Cryptanalysis and Improvement of an Advanced Anonymous and Biometrics-Based Multi-server Authentication Scheme Using Smart Cards

Chunyi Quan[1], Hakjun Lee[1], Dongwoo Kang[1], Jiye Kim[2], Seokhyang Cho[3], and Dongho Won[1(✉)]

[1] Information Security Group, Sungkyunkwan University, Suwon, South Korea
{sikwon, hjlee, dwkang, dhwon}@security.re.kr
[2] Department of Mobile Internet, Daelim University College, Anyang, South Korea
jykim.isg@gmail.com
[3] Department of Information and Communication, Pyeongtaek University, Pyeongtaek, South Korea
cshlch@ptu.ac.kr

Abstract. In conventional single-server environment, a user must register to every server if he/she wants to access numerous network services. It is exceedingly hard for users to generate different robust passwords and remember them with corresponding identities. To solve this problem, many multi-server authentication schemes have been proposed in recent years. In 2017, Chang et al. improved Chuang and Chen's scheme, arguing that their scheme provides higher security and practicability. However, we demonstrate that Chang et al.'s scheme is still vulnerable to outsider attack and session key derived attack. In addition, we also find that both malicious user and server can carry out user impersonation attack in their scheme. In this paper, we propose a new biometrics-based authentication scheme that is suitable for use in multi-server environment. Finally, we show that the proposed scheme improves on the level of security in comparison with related schemes.

Keywords: Authentication · Multi-server · Biometrics · Smart card

1 Introduction

In 1981, Lamport [1] proposed the first remote password authentication scheme under insecure network. However, his scheme is proved to be insecure against guessing attacks. Therefore, smart card based scheme were considered as a solution and came into sight. By utilizing smart cards, instead of keeping a verification table, participants are allowed to store secret information into a smart card which improves security to a new level. After that, other novel schemes [2, 3] which adopt biometrics were introduced for further enhancement. However, all aforementioned schemes [1–3] are designed for single-server environment which makes users extremely inconvenient to

D. Nicholson (ed.), *Advances in Human Factors in Cybersecurity*,
Advances in Intelligent Systems and Computing 593, DOI 10.1007/978-3-319-60585-2_7

access resource from servers because they must register to each server separately. To solve this problem, a new authentication structure for multi-server environment was introduced and several related schemes have been proposed [4–8].

In 2010, Yang and Yang [4] proposed a biometric password-based multi-server authentication scheme using smart card which enables users to register for only once and then be qualified to access all servers. Unfortunately, their scheme costs vast computational resource due to the heavy use of modular exponentiation operations. In the same year, Yoon and Yoo [5] proposed an improved scheme based on elliptic curve cryptosystem. He [6] demonstrated that their scheme cannot resist privileged-insider attack, masquerade attack and stolen smart card attack. In 2014, Chuang and Chen [7] presented a scheme under the assumption that all servers are trusted and achieves both high efficiency and security. However, Chang et al. [8] proved that Chuang and Chen's scheme is insecure against stolen smart card attack, forgery attack and has privacy preservation issue. Furthermore, Chang et al. indicated that in traditional biometric-based scheme the authentication may fails due to the slight difference between imprinted biometrics and original ones. Therefore, they adopted functions defined in Dodis et al.'s work [9] and proposed an enhanced scheme, claiming that their scheme satisfies all desirable security requirements. In this paper, after careful analysis, we find that Chang et al.'s scheme is vulnerable to outsider attack and session key derived attack. In addition, both malicious user and server can carry out user impersonation attack in their scheme. To resolve these vulnerabilities, we propose a new biometric-based authentication scheme that is suitable for multi-server environment. In particular, the comparison on security level between our scheme and other related schemes [2–5, 8] implies that our scheme can defend against a number of attacks including the ones of Chang et al.'s scheme.

The rest of the paper is organized as follows: In Sect. 2, we introduce basic concepts of secure sketch presented by Dodis et al. In Sects. 3 and 4, we review and cryptanalyze Chang et al.'s scheme. Section 5 describes the proposed scheme. Sections 6 and 7 gives a detailed security and performance analysis where our scheme is compared with related schemes, respectively. Finally, in Sect. 7, we conclude this paper.

2 Secure Sketch

The major problem of biometrics-based authentication scheme is that the imprinted biometric can slightly differentiate with the original template since some noise are unavoidably introduced into the reproducing process. To rectify this weakness, Chang et al. [8] adopted Dodis et al.'s function [9] which is defined that a $(\mathcal{M},\ m, m', t)$ secure sketch is a randomized map $SS : \mathcal{M} \rightarrow \{0,1\}^*$ in which m is min-entropy, m' is the lower bound of average m and t refers to the number of tolerated errors.

For distance function dis and vectors $w, w' \in \mathcal{M}$, a deterministic recovery function $Rec(w', SS(w)) = w$ exists which allows to recover w from its sketch $SS(w)$ and w' that is close to w as long as $dis(w, w') < t$ is satisfied. According to this definition, for any given binary $[n, k, 2t+1]$ error correcting code E, we set randomized map SS as a $(\mathcal{M},\ m, m+k-n, t)$-secure sketch and $SS(W; X) = W \oplus E(X)$, where n is string length, k indicates the dimension of codeword, W is uniform and X is a random

parameter. There is a decoding function D can correct t errors maximum that $dis(W, W') < t$. D works as $D(W', S(W; X)) = X$. Lastly, we can set the recovery function $Rec(W', S(W; X)) = SS(W; X) \oplus E(D(W' \oplus SS(W; X))) = W$.

3 Review of Chang et al.'s Scheme

In this section, we briefly review the advanced anonymous and biometrics-based multi-server authentication scheme of Chang et al. [8]. Their scheme consists of following phases: server registration, user registration, login, authentication and password change. The notations used in this paper are described in Table 1.

Table 1. Notations

Notations	Description
U_i, S_j, SC_i	User, server and user's smart card
RC	Registration center
ID_i, SID_j	Identity of U_i and S_j
PW_i, BIO_i	Password and biometrics of U_i
x, y	The secret key and number of RC
$E(\cdot), D(\cdot)$	The encoding and decoding function based on Dodis et al.'s paper [9]
$h(\cdot)$	A secure hash function

3.1 Server Registration Phase

S_j sends a registration request to RC via a secure channel. RC accepts S_j and computes $k_1 = h\big(SID_j \parallel h(y)\big)$ and $k_2 = h(x \parallel y)$. Finally, RC sends k_1 and k_2 back to S_j.

3.2 User Registration Phase

1. U_i freely chooses his/her identity ID_i, password PW_i, and imprints his/her personal biometric information BIO_i into a special device. U_i randomly generates a number r_i that is only retained by himself/herself and computes $\alpha_i = BIO_i \oplus E(r_i)$, $V_i = h(PW_i) \oplus \alpha_i$ and $R_i = h(PW_i \oplus r_i)$. Afterwards, U_i transmits $\{ID_i, V_i, R_i\}$ to RC via a secure channel.
2. After receiving the registration request message from U_i, RC calculates $A_i = h(ID_i \parallel x)$, $B_i = h(ID_i \parallel R_i)$, $C_i = h^2(R_i) \oplus h(y)$, $D_i = h(R_i) \oplus A_i \oplus h(x \parallel y)$ and $E_i = h(A_i \parallel h(x \parallel y)) \oplus h(R_i)$.
3. Lastly, RC stores $\{V_i, B_i, C_i, D_i, E_i, h(.)\}$ into SC_i and sends it to U_i.

3.3 Login Phase

1. U_i inserts his/her SC_i into a card reader, inputs his/her ID_i^* and PW_i^*, imprints personal biometric information BIO_i^* via a special device.

2. SC_i employs inputted information to compute $R_i^* = h\big(PW_i^* \oplus D\big(V_i \oplus h\big(PW_i^*\big) \oplus BIO_i^*\big)\big)$ and verifies whether $h\big(ID_i^* \parallel R_i^*\big)$ equals to B_i. SC_i only proceeds to the next step when they are equal.
3. SC_i generates a random nonce n_i and computes $h(y) = C_i \oplus h^2\big(R_i^*\big)$, $M_1 = h\big(SID_j \parallel h(y)\big) \oplus n_i$, $CID_i = D_i \oplus h\big(R_i^*\big) \oplus h(n_i)$, $G_i = E_i \oplus h\big(R_i^*\big)$ and $CHECK_1 = h\big(h\big(SID_j \parallel h(y)\big) \parallel n_i \parallel G_i\big)$.
4. SC_i sends login request message $\{M_1, CID_i, CHECK_1\}$ to S_j.

3.4 Authentication Phase

1. Upon receiving the login request message from U_i, S_j first employs its secret k_1 to compute random nonce $n_i = M_1 \oplus k_1$ to check its freshness. If n_i is fresh, S_j subsequently computes $A_i = CID_i \oplus h(n_i) \oplus k_2$ and verifies whether $h(k_1 \parallel n_i \parallel h(A_i \parallel k_2))$ equals to $CHECK_1$. If it holds, S_j considers U_i as valid user.
2. S_j generates a random number n_j and computes $M_2 = n_j \oplus n_i \oplus k_1$, $SK = h(h(A_i \parallel k_2) \parallel n_i \parallel n_j)$ and $CHECK_2 = h(SK)$, followed by sending a response message $\{M_2, CHECK_2\}$ to U_i via a public channel.
3. SC_i retrieves random nonce n_j by computing $n_j = M_2 \oplus h\big(SID_j \parallel h(y)\big) \oplus n_i$ and checks its freshness. If n_j is fresh, SC_i then computes $SK = h\big(G_i \parallel n_i \parallel n_j\big)$ and checks if $h(SK)$ equals to $CHECK_2$. If the verification succeeds, SC_i computes $CHECK_3 = h\big(SK \parallel n_j\big)$ and sends it to S_j via a public channel.
4. After receiving $CHECK_3$ from U_i, S_j verifies whether $h\big(SK \parallel n_j\big)$ equals to $CHECK_3$ to reconfirm the authenticity of U_i. Then, U_i and S_j can start to communicate with the other party using the shared session key.

3.5 Password Change Phase

1. U_i inserts his/her SC_i into a card reader and inputs ID_i, PW_i and BIO_i.
2. SC_i computes $\alpha_i = V_i \oplus h(PW_i)$, $r_i = D(BIO_i \oplus \alpha_i)$ and $R_i = h(PW_i \oplus r_i)$, and verifies the condition $h(id_i \parallel R_i) = ?B_i$. If the it holds, SC_i asks U_i to submit a new password, otherwise password change request can be dropped.
3. U_i submits a new password PW_i^{new} and then SC_i employs it to compute $V_i^{new} = V_i \oplus h(PW_i) \oplus h\big(PW_i^{new}\big)$, $R_i^{new} = h\big(PW_i^{new} \oplus r_i\big)$, $B_i^{new} = h\big(ID_i \parallel R_i^{new}\big)$, $C_i^{new} = C_i \oplus h^2(R_i) \oplus h^2\big(R_i^{new}\big)$, $D_i^{new} = D_i \oplus h(R_i) \oplus h\big(R_i^{new}\big)$ and $E_i^{new} = E_i \oplus h(R_i) \oplus h\big(R_i^{new}\big)$. Finally, SC_i replaces V_i, B_i, C_i, D_i and E_i with V_i^{new}, B_i^{new}, C_i^{new}, D_i^{new} and E_i^{new}.

4 Cryptanalysis of Chang et al.'s Scheme

In this section, we cryptanalyze Chang et al.'s scheme [8] and demonstrate that their scheme possesses some security vulnerabilities. According to the threat model described in [10–12], an adversary can eavesdrop, modify and intercept any message in the public channel, and that an adversary can extract all information stored in the smart card by carrying out power analysis [11]. Under these two assumptions, the scheme has the following security problems and the descriptions are given below.

4.1 Outsider Attack

A malicious server $\mathcal{A}$ is aware of secrets k_1 and k_2 that are authenticated from RC and can retrieve A_i and n_i after receiving login request message $\{M_1, CID_i, CHECK_1\}$ from U_i during the authentication phase. If $\mathcal{A}$ steals SC_i which belong to the user he/she is communicating with and extracts parameters $\{C_i, D_i\}$ from it, he/she can compute $h(R_i) = D_i \oplus A_i \oplus k_2$ and then obtains the encrypted secret number of RC by calculating $h(y) = C_i \oplus h^2(R_i)$, which is the same for each user. Therefore, $\mathcal{A}$ may be able to launch other attacks with the knowledge of RC's secret $h(y)$.

4.2 Session Key Derived Attack

Suppose a malicious server $\mathcal{A}$ obtains RC's secret $h(y)$ in the previous attack. He/she can easily compute the session key that is transmitted between any user and server. The attack proceeds as follows:

1. $\mathcal{A}$ eavesdrops login request message $\{M_1, CID_i, CHECK_1\}$ between U_i and S_j, and computes $n_i = h\big(SID_j \parallel h(y)\big) \oplus M_1$ and $A_i = CID_i \oplus h(n_i) \oplus k_2$.
2. Then, $\mathcal{A}$ eavesdrops S_j's response message $\{M_2, CHECK_2\}$, retrieves the nonce n_j by computing $n_j = M_2 \oplus h\big(SID_j \parallel h(y)\big) \oplus n_i$. Afterwards, $\mathcal{A}$ can obtain the session key by computing $SK = h\big(h(A_i \parallel k_2) \parallel n_i \parallel n_j\big)$.

4.3 User Impersonation Attack

Although Chang et al. [8] claim that their scheme can endure user impersonation attack, however after careful analysis we find that an adversary $\mathcal{A}$ can still impersonate as a legitimate user to cheat with S_j. Especially in Chang et al.'s scheme, $\mathcal{A}$ can either be a malicious server or user. Suppose $\mathcal{A}$ is a malicious server who obtains RC's secret $h(y)$ by means of the attack we described in Sect. 4.1. In addition, each server is allocated with same secret value k_2 from RC. He/she can perform this attack by follows:

1. $\mathcal{A}$ intercepts the login request message $\{M_1, CID_i, CHECK_1\}$ sent from legal U_i to S_j and computes $n_i = h\big(SID_j \parallel h(y)\big) \oplus M_1$ and $A_i = CID_i \oplus h(n_i) \oplus k_2$.

2. $\mathcal{A}$ generates a random number n_i^*, then computes $M_1^* = h\big(SID_j \parallel h(y)\big) \oplus n_i^*$, $CID_i^* = A_i \oplus k_2 \oplus h\big(n_i^*\big)$ and $CHECK_1^* = h\big(h\big(SID_j \parallel h(y)\big) \parallel n_i^* \parallel h(A_i \parallel K_2)\big)$ and sends the forged login request message $\big\{M_1^*, CID_i^*, CHECK_1^*\big\}$ to S_j.
3. S_j retrieves $n_i^* = M_1^* \oplus k_1$ using the request message. Since n_i^* is chosen within valid time interval, S_j proceeds to compute $A_i = CID_i \oplus h\big(n_i^*\big) \oplus k_2$ and verify the condition $h(k_1 \parallel n_i \parallel h(A_i \parallel k_2)) = ?CHECK_1$. Obviously, the condition holds, therefore S_j authenticates $\mathcal{A}$ as legal user and computes $M_2 = n_j \oplus n_i^* \oplus k_1$, $SK = h\big(h(A_i \parallel k_2) \parallel n_i^* \parallel n_j\big)$ and $CHECK_2 = h(SK)$, where n_j is the random number generated by S_j. Finally, S_j reply $\mathcal{A}$ with $\{M_2, CHECK_2\}$.
4. After receiving the response message, $\mathcal{A}$ retrieves $n_j = m_2 \oplus h\big(SID_j \parallel h(y)\big) \oplus n_i^*$, $SK = h\big(A_i \oplus k_2 \parallel n_i^* \parallel n_j\big)$ and computes $CHECK_3 = h\big(SK \parallel n_j\big)$. Afterwards, $\mathcal{A}$ sends mutual authentication message $CHECK_3$ to S_j.
5. Upon receiving the authentication message from $\mathcal{A}$, S_j continues to proceed the scheme. Lastly, S_j is mistakenly convinced that $\mathcal{A}$ is a legitimate user and agrees on the session key SK with him/her.

If $\mathcal{A}$ is a malicious user, he/she still can launch this attack by follows:

1. $\mathcal{A}$ obtains RC's secret $h(y)$ by calculating $h(y) = C_a \oplus h^2(R_a)$, where C_a is stored in $\mathcal{A}$'s smart card and R_a can be recovered from $R_a = h(PW_a \oplus D(V_a \oplus h(PW_a) \oplus BIO_a))$. by using his/her PW_a and BIO_a.
2. $\mathcal{A}$ intercepts the login request message $\{M_1, CID_i, CHECK_1\}$ sent from U_i to S_j and computes $n_i = h\big(SID_j \parallel h(y)\big) \oplus M_1$ and $A_i \oplus k_2 = CID_i \oplus h(n_i)$.
3. $\mathcal{A}$ steals SC_i and extracts $\{V_i, B_i, C_i, D_i, E_i, h(.)\}$ from it by using power analysis. Then, $\mathcal{A}$ calculates $h(R_i) = D_i \oplus A_i \oplus k_2$ and $G_i = E_i \oplus h(R_i)$.
4. $\mathcal{A}$ computes $M_1^* = h\big(SID_j \parallel h(y)\big) \oplus n_i^*$, $CID_i^* = A_i \oplus k_2 \oplus h\big(n_i^*\big)$ and $CHECK_1^* = h\big(h\big(SID_j \parallel h(y)\big) \parallel n_i^* \parallel h(A_i \parallel k_2)\big)$, where random number n_i^* is chosen by $\mathcal{A}$ freely. Then $\mathcal{A}$ forges login request message $\big\{M_1^*, CID_i^*, CHECK_1^*\big\}$ and sends it to S_j.
5. Upon receiving the message from $\mathcal{A}$ who manages to impersonate as legal user U_i, the message can successfully pass S_j's verification.
6. Perform steps 3 to 5 in aforementioned attack that $\mathcal{A}$ is a malicious server. Finally, S_j authenticates $\mathcal{A}$ and shares the same session key with him/her.

5 The Proposed Scheme

This section proposes an improved biometrics-based authentication scheme that is suitable for use in multi-server environment. The proposed scheme comprises three participants: user (U_i), server (S_j), registration center (RC), and five phases: server registration, user registration, login, authentication, and password change.

5.1 Server Registration Phase

The server registration phase of proposed scheme is same as Chang et al.'s scheme [8].

5.2 User Registration Phase

1. U_i conducts in the same method as described in step 1 in Sect. 3.2.
2. Upon receiving the registration request message from U_i, RC computes $A_i = h(ID_i \| x)$, $B_i = h(ID_i \| R_i)$, $C_i = h(R_i) \oplus h(y)$, $D_i = A_i \oplus h(x \| y)$ and $E_i = h(A_i \| h(x \| y)) \oplus h(R_i \| h(y))$.
3. RC issues SC_i which contains $\{V_i, B_i, C_i, D_i, E_i, h(.)\}$ and sends it to U_i.

5.3 Login Phase

1. U_i inserts SC_i into a card reader, inputs ID_i^*, PW_i^* and BIO_i^*. SC_i first computes $R_i^* = h\big(PW_i^* \oplus D\big(V_i \oplus h\big(PW_i^*\big) \oplus BIO_i^*\big)\big)$ and verifies whether $h\big(ID_i^* \| R_i^*\big)$ equals to B_i. If it generates negative result, this phase can be terminated.
2. SC_i generates a random nonce n_i and computes $h(y) = C_i \oplus h\big(R_i^*\big)$, $M_1 = h\big(SID_j \| h(y)\big) \oplus n_i$, $CID_i = D_i \oplus h(n_i)$, $G_i = E_i \oplus h\big(R_i^* \| h(y)\big)$ and $CHECK_1 = h\big(h\big(SID_j \| h(y)\big) \| n_i \| G_i\big)$.
3. SC_i sends the request message $\{M_1, CID_i, CHECK_1\}$ to S_j.

5.4 Authentication Phase

1. S_j first checks the validity of the request message by verifying the freshness of random nonce $n_i = M_1 \oplus k_1$. If it holds, S_j computes $A_i = CID_i \oplus h(n_i)$ and verifies whether $h(k_1 \| n_i \| h(A_i \| k_2))$ equals to $CHECK_1$. If the condition holds, S_j authenticates U_i. Otherwise, the session is aborted.
2. S_j further generates a random number n_j and computes $M_2 = n_j \oplus n_i \oplus k_1$, $SK = h\big(h(A_i \| k_2) \| n_i \| n_j\big)$ and $CHECK_2 = h(SK)$. Then, S_j sends the response message $\{M_2, CHECK_2\}$ to U_i.
3. The rest of the authentication phase is same as Chang et al.'s scheme.

5.5 Password Change Phase

1. U_i inserts his/her SC_i into a card reader, then keys his/her ID_i and PW_i, and imprints personal biometric information BIO_i via a special device.
2. SC_i retrieves $\alpha_i = V_i \oplus h(PW_i)$, $r_i = D(BIO_i \oplus \alpha_i)$ and $R_i = h(PW_i \oplus r_i)$, and verifies the whether $h(id_i \| R_i)$ is equal to B_i. If it holds, U_i is allowed to type a new password, otherwise this phase can be aborted.
3. U_i types a new password PW_i^{new}. SC_i calculates $h(y) = C_i \oplus h(R_i)$, $V_i^{new} = V_i \oplus h(PW_i) \oplus h\big(PW_i^{new}\big)$, $R_i^{new} = h\big(PW_i^{new} \oplus r_i\big)$, $B_i^{new} = h\big(ID_i \| R_i^{new}\big)$, $C_i^{new} = C_i \oplus h(R_i) \oplus h\big(R_i^{new}\big)$ and $E_i^{new} = E_i \oplus h(R_i \| h(y)) \oplus h\big(R_i^{new} \| h(y)\big)$. Lastly, SC_i replaces V_i, B_i, C_i and E_i with V_i^{new}, B_i^{new}, C_i^{new} and E_i^{new}.

6 Cryptanalysis of Proposed Scheme

In this section, we cryptanalyze the proposed scheme and examines its security against various attacks. As described in Sect. 5, to achieve least increase on computational cost, our scheme modifies little in user registration phase and login phase based on Chang et al.'s scheme [8] and provides higher security. Therefore, all security features mentioned in [8] are also met in our scheme. In addition, we comparatively give an analysis between our scheme and previous schemes [2–5, 8], which is illustrated in Table 2.

Table 2. Comparison on security level between proposed scheme and related schemes

Features	Ours	[8]	[3]	[5]	[4]	[2]
Outsider attack	Yes	No	Yes	Yes	Yes	Yes
Session key derived attack	Yes	No	Yes	Yes	Yes	Yes
User impersonation attack	Yes	No	No	No	Yes	No
Off-line password guessing attack	Yes	Yes	Yes	Yes	Yes	No
Server spoofing attack	Yes	Yes	No	Yes	No	No
Stolen smart card attack	Yes	Yes	No	No	Yes	No

Yes: The scheme can resist the attack. No: The scheme cannot resist the attack

6.1 Resistance to Outsider Attack

Assume an adversary $\mathcal{A}$ is a malicious server who is aware of $k_1 = h(SID_j \parallel h(y))$ and $k_2 = h(x \parallel y)$, however he/she cannot obtain $h(y)$ by computing $h(y) = h(R_i) \oplus C_i$, where C_i is stored in SC_i. Only possessing correct ID_i, PW_i and BIO_i can retrieve random number r_i and further compute R_i. The possibility that $\mathcal{A}$ obtains ID_i and PW_i simultaneously is extremely small, and BIO_i cannot be forged or obtained since it is imprinted by U_i via a special device. Furthermore, $h(R_i)$ is only applied to constitute C_i, which means $\mathcal{A}$ is not capable of obtaining it from operating with any other parameters. Therefore, our scheme prevents $\mathcal{A}$ from launching outsider attack.

6.2 Resistance to Session Key Derived Attack

The session key is computed as $SK = h(h(A_i \parallel k_2) \parallel n_i \parallel n_j)$, where $A_i = h(ID_i \parallel x)$, $k_2 = h(x \parallel y)$, random numbers n_i and n_j are generated by U_i and S_j, respectively. Assume an adversary $\mathcal{A}$ somehow obtains ID_i, he/she cannot compute SK without the knowledge of secrets x and y that are only known by RC. $\mathcal{A}$ cannot retrieve random numbers n_i and n_j neither, since they must be computed by using $h(y)$ and k_1, which indicates that only legal user and server can compute these two random nonces. Therefore, $\mathcal{A}$ cannot reveal session key SK by any means in the proposed scheme.

6.3 Resistance to User Impersonation Attack

Assume that an adversary $\mathcal{A}$ intercepts all messages $\{M_1, M_2, M_3, CID_i, CHECK_1, CHECK_2, CHECK_3\}$ between U_i and S_j through a public network, steals SC_i and extracts all information $\{V_i, B_i, C_i, D_i, E_i, h(.)\}$. However, $\mathcal{A}$ cannot forge login request message $\{M_1, CID_i, CHECK_1\}$, where $M_1 = h(SID_j \parallel h(y)) \oplus n_i$, $CID_i = D_i \oplus h(n_i) = A_i \oplus h(x \parallel y) \oplus h(n_i)$ and $CHECK_1 = h(h(SID_j \parallel h(y)) \parallel n_i \parallel G_i) = h(h(SID_j \parallel h(y)) \parallel n_i \parallel h(A_i \parallel h(x \parallel y)))$, because secrets x and y are only known to RC, n_i is a random nonce that is generated by U_i. Furthermore, $\mathcal{A}$ cannot generate $\{M_1, CID_i, CHECK_1\}$ without A_i, which can be exclusively obtained by S_j. If the adversary $\mathcal{A}$ is a malicious user or server, he/she is capable of retrieving some parameters within $\{n_i, h(SID_j \parallel h(y)), h(x \parallel y), h(y)\}$. However, as described in Subsects. 6.1 and 6.2, it is impossible for $\mathcal{A}$ to obtain all parameters that form a valid login request message $\{M_1, CID_i, CHECK_1\}$ to impersonate as a legitimate user. Hence, our scheme can resist user impersonation attack.

7 Performance Analysis

In this section, we compare our scheme with other related schemes [2–5, 8] on computational cost during login and authentication phase, which is illustrated in detail in Table 3. Notations used in this section are described as follows. T_h refers to the time to execute a one-way hash function for a single time. T_E and T_D are defined as the time taken to perform one encoding or decoding operation based on Dodis et al.'s definition [9]. T_{ecc} is the computation time that one elliptic curve operation requires. T_e indicates the computation time for one modular exponentiation operation. The computational parameter T_f indicates the computation time to execute fuzzy extractor for once. Although our scheme requires one more hash operation during login phase compared with Chang et al.'s scheme, however it consumes an extremely small amount of time. Considering the security enhancement of proposed scheme, the increased computation cost is worthy.

Table 3. Comparison of computational cost in login and authentication phase between proposed scheme and related schemes

Phases	Ours	[8]	[3]	[5]	[4]	[2]
Login	$8T_h + 1T_D$	$7T_h + 1T_D$	$4T_h$	$2T_h + 1T_{ecc}$	$4T_h + 1T_e + 1T_f$	$2T_h$
Authentication	$10T_h$	$10T_h$	$13T_h$	$15T_h + 3T_{ecc}$	$4T_h + 4T_e$	$8T_h$
Total	$18T_h + 1T_D$	$17T_h + 1T_D$	$17T_h$	$17T_h + 4T_{ecc}$	$8T_h + 5T_e + 1T_f$	$10T_h$

8 Conclusions

In this paper, we analyze Chang et al.'s scheme and demonstrate that it possesses a number of security vulnerabilities including outsider attack, session key derived attack and user impersonation attack. To overcome these flaws, we propose an improved biometrics-based authentication scheme which retains the merits of Chang et al.'s

scheme and also achieves a variety of security features. In addition, the cryptanalysis of this paper shows that our scheme rectifies weaknesses of Chang et al.'s scheme.

Acknowledgements. This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No. R0126-15-1111, The Development of Risk-based Authentication Access Control Platform and Compliance Technique for Cloud Security).

References

1. Lamport, L.: Password authentication with insecure communication. Commun. ACM **24**(11), 770–772 (1981)
2. Das, A.K.: Analysis and improvement on an efficient biometric-based remote user authentication scheme using smart cards. IET Inf. Secur. **5**(3), 145–151 (2011)
3. Li, X., Niu, J.W., Ma, J., Wang, W.D., Liu, C.L.: Cryptanalysis and improvement of a biometrics-based remote user authentication scheme using smart cards. J. Netw. Comput. Appl. **34**(1), 73–79 (2011)
4. Yang, D., Yang, B.: A biometric password-based multi-server authentication scheme with smart card. In: 2010 International Conference on Computer Design and Applications (ICCDA), vol. 5, p. V5-554. IEEE (2010)
5. Yoon, E.J., Yoo, K.Y.: Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem. J. Supercomput. **63**(1), 235–255 (2013)
6. He, D.: Security flaws in a biometrics-based multi-server authentication with key agreement scheme. IACR Cryptology ePrint Archive, 365 (2011)
7. Chuang, M.C., Chen, M.C.: An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics. Expert Syst. Appl. **41**(4), 1411–1418 (2014)
8. Chang, C.C., Hsueh, W.Y., Cheng, T.F.: An advanced anonymous and biometrics-based multi-server authentication scheme using smart cards. Int. J. Netw. Secur. **18**(6), 1010–1021 (2016)
9. Dodis, Y., Reyzin, L, Smith, A.: Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. In: International Conference on the Theory and Applications of Cryptographic Techniques, pp. 523–540. Springer, Heidelberg (2004)
10. Moon, J., Choi, Y., Jung, J., Won, D.: An improvement of robust biometrics-based authentication and key agreement scheme for multi-server environments using smart cards. PloS One **10**(12), e0145263.5 (2015)
11. Jung, J., Kang, D., Lee, D., Won, D.: An improved and secure anonymous biometric-based user authentication with key agreement scheme for the integrated EPR information system. PLoS One **12**(1), e0169414 (2017)
12. Kim, J., Lee, D., Jeon, W., Lee, Y., Won, D.: Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks. Sensors **14**(4), 6443–6462 (2014)