國立勤益科技大學
資訊工程系碩士班

碩士論文

A secure low communication risk biometric-based remote user authentication scheme using smart cards

一個使用生物特徵智慧卡之安全低通訊風險遠端使用者認證系統

研 究 生：林廷諭
指導教授：林宗宏 博士

中華民國 一0二 年 五 月

一個使用生物特徵智慧卡之
安全低通訊風險遠端使用者認證系統

A secure low communication risk biometric-based remote user authentication scheme using smart cards

研 究 生：林 廷 諭　　Student:Ting-Yu Lin
指導教授：林 宗 宏 博士　　Advisor: Dr. Tsung-Hung Lin

國立勤益科技大學
資訊工程系碩士班
碩士論文

A Thesis
Submitted to
Department of Computer Science and Information Engineering
College of Electrical Engineering and Computer Science
National Chin-Yi University of Technology
in Partial Fulfillment of the Requirements
for the Degree of
Master of Science
in
Computer Science and Information Engineering

May 2013
Taiping, Taichung, Taiwan, Republic of China

中華民國 一0二 年 五 月

國立勤益科技大學

博碩士論文全文上網授權書

（提供授權人裝訂於紙本論文書名頁之次頁用）

本授權書所授權之論文為授權人在國立勤益科技大學
資訊工程系 不分 組 102 學年度第 二 學期取得碩士學位之論文。

論文題目：一個使用生物特徵智慧卡之 安全低通訊風險遠端使用者認證系統
指導教授：林宗宏

■ 同意

本人具有著作權之論文全文資料，非專屬、無償授予本人畢業學校圖書館，不限地域、時間與次數，以微縮、光碟或數位化等各種方式重製與利用，提供讀者基於著作權法合理使用範圍內之線上檢索、閱覽、下載及列印。

論文全文上載網路公開之範圍及時間：

校內區域網路	■ 立即公開
校外網際網路	■ 中華民國 104 年 5 月 22 日公開

授 權 人：林廷諭
簽　　名：林廷諭
中　華　民　國 107 年 5 月 22 日

國家圖書館
博碩士論文電子檔案上網授權書

本授權書所授權之論文為授權人在國立勤益科技大學資訊工程系 102 學年度第二學期取得碩士學位之論文。

論文題目：一個使用生物特徵智慧卡之 安全低通訊風險遠端使用者認證系統
指導教授：林宗宏

茲同意將授權人擁有著作權之上列論文全文(含摘要)，提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印，此項授權係非專屬、無償授權國家圖書館及本人畢業學校之圖書館，不限地域、時間與次數，以微縮、光碟或數位化方式將上列論文進行重製，並同意公開傳輸數位檔案。

☐ 立即開放

☐ 上列論文為授權人向經濟部智慧財產局申請專利之附件或相關文件之一（專利申請案號：　　），請於 2015 年 05 月 22 日 後再將上列論文公開或上載網路。

☑ 因上列論文尚未正式對外發表，請於 2015 年 05 月 22 日 後再將上列論文公開或上載網路。

☐ 其他

授權人：林廷諭

親筆簽名及蓋章：林廷諭　　民國 102 年 5 月 日

E-Mail：game951951@hotmail.com

國立勤益科技大學
研究所碩士班
論文口試委員會審定書

本校 資訊工程系 碩士班 林廷諭 君所提論文

一個使用生物特徵智慧卡之
安全低通訊風險遠端使用者認證系統
(A secure low communication risk biometric-based remote user authentication scheme using smart cards)

合於碩士資格水準，業經本委員會評審認可。

學位考試委員會召集人：謝建成 簽章

委員：蔡奎立 簽章

委員：吳嘉琦 簽章

委員：李宗福 簽章

指導教授：林永松 簽章

系（所）主管：張甄英

中華民國 一0二 年 五 月 九 日

一個使用生物特徵智慧卡之
安全低通訊風險遠端使用者認證系統

研 究 生：林 廷 諭　　指導教授：林 宗 宏 博士

國立勤益科技大學資訊工程系

摘要

越來越多的個人訊息透過雲端網路在客戶端以及伺服器端之間傳遞。 只靠唯一的密碼來保護，這個驗證的安全性是不足的，簡單的密碼容易被猜測及破解，而且並沒有身分確認的特性。前者有相關研究使用機密金鑰來使得密碼的複雜度提高，後者則是利用生物系統的辨識性來改善。生物系統是一種特徵辨識系統，可以建立或確認一個人的身分，而智慧卡擁有卡運算系統(COS)、計算能力以及記憶體，所以它非常適合儲存加密金鑰來驗證以及運算個人訊息。在本論文中，我們提出了一個智慧卡認證通訊協定。我們的方法主要利用一種模餘數運算的特性，來進行兩層的密碼認證，第一層可以過濾大部分的密碼輸入錯誤，第二層可以確認密

碼的正確性。利用此種雙重密碼認證，可以有效的抵抗字典攻擊，也可以驗證惡意的密碼攻擊，而不會損失任何的安全性。經過分析以及比較其它協議後，我們證明我們的方法更加的安全而且擁有更低的通訊攻擊風險。

關鍵字:Steganography, biometrics, cryptography, authentication, smart card.

Abstract

More and more personal information is delivered between client and server through cloud networks. The security of authentication is not enough if relying on only passwords. Simple passwords are easy to guess or crack, and it can't check the user's legitimacy. The related research uses secret keys and the biometric system to improve this scheme. Smart card possesses COS (Card Operating System), computing ability, and memory, so it was very suitable for storing encrypted key to verify personal information. In this thesis, we propose a smart card communication protocol for authentications. The authentication used the modular arithmetic operator to build multiple authentication. The first authentication can filter most error passwords and the second authentication can check the correctness of the password. This scheme mainly can effectively resist dictionary attacks and also verify malicious password attacks without loss any security. After the analytic comparison of this protocols with other protocols, the method is more security and prossesses low communication attack risk.

致謝

感謝我的指導教授 林宗宏老師，在這兩年期間指導我，在我迷惘的時候，引導我走去正確的方向，並在我遇到問題的時候，教導我如何去解決問題，也謝謝師母 郭宇婕的諄諄教誨，告訴我人與人之間如何相處，與長輩之間的應對，讓我成長。

這篇論文能夠完成，除了謝謝我的指導教授 林宗宏老師外，還要感謝 李添福教授，謝謝李教授在百忙之中回覆學生的信件，為學生解惑，以及指正，使得論文的內容更加完整。

也要謝謝 謝建成教授，在剛進入研究所時，為學生引薦給 林宗宏老師，讓學生可以向老師學習，以及感謝 吳憲珠教授以及 蔡垂雄教授，因為加入他們的研究團隊的討論，讓學生的思考以及想法可以快速的成長。

此外，我也要感謝研究室的所有成員，在這兩年間不間斷的幫助研究室運作，沒有怨言的一起讓研究室向前邁進。

最後要感謝的是我的家人，感謝您們的包容與關心，讓我可以在這個環境專注課業與研究，我將這份喜悅與您們分享!

廷諭 於民國一〇二年四月三十日星期二

Contents

List of Figures

List of Tables

Keywords: Steganography, biometrics, cryptography, authentication, smart card.

Chapter 1

Introduction

Usually, several password technologies are used to verify the remote users in the cloud network environments [2][8][15]. More and more personal information is delivered between client and server through cloud network. The security of authentication is not enough if relying on only passwords. The user's password itself does not contain any directly related user information [4][10]. Simple passwords are easily guessed or cracked by the malicious attackers, which are why several researchers have proposed security communication protocols to improve password authentication. The user passwords are not easy to memorize, and must be stored in a secure database. Smart card possesses COS (Card Operating System), computing ability, and memory, so it was very suitable for storing encrypted key to verify personal informations. Recently, several remote login systems using smart cards are adopted for ubiquitous personal authentications.

The card systems are friendly for authentications, but they are unable to strictly confirm whether the users are the owners or not. Biometric has been widely used in various fields. Biological features have undeniable characteristics, as they have been confirmed that each person has a unique biometric code. Above all, biometric applications are very suitable for remote password authentications. In addition, the biometric key has several advantages; it is extremely difficult to forge, distribute, copy, or share. Overall, biometric authentications

are safer than traditional password authentications in remote user authentications.

In this thesis, we propose a smart card communication protocol for authentication. The scheme mainly can effectively resist dictionary attacks and also verify malicious password attacks without loss any security. The organizations of this thesis are describing as follows. In chapter 2, we review the traditional methods of remote password authentications. The proposed schemes are introduced in chapter 3, and security analyses are given in chapter 4. In chapter 5, we compare this scheme with other traditional password authentications. Finally, we conclude this research in chapter 6.

Chapter 2

Backgrounds and related works

In 2002, Lee et al. [9] used biometric of fingerprint in the remote user authentication. Their method does not require any password table, uses smart card and fingerprint verification. Their scheme withstands the major malicious attacks containing message replaying attack except for masquerade attack [12] and impersonation attack [3]. In 2003, an introduction of biometric recognition technique is proposed by Jain et al. [14]. In their research, three amusing biometrics issues are proposed, those are "who she is" for individual's identity, "what she possesses" for ID card, and "what she remembers" for password [14]. And then, the security and privacy analysis of biometric recognition was designated by Prabhakar et al. [5] at same year. Lin and Lai (2004) [12] proposed a scheme that overcome the vulnerable of masquerade attack. However, their scheme is susceptible to the server spoofing attack [6]. Afterward, Maltoni et al. [13] implemented a handbook of fingerprint recognition on fingerprint security systems at 2009. The basic function of biometric system is the comparison between extracted biometrics and samples stored in the database. In succession, to use biometric to confirm individual's identity on the remote user authentication are adopted by several researchers [3][4][6][7][9][12].

In 2010, Li and Hwang (2010) [11] proposed an efficient biometric-based remote user authentication. Their scheme doesn't store any password table in the remote authentica-

tion server. The authentication server contains secret information X_s. X_s was protected by one-way hash function and random values between the communications of server and client. In their scheme, they need not store used used random value to resist parallel session attack. But Li-Hwang's scheme has some flaws. In login and authentication phases, they didn't verify the validity of user entered passwords. If the password is valid then this login will works correctly. If the password is a wrongful password, the smart card will sends the request to server. The server receives the wrong messages and produces wrong parameters. And then, the server interrupts the request. The user entered wrongful password will waste extra computing cost and communication cost. Similarly, in password change phase, if the user enters illegal password and causes the secret key decrypt incomplete, then the new password will encryption incorrectly.

After that, A.K. Das (2011)[1] improves Li-Hwang's scheme by adding client password authentication mechanism. In their login and authentication phases, they verify their password directly using smart card before communicating to the server. This measure can effective discovery the incorrect user passwords without any overhead communications. In the process of communication, passwords verified in advance reduces several extra costs and prevents secret messages were intercepted by malicious attackers. In their password change phase, most accurate passwords decrypt the most accurate secret keys and the secret key produces the new encrypted information in the smart card. If the secret key was decrypted incorrectly then the information updated incorrectly too. And then, the user cannot use the new password to login to the remote authentication system [1].

In addition, some methods used hash function to process biometric but it has some problems. There will be some difference of each input biometric. After the operate of hash

function, the difference of biometrics will be magnified. The most serious situation is that the user never passes the biometric verify.

We have to avoid above circumstances, so we propose a secure low communication risk biometric-based remote user authentication scheme using smart cards. The scheme mainly can effectively resist dictionary attacks, low communication risks, and also verify malicious password attacks without loss any security.

Chapter 3

The proposed scheme

In this chapter, we will introduce this protocols and give some notations in the table 3.1. The scheme contains four phases, preparation phase (register phase), login phase, authentication phase, and password changing phase, and describes as follows.

Table 3.1: Notations used in our scheme

Notation	Description
C_i	client user i
TRC	trusted registration center
AS	authentication server
PW_i	C_i's password
ID_i	C_i's identity number
B_i	C_i's biometric
$h(\cdot)$	a secure one-way hash function
X_s	secret information maintained by the server
R_{ci}	a random number generated by C_i
R_s	a random number generated by AS
mod	a modulo operator
X	a modulus number generated by TRC
$\|\|$	a message concatenation operator
$\oplus$	operation of XOR

3.1 Preparation phase

In order to implement the follow-up stages, the users have to register their identification numbers to the system to complete the preparations of preparation phase.

- Preparation phase

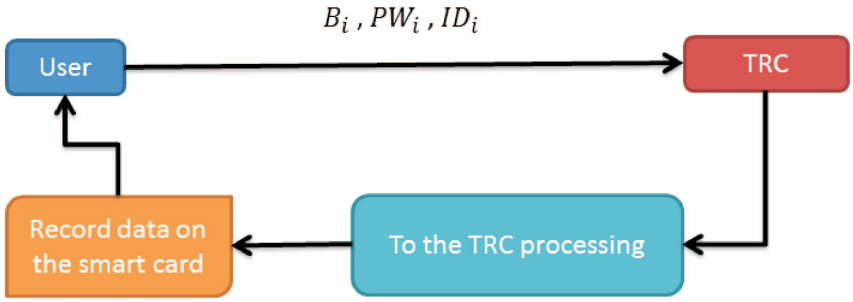

Figure 3.1: Preparation phase part 1

Step 1: First, the user C_i provides his/her identification number ID_i, his/her password PW_i, and inputs his/her biometric B_i on the specific device to the registration center TRC in person.

Step 2: Next, the registration center conducts the operations as following:

$$f_i = h\left(B_i\right)$$

$$r_i = h\left(PW_i\right) \oplus f_i$$

$$m_i = r_i \ mod \ X$$

$$e_i = h\left(ID_i \,||\, X_s\right) \oplus m_i$$

$$p_i = h\left(ID_i \,||\, X_s \,||\, PW_i\right)$$

X_s is secret information generated by the authentication server AS and X is a random number generated by the trusted registration center TRC, respectively. Here, we use modular arithmetic for encryption on r_i. We do not want to directly perform bare password authentications on the smart card. We use modular arithmetic to get the remainder, and then use the remainder to the implementation of the first authentication. To avoid the malicious attackers using dictionary attacks of hash function $h\left(PW_i\right)$, we apply the property of modular authentication to hide the bare user passwords. The property of modulation authentication can effectively filter the most incorrectly guessing password. It is very difficult to pick the exactly correct password in the same remainder from many inputs. If the attacker passed the first authentication by chance, we will prevent them from the following authentications.

Step 3: Finally, the registration center records $(ID_i,\ h\left(\cdot\right),\ B_i,\ e_i,\ m_i,\ p_i,\ X)$ on the user C_i's smart cart and sends it via a secure channel to the user C_i.

Difference with the previous method is the length of PW_i. Because the PW_i is not operated by hash function therefore the PW_i cannot compute with other message by XOR. We propose a solution is using binary of American Standard Code for Information Interchange(ASCII) to map the password and add zero in the lack of length. Thus PW_i can operate with other messages.

In this scheme, we store extra parameters $(B_i,\ m_i,\ p_i,\ X)$ into the smart card. m_i is a modular result that r_i mod X. m_i is only the first line of defense and is used for filter the most incorrectly guessing password. However, we do not use the r_i in this scheme and do not

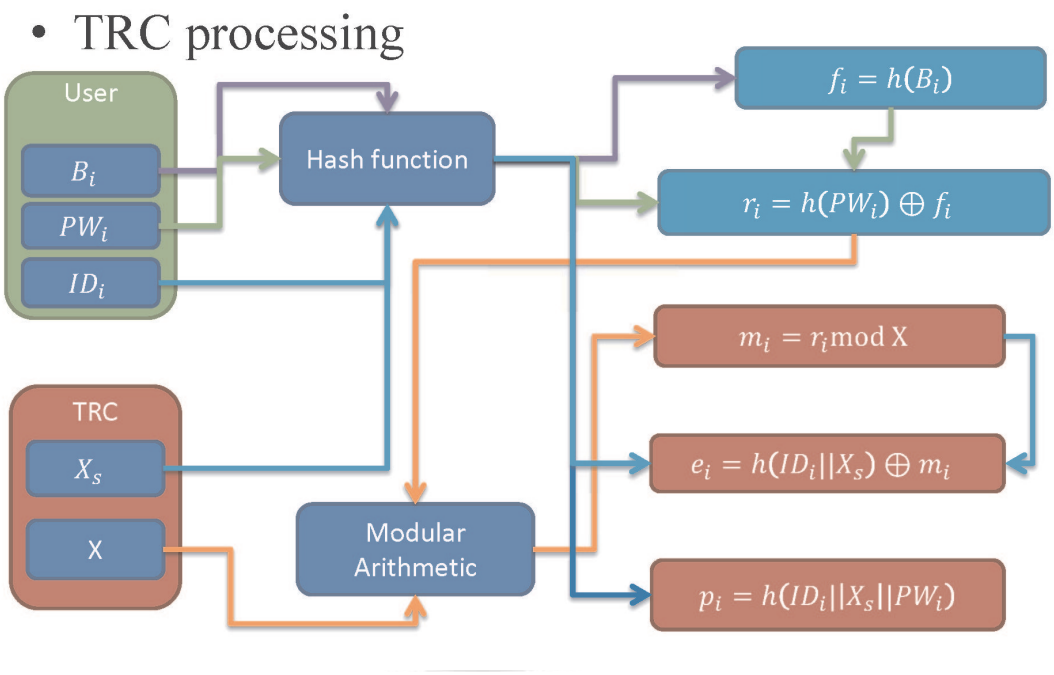

Figure 3.2: Preparation phase part 2

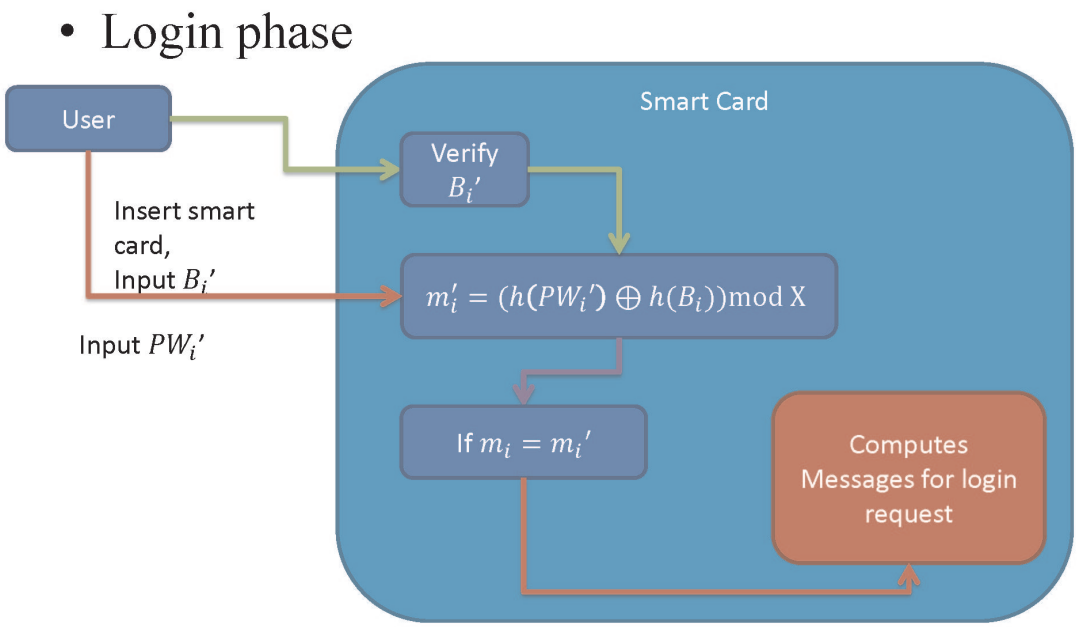

Figure 3.3: Login phase part 1

store it in the smart card because using p_i will achieve the same efficacy. In authentication phase we will describe how to use the p_i.

The biometric B_i do not need to operate by hash function in this scheme. Because the hash function provides the avalanche effect, even a slight change in an input value would cause a substantial change of the output value. As a result the legal user may never login the system access. So we propose an alternative in login phase.

3.2 Login phase

In login phase, if the user C_i wants to login to the remote server, the user C_i must performs the following steps.

- Messages computing

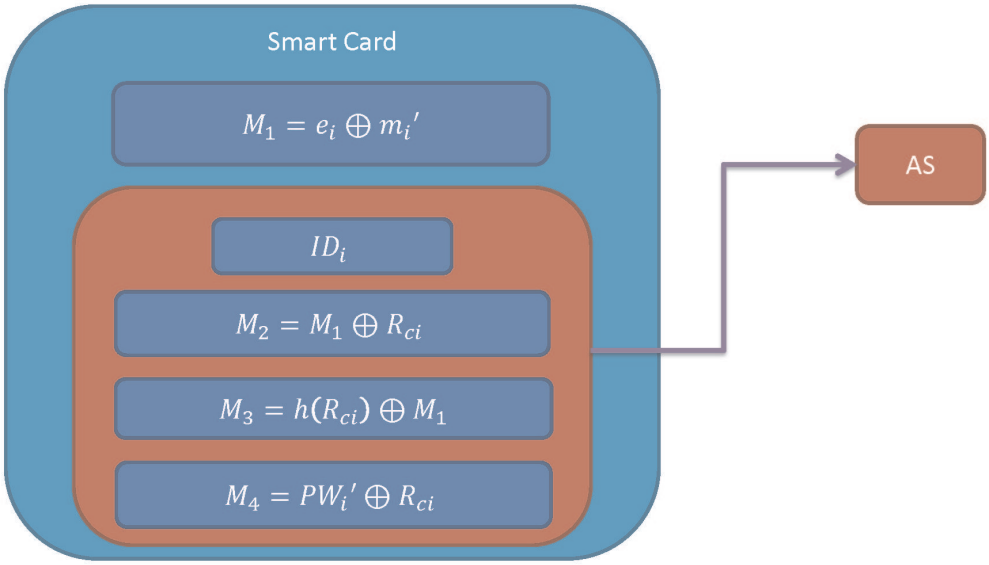

Figure 3.4: Login phase part 2

Step 1: First, the remote user C_i inserts his/her smart card into the card reader and inputs his/her biometric B_i' on the specific device. To confirm the identity of the client user, the smart card compares the biometric B_i with the biometric temple that stored in the smart card.

Step 2: Then, the smart card start to compare the similarity of B_i' with B_i. When the similarity is higher then the threshold, the biometric verification is passed. The threshold is set according to the actual using conditions. If the threshold is higher as well as security but the probability of the authentication succeeds is lower.

Step 3: If the biometric verification failed, then the login request will be interrupted because the user C_i is not the authorized owner. Otherwise, if the verification succeeded, the user C_i passes the biometric authentication and then input his/her password PW_i' to process the next authentication.

Step 4: The smart card computes

$$m_i' = (h(PW_i') \ \oplus \ h(B_i)) \ mod \ X.$$

If m_i' is equal to m_i then the user C_i passes the first round of password authentication. Passing this round of authentication does not mean that the password PW_i' is exactly correct. In the first round of authentication, the majority of incorrectly passwords will be filtered. The property of modulation authentication only allows the remainder and m_i are congruent modulo X. Otherwise, if m_i' is not equal to m_i then the smart card interrupts the login request and reduces the communication risks from malicious attackers.

Step 5: If $m_i' = m_i$ then the client computes the followings message:

$$M_1 = e_i \ \oplus \ m_i'$$

$$M_2 = M_1 \ \oplus \ R_{ci}$$

$$M_3 = h(R_{ci}) \ \oplus \ M_1$$

$$M_4 = PW_i' \ \oplus \ R_{ci}$$

In login phase, we need to mask the message (M_2, M_3, $M4$) using R_{ci}. After masking the messages, the attacker can't extract any information from M_2, M_3, and $M4$. The messages are protected by the one-way hash function and the user's random number R_{ci}. R_{ci} is generated randomly by user's the smart card. M_4 is used in authentication phase to

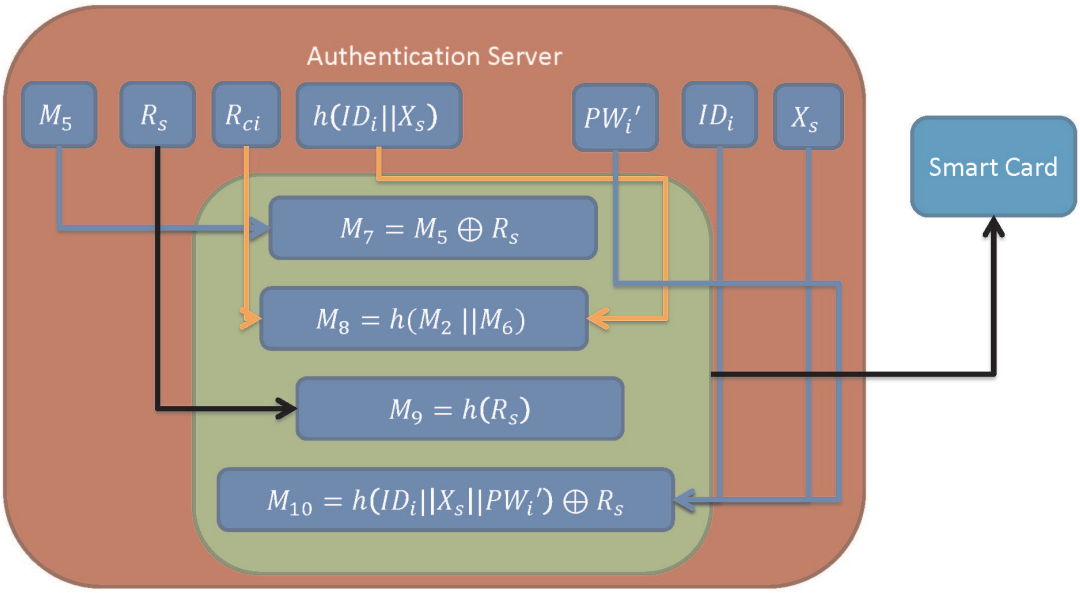

Figure 3.5: Authentication phase 1

verify the password PW_i'. We operate the extra message ei to avoid M_3 and M_4 are stolen to password guessing attacks.

Step 6: After the smart card computes (ID_i, M_2, M_3, M_4), and then the client C_i sends the message tuple (ID_i, M_2, M_3, M_4) to the server AS.

3.3 Authentication phase

In authentication phase, the user C_i and the server AS are going to conduct mutual authentication by random numbers R_{ci} and Rs.

After the server AS received the login request message, the server AS will go on the following steps to verify the remote user C_i:

Step 1: First of all, the server AS confirms the format of ID_i from the login request messages of client user C_i.

Step 2: If the format of ID_i is valid, the server AS derives the corresponding secret information X_s from the server's database and then AS computes the followings:

$$M_5 = h\left(ID_i \,||\, X_s\right)$$

X_s is secret information mapping the user's ID_i.

After the server produced the M_5, then the server keep computing the $M_6 = M_2 \oplus M_5$. If the user C_i entered the incorrect password, then modular number should be computed incorrectly. Thus

$$M_6 = h\left(ID_i \,||\, X_s\right) \oplus m_i \oplus m_i' \oplus R_{ci} \oplus h\left(ID_i \,||\, X_s\right).$$

When M_6 is evaluated, the server AS has to check the R_{ci} is not modified by any attack. So AS checks the value of hash function $h\left(M_6\right)$. If $h\left(M_6\right)$ is equal to $M_3 \oplus M_5$, it means the message is not modified by any attack. Then the server continues the login request. Otherwise, the key parameters may be modified, and then the server rejects the login request. It is impossible to let modified messages pass the authentications because they are protected by the one-way hash function. Next, the server computes the followings:

$$PWi' = M_4 \oplus M_6$$

$$M_7 = M_5 \oplus R_s$$

$$M_8 = h\left(M_2 \,||\, M_6\right) \left(= h\left(h(\, ID_i \,||\, X_s\,) \oplus R_{ci} \,||\, R_{ci}\right)\right)$$

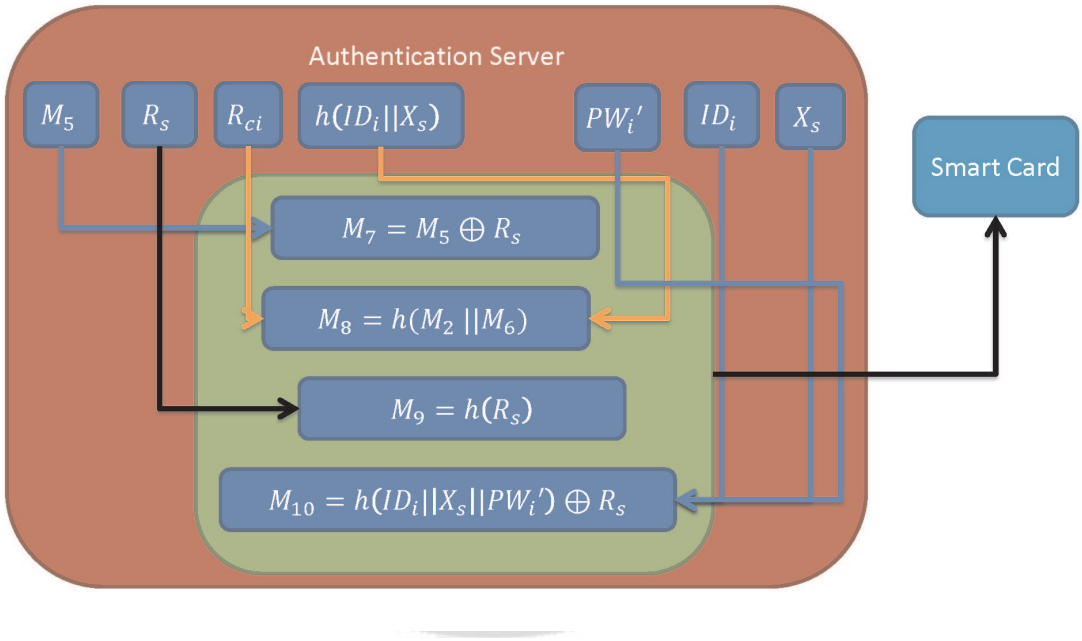

Figure 3.6: Authentication phase 2

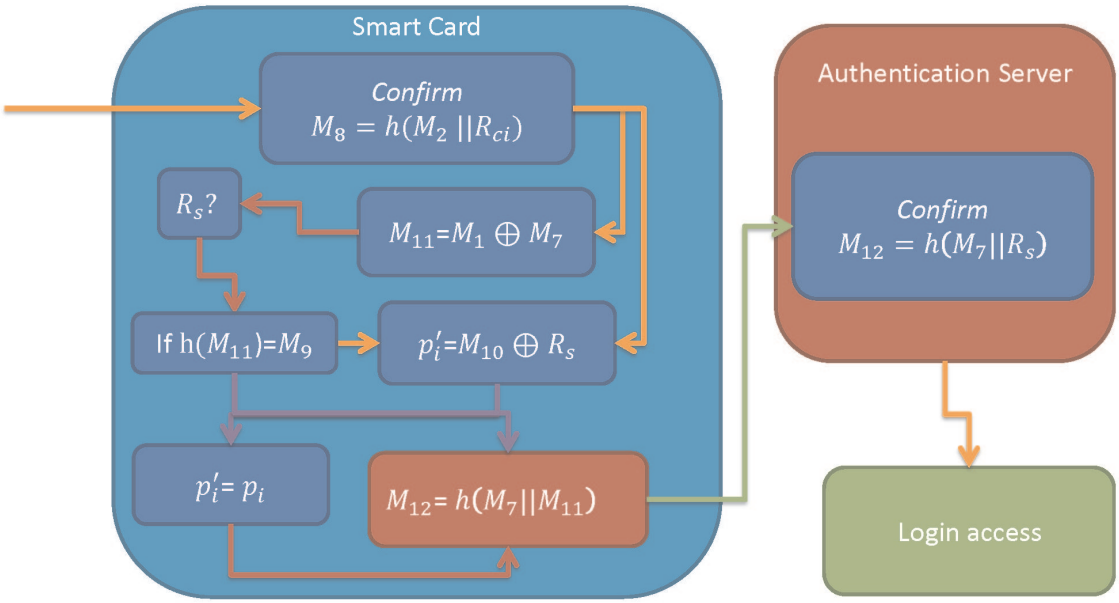

Figure 3.7: Authentication phase 3

$$M_9 = h\,(R_s)$$

$$M_{10} = h\,(ID_i \,||\, X_s \,||\, PW_i') \oplus R_s$$

M_{10} is used to pass the final authentication. The server AS uses the one-way hash function, the identify number ID_i of user, and the mapping secret information Xs to compute the message M_{10}. To prevent the attacker to steal the information $h\,(ID_i \,||\, X_s \,||\, PW_i')$ for final authentication, so we have to mask the information by R_s.

Step 3: The server AS delivers the message tuple (M_8, M_7, M_9, M_{10}) to the remote user C_i.

Step 4: When the remote user C_i receives the message tuple ($M_8,\ M_7,\ M_9,\ M_{10}$), the smart card firstly confirms $M_8 = ?h\,(M_2 \,||\, R_{ci})$. C_i needs to confirm the server AS is a counterfeit or not. If the remote server is a legal server, then the remote server should own the correct information to create M_5. Therefore, the parameter M_5 can decode the R_{ci} correctly. So we only to check the $M8$ is correct or not, then we can verify the server is legal or not. If the certification is failed, the smart card terminates the login request. Otherwise, the client believes the server is legal and goes on next step.

Step 5: After the server verification has already succeeded, the smart card computes the following two parameters for retrieving R_s:

$$M_{11} = M_1 \oplus M_7 = e_i \oplus m_i{}' \oplus h\,(ID_i \,||\, X_s) \oplus R_s = Rs$$

$$p_i{}' = M_{10} \oplus R_s = h\,(ID_i \,||\, X_s \,||\, PW_i{}')$$

Step 6: In the final authentication, smart card uses the p_i stored in the smart card and $p_i{}'$ received from M_{10} to check $p_i = p_i{}'$ or not. If p_i is equal to $p_i{}'$, it means the user C_i entered the valid password PW_i, and then the smart card computes $M_{12} = h\,(M_7 \,||\, M_{11})$ and sends M_{12} to the remote server AS.

If the final authentication failed, it means the user is still illegal, and the system may suffer dictionary attack. If the legal user entered the password incorrectly, the pass probability of the wrong password is very low. Because of anyone wants to lucky pass the first modular authentication is negligible. We assume using the wrong password to pass the first authentication is an attack, so we can set traps in the final authentication to guard these attacks.

Step 7: After the remote server receives the messages, the remote server computes $h\,(M_7 \,||\, R_s)$ and check $h\,(M_7 \,||\, R_s)$ is equal to M_{12} or not. If they are equivalency then the

server accepts the login request. If the result is difference, then the server refuses the login request.

In this section, we use the property of modular authentication to filter incorrectly password. By way of modular verification we can avoid too much overhead cost in authentication phase and also can resistant to the malicious dictionary attack. In this scheme, we use modular property to achieve the first authentication and use the encrypted key stored in the smart card to complete the final authentication. The client and the server use their major metrics R_{ci} and R_s to verify their identity, respectively. In a word, the smart card achieves the client password verification and the remote server authentication.

3.4 Password changing phase

In this scheme, we have to link to the remote server to change the password, we hope that the changing password process is rigorous. If the user wants to change his/her password, he/she has to login to the remote server and then sends the changing password request to the authentication server AS. Changes of any personal information must be verified, so the user has to input his/her information again.

Step 1: When the user successes to login to the remote server, the user offers his/her B_i', PW_i^{old}, and PW_i^{new}.

Step 2:Check the similarity of B_i and B_i'. If the similarity of B_i and B_i' is not enough to pass the verify then smart card interrupt the changing password request. Otherwise, the smart card computes $m_i{}' = h\left(PW_i^{old}\right) \oplus h\left(B_i\right)\ mod\ X$ and continues the next step.

- Password changing phase

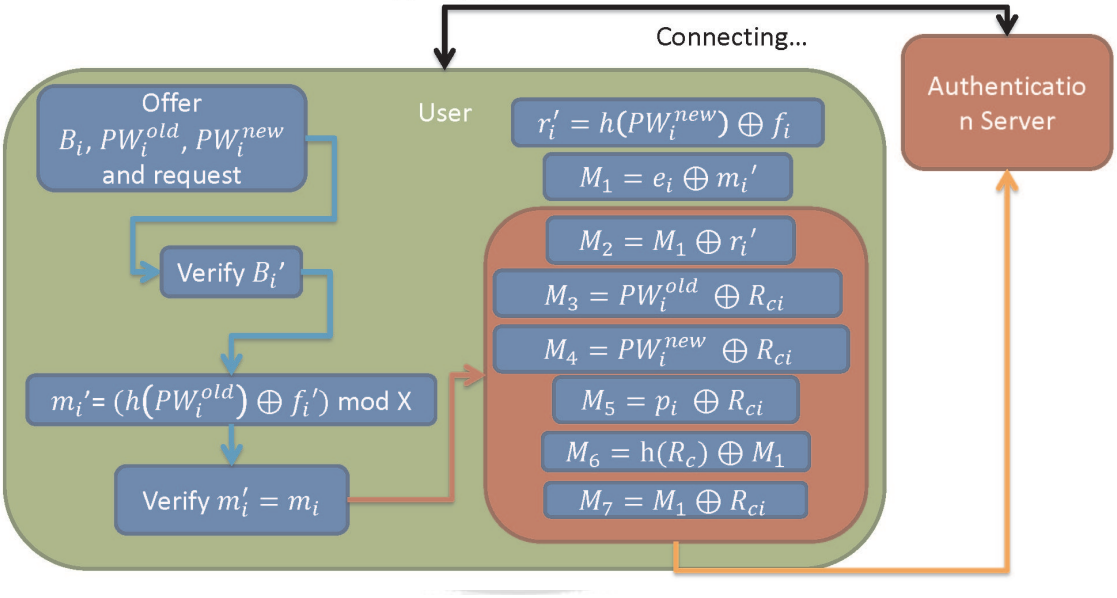

Figure 3.8: Part 1 of Password change phase

Step 3: Verify $m_i' = m_i$ or not, if $m_i' = m_i$ then smart card rejects the request and logouts. Otherwise, the smart card computes the followings:

$$r_i' = h\left(PW_i^{new}\right) \oplus\ h\left(B_i\right)$$

$$M_1 = e_i \oplus m_i'$$

$$M_2 = M_1 \oplus r_i'$$

$$M_3 = PW_i^{old} \oplus R_{ci}$$

$$M_4 = PW_i^{new} \oplus R_{ci}$$

$$M_5 = p_i \oplus R_{ci}$$

$$M_6 = h\left(R_{ci}\right)\ \oplus\ M_1$$

$$M_7 = M_1 \oplus R_{ci}$$

Then the user sends the messages tuple $(ID_i,\ M_2,\ M_3,\ M_4,\ M_5,\ M_6,\ M_7)$ to AS.

Step 4: When the server receives messages, it computes followings:

$$M_8 = h\left(ID_i \,||\, X_s\right)$$

$$M_9 = M_8 \oplus M_7 = R_{ci}$$

If $h\left(M_9\right) \neq M_6\ \oplus\ M_8$, then server aborts the changing password request.

Step 5: Then, server continues following calculations

$$M_{10} = M_8 \oplus M_2 = r_i'$$

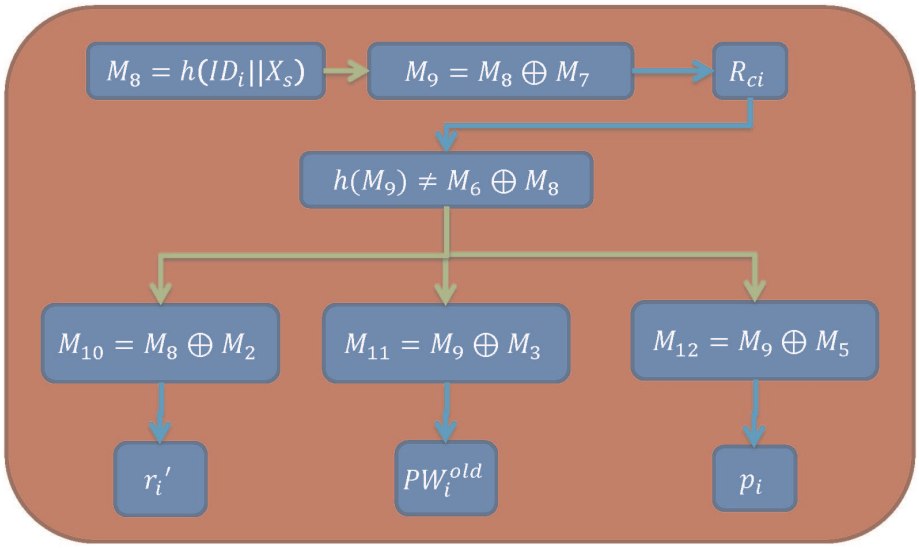

Figure 3.9: Part 2 of Password change phase

$$M_{11} = M_3 \oplus M_9 = PW_i^{old}$$

$$M_{12} = M_9 \oplus M_5 = p_i$$

Step 6: S_i computes $p_i{}' = h\left(ID_i \,||\, X_s \,||\, PW_i^{old}\right)$ and verifies $p_i{}' = p_i$ or not. If it failed, it means the password incorrectly. Otherwise, server AS rejects the changing password request.

Step 7: After the system certified B_i' and PW_i^{old}, the server generates a new X and computes the followings:

$$M_{13} = M_4 \oplus M_9 = PW_i^{new}$$

$$M_{14} = R_s \oplus (r_i{}' \; mod \; X)$$

$$M_{15} = R_s \oplus h\left(ID_i \,||\, X_s \,||\, PW_i^{new}\right)$$

$$M_{16} = R_s \oplus X$$

$$M_{17} = h\left(R_s\right)$$

$$M_{18} = M_8 \oplus R_s$$

Then, the remote server sends messages tuple $(M_{14},\ M_{15},\ M_{16},\ M_{17},\ M_{18})$ to user.

Step 8: The user receives messages tuple $(M_{14},\ M_{15},\ M_{16},\ M_{17},\ M_{18})$ and computes the followings:

$$M_{19} = M_1 \oplus M_{18} = R_s$$

If $h\left(M_{19}\right) \neq M_{17}$, then server aborts the changing password request.

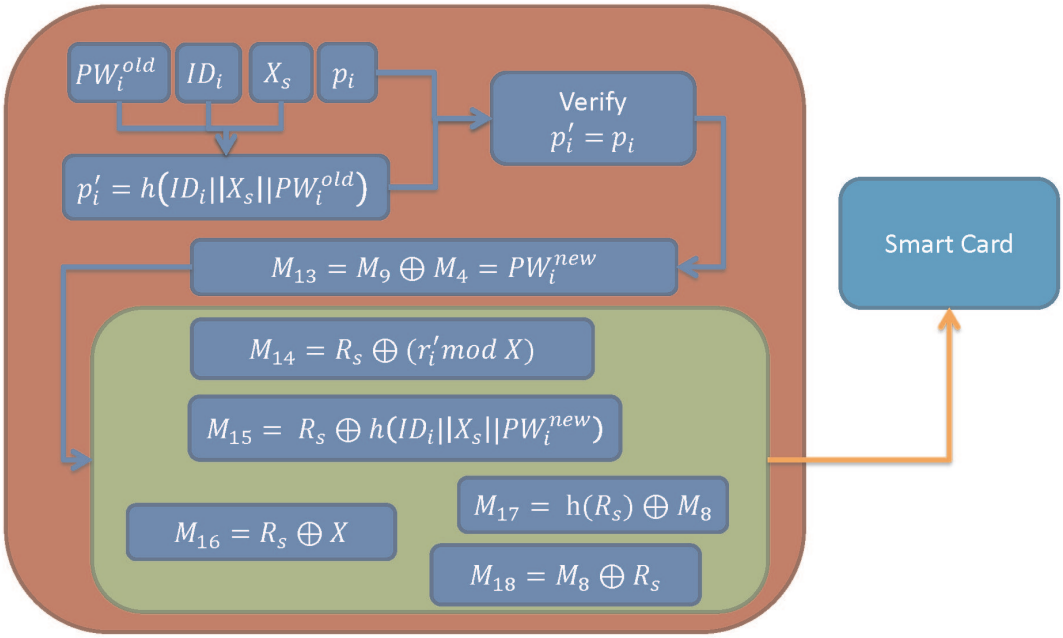

Figure 3.10: Part 3 of Password change phase

Step 9: If $h\left(M_{19}\right) = M_(17)$, the user decrypts messages$(M_{14},\ M_{15},\ M_{16})$ by R_s:

$$M_{20} = M_{14} \oplus R_s$$

$$M_{21} = M_{15} \oplus R_s$$

$$M_{22} = M_{16} \oplus R_s$$

Step 10: Finally, the smart card replaces M_{20} with m_i, M_{21} with p_i, and M_{22} with X. Thus the changing password request is completed.

In this phase, we appeared the method of changing password with the remote server. In order to resist the masquerading attack and the parallel attack, we use a random number to achieve the communication, and we also verify the password and the biometric template again. The authentication avoids someone using his/her new password to change the password in the login state, and interrupt the changing password request when the user entered his/her password incorrectly.

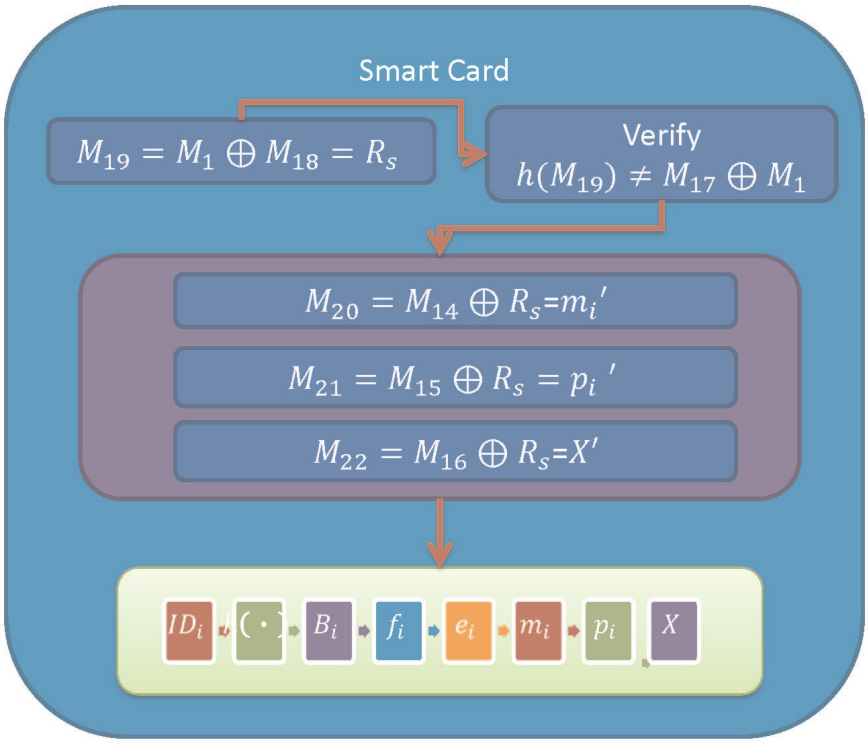

Figure 3.11: Part 1 of Password change phase

Chapter 4

Security analysis and comparisons

We will analysis the security of this scheme in the chapter and compare to related schemes. The analysis will consult Li-Hwang and A.K. Das's schemes.

4.1 Security analysis of the proposed method

4.1.1 Messages were stole during communication

In this scheme, we only stored X_s in the remote server, and didn't stored password table or any extra information. An attacker cannot extract the X_s from M_2 in login phase and (M_8, $M10$, M_{11})in authentication phase. These messages were protected by random value (R_{ci}, R_s), and they were also protected by one-way hash function. One-way hash function cannot be operation inversely operation and the attacker cannot get the random values to extract X_s.

4.1.2 Resisting replay attacks

Assume an attack want to steal the certificate information by interception of communications to complete the replay attack. It is impossible because these messages were protected by R_{ci}, R_s and hash function. Moreover, R_{ci} and R_s both are one-time random number. Based on the above conditions, the system can effective resisting replay attack.

4.1.3 Resisting masquerade attacks

Assume that, an attack tries to intercept the messages ID_i, M_2, $M3$, and M_4 from client and then masquerades as remote server. In authentication phase, the attacker has to compute the M_8 correctly, otherwise the client doesn't believe it is the server. It is impossible to masquerades correctly because the attacker without secret information X_s and random value R_{ci}.

4.1.4 Resisting parallel session attacks

As in A.K. Das and Li-Hwang's schemes, the parallel session attack can be effectively resistance in this scheme. The scheme didn't store all random values, if an attacker steals and resend login messages ($ID_i,\ M_2,\ M3,\ M_4$) to the server during login phase, this message will be verify in authentication and responses messages ($M_8,\ M_7,\ M9,\ M_{10}$) that was protected by difference R_s in every session. As a result, the attack will be failed in authentication.

4.1.5 Resisting smart-card-theft attacks

If any legal user lost his/her smart card, the finder of the card cannot logins to the system or changes the password because he/she has to pass the biometric verify. The illegal user's biometric template is almost impossible match with the legal user's biometric template in the smart card.

4.1.6 Resisting password guessing attacks

In Li-Hwang's scheme, there is no verifying mechanism in password change phase. They use the password to enhance security in login phase, but the attacker may destroy the secret

information in the smart card. In A.K. Das's scheme, they use a verify password mechanism for cutting down the server's computing cost in login phase and password change phase. And, they also avoid that the attacker destroys the secret information in the password phase. But the verify password mechanism is processed in the smart card that cannot resist the dictionary attacks. The attackers just go on repeating the same verify password mechanism to observe the result from the smart card. If the smart card sends messages to the server, it means the password correctly. Otherwise, the input passwords are incorrectly and then the attacker continues to try the passwords. In this scheme, we use a property of modulo check for verifying. The property of modulo check for verifying can reduce the risk of the dictionary attacks. If an attacker tries to attack in the smart card, when the attack hits the password that is as same as the remainder in the smart card after modular arithmetic, the attacker passes the first authentication but the password is incorrectly, so the attacker will be reject in step 6 of authentication phase. As a result, the attacker cannot differentiate which password is correctly or not, even if the lucky wrong password can also pass the first authentication.

4.2 Performance comparisons

We show the performance and compare with the related schemes in this section. Table 4.1 shows the results of comparisons. In this scheme, we never use the exponential operations because the exponential operations are very larger power and time wastage. In addition to the one-way hash function is used in this scheme, we used the modular arithmetic function to reduce the dictionary attacks and the communication attack risks. The computational cost of modular arithmetic is not as low as one-way hash function, but the computational cost is much lower than the exponential operations. Although the scheme sacrifices a few computational cost but the victim is replaced by more security.

Table 4.1: Performances comparison with related schemes

	Lin-Lai[12]	Lee-Chiu[10]	Yoon et al.[9]	Chang et al.[?]	Khan et al.[7]	Li -Hwang[11]	A.K. Das[1]	The proposed scheme
P_1	$1H+1E$	$2H+1E$	$1H$	$2H$	$2H$	$3H$	$3H$	$4H$
P_2	$2H+2E$	$2H+1E$	$1H$	$2H$	$2H$	$2H$	$2H$	$2H$
P_3	$1H+2E$	$2H$	$4H$	$6H$	$6H$	$5H$	$8H$	$9H$
P_4	*yes*	*yes*	*yes*	*no*	*yes*	*yes*	*yes*	*yes*
P_5	*no*	*no*	*yes*	*yes*	*yes*	*yes*	*yes*	*yes*
P_6	*no*	*no*	*no*	*yes*	*no*	*yes*	*yes*	*yes*
P_7	*yes*	*no*	*no*	*no*	*yes*	*yes*	*yes*	*yes*
P_8	*no*	*no*	*no*	*no*	*no*	*no*	*yes*	*yes*
P_9	*no*	*no*	*no*	*no*	*no*	*no*	*no*	*yes*

H:operation of one-way hash function,
E:operation of exponential,
P_1:computational operations in registration phase,
P_2:computational operations in login phase,
P_3:computational operations in authentication phase,
P_4:whether supports change password or not,
P_5:whether supports mutual authentication or not,
P_6:whether supports without synchronized clocks or not
P_7:whether provides non-repudiation or not
P_8:support client-side password pre-check
P_9:prevent password dictionary attack

Chapter 5

Conclusion

In this thesis, we proposed a secure low communication risk biometric-based remote user authentication scheme using smart cards. In order to reduce the communication attacks risks, we adopted the client-side password authentication. Resistance to the password dictionary attacks, we used the modular arithmetic mechanism. And, we never use the exponential operations because the exponential operations are very larger power and time wastage. By the way, the scheme still possesses several advantages; without synchronized clock, freely changes password, low computation cost, mutual authentication and non-repudiation. The authentication mechanism in this scheme not only has the above advantages but also more security than related schemes.

Moreover the improved method of biometric verify can solve that the difference of biometric input in hash function.

Although we sacrifice the freely change password, but we achieved a more secure and low risk authentication mechanism in remote user authentication.

Bibliography

[1] A.K. Das. "Analysis and improvement on an efficient biometric-based remote user authentication scheme using smart cards". *Information Security, IET*, 5:145–151.

[2] Lei Fan, J.H. Li, and H.W. Zhu. "An enhancement of timestamp-based password authentication scheme". *Computers and Security*, 21:7:665–667, Nov. 2002.

[3] B.T. Hsieh, H.Y. Yeh, H.M. Sun, and C.T. Lin. "Cryptanalysis of a fingerprint-based remote user authentication scheme using smart cards". In *Proceedings of 37th IEEE conference on security technology*, pages 349–350, 14-16 Oct. 2003.

[4] M.S. Hwang and C.Y. Liu. "Authenticated encryption schemes: current status and key issues". *International Journal of Network Security*, pages 61–73, Feb. 2005.

[5] A.K. Jain, A. Ross, and S. Prabhakar. "An introduction to biometric recognition". *IEEE Transactions on Circuits and Systems for Video Technology*, pages 4–20, Jan. 2003.

[6] M.K. Khan and J. Zhang. "Improving the security of a flexible biometrics remote user authentication scheme". *Computer Standards and Interfaces*, 29:82-85, Jan. 2007.

[7] M.K. Khan, J. Zhang, and X. Wang. "Chaotic hash-based fingerprint biometric remote user authentication scheme on mobile devices". *Chaotic Solitons Fractals*, 35:519-524, Feb. 2008.

[8] L. Lamport. "Password authentication with insecure communication". *Communications of the ACM*, 24:1:770–772, Nov. 1981.

[9] J.K. Lee, S.R. Ryu, and K.Y. Yoo. "Fingerprint-based remote user authentication scheme using smart cards". *Electronic Letters*, 38:554-5, Dec. 2002.

[10] N.Y. Lee and Y.C. Chiu. "Improved remote authentication scheme withsmart card". *Computer Standards and Interfaces*, pages 177–180, Feb. 2005.

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

[12] C.H. Lin and Y.Y. Lai. "A flexible biometrics remote user authentication scheme". *Computer Standards and Interfaces*, 1:19-23, Jan. 2004.

[13] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar. *"Handbook of fingerprint recognition"*. IEEE Secur. Privacy Mag, 2nd edition, Springer, New York, 2009.

[14] S. Prabhakar, S. Pankanti, and A.K. Jain. "Biometric recognition: security and privacy concerns". *IEEE Security and Privacy Magazine*, pages 33–42, Feb. 2003.

[15] J.J. Shen, C.W. Lin, and M.S. Hwang. "Security enhancement for the timestamp-based password authentication using smart cards". *Computers and Security*, pages 591–595, Jul. 2003.