PLOS ONE

RESEARCH ARTICLE

Security enhanced multi-factor biometric authentication scheme using bio-hash function

Younsung Choi[1], Youngsook Lee[1], Jongho Moon[2], Dongho Won[2]*

1 Department of Cyber Security, Howon University, Impi-Myeon, Gunsan-Si, Jeonrabuk-Do 573-718, Korea, **2** Department of Computer Engineering, Sungkyunkwan University, 2066 Seoburo, Suwon, Gyeonggido 440-746, Korea

* dhwon@security.re.kr

Citation: Choi Y, Lee Y, Moon J, Won D (2017) Security enhanced multi-factor biometric authentication scheme using bio-hash function. PLoS ONE 12(5): e0176250. https://doi.org/10.1371/journal.pone.0176250

Editor: Muhammad Khurram Khan, King Saud University, SAUDI ARABIA

Received: January 26, 2017

Accepted: April 8, 2017

Published: May 1, 2017

Data Availability Statement: All relevant data are within the paper.

Funding: 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).

Competing interests: The authors have declared that no competing interests exist.

Abstract

With the rapid development of personal information and wireless communication technology, user authentication schemes have been crucial to ensure that wireless communications are secure. As such, various authentication schemes with multi-factor authentication have been proposed to improve the security of electronic communications. Multi-factor authentication involves the use of passwords, smart cards, and various biometrics to provide users with the utmost privacy and data protection. Cao and Ge analyzed various authentication schemes and found that Younghwa An's scheme was susceptible to a replay attack where an adversary masquerades as a legal server and a user masquerading attack where user anonymity is not provided, allowing an adversary to execute a password change process by intercepting the user's ID during login. Cao and Ge improved upon Younghwa An's scheme, but various security problems remained. This study demonstrates that Cao and Ge's scheme is susceptible to a biometric recognition error, slow wrong password detection, off-line password attack, user impersonation attack, ID guessing attack, a DoS attack, and that their scheme cannot provide session key agreement. Then, to address all weaknesses identified in Cao and Ge's scheme, this study proposes a security enhanced multi-factor biometric authentication scheme and provides a security analysis and formal analysis using Burrows-Abadi-Needham logic. Finally, the efficiency analysis reveals that the proposed scheme can protect against several possible types of attacks with only a slightly high computational cost.

Introduction

Distributed, networked system's allow users to efficiently access resources at their convenience. Web services such as on-line shopping and Internet banking have become common in today's technological world, and this has given rise to serious demand for remote authentication processes that ensure transactions between users and servers are secure. In various server environments, user authentication schemes are required to implemented elevated levels of

ownership. The first password-based scheme was introduced by Lamport in 1981, and since then, various studies have been carried out on the security, efficiency, and costs of authentication schemes. Existing remote authentication schemes are mainly implemented using a public key system, and in most cases, these can be divided into traditional certificate-based authentication schemes and identity-based authentication schemes according to the type of evidence they adopt for authentication. [1–9].

Various identity-based schemes have been proposed to provide secure, efficient, and practical authentication. One class is based on a pairing operation, which is practical but inefficient since a high computational cost is needed to carry out the pairing operation. The second is based on a particular hash function through which identity information is mapped to a point on an elliptic curve, resulting in a complicated structure. The third is a direct ID-based scheme that uses a general cryptographic hash function with a structure that is more simple than that of the second class. Due to this structure's simplicity, authentication can be accomplished only through a three-way handshake. However, it is still easy for a malicious person to cary out an attack. When all of the problems of the three categories mentioned above are taken into account, secure direct identity-based authentication schemes provide the optimum design for mobile device users and real-time applications. [10–20].

Recently, identity-based authentication schemes with a hash function were further divided into three categories according to the methods used in the authentication procedure: (1) knowledge-based scheme, (2) object-based scheme, and (3) biometrics-based scheme. However, each type has its own outstanding performance and limitations [21–37]:

- knowledge-based authentication is simple, convenient, and efficient, but it is weak to information leaks to malicious persons due to the adoption of a password,
- object-based authentication, based on the physical possession of a device such as a smart card, allows an adversary to impersonate legitimate users in a situation where the smart card is lost,
- biometrics-based authentication shows better results than the two types described above. The biometric keys, such as fingerprints or facial features, cannot be lost and forgotten. However, biometric samples, such as facial images, can be captured in various system databases, so biometric keys can remain insecure.

Multi-factor biometric authentication combines the use of a password, biometrics, and smart card protection to improve security and prevent various types of attacks, and it is not affected by the aforementioned defects. Such schemes have recently become a focal point of research, mainly reflected in the work put forward by various researchers. In 2010, Li and Hwang proposed a novel scheme using identity and a public key system, and then Das extended the work of Li *et al*. and made improvements to their weak scheme in 2011. Younghwa An showed that Das's proposed protocol failed to achieve mutual authentication for the server and user in 2012. However, Younghwa An allows for an adversary to masquerade as a legal server or as a user since mutual authentication is not provided. Cao and Ge attempted to improve on Younghwa An's scheme, but their scheme also has various security problems. We show that Cao and Ge's scheme is vulnerable to a biometric recognition error, slow wrong password detection, off-line password attack, user impersonation attack, ID guessing attack, a DoS attack, and also lacks session key agreement. This study then proposes a scheme to provide improved security by resolving the issues inherent to Cao and Ge's scheme [38–44].

The remainder of this paper is organized as follows. Section 2 briefly introduces related work on the bio-hash function and smart card information to help better understand the details of this paper. Section 3 briefly introduces Cao and Ge's scheme. Section 4 mainly

discusses its weaknesses. Section 5 describes countermeasures to solve its problems. Section 6 details the countermeasures to protect against all attacks. Section 7 is devoted to a formal security analysis of the modified scheme by using Burrows-Abadi-Needham logic (BAN-logic), and it compares the results of a security analysis and efficiency analysis with the modified scheme and some existing authentication schemes. The results indicate that the modified scheme has a slightly high computational cost and can protect against several possible attacks. Section 8 then concludes this paper.

Related works

In this section, the adversary's capability, bio-hash function and information for a smart card are explained to have a better understanding of the content of this paper.

Adversary's capability

In this paper, we assume the following about a probabilistic, polynomial-time adversary to properly capture the security requirements of a multi-factor biometric authentication scheme that uses smart cards during the registration phase, password change phase, and login and authentication phase [45].

- The adversary is able to have complete control over all message exchanges between the protocol participants, including a user and a server. That is, the adversary can intercept, insert, modify, delete, and eavesdrop on messages exchanged among the two parties at will.
- The adversary can (1) extract sensitive information from the smart card of a user through a power analysis attack or (2) determine the user's password, possibly via shoulder-surfing or by employing a malicious card reader. However, the adversary cannot compromise both the information of the smart card and the password of the user. It is otherwise clear that there is no way to prevent the adversary from impersonating the user if both factors have been compromised.

Bio-hash function

A hash function refers to a one-way transformation function. The hash function takes an arbitrary input and returns a string with a fixed size, which is referred to as a hash value or as a message digest.

Due to the peculiarity and ability of biometrics to differentiate a particular person from others, various systems have adopted methods to solve authentication and verification problems. However, a small change in biometric data (a little information missing from the biometric, noise, or a change in the order of the data input) may result in a momentous change in the hash value due to the uncertainty inherent to the retrieval of biometric features. In other words, general hash functions result in large differences due to slight differences in input, and recognition errors easily result from slight biometric changes. To resolve this problem, a bio-function system is proposed and studied. In various studies on bio-hashing systems, the bio-hash function must adhere to the following properties:

- similar biometric information should have similar hash values,
- different biometric information should not have similar hashes,
- rotation and translation of the original template should not have a substantial impact on hash values,

- partial biometric information (with missing core and delta) should be matched if sufficient detailed matters are present.

The hash function's certain class can be formulated to be everlasting to the order in which the input pattern is presented to the hash function, and such hash functions are known as bio-hash function or symmetric hash. So, the bio-hash function can resolve the recognition error of general hash function and can authenticate a legal user even if the user's biometric information changes a little [46, 47].

Smart card information

Various researchers have shown that physically monitoring the power consumption can extract confidential information stored in all smart cards, such as by using a simple power analysis and a differential power analysis. When a user forgets an own smart card, an adversary can analyze it and extract all information stored within. Variations of such schemes are weak to password acquisition attacks off-line where an adversary can be authenticated to the server without separately obtaining the user's information for login and authentication, such as their ID, password and biometrics. Therefore, the security-enhanced authentication scheme needs to be studied even if all the information of a user's smart card is revealed [48, 49].

Review of Cao and Ge's authentication scheme

The process for Cao and Ge's authentication scheme is reviewed before conducting the security analysis. Their scheme includes three phases: registration phase, password change phase, and login and authentication phase. The server S_i stores a secret value X_s and a user account database, which includes the legal user's authentication information [50]. For convenience, the notation used throughout this paper are summarized in Table 1.

Registration phase

This phase is the first to be performed once the U_i registers itself with the server S_i. Fig 1 describes the registration phase for Cao and Ge's scheme.

(R1) U_i selects ID_i, PW_i and imprints its own B_i, and generates K. Then, U_i sends the identity ID_i, password information $(PW_i \oplus K)$, and biometric information $(B_i \oplus K)$ to the server S_i by using a secure channel.

(R2) S_i computes $f_i = h(B_i \oplus K)$, $r_i = h(PW_i \oplus K) \oplus f_i$, and $e_i = h(ID_i||X_s) \oplus r_i$.

(R3) S_i creates an entry for user ID_i and stores n_i on this entry in database. Then, S_i computes $EID_i = h(ID_i)||n_i$ and stores EID_i to the entry.

Table 1. Notation.

Notation	Description	Notation	Description
U_i	User	B_i	U_i's biometric template
S_i	Server	$h(\cdot)$	General hash function
ID_i	User's identity	$H(\cdot)$	Bio-hash function
PW_i	User's password	n_i	Counter number
R_c	A random number generated by U_i	$\oplus$	Bitwise XOR operation
R_s	A random number generated by S_i	$\|\|$	Concatenation operation
X_s	Secret key generated by S_i	T_i	ith timestamp

https://doi.org/10.1371/journal.pone.0176250.t001

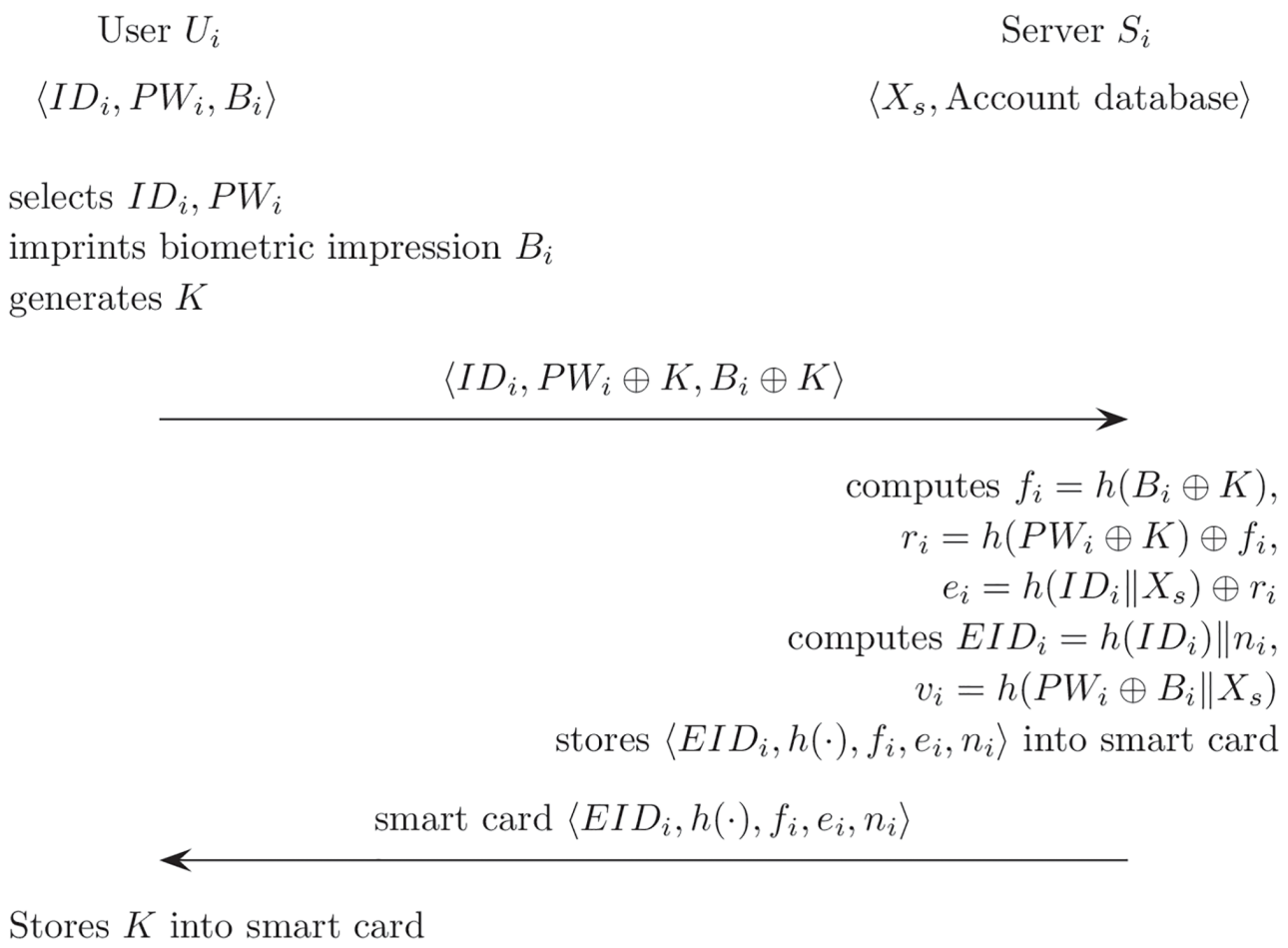

Fig 1. Registration phase for Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g001

(R4) S_i computes $v_i = h(PW_i \oplus B_i \| X_s)$.

(R5) S_i sends a smart card to U_i. It contains $\langle EID_i, h(\cdot), f_i, e_i, n_i \rangle$ using a secure channel. Then U_i stores K in the smart card.

Password change phase

The password change phase is carried out when U_i wants to change the password or the smart card is lost. Fig 2 describes the password change phase on Cao and Ge's scheme.

(RR1) U_i submits the ID_i to S_i, password information $(PW_i \oplus K')$, and biometric information $(B_i \oplus K')$ via a secure channel, K' is the new random number.

(RR2) S_i computes $v_i' = h(h(PW_i) \oplus h(B_i) \oplus X_s)$ and compares v_i' with v_i in the account database. If they are not the same, this phase is terminated.

(RR3) Otherwise, S_i computes $n_{inew} = n_i + 1$. Then, S_i performs the following computations; $f_{inew} = h(B_i \oplus K')$, $r_{inew} = h(PW_i \oplus K') \oplus f_{inew}$, $e_{inew} = h(ID_i \oplus X_s) \oplus r_{inew}$.

(RR4) S_i sends U_i a new smart card that contains $\langle EID_i, h(\cdot), f_{inew}, e_{inew}, n_{inew} \rangle$ by using secure channel. Then U_i stores the random number K' in the smart card.

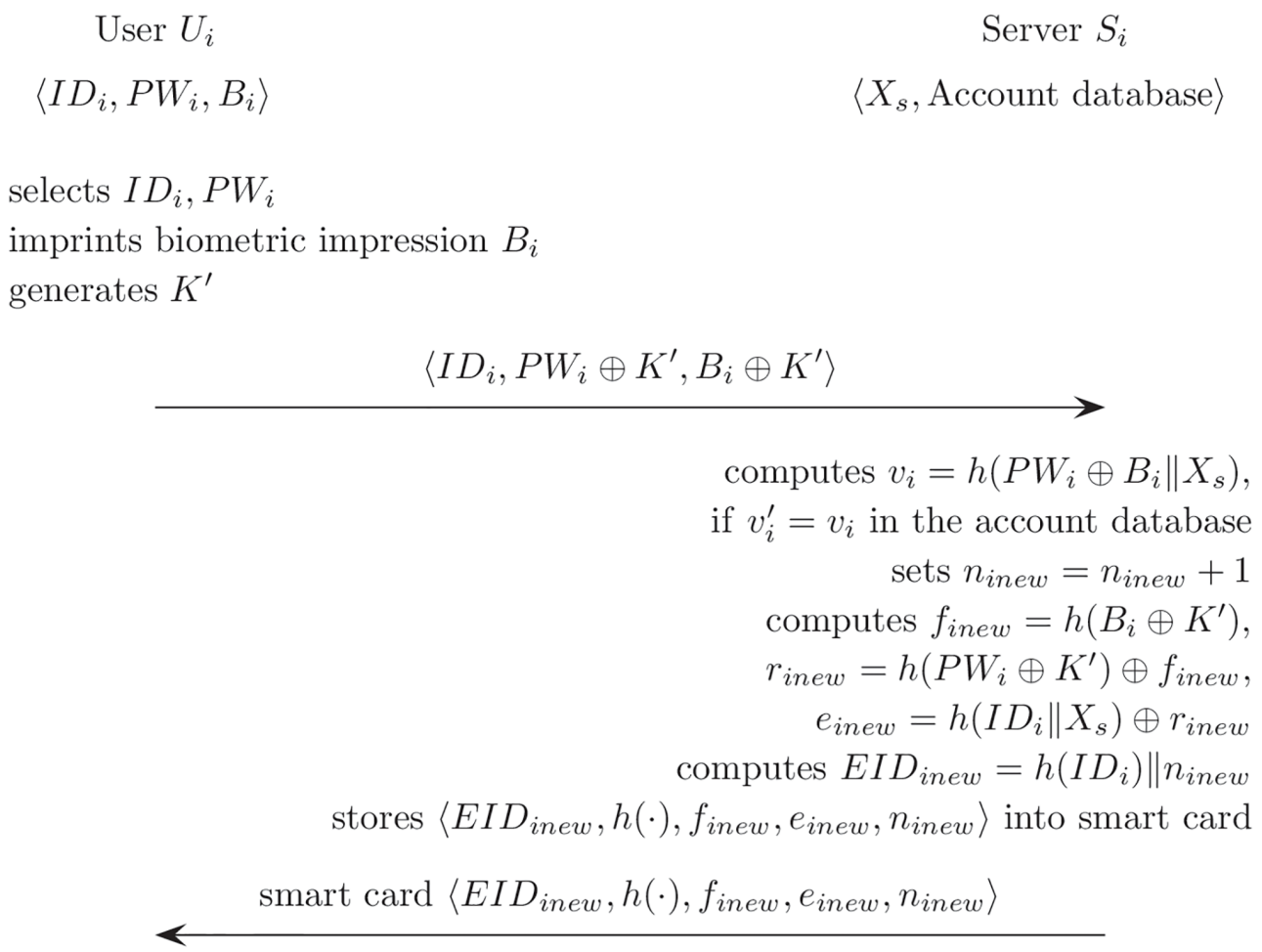

Fig 2. Password change phase on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g002

Login and authentication phase

U_i executes the following steps when U_i wants to authenticate remote S_i. Fig 3 describes the login and authentication phase on Cao and Ge's scheme.

(L1) U_i imprints B_i using a biological feature extraction device, and it computes the information $h(B_i \oplus K)$ using K stored in the smart card. U_i can proceed only if $h(B_i \oplus K)$ matches f_i.

(L2) U_i inputs the ID_i and PW_i and then, the smart card computes

$$\begin{aligned} r_i &= h(PW_i \oplus K) \oplus f_i, \\ M_1 &= e_i \oplus r_i, M_2 = M_1 \oplus R_c \\ M_3 &= h(M_1 \parallel R_c), EID_i = h(ID_i) \parallel n_i. \end{aligned}$$

(L3) The login request message $\langle EID_i, M_2, M_3 \rangle$ is then sent from U_i to S_i.

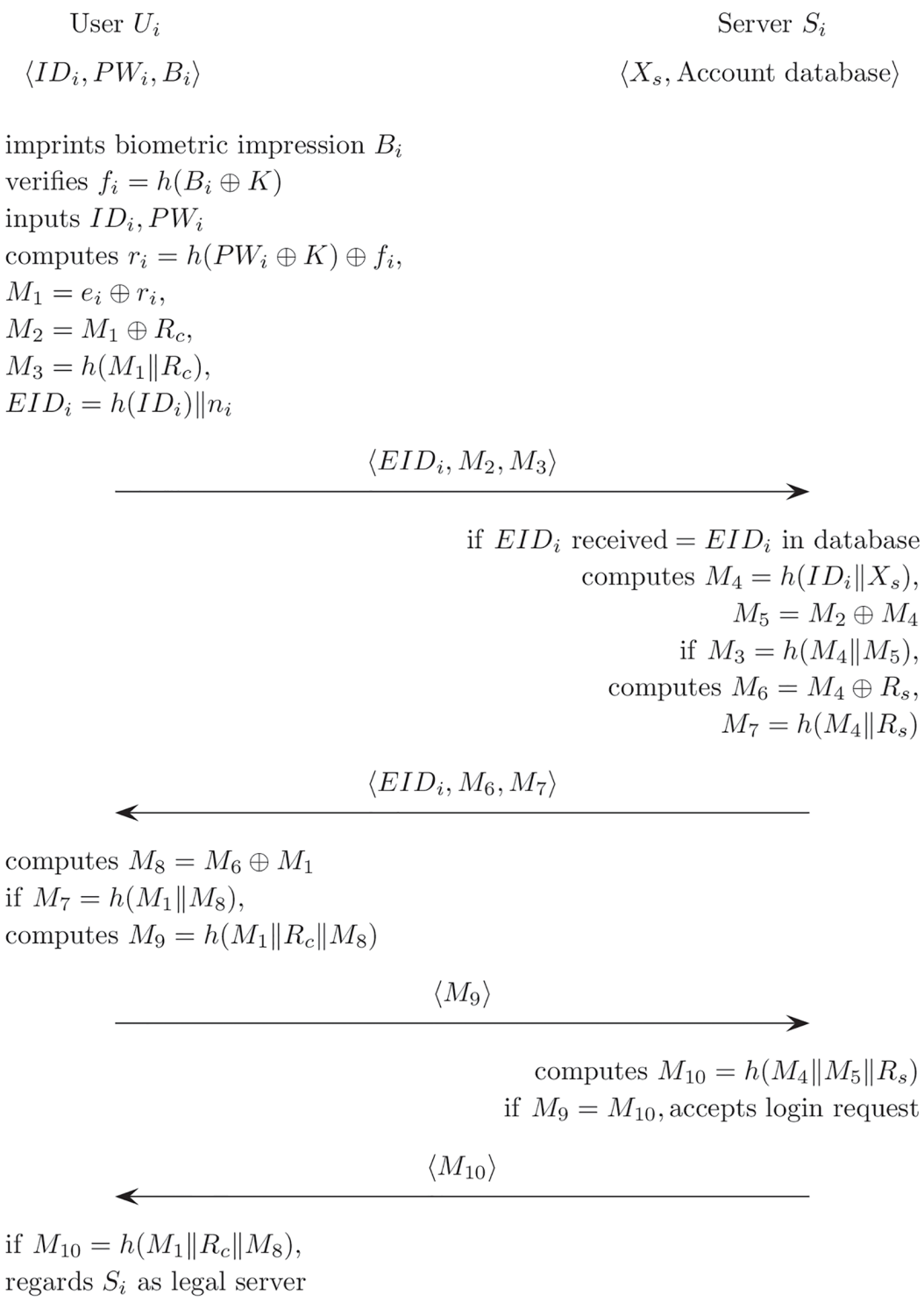

Fig 3. Login and authentication phase for Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g003

The server S_i executes the authentication phase when the message is received.

(A1) S_i makes sure that EID_i satisfies the original format using the database entry and checks the ID_i for the authentication phase.

(A2) If the ID_i is valid when compared with database of S_i, S_i computes

$$M_4 = h(ID_i \parallel X_s), M_5 = M_2 \oplus M_4.$$

(A3) If M_3 is the same as $h(M_4\|M_5)$, S_i computes

$$\begin{aligned} M_6 &= M_4 \oplus R_s, \\ M_7 &= h(M_4 \parallel R_s). \end{aligned}$$

Then, S_i sends the message $\langle M_6, M_7 \rangle$ to U_i.

(A4) U_i computes M_8 and verifies whether $M_7 = h(M_1\|M_8)$ or not. If they are equal, U_i calculates M_9.

$$\begin{aligned} M_8 &= M_6 \oplus M_1, \\ M_9 &= h(M_1 \parallel R_c \parallel M_8). \end{aligned}$$

(A5) U_i sends the message $\langle M_9 \rangle$ to S_i.

(A6) After receiving $\langle M_9 \rangle$, S_i makes sure that M_9 is equal to $M_{10} = h(M_4\|M_5\|R_s)$ and then accepts the user's login request. S_i sends M_{10} to U_i.

$$M_{10} = h(M_4 \parallel M_5 \parallel R_s)$$

(A7) Upon receiving $\langle M_{10} \rangle$, U_i makes sure that M_{10} is equal to $h(M_1\|R_c\|M_8)$ and then regards S_i as a legal server.

$$M_{10} = h(M_1 \parallel R_c \parallel M_8)$$

Cryptanalysis of Cao and Ge's authentication scheme

We analyze Cao and Ge's authentication scheme and identify various security vulnerabilities, including a biometric recognition error, slow wrong password detection, off-line password attack, user impersonation attack, ID guessing attack, DoS attack, and a lack of session key agreement.

Biometric recognition error

Cao and Ge's authentication scheme only uses a general hash function to provide checking biometrics. However, the hash function has a property that causes a slight difference in the input data to result in a very large difference in the output data. Fig 4 describes the biometric recognition error in Cao and Ge's scheme. The output of the imprinted biometrics is not always constant, so biometrics generally have instances of false acceptance and false rejection. Therefore, even when U_i imprints biometrics in the device, it is possible to output a different

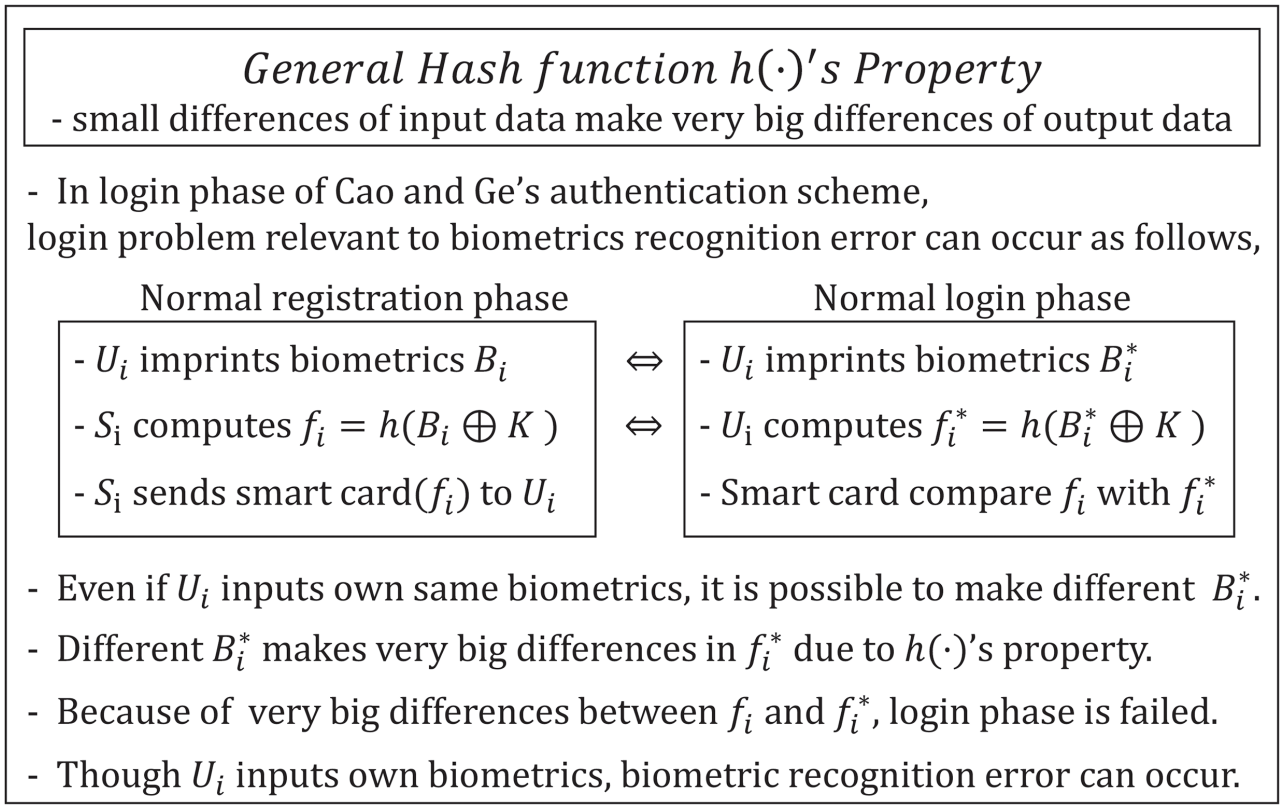

Fig 4. Biometric recognition error on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g004

B_i^*. Therefore, the same user can generate a different output, such as that with B_i during the registration phase and B_i^* during the login phase. The differences between B_i and B_i^* can result in big differences in f_i and f_i^*, and this difference between f_i and f_i^* results in a biometric recognition error in the login phase. Therefore, a normal user does not pass the user biometric verification stage because the smart card compares the computed f_i^* to f_i, which is stored within the smart card. Therefore, even though U_i imprints his/her own biometrics, a biometric recognition error can occur. Thus, the smart card needs to be implemented using more advanced techniques, such as a bio-hash function, to improve the biometrics verification process [51].

Slow wrong password detection

Slow wrong password detection refers to instances in which the user cannot know of a mistake immediately when inputing the wrong password, and the user can know when server S_i notifies there is a wrong user password. In Cao and Ge's authentication scheme, the user's smart card cannot verify the accuracy of the user password during the login phase. Only S_i verifies a legal user by comparing the similarities between M_3 and $h(M_4 \| M_5)$ during authentication phase. Fig 5 specifically describes how slowly the wrong password is detected in Cao and Ge's scheme. Concretely, U_i inputs ID_i and PW_i after the biometric verification, then if U_i selects a wrong password PW_i^*, the smart card is unaware that the password is incorrect. The smart card does not check the PW_i^*, and it only computes various values $\langle r_i^*, M_1^*, M_2^*, M_3^*, EID_i \rangle$ using PW_i^* for login and authentication. The smart card then sends $\langle EID_i, M_2^*, M_3^* \rangle$.

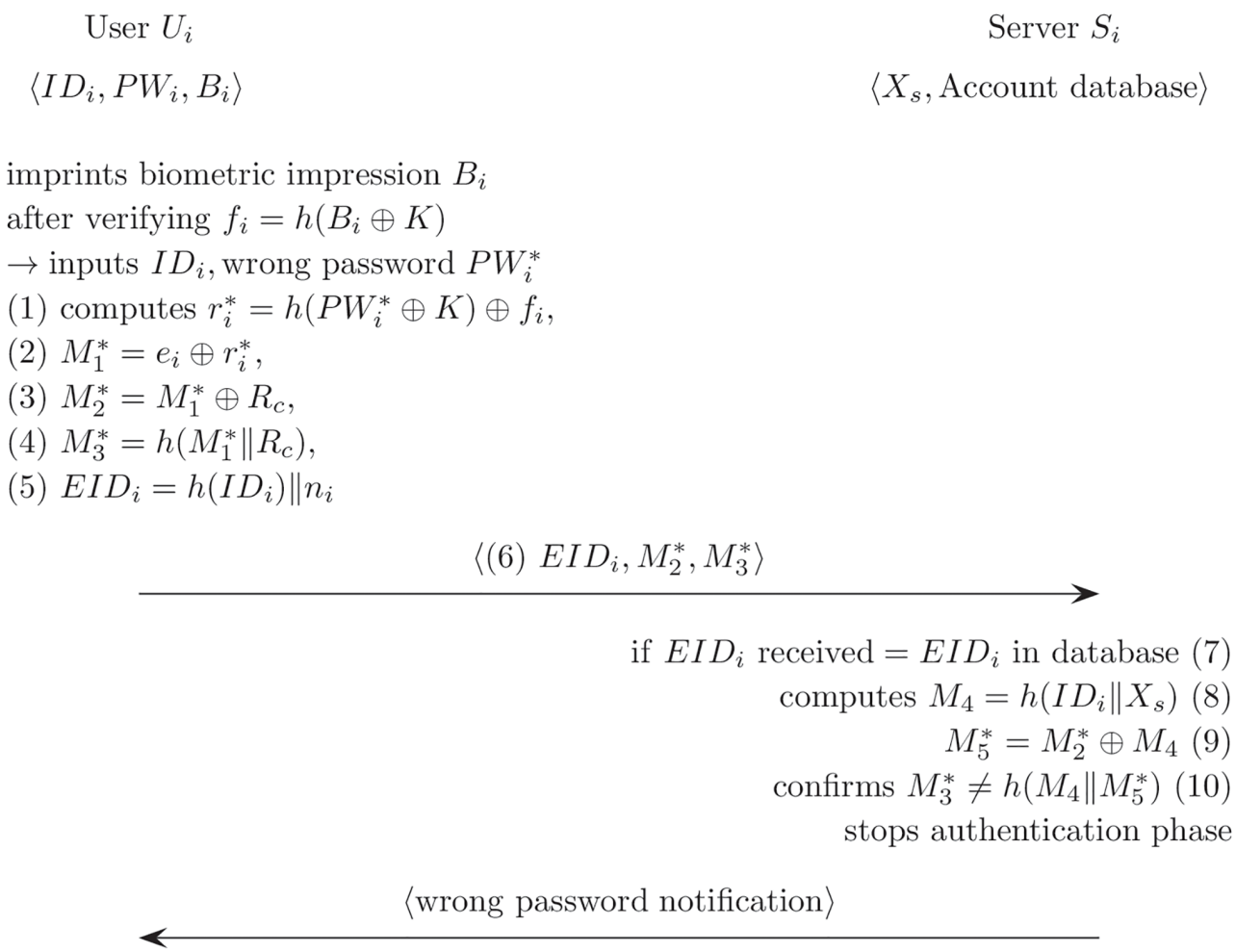

Fig 5. Slow wrong password detection on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g005

S_i is unable to immediately confirm the wrong password after receiving the messages $\langle EID_i, M_2^*, M_3^* \rangle$. First, S_i verifies the received EID_i using EID_i in the database, and then computes $M_4 = h(ID_i \| X_s)$ and $M_5^* = M_2^* \oplus M_4$. Then, because M_3^* is same as $h(M_4 \parallel M_5^*)$, S_i eventually confirms that the received messages are not normal, and maybe U_i could have input the wrong password. Basically, S_i sends the wrong password notification to U_i. In detail, Cao and Ge's scheme requires a lengthy phase that includes value computation and message transmission before confirming that the user input the wrong password. Therefore, a smart card is needed to provide a fast wrong password detection technique during login. When U_i inputs the wrong password during the login phase, the smart card needs to quickly identify the incorrect password and should immediately notify U_i of the mistake.

Off-line password attack

In Cao and Ge's scheme, an adversary can compute the user's password by using public messages and the user's smart card, obtaining M_2 and M_3 from public messages between the user and the server. Fig 6 provides a detailed description of the off-line password attack for Cao and Ge's scheme. Kocher *et al.* and Messerges *et al.* claim that the all confidential information that

- *Adversary* got M_2 and M_3 in precious public communication.
- *Adversary* acquires e_i, f_i, K and $h(\cdot)$ in stolen user smart card.
- *Adversary* knows the formula of all values used in this scheme.

$\rightarrow M_1 = e_i \oplus r_i, M_2 = M_1 \oplus R_c$ and $M_3 = h(M_1 || R_C)$

$\rightarrow$ Due to $R_c = M_1 \oplus M_2$, so M_3 is expressed as follows,

$\Rightarrow M_3 = h(e_i \oplus r_i || M_1 \oplus M_2) \Rightarrow M_3 = h(e_i \oplus r_i || e_i \oplus r_i \oplus M_2)$

$\rightarrow$ Due to $r_i = h(PW_i \oplus K) \oplus f_i$, so M_3 is expressed as follows,

$$\Rightarrow \boldsymbol{M_3 = h(e_i \oplus h(PW_i \oplus K) \oplus f_i \,||\, e_i \oplus h(PW_i \oplus K) \oplus f_i \oplus M_2)}$$

- In this formula, *adversary* already knows all values except PW_i.
- Due to password has low entropy, *adversary* can compute PW_i.

Fig 6. Off-line password attack on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g006

is generally stored in smart cards could be extracted through various forms, such as monitoring the power consumption. Therefore, if a user loses a smart card, all of the information in the smart card can be revealed by an adversary. The smart card stores various types of information, including user login and authentication, so the adversary can acquire the e_i, f_i, K, and hash function $h(\cdot)$ values from the user's smart card. The adversary knows the formula for all values used in Cao and Ge's scheme as follows:

$$M_1 = e_i \oplus r_i \quad , \quad M_2 = M_1 \oplus R_c$$
$$M_3 = h(M_1 \,\|\, R_c) \quad , \quad r_i = h(PW_i \oplus K) \oplus f_i.$$

The adversary uses the determined values, messages, and formula to compute the M_3 formula, as follows:

$$M_3 \quad = \quad h(e_i \oplus h(PW_i \oplus f_i) \oplus f_i \,\|\, e_i \oplus h(PW_i \oplus K) \oplus f_i \oplus M_2).$$

The adversary then knows all values in this formula, except for PW_i. Therefore, the adversary can easily determine the user's password PW_i by mounting an off-line password guessing attack because the password PW_i is not long enough and has a low level of entropy. If the adversary knows the PW_i, various attacks can be facilitated by using the user's password. Therefore, the password needs to be protected by using other values that are not stored in the smart the card with a high entropy, such as biometric information [52].

User impersonation attack

In Cao and Ge's scheme, an adversary can be authenticated with the server by using the user's smart card and the password without access to the user's biometric information. Fig 7

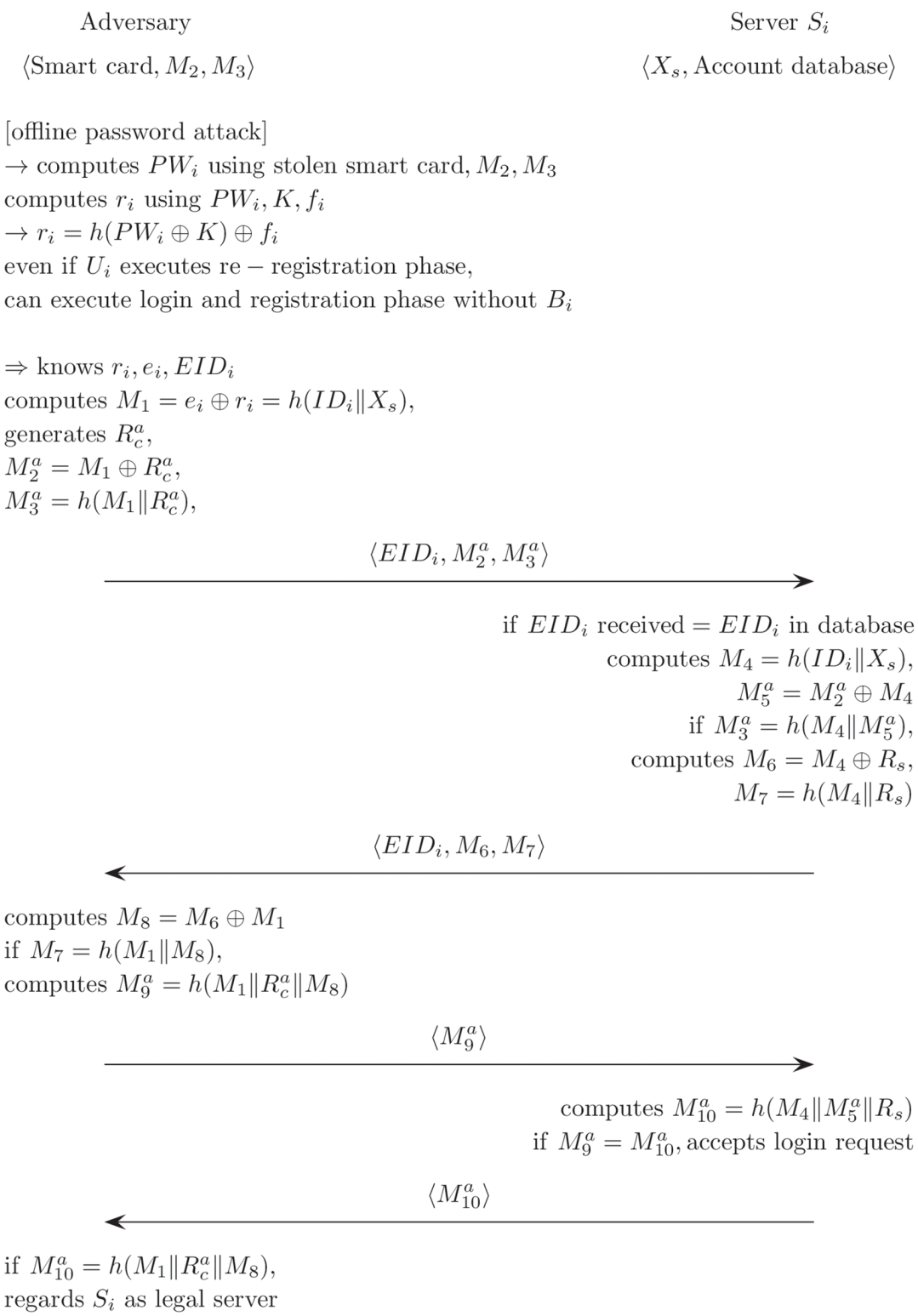

Fig 7. User impersonation attack on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g007

describes in detail a user impersonation attack for Cao and Ge's authentication scheme. In further detail, when an adversary obtains or steals a user's smart card and figures out the user's password, the legitimate user can be easily impersonated. In section 1, an adversary is shown to compute the user's password by using a smart card and public messages. Therefore, this scheme is critically deficient in that the adversary can be authenticated by the server without the user's biometrics.

As described in Fig 6, the adversary can illegally extract all values including K_i, f_i, e_i, and EID_i from the user's smart card by monitoring the power consumption. It then computes PW_i using an off-line password attack computing r_i using PW_i, K_i, f_i as follows:

$$r_i = h(PW_i \oplus K) \oplus f_i$$

Even if U_i successfully executes the password change process, the adversary can still use these to impersonate a legal user, authenticate S_i without knowing the B_i values, and then compute normal authentication messages EID_i, M_2^a, M_3^a using r_i, e_i, EID_i as follows:

$$\begin{aligned} M_1 = e_i \oplus r_i &= h(ID_i \parallel X_s), \text{ generates } R_c^a \\ M_2^a &= M_1 \oplus R_c^a, \\ M_3^a &= h(M_1 \parallel R_c^a). \end{aligned}$$

After S_i receives the messages EID_i, M_2^a, and M_3^a, then, S_i checks the legitimacy of the messages. However, S_i cannot distinguish between a normal M_9 and an abnormal M_9 because the adversary used accurate values like $h(ID_i||X_s)$, but the adversary normally computes $h(ID_i||X_s)$ using r_i, e_i.

Then, S_i sends the authentication messages $\langle EID_i, M_6, M_7 \rangle$ for U_i. These are then used by the adversary to compute the next authentication message M_9^a for S_i as follows,

$$\begin{aligned} M_8 &= M_6 \oplus M_1, \\ \text{if } M_7 &= h(M_1 \parallel M_8), \\ M_9^a &= h(M_1 \parallel R_c^a \parallel M_8). \end{aligned}$$

Next, S_i checks that the received M_9^a is the same as $M_{10}^a = h(M_4 \parallel M_5^a \parallel R_s)$. However, S_i cannot distinguish it from a normal M_9 because the adversary uses accurate values like M_1 $h(ID_i||X_s)$ and R_c^a, which is used for $\langle EID_i, M_2^a, M_3^a \rangle$. Then, S_i accepts the login request for the adversary.

The adversary can be authenticated at S_i because he determined EID_i, e_i and r_i through an off-line password attack, so S_i cannot distinguish between the adversary and a legitimate user. Since the user's biometric information is not used during the login and authentication phase, S_i authenticates the adversary as a normal user. S_i cannot store and check the password and biometric information during the login and authentication phase due to the user's privacy. Thus, to solve this problem, it is necessary to modify the way in which the authentication values $h(ID_i||X_s)$ are computed for the user. This value cannot be stored on the smart card, and it can only be computed by a legitimate user when the user simultaneously inputs the password and biometrics during the login and authentication phase.

ID guessing attack

Cao and Ge's authentication scheme uses EID to protect the user's ID_i in order to ensure user anonymity during public communication. However, the adversary can determine the user's ID_i by using the user's smart card and the public communication message EID_i. Fig 8 describes in detail how to compute the user's ID_i for Cao and Ge's authentication scheme.

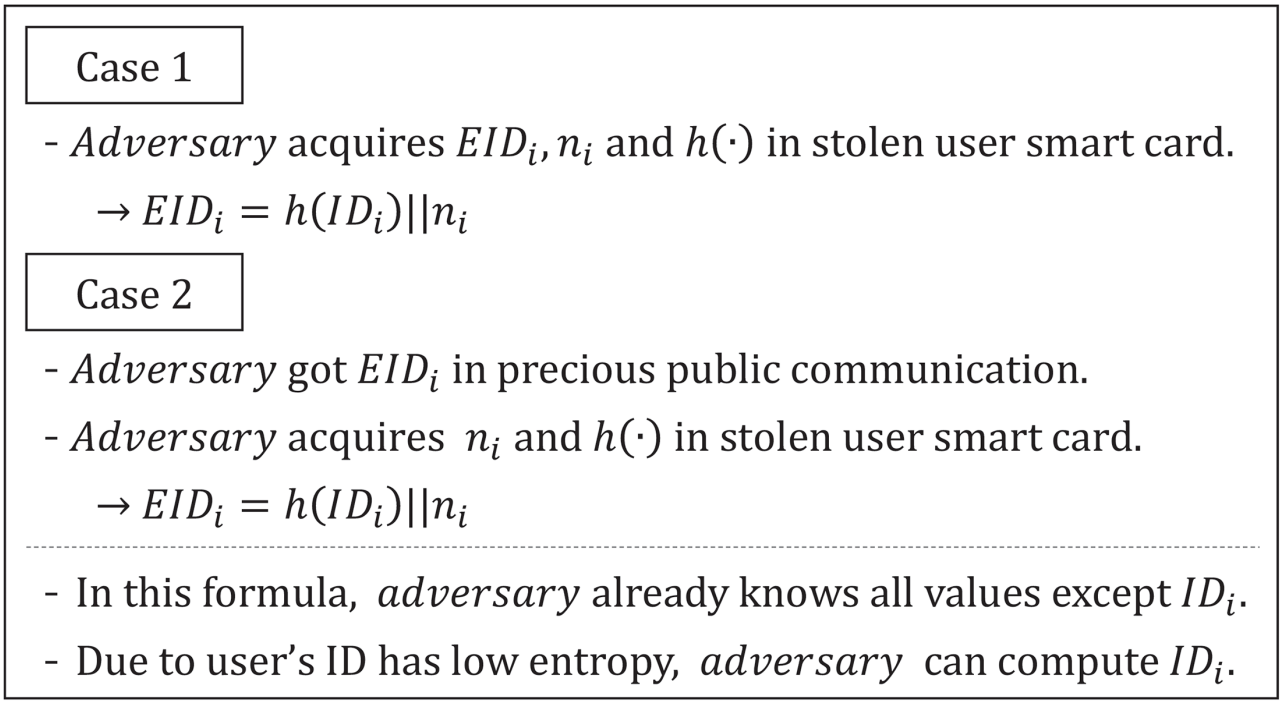

Fig 8. ID guessing attack on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g008

When an adversary obtains or steals a user's smart card, he can extract EID_i, n_i and $h(\cdot)$. Then, the adversary can compute the user ID_i from the formula $EID = h(ID)||n_i$ because he knows all values except for the ID_i. In general, a user ID_i has a low entropy so the adversary is able to easily compute the user ID_i. Basically, if an adversary fails to extract EID_i from the smart card, he can acquire EID from public communication. Therefore, even though the adversary extracts n_i and $h(\cdot)$ from the user's smart card, he can determine the ID_i from $EID = h(ID)||n_i$. The user's ID_i can be used for another attack, and therefore, the user's ID_i needs to be protected using another value that the adversary cannot determine from the user's smart card or from public communication.

Vulnerability to a DoS attack

A DoS attack is such where an adversary attempts to make a server or network resource become unavailable to prevent legitimate users from accessing the normal service. Although there are various ways to accomplish a DoS attack, the server's system or configuration have to prepare for defenses against it. However, in Cao and Ge's scheme, an adversary can execute a DoS attack without difficulty. Fig 9 describes the DoS attack for Cao and Ge's authentication scheme.

An adversary can collect the previous messages $\langle EID_{pi}, M_{p2}, M_{p3}\rangle$ from a legitimate user U_i and a server S_i. Then, the adversary sends the messages to S_i without modification. The S_i unavoidably executes all operations of (2) and sends the (3) messages $\langle EID_{pi}, M_6, M_7\rangle$ to the U_i. This is the reason why S_i cannot verify the freshness of the (1) messages $\langle EID_{pi}, M_{p2}, M_{p3}\rangle$. This operation involves the generation of a random nonce once, executing the hash function twice, calculating the exclusive-or operation twice, conducting the similarities checking function twice, and then, sending (3) messages $\langle EID_{pi}, M_6, M_7\rangle$.

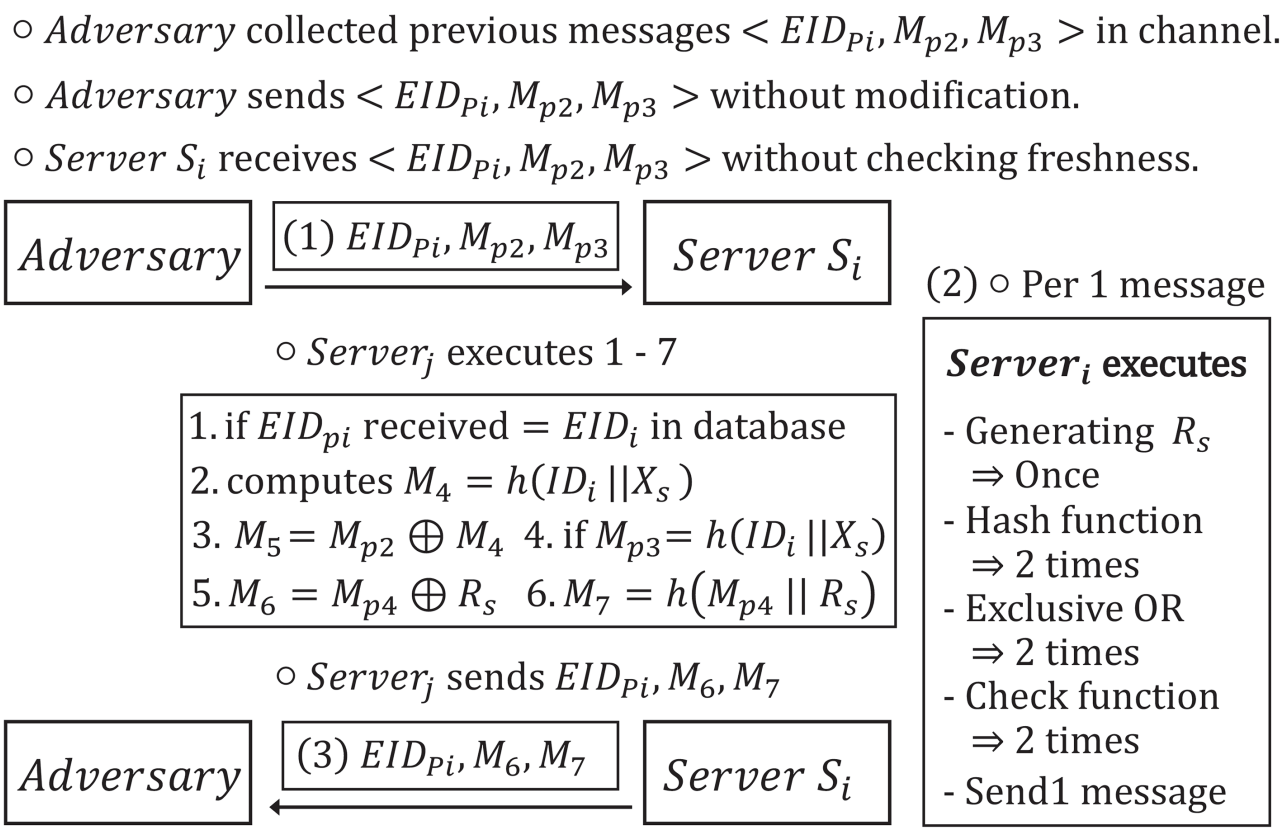

Fig 9. Vulnerability to a DoS attack on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g009

Therefore, the adversary can easily attempt to carry out a DoS attack targeting the server to see if he can obtain an intercepted number from a previous messages. Cao and Ge's scheme does not check the freshness of an authentication message. Therefore, when an adversary sends previous authentication messages to S_i, S_i cannot verify whether the received messages are current or not, and S_i is obligated to execute various operations. In order to defend against a DoS attack, this scheme needs to check the freshness of the messages by considering the timestamps.

Lack of session key agreement

In general, the session key refers to a symmetric key that is used to encrypt all messages in the communication session. Therefore, it can be computed and used for secure communications among communication members after successfully finishing the authentication phase. Fig 10 describes in detail the lack of session key agreement for Cao and Ge's authentication scheme. As described in Fig 10, U_i and S_i finally authenticate each other using M_9 and M_{10}, and then they are accepted and regarded to be legal members. However, secure communication between M_9 and M_{10} is not provided because these do not have a session key after all phases have finished. Therefore, it is necessary to modify the login and authentication phase to provide session key agreement. Moreover, to ensure the security of the scheme, the session key has to be changed for each session and must be secured against various forms of attack.

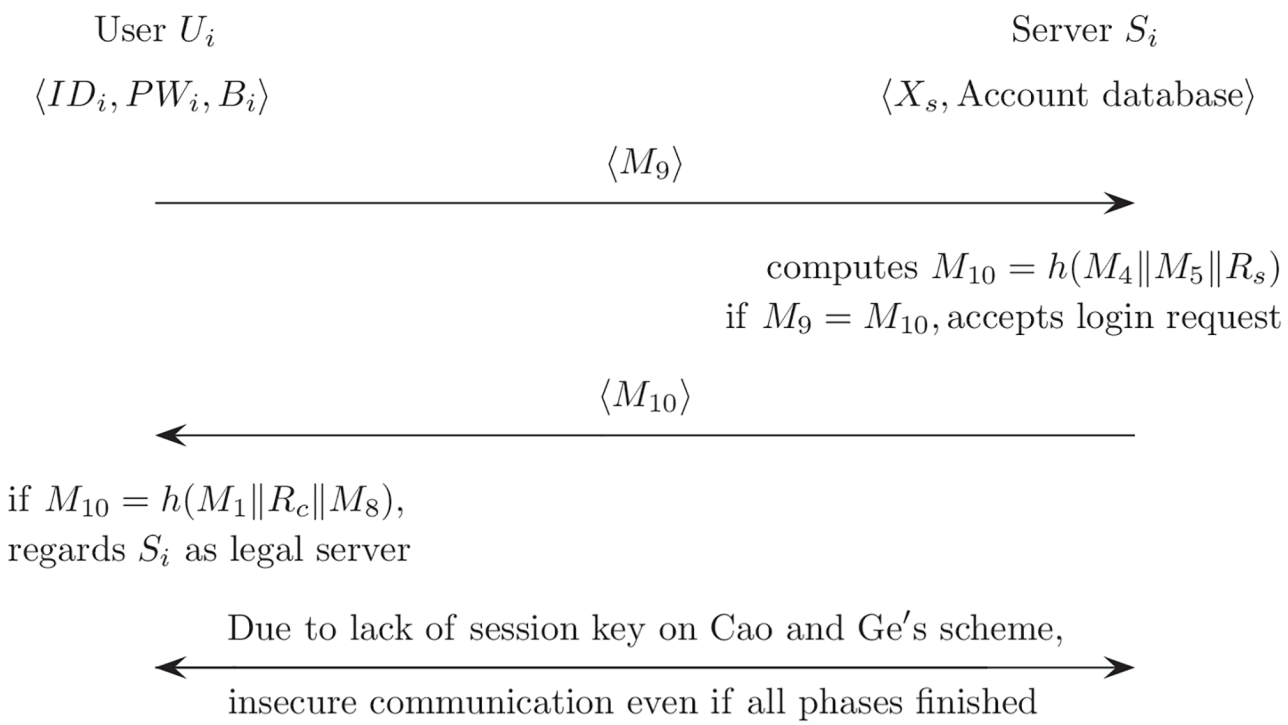

Fig 10. Lack of session key agreement on Cao and Ge's authentication scheme.

https://doi.org/10.1371/journal.pone.0176250.g010

Countermeasures

The reason why Cao and Ge's scheme is vulnerable to the biometric recognition errors is that,

- even if the same user inputs his/her own biometrics to a scanner device, this device can generate slightly different outputs due to the general characteristics of the biometric information;
- the general hash function produces very large differences in the output data from slight differences in the input data.

Thus, a general hash function results in a legal user failing during the login phase when using his/her own biometrics. To prevent a biometric recognition error, we suggest modifying the registration phase from $\langle ID_i, PW_i \oplus K, B_i \oplus K \rangle$ to

$$\langle ID_i, h(PW_i) \oplus K, H(B_i) \oplus K \rangle$$

$H(\cdot)$ is a bio-hash function that produces consistent output for the same biometric information, even if the user's biometric input is slightly different. So, during the login phase, the values need to be modified from $f_i = h(B_i \oplus K)$ to

$$f_i = h(H(B_i) \oplus K)$$

However, by only modifying the scheme to use a bio-hash function, Cao and Ge's authentication scheme is still vulnerable to the slow detection of a wrong password. This type of problem results from,

- the smart card not checking the user's password during the login phase;

- the server can confirm whether a user inputs the wrong password and computes the wrong M_3 during the authentication phase only after extensive computations;

Adding a password verification step during the login phase is suggested to solve the slow wrong password detection problem. Thus, the computations are modified for f_i from f_i from $f_i = h(H(B_i) \oplus K)$ to

$$f_i = h(ID_i \oplus h(PW_i) \oplus H(B_i))$$

However, even with the f_i modified above, an off-line password attack can still be carried out. This vulnerability is due to the fact that;

- an adversary can know and compute all formulas and values except for PW_i;
- it is necessary to check PW_i with values, which the adversary cannot know and compute, such as $H(B_i)$;

Since we check the user's password in f_i, we suggest modifying r_i from $r_i = h(PW_i \oplus K) \oplus f_i$ to

$$r_i = h(H(B_i) \oplus K) \oplus f_i$$

With such a modification, we can also defend against a user impersonation attack because the adversary cannot impersonate the user without the user's password. In other words, the adversary cannot compute r_i without PW_i and then figure out $h(ID_i||X_s)$ to conduct a user impersonation attack due to the lack of a legal M_1.

Next, the possible mechanism to eliminate the vulnerability in Cao and Ge's scheme for an ID guessing attack is presented. This vulnerability is due to the fact that,

- the adversary can obtain the user's ID_i from EID_i using the value n_i stored in the user's smart card.
- Even if EID_i is a public communication message, Cao and Ge's scheme does not provide sufficient protection for EID_i.

To address to the problem on ID guessing attack, we suggest modifying EID_i from $EID_i = h(ID_i)||n_i$ to

$$EID_i = h(ID_i \parallel h(ID_i \parallel X_s) \parallel n_i)$$

$h(ID_i||X_s)$ is not stored in a smart card, and it can be easily computed by S_i. Even if the adversary knows EID_i and n_i, he cannot compute ID_i from EID_i due to the ignorance on $h(ID_i||X_s)$.

However, with the modifications explained above, Cao and Ge's scheme is still vulnerable to a DoS attack. The cause for this vulnerability on DoS attacks is that.

- U_i and S_i perform all operations without checking the freshness of the received authentication messages.
- Moreover, S_i unwillingly executes extensive computations per message before S_i discovers the fault of the received authentication message.

To address the vulnerability of the DoS attack, we suggest using timestamps (T_1, T_2, T_3, T_4) and adding them to the authentication messages. So we propose to modify the computations for M_3, M_3, M_3, and M_{10} from $M_3 = h(M_1||R_c)$, $M_7 = h(M_4||R_s)$, $M_9 = h(M_1||R_c||M_8)$,

$M_{10} = h(M_4 \| M_5 \| R_s)$ to

$$\begin{aligned} M_3 &= h(M_1 \| R_c \| T_1), \\ M_7 &= h(M_4 \| R_s \| T_2), \\ M_9 &= h(M_1 \| R_c \| M_8 \| T_3), \\ M_{10} &= h(M_4 \| M_5 \| R_s \| T_4). \end{aligned}$$

In advance, all transmission messages need to include timestamps to check the freshness, such as from $\langle EID_i, M_2, M_3 \rangle$ to

$$\langle EID_i, M_2, M_3, T_1 \rangle$$

T_1 and M_3 are thus computed by a legal user, and the adversary cannot compute M_3 without T_1, which is current and matched with M_3. So S_i can check the message freshness using T_1, and S_i can verify the the message integrity and freshness by easily checking $M_3 = h(M_1 \| R_c \| T_1)$. In this manner, it is possibly to effectively prevent the DoS attack.

Finally, the problem regarding a lack of a session key is resolved by adding a session key agreement during the login and authentication phase. The session key needs to change for every session in order to enhance the security of the authentication scheme, so computing the session key agreement is proposed as follows;

$$sk = h(h(ID_i \| X_s) \| R_c \| R_s | T_3 \| T_4)$$

For the session key agreement, $h(ID_i \| X_s)$, R_c and R_S are computed only by the legal user and the server. T_3 and T_4 can be used to confirm the freshness of the session key. Therefore, this session key can change every session and can prevent various attacks.

Security enhanced multi-factor biometric authentication scheme

To solve the problems inherent to Cao and Ge's scheme, a security enhanced multi-factor biometric authentication scheme is proposed and divided into three phases: registration phase, password change phase, and login and authentication phase. Before our scheme is executed, S_i generates the server's secure value X_s for security.

Registration phase

The registration phase of the proposed scheme is described in Fig 11. U_i needs to perform the registration phase with S_i by using a secure channel.

(R1) U_i selects ID_i, PW_i; imprints the biometric impression B_i; and generates K. U_i sends the identity ID_i, $h(PW_i) \oplus K$ using the general hash function, and $H(B_i) \oplus K$ using bio-hash function to S_i through a secure channel.

(R2) After receiving these, S_i computes f_i, r_i, and e_i as follows;

$$\begin{aligned} f_i &= h(ID_i \oplus h(PW_i) \oplus H(B_i)), \\ r_i &= h(H(B_i) \oplus K) \oplus f_i, \\ e_i &= h(ID_i \| X_s) \oplus r_i. \end{aligned}$$

(R3) Then, S_i creates an entry of database for the user ID_i and generates n_i.

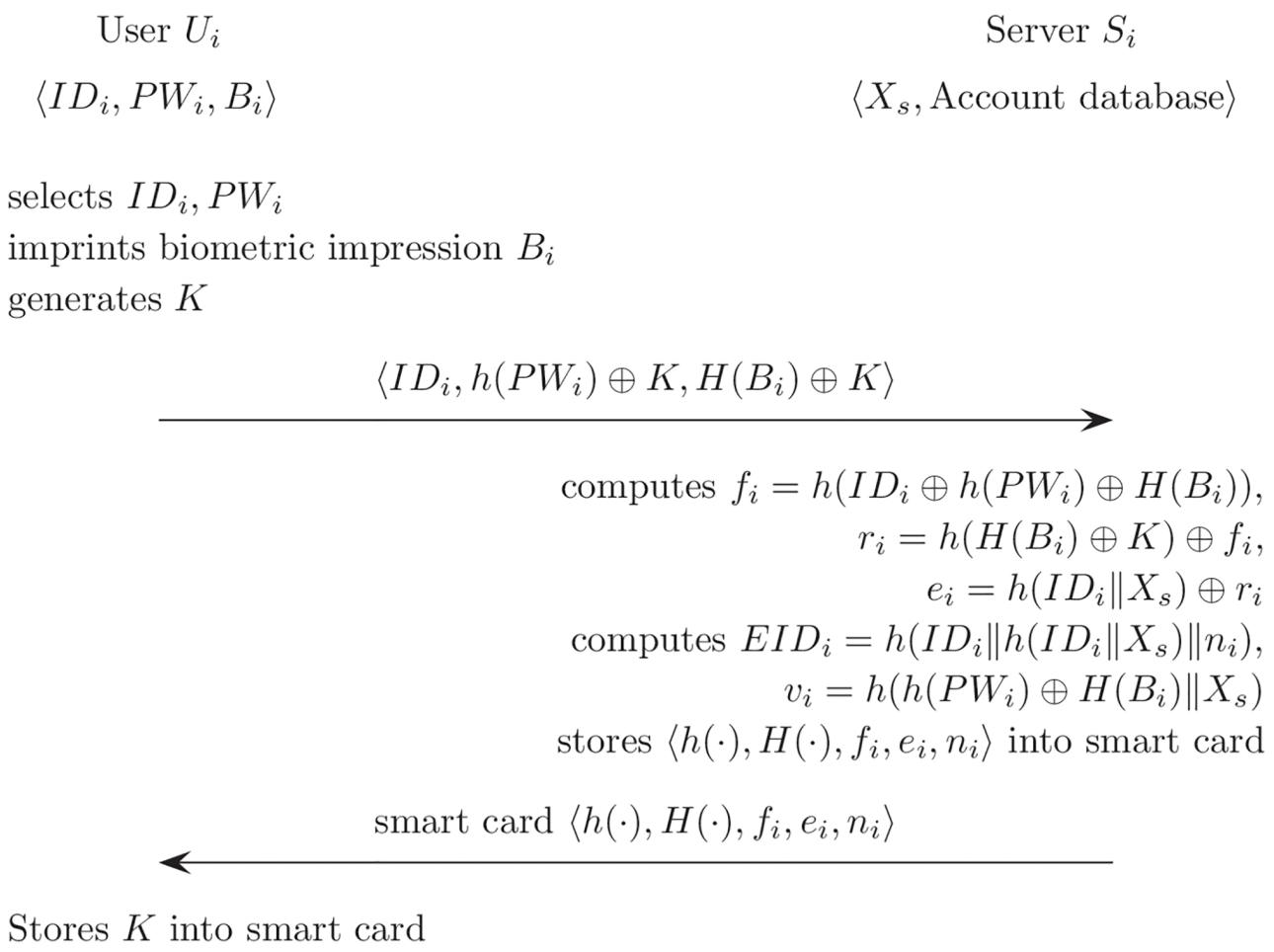

Fig 11. Registration phase for the proposed scheme.

https://doi.org/10.1371/journal.pone.0176250.g011

(R4) S_i computes EID_i and v_i as below, then S_i stores EID_i, ID_i, n_i, v_i for ID_i as an entry in a database.

$$EID_i = h(ID_i \parallel h(ID_i \parallel X_s) \parallel n_i),$$
$$v_i = h(h(PW_i) \oplus H(B_i) \parallel X_s).$$

(R5) S_i sends a smart card to U_i. The smart card contains $\langle h(\cdot), H(\cdot), f_i, e_i, n_i \rangle$ through a secure channel. Then U_i stores K in the smart card.

Password change phase

For the proposed scheme, the password change phase is executed when U_i loses the smart card or wants to update the password. In order to change the password, U_i sends both the old password PW_i and new password PW_{inew}. Fig 12 describes the password change phase for the proposed scheme.

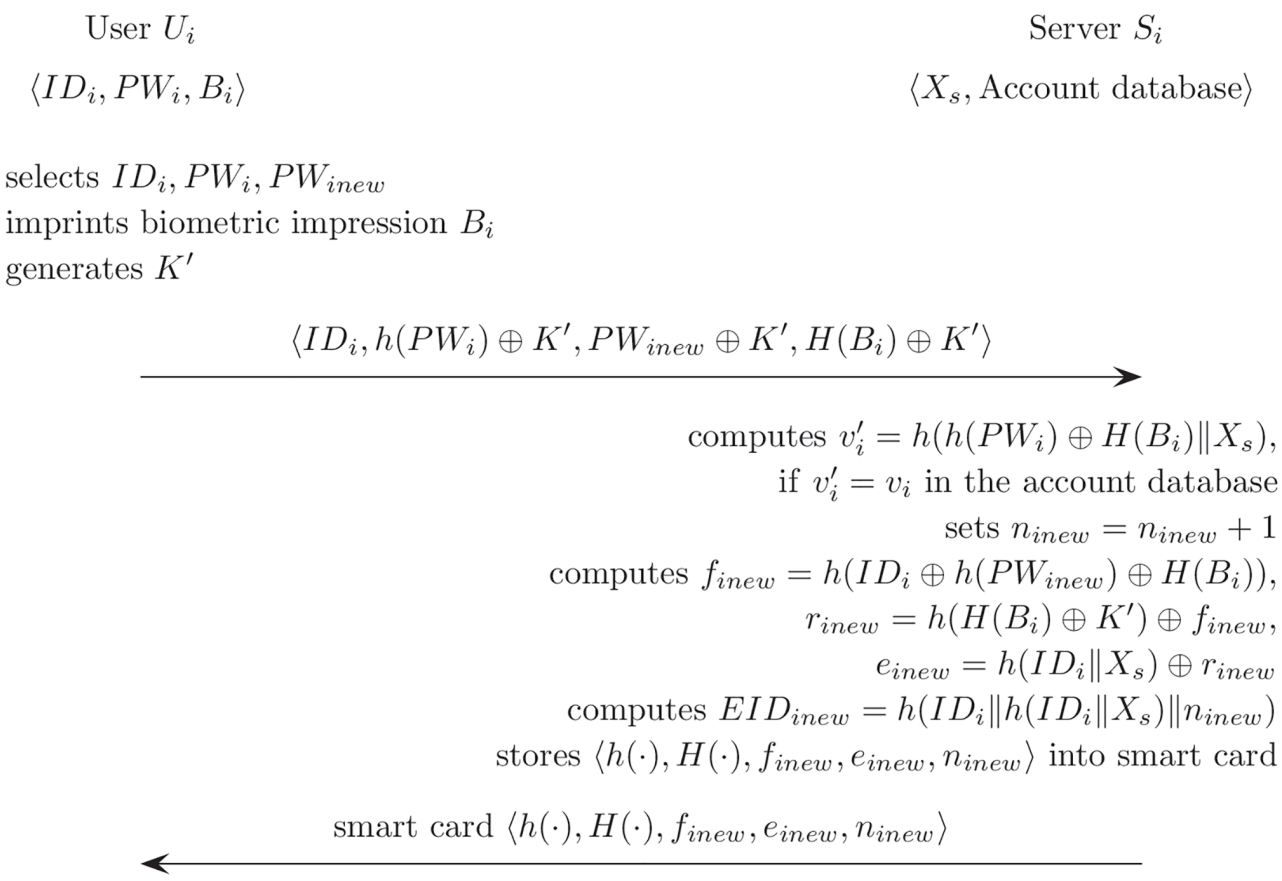

Fig 12. Password change phase for the proposed scheme.

https://doi.org/10.1371/journal.pone.0176250.g012

(RR1) U_i selects and inputs ID_i, PW_i, and PW_{inew}. U_i imprints its own biometric impression B_i and generates a new random value K'. Then, U_i submits $\langle ID_i, h(PW_i) \oplus K', h(PW_{inew}) \oplus K', H(B_i) \oplus K' \rangle$ to S_i through a secure channel.

(RR2) After S_i receives these, S_i checks the database for the ID, and acquires the user's data including EID_i, ID_i, n_i, and v_i. Then, S_i computes $v_i' = h(h(PW_i) \oplus H(B_i) \| X_s)$ and compares v_i' with v_i in the database.

(RR3) S_i sets $n_{inew} = n_i + 1$. Then, S_i carries out the computations as follows:

$$\begin{aligned} f_{inew} &= h(ID_i \oplus h(PW_{inew}) \oplus H(B_i)), \\ r_{inew} &= h(H(B_i) \oplus K') \oplus f_{inew}, \\ e_{inew} &= h(ID_i \| X_s) \oplus r_{inew}. \end{aligned}$$

(RR4) S_i computes $EID_{inew} = h(ID_i \| h(ID_i \| X_s) \| n_{inew})$, then S_i stores EID_{inew}, ID_i, n_{inew} for ID_i to the entry of database.

(RR5) S_i sends a new smart card to U_i that contains $\langle h(\cdot), H(\cdot), f_{inew}, e_{inew}, n_{inew}\rangle$ by using a secure channel. Then U_i stores a new K' in the smart card.

Login and authentication phase

Fig 13 describes the login and authentication phase for the proposed scheme. U_i executes the following steps when U_i wants to authenticate a remote S_i. In this phase, the smart card checks the legitimacy of the user using ID_i, PW_i and B_i.

(L1) U_i inputs the ID_i and PW_i; U_i imprints B_i using a biological feature extraction device; computes $h(PW_i)$ using the general hash function and $H(B_i)$ using the bio-hash function. Then, the smart card computes f_i, and is verified as follows,

$$f_i = h(ID_i \oplus h(PW_i) \oplus H(B_i)).$$

(L2) If they are the same, U_i generates the current timestamp T_i and a random number R_c. Then, U_i computes r_i, M_1, M_2, M_3, EID_i using the user's input values and the smart card storing values as follows;

$$\begin{aligned} r_i &= h(H(B_i) \oplus K) \oplus f_i, \\ M_1 &= e_i \oplus r_i, \\ M_2 &= M_1 \oplus R_c, \\ M_3 &= h(M_1 \parallel R_c \parallel T_1), \\ EID_i &= h(ID_i \parallel h(ID_i \parallel X_s) \parallel n_i). \end{aligned}$$

(L3) U_i sends the login request message $\langle EID_i, M_2, M_3, T_1\rangle$ to S_i.

The server S_i executes the authentication phase when the message is received.

(A1) S_i checks that the EID_i satisfies the original format.

(A2) If the ID_i is valid when compared with the user's entry in the database in S_i, S_i computes M_4 and M_5, and then verifies M_3 as follows,

$$\begin{aligned} M_4 &= h(ID_i \parallel X_s), \\ M_5 &= M_2 \oplus M_4, \\ M_3 &= h(M_4 \parallel M_5 \parallel T_1). \end{aligned}$$

(A3) If M_3 is accurate, S_i generates the current timestamp T_2 and computes M_6 and M_7. Then, S_i sends the message $\langle EID_i, M_6, M_7, T_2\rangle$ to U_i.

$$\begin{aligned} M_6 &= M_4 \oplus R_s, \\ M_7 &= h(M_4 \parallel R_s \parallel T_2). \end{aligned}$$

(A4) U_i computes $M_8 = M_6 \oplus M_1$ and verifies whether $M_7 = h(M_1 \| M_8 \| T_2)$ or not. If they are equal, U_i generate a timestamp T_3 and computes M_9. Then U_i computes sk as follows.

$$\begin{aligned} M_9 &= h(M_1 \parallel R_c \parallel M_8 \parallel T_3), \\ sk &= h(M_1 \parallel R_c \parallel M_8 \parallel T_2 \parallel T_3). \end{aligned}$$

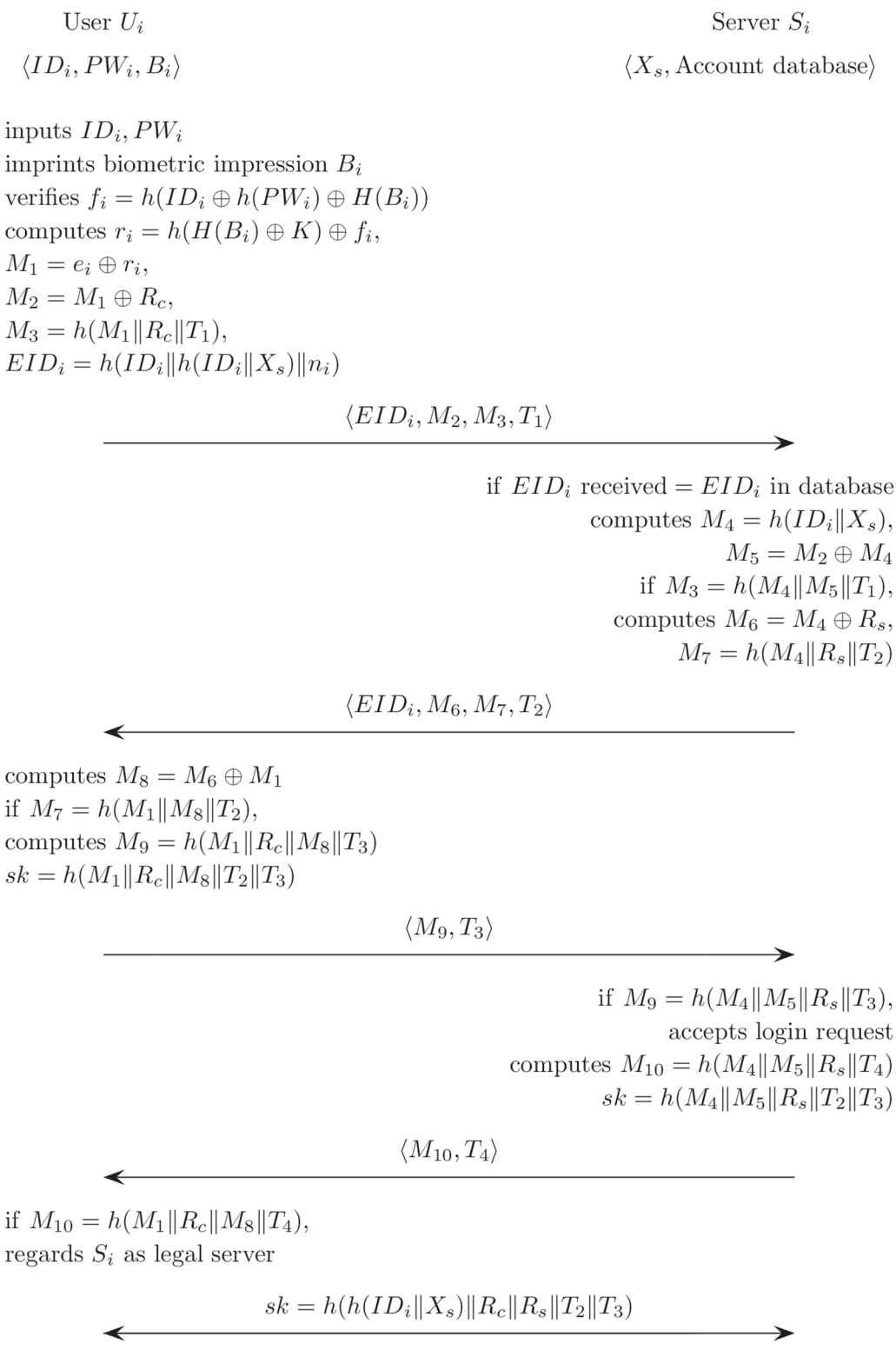

Fig 13. Login and authentication phase for the proposed scheme.

https://doi.org/10.1371/journal.pone.0176250.g013

(A5) U_i sends the message $\langle M_9, T_3 \rangle$ to S_i.

(A6) After receiving $\langle M_9 \rangle$, S_i verifies that M_9 is equal to $h(M_4||M_5||R_s||T_3)$ and then accepts the user's login request. S_i computes $M_{10} = h(M_4||M_5||R_s||T_4)$ and sk. Then, S_i sends $\langle M_{10}, T_4 \rangle$ to U_i.

$$sk = h(M_4 \parallel M_5 \parallel R_s \parallel T_2 \parallel T_3)$$

(A7) After receiving $\langle M_{10}, T_4 \rangle$, U_i verifies that M_{10} is equal to $h(M_1||R_c||M_8||T_4)$ and regards S_i as a legal server.

(A8) Therefore, U_i and S_i share the same session key after all phases have finished.

$$sk = h(h(ID_i \parallel X_s) \parallel R_c \parallel R_s \parallel T_2 \parallel T_3)$$

Analysis

Several analyses were carried out to confirm that the proposed scheme with a bio-hash function improves the security of the authentication process. Ding Wang *et al*. analyzed various smart-card-based password authentication methods and introduced a good solution using the principle of the security-usability trade-off to prevent off-line password attacks. Ding Wang *et al*. proposed that a fuzzy verifier can resolve the trade-off between the security requirement of resistance to smart card loss attack and the usability goal of a local password change [35–37].

In this paper, the proposed scheme uses a bio-hash function, which is similar to a fuzzy verifier to secure the system against various types of off-line guessing attacks. The proposed scheme is investigated by conducting a security analysis, a formal analysis, and an efficiency analysis. Then, the proposed scheme is compared to other authentication schemes, including Cao and Ge's scheme. We follow a security definition with strong secret values (B_i, x) with a high entropy that cannot be guessed in polynomial time and a secure one-way hash function $y = h(x)$. Given x to compute y is easy but y to compute x is much more difficult.

Security analysis

This section describes a security analysis to confirm the security of the proposed scheme.

1. [**Replay attack**] In the proposed scheme, even if an adversary intercepts the messages like $\langle EID_i, M_2, M_3, T_1 \rangle$ and $\langle M_9, T_3 \rangle$ over public communication and replays $\langle EID_i, M_2, M_3, T_1 \rangle$ to S_i, he cannot authenticate with S_i. First, it is hard for the adversary to respond within the allowable time for timestamp T_1, and even though the adversary passes the time limit, he cannot execute the appropriate response for $\langle EID_i, M_6, M_7, T_2 \rangle$. The adversary has only the previous $\langle M_9, T_3 \rangle$, which is not appropriate for the response because he cannot know the new R_c. Only a legal user can know the new R_c using $h(ID_i||X_s)$. Therefore, the adversary cannot succeed in the replay attack due to the timestamps and the lack of knowledge of $h(ID_i||X_s)$ [53].

2. [**Server masquerading attack**] If an adversary wants to masquerade as a legal server, he has to send the appropriate response to the user's request. When the user sends $\langle M_9, T_3 \rangle$ to the adversary, he has to compute the appropriate $\langle M_{10}, T_4 \rangle$ to look like a legal server. However, if the adversary wants to compute $\langle M_{10}, T_4 \rangle$ using M_9, T_3 and T_4, he has to know the R_c and

$h(ID||X_s)$. Only a legal server can compute $\langle M_{10}, T_4\rangle$ because the legal server stored X_s and R_c in the database and the adversary cannot know them. Therefore, the adversary cannot succeed in masquerading as a legal server.

3. [**Mutual authentication**] Mutual authentication means that a user and a server authenticate each other. In the proposed scheme, U_i and S_i authenticate each other by checking for a mutual random number, which is possible for a legal user and server because only they know $h(ID_i||X_s)$. Specifically, S_i authenticates U_i according to the $\langle M_9, T_3\rangle$ that is received because only a legal U_i can compute M_9 using S_i's M_6. U_i authenticates S_i by $\langle M_{10}, T_4\rangle$, and only the server can compute M_{10} from $\langle M_9, T_3\rangle$ because only the legal server can know the user's random number R_c using $h(ID_i||X_s)$, $R_c = M_2 \oplus h(ID_i||X_s)$ [54].

4. [**Biometric recognition error**] The proposed scheme uses a bio-hash function to prevent a biometric recognition error. Cao and Ge's scheme uses a general hash function to verify the user's biometrics, so a biometric recognition error happens as a result of the general hash function's behavior. However, the proposed scheme uses a bio-hash function for the user's biometric information because the bio-hash function provides consistent output for the same biometric information, even when a user's biometrics are input a little differently.

5. [**Slow wrong password detection**] Unlike Cao and Ge's scheme, the proposed scheme can check the user's password during the login phase. Therefore, it is possible to verify whether or not the user has input an accurate password. In the proposed scheme, when a user wants to login and authenticate on a server, he inputs his own ID_i, PW_i, and B_i. Using these, the smart card computes $f_i = h(ID_i \oplus h(PW_i) \oplus H(B_i))$ and computes it with f_i, which is stored in a smart card. If the user inputs the wrong password, the computed f_i and stored f_i will be different, so the user can immediately know whether he needs to input the correct password again.

6. [**Off-line password attack**] An adversary can extract all information stored in the user's smart card by using a side-channel attack, such as by physically monitoring the power consumption. However, in the proposed scheme, the user's password is always used with the user's ID_i and the biometrics information $H(B_i)$ like $f_i = h(ID_i \oplus h(PW_i) \oplus H(B_i))$. The user's ID_i is protected by $EID_i = h(ID_i||h(ID||X_s)||n_i)$. Moreover, B_i has a high entropy, so the adversary cannot carry out the computation. Therefore, even if the adversary extracts f_i using a side channel attack, he cannot compute the user's password because he cannot know both ID_i and $H(B_i)$.

7. [**User impersonation attack**] To successfully carry out a user impersonation attack, an adversary needs to know the user's $h(ID_i||X_i)$. In order to compute $h(ID_i||X_i)$, the adversary must know r_i using f_i and e_i; $f_i = h(ID_i \oplus h(PW_i) \oplus H(B_i))$, $e_i = h(ID_i||X_s) \oplus r_i$.

$$r_i = h(H(B_i) \oplus K) \oplus f_i.$$

However, r_i is protected by $h(H(B_i) \oplus K)$, and the adversary cannot know $H(B_i)$. Therefore the proposed scheme prevents a user impersonation attack.

8. [**ID guessing attack**] Unlike for $EID_i = h(ID_i||n_i)$ in Cao and Ge's scheme, the proposed scheme uses $EID_i = h(ID_i||h(ID_i||X_s)||n_i)$ to protect the user's ID_i. An adversary can extract n_i from the smart card and can obtain EID_i from public communications. However, if $h(ID_i||X_s)$ is not stored in a smart card and can only be easily computed by a legal U_i and S_i, then the adversary cannot compute $h(ID_i||X_s)$. Therefore, even if the adversary knows EID_i and n_i, he cannot compute ID_i from EID_i due to the ignorance of $h(ID_i||X_s)$.

9. [**Vulnerability to a DoS attack**] The proposed scheme checks the freshness of all messages using a timestamp T_1, T_2, T_3, T_4, so it is useless for an adversary to send the previous messages to the server. Moreover, U_i and S_i authenticate each other using the messages including current timestamps; $M_3 = h(M_1||R_c||T_1)$, $M_7 = h(M_4||R_s||T_2)$, $M_9 = h(M_1||R_c||M_8||T_3)$, $M_{10} = h(M_4||M_5||R_s||T_4)$. For example, S_i can check the freshness and legality of M_3 because M_3 and the timestamp T_1 do not match, even if the adversary sends the previous M_3 with the current timestamp. Therefore, the proposed scheme is more secure than Cao and Ge's authentication scheme against a DoS attack.

10. [**Lack of session key agreement**] Cao and Ge's authentication scheme does not provide a session key agreement, so it cannot establish secure communications with an encryption after all phases have finished. To resolve the problem of the lack of a session key, a session key agreement is provided during the login and authentication phase. In order to share the session key $sk = h(h(ID_i||X_s)||R_c||R_s|T_2||T_3)$. $h(ID_i||X_s)$, R_c and R_S are computed only by a legal U_i and S_i. T_2 and T_3 can be used to confirm the freshness of the session key, and the session key of the proposed scheme can be changed at every session to prevent various forms of attack [55].

Table 2 shows a comparison of the security analysis for various multi-factor authentication schemes, including our proposed scheme [14, 38, 39, 50, 56–58].

Formal analysis

BAN logic (Burrows-Abadi-Needham logic) was introduced by Burrows M, and it has consistently drawn attention due to the simplicity and straightforwardness of the analysis of authentication schemes, and in this section, we analyze the proposed scheme using BAN-logic with symbols P and Q representing principals and X and Y representing statements. The main notation of the logic is presented in BAN's paper and main inference rules. The analysis of an authentication scheme using the BAN-logic tool consists of four steps, and the formal analysis of the security of the proposed scheme is described as follows. The analysis shows that a session key can be generated correctly between the communicating parties during authentication. First, the notation of BAN logic being used in this scheme is introduced [59–62].

- $P|\equiv X$: The principal P believes statement X. This means that P believes that in the current run of the scheme, the statement X is true.
- $P \lhd X$: The principal P sees the statement X, which means that P has received a message containing X.

Table 2. Security analysis for various authentication schemes.

Attack resistance	[14]	[38]	[39]	[50]	[56]	[57]	[58]	Ours
Replay attack	O	O	O	O	O	O	O	O
Server masquerading attack	X	X	X	O	X	O	X	O
Mutual authentication	O	O	O	O	X	X	X	O
Biometric recognition error	X	X	X	X	X	X	X	O
Slow wrong password detection	X	X	X	X	O	O	O	O
Off-line password attack	X	O	O	X	X	X	X	O
User impersonation attack	X	O	O	X	X	X	X	O
ID guessing attack	X	X	X	X	X	X	X	O
Vulnerability to a DoS attack	X	X	X	X	X	O	X	O
Lack of session key agreement	X	X	X	X	O	O	O	O

https://doi.org/10.1371/journal.pone.0176250.t002

- $P|\sim X$: The principal P once said the statement X, which means that $P|\equiv X$ when P sent it.
- $P \Rightarrow X$: The principal P has jurisdiction over statement X. This means that P has complete control on the formula X.
- $\sharp(X)$: The formula X is fresh. This means that formula X has not been used before.
- $P \mid\equiv Q \overset{K}{\longleftrightarrow} P$: P believes that the principal P and Q communicate with each other using K.
- $P \overset{K}{\longleftrightarrow} X$: K is shared secret information between P and Q. The secret key K is known only to P and Q, and K is a secret between both parties.
- $\{X\}_K$: The formula X is encrypted using the secret key K.
- $\langle X\rangle_K$: The formula X is combined including the secret key K.
- $(X)_K$: The formula X is hashed including the secret key K.
- sk: The session key used in the current session.

To describe the logical postulates of BAN logic, we present the following rules:

1. Message-meaning rule: $\frac{P|\equiv P\overset{K}{\leftrightarrow}Q, P\triangleleft(X)_K}{P|\equiv Q|\sim X}$: if the principal P believes he/she shares the secret key K with Q, P sees the statement X hashed to include the K. Then P believes that Q once said X.
2. Nonce-verification rule: $\frac{P|\equiv \#(X), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$: if principal P believes that X is fresh and P believes Q once said X, then P believes that Q believes X.
3. The belief rule: $\frac{P|\equiv X, P|\equiv Y}{P|\equiv (X,Y)}$: if principal P believes both X and Y, then P believes (X, Y).
4. Freshness-conjuncatenation rule: $\frac{P|\equiv \#(X)}{P|\equiv \#(X,Y)}$: if principal P believes X is fresh, then P believes (X, Y) is fresh.
5. Jurisdiction rule: $\frac{P|\equiv Q|\Rightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$: if principal P believes that Q has jurisdiction over X and P believes that Q believes X, then P believes X.

According to the analytic procedures of BAN logic and using previously described logical postulates, the proposed scheme needs to satisfy the following goals:

- Goal 1: $S \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$.
- Goal 2: $U \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$.
- Goal 3: $S \mid\equiv U \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$.
- Goal 4: $U \mid\equiv S \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$.

The generic type of proposed scheme is as follows:

- Message 1.
  $U \rightarrow S$: $h(ID_i||h(ID_i||X_s)||n_i)$, $h(ID_i||X_s) \oplus R_c$, $h(h(ID_i||X_s)||R_c||T_1)$, T_1
- Message 2.
  $S \rightarrow U$: $h(ID_i||h(ID_i||X_s)||n_i)$, $h(ID_i||X_s) \oplus R_s$, $h(h(ID_i||X_s)||R_s||T_2)$, T_2
- Message 3.
  $U \rightarrow S$: $h(h(ID_i||X_s)||R_c||R_s||T_3)$, T_3

- Message 4.
  $S \rightarrow U$: $h(h(ID_i||X_s)||R_c||R_s||T_4)$, T_4

The idealized form of proposed scheme is as follows:

- Message 1. $U \rightarrow S$: $(ID_i, n_i)_{h(ID_i||X_s)}$, $\langle R_c \rangle_{h(ID_i||X_s)}$, $(R_c, T_1)_{h(ID_i||X_s)}$, T_1
- Message 2. $S \rightarrow U$: $(ID_i, n_i)_{h(ID_i||X_s)}$, $\langle R_s \rangle_{h(ID_i||X_s)}$, $(R_s, T_2)_{h(ID_i||X_s)}$, T_2
- Message 3. $U \rightarrow S$: $(R_c, R_s, T_3)_{h(ID_i||X_s)}$, T_3, $U \overset{sk}{\longleftrightarrow} S$
- Message 4. $S \rightarrow U$: $(R_c, R_s, T_4)_{h(ID_i||X_s)}$, T_4, $U \overset{sk}{\longleftrightarrow} S$

We make the following assumptions for the initial state of the protocol to analyze the proposed protocol:

- A1: $U \mid\equiv \sharp(T_1)$
- A2: $S \mid\equiv \sharp(T_2)$
- A3: $U \mid\equiv \sharp(T_3)$
- A4: $S \mid\equiv \sharp(T_4)$
- A5: $U \mid\equiv (U \overset{h(ID_i||X_s)}{\longleftrightarrow} S)$
- A6: $S \mid\equiv (U \overset{h(ID_i||X_s)}{\longleftrightarrow} S)$
- A7: $U \mid\equiv S \Rightarrow (U \overset{sk}{\longleftrightarrow} S)$
- A8: $S \mid\equiv U \Rightarrow (U \overset{sk}{\longleftrightarrow} S)$

The idealized form of the proposed protocol based on BAN logic rules and assumptions is analyzed. The main proofs are described as follows.

According to Message 3, we could obtain:

- S1: $S \lhd \{(R_c, R_s, T_3)_{h(ID_i||X_s)}, T_3, U \overset{sk}{\longleftrightarrow} S\}$
  According to the assumption A6 and the message meaning rule, we obtain:
- S2: $S \mid\equiv U \mid\sim \{(R_c, R_s, T_3)_{h(ID_i||X_s)}, T_3, U \overset{sk}{\longleftrightarrow} S\}$
  According to the assumption A3 and the freshness conjuncatenation rule, we can obtain:
- S3: $S \mid\equiv \sharp\{(R_c, R_s, T_3)_{h(ID_i||X_s)}, T_3, U \overset{sk}{\longleftrightarrow} S\}$
  According to the assumption S2, S3 and the nonce verification rule, we obtain:
- S4: $S \mid\equiv U \mid\equiv \{(R_c, R_s, T_3)_{h(ID_i||X_s)}, T_3, U \overset{sk}{\longleftrightarrow} S\}$
  According to S4, we apply the belief rule, we obtain:
- S5: $S \mid\equiv U \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$, We satisfy (**Goal 3**. $S \mid\equiv U \mid\equiv (U \overset{sk}{\longleftrightarrow} S$)
  According to the assumption A8, S5 and the jurisdiction rule, we can obtain the conclusion as follows:
- S6: $S \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$, We satisfy (**Goal 1**. $S \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$)

According to the message 4, we obtain:

Table 3. Computational costs.

Phases	[14]	[38]	[39]	[50]	[56]	[57]	[58]	Ours
Registration phase	3 T_h	3 T_h	3 T_h	7 T_h	5 T_h	7 T_h	4 T_h	7 T_h
Login phase	2 T_h	3 T_h	2 T_h	4 T_h	11 T_h	4 T_h	4 T_h	4 T_h
Authentication phase	5 T_h	6 T_h	8 T_h	7 T_h	4 T_h	11 T_h	13 T_h	9 T_h

https://doi.org/10.1371/journal.pone.0176250.t003

Table 4. Efficiency simulation.

Authentication scheme	[14]	[38]	[39]	[50]	[56]	[57]	[58]	Ours
Execution time (millisecond)	2.0	2.4	2.6	3.6	4.0	4.4	4.2	4.0

https://doi.org/10.1371/journal.pone.0176250.t004

- S7: $U \lhd \{(R_c, R_s, T_4)_{h(ID_i \| X_s)}, T_4, U \overset{sk}{\longleftrightarrow} S\}$
  According to the assumption A5 and the message meaning rule, we obtain:
- S8: $U \mid\equiv S \mid\sim \{(R_c, R_s, T_4)_{h(ID_i \| X_s)}, T_4, U \overset{sk}{\longleftrightarrow} S\}$
  According to the assumption A4 and the freshness conjuncatenation rule, we obtain:
- S9: $U \mid\equiv \sharp\{(R_c, R_s, T_4)_{h(ID_i \| X_s)}, T_4, U \overset{sk}{\longleftrightarrow} S\}$
  According to assumption S8, S9 and the nonce verification rule, we obtain:
- S10: $U \mid\equiv S \mid\equiv \{(R_c, R_s, T_4)_{h(ID_i \| X_s)}, T_4, U \overset{sk}{\longleftrightarrow} S\}$
  According to S10, we apply the belief rule, we obtain:
- S11: $U \mid\equiv S \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$, We satisfy (**Goal 4**. $U \mid\equiv S \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$)
  According to the assumption A7, S11 and the jurisdiction rule, we can obtain the conclusion as follows:
- S12: $U \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$, We satisfy (**Goal 2**. $U \mid\equiv (U \overset{sk}{\longleftrightarrow} S)$)

Efficiency analysis

The computational costs of the modified scheme and others are calculated in Table 3. T_h stands for the computation time of the hash function while the computation time for the exclusive OR operation T_{XOR} does not appear in the table because it can be ignored when compared to T_h.

According to the results obtained in [63], T_h needs a time of about 0.20 ms ($T_h \approx 0.20$ ms) on a system using 3.0 GB RAM with a Pentium IV 3.2 GHz processor. Table 4 shows the efficiency for various authentication scheme obtained through a simulation.

As shown in Tables 3 and 4, the modified scheme requires a slightly higher computational cost than the others, but mainly in the registration phase [38–40, 50]. However, the modified scheme can provide all security properties shown in Table 2.

Conclusions

This paper discusses possible attacks for Cao and Ge's authentication scheme, and a modified scheme is proposed to improve security and protect against various attacks. A security analysis and efficiency analysis are carried out to compare the results of the modified scheme to those of other schemes. In addition, the modified scheme is verified by conducting a formal security analysis using BAN-logic. The results indicate that the modified scheme has a slightly higher

computational cost but that it is more secure than some of the other related schemes. The proposed scheme uses a bio-hash function for multi-factor biometric authentication to improve security. We also intend to conduct further studies on verification techniques, such as a fuzzy verifier and bio-hash function, to resolve the security-usability trade-off.

Acknowledgments

All authors, especially the corresponding author Dongho Won, would like to thank the anonymous reviewers for their time and invaluable comments and suggestions on this paper. This work was supported by Institute for Information and 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).

Author Contributions

Conceptualization: YC YL JM DW.

Data curation: YC YL.

Formal analysis: YC JM DW.

Funding acquisition: YC DW.

Investigation: YC YL DW.

Methodology: YC YL DW.

Project administration: YC YL DW.

Resources: YC YL DW.

Software: YC JM.

Supervision: YC JM.

Validation: YC JM.

Visualization: YC JM.

Writing – original draft: YC YL.

Writing – review & editing: YC JM DW.

References

1. Choi Y, Nam J, Lee D, Kim J, Jung J, and Won D. Security Enhanced Anonymous Multiserver Authenticated Key Agreement Scheme Using Smart Cards and Biometrics. The Scientific World Journal. 2014. https://doi.org/10.1155/2014/281305
2. Huang H, and Cao Z. IDOAKE: strongly secure ID-based one-pass authenticated key exchange protocol. Security and Communication Networks. 2013: 1153–1161.
3. Rivest RL, Shamir A and Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM. 1978: 120–126. https://doi.org/10.1145/359340.359342
4. Lamport L. Password authentication with insecure communication. Communications of the ACM. 1981; 24(11); 770–772. https://doi.org/10.1145/358790.358797
5. ElGamal T. (1985, January). A public key cryptosystem and a signature scheme based on discrete logarithms. In Advances in cryptology. Springer Berlin Heidelberg. 10–18.

6. Choi Y, Lee D, Kim J, Jung J, Nam J, and Won D. Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography. Sensors. 2014; 14(6): 10081–10106. https://doi.org/10.3390/s140610081 PMID: 24919012
7. Koblitz N. Elliptic curve cryptosystems. Mathematics of computation. 1987; 48(177): 203–209. https://doi.org/10.1090/S0025-5718-1987-0866109-5
8. Debiao H, Jianhua C, and Jin H. An ID-based client authentication with key agreement protocol for mobile client-server environment on ECC with provable security. Information Fusion. 2012; 13(3): 223–230. https://doi.org/10.1016/j.inffus.2011.01.001
9. Chaum D, Rivest RL, and Sherman AT. Advances in cryptology. In Proceedings of CRYPTO. 1983; 82: 279–303.
10. Shieh WG, and Wang JM. Efficient remote mutual authentication and key agreement. computers and security. 2006; 25(1): 72–77. https://doi.org/10.1016/j.cose.2005.09.008
11. Lin CH, and Lai YY. A flexible biometrics remote user authentication scheme. Computer Standards and Interfaces. 2004; 27(1): 19–23. https://doi.org/10.1016/j.csi.2004.03.003
12. Hwang MS, Lee CC, and Tang YL. A simple remote user authentication scheme. Mathematical and Computer Modelling. 2002; 36(1): 103–107. https://doi.org/10.1016/S0895-7177(02)00106-1
13. Das ML, Saxena A, and Gulati VP. A dynamic ID-based remote user authentication scheme. Consumer Electronics. IEEE Transactions on. 2004; 50(2): 629–631. https://doi.org/10.1109/TCE.2004.1309441
14. Hwang MS, and Li LH. A new remote user authentication scheme using smart cards. IEEE Transactions on Consumer Electronics. 2000; 46(1): 28–30. https://doi.org/10.1109/30.826377
15. Yoon EJ, and Yoo KY. Robust id-based remote mutual authentication with key agreement scheme for mobile devices on ecc. In Computational Science and Engineering. CSE'09. International Conference on. 2009;2: 633–640.
16. Yang JH, and Chang CC. An ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem. Computers and security. 2009; 28(3): 138–143. https://doi.org/10.1016/j.cose.2008.11.008
17. Islam SH, and Biswas GP. Comments on ID-based client authentication with key agreement protocol on ECC for mobile client-server environment. In Advances in Computing and Communications. Springer Berlin Heidelberg. 2011; 628–635.
18. Zhang F, and Kim K. Efficient ID-based blind signature and proxy signature from bilinear pairings. In Information Security and Privacy. Springer Berlin Heidelberg. 2003; 312–323.
19. Shim K. Efficient ID-based authenticated key agreement protocol based on Weil pairing. Electronics Letters. 2003; 39(8): 653–654. https://doi.org/10.1049/el:20030448
20. Paterson KG. ID-based signatures from pairingson elliptic curves. Electronics Letters. 2002; 38(18): 1025–1026. https://doi.org/10.1049/el:20020682
21. Nandakumar K. Multibiometric systems: Fusion strategies and template security. ProQuest. 2008.
22. Wu M, Chen J, Zhu W, and Yuan Z. Security analysis and enhancements of a multi-factor biometric authentication scheme. International Journal of Electronic Security and Digital Forensics. 2016; 8(4): 352–365. https://doi.org/10.1504/IJESDF.2016.079447
23. Amin R, Islam SH, Biswas GP, Khan MK., and Li X. (2015). Cryptanalysis and enhancement of anonymity preserving remote user mutual authentication and session key agreement scheme for e-health care systems. Journal of medical systems, 39(11), 140. https://doi.org/10.1007/s10916-015-0318-z PMID: 26342492
24. Islam SK, Obaidat MS, and Amin R. An anonymous and provably secure authentication scheme for mobile user. International Journal of Communication Systems. 2016. https://doi.org/10.1002/dac.3126
25. Amin R, Kumar N, Biswas GP, Iqbal R, and Chang V. A light weight authentication protocol for IoT-enabled devices in distributed Cloud Computing environment. Future Generation Computer Systems. 2016. https://doi.org/10.1016/j.future.2016.12.028
26. Amin R, and Biswas GP. A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks. Ad Hoc Networks. 2016; 36: 58–80. https://doi.org/10.1016/j.adhoc.2015.05.020
27. Amin R, and Biswas GP. Design and analysis of bilinear pairing based mutual authentication and key agreement protocol usable in multi-server environment. Wireless Personal Communications. 2015; 84(1): 439–462. https://doi.org/10.1007/s11277-015-2616-7
28. Amin R, Islam SH, Biswas GP, Khan MK, Leng L, and Kumar N. Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks. Computer Networks. 2016; 101: 42–62. https://doi.org/10.1016/j.comnet.2016.01.006

29. Amin R, and Biswas GP. Cryptanalysis and design of a three-party authenticated key exchange protocol using smart card. Arabian Journal for Science and Engineering. 2015; 40(11): 3135–3149. https://doi.org/10.1007/s13369-015-1743-5

30. Amin R. Cryptanalysis and Efficient Dynamic ID Based Remote User Authentication Scheme in Multi-server Environment Using Smart Card. IJ Network Security. 2016; 18(1): 172–181.

31. Li X, Niu JW, Ma J, Wang WD, and Liu CL. Cryptanalysis and improvement of a biometrics-based remote user authentication scheme using smart cards. Journal of Network and Computer Applications, 2011; 34(1): 73–79. https://doi.org/10.1016/j.jnca.2010.09.003

32. Li X, Niu J, Khan MK, and Liao J. An enhanced smart card based remote user password authentication scheme. Journal of Network and Computer Applications. 2013; 36(5): 1365–1371. https://doi.org/10.1016/j.jnca.2013.02.034

33. Li X, Niu J, Wang Z, and Chen C. Applying biometrics to design three factor remote user authentication scheme with key agreement. Security and Communication Networks. 2014; 7(10): 1488–1497.

34. Li X, Niu J, Khan MK, Liao J, and Zhao X. Robust three factor remote user authentication scheme with key agreement for multimedia systems. Security and Communication Networks. 2014. https://doi.org/10.1002/sec.961

35. He D, and Wang D. Robust biometrics-based authentication scheme for multiserver environment. IEEE Systems Journal. 2015; 9(3): 816–823. https://doi.org/10.1109/JSYST.2014.2301517

36. Ma CG, Wang D, and Zhao SD. Security flaws in two improved remote user authentication schemes using smart cards. International Journal of Communication Systems. 2014; 27(10): 2215–2227. https://doi.org/10.1002/dac.2468

37. Wang D, Ma CG, and Wu P. Secure password-based remote user authentication scheme with non-tamper resistant smart cards. In IFIP Annual Conference on Data and Applications Security and Privacy. Springer Berlin Heidelberg. 2012; 114–121.

38. An Y. Security analysis and enhancements of an effective biometric-based remote user authentication scheme using smart cards. BioMed Research International. 2012.

39. Das AK. Analysis and improvement on an efficient biometric-based remote user authentication scheme using smart cards. Information Security. IET. 2011; 5(3): 145–151. https://doi.org/10.1049/iet-ifs.2010.0125

40. Li CT, and Hwang MS. An efficient biometrics-based remote user authentication scheme using smart cards. Journal of Network and computer applications, 2010; 33(1): 1–5. https://doi.org/10.1016/j.jnca.2009.08.001

41. Al-Assam H, and Jassim SA. Multi-factor challenge/response approach for remote biometric authentication. In SPIE Defense, Security, and Sensing. International Society for Optics and Photonics. 2011; 80630V–80630V.

42. Sarier ND. Improving the accuracy and storage cost in biometric remote authentication schemes. Journal of Network and Computer Applications. 2010; 33(3): 268–274. https://doi.org/10.1016/j.jnca.2009.12.017

43. Cox IJ, Miller ML, Bloom JA, and Honsinger C. Digital watermarking. San Francisco: Morgan Kaufmann. 2002;53.

44. Pointcheval D, and Zimmer S. Applied Cryptography and Network Security. Proceedings. Springe: Berlin. 2008;277–295.

45. Choi Y, Lee D, Kim J, Jung J, Nam J, and Won D. Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography. Sensors. 2014; 14(6): 10081–10106. https://doi.org/10.3390/s140610081 PMID: 24919012

46. KAMAL K, GHANY A, MONEIM MA, GHALI NI, HASSANIEN AE, and HEFNY HA. A Symmetric Bio-Hash Function Based On Fingerprint Minutiae And Principal Curves Approach. 2011.

47. Teoh AB, Goh A, and Ngo DC. Random multispace quantization as an analytic mechanism for biohashing of biometric and random identity inputs. Pattern Analysis and Machine Intelligence. IEEE Transactions on. 2006; 28(12): 1892–1901. https://doi.org/10.1109/TPAMI.2006.250

48. Kim J, Lee D, Jeon W, Lee Y, and Won D. Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks. Sensors. 2014; 14(4): 6443–6462. https://doi.org/10.3390/s140406443 PMID: 24721764

49. Nam J, Choo KKR, Kim M, Paik J, and Won D. Dictionary attacks against password-based authenticated three-party key exchange protocols. KSII Transactions on Internet and Information Systems (TIIS). 2013; 7(12): 3244–3260.

50. Cao L, and Ge W. Analysis and improvement of a multi-factor biometric authentication scheme. Security and Communication Networks. 2015; 8(4): 617–625. https://doi.org/10.1002/sec.1010

51. Choi Y, Nam J, Lee Y, Jung S, and Won D. Cryptanalysis of Advanced Biometric-Based User Authentication Scheme for Wireless Sensor Networks. In Computer Science and its Applications. Springer Berlin Heidelberg. 2015;1367–1375.
52. Nam J, Choo KKR, Kim M, Paik J, and Won D. An Offline Dictionary Attack against Abdalla and Pointcheval's Key Exchange in the Password-Only Three-Party Setting. IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences. 2015; 98(1): 424–427. https://doi.org/10.1587/transfun.E98.A.424
53. Syverson P. A taxonomy of replay attacks [cryptographic protocols]. In Computer Security Foundations Workshop VII. CSFW 7. Proceedings. IEEE. 1994; 187–191.
54. Otway D, and Rees O. Efficient and timely mutual authentication. ACM SIGOPS Operating Systems Review. 1987; 21(1): 8–10. https://doi.org/10.1145/24592.24594
55. Blake-Wilson S, Johnson D, and Menezes A. Key agreement protocols and their security analysis. Springer Berlin Heidelberg. 1997; 30–45.
56. Wen F, and Li X. An improved dynamic ID-based remote user authentication with key agreement scheme. Computers and Electrical Engineering. 2012; 38(2): 381–387. https://doi.org/10.1016/j.compeleceng.2011.11.010
57. Chou JS, Huang CH, Huang YS, and Chen Y. Efficient Two-Pass Anonymous Identity Authentication Using Smart Card. IACR Cryptology ePrint Archive. 2013;402.
58. Das AK, and Goswami A. A secure and efficient uniqueness-and-anonymity-preserving remote user authentication scheme for connected health care. Journal of medical systems. 2013; 37(3): 9948. https://doi.org/10.1007/s10916-013-9948-1 PMID: 23660745
59. Boyd C, and Mao W. On a limitation of BAN logic. In Advances in Cryptology-EUROCRYPT'93. Springer Berlin Heidelberg. 1994;240–247.
60. Bleeker A, and Meertens L. A semantics for BAN logic. In Proceedings of the DIMACS Workshop on Design and Formal Verification of Security Protocols. 1997.
61. Burrows M, Abadi M, and Needham RM. A logic of authentication. In Proceedings of the Royal Society of London A: Mathematical. Physical and Engineering Sciences. The Royal Society. 1989; 426: 233–271.
62. Abadi M, and Needham R. Prudent engineering practice for cryptographic protocols. IEEE transactions on Software Engineering. 1996; 1: 6–15. https://doi.org/10.1109/32.481513
63. Xue K, and Hong P. Security improvement on an anonymous key agreement protocol based on chaotic maps. Communications in Nonlinear Science and Numerical Simulation. 2012; 17(7): 2969–2977. https://doi.org/10.1016/j.cnsns.2011.11.025