \parallel (MU_{psw} \oplus MU_r\ ))$ using the identity MU_{id} and password MU_{psw} to create a smart card. Subsequently, MU sends RID to HA.
2) HA receives RID from MU and calculates the secret information $HID = h\ (RID \parallel SK_{HA})$. Then, HA stores the received information $\{RID\}$ in its database. HA then sends HID and hash function $h\ (\cdot)$ to MU.
3) MU receives HID and hash function $h\ (\cdot)$ information from HA and calculates the $SP = HID \oplus h\ (MU_{psw} \parallel (MU_{id} \oplus MU_r))$ and $PV = h\ (MU_{id} \parallel MU_{psw} \parallel MU_r)$ values. On smart card SC, MU stores the random value MU_r. $SC = \{SP, PV, MU_r, h\ (\cdot)\}$.

B. LOGIN AND AUTHENTICATION PHASE

In the login and authentication phase, the user logs in with the smart card created by the user and shares the session key between the mobile user and the foreign agent.

1) MU requests the reader terminal for login by inputting its smart card SC, identity MU_{id}, and password MU_{psw}.
2) Subsequently, the reader terminal that receives the smart card SC information, identity MU_{id}, and password MU_{psw} calculates $PV^* = h\ (MU_{id} \parallel MU_{psw} \parallel MU_r)$ and checks whether the value PV^* matches the information PV in the smart card SC. If they match, the terminal authenticates MU and generates a random value N_M and a timestamp T_M. Then, it calculates $HID = SP \oplus h\ (MU_{psw} \parallel (MU_{id} \oplus MU_r))$, $A_M = h\ ((HID \oplus N_M) \parallel T_M)$, and $V_1 = h\ (HID \parallel T_M) \oplus N_M$ and sends the final values $\{A_M, V_1, HA_{id}, T_M\}$ to FA.
3) Then, FA receives $\{A_M, V_1, HA_{id}, T_M\}$ from MU and verifies the timestamp T_M. If the verification is confirmed, FA generates a random value N_F and timestamp T_F and calculates $A_F = h\ (A_M \parallel T_F \parallel SK_{FA}) \oplus N_F$ and $V_2 = h\ (A_F \parallel (T_F \parallel N_F) \parallel SK_{FA} \parallel (V_1 \oplus A_M))$. Subsequently, FA sends $\{T_M, T_F, FA_{id}, A_F, V_1, V_2\}$ to HA.
4) HA receives $\{T_M, T_F, FA_{id}, A_F, V_1, V_2\}$ from FA and verifies the timestamps T_M and T_F. Subsequently, it calculates $SK_{FA} = h\ (FA_{id} \parallel SK_{HA})$ and determines FA_{id}. Then, HA calculates $N_F^* = A_F \oplus h\ (A_M \parallel T_F \parallel SK_{FA})$, extracts $\{RID\}$ from the database and computes $HID^* = h\ (RID \parallel SK_{HA})$, $N_M^* = h\ (HID^* \parallel T_M) \oplus V_1$, $A_M^* = h\ ((HID^* \oplus N_M^*) \parallel T_M)$, and $V_2^* = h\ (A_F \parallel (T_F \oplus N_F) \parallel SK_{FA} \parallel (V_1 \oplus A_M^*))$. Subsequently, it verifies $V_2 = V_2^*$, calculates $A_H = A_F \oplus N_F^* \oplus N_M^*$, $V_3 = h\ ((HA_{id} \oplus N_H) \parallel (N_F^* \oplus A_H) \parallel SK_{FA} \parallel T_H)$, and $V_4 = h\ ((HID^* \oplus N_F^*) \parallel (HA_{id} \oplus N_M^*) \parallel N_H \parallel T_H)$, and then, sends the information of $\{T_H, A_H, N_H, V_3, V_4\}$ to FA.
5) FA receives information about $\{T_H, A_H, N_H, V_3, V_4\}$ from HA and calculates $V_3^* = h\ ((HA_{id} \oplus N_H) \parallel (N_F \oplus A_H) \parallel SK_{FA} \parallel T_H)$. Furthermore, FA verifies $V_3 = V_3^*$ and authenticates MU and HA. If they are authenticated, FA calculates N_M and $A_F^{'} = A_M \oplus N_M = \oplus N_F$ to determine the session key $SK = h\ (N_F \parallel N_M \parallel N_H)$. Subsequently, FA sends $\{T_H, N_H, A_F^{'}, V_4\}$ to MU.
6) MU receives $\{T_H, N_H, A_F^{'}, V_4\}$ from FA, calculates $NF = A_F^{'} \oplus A_M \oplus N_M$, $V_4^* = h\ ((HID \oplus N_F) \parallel (HA_{id} \oplus N_M) \parallel N_H \parallel T_H)$, and verifies $V_4 = V_4^*$ to authenticate FA and HA. Furthermore, MU calculates the session key $SK = h\ (N_F \parallel N_M \parallel N_H)$.

C. PASSWORD CHANGE PHASE

The password change phase of AMAPG is performed in a secure channel as follows.

1) The mobile user MU logs in with the identity MU_{id} and password MU_{psw}, gives smart card information $SC = \{SP, PV, MU_r, h\ (\cdot)\}$ to the reader terminal, and requests a password change.
2) MU's smart card SC calculates $PV^* = h\ (MU_{id} \parallel MU_{psw} \parallel MU_r)$ and checks the $PV = PV^*$ information. If approved, MU is verified and the smart card SC provides $HID = SP \oplus h\ (MU_{psw} \parallel (MU_{id} \oplus MU_r))$.
3) When MU inputs a new password MU_{psw}^{new} in the reader terminal, the new $PV^{new} = h(MU_{id} \parallel MU_{psw}^{new} \parallel MU_r)$ and $SP^{new} = HID \oplus h\ (MU_{psw}^{new} \parallel (MU_{id} \oplus MU_r))$ updates the old PV, and SP is replaced with PV^{new} and SP^{new} on the smart card $SC = \{SP^{new}, PV^{new}, MU_r, h\ (\cdot)\}$.

IV. ANALYSIS OF RAHMANI ET AL.'S AMAPG

This section describes the above vulnerabilities in AMAPG step by step.

A. LOSS OF SMART CARD INFORMATION

A side-channel attack can steal information on a smart card. In general, three methods exist for side-channel attacks. We assume that smart-card information can easily be extracted through the following attacks [26]:

1) Timing Attacks: These attacks are calculated by measuring the time taken to perform the unit operation.
2) Power Consumption Analysis Attacks: These attacks depend on power consumption analysis during the encryption operation. These types of attacks are subdivided into simple and co-relation power analysis attacks.
3) Fault Analysis Attacks: Fault analysis attacks are recent and powerful cryptanalysis attacks that induce

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

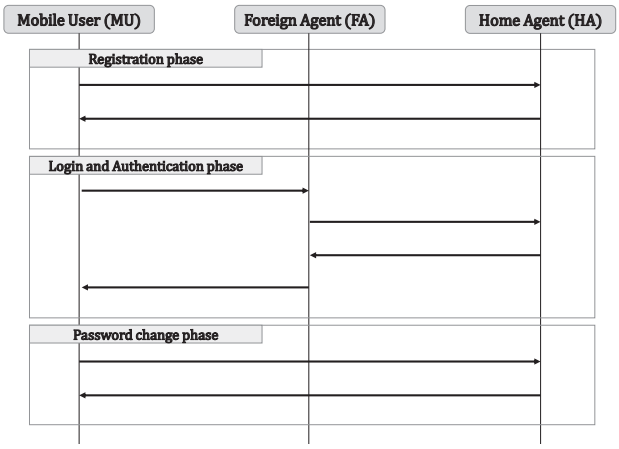

FIGURE 2. Sequence Diagram of SMASG.

faulty operations, with the expectation that the results of the fault operation will leak information regarding the secret keys involved.

B. PASSWORD GUESSING ATTACK

Using stolen smart card information (Section IV-A) and assuming that the user's identity is known, an attack that guesses the user's password can be attempted. The details are as follows.

1) The attacker obtains the PV and MU_r information from the user's smart card. It is also assumed that MU_{id} is known.
2) As it is a $PV = h\left(MU_{id} \parallel MU_{psw} \parallel MU_r\right)$, the attacker enters the user's identity MU_{id}, MU_r, and PV values, and extracts the password.

C. SESSION KEY DISCLOSURE ATTACK

If an attacker is involved in the registration phase, they can steal session keys of the mobile user and the foreign agent.

1) The attacker steals the HID value in the registration phase.
2) The attacker steals the A_M, V_1, and T_M, where MU sends A_F, FA sends N_H, and HA sends A_H, respectively.
3) The attacker computes $N_M = h\left(HID \parallel T_M\right) \oplus V_1$, and $N_F = A_F \oplus A_M \oplus N_M$.
4) Finally, the attacker calculates that $SK = h\left(N_F \parallel N_M \parallel N_H\right)$.

V. SMASG: THE PROPOSED SCHEME

To compensate for the vulnerabilities in AMAPG, we propose a novel scheme, SMASG that uses a fuzzy extractor to authenticate the user's biometric information. It consists of three phases: registration, login and authentication, and password changes, as shown in Figure 2. The details are as follows.

A. REGISTRATION PHASE

MU inputs the user information and receives a smart card SC from the home agent HA. HA provides MU the information required for the smart card and stores the user's information in its database DB. The details are presented in Figure 3.

1) MU inputs identity MU_{id}, password MU_{psw}, and the biometric information MU_{bio}. The fuzzy extractor receives MU_{bio} and generates $(R, P) = GEN\left(MU_{bio}\right)$. Then, MU calculates $x = h\left(MU_{id} \parallel MU_{psw} \parallel R\right)$, $RID = h\left(MU_{id} \parallel R\right)$, and $PID = h\left(MU_{id} \parallel MU_{psw}\right)$ and sends the PID and hash function $h\left(\cdot\right)$ to HA.
2) HA receives information $\{PID, h\left(\cdot\right)\}$ sent by MU and calculates $HID = h\left(r \parallel SK_{HA}\right)$. It stores r and PID in the database $DB = \{r, PID\}$. In addition, HA stores HID in smart card $SC = \{HID\}$ and sends it to MU.
3) MU calculates the SP and PV and stores $\{SP, PV, REP, P, h\left(\cdot\right)\}$ in the smart card SC. The registration step is thus completed.

B. LOGIN AND AUTHENTICATION PHASE

MU and FA must undergo authentication and session-key SK sharing processes. The details are presented in Figure 4.

1) MU inputs the information, such as identity MU_{id}, password MU_{psw}, and biometric information MU_{bio}, to its smart card $SC = \{SP, PV, REP, P, h\left(\cdot\right)\}$ to log in. Biometric information MU_{bio} extracts $R = REP\left(MU_{bio}, P\right)$. MU's smart card SC calculates $x = h\left(MU_{id} \parallel MU_{psw} \parallel R\right)$, $RID = h\left(MU_{id} \parallel R\right)$, $PID = h\left(MU_{id} \parallel MU_{psw}\right)$, $HID = SP \oplus RID$, and $PV = h\left(x \parallel HID\right)$. At this time, the mobile user MU is authenticated; the value of PV is verified to be the same as that of PV in the smart card SC. Subsequently, the reader terminal generates the timestamp T_M and random number N_M and calculates $V_1 = h\left(HID \parallel T_M\right) \oplus N_M$. Finally, MU sends message $\{V_1, HA_{id}, T_M\}$ to FA.
2) Foreign agent FA receives $\{V_1, HA_{id}, T_M\}$ from MU and checks whether T_M is valid. If it is valid, FA generates timestamp T_F and random number N_F. Furthermore, FA computes $V_2 = h\left(V_1 \parallel T_M \parallel T_F \parallel SK_{FA}\right) \oplus N_F$, and $A = h\left(V_2 \parallel N_F\right)$. Finally, FA sends $\{V_1, V_2, T_M, T_F, A\}$ to HA.
3) HA receives $\{V_1, V_2, T_M, T_F, A\}$ from FA to confirm the validity of T_F. If valid, HA calculates $SK_{FA} = h\left(FA_{id} \parallel SK_{HA}\right)$ by checking the identity FA_{id} of FA. After calculating $N_F = h\left(V_1 \parallel T_M \parallel T_F \parallel SK_{FA}\right) \oplus V_2$ and $A = h\left(V_2 \parallel N_F\right)$, we determine whether A matches received message A. HA calculates $HID = h\left(r \parallel SK_{HA}\right)$, where it determines r by mapping PID to the database DB, and $N_M = h\left(HID \parallel T_M\right) \oplus V_1$, and generates timestamp T_H and random number N_H. Then, HA calculates $V_3 =$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

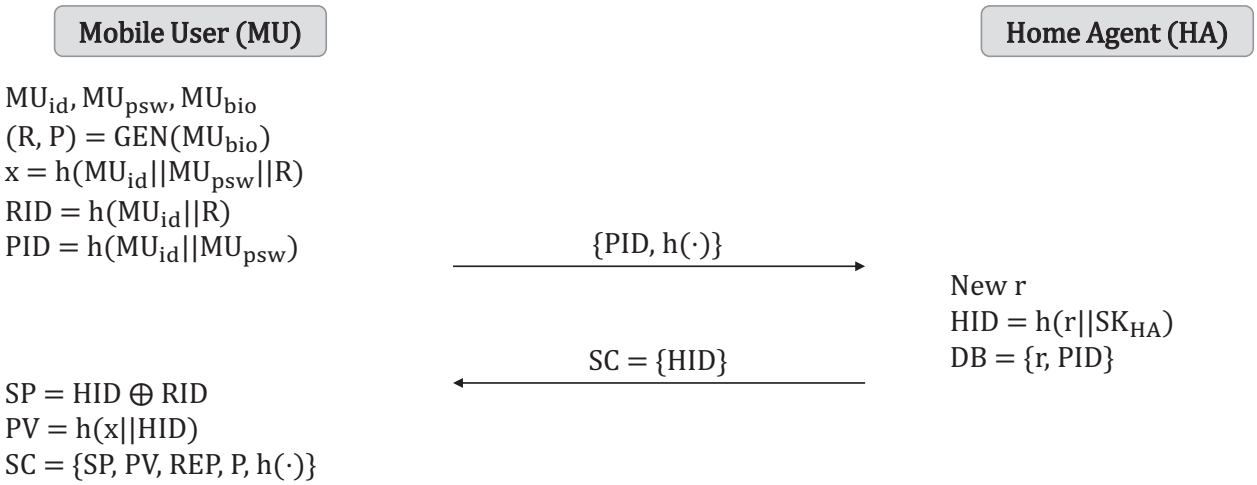

FIGURE 3. Registration Phase of SMASG.

$N_F \oplus N_H$, $V_4 = h\ (V_3 \parallel N_M \parallel N_H \parallel T_H \parallel HID)$, $V_5 = h\ (N_F \parallel T_H)$, and $V_6 = N_M \oplus N_H$, and then, sends $\{V_3, V_4, V_5, V_6, T_H\}$ to FA.

4) FA receives $\{V_3, V_4, V_5, V_6, T_H\}$ from HA and verifies the validity of the timestamp T_H. When the timestamp T_H is validated, FA computes $V_5 = h\ (N_F \parallel T_H)$ and checks whether it is equal to the V_5 of the received message. Subsequently, FA calculates $N_H = V_3 \oplus N_F$, $N_M = V_6 \oplus N_H$, and the session-key $SK = h\ (N_M \parallel N_F \parallel N_H)$. FA creates timestamp T_{F2} and sends $\{V_3, V_4, V_6, T_{F2}, T_H\}$ to MU.
5) MU receives $\{V_3, V_4, V_6, T_{F2}, T_H\}$ from FA and checks the timestamp T_{F2}. Then, we calculate $N_H = V_6 \oplus N_M$, $N_F = V_3 \oplus N_H$, and $V_4 = h\ (V_3 \parallel N_M \parallel N_H \parallel T_H \parallel HID)$, and check whether V_4 is the same as the received V_4. If V_4 is confirmed, we calculate the session key $SK = h\ (N_M \parallel N_F \parallel N_H)$.

C. PASSWORD CHANGE PHASE

We provide users with the opportunity to change their old passwords. What the user has lost is to prepare an option so that the password can be changed regularly for safety if the password is exposed. In SMASG, when MU changes password MU_{psw}, we pursue the following process.

1) MU inputs the original identity MU_{id}, password MU_{psw}^{old}, and biometric information MU_{bio} into its smart card.
2) The smart card calculates $R = REP\ (MU_{bio}, P)$, $x^{old} = h\ (MU_{id} \parallel MU_{psw}^{old} \parallel R)$, $RID = h\ (MU_{id} \parallel R)$, $PID^{old} = h\ (MU_{id} \parallel MU_{psw}^{old})$, $HID = SP \oplus RID$, and $PV^{old} = h\ (x \parallel HID)$, compares PV^{old} with PV^{old} of the smart card SC, and checks whether MU has correctly entered the user information.
3) MU inputs the new password MU_{psw}^{new} into the smart card SC.
4) Smart card SC computes a new $PID^{new} = h\ (MU_{id} \parallel MU_{psw}^{new})$, $x^{new} = h\ (MU_{id} \parallel MU_{psw}^{new} \parallel R)$, and $PV^{new} = h\ (x^{new} \parallel HID)$. MU's smart card sends PID^{new} along with the original PID^{old} to HA on a secure channel, such that HA updates the PID^{old} information in its database DB with PID^{new}.
5) Smart card SC finally updates the original PV^{old} using the information from the new PV^{new}.

VI. SECURITY ANALYSIS OF SMASG

In this section, we analyze the security of SMASG in two ways: formal and informal security analyses. We used the formal protocol verification tool called ProVerif and BAN logic in Section VI-A to demonstrate the security of our scheme. We also provide a theoretical security analysis of this protocol in Section VI-B. Through this verification, we demonstrate the safety of the proposed scheme.

A. FORMAL SECURITY ANALYSIS

We verified the protocol using two well-known security-analysis tools. The first method involves verification using Proverif software. The second method involves verification using the BAN logic. The details are as follows:

1) Security proof through Proverif

We used ProVerif to analyze the security and correctness of the proposed scheme. ProVerif has been widely used to verify security protocols [30], [31], [35]. This software tool formally verifies the security of cryptographic protocols. We define basic cryptographic primitives, such as hash functions, encryption, digital signatures, and bit commitment.

This tool can systematically prove cryptographic properties such as reachability, secrecy, correspondence, and some observational equivalence properties. ProVerif has two unique design characteristics. First, it uses an extension of pi-calculus with cryptography; thus, it supports various types

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

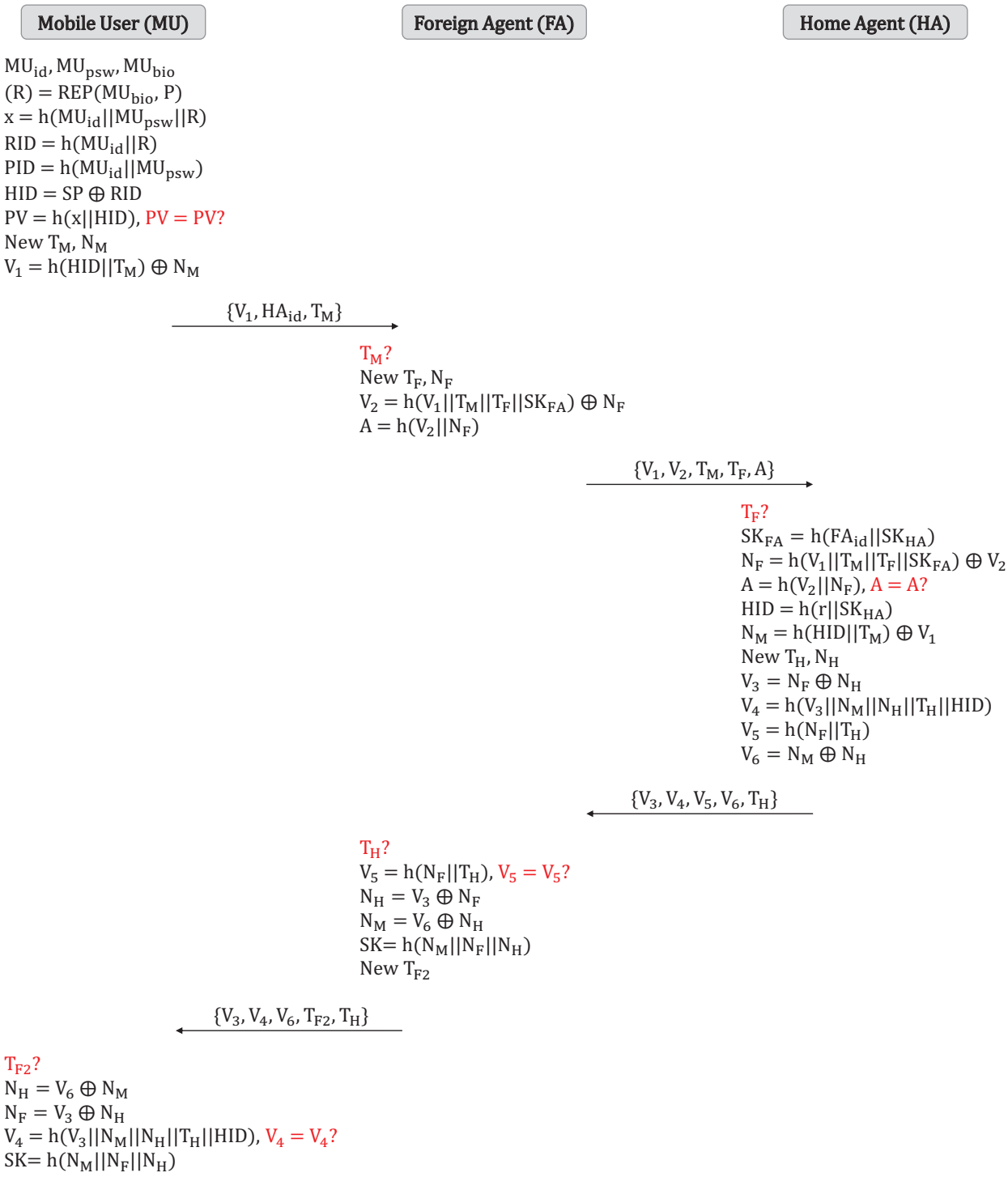

FIGURE 4. Login and Authentication Phase of SMASG.

of cryptographic primitives. In addition, ProVerif analyzes protocols after translating them into Horn clauses; therefore, it can verify the security features in an unbounded number of sessions.

We use three channels: a registration channel (mobile user–home agent channel) (cha), a mobile user–foreign agent channel (chb), and a foreign agent–home agent channel (chc). Table 4 lists the variables, constants, secret keys, functions, and events.

The "Registration" and "Login and Authentication" phases for mobile users (MU) are listed in Table 5. The "Registration" and "Authentication" phases for foreign agent (FA) are shown in Table 6. The "Authentication" phase of the home agent (HA) is presented in Table 7. Tables 2 and 3 list the

TABLE 2. Query.

```
(*—-queries—-*)
query attacker(MUid).
query secret:bitstring; inj-event(endMU(secret)) ==> inj-event(beginMU(secret)).
query secret:bitstring; inj-event(endFA(secret)) ==> inj-event(beginFA(secret)).
query secret:bitstring; inj-event(endHA(secret)) ==> inj-event(beginHA(secret)).
process
((!MU)|(!FA)|(!HA))
```

TABLE 3. Query Results.

```
RESULT inj-event(endMU(secret)) ==> inj-event(beginMU(secret)) is true.
RESULT inj-event(endFA(secret)) ==> inj-event(beginFA(secret)) is true.
RESULT inj-event(endHA(secret)) ==> inj-event(beginHA(secret)) is true.
RESULT not attacker(MUid[]) is true.
```

TABLE 4. Define Values and Functions.

```
(*—-channels—-*)
free cha:channel [private].
free chb:channel.
free chc:channel.
(*—-constants—-*)
free R:bitstring [private].
free MUid:bitstring [private].
free FAid:bitstring [private].
free HAid:bitstring.
free MUpsw:bitstring [private].
(*—-secret key—-*)
free SKHA:bitstring [private].
free SKFA:bitstring [private].
(*—-functions—-*)
fun concat(bitstring, bitstring) : bitstring.
fun xor(bitstring, bitstring) : bitstring.
fun h(bitstring) : bitstring.
equation forall a:bitstring, b:bitstring; xor(xor(a, b), b) = a.
(*—-events—-*)
event beginMU(bitstring).
event endMU(bitstring).
event beginFA(bitstring).
event endFA(bitstring).
event beginHA(bitstring).
event endHA(bitstring).
```

queries and the corresponding results.

When we run the query in Table 2, we obtain the following results:

1) RESULT inj-event(EVENTA) ==> inj-event(EVENTB) is true.
2) RESULT inj-event(EVENTA) ==> inj-event(EVENTB) is false.
3) RESULT not attacker(QUERY) is true.
4) RESULT not attacker(QUERY) is false.

"RESULT inj-event (EVENTA) == > inj-event (EVENTB) is true." indicates that the process from EVENTA to EVENTB has been authenticated. By contrast, "RESULT inj-event (EVENTA) == > inj-event (EVENTB) is false." indicates that the authentication from EVENTA to EVENTB is not successful. "RESULT not attacker (QUERY) is true." implies that an attacker cannot get a free name QUERY, and "RESULT not attacker (QUERY) is false." implies that an attacker can trace the QUERY.

The results for the queries in Table 2 are listed in Table 3. In this case, the authentication process is performed correctly and the attacker cannot obtain $MUid$.

2) Security proof through BAN Logic

We analyzed SMASG using BAN logic, which was created by Burrows, Abadi, and Needham (BAN) [32], and is used to verify the security of many schemes [1], [11]. BAN logic is one of the methods used to verify the scheme authentication and key establishment. To utilize the BAN logic, idealization, assumption, goal, and derivation processes are required; the

TABLE 5. Mobile User Scheme.

```
(*—-MU process—-*)
let MU =
let x = h(concat(h(concat(MUid, MUpsw)), R)) in
let RID = h(concat(MUid, R)) in
let PID = h(concat(MUid, MUpsw)) in
out(cha,(PID));
in(cha,(XHID:bitstring));
let SP = xor(XHID, RID) in
let PV = h(concat(x, XHID)) in
event beginMU(MUid);
new TM:bitstring;
new NM:bitstring;
let V1 = xor(h((concat(XHID, TM)), NM) in
out(chb, (V1, TM));
in(chb, (XXV3:bitstring, XXV4:bitstring, XXV6:bitstring, XTF2:bitstring, XXTH:bitstring));
let XXNH = xor(XXV6, NM) in
let XXNF = xor(XXV3, XXNH) in
let XXXV4 = h(concat(concat(XXV3, NM), concat(XXNH, concat(XXTH, XHID)))) in
if XXXV4 = XXV4 then
let SK = h(concat(h(concat(NM, XXNF)), XXNH)) in
event endMU(MUid).
```

TABLE 6. Foreign Agent Scheme.

```
(*—-FA process—-*)
let FA =
in(chb, (XV1:bitstring, XTM:bitstring));
event beginFA(FAid);
new TF:bitstring;
new NF:bitstring;
let V2 = xor(h(concat(concat(XV1, XTM), concat(TF, SKFA))), NF) in
let A = h(concat(V2, NF)) in
out(chc, (XV1, V2, XTM, TF, A));
in(chc, (XV3:bitstring, XV4:bitstring, XV5:bitstring, XV6:bitstring, XTH:bitstring));
let XXV5 = h(concat(NF, XTH)) in
if XXV5 = XV5 then
let XNH = xor(XV3, NF) in
let XNM = xor(XV6, XNH) in
let XSK = h(concat(h(concat(XNM, NF)), XNH)) in
new TF2:bitstring;
out(chb, (XV3, XV4, XV6, TF2, XTH));
event endFA(FAid).
```

TABLE 7. Home Agent Scheme.

```
(*—-HA process—-*)
let HA =
in(cha, (XPID:bitstring));
new r:bitstring;
let HID = h(concat(r, SKHA)) in
out(cha, (HID));
in(chc, (XXV1:bitstring, XV2:bitstring, XXTM:bitstring, XTF:bitstring, XA:bitstring));
event beginHA(HAid);
let XNF = xor(h(concat(concat(XXV1, XXTM), concat(XTF, SKFA))), XV2) in
let XXA = h(concat(XV2, XNF)) in
if XXA = XA then
let XXNM = xor(h(concat(HID, XXTM)), XXV1) in
new TH:bitstring;
new NH:bitstring;
let V3 = xor(XNF, NH) in
let V4 = h(concat(concat(V3, XXNM), concat(NH, concat(TH, HID)))) in
let V5 = h(concat(XNF, TH)) in
let V6 = xor(XXNM, NH) in
out(chc, (V3, V4, V5, V6, TH));
event endHA(HAid).
```

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

logic verifies whether the results derived through each process are logically reasonable. The BAN logic notations used in this study are as shown in Table 8.

TABLE 8. BAN Logic Notations

Notations	Description
$P\| \equiv X$	P believes that X holds
$P \lhd X$	P sees/holds that X
$P\| \sim X$	P has once said that X
$P \Rightarrow X$	P has complete control over X
$\#(X)$	X is fresh and recent
$P \overset{K}{\longleftrightarrow} Q$	P, and Q shares secret key K
$\langle X \rangle_K$	X encrypted with key K
$(X)_h$	hashed X

We also use the following BAN logic postulates. Assuming that formulas X_1, X_2, ... X_n are performed and Y is performed, it is written as follows:

$$\frac{X_1, X_2, ..., X_n}{Y} \tag{3}$$

According to [1], [11], [32], the following rule is applied.

1) P_1 (Message-meaning rule) : $\dfrac{P| \equiv P \overset{K}{\longleftrightarrow} Q, P \lhd \langle X \rangle_K}{P| \equiv Q| \sim X}$
2) P_2 (Nonce-verification rule) : $\dfrac{P| \equiv \#(X), P| \equiv Q| \sim X}{P| \equiv Q| \equiv X}$
3) P_3 (Believe rule 1) : $\dfrac{P| \equiv X, P| \equiv Y}{P| \equiv (X, Y)}$
4) P_4 (Believe rule 2) : $\dfrac{P| \equiv (X, Y)}{P| \equiv X, P| \equiv Y}$
5) P_5 (Freshness-conjuncatenation rule) : $\dfrac{P| \equiv \#(X)}{P| \equiv \#(X, Y)}$
6) P_6 (Jurisdiction rule) : $\dfrac{P| \equiv Q \Rightarrow X, P| \equiv Q| \equiv X}{P| \equiv X}$

When the message in the registration phase is completed, the messages exchanged in the login and authentication phases are expressed and idealized as follows:

1) When using $M_1 = \{V_1, HA_{id}, T_M\}$, $MU \rightarrow FA$: $V_1 = h(HID \parallel T_M) \oplus N_M$, this is idealized to I_1 : $FA \lhd \langle T_M, N_M, HA_id \rangle_{HID}$.
2) When using $M_2 = \{V_1, V_2, T_M, T_F, A\}$, $FA \rightarrow HA$: $V_1 = h(HID \parallel T_M) \oplus N_M$, $V_2 = h(V_1 \parallel T_M \parallel T_F \parallel SK_{FA}) \oplus N_F$, it is idealized as follows: $I_{21} : HA \lhd \langle T_M, N_M \rangle_{HID}$, $I_{22} : HA \lhd \langle V_1, T_M, T_F, N_F \rangle_{SK_{FA}}$
3) When using $M_3 = \{V_3, V_4, V_5, V_6, T_H\}$, $HA \rightarrow FA$: $V_3 = N_F \oplus N_H$, $V_3 = N_F \oplus N_H$, $V_4 = h(V_3 \parallel N_M \parallel N_H \parallel T_H \parallel HID)$, $V_5 = h(N_F \parallel T_H)$, $V_6 = N_M \oplus N_H$, it is idealized as follows: $I_{31} : FA \lhd \langle N_F, N_M, N_H, T_H \rangle_{HID}$, $I_{32} : FA \lhd \langle N_M, N_H, N_F, T_H \rangle_{SK_{FA}}$
4) When using $M_4 = \{V_3, V_4, V_6, T_H, T_{F2}\}$, $FA \rightarrow MU$: $V_3 = N_F \oplus N_H$, $V_4 = h(V_3 \parallel N_M \parallel N_H \parallel T_H \parallel HID)$, $V_6 = N_M \oplus N_H$, it is idealized to: I_4 : $MU \lhd \langle N_F, N_M, N_H, T_H \rangle_{HID}$

To derive the goal of our scheme, we make the following assumptions:

1) $A_1 : MU| \equiv \#(N_M)$
2) $A_2 : MU| \equiv \#(T_M)$
3) $A_3 : FA| \equiv \#(N_F)$
4) $A_4 : FA| \equiv \#(T_F)$
5) $A_5 : HA| \equiv \#(N_H)$
6) $A_6 : HA| \equiv \#(T_H)$
7) $A_7 : MU| \equiv (MU \overset{\text{HID}}{\longleftrightarrow} HA)$
8) $A_8 : HA| \equiv (MU \overset{\text{HID}}{\longleftrightarrow} HA)$
9) $A_9 : FA| \equiv (FA \overset{SK_{FA} = h(FA_{id} || SK_{HA})}{\longleftrightarrow} HA)$
10) $A_{10} : HA| \equiv (FA \overset{SK_{FA} = h(FA_{id} || SK_{HA})}{\longleftrightarrow} HA)$
11) $A_{11} : MU| \equiv HA \Rightarrow SK$
12) $A_{12} : FA| \equiv HA \Rightarrow SK$

SMASG proves that the following two conditions are satisfied, similar to the method using BAN logic in AMAPG.

1) $G_1 : FA| \equiv SK$
2) $G_2 : MU| \equiv SK$

To prove that $G_1 : FA| \equiv SK$, the following is derived:

1) Given I_{32} and A_9, using P_1, we get $D_1 : FA| \equiv HA | \sim \{N_M, N_H, N_F, T_H\}$
2) When A_5 is applied to P_5, the following result can be obtained: $D_2 : FA| \equiv \#(\{N_M, N_H, N_F, T_H\})$
3) Applying D_1 and D_2 to P_2 gives the following: D_3 : $FA| \equiv HA | \equiv \{N_M, N_H, N_F, T_H\}$
4) When D_3 is applied to P_4, D_4, D_5 and D_6 can be obtained as follows: $D_4 : FA| \equiv HA| \equiv N_M$, $D_5 : FA| \equiv HA| \equiv N_F$, $D_6 : FA| \equiv HA| \equiv N_H$
5) When D_4, D_5, and D_6 are applied to P_3, it is expressed as follows: $D_7 : FA| \equiv (N_M, N_F, N_H)$
6) When D_7 is hashed and applied, it is expressed as follows: $D_8 : FA| \equiv (N_M, N_F, N_H)_h$ and this value is SK. Therefore, $G_1 : FA| \equiv SK$ was proven.

Similarly, to prove that $G_2 : MU| \equiv SK$, the following is derived:

1) When I_4 and A_7 are applied to P_1, the following result appears: $D_9 : MU| \equiv HA| \sim \{N_F, N_M, N_H, T_H\}$
2) Applying A_1 to P_5, we can get $D_{10} : MU| \equiv \#(\{N_F, N_M, N_H, T_H\})$
3) By applying D_9 and D_{10} to P_2, the following can be extracted: $D_{11} : MU| \equiv HA| \equiv \{N_F, N_M, N_H, T_H\}$
4) When D_{11} is applied to P_4, D_{12}, D_{13}, and D_{14} can be obtained as follows: $D_{12} : MU| \equiv HA| \equiv N_M$, D_{13} : $MU| \equiv HA| \equiv N_F$, $D_{14} : MU| \equiv HA| \equiv N_H$
5) When D_{12}, D_{13}, and D_{14} are applied to P_3, D_{15} can be induced as follows: $D_{15} : MU| \equiv HA| \equiv (N_M, N_F, N_H)$
6) Finally, when D_{15} is hashed and applied, SK can be derived as follows: $D_{15} : MU| \equiv SK = (N_M, N_F, N_H)_h$

B. INFORMAL SECURITY ANALYSIS

We performed a formal analysis in Section VI-A. However, according to [33], [34], formal analysis is not sufficient to prove security. Therefore, we further analyzed our scheme using an informal analysis.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3157871, IEEE Access

TABLE 9. Hardware and Software Conditions.

	Specification
CPU	Intel (R) Core(TM) 2 Quad CPU Q8300, 2.50 Hz
Memory	2G
OS	Win7 Professional
Hash Function	SHA-256
The Symmetric Encryption Algorithm	AES
The As-symmetric Algorithm	ECC

TABLE 10. Comparison of Registration Computation Cost.

	Madhusudhan et al. [13]	**Nikooghadama et al. [14]**	**AMAPG [11]**	**SMASG**
Mobile User MU	T_h	$2T_h$	$3T_h$	$4T_h + T_{Rep}$
Foreign Agent FA	0	T_m	0	0
Home Agent HA	$2T_h + T_m$	$6T_h$	T_h	T_h
Total time cost	$3T_h + T_m$	$8T_h + T_m$	$4T_h$	$5T_h + T_{Rep}$
(ms)	= 51.8	= 54.3	= 2	= 3

TABLE 11. Comparison of Login and Authentication Computation Cost.

	Madhusudhan et al. [13]	**Nikooghadama et al. [14]**	**AMAPG [11]**	**SMASG**
Mobile User MU	$3T_h$	$7T_h + 3T_m + 3T_s$	$6T_h$	$7T_h + T_{Rep}$
Foreign Agent FA	$T_h + 2T_s$	$5T_h + 2T_m + 2T_s$	$4T_h$	$4T_h$
Home Agent HA	$2T_h + T_m + 2T_s$	$4T_h + T_s$	$8T_h$	$7T_h$
Total time cost	$6T_h + T_m + 4T_s$	$16T_h + 5T_m + 6T_s$	$18T_h$	$18T_h + T_{Rep}$
(ms)	= 88.1	= 311.7	= 9	= 9.5

We present a theoretical analysis of the SMASG. Subsequently, we briefly explain the results of the informal security analysis.

1) Privileged Insider Attack

In the registration phase, the mobile user (MU) sends the value $PID = h\ (MU_{id} \parallel MU_{psw})$, created using identity MU_{id} and password MU_{psw} to the home agent (HA). At this time, no information is disclosed, and there is no way to know personal information because $RID = h\ (MU_{id} \parallel R)$, $PID = h\ (MU_{id} \parallel MU_{psw})$, $SP = HID \oplus RID$, and $PV = h\ (h\ (MU_{id} \parallel MU_{psw} \parallel R) \parallel HID)$ are encrypted along with MU's information. Therefore, it is safe against privileged insider attacks.

2) Outsider Attack

The information contained in smart card SC is $\{SP, PV, REP, P, h\ (\cdot)\}$, and the mobile user ($MU$) cannot be identified.

3) Offline ID Guessing Attack

MU's identity is not disclosed in the plain text of the scheme. Although the identity of MU contains information in RID, PID, and x, it is encrypted with the hash functions, R and MU_{psw}.

4) Online ID Guessing Attack

As mentioned in the offline ID-guessing attack, the identity of MU is not disclosed in plain text. Therefore, this protects the protocol from online ID-guessing attacks.

5) Session Key Disclosure Attack

Session-key information is expressed as $SK = h\ (N_M \parallel N_F \parallel N_H)$. At this time, N_M, N_F, and N_H are not directly disclosed, and an outside intruder cannot determine the session key because they cannot be calculated unless they are involved.

6) Mobile User Impersonation Attack

The information in MU is authenticated when HA checks the value of $A = h\ (V_2 \parallel N_F)$. Because we calculate the session-key value using the information generated in A and confirm the information of MU through PID, the protocol is safe from mobile user-impersonation attacks.

7) Home Agent Impersonation Attack

In SMASG, foreign agents FA and MU verify the home agent (HA) in a manner that checks $V_4 = h\ (V_3 \parallel N_M \parallel N_H \parallel T_H \parallel HID)$ and $V_5 = h\ (N_F \parallel T_H)$ values, respectively, to prevent impersonation attacks.

8) Replay Attack

An attacker can send the user MU's previous login message back to FA. However, because the attacker does not have access to the HID, he/she cannot create a session key SK, and therefore, cannot perform a replay attack.

VII. PERFORMANCE ANALYSIS OF SMASG

The four symbols necessary for performance analysis are as follows [36]–[38]: T_{Rep} is the time required to check for a match when recognizing a mobile user (MU)'s biometric MU_{bio}. T_h denotes hash time. T_m denotes the time of the

multiplicative operation used in the elliptic curve cryptography (ECC). T_s denotes the time required for the symmetric encryption or decryption. These values are listed in Table 12. Table 9 lists the computer hardware and software used to calculate the algorithm runtime. We compared our scheme with the state-of-the-art schemes proposed by Madhusudhan et al. [13], Nikooghadama et al. [14], and AMAPG [11].

The costs for the registration phases are listed in Table 10. Table 11 compares the costs of the login and authentication phases.

TABLE 12. Notations of Time Symbol.

Symbol	Meaning	Time (ms)
T_{Rep}	the Time of REP and GEN	0.5
T_h	the Time of Hash Operation	0.5
T_m	the Time of Multiplication in ECC	50.3
T_s	the Time of Symmetric Encryption or Decryption	8.7

The scheme of Madhusudhan et al. [13] uses ECC cryptography, symmetric cryptography, and hash functions. Therefore, the time taken for the registration phase is 51.8 ms, and the time taken for the login and authentication phase is 88.1 ms. Nikooghadama et al. [14]'s scheme also uses the ECC encryption method, symmetric encryption method, and hash function to consume 54.3 ms for the registration phase and 311.7 ms for the login and authentication phase. AMAPG [11] only uses a hash function. At this time, it takes 2 ms for the registration phase and 9 ms for the login and authentication phases.

In contrast, our proposed scheme, SMASG, uses a hash function and a biometric fuzzy extractor, and consumes 3 ms in the registration phase and 9.5 ms in the login and authentication phases. The registration computation cost is listed in Table 10, and the login and authentication costs are listed in Table 11.

VIII. DISCUSSION OF PERFORMANCE

The proposed scheme, SMASG, is a secure user authentication scheme that overcomes the weaknesses of AMAPG [11] and uses biometric information from mobile users. We used a fuzzy extractor to safely extract biometric information.

Our study compares the performance of three schemes [13], [14], and [11] in Section VII. Compared to Madhusudhan et al.'s scheme [13], SMASG takes 0.058 times longer for the registration phase, 0.108 times longer for the login and authentication phases, respectively, 0.055 and 0.030 compared to [14] and 1.5 times compares to [11], it takes 1.056 times the time. Because SMASG is an improved scheme of [11], it overcomes the small gap in time by fully addressing their vulnerabilities.

Therefore, on average, the time taken for the registration phase was reduced by 91.67%, the time taken for the login and authentication phases was reduced by 93.03%, and the performance was greatly improved to 1101.11% and 1334.39%, respectively.

IX. CONCLUSION

A recent study proposed AMAPG, a GLOMONET-based authentication scheme. It is efficient because it is designed to be lightweight and involves simple operations such as hash function and XOR operation; however, we found a critical vulnerability in this protocol. First, smart cards store vital information; therefore, the information is exposed when the smart card is stolen. In addition, it is vulnerable to password-guessing attacks. Third, because attackers can steal session-keys, the security of future messages is not guaranteed.

Three elements were used to solve these AMAPG issues : identity, password, and biometric information. Biometrics is a function used in most mobile devices; therefore, there are no technical problems in its use. SMASG, a new scheme using these three elements, provides security verification of the proposed scheme using ProVerif and shows that it performs better than other proposed schemes.

Our proposed method, SMASG, is a lightweight scheme that can be implemented only with a hash function, XOR operation, and fuzzy extractor. The SMASG assumes that a foreign agent is an honest user. However, in some applications, users may not want to trust the foreign agents. This scenario has not been addressed. Our scheme is not suitable for scenarios in which the mobile user does not trust the foreign agent. Therefore, this case is left for future work.

REFERENCES

[1] H. Lee, D. Kang, J. Ryu, D. Won, H. Kim, Y. Lee, "A three-factor anonymous user authentication scheme for Internet of Things environments," *Journal of Information Security and Applications,* Vol. 52, 102494, 2020.

[2] C. Shi, J. Liu, H. Liu, Y. Chen, "Smart user authentication through actuation of daily activities leveraging WiFi-enabled IoT," *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing,* pp. 1–10, 2017.

[3] G. Horn, B. Preneel, "Authentication and Payment in Future Mobile Systems," *European Symposium on Research in Computer Security,* vol. 98, pp. 277-293, 1998.

[4] J. Zhu, J. Ma, "A New Authentication Scheme with Anonymity for Wireless Environments," *IEEE Transactions on Consumer Electronics,* vol. 50, no. 1, pp. 231–235, 2004.

[5] C. Lee, M. Hwang, I. Liao, "Security enhancement on a new authentication scheme with anonymity for wireless environments," *IEEE Transactions on Industrial Electronics,* vol. 53, no. 5, pp. 1683-1687, 2006.

[6] C. Chang, C. Lee, Y. Chiu, "Enhanced authentication scheme with anonymity for roaming service in global mobility networks," *Computer Communications,* vol. 32, no. 4, pp. 611–618, 2009.

[7] T. Zhou and J. Xu, "Provable secure authentication protocol with anonymity for roaming service in global mobility networks," *Computer Networks,* vol. 55, no. 1, pp. 205–213. 2011.

[8] P. Gope, T. Hwang, "Lightweight and energy-efficient mutual authentication and key agreement scheme with user anonymity for secure communication in global mobility networks," *IEEE Systems Journal,* vol. 10, no. 4, pp. 1370–1379, 2015.

[9] G. Xu, J. Liu, Y. Lu, X. Zeng, Y. Zhang, X. Li, "A novel efficient maka protocol with desynchronization for anonymous roaming service in global mobility networks," *Journal of Network and Computer Applications,* vol. 107, pp. 83–92, 2018.

[10] R. Shashidhara, S. Bojjagani, A. K. Maurya, S. Kumari, H. Xiong, "A robust user authentication protocol with privacy-preserving for roaming service in mobility environments," *Peer-to-Peer Networking and Applications,* vol. 13, no. 6, pp. 1943—1966, 2020.

[11] AM. Rahmani, M. Mohammadi, J. Lansky, S. Mildeova, M. Safkhani, S. Kumari, S Karim, M. Hosseinzadeh, "AMAPG: Advanced Mobile Authentication Protocol for GLOMONET," *IEEE Access,* 2021.

[12] D. Kang, H. Lee, Y. Lee, D. Won, "Lightweight user authentication scheme for roaming service in GLOMONET with privacy preserving," *PLOS ONE,* vol. 16, no. 2, e0247441, 2021.

[13] R. Madhusudhan, R. Shashidhara, "Mobile user authentication protocol with privacy preserving for roaming service in GLOMONET," *Peer-to-Peer Networking and Applications,* vol. 13, no. 1, pp. 82–103, 2020.

[14] M. Nikooghadam, H. Amintoosi, S. Kumari, "A provably secure ECC-based roaming authentication scheme for global mobility networks," *Journal of Information Security and Applications,* vol. 54, 102588, 2020.

[15] J. Ryu, H. Lee, H. Kim, D. Won, "Secure and efficient three-factor protocol for wireless sensor networks," *Sensors,* vol. 18, no. 12, 2018.

[16] J. Ryu, T. Song, J. Moon, H. Kim, D. Won, "Cryptanalysis of improved and provably secure three-factor user authentication scheme for wireless sensor networks," *Computational science and technology,* pp. 49–58, 2019.

[17] T. Song, D. Kang, J. Ryu, H. Kim, D. Won, "Cryptanalysis and improvement of an ECC-based authentication protocol for wireless sensor networks," *International Conference on Computational Science and Its Applications,* pp. 50–61, 2018.

[18] H. Lee, J. Ryu, Y. Lee, D. Won, "Security Analysis of Blockchain-based User Authentication for Smart Grid Edge Computing Infrastructure," *International Conference on Ubiquitous Information Management and Communication (IMCOM),* pp. 1–4, 2021.

[19] J. Ryu, D. Kang, H. Lee, H. Kim, D. Won, "A Secure and Lightweight Three-Factor-Based Authentication Scheme for Smart Healthcare Systems," *Sensors,* vol. 20, no. 24, 7136, 2020.

[20] Y. Dodis, L. Reyzin, A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *International conference on the theory and applications of cryptographic techniques,* pp. 523–540, 2004.

[21] J. Jung, D. Kang, D. Lee, D. Won, "An improved and secure anonymous biometric-based user authentication with key agreement scheme for the integrated EPR information system," *PLOS ONE* vol. 12, no. 1, e0169414, 2017.

[22] K. Niinuma, U. Park, AK. Jain, "Soft biometric traits for continuous user authentication," *IEEE Transactions on information forensics and security,* vol. 5, no. 4, pp. 771–780, 2010.

[23] M. Alotaibi, "An enhanced symmetric cryptosystem and biometric-based anonymous user authentication and session key establishment scheme for WSN," *IEEE Access,* vol. 6, pp. 70072–70087, 2018.

[24] Y. Wang, C. Wu, K. Zheng, X. Wang, "Improving reliability: User authentication on smartphones using keystroke biometrics," *IEEE Access,* vol. 7, pp. 26218–26228, 2019.

[25] Z. Rui, Z. Yan, "A survey on biometric authentication: Toward secure and privacy-preserving identification," *IEEE Access,* vol. 7, pp. 5994–6009, 2018.

[26] M. Devi, A. Majumder, "Side-channel attack in internet of things: a survey," *Applications of Internet of Things,* pp. 213–222, 2021.

[27] J. Ryu, H. Kim, Y. Lee, D. Won, "Cryptanalysis of Protocol for Heterogeneous Wireless Sensor Networks for the Internet of Things Environment," *International Conference on Ubiquitous Information Management and Communication (IMCOM),* pp. 1–4, 2020.

[28] S. Banerjee, V. Odelu, AK. Das, S. Chattopadhyay; J. Rodrigues, Y. Park, "Physically secure lightweight anonymous user authentication protocol for internet of things using physically unclonable functions," *IEEE Access,* vol. 7, pp. 85627–85644, 2019.

[29] D. Dolev, AC. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory,* vol. 29, no. 2, pp. 198–208, 1983.

[30] S. Roy, S. Chatterjee, AK. Das, S. Chattopadhyay, S. Jo, "Chaotic map-based anonymous user authentication scheme with user biometrics and fuzzy extractor for crowdsourcing Internet of Thing," *IEEE Internet of Things Journal,* vol. 5, no. 4, pp. 2884–2895, 2017.

[31] S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, N. Kumar, A. Vasilakos, "On the design of provably secure lightweight remote user authentication scheme for mobile cloud computing services," *IEEE Access,* vol. 5, pp. 25808–25825, 2017.

[32] M. Burrows, M. Abadi, R. M. Needham, "A logic of authentication," *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences,* vol. 426, no. 1871, pp. 233–271, 1989.

[33] M. Bellare, P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," *In Proceedings of the ACM conference on Computers and Communication Security,* pp. 62–73, 1993.

[34] Y.H. Chuang, C.L. Lei, and H.J. Shiu, "How to design a secure anonymous authentication and key agreement protocol for multi-server environments and prove its security," *Symmetry,* vol. 13, no. 9, 2021.

[35] W. Liu, X. Wang, W. Peng, "Secure remote multi-factor authentication scheme based on chaotic map zero-knowledge proof for crowdsourcing internet of things," *IEEE Access,* vol. 8, pp. 8754–8767, 2019.

[36] SY. Chiou, Z. Ying, J. Liu, "Improvement of a privacy authentication scheme based on cloud for medical environment," *Journal of medical systems,* vol. 40, no. 4, 101, 2016.

[37] AK. Maurya, VN. Sastry, "Fuzzy Extractor and Elliptic Curve Based Efficient User Authentication Protocol for Wireless Sensor Networks and Internet of Things," *Information,* vol. 8, no. 4, 136, 2017.

[38] Y. Dodis, L. Reyzin, A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *International conference on the theory and applications of cryptographic techniques (EUROCRYPT),* pp. 523—540, 2004.

JIHYEON RYU received the B.S. degree in Mathematic and Computer Science from Sungkyunkwan University, Korea, in 2018. She is currently undertaking a Ph.D. course on Software of Department in Sungkyunkwan University. Her current research interests include Cyber Security, Machine Learning, and User Authentication.

HAKJUN LEE received the B.S. degree in Software Engineering from Korea National University of Transportation, Korea, in 2015. He received M.S. and is currently undertaking a Ph.D in at Electrical and Computer Engineering from Sungkyunkwan University. He is also currently an assistant professor at Cyber Security in Howon University. His current research interest is in the area of cryptography, authentication protocol, and blockchain.

YOUNGSOOK LEE received B.S., M.S. and Ph.D. in Computer Science from Sungkyunkwan University, South Korea. She is currently a professor at ITSoftwareSecurity in Howon University. Her research interests are cryptology and informaion security.

DONGHO WON received B.S., M.S. and Ph.D. in Electronic Engineering from Sungkyunkwan University, South Korea. After working in Electronics and Telecommunication Research Institute for two years, he joined Sungkyunkwan University. He also served as a President of Korea Institute of Information Security and Cryptography. His research interests are cryptology and information security.