ICIC Express Letters
Volume **12**, Number **2**, February **2018**
ICIC International ©2018 ISSN 1881-803X
pp. 175–182

THE WEAKNESSES AND LIGHT IMPROVEMENT OF CHOI ET AL.'S ANONYMOUS MULTI-SERVER AUTHENTICATED KEY AGREEMENT SCHEME USING SMART CARDS AND BIOMETRIC

Ping Yu* and Wen-Gong Shieh

Department of Information Management
Chinese Culture University
No. 55, Hwa-Kang Road, Yang-Ming-Shan, Taipei 11114, Taiwan
*Corresponding author: yp@faculty.pccu.edu.tw; wgshieh@faculty.pccu.edu.tw

Received August 2017; accepted November 2017

Abstract. *Due to the rapid development of network applications and the biometric techniques, users can use a single smart card in multi-server communication environment to get the benefit of different services. Choi et al. proposed an anonymity-preserving biometric-based multi-server authentication scheme using smart card with the functions of session key agreement, mutual authentication and forward secrecy. However, we found that Choi et al.'s scheme is vulnerable to some attacks such as offline identity guessing and insider attack. The scheme has mutual authentication problem and fails to maintain forward secrecy. We find, if adding a simple and experimentally feasible modification to the Choi et al.'s scheme, the modified scheme can protect the session key against some collective attacks and achieve perfect forward secrecy. We also show this simple modification of their scheme with better efficiency.*
Keywords: Authentication, Smart card, Biometric, Anonymity, Forgery attack, Insider attack, Forward secrecy

1. **Introduction.** In recent years, remote authentication has been an important issue for the communication applications in Internet. In 2009, Hsiang and Shih [1] proposed a dynamic ID based remote user authentication scheme for multi-server environment. In 2010, Li and Hwang [2] proposed a biometric-based scheme that was based on the biometrics verification. In 2011, Chen et al. [3] proposed attacks to Wan et al.'s scheme [4] and proposed an improvement scheme. Many researchers found that Chen et al.'s scheme is still vulnerable to the offline password-guessing attack such as Kumari et al. [5] and Yu and Shieh [6]. In particular, Choi et al. [7] showed that Chuang-Chen's [8] multi-server authenticated key agreement scheme does not resist some attack and lacked the smart card and session key verification mechanism. Choi et al. also proposed a remote authentication scheme using smart card that is an improvement from Chuang-Chen's scheme. However, we found that Choi et al.'s scheme is still vulnerable to some attacks. First at all, when the attacker registers himself as a legal user and interprets the login and authentication request/response messages, she/he can compute the important common secret value of each smart card and offline guesses the user identity. Therefore, their scheme suffers the anonymous problem. Secondly, if the insider records the registration values of the registered users in the registration center and is also a legal user who has computed the common secret from herself/himself smart card, even the insider has not the smart card, password or biometrics of users, the insider can get the secret information and attack the scheme like identities and session keys. Thirdly, if the attacker is a legal user and performs the offline identity guessing attack successfully, the attacker also can perform the server spoofing attack. Similarly, if the attacker gets the smart card of user, the attacker also can perform the user impersonation attack. Therefore, their scheme

suffers the mutual authentication problem. Finally, Choi et al.'s scheme cannot provide the forward secrecy. The attacker can get the session keys that only need the messages in the common channel without the long-term secret, smart card, password or biometrics of users. Through entire analysis, we find that Choi et al.'s scheme may be not suitable for applications in the network which requires user privacy and security. We also find, if adding a simple and experimentally feasible modification to the Choi et al.'s scheme, the modified scheme can protect the session key against some collective attacks and achieve perfect forward secrecy. We also show this simple modification of their scheme with better efficiency.

The remainder of this paper is presented as follows. In the next section, a brief review of Choi et al.'s scheme is given. After that, we point out the weakness of Choi et al.'s scheme in Section 3. In Section 4, we propose a simple and experimentally feasible modification to the Choi et al.'s scheme. Then, we discuss the security of the modified scheme and show it with better efficiency than Choi et al.'s scheme in Section 5. Finally, we give our conclusion in the last section.

2. Review of Choi et al.'s Scheme.

There are four phases in Choi et al.'s scheme [7]: the registration, login, authentication, and password change phases. The notations used in this paper are summarized in Table 1 and the description of each phase is as the following.

Table 1. The notations used in this paper

Notations	Description	Notations	Description
RC	Registration center	PW_i	Password of U_i
S_j	Server j	BIO_i	Biometrics of U_i
x	A secret value of RC	AID_i	Anonymous identity of U_i
PSK	A secret key of RC and all S	$h(\cdot)$	One-way hash function
U_i	User i	$\oplus$	bitwise XOR operator
U_a	Attacker	$\|\|$	Concatenation operator
SC_i	Smart card of U_i	⟶	a common channel
ID_i	Identity of U_i	- - - - -▶	a secure channel
SID_j	Identity of S_j		

2.1. Registration phase.

In Choi et al.'s scheme, servers and users must register to the registration center RC. Firstly, U_i chooses the identity ID_i, password PW_i and inserts her/his biometric information BIO_i to compute $h(PW_i \oplus BIO_i)$. Then, U_i sends her/his identity ID_i and $h(PW_i \oplus BIO_i)$ to RC for registration via a secure channel. If RC accepts the request, RC computes $A_i = h(ID_i||x)$, $B_i = h(A_i) = h^2(ID_i||x)$, $C_i = h(PW_i \oplus BIO_i) \oplus B_i$, $D_i = PSK \oplus A_i \oplus h(PSK)$ and $E_i = h(PSK) \oplus h(PW_i \oplus BIO_i)$, where x is the secret key of RC and PSK is the secret key of all servers. Finally, RC gives U_i a smart card SC_i containing $\{ID_i, B_i, C_i, D_i, E_i, h(.)\}$ via a secure channel. In the server side, RC uses the same PSK to all the authorized servers and facilitates the user's authentication procedure.

2.2. Login and authentication phase.

When U_i wants to log in to the server S_j, U_i inserts SC_i and inputs ID_i, PW_i and BIO_i with a sensor. The smart card SC_i computes $B_i' = h(PW_i \oplus BIO_i) \oplus C_i$ and checks if B_i' is equivalent to the stored B_i. If yes, SC_i generates a random number N_1 and computes $h(PSK) = E_i \oplus h(PW_i \oplus BIO_i)$, $M_1 = h(B_i) \oplus N_1 \oplus h(PSK)$, $AID_i = h(N_1) \oplus ID_i$ and $M_2 = h(N_1||AID_i||D_i||SID_i||T_i)$. Then, SC_i sends the message $\{AID_i, M_1, M_2, D_i, T_i\}$ to the server S_j, where T_i is the timestamp of U_i.

After receiving the message form U_i, the server S_j retrieves $A_i = D_i \oplus PSK \oplus h(PSK)$, $N_1 = M_1 \oplus h^2(A_i) \oplus h(PSK)$, and checks whether $M_2' = h(N_1||AID_i||D_i||SID_j||T_i)$ is equivalent to the received M_2 and the freshness of T_i. If it fails, S_j rejects U_i's login request. Otherwise, it accepts the request and generates a random number N_2, computes the session key $SK_{ji} = h(N_1||N_2)$, $M_3 = N_2 \oplus h^2(N_1) \oplus h(PSK)$, $M_4 = h(SID_j||N_2||AID_i)$ and sends the message $\{SID_j, M_3, M_4\}$ to U_i. Upon receiving the message from S_j, SC_i retrieves $N_2 = M_3 \oplus h^2(N_1) \oplus h(PSK)$, and checks whether $M_4' = h(SID_j||N_2||AID_i)$ is equivalent to the received M_4. If it fails, U_i terminates this session. Otherwise, SC_i computes the session key $SK_{ij} = h(N_1||N_2)$, $SK_{ij} \oplus h(N_2)$ and sends $SK_{ij} \oplus h(N_2)$ to S_j. After receiving the message, S_j checks whether $SK_{ij} \oplus h(N_2)$ is equivalent to the received value. If it fails, S_j rejects U_i's request. Otherwise, S_j successfully authenticates U_i.

2.3. **Password change phase.** When U_i wants to change the password, U_i inserts SC_i and inputs ID_i, PW_i, BIO_i and new password PW^*. The smart card checks whether the ID_i and $B_i' = h(PW_i \oplus BIO_i) \oplus C_i \oplus h(PSK)$ is equivalent to the stored ID_i and B'. If it fails, SC_i rejects U_i's request. Otherwise, SC_i accepts this request and computes $C_i^* = C_i \oplus h(PW_i \oplus BIO_i) \oplus h(PW_i^* \oplus BIO_i)$ to replace C_i.

3. **Our Attacks to Choi et al.'s Scheme.** In this section, we demonstrate the weaknesses of Choi et al.'s scheme and follow three assumptions regarding capabilities of an attacker as suggested by Kocher et al. [9], Messerges et al. [10] and Huang et al. [11] respectively. Firstly, an attacker has total control over the common channel connecting the users and the remote server in login/authentication phase that the adversary can intercept, insert, delete, or modify any message transmitted via a common channel. Secondly, an attacker may either steal a user's smart card or obtain a user's password, but not both. Thirdly, the adversary attacker can register as legitimate users and take legal smart cards. Those assumptions are similar to the analyzed Chuang-Chen's scheme of Choi et al. From previous assumptions, we analyze the weaknesses existing in Choi et al.'s scheme.

3.1. **The weakness of anonymity and offline identity guessing attack.** We find the Choi et al.'s scheme has a constant value $h(PSK)$ in all smart card. If attacker U_a is a legal user, using the legal registered smart card $SC_a = \{ID_a, B_a, C_a, D_a, E_i, h(.)\}$ with U_a self-choice identity ID_a, password PW_a and biometric information BIO_a. U_a can get the $h(PSK)$ from $h(PSK) = E_a \oplus h(PW_a \oplus BIO_a)$. From the request and response message $\{AID_i, M_1, M_2, D_i, T_i\}$ and $\{SID_j, M_3, M_4\}$ between S_j and U_i, U_a can guess an identity ID_i' and computes $h(N_1)' = AID_i \oplus ID_i'$, $N_2' = M_3 \oplus h(h(N_1)') \oplus h(PSK)$, $M_4' = h(SID_j||N_2'||AID_i)$. If $M_4' \neq M_4$, U_a repeats same steps. If $M_4' = M_4$, it implies $ID_i' = ID_i$, U_i's identity, and U_a gets the random N_2 generated by the server. After getting the N_2, the attacker can compute the session key from the response message, $SK_{ij} \oplus h(N_2)$ from user to server, using $SK_{ij} = SK_{ij} \oplus h(N_2) \oplus h(N_2)$.

3.2. **The insider attack.** Choi et al.'s scheme supposes that the user U_i never sends plain PW_i and BIO_i to the RC which cannot obtain the user's password or biometrics and cannot compute the PW_i using $h(PW_i \oplus BIO_i)$ because the biometric information has high entropy. So, the insider adversary cannot figure out U_i's PW_i and BIO_i. Therefore, the proposed scheme is secure against the insider attack.

However, we find the insider can successfully attack their scheme. First of all, assume that the insider records the registration values of identity ID_i of all registered users and stores the value in a data base DB. The insider also registers herself/himself as a legal user. From previous Section 3.1, she/he can compute the common secret $h(PSK)$ from her/his smart card. Based on these assumptions, we show the insider attack as follows.

Firstly, the insider U_a intercepts and records the request and response message $\{AID_i, M_1, M_2, D_i, T_i\}$, $\{SID_j, M_3, M_4\}$ and $SK_{ij} \oplus h(N_2)$ between S_j and U_i that classifies the users by the constant value D_i. The attacker can use the identity ID_i' in her/his DB and computes $h(N_1)' = AID_i \oplus ID_i'$, $N_2' = M_3 \oplus h(h(N_1)') \oplus h(PSK)$, $M_4' = h(SID_j||N_2'||AID_i')$. If $M_4' \neq M_4$, then U_a repeats with some other ID_i' and so on until getting success. If $M_4' = M_4$, it implies that U_a has successfully guessed U_i's identity $ID_i' = ID_i$ and get the random N_2 generated by the server. After getting the N_2, the attacker can compute the session key $SK_{ij} = SK_{ij} \oplus h(N_2) \oplus h(N_2)$ from the response message from user to server and records the $\{ID_i, SID_j, D_i, h(N_1), N_2, T_i\}$ as a record in her/his DB to another attack. In this attack, the insider also has not the SC_i, PW_i or BIO_i of U_i that only record the identity of user to register and the intercepted message in the common channel. Therefore, an insider can get the secret information and attack the scheme.

3.3. **The server spoofing attack.** Firstly, the attacker U_a gets $h(PSK)$ and $h(N_1)$ from Section 3.1. Secondly, U_a blocks the response message to U_i, generates a random number N_2' and computes the session key $SK_{ji}' = h(N_1||N_2')$, $M_3' = N_2' \oplus h^2(N_1) \oplus h(PSK)$, $M_4' = h(SID_j||N_2'||AID_i)$ and sends the message $\{SID_j, M_3', M_4'\}$ to U_i. Upon receiving the message from U_a, SC_i retrieves $N_2' = M_3' \oplus h^2(N_1) \oplus h(PSK)$, and finds $M_4'' = h(SID_j||N_2'||AID_i)$ that is equivalent to the received M_4'. SC_i will compute the session key $SK_{ji}' = h(N_1||N_2')$, $SK_{ij}' \oplus h(N_2')$ and sends $SK_{ij}' \oplus h(N_2')$ to S_j. After blocking the response message to S_j, U_a successfully performs the server spoofing attack.

3.4. **The smart card lost and user impersonation attack.** Firstly, the attacker U_a gets $h(PSK)$ and $h(N_1)$ as Section 3.1 and obtains ID_i, B_i, and D_i from the smart card SC_i of U_i. Then, U_a generates a random number N_1' and computes $M_1' = h(B_i) \oplus N_1' \oplus h(PSK)$, $AID_i' = h(N_1') \oplus ID_i$ and $M_2' = h(N_1'||AID_i'||D_i||SID_j||T_a)$. Thirdly, U_a sends the message $\{AID_i', M_1', M_2', D_i, T_a\}$ to the server S_j. After receiving the message from U_a, the server S_j can find $M_2'' = h(N_1'||AID_i'||D_i||SID_j||T_a)$ that is equivalent to the received M_2' and the timestamp T_a is fresh. Thus, S_j accepts U_a's request as U_i. Then, S_j generates a random number N_2 and computes the session key $SK_{ji}' = h(N_1'||N_2)$, $M_3' = N_2 \oplus h^2(N_1') \oplus h(PSK)$, $M_4' = h(SID_j||N_2||AID_i')$ and sends the message $\{SID_j, M_3', M_4'\}$ to U_a. Upon receiving the message from S_j, U_a retrieves $N_2 = M_3' \oplus h^2(N_1') \oplus h(PSK)$ and computes the session key $SK_{ji}' = h(N_1'||N_2)$, $SK_{ij}' \oplus h(N_2)$ and sends $SK_{ij}' \oplus h(N_2)$ to S_j. After receiving the message, S_j can find that the computed $SK_{ji} \oplus h(N_2)'$ is equivalent to the received $SK_{ij}' \oplus h(N_2)$ and successfully authenticate U_a.

3.5. **Forward secrecy problem.** Forward secrecy guarantees that a session key derived from a set of long-term keys cannot be compromised if one of the long-term keys is compromised in the future. However, from Section 3.1, the attacker can compute the session key using the validation message from user to server. Hence, Choi et al.'s scheme cannot provide the perfect forward secrecy.

4. **The Simple Modification to the Choi et al.'s Scheme.** In this section, we propose a simple modification to the Choi et al.'s scheme. In our scheme, we also use the timestamp to avoid replay attack of the Chuang-Chen's scheme and use the characteristic of challenge/response to avoid the failure of perfect forward secrecy of the Choi et al.'s scheme. Our scheme also consists of four phases that describes as the following. Our password change phase is similar to that of Choi et al.'s scheme and thus we skip its description. The symbols in our scheme are defined as in the Choi et al.'s scheme in Table 1.

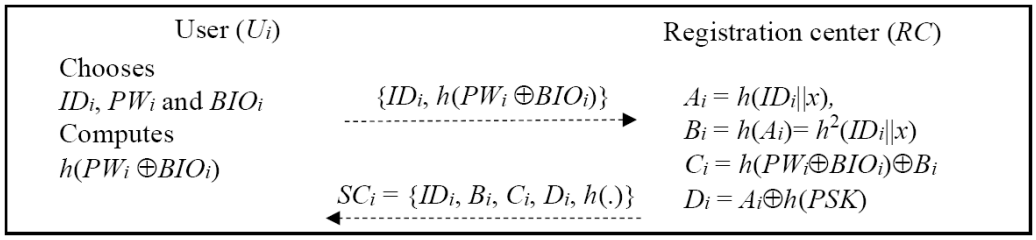

FIGURE 1. Registration phase

4.1. **Registration phase.** Our registration phase uses the hash function to hide the PSK in the last step as $D_i = A_i \oplus h(PSK)$. We note that the Chuang-Chen's and Choi et al.'s schemes both use the PSK in plaintext and abridge the value E_i in the Choi et al.'s scheme. Firstly, U_i chooses the identity ID_i and password PW_i. Secondly, the user U_i computes the value $h(PW_i \oplus BIO_i)$. Then, the user U_i sends ID_i and $h(PW_i \oplus BIO_i)$ to RC via a secure channel. If RC accepts the request, RC computes $A_i = h(ID_i||x)$, $B_i = h(A_i) = h^2(ID_i||x)$, $C_i = h(PW_i||BIO_i) \oplus B_i$ and $D_i = A_i \oplus h(PSK)$. Then, RC gives U_i a smart card SC_i containing $\{ID_i, B_i, C_i, D_i, h(.)\}$ via a secure channel. The registration phase is shown as Figure 1.

4.2. **Login and authentication phase.** When U_i wants to log in to the server S_j, she/he inserts smart card SC_i and inputs ID_i, PW_i and BIO_i. The smart card SC_i computes $B_i' = h(PW_i \oplus BIO_i) \oplus C_i$ and checks whether B_i' is equivalent to the stored B_i. If it is successful, SC_i generates a random number N_1 and computes $h(PSK) = D_i \oplus h(PW_i \oplus BIO_i)$, $M_1 = h(B_i) \oplus N_1 \oplus h(PSK)$, $AID_i = h(N_1) \oplus ID_i$ and $M_2 = h(N_1||AID_i||D_i||SID_j||T_i)$. Then, SC_i sends the message $\{AID_i, M_1, M_2, D_i, T_i\}$ to the server S_j, where T_i is both the current timestamp and the challenge nonce of U_i.

After receiving the message from U_i at time T_s, the server S_j checks the freshness of T_i. Then, S_j retrieves $A_i = D_i \oplus h(PSK)$, $N_1 = M_1 \oplus h^2(A_i) \oplus h(PSK)$, and checks whether $M_2' = h(N_1||AID_i||D_i||SID_j||T_i)$ is equivalent to the received M_2. If it fails, S_j rejects U_i's login request. Otherwise, S_j accepts U_i's request and generates a random number N_2. S_j computes the session key $SK_{ji} = h(N_1||N_2)$, $M_3 = N_2 \oplus h^2(N_1) \oplus h(PSK)$, $M_4 = h(SID_j||N_2||AID_i||T_i)$ and sends the message $\{SID_j, M_3, M_4\}$ to U_i. Upon receiving the message from S_j, SC_i retrieves $N_2 = M_3 \oplus h^2(N_1) \oplus h(PSK)$, and checks whether $M_4' = h(SID_i||N_2||AID_i||T_i)$ is equivalent to the received M_4. If it fails, U_i terminates this session. Otherwise, SC_i computes the session key $SK_{ij} = h(N_1||N_2)$ as the session key. The authentication phase is shown as in Figure 2.

4.3. **Password change phase.** Our registration phase is similar to that of Choi et al.'s scheme. When the user U wants to change her/his password from PW to PW^*, U_i inserts her/his smart card SC_i into a card reader and inputs her/his ID_i, PW_i, BIO_i and PW^*. The smart card SC_i checks whether the ID_i and $B_i' = h(PW_i \oplus BIO_i) \oplus C_i$ are equivalent to the stored ID_i and B'. If it fails, SC_i rejects U_i's request. Otherwise, it accepts U_i's password changing request. The SC_i computes $C_i^* = C_i \oplus h(PW_i \oplus BIO_i) \oplus h(PW_i^* \oplus BIO_i)$ and replaces C_i with C_i^*. The password change phase is shown in Figure 3.

5. **Security and Efficiency Analysis.** In this section, we analyze the security and performance of our scheme. Our scheme is similar to the Choi et al.'s scheme. Our scheme also inherits some weaknesses in the Choi et al.'s scheme. Nevertheless, the replay attack will fail due to our timestamp challenge/response scheme. Based on the scheme, a replay attack cannot pass the subsequent challenges. When the server S_j receives the message $\{AID_i, M_1, M_2, D_i, T_i\}$, it includes a challenge nonce T_i from U_i. Therefore, the S_j must send back to U_i from a corresponding value derived from T_i as the response nonce. When U_i receives the message M_4, it includes the challenge nonce T_i. Note that T_i as challenge

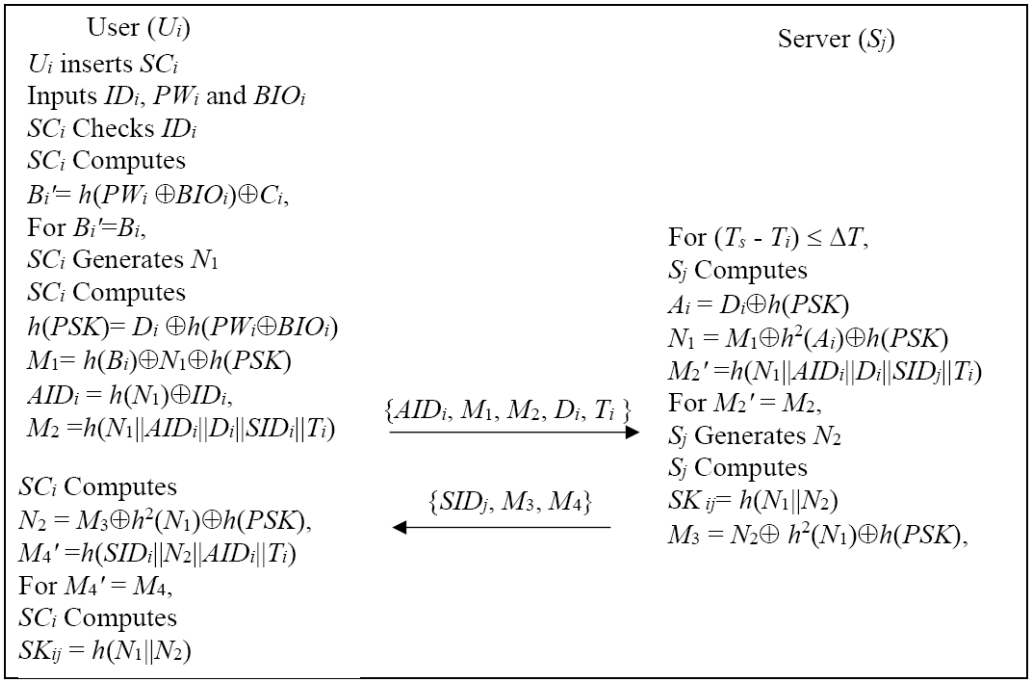

FIGURE 2. The login and authentication phase of our scheme

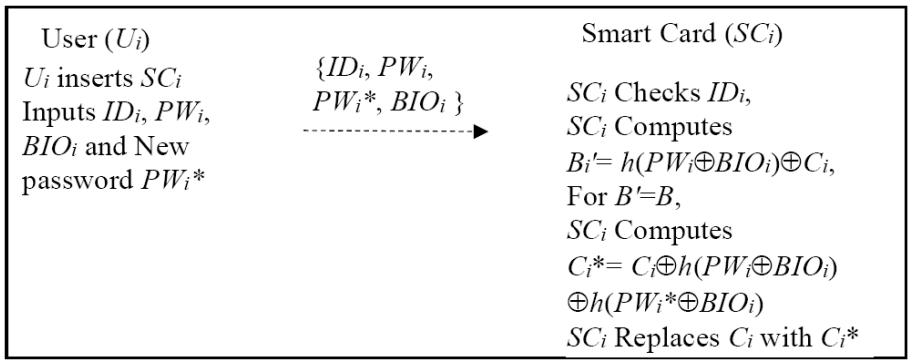

FIGURE 3. The password change phase of our scheme

nonce is fresh. Additionally, we do not send the response message from U_i to S_j. Without knowing the response message $\{SK_{ij} \oplus h(N_2)\}$ from U_i to S_j, the attacker is impossible to create session key $SK_{ij} = h(N_1||N_2)$. Hence, our scheme can provide the perfect forward secrecy.

The performance comparisons of our proposed scheme with Chuang-Chen's and Choi et al.'s schemes are summarized in Table 2. In three schemes, in the registration phase of user, the computation cost is equal. In the registration phase of RC, both performs three hashing operations but Choi et al.'s scheme performs two more XOR operations to adds a value E_i. In login and authentication phase of user, our scheme performs two more XOR and three more concatenation operations than Chuang-Chen's scheme and two less hash and one less XOR operations than Choi et al.'s scheme. In login and authentication phase of server, our scheme performs two less hash operations and three more concatenation operations than Chuang-Chen's scheme because we use timestamp as challenge/response instead of the verification message and avoid the weakness of session key. We also find that the verification message can be neglected in the Choi et al.'s scheme that also uses the timestamp to avoid the replay attack and takes two more hash and one more concatenation operations compared with our scheme. In the password change phase, both our and Chuang-Chen's schemes require two hashing and five XOR

Table 2. Performance comparison of three schemes

	Chuang-Chen's scheme	Choi et al.'s scheme	Our scheme
Registration user	1H + 1X + B	1H + 1X + B	1H + 1X + B
Registration RC	3H + 2X + 1C	3H + 4X + 1C	3H + 2X + 1C
Login and authentication user	6H + 6X + 4C + 1B	8H + 9X + 7C + 1B	6H + 8X + 7C + 1B
Login and authentication server	10H + 5X + 4C	8H + 7X + 7C	8H + 5X + 7C
Password change user	2H + 5X	2H + 7X	2H + 5X

H: number of H() operation; X: number of XOR operation; C: concatenation operation; B: biometric operation.

operations that use two less XOR operations than Choi et al.'s scheme. Regarding the total computation cost of three schemes, we can find that our scheme has the similar efficiency to Chuang-Chen's scheme but better than Choi et al.'s scheme.

6. **Conclusion.** In this paper, we analyze the weaknesses of Choi et al.'s remote user authentication scheme. When the attacker registers as a legal user and intercepts the login and authentication request/response messages, she/he can compute the important common secret value of each smart card and offline guesses the user identity. Secondly, suppose that the insider records the registration values in the *RC*, even the insider does not have the smart card, password or biometrics of users and only records the identity of user to register and the intercepted messages in the common channel. The insider can get the secret information and attack the scheme. Thirdly, if the attacker is a legal user, she/he can perform the server spoofing attack. Therefore, their scheme suffers the mutual authentication problem. Finally, Choi et al.'s scheme cannot provide the forward secrecy. The attacker can get the session keys using only the messages in the common channel without the long-term secret, smart card, password or biometrics of users. Through entire analysis, we find that Choi et al.'s scheme may be not suitable for applications in the network, which require user privacy and security. We also find, if adding a simple and experimentally feasible modification to the Choi et al.'s scheme, the modified scheme can protect the session key and achieve perfect forward secrecy. We also show our simple modification with better efficiency. As security breached is increasing, new authentication techniques need to incorporate biometric to increase security of the remote system. In future, we can develop the more lightweight scheme to be suitable for the resource constrained devices as IoT (Internet of Things), provide the formal verification to confirm its security and set up a test platform to prove the proposed scheme is suitable for applications in the network.

REFERENCES

[1] H.-C. Hsiang and W.-K. Shih, Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment, *Computer Standards & Interfaces*, vol.31, pp.1118-1123, 2009.

[2] C.-T. Li and M.-S. Hwang, An efficient biometrics-based remote user authentication scheme using smart cards, *Journal of Network and Computer Applications*, vol.33, pp.1-5, 2010.

[3] T.-H. Chen, H.-C. Hsiang and W.-K. Shih, Security enhancement on an improvement on two remote user authentication schemes using smart cards, *Future Generation Computer Systems*, vol.27, pp.377-380, 2011.

[4] T. Wan, Z. Liu and J. Ma, Authentication and key agreement protocol for multi-server architecture, *Journal of Computer Research & Development*, vol.53, pp.2446-2453, 2016.
[5] S. Kumari, M. K. Gupta and M. Kumar, Cryptanalysis and security enhancement of Chen et al.'s remote user authentication scheme using smart card, *Central European Journal of Computer Science*, vol.2, pp.60-75, 2012.
[6] P. Yu and W.-G. Shieh, A new scheme of remote user authentication using smart cards, *ICIC Express Letters*, vol.11, no.1, pp.59-64, 2017.
[7] Y. Choi, J. Nam, D. Lee, J. Kim, J. Jung and D. Won, Security enhanced anonymous multiserver authenticated key agreement scheme using smart cards and biometrics, *The Scientific World Journal*, vol.2014, p.15, 2014.
[8] M.-C. Chuang and M. C. Chen, An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics, *Expert Systems with Applications*, vol.41, pp.1411-1418, 2014.
[9] P. Kocher, J. Jaffe and B. Jun, Differential power analysis, *Proc. of Advances in Cryptology*, 1999.
[10] T. S. Messerges, E. A. Dabbish and R. H. Sloan, Examining smart-card security under the threat of power analysis attacks, *IEEE Trans. Computers*, vol.51, pp.541-552, 2002.
[11] X. Huang, X. Chen, J. Li, Y. Xiang and L. Xu, Further observations on smart-card-based password-authenticated key agreement in distributed systems, *IEEE Trans. Parallel and Distributed Systems*, vol.25, pp.1767-1775, 2014.