Peer-to-Peer Networking and Applications (2020) 13:1943–1966
https://doi.org/10.1007/s12083-020-00929-y

A Robust user authentication protocol with privacy-preserving for roaming service in mobility environments

R. Shashidhara[1] · Sriramulu Bojjagani[2] · Anup Kumar Maurya[3] · Saru Kumari[4] · Hu Xiong[5]

Received: 19 August 2019 / Accepted: 8 May 2020 / Published online: 28 June 2020

Abstract

The authentication system plays a crucial role in the context of GLObal MObility NETwork (GLOMONET) where Mobile User (MU) often need to seamless and secure roaming service over multiple Foreign Agents (FA). However, designing a robust and anonymous authentication protocol along with a user privacy is essential and challenging task. Due to the resource constrained property of mobile terminals, the broadcast nature of a wireless channel, mobility environments are frequently exposed to several attacks. Many researchers focus their interests on designing an efficient and secure mobile user authentication protocol for mobility networks. Very recently (in 2018), Xu et al presented the novel anonymous authentication system for roaming in GLOMONET, and insisted that their protocol is more secure than existing authentication protocols. The security strength of Xu et al.'s authentication protocol is analysed and identified that the protocol is vulnerable to stolen verifier attack, privileged insider attack, impersonation attack and denial of service attack. In-fact, the protocol suffers from clock synchronization problem and cannot afford local password-verification to detect wrong passwords quickly. As a remedy, we proposed an efficient and robust anonymous authentication protocol for mobility networks. The proposed mobile user authentication protocol achieves the provable security and has the ability to resist against numerous network attacks. Besides, the correctness of the novel authentication protocol is validated using formal security tool called AVISPA (Automated Validation of Internet Security Protocols & Applications). Finally, the performance analysis and simulation results reveals that the proposed authentication protocol is computationally efficient and practically implementable in resource limited mobility environments.

Keywords Authentication · Global roaming · Mobile networks · Privacy · User anonymity · Computationally efficient

1 Introduction

With the tremendous growth of communication technologies enabled mobile users to roam across the world in order to access ubiquitous services offered by the mobile network [1]. In recent times, GLOMONET becomes one of the emerging environments to provide seamless roaming service in foreign networks. But it is well-known that the wireless and mobility environment is more prone to attacks. The adversary can eavesdrop, modify, or block the sensitive information communicated through the radio link. Accordingly, mutual authentication process between communication entities in the mobility environment is crucial.

A general authentication scenario for roaming service in GLOMONET consists of a MU (Mobile User), HA (Home Agent) and FA (Foreign Agent). The scenario of user authentication for roaming service in GLOMONET is shown in Fig. 1. The mobile user gets register with Home Network (HN) administrated by the home agent to access network services. When MU moves out from the coverage zone of HA and enters into the Foreign Network (FN) managed by the FA, an authentication is essential between MU, FA and HA to prevent unauthorised access from the attackers. In addition, privacy is a major issue in the mobile networks. The protocols may reveal remote user identity, location and roaming route in the mutual authentication phase, such information is highly confidential. This sensitive data can be sneaked and used by several organizations, in order to promote their business [2]. Thus, an efficient, secure and privacy preserving authentication protocol is essential to resist an

✉ Saru Kumari
saru@ccsuniversity.ac.in

Extended author information available on the last page of the article.

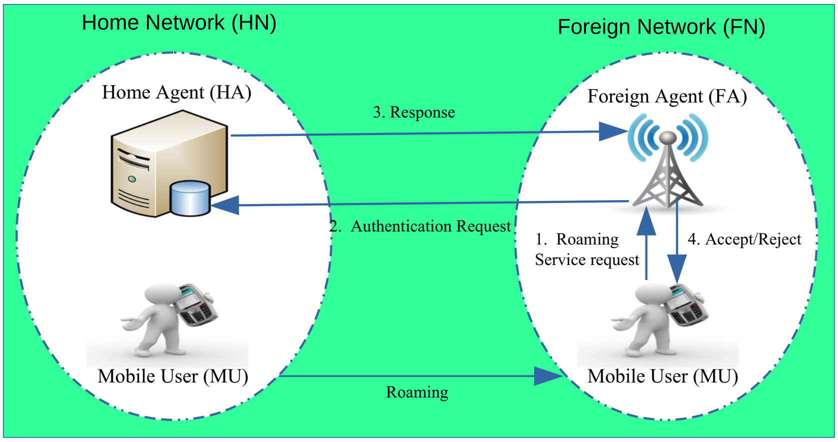


Fig. 1 Mobile user authentication for roaming service in GLOMONET

unauthorised access in the global mobility environments. In this research article, we designed a robust and lightweight authentication protocol for global mobility networks to overcome all security pitfalls identified in [3]. Further, the proposed authentication model is light-weight and computationally efficient.

1.1 Motivation and design goals

The cryptographers all over the world have been aspiring for developing a robust authentication and key establishment protocols using computational mathematics that combat several security threats existing in wireless and global mobility networks. In this context, designing a secure and efficient authentication module, researchers mostly used the computationally intensive security techniques, which could be inefficient for the resource limited mobility environments. As a result, we need a robust authentication protocol for wireless and mobile environments with certain lightweight cryptographic primitives. An attentive review on the existing mobile user authentication protocols under global mobility environment reveals that the most of privacy preserving protocols in the literature have some security pitfalls. Hence, design of a robust and more secure protocol is crucial in this environment [4]. Additionally, the authentication protocol should satisfy all security requirements in GLOMONET and have the ability to ensure low communication and computational complexities. The major design goals of a robust mobile user authentication protocol for GLOMONET are as follows:

– *Secure mutual authentication:* The communication entities MU, FA and HA should authenticate each other without exposing their passwords and identities across insecure channels.
– *Anonymity and untraceability:* The identity information of communication entities should remain secret from the attackers throughout the authentication process. Further, an attacker should not trace the mobile user location and roaming route.
– *Session key security:* The authentication protocol should guarantee that the leakage of session specific information will have no impact on the confidentiality of further sessions.
– *Resilience to various attacks:* The authentication protocol need to resist against replay attack, masquerade attack, insider attack, stolen smart-card attack, stolen verifier attack and so on.
– *Computationally efficient:* The authentication protocol should be lightweight and efficient, in order to cope with resource constrained limitations of mobile devices. In addition, the protocol should be designed using low cost cryptographic primitives.

1.2 Research contributions

Contributions made in this article are outlined as follows:

1. We reveal that the mutual authentication protocol of Xu et al.'s [3] is insecure against impersonation attack, stolen-verifier attack, privileged insider attack, denial-of-service attack, user untraceability attack and suffers from clock synchronization problem. Further, their protocol is unable to provide local password verification to detect wrong passwords quickly.
2. We proposed a robust and lightweight anonymous authentication protocol for global mobility networks to combat all security flaws identified in [3].
3. In the proposed protocol, no trusted third party is involved in the mutual authentication process. This prevents session key attack and decreases overall communication and computational complexities.
4. Through an attentive cryptanalysis, we have proved that the proposed authentication protocol withstand known attacks in the GLOMONET.

5. The correctness of the proposed protocol is verified through a popular formal security tool known as AVISPA.
6. The proposed authentication protocol is compared with other recent security protocols. It is clear that the proposed authentication model achieves low communication and computational costs.
7. Finally, the performance evaluation and simulation results demonstrate that the proposed authentication protocol is lightweight, efficient and practically implementable in resource-constrained mobility environments.

1.3 Road-map of the article

The sequel of the research article as follows: Section 2, includes related work, cryptographic primitives and adversary model. Section 3, gives brief overview of Xu et al. [3] protocol. Section 4, describes the security flaws in [3]. Section 5 outlines our novel authentication protocol for global mobile networks and is corresponding security analysis is presented in Section 6. The formal verification of the protocol using AVISPA is presented in the Section 7. The performance analysis and simulation results is summarized in Section 8. The article is concluded in Section 9.

2 Related work

In order to ensure privacy and secrecy for roaming users, recently numerous user authentication protocols have been designed in the area of wireless and mobile networks. Particularly, In 2004, Zhu et al. [5] introduced a smart-card based two-factor authentication scheme using symmetric and asymmetric crypto primitives for roaming in wireless environments, which preserves anonymity of the user. Subsequently, Lee et al [6] analysed that the protocol presented in [5] cannot achieve backward secrecy, mutual authentication, and is vulnerable to impersonation attacks. In order to combat these security threats, Lee et al. [6] presented a modified version of the protocol in [5] and claims that the enhanced authentication protocol withstands known attacks in GLOMONET. However, Wu et al. [7] analysed the authentication protocol in Lee [6] and found that the protocol cannot provide anonymity and the backward secrecy.

Independently, Yoon et al. [8] presented the user friendly protocol to preserve anonymity in mobile and wireless environments. Unfortunately, Li et al. [9] found that the protocol in [8] is unsuccessful at key agreement, absence of user anonymity and is further proposed a novel authentication protocol for GLOMONET. He et al. [10] designed a lightweight authentication protocol for wireless communications by using XOR and hash functions. Later, Li and Lee [11] proved that the protocol in [10] is unable to provide user anonymity and is also exposed to replay and impersonation attacks. In 2012, Jiang et al. [12] presented an effective anonymous protocol for privacy preserving in mobile networks. However, Wen et al. [13] proved that the authentication scheme in [12] is vulnerable to replay and spoofing attacks and, they proposed an enhanced authentication protocol. In 2014, the authors in [14] pointed that the protocol in [15] cannot achieve mutual authentication, user-friendliness and local password-verification. Further, the protocol is vulnerable to forgery attack. In 2015, the authors in [2] proposed a lightweight and efficient protocol for GLOMONETs. However, Wu et al. [16] showed that the protocol in [2] suffers from de-synchronization problem, unfair key agreement, and impracticality due to the time delay. Besides, they come up with a enhanced mobile user authentication protocol. Later on, some new anonymous and privacy preserving authentication protocols have been proposed using blockchain technology [17–22]. Additionally, the authors in [23–32] proposed a robust authentication protocols for roaming in mobility environments. However, these protocols introduce more computational overhead. Recently, Xu et al. [3] analysed the authentication protocol in [23] and identified that their protocol is susceptible to replay attack, de-synchronization problem and have a large storage burden and is further proposed a new mutual authentication protocol. However, we analysed the security strength of the protocol in [3] and found that the protocol is vulnerable to stolen verifier attack, denial of service attack, privileged insider attack, impersonation attack, clock synchronization problem and unable to provide local password-verification. As a remedy, we presented a robust mobile user authentication protocol for mobility networks.

2.1 Cryptographic primitives

In order to ensure secrecy and efficiency of the authentication system, we prefer to use lightweight and low cost cryptographic primitives like XOR, secure hash functions and symmetric crypto operations in the proposed protocol.

2.1.1 EXCLUSIVE-OR cipher:

In cryptography, the simple EXCLUSIVE-OR algorithm has the successive additive principles:

$$A \oplus A = 0, \; A \oplus 0 = A$$

$$A \oplus (B \oplus C) = (A \oplus B) \oplus C, \; (B \oplus A) \oplus A = B \oplus 0 = B$$

2.1.2 Hash function:

A secure hash function accepts the string of variable length as an input and produces the fixed length output known as hash value [33]. In cryptography, the secure one-way hash operation has following properties:

- For given input I it's very easy to calculate H(I), but its impossible to compute I from given H(I).
- Its very difficult to find the pair of input A and B such that H(A)=H(B), such a pair is known as hash collision.

2.2 Adversary model

Here, we consider the capabilities of an adversary to highlight security threats and privacy issues during authentication in global mobility networks:

1. An attacker $\mathscr{A}$ is assumed to have complete control over insecure wireless channel where the mobile entities MU, FA and HA mutually exchange the authentication messages. This afford to eavesdrop, modify, and delete any sensitive messages flows between mobile user to authentication server [34].
2. Attacker $\mathscr{A}$ is capable to extract or read the parameters from stolen or lost smart card by monitoring power consumption [9, 15].
3. In case of verification tables in the FA or HA, an adversary $\mathscr{A}$ is able to attain confidential information of registered mobile users.
4. The attacker $\mathscr{A}$ can trace the location information of the particular mobile subscriber, when any of the user specific parameter is remains constant in all authentication sessions.
5. The identity and password information of the mobile user is in a finite set, an attacker $\mathscr{A}$ is able to guess them in a polynomial time [35].

3 Review of recently proposed novel authentication scheme [3]

In this part, we review Xu et al.'s [3] novel efficient authentication and key establishment protocol. The cryptographic notations used in the article are listed in Table 1. The phases in Xu et al. authentication protocol are as follows:

3.1 Registration phase

A new MU submits the identity ID_M to HA through secure channel. After receiving MU's request, HA generates two random numbers $n_h,\ n_0$ and computes:

$$K_{uh} = h(ID_M||n_h);\ \ EID = E_K(ID_M||n_0).$$

Table 1 Cryptographic notations in the article

Symbol	Description
PSW_M	Password of the mobile user
ID_M, ID_H, ID_F	Identities of MU, HA and FA
K_{FH}	Shared-secret key
S_{HA}	HA's secret key
$\mathscr{A}$	Adversary
K_{MU}	Counter value of MU
$(E/D)_K$	Symmetric crypto operations using key K
SK	Session key
$\|\|$	Concatenation operation
$h(.)$	Secure-hash function
$\oplus$	XOR operation

Where K_{uh} is a shared secret between MU and HA. K is the private key of HA. Subsequently, HA stores $\{K_{uh}, ID_M\}$ in it's database and sends a message $\{K_{uh}, EID, h(.)\}$ to the MU via secure channel. Upon receiving the message from HA, MU freely selects a password PSW_M and computes:

$$EID^* = EID \oplus h(ID_M||PSW_M);\ K^*_{uh} = K_{uh} \oplus h(ID_M||PSW_M).$$

Hereafter, MU replaces $K_{uh},\ EID$ with the values of $K^*_{uh},\ EID^*$. Finally, the mobile user smartcard contains $\{K^*_{uh}, EID^*, h(.)\}$.

3.2 Mutual authentication and key agreement phase

Hither, a registered mobile user can roam into foreign networks to obtain desired services from a service provider FA. During this roaming process, the mobile user and foreign agent mutually authenticate each other with an assistance of the HA. The procedure of mutual authentication is outlined as follows:

Step 1: MU inputs his identity ID_M, password PSW_M through the card reader terminal. The device generates a nonce N_M and computes:

$$K_{uh} = K^*_{uh} \oplus h(ID_M||PSW_M)$$

$$EID = EID^* \oplus h(ID_M||PSW_M)$$

$$N_X = h(ID_M||K_{uh});\ V_1 = (EID||N_X||T_1||ID_M||K_{uh}).$$

Here T_1 is the timestamp used to prevent replay attacks. Finally, MU forms a message $M_1 = \{EID, N_X, ID_H, V_1, T_1\}$ to the FA.

Step 2: After hearing the message M_1, FA verifies whether current time falls within T_1 time. If the verification is unsuccessful, FA declines M_1. Otherwise, FA generates a nonce N_F and calculates:

$$N_Y = h(K_{FH}) \oplus N_F;$$

$$V_2 = h(EID||N_X||N_Y||T_2||K_{FH}||N_F).$$

After that FA forms a message $M_2 = \{EID, N_X, ID_F, V_1, T_1, N_Y, V_2, T_2\}$ to HA.

Step 3: HA receives the message from FA and checks the freshness of T_2. If the verification is fails, HA aborts the protocol. or else, HA computes:

$$N_F = h(K_{FH}) \oplus N_Y;\ V_2^* = h(EID||N_X||N_Y||T_2||K_{FH}||N_F).$$

Then, HA verifies whether $V_2^* \overset{?}{=} V_2$. If not, HA rejects the message M_2. Otherwise, HA decrypts $D_K(EID)$ to get the values ID_M and n_0. Next, HA computes $V_1^* = (EID||N_X||T_1||ID_M||K_{uh})$ and compares whether $V_1^* \overset{?}{=} V_1$. If the comparison is unsuccessful, HA aborts the authentication protocol. Otherwise, it generates a nonce n_1 and computes $FID = E_K(ID_M||n_1)$, $FID^* = FID \oplus h(ID_M||K_{uh})$. After that HA derives $N_M = h(ID_M||K_{uh}) \oplus N_X$ and computes the following:

$$N'_X = h(K_{uh}||ID_M||N_M) \oplus N_F \oplus n_0$$

$$N'_Y = h(K_{FH}||ID_F||N_F) \oplus N_M \oplus n_0$$

$$V_3 = h(N'_Y||N_F) \oplus K_{FH};\ V_4 = h(N'_X||FID^*||N_M) \oplus K_{uh}.$$

Finally, HA returns a authentication response message $M_3 = \{N'_X, N'_Y, V_3, V_4, FID^*\}$ to the FA.

Step 4: Upon accepting M_3, FA calculates $V_3^* = h(N'_Y||N_F) \oplus K_{FH}$ by using shared key K_{FH} and nonce N_F stored in the FA's verifier-table. Next, FA compares $V_3^* \overset{?}{=} V_3$. If the comparison fails, FA terminates the authentication protocol. Otherwise, it deduce the session key SK as follows:

$$N_M \oplus n_0 = h(K_{FH}||ID_F||N_F) \oplus N'_Y;$$

$$SK = N_M \oplus n_0 \oplus N_F.$$

At last, FA forms the message $M_4 = \{N'_X, V_4, FID^*\}$ to the MU.

Step 5: After hearing a message M_4, MU computes $V_4^* = h(N'_X||FID^*||N_M) \oplus K_{uh}$ and verifies $V_4^* \overset{?}{=} V_4$. If the verification succeeds, the mobile user derives $N_F \oplus n_0 = h(K_{uh}||ID_M||N_M) \oplus N'_X$ and computes $SK = N_M \oplus n_0 \oplus N_F$. Finally, MU and FA communicates each other using a session key SK. The mutual authentication and key agreement phase of Xu et al's scheme is depicted in Fig. 2.

3.3 Password renewal phase

To change the password in protocol [3], MU needs to input a old password PSW_M as well as new password PSW_M^*. After that, MU device computes:

$$K_{uh} = K_{uh}^* \oplus h(ID_M||PSW_M);$$

$$EID = EID^* \oplus h(ID_M||PSW_M)$$

$$K_{uh}^{**} = K_{uh} \oplus h(ID_M||PSW_M^*);$$

$$EID^{**} = EID \oplus h(ID_M||PSW_M^*).$$

Finally, the password is successfully changed and old values $\{K_{uh}^*, EID^*\}$ are replaced with new values $\{K_{uh}^{**}, EID^{**}\}$ in the MU's smartcard.

4 Security weaknesses of Xu et al. scheme

Hither, we analyse the deficiencies of the authentication and key establishment protocol presented in [3].

4.1 Vulnerable to stolen-verifier attack

In this protocol, FA and HA shares a static long-term key K_{FH} in advance. This shared secret key should be stored in both verifier tables of FA as well as HA [13]. In this regard, if the verifier table of FA or HA is stolen or leaked out by the attacker $\mathscr{A}$, then its possible to compute the session key belongs to legal mobile users. The attack on session key is described through the following steps:

1. An adversary $\mathscr{A}$ eavesdrops a authentication request message $M_2 = \{EID, N_X, ID_F, V_1, T_1, N_Y, V_2, T_2\}$ and its corresponding authentication response message $M_3 = \{N'_X, N'_Y, V_3, V_4, FID^*\}$ from the public communication channel.
2. $\mathscr{A}$ computes $N_F = N_y \oplus h(K_{FH})$ using the stolen K_{FH} and the parameter N_Y present in the eavesdropped authentication request M_2.
3. $\mathscr{A}$ derives $N_M \oplus n_0 = h(K_{FH}||ID_F||N_F) \oplus N'_Y$; $SK = N_M \oplus n_0 \oplus N_F$ using eavesdropped message M_3.
4. Finally, $\mathscr{A}$ is capable to compute a session-key $SK = N_M \oplus n_0 \oplus N_F$.

Thus, the mutual authentication protocol presented in [3] is vulnerable to stolen-verifier attack. Furthermore, the protocol is unable to provide a session-key security.

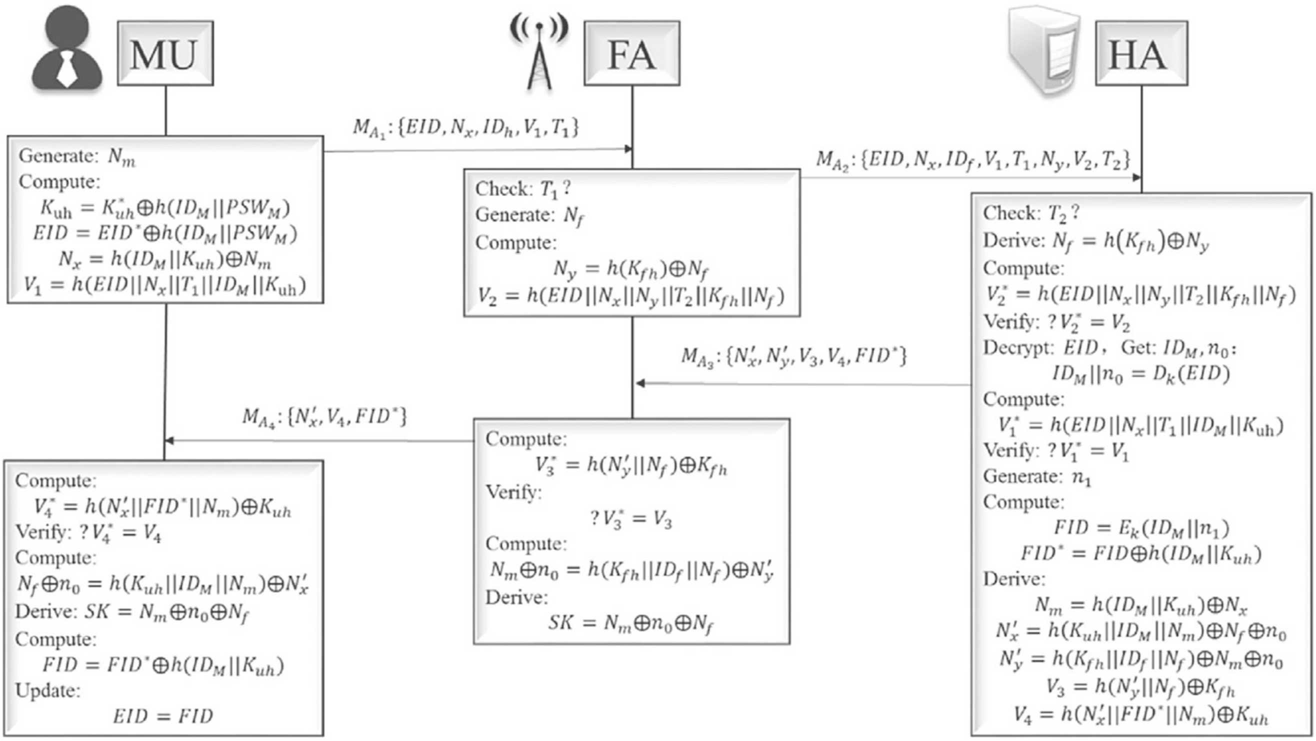

MU
FA
HA
M_{A_1}: {EID, N_x, ID_h, V_1, T_1}
Generate: N_m
Compute:
$K_{uh} = K^*_{uh} \oplus h(ID_M||PSW_M)$
$EID = EID^* \oplus h(ID_M||PSW_M)$
$N_x = h(ID_M||K_{uh}) \oplus N_m$
$V_1 = h(EID||N_x||T_1||ID_M||K_{uh})$
M_{A_2}: {EID, N_x, ID_f, V_1, T_1, N_y, V_2, T_2}
Check: T_1 ?
Generate: N_f
Compute:
$N_y = h(K_{fh}) \oplus N_f$
$V_2 = h(EID||N_x||N_y||T_2||K_{fh}||N_f)$
Check: T_2 ?
Derive: $N_f = h(K_{fh}) \oplus N_y$
Compute:
$V_2^* = h(EID||N_x||N_y||T_2||K_{fh}||N_f)$
Verify: $? V_2^* = V_2$
Decrypt: EID, Get: ID_M, n_0:
$ID_M||n_0 = D_k(EID)$
Compute:
$V_1^* = h(EID||N_x||T_1||ID_M||K_{uh})$
Verify: $? V_1^* = V_1$
Generate: n_1
Compute:
$FID = E_k(ID_M||n_1)$
$FID^* = FID \oplus h(ID_M||K_{uh})$
Derive:
$N_m = h(ID_M||K_{uh}) \oplus N_x$
$N_x' = h(K_{uh}||ID_M||N_m) \oplus N_f \oplus n_0$
$N_y' = h(K_{fh}||ID_f||N_f) \oplus N_m \oplus n_0$
$V_3 = h(N_y'||N_f) \oplus K_{fh}$
$V_4 = h(N_x'||FID^*||N_m) \oplus K_{uh}$
M_{A_3}: {N_x', N_y', V_3, V_4, FID^*}
M_{A_4}: {N_x', V_4, FID^*}
Compute:
$V_3^* = h(N_y'||N_f) \oplus K_{fh}$
Verify:
$? V_3^* = V_3$
Compute:
$N_m \oplus n_0 = h(K_{fh}||ID_f||N_f) \oplus N_y'$
Derive:
$SK = N_m \oplus n_0 \oplus N_f$
Compute:
$V_4^* = h(N_x'||FID^*||N_m) \oplus K_{uh}$
Verify: $? V_4^* = V_4$
Compute:
$N_f \oplus n_0 = h(K_{uh}||ID_M||N_m) \oplus N_x'$
Derive: $SK = N_m \oplus n_0 \oplus N_f$
Compute:
$FID = FID^* \oplus h(ID_M||K_{uh})$
Update:
$EID = FID$

Fig. 2 Xu et al's mutual authentication and key agreement phase

4.2 Fails to realize untraceability

An adversary $\mathscr{A}$ can neither trace MU identity, nor links mutual authentication session in which the same MU is involved. In mutual authentication phase of protocol [3], MU forms an authentication request $M_1 = \{EID, N_X, ID_H, V_1, T_1\}$ to FA, where the authentication parameters EID' and ID_H in M_1 is unchangeable for all authentication sessions. Thus, an attacker can easily identify and trace a mobile user location by linking different authentication sessions. As a result, this protocol fails to provide untraceability.

4.3 Vulnerable to impersonation attacks

In mutual authentication phase, an adversary $\mathscr{A}$ eavesdrops a login message $M_1 = \{EID, N_X, ID_H, V_1, T_1\}$ and authentication request $M_2 = \{EID, N_X, ID_F, V_1, T_1, N_Y, V_2, T_2\}$ from the public network. Then, an adversary selects a random nonce N_F' and calculates $N_Y = h(K_{FH}) \oplus N_F'$; $V_2' = h(EID||N_X||N_Y||T_2'||K_{FH}||N_F')$ using stolen key K_{FH}. Finally, an adversary $\mathscr{A}$ sends an authentication request message $M_2' = \{EID, N_X, ID_F, V_1, T_1, N_Y, V_2', T_2'\}$ to the HA. After accepting a message M_2', HA verifies the freshness of T_2' and computes

$$N_F = h(K_{FH}) \oplus N_Y;$$

$$V_2^* = h(EID||N_X||N_Y||T_2'||K_{FH}||N_F').$$

Then, HA verifies $V_2^* \overset{?}{=} V_2'$ and computes $V_1^* = (EID||N_X||T_1||ID_M||K_{uh})$ and compares whether $V_1^* \overset{?}{=} V_1$. If the comparison fails, HA terminates the protocol. Otherwise, it generates a nonce n_1 and computes $FID = E_K(ID_M||n_1)$, $FID^* = FID \oplus h(ID_M||K_{uh})$. After that HA derives $N_M = h(ID_M||K_{uh}) \oplus N_X$ and computes the following:

$$N_X' = h(K_{uh}||ID_M||N_M) \oplus N_F' \oplus n_0$$

$$N_Y' = h(K_{FH}||ID_F||N_F') \oplus N_M \oplus n_0$$

$$V_3 = h(N_Y'||N_F') \oplus K_{FH};\ V_4 = h(N_X'||FID^*||N_M) \oplus K_{uh}.$$

Finally, HA returns a authentication response message $M_3 = \{N_X', N_Y', V_3, V_4, FID^*\}$ to FA. Upon receiving the authentication response message M_3, an adversary computes $V_3^* = h(N_Y'||N_F') \oplus K_{FH}$ by using shared key K_{FH} and nonce N_F' stored in adversary's verifier-table. Next, $\mathscr{A}$ compares $V_3^* \overset{?}{=} V_3$. If the comparison fails,$\mathscr{A}$ terminates the authentication protocol. Otherwise, he/she derive the session key SK as follows: $N_M \oplus n_0 = h(K_{FH}||ID_F||N_F') \oplus N_Y'$; $SK = N_M \oplus n_0 \oplus N_F$. At last, $\mathscr{A}$ forms the message $M_4 = \{N_X', V_4, FID^*\}$ to MU. After receiving M_4, MU successfully authenticates HA as well as malicious FA. Finally, MU computes $N_F \oplus n_0 = h(K_{uh}||ID_M||N_M) \oplus N_X'$ and derives $SK = N_M \oplus n_0 \oplus N_F$. Hereafter, MU believes that an adversary $\mathscr{A}$ is a legal FA. Therefore, in this protocol an adversary $\mathscr{A}$ impersonate as FA to cheat a mobile user MU and HA.

4.4 Vulnerable to denial of service attack

In login phase, Xu et al.'s authentication protocol is unable to validate the existing password. If an adversary $\mathscr{A}$ gets the lost smartcard or he/she may pick out the MU's smartcard for a short period of time. Then, $\mathscr{A}$ can launch a denial of service attacks. It is described below:

1. An adversary $\mathscr{A}$ with lost/stolen smartcard, enters a random identity ID_M and password PSW_M.
2. MU device computes:

   $$K_{uh} = K_{uh}^{*} \oplus h(ID_M||PSW_M);$$

   $$EID = EID^{*} \oplus h(ID_M||PSW_M)$$

3. After, MU generates a random nonce N_M and computes:

   $$N_X = h(ID_M||K_{uh});\ V_1 = (EID||N_X||T_1||ID_M||K_{uh}).$$

4. Later, $\mathscr{A}$ may attempt to create and sends invalid login requests $M_1' = \{EID, N_X, ID_H, V_1, T_1\}$ by entering fake credentials. It could be detected only at HA not at the MU process.
5. An adversary can repeat this process again and again to overload requests in the network, which makes authentication system busy and restrains accessibility for the legal mobile users. As a result, the above authentication protocol causes denial of service attack.

4.5 Vulnerable to privileged-insider attack

The privileged-insider of the home agent would gets mobile user information $\{ID_M, K_{uh}\}$ from HA's database. Assume that the privileged-insider being an adversary tries to impersonate a legal user with lost/stolen smartcard. Through power analysis $\mathscr{A}$ extracts the sensitive information $\{EID^{*}, K_{uh}^{*}, h(.)\}$ from the smart card. After that $\mathscr{A}$ computes $h(ID_M||PSW_M) = K_{uh}^{*} \oplus K_{uh}$ and $EID = EID^{*} \oplus h(ID_M||PSW_M)$. Subsequently, an adversary $\mathscr{A}$ computes:

$$N_X = h(ID_M||K_{uh});\ V_1' = (EID||N_X||T_1'||ID_M||K_{uh}).$$

Finally, an adversary forms a valid login request $M_1' = \{EID, N_X, ID_H, V_1', T_1'\}$ to the FA. Upon receiving M_1', FA generates N_F and computes:

$$N_Y = h(K_{FH}) \oplus N_F';\ V_2' = h(EID||N_X||N_Y||T_2'||K_{FH}||N_F').$$

Then, FA makes a valid authentication request $M_2 = \{EID, N_X, ID_F, V_1', T_1', N_Y, V_2', T_2'\}$ to the HA. Hereafter, HA successfully authenticates FA, MU and believes that an adversary is a legal MU. Hence, it is clear that the above protocol is vulnerable to an insider attack.

4.6 Wrong password cannot be detected quickly

In Xu et al.'s protocol, MU device is unable to validate the user credentials such as identity ID_M and password PSW_M. Therefore, an unauthorized user or an adversary with lost/stolen smart card could enter the fake login credentials and forms a login request $M_1' = \{EID, N_X, ID_H, V_1', T_1'\}$. This vulnerability can be detected only at the HA, which makes the authentication system inefficient. Therefore, designing of a local password verification mechanism to detect wrong passwords quickly is very essential in the authentication systems.

4.7 Clock synchronization problem

The above protocol employs timestamps mechanism to prevent replay attacks. The timestamps make use of clocks are not suitable for real time mobility environments, the reason is that the additional clocks used at the mobile agents may not be synchronized always. Even a small time variation in clocks could result in refusal of authentication messages in the network [36]. Assume that mobile jammers are deployed to halt mobile devices from transmitting or receiving signals in the places where a mobile devices would be particularly disruptive. In this context, the above authentication protocol transmits a sequence of messages $\{M_1, M_2\}$ associated with timestamps $\{T_1, T_2\}$ across MU, FA, and HA. Due to a network failure or deployment of mobile jammers, the transmission/reception of authentication requests $\{M_1, M_2\}$ will be delayed. After receiving these requests, each entity in the network, verifies the freshness of time-stamp by comparing with it's current time-stamp and expected time interval T. If the verification is unsuccessful, the agents MU,FA, and HA simply rejects the authentication requests. Thus, the mutual authentication process is denied, therefore, the above protocol suffers from clock-synchronization problem.

4.8 Insecure password renewal phase

In the password renewal phase, there is no provision for validating the existing password. Assume that, the attacker $\mathscr{A}$ with stolen/lost smart-card inputs the random password PSW_M, new password PSW_M^{*} and invokes the password renewal phase to change the legal MU's password. After that, the device computes:

$$K_{uh} = K_{uh}^{*} \oplus h(ID_M||PSW_M);$$

$$EID = EID^{*} \oplus h(ID_M||PSW_M)$$
$$K_{uh}^{**} = K_{uh} \oplus h(ID_M||PSW_M^{*});$$
$$EID^{**} = EID \oplus h(ID_M||PSW_M^{*}).$$

Finally, the password is successfully changed and old values $\{K_{uh}^{*}, EID^{*}\}$ are replaced with new values $\{K_{uh}^{**}, EID^{**}\}$

in the smartcard. Hereafter, the legal MU is unable to compute the correct session key with FA any more. Because, an adversary $\mathscr{A}$ is updated his own authentication information in the smart-card. Hence, the password renewal phase is insecure.

5 The proposed scheme

To address the existing security flaws in global mobility networks, we present a robust authentication protocol that resist all security weakness of Xu et al. protocol and improves the computational efficiency. The proposed protocol comprises of: (1) registration phase, (2) login and authentication phase, and (3) the password change phase. Initially, each FA will acquire a dynamic Diffie-Hellman secret-key SK_{FA} from home agent, where $SK_{FA} = h(ID_F||SK_{HA})$.

5.1 Registration phase

In this scenario, a new mobile user MU gets registered with the home agent. The procedure of the mobile user registration is described through the following steps:

R1 : MU selects the identity ID_M and password PSW_M, generates a nonce R_N, and computes $RID = h(ID_M||R_N)$. Afterwards, sends RID to HA via secure channel.

R2 : After accepting RID from MU, HA computes $HID = h(RID||SK_{HA})$, where SK_{HA} is a secret key of HA. Then, HA initializes the counter $K_{MU} = 0$ for MU and keeps $\{RID, K_{MU}\}$ in the database. Finally, HA securely returns $\{HID, K_{MU}, h(.)\}$ to the MU.

R3 : The mobile user MU computes:

$$SP = HID \oplus h(PSW_M||R_N)$$

$$PV = h(ID_M||PSW_M||R_N).$$

MU replaces HID with SP in the smart-card. Finally, MU stores the values $\{SP, PV, R_N, K_{MU}, h(.)\}$ in the smart-card. The registration phase is depicted in Table 2.

5.2 Login and mutual authentication phase

The authentication phase is carried out when a mobile user MU moves into the foreign network (FN) to obtain roaming services assigned by the FA through HA. In this scenario, accomplishment of mutual authentication between MU, FA, and HA is very crucial. Login and authentication procedure is summarized in Table 3.

LA1 : $MU \longrightarrow FA : M_{MF} = \{A_M, V_1, ID_H\}$

MU first inserts the smart card into a card reader terminal, and inputs the identity ID_M and password PSW_M. Then, the device computes $PV^* = h(ID_M||PSW_M||R_N)$ and verifies whether the condition $PV^* = PV$ holds or not. If the verification fails, the device rejects the entered credentials. Otherwise, the legitimacy of user is proved. Then, MU device generates the nonce N_M and computes the following:

$$HID = SP \oplus h(PSW_M||R_N)$$

Table 2 Registration phase of the proposed protocol

Mobile User (MU)		Home Agent (HA)
Select ID_M, PSW_M, R_N Compute:$RID = h(ID_M\|\|R_N)$		
	$\xrightarrow{\boldsymbol{R_1 = \{RID\}}}$	
		$HID = h(RID\|\|SK_{HA})$ HA keeps $\{RID, K_{MU}\}$ in its database
	$\xleftarrow{\boldsymbol{R_2 = \{HID, K_{MU}, h(.)\}}}$	
$SP = HID \oplus h(PSW_M\|\|R_N)$ $PV = h(ID_M\|\|PSW\|\|R_N)$ MU stores $\{SP, PV, R_N, K_{MU}\}$ in the smart card		

Table 3 Login and mutual authentication phase of the proposed protocol

Mobile User (MU)	Foreign Agent (FA)	Home Agent (HA)
Generate N_M Compute:$HID = SP \oplus h(PSW_M \Vert R_N)$ $A_M = h(ID_M \Vert R_N) \oplus N_M$ $V_1 = h(HID \Vert K_{MU}) \oplus N_M$		
$\xrightarrow{\boldsymbol{M_{MF} = \{A_M, V_1, ID_H\}}}$		
	Generate N_F $B_M = h(A_M \Vert SK_{FA}) \oplus N_F$ $V_2 = h(B_M \Vert SK_{FA} \Vert V_1)$.	
	$\xrightarrow{\boldsymbol{M_{FH} = \{ID_F, B_M, V_1, V_2\}}}$	
		$SK_{FA} = h(ID_F \Vert SK_{HA})$ $V_2^* = h(B_M \Vert SK_{FA} \Vert V_1)$ $V_2^* \stackrel{?}{=} V_2$ $HID^* = h(RID \Vert SK_{HA})$ $N_M^* = h(HID^* \Vert K_{MU}) \oplus V_1$ $V_1^* = h(HID^* \Vert K_{MU}) \oplus N_M^*$ $V_1^* \stackrel{?}{=} V_1$ $A_M^* = h(ID_M \Vert R_N) \oplus N_M^*$ $N_F = h(A_M^* \Vert SK_{FA}) \oplus B_M$ $N_M' = h(HID^* \Vert N_M^*) \oplus N_F$ $V_3 = h(ID_H \Vert A_M^* \Vert SK_{FA})$ $V_4 = h(HID^* \Vert ID_F \Vert K_{MU})$ Update $K_{MU} = K_{MU} + 1$
		$\xleftarrow{\boldsymbol{M_{HF} = \{N_M', V_3, V_4\}}}$
	$V_3^* = h(ID_H \Vert A_M \Vert SK_{FA})$ $V_3^* \stackrel{?}{=} V_3$ $SK = h(N_F \Vert A_M \Vert ID_H)$	
	$\xleftarrow{\boldsymbol{M_{FM} = \{N_M', V_4\}}}$	
$V_4^* = h(HID \Vert ID_F \Vert K_{MU})$ $V_4^* \stackrel{?}{=} V_4$ $N_F = N_M' \oplus h(HID \Vert N_M)$ $SK = h(N_F \Vert A_M \Vert ID_H)$ Update $K_{MU} = K_{MU} + 1$		

$$A_M = h(ID_M||R_N) \oplus N_M$$

$$V_1 = h(HID||K_{MU}) \oplus N_M$$

Finally, MU sends the login request $M_{MF} = \{A_M, V_1, ID_H\}$ to the FA.

LA2 : $FA \rightarrow HA : M_{FH} = \{ID_F, B_M, V_1, V_2\}$

Upon receiving the message M_{MF}, FA generates a nonce N_F and computes:

$$B_M = h(A_M||SK_{FA}) \oplus N_F;\ V_2 = h(B_M||SK_{FA}||V_1).$$

After that, FA keeps $\{N_F, A_M\}$ and sends the authentication request $M_{FH} = \{ID_F, B_M, V_1, V_2)\}$ to the HA.

LA3 : $HA \rightarrow FA : M_{HF} = \{N'_M, V_3, V_4\}$

HA receives the authentication request M_{FH} from FA and verifies the existence of ID_F. If its holds, HA finds a secret key $SK_{FA} = h(ID_F||SK_{HA})$ corresponding to ID_F. Next, HA computes $V_2^* = h(B_M||SK_{FA}||V_1)$ and compares with received V_2. If the comparison fails, HA rejects the authentication request. Otherwise, HA successfully authenticates FA, retrieves the values $\{RID, K_{MU}\}$ from its database and computes the following:

$$HID^* = h(RID||SK_{HA});\ N_M^* = h(HID^*||K_{MU}) \oplus V_1$$

$$V_1^* = h(HID^*||K_{MU}) \oplus N_M^*;\ V_1^* \stackrel{?}{=} V_1.$$

If the above condition fails, HA terminates the authentication process. Otherwise, HA authenticates MU and computes the following:

$$A_M^* = h(ID_M||R_N) \oplus N_M^*;\ N_F = h(A_M^*||SK_{FA}) \oplus B_M$$

$$N'_M = h(HID^*||N_M^*) \oplus N_F$$

$$V_3 = h(ID_H||A_M^*||SK_{FA})$$

$$V_4 = h(HID^*||ID_F||K_{MU}).$$

Finally, HA updates the counter $K_{MU} = K_{MU} + 1$ and returns the authentication response $M_{HF} = \{N'_M, V_3, V_4\}$ to the FA.

LA4 : $FA \rightarrow MU : M_{FM} = \{N'_M, V_4\}$

Upon accepting a message M_{HF} from HA, FA computes $V_3^* = h(ID_H||A_M||SK_{FA})$ and verifies the condition $V_3^* \stackrel{?}{=} V_3$. If the comparison is unsuccessful, FA terminates the authentication process. Otherwise, FA successfully authenticates HA, MU, and computes the session key $SK = h(N_F||A_M||ID_H)$. Further, FA returns the message $M_{FM} = \{N'_M, V_4\}$ to the MU.

LA5 : MU receives the authentication response M_{FM} from FA, then computes $V_4^* = h(HID||ID_F||K_{MU})$ and verifies whether $V_4^* \stackrel{?}{=} V_4$. If not, MU aborts the authentication process. Otherwise, MU successfully authenticates FA, HA, and derives the session key as follows:

$$N_F = N'_M \oplus h(HID||N_M);\ SK = h(N_F||A_M||ID_H).$$

Finally, MU updates $K_{MU} = K_{MU} + 1$ in the smart-card.

5.3 Password change phase

MU can use the password change phase to change his default password without the assistance of HA. The steps involved in the password change phase are as follows:

Step 1: MU inputs his identity ID_M, password PSW_M, and submit the password change request through terminal.

Step 2: The mobile user smart card computes $PV^* = h(ID_M)||PSW_M||R_N)$ and verifies whether $PV^* \stackrel{?}{=} PV$. If this condition fails, password change request is rejected. Otherwise, the legality of MU is proved. Next, the smart card derives $HID = SP \oplus h(PSW_M||R_N)$.

Step 3: After that, MU enters the new password PSW_M^* and computes the following:

$$PV_N = h(ID_M||PSW_M^*||R_N)$$

$$SP_N = HID \oplus h(PSW_M^*||R_N).$$

Step 4: At last, MU replaces old values $\{PV, SP\}$ with new values of $\{PV_N, SP_N\}$, respectively. Finally, the smart-card contains $\{PV_N, SP_N, R_N, K_{MU}\}$.

The execution road map of the proposed authentication scheme and interrelationship between registration and mutual authentication phase is depicted in Fig. 3.

6 Security analysis

This section describes the rigorous informal security analysis of the proposed protocol. In addition, we proved that the proposed protocol resist against various attacks in global mobility networks. Here, we assume that the adversary $\mathscr{A}$ wish to break the authentication protocol. We record the difficulties that $\mathscr{A}$ faces to break the security of the proposed protocol.

6.1 Withstand stolen-verifier attack

The proposed protocol make use of dynamic Diffie Hellman key exchange mechanism to share the secret key between HA and FA, which is infeasible for an adversary $\mathscr{A}$ to obtain the dynamic key $SK_{FA} = (ID_F||SK_{HA})$ from the public network. In addition, the entities FA and HA does not keep

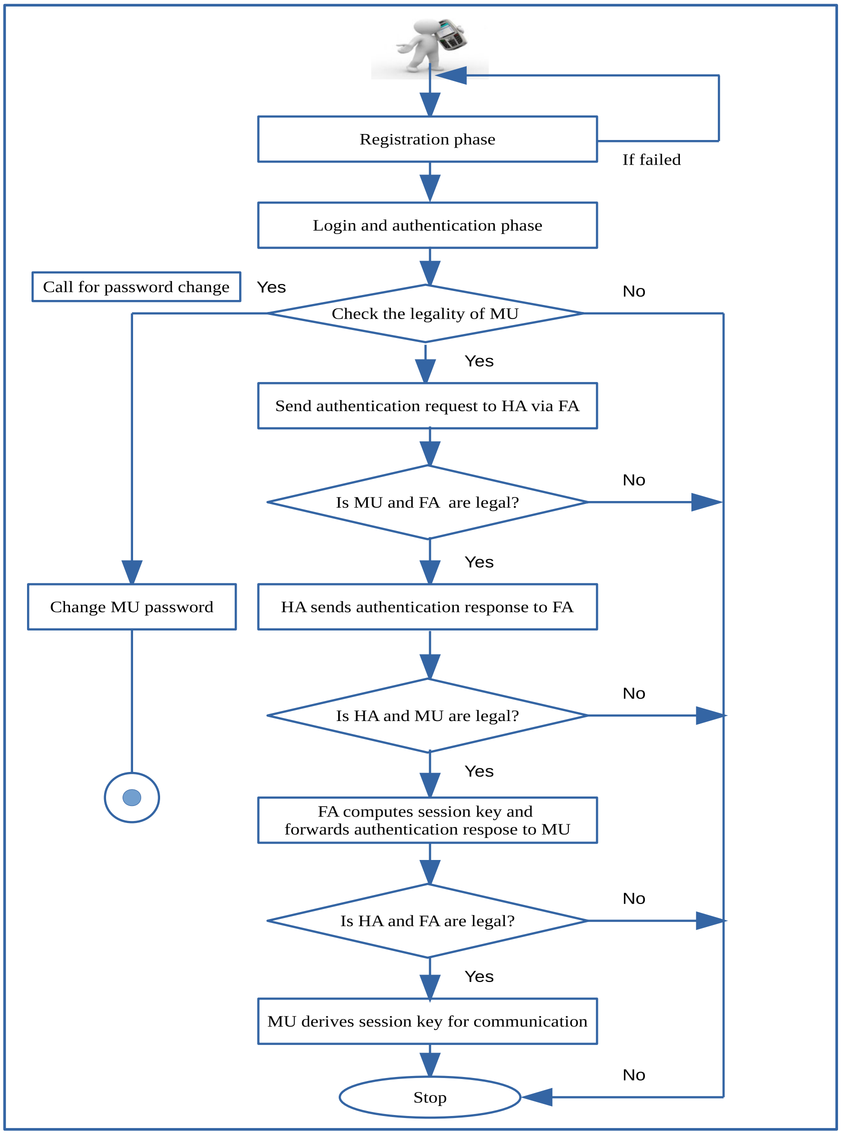

Registration phase
If failed
Login and authentication phase
Call for password change
Yes
Check the legality of MU
No
Yes
Send authentication request to HA via FA
Is MU and FA are legal?
No
Yes
Change MU password
HA sends authentication response to FA
Is HA and MU are legal?
No
Yes
FA computes session key and forwards authentication respose to MU
Is HA and FA are legal?
No
Yes
MU derives session key for communication
Stop
No

Fig. 3 The execution road map of the proposed protocol

any password verifier information in its database. Thus, an adversary have no clue about the users passwords or secret keys related to HA and FA. As a result, the proposed protocol withstand stolen-verifier attack.

6.2 User anonymity and untraceability

Throughout the roaming process, the proposed mutual authentication protocol is remain anonymous across insecure communication channels as described here. In registration phase, MU's identity ID_M is concealed with random nonce that is $RID = h(ID_M||R_N)$, and makes a registration request RID to HA. In this scenario, an adversary $\mathscr{A}$, including legitimate HA unable to deduce the user identity ID_M. In authentication phase, we assume that an adversary $\mathscr{A}$ eavesdrops transmitted messages $M_{MF} = \{A_M, V_1, ID_H\}$, $M_{FH} = \{ID_F, B_M, V_1, V_2\}$, $M_{HF} = \{N'_M, V_3, V_4\}$, $M_{FM} = \{N'_M, V_4\}$ between MU, FA and HA through public channel. We could observe that the above messages are not disclosing MU's identity ID_M. Thus, the anonymity of MU is ensured. If an adversary $\mathscr{A}$ wants to trace MU using an intercepted information from the public channel, then he/she must discover a relationship between communications. In the proposed protocol, messages $M_{MF}, M_{FH}, M_{HF}, M_{FM}$ shared between MU, FA and HA are variable in every session due to employment of nonce's N_M, N_F which means that the values have no relationship with another one and random numbers are unobtainable. This scenario makes communication untraceable and anonymous to an adversaries.

6.3 Resilience to impersonation attacks

Assume that the adversary $\mathscr{A}$ eavesdrops the messages $M_{MF}, M_{FH}, M_{HF}, M_{FM}$ from the public network and tries to impersonate as a legal MU, FA or HA to access the desired services. In this scenario, an adversary would face different challenges, which are outlined here. In order to impersonate the legal mobile user, the adversary $\mathscr{A}$ should have ID_M and PSW_M. In this protocol, mobile user credentials like identity and user passwords are not transmitted in the messages $M_{MF}, M_{FH}, M_{HF}, M_{FM}$ through public channels. Even though, an adversary $\mathscr{A}$ gets the stolen/lost smart card, then he/she is unable to form a valid login request $M_{MF} = \{A_M, V_1, ID_H\}$ to cheat FA and HA. Since an adversary cannot compute the values $HID = SP \oplus h(PSW_M||R_N)$ $A_M = h(ID_M||R_N) \oplus N_M$ without MU's identity and password. Thus, the protocol resist against MU impersonation attack. Without knowledge of FA's random nonce N_F and its secret key SK_{FA}, an adversary cannot forge the message $M_{FH} = \{ID_F, B_M, V_1, V_2\}$. Further, HA and FA make use of dynamic Diffie-Hellman keys to share the secret SK_{FA}, its difficult to cheat either of them due to the intractability of Diffie-Hellman problem. Thus, the proposed protocol resist against FA impersonation attack. Further, Without knowing a HA's secret S_{HA} an adversary cannot compute the message $M_{HF} = \{N'_M, V_3, V_4\}$ to cheat FA and MU. Because, $\mathscr{A}$ cannot derive N_F, A_M to compute session-key $SK = h(N_F||A_M||ID_H)$. Hence, the proposed protocol withstand against HA impersonation attack.

6.4 Protection against denial of service attack

In order to secure against denial-of-service attacks, MU computes $PV^* = h(ID_M||PSW_M||R_N)$ and checks whether $PV^* = PV$ or not. If the comparison succeeds, legality of the mobile user is proved. Otherwise, denies the access to the system. Assume that an unauthorized user obtains the lost/stolen smart-card and attempts to logging into the authentication system by entering his random identity and password. In the proposed protocol, this attack is successfully detected at mobile user side by verifying $PV^* = PV$, this process eliminates the invalid requests sending to entities FA and HA. As a result, the proposed protocol prevents denial of service attack.

6.5 Resilience to privileged-insider attack

In the registration phase as well as mutual authentication phase of the proposed protocol, the users need not to send their plain-text passwords to HA. Therefore, an insider of the HA is unable to get the MU's password information. Assume that the privileged inside adversary of the HA collects all information $\{SP, PV, K_{MU}, R_N\}$ from a stolen/lost smart-card using a power analysis attack. In order to guess a correct password PSW_M of the MU from PV, an adversary needs to know ID_M, R_N. The identity ID_M is neither stored in smart-card or nor in HA's database. Thus, it is computationally infeasible for an adversary to guess PSW_M correctly. As a result, an insider attack is prevented in the proposed protocol.

6.6 Detects the wrong password quickly

The proposed protocol employs the local password verification mechanism. In this process MU device validates the user credentials such as identity ID_M and password PSW_M, before making authentication requests to FA and HA. Here, an adversary $\mathscr{A}$ or an unauthorized user with stolen/lost smart-card, cannot computes the correct $PV^* = h(ID_M||PSW_M||R_N)$. Because, an adversary should have knowledge of MU's ID_M, PSW_M to succeed the verification process $PV^* = PV$ during login phase. Thus, the proposed protocol is designed to avoid unauthorized access by verifying mobile user's passwords locally.

6.7 The clock-synchronization problem

The security protocols making use of time-stamps to furnish message freshness is suffers from replay attacks as the transmission delay is un-predictable in wireless and mobility environments. Thus, the proposed protocol does not use the time-stamps or additional clocks to synchronize the time between communicating parities such as MU, FA and HA. Besides, we used counter K_{MU} to prevent the replay attacks. As a result, the proposed protocol resist against clock-synchronization problem.

6.8 Secure password change phase

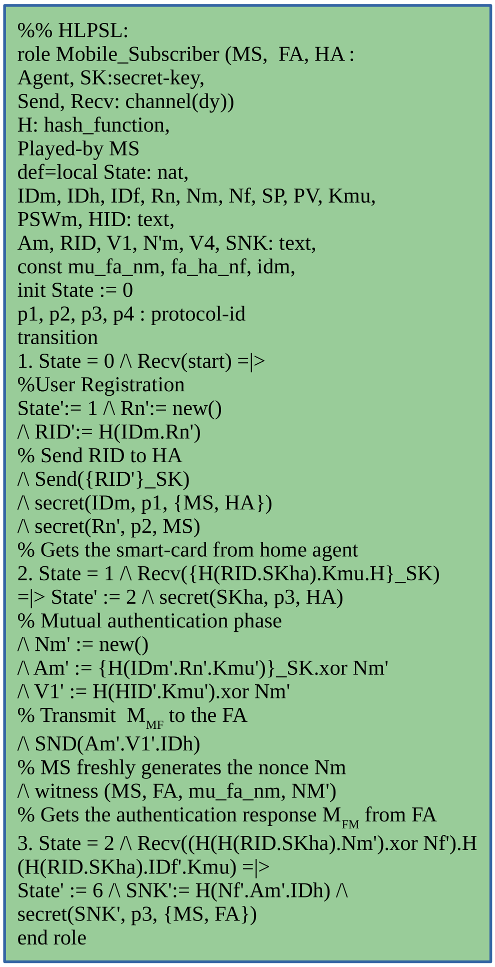
In order to change a password, user needs to input his old identity ID_M and password PSW_M. Then, local password verification process validates the user credentials to deny access of an unauthorized users. In this process, if the the legality of user is proved, the mobile user freely selects his login credentials such as ID_M, PSW_M without the assistance of HA. In other words, the proposed protocol permits user to change PSW_M in a short period of time. Because, the mobile user need not to go through entire login procedure, which obviously reduces the time and minimizes the computational complexity of the system. Thus, the proposed protocol employs secure password change mechanism as well as user-friendly.

7 Formal security verification using avispa tool

In this part, we carry out the formal security verification of the proposed protocol through the widely-accepted AVISPA (Automated Validation of Internet Security Protocols and Applications) tool [37]. Its one of the push button tool consists of four backends: OFMC (On the fly Model Checker), SATMC (SAT based Model Checker), CL-AtSe (Constraint-Logic based Attack Searcher), and TA4SP (Tree Automata based on Automatic Approximations for the Analysis of Security Protocols). In AVIPSA, the authentication protocol is implemented in HLPSL (High-Level Protocol Specification Language). It is a role-oriented language, the roles system specifies each participant, compositions for signifying communication sessions. The adversary $\mathscr{A}$ in HLPSL is specified using a security model called Dolev-Yao (DY) model. Thus, an intruder can take part in an authentic role.

The code implemented in HLPSL is translated into intermediate format (IF) using HLPSL2IF translator. This IF is fed into one of the AVISPA backends in order to produce a output format known as OF. It consists of summary, name of the protocol, goal, statistics and back end details. Finally, with help of output format the AVISPA tool formally verifies whether the given security protocol is safe or unsafe to thwart passive and active attacks in a networked environment. At first, $Mobile\_subscriber$ gets a start-signal and changes it's state from 0 to 1. A particular variable *State* is used to maintain a state value. In registration phase, mobile subscriber sends the registration request RID using Send() operation to HA via the secure channel. Further, mobile subscriber collects a smart-card comprising the parameters $\{HID, K_{MU}, h(.)\}$ from HA using Recv() operation. In login and mutual authentication phase, MS sends $M_{MF} = \{ID_H, A_M, V_1\}$ to FA through a open channel, then FA returns $M_{FM} = \{N'_M, V_4\}$ to MS through the insecure communication channel. The mobile subscriber role using HLPSL is shown in Fig. 4. The role system defines principals, a number of sessions and associated basic roles for security protocols. Some HLPSL supported basic types are: *agent, const, symmetric key, public key, text and nat* for natural numbers. The declaration $played\_by\ MS$ denotes that an agent MS plays the role. In HLPSL transition of the form $P = | > Q$ connects an event P and the activity Q. The goal $secrecy\_of\ \ S$ states that the variable S remains secure.

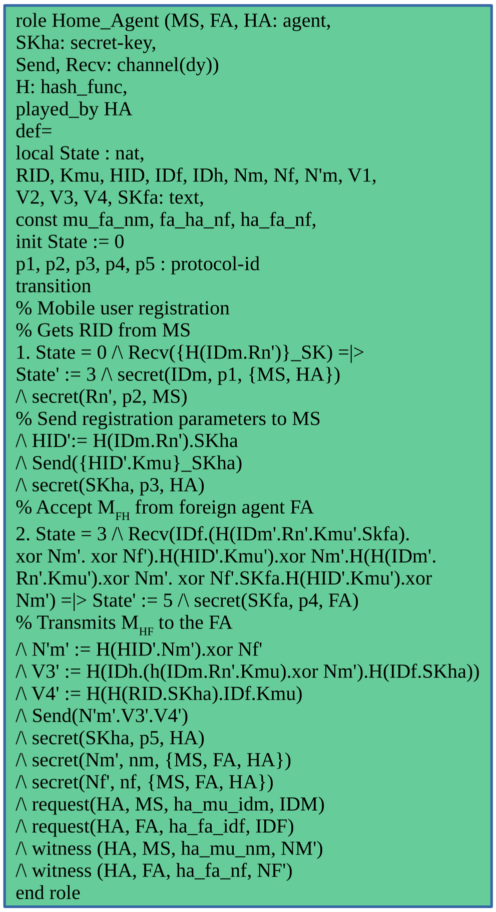
We have implemented communication entity roles in HLPSL namely, $Mobile\_Subscriber$, $Home\_Agent$ and $Foreign\_Agent$ for MU, FA, and HA along with session, environment and goal roles. The proposed protocol in HLPSL implementation covers registration phase, login and authentication phase. The declaration statement $(IDM, s1, MS, HA)$ denotes that an identity ID_M is only known to mobile subscriber, home agent which is identified using $S1$. HA authenticates MU through HID, and MU authenticate HA by using MU's random nonce through N_M. $witness(MS, FA, mu\_ha\_nm, NM')$ specifies the authentication property of agent MS has freshly generated nonce N_M for an agent FA. Likewise, home and foreign agent roles of the proposed protocol under HLPSL implementation is shown in Figs. 5 and 6, respectively.



```
%% HLPSL:
role Mobile_Subscriber (MS,  FA, HA :
Agent, SK:secret-key,
Send, Recv: channel(dy))
H: hash_function,
Played-by MS
def=local State: nat,
IDm, IDh, IDf, Rn, Nm, Nf, SP, PV, Kmu,
PSWm, HID: text,
Am, RID, V1, N'm, V4, SNK: text,
const mu_fa_nm, fa_ha_nf, idm,
init State := 0
p1, p2, p3, p4 : protocol-id
transition
1. State = 0 ∧ Recv(start) =|>
%User Registration
State':= 1 ∧ Rn':= new()
∧ RID':= H(IDm.Rn')
% Send RID to HA
∧ Send({RID'}_SK)
∧ secret(IDm, p1, {MS, HA})
∧ secret(Rn', p2, MS)
% Gets the smart-card from home agent
2. State = 1 ∧ Recv({H(RID.SKha).Kmu.H}_SK)
=|> State' := 2 ∧ secret(SKha, p3, HA)
% Mutual authentication phase
∧ Nm' := new()
∧ Am' := {H(IDm'.Rn'.Kmu')}_SK.xor Nm'
∧ V1' := H(HID'.Kmu').xor Nm'
% Transmit  M_MF to the FA
∧ SND(Am'.V1'.IDh)
% MS freshly generates the nonce Nm
∧ witness (MS, FA, mu_fa_nm, NM')
% Gets the authentication response M_FM from FA
3. State = 2 ∧ Recv((H(H(RID.SKha).Nm').xor Nf').H
(H(RID.SKha).IDf'.Kmu) =|>
State' := 6 ∧ SNK':= H(Nf'.Am'.IDh) ∧
secret(SNK', p3, {MS, FA})
end role
```

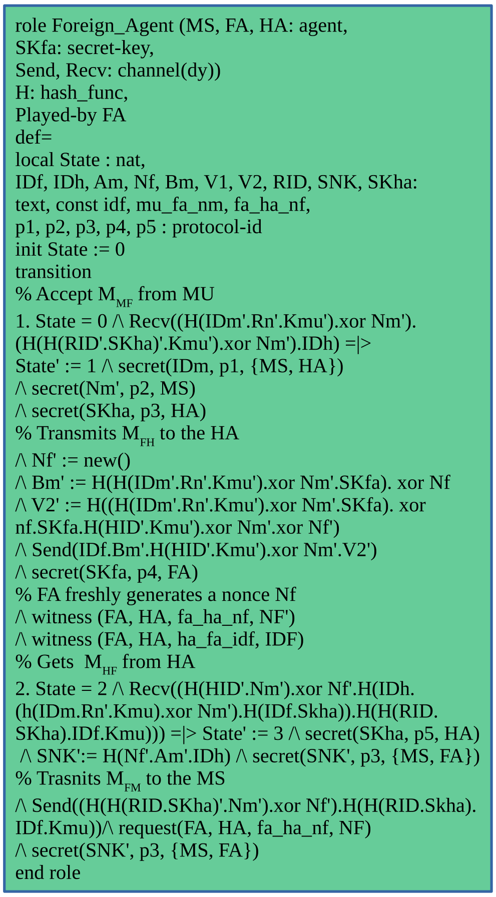
Fig. 4 HLPSL code for implementing the MU process

The authentication among FA and HA is achieved through FA's identity ID_F, and The authentication between FA and HA is achieved through N_F. During the authentication process, it is essential to promise the confidentiality of the parameters $\{ID_M, N_M, N_F, SK\}$ transmitted among the entities MS, FA and HA. The role specification for a session, environment and goal is depicted in Fig. 7. The session role gives the brief explanation about how to combine communicating parties in the specific role. The proposed session is a composition of mobile subscriber, home agent, and foreign agent roles, respectively. The environment role describes a environment at which the proposed authentication protocol



```
role Home_Agent (MS, FA, HA: agent,
SKha: secret-key,
Send, Recv: channel(dy))
H: hash_func,
played_by HA
def=
local State : nat,
RID, Kmu, HID, IDf, IDh, Nm, Nf, N'm, V1,
V2, V3, V4, SKfa: text,
const mu_fa_nm, fa_ha_nf, ha_fa_nf,
init State := 0
p1, p2, p3, p4, p5 : protocol-id
transition
% Mobile user registration
% Gets RID from MS
1. State = 0 /\ Recv({H(IDm.Rn')}_SK) =|>
State' := 3 /\ secret(IDm, p1, {MS, HA})
/\ secret(Rn', p2, MS)
% Send registration parameters to MS
/\ HID':= H(IDm.Rn').SKha
/\ Send({HID'.Kmu}_SKha)
/\ secret(SKha, p3, HA)
% Accept M_FH from foreign agent FA
2. State = 3 /\ Recv(IDf.(H(IDm'.Rn'.Kmu'.Skfa).
xor Nm'. xor Nf').H(HID'.Kmu').xor Nm'.H(H(IDm'.
Rn'.Kmu').xor Nm'. xor Nf'.SKfa.H(HID'.Kmu').xor
Nm') =|> State' := 5 /\ secret(SKfa, p4, FA)
% Transmits M_HF to the FA
/\ N'm' := H(HID'.Nm').xor Nf'
/\ V3' := H(IDh.(h(IDm.Rn'.Kmu).xor Nm').H(IDf.SKha))
/\ V4' := H(H(RID.SKha).IDf.Kmu)
/\ Send(N'm'.V3'.V4')
/\ secret(SKha, p5, HA)
/\ secret(Nm', nm, {MS, FA, HA})
/\ secret(Nf', nf, {MS, FA, HA})
/\ request(HA, MS, ha_mu_idm, IDM)
/\ request(HA, FA, ha_fa_idf, IDF)
/\ witness (HA, MS, ha_mu_nm, NM')
/\ witness (HA, FA, ha_fa_nf, NF')
end role
```

Fig. 5 HLPSL code for implementing the HA process



```
role Foreign_Agent (MS, FA, HA: agent,
SKfa: secret-key,
Send, Recv: channel(dy))
H: hash_func,
Played-by FA
def=
local State : nat,
IDf, IDh, Am, Nf, Bm, V1, V2, RID, SNK, SKha:
text, const idf, mu_fa_nm, fa_ha_nf,
p1, p2, p3, p4, p5 : protocol-id
init State := 0
transition
% Accept M_MF from MU
1. State = 0 /\ Recv((H(IDm'.Rn'.Kmu').xor Nm').
(H(H(RID'.SKha)'.Kmu').xor Nm').IDh) =|>
State' := 1 /\ secret(IDm, p1, {MS, HA})
/\ secret(Nm', p2, MS)
/\ secret(SKha, p3, HA)
% Transmits M_FH to the HA
/\ Nf' := new()
/\ Bm' := H(H(IDm'.Rn'.Kmu').xor Nm'.SKfa). xor Nf
/\ V2' := H((H(IDm'.Rn'.Kmu').xor Nm'.SKfa). xor
nf.SKfa.H(HID'.Kmu').xor Nm'.xor Nf')
/\ Send(IDf.Bm'.H(HID'.Kmu').xor Nm'.V2')
/\ secret(SKfa, p4, FA)
% FA freshly generates a nonce Nf
/\ witness (FA, HA, fa_ha_nf, NF')
/\ witness (FA, HA, ha_fa_idf, IDF)
% Gets  M_HF from HA
2. State = 2 /\ Recv((H(HID'.Nm').xor Nf'.H(IDh.
(h(IDm.Rn'.Kmu).xor Nm').H(IDf.Skha)).H(H(RID.
SKha).IDf.Kmu))) =|> State' := 3 /\ secret(SKha, p5, HA)
 /\ SNK':= H(Nf'.Am'.IDh) /\ secret(SNK', p3, {MS, FA})
% Trasnits M_FM to the MS
/\ Send((H(H(RID.SKha)'.Nm').xor Nf').H(H(RID.Skha).
IDf.Kmu))/\ request(FA, HA, fa_ha_nf, NF)
/\ secret(SNK', p3, {MS, FA})
end role
```

Fig. 6 HLPSL code for implementing the FA process

will be analysed with a initial knowledge of the attacker. The demonstrated scenario is the composition of four sessions: players mu, ha and fa are in a first session, players intruder, ha and fa are in second session, the players mu, intruder and fa are in third session and, finally players mu, ha and intruder are in the last session. The goal role specifies the security requirements which the proposed authentication protocol requires to meet. The proposed mutual authentication protocol is simulated through AVISPA web tool under the ATSE (ATtack SEarcher) and OFMC backends. The AVISPA result comprises of the following segments:

1. **SUMMARY:** Which specifies that whether tested authentication protocols are safe or unsafe.
2. **DETAILS:** Describes under what criteria the tested protocols are concluded as safe or unsafe.
3. **PROTOCOL, GOAL, and BACKEND:** This section denotes a protocol name, goal of the protocol analysis and name of the backend used in AVISPA tool.

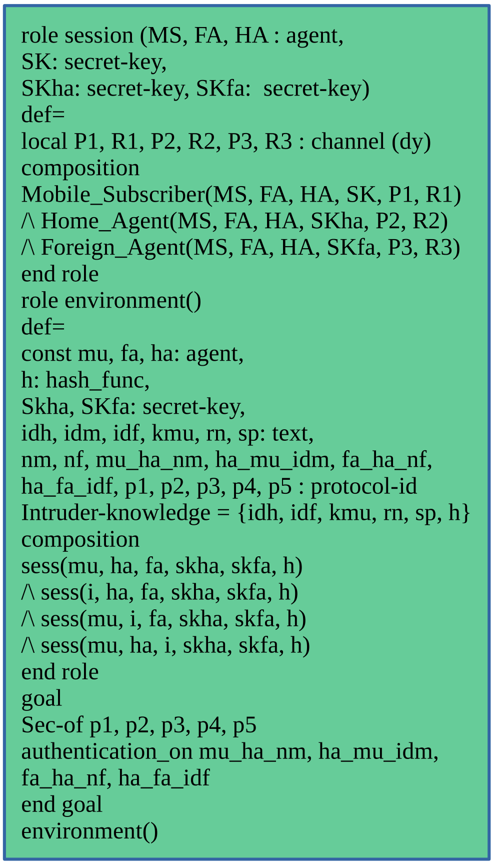
The AVISPA results of security analysis using OFMC as a back end is shown in Fig. 8. Similarly, the security analysis using ATSE backend is shown in Fig. 9. As summarized in simulation results, the number of visited nodes is 130 and the depth of search is 6, it requires 0.49 s. It is evident from the results that the proposed authentication protocol is safe and satisfies the design goals for roaming in global mobility networks. Further, the proposed protocol is verified using SPAN (Security Protocol Animator) tool to detect and build a message sequence chart (MSC) to represent the possible attacks and intruder activities.

8 Performance evaluation

This section evaluates the proposed mobile user authentication protocol and some other recent authentication protocols in terms of functionalities, communicational and computational complexities.

8.1 Functionality comparison

The proposed mutual authentication protocol is analysed and differentiated with recently introduced protocols for roaming services in GLOMONET such as Xu et al. [3],



```
role session (MS, FA, HA : agent,
SK: secret-key,
SKha: secret-key, SKfa:  secret-key)
def=
local P1, R1, P2, R2, P3, R3 : channel (dy)
composition
Mobile_Subscriber(MS, FA, HA, SK, P1, R1)
/\ Home_Agent(MS, FA, HA, SKha, P2, R2)
/\ Foreign_Agent(MS, FA, HA, SKfa, P3, R3)
end role
role environment()
def=
const mu, fa, ha: agent,
h: hash_func,
Skha, SKfa: secret-key,
idh, idm, idf, kmu, rn, sp: text,
nm, nf, mu_ha_nm, ha_mu_idm, fa_ha_nf,
ha_fa_idf, p1, p2, p3, p4, p5 : protocol-id
Intruder-knowledge = {idh, idf, kmu, rn, sp, h}
composition
sess(mu, ha, fa, skha, skfa, h)
/\ sess(i, ha, fa, skha, skfa, h)
/\ sess(mu, i, fa, skha, skfa, h)
/\ sess(mu, ha, i, skha, skfa, h)
end role
goal
Sec-of p1, p2, p3, p4, p5
authentication_on mu_ha_nm, ha_mu_idm,
fa_ha_nf, ha_fa_idf
end goal
environment()
```

Fig. 7 HLPSL code for role specification of session, goal and environment

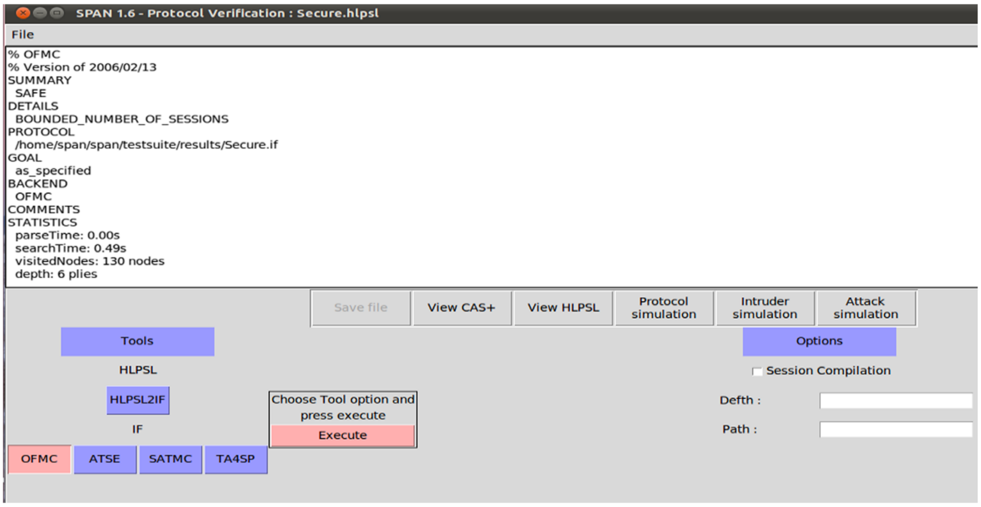
Karuppiah and Saravanan [1], and Reddy et al. [38]. Table 4 summarizes the functionality comparison of the proposed protocol and other authentication protocols [1, 3, 38]. It is clear from Table 4 that the proposed protocol ensures all functional requirements and withstands against security pitfalls in global mobility networks.

8.2 Performance comparison

Mobile terminals have limited resources in terms power, processor and memory. Thus, a major issue in wireless and mobile environments is consumption of the resources by computation and communication operations. Notably, the efficiency measures are accomplished in terms of computational and communicational complexities. The crypto symbols used to compute computational overhead are as follows:

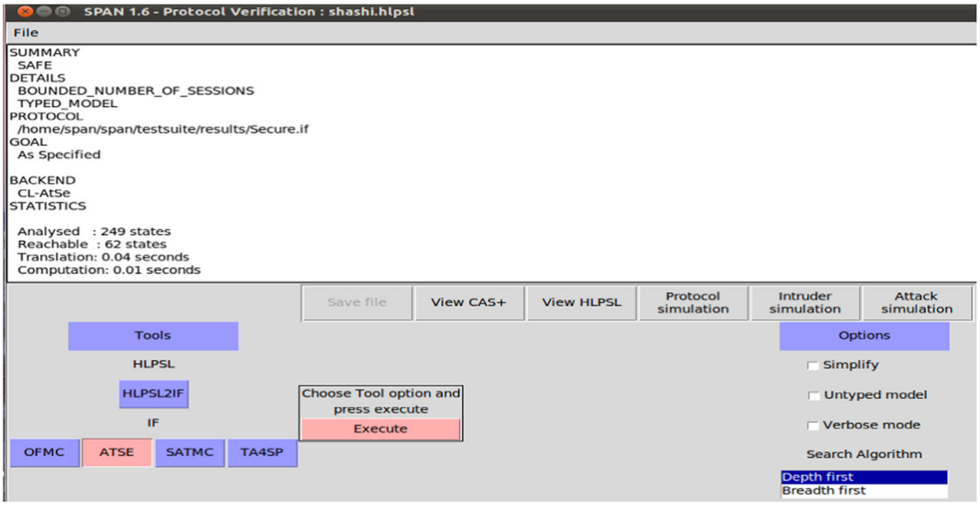
- T_h: Time complexity of a hash operation.
- T_m: Time complexity of a modular operation.
- T_{sym}: Time complexities of the symmetric encryption and decryption operations.
- P: Number of point operations on ECC.
- T_P: Time complexity of an elliptic-curve point multiplication.

In order to analyse the computational performance of the proposed authentication protocol in resource-limited devices, several cryptographic operations have been simulated through the Crypto library on a smartphone. The smartphone runs on the Android operating system of the Arm Cortex-A8 processor with frequency of 0.72 GHz. The crypto algorithms are executed in C++ language under Crypto++ library (MIRACL). Further, the hash operation, symmetric and asymmetric encryption/decryption operations are implemented by the Secure Hash Algorithm (SHA-256), AES-CBC (Advanced Encryption Standard with Cipher Block Chaining) and ECIES (Elliptic Curve Integrated Encryption Scheme). As per experimental results,



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/Secure.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.49s
  visitedNodes: 130 nodes
  depth: 6 plies
```

Fig. 8 Result analysis using OFMC backend



```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
  /home/span/span/testsuite/results/Secure.if
GOAL
  As Specified

BACKEND
  CL-AtSe
STATISTICS

  Analysed   : 249 states
  Reachable  : 62 states
  Translation: 0.04 seconds
  Computation: 0.01 seconds
```

□ Simplify
□ Untyped model
□ Verbose mode
Search Algorithm
Depth first
Breadth first

Fig. 9 Result analysis using ATSE backend

Table 4 Security properties comparison

Security Requirements	Proposed Scheme	Scheme [1]	Scheme [3]	Scheme [38]
User anonymity	✓	✓	×	✓
Mutual authentication	✓	✓	✓	✓
Withstand insider attack	✓	✓	✓	✓
Withstand impersonation attack	✓	×	✓	✓
Withstand smartcard loss attack	✓	✓	✓	✓
Withstand password guessing attack	✓	✓	✓	✓
Withstand replay attack	✓	✓	×	×
Perfect forward secrecy	✓	✓	✓	✓
Withstand stolen-verifier attack	✓	×	×	✓
Local password verification	✓	✓	×	✓
Fair key agreement	✓	✓	×	×
No time synchronization	✓	×	×	✓
Secure password change	✓	✓	✓	✓
Withstand denial-of-service attack	✓	✓	×	✓

Table 5 Execution time of various cryptographic operations

Notation	Description	Execution time (S)
T_h	Hash operation (SHA-256)	0.0005
T_{sym}	Symmetric encryption/decryption (AES-CBC)	0.0087
T_{asym}	Asymmetric encryption/decryption (ECIES)	0.01725
T_m	Modular operation (Diffie-Hellman)	0.522
T_P	Point multiplication on ECC	0.763

Table 6 Performance comparison

Computation	Proposed	Scheme [1]	Scheme [3]	Scheme [38]
MU	$6T_h$	$8T_h + 3T_m$	$7T_h$	$10T_h + 3T_P$
FA	$4T_h$	$3T_h$	$4T_h$	$5T_h + 2T_P$
HA	$10T_h + T_{sym}$	$8T_h + T_m + 3T_{sym}$	$9T_h + 2T_{sym}$	$7T_h + 2T_P$
Total	$\boldsymbol{20T_h + T_{sym}}$	$\boldsymbol{19T_h + 4T_m + 3T_{sym}}$	$\boldsymbol{20T_h + 2T_{sym}}$	$\boldsymbol{22T_h + 7T_P}$
Execution time (s)	**0.0187**	**2.523**	**0.0274**	**5.352**

Table 7 Comparison of computation cost

Scheme Phase	Proposed		Scheme [1]				Scheme [3]		Scheme [38]	
	HF	E/D	HF	E/D	ME	MM	HF	E/D	HF	PM
Registration	4	1	5	1	0	0	3	1	5	1
Authentication	20	1	24	3	3	1	20	2	22	7
Password Change	4	0	10	0	0	0	4	0	6	0
Total operations	**28**	**2**	**39**	**4**	**3**	**1**	**27**	**3**	**33**	**8**

ḢF: Hash Function; Ė/D: Encryption and Decryption; ṖM: Point Multiplication on ECC; ṀE:Modular Exponentiation; ṀM: Modular Multiplication

Table 8 Comparison of the communication overhead (bits)

Phase	Proposed Scheme	Scheme [1]	Scheme [3]	Scheme [38]
Registration	640	1120	640	960
Login	480	800	800	800
Mutual authentication and key agreement	1440	2400	2560	1760
Total cost	**2560**	**4320**	**4000**	**3520**

the execution time of various cryptographic operations are listed in Table 5. Table 6 summarizes the computational cost of MU, FA and HA in login and mutual authentication phase, since this phase is carried out regularly. In the proposed protocol, MU requires six hash operations to send a login message M_{MF} and mutual authentication process. FA needs four hash computations to transmitting messages $\{M_{FH}, M_{FM}\}$ between HA and MU, respectively. HA requires ten hash operations and a symmetric decryption operation to authenticate MU and FA. From Table 6, we can notice that the proposed protocol completes the authentication process in 0.0187 seconds. Therefore, the proposed protocol is computationally efficient than the protocols in [1, 3, 38]. Comparison of the cryptographic operations required in the registration phase, mutual authentication phase and the password change phase of the proposed protocol and some other relevant protocols [1, 3, 38] have been listed in Table 7. We can notice that the proposed protocol uses less hash computations and symmetric operations.

Table 8, presents the comparison of communication overhead of the protocols in [1, 3, 38] and the proposed protocol. In order to evaluate communication overhead, we assumed that the length of the secure hash function (SHA-256), random number, timestamp, and user's information are 160 bits, respectively. The length of the elliptic curve point (ECC) is 320 bits.

In the proposed protocol, the registration messages $R_1 = \{RID\}, R_2 = \{HID, K_{MU}, h(.)\}$ requires (160+160+160+160)=640 bits and the login request $M_{MF} = \{A_M, V_1, ID_H\}$ needs (160+160+160)=480 bits. In order to perform mutual authentication and establishing the session key, the proposed protocol transmits the messages $M_{FH} = \{ID_F, B_M, V_1, V_2\}$, $M_{HF} = \{N'_M, V_3, V_4\}$, $M_{FM} = \{N'_M, V_4\}$, which requires (640+480+320)=1440 bits. Hence, the proposed protocol needs total $(640 + 480 + 1440) = 2560$ bits. From Fig. 10, we can conclude that the proposed protocol has low communication overhead as compared to the protocols [1, 3, 38]. Hence, the proposed protocol is light-weight, computationally efficient and practically implementable in resource-constrained mobile devices.

8.3 GLOMONET simulation Using NS2 simulator

The proposed mutual authentication protocol for roaming in mobility environments is simulated using NS2 simulator. It is a discrete event network simulator, primarily used in research community and teaching to simulate various protocols including routing protocols, TCP/UDP (Transmission Control Protocol/User Datagram Prtocol) over wired, Ad hoc, mobile, and wireless sensor networks [39]. Further, we estimate the network performance of the proposed protocol using NS-2.35 simulator.

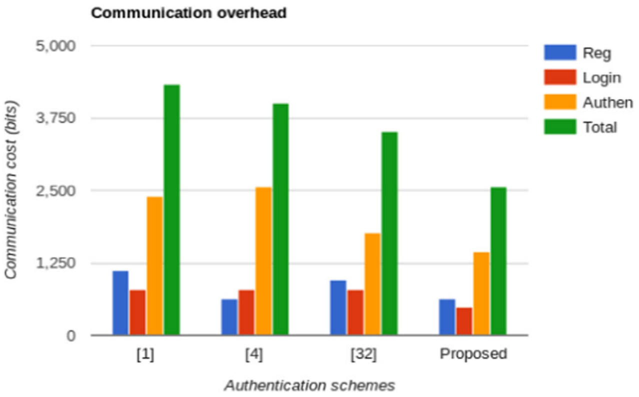


Fig. 10 Comparison of communication cost

8.3.1 Simulation environment

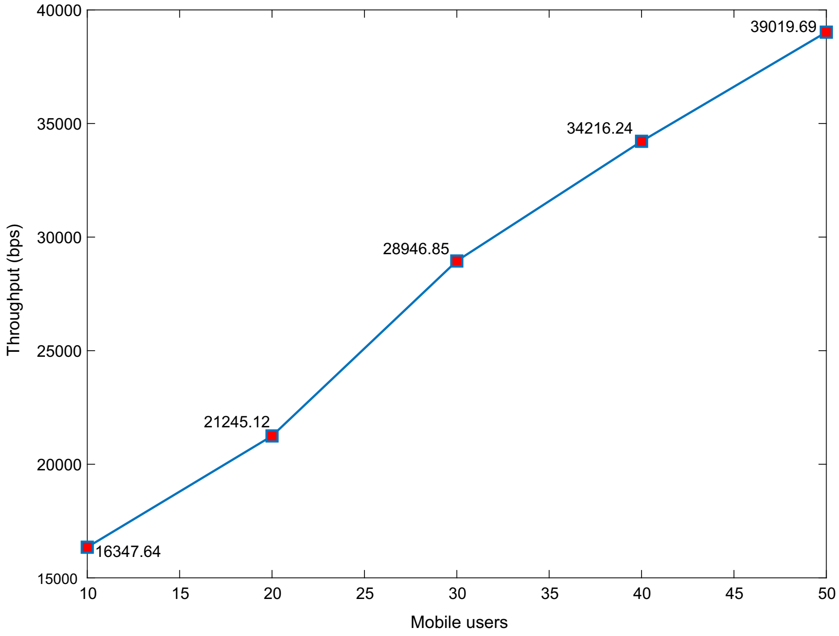
We have used Ubuntu 14.04 LTS platform to run NS-2.35 simulator. The network simulation parameters are:

1. There are 5 Foreign Agents (FAs) communicating with the Home Agent (HA) are placed in a line, each neighbour pair is separated with the distance of 50 m.
2. The mobile users are restricted in the 750 x 750 m^2 area with a speed of 3 m/s. The number of mobile subscribers are multiples of ten, that is 10 users are added first, and next 10 subscribers are added, until the number of mobile users reaches 50.
3. The proposed GLOMONET model is simulated for 10 sec and each mobile user transmits the packets with an interval of 1 sec is taken into account.
4. Wireless environment is used in between MU and FA. In addition, P2P communication medium is established between HA and FA.
5. The proposed protocol consists of four messages between MU, FA and HA in mutual authentication phase. The service request message M_{MF} is size of 480 bits, and authentication request and response messages M_{FH}, M_{HF}, M_{FM} are of sizes 640, 480 and 320 bits, respectively.

8.3.2 Simulation results

During the simulation, the most significant network performance metrics like throughput, packet delivery ratio, load, end to end delay is analysed and computed.

Fig. 11 Impact on throughput



8.3.3 Impact on throughput

Network throughput (in bps) is the amount of data transferred successfully in a given time period. Throughput is calculated as:

$$Throughput = \frac{Recieved\ packets \times Bit\ size\ of\ a\ packet}{Total\ simulation\ time}.$$

From Fig. 11, we can summarize that the throughput will be maximum when the number of mobile subscribers increases in the network. The reason is that the transmitted information will be more in case of huge number of mobile users are interacting to the service provider network.

8.3.4 Impact on packet delivery ratio

It is the ratio of total number of data packets arrived at the destination and the total number of sent packets. Packet delivery ratio (PDR) is calculated as:

$$PDR = \frac{No.\ of\ recieved\ packets}{No.\ of\ sent\ packets}$$

Fig. 12 Impact on packet delivery ratio

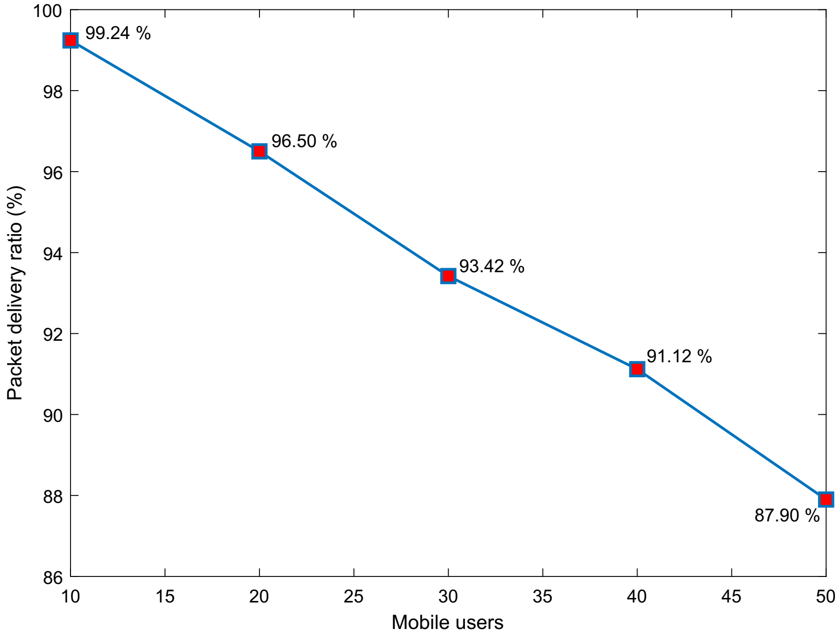
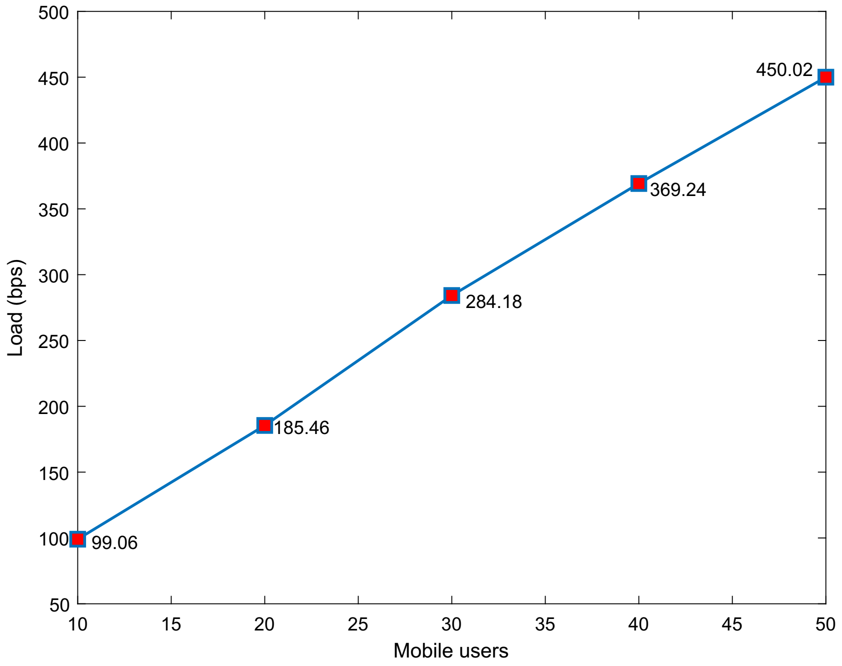


Fig. 13 Impact on load



From Fig. 12, we can see that the packet delivery ratio falls down with increase of mobile users. The reason is that more congestion happening in the network when the mobile subscribers will be higher. Further, if the mobile subscriber is far from HA, then the energy in a sent packet will be dry and dropped at the FA. As the proposed protocol is lightweight and makes use of the smaller packet size. As a result, PDR decrement is small.

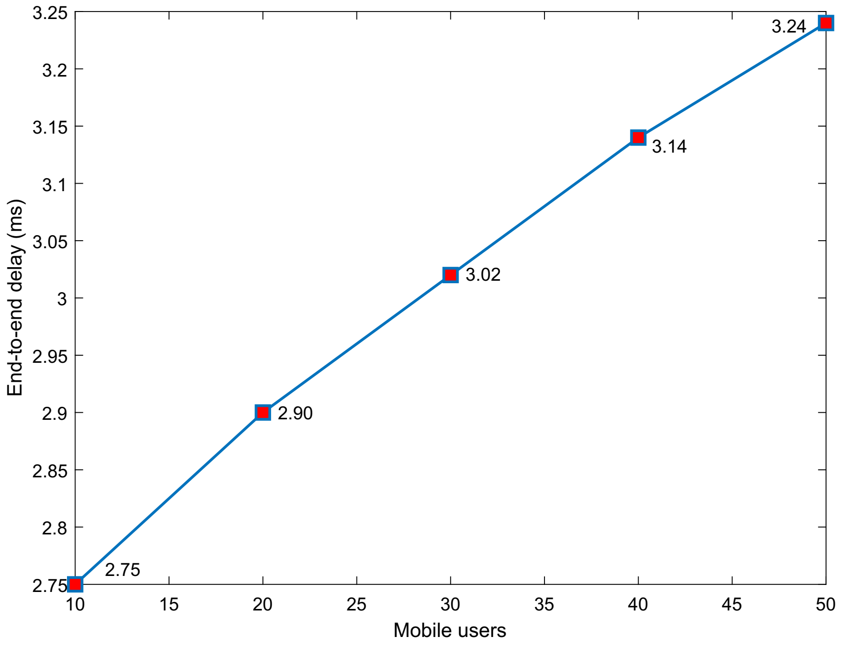
8.3.5 Impact on load

The network load can be computed as:

$$Load = \frac{(Sent + Recieved\ packets) \times Bit\ size\ of\ packet}{Total\ simulation\ time}.$$

In this simulation model, we have computed load of the home agent HA. From Fig. 13, we can note that the network

Fig. 14 Impact of end-to-end delay



load is increasing with more number of the mobile users interacting to a service provider network.

8.3.6 Impact on end to end delay

End to end delay refers to the time taken for a data packet to be sent across a network from source to destination. It can be computed as:

$$EED = \frac{T_{Rec} - T_{Snd}}{T_P}.$$

where T_{Rec} is the time of receiving, T_{Snd} is the time of sending a packet and T_P is the total number of sent packets. In Fig. 14, we can observe that the end-to-end delay keeps increasing when the mobile users increases in the network. It is obvious that maximum flow of messages bring in higher distances and more congestions in the mobile network.

9 Conclusion

In this article, we found some security flaws in the Xu et al.'s authentication and key agreement protocol. We pointed out that their protocol is vulnerable to stolen verifier attack, impersonation attack, privileged insider attack, denial of service attack, clock-synchronization problem and unable to provide local password verification to detect wrong passwords quickly. As a remedy, we proposed a secure and robust mutual authentication protocol for mobility networks. In order to prove the correctness, the proposed protocol is implemented in HLPSL language using AVISPA as the formal verification tool. Besides, the performance evaluation shows that the proposed security protocol is light-weight, secure and computationally efficient. Hence, this authentication system is robust and practically implementable in resource limited mobility environments.

References

1. Karuppiah M, Saravanan R (2015) A secure authentication scheme with user anonymity for roaming service in global mobility networks. Wirel Pers Commun 84(3):2055–2078
2. Gope P, Hwang T (2016) Lightweight and energy-efficient mutual authentication and key agreement scheme with user anonymity for secure communication in global mobility networks. IEEE Syst J 10(4):1370–1379
3. Xu G, Liu J, Lu Y, Zeng X, Zhang Y, Li X (2018) A novel efficient maka protocol with desynchronization for anonymous roaming service in global mobility networks. J Netw Comput Appl 107:83–92
4. Madhusudhan R, Shashidhara R (2019) Mobile user authentication protocol with privacy preserving for roaming service in glomonet. Peer Peer Netw Appl 1–22
5. Zhu J, Ma J (2004) A new authentication scheme with anonymity for wireless environments. IEEE Trans Consum Electron 50(1):231–235
6. Lee C, Hwang M, Liao E (2006) Security enhancement on a new authentication scheme with anonymity for wireless environments. IEEE Trans Ind Electron 53(5):1683–1687
7. Wu C, Lee W, Tsaur W et al (2008) A secure authentication scheme with anonymity for wireless communications. IEEE Commun Lett 12(10):722–723
8. Yoon E, Yoo K (2011) Young, Ha, K.: A user friendly authentication scheme with anonymity for wireless communications. Comput Electr Eng 37(3):356–364
9. Li C, Lee C (2012) A novel user authentication and privacy preserving scheme with smart cards for wireless communications. Math Comput Model 55(1):35–44
10. He D, Ma M, Zhang Y, Chen C, Bu J (2011) A strong user authentication scheme with smart cards for wireless communications. Comput Commun 34(3):367–374
11. Li X, Niu J, Khan MK, Liao J (2013) An enhanced smart card based remote user password authentication scheme. J Netw Comput Appl 36(5):1365–1371
12. Jiang Q, Ma J, Li G, Yang L (2013) An enhanced authentication scheme with privacy preservation for roaming service in global mobility networks. Wirel Pres Commun 68(4):1477–1491
13. Wen F, Susilo W, Yang G (2013) A secure and effective anonymous user authentication scheme for roaming service in global mobility networks. Wirel Pers Commun 73(3):993–1004
14. Zhao D, Peng H, Li L, Yang Y (2014) A secure and effective anonymous authentication scheme for roaming service in global mobility networks. Wirel Pres Commun 78(1):247–269
15. Mun H, Han K, Lee YS, Yeun CY, Choi HH (2012) Enhanced secure anonymous authentication scheme for roaming service in global mobility networks. Math Comput Model 55(1):214–222
16. Wu F, Xu L, Kumari S, Li X, Das AK, Khan MK, Karuppiah M, Baliyan R (2016) A novel and provably secure authentication and key agreement scheme with user anonymity for global mobility networks. Secur Commun Netw 9(16):3527–3542
17. Al Amiri W, Baza M, Mahmoud M, Alasmary W, Akkaya K Towards secure smart parking system using blockchain technology
18. Amiri WA, Baza M, Banawan K, Mahmoud M, Alasmary W, Akkaya K (2019). Privacy-preserving smart parking system using blockchain and private information retrieval. arXiv:1904.09703
19. Baza M, Nabil M, Ismail M, Mahmoud M, Serpedin E, Rahman M (2018). Blockchain-based charging coordination mechanism for smart grid energy storage units. arXiv:1811.02001
20. Baza M, Nabil M, Lasla N, Fidan K, Mahmoud M, Abdallah M (2019) Blockchain-based firmware update scheme tailored for autonomous vehicles. In: 2019 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, pp 1–7
21. Zhang R, Xue R, Liu L (2019). Security and privacy on blockchain. arXiv:1903.07602
22. Baza M, Nabil M, Bewermeier N, Fidan K, Mahmoud M, Abdallah M (2019). Detecting sybil attacks using proofs of work and location in vanets. arXiv:1904.05845
23. Gope P, Hwang T (2016) An efficient mutual authentication and key agreement scheme preserving strong anonymity of the mobile user in global mobility networks. J Netw Comput Appl 62:1–8
24. Guo D, Wen F (2016) A more robust authentication scheme for roaming service in global mobility networks using ecc. IJ Netw Secur 18(2):217–223
25. Madhusudhan R, et al. (2016) An effcient and secure authentication scheme with user anonymity for roaming service in global mobile networks. In: Proceedings of the 6th international conference on communication and network security. ACM, pp 119–126
26. Wu F, Xu L, Kumari S, Li X, Khan MK, Das AK (2017) An enhanced mutual authentication and key agreement scheme for mobile user roaming service in global mobility networks. Ann Telecommun 72(3–4):131-144

27. Bojjagani S, Sastry V (2017) A secure end-to-end sms-based mobile banking protocol. Int J Commun Syst 30(15):3302
28. Bojjagani S, Sastry V A secure end-to-end proximity nfc-based mobile payment protocol. Comput Stand Inter 66
29. Karuppiah M, Kumari S, Li X, Wu F, Das AK, Khan MK, Saravanan R, Basu S (2017) A dynamic idbased generic framework for anonymous authentication scheme for roaming service in global mobility networks. Wirel Pres Commun 93(2):383–407
30. Arshad H, Rasoolzadegan A (2017) A secure authentication and key agreement scheme for roaming service with user anonymity. Int J Commun Syst
31. Madhusudhan R, Shashidhara R (2019) A novel dna based password authentication system for global roaming in resource-limited mobile environments. Multimed Tool Appl 1–28
32. Madhusudhan R, Shashidhara R (2019) A secure anonymous authentication protocol for roaming service in resource-constrained mobility environments. Arab J Sci Eng 1–22
33. Ha J (2015) An efficient and robust anonymous authentication scheme in global mobility networks. Int J Secur Appl 9(10):297–312
34. Kocher P, Jaffe J, Jun B (1999) Differential power analysis. In: Advances in Cryptology—CRYPTO'99. Springer, pp 388–397
35. Wang D, He D, Wang P, Chu C-H (2015) Anonymous two-factor authentication in distributed systems: certain goals are beyond attainment. IEEE Trans Depend Secur Comput 12(4):428–442
36. Madhusudhan R, et al. (2018) A secure and lightweight authentication scheme for roaming service in global mobile networks. J Inform Secur Appl 38:96–110
37. Armando A, Basin D, Cuellar J, Rusinowitch M, Viganó L (2006) Avispa: automated validation of internet security protocols and applications. ERCIM News 64
38. Reddy AG, Das AK, Yoon E-J, Yoo K-Y (2016) A secure anonymous authentication protocol for mobile ser vices on elliptic curve cryptography. IEEE Access 4:4394–4407
39. Odelu V, Das AK, Kumari S, Huang X, Wazid M (2017) Provably secure authenticated key agreement scheme for distributed mobile cloud computing services. Future Gener Comput Syst 68:74–88

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

R. Shashidhara is currently working as an Assistant Professor in the School of Engineering & Applied Sciences, Bennett University, Greater Noida Uttar Pradesh, India. He received his PhD from National Institute of Technology Karnataka Surathkal, India. He is a reviewer for the many reputed journals. His research interests include design of robust authentication protocols for wireless and mobile networks, cross-site scripting attacks, blockchain technology and IoT security.

Sriramulu Bojjagani Dr. Sriramulu Bojjagani is working as an Assistant Professor in the Department of Computer Science and Engineering, SRM University, Amaravathi, AP, India. He received his Ph.D. Degree for his work on "Design, Testing and Formal Verification of Secure Mobile Payment Protocols" from the School of Computer and Information Sciences, University of Hyderabad and Institute for Development and Research in Banking Technology (IDRBT), India in 2019. He obtained his M.Tech in Information Technology from the Andhra University, Visakhapatnam, India. He received his B.Tech from JNTU Hyderabad, India. He is a reviewer for some reputed journals like IJCS:Wiley, Peer-to-Peer Networking and applications, Telecommunication systems: Springer, IEEE Security & Privacy and IEEE Systems. His research interests include design, testing of mobile payment applications, formal methods for verification of secure mobile payment protocols, Information security and privacy in IoV (Internet of Vehicles), security issues in cloud computing and SDN (Software Defined Networks).

Anup Kumar Maurya is presently working as a Senior Lecturer in Big Data Analytics Department of Goa Institute of Management, Goa, India. He has completed his Ph.D. in the area of "Secure Wireless Sensor Networks & Internet of Things" from the University of Hyderabad, India. He obtained his M.Tech. Degree in Artificial Intelligence from the University of Hyderabad, India. He received his Master of Computer Application Degree from Uttar Pradesh Technical University, India, and a Bachelor of Science Degree in Mathematics from D.D.U. Gorakhpur University, India. He has various research publications, including one of special mention where he got the best research paper award at the International Symposium on Security in Computing and Communication (SSCC-2017, Manipal, India). He is a reviewer for the various reputed international journals. He was selected as a visiting researcher at the State University of New York at Buffalo (Centre for Unified Biometrics and Sensors), USA and has worked there on biometric-based efficient user authentication protocols.

Saru Kumari received her Ph.D. degree in Mathematics in 2012 from CCS University, Meerut, UP, India. She has published more than 141 research papers in reputed International journals and conferences, including 122 publications in SCI-Indexed Journals. She is on the Editorial board of AEÜ - International Journal of Electronics and Communications, Elsevier (SCI); International Journal of Communication Systems, Wiley (SCI-E); Telecommunication Systems, Springer (SCI); Human Centric Computing and Information sciences, Springer (SCI-E); Transactions on Emerging Telecommunications Technologies; Wiley (SCI-E), Information Technology and Control, Kaunas University of Technology, Lithuania (SCI-E); KSII Transactions on Internet and Information Systems (SCI-E), published from Taiwan; Information Security: A Global Perspective, Taylor & Francis (ESCI, Scopus); International Journal of Wireless Information Networks (ESCI, Scopus), Springer; Journal of Reliable Intelligent Environments, Springer. (ESCI, Scopus); Security and Privacy, Wiley; Iran Journal of Computer Science, Springer; Azerbaijan Journal of High Performance Computing, published by Azerbaijan State Oil and Industry University, Azerbaijan. She served as Guest Editor of the Special Issue Big-data and IoT in e-Healthcare for Computers and Electrical Engineering, Elsevier (SCI-E), Elsevier. She is Technical Program Committee Member for more than a dozen of International conferences. She is a reviewer of more than 50 reputed Journals including SCI-Indexed Journals of IEEE, Elsevier, Springer, Wiley etc. Her current research interests include information security and applied cryptography.

Hu Xiong received the Ph.D. degree from the School of Computer Science and Engineering, University of Electronic Science and Technology of China (UESTC) in 2009. He is currently a Full Professor with the School of Information and Software Engineering, UESTC. His research interests include cryptographic protocols and network security. Since 2010, Dr. Xiong has participated in 5 R+D projects from both Chinese government and industries. His research interests include public key cryptography and cyber space security. Dr. Xiong has published over 9 IEEE journals and published a book in the CRC press as the 1st author in the field of his research area. Also, he has organized special issue related to security and privacy in Transactions on Emerging Telecommunications Technologies and Mathematical Biosciences and Engineering.

Affiliations

R. Shashidhara[1] · Sriramulu Bojjagani[2] · Anup Kumar Maurya[3] · Saru Kumari[4] · Hu Xiong[5]

R. Shashidhara
eemailshashi@gmail.com

Sriramulu Bojjagani
sriramulu.b@srmap.edu.in

Anup Kumar Maurya
anupmaurya88@gmail.com

Hu Xiong
xionghu.uestc@gmail.com

[1] School of Engineering and Applied Sciences, Bennett University, Greater Noida, 201310, Uttar Pradesh, India

[2] Department of Computer Science and Engineering, SRM University, Amaravati, Neerukonda, Guntur District, Mangalagiri, 522502, AP, India

[3] Goa Institute of Management, Goa, 403505, India

[4] Department of Mathematics, Chaudhary Charan Singh University, Meerut 250004, Uttar Pradesh, India

[5] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China