Journal of Information Hiding and Multimedia Signal Processing ©2015 ISSN 2073-4212
Ubiquitous International Volume **6**, Number **2**, March **2015**

A One-Way Authentication Key Agreement Scheme with User Anonymity Based on Chaotic maps towards Multi-Server Architecture

Hongfeng Zhu, Yifeng Zhang, and Yan Zhang

Software College, Shenyang Normal University
No.253, HuangHe Bei Street, HuangGu District, Shenyang, P.C 110034 - China
zhuhongfeng1978@163.com;{1548452125; 1505733680}@qq.com

Received September, 2014; revised November, 2014

Abstract. *One-way authenticated key agreement protocols, aiming at solving the problems to establish secure communications over public insecure networks, can achieve one-way authentication of communicating entities for giving a specific user strong anonymity and confidentiality of transmitted data. Public Key Infrastructure can design one-way authenticated key agreement protocols, but it will consume a large amount of computation. Because one-way authenticated key agreement protocols mainly concern on authentication and key agreement, we adopt multi-server architecture to realize these goals. About multi-server architecture, which allow the user to register at the registration center (RC) once and can access all the permitted services provided by the eligible servers. In other words, users do not need to register at numerous servers repeatedly. The combination of above-mentioned ideas can lead to a high-practical scheme in the universal client/server architecture. Based on these motivations, the paper firstly proposed a new one-way authenticated key agreement scheme based on multi-server architecture. Compared with related literatures recently, our proposed scheme can not only own high efficiency and unique functions, but is also robust to various attacks and achieves perfect forward secrecy. Finally, we give the security proof and the efficiency analysis of our proposed scheme.*

Keywords: One-way authentication, Key agreement, Multi-server architecture, Anonymity, Chaotic maps

1. **Introduction.** Authenticated key exchange (AKE) is one of the most important cryptographic components which is used for establishing an authenticated and confidential communication channel. Based on the number of participants, we can divide AKE protocols into three categories: two-party AKE protocols [7 - 10], three-party AKE protocols [11 - 13], and N-party AKE protocols [14 - 17]. Furthermore, based on the respective features in detail, the previous AKE protocols [7 - 31] can be classified many categories, we use two-party AKE protocols to set an example: such as using smart card [1 - 3], password-based [1 - 6], chaotic map-based [8 - 13], ID-based [16,17], anonymity [13,18], secret sharing [19, 20] and so on. Recently many researchers achieve AKE in the multi-server environment called multi-server authenticated key agreement (MSAKA) protocols. MSAKA protocols allow the user to register at the registration center (RC) once and can access all the permitted services provided by the eligible servers. In other words, users do not need to register at numerous servers repeatedly. MSAKA protocols mainly want to solve the problems in a traditional single server with authentication schemes [21,22]

which lead to the fact that user has to register to different servers separately. On a macro level MSAKA protocols can be divided into three phases in chronological order: Creative Phase: The pioneer work in the field was proposed by Li et al. [23] in 2001. However, Lin et al. [24] pointed out that Li et al.s scheme takes long time to train neural networks and an improved scheme based on ElGamal digital signature and geometric properties on the Euclidean plane has also been given. Development Phase: the main work in this phase is amended repeatedly. For example, Tsai [25] also proposed an efficient multi-server authentication scheme based on one-way hash function without a verification table. Because Tsais scheme only uses the nonce and one-way hash function, the problems associated with the cost of computation can be avoided in the distributed network environment. However, some researchers [26] pointed out that Tsais scheme is also vulnerable to server spoofing attacks by an insider server and privileged insider attacks, and does not provide forward secrecy. Diversification Phase: the research emphasis shifts to functionality. Therefore, identity-based MSAKA protocols, based on bilinear pairings or elliptic curve cryptosystem (ECC) MSAKA protocols, dynamic identity-based MSAKA protocols and other MSAKA protocols came up recently[26 - 28]. However, most existing AKE or MSAKA protocols have emphasized mutual authentication, in which both parties authenticate themselves to their peer. There are many scenes need not mutual authentication at all and we just need one-way authentication. We can take some facts as examples which are shown in the Fig.1. (1) Readers-to-journalists model: Readers act upon the perceived reputation of a news source, so reputation is a valuable commodity for journalists. No further authentication is required and since the information is public, channel secrecy is not required and does not affect the actions of either party. (2) Patient-to-expert model: On Internet, patients requiring medical advice may wish to do so anonymously, while still ensuring the confidentiality of their request and assurance that the medical advice received comes from an authentic, qualified source.

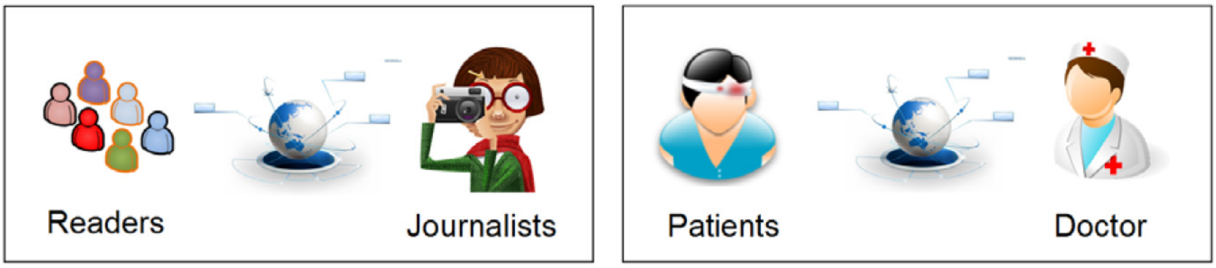

FIGURE 1. No need for mutual authentication environment on Internet

The key idea of one-way AKE is that one party wishes for no one to be able to determine his/her identity, including all the authorities. However, only a few protocols have considered the problem of one-way authentication. Goldberg [29] gave a specialized one-way AKE security definition for the Tor authentication protocol. The literature [30] described an identity-based anonymous authenticated key exchange protocol but with a limited session key secrecy definition based on key recovery, not indistinguishability. Morrissey, Smart, and Warinschi [31] analyzed the security of the Transport Layer Security (TLS) protocol in the context of one-way authentication, but with specialized security definitions. Recently, Goldberg and Stebila [32] provided an intuitive set of goals and present a formal model that captures these goals. Usually, public key encryption can be used for one-way AKE protocols, for example by having the client encrypt a session key under the server's public key. This mechanism is widely used, for example in the

RSA-based cipher suites in TLS [33] and in the KAS1 protocol in NIST SP800-56B [34]. The main contributions are shown as below: The paper firstly presents a new one-way authentication key agreement scheme towards multi-server architecture. Furthermore, the proposed protocol is based on chaotic maps without using modular exponentiation and scalar multiplication on an elliptic curve. In Security aspect, the protocol can resist all common attacks, such as impersonation attacks, man-in-the-middle attacks, etc. About functionality, the protocol also has achieved some well-known properties, such as perfect forward secrecy and execution efficiency. The rest of the paper is organized as follows: Some preliminaries are given in Section 2. Next, a One-Way AKE towards Multi-Server Architecture is described in Section 3. Then, the security analysis and efficiency analysis are given in Section 4 and Section 5. This paper is finally concluded in Section 6.

2. Preliminaries.

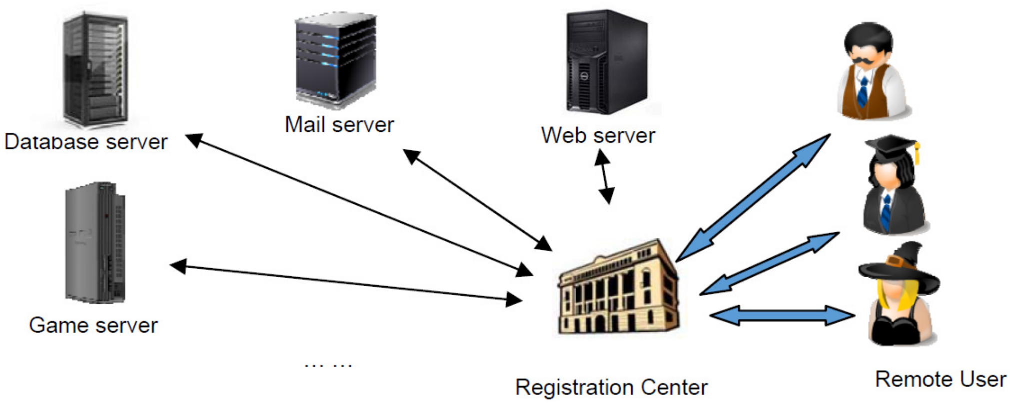

FIGURE 2. The traditional multi-server communication architecture

2.1. Multi-server architecture. In the multi-server environment, each user must perform authentication procedure to login the server for a transaction. If the user is in a single authentication architecture, then the user must register at various servers and memorize the corresponding identifications and passwords, which could not be convenient for a user. In order to make the registration to various servers easier for users, multi-server architecture schemes have been developed and proposed [23-28]. Basically, each user must register with the registration center to obtain a secure account. Then the user uses the secure account to perform the login and authentication procedures with various servers. Fig.2 shows the traditional multi-server environment.

2.2. Definition and properties of Chebyshev chaotic maps. Let n be an integer and let x be a variable with the interval $[-1,1]$. The Chebyshev polynomial $T_n(x) : [-1,1] \to [-1,1]$ is defined as $T_n(x) = \cos(ncos^{-1}(x))$. Chebyshev polynomial map $T_n : R \to R$ of degree n is defined using the following recurrent relation [35]:

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) \tag{1}$$

where $n \geq 2$, $T_0(x) = 1$, and $T_1(x) = x$.

The first few Chebyshev polynomials are:
$T_2(x) = 2x^2 - 1$, $T_3(x) = 4x^3 - 3x$, $T_4(x) = 8x^4 - 8x^2 + 1$,.......
One of the most important properties is that Chebyshev polynomials are the so-called semi-group property which establishes that

$$T_r(T_s(x)) = T_{r \cdot s}(x) \tag{2}$$

An immediate consequence of this property is that Chebyshev polynomials commute under composition

$$T_r(T_s(x)) = T_s(T_r(x)) \tag{3}$$

In order to enhance the security, Zhang [36] proved that semi-group property holds for Chebyshev polynomials defined on interval $(-\infty, +\infty)$. The enhanced Chebyshev polynomials are used in the proposed protocol:

$$T_n\left(x\right) = (2xT_{n-1}(x) - T_{n-2}(x))(\mathrm{mod} N) \tag{4}$$

where $n \geq 2$, $x \in (-\infty, +\infty)$, and N is a large prime number. Obviously,

$$T_{r \cdot s}(x) = T_r(T_s(x)) = T_s(T_r(x)) \tag{5}$$

Definition 2.1. *Semi-group property of Chebyshev polynomials:*
$T_r(T_s(x)) = \cos(r\cos^{-1}(s\cos^{-1}(x))) = \cos(rs\cos^{-1}(x)) = T_{sr}(x) = T_s(T_r(x))$

Definition 2.2. *Given x and y, it is intractable to find the integer s, such that $T_s(x) = y$. It is called the Chaotic Maps-Based Discrete Logarithm problem (CMBDLP).*

Definition 2.3. *Given x, $T_r(x)$,and $T_s(x)$, it is intractable to find $T_{rs}(x)$.It is called the Chaotic Maps-Based Diffie-Hellman problem (CMBDHP).*

2.3. **One-way Hash Function.** A secure cryptographic one-way hash function $h : a \rightarrow b$ has four main properties:
(1) The function h takes a message of arbitrary length as the input and produces a message digest of fixed-length as the output;
(2) The function h is one-way in the sense that given a, it is easy to compute $h\left(a\right) = b$.However, given b, it is hard to compute $h^{-1}\left(b\right) = a$;
(3) Given a, it is computationally infeasible to find a' such that $a' \neq a$, but $h\left(a'\right) = h\left(a\right)$;
(4) It is computationally infeasible to find any pair a, a' such that $a' \neq a$,but $h\left(a'\right) = h\left(a\right)$.

2.4. **Symmetric encryption.** A symmetric encryption scheme $E_k(Kgen, E, D)$ consists of three algorithms as follows:
(1)Randomized Key Generation Algorithm $Kgen$: it returns a key k drawn from the key space $Keys(E_k)$ at random.
(2) Encryption Algorithm E:it takes the key $k \in Keys(E_k)$ and a plaintext $M \in \{0, 1\}^*$ as the inputs and outputs a ciphertext $C \in \{0, 1\}^*$. So it can be written $C = E_k(M)$.
(3)Decryption Algorithm D: it takes the key $k \in Keys(E_k)$ and a ciphertext $C \in \{0, 1\}^*$ as the inputs and outputs a plaintext $M \in \{0, 1\}^*$. So it can be written $M = D_k(C)$.

2.5. **Explanation of some terms.**
(1)*Anonymity vs OTP and ID hiding*
Anonymity ensures that a user may use a resource or service without disclosing the users identity completely. ID hiding[A pseudonym is an identifier of a subject other than one of the subjects real names. ID hiding usually uses pseudonym to realize.] usually means that a user may use a resource or service without disclosing the users identity during the protocol interaction, which is a kind of privacy protection partly. Because the server may store the users identity.
OTP (one-time password) usually means that the password can be used only once but the ID is plaintext during the protocol interaction, so there is no privacy protection. The concrete differences are shown in Table1.

Table 1. Comparisons among Anonymity, OTP and ID hiding

Terms	Privacy Protection	Authentication	Security Level
Anonymity	√√√	One way	√√√
ID hiding	√	Mutual	√√
OTP	×	Mutual	√√

(2)*Anonymity vs Unlinkability*
Unlinkability [37] of two or more items of interest (IOIs, e.g., subjects, messages, actions, ...) from an attackers perspective means that within the system (comprising these and possibly other items), the attacker cannot sufficiently distinguish whether these IOIs are related or not. So in the context of key exchange, unlinkability and anonymity are in a sense equivalent.
(3)*Anonymity vs Untraceability* [37]
Untraceability: The signer is unable to link the message-signature pair with the corresponding view after the blind signature has been revealed to the public by the requester. So anonymity is a general term and untraceability is used in signature usually.
(4)*Anonymity vs Undetectability*
Undetectability [37] of an item of interest (IOI) from an attackers perspective means that the attacker cannot sufficiently distinguish whether it exists or not. So we can defer undetectability is a kind of pseudor anonymity just like pseudorandom number and true random number.
(5)*One-way AKE vs One-flow AKE*
In brief, we can view a one-way AKE protocol as the complement of a one-flow AKE protocol. One-flow AKE protocols are designed to establish a session key using a single message from the client to the server. It can provide mutual authentication by using two static keys (one each from the client and the server) and one ephemeral key (from the client). In contrast, one-way AKE can use one static key (from the server) and two ephemeralkeys (one each from the client and the server), but provides no authentication to the server.

2.6. **Security requirements.**
Secure communication schemes for remote mutual authentication and session key agreement for the multi-server architecture should provide security requirements [38,39], such as mutual/one-way authentication and key agreement, impersonation attack, man-in-the-middle attack, replay attack, known- key security, perfect forward secrecy, data integrity, off-line guessing attack, session key security and key compromise impersonation. The definitions and proofs of above-mentioned security requirements will be illustrated in Appendix A. detailedly.

3. **The Proposed One-Way AKE towards Multi-Server Architecture.** In this section, under the multi-server architecture, a chaotic maps-based one-way authentication key agreement scheme is proposed which consists of two phases: the servers registration

phase, one-way authentication key agreement phase. But firstly some notations are given which used in the proposed scheme.

Remark 3.1. *Because our proposed protocol is an one-way authentication scheme, there is no password update phase in our protocol.*

3.1. **Notations.** In this phase, any participant i has its identity ID_i, and public key $(x, T_{k_i}(x))$ and a secret key k_i based on Chebyshev chaotic maps, a secure one-way hash function $H(\cdot)$, and a pair of secure symmetric encryption/decryption functions $E_K()/D_K()$ with key K.The concrete notations used hereafter are shown in Table2.

TABLE 2. Notations

Symbol	Definition
SID_A	a temporary session
S_i, ID_{S_i}	The ith server, the identity of the ith server, respectively
a, r_i	nonces
$(x, T_k(x))$	public key based on Chebyshev chaotic maps
k	secret key based on Chebyshev chaotic maps
RC	registration center
$E_K() / D_K()$	a pair of secure symmetric encryption/decryption functions with the key K
H	A secure one-way hash function
$\|\|$	concatenation operation

3.2. **Servers registration phase.** Concerning the fact that the proposed scheme mainly relies on the design of Chebyshev chaotic maps-based in multi-server architecture, it is assumed that the servers can register at the registration center in some secure way or by secure channel. The same assumption can be set up for servers Fig.3 illustrates the server registration phase. The steps are performed during the server registration phase as follows.
Step 1. When a server(or an expert) wants to be a new legal service provider, she chooses her identity ID_{S_i} with her identification card in law. Then the server submits ID_{S_i} to the RC via a secure channel. **Step 2.** Upon receiving ID_{S_i} from the server, the RC com-

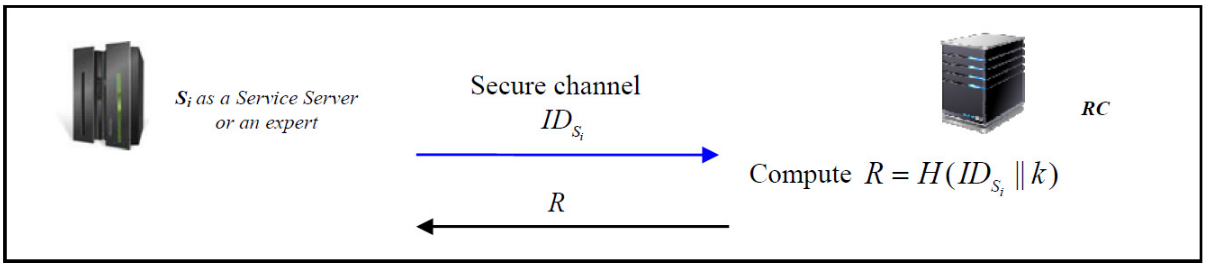

FIGURE 3. Server or a authenticated expert registration phase

putes $R = H(ID_{S_i}||k)$,where k is the secret key of RC.Then the server stores R in a secure way via a secure channel.

3.3. **One-way authenticated key agreement phase.** In this phase, one-way authenticated means that the server or the RC can be authenticated by the other two peers, but the user can not be authenticated by the the server or the RC to keep the user complete anonymity in the multi-server architecture. This concrete process is presented in the following Fig. 4.

Step 1. If Alice(assume Alice as an anonymous user) wishes to consult some personal issues establish with S_i (or an expert) in a secure way, she will choose a random integer number a and a temporary session SID_A. Then the device of Alice will compute $K_{A-RC} = T_aT_k(x)$, $H_A = H(SID_A||ID_{S_i}||T_a(x))$, $C_1 = E_{K_{A-RC}}(SID_A||ID_{S_i}||H_A)$.After that, Alice sends $m_1 = \{SID_A, T_a(x), C_1\}$ to S_i where she wants to get the servers service.

Step 2. After receiving the message $m_1 = \{SID_A, T_a(x), C_1\}$ from Alice, S_i will do the following tasks to ask RC for helping Alice to authenticate itself: S_i selects random r_i and computes $T_{r_i}(x)$, $C_2 = H(ID_{S_i}||m_1||R||T_{r_i}(x))$.And then sends the message $m_2 = \{ID_{S_i}, T_{r_i}(x), C_2, m_1\}$ to RC.

Step 3. Next, RC will help Alice to authenticate S_i and verify the temporary information by helping them to compute the session key. After receiving the message $m_2 = \{ID_{S_i}, T_{r_i}(x), C_2, m_1$, RC will do the following tasks:

(1) Authenticate S_i:Based on ID_{S_i}, RC can compute $R' = H(ID_{S_i}||k)$. Then RC computes $C_2' = H(ID_{S_i}||m_1||R'||T_{r_i}(x))$ and check if $C_2'? = C_2$. If above equations hold, that means S_i are legal participants in this instance because only S_i own R.

(2) Confirm S_i is the server that Alice wants to consult with: RC computes $K_{RC-A} = T_kT_a(x)$ and then decrypts C_1 to get $SID_A||ID_{S_i}||H_A$.Next, RC computes H_A'$= H(SID_A||ID_{S_i}||T_a(x))$. RC verifies H_A'$? = H_A$ and checks if ID_{S_i} in the C_1 equals to ID_{S_i} in plaintext or not. If holds, that means S_i is the server that Alice wants to consult with.

(3) Help S_i and Alice to get the session key: RC computes $C_3 = H(ID_{RC}||ID_{S_i}||m_1||R||T_{r_i}(x))$, $C_4 = E_{K_{RC-A}}(ID_{RC}||ID_{S_i}||m_1||T_{r_i}(x)||H_{RC})$ and $H_{RC} = H(SID_A||ID_{S_i}||ID_{RC}||T_{r_i}(x))$. Then RC sends the message ID_{RC}, C_4 to Alice and sends the message ID_{RC}, C_3 to S_i. If any authentication process does not pass, the protocol will be terminated immediately.

Step 4.

For Alice: After receiving the message ID_{RC}, C_4, Alice uses K_{A-RC} to decrypt C_4. Next Alice computes H_{RC}'$= H(SID_A||ID_{S_i}||ID_{RC}||T_{r_i}(x))$. Check if H_{RC}'$= H_{RC}$.If holds, Alice computes $SK = T_aT_{r_i}(x)$.

For S_i:After receiving the message ID_{RC}, C_3, S_i computes C_3'$= H(ID_{RC}||ID_{S_i}||m_1||R||T_{r_i}(x))$ and checks if C_3'$= C_3$.If holds, then S_i computes $SK = T_{r_i}T_a(x)$.

4. **Security Consideration.** The section analyzes the security of our proposed protocol. Let us assume that there are three secure components, including the two problems CMBDLP and CMBDHP cannot be solved in polynomial-time, a secure one-way hash function, and a secure symmetric encryption. Assume that the adversary has full control over the insecure channel including eavesdropping, recording, intercepting, modifying the transmitted messages. The definitions and analysis of the security requirements will be illustrated in Appendix A. From the Table 3, we can see that the proposed scheme can provide secure session key agreement, perfect forward secrecy and so on. As a result, the proposed scheme is more secure and has much functionality compared with the recent related scheme.

5. **Efficiency Analysis.** Compared to RSA and ECC, Chebyshev polynomial computation problem offers smaller key sizes, faster computation, as well as memory, energy and

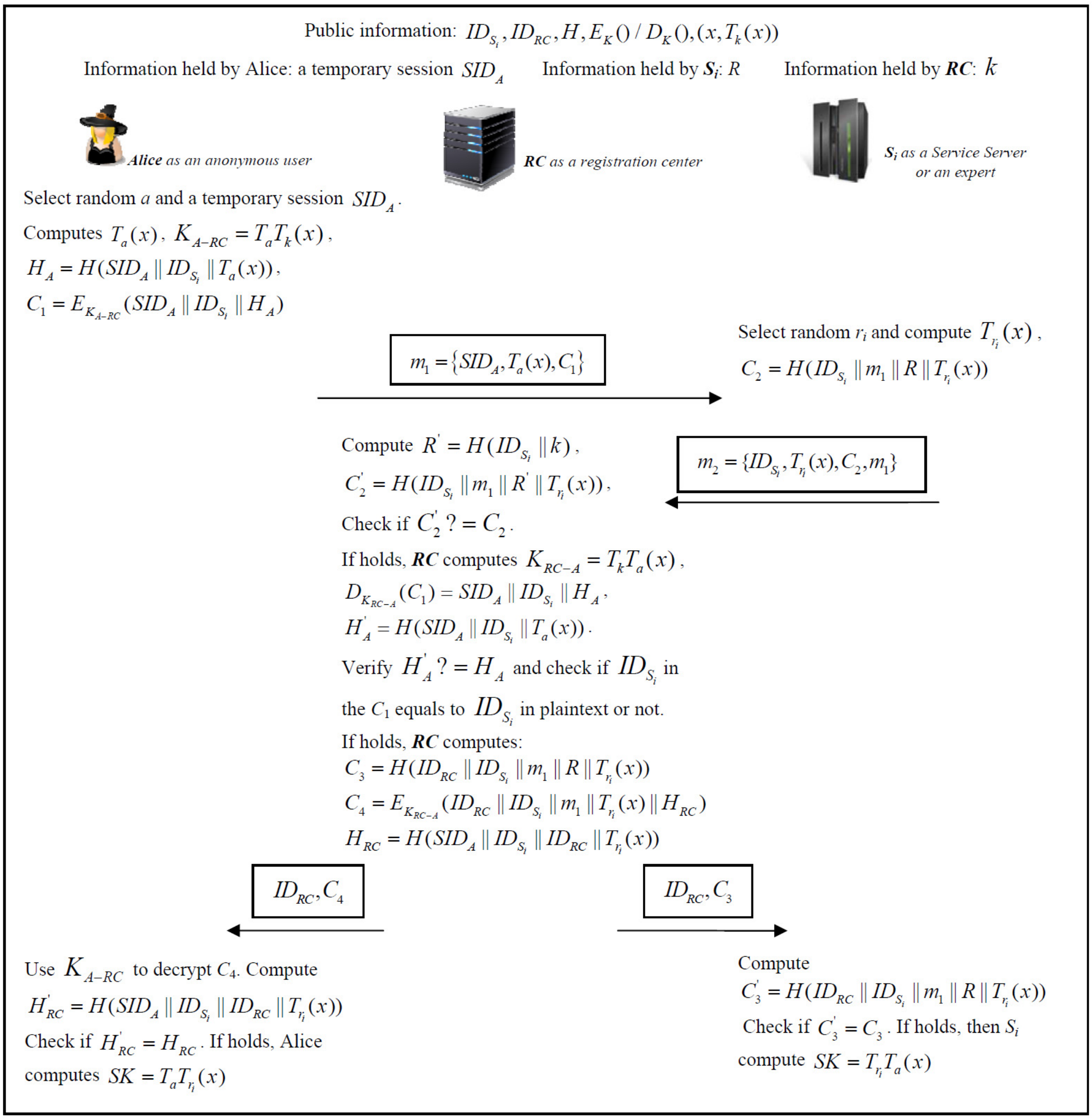

FIGURE 4. One-way authenticated key agreement phase

bandwidth savings. In our proposed protocol, no time-consuming modular exponentiation and scalar multiplication on elliptic curves are needed. However, Wang [35] proposed several methods to solve the Chebyshev polynomial computation problem. For convenience, some notations are defined as follows.
T_{hash}:The time for executing the hash function;
T_{sym}:The time for executing the symmetric key cryptography;
T_{XOR}:The time for executing the XOR operation;
T_{Exp}:The time for a modular exponentiation computation;

Table 3. Architecture and security of our proposed protocol

Criteria	[40](2013)	[41](2008)	[42](2009)	[43](2009)	Ours
Single registration	Yes	Yes	Yes	Yes	Yes (Only for servers)
Authentication	Mutual	Mutual	No	No	One-way
No verification table	Yes	Yes	Yes	Yes	Yes
Securely chosen password	Yes	Yes	Yes	No	No Need
Session key agreement	Yes	Yes	Yes	Yes	Yes
Privacy protection for a user	No	No	ID hiding	ID hiding	Anonymity
Freedom from time synchronization	Yes	No	No	No	Yes
Session key secrecy	Yes	No	No	No	Yes
Perfect forward secrecy	Yes	No	No	No	Yes
Resistance to replay attack	Yes	Yes	Yes	Yes	Yes
Resistance to stolen-verifier attack	Yes	Yes	Yes	Yes	Yes
Resistance to masquerading attack	Yes	No	No	No	Yes

T_{CH}:The time for executing the $T_n(x) \mod p$ in Chebyshev polynomial using the algorithm in literature[44].
Table 4 shows performance comparisons between our proposed scheme and the literature of [40-43] in multi-server architecture. The literature [40] consumes more computations than ours. And the literatures [41-43] own high-efficiency, but in secure aspect, they can not resist some common attacks such as masquerading attack and can not gain some common functionality such as session key secrecy or perfect forward secrecy. Therefore, as in Table 3 and Table 4, we can draw a conclusion that the proposed scheme has achieved the balance of efficiency and security.

Table 4. Efficiency of our proposed scheme

Phase		[40](2013)	[41](2008)	[42](2009)	[43](2009)	Ours
A		$2T_{hash} + 1T_{XOR}$	$2\ T_{hash} + 1T_{XOR}$	$5T_{hash} + 2T_{XOR}$	$8T_{hash} + 4T_{XOR}$	No Need
B		$1T_{hash}$	$1\ T_{hash}$	$1T_{hash}$	$1T_{hash}$	$1T_{hash}$
C		$2T_{hash} + 1T_{XOR} + 1T_{Exp}$	$1T_{hash} + 2T_{XOR}$	$6T_{hash} + 3T_{XOR}$	$7T_{hash} + 7T_{XOR}$	$1T_{svm} + 1T_{hash} + 1T_{CH}$
D	User	$1T_{hash} + 1T_{Exp}$	$4T_{hash} + 3T_{XOR}$	$3T_{hash}$	$2T_{hash}$	$1T_{hash} + 1T_{CH}$
	Server	$2T_{hash} + 2T_{Exp}$	$6T_{hash} + 7T_{XOR}$	$6T_{hash} + 3T_{XOR}$	$8T_{hash} + 6T_{XOR}$	$2T_{hash} + 1T_{CH}$
	RC	$6T_{hash}$	$6T_{hash} + 5T_{XOR}$	0	$5T_{hash} + 7T_{XOR}$	$5T_{hash} + 1T_{CH} + 2T_{svm}$
	Total	$9T_{hash} + 3T_{Exp}$	$16T_{hash} + 15T_{XOR}$	$9T_{hash} + 3T_{XOR}$	$15T_{hash} + 13T_{XOR}$	$8T_{hash} + 3T_{CH} + 2T_{svm}$
E		$2T_{hash} + 2T_{XOR}$	$2T_{hash} + 2T_{XOR}$	$4T_{hash} + 5T_{XOR}$	$4T_{hash} + 4T_{XOR}$	No Need
F		4 rounds	7 rounds	3 rounds	5 rounds	3 rounds
A: User registration **B**: Server registration **C**: Login phase **D**: Authentication phase (Session key establishment included) **E**: Password change phase **F**: Communication cost						

Table 5 presents the effciency in term of modular exponentiations(ME) and chebyshev polynomial(CP) computation of relevant one-way authentication key agreement protocols [32,
46-49]. The Diffie-Hellman protocol [46] is the basic protocol on which most other protocols in the literature are built upon. In the table we refer to the ephemeral-ephemeral variant that succumbs to man-in-the-middle attacks, but is a good benchmark for effiency. About some values (such as 1.33, 1.17 in the Table 5)of modular exponentiations, since the base is the same, squarings in the squareand-multiply algorithm can be parallelized [50] reducing the computational cost to 1.33 exponentiations. Therefore, from Table 5 we can see that the our proposed scheme has achieved the tight security and good efficiency. Moreover, our proposed scheme possesses expandability because it is realized in multi-server architecture.

Table 5. Efficiency in terms of modular exponentiations(ME) and chebyshev polynomial(CP)

Protocol	Efficiency (client)				Efficiency (server)				Authentication	Security	Architecture
	Off-line		On-line		Off-line		On-line				
	ME	*CP*	*ME*	*CP*	*ME*	*CP*	*ME*	*CP*			
DH[46]	1	0	1	0	1	0	1	0	none	insecure	Two-party
ØS[47]	1	0	1	0	1	0	1	0	one-way	insecure	Two-party
MQV[48]	1	0	1.17 (1.5)	0	1	0	1.17 (1.5)	0	mutual	non-tight	Two-party
UM[49]	1	0	2	0	1	0	2	0	mutual	limited	Two-party
NTOR[32]	1	0	2	0	1	0	1.33 (2)	0	one-way	tight	Two-party
Ours	0	2	0	0	0	1	0	0	one-way	tight	Multi-Server
Tight: The only signicant factor between the difficulty of breaking the key agreement protocol and the difficulty of solving the underlying function is the factor that comes from guessing the correct test session. In brief, an adversary only solves some kind of hard problem, the protocol can be compromised.											

6. **Conclusion.** This work provides a new approach to one-way authenticated key establishment towards multi-server architecture. The core ideas of the proposed scheme are the mutual authentication for the servers and RC and the anonymity for the users. Subsequently, we explain the practical motivations for authentication and secrecy assurances of parties engaging in one-way AKE protocols and some related terms. Based on our discussion we proposed a suitable protocol that covers those goals and offered an efficient protocol that formally meets the proposed security definition. Finally, after comparing with related literatures (multi-server schemes and one-way protocols) respectively, we found our proposed scheme has satisfactory security, efficiency and functionality. Therefore, our protocol is more suitable for practical applications.

REFERENCES

[1] L. Lamport, Password authentication with insecure communication, *Communications of the ACM*, vol. 24, no. 11, pp. 770-772, 1981.

[2] W.B. Hsieh, J.S. Leu, Exploiting hash functions to intensify the remote user authentication scheme, *Computers & Security*, vol. 31, pp. 791-798, 2012.

[3] J.L. Tsai, Efficient multi-server authentication scheme based on one-way hash function without verification table, *Computers & Security*, vol. 27, pp. 115121, 2008.

[4] S.M. Bellovin, M. Merritt, Encrypted key exchange: Password-based protocols secure against dictionary attacks, *IEEE Symposium on Research in Security and Privacy*, pp. 7284, 1992.

[5] J. Katz, P. MacKenzie, G. Taban, V. Gligor. Two-server password-only authenticated key exchange, *Journal of Computer and System Sciences*, vol. 78, pp. 651669, 2012.

[6] M. Bellare, P. Rogaway, Random oracles are practical: A paradigm for designing efficient protocols, *Proc. 1st ACM Conference on Computer and Communications Security*, pp. 6273, 1993.

[7] Baptista MS. Cryptography with chaos, *Phys. Lett, A 240*, pp.5054, 1998.

[8] P. Gong, P. Li ,and W. Shi , A secure chaotic maps-based key agreement protocol without using smart cards, *Nonlinear Dyn.70*, pp. 24012406, 2012.

[9] C. Lee, C. Hsu , A secure biometric-based remote user authentication with key agreement scheme using extended chaotic maps, *Nonlinear Dyn.71,* pp. 201211, 2013.

[10] C. Guo, C.C. Chang, Chaotic maps-based password-authenticated key agreement using smart cards, *Commun. Nonlinear Sci. Numer. Simul*, 2012.

[11] H. Lai, J. Xiao, L. Li, and Y. Yang, Applying semigroup property of enhanced Chebyshev polynomials to anonymous authentication protocol, *Math. Probl. Eng*, 2012.

[12] Q. Xie, J.M. Zhao, and X.Y. Yu, Chaotic maps-based three-party password-authenticated key agreement scheme, *Nonlinear Dyn*, vol. 74 , 2013.

[13] C. Lee, C. Li, and C. Hsu, A three-party password-based authenticated key exchange protocol with user anonymity using extended chaotic maps, *Nonlinear Dyn,* vol. 73, pp. 125132, 2013.

[14] E. Bresson, O. Chevassut and D. Pointcheval, Group Diffie-Hellman key exchange secure against dictionary attack, *Asiacrypt*, pp. 497514, 2002.

[15] H. Li, C.K. Wu, and J. Sun, A general compiler for password-authenticated group key exchange protocol, *Information Processing Letters*, pp. 160167, 2010.

[16] T.Y. Wu, Y.M. Tseng, and T.T. Tsai, A revocable ID-based authenticated group key exchange protocol with resistant to malicious participants, *Computer Networks,* vol. 56, no. 12, pp. 2994-3006, 2012.

[17] T.Y. Wu, Y.M. Tseng, Towards ID-based authenticated group key exchange protocol with identifying malicious participants, *Informatica: International Journal*, vol. 23, no. 2, pp. 315-334, 2012.

[18] H. Tseng, R. Jan, and W. Yang, A chaotic maps-based key agreement protocol that preserves user anonymity. *IEEE International Conference on Communications*, pp. 16, 2009.

[19] P. MacKenzie, T. Shrimpton, and M. Jakobsson, Threshold password-authenticated key exchange, *J. Cryptology*, vol. 19, no. 2, pp. 2766, 2006.

[20] M.D. Raimondo, R. Gennaro, Provably secure threshold password-authenticated key exchange, *J. Comput. System Sci*, vol. 72, no. 6, pp. 9781001, 2006.

[21] Y.F. Chang, C.C. Chang, and Y.W. Su , A secure improvement on the user-friendly remote authentication scheme with no time concurrency mechanism, *Proc. of IEEE 20th international conference on advanced information networking and applications*, pp. 741745, 2006.

[22] M.K. Khan, J. Zhang, Improving the security of a flexible biometrics remote user authentication scheme. *Comput Stand Interfaces*, vol. 29, no. 1, pp. 8285, 2007.

[23] L.H. Li, I.C. Lin, and M.s. Hwang, A remote password authentication scheme for multi-server architecture using neural networks, *IEEE Trans. on Neural Networks*, vol. 12, no.6, pp. 14981504, 2001.

[24] I.C. Lin, M.S. Hwang, and L.H. Li, A new remote user authentication scheme for multi-server architecture, *Future Generation Computer Systems*, vol. 19, no. 1, pp.1322, 2003.

[25] J.L. Tsai, Efficient multi-server authentication scheme based on one-way hash function without verification table, *Comput Secur*, vol. 27, no. 34, pp. 115121, 2008.

[26] S.P. Ravi, C.D. Jaidhar, and T. Shashikala, Robust Smart Card Authentication Scheme for Multi-server Architecture. *Wireless Pers Commun*, vol. 72, pp. 729745, 2013.

[27] B. Wang, M. Ma, A smart card based efficient and secured multi-server authentication scheme. *Wireless Personal Communications*, vol. 10, 2012.

[28] E.J. Yoon, K.Y. Yoo, Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem. *J Supercomput*, vol. 63, pp. 235255, 2013.

[29] I. Goldberg, *On the security of the Tor authentication protocol*,LNCS4258, Springer, pp. 316-331, 2006.

[30] A. Kate, G.M. Zaverucha, and I. Goldberg, Pairing-based onion routing with improved forward secrecy, *ACM Trans. on Information and System Security*, vol. 13, no. 4, pp. 29, 2010.

[31] P. Morrissey, N.P. Smart, and B. Warinschi, A modular security analysis of the TLS handshake protocol, *Advances in Cryptology-Proc*, vol. 5350, pp. 55-73, 2008.

[32] I. Goldberg, D. Stebila, and B. Ustaoglu, Anonymity and one-way authentication in key exchange protocols, *Des. Codes Cryptogr*, vol. 67, pp. 245269, 2013.

[33] T. Dierks and A. Christopher, The TLS protocol version 1.0, http://www.ietf.org/rfc/rfc2246.txt. RFC 2246.

[34] NIST National Institute of Standards and Technology, Special Publication 800-56B, Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography, http://csrc.nist.gov/publications/PubsSPs.html.

[35] X.Y. Wang, J.F. Zhao, An improved key agreement protocol based on chaos, *Communications in Nonlinear Science and Numerical Simulation*, vol.15, no. 12, pp.40524057, 2010.

[36] L.Zhang, Cryptanalysis of the public key encryption based on multiple chaotic systems, *Chaos Solitons Fractals,* vol. 37, no. 3, pp. 669674, 2008.

[37] A. Pfitzmann, M. Hansen, A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management, http://dud.inf.tu-dresden.de/Anon_ Terminology.shtml.

[38] R.S. Pippal, C.D. Jaidhar, and S. Tapaswi, Robust Smart Card Authentication Scheme for Multi-server Architecture, *Wireless Personal Communications*, vol. 72, no. 1, pp. 729745, 2013.

[39] E.Y. Yoon, K.Y. Yoo. Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem, *J Supercomput*, vol. 63, pp. 235255, 2013.

[40] T.Y. Chen, C.C. Lee, M.S. Hwang and J.K. Jan, Towards secure and efficient user authentication scheme using smart card for multi-server environments, *J Supercomput*, vol. 66, pp. 10081032, 2013.

[41] J.L. Tsai, Efficient multi-server authentication scheme based on one-way hash function without verification table, *Computers & Security*, vol. 27, no. 3-4, pp.115121, 2008.

[42] Y.P. Liao, S.S. Wang, A secure dynamic ID based remote user authentication scheme for multiserver environment, *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 2429, 2009.

[43] H.C. Hsiang, W.K. Shih, Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment, *Computer Standards & Interfaces*, vol. 31, no. 6, pp.11181123, 2009.

[44] L. Kocarev, S. Lian,*Chaos-Based Cryptography: Theory, Algorithms and Applications*, Springer, Berlin-Heidelberg, 2011.

[45] J. Kar, B. Majhi, An Efficient Password Security of Three Party Key Exchange Protocol based on ECDLP, *The 12th International Conference on Information Technology*, pp. 75-78, 2009.

[46] W. Diffie and M. E. Hellman, New directions in cryptography, *IEEE Trans. on Information Theory*, vol. 22, no.6, pp. 644-654, 1976.

[47] L verlier, P Syverson, *Improving Efficiency and Simplicity of Tor Circuit Establishment and Hidden Services*, LNCS4776, Springer, pp. 134-152, 2007.

[48] L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone, An efficient protocol for authenticated key agreement, *Designs, Codes and Cryptography*, vol. 28, no. 2, pp. 119-134, 2003.

[49] S. Blake-Wilson, D. Johnson, and A. Menezes, *Key agreement protocols and their security analysis*, LNCS1355, Springer, pp.30-45, 2005.

[50] D. M'Rahi, D. Naccache, Batch exponentiation: a fast DLP-based signature generation strategy, *Proc. of the 3rd ACM conference on Computer and communications security*, pp. 58-61, 1996.

7. Appendix A. Security proof of the proposed scheme.

(1)One-way authentication and key agreement

Definition 7.1. *One-way authentication and key agreement refers to only one party authenticating the other suitably and getting the session key simultaneously.*

Theorem 7.1. *The proposed protocol can achieve one-way authentication and key agreement.*

Proof: In our proposed protocol, one-way authentication means that RC helps Alice (an anonymous user) to authenticate S_i.So we can divide the one-way authentication process into three steps:
(1) Alice authenticates RC:Because only RC has the secret k, RC can computes $K_{RC-A} = T_kT_a(x)$ which equals to $K_{A-RC} = T_aT_k(x)$.So if Alice decrypts C_4 to get the necessary information and check if $H^{'}_{RC} = H_{RC}$.If above equation is equal, then that means Alice authenticates RC.
(2)RC and S_i authenticate each other: We can use the shared key R to achieve the task. Firstly, based on ID_{S_i}, RC can compute $R^{'} = H(ID_{S_i}||k)$ by its private key k.Then RC computes $C^{'}_2 = H(ID_{S_i}||m_1||R^{'}||T_{r_i}(x))$ and checks if $C^{'}_2 = C_2$.If above equation is equal, then that means RC authenticates S_i.After receiving the messages $\{ID_{RC}, C_3\}$, S_i computes $C^{'}_3 = H(ID_{RC}||ID_{S_i}||m_1||R||T_{r_i}(x))$ and chesks if $C^{'}_3 = C_3$.If holds, we can say S_i authenticates RC. (3) Alice authenticates S_i:If Alice already authenticates RC,then she can authenticate S_i based on the information $ID_{RC}||ID_{S_i}||m_1||T_{r_i}(x)||H_{RC}$ which were decrypted by RC in C_4.The trust flow is Alice $\rightarrow RC \rightarrow S_i$.
As for the key agreement, after authenticating each other, the temporary $T_a(x), T_{r_i}(x)$ and the $SID_A||ID_{S_i}||ID_{RC}$ were already authenticated by RC.So finally Alice and S_i can make the key agreement simultaneously.

(2) Impersonation attack

Definition 7.2. *An impersonation attack is an attack in which an adversary successfully assumes the identity of one of the legitimate parties in a system or in a communications protocol.*

Theorem 7.2. *The proposed protocol can resist impersonation attack.*

Proof: An adversary cannot impersonate anyone of the S_i and RC.The proposed scheme has already authenticated each other between S_i and RC,and Alice authenticates S_i and RC

(in section Appendix A.(1)) based on the secrets k, R and the nonces a, r_i.So there is no way for an adversary to have a chance to carry out impersonation attack.

Remark 7.1. *Because Alice is an anonymous user, an adversary impersonates Alice is meaningless for the S_i and RC.*

(3) Man-in-the-middle attack

Definition 7.3. *The man-in-the-middle attack is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker.*

Theorem 7.3. *The proposed protocol can resist Man-in-the-middle attack.*

Proof: Because $C_i(1 \leq i \leq 4)$ contain the participants identities or an anonymous users temporary session ID, a man-in-the-middle attack cannot succeed.

(4) Replay attack

Definition 7.4. *A replay attack is a form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed.*

Theorem 7.4. *The proposed protocol can resist replay attack.*

Proof: If an adversary replays any message of Alice, which is meaningless. Because "Alice" is an anonymous user, the adversary can as an anonymous user to initiate the protocol legally as his wish. For the messages between S_i and RC,an adversary cannot start a replay attack against our scheme because of the freshness of a, r_i in each session. If $T_a(x)$ and $T_{r_i}(x)$ have appeared before or the status shows in process, any of the participants in instance protocol will reject the session request. If the adversary wants to launch the replay attack successfully, it must compute and modify $T_a(x)$, $T_{r_i}(x)$ and $C_i(1 \leq i \leq 4)$ correctly which is impossible.

(5) Known-key security

Definition 7.5. *Known-key security is that a protocol can protect the subsequent session keys from disclosing even if the previous session keys are revealed by the intendant user.*

Theorem 7.5. *The proposed protocol can achieve known-key security.*

Proof: Since the session key $SK = T_aT_{r_i}(x) = T_{r_i}T_a(x)$ is depended on the random nonces a and r_i,and the generation of nonces is independent in all sessions, an adversary cannot compute the previous and the future session keys when the adversary knows one session key. And in the secrets update phase, any session key is only used once, so it has known-key security attribute.

(6) Perfect forward secrecy

Definition 7.6. *An authenticated multiple key establishment protocol provides perfect forward secrecy if the compromise of both of the nodes secret keys cannot results in the compromise of previously established session keys [45].*

Theorem 7.6. *The proposed protocol can achieve perfect forward secrecy.*

Proof: In the proposed scheme, the session key $SK = T_aT_{r_i}(x) = T_{r_i}T_a(x)$ is related with a and r_i,which were randomly chosen by Alice and the server S_i,respectively. So any session key has not related with the secret key (such as k)of each of participants. Furthermore because of the intractability of the CMBDLP and CMBDHP problem, an adversary cannot compute the previously established session keys.

(7) Session key security

Definition 7.7. *A communication protocol exhibits session key security if the session key cannot be obtained without any long-term secrets.*

Theorem 7.7. *The proposed protocol can achieve session key security.*

Proof: In the authenticated key agreement phase, a session key SK is generated from a and r_i.These parameter values are different in each session, and each of them is only known by Alice and S_i.Whenever the communication ends between S_i and Alice, the key will immediately self-destruct and will not be reused. Therefore, assuming the attacker has obtained a session key, Alice will be unable to use this session key to decode the information in other communication processes. Because the random point elements a and r_i are all generated randomly and are protected by the CMBDLP, CMBDHP, and the secure symmetric encryption, a known session key cannot be used to calculate the value of the next session key. Additionally, since the values a and r_i of the random elements are very large, attackers cannot directly guess the values a and r_i of the random elements to generate session key. Therefore, the proposed scheme provides session key security.

(8) Resistance to stolen-verifier attacks

Definition 7.8. *An adversary gets the verifier table from servers or RC by a hacking way, and then the adversary can launch any other attack which called stolen-verifier attacks.*

Theorem 7.8. *The proposed protocol can resistance to stolen-verifier attacks.*

Proof: In the proposed scheme, neither the server nor the registration center maintains any verification table. Thus, the stolen-verifier attack is impossible to initiate in the proposed scheme.