IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 3, 2017, accepted November 10, 2017, date of publication November 14, 2017,
date of current version December 5, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2773535

Provably Secure and Efficient Authentication Protocol for Roaming Service in Global Mobility Networks

KISUNG PARK[1], YOUNGHO PARK[1], (Member, IEEE), YOHAN PARK[2], ALAVALAPATI GOUTHAM REDDY[3], (Member, IEEE), AND ASHOK KUMAR DAS[4], (Member, IEEE)

[1]School of Electronics Engineering, Kyungpook National University, Daegu 41566, South Korea
[2]Division of IT Convergence, Korea Nazarene University, Cheonan 31172, South Korea
[3]Department of Computer and Information Security, Sejong University, Seoul 05006, South Korea
[4]Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India

Corresponding author: Youngho Park (parkyh@knu.ac.kr)

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Science, ICT and Future Planning under Grant 2017R1A2B1002147 and in part by the BK21 Plus project funded by the Ministry of Education, Korea under Grant 21A20131600011.

ABSTRACT In global mobility networks, a mobile user can access roaming services using a mobile device at anytime and anywhere. However, mobile users can be vulnerable to various attacks by adversaries, because the roaming services are provided through public network. Therefore, an anonymous mobile user authentication for roaming services is an essential security issue in global mobility networks. Recently, Lee *et al.* pointed out the security weaknesses of a previous scheme and proposed an advanced secure anonymous authentication scheme for roaming services in global mobility networks. However, we found that the scheme proposed by Lee *et al.* is vulnerable to password guessing and user impersonation attacks, and that it cannot provide perfect forward secrecy and secure password altered phase. In this paper, to overcome the security weaknesses of the scheme proposed by Lee *et al.*, we propose an improved secure anonymous authentication scheme using shared secret keys between home agent and foreign agent. In addition, we analyze the security of our proposed scheme against various attacks and prove that it provides secure mutual authentication using Burrows–Abadi–Needham logic. In addition, the formal security analysis using the broadly-accepted real-or-random (ROR) random oracle model and the formal security verification using the widely accepted automated validation of the Internet security protocols and applications tool show that the proposed scheme provides the session key security and protection against replay as well as man-in-the-middle attacks, respectively. Finally, we compare the performance of the proposed scheme with the related schemes, and the results show that the proposed scheme provides better security and comparable efficiency as compared with those for the existing schemes.

INDEX TERMS Global mobility networks, authentication, roaming services, formal security, key agreement, ROR model, BAN logic, AVISPA simulation.

I. INTRODUCTION

Global mobility networks (GLOMONETs) are being expanded, the numerous mobile users can access roaming services using mobile devices at anytime and anywhere. However, because roaming services are provided through public networks, an adversary could modify, delete, replay, intercept or eavesdrop the messages transmitted in public network. For these reasons, a secure mutual authentication for roaming services has become a very important issue.

Nowadays, to ensure privacy of mobile user, many mobile user authentication schemes have been proposed for roaming services in GLOMONETs [1]–[11]. Unfortunately, numerous authentication schemes are vulnerable to various attacks such as man-in-the-middle attack, perfect forward/backward secrecy, replay attack and impersonation attack. To resolve these security weaknesses, in 2004, Zhu and Ma [7] proposed an authentication scheme with anonymity for wireless environments. However, Lee *et al.* [8] found that Zhu and Ma's

authentication scheme cannot achieve backward secrecy and mutual authentication, and cannot withstand forgery attacks. They also proposed a security enhancement on a authentication scheme with anonymity for wireless environments. In 2008, Wu *et al.* [9] showed that Lee *et al.*'s scheme cannot achieve anonymity and perfect backward secrecy, and proposed an improved authentication scheme. However, in 2012, Mun *et al.* [6] claimed that Wu *et al.*'s scheme cannot achieve perfect forward secrecy and discloses the passwords of legitimate users under password guessing attack. Then, they proposed an enhanced secure anonymous authentication scheme. Recently, in 2017, Lee *et al.* [5] showed that Mun *et al.*'s scheme fails to realize real anonymity and perfect forward secrecy, and is vulnerable to masquerading attack and man-in-the-middle attack. They also proposed advanced secure anonymous authentication scheme for roaming service in GLOMONETs.

1) THREAT MODEL

The Dolev-Yao threat (DY) model [12] is widely used in evaluating the security of a protocol. Under the DY model, any two entities can communicate over a public channel. This gives an adversary an opportunity to eavesdrop (read), modify or delete the content of the messages being transmitted over the channel. We also assume that an adversary can have a lost or stolen mobile device, and can then extract all the sensitive information stored in that device using the power analysis attacks [13], [14]. In addition, an adversary can perform various attacks including offline password guessing attack, user impersonation attack, relay attack, man-in-the-middle attack and privileged-insider attack.

2) RESEARCH CONTRIBUTIONS

The contributions made in the paper are listed below:

- We analyze security weaknesses of Lee *et al.*'s scheme and demonstrate that it is vulnerable to password guessing and user impersonation attacks, and then we show that their scheme cannot provide perfect forward secrecy and secure password altered phase.
- To overcome these security weaknesses, we propose an enhanced secure anonymous authentication scheme for the global roaming service in GLOMONETs. The proposed scheme prevents various attacks such as password guessing attack, user impersonation attack and replay attack from malicious adversaries using a shared secret key between a foreign agent and home agent.
- The formal security analysis using the broadly-accepted Real-Or-Random (ROR) random oracle model proves that the proposed scheme provides the session key (SK) security.
- Moreover, our scheme provides secure mutual authentication and perfect forward secrecy, and we prove the secure mutual authentication of our scheme using the BAN logic.

TABLE 1. Notations.

Notation	Description
MU	Mobile user
FA	Foreign Agent
HA	Home agent
ID_X	Identity of an entity X
PW_{MU}	Password of MU
N_X	Random once generated by X
K_{XY}	Session key between X and Y
SK_{HA}	Secret key of HA
SK_{FA}	Secret key of FA
s, s_{new}	Random numbers generated by MU
k_{FA}	Shared secret key between FA and HA
$h(\cdot)$	Collision-resistant cryptographic one-way hash function
$\oplus$	Exclusive-OR operation
$\|\|$	Concatenation operation

- Finally, we compare the performance of our scheme with related schemes to show its security and efficiency.

3) PAPER STRUCTURE

The remainder of this paper is organized as follows. In Section II, we review the authentication scheme of Lee *et al.* In Section III, we cryptanalyze the scheme of Lee *et al.* In Section IV, we propose a secure anonymous protocol for roaming service in GLOMONETs to withstand the security pitfalls found in the authentication scheme of Lee *et al.*, and then we discuss the security of our proposed scheme in Section V. In Section VI, we compare the performance of the proposed scheme with the related existing schemes. Finally, we conclude the paper in Section VII.

II. REVIEW OF LEE *ET AL.*'s SCHEME

In this section, we review the Lee *et al.*'s advanced secure anonymous authentication scheme for roaming service in GLOMONETs. This scheme consists of four phases: registration, authentication and establishment of session key (AESK), session key update, and password altered. The notation used in this paper is defined in Table 1.

A. REGISTRATION PHASE

If a new mobile user MU wants to access the roaming service, MU must register with the Home agent HA firstly. The Lee *et al.*'s mobile user registration phase is shown Figure 1 and the detailed step of this registration phase as follows:

Step 1: MU chooses the password PW_{MU} and random number s, and then computes $EID = h(ID_{MU} \oplus PW_{MU}) \oplus s$ and sends EID to HA through the secure channel.

Step 2: After receiving EID from MU, HA computes $S = h(EID||h(SK_{HA}))$ and sends S to MU.

Step 3: MU computes $SPW = S \oplus h(PW_{MU})$ and stores SPW and s. Consequently, the mobile device contains $\{SPW, s\}$.

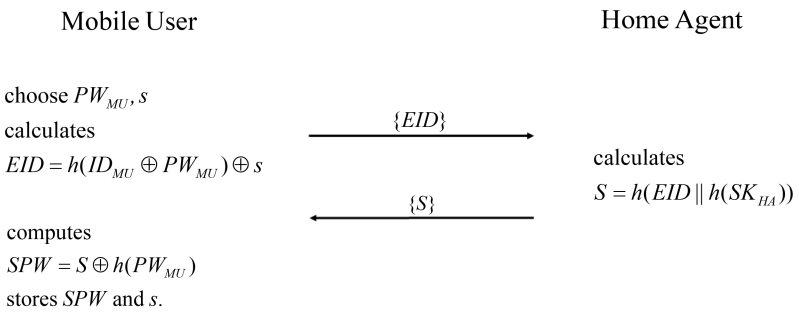

FIGURE 1. Mobile user registration phase of Lee *et al.*'s scheme.

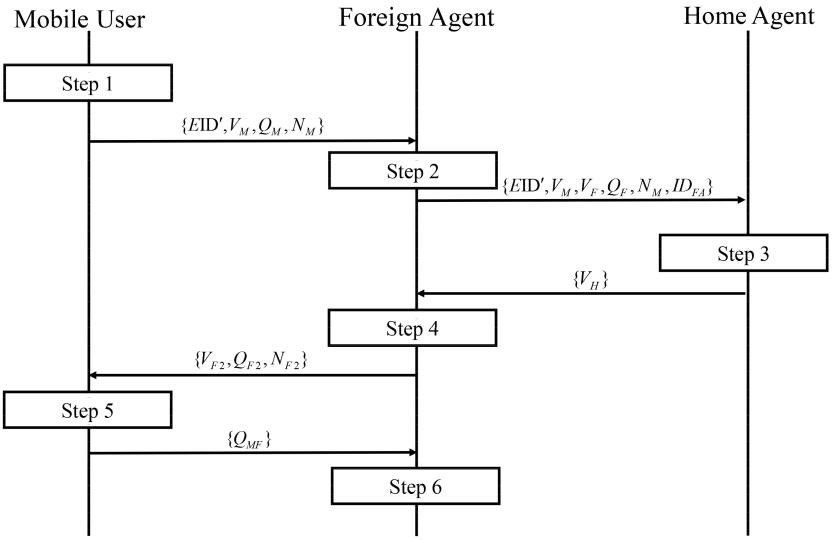

FIGURE 2. AESK phase of Lee *et al.*'s scheme.

B. AESK PHASE

When the mobile user MU wants to use a roaming service, MU sends request messages of accessing the roaming service to foreign agent FA. Lee *et al.*'s scheme also supposed that MU and FA must authenticate each other before sending the request message of roaming service to FA. The Lee *et al.*'s AESK phase is shown in Figure 2 and the detailed step of this AESK phase as follows:

Step 1: MU inputs password PW'_{MU} and selects two random numbers $\{s_{new}, N_M\}$. Then MU computes $EID' = h(ID_{MU} \oplus PW'_{MU}) \oplus s$, $S' = SPW \oplus h(PW'_{MU})$, $EID_{new} = h(ID_{MU} \oplus PW'_{MU}) \oplus s_{new}$, $V_M = EID_{new} \oplus h(S'||N_M)$ and $Q_M = h(EID_{new}||S'||N_M)$, and MU sends $\{EID', V_M, Q_M, N_M\}$ to FA.

Step 2: After receiving EID', V_M, Q_M, N_M from MU, FA chooses a random number N_F, and computes $Q_F = h(Q_M||N_F||SK_{FA})$ and $V_F = N_F \oplus h(SK_{FA})$. Finally, FA sends $\{EID', V_M, Q_F, N_M, V_F, ID_{FA}\}$ to HA.

Step 3: Upon receiving $\{EID', V_M, Q_F, N_M, V_F, ID_{FA}\}$ from FA, HA computes $S' = h(EID' ||h(SK_{HA}))$ and retrieves $EID'_{new} = V_M \oplus h(S' ||N_M)$. Then, HA computes $SK_{FA} = h(ID_{FA} \oplus SK_{HA})$ and $N_F = V_F \oplus h(SK_{FA})$, and HA checks whether $Q_F \stackrel{?}{=} h(h(EID'_{new} ||S' ||N_M) ||N_F ||SK_{FA})$. It it is equal, HA computes $S_{new} = h(EID_{new} ||h(SK_{HA}))$ and $V_H = (EID_{new} ||S|| S_{new}) \oplus h(SK_{FA} ||N_F)$. Finally, HA sends $\{V_H\}$ to FA.

Step 4: After receiving $\{V_H\}$ to HA, FA retrieves $(EID_{new}||S||S_{new}) = V_H \oplus h(SK_{FA}||N_F)$ and checks whether $Q_M \stackrel{?}{=} h(EID_{new}||S||N_M)$. If they are equal, FA chooses a random number N_{F2} and computes $V_{F2} = S_{new} \oplus h(S||N_{F2})$ and $Q_{F2} = h(EID||S_{new}||N_{F2})$. Finally, FA sends $\{V_{F2}, Q_{F2}, N_{F2}\}$ to MU.

Step 5: After receiving $\{V_{F2}, Q_{F2}, N_{F2}\}$ from FA, MU computes $S_{new} = V_{F2} \oplus h(S||N_{F2})$ and checks whether $Q_{F2} \stackrel{?}{=} h(EID||S_{new}||N_{F2})$. Then, MU computes $SPW_{new} = S_{new} \oplus h(PW_{MU})$ and stores $\{SPW_{new}, s_{new}\}$ into device. Finally, MU computes the session key $K_{MF} = h(N_M||N_{F2}||S)$ and sends $Q_{MF} = h(N_M||S||N_{F2}||S_{new})$ to FA.

Step 6: Upon receiving $\{Q_{MF}\}$ from MU, FA checks whether $Q_{MF} \stackrel{?}{=} h(N_M||S||N_{F2}||S_{new})$ and computes the session key $K_{MF} = h(N_M||N_{F2}||S)$. Finally, MU and FA achieve the session key agreement successfully.

C. SESSION KEY UPDATE PHASE

When mobile user MU wants to update the session key, MU can update the session key. The detailed step of this password change phase are follows:

Step 1: MU chooses a new random number N_M^*, and MU computes $U_M = N_M^* \oplus h(S||N_M||N_{F2})$ and $Q_M^* = h(N_M^* \oplus S)$. Then, MU sends U_M and Q_M^* to FA.

Step 2: After receiving $\{U_M, Q_M\}$ from MU, FA computes $N_M^* = U_M \oplus h(S||N_M||N_{F2})$ and checks whether $Q_M^* \stackrel{?}{=} h(N_M^* \oplus S)$. Then, FA chooses a new random number N_F^* and computes $U_F = N_F^* \oplus h(S||N_{F2}||N_M^*)$ and $Q_F^* = h(N_F^* \oplus S)$. After that, FA sends $\{U_F, Q_F^*\}$ to MU.

Step 3: After receiving $\{U_F, Q_F^*\}$ from FA, MU computes $N_F^* = U_F \oplus h(S||N_{F2}||N_M)$ and checks whether $Q_F^* \stackrel{?}{=} h(N_F^* \oplus S)$. If it is equal, MU updates the new session key K_{MF}^* from K_{MF}. After that, MU computes $Q_{MF}^* = h(N_M^* \oplus N_F^* \oplus S)$ and sends $\{Q_{MF}^*\}$ to FA.

Step 4: Upon receiving $\{Q_{MF}^*\}$, FA checks whether $Q_{MF}^* \stackrel{?}{=} h(N_M^* \oplus N_F^* \oplus S)$. If it is equal, FA updates the session key K_{MF}^* from K_{MF}. Finally, the session update phase is completed successfully.

D. PASSWORD ALTERED PHASE

When mobile user MU wants to alter his/her password, MU can alter his/her password freely. The procedure of password altered phase is similar to AESK phase. When MU computes EID_{new} and SPW_{new}, MU computes $EID_{new} = h(ID_{MU} \oplus PW_{new}) \oplus s$ and $SPW_{new} = S_{new} \oplus h(PW_{new})$ using new password PW_{new} instead of existing password PW_{MU}. Therefore, the password altered phase is completed successfully.

III. CRYPTANALYSIS OF LEE *ET AL.*'s SCHEME

In this section, we demonstrate that Lee *et al.*'s scheme cannot prevent user impersonation and password guessing attacks. We also show that their scheme cannot provide perfect forward secrecy, and an adversary can freely perform changing password. We assumed that an adversary MU_a could steal or obtain the MU's mobile device. In addition, an adversary MU_a could extract information $\{SPW, s\}$ from the device and could get previous session messages transmitted through public network. The description of the security weaknesses of Lee *et al.*'s scheme is as follows.

A. OFFLINE PASSWORD GUESSING ATTACK

If the adversary MU_a obtains SPW, MU_a can attempt to guess the password of MU, and then MU_a can guess password of MU successfully. The procedure of offline password guessing attack is as follows:

Step 1: The adversary MU_a chooses a new password PW_{MU}^* and computes $S^* = SPW \oplus h(PW_{MU}^*)$.

Step 2: The MU_a computes $S_{new}^* = V_{F2} \oplus h(S^*||N_{F2})$, where V_{F2} and N_{F2} are previous transmitted messages.

Step 3: The MU_a computes $Q_{MF}^* = h(N_M||S^*||N_{F2}||S_{new}^*)$, and then compares $Q_{MF}^* \stackrel{?}{=} Q_{MF}$, where Q_{MF} is previous transmitted messages. If it is equal, The adversary MU_a guesses password of MU correctly.

The result of this attack indicates that Lee *et al.*'s scheme is vulnerable to offline password guessing attack.

B. USER IMPERSONATION ATTACK

According to Section III-A, the adversary MU_a can obtain password of MU correctly. After that, we show that Lee *et al.*'s scheme cannot resist user impersonation attack using Section III-A. The detailed step of user impersonation attack is as follows.

Step 1: MU_a generates N_a and computes $V_a = EID' \oplus h(S_a||N_a)$ and $Q_a = h(EID'||S_a||N_a)$, where EID' and S_a obtained according to Section III-A. Then, MU_a sends $\{EID', V_M, Q_M, N_M\}$ to FA.

Step 2: After receiving $\{EID', V_a, Q_a, N_a\}$ from MU_a, FA chooses a random number N_F, and computes $Q_F = h(Q_a||N_F||SK_{FA})$ and $V_F = N_F \oplus h(SK_{FA})$. Finally, FA sends $\{EID', V_a, Q_F, N_a, V_F, ID_{FA}\}$ to HA.

Step 3: Upon receiving $\{EID', V_a, Q_F, N_a, V_F, ID_{FA}\}$ from FA, HA computes $S' = h(EID'||h(SK_{HA}))$ and retrieves $EID'_{new} = V_a \oplus h(S'||N_a)$. Then, HA computes $SK_{FA} = h(ID_{FA} \oplus SK_{HA})$ and $N_F = V_F \oplus h(SK_{FA})$, and HA checks whether $Q_F \stackrel{?}{=} h(h(EID'_{new}||S'||N_a)||N_F||SK_{FA})$. It it is equal, HA computes $S_{new} = h(EID_{new}||h(SK_{HA}))$ and $V_H = (EID_{new}||S||S_{new}) \oplus h(SK_{FA}||N_F)$. Finally, HA sends $\{V_H\}$ to FA.

Step 4: After receiving $\{V_H\}$ to HA, FA retrieves $(EID_{new}||S||S_{new}) = V_H \oplus h(SK_{FA}||N_F)$ and checks whether $Q_a \stackrel{?}{=} h(EID_{new}||S||N_a)$. If they are equal, FA chooses a random number N_{F2} and computes $V_{F2} = S_{new} \oplus h(S||N_{F2})$ and $Q_{F2} = h(EID||S_{new}||N_{F2})$. Finally, FA sends $\{V_{F2}, Q_{F2}, N_{F2}\}$ to MU_a.

Step 5: After receiving $\{V_{F2}, Q_{F2}, N_{F2}\}$ from FA, MU_a computes $S_{new} = V_{F2} \oplus h(S_a||N_{F2})$ and checks whether $Q_{F2} \stackrel{?}{=} h(EID||S_{new}||N_{F2})$. Then, MU computes $SPW_{new} = S_{new} \oplus h(PW_{MU})$. Finally, MU_a computes the session key $K_{MF} = h(N_a||N_{F2}||S)$ and sends $Q_{aF} = h(N_a||S||N_{F2}||S_{new})$ to FA.

Step 6: Upon receiving $\{Q_{aF}\}$ from MU_a, FA checks whether $Q_{aF} \stackrel{?}{=} h(N_a||S||N_{F2}||S_{new})$ and computes

the session key $K_{aF} = h(N_a||N_{F2}||S)$. Finally, MU_a and FA achieve the session key agreement successfully.

Therefore, Lee *et al.*'s scheme cannot resist user impersonation attack.

C. PERFECT FORWARD SECRECY

We assume that MU_a intercepts and stores messages transmitted in the previous session, and SK_{FA} is compromised by MU_a. In Lee *et al.*'s scheme, the session key $K_{MF} = h(N_M||N_F||S)$ and MU_a can computes previous session key. First, MU_a computes $N_F = V_F \oplus h(SK_{FA})$, $(EID_{new}||S||S_{new}) = V_H \oplus h(SK_{FA}||N_F)$. After that, MU_a can computes the session key $K_{MF} = h(N_M||N_{F2}||S)$, where N_{F2} is transmitted message in previous session. Therefore, Lee *et al.*'s scheme cannot provide perfect forward secrecy.

D. FLAW OF PASSWORD ALTERED PHASE

In Lee *et al.*'s scheme, when an adversary MU_a wants to change the password of MU, MU_a can change the password. According to Sections II-D and III-B, in AESK phase, MU_a can replace the values $EID_a = h(ID_a \oplus PW_a) \oplus S_a$ and $SPW_a = S_{new} \oplus h(PW_a)$ instead of EID_{new} and SPW_{new}. Therefore, since MU_a can change the password of MU, Lee *et al.*'s scheme has flaw of password altered phase.

E. LACK OF MUTUAL AUTHENTICATION

According to Section III-B, an adversary MU_a can impersonate as mobile user MU. MU_a also can achieve authentication between MU_a and FA. For this reason, Lee et al's scheme cannot provide secure mutual authentication.

IV. THE PROPOSED SCHEME

In this section, to resolve the security weaknesses of Lee *et al.*'s scheme, we propose an improved secure anonymous authentication protocol for roaming service in GLOMONETs. Our scheme consists of five phases: foreign agent (FA) registration, mobile user (MU) registration, authentication and establishment of session key (AESK), session key update and password altered phase.

A. REGISTRATION PHASE

The registration phase of the proposed scheme is composed of the following two registration phases of the foreign agent (FA) and the mobile user (MU).

1) FOREIGN AGENT REGISTRATION PHASE

When the foreign agent FA wants to provide roaming service with home agent HA, FA sends the agent registration request message to HA in advance. The foreign agent registration phase of the proposed scheme is shown in Figure 3 and the detailed steps of this phase are as follows.

Step 1: A foreign agent FA chooses a random number b and computes $k_{FA} = h(ID_{FA}||b)$, and then sends $\{ID_{FA}, k_{FA}\}$ to HA.

Foreign Agent (FA)	Home Agent (HA)
chooses ID_{FA} and b	
computes $k_{FA} = h(ID_{FA}\|\|b)$	calculates
$\{ID_{FA}, k_{FA}\}$ →	$SK_{FA} = h(ID_{FA} \oplus SK_{HA})$
	stores k_{FA} with SK_{FA} in database
	$\{SK_{FA}\}$ ←
stores $\{ID_{FA}, k_{FA}, SK_{FA}\}$ in database	

FIGURE 3. Foreign agent registration phase of the proposed scheme.

Mobile User (MU)	Home Agent (HA)
chooses identity ID_{MU}, password PW_{MU} and a random number s	
computes $EID = h(ID_{MU} \oplus PW_{MU}) \oplus s$	
$\{EID\}$ →	calculates
	$S = h(EID\|\|h(SK_{HA}))$
	$\{S\}$ ←
computes $SPW = S \oplus h(PW_{MU}) \oplus h(ID_{MU}\|\|s)$ and $s' = s \oplus h(ID_{MU}\|\|PW_{MU})$ stores SPW and s' in mobile device	

FIGURE 4. Mobile user registration phase of the proposed scheme.

Step 2: After receiving $\{ID_{FA}, k_{FA}\}$ from FA, HA computes $SK_{FA} = h(ID_{FA} \oplus SK_{HA})$. After that, HA stores k_{FA} with SK_{FA} in a database and sends to SK_{FA} to FA.

Step 3: FA stores $\{ID_{FA}, k_{FA}, SK_{FA}\}$ in its database.

2) MOBILE USER REGISTRATION PHASE

When a new mobile user MU wants to use the roaming service, MU must register with the Home agent HA in advance. The proposed scheme's mobile user registration phase is shown Figure 4 and the detailed steps of this phase are as follows.

Step 1: MU chooses the identity ID_{MU}, password PW_{MU} and a random number s, and then computes $EID = h(ID_{MU} \oplus PW_{MU}) \oplus s$ and sends EID to HA through the secure channel.

Step 2: After receiving EID from MU, HA computes $S = h(EID||h(SK_{HA}))$ and sends S to MU.

Step 3: After receiving S from HA, MU computes $SPW = S \oplus h(PW_{MU}) \oplus h(ID_{MU}||s)$ and $s' = s \oplus h(ID_{MU}||PW_{MU})$, and stores SPW and s'. Consequently, the mobile device contains $\{SPW, s'\}$.

B. AESK PHASE

If the mobile user MU wants to access a roaming service, MU sends request message of accessing the roaming service to foreign agent FA. AESK phase of the proposed scheme is shown in Figure 5 and the detailed steps of this AESK phase are as follows.

Step 1: MU inputs identity ID_{MU} and password PW'_{MU}, and selects two random numbers s_{new} and N_M. Then, MU computes $s = s' \oplus H(ID_{MU}||PW'_{MU})$,

Mobile User (MU)	Foreign Agent (FA)	Home Agent (HA)
inputs identity ID_{MU} and password PW'_{MU} selects two random numbers s_{new} and N_M computes $s = s' \oplus H(ID_{MU}\|\|PW'_{MU})$, $EID' = h(ID_{MU} \oplus PW'_{MU}) \oplus s$, $S' = SPW \oplus h(PW'_{MU}) \oplus h(ID_{MU}\|\|s)$, $EID_{new} = h(ID_{MU} \oplus PW'_{MU}) \oplus s_{new}$, $V_M = EID_{new} \oplus h(S'\|\|N_M)$, $s'_{new} = s_{new} \oplus h(ID_{MU}\|\|PW'_{MU})$, $Q_M = h(EID_{new} \|\|S' \|\|N_M)$ $\{EID', V_M, Q_M, N_M\}$ →		
	chooses a random number N_F computes $Q_F = h(Q_M\|\|N_F\|\|SK_{FA})$, $V_F = N_F \oplus h(SK_{FA}) \oplus h(k_{FA}\|\|N_M)$ $\{EID', V_M, Q_F, N_M, V_F, ID_{FA}\}$ →	
		computes $S' = h(EID' \|\|h(SK_{HA}))$ retrieves $EID'_{new} = V_M \oplus h(S' \|\|N_M)$, $SK_{FA} = h(ID_{FA} \oplus SK_{HA})$, $N_F = V_F \oplus h(SK_{FA}) \oplus h(k_{FA} \|\|N_M)$ checks $Q_F \stackrel{?}{=} h(h(EID'_{new}$ $\|\|S' \|\|N_M) \|\|N_F \|\|SK_{FA})$ It it is equal, computes $S_{new} = h(EID_{new} \|\|h(SK_{HA}))$, $V_H = (EID_{new} \|\|S \|\|S_{new})$ $\oplus h(SK_{FA} \|\|N_F)$ $\{V_H\}$ ←
	retrieves $(EID_{new}\|\|S\|\|S_{new}) =$ $V_H \oplus h(SK_{FA}\|\|N_F)$ checks $Q_M \stackrel{?}{=} h(EID_{new}\|\|S\|\|N_M)$ If they are equal, chooses random number N_{F2} and computes $V_{F2} = S_{new} \oplus h(S\|\|N_{F2})$, $Q_{F2} = h(EID\|\|S_{new}\|\|N_{F2})$ $\{V_{F2}, Q_{F2}, N_{F2}\}$ ←	
computes S_{new}=$V_{F2} \oplus h(S\|\|N_{F2})$ checks $Q_{F2} \stackrel{?}{=} h(EID\|\|S_{new}\|\|N_{F2})$ computes $SPW_{new} = S_{new} \oplus h(PW_{MU})$ $\oplus h(ID_{MU}\|\|s)$ stores SPW_{new} and s'_{new} into device computes session key $K_{MF} = h(N_M\|\|N_{F2}\|\|S)$ $\{Q_{MF}\}$ →		
	checks $Q_{MF} \stackrel{?}{=} h(N_M\|\|S\|\|N_{F2}\|\|S_{new})$ computes session key $K_{MF} = h(N_M\|\|N_{F2}\|\|S)$	
stores session key K_{MF} shared with FA	stores session key K_{MF} shared with MU	

FIGURE 5. AESK of the proposed scheme.

$EID' = h(ID_{MU} \oplus PW'_{MU}) \oplus s$, $S' = SPW \oplus h(PW'_{MU}) \oplus h(ID_{MU}||s)$, $EID_{new} = h(ID_{MU} \oplus PW'_{MU}) \oplus s_{new}$, $V_M = EID_{new} \oplus h(S' ||N_M)$, $s'_{new} = s_{new} \oplus h(ID_{MU} ||PW'_{MU})$ and $Q_M = h(EID_{new} ||S' ||N_M)$, and *MU* sends $\{EID', V_M, Q_M, N_M\}$ to *FA*.

Step 2: After receiving $\{EID', V_M, Q_M, N_M\}$ from *MU*, *FA* chooses a random number N_F, and computes $Q_F = h(Q_M||N_F||SK_{FA})$ and $V_F = N_F \oplus h(SK_{FA}) \oplus h(k_{FA}||N_M)$. Finally, *FA* sends $\{EID', V_M, Q_F, N_M, V_F, ID_{FA}\}$ to *HA*.

Step 3: Upon receiving $\{EID', V_M, Q_F, N_M, V_F, ID_{FA}\}$ from *FA*, *HA* computes $S' = h(EID' ||h(SK_{HA}))$ and retrieves $EID'_{new} = V_M \oplus h(S' ||N_M)$. Then, *HA* computes $SK_{FA} = h(ID_{FA} \oplus SK_{HA})$ and retrieves k_{FA} in database. *HA* also computes $N_F = V_F \oplus h(SK_{FA}) \oplus h(k_{FA} ||N_M)$, and *HA* checks whether $Q_F \stackrel{?}{=} h(h(EID'_{new} ||S' ||N_M) ||N_F ||SK_{FA})$. It it is equal, *HA* computes $S_{new} = h(EID_{new} ||h(SK_{HA}))$ and $V_H = (EID_{new} ||S ||S_{new}) \oplus h(SK_{FA} ||N_F)$. Finally, *HA* sends $\{V_H\}$ to *FA*.

Step 4: After receiving $\{V_H\}$ to *HA*, *FA* retrieves $(EID_{new}||S||S_{new}) = V_H \oplus h(SK_{FA}||N_F)$ and checks whether $Q_M \stackrel{?}{=} h(EID_{new}||S||N_M)$. If they are equal, *FA* chooses a random number N_{F2} and computes $V_{F2} = S_{new} \oplus h(S||N_{F2})$

and $Q_{F2} = h(EID||S_{new}||N_{F2})$. Finally, *FA* sends $\{V_{F2}, Q_{F2}, N_{F2}\}$ to *MU*.

Step 5: After receiving $\{V_{F2}, Q_{F2}, N_{F2}\}$ from *FA*, *MU* computes S_{new}=$V_{F2} \oplus h(S||N_{F2})$ and checks whether $Q_{F2} \stackrel{?}{=} h(EID||S_{new}||N_{F2})$. Then, *MU* computes $SPW_{new} = S_{new} \oplus h(PW_{MU}) \oplus h(ID_{MU}||s)$, and stores SPW_{new} and s'_{new} into the mobile device. Finally, *MU* computes the session key $K_{MF} = h(N_M||N_{F2}||S)$ and sends $Q_{MF} = h(N_M||S||N_{F2}||S_{new})$ to *FA*.

Step 6: Upon receiving $\{Q_{MF}\}$ from *MU*, *FA* checks whether $Q_{MF} \stackrel{?}{=} h(N_M||S||N_{F2}||S_{new})$ and computes the session key $K_{MF} = h(N_M||N_{F2}||S)$. Finally, *MU* and *FA* achieve the session key agreement successfully.

C. SESSION KEY UPDATE PHASE

If a mobile user *MU* wants to update the session key, *MU* can try to update the session key as follows.

Step 1: *MU* chooses a new random number N^*_M, and *MU* computes $U_M = N^*_M \oplus h(S||N_M||N_{F2})$ and $Q^*_M = h(N^*_M \oplus S)$. Then, *MU* sends U_M and Q^*_M to *FA*.

Step 2: After receiving $\{U_M, Q_M\}$ from *MU*, *FA* computes $N^*_M = U_M \oplus h(S||N_M||N_{F2})$ and checks whether $Q^*_M \stackrel{?}{=} h(N^*_M \oplus S)$. Then, *FA* chooses a new random number N^*_F and computes $U_F = N^*_F \oplus h(S||N_{F2}||N^*_M)$ and $Q^*_F = h(N^*_F \oplus S)$. After that, *FA* sends $\{U_F, Q^*_F\}$ to *MU*.

Step 3: After receiving $\{U_F, Q^*_F\}$ from *FA*, *MU* computes $N^*_F = U_F \oplus h(S||N_{F2}||N_M)$ and checks whether $Q^*_F \stackrel{?}{=} h(N^*_F \oplus S)$. If it is equal, *MU* updates the new session key K^*_{MF} from K_{MF}. After that, *MU* computes $Q^*_{MF} = h(N^*_M \oplus N^*_F \oplus S)$ and sends $\{Q^*_{MF}\}$ to *FA*.

Step 4: Upon receiving $\{Q^*_{MF}\}$, *FA* checks whether $Q^*_{MF} \stackrel{?}{=} h(N^*_M \oplus N^*_F \oplus S)$. If it is equal, *FA* updates the session key K^*_{MF} from K_{MF}. Finally, the session update phase is completed successfully.

Session key update phase of the proposed scheme is briefed in Figure 6.

D. PASSWORD ALTERED PHASE

In our scheme, when mobile user *MU* wants to alter his/her password, *MU* can alter his/her password freely. This phase has the following steps:

Step 1: *MU* enters the existing identity ID_{MU} and old password PW_{MU}, and also the new password PW_{new}.

Step 2: *MU* computes $s = s' \oplus h(ID_{MU}||PW_{MU})$ and $EID_{new} = h(ID_{MU} \oplus PW_{new}) \oplus s$ using new password PW_{new} instead of existing password PW_{MU}. *MU* then sends EID_{new} to *HA* securely.

Step 3: After receiving EID_{new} from *MU*, *HA* computes $S_{new} = h(EID_{new}||h(SK_{HA}))$, and sends S_{new} to *MU* securely.

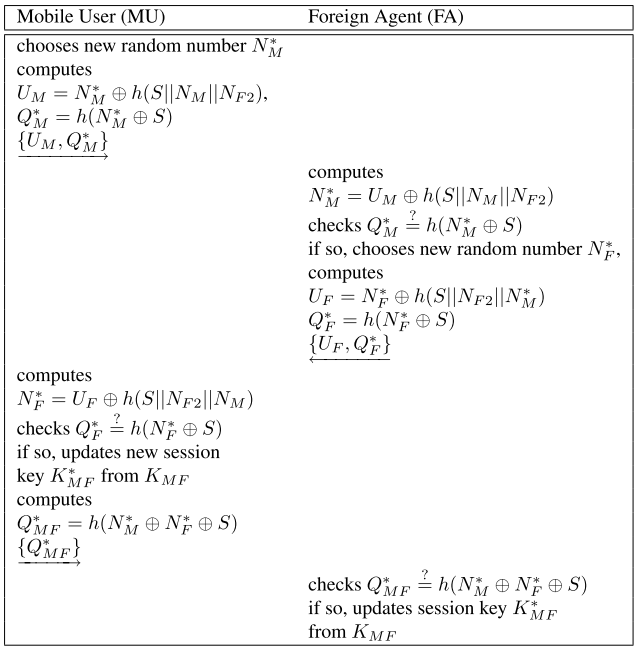

FIGURE 6. Session key update phase of the proposed scheme.

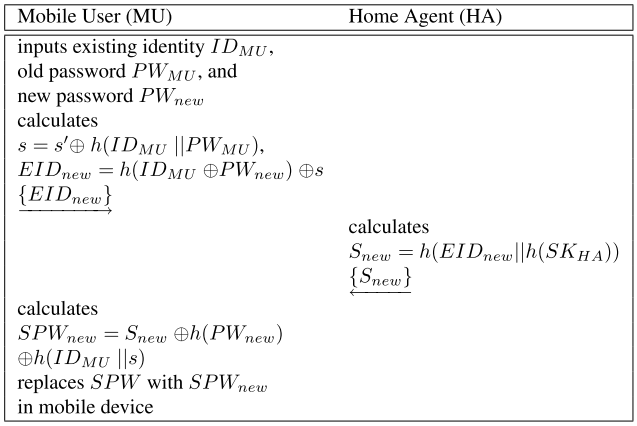

FIGURE 7. Password altered phase of the proposed scheme.

Step 4: *MU* then computes $SPW_{new} = S_{new} \oplus h(PW_{new}) \oplus h(ID_{MU}||s)$ using new password PW_{new} instead of existing password PW_{MU}. *MU* then replaces *SPW* with SPW_{new} in the mobile device. Finally, the password altered phase is completed successfully.

Password altered phase of the proposed scheme is summarized in Figure 7.

V. SECURITY ANALYSIS

In this section, to prove the security of our proposed scheme, we perform the formal security analysis using the broadly-accepted Real-Or-Random (ROR) model [15], mutual authentication proof using the the widely-accepted BAN logic [16] and also formal security verification using the broadly-used Automated Validation of Internet Security Protocols and Applications (AVISPA) simulation [17].

A. FORMAL SECURITY USING ROR MODEL

This section proves the session key (SK) security of the proposed scheme using the widely-accepted Real-Or-Random (ROR) model [15].

1) ROR MODEL

In the AESK phase of the proposed scheme. we have three entities, namely, *MU*, *FA* and *HA*. The brief discussion on the ROR model is given below.

a: PARTICIPANTS

Let $\mathcal{I}_{MU}^{t_1}$, $\mathcal{I}_{FA}^{t_2}$ and $\mathcal{I}_{HA}^{t_3}$ denote the instances t_1, t_2 and t_3 of *MU*, *FA* and *HA*, respectively. These are called *oracles*.

b: ACCEPTED STATE

In an instance $\mathcal{I}^t$ goes to an accept state after receiving the final protocol message, it is said to be that it enters in accepted state. The ordered concatenation of all communications (send and received messages by $\mathcal{I}^t$) is called the session identification (*sid*) of $\mathcal{I}^t$ for current session.

c: PARTNERING

Let $\mathcal{I}^{t_1}$ and $\mathcal{I}^{t_2}$ be two instances. They are called partners to each other, if the following three conditions hold simultaneously: 1) both are in accepted state, 2) both mutually authenticate each other and share the same *sid*, and 3) both are mutual partners of each other.

d: FRESHNESS

If the session key K_{MF} between *MU* and *FA* is not revealed through a reveal query $\mathcal{R}$ defined below, $\mathcal{I}_{MU}^{t_1}$ or $\mathcal{I}_{FA}^{t_2}$ is said to be fresh.

e: ADVERSARY

An adversary $\mathcal{A}$ is modeled using the Dolev-Yao (DY) model as explained in the threat model (Section I-.1). $\mathcal{A}$ can eavesdrop, modify, delete or inject the messages transmitted between the entities involved during the communication with the help of the following defined queries:

Execute$(\mathcal{I}^{t_1}, \mathcal{I}^{t_2})$: The eavesdropping attack is modeled in this *execute* query. It allows $\mathcal{A}$ to eavesdrop the messages communicated among *MU*, *FA* and *HA*.

Send$(\mathcal{I}^t, msg)$: This is the *send* query under which $\mathcal{A}$ transmits a message to $\mathcal{I}^t$, and in response, it also receives the message from $\mathcal{I}^t$. It is further modeled as an active attack.

Reveal$(\mathcal{I}^t)$: This is the *reveal* query that reveals the session key K_{MF} created by $\mathcal{I}^t$ (and its partner) to $\mathcal{A}$ in the current session.

CMD$(\mathcal{I}_{MU}^t)$: Under this *corrupt mobile device* query, $\mathcal{A}$ can fetch all the sensitive secret credentials stored from the lost or stolen mobile device's memory using the power analysis attack [13], [14]. This is also modeled as an active attack.

Test$(\mathcal{I}^t)$: Before the game begins, an unbiased coin c is flipped. Based on the output, the following decision is taken. $\mathcal{A}$ executes this *test* query and if the session key K_{MF} between *MU* and *FA* is fresh, $\mathcal{I}^t$ returns K_{MF} if $c = 1$ or a random number if $c = 0$; otherwise, it returns a null value ($\perp$).

In our formal security analysis, we impose a restriction that $\mathcal{A}$ can access only a limited number of *CMD*$(\mathcal{I}_{MU}^t)$ queries, while an unlimited number of *Test*$(\mathcal{I}^t)$ queries are accessible by $\mathcal{A}$.

f: SEMANTIC SECURITY

Under the semantic security, the indistinguishability of the real session key K_{MF} from a random number by an adversary $\mathcal{A}$ is necessary. The query *Test*$(\mathcal{I}^t)$'s outputs needs to be consistent with the random bit c. If $\mathcal{A}$'s guessed bit is c' and *Succ* is the winning probability in the game, $\mathcal{A}$'s advantage in breaking the session key (SK) security of the proposed scheme, say $\mathcal{P}$ is denoted and defined by $Adv_{\mathcal{P}} = 2|Pr[Succ] - 1|$, where $Pr[.]$ denotes the probability.

g: RANDOM ORACLE

The proposed scheme makes use of the one-way cryptographic hash function $h(\cdot)$. $h(\cdot)$ is modeled as a random oracle, say *Hash* and $h(\cdot)$ is public.

2) SECURITY PROOF

Wang *et al.* [18] mentioned that the user-chosen passwords follow the Zipf's law that is a vastly different distribution from the uniform distribution. The size of password dictionary is generally much constrained in the sense that the users will not use the whole space of passwords, but rather a small space of the allowed characters space [18]. We use the Zipf's law in proving the SK security of the proposed scheme, which is provided in Theorem 1.

Theorem 1: If $Adv_{\mathcal{P}}$ denotes the advantage function of an adversary $\mathcal{A}$ in breaking the SK security of the proposed scheme $\mathcal{P}$, then

$$Adv_{\mathcal{P}} \leq \frac{q_h^2}{|Hash|} + 2C' \cdot q_{send}^{s'}$$

where q_h, q_{send} and $|Hash|$ are the number of *Hash* queries, the number of *Send* queries and the range space of the hash function $h(\cdot)$, respectively, and C' and s' are the Zipf's parameters [18].

Proof: We follow the similar proof as applied in [19]–[22]. We define a sequence of four games, say GM_i for $i = 0, 1, 2, 3$. Let $Succ_i$ be the probability associated with the game GM_i in which an adversary $\mathcal{A}$ wins the game GM_i. The detailed discussion of these four games is given below.

- *Game* GM_0: This is the initial game in which $\mathcal{A}$ chooses the bit c. Since GM_0 and the real protocol in the ROR model are identical to each other, we have,

$$Adv_{\mathcal{P}} = |2Pr[Succ_0] - 1|. \quad (1)$$

- *Game* GM_1: Under this game, the eavesdropping attack is implemented by $\mathcal{A}$. $\mathcal{A}$ makes *Execute* query, and at the end of the game, $\mathcal{A}$ makes *Test* query. The output of the *Test* query decides if $\mathcal{A}$ obtains the real session key K_{MF} or a random number. In the proposed scheme, the session key K_{MF} is computed by both *MU* and *FA* as

$K_{MF} = h(N_M \,||N_{F2}\,||S)$, where $S = h(EID\,||h(SK_{HA}))$. To derive K_{MF}, $\mathcal{A}$ requires both temporal secrets N_M and N_{F2}, and also the permanent secrets SK_{HA} and EID. Hence, winning the game GM_1 by $\mathcal{A}$ is not increased by eavesdropping attack. Since both the game $Game_0$ and $Game_1$ are indistinguishable, it follows that

$$Pr[Succ_1] = Pr[Succ_0]. \tag{2}$$

- *Game* GM_2: Under this game, the simulations of *Send* and *Hash* queries are added so that GM_1 in converted to GM_2. This game is also modeled as an active attack wherein $\mathcal{A}$ eavesdrops all the messages $\{EID', V_M, Q_M, N_M\}$, $\{EID', V_M, Q_F, N_M, V_F, ID_{FA}\}$, $\{V_H\}$, $\{V_{F2}, Q_{F2}, N_{F2}\}$ and $\{Q_{MF}\}$. It is worth noticing that the messages involve the random nonces. Therefore, no collision in hash outputs (message digests) occurs when $\mathcal{A}$ makes the *Hash* queries. Using the birthday paradox result, we have,

$$|Pr[Succ_2] - Pr[Succ_1]| \leq \frac{q_h^2}{2|Hash|}. \tag{3}$$

- *Game* GM_3: It is the final game, which implements the $CMD(\mathcal{I}_{MU}^t)$ query wherein $\mathcal{A}$ can extract all the information $\{SPW, s'\}$ from the lost or stolen device of MU. Note that $SPW = S \oplus h(PW_{MU}) \oplus h(ID_{MU}||s)$, $S = h(EID\,||h(SK_{HA}))$ and $s' = s \oplus h(ID_{MU}\,||PW_{MU})$. Without having the secret credential s of MU and secret key SK_{HA} of HA, it is computationally difficult for $\mathcal{A}$ to guess password PW_{MU} of MU correctly through the *send* queries. Since the games GM_2 and GM_3 are identical when the password guessing attack is absent, using the Zipf's law on passwords [18], it follows that

$$|Pr[Succ_3] - Pr[Succ_2]| \leq C' \cdot q_{send}^{s'} \tag{4}$$

where C' and s' are the Zipf's parameters [18].

Since all the games are executed, it is remained for $\mathcal{A}$ to guess the correct bit c. Thus, we have,

$$Pr[Succ_3] = \frac{1}{2}. \tag{5}$$

Eqs. (1) and (2) give the following result:

$$\begin{aligned} \frac{1}{2}Adv_{\mathcal{P}} &= |Pr[Succ_0] - \frac{1}{2}| \\ &= |Pr[Succ_1] - \frac{1}{2}|. \end{aligned} \tag{6}$$

Again, Eqs. (5) and (6) produce the following result:

$$\frac{1}{2}Adv_{\mathcal{P}} = |Pr[Succ_1] - Pr[Succ_3]|. \tag{7}$$

Using the triangular inequality, and Eqs. (3) and (4), we obtain the following:

$$\begin{aligned} |Pr[Succ_1] - Pr[Succ_3]| &\leq |Pr[Succ_1] - Pr[Succ_2]| \\ &\quad + |Pr[Succ_2] - Pr[Succ_3]| \\ &\leq \frac{q_h^2}{2|Hash|} + C' \cdot q_{send}^{s'}. \end{aligned} \tag{8}$$

TABLE 2. Notations of the BAN logic.

Notation	Description
$P\| \equiv X$	Principal P **believes** a statement X
#X	Formula X is **fresh**
$P \lhd X$	Principal P **sees** X
$P\| \sim X$	Principal P once **said** X
$P \Rightarrow X$	Principal P **controls** X
$< X >_Y$	Formula X is **combined** with the formula Y
$\{X\}_K$	Formula X is **encrypted** by the key K
$P \overset{K}{\leftrightarrow} Q$	P and Q use the **shared key** K to communicate
SK	Session key uesd in the current session

Eqs. (7) and (8) give the following result:

$$\frac{1}{2}Adv_{\mathcal{P}} \leq \frac{q_h^2}{2|Hash|} + C' \cdot q_{send}^{s'}. \tag{9}$$

Finally, multiplying both sides of Eq. (9) by a factor of 2, we have,

$$Adv_{\mathcal{P}} \leq \frac{q_h^2}{|Hash|} + 2C' \cdot q_{send}^{s'}.$$

Hence, the theorem is proved. ■

B. SECURITY PROOF USING BAN LOGIC

To prove the validity of our authentication scheme, we perform the BAN logic [16] analysis which is well-known as formal security model. First, we define the notations of BAN logic in Table 2 and describe logical postulates of BAN logic. Next, we demonstrate that our proposed scheme can achieves mutual authentication between MU and FA.

1) POSTULATES OF BAN LOGIC

The following postulates of the BAN logic are given below:

1. Message meaning rule :

$$\frac{P \,|\equiv P \overset{K}{\leftrightarrow} Q, \quad P \lhd \{X\}_K}{P \,|\equiv Q\,|\sim X}$$

2. Nonce verification rule :

$$\frac{P \,|\equiv \#(X), \quad P\,|\equiv Q \,|\sim X}{P \,|\equiv Q\,|\equiv X}$$

3. Jurisdiction rule :

$$\frac{P \,|\equiv P\,|\Longrightarrow X, \quad P \,|\equiv Q\,|\equiv X}{P \,|\equiv X}$$

4. Freshness rule :

$$\frac{P \,|\equiv \#(X)}{P \,|\equiv \#(X, Y)}$$

5. Believe rule :

$$\frac{P \mid \equiv (X, Y)}{P \mid \equiv X.}$$

In order to perform the analytic procedures of BAN logic, we first list the verification goals and idealized form of our proposed scheme. After that, to analyze the proposed scheme, we make the initial state assumptions and demonstrate that our proposed scheme can achieve the secure mutual authentication.

2) GOALS

We have the following goals related to our proposed scheme:

Goal 1: $MU \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$
Goal 2: $FA \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$
Goal 3: $MU \mid\equiv FA \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$
Goal 4: $FA \mid\equiv MU \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$

3) IDEALIZED FORMS

The idealized forms of the messages are as follows:

Msg_1: $MU \rightarrow FA$: $(EID, EID_{new}, N_M)_S$
Msg_2: $FA \rightarrow HA$: $(EID, EID_{new}, S, N_M, N_F, ID_{FA}, FA \overset{K_{FA}}{\longleftrightarrow} HA)_{SK_{FA}}$
Msg_3: $HA \rightarrow FA$: $(EID_{new}, S, S_{new}, N_F, R_i)_{SK_{FA}}$
Msg_4: $FA \rightarrow MU$: $(EID, N_{F2}, S_{new})_S$
Msg_5: $MU \rightarrow FA$: $(N_M, N_{F2}, S_{new})_S$

4) ASSUMPTIONS

We make the following assumptions in the BAN logic proof:

A_1: $FA \mid\equiv (MU \overset{S}{\longleftrightarrow} FA)$
A_2: $FA \mid\equiv \#(N_M)$
A_3: $HA \mid\equiv (HA \overset{SK_{FA}}{\longleftrightarrow} FA)$
A_4: $HA \mid\equiv \#(N_F)$
A_5: $FA \mid\equiv (HA \overset{SK_{FA}}{\longleftrightarrow} FA)$
A_6: $FA \mid\equiv \#(N_F)$
A_7: $MU \mid\equiv (MU \overset{S}{\longleftrightarrow} FA)$
A_8: $MU \mid\equiv \#(N_{F2})$
A_9: $FA \mid\equiv \#(N_{F2})$
A_{10}: $MU \mid\equiv FA \Rightarrow (MU \overset{K_{MF}}{\longleftrightarrow} FA)$
A_{11}: $FA \mid\equiv MU \Rightarrow (MU \overset{K_{MF}}{\longleftrightarrow} FA)$

5) PROOF USING BAN LOGIC

The main proof consists of the following steps:

Step 1: According to Msg_1, we could get:

$$S_1 : FA \lhd (EID, EID_{new}, N_M)_S$$

Step 2: From S_1 and assumption A_1, we apply the message meaning rule to obtain:

$$S_2 : FA \mid\equiv MU \sim (EID, EID_{new}, N_M)_S$$

Step 3: From S_2 and A_2, we apply the freshness rule to obtain:

$$S_3 : FA \mid\equiv \#(EID, EID_{new}, N_M)_S$$

Step 4: According to S_2 and S_3, we apply the nonce verification rule:

$$S_4 : FA \mid\equiv MU \mid\equiv (EID, EID_{new}, N_M)_S$$

Step 5: From S_4, we apply the believe rule to obtain:

$$S_5 : FA \mid\equiv MU \mid\equiv (N_M)_S$$

Step 6: According to Msg_2, we could get:

$$S_6 : HA \lhd (EID, EID_{new}, S, N_M, N_F, ID_{FA}, FA \overset{K_{FA}}{\longleftrightarrow} HA)_{SK_{FA}}$$

Step 7: From S_6 and A_3, we apply the message meaning rule to obtain:

$$S_7 : HA \mid \equiv FA \sim (EID, EID_{new}, S, N_M, N_F, ID_{FA}, FA \overset{K_{FA}}{\longleftrightarrow} HA)_{SK_{FA}}$$

Step 8: From S_6 and A_4, we apply the freshness rule to obtain:

$$S_8 : HA \mid \equiv \#(EID, EID_{new}, S, N_M, N_F, ID_{FA}, FA \overset{K_{FA}}{\longleftrightarrow} HA)_{SK_{FA}}$$

Step 9: According to S_7 and S_6, we apply the nonce verification rule to obtain:

$$S_9 : HA \mid \equiv FA \mid\equiv (EID, EID_{new}, S, N_M, N_F, ID_{FA}, FA \overset{K_{FA}}{\longleftrightarrow} HA)_{SK_{FA}}$$

Step 10: According to Msg_3, we could get:

$$S_{10} : FA \lhd (EID_{new}, S, S_{new}, N_F, R_i)_{SK_{FA}}$$

Step 11: From S_{10} and A_5, we apply the message meaning rule to obtain:

$$S_{11} : FA \mid \equiv HA \sim (EID_{new}, S, S_{new}, N_F, R_i)_{SK_{FA}}$$

Step 12: From S_{11} and A_6, we apply the freshness rule to obtain:

$$S_{12} : FA \mid\equiv \#(EID_{new}, S, S_{new}, N_F, R_i)_{SK_{FA}}$$

Step 13: According to S_{12} and S_{11}, we apply the nonce verification rule to obtain:

$$S_{13} : FA \mid \equiv HA \mid\equiv (EID_{new}, S, S_{new}, N_F, R_i)_{SK_{FA}}$$

Step 14: According to Msg_4, we could get:

$$S_{14} : MU \lhd (EID, N_{F2}, S_{new})_S$$

Step 15: From S_{14} and assumption A_7, we apply the message meaning rule to obtain:

$$S_{15} : MU \mid\equiv FA \sim (EID, N_{F2}, S_{new})_S$$

Step 16: From S_{15} and A_8, we apply the freshness rule to obtain:

$$S_{16} : MU \mid\equiv \#(EID, N_{F2}, S_{new})_S$$

Step 17: According to S_{16} and S_{15}, we apply the nonce verification rule:

$$S_{17} : MU \mid\equiv FA \mid\equiv (EID, N_{F2}, S_{new})_S$$

Step 18: From S_{17}, we apply the believe rule to obtain:

$$S_{18} : MU \mid\equiv FA \mid\equiv (N_{F2})_S$$

Step 19: According to Msg_5, we could get:

$$S_{19} : FA \lhd (N_M, N_{F2}, S_{new})_S$$

Step 20: From S_{19} and assumption A_7, we apply the message meaning rule to obtain:

$$S_{15} : FA \mid\equiv MU \sim (EID, N_{F2}, S_{new})_S$$

Step 21: From S_{15} and A_1, we apply the freshness rule to obtain:

$$S_{21} : FA \mid\equiv \#(N_M, N_{F2}, S_{new})_S$$

Step 22: According to S_{21} and S_{20}, we apply the nonce verification rule:

$$S_{22} : FA \mid\equiv MU \mid\equiv (N_M, N_{F2}, S_{new})_S$$

Step 23: Because of the session key $K_{MF} = h(N_M||N_{F2}||S)$, according to S_5, S_9, S_{13}, S_{18} and S_{22} we could get :

$$S_{23} : MU \mid\equiv FA \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$$

(Goal 3)

$$S_{24} : FA \mid\equiv MU \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$$

(Goal 4)

Step 24: According to S_{23} and A_{10}, we apply the jurisdiction rule to obtain:

$$S_{25} : MU \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$$

(Goal 1)

Step 25: According to S_{24} and A_{11}, we apply the jurisdiction rule to obtain:

$$S_{26} : FA \mid\equiv (MU \overset{K_{MF}}{\longleftrightarrow} FA)$$

(Goal 2)

From the goals 1–4, it is clear that our proposed scheme can achieve mutual authentication between MU and FA.

C. INFORMAL SECURITY ANALYSIS

In this section, we perform an informal analysis of our proposed scheme to demonstrate that it is secure against various attacks such as offline password guessing, user impersonation, replay and privileged-insider attacks. We also show that our scheme provides perfect forward secrecy, secure password altered phase and secure mutual authentication.

1) OFFLINE PASSWORD GUESSING ATTACK

In our proposed scheme, an adversary MU_a can try to guess the password of MU. To guess an password of MU correctly, MU_a can know the value $h(MU_{ID}||s)$ however MU_a cannot know values MU_{ID} and $h(MU_{ID}||s)$. Therefore, our proposed scheme can resist offline password guessing attack.

2) USER IMPERSONATION ATTACK

If an adversary MU_a attempts to impersonate mobile user MU, MU_a must know the password of MU correctly. However, according to section V-C.1, MU_a cannot know the password of MU. As a result, since the MU_a cannot generate authentication request messages, our proposed scheme can prevent user impersonation attack.

3) REPLAY ATTACK

We assume that an adversary MU_a can know the messages transmitted in previous session and wants to access to the foreign agent FA.

To access the roaming service, MU_a may resend the previous authentication request message $\{EID', V_M, Q_M, N_M\}$ and receives the response messages $\{V_{F2}, Q_{F2}, N_{F2}\}$ from FA. However, MU_a cannot computes S_{new} because MU_a cannot know the value S. In addition, N_M and N_{F2} are changed in every session. Therefore, since MU_a cannot authenticate with FA, the proposed scheme can resist replay attack.

4) PERFECT FORWARD SECRECY

We assume that MU_a intercepts and stores messages transmitted in the previous session, and SK_{FA} is compromised by MU_a. In our scheme, the session key $K_{MF} = h(N_M||N_F||S)$ and MU_a can attempt to compute previous session key. However, MU_a cannot retrieve N_F because MU_a cannot know the shared secret key k_{FA} between FA and HA. For this reason, MU_a cannot compute $K_{MF} = h(N_M||N_F||S)$ and our proposed scheme can provide perfect forward secrecy.

5) SECURE PASSWORD ALTERED PHASE

In our scheme, we suppose that an adversary MU_a wants to change the password of MU. However, since the password altered phase is similar to AESK phase, MU_a cannot change the password freely without password of MU. Therefore, our scheme can provide secure password altered phase.

6) SECURE MUTUAL AUTHENTICATION

In our scheme, the values $Q_{F2} = h(EID||S_{new}||N_{F2})$ and $Q_{MF} = h(N_M||S||N_{F2}||S_{new})$ are checked by MU and FA respectively. However, the random number N_{F2} and N_M must use each session. Beside, to compute the session key $K_{MF} = h(N_M||N_{F2}||S)$, they must know the value S. Therefore, our proposed scheme can provide secure mutual authentication because the value S only can know FA, MU and HA.

7) PRIVILEGED-INSIDER ATTACK

In a privileged-insider attack, an insider user of a trusted entity being an adversary $\mathcal{A}$ tries to defeat the security of the system. This is considered as a serious attack in an authentication scheme [20], [23], [24]. Suppose $\mathcal{A}$ residing in the *HA* knows the information $\{EID\}$ which was delivered to the *HA* securely during the mobile user registration phase (see Section IV-A.2), where $EID = h(ID_{MU} \oplus PW_{MU}) \oplus s$ and s being a secret random of the mobile user MU. Also, assume that $\mathcal{A}$ attains the lost or stolen mobile device of MU after the registration process if finished. Then, $\mathcal{A}$ can easily extract all the sensitive information $\{SPW, s'\}$ stored in the device using the power analysis attacks [13], [14] as described in the threat model (Section I-.1), where $SPW = S \oplus h(PW_{MU}) \oplus h(ID_{MU}||s)$, $S = h(EID\ ||h(SK_{HA}))$ and $s' = s \oplus h(ID_{MU}||PW_{MU})$. Now, without having the secret credentials s of MU and secret key SK_{HA} of *HA*, it is computationally infeasible problem for $\mathcal{A}$ to guess ID_{MU} as well as password PW_{MU} of MU, and S correctly. This assures that the proposed scheme is tolerable to the privileged-insider attack.

D. FORMAL SECURITY VERIFICATION USING AVISPA TOOL: SIMULATION STUDY

In this section, the proposed scheme is tested for the formal security verification using the widely-accepted AVISPA tool [17]. AVISPA tool has gained popularity in recent years and has been considered as one of the powerful tool for testing whether a security protocol is safe or unsafe against replay and man-in-the-middle attacks [19], [20], [23]–[27].

In AVISPA, a designed security protocol needs to be implemented using the role-oriented language, called High-Level Protocol Specification Language (HLPSL) [28]. Under the HLPSL, the various basic roles and two other mandatory roles, known as *session* and *environment* to be defined. In the HPLSL implementation of the proposed scheme, we have three basic roles: 1) *mobileuser* for a mobile user (MU) shown in Figure 8, 2) *homeagent* for the home agent shown in Figure 9 and 3) *foreignagent* for a foreign agent (*FA*) shown in Figure 10. The roles for the session and environment are termed as *session* and *environment*, respectively. The *environment* role further contains the secrecy goals under which the protocol becomes secure or insecure. The detailed documentation on AVISPA and its HLPSL specification is available online [17].

1) DEFINING ROLE SPECIFICATION

Consider the role of a mobile user (MU) which is implemented in HLPSL in Figure 8. At first, MU receives the start signal and then makes its initialized state (State = 0) to 1, and sends the registration request message $\{EID\}$ to *HA* securely during the mobile user registration phase. After that MU receives the registration reply $\{S\}$ from MU securely, and updates its state from 2 to 4. During the authentication and establishment of session key (AESK) phase, MU transmits the message $\{EID', V_M, Q_M, N_M\}$ to *FA* via open channel (using *channel(dy)*). Here, *channel(dy)* means the channel is insecure and follow the DY model as discussed in the threat model (Section I-.1). After sending this message, MU declares that he/she has freshly generated random numbers N_M and S_{new} for *FA*, which are reflected in the declarations: 1) witness(MU, FA, mu_fa_nm, Nm') and 2) witness(MU, FA, mu_fa_sn, Sn'). Once MU receives the message $\{V_{F2}, Q_{F2}, N_{F2}\}$ from *FA*, the state is changed from 4 to 6. Finally, MU sends the message $\{Q_{MF}\}$ to *FA* for mutual authentication purpose. In this role, MU authenticates *FA* based on the random value N_{F2} using the declaration request(FA, MU, fa_ha_nf2, Nf2'). In a similar manner, the roles for *HA* and *FA* are defined in HLPSL in Figures 9 and 10.

```
%%%%%%% Role for mobile user MU %%%%%%

role mobileuser (MU, HA, FA : agent, SKmuha : symmetric_key,
        % H is one−way hash function
        H: hash_func, SND, RCV: channel(dy))
% Player: the mobile user MU
played_by MU
def=
local State: nat,
    S, IDmu, PWmu, EID, SKha, Sn, Nm, EIDn,
  Sn1, Qm, Vm, Nf2, Qmf : text
const sp1, sp2, sp3, mu_fa_nm, mu_fa_sn : protocol_id
init  State := 0
transition
% Mobile user registration phase
1. State = 0 ∧ RCV(start) =|>
State' := 2  ∧ S' := new() ∧ EID' :=xor(H(xor(IDmu,PWmu)),S')
% Send registration request <EID> to HA securely
   ∧ SND({EID'}_SKmuha)
   ∧ secret({IDmu}, sp1, {MU,HA}) ∧ secret({PWmu,S'}, sp2, {MU})
% Receive registration reply from HA securely
2. State = 2 ∧ RCV({H(xor(H(xor(IDmu,PWmu)),S').H(SKha))}_SKmuha)=|>
 State' := 4 ∧ secret({SKha}, sp3, {HA})
% Authentication & establishment of session key phase
        ∧ Sn' := new() ∧ Nm' := new()
        ∧ EIDn' := xor(H(xor(IDmu,PWmu)),Sn')
        ∧ Vm' := xor(EIDn', H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).Nm'))
        ∧ Sn1' := xor(Sn',H(IDmu.PWmu))
        ∧ Qm' := H(EIDn'.Sn1'.Nm')
% Send message {EID',Vm,Qm,Nm} to FA via open channel
        ∧ SND(EID'.Vm'.Qm'.Nm')
% MU has freshly generated random numbers Nm and Sn for FA
   ∧ witness(MU, FA, mu_fa_nm, Nm')
   ∧ witness(MU, FA, mu_fa_sn, Sn')
% Receive message <Vf2, Qf2, Nf2> from FA via open channel
3. State = 4 ∧ RCV(xor(H(xor(H(xor(IDmu,PWmu)),Sn').H(SKha)),
            H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).Nf2')).
            H(xor(H(xor(IDmu,PWmu)),S').H(xor(H(xor(IDmu,PWmu)),Sn').
            H(SKha)).Nf2').Nf2') =|>
State' := 6 ∧ Qmf' := H(Nm'.H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).
                Nf2'.H(xor(H(xor(IDmu,PWmu)),Sn').H(SKha))))
% Send message <Qmf> to FA via open channel
        ∧ SND(Qmf')
% MU's acceptance of the value Nf2 generated for FA by MU
        ∧ request(FA, MU, fa_ha_nf2, Nf2')
end role
```

FIGURE 8. Role specification for the mobile user (*MU*).

2) DISCUSSION ON SIMULATION RESULTS

The broadly-used Constraint Logic based Attack Searcher (CL-AtSe) backend has been selected for formal security verification purpose in order to find if there are any attacks on the proposed scheme [17]. For the replay attack checking,

```
%%%%%%% Role for HA %%%%%
role homeagent (MU, HA, FA : agent, SKmuha : symmetric_key,
        H: hash_func, SND, RCV: channel(dy))
% Player: the home agent HA
played_by HA
def=
 local State : nat,
     IDmu, PWmu, S, S1, SKha, Kfa, Sn, Nm,Sn1, Nf, SKfa, IDfa, Vh: text
 const sp1, sp2, sp3, fa_ha_nf: protocol_id
init  State := 0
transition
% Mobile user registration phase
% Receive registration request <EID> to from MU securely
1. State = 0 /\ RCV({xor(H(xor(IDmu,PWmu)),S')}_SKmuha) =|>
  State' := 3 /\ secret({IDmu}, sp1, {MU,HA})
          /\ secret({PWmu,S'}, sp2, {MU})
          /\ S1' := H(xor(H(xor(IDmu,PWmu)),S').H(SKha))
          /\ secret({SKha}, sp3, {HA})
% Send registration reply to MU securely
          /\ SND({S1'}_SKha)
% Authentication & establishment of session key phase
% Receive message {EID',Vm,Qf,Nm,Vf,IDfa} from FA via open channel
2. State = 3 /\ RCV(xor(H(xor(IDmu,PWmu)),S').xor(xor(H(xor(IDmu,PWmu)),Sn'),
              H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).Nm')).
              H(H(xor(H(xor(IDmu,PWmu)),Sn').Sn1'.Nm').Nf'.SKfa').Nm'.
              xor(Nf',H(SKfa'),H(Kfa.Nm')).IDfa) =|>
 State' := 6 /\ secret({Kfa}, sp4, {FA,HA})
         /\ Vh' := xor((xor(H(xor(IDmu,PWmu)),Sn').
                H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).
                H(xor(H(xor(IDmu,PWmu)),Sn').H(SKha))), H(SKfa.Nf'))
% Send message {Vh} to FA via open channel
         /\ SND(Vh')
% HA's acceptance of the value Nf generated for HA by FA
         /\ request(FA, HA, fa_ha_nf, Nf')
end role
```

FIGURE 9. Role specification for the home agent (*HA*).

CL-AtSe checks if the specified protocol can be executed by the legitimate parties by searching for a passive intruder. The back-end then gives the intruder (which is always denoted by the special symbol i) with information about a few normal sessions between the valid parties. In addition, for the DY model checking, the CL-AtSe also checks if there is any possibility of man-in-the-middle attack by the intruder (i). Note that the intruder (i) has knowledge of all public parameters and can also play a legitimate role in AVISPA protocol execution. For this purpose, the role of the intruder (i) is also included in the role *environment* as shown in Figure 11.

The proposed scheme is finally simulated using the SPAN, the Security Protocol ANimator for AVISPA tool [29] for the CL-AtSe backend. The simulation results are organized in the output format (OF), which has the following sections:

- SUMMARY: This section either indicates that the proposed scheme is safe or unsafe or that the analysis is inconclusive.
- DETAILS: This section explains under which the conditions the proposed scheme is safe or when attacks are possible or the reason for an inconclusive analysis.
- BACK-END, GOAL and PROTOCOL: These sections indicate which the backend used to analyze, the goal of the analysis and the name of the protocol, respectively.
- Finally, if an attack is found during the protocol execution, the trace of the attack is also printed in the standard Alice-Bob format with a few statistics and comments.

The simulation results shown in Figure 12 shows that 63 states were analyzed and out of these states, 15 states were reachable. The translation and computation time taken during the execution were 0.09 seconds and 0.36 seconds. The reported simulation results in Figure 12 assure that the proposed scheme is safe against replay and man-in-the-middle attacks.

```
%%%%%%% Role for FA %%%%%
role foreignagent (MU, HA, FA : agent, H: hash_func, SND, RCV: channel(dy))
% Player: the foreign agent FA
played_by FA
def=
 local State: nat,
     S, IDmu, PWmu, EID, SKha, Nm, Sn, SKfa,
  Nf, IDfa, Qf, EIDn, Sn1, Kfa, Vf, Nf2, Qf2, Vf2: text
const sp1, sp2, sp3, mu_fa_nm, mu_fa_sn, fa_ha_nf, fa_ha_nf2: protocol_id
init  State := 0
transition
% Authentication & establishment of session key phase
% Receive message {EID',Vm,Qm,Nm} from MU via open channel
1. State = 0 /\ RCV(xor(H(xor(IDmu,PWmu)),S').xor(xor(H(xor(IDmu,PWmu)),Sn'),
              H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).Nm')).
              H(xor(H(xor(IDmu,PWmu)),Sn').xor(Sn',H(IDmu.PWmu)).Nm').Nm')=|>
State' := 1 /\ secret({IDmu}, sp1, {MU,HA}) /\ secret({PWmu,S'}, sp2, {MU})
         /\ secret({SKha}, sp3, {HA}) /\ secret({Kfa}, sp4, {FA,HA})
         /\ Nf' := new() /\ SKfa' := H(xor(IDfa,SKha))
         /\ Qf' := H(H(EIDn'.Sn1'.Nm').Nf'.SKfa')
         /\ Vf' := xor(Nf',H(SKfa'),H(Kfa.Nm'))
% Send message {EID',Vm,Qf,Nm,Vf,IDfa} to HA via open channel
         /\ SND(xor(H(xor(IDmu,PWmu)),S').xor(xor(H(xor(IDmu,PWmu)),Sn'),
              H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).Nm')).
              Qf'.Nm'.Vf'.IDfa)
% FA has freshly generated random Nf for HA
         /\ witness(FA, HA, fa_ha_nf, Nf')
2. State = 1 /\ RCV(xor((xor(H(xor(IDmu,PWmu)),Sn').
              H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).
              H(xor(H(xor(IDmu,PWmu)),Sn').H(SKha))), H(SKfa.Nf'))) =|>
 State' := 3 /\ Nf2' := new()
          /\ Vf2' := xor(H(xor(H(xor(IDmu,PWmu)),Sn').H(SKha)),
                H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).Nf2'))
          /\ Qf2' := H(xor(H(xor(IDmu,PWmu)),S').H(xor(H(xor(IDmu,PWmu)),Sn').
                H(SKha)).Nf2')
% Send message {Vf2, Qf2, Nf2} to MU via open channel
          /\ SND(Vf2'.Qf2'.Nf2')
% FA has freshly generated random Nf2 for MU
         /\ witness(FA, HA, fa_ha_nf2, Nf2')
% Receive message {Qmf} from MU via open channel
3. State = 3 /\ RCV(H(Nm'.H(H(xor(H(xor(IDmu,PWmu)),S').H(SKha)).
              Nf2'.H(xor(H(xor(IDmu,PWmu)),Sn').H(SKha))))) =|>
% FA's acceptance of the values Nm and Sn generated for FA by MU
 State' := 5 /\ request(MU, FA, mu_fa_nm, Nm')
          /\ request(MU, FA, mu_fa_sn, Sn')
end role
```

FIGURE 10. Role specification for the foreign agent (*FA*).

VI. PERFORMANCE COMPARISON

In this section, we compare the performance our proposed scheme with the existing related schemes, such as the schemes of Lee *et al.* [5], Mun *et al.* [6], and Wu *et al.* [9].

A. SECURITY FEATURES COMPARISON

We have analyzed the security properties of our proposed scheme with the related schemes in Table 3. From this table, it is clear that the existing schemes do not protect various attacks, whereas the proposed scheme is secure against the attacks. Specifically, perfect forward secrecy (SP5) and mutual authentication (SP7) are not supported in the existing schemes. Thus, the proposed scheme provides better security as compared to other schemes of Lee *et al.* [5], Mun *et al.* [6], and Wu *et al.* [9].

```
%%% Role for the session %%%

role session (MU, HA, FA : agent,
          SKmuha : symmetric_key,
          H: hash_func)
def=
 local  SN1, SN2, SN3, RV1, RV2, RV3: channel (dy)
 composition
    mobileuser (MU, HA, FA, SKmuha, H, SN1, RV1)
  /\ homeagent (MU, HA, FA, SKmuha, H, SN2, RV2)
  /\ foreignagent (MU, HA, FA, H, SN3, RV3)
end role

%%% Role for the goal and environment %%%

role environment()
def=
 const mu, ha, fa: agent,
       skmuha: symmetric_key,
       h : hash_func,
       idha, idfa: text,
       mu_fa_x, fa_ha_y: protocol_id,
       sp1, sp2, sp3, sp4: protocol_id
 intruder_knowledge = {mu, ha, fa, h, idfa}
 composition
  session(mu, ha, fa, skmuha, h)
/\ session(i, ha, fa, skmuha, h)
       /\ session(mu, i, fa, skmuha, h)
       /\ session(mu, ha, i, skmuha, h)
end role
goal
 secrecy_of sp1, sp2, sp3, sp4
 authentication_on fa_ha_nf, fa_ha_nf2
 authentication_on mu_fa_nm, mu_fa_sn
end goal
environment()
```

FIGURE 11. Role specification for the session, goal and environment.

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  C:\progra~1\SPAN\testsuite
    \results\auth.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 63 states
  Reachable  : 15 states
  Translation: 0.09 seconds
  Computation: 0.36 seconds
```

FIGURE 12. The simulation results using the CL-AtSe backend.

B. COMPUTATION OVERHEAD COMPARISON

For computation overhead comparison, we use the following notations. T_h, T_{XOR}, T_{sym}, T_{asym} and T_{ecm} denote the time needed for the one-way hash operation, exclusive-OR operation, symmetric key encryption/decryption, asymmetric key encryption/decryption and an elliptic curve point multiplication, respectively. For rough estimation, we consider the existing experimental results reported by Lee *et al.* [30]. Lee *et al.* [30] did an evaluation on a four-core 3.2 GHz machine with 8 GB memory, and the results were averaged over 300 randomized simulation runs. It was mentioned that one symmetric encryption/decryption is at least 100 times faster than one public-key encryption/decryption and one exponential operation is approximately equal to 60 symmetric encryptions/decryptions. Based on the results, T_{ecm}, T_{sym} and T_h are $0.063075s$ seconds, 0.0087 seconds and 0.0005 seconds, respectively [21], [30]. Then, we have, $T_{asym} \approx 100$ $T_{sym} \approx 0.87$ seconds. Since exclusive-OR operation is negligible as compared to other operations, it is ignored in the rough estimation calculation. The computation overheads between the proposed scheme and other schemes are shown in Table 4. It is worth noticing that the computation time needed for the schemes of Lee *et al.* [5], Mun *et al.* [6], and Wu *et al.* [9] are 2.6421 seconds, 0.13315 seconds and 0.0095 seconds, respectively, whereas it is 0.0135 seconds for the proposed scheme. It is clear that the proposed scheme is better in term of computation cost as compared to the existing Wu *et al.*'s scheme [9] and Mun *et al.*'s scheme [6]. Though Lee *et al.*'s scheme [5] needs little less computation cost as compared to the proposed scheme, our scheme provides better security as compared to other schemes.

TABLE 3. Security properties of our proposed scheme with other related schemes.

Security property	Wu *et al.* [9]	Mun *et al.* [6]	Lee *et al.* [5]	Our
SP_1	○	○	×	○
SP_2	×	×	×	○
SP_3	×	×	×	○
SP_4	×	○	○	○
SP_5	×	×	×	○
SP_6	○	×	○	○
SP_7	×	×	×	○

SP_1: offline password guessing attack; SP_2: user anonymity; SP_3: impersonation attack; SP_4 replay attack; SP_5: perfect forward secrecy; SP_6: man in the middle attack; SP_7: mutual authentication.
○: preserves the security properties; ×: does not preserve the security properties;

C. COMMUNICATION OVERHEAD COMPARISON

For communication overhead comparison, we assume that identity of an entity, timestamp, random nonce/secret, hash output (if we apply Secure Hash Standard (SHA-1) algorithm [31]), elliptic curve point $P = (P_x, P_y)$ and ciphertext using public key cryptosystem (if we apply RSA public key cryptosystem [32]) are 160 bits, 32 bits, 128 bits, 160 bits, (160+160) = 320 bits and 1024 bits, respectively. In addition, $|Cert_{FA}|$ and $|Cert_{HA}|$ represent the bit lengths in the certificates $Cert_{FA}$ of FA and $Cert_{HA}$ of HA, respectively. In the proposed scheme, the messages $\{EID', V_M, Q_M, N_M\}$, $\{EID', V_M, Q_F, N_M, V_F, ID_{FA}\}$, $\{V_H\}$, $\{V_{F2}, Q_{F2}, N_{F2}\}$ and $\{Q_{MF}\}$ need $(160 + 160 + 160 + 128) = 608$ bits,

TABLE 4. Comparison of computation overheads.

	Wu *et al.* [9]	Mun *et al.* [6]	Lee *et al.* [5]	Our
MU	$3T_h + 1T_{XOR} + 1T_{sym}$	$5T_h + 2T_{XOR} + 1T_{ecm}$	$6T_h + 6T_{XOR}$	$9T_h + 9T_{XOR}$
FA	$5T_h + 3T_{asym}$	$4T_h + 2T_{XOR} + 1T_{ecm}$	$6T_h + 6T_{XOR}$	$10T_h + 4T_{XOR}$
HA	$4T_h + 3T_{XOR} + 2T_{sym}$	$5T_h + 3T_{XOR}$	$7T_h + 4T_{XOR}$	$8T_h + 4T_{XOR}$
Total cost	$12T_h + 4T_{XOR} + 3T_{sym} + 3T_{asym}$	$14T_h + 7T_{XOR} + 2T_{ecm}$	$19T_h + 16T_{XOR}$	$27T_h + 17T_{XOR}$
Rough estimation	2.6421 s	0.13315 s	0.0095 s	0.0135 s

TABLE 5. Comparison of communication overheads.

Scheme	No. of messages	No. of bits
Wu *et al.* **[9]**	4	$6944 + \|Cert_{FA}\| + \|Cert_{HA}\|$
Mun *et al.* **[6]**	5	2304
Lee *et al.* **[5]**	5	2592
Our	5	2592

$(160 + 160 + 160 + 128 + 160 + 128) = 896$ bits, $\max\{(160 + 160 + 160), 160\} = 480$ bits, $(160 + 160 + 128) = 448$ bits and 160 bits, respectively. Thus, the total communication overhead for five messages transmission is $(608 + 896 + 480 + 448 + 160) = 2592$ bits. The results shown in Table 5 indicate that the communication overhead of the proposed scheme is comparable with that for Mun *et al.*'s scheme [6] and Lee *et al.*'s scheme [5], whereas it is much better than that for Wu *et al.*'s scheme [9].

VII. CONCLUSION

In this paper, we demonstrated that Lee *et al.*'s scheme is vulnerable to various attacks such as offline password guessing and impersonation attacks, and then we showed that it cannot provide perfect forward secrecy, secure mutual authentication and secure password altered phase. To overcome these security weaknesses, we have presented an improved secure anonymous authentication protocol for roaming service in GLOMONETs. The proposed scheme can prevent offline password guessing, user impersonation and replay attacks, and our scheme can provide perfect forward secrecy, secure password altered phase and secure mutual authentication using secret key k_{FA} between *FA* and *HA*. We also demonstrated that our scheme can provide secure mutual authentication between *MU* and *FA* using BAN logic. Moreover, we compared the performances and security properties with related schemes. Therefore, our proposed scheme is more secure than related schemes, and can be applicable to roaming services in GLOMONETs.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and the Associate Editor for their valuable feedback on the paper which helped us to improve its quality and presentation.

REFERENCES

[1] T.-Y. Youn, Y.-P. Park, and J. Lim, "Weaknesses in an anonymous authentication scheme for roaming service in global mobility networks," *IEEE Commun. Lett.*, vol. 13, no. 7, pp. 471–473, Jul. 2009.
[2] P. Gope and T. Hwang, "Enhanced secure mutual authentication and key agreement scheme preserving user anonymity in global mobile networks," *Wireless Pers. Commun.*, vol. 82, no. 4, pp. 2231–2245, Jun. 2015.
[3] C. Chen, D. He, S. Chan, J. Bu, Y. Gao, and R. Fan, "Lightweight and provably secure user authentication with anonymity for the global mobility network," *Int. J. Commun. Syst.*, vol. 24, no. 3, pp. 347–362, 2011.
[4] Q. Jiang, J. Ma, G. Li, and L. Yang, "An enhanced authentication scheme with privacy preservation for roaming service in global mobility networks," *Wireless Pers. Commun.*, vol. 68, no. 4, pp. 1477–1491, 2013.
[5] C.-C. Lee, Y.-M. Lai, C.-T. Chen, and S.-D. Chen, "Advanced secure anonymous authentication scheme for roaming service in global mobility networks," *Wireless Pers. Commun.*, vol. 94, no. 3, pp. 1281–1296, 2017.
[6] H. Mun, K. Han, Y. S. Lee, C. Y. Yeun, and H. H. Choi, "Enhanced secure anonymous authentication scheme for roaming service in global mobility networks," *Math. Comput. Model.*, vol. 55, nos. 1–2, pp. 214–222, 2012.
[7] J. Zhu and J. Ma, "A new authentication scheme with anonymity for wireless environments," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 231–235, Feb. 2004.
[8] C. C. Lee, M. S. Hwang, and I. E. Liao, "Security enhancement on a new authentication scheme with anonymity for wireless environments," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1683–1687, Oct. 2006.
[9] C. C. Wu, W. B. Lee, and W. J. Tsaur, "A secure authentication scheme with anonymity for wireless communications," *IEEE Commun. Lett.*, vol. 12, no. 10, pp. 722–723, Oct. 2008.
[10] C.-C. Chang, C.-Y. Lee, and Y.-C. Chiu, "Enhanced authentication scheme with anonymity for roaming service in global mobility networks," *Comput. Commun.*, vol. 32, no. 4, pp. 611–618, 2009.
[11] T.-F. Lee, "User authentication scheme with anonymity, unlinkability and untrackability for global mobility networks," *Secur. Commun. Netw.*, vol. 6, no. 11, pp. 1404–1413, 2013.
[12] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.
[13] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 541–552, May 2002.
[14] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. 19th Annu. Int. Cryptol. Conf.*, Santa Barbara, CA, USA, Aug. 1999, pp. 388–397.
[15] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Proc. 8th Int. Workshop Theory Pract. Public Key Cryptogr. (PKC)*, Les Diablerets, Switzerland, 2005, pp. 65–84.
[16] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, 1990.

[17] AVISPA. *Automated Validation of Internet Security Protocols and Applications*. Accessed: Apr. 2017. [Online]. Available: http://www.avispa-project.org/
[18] D. Wang, H. Cheng, P. Wang, X. Huang, and G. Jian, "Zipf's law in passwords," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2776–2791, Nov. 2017.
[19] S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos, "Secure biometric-based authentication scheme using Chebyshev chaotic map for multi-server environment," *IEEE Trans. Depend. Sec. Comput.*, to be published, doi: 10.1109/TDSC.2016.2616876.
[20] A. K. Das, M. Wazid, N. Kumar, M. K. Khan, K.-K. R. Choo, and Y. Park, "Design of secure and lightweight authentication protocol for wearable devices environment," *IEEE J. Biomed. Health Inform.*, to be published, doi: 10.1109/JBHI.2017.2753464.
[21] M. Wazid, A. K. Das, S. Kumari, X. Li, and F. Wu, "Provably secure biometric-based user authentication and key agreement scheme in cloud computing," *Secur. Commun. Netw.*, vol. 9, no. 17, pp. 4103–4119, 2016.
[22] C.-C. Chang and H.-D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 357–366, Jan. 2016.
[23] M. Wazid, A. K. Das, N. Kumar, M. Conti, and A. V. Vasilakos, "A novel authentication and key agreement scheme for implantable medical devices deployment," *IEEE J. Biomed. Health Inform.*, to be published, doi: 10.1109/JBHI.2017.2721545.
[24] S. Challa *et al.*, "Secure signature-based authenticated key establishment scheme for future IoT applications," *IEEE Access*, vol. 5, pp. 3028–3043, 2017.
[25] V. Odelu, A. K. Das, S. Kumari, X. Huang, and M. Wazid, "Provably secure authenticated key agreement scheme for distributed mobile cloud computing services," *Future Generat. Comput. Syst.*, vol. 68, pp. 74–88, Mar. 2017.
[26] A. K. Das, A. K. Sutrala, S. Kumari, V. Odelu, M. Wazid, and X. Li, "An efficient multi-gateway-based three-factor user authentication and key agreement scheme in hierarchical wireless sensor networks," *Secur. Commun. Netw.*, vol. 9, no. 13, pp. 2070–2092, 2016.
[27] A. G. Reddy, E.-J. Yoon, A. K. Das, V. Odelu, and K.-Y. Yoo, "Design of mutually authenticated key agreement protocol resistant to impersonation attacks for multi-server environment," *IEEE Access*, vol. 5, pp. 3622–3639, 2017.
[28] D. von Oheimb, "The high-level protocol specification language HLPSL developed in the EU project AVISPA," in *Proc. 3rd APPSEM II Workshop Appl. Semantics (APPSEM)*, Frauenchiemsee, Germany, 2005, pp. 1–17.
[29] AVISPA. *SPAN: Security Protocol ANimator for AVISPA*. Accessed: Apr. 2017. [Online]. Available: http://www.avispa-project.org/
[30] C.-C. Lee, C.-T. Chen, P.-H. Wu, and T.-Y. Chen, "Three-factor control protocol based on elliptic curve cryptosystem for universal serial bus mass storage devices," *IET Comput. Digit. Techn.*, vol. 7, no. 1, pp. 48–56, Jan. 2013.
[31] National Institute of Standards and Technology (NIST), document Secure Hash Standard, FIPS PUB 180-1, U.S. Department of Commerce, Apr. 1995. Accessed: Sep. 2017. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf
[32] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

KISUNG PARK received the B.S. and M.S. degrees in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the School of Electronics Engineering. His research interests include authentication, computer networks, Internet of Things, post-quatum cryptography, VANET, and information security.

YOUNGHO PARK (M'17) received the B.S., M.S., and Ph.D. degrees from Kyungpook National University, Daegu, South Korea, in 1989, 1991, and 1995, respectively, all in electronic engineering. From 1996 to 2008, he was a Professor with the School of Electronics and Electrical Engineering, Sangju National University, South Korea. From 2003 to 2004, he was a Visiting Scholar with the School of Electrical Engineering and Computer Science, Oregon State University, USA. He is currently a Professor with the School of Electronics Engineering, Kyungpook National University. His research interests include computer networks, multimedia, and information security.

YOHAN PARK received the B.S., M.S., and Ph.D. degrees from Kyungpook National University, Daegu, South Korea, in 2006, 2008, and 2013, respectively, all in electronic engineering. He is currently an Assistant Professor with the Information and Communication Department, Division of IT Convergence, Korea Nazarene University. His research interests include computer networks, mobile security, and information security.

ALAVALAPATI GOUTHAM REDDY (M'17) received the Ph.D. degree from the Information Security Laboratory, Kyungpook National University, South Korea, in 2017. He was a Visiting Researcher with the KINDI Center for Computing Research, Qatar University, Qatar. He is currently an Assistant Professor (Research-Focused) with the Department of Computer and Information Security, Sejong University, South Korea. He holds several publications in cryptographic authentication protocols. His primary research interests revolve around cryptography and information security. He is a Professional Member of the ACM.

ASHOK KUMAR DAS (M'17) received the Ph.D. degree in computer science and engineering, the M.Tech. degree in computer science and data processing, and the M.Sc. degree in mathematics from IIT Kharagpur, India. He is currently an Assistant Professor with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad, India. His current research interests include cryptography, wireless sensor network security, hierarchical access control, security in vehicular ad hoc networks, smart grid, Internet of Things, cyber-physical systems and cloud computing, and remote user authentication. He has authored over 150 papers in international journals and conferences in the above areas. Some of his research findings are published in top cited journals such as the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, the IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, the IEEE TRANSACTIONS ON SMART GRID, the IEEE INTERNET OF THINGS JOURNAL, the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, the IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, the *IEEE Electronics Magazine*, the IEEE ACCESS, the *IEEE Communications Magazine*, *Future Generation Computer Systems*, and the *Journal of Network and Computer Applications*. He was a recipient of the Institute Silver Medal from IIT Kharagpur. He is in the Editorial Board of *KSII Transactions on Internet and Information Systems*, and the *International Journal of Internet Technology and Secured Transactions* (Inderscience), and a Guest Editor for the *Computers and Electrical Engineering* (Elsevier) for the special issue on Big data and IoT in e-healthcare, and has served as a program committee member in many international conferences.