A Secure Three-Factor Authenticated Key Agreement Scheme for Multi-Server Environment

Meichen Xia[1, *], Shiliang Li[1] and Liu Liu[2]

Abstract: Multi-server authenticated key agreement schemes have attracted great attention to both academia and industry in recent years. However, traditional authenticated key agreement schemes in the single-server environment are not suitable for the multi-server environment because the user has to register on each server when he/she wishes to log in various servers for different service. Moreover, it is unreasonable to consider all servers are trusted since the server in a multi-server environment may be a semi-trusted party. In order to overcome these difficulties, we designed a secure three-factor multi-server authenticated key agreement protocol based on elliptic curve cryptography, which needs the user to register only once at the registration center in order to access all semi-trusted servers. The proposed scheme can not only against various known attacks but also provides high computational efficiency. Besides, we have proved our scheme fulfills mutual authentication by using the authentication test method.

Keywords: Authenticated key agreement, three-factor, multi-server, authentication test method.

1 Introduction

Authenticated key agreement is an important cryptography mechanism through which two communication parties could authenticate each other and establish a confidential communication channel between them. In conventional single-server authentication schemes, when the user wishes to log in in various servers for different services, he/she must register identity and password at these servers. Therefore, in order to remember various user identities and passwords, the user often has to write down this information, which will lead to extremely tedious work and easy to leak out. With the rapid development of computer networks, multi-server authentication schemes become a wide range of applications. The remote user authentication schemes for multi-server environment only need user to register once at the registration center, then the user can access all the registered servers. Multi-server authentication schemes provide convenience to users, but it is also accompanied with security problems. Recently, a large number of authentication schemes have been proposed by researchers. Among them, three-factor schemes have

[1] School of Computer and Software Engineering, Xihua University, Chengdu, 610039, China.

[2] School of Software and Electrical Engineering, Swinburne University of Technology, Hawthorn, VIC 3122, Australia.

[*] Corresponding Author: Meichen Xia. Email: xiameichen123@gmail.com.

Received: 15 February 2020; Accepted: 23 April 2020.

CMC. doi:10.32604/cmc.2020.010177 www.techscience.com/journal/cmc

gained significant attention due to their inherently reliable attributions [Mishra (2016); Yu, Wang, Gao et al. (2014)]. There are three factors generally admitted by human authentication, namely [Pointcheval and Zimmer (2008)], (1) something you know (as a secret password); (2) something you have (as an unclonable secure device with a secret key such as smart card); (3) something you are (like a biometric, for example, fingerprint, flaw print, iris). Combining the three factors in the protocol may increase the security of system since the adversary would have to break all of them in order to succeed.

In the last few years, many studies on the authentication protocol in single environment have been studied [Pointcheval and Zimmer (2008); Jiang, Khan, Lu et al. (2016); Wu, Xu, Kumari et al. (2015); Zhang, Zhang and Zhang (2015)]. However, these schemes cannot be efficiently applied for multi-server environment. Li et al. [Li, Lin and Hwang (2001)] firstly proposed multi-server authentication scheme based on password. Their scheme spends too much time on training and constructing neural networks. In addition, maintaining neural networks in every single server will also add up extra computation cost. Later, Lin et al. [Lin, Hwang and Li (2003); Tsaur, Wu and Lee (2004)] improved this scheme, but the efficiency is still low, and it is vulnerable to insider attack. Tsai [Tsai (2008)] proposed another scheme based on the nonce and hash function which does not need to store any verification table in the server and registration center. This scheme is also very efficient as compared with the above protocols because it does not use any symmetric and asymmetric encryption algorithm. However, it is also vulnerable to insider attack. In order to solve this problem, Tsaur et al. [Tsaur, Li and Lee (2012)] proposed another authenticated key agreement scheme based on self-verified timestamp technique. Nevertheless, all the schemes given above do not consider privacy for multi-server environment because identities used in the schemes are static. As a result, the leakage identities of users may reveal their movements and locations so as to influence their normal life [Gu, Yang and Yin (2018); He, Zeng, Xie et al. (2017); Yin, Ju, Yin et al. (2019)]. In order to solve this problem, Liao et al. [Liao and Wang (2007)] proposed a dynamic identity authentication scheme for the multi-server environment. This scheme is intended to resist various attacks such as reply attack, masquerade attack, anonymity and so on. Unfortunately, it is still vulnerable to insider attack, masquerade attack, server spoofing attack and registration center spoofing attack. Later, Hsiang et al. [Hsiang and Shih (2009)] proposed an improved scheme which has overcome the failing found in Liao and Wang's scheme, but it is vulnerable to server spoofing attack. Subsequently, Sood et al. [Sood, Sarje and Singh (2011); Lee, Lin and Chang (2011)] proposed enhanced schemes but in which some weakness have been discovered. To enhance security and provide perfect forward secrecy, Yoon et al. [Yoon and Yoo (2013)] firstly proposed three-factor authenticated key agreement protocol for multi-serve using ECC. Later on, Shen et al. [Shen, Gao, He et al. (2015)] demonstrated the vulnerability of Yoon et al.'s [Yoon and Yoo (2013)] scheme to insider attack, user impersonation attack and stolen smart card attack. To overcome these weaknesses, Shen et al. proposed a modified scheme. However, this scheme was found vulnerable to spoofing attack, wrong password detection mechanism in Li et al.'s recent study [Li, Wang, Shen et al. (2016)]. Then, Li et al. [Li, Wang, Shen et al. (2016)] proposed an improved scheme. Unfortunately, Li's scheme cannot resist to impersonation attack in the registration phase and server spoofing attack. Meanwhile, it does not consider user revocation. Most recently, Odelu et al. [Odelu, Das and Goswami (2015)] designed a secure three-factor multi-server authentication key agreement protocol, which can resist various attacks. However, their scheme is vulnerable to three-factor security. To address this issue, this paper proposed a secure three-factor authentication protocol for multi-server environment. The main contributions of this paper are listed below:

(1) Our proposed scheme can resist impersonation attack in registration phase, server spoofing attack, etc.

(2) Our proposed scheme can provide three-factor security, whereas Odelu's scheme has a weakness that three factors may leaked to adversary.

(3) The proposed scheme can provide security in strand space model, and is more secure but lower computational cost.

The rest of the article is arranged as follows: Section 2 is preliminary which gives theoretical basis. Section 3 gives our proposed scheme. Subsequently, we give security analysis of our scheme and discuss functionality and performance comparisons in Section 4. Finally, we draw the conclusion in Section 5.

2 Preliminary

2.1 Notation

The notations used in our proposed scheme are listed in Tab. 1.

Table 1: Notations using throughout the paper

Notation	Descriptions
U_i	User *i*
ID_i	Unique identity of user *i*
PW_i	The password of user *i*
T_i	Timestamp generated by entity *i*
B_i	The biometric template of user *i*
S_j	*j*th server in the system
SID_j	Unique identity of *j*th server
RC	A trustworthy registration center
x, u_i	*RC*'s secret key for user registration
X	*RC*'s public key
y, su_j	*RC*'s secret key for server registration
$Ep(a, b)$	An elliptic curve
G	Additive group of points of *Ep (a, b)*
P	A generator of G
h	One-way hash functions
(Gen, Rep)	A fuzzy tractor for biometrics template

2.2 Security properties of authentication schemes

Assume probabilistic polynomial-time (PPT) adversary A has controlled communication channel. Hence, A can intercept, insert, delete, or modify all transmitted message through the channel between users and server. Therefore, security properties discussed in published paper summarize as follows [Odelu, Das and Goswami (2015)].

-Impersonation attack is an attack in which an adversary successfully assumes the identity of one of the legitimate parties in the system. It is clear that resistant to impersonation attack is basic security request in multi-server authentication schemes.

-Spoofing attack is a situation in which an adversary successfully masquerades as another by falsifying data, thereby gaining an illegitimate advantage. In multi-server environment, user spoofing or server spoofing is not permitted.

-Assuming that an adversary has been obtained all the authentication factors of the user, he can impersonate the user to RC and server. But even in this case it is desirable to prevent the adversary from impersonating RC and server to the user. Therefore, resistant to three-factor compromise impersonation attack is necessary in the multi-server environment.

-A user who forgets password or an adversary inputs incorrect password in the login phase, although, he cannot login server successfully, it is a serious security pitfall which users mistake may cause the denial-of-service attack. Thus, the mistake in login phase should not outcome denial-of-service attack.

-A user who forgets password or an adversary inputs incorrect password in password update phase, if there is no password detection mechanism, the wrong password and its verification information will be used in later log in phase and cause denial-of-service attack. Moreover, once onetime mistake in password update phase, a valid user no longer login to the server using the same smart card. Therefore, multi-server authentication scheme should consider quickly detection mechanism so that avoiding denial-of-service attack in password update phase.

-Mutual authentication is a very important security feature for user authentication scheme, which allows any participant to authenticate the other participants.

-Perfect forward secrecy means that previously established session agreement key remains secure when the long-term key of the user, server and RC are disclosed. It is also a very important feature in multi-server authentication scheme.

2.3 Review of authentication test method

Authentication test method verifies protocol security property using challenge response which is based on the strand space theory [Guttman and Fbrega (2000, 2002); Perrig and Song (2000)]. Compared with ideal and honest method, and the minimal component method in the strand space model, authentication test method can be simpler and more soundness. In this section, we give brief introduction of three theorems of authentication test method which are $n_0 \Rightarrow^+ n_1$used to prove mutual authentication property.

i. ***Authentication Test 1*** Outgoing Test Proposition: Let C be a bundle, $n_0, n_1 \in \mathrm{C}$, the edge $n_0 \Rightarrow^+ n_1$ is outgoing test for a in $t = h_k$ Then (1) there exist regular nodes $m, m' \in C$ such that t is component of node m, and the edge $m \Rightarrow^+ m'$ is a transforming edge for a.(2) Suppose in addition that a occurs only in $t' = h'_k$ of node m', that t' is not a proper subterm of any regular component, and that $K^{-1} \notin K_p$. Then there is a negative node m'' with t' as a component.

ii. ***Authentication Test 2*** Incoming Test Proposition: Let C be a bundle, $n_0, n_1 \in C$, the edge $n_0 \Rightarrow^+ n_1$ is incoming test for a in $t = h_k$. Then there exist $m, m' \in C$ such that t is component of node m', and the edge $m \Rightarrow^+ m'$ is a transforming edge for a.

iii. ***Authentication Test 3*** Unsolicited Test Proposition: Let C be a bundle, $n \in C$ and n is unsolicited test for a in $t = h_K$ then there exists regular positive node $m \in C$ such that t is component of node m.

3 The proposed scheme

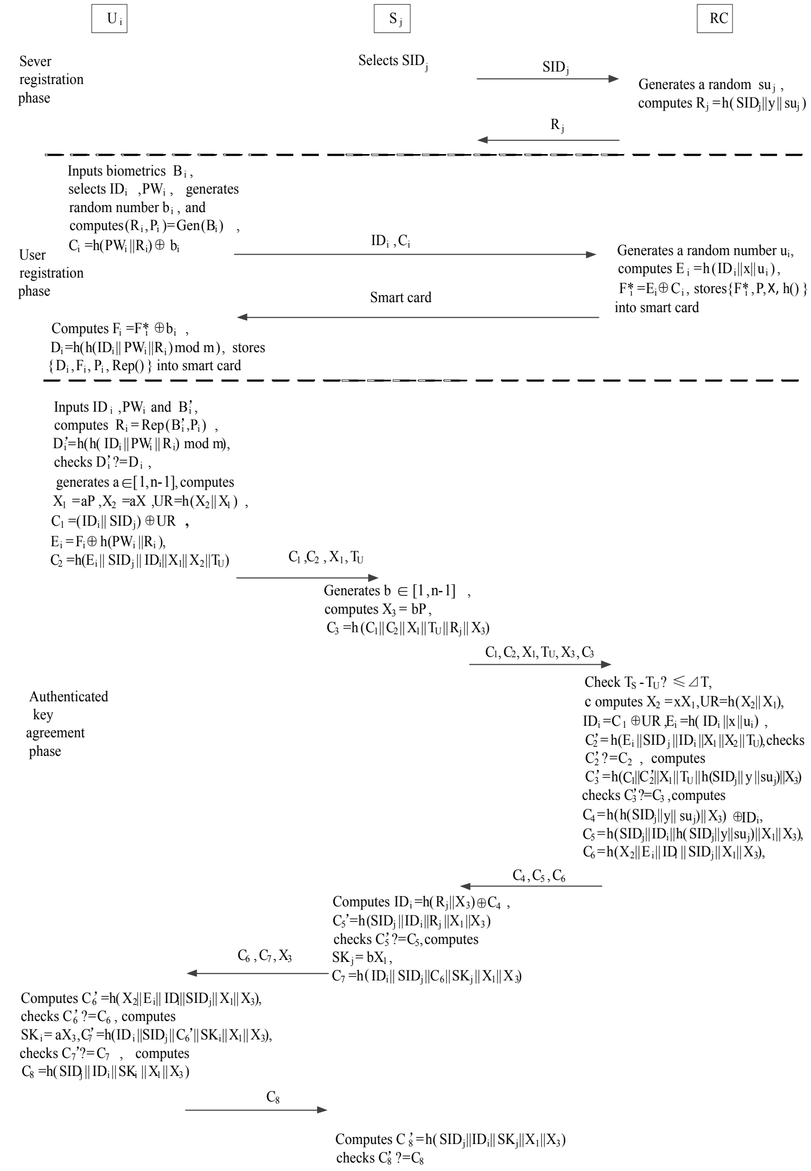
U_i
S_j
RC
Sever registration phase
Selects SID_j
SID_j
Generates a random su_j, computes $R_j = h(SID_j \| y \| su_j)$
R_j
Inputs biometrics B_i, selects ID_i, PW_i, generates random number b_i, and computes $(R_i, P_i) = Gen(B_i)$, $C_i = h(PW_i \| R_i) \oplus b_i$
User registration phase
ID_i, C_i
Generates a random number u_i, computes $E_i = h(ID_i \| x \| u_i)$, $F_i^* = E_i \oplus C_i$, stores $\{F_i^*, P, X, h()\}$ into smart card
Smart card
Computes $F_i = F_i^* \oplus b_i$, $D_i = h(h(ID_i \| PW_i \| R_i) \bmod m)$, stores $\{D_i, F_i, P_i, Rep()\}$ into smart card
Inputs ID_i, PW_i and B_i', computes $R_i = Rep(B_i', P_i)$, $D_i' = h(h(ID_i \| PW_i \| R_i) \bmod m)$, checks $D_i' ?= D_i$, generates $a \in [1, n-1]$, computes $X_1 = aP$, $X_2 = aX$, $UR = h(X_2 \| X_1)$, $C_1 = (ID_i \| SID_j) \oplus UR$, $E_i = F_i \oplus h(PW_i \| R_i)$, $C_2 = h(E_i \| SID_j \| ID_i \| X_1 \| X_2 \| T_U)$
C_1, C_2, X_1, T_U
Generates $b \in [1, n-1]$, computes $X_3 = bP$, $C_3 = h(C_1 \| C_2 \| X_1 \| T_U \| R_j \| X_3)$
$C_1, C_2, X_1, T_U, X_3, C_3$
Check $T_S - T_U ? \leqslant \Delta T$, c omputes $X_2 = xX_1$, $UR = h(X_2 \| X_1)$, $ID_i = C_1 \oplus UR$, $E_i = h(ID_i \| x \| u_i)$, $C_2' = h(E_i \| SID_j \| ID_i \| X_1 \| X_2 \| T_U)$, checks $C_2' ?= C_2$, computes $C_3' = h(C_1 \| C_2' \| X_1 \| T_U \| h(SID_j \| y \| su_j) \| X_3)$ checks $C_3' ?= C_3$, computes $C_4 = h(h(SID_j \| y \| su_j) \| X_3) \oplus ID_i$, $C_5 = h(SID_j \| ID_i \| h(SID_j \| y \| su_j) \| X_1 \| X_3)$, $C_6 = h(X_2 \| E_i \| ID_i \| SID_j \| X_1 \| X_3)$,
Authenticated key agreement phase
C_4, C_5, C_6
Computes $ID_i = h(R_j \| X_3) \oplus C_4$, $C_5' = h(SID_j \| ID_i \| R_j \| X_1 \| X_3)$ checks $C_5' ?= C_5$, computes $SK_j = bX_1$, $C_7 = h(ID_i \| SID_j \| C_6 \| SK_j \| X_1 \| X_3)$
C_6, C_7, X_3
Computes $C_6' = h(X_2 \| E_i \| ID_i \| SID_j \| X_1 \| X_3)$, checks $C_6' ?= C_6$, computes $SK_i = aX_3$, $C_7' = h(ID_i \| SID_j \| C_6' \| SK_i \| X_1 \| X_3)$, checks $C_7' ?= C_7$, computes $C_8 = h(SID_j \| ID_i \| SK_i \| X_1 \| X_3)$
C_8
Computes $C_8' = h(SID_j \| ID_i \| SK_j \| X_1 \| X_3)$ checks $C_8' ?= C_8$

Figure 1: Registration and authenticated key agreement phase of our scheme

This section will introduce our proposed scheme, which includes server registration phase, user registration phase, authenticated key agreement phase and password update phase. As shown in Fig. 1, the detail of our authenticated key agreement scheme is as follows.

3.1 Server registration phase

A legal server S_j needs to request register to RC, the process of registration is given bellow:

(1) S_j chooses his identity SID_j and transmits it to RC through secure channel.

(2) Upon receiving the registration request, RC searches whether SID_j exists in the server identity information table, if it exists, RC rejects this request. Otherwise, RC generates a random number su_j, and computers

$$R_j = h\left(SID_j||y||su_j\right). \quad (1)$$

Then RC updates the server identity information table with new entry SID_j, su_j, and transmits secret parameter R_j to S_j.

(3) After receiving the R_j from RC, S_j stores it secretly.

3.2 User registration phase

When a user U_i wants to access services provided by the system, he/she needs to register in RC first. The process of registration is show in Fig. 1.

(1) U_i imprints his/her biometric template B_i through sensor, and selects identity ID_i and password PW_i. Then, U_i generates a random number b_i and computes

$$(R_i, P_i) = Gen(B_i) \quad (2)$$

$$C_i = h(PW_i||R_i)^{\oplus} b_i \quad (3)$$

U_i transmits ID_i, C_i to RC through a secure channel.

(2) After receiving the registration request, RC checks whether ID_i exists in the user information table, if it exists, RC reject this request. Otherwise, RC generates a random number u_i, and computes

$$E_i = h(ID_i||x||u_i) \quad (4)$$

$$F_i^* = E_i {}^{\oplus} C_i \quad (5)$$

After that, RC updates the user identity information table with new entry $\{ID_i, U_i\}$ writes $\{F_i^*, P, X, h(\cdot)\}$ into the smart card and transmits it to U_i via a secure channel.

(3) U_i computes

$$F_i = F_i^* {}^{\oplus} b_i \quad (6)$$

$$D_i = h(h(ID_i||PW_i||R_i) mod\ m) \quad (7)$$

where m is medium number, $2^8 \leq m \leq 2^{16}$, which determines the capacity of the pool of the $\langle ID_i, PW_i \rangle$ pair against off-line password guessing attack [Jiang, Khan, Lu et al. (2016)], Then U_i stores $\{ID_i, F_i, P_i, Rep(,)\}$ into smart card, where F_i is to replace F_i^*, thus the smart card contains $\{D_i, F_i, P_i, P, X, h(\cdot)\}$.

3.3 Authenticated key agreement phase

If a user U_i wants to login S_j, the following steps are executed among U_i, S_j and RC.

(1) U_i inputs ID_i, PW_i and B_i into smart card. The smart card computes

$$R_i' = Rep(B_i', P_i) \quad (8)$$

$$D_i' = h(h(ID_i||PW_i||R_i') \, mod \, m) \quad (9)$$

and checks whether D_i' equals to D_i. If not, smart card fails to authenticate user. The login request is rejected by smart card. Otherwise, the smart card randomly generates a integer $a\epsilon[1, n-1]$, and computes

$$X_1 = aP \quad (10)$$

$$X_2 = aX \quad (11)$$

$$E_i = F_i \oplus h(PW_i||R_i') \quad (12)$$

$$UR = h(X_2||X_1) \quad (13)$$

$$C_1 = (ID_i||SID_j) \oplus UR \quad (14)$$

$$C_2 = h(E_i||SID_j||ID_i||X_1||X_2||T_U) \quad (15)$$

where T_U is the timestamp of current system. Then U_i transmits message $M_1 = \{C_1, C_2, X_1, T_U\}$ to S_j.

(2) S_j generates a random number $b\epsilon[1, n-1]$, and computes

$$X_3 = bP \quad (16)$$

$$C_3 = h(C_1||C_2||X_1||T_U||R_j||X_3)) \quad (17)$$

Then S_j transmits message $M_2 = \{C_1, C_2, X_1, T_U, X_3, C_3\}$ to RC.

(3) After receiving message, RC verifies whether $T_s - T_u < \Delta T$ hold, where T_s is the message receiving time, and ΔT is time threshold. If it is not hold, it means that the session is invalid and RC rejects this session. Otherwise, RC computes

$$X_2 = xX_1 \quad (18)$$

$$UR = h(X_2||X_1) \quad (19)$$

$$ID_i||SID_j = C_1 \oplus UR \quad (20)$$

$$E_i = h(ID_i||x||u_i) \quad (21)$$

$$C_2' = h(E_i||SID_j||ID_i||X_1||X_2||T_U) \quad (22)$$

and checks whether C_2' equals to C_2, if they are not equal, RC fails to authenticate user U_i and the session is terminated. Otherwise, RC verifies U_i successful. After that, RC computes

$$C_3' = h(C_1||C_2'||X_1||T_U||h(SID_j||y||su_j)||X_3) \quad (23)$$

and checks whether C_3' equals to C_3, if they are not equal, RC fails to authenticate server S_j, and the session is terminated. Otherwise, RC verifies S_j successful. Next, RC computes

$$C_4 = h(h(SID_j||y||su_j)||X_3) \oplus ID_i \quad (24)$$

$$C_5 = h(SID_j||ID_i||h(SID_j||y||su_j)||X_1||X_3) \tag{25}$$

$$C_6 = h(X_2||E_i||ID_i||SID_j||X_1||X_3) \tag{26}$$

and transmits $M_3 = \{C_4, C_5, C_6\}$ to S_j.

(4) After S_j receives message from RC, S_j computes

$$ID_i = h(R_j\ ||X_3) \oplus C_4 \tag{27}$$

$$C_5' = h(SID_j||ID_i||R_j||X_1||X_3) \tag{28}$$

and checks whether C_5' equals to C_5. If not, S_j fails to authenticate server RC, and the session is terminated. Otherwise, S_j verifies RC, successful. Then S_j computes the session key

$$SK_j = bX_1 \tag{29}$$

$$C_7 = h(ID_i||SID_j||C_6||SK_j||X_1||X_3) \tag{30}$$

and transmits $M_4 = \{C_6, C_7, X_3\}$.

(5) When U_i receives message from S_j computes

$$C_6' = h\left(X_2\middle||E_i||ID_i\middle||SID_j\middle||X_1\middle||X_3\right) \tag{31}$$

and checks whether C_6' equals to C_6, if they are not equal, U_i fails to authenticate RC, and the session is terminated. Otherwise, U_i verifies RC successful. Next, U_i computes the session key

$$SK_i = aX_3 \tag{32}$$

$$C_7' = h(ID_i||SID_j||\ C_6'||SK_i||X_1||X_3) \tag{33}$$

and checks whether C_7' equals to C_7. If not, U_i fails to authenticate S_j, and the session is terminated. Otherwise, U_i verifies S_j successful. Then U_i computes

$$C_8 = h(SID_j||ID_i||SK_i||X_1||X_3) \tag{34}$$

and transmits it to S_j.

(6) S_j computes

$$C_8' = h(SID_j||ID_i||SK_j||X_1||X_3) \tag{35}$$

and checks whether C_8' equals to C_8. If not, S_j fails to authenticate user, and the session is terminated. Otherwise, S_j verifies user successful.

3.4 Password update phase

When a user U_i wants to update password, he/she should run as follows:

(1) U_i inputs ID_i, PW_i and B_i into smart card. The smart card computes

$$R_i' = Rep(B_i', P_i), \tag{36}$$

$$D_i' = h(h(ID_i||PW_i||R_i')\ mod\ m) \tag{37}$$

and checking whether D_i' equals to D_i, if they are not equal, the smart card fails to authenticate user, and rejects the request of password update. Otherwise U_i inputs a new password PW_i^*.

(2) The smart card computes

$$D_i^* = h(h(ID_i||PW_i^*||mod\ m) \tag{38}$$

$$F_i^* = F_i \oplus h(PW_i||R_i') \oplus h(PW_i^*||R_i') \tag{39}$$

Finally, D_i^* and F_i^* are stored in the smart card to replace D_i and F_i respectively.

4 Security analysis and performance comparisons

4.1 The proof of authentication based on authentication test method

The process of our improved scheme is as follow.

Step 1 $U_i \rightarrow S_j : C_1, C_2, X_1, T_U$

Step 2 $S_j \rightarrow RC : C_1, C_2, X_1, T_U, X_3, C_3$

Step 3 $RC \rightarrow S_j : C_4, C_5, X_6$

Step 4 $S_j \rightarrow U_i : C_6, C_7, X_3$

Step 5 $U_i \rightarrow S_j : C_8$

where X_1 in Eq. (10), X_2 in Eq. (11), $\mathrm{E_i}$ in Eq. (12), UR in Eq. (13), C_1 in Eq. (14), C_2 in Eq. (15), C_3 in Eq. (17), C_4 in Eq. (24), C_5 in Eq. (25), C_6 in Eq. (26), C_7 in Eq. (30), C_8 in Eq. (34). In our scheme, $\mathrm{E_i}$ in Eq. (4) is the secret parameter sharing by $\mathrm{U_i}$ and RC and $\mathrm{R_j}$ in Eq. (1) is the secret parameter sharing by S_j and RC. x and X are private secret key and public key of RC respectively. X_2 is the agreement session key between U_i and RC. SK_i and SK_j are the agreement session key between U_i and S_j. X_1 and X_3 are fresh temporary public keys. $h(k||m)/h(k||m)/h(m_1|| k||m_2)$ denote the hash value of message $k||m/k||m/m_1|| k||m_2$. Assuming k is a secret parameter shared by U_i and RC or S_j and RC, nobody is able to compute hash value expect who has learned the secret parameter k. It is obviously that this situation is equal to $\{m\}_k$ in the authentication test method.

Let $(\sum, P)$ be an infiltrated strand space, the strands space model given in Fig. 2. $(\sum, P)$ is an our improved scheme space if $\sum$ have four kinds of strands.

(1) Penetrator strands.

(2) User strands with trace User $[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, defined to be $\langle +\{C_1, C_2, X_1, T_U\}, -\{C_6, C_7, X_3\}, +\{C_8\} \rangle$.

(3) Server strands with trace Ser $[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$ defined to be $\langle -\{C_1, C_2, X_1, T_U\}, +\{C_1, C_2, X_1, T_U, SID_j, X_3, C_3\}, -\{C_4, C_5, C_6\}, +\{C_6, C_7, X_3\}, \rangle$, $\langle -\{C_8\} \rangle$.

(4) RC strands with trace $[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, defined to be $\langle -\{C_1, C_2, X_1, T_U, SID_j, X_3, C_3\}, +\{C_4, C_5, C_6\} \rangle$.

The users guarantees. We suppose:

(1) $\sum$ is our improved scheme space. C is a buddle containing a user's strand S_i with trace $S_i \in \mathrm{User}[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, and the $\mathrm{C}-$ height is 3.

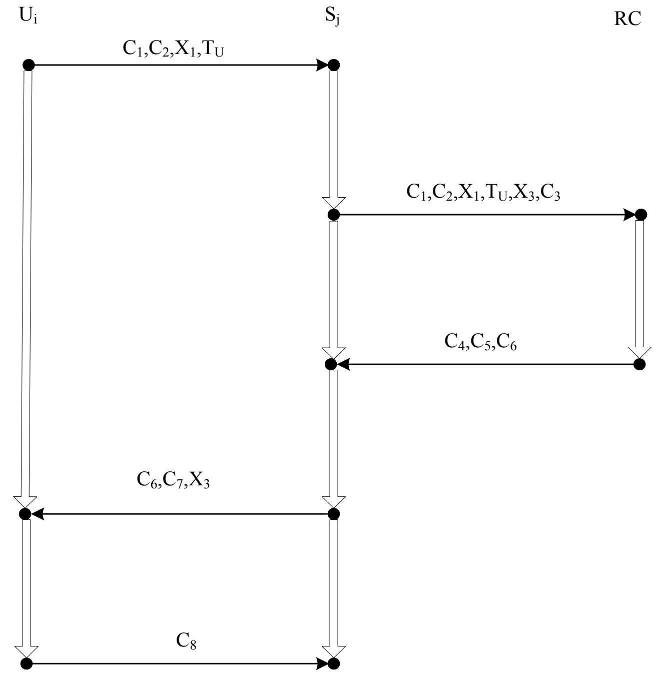

Figure 2: The strands space model of our scheme

(2) $E_i \notin K_p, SK_i \notin K_p, SK_j \notin K_p$.

(3) X_1, X_3 is uniquely originating in $\sum$.

***Lemma** 1* (User authenticate RC): The user can authenticate RC if the bundle contains RC's stand with trace $S_R \in RC[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, and the C −height is 2.

Proof: According to our improved scheme protocol space, X_1 is uniquely originating at node $\langle S_i, 1\rangle$ in $\sum$. The edge $\langle S_i, 1\rangle \Rightarrow \langle S_i, 2\rangle$ is a transformed edge for X_1, and $E_i \notin K_p$, then the edge $\langle S_i, 1\rangle \Rightarrow \langle S_i, 2\rangle$ is an incoming test for X_1 in $h(X_2||E_i||ID_i||SID_j||X_1||X_3)$. $h(X_2||E_i||ID_i||SID_j||X_1||X_3)$ is a test component for X_1. According to the incoming test proposition, there exist regular nodes $m, m' \in C$ such that $h(X_2||E_i||ID_i||SID_j||X_1||X_3)$ is a component of m', and the edge $m \Rightarrow^{+} m'$ is a transforming edge for X_1. Thus, the transforming edge $m \Rightarrow^{+} m'$ must be $\langle S_i, 1\rangle \Rightarrow \langle S_i, 2\rangle$, and S_j has C −height 2. This proves that the user successfully authenticates RC in our improved protocol space.

Lemma 2 (User authenticate Server): The user can authenticate a server if the bundle contains the server's stand with trace $S_j \in Ser[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, and the C −height is 5.

Proof: According to our improved scheme protocol space, X_1 is uniquely originating at node$\langle S_i, 1\rangle$ in $\sum$. The edge $\langle S_i, 1\rangle \Rightarrow \langle S_i, 2\rangle$ is a transformed edge for X_1, and $SK_i \notin K_p, SK_j \notin K_p$, then the edge $\langle S_i, 1\rangle \Rightarrow \langle S_i, 2\rangle$ is an incoming test for X_1 in

$h(ID_i||SID_j||C_6||SK_j||X_1||X_3)$. $h(ID_i||SID_j||C_6||SK_j||X_1||X_3)$ is test component for X_1. According to incoming test proposition, there exist regular nodes $m, m' \in C$ such that $h(ID_i||SID_j||C_6||SK_j||X_1||X_3)$ is a component of m', and the edge $m \Rightarrow^+ m'$ is a transforming edge for X_1. Thus, the transforming edge $m \Rightarrow^+ m'$, must be $\langle S_i, 1\rangle \Rightarrow \langle S_i, 2\rangle$, and S_j has $\mathrm{C}-$ height 5. This proves that the user successfully authenticates server in our improved protocol space.

Lemma 3 (RC authenticate user): RC can authenticate a user if the bundle contains the user's stand with trace $S_i \in User[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, and $\mathrm{C}-$height is 3.

Proof: According to our improved scheme protocol space X_1 is uniquely originating at node $\langle S_i, 1\rangle$ in $\sum$. Because $E_i \notin K_p$, then the node $\langle S_i, 1\rangle$ is an unsolicited Test for X_1 in $h(E_i||SID_j||ID_i||X_1||X_2||T_U)$. $h(E_i||SID_j||ID_i||X_1||X_2||T_U)$ is test component for X_1. According to unsolicited Test Proposition, there exist regular nodes such that $h(E_i||SID_j||ID_i||X_1||X_2||T_U)$ is a component of m. Thus, the node m must be $\langle S_i, 1\rangle$, and S_i has $\mathrm{C}-$height 3. This proves that RC successfully authenticates the user in our improved protocol space. The servers guarantees, suppose:

(1) $\sum$ is our improved scheme space, and C is a bundle containing a server's strand S_j with trace $S_j \in Server\left[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j\right]$ and the $\mathrm{C}-$height is 3.

(2) $R_j \notin K_p, SK_i \notin K_p, SK_j \notin K_p$.

(3) X_1, X_3 is uniquely originating in $\sum$.

Lemma 4 (server authenticate RC): The server can authenticate RC if the bundle contains RC's stand with trace $S_R \in RC[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, and the $\mathrm{C}-$height is 2.

Proof: According to our improved scheme protocol space, X_3 is uniquely originating at node $\langle S_i, 1\rangle$ in $\sum$. The edge $\langle S_j, 1\rangle \Rightarrow \langle S_j, 2\rangle$ is a transformed edge for X_3, and $R_j \notin K_p$, then the edge $\langle S_j, 1\rangle \Rightarrow \langle S_j, 2\rangle$ is an incoming test for X_3 in $h(SID_j||ID_i||R_j||X_1||X_3)$. $h(SID_j||ID_i||R_j||X_1||X_3)$ is test component for X_3. According to incoming test proposition, there exist regular nodes $m, m' \in C$ such that $h(SID_j||ID_i||R_j||X_1||X_3)$ is a component of m', and the edge $m \Rightarrow^+ m'$ is a transforming edge for X_1. Thus, the transforming edge $m \Rightarrow^+ m'$ must be $\langle S_j, 1\rangle \Rightarrow \langle S_j, 2\rangle$, and S_j has $\mathrm{C}-$height 5. This proves that the server successfully authenticates RC in our improved protocol space.

Lemma 5 (server authenticate user): The server can authenticate a user if the bundle contains server's stand with trace $S_i \in User[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, and the $\mathrm{C}-$height is 3.

Proof: According to our improved scheme protocol space, X_3 is uniquely originating at node$\langle S_j, 1\rangle$ in $\sum$. The edge $\langle S_j, 1\rangle \Rightarrow \langle S_j, 5\rangle$ is a transformed edge for X_3, and $SK_i \notin K_p$, $SK_j \notin K_p$, then the edge $\langle S_j, 1\rangle \Rightarrow \langle S_j, 5\rangle$ is an incoming test for X_3 in $h(SID_j||ID_i||SK_i||X_1||X_3)$. $h(SID_j||ID_i||SK_i||X_1||X_3)$ is test component for X_3 . According to incoming test proposition, there exist regular nodes $m, m' \in C$ such that $h(SID_j||ID_i||SK_i||X_1||X_3)$ is a component of m' , and the edge $m \Rightarrow^+ m'$ is a transforming edge for X_1. Thus, the transforming edge $m \Rightarrow^{+} m'$ must be $\langle S_j, 1\rangle \Rightarrow \langle S_j, 5\rangle$, and S_i has C −height 3. This proves that the server successfully authenticates the user in our improved protocol space.

Lemma 6 (RC authenticate server): RC can authenticate user if buddle contain user's stand with trace $S_j \in Server[ID_i, SID_j, X, X_1, X_3, T_U, E_i, R_j]$, and the C −height is 5.

Proof**:** According to our improved scheme protocol space X_3 is uniquely originating at node $\langle S_j, 1\rangle$ in $\sum$. Because $R_j \notin K_p$, then the node $\langle S_i, 1\rangle$ is an unsolicited Test for X_3 in $h(C_1||C_2||X_1||T_U||R_j||X_3)$. $h(C_1||C_2||X_1||T_U||R_j||X_3)$ is test component for X_3 . According to unsolicited Test Proposition, there exist regular nodes such that $h(C_1||C_2||X_1||T_U||R_j||X_3)$ is a component of m. Thus, the node m must be $\langle S_j, 1\rangle$, and S_j has C −height 5. This proves that RC successfully authenticates server in our improved protocol space.

Theorem 1 (Mutual Authentication): In our proposed protocol, if (1) $\sum$ is our improved scheme space and C is a bundle containing an users strand S_i and a servers strand S_j ,and the C −height of user strand is 3 and the C − height of server strand is 5. (2) $E_i \notin K_p, R_j \notin K_p, SK_i \notin K_p, SK_j \notin K_p$. (3) X_1, X_3 is uniquely originating in $\sum$, then our improved scheme is a secure mutual authentication scheme among user, server and RC.

Proof**:** According to Lemma 1 Lemma 6, user can authenticate RC and server successful, the server can authenticate RC and user successful, RC can authenticate user and the server. Therefore, Theorem 1 holds.

4.2 Further security analysis of our proposed scheme

Resistant to privileged insider attack**:** In user registration phase, U_i sends ID_i instead of PW_i in plain text, C_i in Eq. (3) to RC, in which b_i is a random number unknown to RC. In this process, the insider cannot access the password PW_i due to the irreversible property of the one-way hash function. Thus, our scheme can resist privilege insider attack.

Resistant to stolen-verifier attack**:** In stolen-verifier attack, an adversary obtains the verification information stored in the server. In our scheme, the server maintains two identity information table, one is the user's and the other is the server's. Two table contain no information related to the password. RC only needs to maintain the private key x and y. Therefore, our scheme can resist stolen-verifier attack.

Resistant to user impersonation attack**:** In our scheme, in order to impersonate as $\mathrm{U_i}$, the adversary has to generate a valid login request $\mathrm{M_1} = \{\mathrm{C_1}, \mathrm{C_2}, \mathrm{X_1}, \mathrm{T_U}\}$, where $\mathrm{X_1}$ see Eq. (10), $\mathrm{X_2}$ see Eq. (11), $\mathrm{E_i}$ see Eq. (12), UR see Eq. (13), $\mathrm{C_1}$ see Eq. (14), $\mathrm{C_2}$ see Eq. (15). An adversary who wants to impersonate user U_i must know the user identity ID_i and the secret parameter E_i at same time. If the adversary has obtained the user identity ID_i, he still does not learn E_i without knowing x and u_i according to Eq. (4). In addition, he cannot obtain E_i in user re-registration phase. Therefore, our proposed scheme can resist user impersonation attack.

Resistant to server spoofing attack**:** In a multi-server environment, the server is a semi-trusted party. So, the server may try to masquerade as a user to fool RC. In our proposed scheme, we use time stamp to prevent this action. When RC receives message, he will verify time stamp transmitted by user, if the time stamp is invalid, RC rejects this session. Moreover, the time stamp of user is protect by C_2 see Eq. (15), no one can forge C_2 without knowing the secret parameter E_i and ID_i . Therefore, when RC has been authenticated by a user, the session must be user initialized one rather than server masquerade session. Thus, our proposed scheme could withstand server spoofing attack.

***Resistant to replay attack*:** In our improved scheme, we use time stamp and random number to prevent replay attack. The random a and b are fresh for current session. The time stamp is the current time of the system. When the adversary replays previous message, RC cannot pass the time stamp verification. In addition, if the session key agreed by them are not equal, U_i and S_j cannot authenticate each other successful. Therefore, when user, server and RC authenticated each other successfully, it must be current session, not previous session. So, our scheme can avoid replay attack.

***Three-factor security*:** Firstly, we assume that an adversary A has been obtained the user's password and biometric. Obviously, A cannot forge a legitimate user. Then, we assume that A has obtained the secret parameters in the smart card. Unfortunately, A still cannot guess the correct value of password. The reason of this is that $|D_{pw}/m|$ candidates of the password are existed, where $|D_{pw}|$ is the space of password [Xiong, Li, Zeng et al. (2019)]. Therefore, the proposed scheme is secure in three-factor security.

***Resistant to wrong password and biometric login attack*:** In our scheme, secret information D_i in Eq. (7) stored in the smart is designed to check user login. If the user inputs wrong password PW_i^* or biometric R_i^*, the smart card will reject user login by checking whether D_i and $h(h(ID_i||PW_i||R_i) mod\ m)$ are equal. Therefore, our scheme can quickly detect unauthorized login with wrong password.

***Resistant to denial-of-service attack in password update phase*:** In our scheme, password update can be accomplished in the smart card, and it is not assisted by server or RC. Wrong password will be rejected by the smart card through password detection mechanism. Therefore, our scheme can resist denial-of-service attack in password update phase.

Mutual authentication among user, server and RC*:** Mutual authentication is a very important security feature for user authentication scheme, which allows any participant to authenticate the other participants. In Li et al.'s scheme [Li, Wang, Shen et al. (2016)], server only knows user communicated with him is a legal user, but he does not know who the user is without learning the identity of user. In our scheme, after RC has been authenticated by U_i and S_j, he sends the identity of U_i to S_j by computing C_4 see Eq. (4). Thus S_j knows who will communicate with him. Moreover, from ***Theorem 1 we know that mutual authentication can be achieved among user, server and RC by using authentication test method in our scheme. Therefore, our scheme could provide mutual authentication among user, server and RC.

***Known key security*:** After mutual authentication between user and server, they will agree on session key $SK_i = abP = SK_j$. Using this agreed session key user and server establish a confidentiality channel. In our scheme, the agreed session key is independent and different from other session keys. If some agreed session keys are disclosed, the other agreed session keys still remain secure. Therefore, our proposed scheme can provide known key security.

***Perfect forward secrecy*:** Perfect forward secrecy means that previous established session agreement key remains secure when the long-term key of the user, server and RC are disclosed. In our scheme, the long-term key consists of three kinds of key: the secret parameter E_i of user and RC, the secret parameter R_j of server and RC, and the private key x of RC. Even if the value E_i, R_j, and x are compromised, the session agreement $SK_i = abP = SK_j$ of previous session remains secure, because the adversary cannot compute session key with X_1 and X_3 due to the hardness of ECCDH problem. Therefore, out proposed scheme can achieve perfect forward secrecy.

4.3 Functionality comparisons

In this section, we compare security features of our improved scheme with Li et al.'s scheme [Li, Wang, Shen et al. (2016)] and Odelu et al.'s scheme [Odelu, Das and Goswami (2015)]. The results of comparison are listed in Tab. 2. From Tab. 2, we can see that our scheme is the only one that is capable of resisting all known attacks and fulfills the desirable security features.

Table 2: Functionality comparisons

Security attribute	**Li's scheme**	**Odelu's scheme**	**Ours scheme**
Resistant to privileged insider attack	Yes	Yes	Yes
Resistant to stolen-verifier attack	Yes	Yes	Yes
Resistant to user impersonation attack in registration phase	No	Yes	Yes
Resistant to server spoofing attack	No	Yes	Yes
Resistant to replay attack	Yes	Yes	Yes
Three-factor security	No	No	Yes
Resistant to wrong password and biometric login attack	Yes	Yes	Yes
Resistant to denial-of-service attack in password update phase	Yes	Yes	Yes
Mutual authentication among user, server and RC	Yes	Yes	Yes
Known key security	Yes	Yes	Yes
Perfect forward secrecy	Yes	Yes	Yes

4.4 Performance comparisons

For efficiency analysis, we compare the time complexity of our scheme with Li et al.'s scheme [Li, Wang, Shen et al. (2016)] and Odelu et al.'s scheme [Odelu, Das and Goswami (2015)], including the server registration phase, user registration phase and authenticated key agreement phase. To facilitate analysis, the notations are defined as follows.

T_m: the time complexity for ECC point multiplication operation.

T_h: the time complexity of one-way hash function.

The results of performance comparisons are summarized in Tab. 3. From Tab. 3, we can see that the computation cost of our scheme is more economic than that of Li's scheme [Li, Wang, Shen et al. (2016)] and Odelu's et al. scheme [Odelu, Das and Goswami (2015)]. Because the time complexity of hash function is low [Xu and Wu (2015)], the time cost of three schemes are almost at the same level. However, our scheme provides higher security functionality compared with the other two schemes. Therefore, our scheme is more secure but costs less.

Table 3: Performance comparisons

Schemes	**Li's scheme**	**Odelu's scheme**	**Ours scheme**
Server registration phase	T_h	$2T_h$	T_h
User registration phase	$5T_h$	$4T_h$	$4T_h$
Authenticated key agreement phase	$6T_m+22T_h$	$6T_m+24T_h$	$6T_m+21T_h$
Overall computation cost	$6T_m+28T_h$	$6T_m+30T_h$	$6T_m+26T_h$

5 Conclusions

In this paper, we have proposed a new three-factor authenticated key agreement protocol to remedy the problem of three-factor in Odelu's scheme [Odelu, Das and Goswami (2015)]. We have proved our scheme fulfills mutual authentication by using the authentication test method. Moreover, through the informal security analysis, we have shown that our scheme can resist various known attacks and provide more security features. At last, our scheme has been compared with two related schemes. The comparison has shown that our improved scheme provides not only useful and security functional features such as server anonymity and mutual authentication but also has high computational efficiency.

Funding Statement: This work is supported by the Sichuan education department research project (No. 16226483), Sichuan Science and Technology Program (No. 2018GZDZX0008), Chengdu Science and Technology Program (No. 2018-YF08-00007-GX), the National Natural Science Foundation of China (No. 61872087).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Gu, K.; Yang, L. H.; Yin, B. (2018): Location data record privacy protection based on differential privacy mechanism. *Information Technology and Control*, vol. 47, no. 4, pp. 639-654.

Guttman, J. D.; Fbrega, F. J. T. (2000): Authentication tests. *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 96-109.

Guttman, J. D.; Fbrega, F. J. T. (2002): Authentication tesssts and the structure of bundles. *Theoretical Computer Science*, vol. 283, no. 2, pp. 333-380.

He, D. B.; Wang, D. (2015): Robust biometrics-based authentication scheme for multi-server environment. *IEEE Systems Journal*, vol. 9, no. 3, pp. 816-823.

He, S. M.; Zeng, W. N.; Xie, K.; Yang, H. M.; Lai, M. Y. et al. (2017): PPNC: privacy preserving scheme for random linear network coding in smart grid. *KSII Transactions on Internet & Information Systems*, vol. 11, no. 3, pp. 1510-1532.

Hsiang, H. C.; Shih, W. K. (2009): Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment. *Computer Standards & Interfaces*, vol. 31, no. 6, pp. 1118-1123.

Jiang, Q.; Khan, M. K.; Lu, X.; Ma, J.; He, D. (2016): A privacy preserving three-factor authentication protocol for e-Health clouds. *The Journal of Supercomputing*, vol. 72, no. 10, pp. 3826-3849.

Lee, C. C.; Lin, T. H.; Chang, R. X. (2011): A secure dynamic ID based remote user authentication scheme for multi-server environment using smart cards. *Expert System with Applications*, vol. 38, no. 11, pp. 13863-13870.

Li, L. H.; Lin, I. C.; Hwang, M. S. (2001): A remote password authentication scheme for multi-server architecture using neural networks. *IEEE Transaction on Neural Network*, vol. 12, no. 6, pp. 1498-1504.

Li, X.; Wang, K. H.; Shen, J. (2016): An enhanced biometric-based user authentication scheme for multi-server environment in critical systems. *Ambient Intelligence and Humanized Computing*, vol. 7, pp. 427-443.

Liao, Y. P.; Wang, S. S. (2009): A secure dynamic ID based remote user authentication scheme for multi-server environment. *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 24-29.

Lin, I. C.; Hwang, M. S.; Li, L. H. (2003): A new remote user authentication scheme for multi-server architecture. *Future Generation Computer Systems*, vol. 19, pp. 13-22.

Mishra, D. (2016): Design and analysis of a provably secure multi-server authentication scheme. *Wireless Personal Communications*, vol. 86, pp. 1095-1119.

Odelu, V.; Das, A. K.; Goswami, A. (2015): A secure biometrics-based multi-server authentication protocol using smart cards. *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1953-1966.

Perrig, A.; Song, D. (2000): Looking for diamonds in the desert-extending automatic protocol generation to three-party authentication and key agreement. *Proceedings of the 13th IEEE Computer Security Foundations Workshop*, pp. 64-76.

Pointcheval, D.; Zimmer, S. (2008): Multi-factor authenticated key exchange. *Applied Cryptography and Network Security*, pp. 277-295.

Shen, H.; Gao, C. Z.; He, D. B.; Wu, L. B. (2015): New biometrics-based authentication scheme for multi-server environment in critical systems. *Ambient Intelligence and Humanized Computing*, vol. 6, pp. 825-834.

Sood, S. K.; Sarje, A. K.; Singh, K. (2011): A secure dynamic identity-based authentication protocol for multi-server architecture. *Network and Computer Applications*, vol. 34, no. 2, pp. 609-618.

Tsai, J. L. (2008): Efficient multi-server authentication scheme based on one-way hash function without verification table. *Computers & Security*, vol. 27, pp. 115-121.

Tsaur, W. J.; Li, J. H.; Lee, W. B. (2012): An efficient and secure multi-server authentication scheme with key agreement. *Journal of Systems and Software*, vol. 85, no. 4, pp. 876-882.

Tsaur, W. J.; Wu, C. C.; Lee, W. B. (2004): A smart card-based remote scheme for password authentication in multi-server Internet services. *Computer Standards & Interfaces*, vol. 27, no. 1, pp. 39-51.

Wu, F.; Xu, L. L.; Kumari, S.; Xiong, L. (2015): A novel and provably secure biometrics-based three-factor remote authentication scheme for mobile client-server networks. *Computers and Electrical Engineering*, vol. 45, pp. 274-285.

Xiong, L.; Li, F.; Zeng, S.; Peng, T.; Liu, Z. (2019): A blockchain-based privacy-awareness authentication scheme with efficient revocation for multi-server architectures. *IEEE Access*, vol. 7, pp. 125840-125853.

Xu, L.; Wu, F. (2015): Cryptanalysis and improvement of a user authentication scheme preserving uniqueness and anonymity for connected health care. *Medical Systems*, vol. 39, no. 2, pp. 1-9.

Yin, C. Y.; Ju, X. K.; Yin, Z. C.; Wang, J. (2019): Location recommendation privacy protection method based on location sensitivity division. *Journal on Wireless Communications and Networking*, https://doi.org/10.1186/s13638-019-1606-y.

Yoon, E. J.; Yoo, K. Y. (2013): Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem. *The Journal of Supercomputing*, vol. 63, no. 1, pp. 235-255.

Yu, J. S.; Wang, G. L.; Mu, Y.; Gao, W. (2014): An efficient generic framework for three-factor authentication with provably secure instantiation. *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 12, pp. 2302-2313.

Zhang, M.; Zhang, J. S.; Zhang, Y. (2015): Remote three-factor authentication scheme based on fuzzy extractors. *Security and Communication Networks*, vol. 8, pp. 682-693.