A Project Report on

Some Studies on Remote User Authentication Scheme

Submitted to the Department of Information Technology

**For the partial fulfilment of the degree of B.Tech. in**

**Information Technology**

by

**Kishlay Verma** Registration number: 510814011 4th year

**Suraj Kumar** Registration number: 510814004 4th year

Under the supervision of

**Shyamalendu Kandar**



Department of Information Technology

INDIAN INSTITUTE OF ENGINEERING SCIENCE AND

TECHNOLOGY, SHIBPUR

*December 2017*

****

**Department of Information Technology**

**Indian Institute of Engineering Science and Technology,**

**Shibpur**

**CERTIFICATE**

This is to certify that the work presented in this report entitled “Some Studies on Remote User Authentication Scheme”, submitted by Kishlay Verma having the examination roll number 510814011, and Suraj Kumar, having the examination roll number 510814004, has been carried out under my supervision for the partial fulfilment of the degree of Bachelor of Technology in Information Technology during the session 2017-18 in the Department of Information Technology, Indian Institute of Engineering Science and Technology, Shibpur.

—————————————– —————————————–

SHYAMALENDU KANDAR DR. ARINDAM BISWAS

Assistant Professor Head of the Department

Department of Information Technology Department of Information Technology

Indian Institute of Engineering Science Indian Institute of Engineering Science

and Technology, Shibpur and Technology, Shibpur

—————————————– Date:

Dean (Academic)

Indian Institute of Engineering Science

and Technology, Shibpur

**Acknowledgements**

It is our genuine pleasure to express a deep sense of thanks and gratitude to our professor and project mentor Shyamalendu Kandar for granting us the opportunity to work on the topic “Some Studies on Remote User Authentication Scheme”. This work might not have reached this intermediate level without his valuable guidance, input, encouragement, wholehearted cooperation and constructive motivation throughout the duration of the work. We take this opportunity to thank all our teachers who have directly or indirectly provided us their guidelines to complete this project.

Date: —————————————–

**KISHLAY VERMA**

**SURAJ KUMAR**

Department of Information Technology

Indian Institute of Engineering Science

And Technology, Shibpur

**Abstract**

Remote user authentication is a mechanism in which the remote server veriﬁes the authenticity of a user over an insecure communication channel. Password based authentication schemes have been widely used to verify the authenticity of remote users, as password authentication is one of the simplest and the most convenient authentication mechanism over insecure networks. There have been ample of remote user authentication schemes published in the literature and each published schemes have its own merits and demerits. Recently, many schemes proposed are based on the one-way hash function. In this work, we have studied and implemented a scheme that uses the combination of a user’s biometrics and password to design a remote user authentication scheme that enhances the level of the security. In addition, we have also deﬁned all the security requirements and the goals that an ideal password authentication schemes should satisfy.

i

**Contents**

**1. INTRODUCTION 1**

**2. RELATED WORKS 2**

**3. PRELIMINARIES 4**

3.1. User requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

3.2. Different types of Attacks. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

3.3. Security requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

3.4. Advantages of using biometrics . . . . . . . . . . . . . . . . . . . . . . . . . 6

3.5. The features of hash functions . . . . . . . . . . . . . . . . . . . . . . . . . . 6

**4. IMPLEMENTED APPROACH 7**

4.1 Server registration procedure . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

4.2 User registration procedure. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

4.3 Login procedure. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

4.4 Authentication procedure . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

**5. ANALYSIS 13**

**6. CONCLUSIONS 15**

**7**. **REFRENCES 16**

ii

**1. INTRODUCTION**

In recent years, wireless communications and Internet technologies have undergone rapid development and many people now use mobile devices (e.g., mobile phones, notebooks etc.) at any time and from anywhere to access all kinds of application services from the Internet, such as Web-browsing, video conferencing, and multimedia applications. Other services include Remote Login Systems, Automated Teller Machines (ATM’s), and Database Management Systems, etc. However, using these services over the internet calls for an authentication mechanism to protect the valid user from different kinds of attacks i.e. Denial of Service Attack (Dos), Password Guessing Attack, Replay Attack, Stolen-Veriﬁer Attack, Insider Attack and others.

Remote user authentication scheme is one of the simplest and most convenient authentication mechanisms for insecure networks. It provides the legal users to use the resources of the remote system. Different techniques have been proposed in the recent years. These techniques includes (i) *2-factor authentication techniques* in which each user should have an identity and password (IDi, PWi) to access these resources, or (ii) *3-factor authentication techniques* in which each user should have an identity and password (IDi, PWi) along with a system generated OTP, or (iv) the ongoing technique i.e. *4-factor authentication techniques* in which each user should have an identity and password (IDi, PWi), OTP and User’s Biometric details.

Traditionally, the identification entities are maintained by the remote system and when a user wants to login to a remote server, he simply submits his (IDi, PWi) and other identification entities (like OTP in 3-factor authentication and OTP and Biometric Details in 4-factor authentication) to the server. On receiving the login message, the remote server compares the submitted entities with stored one in identification table. If match found, user will be granted to access the server resources. Due to their eﬃciency and one-way property, one way hash functions have been used and as the basis on which more and more cryptosystems including password authentication systems, are being deployed.

In this work, we have implemented an anonymous multi-server authenticating key agreement scheme based on trust computing using smart cards, password, and biometrics. Smart Cards have embedded integrated circuits that provide personal identification, authentication, data storage, and application processing. They are designed in a way that provide strong security authentication for in-memory information within organizations.

Our scheme not only is a lightweight authentication scheme which only uses the nonce and a hash function but also satisfies various security properties.

1

**2. RELATED WORKS**

In 1981, Lamport[3] proposed a novel password authentication scheme using cryptography hash function. However, high hash overhead and the necessity for password resetting decrease its suitability for practical use. Since then, many improved password authentication schemes have been proposed.

One of the common feature of these schemes is that the server has to securely store a veriﬁcation table, which contains the veriﬁers of user’s passwords. If the veriﬁcation table is stolen by an adversary, the system may be partially or totally broken. To resist such a stolen veriﬁer attack, Hwang et al in 1990[4], proposed a non-interactive password authentication scheme and its enhanced version, which additionally uses smart cards. In Hwang et al scheme, the server requires neither storing the veriﬁers of user’s passwords nor keeping any secret of the user.

In 2000, Hwang and Li[5] proposed a veriﬁer-free password authentication scheme using smart cards based public-key technique. However, Hwang-Li’s[5] scheme doesn’t allow users to freely choose and change their passwords. Furthermore, Hwang-Li’s[5] scheme had found to be vulnerable to impersonation attack. To improve their eﬃciency, Sun[6] proposed a light-weight veriﬁer-free password authentication scheme using smart cards based on cryptographic hash functions. The major drawbacks of Sun’s[6] scheme was that the password is not easily memorisable and the user can’t freely choose/change his password.

Later, in 2002 H.Y.Chien[7] pointed out that Sun’s[6] scheme achieves unilateral user authentication and he also proposed a protocol to achieve mutual authentication. In addition, the user can freely choose his password and the smart card not containing user’s IDi can avoid the risk of IDi-theft or impersonation. Unfortunately, Chien’et al[7] scheme can’t withstand a parallel session attack. Further Ku’s et al[8] pointed out the Chien et al[7] scheme is vulnerable to reﬂection attack, insider attack, guessing attack and is not repairable. However, Yoon et al[9] showed that Ku et al[8] scheme was susceptible to parallel session attack and was insecure for changing the user’s password, and also proposed an enhancement to Ku et al’s[8] scheme to overcome such problems. Due to the power constraints of smart cards, the cost of implementation should be low as the lower the cost, the great chance of success in practical realization.

2

Among those smart card based schemes, Ku et al’s[8] and Yoon et al’s[9] schemes require only several hash operations instead of the costly modular exponentiations. Therefore, their schemes exhibits great application potentiality in smart card ﬁeld, regardless of their security. In continuation process 2005, Yoon et al pointed out Lee et al scheme is also vulnerable to some insidious attacks, reﬂection attack, stolen veriﬁer attack, parallel session attack, replay attack, etc.

To remedy these pitfalls, In 2007, X-M Wang pointed out that Ku et al’s[8] and Yoon et al’s[9] scheme are still vulnerable to the guessing attack, forgery attack and denial of service attack. As a result, only requiring few additional hash operations, X-M-Wang scheme can withstand the previously proposed attacks. In addition, wrong passwords input by the users can be deducted immediately and session key is provided after authentication phase. The computational cost and eﬃciency of the improved scheme are encouraging for the practical implementation in the resource-constraints environment.

3

**3. PRELIMINARIES**

This section describes the common user requirements, the security requirements of the system, the advantage of using biometrics, and the feature of the hash function.

**3.1. User requirements**

Given that the designed authentication scheme should be user-friendly, the following user requirements need to be considered.

1. *Simple and secure password choice and modification*: The system allows users to choose and changing their passwords easily and securely. In other words, the user can change the password without the help of any third trusted party after assuring the legality of cardholder.
2. *Single registration*: The user only needs to register with the registration centre once and then can access different application servers. Moreover, the single registration can reduce the overhead of the registration centre and the network.
3. *Anonymity*: The privacy of the user has attracted increasing attention from both industry and academia. Therefore, anonymous authentication involves verifying that a user does not use the real identity to execute the authentication procedure.

**3.2. Different types of Attacks**

1. *Denial of Service Attack:* This attack rejects all or speciﬁc users by means of an oﬀensive action on the server or by means of a falsiﬁcation of user’s password-veriﬁer. In this attack, an attacker can inconvenience the user but cannot imitate the user.
2. *Password Guessing Attack:* Most passwords have such low entropy that it is vulnerable to password guessing attack, where an attacker intercepts authentication messages and store it locally and then attempts to use a guessed password to verify the correctness of his guess using these authentication messages.

4

1. *Replay Attack:* Having intercepted previous communications, an attacker can impersonate as the legal user and login to the system. The attacker can replay the intercepted messages. An attack in which a valid data transmission is maliciously or fraudulently repeated either by the originator or by an adversary who intercepts the data and retransmits it possibly as part of a masquerade attack.
2. *Stolen-Veriﬁer Attack:* In most of the application, the server stores hashed passwords instead of clear text passwords. In the stolen veriﬁer attack, an adversary who steals the password veriﬁer (e.g. hashed password) from the server can use it directly to masquerade as a legitimate user during the user authentication phase.
3. *Insider Attack:* An insider attack is intentional misuse by individuals who are authorized to use the servers and the networks. Insider of the server can perform an oﬀ-line guessing attack to obtain password. If succeeds, the insider of the server can try to use password to impersonate users to login other servers employing normal password authentication methods.

**3.3. Security requirements**

Since a remote user authentication scheme is susceptible to attack from adversaries, our objective is to design a scheme that is robust enough to resist such attacks. Based on related studies, we define the following key requirements for securing authentication.

1. *Mutual authentication*: A mutual authentication process is required insofar as the server needs to verify that the user is a legal one, and the user needs to ensure that the server is not a forged one.

1. *No verification table*: In most applications, the registration centre stores the password table of the user resulting in the stolen-verifier attack, and as such, the designed scheme needs to avoid maintaining the password verification table of the user.
2. *Integrity*: The message integrity means that data cannot be modified without detection.

5

1. *Session key agreement*: After the authentication procedure, the session key is generated between the user and the server to provide a secure communication, and it can achieve for-ward secrecy.

**3.4. Advantage of using biometrics**

The weakness of a secret key is its low value of entropy, which can be guessed or cracked easily. For example, there is no way to prevent the attacker from impersonating the user if both the user’s smart card and password were stolen. For this reason, many schemes guarantee the system security when either the smart card or his password is stolen, but not both. On the other hand, a strong secret key which combines passwords with biometrics and smart cards (called three-factor security) has the value of high entropy which cannot be guessed easily. Moreover, the main feature of the biometric is uniqueness in that everyone has a different biometric, and it is difficult for the user’s biometric to be stolen because only the user inputs his biometric into his own smart card.

**3.5. The feature of hash function**

The security property of the proposed scheme is based on a collision-free one-way hash function, such as MD5, SHA, and RIPEMD. The bit size is longer, and the system is more robust. For a one-way hash function h (), when the value of x is given, it is easy to compute h(x); however, if the value of h(x) is given, computing x is very difficult or it incurs a high computational cost.

6

**4. IMPLEMENTED APPROACH**

This section describes the proposed anonymous multi-server authenticated key agreement scheme which involves five procedures: server registration, user registration, login, authentication, and change password. All notations are summarized in Table 1.

**4.1. Server registration procedure**

The application server sends the RC a join message if it would like to become an authorized server. Then, the RC replies with the key PSK to the server via the some Protocol. Afterwards, the authorized server uses this key (i.e., PSK) to facilitate the user’s authentication procedure.

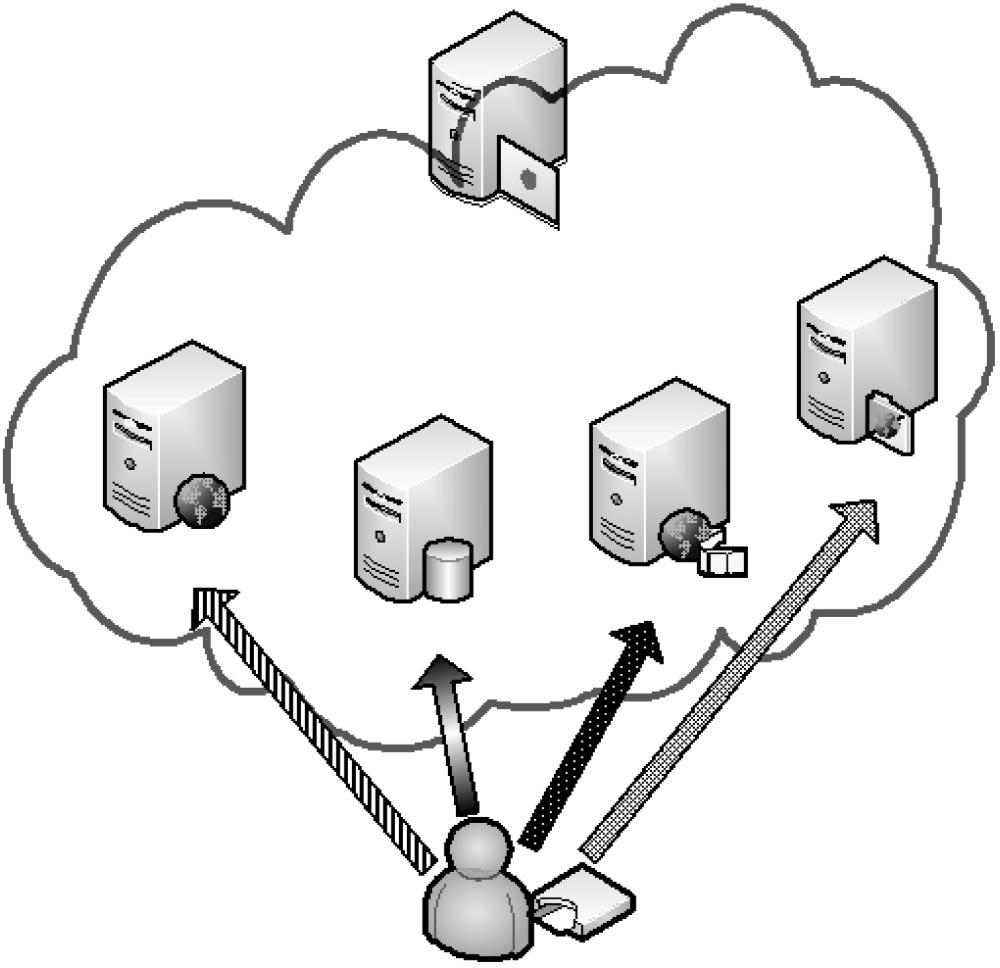


FIG 1.The User accesses the multiple application Server

7

**4.2. User registration procedure**

Initially, every user needs to perform the user registration procedure with the registration centre via a secure channel. Moreover, we assume that the authorized application servers are trusted according to the trust computing and

that the PSK cannot be extracted from the RC and application servers. Fig. 2 depicts the user registration procedure. The steps of the procedure are described as follows.

**Table 1**

*Notations*

-----------------------------------------------------------------------------------------------

X A secret value of the registration centre

RC The registration centre

IDi The public identification of user i

SIDi The public identification of server j

PWi The password of user i

BIOi The biometrics information of user i

AIDi  The anonymous identification of user i

h() A one-way collision-resistant hash function

Ni A random number

PSK A secure pre-shared key among authorized application servers and

the registration centre

⊕ The bitwise XOR operator

|| The string concatenation operator

X -> Y User X sends a message to user Y through a secure channel

X -> Y User X sends a message to user Y through a common channel

------------------------------------------------------------------------------------------------

*Step 1:* User -> RC: The user sends his registration information (i.e., his identification IDi and h(PWi ⊕BIOi)) to the RC via a secure channel.

*Step 2:* After receiving the information, the RC calculates the authentication parameters of the user as follows:

Ai = h(IDi ||x),

Bi = h2(IDi ||x) = h(Ai),

Ci = h(PWi ⊕ BIOi) ⊕ Bi, and

Di = PSK ⊕ Ai.

8

*Step 3:* RC -> User: The RC stores these authentication parameters { IDi, Bi, Ci, Di, h()}.

RC

Useri

( 1) IDi , h(PWi⊕BIOi) (2)

1. Ai=h(IDi|| x)

2. Bi=h2(IDi||x)=h(Ai)

3. Ci=h(PWi⊕BIOi)⊕Bi

Smartcard

(3) IDi ,Bi ,Ci , Di

4. Di=PSK⊕Ai

FIG 2. The User Registration Procedure

Here the RC does not obtain the user’s verification information (e.g., the password and biometrics information). Therefore, we can prevent the possibility of a *stolen-verifier and insider attacks*. In addition, the registered user cannot fabricate a valid user successfully when the user obtains these parameters (i.e., IDi, Bi, Ci, Di, h()). This is because the user does not know the secret value of the RC (i.e., x) and the PSK. Here, we also maintain the assumption that the biometric matching is exact matching.

9

**4.3. Login procedure**

The login procedure is the first checkpoint. The smart card detects an error event immediately if the user is not authorized to gain access (i.e., the user keys in the wrong identification, password, or biometrics information). Fig. 3 shows the steps of the lo-gin procedure.

*Step 1:* At the application server, the user inserts his IDi and PWi. Then, he scans his biometric information (e.g., fingerprint or user input) BIOi at the sensor.

*Step 2:* The smart card checks the IDi and then verifies whether h(PWi ⊕ BIOi) ⊕ Ci is equal to Bi. If the information is verified, then the smart card generates a nonce N1, calculates the message M1 as h(Bi) ⊕ N1, computes the alias AIDi as h(N1) ⊕ IDi, and generates the message M2 as h(N1|| AIDi ||Di), where Bi and Ci are obtained from the user registration procedure.

Smart card

Useri

1. IDi ,PWi , BIOi  (2)
2. Check IDi

2. Check h(PWi⊕Bi) ⊕Ci=Bi

3. Generate N1

4. M1= h(Bi)⊕N1

5. AIDi=h(N1)⊕IDi

6. M2= h(N1|| AIDi|| Di)

FIG 3. The Login Procedure

10

**4.4. Authentication procedure**

The smart card sends the server an authentication message after the user finishes the login procedure. Note that the smart card never uses the real identity (i.e., IDi) to perform the authentication procedure. Fig. 4 shows the steps of the authentication procedure.

*Step 1:* Smart Card -> Server: The smart card sends the server an authentication message (i.e., AIDi, M1, M2, Di), where Di is obtained from the user registration procedure.

*Step 2:* The server verifies the user: On receipt of the authentication request (i.e., AIDi, M1, M2, Di), the RC uses a secure pre-shared key (i.e., PSK) to obtain Ai (i.e., Ai = Di ⊕ PSK). The RC retrieves the value of N1 (i.e., N1 = M1 ⊕ h2(Ai)) and then checks whether h(N1|| AIDi ||Di) is equal to M2. The server rejects this authentication request and terminates this session if the result is not equal. This is because the authentication message has been modified. Next, the RC generates a random number N2 and calculates a session key SKij as h(N1||N2). Finally, the RC computes the authentication reply message (i.e., M3, M4), where M3 as N2 ⊕ h2(N1) and M4 as h(SIDj||N2).

*Step 3:* Server -> Smart Card: The server sends back the authentication reply message (i.e., SIDj, M3, M4) to the smart card.

*Step 4:* The smart card verifies the RC: The smart card computes the value of h2(N1), retrieves the random number N2 (i.e., N2 = M3 ⊕ h2(N1)), and checks whether h(SIDj||N2) is equal to M4. If the values are equal, the smart card computes the session key (i.e., SKij = h(N1||N2)).

*Step 5:* Smart Card -> Server: The smart card sends the message (i.e., SKij ⊕ h(N )) to the RC.

*Step 6:* The server uses the session key SKij to retrieve the value (i.e., h(N2)), and then it checks this value to prevent an invalid user from executing the replay attack.

11

Smart card

Server

1. AIDi ,M1,M2 (2)
2. Ai=Di⊕PSK
3. N1=M1⊕h2(Ai)
4. Check h(N1||AIDi||Di)=M2
5. Generate N2
6. SKij=h(N1||N2)
7. M3=N2⊕h2(N1)
8. M4=h(SIDj||N2)

(4) (3)SIDj ,M3, M4

1. Compute h2(N1)

2. N2=M3⊕h2(N1)

1. Check h(SIDj||N2)=M4
2. SKij=h(N1||N2)

(5)SKij ⊕ h(N2) (6) check h(N2) ⊕SKij

FIG 4. The Authentication Procedure

12

**5. ANALYSIS**

1. ***No verification table:*** The registration centre and application servers do not store the password table and the biometrics database of the user. Therefore, even if an intruding adversary accesses the database of the RC, he still cannot obtain the authentication information of users.
2. ***Anonymity:*** Under the proposed scheme, the original identity of a user is always converted into an alias that is based on a random number (i.e., AIDi = h(N1)⊕IDi). Therefore, an adversary cannot determine the original identity of the user without knowing the random number N1 chosen by the smart card. In addition, the unauthorized server cannot decrypt the user’s authentication message successfully since it does not obtain the PSK. As a result, it cannot retrieve the user’s real identity, i.e., the unauthorized server cannot get N1 because it cannot retrieve Ai from the communication messages. Our anonymity mechanism is a dynamic identification process.
3. ***Mutual authentication:*** A mutual authentication process is required. The server needs to verify that the user is a legal one, and the user needs to ensure that the server is not a forged one. In the authentication procedure, Step 2 shows that the server authenticates the user, and Step 4 shows that the user authenticates the server. If the attacker intercepts the messages and wants to forge a valid server/user, it must generate a valid reply message to the user/server. However, the attacker cannot compute a valid message because he does not know the secure key (i.e., PSK) and the random number (i.e., N1 and N2).
4. ***Resistance to replay attacks:*** To protect the proposed scheme from replay attacks, we add a random number into the message. Hence, if an adversary intercepts the previous authentication message (i.e., AIDi, M1, M2, Di) and tried to impersonate the valid user by immediately replaying the message, the server would obviously reject the request because the invalid random number (i.e., N1) will be detected in Step 2 of the authentication procedure. Moreover, the user also checks the random number which is sent from the server to prevent the replay attack.

13

1. ***Session key agreement:*** We only use one round trip between the user and the server to generate the session key. Then, we can use the session key to encrypt the following packets to ensure the communications are confidential. Moreover, the session key is generated by the random number and a one-way hash function (i.e., SKij=h(N1||N2)). Hence, this session key is different in each session, and it is difficult for the adversary to derive the session key from the intercepted messages.
2. ***Resistance to off-line password guessing attacks:*** Since PWi, PSK, BIOi, and x are unknown to the adversary, the sys-tem is secure even if the stored information Bi, Ci, and Di are revealed. The password and the biometrics of the user (i.e., PWi and BIOi) are protected by the one-way hash function h(), which means that the adversary cannot check whether or not each of his guessed password (i.e., PWi) is correct (i.e., h(PWi ⊕BIOi) = Ci⊕ Bi). Moreover, it is impossible for any two people to have the same biometrics template, such as a fingerprint. Therefore, our scheme can defeat the off-line password guessing attack.
3. ***Resistance to insider attacks:*** In the registration procedure, the user sends RC a registration message, which is h(PWi ⊕ BIOi) instead of PWi and BIOi. The RC cannot obtain the user’s password and biometrics directly. Moreover, it is difficult for the attacker to retrieve the user’s information since the hash function has the one-way property. As such, the proposed scheme can resist insider attacks.

14

**6. CONCLUSIONS**

In this paper, we propose a secure remote user authentication scheme which supports single-server environment to give an overview of the connection and information transfer between RC and application server. It possesses high security properties to protect the valid user against attacks with minimal computational cost. Our scheme is suitable for real-life applications because it is a true lightweight authentication scheme that only uses the hash function. Moreover, our scheme satisfies the following security properties: anonymity, no verification tables, mutual authentication, resistance to forgery attacks, no clock synchronization problem, resistance to modification attacks, resistance to replay attacks, fast error detection, resistance to off-line guessing attacks, resistance to insider attacks, simple and secure choice and change of passwords, biometric template protection, and session key agreement.

In the future, we will propose a similar scheme to prove that our authentication mechanism is secure and can be implemented in a multi-server and multi-user environment and will discuss the biometric matching issue in detail. Moreover, we will build a biometric-based authentication testbed and extend our scheme for other services.

15

**7. REFERENCES**

[1] Ming-Chin Chuang, Meng Chang Chen. An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometric (2014).

[2] Stack Overflow. Working and implementation of one-way hash functions work. [*https://stackoverflow.com/questions/2112685/how-do-one-way-hash-functions-work*](https://stackoverflow.com/questions/2112685/how-do-one-way-hash-functions-work)*.*

[3] Lamport: Password authentication with insecure communication (1981).

[4] Hwang, T., Chen, Y., Laih, C.S.: Non interactive password authentication without password tables (1990).

[5] Hwang, Hwang, L.M.S., Li, L.H.: A new remote user authentication scheme using smart card (2000).

[6] Sun, H.M.: An eﬃcient remote user authentication scheme using smart card (2000).

[7] Chien, H.Y., Jan, J.K., Tseng, Y.M.: An eﬃcient and practical solution to remote authentication: smart cards (2002).

[8] Ku, W.C., Chen, S.N.: weakness and improvement of an eﬃcient password based remote user authentication scheme using smart cards (2004).

[9] Yoon, E.J., Ryu, E.K., Yoo, K.Y.: Further improvement of an eﬃcient password based remote user authentication scheme using smart cards (2004).

16