An Improved Authentication and Key Agreement scheme for Session Initial Protocol

Libing Wu[1,2 *], Jing Fan[2], Yong Xie[2,3] and Jing Wang[2]
[1] State Key Laboratory of Software Engineering, Wuhan University
Wuhan 430072, China
[e-mail: wu@whu.edu.cn]
[2] School of Computer Science, Wuhan University
Wuhan, China
[e-mail: wu@whu.edu.cn]
[3] Jingdezhen Ceramic Institute
Jingdezhen 333403, China
[e-mail: xieyongdian@whu.edu.cn]
*Corresponding author: Libing Wu

Received November 6, 2016; revised March 19, 2017; accepted April 17, 2017; published August 31, 2017

Abstract

Session initiation protocol (SIP) is a kind of powerful and common protocols applied for the voice over internet protocol. The security and efficiency are two urgent requirements and admired properties of SIP. Recently, Hamed et al. proposed an efficient authentication and key agreement scheme for SIP. However, we demonstrate that Hamed et al.'s scheme is vulnerable to de-synchronization attack and cannot provide anonymity for users. Furthermore, we propose an improved and efficient authentication and key agreement scheme by using elliptic curve cryptosystem. Besides, we prove that the proposed scheme is provably secure by using secure formal proof based on Burrows-Abadi-Needham logic. The comparison with the relevant schemes shows that our proposed scheme has lower computation costs and can provide stronger security.

Keywords: Session initial protocol; Mutual authentication; VoIP; Anonymity; Elliptic curve cryptosystem

This work is supported by Natural Science Foundation of China(61472287,61572370), Science and Technology Support Program of Hubei Province (2015CFA068) and Science and Technology Plan Projects of Wuhan City(2016060101010047)

https://doi.org/10.3837/tiis.2017.08.015 **ISSN : 1976-7277**

1. Introduction

Session initiation protocol (SIP) [1] is a signaling protocol that used to create, modify and terminate multimedia sessions among one or more users. It is widely used by many Internet telephony service providers (ITSPs) to support their voice over Internet protocol (VoIP) services. With fast-developing of communication systems based on VoIP, the protection of communicating information in VoIP communication is of vital importance, because the messages of VoIP are transmitted through public channels. This makes VoIP are vulnerable to various security attacks, privacy-leaking and compromise. Therefore, prior to deployment of the VoIP, the security and privacy issues should be solved.

As one of significant roles for ensuring the communication security of VoIP, session initiation protocol (SIP), first proposed in 1999 [2], has aroused wide public concern. Deriving from HTTP digest authentication [3], SIP adopted an insecure channel to send packets to an intended recipient, which resulted in some vulnerabilities against various attacks -- A malicious attacker can eavesdrop communication messages, and further reveal user privacy, modify authentication messages, crack the server and impersonate identities [1,4,5]. Thus, to achieve the smooth functioning, a robust secure mechanism is urgently required for SIP. Fortunately, more and more attentions have been paid on the security during the last decade.

Recently, Hamed *et al.*'s [20] analyzed the authentication protocol proposed by Irshad *et al.* [21], and further put forward a secure and efficient scheme. However, we demonstrate that Hamed *et al.*'s scheme has a few security loopholes, then we propose an enhanced security scheme over Hamed *et al.*'s scheme.

In a nutshell, there are threefold contributions of this paper:

1. We demonstrate that Hamed *et al.*'s scheme is apt to suffer from the de-synchronization attack and cannot provide user anonymity, which leads to other privacy problems.
2. We propose an enhanced security scheme over Hamed *et al.*'s scheme as our main contribution. Our proposed scheme can provide user anonymity and resist de-synchronization attack, the performance evaluation show that our scheme has lower computation costs while makes up the missing security requirement on Hamed *et al.*'s scheme for real-life applications.
3. The proposed scheme is proved to be secure by using secure formal proof based on Burrows-Abadi-Needham logic [22,23,24] .

The rest of this paper is as follows. Sect. 2 describe the relate work about SIP. Sect. 3 reviews Hamed *et al.*'s scheme. Sect. 4 analyses the weakness of this scheme. Our enhanced security scheme is proposed in Sect. 5. Sect.6 demonstrates the security proof. The security analysis and comparisons are discussed in Sect. 7. Finally, conclusion is drawn in Sect. 8.

2. Relate Work

As the authentication is one of most significant and essential parts of SIP, there are many relative researches are proposed [2,8,9,10,15,20,23,25,26,27]. The HTTP digest authentication for SIP was first proposed by Franks *et al.* [2] and has becoming one of most common authentication mechanism in terms of SIP. However, it is apt to suffer from off-line password guessing attack and cannot provide mutual authentication. Yang *et al.* [8] proposed a new authentication scheme based on the computational difficulty of discrete logarithm problem (DLP), nevertheless, their scheme is demonstrated to be subject to the Denning-Sacco

attack and stolen verifier attack [9]. Durlarik in [9] put forward an improved authentication scheme over Yang *et al.*'s scheme by using elliptic curve discrete logarithm problem (ECDLP).They proved their scheme embodied the superiority on both computational efficiency and security than Yang's scheme. But Durlarik's scheme still suffers from the Denning-Sacco attack and stolen verifier attack [10]. Wu *et al.* [11] presented a provably secure authentication by using elliptic curve cryptosystem (ECC)to achieve the goal of low computational overhead, but their scheme is vulnerable to suffer offline password guessing attack [12,13]. An improve scheme over Wu *et al.*'s scheme was proposed in [12], unfortunately, it still has the same vulnerabilities with Wu *et al.*'s scheme. Choi *et al.* [13] proposed an authentication scheme by using multiple keys for encryption, authentication and integrity checking. Most of authentication schemes based on ECDLP can resist offline password guessing attacks except the scheme proposed in [11]. Xie [14] proposed an improved secure authentication scheme based on elliptic curve cryptosystems. While it is demonstrated to be insecure against the impersonation attack and off-line password guessing attack [15].

With privacy-preserving concerns being raised rapidly among individuals and organizations in recent years [16], the privacy-preserving has becoming an urgent requirement and admired property of SIP. What is more, the privacy-preserving concerns are no longer just a unique notion of "keep user being anonymous", but also "user identity protection", "user un-traceability" and "Non-linkability of user information" [17,18]. As we know, users would like to keep their information based on SIP in secrete forever, such as identity, time, location and so on. However, most of the existing researches on SIP pay most attention on security of authentication, and overlook some potential attacks, such as privacy leakage and de-synchronization attack. To the best of our knowledge, the de-synchronization attack is a new kind of inconspicuous attack but has aroused many intensive discussions [18,19]. It should emphasize the importance of potential threats when designing a SIP [21,28]. However, there is a crux that how to maintain the privacy-preserving property without adding computation costs and incurring de-synchronization problem.

Nowadays, SIP has also been gradually extended into video session, instant messaging, file transmission and even multiplayer games. And also, many new technology has been imported for current researches, such as three-factor authentication [31], smart card [32].

3. Review of Hamed *et al.*'s scheme

In this section, we will succinctly review Hamed *et al.*'s scheme for SIP [20]. Their scheme consists of four phases: system setup phase, registration phase, authentication and key agreement phase, and password change phase. For concise presentation, we list the notations and their corresponding descriptions used in this paper in **Table 1**.

Table 1. Notations used and Description

Symbol	Description
$E_p(a,b)$	An elliptic curve, n is larger prime number as curve's order
P	The base point of the elliptic curve
k_s	The secret key of server
K_s	$K_s = k_s P$, it denotes the public key of server.
ID_i	The i thclient's identity
PW_i	The i thclient's password

$\|$	The concatenate operation
$\oplus$	The bit-wise exclusive-or(XOR) operation
$\mathrm{E}_k/\mathrm{D}_k(.)$	The symmetric encryption/ decryption algorithm using key K
$\rightarrow$	A common and unsecure channel
$\Rightarrow$	A secure channel for registration phase
$\mathrm{h}(.)$	A function using one-way hash

3.1 System setup phase

The server establishes the initial parameters for SIP, namely, chooses an elliptic curve $E_p(a, b)$ with prime order n, and sets a point P as base point over $E_p(a, b)$with order n. Then takes a secure one-way hash function $\mathrm{h}(.)$ as a hash function used in scheme, selects a key k_s as its eternal private secret key, and publishes $K_s = k_s P$ as its public key, at last, publishes ($E_p(a, b)$, n, P, $h(.)$,K_s) as system parameter.

3.2 Registration phase

In this phase, the client will execute the following steps with the server if he/she registers to be a legal user.

Step R1. Client →Server: (ID_i, v_i)

The client with identity ID_ichoose a random number $N_c \in \mathrm{Z}_P^*$ and a password PW_i, then computes and stores $v_i = h(ID_i \| PW_i \| N_c)$ in his/her memory device, such as portable HDDs or USB stick, finally sendsv_i and ID_i to the server via a secure channel.

Step R2. Server:

Upon receiving the registration message of a client, the server calculates$\mathrm{V}_i = h(ID_i \| k_s) \oplus v_i$ and stores tuple about ($\mathrm{ID}_i, \mathrm{V}_i$) in its database if the ID_i is not in the database.

3.3 Authentication and key agreement phase

By the time a client wants to establish a new communication with the server, he/she does as follows.

Step A1. Client →Server: *REQ* (ID_i, R_C)

The client generates a random integer $d_c \in \mathrm{Z}_P^*$, and sends the *REQ*(ID_i, R_C) to the server via the public channel, where $R_C = d_c K_s$.

Step A2. Server→ Client: *CHA* $(realm, Q_S, V_S)$

On receiving the message *REQ* (ID_i, R_C), the server stops this session if this ID_i is not in the database.Or, the server generates a random integer $d_s \in \mathrm{Z}_P^*$, computes $\mathrm{Q}_s = d_s P, \mathrm{Q}_{sc} = d_s k_s^{-1} \mathrm{R}_c$, and $\mathrm{V}_s = h(ID_i \| \mathrm{Q}_s \| \mathrm{Q}_{sc})$, then sends a challenge message as *CHA* $(realm, Q_S, V_S)$ to the client via the public channel.

Step A3. Client →Server: *RES* $(ID_i, realm, V_C)$

Upon receiving the message *CHA* $(realm, Q_S, V_S)$, the client checks whether the received V_C is equal to $h(ID_i \| \mathrm{Q}_s \| \mathrm{Q}'_{sc})$ or not, where, $\mathrm{Q}'_{sc} = d_c \mathrm{Q}_s = d_c d_s P$. If it is not true, the client stops this session. Otherwise, the client retrieves N_c from his/her memory device, and computes $\mathrm{V}_c = h(ID_i \| \mathrm{Q}_s \| \mathrm{ram} \| \mathrm{Q}_{sc} \| h(ID_i \| PW_i \| N_c))$, then send message *RES*

$(ID_i, realm, V_C)$ to server via the public channel. The share session key *SK* is computed by $SK = h(ID_i \| Q_s \| Q_{sc} \| ram)$.

Step A4. Server:

On receiving the message *RES* $(ID_i, realm, V_C)$, the server computes $v_i = V_i \oplus h(ID_i \| k_s) = h(ID_i \| PW_i \| N_C)$, then checks whether the receivedV_C is equal to $h(ID_i \| Q_s \| ram \| Q_{sc} \| v_i)$ or not. If not hold, the server drops the session, otherwise, the server confirms the client and accepts the client's request. At last the server calculates the new shared session key according to $h(ID_i \| Q_s \| Q_{sc} \| ram)$.

3.4 Password change phase

After being authenticated and obtaining the session key, the client can change his/her password PW_i according to the following steps.

Step P1. Client →Server: *CHAPWD* (ID_i, Z, V_z)

The client selects a random integer $N_c \in Z_P^*$ and a new password PW_i^*, then retrieves the random number N_C in his/her memory device, and computes $v_i^* = h(ID_i \| PW_i^* \| N_c^*)$,$z = h(ID_i \| PW_i \| N_c) \oplus v_i^*$, $Z = z \oplus h(ID_i \| SK)$, and $V_z = h(ID_i \| v_i^* \| SK \| v_i)$, where *SK* is the current session key. Then client sends the message *CHAPWD* (ID_i, Z, V_z) to the server.

Step P2. Server→ Client: *ACC* $(h(ID_i \| v_i \| accept \| v_i^* \| SK))$

Upon receiving the message *CHAPWD* (ID_i, Z, V_z), the server computes $v_i = V_i \oplus h(ID_i \| k_s)$, $v_i^* = Z \oplus v_i \oplus h(ID_i \| SK)$, and rejects the *CHAPWD* message if $h(ID_i \| v_i^* \| SK \| v_i)$ is not equal to the received V_z. Or, the server calculates $V_i^* = V_i \oplus Z = h(ID_i \| k_s) \oplus h(ID_i \| PW_i^* \| N_c^*)$ and replaces V_i with V_i^* in server database. At last the server sends the message *ACC*$(h(ID_i \| v_i \| accept \| v_i^* \| SK))$ to the client.

Step P3. Client:

The client checks whether $h(ID_i \| v_i \| accept \| v_i^* \| SK)$ is the same with the *ACC* received from the server. If they are the same, the client replaces N_c with N_c^*.

Although Hamed et al. claimed that their scheme was secure against various types of attacks, we demonstrate their scheme is not secure against the trajectory tracking attack since each client's identity is exposed in their scheme. What is worse, their scheme is vulnerable to de-synchronization attack, because an adversary can intercept the message ACC and instead it with a false one to pass the verification of the other legal client. The details of implementing these attacks are presented in the following section.

4. Cryptanalysis of Hamed *et al.*'s scheme

Next, we will demonstrate that Hamed *et al.*'s scheme [20] cannot provide user anonymity and is apt to suffer trajectory tracking attack and de-synchronization attack in this section.

4.1 No provide user anonymity and suffering trajectory tracking attack

With frequent modern privacy violations, people are more aware of privacy protection than before. The user anonymity has becoming an urgent requirement and admired property of SIP. However, the client's identity ID_i is exposed to everyone in Hamed *et al.*'s scheme. A malicious adversary obtains all of the user's conversation data without any difficulty, and then mounts a trajectory tracking attack to disclose more private information from these conversation data. It is worth to note that user anonymity, un-traceability and non-linkability

of user trajectory are admired security requirement, especially in the current era of frequent occurrence of privacy disclosure.

4.2 De-synchronization attack

During registration phase of Hamed *et al.*'s scheme, the client calculates his/her verification information v_i by $v_i = h(ID_i \| PW_i \| N_c)$, then send v_i to the server via secure channel. Upon receiving v_i, the server stores the client's verification with $V_i = h(ID_i \| k_s) \oplus v_i$. So, the server and the client preserve the secret sharing data v_i. In the authentication and key agreement phase of Hamed *et al.*'s scheme, v_i is a critical data for the client when authenticated by the server. Thereby, it needs an additional synchronization mechanism to keep the consistency during the password change phase of Hamed *et al.*'s scheme. However, we notice that if this consistence is broken, the client cannot pass authentication and key agreement with the server again. Let us illustrate a concrete example, as shown in **Fig. 1**. Assume the server has performed **Step P2** during password change phase (it means the server has replaced V_i with $V_i^* = h(ID_i \| k_s) \oplus v_i^* = h(ID_i \| k_s) \oplus h(ID_i \| PW_i^* \| N_c^*)$) in the server database) and sends out message *ACC* to the client. Before *ACC* reaches to the client, the malicious adversary intercepts it and blocks it or sends a false message *ACC* to the client. As a result, the client cannot get a correct verification of the equation about $ACC = h(ID_i \| v_i \| \text{accept} \| v_i^* \| SK)$, and of course, the client will not update the PW_i and N_c. In this way, the consistency of the v_i (or v_i^*) between the server and the client is eroded. In the next authentication and key agreement request, this client with the old $v_i = h(ID_i \| PW_i \| N_c)$ will always be rejected by the server, for there is no corresponding v_i in the server database.

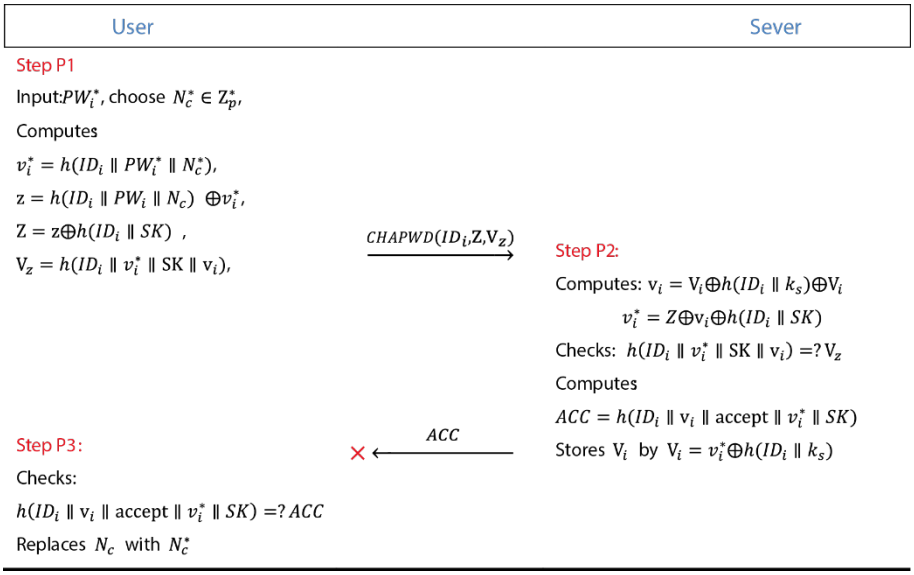

Fig. 1. De-synchronization attack on Hamed *et al.*'s scheme

5. Our improved scheme

Next, an enhanced security scheme over Hamed *et al.*'s scheme will be proposed in this section. This scheme is comprised of four phases: system setup phase, registration phase, authentication and key agreement phase, and password change phase. Similarly, the relevant

notations and their corresponding descriptions used in our scheme are shown in **Table 1**. We describe the details of each phase of scheme as follows.

5.1 System setup phase

Our scheme has the same system setup phase with Hamed *et al.*'s scheme's [20], so we do not repeat it here.

5.2. Registration phase

If a client wants to be a register member of the server, he/she and the server will execute as follows.

Step R1. Client →Server: (ID_i, HID_i)

The client chooses a random number $N_c \in \mathrm{Z}_P^*$ and a password PW_i, then computes $HID_i = h(ID_i \| N_c)$, and sends ID_i and HID_i to the server via a secure channel. At last, the client calculates $\mathrm{VR_{si}} = \mathrm{h}(\mathrm{ID_i} \| \mathrm{h}(\mathrm{N_i} \| \mathrm{PW_i}))$, $\mathrm{R_{si}} = \mathrm{h}(\mathrm{ID_i} \| \mathrm{PW_i}) \oplus \mathrm{N_i}$, and stores $\{R_{si}, \mathrm{V}R_{si}\}$ in his/her memory device.

Step R2. Server:

Upon receiving the message from the client, the server calculates $ID_{si} = h(k_s \| ID_i)$ and $V_{si} = \mathrm{h}(k_s \| \mathrm{HID}_i)$, and then stores $(\mathrm{ID}_{si}, \mathrm{V}_{si})$ in its database if its ID_{si} is not in the database.

5.3 Authentication and key agreement phase

When a client wants to establish a new communication with the server, he/she and the server perform actions as shown in **Fig. 2**. The details of corresponding actions are as follows.

Step A1. Client →Server: *REQ*(R_i, e_i, V_i, T_i)

The client generates a random integer $d_c \in \mathrm{Z}_P^*$ and current timestamps T_i, and inputs ID_i, PW_i, retrieves R_{si} from his/her memory device, then computes $N_c = R_{si} \oplus \mathrm{h}(ID_i \| PW_i)$, $\mathrm{HID}_i = \mathrm{h}(ID_i \| N_c)$, $\mathrm{R}_1 = d_c\mathrm{Ks}$, and computes $\mathrm{R}_2 = d_c\mathrm{P} = (\mathrm{R}_{2x}, \mathrm{R}_{2y})$ to obtains R_{2x}, encrypts message $e_i = \mathrm{E}_{\mathrm{R}_{2x}}(\mathrm{HID}_i, \mathrm{ID}_i)$ by using R_{2x}. Next, the client computes $\mathrm{V}_i = \mathrm{h}(\mathrm{HID}_i \| e_i \| \mathrm{T}_i)$, and send message as *REQ* (R_i, e_i, V_i, T_i) to the server via a public channel.

Compared to Hamed et al.'s scheme, client's identity ID_i is not directly transmitted through the public channel, since we encrypt it as $e_i = \mathrm{E}_{\mathrm{R}_{2x}}(\mathrm{HID}_i, \mathrm{ID}_i)$.

Step A2. Server→ Client: *CHA* (T_S, V_S)

After receiving the message *REQ* (R_i, e_i, V_i, T_i), the server checks whether T_i is fresh or not. If T_i is not fresh, the server refuses this session. Otherwise, the server calculates $\mathrm{R}'_2 = \mathrm{k}_s^{-1}\mathrm{R}_1 = (\mathrm{R}'_{2x}, \mathrm{R}'_{2y})$ to obtains R'_{2x}, then it decrypts e_i by using R'_{2x} to reveal $(\mathrm{HID}'_i, \mathrm{ID}'_i)$. Next the server computes $ID_{si} = h(k_s \| ID'_i)$, $\mathrm{S}_{vi} = \mathrm{h}(k_s \| \mathrm{HID}'_i)$, and checks whether $(\mathrm{ID}_{si}, \mathrm{S}_{vi})$ is in the server database or not. If $(\mathrm{ID}_{si}, \mathrm{S}_{vi})$ is not in database, the server refuses this session. Else, the server selects the current timestamps T_s, computes $V_s = h(HID'_i \| R'_{2x} \| T_s)$. Then it sends message *CHA*(T_S, V_S) to the client via a public channel.

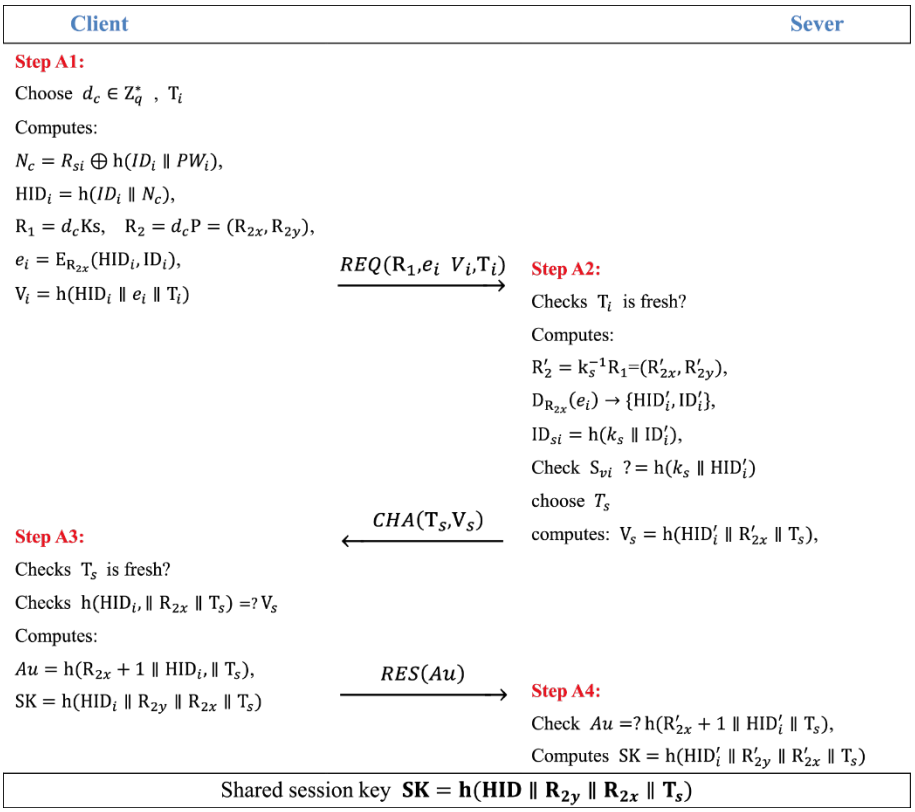

Fig. 2. Authentication and key agreement phase of the proposed scheme

Step A3. Client →Server: *RES(Au)*

Upon receiving the message *CHA* (T_S, V_S), the client checks whether T_s is fresh or not. If not, the client aborts the session. Otherwise, the client checks whether the received V_s is equal to $h(HID_i \| R_{2x} \| T_s)$ or not. If not, the client terminates this session. Or the client calculates $Au = h(R_{2x} + 1 \| HID_i \| T_s)$ and sends *RES(Au)* to the server via a public channel. Furthermore, the client calculates $SK = h(HID_i \| R_{2y} \| R_{2x} \| T_s)$ as the shared session key.

Step A4. Server:

On receiving the message *RES(Au)* from the client, the server checks whether the received *RES(Au)* is the same with $h(R'_{2x} + 1 \| HID'_i \| T_s)$ or not. If they are the same, the server computes $K = h(HID'_i \| R_{2y} \| R_{2x} \| T_s)$ as the shared session key; otherwise, the server terminates this session.

5.4 Password change phase

If a client wants to renew his/her password, he/she can finish this phase anytime without going through the authentication phase and participation of the server. This phase goes as follows:

The client first chooses a new password PW_i^*, and inputs ID_i, PW_i, then retrieves R_{si} and VR_{si} from his/her memory device, next computes $N_i = R_{si} \oplus h(ID_i \| PW_i)$ and checks whether VR_{si} and $h(ID_i \| h(N_i \| PW_i))$ are equal or not. If they are not equal, the client aborts the next steps; otherwise, the client computes $R_{si}^* = h(ID_i \| PW_i^{new}) \oplus N_i$, $VR_{si}^* = h(ID_i \| h(N_i \| PW_i^{new}))$, finally replaces R_{si},VR_{si} with R_{si}^*, VR_{si}^* respectively.

Obviously, the password change phase happens only in the client side, it does not rely on the message $h(ID_i \| v_i \| \text{accept} \| v_i^* \| SK)$ sent by the server. Therefore, the client does not need to worry that the message is lost in the public channel or modified by adversaries, since it can change the password autonomously. That is, our proposed scheme can provide the resistance of de-synchronization attack.

6. Security proof

In this section, we present the security proof of our proposed scheme. And the Burrows-Abadi-Needham (BAN) logic is introduced to demonstrate that the proposed scheme is provably secure.

Among many formal analysis methods, BAN logic is one of the most influential methods. It was proposed by Burrows, Abadi and Needham in 1989 [22]. In order to definite and analyze information exchange protocols, BAN logic designs a set of rules/principals. BAN is a model base on faith and its rules/principals will change its faith along with the message exchange. It contains no logical inversions; therefore, it cannot be used to prove a protocol flawed. **Table 2** lists some BAN logical modal operations and corresponding descriptions.

Table 2.. Logical modal operation and descriptions

Operation	Description
A| ≡ X	A believes X
A|~X	A once said X
A ⇒ X	A has jurisdiction over X
A ◁ X	A sees X
#(X)	X is fresh
(X, Y)	X and Y area part of (X, Y) respectively
SK	The shared session key
A → B	A said to B
A $\overset{K}{\leftrightarrow}$ B	A and B can communicate by using key K
$(Y)_X$	Y is hash by using key X
A $\overset{X}{\Leftrightarrow}$ B	A and B share secret X

With help of BAN logic, users can judge whether the information exchanged in the communication protocols is secured against being eavesdropped, authentic. BAN logic has three typical logical verifications, i.e., the first is to verify message origin, the second is to verify message freshness, and the last is to verify the origin's trustworthiness.

Next, our scheme is proved to be secured in BAN logic. In our scheme, the registration phase goes via a secure channel and the password changing phase just does on client's devises without communication. As the result, if the authentication and key agreement is proved to be

secure, we can confirm that our scheme is secure. For simplicity, $\mathcal{C}$ and $\mathcal{S}$denote as the client and the server respectively during the following analysis.

In the authentication and key agreement phase, there are two main goals for our proposed scheme, the goals denote that the client believes itself can establish a secure session with the server through an only shared session key between them, and vice versa, i.e.,

GoL 1: $\mathcal{C}| \equiv (\mathcal{C} \overset{SK}{\leftrightarrow} \mathcal{S})$.

GoL 2: $\mathcal{S}| \equiv (\mathcal{C} \overset{SK}{\leftrightarrow} \mathcal{S})$.

Now, we transform the authentication and key agreement phase of our scheme to an idealized form as follows:

Msg 1: $\mathcal{C} \to \mathcal{S}$: $\{\mathrm{R}_1,\ T_i,\ \{HID_i, ID_i\}_{R_2}, \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S}\}$.

Msg 2: $\mathcal{S} \to \mathcal{C}$: $\{T_s,\ \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S},\ \mathcal{C} \overset{HID_i}{\longleftrightarrow} \mathcal{S}\}$.

Msg 3: $\mathcal{C} \to \mathcal{S}$: $\{T_s,\ \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S},\ \mathcal{C} \overset{HID_i}{\longleftrightarrow} \mathcal{S}\}$.

Here, we state the assumption about the original messages as follows:

Asp 1: $\mathcal{C}| \equiv \#(R_2)$.

Asp 2: $\mathcal{S}| \equiv \#(T_s)$.

Asp 3: $\mathcal{C}| \equiv \#(\mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S})$.

Asp 4: $\mathcal{S}| \equiv \#(\mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S})$.

Asp 5: $\mathcal{C}| \equiv (\mathcal{C} \overset{HID_i}{\longleftrightarrow} \mathcal{S})$.

Asp 6: $\mathcal{S}| \equiv (\mathcal{C} \overset{HID_i}{\longleftrightarrow} \mathcal{S})$.

Asp 7: $\mathcal{S}| \equiv \mathcal{C} \Rightarrow (R_2)$.

Asp 8: $\mathcal{C}| \equiv \mathcal{S} \Rightarrow (T_s)$.

Asp 9: $\mathcal{S}| \equiv (\mathcal{C} \overset{T_s}{\leftrightarrow} \mathcal{S})$.

Asp 10: $\mathcal{C}| \equiv (\mathcal{C} \overset{T_s}{\leftrightarrow} \mathcal{S})$.

Last, according to the above-mentioned assumptions and logical postulates, we can prove the security of the authentication and key agreement phase based on BAN logic as following:

According to Msg 1, we can get:

$\mathcal{S} \lhd \{R_1, T_i,\ \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S},\ \{HID_i, \mathrm{ID}_i\}_{R_{2x}}\}$.

According to the jurisdiction rule, we can get:

$\mathcal{S} \lhd \{\{\mathrm{HID}_i, \mathrm{ID}_i\}_{R_{2x}}\}$.

According to Asp5, Asp 7 and the message rules, we can get:

$\mathcal{S}| \equiv \mathcal{C}|\sim(\mathrm{HID}_i, \mathrm{ID}_i, R_2)$.

According to Msg 2, we can get:

$\mathcal{C} \lhd \{T_s,\ \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S},\ \mathcal{C} \overset{HID_i}{\longleftrightarrow} \mathcal{S}\}$.

According to the assumption Asp2 and the message meaning rules, we can get:

$\mathcal{C}| \equiv \mathcal{S}|\sim(T_s,\ \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S},\ \mathcal{C} \overset{HID_i}{\longleftrightarrow} \mathcal{S})$.

According to the assumption Asp 1, Asp 2 and Asp 3,

$\mathcal{C}| \equiv \#(R_2, T_i, T_s)$.

According to the nonce-verification rules, we can get:

$\mathcal{C}| \equiv \mathcal{S}| \equiv (R_2, T_i, T_s)$.

According to the assumption Asp 8 and the jurisdiction rules, we can get:

$\mathcal{C}| \equiv (R_2, T_i, T_s)$.

According to the jurisdiction rules, we can get:

$\mathcal{C}| \equiv T_s$.

According to the shared session key $SK = h(HID_i' \| R_{2y}' \| R_{2x}' \| T_s)$ and the assumption Asp 1, Asp 5, we can get:

$\mathcal{C}| \equiv (\mathcal{C} \overset{SK}{\leftrightarrow} \mathcal{S})$ **(GoL1)**.

According to the message 3, we can get:

$\mathcal{S} \lhd \{T_s, \ \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S}, \ \mathcal{C} \overset{HID}{\longleftrightarrow} \mathcal{S}, \mathcal{C} \overset{SK}{\leftrightarrow} \mathcal{S}\}$.

According to Asp10 and message-meaning rules, we can get:

$\mathcal{S}| \equiv \mathcal{C}|\sim(T_s, \ \mathcal{C} \overset{R_2}{\leftrightarrow} \mathcal{S}, \ \mathcal{C} \overset{HID_i}{\longleftrightarrow} \mathcal{S})$.

According to Asp 2, Asp 4 and the freshness-conjuncatenation rules, we can get:

$\mathcal{S}| \equiv \#(R_2, T_s)$.

According to the nonce-verification rules, we can get:

$\mathcal{S}| \equiv \mathcal{C}| \equiv (R_2, T_i, T_s)$.

According to the Asp 7, we can get:

$\mathcal{S}| \equiv (R_2, T_i, T_s)$.

According to the assumption Asp 5, Asp 7 and the shared session key $SK = h(HID_i' \| R_{2y}' \| R_{2x}' \| T_s)$, we can get:

$\mathcal{S}| \equiv (\mathcal{C} \overset{SK}{\leftrightarrow} \mathcal{S})$ **(GoL2)**.

Therefore, our proposed scheme is provably secure based on BAN logic for achieving GoL1 and GoL2.

7. Security analysis and complexity comparisons

In this section, we will show that our proposed scheme has lower computation costs while it is more secure than other relevant schemes.

7.1 security analysis

In this section, detailed security analysis of our proposed scheme shows that our proposed scheme is secure against a variety of relative attacks.

7.1.1 Relay attacks

Relay attacks occur when an adversary impersonates one of the communicating parties by reusing the authentication messages obtained in a SIP [12].

Our proposed scheme adopts a fresh random integer number d_c, two fresh timestamps T_i and T_s to resist replay attacks in each session. Suppose that the client's previous *REQ*(R_i, e_i, V_i, T_i) has been intercepted by an adversary in step A1. In order to impersonate the client, the adversary relays *REQ* to the server. However, in step A3, without knowledge of R_2 and d_c, the adversary cannot construct a right *RES(Au)* to pass the server's verification. In order to obtain R_2 and d_c, the adversary needs to retrieve R_2 from R_1. Obviously, the adversary cannot successfully retrieve R_2 because he/she cannot resolve an ecliptic curve discrete logarithm problem without knowing the server secrete key k_s. As a result, our scheme can resist relay attacks effectively.

7.1.2 De-synchronization attack

In our proposed scheme, there is no need to synchronize data between the server and the clients during the authentication and key phase. Similarly, it does not need to synchronize data during the password change phase as well. So, an adversary cannot mount a de-synchronization attack. As a result, our scheme could completely resist the de-synchronization attack.

7.1.3 Provide user anonymity

The client's ID is encrypted by secure symmetric encryption with a random secrete key in our proposed scheme, i.e., $e_i = \mathrm{E}_{\mathrm{R}_{2x}}(\mathrm{HID}_i, \mathrm{ID}_i)$, and the value of R_{2x} is obtained from $\mathrm{R}_2 = d_c\mathrm{P} = (\mathrm{R}_{2x}, \mathrm{R}_{2y})$ with a random integer $d_c \in \mathrm{Z}_P^*$, so R_{2x} and e_i are different in each session.R_{2x} is protected under elliptic curve discrete logarithm problem (DLP), and only can be obtained by the one who has the server private key k_s. Assume the messages transmitted between the client and the server have been recorded by an adversary, but the adversary cannot retrieve ID_i or trace the client by $e_i$without knowing the secret key k_s. Thus, better than Hamed et al.'s scheme, our proposed scheme can provide user anonymity and preserve user privacy.

7.1.4 Provide Mutual authentication

As an essential and important requirement of SIP, the mutual authentication is one of key goals during protocol design. In our proposed scheme, the client sends $e_i = \mathrm{E}_{\mathrm{R}_{2x}}(\mathrm{HID}_i, \mathrm{ID}_i)$ to server in Step A1, where only the legal user ID_i has his/her secrete HID_i. Then in step A2, the server authenticates the client by checking whether two equations do hold or not, i.e., $ID_{si} = h(k_s \parallel ID_i')$ and $\mathrm{S}_{vi} = \mathrm{h}(k_s \parallel \mathrm{HID}_i')$, where ID_{si} and S_{vi} are retrieved from the server database. In step A3, the client authenticates the server by checking whetherV_s is equal to $h(HID_i \parallel R_{2x} \parallel T_s)$ or not, this is because no one can retrieve R_{2x} and HID_i without knowledge of the server secrete key k_s. Therefore, our proposed scheme could provide mutual authentication.

7.1.5 Privilege insider attacks

As shown in section 4.2, in the registration phase, the server stores the clients' registration information in the verification table by using $ID_{si} = h(k_s \parallel ID_i)$ and $\mathrm{S}_{vi} = \mathrm{h}(k_s \parallel HID_i)$, which makes the verification information in a chaotic state. When the privilege attacker gets the verification table, he/she cannot obtain any useful information and cannot retrieveID_i or HID_iwithout knowledge of server's secrete key k_s, and he/she can never derive the client's password because no password information is stored in verification table. Therefore, our scheme is secure against the privilege insider attack.

7.1.6 Impersonation attacks

A malicious adversary maybe intercepts all transmitted messages in our proposed scheme, such as R_1, e_i, V_i, T_i, T_S, V_S, Au. But the adversary cannot compute the values of $\mathrm{R}_{2x}, \mathrm{R}_{2y}, d_c, ID_i$ andHID_i, so that the adversary cannot construct a correct *RES(Au)* to pass the

server's verification. With no knowledge of the server's secrete key k_s, the adversary cannot compute the corrected values of $h(HID_i \parallel R_{2x} \parallel T_s)$ to pass the client's verification process. Hence, the proposed scheme can resist the impersonation attacks.

7.1.7 Password guessing attacks

In the authentication and key agreement phase of our scheme, the password of client is used in step A1. Suppose that an adversary has retrieves values, i.e. R_{si} and $\mathrm{VR_{si}}$, from the client's devise, and intercepts all transmitted values, such as $\mathrm{R_1}, e_i, V_i, T_i$, the adversary cannot retrieve$\mathrm{ID_i}$ by $e_i = \mathrm{E_{R_{2x}}(HID_i, ID_i)}$ without knowledge of the secrete server key k_s, then the adversary cannot launch off-line password guessing attack by analysis of $\mathrm{VR_{si}} = \mathrm{h(ID_i \parallel h(N_i \parallel PW_i))}$ and $\mathrm{R_{si}} = \mathrm{N_i} \oplus \mathrm{h(ID_i \parallel PW_i)}$ without knowing $\mathrm{N_i}$ and $\mathrm{ID_i}$. As a result, our proposed scheme is secure against password guessing attacks.

7.1.8 Perfect forward secrecy

The shared session key SK is computed by $h(\mathrm{HID}_i' \parallel R_{2y} \parallel R_{2x} \parallel T_s)$ in the authentication and key agreement phase. Suppose the adversary has the server secrete key $k_s$or the client's ID or password, without knowing random number d_c, he/she cannot compute the previous session keys, because it is impossible to computed_cP, i.e. (R_{2y},R_{2x}). Therefore, our proposed scheme could provide perfect forward secrecy.

7.1.9 Modification attacks

In the authentication and key agreement phase, each authentication message includes the verification V_i and V_s. Both V_i and V_s are computed by hash function according to R_2, Without knowing the point $R_2$of the elliptic curve, the adversary cannot launch modification attack in our scheme because he/she cannot calculate the correct V_i and V_s. Hence, the proposed scheme is secure against the modification attacks.

7.2 Performance comparisons

Next, we will evaluate the computing complexity of our scheme. Furthermore, our scheme is compared with some relevant schemes [20,21,25,29] in terms of security properties and computational cost.

7.2.1 Computational cost comparison

For the convenience of evaluation of computational cost, let $\mathrm{T_{HP}}$, $\mathrm{T_H}$, $\mathrm{T_G}$, and $\mathrm{T_E}$ denote the times cost of executing a hash to point function operation, a hash to number function operation, a random number generation and a symmetric encryption/decryption respectively; let $\mathrm{T_{PM}}$, $\mathrm{T_{PA}}$, $\mathrm{T_{IN}}$, and $\mathrm{T_M}$ denote the times cost of executing an elliptic curve point multiplication, an elliptic curve point addition, an 160-bite modular inversion and a 160-bit modular multiplication respectively. In order to achieve a more equitable and accurate comparison on computation cost, the experiment data of function operation reported in [1,15,16] are used in this paper, $\mathrm{T_{HP}}$, $\mathrm{T_H}$ and $\mathrm{T_G}$ are approximately 0.974, 0.0023 and 0.539 *ms* respectively, $\mathrm{T_{PA}}$, $\mathrm{T_{PM}}$, $\mathrm{T_M}$, $\mathrm{T_E}$ and $\mathrm{T_{IN}}$ are approximately 0.0288, 2.226, 0.00186, 0.0371 and 0.00556 *ms* respectively. According to [15,30], the time of an exclusive-or operation (XOR) and

concatenate operation are negligible because they are much less than a hash to number function operation.

As we all know, registration is a one-time operation for all communicating parties, it is not necessary to compare the computation cost of registration phase in detail. And the main computation costs of SIP come from the authentication and key agreement phase, so computation cost of this phase is selected as the comparisons of computation cost among our proposed scheme and several relevant schemes [20,21,25,29]. As shown in the **Fig. 2** and Section 5.4, in the authentication and key agreement phase of our proposed scheme, it requires three elliptic curve point multiplications, nine hash function operations, one random number generation, one 160-bite modular inversion and two symmetric encryption/decryption operations. Thus, the computation cost of the authentication and key agreement phase of our scheme is $3T_{PM} + 9T_H + 1T_G + 1T_{IN} + 2T_E$. And in all the computational cost of this phase, the client takes $5\mathrm{T_H} + 2\mathrm{T_{PM}} + \mathrm{T_E} + T_G$ and the server takes $4\mathrm{T_H} + \mathrm{T_{PM}} + \mathrm{T_{IN}}$. The corresponding cost of Hamed's scheme [20], Irshad's scheme [21], Zhang's scheme [25] and Tang's scheme [29] can be obtained from their own articles. The results are shown in **Table 3**. Close analysis of data in **Fig. 3** and **Table 3** indicate that our scheme has lower computation cost than other relevant schemes.

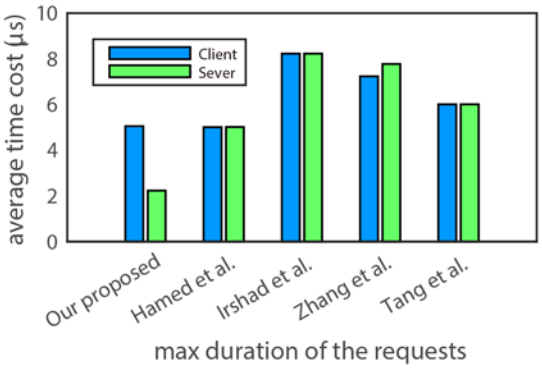

Fig. 3. Computational cost between client and server

Table 3. Performance comparison among relevant schemes

scheme	computational cost of authentication and key agreement phase		
	client	server	Total Time (ms)
Our proposed	$5\mathrm{T_H} + 2\mathrm{T_{PM}} + \mathrm{T_E} + T_G$	$4\mathrm{T_H} + \mathrm{T_{PM}} + \mathrm{T_{IN}}$	7.2804
Hamed *et al.*[20] (2014)	$4\mathrm{T_H} + 2\mathrm{T_{PM}} + \mathrm{T_G}$	$4\mathrm{T_H} + 2\mathrm{T_{PM}} + \mathrm{T_G} + \mathrm{T_M} + T_{IN}$	10.0078
Irshad *et al.*[21] (2014)	$2\mathrm{T_H} + 3\mathrm{T_{PM}} + \mathrm{T_{HP}} + \mathrm{T_{PA}} + \mathrm{T_G}$	$3\mathrm{T_H} + 3\mathrm{T_{PM}} + \mathrm{T_{HP}} + \mathrm{T_{PA}} + \mathrm{T_G}$	16.4511
Zhang *et al.*[25] (2014)	$6\mathrm{T_H} + 3\mathrm{T_{PM}} + \mathrm{T_G} + T_M$	$4\mathrm{T_H} + 3\mathrm{T_{PM}} + 2\mathrm{T_G} + 2\mathrm{T_M}$	15.0016
Tang *et al.*[29] (2013)	$3\mathrm{T_H} + 2\mathrm{T_{PM}} + \mathrm{T_{HP}} + \mathrm{T_{PA}} + T_G$	$4\mathrm{T_H} + 2\mathrm{T_{PM}} + \mathrm{T_{HP}} + \mathrm{T_{PA}} + \mathrm{T_G}$	12.0037

7.2.2 Security Comparison of relevant works

The security analysis is shown in the **Table 4**. P1…P6 mean different security properties which are shown in the footnote of **Table 4**. The symbol '✓' means the corresponding scheme is secure or can provide the security property. N/A and symbol ✖ means not applicable or

cannot provide the security. As shown in **Table 4**, comparisons of security and functionality between our scheme and [20], [21], [25] and [29], can illustrate that our proposed scheme can achieve more distinctive features, such as user anonymity, resisting de-synchronization attack and, secure and free updating password phase (i.e. updating password without connecting the server). And the Hamed et al.'s scheme cannot provide user anonymity and secure or free updating password phase. It also cannot resist to the de-synchronization attack. To the best of our knowledge, these new security requirements and functionality requirements have become extremely urgent while people are more sensitive to privacy protection.

Table 4. Security comparison among relevant schemes

scheme	Security and functionality					
	P1	P2	P3	P4	P5	P6
Our proposed	✓	✓	✓	✓	✓	✓
Hamed *et al.*[20] (2014)	×	×	✓	✓	✓	×
Irshad *et al.*[21] (2014)	×	✓	✓	✓	✓	N/A
Zhang *et al.*[25] (2014)	×	✓	✓	✓	×	N/A
Tang *et al.*[29] (2013)	×	×	✓	✓	×	×

P1: Provide user anonymity, **P2:** Resist de-synchronization attack, **P3:** Resist relay attack, **P4:** Resist impersonation attack, **P5:** Resist privilege insider attacks, **P6:** Provide secure and free updating password phase.

From above discussion, it can draw a conclusion that our proposed scheme not only provides better security prosperities and privacy preserving, but also is more efficient than the relevant schemes. Especially, comparing with Hamed et al.'s scheme our proposed scheme has 27.25% lower computation costs while has better security properties.

8. Conclusions

In this paper, we have demonstrated that Hamed *et al.*'s authentication scheme for session initial protocol cannot provide user anonymity and it is also apt to suffer from the trajectory tracking attack and de-synchronization attack. In order to make up for these deficiencies, we propose an enhanced security scheme for session initial protocol over Hamed *et al.*'s authentication scheme. The security analysis proves that our proposed scheme could resist all known security attacks. The comparison with the relevant schemes shows that our proposed scheme has 27.25% lower computation costs while has better security properties than Hamed et al.'s scheme. Further, in order to improve the efficiency of message authentication, next, we will research on cooperating authentication scheme among users.

References

[1] Kilinc H H, Yanik T., "A Survey of SIP Authentication and Key Agreement Schemes [J]," *Communications Surveys & Tutorials, IEEE*, 16(2): 1005-1023, 2014. Article (CrossRef Link)

[2] Franks J, Hallam-Baker P, Hostetler J, et al., "HTTP authentication: Basic and digest access authentication [Z]," *RFC 2617*, June, 1999. Article (CrossRef Link)

[3] Li J, Kao C, Tzeng J., "VoIP secure session assistance and call monitoring via building security gateway [J]," *International Journal of Communication Systems*, 24(7): 837-851, 2011. Article (CrossRef Link)

[4] Farash MS, Attari MA, "An enhanced authenticated key agreement for session initiation protocol," *Inform Techno Control*, 42(4):333-342, 2013. Article (CrossRef Link)
[5] Jiang Q, Ma J, Tian Y., "Cryptanalysis of smart-card-based password authenticated key agreement protocol for session initiation protocol of Zhang et al [J]," *International Journal of Communication Systems*, 28(7): 1340-1351, 2015. Article (CrossRef Link)
[6] Chen CL, Lee CC, Hsu CY., "Mobile device integration of a fingerprint biometric remote authentication scheme [J]," *International Journal of Communication Systems*, 25(6):585-597, 2012. Article (CrossRef Link)
[7] Jiang Q, Ma J, Li G, et al., "An Enhanced Authentication Scheme with Privacy Preservation for Roaming Service in Global Mobility Networks [J]," *Wireless Personal Communications*, 68(4): 1477-1491, 2013. Article (CrossRef Link)
[8] Yang CC, Wang RC, Liu WT., "Secure authentication scheme for session initiation protocol [J]," *Computers & Security*, 24(5):381–386, 2005. Article (CrossRef Link)
[9] Durlanik A, Sogukpinar I., "SIP authentication scheme using ECDH [C]," in *Proc. of World Academy of Science, Engineering and Technology*, 8:350-353, 2005. Article (CrossRef Link)
[10] Yoon EJ, Yoo KY., "Cryptanalysis of DS-SIP authentication scheme using ECDH [C]," in *Proc. of International Conference on New Trends in Information and Service Science*, Beijing, 642-647, 2009. Article (CrossRef Link)
[11] Wu L, Zhang Y, Wang F., "A new provably secure authentication and key agreement protocol for SIP using ECC [J]," *Comput Stand Interfaces*, 31(2):286–291, 2009. Article (CrossRef Link)
[12] Yoon EJ, Yoo KY, Kim C, Hong Y, Jo M, Chen H., "A secure and efficient SIP authentication scheme for converged VoIP networks [J]," *Comput Commun*, 33(14):1674-1681, 2010. Article (CrossRef Link)
[13] Jaeduck C, Souhwan J, Kwangyong B, et al., "A lightweight authentication and hop-by-hop security mechanism for SIP network[C]," in *Proc. of Advanced Technologies for Communications, International Conference on*, Hanoi, (1): 235-238, 2008. Article (CrossRef Link)
[14] Xie Q., "A new authenticated key agreement for session initiation protocol [J]," *Int J Commun Syst.*, 25(1):47-54, 2012. Article (CrossRef Link)
[15] Chaudhry SA, Farash MS, Naqvi H, Sher M., "A secure and efficient authenticated encryption for electronic payment systems using elliptic curve cryptography," *Electronic Commerce Research*, 16(1):113-139, 2016. Article (CrossRef Link)
[16] Acquist A, Brandimarte L, Loewenstein G., "Privacy and human behavior in the age of information [J]," *Science*, 347(6221):509-514, 2015. Article (CrossRef Link)
[17] Hughes D, shmatukov V., "Information hiding, anonymity and privacy: a modular approach [J]," *Journal of Computer Security*, 12(1):3-36, 2014. Article (CrossRef Link)
[18] He D, Chen C, Bu J, et al., "Security and efficiency in roaming services for wireless networks: challenges, approaches, and prospects [J]," *Communications Magazine, IEEE*, 51(2): 142-150, 2013. Article (CrossRef Link)
[19] Wang RC, Juang WS, Lei CL., "Robust authentication and key agreement scheme preserving the privacy of secret key [J]," *Computer Communications*, 34(3): 274-280, 2011. Article (CrossRef Link)
[20] Arshad H, Nikooghadam M., "An efficient and secure authentication and key agreement scheme for session initiation protocol using ECC [J]," *Multimedia Tools and Applications*, 1-17, 2014. Article (CrossRef Link)
[21]Irshad A, Sher M, Faisal M S, et al., "A secure authentication scheme for session initiation protocol by using ECC on the basis of the Tang and Liu scheme [J]," *Security and Communication Networks*, 7(8): 1210-1218, 2014. Article (CrossRef Link)
[22] Burrows M, Abadi M, "Needham RM., "A logic of authentication[C]," in *Proc. of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, The Royal Society, 426(1871): 233-271, 1990. Article (CrossRef Link)
[23] Khan M K, He D., "A new dynamic identity-based authentication protocol for multi-server environment using elliptic curve cryptography [J]," *Security and Communication Networks*, 5(11): 1260-1266, 2012. Article (CrossRef Link)

[24] Zhang Y, Chen J, Huang B, et al., "An Efficient Password Authentication Scheme Using Smart Card Based on Elliptic Curve Cryptography [J]," *Information Technology and Control*, 43(4): 390-401, 2014. Article (CrossRef Link)
[25] Zhang L, Tang S, Cai Z., "Cryptanalysis and improvement of password authenticated key agreement for session initiation protocol using smart cards [J]," *Security and Communication Networks*, 7(12): 2405-2411, 2014. Article (CrossRef Link)
[26] Zhang L, Tang S, Cai Z., "Efficient and flexible password authenticated key agreement for voice over internet protocol session initiation protocol using smart card [J]," *International Journal of communication systems*, 27(11): 2691-2702, 2014. Article (CrossRef Link)
[27] Li, X., Niu, J., Liao, J. and Liang, W., "Cryptanalysis of a dynamic identity-based remote user authentication scheme with verifiable password update," *International Journal of Communication Systems*, 28(2), pp.374-382, 2015. Article (CrossRef Link)
[28] Zhang L, Tang S, Chen J, et al., "Two-Factor Remote Authentication Protocol with User Anonymity Based on Elliptic Curve Cryptography[J]," *Wireless Personal Communications*, 1-23, 2014. Article (CrossRef Link)
[29] Tang H, Liu X., "Cryptanalysis of Arshad et al.'s ECC-based mutual authentication scheme for session initiation protocol [J]," *Multimedia tools and applications*, 65(3): 321-333, 2013. Article (CrossRef Link)
[30] Koblitz N, Menezes A, Vanstone S., "The state of elliptic curve cryptography [M]," *Towards a Quarter-Century of Public Key Cryptography*, Springer US, 103-123, 2000. Article (CrossRef Link)
[31] Li, X., Niu, J., Khan, M.K., Liao, J. and Zhao, X., "Robust three-factor remote user authentication scheme with key agreement for multimedia systems," *Security and Communication Networks*, 13(9), pp.1916-1927, 2016. Article (CrossRef Link)
[32] Li X, Niu J, Kumari S, Liao J, Liang W., "An enhancement of a smart card authentication scheme for multi-server architecture," *Wireless Personal Communications*, Jan 1;80(1):175-192, 2015. Article (CrossRef Link)

.

Wu Libing received the B.Sc. and M.Sc. degrees from Central China Normal University, Wuhan, China, in 1994 and 2001, respectively, and the Ph.D. degree from Wuhan University, Wuhan, in 2006, all in computer science. He was a Visiting Scholar with the Advanced Networking Laboratory, University of Kentucky, USA, in 2011. He is currently a Professor with the Department of Computer Science, Wuhan University. His research interests include wireless sensor networks, network management, and distributed computing.

Fan Jing received M.Sc. degree in computer sciences from Central China Normal University, Wuhan, China, in 2013. He is currently pursuing the Ph.D. degree in computer sciences from Wuhan University, Wuhan, China. His main research interests include next generation Internet, Internet of vehicles and distributed computing.

Yong Xie received M.Sc. degree in computer sciences of Jingdezhen Ceramic Institute, Jingdezhen, China, in 2005 and received Ph.D. degree in computer science from Wuhan University, Wuhan, China, in 2016. His current research interests include next generation Internet, network protocol and protocol security.

Jing Wang received her B.Sc. degree from Wuhan University of China in 2016. She is currently pursuing the M.Sc. degree in computer science from Wuhan University, Wuhan, China. Her research interests are in the areas of digital signatures and secure cloud storage.