A Novel Secure Remote User Authentication Protocol using Three Factors

Yogita Borse
University of Mumbai
K J Somaiya College of Engineering
Mumbai

Irfan Siddavatam
University of Mumbai
K J Somaiya College of Engineering
Mumbai

ABSTRACT

According to the recent work done in the area of remote user authentication, biometrics based password authentication using smart card is the most interesting and upcoming technology. Many protocols has been designed aiming to combine three authentication factors efficiently in order secure the process of remote user authentication, but failed to do so. One of the many possible reasons is biometrics comparison. Basically, biometric is used to uniquely identify the user. It has been observed that, the biometrics comparison during the verification is done using its hash value, which is infeasible due to it's avalanch effect property. Moreover, impersonation, server spoofing, man-in-the-middle, denial-of-service etc attacks needs to handle properly to guarantee the security of the protocol. The main objective of this paper is to focus on biometrics comparison and making the protocol immune to above mentioned attacks.

General Terms:

Security, Biometrics, Authentication protocols

Keywords:

Remote users, Authentication, Passwords, Biometrics, Smart Cards, Three factor Authentication.

1. INTRODUCTION

Considering today's lifestyle, use of remote services is increased globally. It is necessary for such service providers to identify the remote users and maintain their accountability for smooth running of their business. But it is been reviewed in literature that the available security mechanisms are not meeting the requirements.
Authentication is basically identification/recognition of the connecting party. It is of two types, unilateral authentication where only one party authentication happens(only client/server) or bilateral/mutual authentication where both/all parties needs to be authenticated(client(s) as well as server(s)).
There are three factors of authentication.

- Password (Something that you know),
- Smart card (Something that you have) and
- Biometric (Something that you are).

Different combinations of these factors have been used to enhance the security. In early days of authentication systems, right from the 1981's Lamport's [1] simple authentication protocol till Kerberos authentication system, all were using only one authentication factor that is nothing but password. These schemes were based on the verification of user's legitimacy with the password stored in the server side database, which is vulnerable as shown by [2, 3]. The most important limitation of password based protocol was server side password database. Many cryptographic solutions were experimented to make the password database secure [4], but all were fail. This has then become the driving force for hardware tokens like smart cards. The smart-card-based password authentication has then become one of the most common authentication mechanisms [5, 6, 7, 8]. To overcome other limitations of passwords, biometric was included as a third authentication factor.
Biometrics, where users are identified by their measurable human characteristics, such as fingerprint, voice print, and iris scan etc. Biometric characteristics are believed to be a reliable authentication factor since they provide a potential source of high-entropy information and cannot be easily lost or forgotten. Biometric feature of an user is so unique that even twines cannot have the same biometrics. Despite these merits, biometric authentication has some imperfect features. Unlike password, biometric characteristics cannot be easily compare, change or revoke. Moreover, these security mechanisms are costlier. But biometrics are most reliable among all three factors of authentication, therefore many recent authentication protocols are using the combination of these factors to enhance the security [9, 10, 11]. Therefore, this paper tried to put focus on the efficient use of biometrics using best of cryptographic techniques, while keeping overall authentication process as simple as possible.
Rest of the paper is organized as follows- in section (2), the review of literature survey done in the direction of evolution of authentication protocols is given. The proposed secure remote user authentication protocol is explained in Section (3). In Section (4) proposed protocol's design criteria fulfillment, security analysis against different attacks and time & space based performance analysis is given. Finally, Section (5) gives the overall conclusion of the paper.

2. LITERATURE SURVEY

In 2009 [12], Ou Qingyu et. al. cryptanalyze the Yoon et al's scheme, they pointed out that one of the parameter stored in the smart card was completely insecure. With that parameter, the password was easily revealed and both the login message and response message were easily forged. So, Ou Qingyu et. al. proposed the scheme to fix this flaw.
In 2010 [13], Ronggong Song has proved that in Xu et. al. [7] scheme the user's identity is independent of the server's secret value stored on the smart card due to which the user is able to mount an impersonation attack. In order to resist the impersonation attack, the authentication scheme must ensure that the server must use the user's identity in order to recover authentication data. Ronggong Song has proposed a new efficient strong smart card based password authentication protocol that satisfies not only the minimum conditions but also advanced requirements like efficiency and mutual authentication.
In the same year 2010 [14] Rafael et al. have analyzed the Das et. al. scheme proposed in 2004 [15] and shown that it is vulnerable to insider, masquerade, and server spoofing attacks as well as fails to provide mutual authentication. Hence, they proposed the solution which resolves all the flaw they shown in Das scheme.
Rafael et al. scheme is based on one-way secure hash function. It also provides mutual authentication and session key generation for secure communication.
Meanwhile, many people were working on combining three factors to authenticate a remote user. In 2004 [10], Lin and Lai analyzed Lee-Ryu-Yoo scheme proposed in 2002 [9] which was based on three factors of authentication. They proved that Lee-Ryu-Yoo scheme has serious flaw of masquerading. The authors have suggested the solution to the flaw as well as proposed a new scheme which is based on Lee-Ryu-Yoo scheme. In the proposed scheme authors has used ElGamal's crypto system and a one-way secure hash function. But in 2007 [11], Zhang et al. analyzed the Lin-Lai scheme [10] and pointed out that their scheme is vulnerable to server spoofing attack. Zhang et al. shown that Lin-Lai's scheme performs only unilateral authentication (only client authentication) and there is no mutual authentication between user and remote system, thus their scheme is susceptible to the server spoofing attack. To overcome this security flaw, Zhang et. al. presented an improved security patch.
Later in 2013 [16], authors of this paper analyzed Lin-Lai's scheme as well as Zhangs et. al. improved patch and proved that Zhang et. al. improved patch is only focusing on one vulnerability of Lin-Lai's scheme. Authors shown that Lin-Lai's scheme has serious vulnerabilities in login, authentication and password change phases. Moreover it is also susceptible to stolen smart card attacks, Man-in-the-Middle attack, outsider attacks and proposed a complete new protocol based on Lin-Lai's scheme.
In this paper, the proposed authentication protocol based on biometrics, password and smart card is presented. The protocol is design in order to overcome the aforementioned vulnerabilities and flaws.

3. PROPOSED PROTOCOL

The proposed secure authentication protocol(SA) uses three factors of authentication. The first factor password along with the user identity is used to identify the owner of the smart card, the second factor smart card is used to securely store the user's secrete data and the third factor finger print is used to uniquely identify the user.

Table 1 shows the notations used in the protocol. The proposed protocol comprises of the following phases

Table 1. **Table of notations used in the proposed protocol**

Notation	Description
Ui	Registered user
IDi	Ui's identity
PWi	Ui's password
Si	Ui's finger print minutiae
Xs	Server's secrete
r, rs	nonce generated from Si
A⊕B	XOR operation on data A and B
A‖B	concatenation of data A and B
IDi', PWi', Si'	Ui's current identity, password and fingerprint minutiae respectively.

3.1 Registration phase

This phase is protected using SSL protocol. Every new user has to first register to login to the system. For registration, new user first selects his/her password PWi and IDi information. Then he/she provides his/her finger print Si on a fingerprint scanner. Then client machine performs following

1. User inputs his/her password PWi & identity IDi on client machine and finger print Si on a fingerprint scanner.
2. The client machine then computes and sends message (IDi, h(PWi), E_{PWi}(Si)) to registration server, where, h(PWi) is hashed password and E_{PWi}(Si) is encrypted Si using PWi.
3. On receiving registration request from user, registration server computes
   ai=h(IDi‖Xs)
   Ai=h(ai)
   Bi=ai⊕h(IDi‖h(PWi)) and
   stores (Ai, Bi, E_{PWi}(Si), h(.)) on smart card.
4. Finally, registration server sends registration completed message to the user and issues smart card through secure means.

3.2 Login phase

In this phase, a registered user Ui needs to login to the server inorder to access it's services. At the client side Ui's identity, password and fingerprint gets verified against the data stored on the Ui's smart card. Only on successful verification, client sends authentication message to the server(avoids DoS). Following are the detailed steps for login to the server

1. Ui first inserts his/her smart card into the smart card reader of client machine and enters IDi' and password PWi'.
2. Client machine first verifies the password by computing
   ai' = Bi⊕h(IDi'‖h(PWi'))
   Ai' = h(ai') and
   matching Ai' with Ai stored in smart card. If this verification fails, client rejects the user login request. Otherwise, it asks the Ui to scan fingerprint, let's say Si' is Ui's fingerprint minutiae at the time of login.
3. Then client decrypts the E_{PWi}(Si) stored on Ui's smart card and Si' is matched against the decrypted E_{PWi}(Si). If this verification fails, client rejects the user login request. Otherwise Ui passes the login phase.

3.3 Authentication phase

On successful login, client computes authentication parameters as follows

1. Generates a random number 'r' using the Si' and computes
   cid = ai' $\oplus$ r
   uid = h(PWi) $\oplus$ r
   ei = h(Bi$\|$cid$\|$uid$\|$r)
   and sends message (IDi, cid, uid, ei) to the server for authentication.
2. On receiving authentication message from Ui, authentication server looks for the IDi in the user database. If IDi is present in database, computes
   ai = h(IDi$\|$Xs)
   r'= ai $\oplus$ cid
   and r' is matched against the random number stored in the IDi's record r$_pre$. If it is matched, that means it may be a replay attack and rejects the authentication request by terminating the session.
3. Otherwise, replaces random number in IDi's record with r' and computes
   h(PWi)' = uid $\oplus$ r'
   Bi' = ai$\oplus$h(IDi$\|$h(PWi)')
   ei' = h(Bi' $\|$ cid $\|$ uid $\|$ r')
   and verifies whether ei ?= ei'. If this verification fails, server rejects Ui's authentication request and terminates session. Otherwise it proves the authenticity of Ui.
4. On successful client authentication, server computes mutual authentication message for the client to verify as following
   generates a random number 'rs' from r'
   computes Fs = h(ai $\|$ Bi' $\|$ (r'+1))
   and sends message (Fs,rs) to the client.
5. On receiving response from the server, client verifies whether Fs ?= h(ai' $\|$ Bi $\|$ (r+1)). If this verification fails, client rejects the server's mutual authentication and terminates the session. Otherwise, it proves the legitimacy of the server and client acknowledge the server.
6. Finally, on successful authentication both server and client generates shared session key SK = h(ai $\|$ Bi $\|$ r $\|$ rs)

3.4 Change password

Whenever, a user Ui decides to change his/her old password, he/she needs to first pass the login phase, then the protocol asks for the new password and updates the smart card accordingly. The steps are as shown below.

1. After successful login, client machine asks Ui to enter new password, let's assume Ui's new password is PWi*.
2. Then client machine performs following
   Decrypts Si stored on smart card using old password PWi
   Si = $D_{PWi}(E_{PWi}(Si))$ and encrypts using PWi*
   $E_{PWi*}(Si)$ and computes
   PWi' = h(IDi$\|$h(PWi*)),
   ai = Bi$\oplus$h(IDi$\|$h(PWi)),
   Bi* = ai $\oplus$ h(IDi$\|$h(PWi*)).
   Finally, replaces Bi with Bi* and $E_{PWi}(Si)$ with $E_{PWi*}(Si)$ on smart card.

4. RESULTS

Results of the proposed protocol are shown in three different ways, first: fulfillment of evaluation criteria suggested by [17], second: security analysis of the proposed protocol against different attacks and third: performance comparison of the proposed protocol on the basis of time requirement.

4.1 Evaluation Criteria Fulfillment

Proposed protocol is design such that it can fulfill the all twelve evaluation criteria suggested by [17]. The details are as below-
To achieve C1, we are using the smart cards to store user credential and so, server is not maintaining any password database. To achieve C2 and C4 at the same time, a verification of the authenticity of the original password before updating the value of Bi and Si in the memory of smart card is essential. And thus, besides Bi, additional parameter 'ai' is required during verification of password who's hash value 'Ai' is stored on smart card. Getting 'ai' from its hash value is computationally infeasible. Hence, this resist the adversary from getting 'ai' without the knowledge of IDi and PWi and vice-versa. Moreover, if an adversary got the Ui's smart card and successful in extracting the parameters like Ai, Bi or encrypted Si, he/she will not be able to get any knowledge about secrets PWi or Xs. To achieve C3, registration phase is protected using SSL protocol. Moreover, during the registration phase, Ui sends h(PWi) to the server and not the plain text. According to the one-way property of the hash function it is difficult for an adversary to derive the password from its hash value. To achieve C6, an entry (IDi, r_{reg}, T_{reg}) corresponding to Ui is stored in the server's database, only T_{reg} needs to be updated when Ui revokes smart card. According to C7, proposed protocol generates the shared secret session key SK on successful authentication; and according to C8, proposed protocol uses random nonce instead of time-stamps to prove the freshness of the messages. To achieve C9, during the login phase Ui provides password and identity, which is verified against the Bi stored on the smart card before change of password. However, only after successful login user can connect to the server. To achieve C10, during authentication phase, server reply with Fs=h(ai $\|$ Bi' $\|$ (r'+1)), where 'ai' calculation requires server secret Xs which is known to the server only. So, on client side when client finds Fs and h(ai $\|$ Bi $\|$ (r+1)) equal, it proves the legitimacy of the server. In proposed protocol even though IDi is traveling in plain form, having the knowledge of IDi attacker cannot harm the system. Fulfillment of C5 and C12, is as shown in the section 4.2.

4.2 Security Analysis

The following section discuss security analysis of the proposed protocol against different attacks and shown how proposed protocol resist different attacks. The protocol's security is rely on secure one-way hash function, cryptographic algorithm and uniqueness of biometrics. In the analysis it is assumed that the smart card is not temper proof [18, 19].

1. Off-line Password Guessing attack- In proposed protocol, if an adversary gets Ui's smart card and finds out the value of (Ai, Bi) where
   Ai = h(ai) = h(IDi $\|$ h(PWi)) and
   Bi = ai $\oplus$ h(IDi $\|$ h(PWi)).
   For off-line password guessing, an adversary require ai which is masked with secure one-way hash function. So, it is clear that using stolen smart card an adversary cannot launch off-line password guessing attack.
2. Denial-of-service(DoS) attack- In DoS attack, attacker simply wants to overload the server, so that it can not provide the services to the legal users. In the extreme case, server may crash. As mentioned in the login-authentication phase, until

and unless user is not passing the login phase he/she cannot connect to the server. So for an adversary without having legal user's IDi, PWi, Si and smart card, it is impossible to send fake authentication request to the server. Also, in case of stolen smart card, attacker will not be able to create fake login request as explained in of-line password guessing attack. However, for successful login, attacker should also know the Si, which is securely stored on smart card using symmetric encryption.

3. Replay attack - Proposed protocol uses random nonces 'r' and 'rs' instead of time-stamp to withstand replay attacks. Suppose that the attacker has intercepted a previous login request message (IDi, cid, uid, ei) from Ui, and replay the same message to server. Upon receiving the message when server will find r=r_pre will recognize the possible replay attack and immediately terminate the session.
4. Parallel Session attack - If the attacker intercepts the acknowledgment message (Fs, rs) from one session and try to use it in parallel session, he/she will fail. He/she cannot re-use it to create a valid login request message (IDi, cid, uid, ei) because the acknowledgment message does not contain information to construct a valid login request message.
5. Insider attack - If an attacker obtains Ai, Bi and E_{PWi}(Si) from Uis smart card, he/she cannot extract sensitive information, like PWi, Xs because it is computationally infeasible to invert the one-way hash function. Moreover, he/she cannot decrypt E_{PWi}(Si) without the knowledge of PWi. Furthermore, the server does not maintain any PWi verification table. Even if the attacker is a legal user Ui, he/she cannot obtain server's secret Xs from his/her smart card because it is masked with one-way hash function.
6. Client Impersonation attack - An illegal user may try to modify a login request massage (IDi, cid, uid, ei) into (IDi, cid*, uid*, ei*). However, such modification will fail in the authentication and session key agreement phase, because there is no way of obtaining the value of ai and secret password PWi to computes the valid parameters which will be accepted by the server. In addition, a legal user also will fail to calculate valid uid and hence, will fail during authentication phase.
7. Server Spoofing attack - An attacker cannot masquerade as a legal server because he/she cannot compute ai and Bi without knowing Xs. Moreover, he/she cannot extract nonce r and therefore, h(PWi) without the knowledge of ai. Furthermore, he/she cannot compute a correct session key SK. In addition, legal user also cannot masquerade as legal server because of server secret Xs, which is known to legal server only because though Xs is stored on user's smart card, there is no way to extract it.
8. Stolen Smart Card attack - In case, a users smart card is lost or stolen by the attacker, the attacker cannot use this card without knowing the valid IDi and PWi. If he/she can extract the secret information (Ai Bi, E_{PWi}(Si), h (.)) stored in smart card where,
   Ai=h(ai) and ai = h(IDi $\|$ Xs),
   Bi = ai $\oplus$ h(IDi $\|$ h(PWi))
   It is not possible to guess valid IDi and PWi at the same time.
9. **Man in the Middle(MiM) attack** The main purpose of MiM attack is to sit between the two remotely communicating parties and interrupt the communication. MiM is possible because of insecure sessions. In proposed protocol, at the end of authentication phase client and server generates a session key SK which helps in protecting the system from MiM attack.
10. Perfect Forward Secrecy(PFS) - According to evaluation criteria C12, proposed protocol's PFS is proven using following way,
    - Compromise of current long term key should not compromise future long term key
    - Compromise of old long term key should not compromise current long term key.
    - Compromise of current long term key should not compromise current or past session keys.
    - Compromise of current session key should not compromise current long term key.

    In proposed protocol, two long term keys are used namely PWi and Xs. And one short term key SK is used. Both long term keys are chosen by respective parties independently, so compromise in one key cannot leak the other. A short term key SK is computed using random nonces along with ai and Bi, so compromised PWi or Xs will not result in compromised SK(current, past or future). Moreover, the interchanging messages between client and server are protected using secure hash function and not secured using any of long term keys, so compromised in any of these keys will not reveal any information from past messages.
    A short term key SK is comprises of ai, Bi, r and rs out of which r and rs are nonces - random number, so in every session SK will be different and so compromised SK of one session will not help attacker to extract any information from the session messages. As well as because of its randomness attacker will not be able to compute or derive any long term key from it.

4.3 Performance Analysis

This section, summarizes performance of the proposed protocol by comparing it with related schemes in terms of computational cost and storage capacity. The paper mainly focus on the computations of registration, login, authentication phases since these phases are the main body of the proposed protocol. In order to carry out the computational cost evaluation, following notations are used: T_h, T_{cry} and T_c are defined as the execution times for one-way hash functions, cryptography and comparison operations, respectively. Because exclusive-or operation requires very low execution time, it is usually neglected considering its computational cost. The time complexity associated with the different operations can be expressed as $T_{XOR} \ll T_h \leq T_c \leq T_{cry}$. The computational cost is defined as the total time of various operations executed in each step. According to the above definition, the computation cost in the registration phase is $4T_h + 1T_{cry}$ time. The login phase requires $3T_h + 1T_{cry} + 2T_c$ time. Where as mutual authentication and session key generation phase requires $9T_h + 3T_c$ time. In addition, proposed protocol is evaluated and compared in terms of storage capacity also. It is assumed that the output size of a one-way hash function, random numbers and secret keys are 32-bytes in length, identity is 15-bytes in length and encrypted fingerprint is ¡1024-bytes. So, the memory needed in the user's smart card is (ai, Bi, $E_{PWi}(Si)$) = (32+32+1024) = 1088-bytes.
Here, SA protocol is compaered with the Sonwanshi et. al. protocol [20], as it forms the basic two factor based protocol for SA protocol. Table 2 shows that SA protocol require more computational cost than Sonwanshi et. al. scheme in registration and login phase. However, our scheme is resistance to replay, insider, leak of password, and masquerade attacks. Note that server computes three one-way hash functions while user Ui computes just a single one-way hash function and a single encryption considering the computational power of current servers,

Table 2. **Performance Comparison**

Cost/space	**Location**	**SA protocol**	**Sonwanshi et. al.**
Registration Phase	Client Server	T_h+T_{cry} $3T_h$	T_h $2T_h$
Login Phase	Client	$3T_h$+T_{cry}+$2T_c$	$2T_h$+T_c
Authentication Phase	Client Server	$2T_h$ $3T_h$+$2T_c$	$2T_h$ $5T_h$+T_c
Mutual Authentication & Session Key gen. Phase	Client Server	$2T_h$+T_c $2T_h$	$2T_h$+$2T_c$ T_h
Authorization Phase	Client Server	T_{cry} T_{cry}+T_c	-
Token generation Phase	Server	$2T_{cry}$	- -
Storage Required	Smart Card Server	10-11KB Xs = 32-bytes per Ui = 96-bytes	64-bytes X = 32-bytes per Ui =32-bytes

the execution time of three one-way hash functions is extremely very low. Moreover, because of using biometrics our scheme provieds non-repudiation which is not there in Sonwanshi et. al. scheme. In addition, our scheme provides authorization as well as token generation for achieveing single-sign-on(SSO). The storage capacity evaluation demonstrated that our scheme requires more space on smart cards because of biometrics, but it provides higher level of security and protects the system in case of compromised password and/or stolen smart card. It also provides non-repudiation which is difficult to achieve otherwise. Furthermore, in our scheme, the server requires more byte to store user data because the additional information related to user helps in revokation of smart card which Sonwanshi et. al. scheme fails to do.

5. CONCLUSION

This paper has introduced a novel remote user authentication protocol using fingerprint, smart card and password. The security analysis of the proposed protocol shows that it fulfills all twelve evaluation criteria which are the benchmarks for any secure authentication protocol, available in literature. Proposed protocol also resist the different attacks such as off-line password guessing attack, replay attack, DoS attack, server spoofing attack, masquerading attack, Man-in-the-Middle attack, parallel session attack, insider attack as well as attacks against compromised/stolen smart card. However, it also provides forward secrecy and SSO feature which are essential in distributed environment.

6. REFERENCES

[1] Leslie Lamport. Password authentication with insecure communication. *Commun. ACM*, 24(11):770–772, November 1981.

[2] Min-Shiang Hwang and Li-Hua Li. A new remote user authentication scheme using smart cards. *Consumer Electronics, IEEE Transactions on*, 46(1):28–30, 2000.

[3] M.K. Khan. Cryptanalysis and security enhancement of two password authentication schemes with smart cards. In *Multitopic Conference, 2007. INMIC 2007. IEEE International*, pages 1–4, 2007.

[4] Seung Wook Jung and Souhwan Jung. Secure password authentication for distributed computing. In *Computational Intelligence and Security, 2006 International Conference on*, volume 2, pages 1345–1350, 2006.

[5] Chun-Ta Li and Min-Shiang Hwang. An efficient biometrics-based remote user authentication scheme using smart cards. *J. Netw. Comput. Appl.*, 33(1):1–5, January 2010.

[6] Narn-Yih Lee and Yu-Chung Chiu. Improved remote authentication scheme with smart card. *Computer Standards & Interfaces*, 27(2):177–180, 2005.

[7] Jing Xu, Wen-Tao Zhu, and Deng-Guo Feng. An improved smart card based password authentication scheme with provable security. *Comput. Stand. Interfaces*, 31(4):723–728, June 2009.

[8] Ronggong Song. Advanced smart card based password authentication protocol. *Comput. Stand. Interfaces*, 32(5-6):321–325, October 2010.

[9] J.K. Lee, S. R. Ryu, and K.Y. Yoo. Fingerprint-based remote user authentication scheme using smart cards. *Electronics Letters*, 38(12):554–555, 2002.

[10] Chu-Hsing Lin and Yi-Yi Lai. A flexible biometrics remote user authentication scheme. *Computer Standards & Interfaces*, 27(1):19–23, 2004.

[11] Muhammad Khurram Khan and Jiashu Zhang. Improving the security of 'a flexible biometrics remote user authentication scheme'. *Computer Standards and Interfaces*, pages 82–85, 2007.

[12] Ou Qingyu, Huang Kai, and Li Guang. Cryptanalysis and improvement of a remote user authentication scheme. In *Intelligent Computation Technology and Automation, 2009. ICICTA '09. Second International Conference on*, volume 4, pages 49–52, 2009.

[13] Ronggong Song. Advanced smart card based password authentication protocol. *Comput. Stand. Interfaces*, 32(5-6):321–325, October 2010.

[14] R. Martinez-Pelaez, F. Rico-Novella, C. Satizabal, and J. Pomykala. Improvement of the dynamic id-based remote user authentication scheme. In *Information Society (i-Society), 2010 International Conference on*, pages 168–172, 2010.

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

[16] Yogita Borse and Irfan Siddavatam. Article: Mitigating vulnerabilities in 3-factor based authentication. *International Journal of Computer Applications*, 76(10):19–23, August 2013. Published by Foundation of Computer Science, New York, USA.

[17] Ding Wang and Chunguang Ma. Robust smart card based password authentication scheme against smart card loss problem. *IACR Cryptology ePrint Archive*, page 439. informal publication.

[18] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '99, pages 388–397, London, UK, UK, 1999. Springer-Verlag.

[19] Thomas S. Messerges, Ezzat A. Dabbish, and Robert H. Sloan. Examining smart-card security under the threat of power analysis attacks. *IEEE Trans. Comput.*, 51(5):541–552, May 2002.

[20] S.S. Sonwanshi, R.R. Ahirwal, and Y.K. Jain. An efficient smart card based remote user authentication scheme using hash function. In *Electrical, Electronics and Computer Science (SCEECS), 2012 IEEE Students' Conference on*, pages 1–4, 2012.