Multimedia Tools and Applications (2019) 78:11129–11153
https://doi.org/10.1007/s11042-018-6620-2

Perfect forward secrecy in VoIP networks through design a lightweight and secure authenticated communication scheme

Niloofar Ravanbakhsh [1] · Mohadeseh Mohammadi [1] · Morteza Nikooghadam [1]

Received: 26 November 2017 / Revised: 17 June 2018 / Accepted: 27 August 2018 /
Published online: 19 September 2018

Abstract
With the growth of the internet, development of IP based services has increased. Voice over IP (VoIP) technology is one of the services which works based on the internet and packet switching networks and uses this structure to transfer the multimedia data e.g. voices and images. Recently, Chaudhry et al., Zhang et al. and Nikooghadam et al. have presented three authentication and key agreement protocols, separately. However, in this paper, it is proved that the presented protocols by Chaudhry et al. and also Nikooghadam et al. do not provide the perfect forward secrecy, and the presented protocol by Zhang et al. not only is vulnerable to replay attack, and known session-specific temporary information attack, but also does not provide user anonymity, re-registration and revocation, and violation of fast error detection. Therefore, a secure and efficient two-factor authentication and key agreement protocol is presented. The security analysis proves that our proposed protocol is secure against various attacks. Furthermore, security of proposed scheme is formally analyzed using BAN logic and simulated by means of the AVISPA tool. The simulation results demonstrate security of presented protocol against active and passive attacks. The communication and computation cost of the proposed scheme is compared with previously proposed authentication schemes and results confirm superiority of the proposed scheme.

Keywords Authentication · Cryptanalysis · Key agreement · Lightweight design · Session initiation protocol (SIP)

✉ Morteza Nikooghadam
morteza.nikooghadam@gmail.com; m.nikooghadam@Imamreza.ac.ir

Niloofar Ravanbakhsh
niloofar_ravanbakhsh@yahoo.com

Mohadeseh Mohammadi
mohadesehmohammadi@ymail.com

[1] Department of Computer Engineering and Information Technology, Imam Reza International University, Mashhad, Iran

1 Introduction

VoIP technology has provided the possibility to use the internet to make phone calls in comparison with traditional phones. In VoIP technology, voice is sent by IP information packets through the internet. VoIP technology in comparison to traditional phone networks has many advantages such as easy expansibility, implementation flexibility, concentration management and lower cost [48]. VoIP is composed of session initiation protocol (SIP) and real-time transportprotocol (RTP). SIP was first developed by Internet Engineering Task Force on 1999 [13, 31]. It is an application layer signaling protocol whose function is to initiate, manage and terminate a session between two or more systems based on IP. H323 and MGCP are some of rival protocols of SIP [48, 34]. Authentication and key agreement schemes in SIP are either based on password, which are called one-factor schemes [6, 23, 32, 35], or based on password and smart card which are called two-factor schemes [1, 5, 12, 22, 26–29, 48].

In order to provide a secure key agreement on SIP, many protocols have been proposed in recent years [4, 10, 11, 15–18, 23, 27, 30, 36–39, 42–48]. One of the oldest technologies used for authentication and key agreement in SIP is HTTP digest single factor scheme which was dismissed due to inefficiency in providing security requirements such as mutual authentication, off-line password guessing and stolen verifier attacks [14, 33, 42]. Furthermore, some schemes based on Diffie-Hellman key agreement were presented which suffer from considerable computational costs. Nowadays, considering the importance of decreasing computation and communication costs, protocols are up to providing this matter.

Because of the efficiency of elliptic curve, the difficulty of Discrete Logarithm Problem, and shorter key length, nowadays most one-factor and two-factor authentication schemes use Elliptic Curve Cryptography to provide full security and reduce the computational cost [16, 22, 27, 29, 36, 39, 48].

In 2005, Yang et al. [42] showed that previous mechanism used for authentication and key agreement in SIP (HTTP digest) is vulnerable against offline password guessing and server spoofing attacks. Therefore, a new scheme based on Diffie-Hellman key exchange was presented that the security of this scheme was based on the difficulty of solving the Discrete Logarithm Problem. Durlanik and Sogukpinar [10] showed that Yang et al.'s protocol [42] was not suitable for resource-constrained equipment due to its high computational costs. In order to decrease computational costs, a new scheme based on Elliptic Curve Diffie–Hellman was presented [44]. Yoon and Yoo in 2009 [44] proved that the mentioned scheme [10] is vulnerable against some attacks including stolen verifier and Denning Sacco attacks. Wu et al. in 2009 [39] presented a key agreement and secure authentication scheme for SIP based on the difficulty of Elliptic Curve Cryptography problem. However, Yoon et al. [45] showed that their scheme [39] is vulnerable against offline password guessing attack. Later, it is shown that the presented scheme by Yoon et al. [45] is vulnerable against password guessing attack [21]. Tsai [37] presented a lightweight scheme for SIP. However, Arshad and Ikram [4] proved that the presented scheme by Tsai is vulnerable against offline password guessing and stolen verifier attacks, and does not provide perfect forward secrecy (PFS) and known key secrecy. They also presented a new mutual authentication scheme based on Elliptic Curve Cryptography. Some other researcher, [15, 24, 30, 36], illustrate vulnerability of Arshad and Ikram's scheme [4] against offline password guessing, internal and masquerade attacks. Irshad

et al. [16] demonstrated that the proposed scheme in [36] was insecure against server impersonation attack and they proposed a new scheme, however, their scheme was vulnerable to privileged insider attack [6]. Zhang et al. [46] presented an authentication and key agreement scheme for SIP based on password and smart card which unlike the mentioned schemes doesn't need to store passwords in server or hold a verification table. Zhang et al. [46] claimed that their scheme is resistant to well-known attacks. Nevertheless, some other researchers affirmed that Zhang et al.' scheme [46] is vulnerable against user impersonation and malicious insider attacks [18, 38, 47]. Irshad et al. [17] proved that Zhang et al. [46] scheme is vulnerable against Denial of Service attack. Arshad and Nikooghadam [5] proved that the presented protocol by Jiang et al. [18] is vulnerable against user impersonation attack, and Yeh et al.'s scheme [43] is not robust against offline password guessing attack, and can't provide perfect forward secrecy. Recently, Chaudhry et al. [8] presented an authentication and key agreement scheme preserving privacy. In this paper, we proved their scheme doesn't provide perfect forward secrecy.

Lu et al. [23] introduced a secure and efficient scheme for SIP in 2016. Nevertheless, Chaudhry et al. [9], Xu et al. [41] and Kumari et al. [20] pointed Lu et al.'s scheme [23] is still vulnerable against user and server impersonation attack, stolen verifier attack and identity guessing attack. Therefore, Xu et al. [41] in 2017 designed a provably secure anonymous mutual authentication scheme.

In 2016, Zhang et al. [48] also presented an efficient authentication and key agreement scheme for VoIP networks. Besides, Nikooghadam et al. [28] presented a scheme for key agreement to preserve privacy through anonymity. In this paper, security flaws of Zhang et al. [48] and Nikooghadam et al. [28] schemes are scrutinized and is demonstrated that Zhang et al. scheme [48] is vulnerable against replay attacks, known session specific temporary information attacks, ignoring re-registration and revocation, violation of fast error detection and does not provide user anonymity, Furthermore, Nikooghadam et al.'s scheme [28] does not provide perfect forward secrecy.

Our contribution: The contribution of this paper is as follows.

- Cryptanalysis of Chaudhry et al.'s scheme [8], Zhang et al. 's scheme [48] and Nikooghadam et al. 's scheme [28], and demonstrating their security challenges and flaws.
- Presenting a two-factor authentication and key agreement protocol that (a) solves the security challenges of schemes [8, 28, 48]; (b) provides perfect forward secrecy and user anonymity; (c) is more efficient than most recent schemes in terms of both communication and computation costs; (d) uses no expensive operations over elliptic curves.
- Security analysis of the proposed scheme in the BAN logic and the AVISPA tool.

In the first section of this paper, VoIP, SIP and some related works are reviewed. Nikooghadam et al.'s scheme [28] and its security weakness is described in the second section. Chaudhry et al.'s scheme [8] is reviewed in Section 3. security weakness of Chaudhry et al.'s scheme [8] is also explained in this section. Zhang et al.'s scheme [48] is reviewed and analyzed in Section 4. In Section 5, we propose our suggested protocol. In Section 6, security of our suggested protocol is verified through informal and formal analyses. In Section 7, we analyze and compare performance of the suggested protocol with other protocols and finally, the conclusion part is presented.

2 Review of Nikooghadam et al.'s scheme

Nikooghadam et al.'s scheme [28] includes three phases: registration, login and authentication, and password changing phases. In this section, we briefly review the first two phases to analyze their proposed scheme, then its security flaw is demonstrated. The definition of notations used in Nikooghadam et al.'s scheme [28], Chaudhry et al.'s scheme [8] and Zhang et al.'s scheme [48] are shown in Table 1.

2.1 Registration phase

In registration phase, user performs the following steps over a secure channel. At the end of this phase, a smart card is issued by server to a user.

Step 1. First, an identity ID_i, a password PW_i, and a random number, r, are chosen by the user. Then the masked password $MPW_i = h(ID_i \| r \| PW_i)$ is computed by the user. The values of ID_i and MPW_i are transferred to the server via a secure channel.

Step 2. After receiving $\{ID_i, MPW_i\}$, the server searches registered user's table whether received ID_i exists or not. If received ID_i was repetitive, the server asks for a new ID_i. Otherwise, the server computes the values of $A_i = h(ID_i \| x)$, $B_i = A_i \oplus MPW_i$ and selects a random number, N, and calculates masked identity of user $MID_i = E_x(ID_i \| N)$. At last, the server stores ID_i to the table of registered user, and issues a smart card that, includes $\{B_i, MID_i, E_{key}(.)/D_{key}(.), h(.)\}$. Then the server sends the smart card to user through a secure channel.

Table 1 The notations used in Nikooghadam et al., Chaudhry et al., and Zhang et al.'s schemes

Symbol	Definition
U_i	A user
ID_i	The identity for U_i
PW_i	The password for U_i
x	The server private key
MID_i	The masked identity for U_i
MPW_i	The masked password for U_i
SK	The session key between the user and the server
$\oplus$	The exclusive-OR operation (XOR)
$\|$	The concatenation operation
$D_k(.)/E_k(.)$	The symmetric decryption/encryption with the key k
$h(.)$	A secure one-way hash function
PID_i	Pseudo identity of $\boldsymbol{U_i}$
k_{s1}, k_{s2}	The secret keys (number) maintained by server
T_{Ui}	Timestamp of U_i
T_{Si}	i^{th} time stamps of S
k_i	Unique random number of U_i
S	SIP server
s	A high-entropy secret key of S
p	A prime power
P	A generator point with the order n over $E_p(a, b)$
F_p	A prime finite field
$E_p(a, b)$	An elliptic curve equation
r, r_1, r_2, r_3, r_4	High-entropy random numbers
$(Q)_x/(Q)_y$	x-coordinate value or y-coordinate value of elliptic curve point Q

Step 3. As soon as the user receives smart card, stores the value of random number r into the smart card.

2.2 Login and authentication phase

Step 1. The user inserts his/her smart card into the card reader and inputs his/her ID_i and PW_i. Then the smart card selects a random number RN_i, captures the current time stamp T_i, and calculates $A_i = B_i \oplus h(ID_i \| r \| PW_i) = h(ID_i \| x)$, $M_1 = E_{A_i}(ID_i \| RN_i \| T_i \| MID_i)$. Then login request message REQUEST$\{MID_i, M_1, T_i\}$ is transmitted to the server via an insecure channel.

Step 2. when, the server receives login request message at T_s, checks condition $|T_s - T_i| \leq \Delta T$ holds or not. If $|T_s - T_i| \leq \Delta T$ holds, then the server calculates $D_x(MID_i) = (ID_i \| N)$ and $A_i^* = h(ID_i \| x)$. After that, the server decrypts received M_1 by A_i^* as $D_{A_i^*}(M_1) = (ID_i \| RN_i \| T_i \| MID_i)$, acquires $(ID_i \| RN_i \| T_i \| MID_i)$ and compares them with received MID_i and T_i. If they are equal, then the server selects two random numbers N^{new} and RN_s and computes $MID_i^{new} = E_x(ID_i \| N^{new})$, $M_2 = E_{A_i^*}(MID_i^{new} \| RN_s \| ID_i \| RN_i)$ and transmits a challenge message CHALLENGE $\{M_2\}$ to the user.

Step 3. Upon receiving $\{M_2\}$, user decrypts M_2 as $D_{A_i}(M_2) = (MID_i^{new} \| RN_s \| ID_i \| RN_i)$ and verifies ID_i and RN_i. After calculating $M_3 = h(RN_s \| MID_i^{new} \| RN_i)$ and $SK = h(RN_i \| A_i \| RN_s)$ by the user, he/she replaces MID_i with MID_i^{new} and sends a response message RESPONSE $\{M_3\}$ to the server.

Step 4. Upon the server receives $\{M_3\}$, calculates $M_3^* = h(RN_s \| MID_i^{new} \| RN_i)$ and verifies condition $M_3^* = ?M_3$. If this condition holds, the user is authenticated by the server. Then server and user agree on same session key $SK = h(RN_i \| A_i^* \| RN_s)$.

2.3 Cryptanalysis of Nikooghadam et al.'s scheme

If the adversary obtains the private key of the server, x, he/she can acquire session key $SK = h(RN_i \| A_i \| RN_s)$ by performing the following steps:

Step 1. In first step of login and authentication phase of Nikooghadam et al.'s scheme [28], M_1 and MID_i are sent through a public channel. So, the adversary can decrypt MID_i with private key of server (x) as $D_x(MID_i) = (ID_i \| N)$, and obtains ID_i and N. Then the adversary calculates A_i as $A_i = h(ID_i \| x)$ and decrypts M_1 using A_i as $D_{A_i}(M_1) = (ID_i \| RN_i \| T_i \| MID_i)$, and derives RN_i.

Step 2. In second step of login and authentication phase, M_2 is sent to user via an insecure channel, therefore the adversary reaches RN_s with decrypting M_2 with A_i as $D_{A_i}(M_2) = (MID_i^{new} \| RN_s \| ID_i \| RN_i)$. As a result, by exposing the private key of server, the adversary is able to derive RN_i, RN_s,and A_i, then calculates session key $SK = h(RN_i \| A_i \| RN_s)$. Therefore, Nikooghadam et al.'s scheme does not provide the perfect forward secrecy.

3 Review of Chaudhry et al.'s scheme

In this section, first two phases of Chaudhry et al.'s scheme [8] is briefly discussed and then we prove that it does not provide perfect forward secrecy. Chaudhry et al.'s scheme [8] includes three phases: registration phase, login and authentication phase, and password change phase. The used notations in their scheme are listed in the Table 1.

3.1 Registration phase

In this phase, user and server performs the following steps:

Step 1. The user chooses an identity ID_i, a password PW_i, a random number c, and calculates $RP_i = h(c \| ID_i)$. Then he/she transmits $\{ID_i, RP_i\}$ to the server through a secure channel.

Step 2. The server computes pseudo-identity $PID_i = E_{ks2}(ID_i \| T_{s0})$ for the user. After that, the values of $G_i = h(ID_i \| k_{s1}) \oplus RP_i$, $K_i = k_i \oplus RP_i$, $H_i = h(ID_i \| k_i \| RP_i)$ and $J_i = k_i \oplus h(k_{s2} \| ID_i)$ are calculated by the server, the server saves $\{K_i, H_i, J_i, PID_i, h(.)\}$ into the smart card and forwards the smart card SC_{ui} and G_i to the user over a secure channel.

Step 3. When the user receives $\{SC_{ui}, G_i\}$, he/she computes $R_i = (ID_i \| PW_i) \oplus c$ and $L_i = G_i \oplus c$, and stores both R_i and L_i in SC_{ui}. At last, the smart card SC_{ui} subtends $\{K_i, H_i, J_i, PID_i, h(.), R_i, L_i\}$.

3.2 Login and authentication phase

Through the login phase, a valid user is able to login to the server after the following calculation.

Step 1. The user inserts his/her SC_{ui}, ID_i and PW_i. Then, SC_{ui} computes $c = R_i \oplus (ID_i \| PW_i)$, $RP_i = h(c \| PW_i)$, $h(ID_i \| k_{s1}) = L_i \oplus RP_i \oplus c$, $k_i = K_i \oplus h(c \| PW_i)$ and $H_i^* = h(ID_i \| k_i \| RP_i)$.

Step 2. The smart card SC_{ui} compares $H_i^* = ?H_i$, if this condition does not hold, the smart card SC_{ui} stops this session.

Step 3. After that, the smart card SC_{ui} calculates $h(k_{s2} \| ID_i) = k_i \oplus J_i$, $G_i = L_i \oplus c$, $\overline{G_i} = G_i \oplus h(k_i \| T_{ui})$, $Q_i = h(G_i \| k_i \| P_i \| T_{ui})$, $P_i = G_i \oplus RP_i$, and $S_i = k_i \oplus (h(k_{s2} \| ID_i) \| T_{ui})$, transmits authentication request message $\{PID_i, \overline{G_i}, Q_i, S_i, T_{ui}\}$ to the server.

Step 4. Upon receiving the authentication request message, the server S validates time stamp T_{ui} and calculates $(ID_i \| T_{s0}) = D_{k_{s2}}(PID_i)$, $k_i = S_i \oplus (h(k_{s2} \| ID_i) \| T_{ui})$, $G_i = \overline{G_i} \oplus h(k_i \| T_{ui})$, $P_i^* = h(ID_i \| k_{s1})$ and $Q_i^* = h(G_i \| k_i \| P_i^* \| T_{ui})$.

Step 5. The server S checks whether Q_i is equal with Q_i^*. If they are equal, S authenticates the user U_i.

Step 6. The values $a = h(P_i^* \| k_i \| T_{s2})$ and $Z_i = P_i \oplus E_{k_{s2}}(ID_i \| T_{s1})$ are computed and $\{a, T_{s2}, Z_i\}$ are sent by the server to the user.

Step 7. After receiving $\{a, T_{s2}, Z_i\}$, initially the user verifies T_{s2}, then computes $a^* = h(P_i \| k_i \| T_{s2})$ and compares it with received $\{a\}$. If a^* is equal with received a, the user authenticates the server as a valid server.

Step 8. The server S and the user U_i calculates the shared key as $SK = h(P_i \| k_i \| T_{ui} \| T_{s2} \| h(k_{s2} \| ID_i))$.

3.3 Cryptanalysis of Chaudhry et al.'s scheme

Assume secret keys k_{s1} and k_{s2} of the server are disclosed, by executing the following calculations, the attacker is able to acquire the session key $SK = h(P_i \| k_i \| T_{ui} \| T_{s2} \| h(k_{s2} \| ID_i))$.

The attacker eavesdrops exchanged messages over a public channel and obtains values $\{PID_i, T_{ui}, T_{s2}, S_i, Z_i\}$. Then, the attacker decrypts PID_i with k_{s2} as $D_{k_{s2}}(PID_i) = (ID_i \| T_{s0})$ and evolves ID_i. For deriving k_i and P_i, the attacker computes $k_i = S_i \oplus (h(k_{s2} \| ID_i) \| T_{ui})$ and $P_i = PID_i \oplus Z_i$. Thus, in this scheme, perfect forward secrecy does not provide.

4 Review of Zhang et al.'s scheme

In this section, Initialization, registration and authentication phases of Zhang et al.'s scheme [48] is reviewed, then its security flaws are explained. Zhang et al.'s scheme [48] contains four phases: Initialization, registration, authentication, and password changing phases. The used notations in their scheme are listed in the Table 1.

4.1 Initialization phase

Step 1. An Elliptic Curve equation $E_p(a, b) : y^2 = x^3 + ax + b(mod\ p)$ over a prime finite field F_p, (where a, $b \in F_p$ and $4a^3 + 27b^2 \neq 0(mod\ p)$) are chosen by the SIP server. Then, the SIP server selects a base point P over $E_p(a, b)$.

Step 2. The server selects a high entropy random integer s as its secret key and a one-way hash function $h(.) : \{0, 1\}^* \rightarrow \{0, 1\}^k$, next computes $P_{pub} = sP$.

Step 3. The secret key s is protected by server and $\{E_p(a, b), P, P_{pub}, h(.)\}$ as public parameters are published.

4.2 Registration phase

In this phase, a new user registers in the SIP server and at the end of this phase, he/she receives a smart card.

Step 1. Initially, the user chooses his/her identity ID_i, password PW_i, a high entropy random integer r, and calculates $C_1 = h(PW_i \oplus r)$. Next, $\{ID_i, C_1\}$ is sent to the SIP server through a secure channel.

Step 2. The SIP server calculates C_2 and C_3 as $C_2 = h(ID_i \oplus s)$ and $C_3 = C_1 \oplus C_2$, respectively. Also, it stores C_3 in the memory of the smart card and sends the smart card to the user, via a secure channel.

Step 3. After receiving the smart card by the user, he/she saves r in to the smart card.

4.3 Authentication phase

At the end of this phase, the user and the SIP server agree to a same session key.

Step 1. The user inserts his/her smart card into the card reader, and inputs his/her ID_i and PW_i. The smart card chooses a high entropy random integer r_1, a random integer r_2, and calculates the following computations:

$$C_2 = C_3 \oplus h(PW_i \oplus r) = h(ID_i \oplus s),$$

$$C_4 = r_1 P,$$

$$C_5 = r_1 C_2 P_{pub},$$

$$C_6 = h(C_5) \oplus \Big(h(ID_i \oplus s) \oplus r_2 \big\| (C_5)_x \big\| (C_5)_y \Big).$$

where $(C_5)_x$ and $(C_5)_y$ are x/y-coordinates values of elliptic curve point C_5, respectively.

Finally, the request message REQUEST $\{ID_i, C_4, C_6\}$ is sent by the user to the SIP server through a public channel.

Step 2. The SIP server computes $C_2 = h(ID_i \oplus s)$ and retrieves $(h(ID_i \oplus s) \oplus r_2 \| (C_5)_x \| (C_5)_y)$ as $(h(ID_i \oplus s) \oplus r_2 \| (C_5)_x \| (C_5)_y = h(sC_2C_4) \oplus C_6$. Next, it verifies whether the condition $(C_5)_x \| (C_5)_y = (s\ C_2\ C_4)_x \| (s\ C_2\ C_4)_y$ holds or not. If it is not equal, the request is rejected. Otherwise, the SIP server acquires r_2 as $C_2 \oplus h(ID_i \oplus s) \oplus r_2$, then it selects two random integers (r_3, r_4) and calculates $C_7 = r_3 P$ and session key $SK = h(C_4 \| r_3\ C_4 \| C_7)$. In the following, the SIP server calculates an authentication message $Auth_s = h(h(ID_i \oplus s) \| r_2 \| (SK)_x \| (C_5)_x \| (SK)_y \| (C_5)_y)$ and forwards a challenge message CALLENGE $\{realm, C_7, Auth_s, r_4\}$ to the user U_i.

Step 3. Upon receiving the message CALLENGE $\{realm, C_7, Auth_s, r_4\}$, the smart card calculates the session key $SK = h(C_4 \| r_1\ C_7 \| C_7)$, and then it computes $h(C_2 \| r_2 \| (SK)_x \| (C_5)_x \| (SK)_y \| (C_5)_y)$, and checks the condition $h(C_2 \| r_2 \| (SK)_x \| (C_5)_x \| (SK)_y \| (C_5)_y) = ? Auth_s$. If it was true, the user U_i and the SIP server S agree to a same session key SK. The user U_i computes the authentication information $Auth_u = h((SK)_x \| (r_4 + 1) \| (SK)_y)$; otherwise, it aborts the session. At last, U_i transmits a response message RESPONSE $\{realm, Auth_u\}$ to the SIP server S.

Step 4. The SIP server checks whether $Auth_u$ is equal with $h((SK)_x \| (r_4 + 1) \| (SK)_y)$ or not. If it is not equivalent, the authentication process is terminated. Otherwise, the session key is equal to $SK = r_1\ r_3 P$.

4.4 Security weaknesses of Zhang et al.'s scheme

Lack of user anonymity and probability of user traceability At the end of first step of authentication phase of Zhang et al.'s scheme [48], identity of user is revealed in message REQUEST$\{ID_i, C_4, C_6\}$. So, the adversary is able to eavesdrop request message and obtain

identity ID_i of user. Also, the adversary can trace a certain user. Thus, in Zhang et al.'s scheme [48] both user anonymity and user untraceability are violated.

Ignoring re-registration and revocation The adversary can retrieve the identity of user, ID_i, because ID_i is sent as a plaintext via an insecure channel. So, if an outsider attacker selects a new password $PW_i^{'}$ and a new random integer r', he/she can register in the SIP server S using identity of legal user and the SIP server S cannot distinguish that formerly another user with same identity is registered and thus, it issues a new smart card for outsider attacker. Also, if user's smart card is stolen/lost, there is no mechanism to prevent misuse stolen/lost smart card.

Known-session-specific temporary information attack Assume a random number r_1 that it is chosen by smart card in the first step of authentication phase, is unexpectedly revealed to the adversary. Since the values C_4 and C_7 are sent through an insecure channel. The adversary is able to compare the value of r_1P with C_4. If r_1P is equal with C_4, the adversary can retrieve the session key SK as $h(\ C_4 \| \ r_1\ C_7 \| \ C_7)$.

Replay attack In authentication phase of Zhang et al.'s scheme [48], assume the adversary obtains overheard message REQUEST $\{ID_i, C_4, C_6\}$ and resent it to the SIP server S, he/she is able to login to the SIP server S. Because the SIP server S does not check the freshness of received message REQUEST $\{ID_i, C_4, C_6\}$. The server calculates all of computations in Step 2 of authentication phase and sends message CALLENGE$\{realm, C_7, Auth_s, r_4\}$. Therefore, time and energy of the SIP server S is wasted.

Violation of fast error detection In the first step of authentication phase of Zhang et al.'s scheme [48], the smart card calculates $(\ C_2, C_4, C_5, C_6)$ without verification through ID_i and PW_i, and sends REQUEST $\{ID_i, C_4, C_6\}$ to the server. An attacker can enter incorrect ID_i and PW_i, and creates denial of service attack, because the smart card doesn't check the entered information.

5 The proposed scheme

In this section, we explain our secure and efficient proposed scheme. The proposed scheme includes three phases: (1) registration phase, (2) authentication and key agreement phase, and (3) password update phase. The used symbols in the proposed scheme are shown in the Table 2.

5.1 Registration phase

The server and the user perform the following steps. At the end of this phase, server issues a smart card to the user.

Step 1: The user selects identity ID_i, password PW_i and two high-entropy random numbers r_i and b_i, then computes values RB and IP_i as $RB = h(\ r_i \| \ b_i)$ and $IP_i =$

Table 2 The notations used in proposed scheme

Symbol	Definition
U_i	User i
S	The SIP server
ID_i	Identity of the U_i
PW_i	Password of U_i
ID_{sc}	Identity of smart card
x_s	A high-entropy secret key of S
r_i, b_i, r_c, r_s	High-entropy random numbers
SK	The shared one-time session key
$E_k(.)/D_k(.)$	Symmetric key encryption/decryption by key k
$\oplus$	Bitwise XOR operation
$\|\|$	Concatenation operation
T_1, T_2, T_3, T_4, T_5	The current time of user's system/server's system
$h(.)$	Secure one-way hash function

$h(h(\ ID_i\|\|\ PW_i\|\|RB)\ mod\ m)$ where m is an integer between 2^8 and 2^{16} to avoid simultaneous guess identity and password [19, 25]. Next, he/she sends $\{ID_i, IP_i, b_i\}$ to the server through a secure channel.

Step 2: After receiving the registration request $\{ID_i, IP_i, b_i\}$, the server calculates A_i, B_i and NID_i as $A_i = h(\ SID_i\|\ x_s\|\ ID_{sc}\|\ ID_i)$, $B_i = A_i \oplus IP_i$ and $NID_i = ID_i \oplus IP_i$, then selects time stamp T_1 and computes $RID_i = E_{x_s}(\ ID_{sc}\|\ ID_i\|\ T_1)$. The server stores RID_i and B_i in the smart card and forwards it through a secure channel. Also, the server saves $\langle NID_i, status, b_i\rangle$ in its database. If the user is logged in, set a $status$ bit, otherwise $status = 0$. Note: If the user logged out and $status = 0$, it is impossible that the attacker logs in, because the proposed scheme provides user anonymity and resists against password guessing attack.

Step 3: when the user receives his/her smart card, computes $K_i = B_i \oplus r_i$, $W_i = h(h(\ b_i\|\ PW_i\|\ ID_i\|\ ID_{sc})\ mod\ m)$, and $H_i = W_i \oplus r_i$. Then, he/she deletes B_i and saves $\{b_i, K_i, H_i, ID_{sc}\ \}$ in the smart card. Finally, his/her smart card contains values $\{RID_i, b_i, K_i, H_i, ID_{sc}, E_{key}(.)/D_{key}(.), h(.)\ \}$. We suppose that database of server is secure and only the server has access to the database. The registration phase of proposed scheme is shown in Fig. 1.

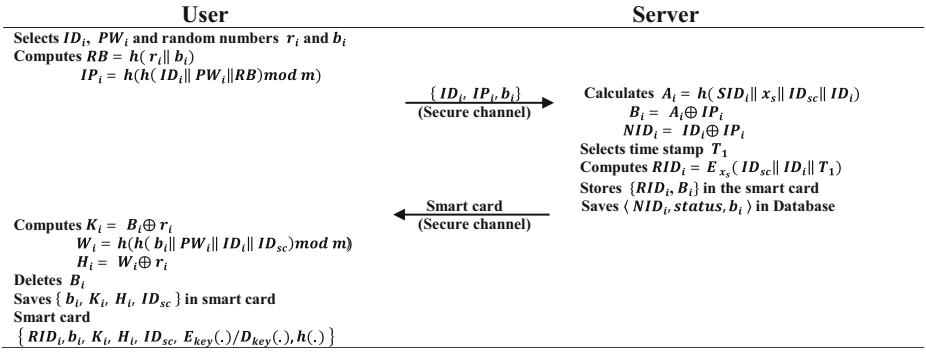

Fig. 1 Registration phase of proposed scheme

5.2 Authentication and key agreement phase

Step 1: In the beginning of this phase, the user inserts his/her smart card and next, enters his/her identity ID_i and password PW_i. Then, the smart card computes the following calculations:

$$W_i^* = h\big(h(\ b_i \big\| PW_i^* \big\| ID_i^* \big\| ID_{sc}^*) mod\ m\big),$$

$$r_i^* = H_i \oplus W_i^*,$$

$$B_i^* = K_i \oplus r_i^*,$$

$$RB^* = h\big(r_i^* \| \ b_i\big),$$

$$IP_i^* = h\big(h\big(ID_i^* \big\| PW_i^* \big\| RB^*\big) mod\ m\big),$$

$$A_i^* = B_i^* \oplus IP_i^*.$$

Then, it chooses a time stamp T_2 and a random number r_c, calculates $E_{A_i^*}(A_i^* \| r_c \| T_2 \big\| IP_i^*) = E A_i$ and forwards message REQUEST $\{EA_i, RID_i, T_2\}$ to the server via an insecure channel.

Step 2: When the server receives the login request message REQUEST $\{EA_i, RID_i, T_2\}$, checks whether $|T_3 - T_2| \le \Delta T$ is true or not. If this condition holds, the server decrypts RID_i with secret key x_s as $D_{x_s}(\ RID_i) = \big(ID_{sc}^* \big\| ID_i^* \big\| T_1\big)$. Next, it calculates $A_i^* = h(SID_i \| \ x_s \| \ ID_{sc} \| \ ID_i)$ and decrypts EA_i with A_i^* and extracts A_i, r_c, T_2' and IP_i. Then, the server compares A_i with A_i^*, and verifies $|T_3 - T_2'| \le \Delta T$, if these values are not equal, the session is aborted. Otherwise, the user is authenticated by the server. The server calculates $NID_i^* = ID_i \oplus IP_i$, searches NID_i^* in its database and extracts $\langle NID_i, status, b_i \rangle$, if the value of $status = 1$, then, the session is terminated, otherwise the server chooses a time stamp T_4 and a random number r_s. Afterward, it computes the following calculations and forwards a challenge message CHALLENGE $\{RID_i', T_4, EA_2\}$:

$$SK = h(A_i \| r_s \| r_c \| IP_i \| b_i),$$

$$m = h(IP_i \| r_c \| r_s \| SK),$$

$$EA_2 = E_{IP_i}(IP_i \| r_c \| r_s \| T_4),$$

$$RID_i^{'} = E_{x_s}(ID_{sc} \| ID_i \| T_4).$$

Step 3: When the user receives the challenge message at the time T_5, he/she verifies $|T_5 - T_4| \le \Delta T$ is true or not. If not, the session is aborted. Otherwise, the smart card decrypts EA_2 using IP_i and extracts (IP_i^*, r_c^*, r_s^*, T_4). Then, it compares IP_i^* with IP_i and r_c^* with r_c. If, $IP_i^* = IP_i$ and $r_c^* = r_c$, the smart card authenticates the server and calculates the session key SK and m as $SK = h(A_i \| r_s \| r_c \| IP_i \| b_i)$ and $m^* = h(IP_i \| r_c \| r_s \| SK)$ and replaces RID_i with RID_i'. Finally, the smart card sends response message RESPONSE $\{m^*\}$ to the server through a public channel.

Step 4: After receiving RESPONSE $\{m^*\}$, the server verifies condition $m^* = ?\, m$ is valid or not. If the condition holds, the server updates $status = 1$ and agrees with the user on the session key. After computing the session key, the server changes the value of *status* to zero. The authentication and key agreement phase of proposed scheme is shown in Fig. 2.

5.3 Password update phase

In this phase, a legal user is able to change his/her Password, securely.

Step 1: This step is same as the first step of authentication and key agreement phase.
Step 2: This step is same as the second step of authentication and key agreement phase.
Step 3: When the user receives challenge message at the time T_5, he/she verifies $|T_5 - T_4| \leq \Delta T$ is true or not. If not, the session is aborts. Otherwise, the smart card decrypts EA_2 using IP_i and extracts (IP_i^*, r_c^*, r_s^*, T_4). Then, it compares IP_i^* with IP_i and r_c^* with r_c. If, $IP_i^* = IP_i$ and $r_c^* = r_c$, the smart card authenticates the server and calculates $SK = h(A_i \| r_s \| r_c \| IP_i \| b_i)$. Then, the smart card requests the user to enter his/her new password PW_i^{new}, and calculates values (IP_i^{new}, B_i^{new}, W_i^{new}, K_i^{new}, H_i^{new}, NID_i^{new}) as following:

User		Server
Inserts smart card **Enters** ID_i **and** PW_i **Computes** $W_i^* = h(h(b_i \| PW_i^* \| ID_i^* \| ID_{sc}^*) mod\ m)$ $r_i^* = H_i \oplus W_i^*$ $B_i^* = K_i \oplus r_i^*$ $RB^* = h(r_i^* \| b_i)$ $IP_i^* = h(h(ID_i^* \| PW_i^* \| RB^*) mod\ m)$ $A_i^* = B_i^* \oplus IP_i^*$ **Selects a time stamp** T_2 **and a random number** r_c **Calculates** $E_{A_i^*}(A_i^* \| r_c \| T_2 \| IP_i^*) = EA_i$	**REQUEST** $\{EA_i, RID_i, T_2\}$ **(Insecure channel)**	**Checks** $\|T_3 - T_2\| \leq \Delta T$ **Decrypts** $RID_i\ D_{x_s}(RID_i) = (ID_{sc}^* \| ID_i^* \| T_1)$ **Calculates** $A_i^* = h\ SID_i \| x_s \| ID_{sc} \| ID_i)$ **Decrypts** $EA_i\ D_{A_i}(EA_i) = (A_i \| r_c \| T_2' \| IP_i)$ **Checks** $A_i = ?A_i^*$, **and** $\|T_3 - T_2'\| \leq \Delta T$ **Calculates** $NID_i^* = ID_i \oplus IP_i$, **Searches** NID_i^* **in database ,** **extracts**$\langle NID_i, status, b_i \rangle$ **Chooses a time stamp** T_4 **and a random number** r_s **Computes** $SK = h(A_i \| r_s \| r_c \| IP_i \| b_i)$ $EA_2 = E_{IP_i}(IP_i \| r_c \| r_s \| T_4)$ $RID_i' = E_{x_s}(ID_{sc} \| ID_i \| T_4)$ $m = h(IP_i \| r_c \| r_s \| SK)$
	CHALLENGE $\{RID_i', T_4, EA_2\}$ **(Insecure Channel)**	
Checks $\|T_5 - T_4\| \leq \Delta T$ **Decrypts**$EA_2\ D_{IP_i}(EA_2) = (IP_i^* \| r_c^* \| r_s^* \| T_4)$ **Checks** $IP_i^* = ?\, IP_i$ **and** $r_c^* = ?\, r_c$ **Calculates** $SK = h(A_i \| r_s \| r_c \| IP_i \| b_i)$ **and** $m^* = h(IP_i \| r_c \| r_s \| SK)$ **Replaces** RID_i **with** RID_i'		
	RESPONSE $\{m^*\}$ **(Insecure channel)**	**Checks** $m^* = ?\, m$ **Updates** $status = 1$
Agrees with the user on the session key $SK = h(A_i \| r_s \| r_c \| IP_i \| b_i)$		

Fig. 2 Authentication and key agreement phase of proposed scheme

$$IP_i^{new} = h(h(\ ID_i||PW_i^{new}||RB) mod\ m),$$

$$B_i^{new} = B_i^{old} \oplus IP_i^{old} \oplus IP_i^{new},$$

$$W_i^{new} = h(h(\ b_i||PW_i^{new}||\ ID_i||\ ID_{sc}) mod\ m),$$

$$K_i^{new} = B_i^{new} \oplus r_i,$$

$$H_i^{new} = W_i^{new} \oplus r_i,$$

$$NID_i^{new} = ID_i \oplus IP_i^{new}.$$

Finally, the smart card replaces (K_i, H_i) with (K_i^{new}, H_i^{new}), respectively, and encrypts ($b_i||NID_i^{new}$) with SK as $E_{SK}(\ b_i||NID_i^{new}) = ESK$. Next, it sends ESK to the server.

Step 4: The server decrypts ESK, with SK and extracts ($b_i||NID_i^{new}$). Then, it searches b_i in its database, extracts $\langle NID_i, status, b_i\rangle$ and replaces NID_i with NID_i^{new} in the database.

6 Security analysis

6.1 Informal security analysis

Anonymity In REQUEST, CHALLENGE and RESPONSE messages, identity of user will not be sent plain to the server. Also, if the attacker steals/finds smart card of user and extracts its stored information as $\{RID_i, b_i, K_i, H_i, ID_{sc}, E_{key}(.)/D_{key}(.), h(.)\}$,and interrupts the login request message REQUEST $\{EA_i, RID_i, T_2\}$, the challenge message CHALLENGE $\{RID_i', T_4, EA_2\}$, and response message RESPONSE $\{m^*\}$, he/she is not able to acquire identity of user ID_i.

$$RID_i = E_{x_s}(\ ID_{sc}||\ ID_i||\ T_1),$$

$$EA_i = E_{A_i^*}\left(A_i^*||r_c||T_2\middle||IP_i\right),$$

$$EA_2 = E_{IP_i}(IP_i||r_c||r_s||T_4),$$

$$RID_i^{'} = E_{x_s}(ID_{sc}||ID_i||T_4),$$

$$m^* = h(IP_i||r_c||r_s||SK).$$

For obtaining identity of user, ID_i, the attacker needs to have secret key of the server x_s, whereas secret key of the server x_s has been kept securely by the server, therefore, anonymity of user is preserved.

Perfect forward secrecy If user's password PW_i and server's secret key x_s is disclosed, without knowing random numbers r_s, r_c and b_i, the attacker cannot compute the session key $SK = h(A_i||r_s||r_c||IP_i||b_i)$. For obtaining r_s and r_c, the attacker should know IP_i, that $IP_i = h(h(ID_i||\ PW_i||RB)\ mod\ m)$. Since, value of RB is secret, thus, the attacker is not able to

compute IP_i. Therefore, the proposed scheme provides perfect forward secrecy. Also, the attacker has no way to find the value of r_i.

Known-key secrecy Since the session key is equal to $SK = h(A_i \| r_s \| r_c \| IP_i \| b_i)$ and random numbers r_s and r_c are generated randomly and different to selected random numbers in the other sessions, and also, random numbers r_s and r_c are not related to selected random numbers in previous sessions, even if the session key is disclosed, the attacker is not able to compute other session keys.

Session key security In our proposed scheme, only the user and the server know the session key. Whereas, the attacker is not able to calculate b_i, A_i and IP_i, therefore, he/she cannot obtain random numbers r_s and r_c, and cannot compute the session key.

Known-session-specific temporary information attack If session random numbers r_s, r_c and b_i are unexpectedly revealed to the attacker, he/she is not able to compute session key SK. Because the attacker cannot calculate IP_i and A_i. Since, the attacker doesn't know secret key of the server x_s, user's password PW_i and identity of user ID_i, the proposed scheme is robust against known-session-specific temporary information attack.

Offline guessing attack In our proposed scheme, if the attacker interrupts REQUEST, CHALLENGE and RESPONSE messages, he/she acquires following values.

$$RID_i = E_{x_s}(ID_{sc} \| ID_i \| T_1),$$

$$EA_i = E_{A_i^*}\left(A_i^* \| r_c \| T_2 \middle\| IP_i\right),$$

$$EA_2 = E_{IP_i}(IP_i \| r_c \| r_s \| T_4),$$

$$RID_i^{'} = E_{x_s}\left(ID_{sc} \| ID_i \middle\| T_4\right),$$

$$m^* = h(IP_i \| r_c \| r_s \| SK).$$

According to $IP_i = h(h(\ ID_i \| \ PW_i \| RB)\ mod\ m)$, for calculating user's password, the attacker should know IP_i, because the attacker does not aware A_i and B_i ($IP_i = A_i \oplus B_i$), he/she is not able to compute IP_i. So, using any of aforementioned messages, the attacker cannot acquire password of user.

Whereas stored parameters in the smart card are $\{RID_i, b_i, K_i, H_i, ID_{sc}, E_{key}(.)/D_{key}(.), h(.)\}$, even if the attacker finds/steals smart card, he/she is not able to find PW_i. Also, in our presented scheme, to avoidance guessing two parameters PW_i and ID_i, we use $mod\ m$ which $2^8 < m < 2^{16}$.

User impersonation attack If the attacker wants to impersonate user and send a reliable REQUEST and RESPONSE messages to the server. He/she needs to calculate valid EA_i^* and m^*. To compute a valid EA_i^*, the attacker should obtain PW_i, ID_i and RB. But, these values are kept securely by the user. Also, to calculate a valid m^*, the attacker should know IP_i, r_s, r_c, and SK. The attacker is not able to acquire session key SK and he/she does not know selected random numbers by the server, r_s, and the user, r_c.

Server impersonation attack If the attacker wants to forge the server, he/she should produce a valid challenge message as $\{RID_i'^*, T_4^*, EA_2^*\}$. To compute a valid EA_2^* and $RID_i'^*$, the attacker should know IP_i and the random number generated by the user as r_c. Since the attacker is not able to detect PW_i and ID_i, also, he/she doesn't know secret key of server, x_s, and RB, thus the attacker cannot impersonate the server. Therefore, the proposed scheme resists against impersonation attacks.

Stolen-verifier attack In our presented scheme, neither password nor secret key of the server x_s are stored in the server's database. Therefore the attacker can not retrieve the verification information.

Denning-sacco attack In this attack, the attacker tries to find a long term private key e.g. user's password or other session keys through obtained old session key. In the proposed scheme, if the attacker acquires old session key, he/she cannot retrieve user's password, server's secret key or other session keys. Because, the session key is equal to $SK = h(A_i \| r_s \| r_c \| IP_i \| b_i)$ that random numbers r_s and r_c are chosen by the server and the user, randomly. So, the attacker cannot detect other session keys using an old session key. Hence, the proposed scheme can resist against denning-sacco attack.

Replay attack Assume an attacker *A,* replays the old message REQUEST$\{EA_i, RID_i, T_2\}$ to the server. In our scheme, the server will find out that this message is repetitive and old. At first, the server verifies$|T_3 - T_2| \leq \Delta T$, and if this condition is not true, the session terminates. Even if the attacker changes T_2 with current time T_2^* and sends $\{EA_i, RID_i, T_2^*\}$ to the server, the server is able to distinguish that the message is old. The server computes A_i^*, decrypts EA_i with A_i^* and compares T_2' with T_3, (T_2' extracts from decryption EA_i). Since the attacker is not able to calculate new EA_i' with current time T_2^*, so $|T_3 - T_2'| \nleq \Delta T$, our proposed scheme is resist against replay attack.

Privileged insider attack Assume an insider obtains the registration information $\{ID_i, IP_i, b_i\}$ from the registration request message. The insider needs to obtain b_i and r_i to guess the user's password, but, b_i and r_i are random selected numbers by the legal user and they are high entropy. Hence, insider user of the server is not able to guess b_i and r_i during a polynomial time. As a result, our scheme is secure against privileged insider attack.

6.2 Formal security analysis using AVISPA tool

We prove security of our proposed scheme by means of AVIPSA tool [3]. The two widely-accepted back-ends, OFMC and CL-AtSe are selected for the execution tests and a bounded number of sessions model checking. For verification whether the replay attack is possible, these back-ends check whether the legitimate agents can execute the specified protocol by performing a search of a passive intruder. The back-ends then provide the intruder the knowledge of some normal sessions among the legitimate agents. For the Dolev–Yao model checking, these back-ends also verify possibility man-in-the-middle attack by the attacker. The output format contains the following sections:

- SUMMARY shows that the proposed scheme is safe, unsafe, or whether the analysis is also inconclusive.

- DETAILS denotes that under what condition the proposed scheme is safe, or what conditions are used to detect an attack, or why the result was inconclusive.
- PROTOCOL, GOAL and BACKEND are other sections for the protocol name, goal of the analysis and the back-end that has been used, respectively.
- Finally, the trace of an attack (if there is) is also presented in the standard Alice–Bob format [26]. Eventually, we simulate our scheme under OFMC and CL-AtSe back-ends using AVISPA tool.

6.2.1 Simulation results

In this section, we present the simulation results of our proposed scheme. Figs. 3 and 4 show the simulation results for the two back-ends OFMC and CL-AtSe. The results affirm that the proposed scheme is SAFE and secure against active and passive attacks, including replay attack and man-in-the-middle attack.

6.3 Formal security proof based BAN logic

Table 3 defines the symbols that are used in the BAN logic rules and the assumptions. In this section, using the BAN logic [7], we analyze our proposed scheme to affirm the correctness of the proposed protocol. In the following, the BAN logic's rules, assumptions, security goals, and idealized form are defined.

Rules:

- The message meaning rule: $\frac{P|\equiv P\overset{K}{\leftrightarrow}Q, P\lhd\{X\}_K}{P|\equiv Q|\sim X}$.
- The freshness rule: $\frac{P|\equiv\#(X)}{P|\equiv\#(X,Y)}$.
- The nonce verification rule: $\frac{P|\equiv\#(X), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$.
- The jurisdiction rule: $\frac{P|\equiv Q|\Rightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$.
- The belief rule: $\frac{P|\equiv(X), P|\equiv(Y)}{P|\equiv(X,Y)}$.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/niloo/Documents/avispa-1.1/testsuite/results/voip.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.02s
  visitedNodes: 4 nodes
  depth: 2 plies
```

Fig. 3 The simulation result of the analysis using *OFMC* of our proposed scheme

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/niloo/Documents/avispa-1.1/testsuite/results/voip.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 0 states
  Reachable  : 0 states
  Translation: 0.01 seconds
  Computation: 0.00 seconds
```

Fig. 4 The simulation result of the analysis using *CL-AtSe* of our proposed scheme

Assumptions:

- A1: $U_i|\equiv\left(U_i \overset{A_i}{\longleftrightarrow} S\right)$.
- A2: $S|\equiv\left(S \overset{A_i}{\longleftrightarrow} U_i\right)$.
- A3: $U_i \mid \equiv \#(r_c)$.
- A4: $U_i \mid \equiv \#(T_2)$.
- A5: $S \mid \equiv \#(r_s)$.
- A6: $S \mid \equiv \#(T_4)$.
- A7: $U_i|\equiv S \Rightarrow \left(U_i \overset{SK}{\longleftrightarrow} S\right)$.
- A8: $S|\equiv U_i \Rightarrow \left(U_i \overset{SK}{\longleftrightarrow} S\right)$.
- A9: $U_i|\equiv\left(U_i \overset{IP_i}{\longleftrightarrow} S\right)$.
- A10: $S|\equiv\left(S \overset{IP_i}{\longleftrightarrow} U_i\right)$.
- A11: $U_i|\equiv\left(U_i \overset{b_i}{\longleftrightarrow} S\right)$.
- A12: $S|\equiv\left(S \overset{b_i}{\longleftrightarrow} U_i\right)$.

Table 3 The BAN notations

Symbol	Description
$P \mid \equiv X$	The principal P believes a statement X.
$P \lhd X$	The principal P sees a statement X.
$P \mid \sim X$	The principal P once said a statement X.
$P \Rightarrow X$	The principal P has jurisdiction over X.
$\#(X)$	The message X is fresh.
$P \equiv Q \overset{K}{\longleftrightarrow} P$	The secret key K is used by P and Q for communicating.
$\{X\}_K$	The formula X is encrypted with the key K.
$<X>_K$	The formula X is xored with the key K.
$(X)_K$	The formula X is hashed with the key K.

Goals:

- Goal1: $U_i|\equiv\left(U_i\overset{SK}{\longleftrightarrow}S\right)$.
- Goal2: $S|\equiv\left(U_i\overset{SK}{\longleftrightarrow}S\right)$.

Idealized form:

- Message1: $U_i\rightarrow S:\left(\{A_i,r_c,T_2,IP_i\}_{A_i},\{ID_{sc},ID_i,T_1\}_{x_s},T_2\right)$.
- Message2: $S\rightarrow U_i:\left(\{ID_{sc},ID_i,T_4\}_{x_s},T_4,\{IP_i,r_c,r_s,T_4\}_{IP_i}\right)$.
- Message3: $U_i\rightarrow S:\left(\left(r_c,r_s,U_i\overset{SK}{\longleftrightarrow}S\right)_{IP_i}\right)$.

According to the BAN logic rules and the assumptions, the idealized form of the proposed scheme are analyzed as follows:

Based on the message1, we can achieve the following:

R1) $S\triangleleft\left(\{A_i,r_c,T_2,IP_i\}_{A_i},\{ID_{sc},ID_i,T_1\}_{x_s},T_2\right)$

According to the assumption A2, after exerting the message meaning rule to R1, we can conclude R2:

R2) $S\mid\equiv U_i\mid\sim(A_i,r_c,T_2,IP_i)$.

Based on the message2, we can derive R3:

R3) $U_i\triangleleft\left(\{ID_{sc},ID_i,T_4\}_{x_s},T_4,\{IP_i,r_c,r_s,T_4\}_{IP_i}\right)$.

Based on assumption A9, after applying the message meaning rule to R3, we can conclude R4:

R4) $U_i|\equiv S|\sim(IP_i,r_c,r_s,T_4)$.

According to A3, after applying the nonce verification rule to R4, we can derive R5:

R5) $U_i|\equiv S|\equiv(IP_i,r_c,r_s,T_4)$.

Based on R5, assumptions A2, A12, and session key $SK=h(A_i\|r_s\|r_c\|IP_i\|b_i)$, R6 is concluded:

R6) $U_i|\equiv S|\equiv\left(U_i\overset{SK}{\longleftrightarrow}S\right)$.

According to assumption A7, we apply the jurisdiction rule to R6 and we can conclude the goal1:

R7) $U_i|\equiv\left(U_i\overset{SK}{\longleftrightarrow}S\right)$.(Goal1)

Based on the message3, we can achieve R8:

$$\text{R8) } S \lhd \left(\left(r_c, r_s, U_i \overset{SK}{\longleftrightarrow} S \right)_{IP_i} \right).$$

According to assumption A10, we apply the message meaning rule to R8, and derive R9:

$$\text{R9) } S \mid \equiv U_i \mid \sim \left(r_c, r_s, U_i \overset{SK}{\longleftrightarrow} S \right).$$

Based on A5, after applying the nonce verification rule to R9, R10 is retrieved as follow:

$$\text{R10) } S \mid \equiv U_i \mid \equiv \left(r_c, r_s, U_i \overset{SK}{\longleftrightarrow} S \right).$$

We apply the belief rule on R10 and can obtain R11:

$$\text{R11) } S \mid \equiv U_i \mid \equiv \left(U_i \overset{SK}{\longleftrightarrow} S \right).$$

According to assumption A8, after applying the jurisdiction rule to R11, we can achieve to goal2:

$$\text{R12) } S \mid \equiv \left(U_i \overset{SK}{\longleftrightarrow} S \right). (\text{Goal2})$$

7 Analysis of performance and features

The notations T_{hf}, T_{mu}, T_{ad}, $T_{en/d}$, T_{mm}, T_{inv} are considered to show the computing complexity of one-way hash function, performing the scalar multiplication operation of elliptic curve, a point addition operation of elliptic curve, performing symmetric encryption/decryption and the modular multiplication and modular inversion, respectively. Due to the low computation cost of concatenation (||) and XOR operation, their cost is not considered. According to [2, 40], execution time of T_{hf}, T_{mu}, T_{ad}, $T_{en/d}$, T_{mm}, T_{inv} are 0.0004 *ms*, 7.3529 *ms*, 0.009 *ms*, 0.1303 *ms*, 0.0147 *ms*, and 0.1032 *ms*, respectively.

The computation costs of recent protocols including the registration, authentication, and key agreement phases for clients and server are represented in Table 4.

As it is shown in Table 4 and Fig. 5, our proposed scheme comes third in terms of computation cost. The protocol of Chaudhry et al. [8] has the least calculation costs but Chaudhry et al.'s scheme [8] and Nikooghadam et al.'s scheme [28] do not provide the perfect forward secrecy security requirement.

According to [40], communication costs for sending identity ID_i is considered to be 160 bits, for timestamp 32-bits, for symmetric encryption and decryption operations is 128-bits, elliptic curve point multiplication and output hash function are 320 bits and 160 bits, respectively.

According to the Table 5, in the login phase of our proposed scheme, three messages will be sent. Communication cost of our proposed scheme based on the given aforementioned numbers is:

The sending request: $\left\lceil \frac{160+128}{128} \right\rceil * 128 + \left\lceil \frac{160+128}{128} \right\rceil * 128 + 32 = 800$ bits.
The sending challenge: $32 + \left\lceil \frac{160+128}{128} \right\rceil * 128 + \left\lceil \frac{160+128}{128} \right\rceil * 128 = 800$ bits.
The sending response: 160 bits.

Table 4 Computation cost comparison between recent protocols and proposed protocol

Schemes	Computation cost			Time (ms)
	Registration phase	Login and Authentication phase	Total	
The proposed	$6T_{hf}+T_{en/d}$	$11T_{hf}+6T_{en/d}$	$17T_{hf}+7T_{en/d}$	0.9189
[38]	$2T_{hf}+T_{mu}$	$10T_{hf}+7T_{mu}+T_{ad}$	$12T_{hf}+8T_{mu}+T_{ad}$	58.837
[8]	$4T_{hf}+T_{en/d}$	$14T_{hf}+2T_{ed/n}$	$18T_{hf}+3T_{ed/n}$	0.3981
[18]	$3T_{hf}+2T_{mu}+T_{inv}$	$9T_{hf}+7T_{mu}+2T_{ad}+3T_{mm}$	$12T_{hf}+9T_{mu}+2T_{ad}+3T_{mm}+T_{inv}$	66.3462
[5]	$2T_{hf}+T_{inv}$	$10T_{hf}+6T_{mu}+2T_{mm}$	$12T_{hf}+6T_{mu}+2T_{mm}+T_{inv}$	44.2548
[28]	$2T_{hf}+T_{en/d}$	$6T_{hf}+6T_{en/d}$	$8T_{hf}+7T_{en/d}$	0.9153
[48]	$2T_{hf}$	$9T_{hf}+6T_{mu}$	$11T_{hf}+6T_{mu}$	44.1218
[9]	$3T_{hf}$	$4T_{hf}+6T_{mu}$	$7T_{hf}+6T_{mu}$	44.1202
[41]	$3T_{hf}$	$6T_{hf}+6T_{mu}$	$9T_{hf}+6T_{mu}$	44.121
[24]	$3T_{hf}$	$7T_{hf}+7T_{mu}+2T_{en/d}$	$10T_{hf}+7T_{mu}+2T_{en/d}$	53.7776
[20]	$2T_{hf}$	$10T_{hf}+4T_{mu}$	$12T_{hf}+4T_{mu}$	29.4164

According to the performed computation, the cost of communication in our proposed protocol is 1760 bits. Communication costs of the other protocols are calculated in the same way and given in Table 5.

Although the communication cost of the proposed protocol in comparison with schemes [8, 9, 20, 28, 41, and] is increased, the schemes [9] and [41] are vulnerable

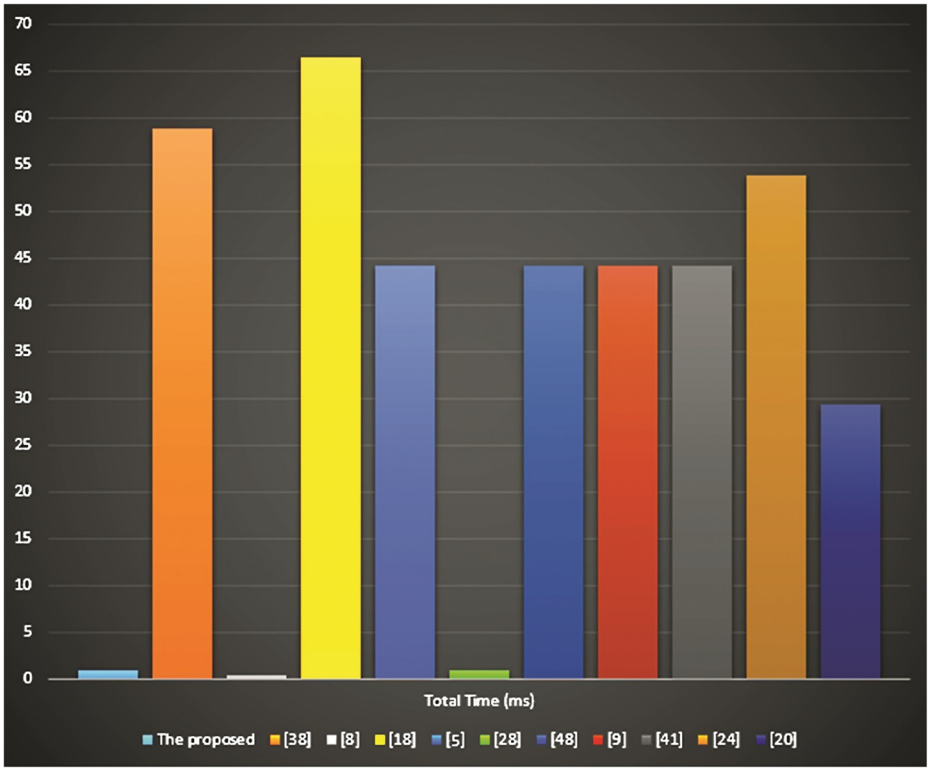

Fig. 5 Comparison of execution time between our proposed protocol and other protocols

Table 5 Communication costs comparison between recent protocols and proposed protocol

Schemes	Communication cost			Total (Bits)
	REQUEST	CHALLENGE	RESPONSE	
The proposed	EA_i, RID_i, T_2	EA_2, $RID_i^{'}$, T_4	M	1760
[8, 18, 38]	*Username*, *V*, *W*	$Auth_s$, *C*, *r*, *realm*	*realm*, $Auth_u$	1760
	$G_i^{'}$, PID, Q_i, s_i, T_{ui}	*A*, z_i, Ts_2	*realm*, $Auth_u$	1472
	Username, *V*, *W*	$Auth_s$, *S*, *r*, *realm*		1920
[5, 28, 48]	R_u, T_u, ID_u, V_1	R_s, *realm*, V_2	ID_u, *realm*, V_3	1792
	C_4, C_6, *ID*	C_7, $Auth_s$, r_4, *realm*	*realm*, $Auth_u$	1760
	MID, M_1, T_1	M_2	M_3	1344
[9]	$Auth_u$, *DID*, T_x, *B*	$Auth_s$, *E*		1312
[41]	$Auth_u$, *DID*, T_x, *B*	$Auth_s$, *E*		1152
[24]	*M*, *T*, *U*	H_s, *realm*, *Auth*	V_u, *realm*	1824
[20]	*B*, *Hld*, *C*, *PId*, T_u	$AUTH_s$, *realm*, *G*, T_s	$AUTH_u$, *realm*	1696

against known session specific temporary information attack and ignore re-registration and revocation, scheme [20] is vulnerable against ignoring re-registration and revocation, schemes [8] and [28] do not provide the perfect forward secrecy. As shown in Table 5, this additional communication cost provides all security aspects that are provided by the proposed scheme.

The performed analysis of the recent protocols are presented in Table 6. As it can be observed in Table 6, our suggested protocol is resistant to all of the attacks and provides security requirements such as perfect forward secrecy and known key secrecy. Moreover, the user anonymity is provided in our presented protocol. Therefore, our proposed scheme is a secure scheme with reasonable cost among all latest authentication schemes.

Table 6 Functionality comparison between recent protocols and proposed protocol

Functionality comparison	[38]	[8]	[18]	[48]	[28]	[9]	[41]	[24]	[20]	The Proposed
FC_1	–	*	*	–	*	*	*	*	*	*
FC_2	*	–	*	*	*	*	*	*	*	*
FC_3	*	*	*	*	*	*	*	*	*	*
FC_4	*	*	*	*	*	*	*	*	*	*
FC_5	*	*	*	–	*	–	–	–	*	*
FC_6	*	*	*	*	*	*	*	*	*	*
FC_7	*	–	*	*	–	*	*	*	*	*
FC_8	*	*	*	*	*	*	*	*	*	*
FC_9	*	*	*	*	*	*	*	*	*	*
FC_{10}	*	*	*	*	*	*	*	*	*	*
FC_{11}	–	*	–	–	*	–	–	–	–	*
FC_{12}	–	*	–	*	*	*	*	*	*	*
FC_{13}	–	*	*	–	*	*	*	*	*	*

FC_1: Replay attack resistance; FC_2: Off-line password guessing attack resistance; FC_3: Stolen verifier attack resistance; FC_4: Denning sacco attack resistance; FC_5: Known session specific temporary information attack resistance; FC_6: Providing known key secrecy; FC_7: Providing perfect forward secrecy; FC_8: Privileged insider attack resistance; FC_9: Man in the middle attack resistance; FC_{10}: Providing mutual authentication; FC_{11}: Ignoring re-registration and revocation; FC_{12}: User/Server impersonation attack resistance; FC_{13}: Providing user anonymity.

*: The scheme is secure or supports this feature; −: The scheme is insecure or does not provide this property

8 Conclusion

In this paper, we first investigate the security weaknesses in the represented protocols by Chaudhry et al., Nikooghadam et al. and Zhang et al. As it was investigated, the presented protocols by Chaudhry et al. and Nikooghadam et al. do not provide the perfect forward secrecy and the represented protocol by Zhang et al. does not resist against the replay attack, known session-specific temporary information attack and also, does not provide user anonymity, fast error detection and ignores re-registration and revocation. Therefore, in order to resolve the mentioned security flaws, we present an authentication and key agreement scheme based on password and smart card. Then, the presented protocol is analyzed formally and informally. Security verifications show security of the proposed protocol against various attacks. Informal proof shows that the presented scheme provides significant security requirements. Then, we show that the presented scheme has low calculating costs as it does not use expensive operators in elliptic curve including multiplication on elliptic curve. Thus, we present an efficient scheme that it provides security and has low computation and communication costs, therefore, this is an efficient and secure scheme for VoIP.

References

1. Amin R, Islam SH, Biswas G, Khan MK, Kumar N (2015) An efficient and practical smart card based anonymity preserving user authentication scheme for TMIS using elliptic curve cryptography. J Med Syst 39(11):1–18
2. Amin R, Islam SH, Biswas G, Khan MK, Obaidat MS (2015) Design and analysis of an enhanced patient-server mutual authentication protocol for telecare medical information system. J Med Syst 39(11):1–20
3. Armando A, Basin D, Boichut Y, Chevalier Y, Compagna L, Cuéllar J, Drielsma PH, Héam P-C, Kouchnarenko O, Mantovani J (2005) The AVISPA tool for the automated validation of internet security protocols and applications. In: International Conference on Computer Aided Verification. Springer, pp 281–285
4. Arshad R, Ikram N (2013) Elliptic curve cryptography based mutual authentication scheme for session initiation protocol. Multimedia Tools Appl 66(2):165–178
5. Arshad H, Nikooghadam M (2015) Security analysis and improvement of two authentication and key agreement schemes for session initiation protocol. J Supercomput 71(8):3163–3180
6. Arshad H, Nikooghadam M (2016) An efficient and secure authentication and key agreement scheme for session initiation protocol using ECC. Multimedia Tools Appl 75(1):181–197
7. Burrows M, Abadi M, Needham RM (1989) A logic of authentication. Proc R Soc Lond A 426(1871):233–271
8. Chaudhry SA, Farash MS, Naqvi H, Kumari S, Khan MK (2015) An enhanced privacy preserving remote user authentication scheme with provable security. Security and Communication Networks 8(18):3782–3795
9. Chaudhry SA, Khan I, Irshad A, Ashraf MU, Khan MK, Ahmad HF (2016) A provably secure anonymous authentication scheme for session initiation protocol. Security and Communication Networks 9(18):5016–5027
10. Durlanik A, Sogukpinar I (2005) SIP authentication scheme using ECDH. Screen 137:3367
11. Farash MS (2016) Security analysis and enhancements of an improved authentication for session initiation protocol with provable security. Peer-to-Peer Networking and Applications 9(1):82–91
12. Farash MS, Kumari S, Bakhtiari M (2016) Cryptanalysis and improvement of a robust smart card secured authentication scheme on SIP using elliptic curve cryptography. Multimedia Tools and Applications 75(8):4485–4504

13. Franks J, Hallam-Baker P, Hostetler J, Lawrence S, Leach P, Luotonen A, Stewart L (1999) HTTP authentication: Basic and digest access authentication
14. Geneiatakis D, Dagiuklas T, Kambourakis G, Lambrinoudakis C, Gritzalis S, Ehlert KS, Sisalem D (2006) Survey of security vulnerabilities in session initiation protocol. IEEE Communications Surveys & Tutorials 8(3):68–81
15. He D, Chen J, Chen Y (2012) A secure mutual authentication scheme for session initiation protocol using elliptic curve cryptography. Security and Communication Networks 5(12):1423–1429
16. Irshad A, Sher M, Faisal MS, Ghani A, Ul Hassan M, Ashraf Ch S (2014) A secure authentication scheme for session initiation protocol by using ECC on the basis of the tang and Liu scheme. Security and Communication Networks 7(8):1210–1218
17. Irshad A, Sher M, Rehman E, Ch SA, Hassan MU, Ghani A (2015) A single round-trip SIP authentication scheme for voice over internet protocol using smart card. Multimedia Tools and Applications 74(11):3967–3984
18. Jiang Q, Ma J, Tian Y (2015) Cryptanalysis of smart-card-based password authenticated key agreement protocol for session initiation protocol of Zhang et al. Int J Commun Syst 28(7):1340–1351
19. Jiang Q, Khan MK, Lu X, Ma J, He D (2016) A privacy preserving three-factor authentication protocol for e-health clouds. J Supercomput:1–24
20. Kumari S, Karuppiah M, Das AK, Li X, Wu F, Gupta V (2017) Design of a secure anonymity preserving authentication scheme for session initiation protocol using elliptic curve cryptography. J Ambient Intell Humaniz Comput:1–11
21. Liu F, Koenig H (2011) Cryptanalysis of a SIP authentication scheme. In: IFIP International Conference on Communications and Multimedia Security. Springer, pp 134–143
22. Lu Y, Li L, Peng H, Yang Y (2015) An anonymous two-factor authenticated key agreement scheme for session initiation protocol using elliptic curve cryptography. Multimedia Tools and Applications:1–15
23. Lu Y, Li L, Peng H, Yang Y (2016) A secure and efficient mutual authentication scheme for session initiation protocol. Peer-to-Peer Networking and Applications 9(2):449–459
24. Lu, Y., Li, L., Peng, H., & Yang, Y. 2016. An advanced elliptic curve cryptography based mutual authentication scheme for session initiation protocol. ITC 45, 4, pp. 393–400)
25. Mishra D (2015) On the security flaws in id-based password authentication schemes for telecare medical information systems. J Med Syst 39(1):1–16
26. Mishra D, Das AK, Chaturvedi A, Mukhopadhyay S (2015) A secure password-based authentication and key agreement scheme using smart cards. Journal of Information Security and Applications 23: 28–43
27. Mishra D, Das AK, Mukhopadhyay S (2016) A secure and efficient ECC-based user anonymity-preserving session initiation authentication protocol using smart card. Peer-to-peer networking and applications 9(1): 171–192
28. Nikooghadam M, Jahantigh R, Arshad H (2016) A lightweight authentication and key agreement protocol preserving user anonymity. Multimedia Tools and Applications:1–23
29. Odelu V, Das AK, Goswami A (2015) An efficient ECC-based privacy-preserving client authentication protocol with key agreement using smart card. Journal of Information Security and Applications 21:1–19
30. Pu Q, Wang J, Wu S (2013) Secure SIP authentication scheme supporting lawful interception. Security and Communication Networks 6(3):340–350
31. Rosenberg J, Schulzrinne H, Camarillo G, Johnston A, Peterson J, Sparks R, Handley M, Schooler E (2002) SIP: session initiation protocol
32. Ruan O, Kumar N, He D, Lee J-H (2015) Efficient provably secure password-based explicit authenticated key agreement. Pervasive and Mobile Computing 24:50–60
33. Salsano S, Veltri L, Papalilo D (2002) SIP security issues: the SIP authentication procedure and its processing load. IEEE Netw 16(6):38–44
34. Sisalem D, Kuthan J, Ehlert S (2006) Denial of service attacks targeting a SIP VoIP infrastructure: attack scenarios and prevention mechanisms. IEEE Netw 20(5):26–31
35. Sutrala AK, Das AK, Odelu V, Wazid M, Kumari S (2016) Secure anonymity-preserving password-based user authentication and session key agreement scheme for telecare medicine information systems. Comput Methods Prog Biomed 135:167–185
36. Tang H, Liu X (2013) Cryptanalysis of Arshad et al.'s ECC-based mutual authentication scheme for session initiation protocol. Multimedia tools and applications 65(3):321–333
37. Tsai JL (2009) Efficient Nonce-based Authentication Scheme for Session Initiation Protocol. IJ Network Security 9(1):12–16
38. Tu H, Kumar N, Chilamkurti N, Rho S (2015) An improved authentication protocol for session initiation protocol using smart card. Peer-to-Peer Networking and Applications 8(5):903–910

39. Wu L, Zhang Y, Wang F (2009) A new provably secure authentication and key agreement protocol for SIP using ECC. Computer Standards & Interfaces 31(2):286–291
40. Xu L, Wu F (2015) Cryptanalysis and improvement of a user authentication scheme preserving uniqueness and anonymity for connected health care. J Med Syst 39(2):1–9
41. Xu D, Zhang S, Chen J, Ma M (2017) A provably secure anonymous mutual authentication scheme with key agreement for SIP using ECC. Peer-to-Peer Networking and Applications:1–11
42. Yang C-C, Wang R-C, Liu W-T (2005) Secure authentication scheme for session initiation protocol. Computers & Security 24(5):381–386
43. Yeh H-L, Chen T-H, Shih W-K (2014) Robust smart card secured authentication scheme on SIP using elliptic curve cryptography. Computer Standards & Interfaces 36(2):397–402
44. Yoon E-J, Yoo K-Y (2009) Cryptanalysis of DS-SIP authentication scheme using ECDH. In: New Trends in Information and Service Science. NISS'09. International Conference on, 2009. IEEE, pp 642–647
45. Yoon E-J, Yoo K-Y, Kim C, Hong Y-S, Jo M, Chen H-H (2010) A secure and efficient SIP authentication scheme for converged VoIP networks. Comput Commun 33(14):1674–1681
46. Zhang L, Tang S, Cai Z (2014) Efficient and flexible password authenticated key agreement for voice over internet protocol session initiation protocol using smart card. Int J Commun Syst 27(11):2691–2702
47. Zhang L, Tang S, Cai Z (2014) Cryptanalysis and improvement of password-authenticated key agreement for session initiation protocol using smart cards. Security and Communication Networks 7(12):2405–2411
48. Zhang L, Tang S, Zhu S (2016) An energy efficient authenticated key agreement protocol for SIP-based green VoIP networks. J Netw Comput Appl 59:126–133

Niloofar Ravanbakhsh received the B.Sc. degree in Information Technology Engineering (IT) from Birjand University, Birjand, Iran, in 2012, and M.Sc. degree in Secure Communication from Imam Reza International University, Mashhad, Iran, in 2016. Her research interests include Cryptography, Data Security and Authentication Protocols. Email: n.ravanbakhsh@imamreza.ac.ir

Mohadeseh Mohammadi received the B.Sc. degree in Information Technology Engineering (IT) from Birjand University of Technology, Birjand, Iran, in 2014, and M.Sc. degree in Information Security of Imam Reza International University, Mashhad, Iran in 2017. Her research interests include Cryptography, Data Security and Authentication Protocols. Email: m.mohammadi@imamreza.ac.ir

Morteza Nikooghadam received the B.Sc. degree from university of Sadjad, Iran, in 2006, M.Sc. from the Shahid Beheshti University, Iran, in 2008, and Ph.D. from Shahid Beheshti University, Iran, in 2012. He is currently an assistant professor in the Department of Computer Engineering and Information Technology at Imam Reza International University, Mashhad, Iran. His research focuses on Data Security, Cryptography, and Sensor Network Security. His current research interests are Reconfigurable Architectures for multipliers under Galois Field GF(2^m). Email: m.nikooghadam@imamreza.ac.ir