Improvement on a Biometric-Based Key Agreement and Authentication Scheme for the Multi-server Environments

Jongho Moon[1], Youngsook Lee[2], Hyungkyu Yang[3], Hakjun Lee[1], Sewan Ha[1], and Dongho Won[1](✉)

[1] Department of Electrical and Computer Engineering, Sungkyunkwan University, 2066 Seobu-ro, Jangan-gu, Suwon-si, Gyeonggi-do 16419, Korea
{jhmoon,hjlee,hsewan,dhwon}@security.re.kr

[2] Department of Cyber Security, Howon University, 64 Howondae 3-gil, Impi-myeon, Gunsan-si, Jeonrabuk-do 54058, Korea
ysooklee@howon.ac.kr

[3] Department of Computer and Media Information, Kangnam University, 40 Gangnam-ro, Giheung-gu, Yongin-si, Gyeonggi-do 16979, Korea
hkyang@kangnam.ac.kr

Abstract. With the rapid spiraling network users expansion and the enlargement of communication technologies, the multi-server environment has been the most common environment for widely deployed applications. Wang et al. recently have shown that Mishra et al.'s biohasing-based authentication scheme for multi-server was insecure, and then presented a fuzzy-extractor-based authentication protocol for key-agreement and multi-server. They continued to assert that their protocol was more secure and efficient. After a prudent analysis, however, their enhanced scheme still remains vulnerabilities against well-known attacks. In this paper, the weaknesses of Wang et al.'s protocol such as the outsider and user impersonation attacks are demonstrated, followed by the proposal of a new fuzzy-extractor and smart card-based protocol, also for key agreement and multi-server environment. Lastly, the authors shows that the new key-agreement protocol is more secure using random oracle method and Automated Validation of Internet Security Protocols and Applications (AVISPA) tool, and that it serves to gratify all of the required security properties.

Keywords: Multi-server · Authentication · Fuzzy-extractor Biometrics

1 Introduction

Transmission environments of the information become more open and dynamic, research on the trustworthiness of large-scale network has become progressively more crucial [1]. The typical previous user authentication schemes verify the

O. Gervasi et al. (Eds.): ICCSA 2018, LNCS 10960, pp. 541–557, 2018.
https://doi.org/10.1007/978-3-319-95162-1_37

entered credentials with the stored databases. Since the first authentication scheme that is based on password was presented by Lamport [2] in 1981, a variety of authentication schemes [3–5] which are based on password have been presented. Regarding password authentication scheme, however, a server needs to store a list which is stored the password for the identification of the credentials of a remote user; the server thus must make arrangements for additional storage or memory for the storage of the password table. Furthermore, several researcher studies have shown that the password-based authentication protocols are vulnerable against some attacks such as the off-line password guessing or stolen smart card [6,7]. For these reasons, many researchers have suggested a new user authentication protocol for key-agreement using biometrics. The biometrics has a major characteristic which is the uniqueness. Numerous remote user authentication schemes [8–11] have used biological characteristics. In multi-server environments (MSE), each user can approach any type of application server, regardless of their physical location, by using a single registration; for this reason, a secure remote user authentication protocol is required in the MSE. Figure 1 delineates this structure, which incorporates a one-time registration, a single smart card, and the same credentials. For this reason, the MSE requires a secure and forceful remote user authentication protocol.

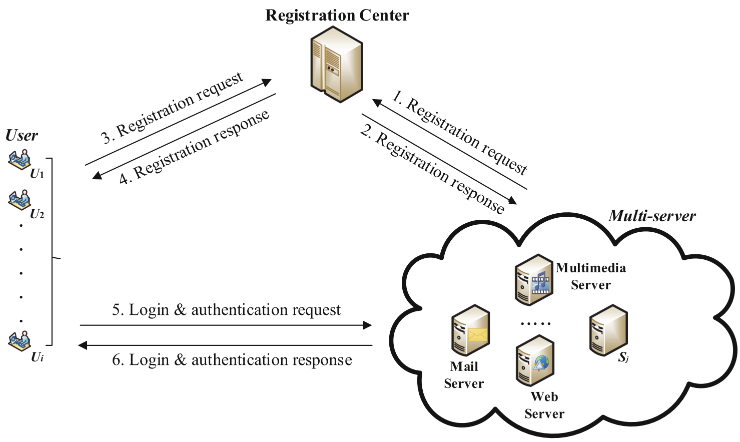

Fig. 1. The basic architecture of multi-server

During the past decade, many researchers have presented user authentication protocols for the MSE. In 2008, Tsai [12] proposed user authentication scheme without verification record using hash function; after that, Liao and Wang [13] presented an user authentication protocol using a dynamic identity. Hsiang and Shih [14], however, have shown that Liao and Wang's protocol was vulnerable against the replay, server spoofing, and stolen-verifier attacks, and aimed to provide mutual authentication, forward secrecy, and user anonymity.

In 2012, Li et al. [15] presented an user authentication protocol using a dynamic identity and smart card; however, Xue et al. [16] showed that Li et al.'s protocol was insecure against some attacks which are replay, eavesdropping, insider, impersonation, and denial of service (DoS), and then presented a new authentication protocol for key agreement using dynamic identity. Nevertheless, Lu et al. [17] have shown that Xue et al.'s protocol was vulnerable against some type of attacks which are masquerade, off-line password guessing and insider. To overcome these vulnerabilities, Lu et al. then presented a meliorated identity-based key-agreement protocol. Chuang and Chen [18] presented a trust computing-based authentication protocol that uses biometrics and smart cards, and they asserted that improved protocol can achieve a variety of security features; unfortunately, Mishra et al. [19] have shown that their protocol was not secure to user impersonation, server spoofing and stolen smart card attacks, and presented a new authentication protocol for key agreement using biometrics; however, Lu et al. [20] have shown that Mishra et al.'s protocol was insecure to the replay attack, and also does not provide an effective password change phase; furthermore, Wang et al. [21] have shown that Mishra et al.'s protocol was vulnerable against masquerade, replay and DoS attacks, and it cannot satisfy perfect forward secrecy. To overcome these problems, Wang et al. suggested a meliorated, authentication protocol for key agreement using biometrics; unfortunately, their proposed protocol is still insecure against some type of attacks which are outsider and user impersonation.

In this paper, we review the authentication protocol of Wang et al. [21] and show how the adversary can impersonate a legal user. Wang et al. [21] have improved the vulnerabilities of previous authentication schemes, and shown the efficient computational cost. Their scheme consists only a hash function and fuzzy-extraction technique. After demonstrating these problems, an improved fuzzy extractor-based authentication protocol is presented for MSE. Our contribution is to prove and overcome the weaknesses of Wang et al.'s protocol [21]. Lastly, the improved protocol is analyzed according to the security properties and the computational cost.

The remainder of the paper is constituted as follows: Some definitions such as threat assumptions and fuzzy extractor that are adopted for the proposed scheme are briefly introduced in Sect. 2; in Sects. 3 and 4, Wang et al.'s protocol is reviewed and analyzed, respectively; in Sect. 5, an improved fuzzy extractor-based authentication scheme is presented; in Sect. 6, a formal and informal analysis and simulation result of the improved protocol is demonstrated; Sect. 7 shows the comparison of security and performance of the improved protocol with the previous protocols; lastly, in Sect. 8, the conclusion is demonstrated.

2 Preliminaries

Some definitions of the threat assumptions and the fuzzy extractor which are useful to understand this paper are demonstrated in this section.

2.1 Threat Assumptions

The Dolev-Yao threat model [22] is introduced here, and the risk of side-channel attacks [23] is considered for the construction of the threat assumptions [8] that are demonstrated as follows:

(TA1) A remote user can be either an adversary $\mathcal{A}D$ or a legal user. In other words, a legitimate user can perform as any adversary $\mathcal{A}D$.
(TA2) The $\mathcal{A}D$ can intercept, modification such as insert or delete, or reroute any transmitted communication message over public channel.
(TA3) Using the examining the power consumption, the $\mathcal{A}D$ can pull out the stored information from the any issued smart card.

2.2 Fuzzy Extractor

The fuzzy extractor can convert from the biometrics to a random string, is described here. Based on the Refs. [24,25], the fuzzy extractor is made of the two procedures (*Gen*, *Rep*).

- $Gen(Biometrics) \rightarrow \langle \alpha, \beta \rangle$
- $Rep(Biometrics^{*}, \beta) = \alpha \; if \; Biometrics^{*} \; is \; reasonably \; close \; to \; Biometrics.$

The probabilistic generation procedure *Gen* can extract some binary string $\alpha \in \{0,1\}^k$ and string $\beta \in \{0,1\}^*$ from the biometrics, where α is nearly random string and β is an auxiliary binary, and the deterministic reproduction procedure *Rep* can recover a nearly random binary string α from the auxiliary string β and any biometrics $Biometrics^{*}$ when the $Biometrics^{*}$ is pretty similar the $Biometrics$. Additional information can be found in the research [26].

3 Review of Wang et al.'s Protocol

Wang et al.'s fuzzy extractor-based authentication protocol for key agreement is reviewed here. Their protocol consists of three entities, as follows: user, server, and registration authority. Six phases relate to their protocol, and they are the server registration, user registration, login, authentication, password changing, and revocation or re-registration phases. For convenience, Table 1 describes some of the expressions that are used in this paper.

3.1 Server Registration

(SR1) S_j sends the message to the registration authority RA for server registration request.
(SR2) RA sends PK which is the pre-shared key to S_j through Internet Key Exchange Protocol version 2 (IKEv2) [27] by using a secure communication route.

Table 1. Expressions

Notation	Description
U_i, S_j, SC_i	User i, server j and smart card of U_i
$\mathcal{A}D$	Adversary
RA	Registration authority
ID_i, PW_i, BIO_i, DID_i	Identity, password, biometrics and dynamic identity of U_i
SID_j	Identity of S_j
TR_i	Registration time of U_i
R_i	Positive random integer unique to U_i
x	Master secret key selected by RA
α_i, β_i	U_i's nearly random and auxiliary binary strings
PK	Secure key pre-shared by RA and S_j
$\oplus$, $\|$	XOR and concatenation operation
$h(\cdot)$	Collision-resistance one-way hash function

3.2 User Registration

(UR1) U_i gives one's biometrics BIO_i at the biometrics scan sensor. The sensor then scans the BIO_i, pulls out the two random strings (α_i, β_i) from the computation $Gen(BIO_i) \rightarrow (\alpha_i, \beta_i)$, and keeps the β_i in the temporary storage. U_i hence chooses ID_i and PW_i, and calculates $DPW_i = h(PW_i \| \alpha_i)$. Lastly, U_i sends the message $\langle ID_i, DPW_i \rangle$ to RA for user registration by using a secure communication network.

(UR2) RA registers a new user record $\langle ID_i, UR_i = 1 \rangle$ into the database, where UR_i is the registration frequency of U_i. RA then calculates $V_i = h(ID_i \| x \| TR_i)$, $W_i = DPW_i \oplus h(V_i)$, $X_i = W_i \oplus h(PK)$, $Y_i = PK \oplus V_i \oplus h(PK)$ and $Z_i = h(ID_i \| DPW_i)$, where TR_i is the registration time of U_i.

(UR3) RA replies a new SC_i to U_i, which is composed of $\langle W_i, X_i, Y_i, Z_i, h(\cdot) \rangle$ by using a secure communication network.

(UR4) After receiving the smart card, U_i stores β_i into SC_i.

3.3 Login

(L1) U_i inserts own SC_i into a card recognizing device, enters ID_i and PW_i, and gives BIO_i^* at the biometrics scan sensor. The sensor hence scans the BIO_i^*, and recovers α_i from the $Rep(BIO_i^*, \beta_i) \rightarrow \alpha_i$.

(L2) SC_i then computes $DPW_i = h(PW_i \| \alpha_i)$, and checks whether $h(ID_i \| DPW_i)$ is same to the stored Z_i. If this holds, SC_i further calculates $h(PK) = W_i \oplus X_i$.

(L3) Next, SC_i chooses some random digits RN_1, and calculates $DID_i = ID_i \oplus h(RN_1)$, $M_1 = DPW_i \oplus RN_1 \oplus h(PK)$ and $M_2 = h(DID_i \parallel RN_1 \parallel DPW_i \parallel SID_j \parallel TS_i)$, where TS_i is the timestamp.

(L4) Lastly, SC_i sends the message $\langle DID_i, M_1, M_2, W_i, Y_i, TS_i \rangle$ to S_j for login request by using a public communication network.

3.4 Authentication

(A1) S_j verifies whether $TS_j - TS_i \leq \triangle TS$ is reasonable, where $\triangle TS$ is the minimum acceptable time interval and TS_j is the actual arrival time of the message. If this holds, S_j proceeds on the next stage; otherwise, S_j rejects the request.

(A2) S_j computes $V_i = PK \oplus Y_i \oplus h(PK)$, $DPW_i = W_i \oplus h(V_i)$, $RN_1 = DPW_i \oplus M_1 \oplus h(PK)$, and checks whether $h(DID_i \parallel RN_1 \parallel DPW_i \parallel SID_j \parallel TS_i)$ is same to the received M_2.

(A3) If this holds, S_j chooses some random digits RN_2, and calculates the common session secret key $SK_{ji} = h(DID_i \parallel SID_j \parallel RN_1 \parallel RN_2)$.

(A4) S_j computes $M_3 = RN_2 \oplus h(DID_i \parallel RN_1) \oplus h(PK)$ and $M_4 = h(SID_j \parallel RN_2 \parallel DID_i)$, and replies the response message $\langle SID_j, M_3, M_4 \rangle$ to U_i by using a public communication network.

(A5) SC_i computes $RN_2 = M_3 \oplus h(DID_i \parallel RN_1) \oplus h(PK)$, $SK_{ij} = h(DID_i \parallel SID_j \parallel RN_1 \parallel RN_2)$, and then checks whether $h(SID_j \parallel RN_2 \parallel DID_i)$ is same to the received M_4. If this holds, SC_i calculates $M_5 = h(SK_{ij} \parallel RN_1 \parallel RN_2)$, and sends $\langle M_5 \rangle$ to S_j by using a public communication network.

(A6) S_j checks whether $h(SK_{ji} \parallel RN_1 \parallel RN_2)$ is equal to the received M_5. If this holds, S_j can accept the session key SK_{ji} in this session; otherwise, S_j rejects any request message.

3.5 Password Change

(P1) U_i first inserts own SC_i into a card recognizing device, enters ID_i and PW_i, and gives BIO_i^* at the biometrics scan sensor. The sensor then scans BIO_i^*, and recovers α_i from the computation $Rep(BIO_i^*, \beta_i) \rightarrow \alpha_i$.

(P2) SC_i then computes $DPW_i = h(PW_i \parallel \alpha_i)$, and checks whether $h(ID_i \parallel DPW_i)$ is same to the stored Z_i. If this holds, SC_i trying to ask the user about the new password; otherwise, SC_i immediately terminates the password change phase.

(P3) After inputting the new PW_i^{new}, SC_i computes $DPW_i^{new} = h(PW_i^{new} \parallel \alpha_i)$, $W_i^{new} = W_i \oplus DPW_i \oplus DPW_i^{new}$, $X_i^{new} = X_i \oplus W_i \oplus W_i^{new}$ and $Z_i^{new} = h(ID_i \parallel DPW_i^{new})$.

(P4) Lastly, SC_i replaces W_i, X_i and Z_i with W_i^{new}, X_i^{new} and Z_i^{new}.

3.6 Revocation or Re-registration

If any user U_i wants to revoke his/her right, it is necessary that the U_i sends the message $\langle DPW_i \rangle$ to RA for revocation and verification by using a secure communication network. RA checks whether U_i is legitimate. If this holds, RA then updates the user's record by setting $\langle ID_i, UR_i = 0 \rangle$. Similarly, after receiving the message for re-registration request by using a public communication network, RA performs the same steps explained in Sect. 3.2, and it changes the user record from $\langle ID_i, UR_i \rangle$ to $\langle ID_i, UR_i = UR_i + 1 \rangle$.

4 Cryptanalysis of Wang et al.'s Protocol

Security weaknesses of Wang et al.'s protocol is shown here, and the authors shows that Wang et al.'s protocol is vulnerable to outsider, user impersonation and privileged insider attacks.

4.1 Outsider Attack

Outsider attack means that a legitimate user who issued a smart card uses his/her card to extract a meaningful value for attack. Let $\mathcal{AD}$, who is the legitimate user but malicious, he/she then can extract the stored information $\{W_{\mathcal{AD}}$, $X_{\mathcal{AD}}$, $Y_{\mathcal{AD}}$, $Z_{\mathcal{AD}}$, $\beta_{\mathcal{AD}}$, $h(\cdot)\}$ from the one's smart card; then, the $\mathcal{AD}$ can easily calculate $h(PK) = W_{\mathcal{AD}} \oplus X_{\mathcal{AD}}$, which is the same for any legitimate user and the pre-shared server key's hash result.

4.2 User Impersonation Attack

Suppose an adversary $\mathcal{AD}$ eavesdrops any user U_i's request message $\langle DID_i$, M_1, M_2, W_i, Y_i, $TS_i \rangle$ for login. $\mathcal{AD}$ can then perform the user impersonate attack by using message modification.

- (UA1) Outsider adversary $\mathcal{AD}$ obtains $h(PK) = W_{\mathcal{AD}} \oplus X_{\mathcal{AD}}$ from his/her smart card.
- (UA2) $\mathcal{AD}$ randomly generates some nonce $RN_{\mathcal{AD}}$.
- (UA3) $\mathcal{AD}$ then computes $W_i^* = W_i \oplus h(PK)$, $Y_i^* = h(PK)$, $M_1^* = W_i \oplus RN_{\mathcal{AD}} \oplus h(PK)$ and $M_2^* = h(DID_i \parallel RN_{\mathcal{AD}} \parallel W_i \parallel SID_j \parallel TS_{\mathcal{AD}})$, where the $TS_{\mathcal{AD}}$ is the current timestamp.
- (UA4) $\mathcal{AD}$ sends the message $\langle DID_i$, M_1^*, M_2^*, W_i^*, Y_i^*, $TS_{\mathcal{AD}} \rangle$ to the server S_j for login by using a public communication network.
- (UA5) S_j checks whether $TS_j - TS_{\mathcal{AD}} \leq \triangle TS$ is valid. This holds, because the $TS_{\mathcal{AD}}$ has a fresh value.
- (UA6) S_j retrieves $V_i = PK \oplus Y_i^* \oplus h(PK) = PK \oplus h(PK) \oplus h(PK) = PK$, $DPW_i = W_i^* \oplus h(V_i) = W_i \oplus h(PK) \oplus h(PK) = W_i$ and $RN_{\mathcal{AD}} = DPW_i \oplus M_1^* \oplus h(PK) = W_i \oplus W_i \oplus RN_{\mathcal{AD}} \oplus h(PK) \oplus h(PK)$, and verifies whether $h(DID_i \parallel RN_{\mathcal{AD}} \parallel W_i \parallel SID_j \parallel TS_{\mathcal{AD}})$ is equal to the received M_2^*.

(UA7) This holds, S_j then proceeds on the protocol without being detected. Lastly, $\mathcal{A}D$ and S_j "successfully" conclude the session; unfortunately, the S_j faultily decides that he/she is communicating with U_i.

5 The Improved Authentication Protocol

In this section, a new fuzzy extractor-based authentication protocol is proposed. Six phases relate to the proposed protocol, and they are the server registration, user registration, login, authentication, password changing, and revocation or re-registration phases.

5.1 Server Registration

(SR1) S_j sends the message to RA for registration request.
(SR2) RA replies PK which is pre-shared key and second master key x to S_j using the Internet Key Exchange Protocol version 2 (IKEv2) [27] by using secure communication network.

5.2 User Registration

(UR1) U_i imprints own biometrics BIO_i at the biometrics scan sensor. The sensor then scans the BIO_i, pulls out the two random strings (α_i, β_i) from the computation $Gen(BIO_i) \rightarrow (\alpha_i, \beta_i)$, and keeps the β_i in the temporary storage. U_i hence chooses ID_i and PW_i, and calculates $T_i = h(ID_i \parallel \alpha_i)$ and $DPW_i = h(PW_i \parallel \alpha_i)$. Lastly, the U_i sends the request message $\langle ID_i, DPW_i \rangle$ to RA for user registration by using a secure communication network, and stores T_i in the memory.
(UR2) RA registers a new user record $\langle ID_i, UR_i = 1 \rangle$ to the database, where UR_i is the registration frequency of U_i. RA then calculates $V_i = h(ID_i \parallel x \parallel R_i)$, $W_i = DPW_i \oplus h(V_i)$, $X_i = h(R_i \parallel PK)$, $Y_i = PK \oplus R_i \oplus h(PK)$ and $Z_i = h(ID_i \parallel DPW_i)$, where R_i is a positive random integer unique to the user.
(UR3) RA replies the new SC_i to U_i, which is composed of $\{W_i, X_i, Y_i, Z_i, h(\cdot)\}$ by using a secure communication network.
(UR4) U_i computes $X_i^* = X_i \oplus T_i$, replaces X_i with X_i^*, stores β_i into SC_i, removes β_i and T_i from the memory, and initialize the authentication environments.

5.3 Login

(L1) U_i first inserts own SC_i into a card recognizing device, enters ID_i and PW_i, and gives BIO_i^* at the biometrics scan sensor. The sensor then scans the BIO_i^*, and recovers α_i from the computation $Rep(BIO_i^*, \beta_i) \rightarrow \alpha_i$.
(L2) SC_i then computes $DPW_i = h(PW_i \parallel \alpha_i)$, and checks whether $h(ID_i \parallel DPW_i)$ is same to the stored Z_i. If this holds, SC_i further calculates $h(R_i \parallel PK) = X_i \oplus h(ID_i \parallel \alpha_i)$.

(L3) Next, SC_i chooses some random digits RN_1, and calculates $DID_i = ID_i \oplus h(RN_1)$, $M_1 = RN_1 \oplus h(R_i \parallel PK)$ and $M_2 = h(DID_i \parallel RN_1 \parallel DPW_i \parallel SID_j \parallel TS_i)$, where TS_i is the current timestamp.
(L4) Lastly, SC_i sends the message $\langle DID_i, M_1, M_2, W_i, Y_i, TS_i \rangle$ to S_j for login request by using a public communication network.

5.4 Authentication

(A1) S_j verifies whether $TS_j - TS_i \leq \triangle TS$ is reasonable, where $\triangle TS$ is the minimum acceptable time interval and TS_j is the actual arrival time of the message. If this holds, S_j proceeds on the next stage; otherwise, S_j rejects the login request.
(A2) S_j retrieves $R_i = PK \oplus Y_i \oplus h(PK)$, $RN_1 = M_1 \oplus h(R_i \parallel PK)$ and $ID_i = DID_i \oplus h(RN_1)$, and computes $V_i^* = h(ID_i \parallel x \parallel R_i)$ and $DPW_i = W_i \oplus h(V_i^*)$, and checks whether $h(DID_i \parallel RN_1 \parallel DPW_i \parallel SID_j \parallel TS_i)$ is same to the received M_2.
(A3) If this holds, S_j chooses some random digits RN_2, and calculates the common session secret key $SK_{ji} = h(DID_i \parallel SID_j \parallel h(V_i) \parallel RN_1 \parallel RN_2)$.
(A4) S_j computes $M_3 = RN_2 \oplus RN_1$ and $M_4 = h(SID_j \parallel SK_{ji} \parallel RN_1 \parallel RN_2 \parallel DID_i)$, and replies the authentication response message $\langle M_3, M_4 \rangle$ to U_i by using a public communication network.
(A5) SC_i computes $RN_2 = M_3 \oplus RN_1$, $SK_{ij} = h(DID_i \parallel SID_j \parallel W_i \oplus DPW_i \parallel RN_1 \parallel RN_2)$, and then checks whether $h(SID_j \parallel SK_{ij} \parallel RN_1 \parallel RN_2 \parallel DID_i)$ is same to the received M_4. If this holds, SC_i can accept the session key SK_{ij} in this session; otherwise, U_i terminates this session.

5.5 Password Change

(P1) U_i first inserts own SC_i into a card recognizing device, enters ID_i and PW_i, and gives BIO_i^* at the biometrics scan sensor. The sensor then scans the BIO_i^*, and recovers α_i from the computation $Rep(BIO_i^*, \beta_i) \rightarrow \alpha_i$.
(P2) SC_i then computes $DPW_i = h(PW_i \parallel \alpha_i)$, and checks whether $h(ID_i \parallel DPW_i)$ is same to the stored Z_i. If this holds, SC_i trying to ask the user about the new password; otherwise, SC_i immediately terminates the password change phase.
(P3) After inputting the new PW_i^{new}, SC_i computes $DPW_i^{new} = h(PW_i^{new} \parallel \alpha_i)$, $W_i^{new} = W_i \oplus DPW_i \oplus DPW_i^{new}$ and $Z_i^{new} = h(ID_i \parallel DPW_i^{new})$.
(P4) Lastly, SC_i replaces W_i and Z_i with W_i^{new} and Z_i^{new} into the smart card.

5.6 Revocation or Re-registration Phase

User revocation phase is same as the user revocation phase in Wang et al.'s protocol. If user U_i want to re-registration, the registration authority RA reissues the smart card to the user. The RA checks the UR_i value at the time of the user's login request, and if the UR_i is greater than 1, RA uses the value UR_i to calculate V_i^*.

(RR1) After receiving the request from U_i for re-registration, RA updates a user record $\langle ID_i, UR_i = UR_i + 1\rangle$ to the database. RA then calculates $V_i = h(ID_i \parallel x \parallel UR_i \parallel R_i)$, $W_i = DPW_i \oplus h(V_i)$, $X_i = h(R_i \parallel PK)$, $Y_i = PK \oplus R_i \oplus h(PK)$ and $Z_i = h(ID_i \parallel DPW_i)$, where R_i is a positive random integer unique to the user.

(RR2) RA replies the new SC_i to U_i, which is composed of $\{W_i, X_i, Y_i, Z_i, h(\cdot)\}$ by using a secure communication network.

(RR3) U_i computes $X_i^* = X_i \oplus T_i$, replaces X_i with X_i^*, stores β_i into SC_i, removes β_i and T_i from the memory, and initialize the authentication environments.

6 Cryptanalysis of the Proposed Protocol

The improved protocol, which maintains the merits of Wang et al.'s protocol, is demonstrated, and it can resist some type of possible attacks and supports all of the security features. The cryptanalysis of the improved protocol was organized with threat assumptions.

6.1 Informal Security Analysis

We explain the improved protocol can resist various kinds of known attacks.

Outsider Attack. Outsider attack means that a legitimate user who issued a smart card uses his/her card to extract a meaningful value for attack. Assume that an adversary $\mathcal{A}D$ who issued a smart card extracts $\{W_{\mathcal{A}D}, X_{\mathcal{A}D}, Y_{\mathcal{A}D}, Z_{\mathcal{A}D}, \beta_{\mathcal{A}D}, h(\cdot)\}$ from the one's smart card. $\mathcal{A}D$ can retrieve $h(R_{\mathcal{A}D} \parallel PK) = X_{\mathcal{A}D} \oplus h(ID_{\mathcal{A}D} \parallel \alpha_{\mathcal{A}D})$; however, $R_{\mathcal{A}D}$ is a positive random integer that has the different value, and PK is the pre-shared key between RA and S_j. $\mathcal{A}D$ cannot obtain and use this value to the other attack, and the proposed protocol can therefore avoid the outsider attack.

Modification Attack. Assume that $\mathcal{A}D$ intercepts the transmitted informations $\{DID_i, M_1, M_2, W_i, Y_i, TS_i, M_3, M_4\}$; however, the $\mathcal{A}D$ cannot retrieve RN_1, RN_2, R_i and PK from these messages. Even if $\mathcal{A}D$ uses his/her $h(R_{\mathcal{A}D} \parallel PK)$, A cannot generate M_1 without the DPW_i. To compute DPW_i, the second master key x is needed. The proposed protocol can therefore avoid the modification attack.

Off-Line Password Guessing Attack. Assume that U_i's SC_i is lost or $\mathcal{A}D$ steals SC_i of U_i, $\mathcal{A}D$ can then obtain $\{W_i, X_i, Y_i, Z_i, \beta_i, h(\cdot)\}$; however, he/she cannot guess the password of U_i. To guess the password from $h(PW_i \parallel \alpha_i)$, α_i is needed; however, α_i is in possession of the high entropy; moreover, the same biometrics are not present between any two people. The proposed protocol can therefore avoid the off-line password guessing attack.

User Impersonation Attack. Assume that $\mathcal{AD}$ intercepts the transmitted informations $\{DID_i,\ M_1,\ M_2,\ W_i,\ Y_i,\ TS_i,\ M_3,\ M_4\}$; however, $\mathcal{AD}$ cannot make the reasonable message $\{DID_i,\ M_1,\ M_2,\ W_i,\ Y_i,\ TS_i\}$ for login request. This is because R_i is a positive random integer that is different from the other user's thing, and RN_1 is some random digits that is selected by U_i. To make M_2, the second master key x is needed. The proposed protocol can therefore avoid the user impersonation attack.

Stolen Smart Card Attack. Suppose that $\mathcal{AD}$ steals U_i's SC_i, he/she then extracts $\{W_i,\ X_i,\ Y_i,\ Z_i,\ \beta_i\ h(\cdot)\}$.; however, $\mathcal{AD}$ cannot obtain any sensitive information of U_i. Although $\mathcal{AD}$ obtains the $h(R_{\mathcal{AD}} \parallel PK)$ from one's smart card, $R_{\mathcal{AD}}$ and R_i are the different values. The proposed protocol can therefore avoid the stolen smart card attack.

Table 2. Algorithm $EXP_{HASH,A}^{BASMK}$

Eavesdrop the login request message $\langle DID_i,\ M_1,\ M_2,\ W_i,\ Y_i,\ TS_i\rangle$
Call the oracle. Let $(RN_1',\ DPW_i') \leftarrow Reveal(M_2)$
Eavesdrop the authentication response message $\langle M_3,\ M_4\rangle$
Use the oracle. Let $(SK_{ji}',\ RN_1'',\ RN_2') \leftarrow Reveal(M_4)$
if $(RN_1' = RN_1'')$ **then**
  Compute $ID_i' = DID_i \oplus h(RN_1')$ and $H_1 = M_1 \oplus RN_1' = h(R_i \parallel PK)$
  Use the oracle. Let $(R_i', PK') \leftarrow Reveal(H_1)$
  Compute $H_2 = Y_i \oplus R_i' \oplus PK' = h(PK)$
  Use the oracle. Let $(PK'') \leftarrow Reveal(H_2)$
    if $(PK' = PK'')$ **then**
      Compute $RN_2'' = M_3 \oplus RN_1'$
      if $(RN_2' = RN_2'')$ **then**
        Call the oracle. Let $(PW_i',\ \alpha_i') \leftarrow Reveal(DPW_i')$
        Compute $h(V_i) = W_i \oplus DPW_i'$
        Compute $SK_{ij}' = h(DID_i \parallel SID_j \parallel h(V_i) \parallel RN_1' \parallel RN_2'')$
          if $(SK_{ji}' == SK_{ij}')$ **then**
            Accept $ID_i',\ PW_i',\ \alpha_i',\ R_i'$ as the correct $ID_i,\ PW_i,\ \alpha_i,\ R_i$, $PK',\ SK_{ij}$ as the correct PK and SK_{ij}, respectively.
            return 1 (Success)
          else
            return 0 (Failure)
      else
        return 0 (Failure)
      end if
    else
      return 0 (Failure)
    end if
else
  return 0 (Failure)
end if

6.2 Formal Security Analysis

The formal analysis using random oracle method is demonstrated here, and its security is shown. First, the following hash function is defined Refs. [8,28]:

Definition 1. *The secure and collision-resistance hash function* $\mathcal{H}(\cdot) : \{0, 1\}^* \to \{0, 1\}^k$ *picks up any input as a binary string* $a \in \{0, 1\}^*$ *which has a randomly length, extracts a binary string* $\mathcal{H}(a) \in \{0, 1\}^k$*, and satisfies the following conditions:*

(i) Given the $b \in B$, it's mathematically impossible to find out a $a \in A$ such that $b = \mathcal{H}(a)$.
(ii) Given the $a \in A$, it's mathematically impossible to find out the another $a' \neq a \in A$, such that $\mathcal{H}(a') = \mathcal{H}(a)$.
(iii) It's mathematically impossible to find out a pair $(a', a) \in A' \times A$, with $a' \neq a$, such that $\mathcal{H}(a') = \mathcal{H}(a)$

Theorem 1. *According to the assumptions that if the hash function* $\mathcal{H}(\cdot)$ *closely performs like an oracle, then the protocol is certainly secure to the adversary* $\mathcal{AD}$ *for the protection of the meaningful information including the identity* ID_i*, the password* PW_i*, the nearly random binary string* α_i*, the positive random integer* R_i*, the pre-shared key* PK *and the common session key* SK_{ij}*.*

Proof. Formal proof of the proposed protocol is analogous to those in Refs. [8, 20,28,29], and it uses the following random oracle model to construct the $\mathcal{AD}$, who will have the ability to recover the ID_i, PW_i, α_i, R_i, PK and SK_{ij}.

Reveal. The random oracle can obtain the input a from the hash result $b = \mathcal{H}(a)$ without failure. $\mathcal{AD}$ now performs the experimental algorithm as shown in Table 2, $EXP_{HASH,\ A}^{BASMK}$ for the proposed protocol as BASMK. Let's define the probability of success for $EXP_{HASH,A}^{BASMK}$ as $Success_{HASH,\ A}^{BASMK} = |Pr[EXP_{HASH,\ A}^{BASMK} = 1] - 1|$, where $Pr(\cdot)$ means the probability of $EXP_{HASH,A}^{BASMK}$. The advantage function for this algorithm then becomes $Adv_{HASH,\ A}^{BASMK}(t,\ q_R) = max_{Success}$, where t and q_R are the execution cost and number of queries. Consider the algorithm as shown in Table 2. If the $\mathcal{AD}$ has the capability to crack the problem of hash function given in Definition 1, $\mathcal{AD}$ can then immediately obtain the ID_i, PW_i, α_i, R_i, PK and SK_{ij}. In that case, $\mathcal{AD}$ will detect the complete connections between the U_i and S_j; however, the inversion of the input from the given hash result is impossible computationally, i.e., $Adv_{HASH,\ A}^{BASMK}(t) \leq \epsilon$, for all $\epsilon > 0$. Therefore, $Adv_{HASH,\ A}^{BASMK}(t,\ q_R) \leq \epsilon$, since $Adv_{HASH,\ A}^{BASMK}(t,\ q_R)$ depends on $Adv_{HASH,\ A}^{BASMK}(t)$. In conclusion, it is no method for $\mathcal{AD}$ to detect the complete connections between the U_i and S_j, the proposed protocol thus is certainly secure to $\mathcal{AD}$ for retrieving $(ID_i,\ PW_i,\ \alpha_i,\ R_i,\ PK,\ SK_{ij})$.

Table 3. The result of the analysis using OFMC backend

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/testrv4.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
 parseTime: 0.00s
 searchTime: 71.86
 visiteNodes: 11440 nodes
 depth: 9 piles
```

6.3 Simulation Using AVISPA

We perform to simulate the improved protocol for formal analysis using the widely accepted AVISPA. The main contribution of this simulation is to verify whether the proposed protocol is invulnerable to two attacks which are replay and man-in-the middle. AVISPA is composed of four back-ends: (1) On-the-fly Model-Checker; (2) Constraint-Logic-based Attack Searcher; (3) SAT-based Model Checker; and (4) Tree Automata based on Automatic Approximations for the Analysis of Security Protocols. The protocol is implemented in High Level Protocol Specification Language (HLPSL) [28] in AVISPA. The fundamental classes available in the HLPSL are [30]. The simulation result of the proposed protocol using OMFC is shown in Table 3. The result shows that two attacks which are man-in-the middle and replay have no effect on the proposed protocol.

7 Functionality and Performance Analysis

The comparisons of the functionality and computational cost of the proposed protocol with the other previous protocols [15,16,18–21] are demonstrated here.

7.1 Functionality Analysis

Table 4 itemizes the avoidance comparisons of various biometric-based key agreement protocols for MSE. The result shows that the proposed protocol is distinctly secure and achieves all of the security requirements.

Table 4. The comparison of the attack resistance

	Li et al. [15]	Xue et al. [16]	Chuang et al. [18]	Mishra et al. [19]	Lu et al. [20]	Wang et al. [21]	Ours
P1	×	×	×	×	×	×	√
P2	×	×	√	×	√	√	√
P3	√	√	√	√	√	×	√
P4	×	×	√	√	√	√	√
P5	×	×	√	√	√	√	√
P6	√	×	√	√	√	×	√
P7	×	√	×	√	×	×	√
P8	√	√	×	√	×	×	√
P9	√	√	×	×	√	√	√
P10	√	×	√	√	√	√	√

√: Resist to the attack; ×: Vulnerable to the attack; P1: outsider attack; P2: replay attack; P3: modification attack; P4: stolen verifier attack; P5: off-line guessing attack; P6: insider attack; P7: stolen smart card attack; P8: user impersonation attack; P9: DoS attack; P10: server spoofing attack.

Table 5. The comparison of computational cost

	Registration	Login	Authentication	Total	Time(ms)
Li et al. [15]	$6T_H$	$6T_H$	$13T_H$	$25T_H$	5.0
Xue et al. [16]	$7T_H$	$6T_H$	$19T_H$	$31T_H$	6.4
Chuang et al. [18]	$3T_H$	$4T_H$	$13T_H$	$20T_H$	4.0
Mishra et al. [19]	$7T_H$	$6T_H$	$11T_H$	$24T_H$	4.8
Lu et al. [20]	$5T_H$	$5T_H$	$12T_H$	$22T_H$	4.4
Wang et al. [21]	$5T_H$	$4T_H$	$11T_H$	$20T_H$	4.0
Our proposed	$7T_H$	$5T_H$	$9T_H$	$21T_H$	4.2

7.2 Performance Anaylsis

The computational costs are compared. Table 5 itemizes a comparison of the computational spending of the protocol with the related previous protocols, where the definition of T_H is hash function's computational times. According to the results obtained in [31], T_H is less than 0.2 ms on average, in MSE (Core: 3.2 GHz, Memory: 3.0 G). Compared with Wang et al.'s protocol, the proposed protocol requires a slightly higher computational overhead, as the proposed scheme computes the one extra hash operations; however, the proposed scheme possesses all of the properties in terms of the security.

8 Conclusion

Recently, Wang et al. demonstrated the security weaknesses of Mishra et al., and presented a fuzzy extractor-based authentication protocol. They also asserted that their protocol is more secure and guarantees user anonymity; however, Wang et al.'s protocol was insecure to outsider and user impersonation attacks. To overcome these security weaknesses, the authors propose an improved fuzzy extractor-based authentication protocol for the multi-server environment that continues to have the merits of Wang et al.'s scheme. Furthermore, the proposed protocol comprises inclusive security properties. The formal and informal analysis of this paper make clear or explain why the proposed protocol is more secure.

Acknowledgments. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2010-0020210).

References

1. Zhang, X., Li, W., Zheng, Z.M., Guo, B.H.: Optimized statistical analysis of software trustworthiness attributes. Sci. China Inf. Sci. **55**(11), 2508–2520 (2012)
2. Lamport, L.: Password authentication with insecure communication. Commun. ACM **24**(11), 770–772 (1981)
3. Jeon, W., Kim, J., Nam, J., Lee, Y., Won, D.: An enhanced secure authentication scheme with anonymity for wireless environments. IEICE Trans. Commun. **95**(7), 2505–2508 (2012)
4. 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)
5. Sun, D.Z., Huai, J.P., Sun, J.Z., Li, J.X., Zhang, J.W., Feng, Z.Y.: Improvements of Juang's password authenticated key agreement scheme using smart cards. IEEE Trans. Ind. Electron. **56**(6), 2284–2291 (2009)
6. Khan, M.K., Zhang, J.: Improving the security of 'a flexible biometrics remote user authentication scheme'. Comput. Stand. Interfaces **29**(1), 82–85 (2007)
7. He, D., Kumar, N., Khan, M.K., Lee, J.H.: Anonymous two-factor authentication for consumer roaming service in global mobility networks. IEEE Trans. Consum. Electron. **59**(4), 811–817 (2013)
8. 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), 1–15 (2015)
9. Moon, J., Choi, Y., Kim, J., Won, D.: An improvement of robust and efficient biometrics based password authentication scheme for telecare medicine information systems using extended chaotic maps. J. Med. Syst. **40**(3), 1–11 (2016)
10. Lu, Y., Li, L., Peng, H., Yang, Y.: An enhanced biometric-based authentication scheme for telecare medicine information systems using elliptic curve cryptosystem. J. Med. Syst. **39**(3), 1–8 (2015)
11. Choi, Y., Nam, J., Lee, D., Kim, J., Jung, J., Won, D.: Security enhanced anonymous multi-server authenticated key agreement scheme using smart cards and biometrics. Sci. World J. Article ID 281305, 1–15 (2014)

12. Tsai, J.L.: Efficient multi-server authentication scheme based on one-way hash function without verification table. Comput. Secur. **27**(3–4), 115–121 (2008)
13. Liao, Y.P., Wang, S.S.: A secure dynamic ID based remote user authentication scheme for multi-server environment. Comput. Stand. Interfaces **31**(1), 24–29 (2009)
14. Hsiang, H.C., Shih, W.K.: Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment. Comput. Stand. Interfaces **31**(6), 1118–1123 (2009)
15. Li, X., Ma, J., Wang, W., Xiong, Y., Zhang, J.: A novel smart card and dynamic ID based remote user authentication scheme for multi-server environments. Math. Comput. Model. **58**(1–2), 85–95 (2013)
16. Xue, K.P., Hong, P.L., Ma, C.S.: A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server arahitecture. J. Comput. Syst. Sci. **80**(1), 195–206 (2013)
17. Lu, Y., Li, L., Peng, H., Yang, X., Yang, Y.: A lightweight ID based authentication and key agreement protocol for multi-server architecture. Int. J. Distrib. Sens. Netw. **11**(3), 1–9 (2015). 635890
18. 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)
19. Mishra, D., Das, A.K., Mukhopadhyay, S.: A secure user anonymity-preserving biometric-based multiserver authenticated key agreement scheme using smart cards. Expert Syst. Appl. **41**(18), 8129–8143 (2014)
20. Lu, Y., Li, L., Yang, X., Yang, Y.: Robust biometrics based authentication and key agreement scheme for multi-server environments using smart cards. PLoS ONE **10**(5), 1–13 (2015)
21. Wang, C., Zhang, X., Zheng, Z.: Cryptanalysis and improvement of a biometric-based multi-server authentication and key agreement scheme. PLoS ONE **11**(2), 1–25 (2016)
22. Dolev, D., Yao, A.C.: On the security of public key protocols. IEEE Trans. Inf. Theory **29**(2), 198–208 (1983)
23. Kocher, P., Jaffe, J., Jun, B., Rohatgi, P.: Introduction to differential power analysis. J. Cryptogr. Eng. **1**(1), 5–27 (2011)
24. Das, A.K.: A secure and effective biometric-based user authentication scheme for wireless sensor networks using smart card and fuzzy extractor. Int. J. Commun. Syst. **30**(1), 1–25 (2015)
25. Dodis, Y., Kanukurthi, B., Katz, J., Reyzin, L., Smith, A.: Robust fuzzy extractors and authenticated key agreement from close secrets. IEEE Trans. Inf. Theory **58**(9), 6207–6222 (2012)
26. Dodis, Y., Reyzin, L., Smith, A.: Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 523–540. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24676-3_31
27. RFC 4306: Internet key exchange (IKEv2) protocol (2005)
28. Das, A.K.: A secure and effective user authentication and privacy preserving protocol with smart cards for wireless communications. Netw. Sci. **2**(1–2), 12–27 (2013)
29. Das, A.K., Paul, N.R., Tripathy, L.: Cryptanalysis and improvement of an access control in user hieraRAhy based on elliptic curve cryptosystem. Inf. Sci. **209**, 80–92 (2012)

30. von Oheimb, D.: The high-level protocol specification language HLPSL developed in the EU project AVISPA. In: Proceedings of the Applied Semantics 2005 Workshop, Frauenchiemsee, Germany, pp. 1–17 (2005)
31. Xue, K., Hong, P.: Security improvement on an anonymous key agreement protocol based on chaotic maps. Commun. Nonlinear Sci. Numer. Simul. **17**(7), 2969–2977 (2012)