Journal of Information Hiding and Multimedia Signal Processing ©2017 ISSN 2073-4212
Ubiquitous International Volume **8**, Number **6**, November **2017**

Enhancing The Security of Chaotic Maps-based Password-Authenticated Key agreement Using Smart Card

Na Lin

School of Computer
Shenyang Aerospace University
Daoyi South Street, No.37, Daoyi Economic Development Zone, Shenyang, P.C 110136 - China
lin_na2000@163.com

*Hong-feng Zhu

Software College
Shenyang Normal University
No.253, HuangHe Bei Street, HuangGu District, Shenyang, P.C 110034 - China
*Corresponding Author: zhuhongfeng1978@163.com

Received April, 2017; revised June, 2017

Abstract. *The security of key exchange has always been the focus of researchers to explore, which devoted to creating a common session key between two-party communication. Recently, Guo and Chang proposed a novel password-authentication key agreement (PAKA) using smart card, this protocol utilizes chaotic maps instead of modular exponentiation and scalar multiplication operation. However, Lin pointed out that the protocol proposed by Guo and Chang failed to protect user identity and an adversary enable obtain the session key between the user and the server using their previous messages. Hence Lin revised the former protocol, and proposed an improved PAKA protocol based on chaotic maps. Meanwhile, Lin declared that the scheme enable implement full protection for users identity and establish a secure common session key between the user and the server. Unfortunately, after detailed analysis we find that Lins protocol fails to resist key-compromise impersonation (KCI) attack and denial-of-service attack (DoS). Therefore, in this paper, we propose a secure chaotic maps-based password-authenticated key agreement using smart card. Meanwhile, our protocol promises to provide identity protection and enable eliminate the weaknesses which appeared in previous schemes.*
Keywords: Chaotic maps; Authentication; Key agreement; Smart card

1. **Introduction.** An authenticated key agreement protocol aiming to implement mutual authentication among the communication parties, and establishes a secure common session key for information exchange. However, both the mutual authentication and the common session key are bulit in open communication channel. In 1981, Lamport [1] first proposed a password-based authentication with insecure communication, which open the model of password authentication. However, this protocol has a serious drawback that the server needs to maintain the password table for user authentication. In order to construct a secure key agreement protocol, various password-based authentication have been proposed. In 1991, Wu and Chang proposed a novel password authentication protocol based on discrete logarithm algorithm. In 2004, Juang proposed a password authentication key

agreement in multi-server environment. In 2007, Byuna et al. proposed a client-to-client password-authenticated key agreement.

Considering the smart card owns the nature of messages encryption, convenient protability and low computational cost. Recently, lots of user authentication agreements based on smart card have been proposed [2, 3]. In the initial study of password-authenticated key agreement schemes, most of them developed are mainly based on bilinear pairing algorithm. Many researchers have discovered the connection between the chaos theory and cryptography [4, 5, 6], hence they combined the significant properties of chaos with the cryptography and then construct a new encryption method to achieved secure communication. In 2009, Tseng et al. [7] proposed a novel password-authenticated key agreement based on chaotic maps, they claimed that their scheme not only achieve mutual authentication without verification table, but also keep the user anonymity property. However, in 2011, Niu and Wang showed that Tseng et al.s scheme failed to provide user anonymity and cannot achieve forward secrecy and then proposed an anonymous key agreement protocol based on chaotic maps [8] with a trusted third party. In 2012, Xue and Hong [9] found that the scheme of Niu and Wang has several drawbacks, so they presented an improved protocol without the third party. Meanwhile, in the same year, Yoon also demonstrated that Niu and Wangs scheme unable to resist Denial of Service (DoS) attack and it also has calculation efficiency problem [10]. Recently, in 2013, Guo and Chang [11] proposed a novel password-based authenticated key agreement using smart card, which utilized Chebyshev chaotic maps instead of the traditional modular exponentiation and scalar multiplication to improve the operation efficiency. They have promised that their scheme enable resist insider attack, reply attack and implement the fundamental security requirements. Unfortunately, in 2015, Yau and Phan [12] pointed out that Guo and Changs scheme failed to achieve Key-compromise impersonation (KCI) attack and parallel session attack, and it have two weaknesses in password change phase. Later, in the same year, Lin [13] also showed that the protocol of Guo and Chang cannot protect users identity because of its fixed-parameters, and according to the Chebyshev chaotic maps an outsider enable obtain the secret session key. Then Lin put forward an improved protocol to correct these weaknesses. Unfortunately, in this paper, we demonstrate that Lins protocol fails to resist key-compromise impersonation (KCI) attack and denial-of-service attack (Dos). So, in order to eliminate above weaknesses, we propose an enhanced chaotic maps-based password-authenticated key agreement, our protocol also takes advantage of smart card and promise to improve security requirements.

The rest of the paper is organized as follows: Chebyshev chaotic maps are given in Section 2. Next, we review Lins protocol and point out their vulnerabilities in Section 3. Then, a secure and efficient chaotic maps-based password-authenticated key exchange scheme using smart card is described in Section 4. In Section 5, we give the security analysis of our proposed protocol. In Section 6, we display the efficiency analysis in this section. Finally, this paper is concluded in Section 7.

2. Chebyshev Chaotic Maps.

Zhang [14] proved that semi-group property holds for Chebyshev polynomials defined on interval $(-\infty, +\infty)$. The enhanced Chebyshev polynomials are used in the proposed protocol:

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

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

$$T_{rs}(x) = T_r(T_s(x)) = T_s(T_r(x)).$$

Definition 2.1. *(Enhanced Chebyshev polynomials) The enhanced Chebyshev maps of degree* $n(n \in N)$*are defined as:* $T_n(x) = (2xT_{n-1}(x) - T_{n-2}(x))(\mathrm{mod}p)$*, where* $n \geq 2$*,*$x \in (-\infty, +\infty)$*, and* p *is a large prime number. Obviously,* $T_{rs}(x) = T_r(T_s(x)) = T_s(T_r(x))$*.*

Definition 2.2. *(DLP, Discrete Logarithm Problem) Given an integer* a*,find the integer* r*,such that* $T_r(x) = a$*.*

Definition 2.3. *(CDH, Computational DiffieHellman Problem) Given an integer* x*,and the values of* $T_r(x), T_s(x)$*,what is the value of* $T_{rs}(x) =?$*.*

It is widely believed that there is no polynomial time to solve DLP, CDH with a non-negligible probability.

3. **Review of Lins Scheme.** In 2015, Lin pointed out two security weaknesses of Guo-Changs scheme, which fails to user identity protection and an adversary enable obtain the session key based on the previous messages. Therefore, in order to eliminate the above weaknesses, Lin proposed an improved smart card-based password-authenticated key agreement (PAKA) using Chebyshev chaotic maps algorithm.

3.1. **Parameter generation phase.** First,S chooses a random number x to compute $T_r(x)$ which the number r as private key. The parameters $(x, T_r(x))$ are stored in smart card instead of being released. Then the server selects a random number mk as the master key, chooses $h(\cdot)$ and symmetric encryption $E_k(x)$ and decryption $D_k(x)$ with symmetric key k.

3.2. **Registration phase.** Step 1: User U_i chooses an identity ID_i, password PW_i and a random number t, computes $H = h(PW_i||t)$. Then,U_i sends $\{ID_i, H = h(PW_i||t)\}$ to the server S.

Step 2: After receiving $\{ID_i, H = h(PW_i||t)\}$ from U_i, server S computes $R = E_{mk}(ID_i||H)$ with mk and $D = H \oplus (x||T_r(x))$ .Stores $(R, h(\cdot), E_k(\cdot), D_k(\cdot), D)$ into the smart card. Then, S sends the smart card to U_i.

Step 3: After receiving the smart card from S, user U_i puts the random number t into this smart card.

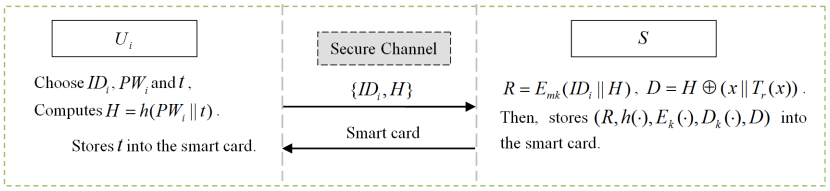

FIGURE 1. Registration Phase of Lins protocol

3.3. **Authentication phase.** In this section, U_i and S achieve mutual authentication, the detailed processes are shown in Fig.2.

Step 1: U_i inputs identity ID_i and password PW_i, the smart card computes $(x||T_r(x)) = H \oplus D$. Then, chooses a random number j computes $m = T_j T_r(x)$,$Q = h(ID_i||H)$,$Z = E_m(Q||R||T_1)$ where T_1 is current timestamp. Then U_i sends $\{R, T_j(x), Z\}$ to S.

Step 2: After receiving $\{R, T_j(x), Z\}$ from U_i,S computes $m = T_r T_j(x)$. Decrypts Z by $\{Q, R, T_1\} = D_m(Z)$,checks if the timestamp T_1 is valid. If holds,S decrypts received R with the mk to obtain the(ID_i', H') , and compares whether $h(ID_i'||H') \stackrel{?}{=} Q$.If holds,the server S chooses a random number s and computes $T_s(x)$,$N = h(ID_i||T_2)$ and

$W = E_m(T_s(x)||N||T_2)$ where T_2 is the current timestamp. Finally,S sends $\{W\}$ to U_i. Meanwhile, S computes the common session key $SK = T_sT_j(x)$ with U_i.

Step 3: After receiving $\{W\}$ from S, the smart card computes $D_m(W) = \{T_s(x), N, T_2\}$. Checks whether the timestamp T_2 is acceptable, and computes $N' = h(ID_i||T_2)$. Compares whether $N' \stackrel{?}{=} N$, if holds, U_i computes the common session key $SK = T_jT_s(x)$ with S.

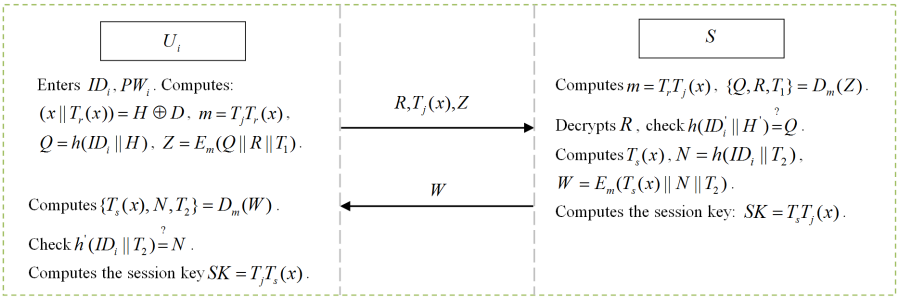

FIGURE 2. Authentication Phase of Lins protocol

3.4. **Password change phase.** If U_i intends to update the password, he/she should proceed as Fig.3.

Step 1: U_i inputs original and new password (PW_i, PW_i'), the smart card computes $H = h(PW_i||t)$ and $(x||T_r(x)) = H \oplus D$ with the old password. Then, selects a random number j', computes $m' = T_{j'}T_r(x)$,$(x||T_r(x)) = H \oplus D$ and $Z' = E_{m'}(H||H'||R)$. Finally,U_i sends the message $\{R, T_{j'}(x), Z'\}$ to S.

Step 2: After receiving $\{R, T_{j'}(x), Z'\}$,S computes $m' = T_rT_{j'}(x)$, decrypts Z' as $\{H, H', R\} = D_{m'}(Z')$ and decrypts R with s as $\{ID_i, H\} = D_{mk}(R)$. Checks whether $H' \stackrel{?}{=} H$, if holds,S computes $R' = E_{mk}(ID_i||H')$. Finally,S sends the message $\{R'\}$ to U_i.

Step 3: U_i replace R with R'.

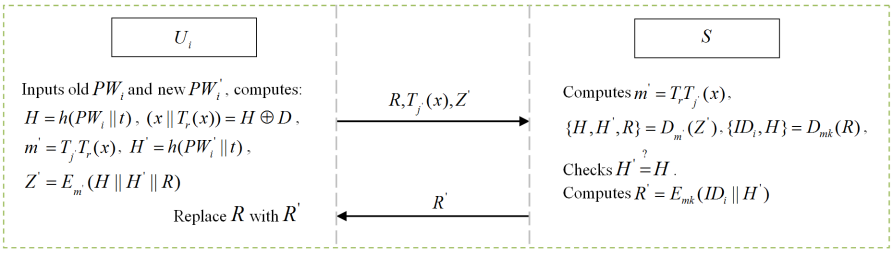

FIGURE 3. Password Change Phase of Lins protocol

3.5. **Security analysis of Lins protocol.** Lin claimed that this protocol has achieved secure communication between the user and the server, and has provided a perfect method of establishing the common session key. In fact, there are several loopholes in authentication and password change phase. Firstly, Lins scheme fails to resist key-compromise impersonation (KCI) attack. Secondly, the user cannot update the password successfully in password change phase and easily to suffer denial-of-service (DoS) attack. In this part, we make a detailed discussion of these securiy problems.

3.5.1. *Key-compromise impersonation attack (KCI).*

Definition 3.1. *If the compromised of a participants long-term key cannot lead to an adversary impersonate other legitimate participants, we refer to this agreement provide key-compromise impersonation resilience.*

Suppose that there are two parties A and B, the relationship between the two participants is user-to-user or user-to-server. If As long-term key or secret is compromised, but it not allow the adversary impersonate the honest participant B to establish the common session key with A, so we believe that the protocol achieves KCI resilience. In fact, researchers have drawn attention to study KCI resilience because of the affect of this attack.

Proof: If the server have compromised the long-term secret key s, then an adversary enable impersonate the user with the help of these derived messages. Next, we give the detailed steps of the KCI attack:

Step 1: In the authentication phase, an adversary A eavesdropping the communication between the user U_i and the server S, and then A could obtain the message $\{R, T_j(x), Z\}$ from U_i to S.

Step 2: After obtain the value R, adversary A decrypts R and retrieve $\{ID_i, h(PW_i||t)\}$ based on the servers compromised long-term key s.

Step 3: Now,adversary A enable impersonate U_i to establish a new session key to S as follows:

(a). Adversary A computes $T_{j^*}(x)$,$m^* = T_{j^*}T_r(x)$,
$Z^* = E_{m^*}(Q||R||T_1) = E_{m^*}(h(ID_i||H)||R||T_1^*) = E_{m^*}(h(ID_i||(PW_i||t))||R||T_1^*)$
where T_1^* is a timestamp, then A sends $\{R, T_{j^*}(x), Z^*\}$ to S.

(b). After receiving $\{R, T_{j^*}(x), Z^*\}$,S computes $m^* = T_rT_{j^*}(x)$and decrypts the value Z^* to obtain $\{h(ID_i||(PW_i||t))||R||T_1^*\}$. The timestamps T_1^* is valid, so S decrypts R to obtain $\{ID_i, h(PW_i||t)\}$ and verify whether $h(ID_i||h(PW_i||t)) \stackrel{?}{=} Q$. If holds, it means S authenticates A as the legitimate user U_i.
Then,S computes $T_s(x)$,$W = E_{m^*}(T_s(x)||N||T_2) = E_{m^*}(T_s(x)||h(ID_i||T_2)||T_2)$. Finally,$S$ sends $\{W\}$ to A who the server thinks as U_i. Meanwhile,S computes the common session key $SK = T_sT_{j^*}(x)$.

(c). After receiving $\{W\}$,A decrypts W with m^* to obtain $\{T_s(x), h(ID_i||T_2), T_2\}$. It means that A impersonates the legitimate user U_i and achieves authentication phase. Finally,A computes the common session key $SK = T_{j^*}T_s(x)$.

3.5.2. *Denial-of-service attack (DoS).*

Definition 3.2. *Denial-of-service attack is one of the common attacks in cryptography, which disallow legitimate users obtain the provided service from the other side, thereby affecting the connection with users.*

Proof: According to our detailed analysis, we find that Lins protocol is easy to suffer denial-of-service attack, which the legitimate user cannot login to the server successfully. Suppose that an adversary A change the message R' to R^*. Consequently, the smart card replace R with R^*. However, when U_i intends to establish the connection to S,S will reject the request. Therefore, the user has to register to the server again, which increase the computational cost and running time. Then, we show the detailed steps as follow:

Step 1: U_i inputs the password PW_i', computes $H' = h(PW_i'||t)$ and $Q' = h(ID_i||H')$. The smart card computes $T_{j^*}(x)$,$m^* = T_{j^*}T_r(x)$ and $Z^* = E_{m^*}(Q'||R^*||T_1^*)$. Then, U_i sends $\{R^*, T_{j^*}(x), Z^*\}$ to S.

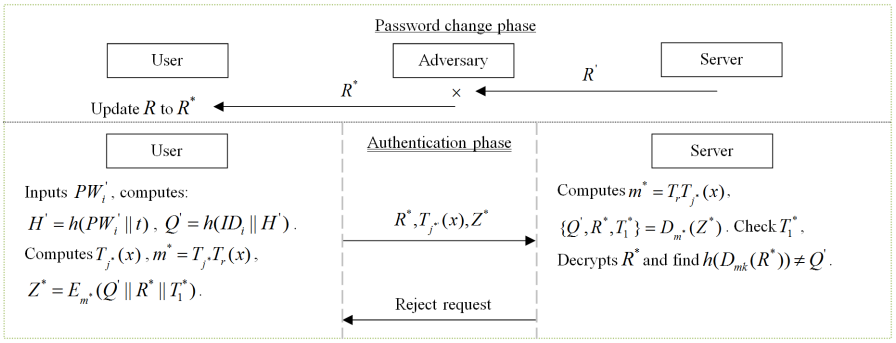

FIGURE 4. Denial-of-service attack on Lins scheme

Step 2: After receiving $\{R^*, T_{j^*}(x), Z^*\}$, S computes $m^* = T_r T_{j^*}(x)$ and decrypts Z^* with m^* to obtain the value $\{Q', R^*, T_1^*\}$,S checks the validation of the timestamp T_1^*.Then,S decrypts R^* with mk to compute $Q^* = h(D_{mk}(R^*))$ and finds $Q^* \neq Q'$. It is worth noting that whatever R^* is a random text or previous information, the decrypted value cannot equal to the current value Q'. Therefore,S rejects this request.

4. **The Proposed Scheme.** In the processing of analysis, we find that Lins scheme fails to resist key-compromised impersonation attack and denial-of-service attack. For the purpose of eliminate these weaknesses, in this section, we propose an enhanced password-authenticated key agreement based on chaotic maps, our protocol also use the smart card to achieve messages storage and computation. The proposed protocol consists of three phase: the registration phase, the authentication phase and the password change phase.

The user i: U_i; The server: S; User $i's$ identity: ID_i; User $i's$ password: PW_i; $E_k(\cdot)/D_k(\cdot)$ means that symmetric encryption/decryption with the secret keyb k; $h(\cdot)$ represents one-way hash function; T_1, T_2 are current timestamp respectively, while SK is the common session key between the user and server.

4.1. **Registration phase.** In this phase, the user register to the server is confidential, which means that the messages delivered between U_i and S is in a secure channel, the detail processes are shown in Fig.5.

Step 1: U_i chooses an identity ID_i, the password PW_i, a random number t and computes $K = h(PW_i||t)$. Then, U_i sends the message $\{ID_i, K = h(PW_i||t)\}$ to S.

Step 2: After receiving $\{ID_i, K\}$,S computes $R = E_s(ID_i||K)$ with the private master key s.S computes $M = K \oplus R$ and $N = R \oplus (x||T_r(x))$. Then,S sends the smart card $\{(M, N, h(\cdot), E_k(\cdot), D_k(\cdot))\}$ to U_i.

Step 3: After receiving the smart card, U_i inputs t into the smart card, so $SC = \{(M, N, h(\cdot), E_k(\cdot), D_k(\cdot), t)\}$.

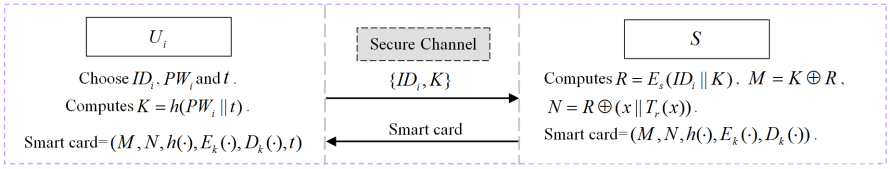

FIGURE 5. Registration Phase of our protocol

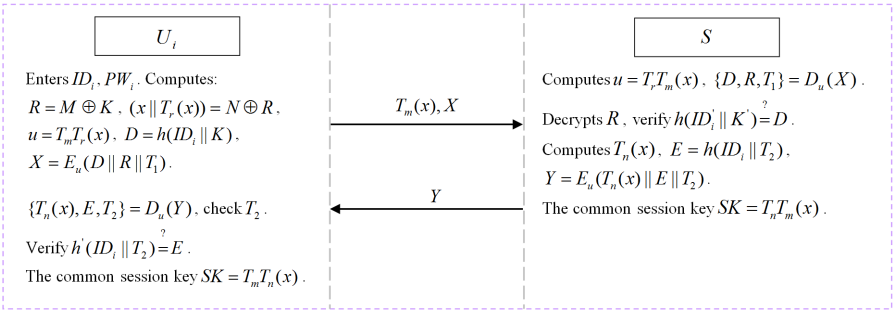

FIGURE 6. Authentication Phase of our protocol

4.2. **Authentication phase.** A common session key SK is established between U_i and S, the detailed processes are shown in Fig.6.

Step 1:U_i enters his/her ID_i and PW_i, the smart card computes $R = M \oplus K$,$(x||T_r(x)) = N \oplus R$. Next, U_i chooses a random number j and computes $u = T_m T_r(x)$,$D = h(ID_i||K)$,$X = E_u(D||R||T_1)$ where T_1 is current timestamp. Then, U_i sends the message $\{T_m(x), X\}$ to S.

Step 2: After receiving $\{T_m(x), X\}$,S computes $u = T_r T_m(x)$. Decrypts Z by $\{D, R, T_1\} = D_u(X)$, checks the timestamp T_1. If holds,S decrypts R with the master key s to obtain (ID_i', K'), checks if $h(ID_i'||K') \stackrel{?}{=} D$. If holds,S chooses n and computes $T_n(x)$,$E = h(ID_i||T_2)$ and $Y = E_u(T_n(x)||E||T_2)$, where T_2 is the current timestamp. Finally,S sends $\{Y\}$ to U_i. Meanwhile,Scomputes the session key $SK = T_n T_m(x)$.

Step 3: After receiving $\{Y\}$,U_i computes $D_u(Y) = \{T_n(x), E, T_2\}$. Checks whether the timestamp T_2 is acceptable, verify whether $h'(ID_i||T_2) \stackrel{?}{=} E$, if holds, U_i computes the common session key $SK = T_m T_n(x)$.

4.3. **Password change phase.** Password change phase intends to help the legitimate user to update his/her password. The detailed processes are shown as Fig.7.

Step 1: U_i enters his/her original password and new password (PW, PW'), the smart card computes $K = h(PW_i||t)$. Then, the smart card selects a random number m', computes $u' = T_{m'} T_r(x)$,$K' = h(PW_i'||t)$ and $X' = E_{u'}(K||K'||R)$. Finally, U_i sends the message $\{T_{m'}(x), X'\}$ to S.

Step 2: After receiving $\{T_{m'}(x), X'\}$,S computes $u' = T_r T_{m'}(x)$, decrypts X' with u' as $\{K, K', R\} = D_{u'}(X')$ and decrypts R with its master key s as $\{ID_i, K\} = D_s(R)$. Verify whether $K' \stackrel{?}{=} K$, if holds,S computes $R' = E_s(ID_i||K')$ and $M' = K' \oplus R'$. Then,S sends the message $\{M'\}$ to U_i.

Step 3: After receiving $\{M'\}$ from S, the user U_i computes $R' = M' \oplus K'$, and then replace R with R'.

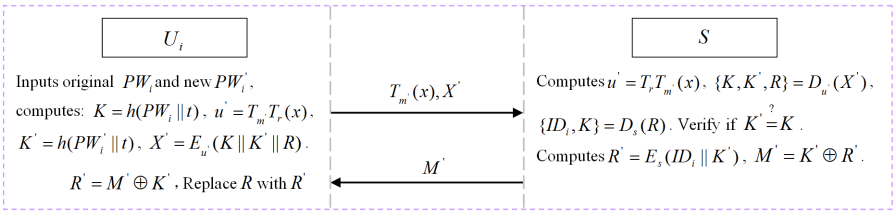

FIGURE 7. Password Change Phase of our protocol

5. **Security Consideration.** In this paper, we propose an improved password-authenticated key agreement using smart card based on chaotic maps. From analyzed Lins protocol, we found that it fails to resist key-compromise impersonation (KCI) attack and denial-of-service (DoS) attack. Therefore, we propose an improved protocol to eliminate above weaknesses. Then, we give a detailed security analysis of our scheme.

5.1. **Key-compromise impersonation attack.**

Definition 5.1. *In the user-to-server model, if servers long-term key is compromised, any adversary cannot impersonate the legitimate user to establish a session key with the compromised server.*

Proof: Suppose that an adversary A eavesdrops the communication between the user U_i and the server S during the authentication phase. Then A obtains $\{T_m(x), X\}$ from U_i to S.In our protocol, the value $(x, T_r(x))$ is protect by U_i and S, the third part cannot obtain the secret values. So A cannot compute $u = T_m T_r(x)$ and $X = E_u(D||R||T_1)$. Therefore, even A owns servers long-term key,A still cannot launch key-compromise impersonation attack. Considering that our protocol based on the smart card, so A could steal the smart card to obtain the message and launch KCI attack. Unfortunately, although A steal the smart card from U_i and obtain $\{M, N, h(\cdot), E_k(\cdot), D_k(\cdot), t\}$,$M \oplus N = K \oplus R \oplus R \oplus (x||T_r(x))$. However, the message K and $(x||T_r(x))$ are not open in our protocol. Hence,A cannot get the message R under this computation. In other words,A fails to launch key-compromise impersonation attack.

5.2. **Denial-of-service attack.**

Definition 5.2. *Denial-of-service attack means that the adversary changes the transmitted messages between the user and the server in the password change phase, aiming to disrupt the communication between them.*

Proof: Suppose that an adversary A eavesdrops the communication between the user U_i and the server S, obtain the message $\{T_{m'}(x), X'\}$ and $\{M'\}$. Then,A changes $\{M'\}$ to $\{M^*\}$ and sends it to U_i. After receiving $\{M^*\}$ from S, U_i intends to retrieve R' from M^* with K'. However, U_i cannot retrieve R' because of the wrong message from S. So, U_i sends a reject response to S in the password change phase. It means that U_i fails to update the password. Therefore, we believe that the adversary A unable launch the Denial-of-service attack successfully. The concrete process can be found in Fig.8.

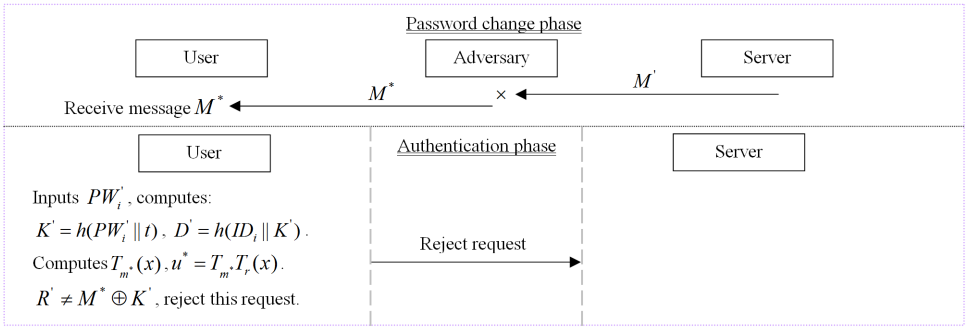

FIGURE 8. Resist to Denial-of-service attack

5.3. **Server impersonation attack.**

Definition 5.3. *Server impersonation attack means that a forge server could be identified by the legal user, and this attack could threat the security of protocol.*

Proof: An authorized server holds the message $\{(M, N, h(\cdot), E_k(\cdot), D_k(\cdot))\}$ and secret value $(x||T_r(x))$. Suppose an adversary A intends to impersonate a server to launch a server impersonation attack,A intercepts $\{T_m(x), X\}$ in the authentication phase. Then,A computes $u' = T_{r'}T_m(x)$ with $T_{r'}(x)$, decrypt X with u'. However,A cannot derive the value from X due to $u' \neq u$. Meanwhile, based on Chebyshev chaotic maps hard problem,A unable guess the same value as r. Therefore, we believe that an attacker cannot launch server impersonation attack.

TABLE 1. Comparisons between our proposed scheme other literatures

	Identity Protection	Known-key Secrecy	Mutual Authentication	Impersonation Attack	KCI Attack	DoS Attack
Our protocol	Yes	Yes	Yes	Yes	Yes	Yes
Lin [13](2015)	No	Yes	Yes	Yes	No	No
Guo et al. [11](2013)	No	Yes	Yes	No	No	No
Li et al. [4](2010)	No	No	Yes	No	No	Yes

6. **Efficiency Analysis.** Chaos refers to the seemingly random irregular movement in a deterministic system, aiming to reveal the simple rules behind it. Chaos system has its unique characteristics, such as uncertainly, boundness, ergodicity and unpredictability, etc. In chaos theory systems, chaotic maps-based Chebyshev polynomials algorithm has been widely used in the secure communication, which as a category of single chaotic public-key algorithm. Chaotic maps encryption algorithm based on two difficult problemsDLP and CDH, and owns the unique semi-group nature to achieve user encryption. Compared to the RSA and ECC encryption algorithm, chaotic maps encryption algorithm avoids scalar multiplication and modular exponentiation computation, offers faster computation, smaller key sizes as well as memory, energy and bandwidth savings.

In this section, we make a comparison among Lins, Guo et al.s and our protocol. From the analysis, we can find that although the efficiency of our protocol is not most perfect, our improved scheme eliminates the weaknesses of other protocols. Therefore, compared to improve the efficiency, our protocol focus on solve these loopholes.The detailed efficiency analysis is shown in Table 2 as follows.

7. **Conclusion.** In this paper, we propose an improved password-authenticated key agreement using smart card. Our protocol based on Chebyshev chaotic maps, utilizes two hard problems and semi-group property to achieve secure communication. We analyzed previous schemes and found that these protocols cannot resist malicious attacks from the third party. Therefore, we propose a secure protocol to help establish a common session key between the user and the server. Our protocol not only provide mutual authentication in user-to-server model, , but also withstand a series of attacks. Compared with other related protocols, the application prospect of our protocol is considerable.

TABLE 2. Computation cost between our protocol and others

	C1	C2	C3	C4	C5	C6	Total
Our scheme	1H	1T+1S+2X	2T+2H +2S+2X	3T+2H+3S	1T+2H +1S+1X	1T+3S+1X	8T+7H +10S+6X
Lin's	1H	1T+1S+1X	2T+2H +2S+1X	3T+2H+3S	1T+2H +1S+1X	1T+3S	8T+7H +10S+3X
Guo's	1H	1T+1S	2T+2H+2S	3T+2H+3S	1T+2H+1S	1T+3S	8T+7H+10S

C1: user in registration phase; C2:server in registration phase; C3: user in authentication phase;
C4: server in authentication phase; C5: user in password change phase;
C6: server in password change phase.H: Time for Hash; X: Time for XOR; S: Time for symmetric encryption/decryption algorithm; T: Time for executing $T_n(x) \mathrm{mod} p$.

Acknowledgment. This work is supported by the Liaoning Natural Science Foundation of China (Grant No. 2015020008) and Liaoning Provincial Natural Science Foundation of China (Grant No. 201602680).

REFERENCES

[1] L. Lamport, A. K. Jones,Password authentication with insecure communication, *Communications of the Association for Computing Machinery*, vol.24, no.11, pp.770–772, 1981.
[2] J. L. Tsai, N. W. L, A chaotic maps-based anonymous multi-server authenticated key agreement protocol using smart card, *International Journal of Communication systems* , vol.28, no.13, pp.1955–1963, 2015.
[3] M. S. Hwang, L. H. Li, A new remote user authentication scheme using smart cards, *IEEE Transactions on Consumer Electronics*, vol.46, no.1, pp.28–30, 2000.
[4] X. Li, W. Qiu, D. Zheng, K. Chen, J.Li, Anonymity enhancement on robust and efficient password-authenticated key agreement using smart cards, *IEEE Transactions on Industrial Electronics*, vol.57, no.2, pp.793–800, 2010.
[5] C. M. Chen, L. L. Xu, T. Y. Wu, C. R. Lin, On the Security of a Chaotic Maps-based Three-party Authenticated Key Agreement Protocol, *Journal of Network Intelligence*, vol.1, no.2, pp.61–66, 2016.
[6] H. J. Wang, H. Zhang, J. X. Li, C. Xu, A(3,3) visual cryptography scheme for authentication, *Shenyang Normal University (Natural Science Edition)*, vol.31, no.101(03), pp.397–400, 2013.
[7] H. R. Tseng, R. H. Jan, W. Yang, A chaotic maps-based key agreement protocol that preserves user anonymity, *IEEE International Conference on Communications (ICC09)*, Dresden, Germany, pp.1–6, 2009.
[8] Y. Niu, X. Wang, An anonymous key agreement protocol based on chaotic maps, *Communications in Nonlinear Science and Numerical Simulation*, vol.16, no.4, pp.1986-1992, 2011.
[9] K. Xue, P. Hong, Security improvement on an anonymous key agreement protocol based on chaotic maps, *Communications in Nonlinear Science and Numerical Simulation*, vol.17, no.7, pp.2969–2977, 2012.
[10] E. J.Yoon, Efficiency and security problems of anonymous key agreement protocol based on chaotic maps, *Communications in Nonlinear Science and Numerical Simulation*, vol.17, no.7, pp.2735–2740, 2012.
[11] C. Guo, C. C. Chang, Chaotic maps-based password authenticated key agreement using smart cards, *Communications in Nonlinear Science and Numerical Simulation*, vol.18, no.6, pp.1433–1440, 2013.
[12] W. C. Yau, R. C. W. Phan, Cryptanalysis of a chaotic map-based password-authenticated key agreement protocol using smart cards, *Nonlinear Dynamics*, vol.79, no.2, pp.809–821, 2015.
[13] H. Y. Lin, Improved chaotic maps-based password-authenticated key agreement using smart cards, *Communications in Nonlinear Science and Numerical Simulation*, vol.20, no.2, pp.482–488, 2015.
[14] L. Zhang, Cryptanalysis of the public key encryption based on multiple chaotic systems, *Chaos Solitons Fractals*, vol.37, no.3, pp.669–674, 2008.