CRYPTANALYSIS AND IMPROVEMENT OF DYNAMIC ID BASED REMOTE USER AUTHENTICATION SCHEMES USING SMART CARD

Thesis

Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

MANJUNATH VISHWESHWAR HEGDE

DEPARTMENT OF MATHEMATICAL & COMPUTATIONAL SCIENCES

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE - 575025

July, 2019

Dedicated to
My Family

DECLARATION

By the Ph.D. Research Scholar

I hereby declare that the Research Thesis entitled **CRYPTANALYSIS AND IMPROVEMENT OF DYNAMIC ID BASED REMOTE USER AUTHENTICATION SCHEMES USING SMART CARD** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfillment of the requirements for the award of the Degree of **Doctor of Philosophy** in **Mathematical and Computational Sciences** is a bonafide report of the research work carried out by me. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.

Manjunath Vishweshwar Hegde
Reg. No.: 148007 MA14F04
Department of Mathematical and Computational Sciences

Place: NITK, Surathkal.
Date: 12 July 2019

CERTIFICATE

This is to certify that the Research Thesis entitled **CRYPTANALYSIS AND IMPROVEMENT OF DYNAMIC ID BASED REMOTE USER AUTHENTICATION SCHEMES USING SMART CARD** submitted by **Manjunath Vishweshwar Hegde**, (Reg. No.: 148007 MA14F04) as the record of the research work carried out by him, is accepted as the Research Thesis submission in partial fulfillment of the requirements for the award of degree of **Doctor of Philosophy**.

(Dr. R Madhusudhan)
Research Supervisor

Chairman - DRPC

ACKNOWLEDGMENT

I would like to acknowledge the moral and intellectual supports given to me by my supervisor Dr. R. Madhusudhan during my Ph.D. program. His way to do research, as well as his attitude towards study and analysis of any particular subject, influenced me immensely, and I still feel there is a lot to learn from him. Among other things, I have always admired his ability to discuss research problems from scratch to formalize rigorously, his scientific bravery in supporting ideas that sounded utterly at first glance. Also, his ideas and his style of writing have always impressed me and influenced my style of work. Most important of all was probably his calm and constant belief in my ability to get a Ph.D.

I would like to take this opportunity to thank Dr. B. R. Shankar, Head of the department, Mathematical and Computational Sciences, for his guidance and cooperation during my coursework. I also extend my sincere thanks to Dr. Annappa, Professor, Department of Computer Science and Engineering for his advice during my coursework.

I would also like to thank my parents, my wife and my sisters for their moral support and patience during my research work.

Finally, I would like to thank all of them whose names are not mentioned here but have helped me in any way to accomplish the work.

Place: NITK, Surathkal Manjunath Vishweshwar Hegde

Date: 12 July 2019

ABSTRACT

Distribution of resources and services via open network has become the latest trend in information technology. In the open network, hackers can easily obtain the communication data. Therefore, open network demands the security to protect data and information. Hence, network security is a most important requirement in a distributed system. In the security system, authentication plays a major role. User authentication is a central component of any security infrastructure. Other security measures depend upon verifying the identity of the sender and receiver of information. Authorization grants privileges based upon identity. Audit trails would not provide accountability without authentication. Confidentiality and integrity are broken if we can't reliably differentiate an authorized entity from an unauthorized entity. Remote user authentication is a mechanism to identify the remote users over an insecure communication network. In remote user authentication, password authentication is the simplest method to authenticate the user. But, the limitations in the password authentication approach leads towards the development of two-factor authentication. There are hundreds of remote user authentication schemes have been proposed by many researchers. None of the schemes achieve all the security goals and many schemes fail to provide security against various attacks. Even though some of the schemes provide the security, they are not efficient in terms of computation and communication cost. Hence, it is necessary to design an efficient and secure authentication scheme.

This thesis aims to provide efficient and secure remote user authentication schemes in distributed systems and networks. There are many factors involved in authentication schemes and these factors use the characteristics of the password, smart card and biometric. This research concentrates on cryptanalysis and improvements of the smart card based two-factor remote user authentication schemes. Till date, many smart card based remote user authentication schemes have been proposed. But, every scheme has its security flaws. None of the schemes have succeeded to achieve all the security re-

quirements and goals. Also, many schemes do not provide a strong formal proof to prove the security of the scheme. In this thesis, cryptanalysis of the recently proposed remote user authentication schemes has been done to identify the vulnerabilities. New schemes have been proposed to overcome the identified security flaws. Security of the proposed schemes has been formally analyzed using BAN logic. Furthermore, the proposed schemes have been simulated using Automated Validation of Internet Security Protocols and Applications (AVISPA) tool. Through this simulation, it has been ensured that the proposed scheme is secure against all attacks.

In the literature study, it is observed that to avoid the replay attack, many remote user authentication schemes depend on clock synchronization. But the clock synchronization has its own disadvantages. Also, the schemes, which are independent of clock synchronization are vulnerable to replay attack. To fix these weaknesses, a novel authentication scheme has been proposed. By employing the Elliptic Curve Diffie-Hellman (ECDH) key exchange algorithm, the proposed scheme resists the replay attack. Through the security analysis, it is proved that the scheme achieves all the security goals and resists well-known attacks like insider attack, offline password guessing attack, etc. The proposed scheme security have been analyzed using BAN logic and simulated in AVISPA tool. Through these result, it is ensured that the proposed scheme resists all security attacks.

The contributions of this thesis is to the improve the security of the existing authentication schemes. In particular, this research analyzes the Wen and Li, Ding et al. and Troung et al.'s schemes. However, the analyzed schemes have many security flaws like fail to provide user anonymity and forward secrecy, vulnerable to the stolen smart card attack, insider attack, guessing attack etc. Based on the analysis, this research proposes improved schemes to overcome the identified weaknesses. Furthermore, a novel authentication scheme has been proposed to avoid the replay attack without clock synchronization. Finally, the thesis presents concluding remarks and discusses the future scope.

Keywords: Network Security, Authentication, Dynamic ID, Smart Card, Two-factor Authentication, Cryptography, Security, Wireless Technology

Contents

List of Figures

Acronyms

1. ID - Identity
2. SG - Security Goals
3. SR - Security Requirement
4. U - User
5. PW - Password
6. S - Server

Chapter 1

INTRODUCTION

1.1 Overview of Network Security

The Internet has become an integral part of everyday life. With the rapid development of the Internet technology, people can access any service from any place and at any time. Major commercial organizations, educational institutes, governments and individuals depend upon the Internet for providing their services. Most of the information exchange will be done through the Internet. Nowadays the number of threats are rising, hacker tools are becoming more sophisticated and powerful. If the network is not secure then the rate of unauthorized access, use, alteration, theft or physical damage to an object that maintaining high confidential information will increase. Therefore, in the present situation, it is important to provide security against numerous malicious users and attacks. They not only disrupt the services but also can steal the sensitive information. Therefore network security has become more important to every user of the Internet.

According to National Institute of Standards and Technology (Guttman and Roback, 1995) computer security is the protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability and confidentiality of information system resources (includes hardware, software, firmware, informal/data and telecommunication). Network security uses the same basic set of controls as computer security.

Most definitions of network security have the intent to consider the security of the network as a whole, rather than as an endpoint issue. A comprehensive network secu-

rity plan must encompass all the elements that make up the network and provide five important services:

1. *Authentication:* Authentication is a service which provides a system with the capability to verify that a user is the one who claims to be authenticated based on what the user is, knows, and has.

2. *Access Control:* In network security, access control is the ability to limit and control the access to host systems and applications via communication link.

3. *Data Confidentiality:* Data confidentiality ensures that the information transmitted across the network is accessible only by the intended recipients.

4. *Data Integrity:* The integrity service protects data against the active threats such as those that may alter data. It ensures that a message has not been modified in transit. It is an assurance of the exactness of received data from the authorized user.

5. *Non Repudiation:* This is a security service that provides proof of origin and delivery of service and/or information. It ensures that the originator of the message cannot deny that he/she sent the message.

1.2 Overview of Authentication

Authentication is a basic unit of network security. According to Kizza (2009) authentication is a service used to identify a user. User authentication is a central component of any security infrastructure. Other security measures depend upon verifying the identity of the sender and receiver of information. Authorization grants privileges based upon identity. Audit trails would not provide accountability without authentication. Confidentiality and integrity are broken if we can't reliably differentiate an authorized entity from an unauthorized entity.

There are many authentication methods used in the current network system. Out of these methods, password authentication is the simplest and widely used authentication method in current technology. But some vulnerabilities affect the security of

the password authentication system. Making poor password choices, which leave them vulnerable to password cracking. Sharing the password with friends and colleagues, so that the secret knowledge becomes public. A shared secret no longer remains under the control of the legitimate user. Sticking with the same password for long periods increases the opportunity for an impostor in the event of the password discovery and using the same password on multiple systems, with the consequence that a breach on one system potentially renders the others vulnerable.

The problems in the password authentication made an approach to introduce the smart cards authentication. There are many reasons to use the smart card, but the main reasons are the built-in security features and cost. Due to low cost, the portability, efficiency, and the cryptographic capacity, smart cards have been widely adopted in many E-Commerce applications and network security protocols. In this thesis, we mainly focused on the smart card based remote user authentication schemes, their weaknesses, and the possible improvements of those schemes. More information about smart card based authentication is given in further sections and chapters.

In traditional cryptosystem, most of the user authentication methods are based on secret keys and passwords. But if the keys are not kept secret or shared with nonlegitimate users, then it is not possible to provide security (Xiao, 2005). Therefore authentication systems are designed based on physiological and behavioral characteristics of human beings, like face, eye, handwriting, voice and fingerprint (Uludag et al., 2004). These authentication systems are also known as biometric authentication systems. According to Wayman et al. (2005), biometric technologies are automated methods of verifying or recognizing the identity of a living person based on physiological or behavioural characteristics.

Compared to other authentication systems, biometric system has the failure in the rate of registration. Not all users can use any given biometric system. For example, people without hands cannot use fingerprint or hand-based systems. Similarly, visually impaired people have difficulties using iris or retina based techniques. In Biometric system, the user authentication can be successful only when user's biometric characteristics are fresh and have been collected from the user being authenticated. This implies

that the biometric input device must be trusted. One more disadvantage of the biometric system is the use of this system may also imply the loss of anonymity. While one can have multiple identities when authentication methods are based on something the password or smart card, biometric systems can sometimes link all user actions to a single identity (Xiao, 2005).

1.3 Smart card authentication

The smart card is a type of chip-based identification card. Its characteristic feature is, an integrated circuit embedded in the card, which has a component for transmitting, storing and processing the data. Data can be transmitted using either contact on the surface of the card or electromagnetic fields without any contact.

1.3.1 Smart card authentication system model

The traditional model of smart card based remote user authentication system is consists of three phases, (1) registration phase, (2) login and authentication phase and (3) password change phase. In the registration phase, the user selects the identity (ID), password (PW) and submit it to the server. After submission of user credentials, server issues the smart card to the user. Registration phase of smart card authentication system is shown in Figure 1.1

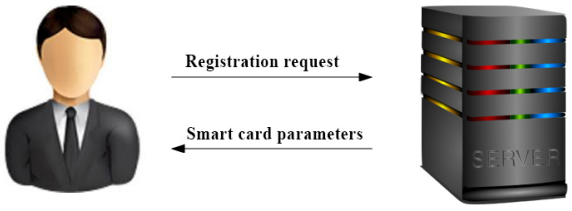

Figure 1.1 Registration phase of smart card authentication system

In login and authentication phase, a user inserts the smart card into the card reader and enters ID and PW. The smart card computes login message and sends it to the server S. On the server side, S receives login message and verifies it. Server rejects the login

message if the ID and PW are wrong. Otherwise it provides the service to the user. During the password change phase, the user can change his/her password. This can be done locally or interaction with the server. The login and authentication phase of smart card authentication system is shown in figure 1.2.

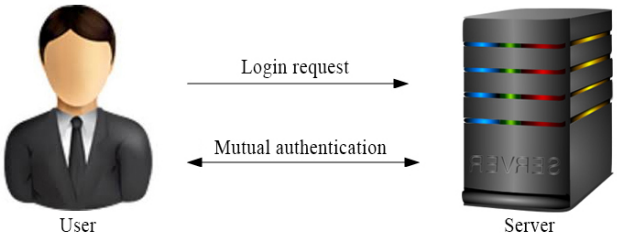

Figure 1.2 login and authentication phase of smart card authentication system

1.3.2 Advantages of smart card authentication

There are many reasons to use the smart card, but the main reasons are the built-in security features and cost. Due to low cost, the portability, efficiency, and the cryptographic capacity, smart cards have been widely adopted in many E-Commerce applications and network security protocols.

Another reason for using smart card authentication is, the card itself provides a computing platform on which information can be stored and computations can be performed securely. Most of today's systems need proper identification technique and it is a crucial part of the access control that makes the major building block of any system's security.

1.4 Dynamic ID based authentication

In a network environment, it is important to take care of information that communicates through the network. In the traditional distributed environment, to access the services, the user must login to the server. The process of the login and authentication are already illustrated in the previous section. To perform secure authentication, a strong scheme is required. Earlier authentication schemes are based on static login ID. Das et al. (2004) claimed that the static login ID leaks partial information about user'slogin message to the adversary. They also said that the adversary can intercept the login ID and he/she

can try to manipulate it with other intercepted parameters to forge the login ID.

To overcome this problem Das et al. (2004) proposed dynamic ID based remote user authentication scheme. Dynamic ID is a temporary user identity, calculated using user's static identity. The client system will calculate this dynamic ID and with this temporary ID, the system will send the login message to the server. Dynamic ID resists ID theft. Therefore, the concept of Dynamic ID has been used in our research.

1.5 Related Work

Lamport (1981) proposed first remote user authentication scheme. This scheme maintains a password table in server side to verify the user. Later, Hwang and Li (2000) proved that Lamport's scheme is inefficient if the intruder steals the password table or modify the table. Further, Hwang et al. (1990) proposed a non-interactive authentication scheme without password table. This idea was inspired from Shamir (1984) ID-based signature scheme. These schemes have used only password for authentication.

Yang and Shieh (1999) proposed a smart card based authentication scheme. Later, Hsu (2004); Hwang and Li (2000); Song (2010); Sun (2000); Xu et al. (2009); Yoon et al. (2005) and many other researchers proposed authentication schemes using the smart card. But, these schemes used static identity for an authentication process. i.e. over insecure communication channel user ID will be transmitted in plain text format. The user ID holds some information about the user. Hence, transmission of user ID over an open channel in the plain text formats violates users privacy, if an attacker steals these user ID. Therefore to protect the ID theft, researches adopted the dynamic ID technique in remote user authentication scheme. Dynamic ID is a temporary user identity, calculated by the user system. This dynamic ID will be different in every login session.

Das et al. (2004) introduced dynamic ID-based remote user authentication scheme to provide user anonymity. Later, Liao et al. (2005) identified the guessing attack in Das et al.'s scheme and developed an improved scheme. Further, Yoon and Yoo (2006) proved that Liao et al.'s scheme is vulnerable to reflection and insider attacks. To enhance the security of Liao et al.'s scheme, Yoon and Yoo proposed a new scheme. Wang

et al. (2009) reviewed Das et al.'s scheme and showed that user authentication can be done independently of the password and proposed a new scheme to resolve identified weaknesses.

Yeh et al. (2010) identified the impersonation and man in middle attacks in Wang et al. (2009) scheme and developed an authentication scheme to withstand the identified weaknesses. Later, Wang et al. (2011) analyzed the Wang et al. (2009) scheme and pointed out the known key attack and smart card loss attack in the scheme. To resist the identified weaknesses, Wang et al. proposed an elliptic curve cryptosystem based remote user authentication scheme. Wang et al. (2009) scheme was also analyzed by Wen and Li (2012). They showed that the scheme is vulnerable to impersonation attack and insider attack. To overcome identified weaknesses Wen and Li proposed a new remote user authentication with key agreement scheme.

Khan et al. (2011) reviewed Wang et al.'s scheme and proved that the scheme failed to provide security against user anonymity and vulnerable to stolen smart card attack. They also proposed an improved scheme to resist the identified weaknesses. Further, An (2013) cryptanalyze Khan et al. scheme and proved that Khan et al.'s scheme is sensitive to password guessing and forgery attacks. He also proved that the scheme does not provide user anonymity. To overcome pointed weaknesses he proposed an improved scheme. Later, Ding et al. (2012) analyzed Chen et al. (2011) scheme and identified that Chen et al. scheme is vulnerable to offline password guessing attack, key compromise impersonation attack, and known key attack. To overcome identified weaknesses, Ding et al. developed an upgraded methodology.e. Further, Li et al. (2013a) reviewed Lee et al. (2011) scheme and identified the forgery and server spoofing attacks in the scheme. They also identified the improper authentication and inefficient password change phase in Lee et al. 's scheme. To improve the identified weaknesses, Li et al. proposed a new dynamic id remote user authentication scheme. In the same year, Li et al. (2013b) reviewed Chen et al. (2014) scheme and identified the perfect forward secrecy, delay in wrong password detection and unfriendliness and inefficiency in password change phase. To overcome these weaknesses Li et al. proposed a new authentication scheme.

Kumari et al. (2014) reviewed the Chang et al. (2014) scheme and identified that,

the scheme is vulnerable to password guessing attack, impersonation attack, and masquerading attack. To overcome these weaknesses Kumari et al. proposed an improved remote user authentication scheme. Next, Truong et al. (2014) reviewed Sood et al. (2010) scheme and pointed out that Sood et al.'s scheme is vulnerable to spoofing attack and verifier attack. To overcome these weaknesses, Troung et al. proposed an elliptic curve cryptography based enhanced authentication scheme. Later, Li et al. (2015) reviewed Chang et al. (2014) scheme and pointed the offline password guessing attack, stolen smart card attack, insider attack and impersonation attack in the scheme. They also identified the user identity traceability and password change inefficiency in the Chang et al.'s scheme.

Maitra et al. (2017) reviewed a ElGamal based remote user authentication scheme, proposed by Lee et al. (2014). They identified forgery attack, stolen smart card attack, password guessing attack and smart card forgery attack in Lee et al.'s scheme. To overcome these vulnerabilities, Maitra et al. proposed an improved scheme. Recently, Wang et al. analyzed Maitra et al. (2017) scheme and identified offline password guessing attack and insider attack. Also, Wang et al. proved that Maitra et al.'s scheme does not provide perfect forward secrecy. To overcome these weaknesses, Wang et al. proposed a lightweight password authentication scheme.

Wang et al. (2016) made a comparative study of two-factor authentication schemes. In this study, they reviewed Li et al. (2013b) scheme, Odelu et al. (2015) scheme and Kumari and Khan (2014) scheme. They identified several weaknesses in the schemes based on the criteria set proposed by Madhusudhan and Mittal (2012). Nikooghadam et al. (2017) reviewed Kumari et al. (2014) scheme and Chaudhry et al. (2015) scheme and identified various pitfalls in the reviewed schemes. Then they proposed the new remote user authentication scheme to overcome the identified weaknesses. Li et al. (2017) proposed an anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks. Chandrakar and Om (2018) illustrated that existing schemes are not providing security against various threats and they proposed a remote user authentication and session key agreement scheme using the Rabin cryptosystem.

1.6 Security requirements and goals

Liao et al. (2006) proposed a set of ten requirements for evaluation of smart card security. They assumed that these requirements provide proper security to the smart card based authentication schemes. Yang et al. (2006) made an argument over Liao et al.'s criteria. and proposed a new set of five criteria as a solution. But, Yang et al.'s criteria set was a bit more theoretical and difficult to adopt in real applications. Later, Tsai et al. (2006) presented another group of security property. This has nine security requirement and ten desirable features, which are based on the tamper-resistant assumptions. Ambiguities and redundancies are identified in previous criteria by Madhusudhan and Mittal (2012). They also proposed a new set of nine security requirements and ten security goals. They are as follows:

1. SR1 *Denial of Service (DoS) Attack:* This attack blocks the normal use of communications facilities. An attacker can update the false verification information of a legal user for the next login phase. So that the legal user will not login successfully any more. Server rejects the login request of a specific user until re-registration.

2. SR2 *Forgery attack:* To access a remote system, attacker attempts to modify intercepted communications to act as the legal user. Similarly attacker also moves as a legal server to manipulate the data of the legal users.

3. SR3 *Parallel session attack:* By creating a valid login message, attacker can enter into the server as a legal user. Later attacker may start a parallel attack by replaying the server's response message as the user's login message.

4. SR4 *Password guessing attack:* Most of the passwords have low entropy that it is vulnerable to password guessing attacks. Also password that is easy to remember is also generally easy for an attacker to guess. Attacker intercepts authentication messages and stores them locally and then attempts to use a guessed password to verify the correctness of his guess using the authentication messages.

5. SR5 *Replay attack:* By intercepting the previous communications, an attacker can

imitate the legal user to login the system. The attacker can replay the intercepted messages. Attacker captures login messages of authorized user and resents it to the server, even though the messages may be encrypted and the attacker might not know what the actual user id and passwords. The retransmission of valid login message is sufficient to access for the server.

6. SR6 *Stolen-verifier attack:* An attacker who robs the password-verifier from the server and use that stolen-verifier to impersonate a legal user to login to the system. In many applications, the server stores hashed passwords instead of clear text passwords. That hashed passwords are enough to act as legal user.

7. SR7 *Insider attack:* It is a primary threat to the system. On many systems, the access control settings for security-relevant objects do not reflect the organization's security policy. This allows the insider to browse through sensitive data. Insider of the server can perform an off-line guessing to obtain password.

8. SR8 *Smart card loss attack:* When the smart card is loss or stolen, illegal users can easily change password of the smart card, or can guess the password of the user using password guessing method, or can impersonate the user to login the system

9. SR9 *Reflection attack:* It is a method of attacking a challenge-response authentication system that uses the same protocol in both directions. The essential idea of the attack is to trick the target into providing the answer to its own challenge.

A robust password authentication scheme should withstand all of the above attacks. It should also achieve the following goals.

1. G1 *Forward secrecy:* A scheme with perfect forward secrecy means, the system will never reveal the previous session keys, even if the system secret key is compromised. Once the session key is obtained, the entire session will become completely insecure.

2. G2 *User anonymity:* It is important to preserve the privacy of a user because user ID leaks partial information about the users login message to the adversary. With

the use of user ID, intruder can try to modify the other parameters.

3. G3 *Efficiency for wrong password login:* If the user inputs wrong password in login phase, instead of sending the user's login request to the server, without any delay user should come to know the wrong password entry, with the help of the error message.

4. G4 *Smart card revocation:* It is one of the requirements of smart card-based authentication schemes, that in case of cards getting lost, there should be provision in the system for invalidation of the further use of lost smart card, otherwise an adversary can impersonate the registered user.

5. G5 *Freely chosen password by the users:* The passwords can be chosen and changed freely by the users. If the password is chosen by the remote server without permission of the user, then he/she has no choice of choosing his own password. Also, password chosen by the server could be long or random. For a registered user, it might be difficult to remember.

6. G6 *Session key agreement:* A session key should be established during the password authentication process. Using session key a private secure channel has been established over a published channel. It is suitable that after the successful authentication process, both parties will communicate some secret messages, which should be encrypted to provide the confidentiality and secrecy of transmitted data.

7. G7 *Free from verification table:* The passwords or verification tables are not stored in the system or server. The remote system should not have a dictionary of verification tables even in the hash format.

8. G8 *Free from password reveal:* The passwords cannot be revealed by the insider of the server. If the user's password is revealed to the server, administrator of the server can try to use the same password for login to the other servers that adopt normal remote user password authentication schemes.

9. G9 *Password dependent:* The password independent scheme means that the scheme is equivalent to no password scheme. That means without using password or

sometimes using wrong password attacker can enter into the system as a legal user. So, the authentication scheme should be password dependent.

10. G10 *Mutual authentication:* Mutual authentication is a process in which both entities in a communication channel authenticate each other. In a network environment, the user and the server both authenticates each other before begin the communication. If the system is mutually authenticated with the user, then it is difficult for an attacker to manipulate sensitive data of the legal users.

1.7 Research objectives

Objectives of this research are as follows:

1. In the literature, several remote user authentication schemes have been proposed. However, most of them have security flaws and their improvements are also insecure against some possible attacks. Thus aim of this research is to give an insight into the most recent remote user authentication schemes and identify the security flaws, issues and challenges. In order to remove the security flaws, robust and efficient remote user authentication scheme has been proposed.

2. The second objective of this research is to propose a secure dynamic authentication scheme using smart card. The security of the scheme should be proved with a formal method and must have simulation results. To make the scheme more robust, elliptic curve cryptography has been adopted in the smart card based remote user authentication. This preserves the privacy and secrecy efficiently by taking minimum key length.

3. The third objective of this research is to propose a novel remote user authentication scheme using the smart card. In the literature study, it is observed that to avoid the replay attack, many remote user authentication schemes depend on clock synchronization. But the clock synchronization has its own disadvantages. Therefore, the proposed scheme should resist replay attack without involvement of clock synchronization.

1.8 Chapter organisation

This thesis contains the cryptanalysis and improvement of dynamic ID based remote user authentication schemes using the smart card. This chapter presents the introduction to the network security and the importance of user authentication in network security. Further, the chapter briefly gives an overview of authentication. Introduction to smart card based authentication, advantages of the smart card authentication and the traditional system model of the smart card authentication are illustrated in this chapter. This chapter also presents the security requirements and goals of an ideal smart card based authentication should satisfy and achieve respectively. The literature survey has been taken throughout the research is presented in this chapter. Finally, the discussion of the research objectives are presented in this chapter.

In Chapter 2, Wen and Li (2012) scheme and Ding et al. (2012) schemes have been reviewed. Through the cryptanalysis, the weaknesses in both the schemes have been identified. In Wen and Li's scheme, it is identified that, the scheme does not detect wrong password quickly, also the scheme is vulnerable to insider and stolen smart card attack. Further, the scheme does not provide forward secrecy. Similarly in Ding et al.'s scheme, it is identified that, the scheme does not provide user anonymity, vulnerable to offline password guessing attack, stolen smart card attack and insider attack. It is also identified that Ding et al.'s scheme does not provide proper perfect forward secrecy and smart card revocation. To solve these security weaknesses, an improved scheme has been proposed and proved its security through the cryptanalysis. The computation cost comparison with the proposed scheme and Wen and Li's scheme and Ding et al.'s scheme also done in this chapter. By the computation cost comparison, it is proved that the proposed scheme uses less number of hash operations and resists all attacks.

In Chapter 3, Truong et al. (2014) remote user authentication scheme has been reviewed. In this scheme, it is identified that the scheme is vulnerable to replay attack, does not provide user anonymity and perfect forward secrecy. It is also observed that the scheme fail to secure the server secret key and it does not allow the user to choose his/her own password. To overcome the these weaknesses, a new scheme has been proposed using elliptic curve cryptography. The security of the proposed scheme is

proved using BAN logic and it is simulated using AVISPA tool. The result of formal verification using AVISPA tool is presented in this chapter. By this simulation result, it is proved that the scheme is secure from all defined attacks. Efficiency factors of the proposed scheme with other related schemes are also discussed in this chapter. The computation cost comparison result of the proposed scheme with Wen and Li (2012), Ding et al. (2012) and Truong et al. (2014) schemes is presented. The result shows that the proposed scheme is more robust when compared to other schemes.

Chapter 4 attempts to resist replay attack without involvement of the clock synchronization. The proposed scheme includes Elliptic Curve Diffie-Hellman (ECDH) key exchange algorithm to overcome replay attack without clock synchronization. In the proposed scheme the dynamic public key methodology has been used i.e. generated ECDH public keys are different at every session. This technique helps to avoid replay attack and also protects the encryption and decryption keys. Through the security analysis, it can be conclude that the proposed scheme is more robust and efficient to implement practically. The formal verification proof has been presented using BAN logic. The simulation result of the proposed scheme using AVISPA tool also presented in this chapter. The functionality and computation efficiency of the proposed scheme has been calculated and compared the result with Wang et al. (2011), Khan et al. (2011), Chen et al. (2011), Wen and Li (2012), An (2013), Ding et al. (2012), Chang et al. (2014), Kumari et al. (2014), Truong et al. (2014) and Wang and Wang (2016) remote user authentication schemes. The comparison has been done on the basis of (C1) Resistance to replay attack without time synchronization, (C2) Session key security, (C3) User credentials privacy, (C4) Secure server secret key (C5) Resistance to stolen smart card attack.

Chapter 5 concludes the thesis with the summary of contributions, limitations of research and future work.

Chapter 2

SECURITY BOUND ENHANCEMENT OF REMOTE USER AUTHENTICATION SCHEMES USING SMART CARD

2.1 Introduction

With the rapid growth in the communication technology, people are carrying out various transactions through the Internet. Many sensitive and secret information is stored in the remote systems. This information can be accessed through the network. Presently, the Internet threats and hacker tools are becoming more powerful. When attacker becomes more strong, communication network becomes insecure. If the network is insecure, then the rate of unauthorized access, use, alteration, theft or physical damage to an object that maintains confidential information will increase. Therefore, knowledge protection of any system is a challenging problem.

Steiner et al. (1988) says that authentication is a fundamental building block for a secure networked environment. If, for example, a server knows for certain the identity of a client, it can decide whether to provide the service, whether the user should be given special privileges, who should receive the bill for the service and so far. Generally, authentication will be done based on what the user knows (Password), what the user has (hardware tokens, Smart cards) and what the user is (Biometric). Among these three methods, password authentication is the simplest and widely used authentication method. But, the limitations of traditional password authentication approached the development of two-factor authentication. Today, the systems which require more security like e-commerce, banking, healthcare are adopting the two-factor authentication.

This study aims to enhance the security boundary of smart card based remote user authentication. By the literature study, it has identified that none of the schemes satisfies all security requirements. Some schemes were not suitable for practical implementation. Therefore it is required to propose a new scheme which is efficient, practically implementable and fulfills all security requirements.

In this chapter, two remote user authentication schemes has been reviewed proposed by Wen and Li (2012) and Ding et al. (2012). Through the cryptanalysis, it is identified the weaknesses in both the schemes. To overcome the identified weaknesses, a new remote user authentication scheme using the smart card has been proposed.

The chapter organization is as follows. Section 2.2 presents the review of Wen and Li scheme. Section 2.3 discusses the cryptanalysis of Wen and Li scheme. Section 2.4 illustrates the review of Ding et al.'s scheme. Section 2.5 presents the cryptanalysis of Ding et al.'s scheme. Section 2.6 proposes a new scheme, which overcomes all the identified attacks. Section 2.7 discusses the security aspects of the proposed scheme followed by the comparison of computational cost and performance analysis of the proposed scheme with the reviewed schemes in section 2.8. Section 2.9 depicts the concluding remarks of the chapter.

2.2 Review of Wen and Li (2012) scheme

Wen and Li (2012) proposed an improved dynamic ID-based remote user authentication with the key agreement scheme. Through cryptanalysis, it is identified that the scheme does not detect the wrong password quickly. Also, the scheme is vulnerable to insider attack and stolen smart card attack. Further, the scheme does not provide forward secrecy. This section presents the review of Wen and Li's remote user authentication scheme. The notations used in the reviewed and the proposed schemes are presented in Table 2.1.

This scheme comprises six phases: Registration phase, Login phase, Authentication and key exchange phase, Mutual authentication and key confirmation phase, Off-line password change phase and Online secret renew phase. The phases of Wen and Li's scheme described one by one as follows:

Table 2.1 Notations and Descriptions.

Notations	Descriptions
S	Server
T_u, T	Current timestamps acquired at the user side
U_i	i^{th} user
ID_i	Identity of user
PW_i	Password of U_i
h(.)	One-way hash function
$\oplus$	Bitwise XOR operator
$\|\|$	Concatenation operator
T', T'', T_{ur}, T_s	Current timestamp acquired at the server side
x, d	Secret keys maintained by S

2.2.1 Registration phase

1. When U_i wants to register for the first time, he/she selects ID_i and PW_i and sends it to the remote server S through a secure channel.

2. S computes $n_i = \text{h}(ID_i \,||\, PW_i)$, where n_i is the user's ID number and it is kept by S to check the validity of the smart card.
   Further, S computes $m_i = n_i \oplus \text{x}$,
   $N_i = \text{h}(ID_i) \oplus \text{h}(PW_i) \oplus \text{h}(\text{x}) \oplus \text{h}(m_i)$
   where x is the server's secret number

3. S personalizes the smart card with the following parameters $\{\text{h}(.), N_i, n_i\}$.

4. S sends the smart card to U_i through the secure channel.

2.2.2 Login phase

When a user U_i wants to login to S, then U_i inserts his/her smart card in the terminal and inputs ID_i and PW_i. The smart card performs the following steps:

1. Computes $A_i = \text{h}(ID_i) \oplus \text{h}(PW_i)$,
   $B_i = N_i \oplus \text{h}(ID_i) \oplus \text{h}(PW_i) = \text{h}(\text{x}) \oplus \text{h}(m_i)$,
   $CID_i = \text{h}(A_i) \oplus \text{h}(\text{h}(n_i) \oplus B_i \oplus \text{h}(N_i) \oplus \text{T})$.

2. U_i sends the login request $M_1 = \{CID_i, n_i, N_i, \text{T}\}$ to the remote server S.

2.2.3 Authentication and key exchange phase

1. S receives the login request message $\{CID_i, n_i, N_i, \mathrm{T}\}$ and checks the validity of the timestamp T. S generates the timestamp T' and verifies the validity of the received message time T. Server checks the condition T'- T $\leq \Delta$T. If this condition holds and n_i is in the registered list then S performs further calculation, else it rejects the request message.

2. S computes $m_i = n_i \oplus \mathrm{x}$,
   $B_i = \mathrm{h(x)} \oplus \mathrm{h}(m_i)$,
   $A_i = N_i \oplus B_i = \mathrm{h}(ID_i) \oplus \mathrm{h}(PW_i)$.

3. S verifies the equation
   $CID_i \oplus \mathrm{h}(A_i) = \mathrm{h(h}(n_i) \oplus B_i \oplus \mathrm{h}(N_i) \oplus \mathrm{T})$. If it is equal, S computes
   C_i'= $\mathrm{h}(A_i \oplus \mathrm{T'} \oplus \mathrm{h}(n_i))$,
   SK = $\mathrm{h}(A_i \parallel \mathrm{T} \parallel B_i \parallel \mathrm{T'})$,
   KC = h $(B_i \parallel \mathrm{SK} \parallel \mathrm{T'})$.

4. S sends the replay message $M_2 = \{C_i', \mathrm{KC}, \mathrm{T'}\}$.

2.2.4 Mutual authentication and key confirmation phase

Once U_i receives the message $\{C_i', \mathrm{KC}, \mathrm{T'}\}$ from S at the time T˝and U_i performs the following:

1. Checks the validity of timestamp T'is valid or not.

2. If the time interval is valid, U_i computes $C_i = \mathrm{h}(A_i \oplus \mathrm{T'} \oplus \mathrm{h}(n_i))$ and verifies C_i with received C_i'. If $C_i = C_i'$, then the mutual authentication is successfully compleated.

3. Further, U_i computes
   SK = $\mathrm{h}(A_i \parallel \mathrm{T} \parallel B_i \parallel \mathrm{T'})$, and then checks whether the key confirmation message KC'is correct. If so, U_i computes KC = h $(A_i \parallel \mathrm{SK} \parallel \mathrm{T}˝)$.

4. U_i sends the last key confirmation message $M_3 = \{\mathrm{KC}, \mathrm{T}˝\}$.

5. S verifies the last key confirmation message, if the equation
   KC = h (A_i || SK || T´´) holds, this scheme completes the authentication process and allows the user to access the server.

2.2.5 Offline password change phase

When the user wants to change the password, he/she inserts the smart card into the terminal and enters ID_i with PW_i and the new password PW_i^*, then the smart card performs the following offline password change phase:
$N_i^* = N_i \oplus h(PW_i) \oplus h(PW_i^*)$,
then the smart card replaces N_i with the new value N_i^*.

2.2.6 Online secret renew phase

When the remote server wants to renew its secret value x, S should interact with its clients and perform the following steps: When U_i and S have authenticated each other and established the secure session key SK by executing the related operations, a private secure channel has been established over a published channel. Then S computes:
$m_i = n_i \oplus x$,
$m_i^* = n_i \oplus x^*$,
$N_i^* = h(ID_i) \oplus h(PW_i) \oplus h(x^*) \oplus h(m_i^*)$, where x^* is the new secret key chosen by S.
S Sends N_i^* to U_i over the established private secure channel. At last, the smart card replaces N_i with N_i^*.

2.3 Cryptanalysis of Wen and Li key agreement scheme

This section discuss the security flaws of Wen and Li's Scheme. Through the security analysis it is proved that the scheme is vulnerable to insider attack and stolen smart card attack. It is also identified that the scheme is weak against perfect forward secrecy and inefficient to identify the wrong password quickly. The proofs are presented below.

2.3.1 No perfect forward secrecy

In Wen and Li's scheme, it is possible to calculate the previous session key, even if the server secret key is unknown.

Assume that the smart card is stolen, and the adversary has got security parameters of the smart card {h(.), N_i, n_i}, previous login and authentication information {CID_i, n_i, N_i, T}, {C_i', KC, T'} and key confirmation message {KC, T´´}. The intruder can calculate previous session key as follows. Consider the equation C_i'= h($A_i \oplus$ T'$\oplus$ h(n_i)). Here, adversary $\mathscr{A}$ can compute h(n_i) using the smart card parameter n_i, he/she can obtain C_i' and T' from authentication message. Therefore, $\mathscr{A}$ can get all the paramers except A_i.

Now adversary can guess the value of A_i as follows: $\mathscr{A}$ first guesses value A_i*, computes C_i* = h(A_i* $\oplus$ T'$\oplus$ h(n_i)) and compares C_i* with C_i'. If bothare equal then $\mathscr{A}$ has guessed A_i else $\mathscr{A}$ repeates the procedure until he/she get the correct A_i. Once $\mathscr{A}$ obtain A_i, adversary can compute $B_i = N_i \oplus$ h(ID_i) $\oplus$ h(PW_i) = $N_i \oplus A_i$,
where A_i = h(ID_i) $\oplus$ h(PW_i). Now, $\mathscr{A}$ has obtained all the parameters required to calculate the session key. Hence $\mathscr{A}$ calculates the session key SK = h(A_i || T || B_i || T'). Once the session key is obtained, the entire session will become completely insecure.

2.3.2 Vulnerable to smart card stolen attack

In Wen and Li's scheme, an intruder can enter into the server without using any password if he/she stolen the smart card. Assume that, the smart card has stolen and adversary $\mathscr{A}$ got security parameters of the smart card {h(.), N_i, n_i}, previous login and authentication informations {CID_i, n_i, N_i, T}, {C_i', KC , T'} and key confirmation message {KC, T´´}, then adversary can login to the server as follows.

Consider the equation C_i'= h($A_i \oplus$ T'$\oplus$ h(n_i)). $\mathscr{A}$ can calculate A_i using this equation. The method of obtaining A_i is illustrated in the section 2.3.1.

After computation of A_i intruder computes T´´- T'to obtain ΔT. Further, $\mathscr{A}$ calculates T* = T´´$\oplus$ ΔT and CID_i* = h(A_i) $\oplus$ h(h(n_i) $\oplus$ B_i $\oplus$ h(N_i) $\oplus$ T*). Further, $\mathscr{A}$ sends the crafted login message {CID_i*, n_i, N_i, T*} to the server.

On the other side, server S receives the {CID_i*, n_i, N_i, T*}, generates timestamp T** and checks the validity of T*. Here, the condition T** - T* $\leq \Delta$ will definitly holds and S receives the login message. Further, S computes $m_i = n_i \oplus$ x, B_i = h(x) $\oplus$ h(m_i), $A_i = N_i \oplus B_i$ = h(ID_i) $\oplus$ h(PW_i) and verifies whether the equation CID_i* $\oplus$ h(A_i) = h(h(n_i) $\oplus$ B_i $\oplus$ h(N_i) $\oplus$ T*) is holds or not. Because of message crafting thecondition

willdefinitly satisfies and the $\mathscr{A}$ will login without using any password.

2.3.3 Vulnerable to insider attack

In the registration phase of Wen and Li's scheme, user U_i submits ID_i and PW_i to the server S in the form of plain text. This facilitates insider to directly access user's password PW_i from the server S. Using user's password, an insider may spoof the legal user of the system.

2.3.4 Wrong password cannot be quickly detected

If the user inputs the wrong password in the login phase, instead of sending user's login request to the server, without any delay, the user comes to know the wrong password entry, with the help of error message. In the login phase of Wen and Li's Scheme, after entering the ID_i and PW_i, the smart card does not verify whether the entered password is correct or not. Instead of that, it sends the login request message to the server and the server checks whether the entered password is correct or not.

Assume that the user U_i entered a wrong password PW_i* in login phase. The smart card computes: $A_i = \mathrm{h}(ID_i) \oplus \mathrm{h}(PW_i*)$, $B_i = N_i \oplus \mathrm{h}(ID_i) \oplus \mathrm{h}(PW_i*)$,
$CID_i = \mathrm{h}(A_i) \oplus \mathrm{h}(\mathrm{h}(n_i) \oplus B_i \oplus \mathrm{h}(N_i) \oplus \mathrm{T})$, and U_i sends the login request $M_1 = \{CID_i, n_i, N_i, \mathrm{T}\}$ to the server. Other side, S receives the M_1 and checks the validity of timestamp T, if T'- T $\leq \Delta$T holds and n_i is in the registered list,
S computes $m_i = n_i \oplus \mathrm{x}$, $B_i = \mathrm{h}(\mathrm{x}) \oplus \mathrm{h}(m_i)$ and $A_i = N_i \oplus B_i = \mathrm{h}(ID_i) \oplus \mathrm{h}(PW_i)$,
Then S verifies whether the equation $CID_i \oplus \mathrm{h}(A_i) = \mathrm{h}(\mathrm{h}(n_i) \oplus B_i \oplus \mathrm{h}(N_i) \oplus \mathrm{T})$ holds or not. This equation actually verifies the correctness of the entered password. If the U_i inputs a wrong password at login phase, it cannot detects as early as possible and user U_i completely unaware about the wrong PW_i entry.

2.4 Review of Ding et al. (2012) scheme

Ding et al. (2012) analyzed Chen et al. (2011) scheme and identified that Chen et al.'s scheme is vulnerable to offline password guessing attack, key compromise impersonation attack, and known key attack. They also identified that Chen et al.'s scheme does not provide perfect forward secrecy and user anonymity. To overcome identified weak-

nesses, Ding et al. proposed a new scheme. By the cryptanalysis of Ding et al.'s scheme, it has identified that Ding et al.'s scheme is vulnerable to offline password guessing attack, server masquerading attack, stolen smart card attack and insider attack. It is also found that the scheme does not provide user anonymity, perfect forward secrecy and proper smart card revocation. To overcome these weaknesses, an authentication scheme has been proposed.

Ding et al.'s scheme mainly comprises of five phases: registration phase, login phase, authentication phase, password change phase and smart card revocation phase.

2.4.1 Registration phase

Before the registration of the new user, server S computes the private and public keys. To calculate these keys, S generates two large prime numbers p and q, computes $n = pq$ and z = (p − 1)(q − 1). Further, It selects a prime number e and d, where e, d ∈ [1, z] such that ed ≡ 1 mod z. At the end, (n, e) will be the public key and (n, d) will be the private key.

Registration phase involves the following steps:

1. U_i selects ID_i, PW_i and a random number b.

2. U_i sends the registration request message $\{ID_i, \text{h}(\text{b} \oplus PW_i)\}$ to S

3. Server receives registration message, generates a random value y_i and calculates the following:
   N_i=h($ID_i \parallel$ h(b ⊕ PW_i)) ⊕ h(d)
   A_i=h(h(b ⊕ PW_i)$\parallel$ ID_i) ⊕ h(y_i)
   B_i=y_i ⊕ ID_i ⊕ h(b ⊕ PW_i) and
   D_i=h(h($ID_i \parallel y_i$) ⊕ d).
   The server S deposits y_i ⊕ h(h(d) ∥ d) and ID_i ⊕ h(d ∥ y_i) corresponding to D_i into the database.

4. The parameters of smart card $\{N_i, A_i, B_i$, n, e, h(.)$\}$ are sent to the user.

5. U_i enters b into smart card.

2.4.2 Login phase

Login phase of Ding et al.'s scheme consists of the following steps.

1. U_i inserts smart card and inputs ID_i and PW_i.

2. The smart card computes:
   y_i=$B_i \oplus ID_i \oplus$ h(b $\oplus$ PW_i)
   A_i^*= h(h(b $\oplus$ PW_i) $\|$ ID_i) $\oplus$ h(y_i).
   Smart card verifies A_i^* with the stored A_i. If it holds, then computes
   h(d)=$N_i \oplus$ h(ID_i $\|$ h(b $\oplus$ PW_i))
   CID= h(ID_i $\|$ y_i) $\oplus$ h(h(d) $\|$ N_u $\|$ T_u)
   C_1= N_u^e mod n and
   C_2=h(ID_i $\|$ h(d) $\|$ y_i $\|$ T_u $\|$ N_u), where T_u is current timestamp.

3. U_i sends login message {CID, C_1, C_2, T_u} to S.

2.4.3 Authentication phase

In this phase S receives the login message {CID, C_1, C_2, T_u} and performs the following steps:

1. Verifies the validity of the timestamp by checking T_{ur} - $T_u \leq \Delta$T. If it does not holds, server rejects the login message.

2. Server decrypts N_u from C_1 using its private key d and computes:
   D_i* = h(CID $\oplus$ h(h(d) $\|$ N_u $\|$ T_u) $\oplus$ d) and finds D_i corresponding to D_i* in the database. If D_i is not found, then S extracts $y_i \oplus$ h(h(d) $\|$ d) and $ID_i \oplus$ h(d $\|$ y_i) corresponding to D_i* from database. S calculates y_i from $y_i \oplus$ h(h(d)$\|$d) and ID_i from $ID_i \oplus$ h(d $\|$ y_i).

3. S computes C_2'= h(ID_i $\|$ h(d) $\|$ y_i $\|$ T_u $\|$ N_u) and compares C_2' with received C_2. If both are equal then the server authenticates the user and begins the mutual authentication.

4. To perform mutual authentication, S computes
   SK= h(ID_i || h(d) || y_i || T_u || T_s || N_u) and
   C_3 = h(h(d) || ID_i || y_i || T_s || N_u || SK), where T_s is the current timestamp.

5. S sends {C_3, T_s} to U_i

6. U_i receives {C_3, T_s}, checks the validity of T_s and computes
   SK' = h(ID_i || h(d) || y_i || T_u || T_s || N_u) and
   C_3' = h(h(d) || ID_i || y_i || T_s || N_u || SK').
   U_i compares computed C_3' with the received C_3. If they are equal, user authenticates the server.

2.4.4 Password change phase

1. U_i inserts the smart card into the card reader and enters ID_i and PW_i, then he/she requests to change the password.

2. The smart card computes
   y_i= $B_i \oplus ID_i \oplus$ h(b $\oplus$ PW_i),
   A_i^*=h(h(b $\oplus$ PW_i) || ID_i) $\oplus$ h(y_i).
   Smart card verifies the validity of A_i^* by checking whether A_i^* is equal to the stored A_i. On the verification failure, the smart card rejects the request. If the number of failures exceeds a predefined value, smart card locks automatically.

3. The smart card computes:
   N_i^{new}=$N_i \oplus$ h(ID_i || h(b $\oplus$ PW_i)) $\oplus$ h(ID_i || h(b $\oplus$ PW_i^{new}))
   A_i^{new}=$A_i \oplus$ h(h(b $\oplus$ PW_i)|| ID_i) $\oplus$ h(h(b $\oplus$ PW_i^{new})|| ID_i) and
   B_i^{new} = $B_i \oplus$ h(b $\oplus$ PW_i) $\oplus$ h(b $\oplus$ PW_i^{new}), and then updates N_i, A_i and B_i with N_i^{new}, A_i^{new} and B_i^{new}, respectively.

2.4.5 Smart card revocation phase

In Ding et al.'s scheme, to revoke the lost smart card, U_i uses the previous ID_i and PW_i. Here server selects a new random number y_i^{new} corresponding to U_i and computes the new N_i, A_i, B_i and D_i. Remaining procedure of this phase is same as the registration phase.

2.5 Cryptanalysis of Ding et al.'s scheme

This section discusses the security problems of Ding et al.'s scheme, which has presented below.

2.5.1 Insecure server secret key

In Ding et al.'s authentication scheme, encryption and decryption operations are carried out with the help of server's secret key. Similarly, the secret key also helps to calculate the session key. Once the secret key has compromised, there is a possibility of leakage of sent and delivered messages. Assume that adversary $\mathscr{A}$ has registered to the server as an authorized user and has obtained a smart card containing the parameters $\{N_i, A_i, B_i, \text{n}, \text{e}, \text{h}(.)\}$. Using these values, an adversary can calculate the server's secret key, d as follows:

Consider the equation $\text{h(d)}=\text{h}(ID_i \parallel \text{h}(\text{b} \oplus PW_i)) \oplus N_i$, where ID_i, PW_i, b are chosen by the adversary and N_i can be extracted from the smart card. To obtain d from h(d), first $\mathscr{A}$ guesses a value d', computes h(d') and compares h(d) with h(d'). If h(d) = h(d'), it indicates that $\mathscr{A}$ has guessed the correct value of d. If they are not equal, $\mathscr{A}$ repeats the procedure until he/she guesses the correct value of d

2.5.2 Traceable user's identity

In Ding et al.'s scheme, an identity of the user does not travel in a public channel. A dynamic ID has computed in every session. But, if the smart card has lost, the adversary can trace the identity of the user if he/she knows the secret key of the server. Suppose smart card is lost/stolen, adversary has extracted the parameters $\{N_i, A_i, B_i, \text{n}, \text{e}, \text{h}(.)\}$ of smart card. Also, the adversary can obtain login message $\{\text{CID}, C_1, C_2, T_u\}$ and authentication message $\{C_3, T_s\}$ from public channel.

Now, the adversary calculates server secret key 'd' as explained in section 2.5.1. Further, he/she can calculate the user ID_i as follows:
Adversary computes $\text{h}(ID_i \parallel \text{h}(\text{b} \oplus PW_i)) = \text{h(d)} \oplus N_i$, where N_i is the parameter stored in the smart card. Further, he/she computes $\text{h}(y_i) = \text{h}(ID_i \parallel \text{h}(\text{b} \oplus PW_i)) \oplus A_i$, where A_i is the stored value in the smart card. Hence adversary can guess y_i from the knowledge of $\text{h}(y_i)$. Now he/she continues the computing as follows:

h($ID_i \parallel y_i$) = CID $\oplus$ h(h(d) $\parallel N_u \parallel T_u$)

Adversary calculates N_u by discrete logarithm method using equation $C_1 = N_u^e$ mod n.

In equation h($ID_i \parallel y_i$) = CID $\oplus$ h(h(d) $\parallel N_u \parallel T_u$), all values are known except ID_i, where CID and T_u will be available from login message {CID, C_1, C_2, T_u} and it is proved that adversary can calculate h(d), N_u and y_i. Now, adversary guesses a random identity ID_i*, calculates h(ID_i* $\parallel y_i$) and compares the result with h($ID_i \parallel y_i$). If both are equal, the adversary has guessed the correct ID_i, else he/she repeats the above procedure till he/she guesses the correct ID_i. Hence, it is proved that Ding et al.'s scheme does not provide user anonymity.

2.5.3 Vulnerable to offline password guessing attack

Consider the scenario that, the smart card is stolen and adversary has extracted the parameters {N_i, A_i, B_i, n, e, h(.)} stored in the smart card. Adversary can obtain the login message {CID, C_1, C_2, T_u} and authentication message {C_3, T_s} from public channel. If the adversary knows server secret key and user id, he/she can guess the password PW_i using the equation h(b $\oplus$ PW_i)=$B_i \oplus ID_i \oplus y_i$.

In section 2.5.1 and 2.5.2, It has explained how the adversary can obtain the secret key 'd' and ID_i. Consider the equation h(b $\oplus$ PW_i) = $B_i \oplus ID_i \oplus y_i$. Using known values B_i, b, ID_i and y_i, adversary can perform offline password guessing attack. Adversary guesses a random password PW_i*, calculates h(b $\oplus$ PW_i*) and compares with h(b $\oplus$ PW_i). If they are equal, adversary has guessed the correct password. If they are unequal $\mathscr{A}$ continues guessing the password till he guesses the correct PW_i. So it can be concluded that Ding et al.'s scheme is vulnerable to offline password guessing attack.

2.5.4 No perfect forward secrecy

Suppose the smart card is lost/stolen and adversary has extracted the smart card parameters {N_i, A_i, B_i, n, e, b, h(.)}. Adversary also can obtain login message {CID, C_1, C_2, T_u} and authentication message {C_3, T_s} from public channel. With these information, adversary can calculate the previous session key SK as follows.

In Ding et al.'s scheme, session key is SK= h($ID_i \parallel$ h(d) $\parallel y_i \parallel T_u \parallel T_s \parallel N_u$). The adversary obtain timestamp T_u and T_s from login and authentication messages and secret

key d, ID, y_i and N_u can be computed by the procedure explained in 2.5.1 and 2.5.2. Now, every parameters required to compute SK are available to the adversary and he/she can calculate previous session key by applying values to the equation SK= h(ID_i || h(d) || y_i || T_u || T_s || N_u). Once the session key is found, the entire session will become insecure.

2.5.5 Vulnerable to insider attack

In registration phase, U_i sends registration request message {ID_i, h(b $\oplus$ PW_i)} to the server S through secure channel. Even though the request message sent through the secure channel, an insider can obtain that message once it received by the server. From this message, an insider can get the user identity. Suppose insider has stolen the smart card and extracts the information {N_i, A_i, B_i, n, e, b, h(.)}, then he/she can obtain the random number b.

Now insider has the value h(b $\oplus$ PW_i) and random number b. Insider guesses the password PW_i*, calculates h(b $\oplus$ PW_i*) and compares with h(b $\oplus$ PW_i). If they are equal, an insider has guessed the correct password. If not he/she continues the process of guessing password till he guesses the correct PW_i. Hence Ding et al.'s scheme is vulnerable to insider attack

2.5.6 Vulnerable to stolen smart card attack

In Ding et al.'s scheme, an intruder can enter into the server by crafting the login message, once the smart card has stolen. Assume that smart card has stolen by the adversary and he/she has extracted the parameter {N_i, A_i, B_i, n, e, h(.)} stored in the smart card. The adversary obtain login message {CID, C_1, C_2, T_u} and authentication message {C_3, T_s} from public channel.

As illustrated in the previous sections (section 2.5.1 and 2.5.2), adversary can calculate server secret key, user ID and random number y_i. In equation C_1= N_u^e mod n, we can calculate N_u using discrete logarithm. With these informations adversary craft the message as follows:
Using timestamps T_s from authentication message, adversary computes CID* = h(ID_i || y_i) $\oplus$ h(h(d) || N_u || T_s). Further he/she computes C_2*=h(ID_i || h(d) || y_i || T_s || N_u) and

crafts the login message {CID, C_1, C_2, T_u} to {CID*, C_1, C_2*, T_s}. This message will be sent to the server S.

On the other hand, the server S receives {CID*, C_1, C_2*, T_s} and checks the login message validity with the help of the time stamp T_{sr} i.e. T_{sr} - $T_s \leq \Delta$T holds or not. This will hold and random number N_u will be decrypted from C_1 by the server S using its private key d. Further it calculates D_i* = h(CID* $\oplus$ h(h(d) $\|$ N_u $\|$ T_s) $\oplus$ d). Then server S calculates y_i from $y_i \oplus$ h(h(d)$\|$d) and ID_i from $ID_i \oplus$ h(d $\|$ y_i) and computes C_2'= h(ID_i $\|$ h(d) $\|$ y_i $\|$ T_s $\|$ N_u).

C_2' will be compared with C_2* and C_2'= C_2* will definitely hold. Hence the server grants access to the adversary. In this way, an adversary can login to the server by forging the login message if the smart card has stolen.

2.5.7 Weak to revoke lost smart card

In Ding et al.'s scheme, to revoke the lost smart card, the client uses the previous PW_i and the ID_i. But it is already proved that (section 2.5.3) Ding et al's scheme is vulnerable to password guessing attack. Assume that adversary steals the smart card, guesses the PW_i correctly and enters the server. He/she can change the password. Once the password changed, the legal user cannot revoke the lost smart card using the previous password. Hence, Ding et al.'s scheme is weak to revoke the lost smart card.

2.6 The Proposed Scheme

There is a need to develop an enhanced scheme to solve the security issues identified in reviewed schemes. The Proposed scheme contains four phases. (1) Registration phase, (2)login and authentication phase, (3) password change phase and (4) smart card revocation phase. The four phases of the proposed scheme are illustrated below.

2.6.1 Registration phase

If U_i wants to register for first time, he/she follow the given procedure:

1. User selects ID_i, PW_i and a random number b.

2. Computes SP = h($ID_i \parallel PW_i \parallel$ b),
   $A_i = ID_i \oplus$ b and sends registration request {SP, A_i} to server S via secure channel.

3. After collecting {SP, A_i} from user, S generates random number r_i and computes
   m_i = h($r_i \parallel$ x) $\oplus A_i$,
   M_i = h(SP) $\oplus$ h($r_i \parallel$ x),
   N_i = h($A_i \parallel$ SP) and
   V_i = h(SP $\parallel A_i$) $\oplus M_i$.

4. S stores r_i and m_i into the database and deposit {h(.), V_i, N_i} into the smart card SC and remits {SC, M_i} to U_i through secure channel.

5. On receiving {SC, M_i} from S, user computes
   $D_i = A_i \oplus M_i$
   and Z_i = h($ID_i \parallel PW_i$) $\oplus$ b.
   U_i stores D_i, Z_i into the smart card. Finally the smart card contains the values {h(.), V_i, N_i, D_i, Z_i }.

2.6.2 Login and authentication phase

In order to login into the system, first U_i inserts the smart card into card reader and inputs ID_i with PW_i. The card executes the following steps:

1. Computes b* = h($ID_i \parallel PW_i$) $\oplus Z_i$
   SP* = h($ID_i \parallel PW_i \parallel$ b)
   A_i* = $ID_i \oplus$ b*
   M_i* = $A_i \oplus D_i$ and V_i* = h(SP* $\parallel A_i$*) $\oplus M_i$*

2. SC verifies, if the computed V_i* = V_i or not. This comparison gives the correctness of ID_i and PW_i. If both V_i* and V_i are not equal, the smart card drops the session, else U_i computes
   $Q_i = M_i \oplus$ h(SP)
   C_1 = h($N_i \parallel Q_i \parallel A_i$),
   CID = h($C_1 \parallel$ T $\parallel A_i$) $\oplus$ b
   C_2 = h(CID $\parallel C_1 \parallel$ b $\parallel N_i$).

3. SC sends the login request {CID, N_i, C_2, T} to the server S via public channel.

The server receives the login request {CID, N_i, C_2, T} and authenticates the user as follows:

1. Server first checks the freshness of login request. Examination of time stamp (T) validity provides the login request freshness result. Server verifies if T'- T $\leq \Delta$T and no other login message is with same parameter {CID, N_i, C_2, T} within the time period (T'+ ΔT) and (T'- ΔT). If these condition satisfies then, S proceeds to the further calculation, else the S rejects the login request.

2. Further, S computes Q_i'=h(r_i || x)
   A_i'= $m_i \oplus Q_i$',
   C_1 = h(N_i || Q_i || A_i)
   b'= CID $\oplus$ h(C_1'|| T || A_i')
   and C_2'= h(CID || C_1'|| b || A_i').

3. S compares C_2' with the received C_2. If $C_2' \neq C_2$, then server drops the session, else the server computes,
   SK = h(Q_i'|| b'|| N_i || C_1').
   C_3 = h(r_i || x) $\oplus$ h(SK || C_1'|| T') and transmits mutual authentication message {C_3, T'} to U_i via public channel.

4. User receives the mutual authentication message and obtains current time T˝and verifies the validity of the T'. User checks if T˝- T'$\leq$ ΔT. If this condition is not satisfied, then U_i rejects the mutual authentication message. Otherwise U_i computes
   SK'= h(Q_i || b || N_i || C_1),
   C_3'= $M_i \oplus$ h(SP) $\oplus$ h(SK || C_1 || T') and verifies C_3'with received C_3. If they are equal then mutual authentication completes successfully.

5. Once the connection is successfully established, further communication will be done using independently calculated common sessions key
   SK = h(Q_i'|| b'|| C_1' || N_i) and
   SK'= h(Q_i || b || C_1 || N_i).

2.6.3 Password change phase

In this phase, user U_i can change the password from PW_i to PW_i*. This phase is applied as follows:

1. U_i inserts the smart card and inputs the ID_i and PW_i. The operations performed are as follows:
   U_i computes b* = h($ID_i \parallel PW_i$) $\oplus$ Z_i,
   SP* = h($ID_i \parallel PW_i \parallel$ b),
   A_i* = $ID_i \oplus$ b*
   M_i* = $A_i \oplus D_i$ and V_i* = h(SP* $\parallel A_i$*) $\oplus M_i$*

2. SC verifies, if the computed V_i* = V_i or not. If they are not equal then smart card drops the session, else it asks for a new password PW_i*.

3. Once the user inputs new password PW_i*, smart card SC computes
   SP^{new} = h($ID_i \parallel PW_i$* $\parallel$ b),
   M_i^{new} =$M_i \oplus$ h(SP) $\oplus$ h(SP^{new})
   N_i^{new} = $N_i \oplus$ h($A_i \parallel$ SP) $\oplus$ h($A_i \parallel SP^{new}$),
   V_i^{new} = h($SP^{new} \parallel A_i$) $\oplus M_i^{new}$,
   Z_i^{new} = h($ID_i \parallel PW_i$*) $\oplus$ b and
   D_i^{new} = $A_i \oplus M_i^{new}$.

4. Smart card SC stores N_i^{new}, V_i^{new}, Z_i^{new} and D_i^{new} in place of N_i, V_i, Z_i, D_i.

2.6.4 Smart card revocation phase

In the proposed scheme, to revoke the lost smart card, no need to use previous password. The procedure for smart card revocation will be illustrated below.

1. When a user U_i inputs previous ID_i and a new password PW_i^{new}, system calculates h($ID_i \parallel PW_i^{new}$) and sends {$ID_i$, h($ID_i \parallel PW_i^{new}$)} to the server.

2. Server calculates A_i = h($r_i \parallel$ x) $\oplus$ m_i, and sends A_i to the user.

3. User system extracts random number by b=$A_i \oplus ID_i$, calculates SP^{new} = h($ID_i \parallel PW_i^{new} \parallel$ b) and sends SP^{new} to the server.

4. After collecting SP^{new} from user, S generates new random number r_i* and computes,
   m_i* = h(r_i* $\|$ x) $\oplus$ A_i,
   $M_i^{new} = SP^{new} \oplus$ h(r_i* $\|$ x),
   N_i^{new} = h($A_i \| SP^{new}$)
   and V_i = h($SP^{new} \| A_i$) $\oplus$ M_i^{new}.

5. S replaces r_i* and m_i* with r_i and m_i respectively in the database and deposit {h(.), N_i^{new}, V_i^{new}, A_i} into the smart card SC and delivers {SC^{new}, M_i^{new}} to U_i via secure channel.

6. On receiving {SC^{new}, M_i^{new}} from S, user computes
   $D_i^{new} = A_i \oplus M_i^{new}$
   and Z_i^{new} = h($ID_i \| PW_i$) $\oplus$ b.
   U_i stores Z_i^{new} and D_i^{new} into the smart card. Finally smart card contains the values
   SC^{new} = {h(.), N_i^{new}, V_i^{new}, Z_i^{new}, D_i^{new} }.

2.7 Cryptanalysis of the Proposed Scheme

In this section, cryptanalysis of the proposed scheme has presented. We have proved that the security of the proposed scheme overcomes all the identified attacks. The security analysis of the proposed scheme has explained below.

2.7.1 Secured server secret key

As discussed before, server secret key involves in encryption, decryption and session keys calculation. Therefore, it is important to provide security to the server secret key. Hence, it is better to develop a scheme, which protects the secret key from both legal user and adversary. In the proposed scheme, for any operation, server secret key will not be used in plain text format. S generates random number r_i and computes m_i = h(r_i $\|$ x). This m_i has used for further operations. Hence, it is not possible to guess the server secret key by any user.

2.7.2 Untraceable User's identity

In the proposed scheme, an attacker cannot trace the identity of a user either from the smart card or an intercepted login and authentication message. In the proposed scheme, U_i will not send ID_i to the server in plain text format even in the secure channel. In the registration phase itself, the user's ID has well protected.

When a U_i wants to register, he/she selects ID_i, PW_i and a random number b. In client side, system first computes SP = h($ID_i \parallel PW_i \parallel$ b) and $A_i = ID_i \oplus$ b, then it sends {SP, A_i} to the server via secure channel. The advantage of A_i calculation is to secure the ID_i. In the proposed scheme, ID_i is involved in many operations like $Q_i = M_i \oplus$ h(SP), C_1 = h($N_i \parallel Q_i \parallel A_i$), CID = h($C_1 \parallel$ T $\parallel A_i$) $\oplus$ b, but ID_i is not used independently. Therefore, the attacker cannot differentiate the user ID.

2.7.3 Resists offline password guessing attack

In the proposed scheme, it is difficult to guess the password even if an adversary obtains the smart card parameters, login and authentication messages. Assume that the adversary has stolen the smart card and extracted the parameters {h(.), V_i, N_i, D_i, Z_i}. Also he/she has obtained login message {CID, N_i, C_2, T} and authentication messages {C_3, T'}. But using this information adversary cannot calculate user ID_i. To calculate PW_i, an adversary must know SP. Even though he/she calculates SP, it is not possible to guess PW_i without knowledge of ID_i and random number b. Hence the proposed scheme protects the user from offline password guessing attack.

2.7.4 Provides perfect forward secrecy

In the proposed scheme, user and the system calculates the session key SK = h($Q_i'\|$ b'$\|$ $N_i \parallel C_1'$). It is already proved that the scheme secures the server secret key. Even though the server secret key x is compromised, our scheme will remain secure because, to calculate the session key it requires not only server secret key x but also need to calculate $Q_i = M_i \oplus$ h(SP) where M_i = h(SP) $\oplus$ h($r_i \parallel$ x).

Assume that adversary has got smart card and extracts the saved parameter {h(.), V_i, N_i, D_i, Z_i}, he/she also obtained login message {CID, N_i, C_2, T} and authentication message {C_3, T'} from the public channel. Even though the server secret key has com-

promised, it is difficult to find the values of Q_i, SP, random number b and r. Hence, it can be concluded that the information of secret key, user's smart card parameters, login and authentication message will not help an attacker to calculate the session key.

2.7.5 Security against insider attack

In the proposed scheme, password will not be submitted in the form of plain text. User creates one secure password value by using random number b. i.e. SP = h($ID_i \parallel PW_i \parallel$ b), $A_i = ID_i \oplus$ b and submits {SP, A_i} to the server. Using SP, adversary can not guess PW_i or ID_i without knowledge of all three parameters (PW_i, ID_i and b). Hence the proposed scheme provides complete security against insider attack.

2.7.6 Security against stolen smart card attack

In the proposed scheme, it is not possible to enter into the system without using the ID_i and PW_i. Assume that, smart card is stolen and parameters {h(.), V_i, N_i, D_i, Z_i} are extracted. Also the login message {CID, N_i, C_2, T} and the authentication message {C_3, T'} are obtained. Even then, the adversary cannot enter into the system because the proposed scheme provides protection against password guessing attack and protects user anonymity.

In the proposed scheme, adversary cannot able to calculate CID_i* by changing the time stamp from T_u to T_s. Before calculation of CID_i*, intruder must know the remaining values which are required to calculate CID_i, i.e. CID_i = h($C_1 \parallel$ T $\parallel A_i$) $\oplus$ b. Without knowledge of the ID_i and PW_i, it is not possible to extract the value of b, where b = Z_i $\oplus$ h($ID_i \parallel PW_i$). Hence, adversary fails to craft the login message. Therefore, without knowledge of ID_i and PW_i, adversary will not be able to login to the system. So the proposed scheme provides security against smart card stolen attack.

2.7.7 Provides proper smart card revocation

We have proved that to revoke the lost smart card, usage of previous ID_i and PW_i is not a secure method. This type of user revocation method might affect the user re-registration if adversary guesses the password. In the proposed scheme, a solution in which user can revoke his/her card with the new password has been given. So that, even though

adversary changes the password, their won't be any issues. Also, it is not necessary to remember the old password even after the smart card has lost.

2.7.8 Wrong password entry quickly detected

In the proposed scheme, PW_i will be verified in the user side before sending the login request message. When U_i wants to login to the server, he/she inputs the ID_i and PW_i. Further, smart card computes SP* = h($ID_i \parallel PW_i \parallel$ b)

$M_i = Z_i \oplus$ SP,

$A_i = D_i \oplus$ SP

V_i* = h(SP* $\parallel A_i) \oplus M_i$. Further the smart card verifies the computed V_i* with stored V_i. If the input ID_i and PW_i are correct then obviously V_i* = V_i. If ID_i or PW_i is wrong then both V_i* and V_i will not be equal and immediately the smart card drops the session. Thus the proposed scheme quickly detects the wrong PW_i entry.

2.8 Computational efficiency

To show the efficiency of an authentication scheme, it is necessary to consider the performance. Performance of a scheme can be measured by computation cost. This section compares the computation cost of the proposed scheme with Wen and Li scheme and Ding et al.'s scheme. Table 2.2 demonstrates the computational cost of the proposed scheme. Here, T_h represents the execution time of the one-way hash function, $T_\oplus$ symbolizes the execution time of the XOR operation, $T_\parallel$ denotes the execution time of the concatenation operation and T_{me} signifies the execution time of modular operation.

Table 2.2 Comparison of Computational efficiency.

Phase	**Wen and Li scheme**	**Ding et al.'s scheme**	**Proposed scheme**
Registration Phase(U_i)	-	$1T_h + 2T_\oplus$	$2T_h + 3T_\oplus + 3T_\parallel$
Registration Phase(S)	$5T_h + 4T_\oplus + 1T_\parallel$	$8T_h + 10T_\oplus + 5T_\parallel$	$3T_h + 3T_\oplus + 2T_\parallel$
Login and authentication Phase(U_i)	$11T_h + 9T_\oplus + 7T_\parallel$	$9T_h + 8T_\oplus + 19T_\parallel + 1T_{me}$	$9T_h + 8T_\oplus + 18T_\parallel$
Login and authentication Phase(S)	$10T_h + 9T_\oplus + 7T_\parallel$	$7T_h + 4T_\oplus + 18T_\parallel + 2T_{me}$	$6T_h + 3T_\oplus + 13T_\parallel$

High hash overhead remote user authentication schemes decreases the practical suitability. Compare to Wen and Li scheme and Ding et al.'s scheme, the proposed scheme uses less number of hash functions. Registration phase of Wen and Li scheme takes $5T_h$ + $4T_{\oplus}$ + $1T_{||}$ and Ding et al.'s scheme takes $9T_h$ + $12T_{\oplus}$ + $5T_{||}$ to execute the scheme. Similarly login and authentication phase of Wen and Li scheme and Ding et al.'s scheme takes $21T_h + 18T_{\oplus} + 14T_{||}$ and $16T_h + 12T_{\oplus} + 37T_{||} + 3T_{me}$ respectively. But the proposed scheme computational cost is $5T_h$ + $6T_{\oplus}$ + $5T_{||}$ in registration phase and $15T_h$ + $11T_{\oplus}$ + $31T_{||}$ in login and authentication phase. Compared to Wen and Li's scheme,the proposed scheme reduces 6 hash operations and reduces 5 hash operations when compared to the Ding et al.'s scheme. These reductions makes the scheme more lightweight.

Table 2.3 Comparison of security attacks and characteristics

Attacks and characteristics	**Wen and Li scheme**	**Ding et al.'s scheme**	**Proposed scheme**
Resistance to denial of Service attack	Y	Y	Y
Resistance to password guessing attack	Y	N	Y
Resistance to parallel session attack	Y	Y	Y
Resistance to server masquerading attack	Y	N	Y
Resistance to insider attack	N	N	Y
Resistance to stolen smart card attack	N	N	Y
Provides quick wrong password verification	N	Y	Y
Provides security for server secret key	Y	N	Y
Provides user anonymity	Y	N	Y
Provides smart card revocation	N	N	Y
Provides perfect forward secrecy	N	N	Y
Provides mutual authentication	Y	Y	Y

Table 2.3 presents the comparison result of various attacks and characteristics of both Wen and Li and Ding et al.'s schemes with the proposed scheme. Here the security attacks include denial of service, password guessing, parallel session, server masquerading, insider attack and stolen smart card attack. Also, it presents some characteristics like verification mechanism in SC_i, security for server secret key, user anonymity, user revocation and mutual authentication. In Table 2.3, Y denotes the scheme provides

the security for a particular attack and N represents the scheme does not provides any security for the respective attack and characteristics. The proposed scheme resists all attacks and satisfies all features.

2.9 Conclusion

In this study, the cryptanalysis of Wen and Li's and Ding et al.'s dynamic ID-based remote user authentication schemes has been presented and its vulnerabilities are identified. To overcome the identified security problems, a new scheme has been proposed. Through the computation cost comparison, it has been observed that the proposed schemes takes less number of hash functions and resists all the security attacks. Hence, it can be concluded that the proposed scheme is more robust and easy to implement practically.

Chapter 3

A SECURE ELLIPTIC CURVE CRYPTOGRAPHY BASED DYNAMIC AUTHENTICATION SCHEME USING SMART CARD

3.1 Introduction

The construction of the secure remote user authentication scheme relies on robust encryption techniques and well built security proofs. Wang et al. (2015) said that authentication schemes, which does not adopt public key cryptography are unable to achieve user privacy and session security. Therefore, it is good to employ public key cryptography while designing the authentication schemes. Many schemes have adopted the public key cryptography. But, they are vulnerable to many attacks.

To solve these problems, Truong et al. (2014) remote user authentication scheme has been reviewed. This scheme designed using elliptic curve cryptography. By cryptanalysis of Troung et al.'s scheme, it is found that the scheme is vulnerable to replay attack, the scheme does not provide user anonymity and perfect forward secrecy. It is also observed that the scheme is failed to provide security to the server secret key and also will not allow the user to choose his/her own password. To overcome these weaknesses, a dynamic authentication scheme using elliptic curve cryptography has been proposed. To prove the security of the proposed scheme by the formal method, BAN logic method is employed and simulated the proposed scheme using AVISPA tool. The simulation demonstrates that the proposed scheme is safe. Furthermore, the computation efficiency

of the proposed scheme is also discussed and compared with other related schemes.

Rest of this chapter is organized as follows. Section 3.2 reviews the Troung et al.'s remote user authentication. Section 3.3 presents the cryptanalysis of Troung et al.'s scheme. Section 3.4 proposes a new scheme, which is an improvement of Troung et al.'s scheme. Section 3.5 discusses the cryptanalysis of the proposed scheme. Formal analysis of the proposed scheme using BAN logic is illustrated in section 3.6. Security verification of the proposed scheme using AVISPA tool is presented in section 3.7. Section 3.8 compares the performance of the proposed scheme with other schemes. Section 3.9 depicts the concluding remarks of this chapter.

3.2 Review of Truong et al. (2014) scheme

In this section presents the review of Troung et al.'s scheme. The scheme mainly contains 3 phases, (1) Registration phase, (2) Login phase, authentication phase, (3) Password change phase. The notations used in this chapter are shown in the Table 3.1.

Table 3.1 Notations and Descriptions.

Notations	**Descriptions**
S	Server
U_i	i^{th} User
ID_i	Identity of U_i
PW_i	i^{th} user password
x	Secret key of S
$\oplus$, $\|\|$	Bitwise XOR and concatenation operator
T, T'	Timestamps acquired at the user side and server side
E_p	Elliptic curve
F_p	Finite field
P	Base point of E_p
h(.)	One-way hash function
$\mathscr{A}$	Adversary

3.2.1 Registration phase

Before registration, S selects an elliptic curve E_p over the finite field F_p with a large prime number 'p' and a one way hash function h(.). Server also choose a base point 'P'

of order 'n' and selects 'x' as private key and publish the parameters $\{E_p, P, F_p, h(.)\}$.

1. In the registration phase, first U_i selects ID_i and sends it to the server.

2. Server selects PW_i, random number 'e' and random value N. This N will be zero at first registration.

3. S computes $RPW_i = h(PW_i \| N)$
   $R_i = h(x \| e) \oplus h(RPW_i \| ID_i)$
   and $V_i = h(h(x \| e) \| ID_i \| RPW_i)$.

4. S stores $\{h(.), V_i, R_i, e, N\}$ into the smart card and sends to U_i with PW_i through the secure channel.

5. U_i receives the smart card and changes PW_i.

3.2.2 Login phase

To login into the server S, U_i inserts the card, inputs ID_i and PW_i. Further, the smart card performs the following operations:

1. Computes $RPW_i = h(PW_i \| N)$
   $h(x \| e) = R_i \oplus h(RPW_i \| ID_i)$
   $V_i = h(h(x \| e) \| ID_i \| RPW_i)$

2. SC verifies, $V_i^* = V_i$ or not. If equal then system generates random number $r_u \in [2, n-1]$ and computes
   $CID_i = h(r_u{*}P \| h(x \| e)) \oplus ID_i$ and
   $MAC_u = h(ID_i \| h(x \| e) \| r_u{*}P)$.

3. SC transmits the login request $\{CID_i, r_u{*}P, MAC_u, e\}$ to S via public channel.

3.2.3 Mutual authentication phase

In mutual authentication phase S receives the login request message $\{CID_i, r_u{*}P, MAC_u, e\}$ and authenticates the user as follows:

1. S reconstructs $h(x \| e)$, using the information e.

2. Extracts ID_i = h(r_u*P || h(x || e)) $\oplus$ CID_i and verifies the validity of ID_i.

3. S checks the condition, MAC_u'= h(ID_i || h(x || e) || r_u*P) with the received MAC_u.

4. Further S generates random number $r_s \in$ [2, n-1] and computes,
   SK = h(r_u*r_s*P || h(x || e) || ID_i) and MAC_s = h(SK || h(x || e) || ID_i).
   S transmits {MAC_s, r_s*P } to U_i for mutual authentication.

5. User receives the message {MAC_s, r_s*P }, then calculates
   SK'= h(r_u*r_s*P || h(x || e) || ID_i),
   MAC_s'= h(SK' || h(x || e) || ID_i). Verifies MAC_s'with received MAC_s.
   If MAC_s'= MAC_s, further it computes R_{US}= h(SK') and sends R_{US} to the server.

6. Server receives R_{US}, computes R_{US}'= h(SK) and compares R_{US}' with received R_{US}. If both R_{US} and R_{US}' are equal then S permits the U_i to access the service. Otherwise, S terminates the session.

3.2.4 Offline password change phase

In this phase U_i can change the password from PW_i to PW_i^{new} as follows:

1. U_i inserts the card, inputs ID_i and PW_i. Smart card computes
   RPW_i = h(PW_i || N), h(x || e) = R_i $\oplus$ h(RPW_i || ID_i),
   V_i = h(h(x || e)|| ID_i || RPW_i).

2. Smart card verifies, V_i* = V_i or not. If it is, then asks for a new password PW_i^{new}.

3. Once the user enters new password PW_i^{new}, smart card checks for N. If N is zero then it selects select new value for N, i.e. N^{new}. Further, smart card computes the following:
   RPW^{new} = h(PW_i^{new} || N^{new}),
   R_i^{new} = R_i $\oplus$ h(ID_i || RPW) $\oplus$ h(ID_i || RPW^{new}),
   V_i^{new} = h(h(x || e) || RPW^{new} || ID_i)

4. Smart card stores R_i^{new}, V_i^{new}, N^{new} in place of R_i, V_i, N.

3.3 Cryptanalysis of Troung et al.'s scheme

In cryptanalysis of the Troung et al.'s scheme, it is identified that the scheme is vulnerable to replay attack, does not provide user anonymity and perfect forward secrecy. It is also observed that Troung et al.'s scheme does not provide security to the server secret key and does not allow the user to choose his/her own password. This section discusses the security weaknesses of Troung et al.'s scheme.

3.3.1 Password selection is done by the server

During the registration phase of Troung et al.'s scheme, the password is selected by the server S. User has no choice of his/her own password selection. In Troung et al.'s scheme, if the user wants to choose his/her own password, then U_i must perform the password change phase after registration. Suppose the user forgot to change the password after registration, then the user should remember the server generated password to access the server next time. But, server-generated passwords are long, random and difficult to remember. Hence, the selection of password must be done by the user. Unfortunately, Troung et al.'s scheme does not allow the user to choose his/her own password.

3.3.2 Insecure server secret key

In Troung et al.'s authentication scheme encryption and decryption operations are done with the help of server's secret key. Once the secret key is compromised, there is a possibility of leakage of passwords. An adversary can get the secret key as follows.

Assume that an adversary $\mathscr{A}$ registers to the server as a legal user and obtains a smart card, containing the parameters $\{R_i, V_i, \text{N}, \text{e}, \text{h}(.)\}$. Using these smart card parameter, $\mathscr{A}$ first computes $\text{h}(\text{x} \parallel \text{e}) = \text{h}(RPW_i \parallel ID_i) \oplus R_i$. Here, $\mathscr{A}$ can obtain R_i and e from the smart card. Further and he/she compute $RPW_i = \text{h}(PW_i \parallel \text{N})$ where N is theparameter obtained from the smart card.

To calculate the server's secret key from $\text{h}(\text{x} \parallel \text{e})$, adversary guesses the random value x', computes $\text{h}(\text{x}' \parallel \text{e})$ and compares with obtained $\text{h}(\text{x} \parallel \text{e})$. If $\text{h}(\text{x}' \parallel \text{e}) = \text{h}(\text{x} \parallel \text{e})$ then adversary guessed the server secret key. If not, $\mathscr{A}$ repeats the above procedure,

until he/she get the correct value x. Hence, Troung et al.'s scheme do not provide any security to the server secret key.

3.3.3 Traceable user's identity

Assume that, an adversary $\mathscr{A}$ has intercepted the login message $\{CID_i, MAC_u, r_u*P, e\}$. Now, $\mathscr{A}$ can trace the identity of the user, if he/she has the knowledge of server secret key 'x'. In section 3.3.2 it has been proved that Troung et al.'s scheme, does not provide security to the server's secret key. Using the procedure explained in the section 3.3.2, $\mathscr{A}$ computes the server secret key 'x'.

Now, consider the equation $ID_i = h(r_u*P \parallel h(x \parallel e)) \oplus CID_i$. Server uses this equation to extract ID_i in mutual authentication phase. Here, $\mathscr{A}$ can obtain the values CID_i, e and r_u*P from intercepted login message $\{CID_i, MAC_u, r_u*P, e\}$. Once $\mathscr{A}$ gets the server secret key 'x', he/she computes $h(x \parallel e)$ and $ID_i = h(r_u*P \parallel h(x \parallel e)) \oplus CID_i$ to get ID_i. Hence, Troung et al.'s scheme does not provide user untraceability.

3.3.4 Replay attack

In Troung et al.'s scheme, when login message $\{CID_i, MAC_u, r_u*P, e\}$ sent from U_i to server S, server starts further computation without verifying freshness of the received message.

If $\mathscr{A}$ intercepts the previous successfully authenticated login request message $\{CID_i, MAC_u, r_u*P, e\}$ and sent back the same message to the server. Then, the server will not verify the freshness of the received message. Server directly calculates $h(x\|e)$, $ID_i = h(r_u*P \parallel h(x \parallel e)) \oplus CID_i$ and $MAC_u' = h(ID_i \parallel h(x \parallel e) \parallel r_u*P)$. Server compares MAC_u' with MAC_u. Obviously, both MAC_u' and MAC_u are equal because the message which $\mathscr{A}$ sent to the server is previously successful authenticated message. The server gives access to $\mathscr{A}$. Hence, Troung et al.'s scheme does not provide security against replay attack.

3.3.5 No perfect forward secrecy

During mutual authentication phase of Troung et al.'s scheme, U_i sends authentication message R_{us} to the server through the public channel, where $R_{us} = h(SK)$. if adversary

intercepts the authentication message, then he/she can guess the value of SK directly as follows:

First, adversary guesses the session key SK', calculates R_{us}' = h(SK') and compares R_{us} = R_{us}'. If both R_{us} and R_{us}' are equal, then $\mathscr{A}$ has guessed the correct value of SK, else adversary repeats the above process until he/she gets the correct value. Correct identification of session key makes entire session insecure. Hence, Troung et al.'s scheme does not provide perfect forward secrecy.

3.4 The proposed scheme

The proposed scheme contains registration phase, login and authentication phase and password change phase. These phases are illustrated below.

3.4.1 Registration phase

Before registration, system initializes some values. S selects an elliptic curve E_p over the finite field F_p with a large prime number 'p' and a one way hash function h(.). Server also choose a base point 'P' of order 'n' and selects 'x' as private key and publish the parameters $\{E_p, P, F_p, h(.)\}$.

Registration phase of the proposed scheme is shown in Figure 3.1. When the user U_i wants to register for the first time, he/she follow the given procedure:

1. U_i selects ID_i, PW_i and a random number b.

2. After selection of the ID_i and PW_i, client system computes
   RPW = h($ID_i \parallel PW_i \parallel$ b),
   $A_i = ID_i \oplus$ b and sends {RPW, A_i} to the server S through secure channel.

3. Upon receiving {RPW, A_i} from user, server (S) generates random number 'e', 'n_0' and computes the following:
   B = e.P, m_i = h(e $\parallel$ x)
   $R_i = m_i \oplus$ h($A_i \parallel$ RPW)
   V_i = h($R_i \parallel A_i$) mod n_0.

4. After computation, S stores 'e', 'P' into the database and {h(.), V_i, R_i, B, P, n_0} into the smart card. The stored parameters {h(.), V_i, R_i, B, P, n_0} will be delivered to the U_i through secure channel.

5. U_i receives {h(.), V_i, R_i, B, P, n_0}, computes Z_i = h(ID $\|$ PW_i) $\oplus$ b and store Z_i into the smart card. At last parameters stored in the smart card are {h(.), V_i, R_i, Z_i, B, P, n_0}.

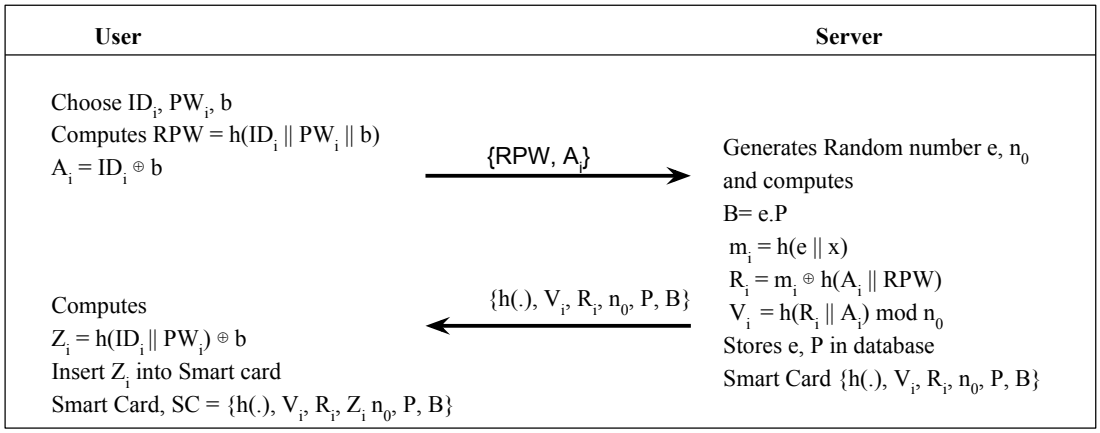

Figure 3.1 Registration phase of the proposed scheme

3.4.2 Login and authentication phase

Login and authentication phase of the proposed scheme is shown in Figure 3.2. In the login phase, U_i inserts the smart card, inputs ID_i and PW_i. The card performs the following procedure:

1. Computes b* = h(ID_i $\|$ PW_i) $\oplus$ Z_i

   A_i* = ID_i $\oplus$ b* and

   V_i* = h(R_i $\|$ A_i*) mod n_0

2. SC verifies the condition V_i* = V_i. This condition fails only if entered ID_i or PW_i is wrong. If this condition is false, then immediately session will be dropped, else computes RPW* = h(ID_i $\|$ PW_i $\|$ b)

   m_i* = R_i $\oplus$ h(RPW* $\|$ A_i*).

User generates random number $w \in [2, n-1]$ and computes

$C_u = w.P$, $K_u = w.B$

$CID = h(K_u \| T \| m_i) \oplus b$ and

$MAC_u = h(CID \| m_i \| K_u \| b)$.

3. Smart card sends the request message $\{CID, C_u, MAC_u, T\}$ for login to the server S through public channel.

S receives the login request message $\{CID, C_u, MAC_u, T\}$ from the user, and authenticates as follows:

1. Server takes the present time T' and verifies the validity of the received message time T. First server verifies the condition $T' - T \leq \Delta T$ and also confirms that there is no other request with same parameter within the period of $(T' + \Delta T)$ and $(T' - \Delta T)$, then S performs further calculation, else it rejects the request message $\{CID, C_u, MAC_u, T\}$ and drops the session.

2. S Computes $m_i = h(e \| x)$

   $K_u' = C_u.e$

   $b' = CID \oplus h(K_u' \| T \| m_i')$

   and $MAC_u' = h(CID \| m_i \| K_u' \| b)$.

3. S verifies the condition $MAC_u = MAC_u'$. If this condition is true, server proceeds to next step, else drops the session.

4. S generates random number $y \in [2, n-1]$ and computes $C_s = y.P$, $K_s = y.C_u$

   $SK = h(K_s \| b' \| m_i)$ and

   $MAC_s = h(SK \| b' \| T' \| m_i)$.

   S transmits $\{MAC_s, C_s, T'\}$ to U_i for mutual authentication through the public channel.

5. User receives $\{MAC_s, C_s, T'\}$ and checks the validity of the time stamp T'. If $T'' - T' \geq \Delta T$, then smart card rejects the message and drops the session. Otherwise U_i Computes

   $K_s' = w.C_s$

SK'= h(K_s'|| b || m_i),

MAC_s'= h(SK'|| b'|| T'). Verifies MAC_s'with received MAC_s. If MAC_s'= MAC_s further it computes R_{US}= h(SK || b) and sends R_{US} to the server.

6. Server receives R_{US}, computes R_{US}'= h(SK'|| b) and compares R_{US} and R_{US}'. If it is equal then the server is assured that the authentication done with the legal user.

7. After successful authentication, further communication will be continued through the common sessions keys. The session key of user, which is shared with the server is SK'= h(K_s'|| b || m_i) and the session key of the server, which is shared with the user is SK = h(K_s || b'|| m_i).

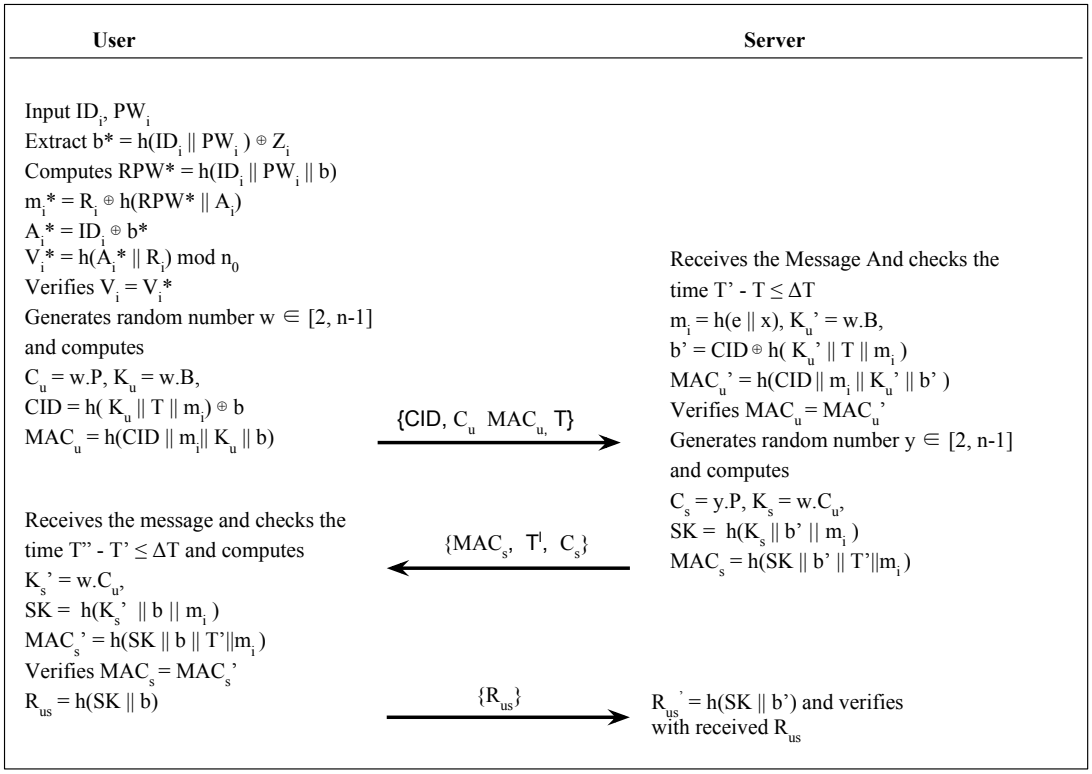

Figure 3.2 Login and authentication phase of the proposed scheme

3.4.3 Offline password change phase

In this phase U_i can change the password PW_i to PW_i*. Procedure to change the password is as follows:

1. U_i inserts the smart card, inputs ID_i and PW_i. Smart card computes

   b* = h($ID_i \parallel PW_i$) $\oplus$ Z_i

   RPW* = h($ID_i \parallel PW_i \parallel$ b)

   m_i* = $R_i \oplus$ h(RPW* $\parallel A_i$*) where

   A_i* = $ID_i \oplus$ b* and

   V_i* = h($R_i \parallel A_i$*) mod n_0

2. Smart card verifies the condition V_i* = V_i or not. If it is equal then the entered password is correct and it asks for a new password PW_i* else smart card drops the session.

3. Once the user choose PW_i*, card computes

   RPW^{new} = h($ID_i \parallel PW_i$* $\parallel$ b),

   R_i^{new} = $R_i \oplus$ h($A_i \parallel$ RPW) $\oplus$ h($A_i \parallel RPW^{new}$),

   V_i^{new} = h($R_i^{new} \parallel A_i$) mod n_0

   Z_i^{new} = h($ID_i \parallel PW_i$*) $\oplus$ b.

4. Smart card stores R_i^{new}, V_i^{new} and Z_i^{new} in place of R_i, V_i, Z_i.

3.5 Cryptanalysis of the proposed scheme

This section presents the cryptanalysis of the proposed scheme. Through this cryptanalysis, it can be say that the proposed scheme overcomes all the pointed weaknesses, which are identified in Troung et al.'s authentication scheme. The cryptanalysis of the proposed scheme is explained below.

3.5.1 Password selection is done by the user

In the proposed scheme, unlike Troung et al.'s authentication scheme, password selection is done by the user. If the server selects the password, it could be long and random, which is difficult to remember for a registered user, if he/she does not use the system frequently. If the user is allowed to choose his password, then he/she has a choice of choosing his/her own password. In many real-life applications like online banking, social media subscriptions and many more accounts, it would be good if the passwords are memorable. Therefore the proposed scheme allows the user to choose the password.

3.5.2 Provides security to server secret key

The server's secret key is involved in encryption, decryption operations. Therefore it is important to provide security to server secret key. It is good to develop a scheme, which protects the server secret key from both legal user and the adversary.

In the proposed scheme, plain text format of server secret key is not used in any operation. Like Troung et al.'s scheme, the proposed scheme also generates random number 'e' and calculates h(x $\|$ e) on the server side of the registration phase. But unlike Troung et al.'s scheme, random number 'e' is not stored in the smart card. In the proposed scheme, random number 'e' stored in the database of the server. This secures the server secret key even though the smart card is lost. Hence, the proposed scheme provides security to the server secret key.

3.5.3 Untraceable user's identity

In the proposed scheme, untraceability of U_i is preserved in each login request. The scheme computes dynamic identity CID = h(K_u $\|$ T $\|$ m_i) $\oplus$ b in each session. This CID is different at each login attempt. In the proposed scheme, verification of ID_i and PW_i are done at the user side. Therefore, It is not required to use ID_i directly for the calculation of CID. Consider the equation CID = h(K_u $\|$ T $\|$ m_i) $\oplus$ b. Here, user ID is not used directly in the equation. Instead of ID, m_i is used to calculate CID in terms m_i = $R_i \oplus$ h(RPW $\|$ A_i) where RPW = h(ID_i $\|$ PW_i $\|$ b) and $A_i = ID_i \oplus$ b. It is not possible to guess the ID_i, even if $\mathscr{A}$ has the knowledge of m_i. To calculate ID_i, an adversary must know the PW_i and b, which are unknown to $\mathscr{A}$. Hence the proposed scheme provides untraceable user identity.

3.5.4 Provides security against replay attack

To avoid the replay attack, the server first verifies the freshness of the login message. To verify the login message freshness time stamp has been used. In the proposed scheme, once server receives the login message, it checks the validity of the time stamp T. Server verifies the condition T' - T $\leq \Delta$T and no other login message is with same parameter {CID, C_u, MAC_u, T} within the time period (T' + ΔT) and (T' - ΔT) where T' is the

timestamp, generated by the server. If this condition satisfied, S proceeds to the further calculation. If not, S rejects the login request message.

Assume that, an adversary intercepts the previous successfully authenticated login request message {CID, C_u, MAC_u, T} and sent the same message to the server. Here, S first checks the freshness of the login message by generating the new timestamp T*. If the timestamp T is not modified in the login request message, then the condition T* - T $\leq \Delta$T will fails and S rejects the login message. Suppose, an adversary $\mathscr{A}$ changed the time stamp T to T** in the login message. Then also server drops the session because the component values of login message involves the respective timestamp sent in the login message. Therefore, the proposed scheme provides security against the replay attack.

3.5.5 Perfect forward secrecy

In the proposed scheme, R_{US}= h(SK $\|$ b) is the authentication message sent from user to the server. An adversary $\mathscr{A}$ cannot calculate SK by obtaining the value R_{US}. To calculate SK, $\mathscr{A}$ must have the knowledge of R_{US} and b. $\mathscr{A}$ can get the information of R_{US} from public channel, but it is not possible to obtain the random number b, which is chosen by the user. Therefore to calculate SK, $\mathscr{A}$ has to guess both SK and b at the same time, which is not possible. Hence the proposed scheme provides perfect forward secrecy.

3.6 Formal analysis of the proposed scheme using BAN Logic

BAN-logic (Burrows et al., 1989) is a formal method for the analysis of cryptographic protocols. This section presents the formal proof for security using BAN logic. Before step into the proof, notations and the logical postulates are presented. In BAN logic, principles, encryption keys and formulas are considered as objects. To illustrate these objects following notations are used. In general notation, P and Q will be principles, X and Y as the range over statements and K will be the key. The symbols and notations used in the entire BAN model are given below.

The *BAN* Statements (Burrows et al., 1989)

Formal definition of BAN statements are given below:

P believes X **or** $P \mid\equiv X$**:** Principle P would be entitled to believe X. In particular, P can take X as true.

P sees X **or** $P \triangleleft X$: Principle P has received some message X and is capable of reading and repeating it.

P said X **or** $P \mid\sim X$**:** Principle P at some time sent a message including the statement X. It is not known whether this is a replay, though it is known that P believed X when he sent it.

P controls X **or** $P \Rightarrow X$**:** The principal P is an authority on X and should be trusted on this matter.

fresh (X) **or** #(X)**:** The message X is *fresh*; that is, X has not been sent in a message at any time before the current run of the protocol. This is usually true for *nonces*.

$P \overset{K}{\leftrightarrow} Q$**:** P and Q may use the *shared key* K to communicate. The key K is good in that it will be known only by P and Q.

Logical postulates

1. *Message meaning rules* concern the interpretation of messages. They all derive beliefs about the origin of messages:

$$\frac{P \mid\equiv Q \overset{K}{\leftrightarrow} P, P \triangleleft [X]_K}{P \mid\equiv Q \mid\sim X}$$

   If P believes that the key K is shared with Q and sees X encrypted under K, then P believes that Q once said X.

2. The *nonce-verification* rule expresses the check that a message is recent, and hence, that the sender still believes in it:

$$\frac{P \mid\equiv \#(X),\ P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$$

3. The *jurisdiction* rule states that if P believes that Q has jurisdiction over X, then P trusts Q on the truth of X:
$$\frac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$$

4. If one part of the formula is fresh, then the entire formula must be fresh:
$$\frac{P \mid\equiv \#(X)}{P \mid\equiv \#(X,Y)}.$$

3.6.1 Proposed scheme goals

Formal analysis of the proposed scheme is nothing but verification of goal (G) achievement. Our goal is, secure communication of session key (SK) between client and server. Therefore, to prove this U and S should trust each other. Hence there are mainly four goals (G_1, G_2, G_3, G_4) i.e.

G_1**:** $U \mid\equiv U \overset{sk}{\leftrightarrow} S$

G_2**:** $S \mid\equiv S \overset{sk}{\leftrightarrow} U$

G_3**:** $U \mid\equiv S \mid\equiv U \overset{sk}{\leftrightarrow} S$

G_4**:** $S \mid\equiv U \mid\equiv U \overset{sk}{\leftrightarrow} S$

3.6.2 Proposed scheme assumptions

For analysis of the proposed scheme using BAN logic, some assumptions are required to achieve the goal and those assumptions (A_1, A_2, A_3, A_4, A_5, A_6) are given below.

A_1**:** $U \mid\equiv U \overset{H(x\|e)}{\leftrightarrow} S$

A_2**:** $U \mid\equiv S \Rightarrow U \overset{sk}{\leftrightarrow} S$

A_3**:** $S \mid\equiv U \Rightarrow U \overset{sk}{\leftrightarrow} S$

A_4**:** $S \mid\equiv S \overset{H(x\|e)}{\leftrightarrow} U$

A_5**:** $S \mid\equiv \#T$

A_6**:** $S \mid\equiv \#T''$

3.6.3 Communicated messages

In the analysis of the proposed scheme, using BAN logic model it is to prove that, the proposed scheme authenticates mutually and shares a common session key. To prove this, the communication messages messages has been taken, which were send and received trough insecure channel between client and server.

Message 1: {CID, C_u, MAC_u, T}

Message 2: {MAC_s, C_s, T' }

Message 3: {R_{US} }

3.6.4 Idealized form of proposed scheme

To describe BAN logic model, the scheme messages should be change to the idealized forms, which are given below.

MAC_u**:** ($U \overset{H(x\|e)}{\leftrightarrow} S$, $w.e$.P, b, T)

MAC_s**:** ($U \overset{H(x\|e)}{\leftrightarrow} S$, $U \overset{sk}{\leftrightarrow} S$, b, T')

R_{US}**:** ($w.y$.P, $U \overset{sk}{\leftrightarrow} S$, b')

3.6.5 Security analysis proof

By the security proof of the proposed scheme, it is proved that the user and server securely share common session key SK. The security proofs are given below.

Apply message meaning rule with A_1 and MAC_s i.e.

$$\frac{U \mid\equiv U \overset{H(x\|e)}{\leftrightarrow} S, U \triangleleft (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, b, T')}{U \mid\equiv S \mid\sim (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, b, T')} \tag{3.1}$$

According to A_5 and MAC_s, apply the freshness rule. i.e.

$$\frac{U \mid\equiv \#T'}{U \mid\equiv \#(U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, b, T')}. \tag{3.2}$$

According to the equation (3.1) and (3.2) apply the nonce verification rule. i.e.

$$\frac{U \mid\equiv \#\,(U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, b, T'),\ U \mid\equiv S \mid\sim (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, b, T')}{U \mid\equiv S \mid\equiv S \overset{sk}{\leftrightarrow} U} \tag{3.3}$$

Which satisfies G_3.

According to A_2 and equation (3.3) apply jurisdiction rule. i.e.

$$\frac{U|\equiv S \Rightarrow U \overset{sk}{\leftrightarrow} S, U \mid\equiv S \mid\equiv U \overset{sk}{\leftrightarrow} S}{U \mid\equiv U \overset{sk}{\leftrightarrow} S} \tag{3.4}$$

Which satisfies G_1.

Apply message meaning rule with A_4 and R_{us} i.e.

$$\frac{S \mid\equiv S \overset{H(x\|e)}{\leftrightarrow} U, S \triangleleft (w.y.P, U \overset{sk}{\leftrightarrow} S, b)}{S|\equiv U \mid\sim (w.y.P, U \overset{sk}{\leftrightarrow} S, b)} \tag{3.5}$$

According to A_5 and MAC_s, apply the freshness rule. i.e.

$$\frac{S \mid\equiv \#T}{S \mid\equiv \#(w.y.P, U \overset{sk}{\leftrightarrow} S, b)}. \tag{3.6}$$

According to the equation (3.5) and (3.6) apply the nonce verification rule. i.e.

$$\frac{S \mid\equiv \#\,(w.y.P, U \overset{sk}{\leftrightarrow} S, b),\ S \mid\equiv U \mid\sim (w.y.P, U \overset{sk}{\leftrightarrow} S, b)}{S \mid\equiv U \mid\equiv (w.y.P, U \overset{sk}{\leftrightarrow} S, b)} \tag{3.7}$$

From the equation (3.7)

$$\frac{S \mid\equiv U \mid\equiv (w.y.P, U \overset{sk}{\leftrightarrow} S, b)}{S \mid\equiv U \mid\equiv U \overset{sk}{\leftrightarrow} S}. \tag{3.8}$$

Which satisfies G_4.

According to A_2 and (3.8) apply jurisdiction rule. i.e.

$$\frac{S|\equiv U \Rightarrow (w.y.P, U \overset{sk}{\leftrightarrow} S, b) S \mid\equiv U \mid\equiv (w.y.P, U \overset{sk}{\leftrightarrow} S, b)}{S \mid\equiv (w.y.P, U \overset{sk}{\leftrightarrow} S, b)} \tag{3.9}$$

From the equation (3.9)

$$\frac{S \mid\equiv (w.y.P, U \overset{sk}{\leftrightarrow} S, b)}{S \mid\equiv S \overset{sk}{\leftrightarrow} U}. \tag{3.10}$$

Which satisfies G_2.

In accordance with the proof, all the goals (G_1, G_2, G_3, G_4) are achieved. Now, it can be concluded that both the server and user believes that other believes the common session key SK is shared between the authorized user and the server.

3.7 Result of formal security verification using AVISPA tool

Automated Validation of Internet Security Protocols and Applications(AVISPA) is a tool, used to verify security of protocols. This tool provides a formal language to state

the protocols, their security properties and gives different backends that implement different state of automatic analysis techniques. On-the-fly Model-Checker (OFMC), CL-AtSe (Constraint Logic-based Attack Searcher), SAT-based Model checker and Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP) are the four different backends supported by AVISPA used for security analysis. To implement security protocols AVISPA uses a language called High-Level Protocol Specification Language (HLPSL).

HLPSL is a role-oriented language in which every participant plays a role during the execution. Each role is independent and through the channels, it communicates with others. In AVISPA, Dolev and Yao (1983) model used to design the intruder. This model helps to play the legitimate role in the system. While execution of protocol, HLPSL code is converted into the Intermediate Format(IF) using the translator hlpsl2if. Further, back-end reads the Intermediate Format and analyses if the security goals are satisfied or not. If the protocol satisfies all goals then AVISPA gives output as SAFE else it gives UNSAFE as output. The system also describes the number of sessions, principals and the roles. Based on back-ends, the Output Format (OF) is generated. After execution, the Output Format describes the result whether the protocol is SAFE or UNSAFE.

In the proposed scheme, registration phase and the login and authentication phase have been implemented using AVISPA. In this implementation, there are mainly two basic roles called alice and bob, which represents the participants as U_i and S. The role of U_i is given in the Figure 3.3. Here, the process starts by receiving start signal. After receiving the signal it changes state from 0 to 1, computes RPW and A_i and sends registration request message $\{\text{RPW}, A_i\}$ to S using symmetric key SKuisj. To send and receive the parameters between client and server, Snd() and Rcv() functions are used respectively. Similarly, the server S begins the process by receiving the message $\{\text{RPW}, A_i\}$ from the U_i. The role of S is given in the Figure 3.4.

In the Figure 3.5 and 3.6, the session and the environment roles specification has been given. In the session role, all the basic roles are included for composition. In the environment role, the global constants have been specified and one more session is

included. Here, an adversary can play as legitimate user role. In the environment role, the goals of the proposed scheme also specified.

```
role alice (Ui, Sj : agent,
      P: public_key,
      SKuisj : symmetric_key,
      H, Mul, Mod:hash_func,
      % H is hash function
      Snd,Rcv :channel(dy))
% Ui is the user; Sj is the server
played_by Ui
def=
 local State : nat,
          IDi, PWi, Rpw, Ai, Vi, Zi, X, B,Ri, Ku, Ks, E, Rus, Cu, Cs, D, Mi,Sks,Sku,
          T1, T2, CID, MACu, MACs, W, Y: text
     const alice_bob_T1, bob_alice_T2,
          alice_bob_Ri, bob_alice_Fi,
          subs1, subs2 : protocol_id
init State:=0
          transition
1.        State=0 /\ Rcv(start)=|>
          State':=1 /\ B':=new()
          /\ Rpw' := H(IDi.PWi.B')
          /\Ai' := xor(IDi, B')
          /\ Zi':= xor(H(IDi.PWi),B')
% Send the registration request message
          /\ Snd({Ai.Rpw'}_SKuisj)
          /\ secret({IDi}, subs1, Sj)
          /\ secret({PWi, B'}, subs2, Ui)
% Receive the smart card from the registration server Sj
 2. State = 1 /\Rcv({Mod(H(H(xor(H(E'.X)), H(H(IDi.PWi.B').xor(IDi, B'))).H(E'.X))).
Mul(E'.P). H(xor(H(E'.X)), H(H(IDi.PWi.B').xor(IDi, B')))}_SKuisj) =|>
% Login phase
 State':= 2
           /\ B' := xor(H(IDi.PWi),Zi)
           /\ Rpw' := H(IDi.PWi.B')
           /\ Ai' := xor(IDi, B')
           /\ Mi' := xor(Ri,H(Rpw'.Ai'))
           /\ Vi':= Mod(H(H(xor(Mi).H(Rpw.Ai)).Ai))
           /\ T1' := new()
           /\ W' := new()
           /\ Cu' := Mul(W'.P)
           /\ Ku' := Mul(W'.Mul(E'.P))
           /\ CID' := xor(H(Ku'.T1'.Mi'), B')
           /\ MACu' := H(CID'.Mi'.Ku'.B')
           /\ Snd({CID'.Ku'.MACu'.T1'}_SKuisj)
           /\ witness(Ui, Sj, alice_bob_T1, T1')
% Authentication phase
3. State = 2 /\ Rcv (H(H(Mul(Y'.Mul(W'.P)).B'. Mi').B'.T2'.Mi'), T2, Mul(Y'.P)) =|>
 State' := 3
      /\ Ks' := Mul(W'.Mul(Y'.P))
      /\ Sku' :=H(Ks'.B'. Mi')
      /\ MACs':= H(Sku'.B'.T2'.Mi)
      /\ Rus' := H(Sku'.B')
      /\ Snd({Rus'}_SKuisj)
      end role
```

Figure 3.3 Role specification for the U_i of the proposed scheme

```
role bob (Ui, Sj : agent,
      P: public_key,
      SKuisj : symmetric_key,
      H, Mul, Mod:hash_func,
      % H is hash function
      Snd,Rcv :channel(dy))
% Ui is the user; Sj is the server
played_by Sj
def=
 local State : nat,
 IDi, PWi, Rpw, Ai, Vi, Zi, X, B, D, Mi, Ri, E, Ku, Ks, Rus, Cu, Cs, Sks,Sku, T1, T2, T3, CID, MACu, MACs, W,Y: text
            const alice_bob_T1, bob_alice_T3,
            alice_bob_Ri, bob_alice_Fi,
            subs1, subs2 : protocol_id
     init State := 0
     transition
% Registration phase
% Receive the registration request message from the user
1.         State = 0 /\ Rcv({H(IDi.PWi.B').xor(IDi, B')}_SKuisj) =|>
% Keep X and B' secret to Sj and PWi, B to Ui
           State' := 1 /\ secret({X',B'}, subs1, Sj)
           /\ secret({PWi, B'}, subs2, Ui)
            /\E' := new()
            /\D' := Mul(E'.P)
            /\ Mi' := H(E'.X)
            /\Ri' := H(xor(H(E'.X)), H(H(IDi.PWi.B').xor(IDi, B')))
            /\Vi' := Mod(H(Ri'.Mi'))
            /\ Snd ({Vi. D'. Ri}_SKuisj)
% Login phase
% Receive the login request message
2. State = 1 /\ Rcv({xor(H(Mul(W'.Mul(E'.P)).T1'.Mi'), B').Mul(W'.P).H(xor(H(Mul(W'.Mul(E'.P)).
T1'.Mi'), B').H(E'.X).Mul(W'.Mul(E'.P)).xor(H(IDi.PWi),Zi)).T1'}_SKuisj) =|>
 State' := 2
     /\ T3' := new()
     /\ Mi' := H(E'.X)
     /\ Ku' := Mul(E'.Mul(W'.P))
     /\ B' := xor(xor(H(Mul(W'.Mul(E'.P)).T1'.Mi'), B'), H(Mul(W'.Mul(E'.P)).T1'.Mi'))
     /\ MACu' := H(xor(H(Mul(W'.Mul(E'.P)).T1'.Mi'), B').Mi'.Ku'.B')
     /\ Y' := new()
     /\ Cs' := Mul(Y'.P)
     /\ Ks' := Mul(Y'.Mul(W'.P))
     /\ Sks' := H(Ks'.B'. Mi')
     /\ MACs' := H(Sks'.B'.T2'.Mi')
% Send the authentication request message
     /\ Snd ({MACs'. T2'. Cs'}_SKuisj)
     3. State = 2 /\ Rcv ({H(Sku'.B')}_SKuisj) =|>
     State' := 3  /\ Rus' := H(Sks'.B')

end role
```

Figure 3.4 Role specification for the S of the proposed scheme

```
role session(Ui, Sj: agent,
P : public_key,
SKuisj : symmetric_key,
H, Mul, Mod : hash_func)
def=
local SI, SJ, RI, RJ: channel (dy)
composition
        alice(Ui, Sj, P, SKuisj, H, Mul, Mod, SI, RI)
        /\ bob(Ui, Sj, P, SKuisj, H, Mul, Mod, SJ,
RJ)
end role
```

Figure 3.5 Role specification for the session of the proposed scheme

```
role environment()
def=
const ui, sj: agent,
skuisj : symmetric_key,
p : public_key,
h, mul, mod: hash_func,
alice_bob_T1, bob_alice_T2,
alice_bob_Ri, bob_alice_Fi,
subs1, subs2 : protocol_id
intruder_knowledge = {ui, sj, p, h,mul, mod}
composition
session(ui, sj, p, skuisj, h, mul, mod)
/\session(ui, sj, p, skuisj, h, mul, mod)
end role
goal
secrecy_of subs1
secrecy_of subs2
authentication_on alice_bob_T1
authentication_on alice_bob_Ri
authentication_on bob_alice_T2
authentication_on bob_alice_Fi
end goal
environment()
```

Figure 3.6 Role specification for the goal and environment of the proposed scheme

The obtained results of formal verification using the AVISPA tool has been presented in Figure 3.7 and 3.8. Here, OFMC and CLAtSe are used as the backend to get the simulation result of the proposed scheme. By this simulation result, it has been proved that the scheme is secure from all defined weaknesses.

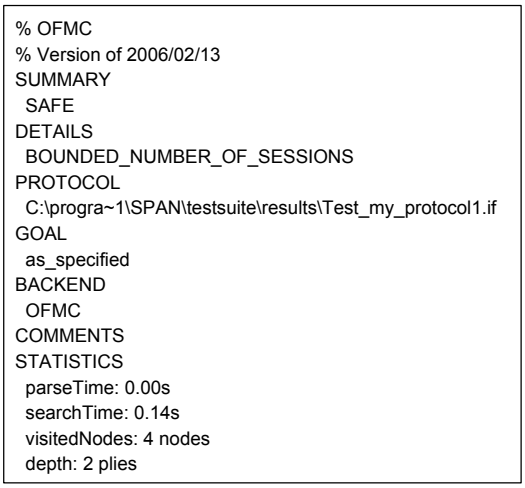

Figure 3.7 OFMC simulation result of proposed scheme

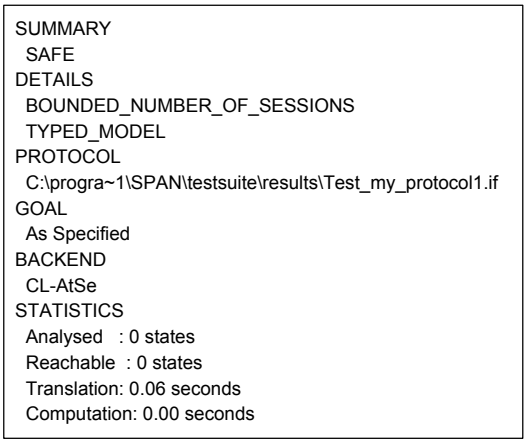

Figure 3.8 CLAtSe simulation result of proposed scheme

3.8 Computation efficiency

In the performance analysis has mainly focused on computation cost. Here, the computation cost of the proposed scheme is compared with the Wen and Li (2012), Ding et al. (2012) and Truong et al. (2014) schemes. Comparison results are presented in the Table 3.2. Here, H represents the one-way hash function, ME denotes modular exponential operation, and ECC signifies the elliptic curve scalar multiplication. The estimated execution time of the proposed scheme and Wen and Li (2012), Ding et al. (2012) and

Truong et al. (2014) schemes also computed and presented in the same table. The simulation result illustrated by Xie et al. (2017) has been taken to compute the execution time. According to Xie et al. (2017), simulation has been done using Miracl library. The environment used for simulation is Windows 7 sp1 64-bit PC, Intel Core i5-3210M CPU of 2.5 GHz, 8GB RAM. According to this simulation, the execution time of one hash operation takes 0.068 ms (millisecond), the execution time of one block encryption/decryption is 0.56 ms, the execution time of one modular exponentiation is 3.043 ms and the execution time of one scalar multiplication on the elliptic curve is 2.501 ms.

Table 3.2 Computational cost comparison

Schemes	**Computation cost**	**Estimated Time (ms)**
Wen and Li (2012)	$21H$	1.768 ms
Ding et al. (2012)	$16H + 3ME$	10.829 ms
Truong et al. (2014)	$13H + 3ECC$	8.659 ms
Proposed scheme	$14H + 6ECC + 1ME$	19.001 ms

The proposed scheme takes $14H + 6ECC + 1ME$ in login and authentication phase to execute the scheme. Estimated computation time to run the scheme is 19.001ms. Compare to the other schemes; the proposed scheme takes more computation time to execute the scheme because, in the proposed scheme, the fuzzy-verifier technique has been adopted gintroduced by Wang and Wang (2016) for quick password verification. This technique also overcomes the dilemma of usability maintenance while achieving security even after smart card tampering. The proposed scheme also use the public key cryptography to preserve the anonymity and secrecy. The proposed scheme resists all attacks whereas other schemes are vulnerable to different attacks as mentioned in Table 3.3.

Table 3.3 presents the comparison result of various security issues in the Wen and Li (2012), Ding et al. (2012) and Truong et al. (2014) schemes with the proposed scheme. Here, Y denotes that the scheme provides security for the particular attack or goal and N represents that the scheme does not provide any security for the particular attack or goal. The proposed scheme resists all attacks and satisfies all goals. This result shows

that the proposed scheme is more robust when compared to other schemes.

Table 3.3 Comparison of security attacks and characteristics

Goals	**S1**	**S2**	**S3**	**Proposed Scheme**
Resilience to denial of service attack	Y	Y	N	Y
Resilience to password guessing attack	Y	N	Y	Y
Resilience to parallel session attack	Y	Y	Y	Y
Resilience to server masquerading attack	Y	N	Y	Y
Resilience to insider attack	N	N	Y	Y
Resilience to stolen smart card attack	N	N	Y	Y
Resilience to replay attack	Y	Y	N	Y
Resilience to impersonation attack	Y	N	N	Y
Provides quick password verification	N	Y	Y	Y
Provides security for server secret key	Y	N	N	Y
Provides user anonymity	Y	N	N	Y
Provides perfect Forward Secrecy	N	N	N	Y

S1: Wen and Li (2012) scheme, S2: Ding et al. (2012) scheme, S3: Truong et al. (2014) scheme

3.9 Conclusion

In this study, cryptanalysis of Troung et al.'s scheme has been presented and its vulnerabilities were identified. To overcome pointed security issues, a secure authentication scheme has proposed. Through the security analysis, it is proved that the proposed scheme protects all pointed weaknesses. The proposed scheme security is also analyzed using BAN logic and simulated using AVISPA tool. The computation cost of the proposed scheme is calculated, estimated its execution time and compared it with other related schemes. Through the result, it is observed that the proposed scheme takes more execution time when compared to the other scheme. But, the scheme resists all attacks and overcomes the dilemma of usability maintenance while achieving security even after the smart card tampering. Hence it can be concluded that the proposed scheme is secure, robust and practically implementable.

Chapter 4

A NOVEL TWO-FACTOR REMOTE USER AUTHENTICATION SCHEME FOR RESOURCE LIMITED WIRELESS ENVIRONMENTS

4.1 Introduction

Wireless network technologies are rapidly extending their capabilities. In recent years it has become more available, affordable and easy to use. Home users and small businesses are adopting wireless technology in great number. With the growth of the wireless network, the vulnerabilities and threats are also increasing. Therefore, security is the main challenge of the whole system. The wireless network is vulnerable to security attacks because of its transparency in transmission media. Anyone within the range of a wireless device can intercept the communication messages without interrupting the flow of data communicating between the user and the server. Zhu and Ma (2004) said that wireless communication suffers from threats inherited from wired networks and that are specific in the wireless environment. Because of its limited resource and a higher channel error rate than wired networks, those security schemes in the wired network couldn't be used directly in the wireless environment.

Authentication is the essential security mechanism in the wireless environment to ensure that the authorized user properly using the service. There are many authentication schemes have been proposed in the wireless environment. But, there are many vulnerabilities in those remote user authentication schemes, which may breaks its secu-

rity. Therefore, proposing a secure remote user authentication scheme is essential. But, it is important to consider the limited resource, while designing the security scheme.

4.1.1 Resistance to replay attack without time synchronization

The replay attack is one of the major active attacks in the wireless environment. In this attack, an adversary impersonates a legal user using his/her login message. The attacker intercepts the previous successfully authenticated login request message of a user and resends the same message to the server. If the scheme doesn't resist replay attack, then the server accepts the login message and authenticates the user.

To avoid the replay attack, the server must accept a fresh login request message at every session. Till date, timestamp or time synchronization is the efficient solution for freshness verification of login request messages. But, timestamp-based remote user authentication schemes having some disadvantages, which are given below:

- It is difficult to verify the timestamp, when entities are located in different time zones or when there is a congested network environment.

- Because of the unpredictable transmission delay in the network environment communication will be delayed. In timestamp based schemes time delay causes the denial of service.

- Clock synchronization mechanism is expensive for some applications.

To avoid replay attack without time synchronization, some researchers have proposed the authentication schemes using the nonce. But still, the schemes fail to resist replay attack. Therefore it can be concluded that the nonce based schemes are also not efficient to withstand the replay attack.

In this chapter, a novel remote user authentication scheme for the wireless environment has been proposed. This scheme resists replay attack without the involvement of time synchronization. To achieve this, Elliptic Curve Diffie-Hellman key exchange (ECDH) method has been employed in the scheme. Through this technique, the scheme checks the freshness of the login request message. The scheme also overcomes other

security attacks and achieves the security goals. Through the security analysis, the proposed scheme security has been discussed and proved that the proposed scheme provides security to all the possible attacks. The formal verification proof of the proposed scheme using BAN login has been given and simulated the scheme using the AVISPA tool. Computation cost and communication cost of the proposed scheme are also calculated and compared the result with the other remote user authentication schemes.

The rest of the chapter has been assembled as follows. Section 4.2 briefly illustrates cryptographic preliminaries needed propose the scheme. Section 4.3 proposes the novel two-factor remote user authentication scheme. Section 4.4 discusses the security aspects of the proposed scheme in detail followed by formal analysis of the result using BAN logic of the proposed scheme in section 4.5. Section 4.6 presents the formal verification of the proposed scheme using AVISPA tool. Section 4.7 compares the performance of the proposed scheme with recently proposed schemes. Section 4.8 depicts the concluding remarks of this work.

4.2 Cryptographic preliminaries

This section illustrates about cryptographic preliminaries needed to discuss the proposed scheme.

Elliptic Curve Discrete Logarithm Problem(ECDLP): An elliptic curve E defined over a finite field $GF(q)$ and two points P, Q $\in$ E, it is hard to find an integer x$\in Z_p$* such that Q=xP.

Elliptic Curve Diffie Hellman (ECDH): Elliptic Curve Diffie Hellman (ECDH) key exchange is the the classical Diffie-Hellman key exchange which exchanges secret information or secure keys between two parties. The keys are exchanged between A and B as follows.

System selects an elliptic curve E_p over the prime finite field F_p where 'p' is a large prime number and select a point on elliptic curve P of order n. The algorithm is as follows.

1. A generates a random number k_A in the interval $[1, n-1]$ and performs a

scalar multiplication $Q_A = k_A * P$ Then, sends Q_A to B

2. B also generates a random number k_B and computes $Q_B = k_B * P$ by scalar multiplication in the same way as described above and sends Q_B to A.
3. After receiving Q_B from B, A computes $S_1 = k_A * Q_B$. Similarly, B receives Q_A from A and computes $S_2 = k_B * Q_A$.
4. A and B shares S_1 and S_2 between them. Thus, two entities exchanges the keys securely.

4.3 Proposed Scheme

The proposed scheme contains three phases (1) Registration phase, (2) Login and authentication phase, (3) Password change phase. The notations used throughout this chapter is given in the Table 4.1

Table 4.1 Notations and Descriptions.

Notations	Descriptions
S	Server
U_i	i^{th} User
ID_i	i^{th} identity of U_i
x	Secret key maintained by S
PW_i	Password of i^{th} User
SC_i	Smart card of U_i
$\oplus$	Bitwise XOR operator
$\|\|$	Concatenation operator
h(.)	One-way hash function

4.3.1 Registration phase

If a user desire to access the service or information from a remote system, first he/she has to register and get the smart card. The Figure 4.1 presents the registration phase of the proposed scheme. The steps involved in new user registration are given below:

1. U_i selects ID_i, PW_i and a random number b.
2. User system computes RPW = h($ID_i \,||\, PW_i \,||$ b), $A_i = ID_i \oplus$ b and sends registration request {RPW, A_i} to the server S via secure channel.

3. S receives the registration request message {RPW, A_i} from U_i, generates random number 'e' and computes

   m_i = h(e || x)

   $N_i = m_i \oplus A_i \oplus$ RPW.

4. S stores 'e' into the database, deposits {h(.), N_i} into the smart card and delivers it to U_i via secure channel.

5. U_i receives {h(.), N_i} from S and computes

   Z_i = h(ID_i || PW_i) $\oplus$ b.

   R_i = h(h(ID_i || PW_i) || b || N_i)

   U_i store Z_i and R_i into the smart card. Finally smart card contains the values {h(.), N_i, R_i, Z_i}.

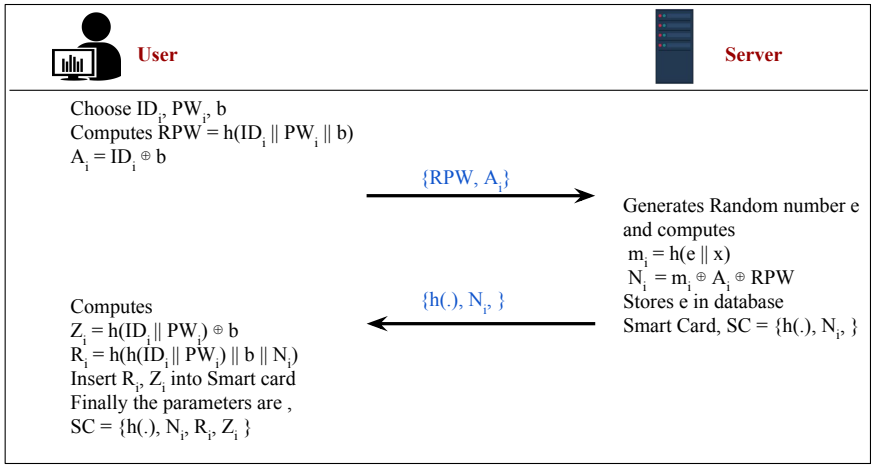

Figure 4.1 Registration phase of the proposed scheme

4.3.2 Login and authentication phase

Figure 4.2 presents the login and authentication phase of the proposed scheme. Whenever U_i wants to login into the server, he/she insert the smart card into the terminal device and inputs ID_i and PW_i. The steps involved in the login and authentication phase are given below:

1. User system extracts b* = h($ID_i \parallel PW_i$) $\oplus$ Z_i and computes
   R_i* = h(h($ID_i \parallel PW_i$) $\parallel$ b* $\parallel$ N_i)

2. User system verifies the condition R_i* = R_i or not. If R_i* and R_i are not equal, then the entered ID_i or PW_i is wrong. Therefore, the card drops the session. If both ID_i and PW_i are correct then the condition R_i* = R_i will be true and user system continue with the next step.

3. User system begins the ECDH key exchange algorithm. First, system selects an elliptic curve E_p over the finite field F_p with a large prime number 'p'. System picks a base point 'P' of order 'n'. Further, system generates a random number w $\in$ [2, n-1] and computes $M_1 = w * P$ and sends $\{M_1, P\}$ to the server.

4. Server receives $\{M_1, P\}$ and generates a random number $y \in$ [2, n-1], and computes $M_2 = y * P$ and sends $\{M_2\}$ to the user.

5. After receiving $\{M_2\}$ from the server, U_i computes
   RPW = h($ID_i \parallel PW_i \parallel$ b)
   $A_i = ID_i \oplus$ b
   $m_i = N_i \oplus$ RPW $\oplus A_i$
   $K_A = w * M_2$
   CID = h($K_A \parallel m_i$) $\oplus$ b and
   C_1 = h(CID $\parallel m_i \parallel K_A \parallel$ b).

6. U_i transmits the login request {CID, K_A, C_1} to the server S via public channel.

After receiving the login request message {CID, K_A, C_1 } from user U_i, server begins the authentication process by checking the freshness of the login message. Steps involved in the authentication at the server side is as follows:

1. To check the freshness of the login message, server computes $K_B = y * M_1$ and compares K_B with received K_A. If $K_A = K_B$, then the received message is fresh and S accepts the login message. Otherwise, S rejects the login request message.

2. S computes m_i = h(e $\|$ x)
   b' = CID $\oplus$ h(K_B $\|$ m_i)
   and computes C_1' = h(CID $\|$ m_i $\|$ K_B $\|$ b').

3. S compares C_1' with the received C_1. if $C_1 \neq C_1$', then the server drops the session, else proceeds the mutual authentication.

4. In mutual authentication, S computes
   SK = h(K_B $\|$ b'$\|$ m_i),
   C_2 = h(SK $\|$ b'$\|$ m_i $\|$ K_B).
   S transmits the mutual authentication message $\{C_2\}$ to U_i via public channel.

5. U_i receives the mutual authentication message $\{C_2\}$ and computes
   SK'= h(K_A $\|$ b $\|$ m_i),
   C_2'= h(SK $\|$ b $\|$ m_i $\|$ K_A). Verifies C_2' with received C_2.
   If C_2'= C_2 then server will assure that, authentication done with legal user.

6. After successful authentication, further communication between the user and the server will continue using the sessions keys SK = h(K_B $\|$ b'$\|$ m_i) and SK'= h(K_A $\|$ b $\|$ m_i).

4.3.3 Password change phase

Whenever U_i wish to change the password, he/she should perform the following steps:

1. U_i inserts the smart card and enters the ID_i and PW_i.

2. U_i Computes b* = h(ID_i $\|$ PW_i) $\oplus$ Z_i and
   R_i* = h(h(ID_i* $\|$ PW_i*) $\|$ b* $\|$ N_i)
   smart card verifies, if the computed R_i* = R_i or not. If R_i* and R_i are not equal, then the entered ID_i or PW_i are wrong and the smart card drops the session, else smart card computes RPW = h(ID_i $\|$ PW_i $\|$ b)
   A_i = ID_i $\oplus$ b and asks for a new password PW_i^{new}

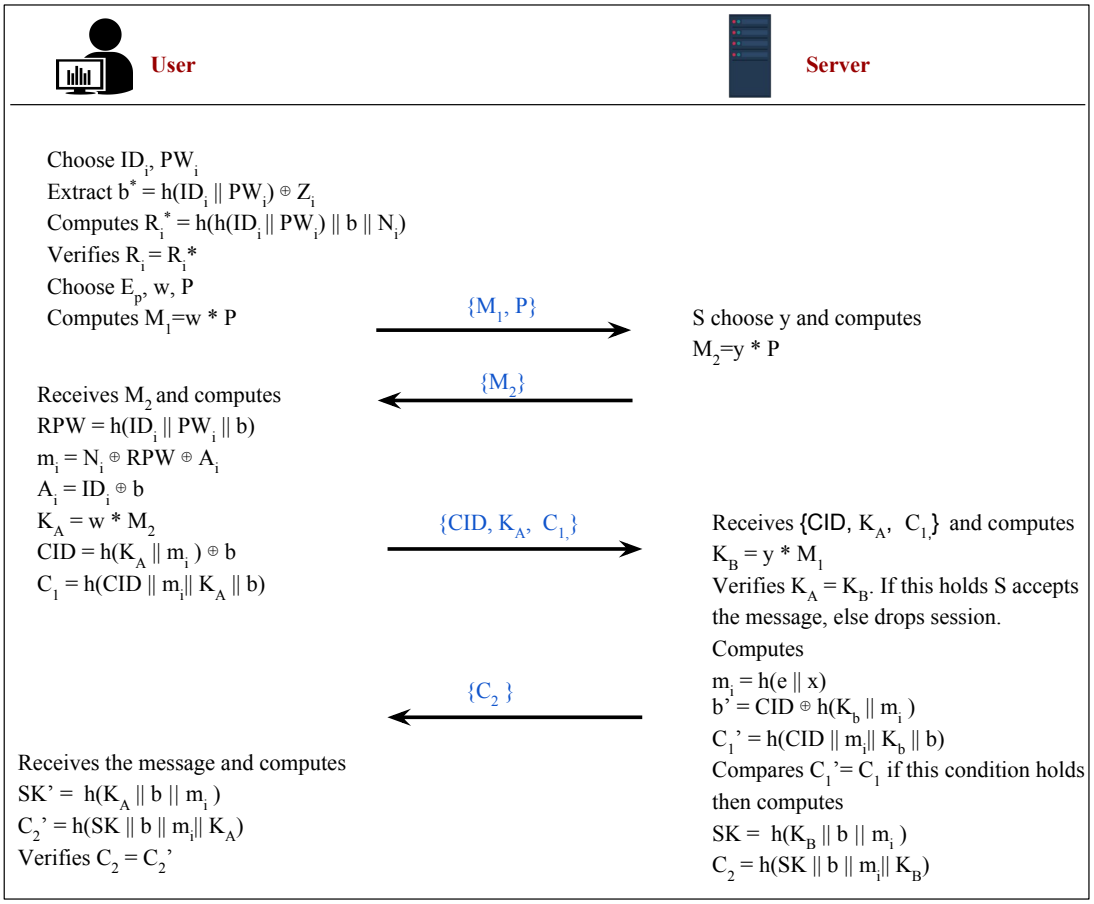

Figure 4.2 Login and authentication phase of the proposed scheme

3. Once the user inputs new password PW_i^{new}, the smart card computes

   $RPW^{new} = h(ID_i \| PW_i^{new} \| b)$,

   $N_i^{new} = N_i \oplus RPW \oplus RPW^{new}$,

   $R_i^{new} = h(h(ID_i \| PW_i^{new}) \| b \| N_i^{new})$ and

   $Z_i^{new} = h(ID_i \| PW_i^{new}) \oplus b$.

4. Smart card stores R_i^{new}, N_i^{new} and Z_i^{new} in place of R_i, N_i, Z_i.

4.4 Cryptanalysis of the Proposed Scheme

This section presents cryptanalysis of the proposed scheme. The detailed security analysis is explained below.

4.4.1 Resists replay attack

To avoid the replay attack, the server must accept a fresh login request message at every session. To check the freshness of the login message, time synchronization is one of the possible methods. Time synchronization has it's own disadvantages. Therefore, ECDH key exchange method has been used to verify the freshness of the login message.

In the proposed scheme, communication with the server will begin with ECDH key exchange method. According to this, user system first selects an elliptic curve E_p over the finite field F_p with a large prime number 'p'. Also chooses a base point 'P' of order 'n'. Then, user system first selects a random number w and computes $M_1 = w * P$ and sends $\{M_1, P\}$ to the server. Server receives the message $\{M_1, P\}$ and selects y, and computes $M_2 = y * P$ and sends $\{M_2\}$ to the user. After receiving $\{M_2\}$, user system starts to compute the login request parameters $\{\text{CID}, K_A, C_1\}$ where $K_A = M_2 * w$, CID = h($K_A \parallel m_i$) $\oplus$ b and C_1 = h(CID $\parallel m_i \parallel K_A \parallel$ b). In the server side, S receives the login message $\{\text{CID}, K_A, C_1\}$ and calculates $K_B = M_1 * y$ and compares K_B with received K_A. If $K_A = K_B$, then the received message is fresh and S accepts the login message. Otherwise S drops the session.

The values of w, y and P are generated by the system in every session. These values are different in every time and will not be available after session termination. Hence, adversary cannot resend the message $\{M_1, P\}$ to perform replay attack. Assume that, adversary resends previous successful message $\{M_1, P\}$ to the server. Then, server receives the message and generates new random number y', computes M_2'= $y' * P$ and sends $\{M_2'\}$ to the user. Here, adversary must compute new values for K_A, CID and C_1 before sending the message $\{\text{CID}, K_A, C_1\}$ where $K_A = M_2 * w$, CID = h($K_A \parallel m_i$) $\oplus$ b and C_1 = h(CID $\parallel m_i \parallel K_A \parallel$ b). To move on to the next step, adversary has to calculate K_A'= $M_2' * w$. But adversary does not have any knowledge about w. Hence, he/she cannot calculate K_A'. He can choose new random number w' to calculate K_A'= $M_2 * w'$. But, adversary cannot calculate new CID' and C_1' because b and m_i are unknown.

If adversary sends previous login message $\{\text{CID}, K_A, C_1\}$ directly to the server after getting $\{M_2'\}$, server computes K_B'= $M_1 * y'$ and compares K_B' with received K_A. But, K_A is calculated using the old random number y and K_B' computation has been done

using new random number y'. Therefore $K_B' \neq K_A$ and server rejects the message and drops the session. Hence, the proposed scheme resists the replay attack.

4.4.2 Provides security to the server secret key

In the proposed scheme, server secret key 'x' is not used as plain text in any operation. In the registration phase of the proposed scheme, server generates a random number 'e' and calculates m_i = h(x $\|$ e). Instead of server's secret key x, the computed m_i is used for further calculations. Hence the server's secret key will be secured from adversary. Even though, the adversary succeeds to get m_i, he/she cannot get the server secret key x because, server generated random number e is stored in the database and adversary will not get this. Hence, the proposed scheme provides security to the server secret key.

4.4.3 Untraceable user's identity

In the proposed scheme, secrecy of U_i is preserved in each session. The proposed scheme computes dynamic identity CID = h($K_A \| m_i$) $\oplus$ b and this CID is different at every login attempt. In the proposed scheme, ID_i is not used in the plain text format to calculate CID. To calculate CID, $m_i = R_i \oplus$ h(RPW $\|$ A_i) is used instead of ID_i, where RPW = h($ID_i \| PW_i \|$ b) and $A_i = ID_i \oplus$ b. It is not possible to guess the ID_i, even if adversary has knowledge of m_i. To calculate ID_i from RPW, adversary must know the unknown values PW_i, RPW and b. Adversary cannot guess all three parameters simultaneously in polynomial time. Also, adversary cannot be able to calculate ID_i from $A_i = ID_i \oplus$ b. Here, adversary must have the knowledge of two unknowns A_i and b and he/she cannot guess these two values at the same time. Hence the proposed scheme provides untraceable user identity.

4.4.4 Provides perfect forward secrecy

Perfect forward secrecy assures that, the session key SK remains safe, even though the server secret key is compromised. Assume that the server secret key is compromised. In the proposed scheme, server's session key is SK = h($K_B \|$ b'$\| m_i$) and the user's session key is SK'= h($K_A \|$ b $\| m_i$) where m_i = h(x $\|$ e). Here, K_A, K_B, b and e are involved to calculate SK. Adversary can get K_A from the login request message {CID, K_A, C_1} and

according to Diffie-Hellman key exchange technique adversary can assume K_A=K_B. But, adversary cannot get the random numbers b and e, where e is stored in server database and b is the random number chosen by the user. Therefore, the session key remains secure. Hence the proposed scheme provides proper perfect forward secrecy.

4.4.5 Resists insider attack

In the proposed scheme user does not submit plain text ID_i and PW_i to the server. To protect the password from insider attack, scheme uses the random number b. In the proposed scheme user computes RPW = h($ID_i \parallel PW_i \parallel$ b), $A_i = ID_i \oplus$ b and submits {RPW, A_i} to the server. Therefore insider of the server cannot obtain the user's password. As all values of RPW i.e ID_i, PW_i and b are unknown to the insider, it is not possible to guess all three parameters simultaneously. Hence, the proposed scheme resists insider attack.

4.4.6 Resists offline password guessing attack

In the proposed scheme, adversary cannot find PW_i from smart card parameter or communicated messages. Assume that the adversary steals smart card and obtains the parameters {h(.), N_i, R_i, Z_i}, which are stored in the smart card. In stored parameter, Z_i = h($ID_i \parallel PW_i$) $\oplus$ b and R_i = h(h($ID_i \parallel PW_i$) $\parallel$ b $\parallel N_i$). Here, adversary can obtain Z_i, R_i and N_i from the smart card. But, to guess the password using above equations, adversary must know ID_i and b. In the section 4.5.3 it is proved that, the proposed scheme provides untraceable user's identity. Therefore, it is not possible for the adversary to guess password and verify his/her guess using Z_i = h($ID_i \parallel PW_i$) $\oplus$ b and R_i = h(h($ID_i \parallel PW_i$) $\parallel$ b $\parallel N_i$). Because, every time adversary must guess atleast two unknown values in polynomial time, which is not possible. Hence, the proposed scheme resists offline password guessing attack.

4.5 Formal analysis of the proposed scheme using BAN Logic

This section presents the formal evaluation results using BAN logic. The symbols and notations used in entire BAN model are illustrated in the section 3.6 of the Chapter 3.

4.5.1 Proposed scheme goals

Formal analysis of the proposed scheme is nothing but verification of goals (G). Our goal is to secure the session key (SK) between the user and the server. Therefore, to prove this U and S should trust each other. Hence there are mainly two goals i.e.

$G_1 : S \mid\equiv S \overset{sk}{\leftrightarrow} U$

$G_2 : S \mid\equiv U \mid\equiv U \overset{sk}{\leftrightarrow} S$

4.5.2 Proposed scheme assumptions

For analysis of the proposed scheme using BAN logic, some assumptions are required to achieve the goals and those assumptions are given below.

$A_1 \;\; U \mid\equiv U \overset{H(x\|e)}{\leftrightarrow} S$

$A_2 \;\; U \mid\equiv S \Rightarrow U \overset{sk}{\leftrightarrow} S$

$A_3 \;\; S \mid\equiv U \Rightarrow U \overset{sk}{\leftrightarrow} S$

$A_4 \;\; S \mid\equiv S \overset{H(x\|e)}{\leftrightarrow} U$

$A_5 \;\; S \mid\equiv \#K_A$

4.5.3 Communicated messages

In analysis of the proposed scheme, BAN logic model has been used to prove that, the proposed scheme authenticates mutually and shares a common session key. To prove this, the communication message between client and server trough insecure channel has been used.

1. **Message 1:** $\{\text{CID}, K_A, C_1\}$

2. **Message 2:** $\{C_2\}$

4.5.4 Idealized form of proposed scheme

To describe BAN logic model, the scheme's messages should be changed to the idealized forms, which are given below.

C_1 : ($U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S$, b)

C_2 : ($U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S$, b)

4.5.5 Security analysis proof

Analysing the idealized form of the proposed scheme, it is proved that the user and the server securely share common session key SK.

Apply message meaning rule with A_4 and C_2 i.e.

$$\frac{S \mid\equiv S \overset{H(x\|e)}{\leftrightarrow} U, S\triangleleft\{(U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)\}}{S\mid\equiv U \mid\sim(U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)} \tag{4.1}$$

According to A_5 and C_2, apply the freshness rule. i.e.

$$\frac{S \mid\equiv \#K_A}{S \mid\equiv \#(U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)}. \tag{4.2}$$

According to equation (4.1) and (4.2) apply the nonce verification rule and we get

$$\frac{S \mid\equiv \#\,(U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b),\ S \mid\equiv U \mid\sim\ (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)}{S \mid\equiv U \mid\equiv (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)} \tag{4.3}$$

From the equation (4.3) we get

$$S \mid\equiv U \mid\equiv U \overset{sk}{\leftrightarrow} S. \tag{4.4}$$

Which satisfies G_2.

According to A_2 and (4.4) apply jurisdiction rule. i.e.

$$\frac{S\mid\equiv U \Rightarrow (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b) S \mid\equiv U \mid\equiv (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)}{S \mid\equiv (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)} \tag{4.5}$$

From the equation (4.5) we get

$$\frac{S \mid\equiv (U \overset{H(x\|e)}{\leftrightarrow} S, U \overset{sk}{\leftrightarrow} S, U \overset{wy}{\leftrightarrow} S, b)}{S \mid\equiv S \overset{sk}{\leftrightarrow} U}. \tag{4.6}$$

Which satisfies G_1.

In accordance with proof goals G_1 and G_2 are achieved. Now, it can be conclude that both the server and user believes, they have shared shared the common session key SK securely between them.

4.6 Result of formal security verification using AVISPA tool

In the proposed scheme, the registration phase, the login and authentication phases have been implemented in AVISPA. In this implementation, there are mainly two basic roles called alice and bob, which represents the participants as U_i and S respectively. The role of U_i is given in Figure 4.3. Here, the process starts by receiving the start signal. After receiving the signal, it changes state from 0 to 1. Further, it computes RPW, A_i and sends registration request message {RPW, A_i} to S using symmetric key SKuisj. To send and receive the parameters between client and server, Snd() and Rcv() functions are used respectively. Similarly, the server S begins the process after getting the message {RPW, A_i} from the U_i. The role of S has been given in Figure 4.4.

The Figure 4.5 and 4.6 presents the session and environment roles specification. The session role presents the composition of the basic roles. The environment role specifies the global constants and included one more session, where the adversary can play as legitimate user roles. The role environment also specifies the goals of the proposed scheme. The specified and achieved goals are given below.

1. secrecy_of s1, s2, s3, s4, s5, s6: It represents various secret information and credentials that are kept secret from the adversary while execution.

2. authentication_on alice_bob_u: Authenticates U_i in the server side after receiving the login message from U_i

3. authentication_on bob_alice_s: Authenticates S in the user side after receiving the mutual authentication message from U_i

The obtained AVISPA result has been presented in the Figure 4.7 and 4.8.

The OFMC and CLAtSe back ends are used to obtain the simulation result of the proposed scheme. By this simulation result, it can prove that the proposed scheme is secure against all defined goals.

```
role alice (Ui, S : agent,
P : public_key,
SKuisj : symmetric_key,
H, Mul:hash_func,
Snd, Rcv : channel (dy))
% Ui is the user, S is the server
played_by Ui
def=
local State : nat,
 IDi, PWi, B, Rpw, Ai, Ni, Zi, Mi, Ri, Ka, M1,M2, W, CID, C1, C2, SKu, E, Y, U, V: text
     const s1, s2, s3, s4, s5, s6, x,  alice_bob_u, bob_alice_s : protocol_id
     init State := 0
     transition
          1. State = 0  /\ Rcv(start) =|>
             State':= 1  /\ B':=new()
                    /\ Rpw' := H(IDi.PWi.B')
                    /\Ai' := xor(IDi, B')
% Send the registration request message
                    /\ Snd({Ai.Rpw'}_SKuisj)
                    /\ secret({IDi}, s1, {Ui,S})
                    /\ secret({PWi, B'}, s2, {Ui})
% Receive the smart card from the registration server S
     2. State = 1 /\ Rcv({xor(xor(H(E.x),xor(IDi, B')), H(IDi.PWi.B'))}_SKuisj) =|>
% Login phase
         State':= 2 /\B' := xor(H(IDi.PWi), xor(H(IDi.PWi), B'))
                /\Ri' := H(H(IDi.PWi).B'.xor(xor(H(E.x),xor(IDi, B')), H(IDi.PWi.B')))
                /\ W':=new()
                /\M1':= Mul(W'.P)
                /\ Snd({M1'.P}_SKuisj)
                /\secret({W'}, s3, {Ui})
% Authentication phase
% Receive the authentication request message
     3. State = 2 /\ Rcv({Mul(Y'.P).P}_SKuisj) =|>
        State' := 3 /\ Rpw' := H(IDi.PWi.B')
                /\ Ai' := xor(IDi, B')
          /\ Mi' := xor(xor(H(IDi.PWi.B'), xor(IDi, B')), xor(xor(H(E.x),Ai'), Rpw'))
          /\ Ka' := Mul(Mul(Y'.P).W)
          /\ CID' := xor(H(Ka'.Mi'), B')
          /\ C1' := H(CID'.Mi'.Ka'.B')
          /\ Snd({CID'.Ka'.C1'}_SKuisj)
          /\secret({IDi, PWi}, s4, {Ui})
          /\witness(Ui, S, alice_bob_u, U')
     4.   State = 3 /\Rcv({H(H(Mul(Mul(W'.P).Y').xor(xor(H(Mul(Mul(Y'.P).W). xor(xor(H(IDi.PWi.B'),
xor(IDi, B')), xor(xor(H(E.x),xor(IDi, B')), H(IDi.PWi.B')))), xor(H(IDi.PWi),  xor(H(IDi.PWi), B'))),
H(Mul(Mul(W'.P).Y'). H(E.x)))). xor(xor(H(Mul(Mul(Y'.P).W).xor(xor(H(IDi.PWi.B'),  xor(IDi, B')),
xor(xor(H(E.x),xor(IDi, B')), H(IDi.PWi.B')))), xor(H(IDi.PWi), xor(H(IDi.PWi), B'))),  H(Mul(Mul(W'.P).Y').
H(E.x))). H(E.x). Mul(Mul(W'.P).Y'))}_SKuisj) =|>
           State' := 4 /\ SKu' := H(Ka'.B'.Mi')
                    /\ C2' := H(SKu'.B'.Mi'.Ka')
                    /\request (S, Ui, bob_alice_s, V')
     end role
```

Figure 4.3 Role specification for the U_i of the proposed scheme

```
role bob (S, Ui : agent,
P : public_key,
SKuisj : symmetric_key,
H, Mul:hash_func,
Snd, Rcv : channel (dy))
 % Ui is the user; S is the server

played_by S def=
local State : nat,
IDi, PWi, Rpw, B, Ai, Ni, Zi, Mi, Ri, Kb, M1, M2, Y, CID, C1, C2, SKs, E, W, U, V: text
const s1, s2, s3, s4, s5, s6, x, alice_bob_u, bob_alice_s : protocol_id
init State := 0
transition
% Registration phase
% Receive the registration request message from the user
    1. State = 0 /\ Rcv({H(IDi.PWi.B').xor(IDi, B')}_SKuisj) =|>
       State':= 1 /\E' := new()
            /\ Mi' := H(E.x)
            /\Ni' := xor(xor(H(E.x),xor(IDi, B')), H(IDi.PWi.B'))
% Send the smart card to the user
            /\ Snd ({Ni'}_SKuisj)
            /\ secret({IDi}, s1, {Ui,S})
    2. State = 1 /\ Rcv({Mul(W'.P).P}_SKuisj) =|>
       State':= 2  /\ Y':=new()
            /\M2':= Mul(Y'.P)
            /\ Snd({M2'}_SKuisj)
            /\secret({Y'}, s5, {S})
% Login phase
% Receive the login request message
3. State = 2 /\ Rcv({H(xor(H(Mul(Mul(Y'.P).W).xor(xor(H(IDi.PWi.B'), xor(IDi, B')), xor(xor(H(E.x),xor(IDi, B')),
H(IDi.PWi.B')))), xor(H(IDi.PWi), xor(H(IDi.PWi), B'))). xor(xor(H(IDi.PWi.B'), xor(IDi, B')), xor(xor(H(E.x),xor(IDi,
B')), H(IDi.PWi.B'))). Mul(Mul(Y'.P).W). xor(H(IDi.PWi), xor(H(IDi.PWi), B'))). Mul(Mul(Y'.P).W).
xor(H(Mul(Mul(Y'.P).W).xor(xor(H(IDi.PWi.B'), xor(IDi, B')), xor(xor(H(E.x),xor(IDi, B')), H(IDi.PWi.B')))),
xor(H(IDi.PWi), xor(H(IDi.PWi), B')))}_SKuisj)=|>

State':= 3   /\ Kb' := Mul(Mul(W'.P).Y')
        /\ B' := xor(xor(H(Mul(Mul(Y'.P).W).xor(xor(H(IDi.PWi.B'), xor(IDi, B')), xor(xor(H(E.x),xor(IDi, B')),
H(IDi.PWi.B')))), xor(H(IDi.PWi), xor(H(IDi.PWi), B'))), H(Mul(Mul(W'.P).Y'). H(E.x)))
        /\ C1' := H(xor(H(Mul(Mul(Y'.P).W).xor(xor(H(IDi.PWi.B'), xor(IDi, B')), xor(xor(H(E.x),xor(IDi, B')),
H(IDi.PWi.B')))), xor(H(IDi.PWi), xor(H(IDi.PWi), B'))). H(E.x). Mul(Mul(W'.P).Y').
xor(xor(H(Mul(Mul(Y'.P).W).xor(xor(H(IDi.PWi.B'),  xor(IDi, B')), xor(xor(H(E.x),xor(IDi, B')), H(IDi.PWi.B')))),
xor(H(IDi.PWi), xor(H(IDi.PWi), B'))), H(Mul(Mul(W'.P)).Y'). H(E.x)))

        /\ SKs' := H(Mul(Mul(W'.P).Y').B'.H(E.x))
        /\ C2' := H(H(Mul(Mul(W'.P).Y').B'.H(E.x)).B'.H(E.x). Mul(Mul(W'.P).Y'))

% Send the authentication request message
        /\ Snd({C2'}_SKuisj)
        /\secret({SKs},s6, {S, Ui})
        /\request(Ui, S, alice_bob_u, U')
        /\witness(S, Ui, bob_alice_s, V')
    end role
```

Figure 4.4 Role specification for the S of the proposed scheme

```
role session (Ui, S: agent,
P : public_key, SKuisj : symmetric_key,
H, Mul:hash_func)
def=
local SI, SJ, RI, RJ: channel (dy)
composition
alice(Ui, S, P, SKuisj, H, Mul, SI, RI)
        /\ bob(S, Ui, P, SKuisj, H, Mul, SJ, RJ)
end role
```

Figure 4.5 Role specification for the session of the proposed scheme

```
role environment()
def=
const ui, s : agent,
p : public_key,
skuisj : symmetric_key,
h, mul: hash_func,
idi, pwi, rpw, b, ai, ni, zi, mi, ri, kb, m1, m2: text,
y, cid, c1, c2, sks, sku, e,w, u, v: text,
 s1, s2, s3, s4,s5,s6, alice_bob_u, bob_alice_s : protocol_id
intruder_knowledge = {ui,s,p,h}
composition
session(ui, s, p, skuisj, h,mul)
/\session(ui, s, p, skuisj, h,mul)
end role

goal
authentication_on alice_bob_u
authentication_on bob_alice_s
secrecy_of s1, s2, s3, s4, s5, s6
end goal
environment()
```

Figure 4.6 Role specification for the goal and environment of the proposed scheme

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL

C:\progra~1\SPAN\testsuite\results\replay_resist1.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 1.88s
  visitedNodes: 4 nodes
  depth: 2 plies
```

Figure 4.7 OFMC simulation result of proposed scheme

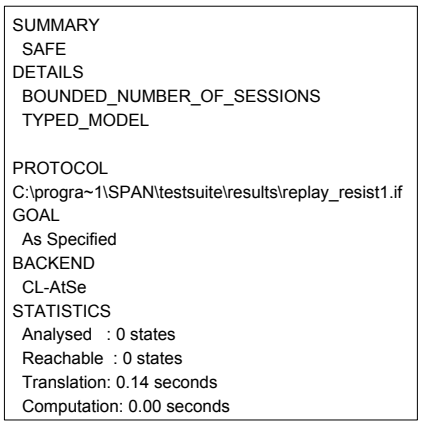

Figure 4.8 CLAtSe simulation result of proposed scheme

4.7 Functionality and performance analysis

4.7.1 Functionality analysis

This section presents the functionalities comparision of the proposed scheme with Wang et al. (2011), Khan et al. (2011), Chen et al. (2011), Wen and Li (2012), An (2013), Ding et al. (2012), Chang et al. (2014), Kumari et al. (2014), Truong et al. (2014), Wang and Wang (2016) schemes. The result of the functionalities comparision has been presented in the table 4.2. Here, there are mainly five functinalities are compared such as (C1) Resistance to replay attack without time synchronization, (C2) Session key security, (C3) User credentials privacy, (C4) Secure server secret key and (C5) Resistance to stolen smart card attack. From the comparison result it is clear that the proposed scheme achieves all the functionalities.

4.7.2 Performance analysis

In performance analysis mainly focused on the comparision of the computation and communication costs of the proposed scheme with Wang et al. (2011), Khan et al. (2011), Chen et al. (2011), Wen and Li (2012), An (2013), Ding et al. (2012), Chang et al. (2014), Kumari et al. (2014), Truong et al. (2014), Wang and Wang (2016) schemes. Computation cost comparison result is presented in Table 4.3 and Communication cost comparison results are given in Table 4.4.

In Table 4.3 H represents the one way hash function, ME denotes modular operation

Table 4.2 Functionality analysis

Scheme	**C1**	**C2**	**C3**	**C4**	**C5**
Wang et al. (2011)	N	Y	Y	Y	Y
Khan et al. (2011)	N	N	N	Y	N
Chen et al. (2011)	N	Y	N	N	Y
Wen and Li (2012)	N	N	Y	Y	N
An (2013)	N	Y	Y	N	N
Ding et al. (2012)	N	N	N	N	N
Chang et al. (2014)	N	N	N	N	N
Kumari et al. (2014)	N	Y	N	Y	Y
Truong et al. (2014)	N	N	N	N	Y
Wang and Wang (2016)	N	Y	Y	Y	Y
Proposed Scheme	Y	Y	Y	Y	Y

and ECC signifies the elliptic curve scalar multiplication. The registration phase of the proposed scheme takes $4H$, The login and authentication phase takes $12H + 4ECC$ to execute the scheme. In the table 4.3, many schemes designed using only hash functions. Those schemes does not adopts public key cryptography. But, Wang et al. said that the scheme which employs the public key cryptography has achieves user anonymity under the tamper resistance assumption. Therefore the proposed scheme adopts the public key cryptography. The proposed scheme also employs ECDH technique to resist the replay attack without time synchronization. Hence the overall computation cost of the proposed scheme is $16H + 4ECC$.

Table 4.3 Computational cost comparision

Schemes	**Computation cost**		**Total cost**	**Estimated Time (ms)**
	Registration phase	Login and Authentication phase		
Wang et al. (2011)	$2H$	$11H$	$13H$	0.884 ms
Khan et al. (2011)	$3H$	$10H$	$13H$	0.884 ms
Chen et al. (2011)	$3H$	$8H$	$11H$	0.748 ms
Wen and Li (2012)	$5H$	$21H$	$26H$	1.768 ms
An (2013)	$3H$	$8H + 4ME$	$11H + 4ME$	12.92 ms
Ding et al. (2012)	$9H$	$16H + 3ME$	$25H + 3ME$	10.829 ms
Chang et al. (2014)	$2H$	$15H$	$17H$	1.156 ms
Kumari et al. (2014)	$3H$	$15H$	$18H$	1.224 ms
Truong et al. (2014)	$4H$	$13H + 3ECC$	$17H + 3ECC$	8.659 ms
Wang and Wang (2016)	$4H + 1ME$	$15H + 6ME$	$19H + 7ME$	22.593 ms
Proposed scheme	$4H$	$12H + 4ECC$	$16H + 4ECC$	11.092 ms

Table 4.3 also presents the estimated execution time of the proposed scheme. The simulation result illustrated by Xie et al. (2017) has been taken to compute the exe-

cution time. According to Xie et al. (2017), simulation has been done using Miracl library. The environment used for simulation is, Windows 7 sp1 64-bit PC, Intel Core i5-3210M CPU of 2.5 GHz, 8GB RAM. According to this simulation, execution time of one hash operation takes 0.068 ms (millisecond), execution time of one block encryption/decryption is 0.56 ms, execution time of one modular exponentiation is 3.043 ms and execution time of one scalar multiplication on elliptic curve is 2.501 ms.

Estimated computation time to execute registration phase of the proposed scheme is 0.272ms. Similarly, estimated computation time to execute login and authentication phase is 10.82ms. Overall estimated time of the proposed scheme is 11.092 ms. From Table 4.3, Wang et al. (2011), Khan et al. (2011), Chen et al. (2011), Wen and Li (2012), Chang et al. (2014), Kumari et al. (2014) schemes are efficient compared to other since they are based on only hash functions. But, these schemes failed to achieve all functional requirements which are presented in the table 4.2. Similarly, Ding et al. (2012) and Truong et al. (2014) schemes also takes less computation cost than the proposed scheme. Even though usage of public key in the scheme they failed to achieve security goals illustrated in the motivation section. An (2013), Wang and Wang (2016) schemes takes more computation time when compared to the proposed scheme.

Table 4.4 Communication cost comparison

Schemes	**Communication cost(bits)**
Wang et al. (2011)	5*128 = 640
Khan et al. (2011)	6*128 = 768
Chen et al. (2011)	6*128 = 768
Wen and Li (2012)	9*128 = 1152
An (2013)	7*128 = 896
Ding et al. (2012)	6*128 = 768
Chang et al. (2014)	6*128 = 768
Kumari et al. (2014)	6*128 = 768
Truong et al. (2014)	7*128 = 896
Wang and Wang (2016)	7*128 = 896
Proposed scheme	7 * 128 = 896

The communication cost of the proposed scheme is calculated and compared with the related scheme. Table 4.4 lists the statistics of the communication cost. This includes the cost of the data communicating in the login and authentication phase of one

complete session. For consistency purpose, the length of the data communicated in the channel has fixed as 128 bits. The proposed scheme takes 7 * 128 = 896 bits as communication cost in login and authentication phase.

4.8 Conclusion

To eliminate replay attack, many researchers have used timestamp. But, the time synchronization has its own drawbacks. Few researchers have used nonce but they are prone to different attacks. Our aim was to propose a scheme, which does not use the timestamp to resist replay attack and also resists all other attacks. Hence, in this article, a novel remote user authentication scheme has been proposed. The proposed scheme overcomes security weaknesses and achieves all security goals of the remote user authentication. The proposed scheme employs the ECDH key exchange method to resist the replay attack without time synchronization. The scheme is developed with less number of hash functions to make the scheme more lightweight and easy for practical implementation. Communication and computation cost comparison shows the efficiency of the proposed scheme. Hence, it can be conclude that the proposed scheme is robust and easy to implement practically.

Chapter 5

CONCLUSION

The conclusion of the thesis first summarizes the contributions and highlights the roadmap for future research in the authentication security.

5.1 Contributions

This thesis primarily focuses on improvement of smart card based remote user authentication schemes and propose a novel remote user authentication scheme for the wireless environment. The research has achieved the following objectives: (1) Analyze the recently proposed remote user authentication schemes, identify the security flaws and propose a new scheme to resolve the identified flaws. (2) Propose a secure dynamic authentication scheme by including the public key cryptosystem and prove the security with formal and simulation proof. (3) Develop a novel remote user authentication scheme using the smart card in the wireless environment. The main contributions of this thesis are as follows:

Chapter 2 first reviews Wen and Li's and Ding et al.'s dynamic ID based remote user authentication schemes. It points out the security flaws in reviewed schemes. To overcome identified security weaknesses, an improved scheme has been proposed. This chapter also presents computation cost results, and it shows that the proposed scheme is more lightweight by having less number of hash functions.

Chapter 3 mainly focuses on the construction of the secure remote user authentication scheme by employing an efficient public key cryptosystem. Here, Troung et al.'s remote user authentication scheme has been reviewed. This scheme is designed using elliptic curve cryptography. In this scheme, few vulnerabilities are identified. To overcome the identified weaknesses, a new dynamic authentication scheme using elliptic curve cryptography has been proposed. This chapter also focuses on the security proofs. Formal proof of the proposed scheme has been given using BAN logic. Simulation has been done using the AVISPA tool. Furthermore, the computation efficiency

of the proposed scheme is also discussed and compared with other related schemes.

Chapter 4 proposes a novel remote user authentication scheme for the wireless environment. This chapter mainly focuses on eliminating the replay attack without the involvement of clock synchronization. In the literature study, it is observed that to avoid the replay attack, many remote user authentication schemes depending on the clock synchronization. But the clock synchronization has its disadvantages. These drawbacks will effects mainly on the practical implementation of the scheme. Also, the schemes which are independent of clock synchronization are vulnerable to replay attack. To eliminate this problem, The Elliptic Curve Diffie-Hellman (ECDH) key exchange algorithm has been used in the proposed scheme. The proposed scheme is also preventing privacy of user credentials and keeps the secrecy of the encryption and decryption keys. The security proof of the proposed scheme using BAN logic has been given and simulated the scheme in AVISPA tool. The computation efficiency of the proposed scheme is discussed and compared it with other related schemes.

5.2 Future Work

This section presents the few directions for future works. Several research directions are worth investigating as follows.

One of the future work includes extending the scheme illustrated in Chapter 4 into the distributed cloud environment. In the distributed cloud, presently Single-Sign-On (SSO) authentication mechanism is efficient for resource access. But, Tsai and Lo (2015) identified the time delay and third-party dependency problems in the Open ID based SSO authentication. To overcome these problems, Tseng et al. (2016), Tsai and Lo (2015), Odelu et al. (2017) and many more researchers proposed user authentication schemes. Still, there is no scheme which overcomes all the security weaknesses efficiently.

Another future direction includes proposing a novel user authentication scheme using the wireless sensor network (WSN) for agriculture monitoring. Ali et al. (2018) proposed an authentication scheme for agriculture monitoring using WSN. In this work, **?** proposed requirements set for secure authentication in WSN. It is necessary to reshape those requirements set and need to propose a new and novel remote user authentication scheme for agriculture monitoring.

The proposed schemes are suitable for only client-server architecture. Adopting the schemes in distributed cloud or multi-server environment is complicated. One more future direction is to extend the security of the proposed schemes into the multi-server environment and cloud environment.

Bibliography

Ali, R., Pal, A. K., Kumari, S., Karuppiah, M., and Conti, M. (2018). A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring. *Future Generation Computer Systems*, 84, 200–215.

An, Y. H. (2013). Security improvements of dynamic id based remote user authentication scheme with session key agreement. In *Advanced Communication Technology (ICACT), 2013 15th International Conference on*, 1072–1076. IEEE.

Burrows, M., Abadi, M., and Needham, R. M. (1989). A logic of authentication. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 426, 233–271. The Royal Society.

Chandrakar, P. and Om, H. (2018). An efficient two-factor remote user authentication and session key agreement scheme using rabin cryptosystem. *Arabian Journal for Science and Engineering*, 43(2), 661–673.

Chang, Y. F., Tai, W. L., and Chang, H. C. (2014). Untraceable dynamic identity based remote user authentication scheme with verifiable password update. *International Journal of Communication Systems*, 27(11), 3430–3440.

Chaudhry, S. A., Farash, M. S., Naqvi, H., Kumari, S., and Khan, M. K. (2015). An enhanced privacy preserving remote user authentication scheme with provable security. *Security and Communication Networks*, 8(18), 3782–3795.

Chen, B.-L., Kuo, W.-C., and Wuu, L.-C. (2014). Robust smart-card-based remote user password authentication scheme. *International Journal of Communication Systems*, 27(2), 377–389.

Chen, T. H., Hsiang, H. C., and Shih, W. K. (2011). Security enhancement on an improvement on two remote user authentication schemes using smart cards. *Future Generation Computer Systems*, 27(4), 377–380.

Das, M. L., Saxena, A., and Gulati, V. P. (2004). A dynamic id based remote user authentication scheme. *Consumer Electronics, IEEE Transactions on*, 50(2), 629–631.

Ding, W. et al. (2012). Cryptanalysis and security enhancement of a remote user authentication scheme using smart cards. *The Journal of China Universities of Posts and Telecommunications*, 19(5), 104–114.

Dolev, D. and Yao, A. (1983). On the security of public key protocols. *IEEE Transactions on information theory*, 29(2), 198–208.

Guttman, B. and Roback, E. A. (1995). *An introduction to computer security: the NIST handbook*. DIANE Publishing.

Hsu, C. L. (2004). Security of chien et al.'s remote user authentication scheme using smart cards. *Computer Standards & Interfaces*, 26(3), 167–169.

Hwang, M. S. and Li, L. H. (2000). A new remote user authentication scheme using smart cards. *IEEE Transactions on Consumer Electronics*, 46(1), 28–30.

Hwang, T., Chen, Y., and Laih, C. S. (1990). Non interactive password authentications without password tables. In *Computer and Communication Systems, 1990. IEEE TENCON'90., 1990 IEEE Region 10 Conference on*, 429–431. IEEE.

Khan, M. K., Kim, S. K., and Alghathbar, K. (2011). Cryptanalysis and security enhancement of a âĂŸmore efficient & secure dynamic id based remote user authentication schemeâĂŹ. *Computer Communications*, 34(3), 305–309.

Kizza, J. M. (2009). *A guide to computer network security*. Springer.

Kumari, S. and Khan, M. K. (2014). Cryptanalysis and improvement of a robust smart-card-based remote user password authentication scheme. *International Journal of Communication Systems*, 27(12), 3939–3955.

Kumari, S., Khan, M. K., and Li, X. (2014). An improved remote user authentication scheme with key agreement. *Computers & Electrical Engineering*, 40(6), 1997–2012.

Lamport, L. (1981). Password authentication with insecure communication. *Communications of the ACM*, 24(11), 770–772.

Lee, C.-C., Lin, T.-H., and Chang, R.-X. (2011). A secure dynamic id based remote user authentication scheme for multi-server environment using smart cards. *Expert Systems with Applications*, 38(11), 13863–13870.

Lee, Y.-C., Hsieh, Y.-C., Lee, P.-J., and You, P.-S. (2014). Improvement of the elgamal based remote authentication scheme using smart cards. *Journal of applied research and technology*, 12(6), 1063–1072.

Li, X., Ibrahim, M. H., Kumari, S., Sangaiah, A. K., Gupta, V., and Choo, K.-K. R. (2017). Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks. *Computer Networks*, 129, 429–443.

Li, X., Ma, J., Wang, W., Xiong, Y., and Zhang, J. (2013a). A novel smart card and dynamic id based remote user authentication scheme for multi-server environments. *Mathematical and Computer Modelling*, 58(1-2), 85–95.

Li, X., Niu, J., Khan, M. K., and Liao, J. (2013b). An enhanced smart card based remote user password authentication scheme. *Journal of Network and Computer Applications*, 36(5), 1365–1371.

Li, X., Niu, J., Liao, J., and Liang, W. (2015). Cryptanalysis of a dynamic identity-based remote user authentication scheme with verifiable password update. *International Journal of Communication Systems*, 28(2), 374–382.

Liao, I. E., Lee, C. C., and Hwang, M. S. (2005). Security enhancement for a dynamic id based remote user authentication scheme. In *Next Generation Web Services Practices, 2005. NWeSP 2005. International Conference on*, 4–pp. IEEE.

Liao, I. E., Lee, C. C., and Hwang, M. S. (2006). A password authentication scheme over insecure networks. *Journal of Computer and System Sciences*, 72(4), 727–740.

Madhusudhan, R. and Mittal, R. (2012). Dynamic id based remote user password authentication schemes using smart cards: A review. *Journal of Network and Computer Applications*, 35(4), 1235–1248.

Maitra, T., Obaidat, M. S., Amin, R., Islam, S., Chaudhry, S. A., and Giri, D. (2017). A robust elgamal-based password-authentication protocol using smart card for client-server communication. *International Journal of Communication Systems*, 30(11).

Nikooghadam, M., Jahantigh, R., and Arshad, H. (2017). A lightweight authentication and key agreement protocol preserving user anonymity. *Multimedia Tools and Applications*, 76(11), 13401–13423.

Odelu, V., Das, A. K., and Goswami, A. (2015). A secure biometrics-based multi-server authentication protocol using smart cards. *IEEE Transactions on Information Forensics and Security*, 10(9), 1953–1966.

Odelu, V., Das, A. K., Kumari, S., Huang, X., and Wazid, M. (2017). Provably secure authenticated key agreement scheme for distributed mobile cloud computing services. *Future Generation Computer Systems*, 68, 74–88.

Shamir, A. (1984). Identity based cryptosystems and signature schemes. In *Advances in cryptology*, 47–53. Springer.

Song, R. (2010). Advanced smart card based password authentication protocol. *Computer Standards & Interfaces*, 32(5), 321–325.

Sood, S. K., Sarje, A. K., and Singh, K. (2010). An improvement of liou et al. s authentication scheme using smart cards. *International Journal of Computer Applications*, 1(8), 16–23.

Steiner, J. G., Neuman, B. C., and Schiller, J. I. (1988). Kerberos: An authentication service for open network systems. In *Usenix Winter*, 191–202.

Sun, H. M. (2000). An efficient remote use authentication scheme using smart cards. *Consumer Electronics, IEEE Transactions on*, 46(4), 958–961.

Truong, T. T., Tran, M. T., and Duong, A. D. (2014). Enhanced dynamic authentication scheme (edas). *Information Systems Frontiers*, 16(1), 113–127.

Tsai, C. S., Lee, C. C., and Hwang, M. S. (2006). Password authentication schemes: Current status and key issues. *IJ Network Security*, 3(2), 101–115.

Tsai, J.-L. and Lo, N.-W. (2015). A privacy-aware authentication scheme for distributed mobile cloud computing services. *IEEE systems journal*, 9(3), 805–815.

Tseng, Y.-M., Huang, S.-S., Tsai, T.-T., and Ke, J.-H. (2016). List-free id-based mutual authentication and key agreement protocol for multiserver architectures. *IEEE Transactions on Emerging Topics in Computing*, 4(1), 102–112.

Uludag, U., Pankanti, S., Prabhakar, S., and Jain, A. K. (2004). Biometric cryptosystems: issues and challenges. *Proceedings of the IEEE*, 92(6), 948–960.

Wang, C., Wang, D., Xu, G., and Guo, Y. A lightweight password-based authentication protocol using smart card. *International Journal of Communication Systems*.

Wang, D., Gu, Q., Cheng, H., and Wang, P. (2016). The request for better measurement: A comparative evaluation of two-factor authentication schemes. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, 475–486. ACM.

Wang, D., Wang, N., Wang, P., and Qing, S. (2015). Preserving privacy for free: efficient and provably secure two-factor authentication scheme with user anonymity. *Information Sciences*, 321, 162–178.

Wang, D. and Wang, P. (2016). Two birds with one stone: Two factor authentication with security beyond conventional bound. *IEEE Transactions on Dependable and Secure Computing*.

Wang, R. C., Juang, W. S., and Lei, C. L. (2011). Robust authentication and key agreement scheme preserving the privacy of secret key. *Computer Communications*, 34(3), 274–280.

Wang, Y. Y., Liu, J. y., Xiao, F. x., and Dan, J. (2009). A more efficient and secure dynamic id based remote user authentication scheme. *Computer communications*, 32(4), 583–585.

Wayman, J., Jain, A., Maltoni, D., and Maio, D. (2005). *An introduction to biometric authentication systems*. Springer.

Wen, F. and Li, X. (2012). An improved dynamic id based remote user authentication with key agreement scheme. *Computers & Electrical Engineering*, 38(2), 381–387.

Xiao, Q. (2005). Security issues in biometric authentication. In *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC*, 8–13. IEEE.

Xie, Q., Wong, D. S., Wang, G., Tan, X., Chen, K., and Fang, L. (2017). Provably secure dynamic id-based anonymous two-factor authenticated key exchange protocol with extended security model. *IEEE Transactions on Information Forensics and Security*, 12(6), 1382–1392.

Xu, J., Zhu, W. T., and Feng, D. G. (2009). An improved smart card based password authentication scheme with provable security. *Computer Standards & Interfaces*, 31(4), 723–728.

Yang, G., Wong, D. S., Wang, H., and Deng, X. (2006). Formal analysis and systematic construction of two factor authentication scheme (short paper). In *International Conference on Information and Communications Security*, 82–91. Springer.

Yang, W. H. and Shieh, S. P. (1999). Password authentication schemes with smart cards. *Computers & Security*, 18(8), 727–733.

Yeh, K. H., Su, C., Lo, N. W., Li, Y., and Hung, Y. X. (2010). Two robust remote user authentication protocols using smart cards. *Journal of Systems and Software*, 83(12), 2556–2565.

Yoon, E. J., Ryu, E. K., and Yoo, K. Y. (2005). An improvement of hwang–lee–tang's simple remote user authentication scheme. *Computers & Security*, 24(1), 50–56.

Yoon, E. J. and Yoo, K. Y. (2006). Improving the dynamic id based remote mutual authentication scheme. In *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, 499–507. Springer.

Zhu, J. and Ma, J. (2004). A new authentication scheme with anonymity for wireless environments. *IEEE Transactions on Consumer Electronics*, 50(1), 231–235.

LIST OF PUBLICATIONS:

Journal papers

- Madhusudhan, R., and Manjunath Hegde. "Security bound enhancement of remote user authentication using smart card", *Journal of Information Security and Applications*, 36 (2017): 59-68 (Scopus and ESCI Journal).

- Madhusudhan R., Manjunath Hegde, I. Memon. "A Secure and Enhanced Elliptic Curve Cryptography Based Dynamic Authentication Scheme Using Smart Card" *International Journal of Communication Systems* 2018;e3701.
  DOI: https://doi.org/10.1002/dac.3701. (SCIE)

Conference papers / Presentation

- Madhusudhan, R., and Manjunath Hegde. *Cryptanalysis and Improvement of Remote User Authentication Scheme Using Smart Card* published in IEEE International Conference on Computer and Communication Engineering (ICCCE), 2016, pp. 84-89.

BIO-DATA

Name	:	Manjunath Vishweshwar Hegde
Email Id	:	manjuchavatti@gmail.com
Mobile	:	+91-8762329150
Date of Birth	:	July 30, 1990
Address	:	S/o. Vishweshwar Hegde,
		# 32, at and post chavatti,
		Yellapur Taluk,
		Uttara Kannada - 581347.
		Karnataka, India.

Educational Qualifications:

Degree	Year of Passing	University
B.Sc.	2012	Mangaluru University, Mangaluru.
M.Sc.	2014	Mangaluru University, Mangaluru.