Journal of Computer Science

Original Research Paper

A Practical and Secure Hash Function-Based Password Authentication Scheme

[1]Muhammad Helmi Ali, [2]Eddie Shahril Ismail and [3]Firdaus Mohamad Hamzah

[1,2]Centre for Modelling and Data Science, Faculty of Science and Technology,
Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
[3]Smart and Sustainable Township Research Centre, Faculty of Engineering and Built Environment,
Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia

Article history
Received: 16-03-2019
Revised: 18-06-2019
Accepted: 13-07-2019

Corresponding Author:
Eddie Shahril Ismail
Centre for Modelling and Data Science, Faculty of Science and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
Email: esbi@ukm.edu.my

Abstract: In this study, we propose a practical and secure hash function-based password authentication scheme using smart cards. Our proposed scheme offers some advantages and interesting features. Firstly, the scheme does not require a verification table and is secure against the replay attacks, an attack that most of the existing schemes suffer. Secondly, any user of the scheme be allowed to change his or her account's password efficiently. Thirdly, the time complexity for each algorithm in the proposed scheme are relatively low and minimal compared to some existing well-known password authentication schemes.

Keywords: Cryptography, Password Authentication Scheme, Hash Function, Smart Cards

Introduction

The global network channel with today's technology is a must for it conveniently provides access to various type of online services through remote systems, an open system that enable users from over the world to communicate confidently and safely. However, the open environment of this kind communication has normally led to the source of security risks and threats. These include data disclosure, data modification and even worse identity impersonation. The password authentication scheme provides a reliable and solid protection from these three main security risks. The scheme allows one to verify the legitimacy of a user over an insecure channel and simultaneously enabling the server to authenticate the user prior to providing access to network database and services. Conventional password authentication schemes normally request users to generate and submit their private username or user ID and password as inputs into the system. The system is designed to match and verify the input data from the verification table in the server. Many existing schemes use this verification table as a mechanism and platform to identify and validate the users. However, it requires one to maintain and administer the space and storage of the table in the scheme. Usually the verification table is protected by the server and is assumed secure against any form of attacks. Some previous verification table-based password authentication schemes were not designed to withstand replay attack, an attack where an intruder observes on a secure network communication, intercepts a message and then fraudulently delays or resends the message to misdirect the receiver into doing what the intruder wants. In other words, the intruder could "impersonate" the user by intercepting the message from the network and later log in and communicate to the server as a "legitimate" user. This attack is commonly due to improper and inappropriate selection of keys and parameters in their algorithms and in the design of the mathematical equations involving these keys and parameters of the scheme. In short, verification table allows intruders to modify the password verification table which indirectly resulted in uncontrollable cost of managing and protecting the table. Evans *et al*. (1974) proposed a method for protecting the verification table by preventing passwords from being disclosed. However, the proposed technique failed as users could be impersonated. Lennon *et al*. (1981) then proposed a secure scheme which used a test pattern and several secret keys to construct a relationship between the user ID and their password. However, Hwang (1983) and later Harn *et al*. (1989) showed that the scheme has a major drawback wherein all the user's password could be retrieved by an intruder subject to changes in secret keys. Lamport (1981) proposed a password authentication scheme and claimed that the scheme secure against replay attacks. However, it was later noted by Chang and Wu (1991) that intruders could modify the user's passwords stored in the password table within the system. Few years later, several password authentication schemes using smart cards have been proposed by researchers (Chang and Liao, 1994; Chien *et al*., 2001; Sun, 2000; Liao *et al*., 2006; Hwang, 1999; Wang and Chang, 1996; Wu, 1995; Lee *et al*., 2002; Kumar, 2010; Sood, 2012; Thandra *et al*., 2016). However, a common drawback from many schemes is that users are not allowed to change their passwords as no such mechanism are provided in the schemes for example in (Sun, 2000; Chien *et al*., 2002; Xu *et al*., 2009; Rajaram and Amutha, 2012). The practical solutions to the above systems have been demonstrated by many researchers such as Hwang *et al*. (2002) and Chang and Lee (2006). Wang and Chang (1996) applied the concept of timestamp to an improved authentication scheme based on ElGamal signature (1985). The authors showed the ability for a remote system to determine the validity of the authentication message, but unable to validate the identity of the user. Hwang *et al*. (2002; Lee *et al*., 2002; Kumar, 2010; Sood, 2012; Rajaram and Amutha, 2012; Lee, 2013; Thandra *et al*., 2016; Pooja and Pramav, 2016; Liu *et al*., 2017) enhanced the authentication scheme using smart cards by allowing users to change and select their passwords without revealing them to the server.

Sun (2000) proposed an efficient and practical remote user authentication scheme. No password table is required to keep in his system and therefore the communication and computation costs are reduced. Hwang *et al*. (2002; Chien *et al*., 2002; Lee *et al*., 2002) respectively proposed their simple remote user authentication schemes. In those schemes, the authors claimed that their schemes could achieve the following goals: requires no verification table on the server's side; low communication and computation costs, the replay attack problem completely solved and users' freedom to choose their own passwords.

Yoon *et al*. (2005) next improved upon Hwang *et al.*'s simple remote user authentication scheme using smart cards. Xu *et al. (*2009) proposed an exponential based smart card authentication scheme and claimed that it can resist various feasible attacks. Kumar (2010) proposed a scheme wherein the server and user authenticate one another and then generate a secret session key for secure communication. In this scheme, the remote user is free to change his or her password without connecting to the server. Next, Rajaram and Amutha (2012) proposed an efficient password authentication scheme for smart cards by using RSA algorithm which offers minimum computational costs. Sood (2012) proposed a protocol that allow the user to choose and change password at their choice and provides mutual authentication between the user and the server to protect it from forgery attack. Lee (2013) proposed an improved scheme claimed that it can provides dynamic identity and user anonymity, obtaining forward or backward secrecy and mutual authentication and can withstanding the replay attack, insider attack and impersonation attack. Pooja and Pramav (2016) proposed a biometrical authentication scheme based on geometric approach using smart card. The combination of passwords, smart card and biometric is used to construct a secure three factor authentication scheme. Thandra *et al*. (2016) cryptanalyzed and improved Rajaram and Amutha's scheme and achieved mutual authentication during the login phase. Liu *et al*. (2017) proposed a smart card-based password authentication scheme to overcome the weaknesses of Li *et al*. (2013)'s scheme and claimed that it can resist various type of attacks with better computational efficiency.

In this study, we propose a new practical and secure password authentication scheme using smart cards based on hash functions. The new scheme is designed to resist replay attacks, guess and impersonation attacks and allows users to freely change their passwords. In the event of a lost smart card, the system also facilitates for a secure transfer into a new system without creation of a new ID. One of the advantages of this system is the lack of necessity to maintain a password verification table for verifying the legitimacy of the user. The time complexity, communication and computational costs of our proposed scheme are relatively low and minimal compared to some existing schemes.

The Design of Password Authentication Scheme

The conventional password authentication scheme consists of four components; initialization phase, registration phase, login phase and authentication phase. Our scheme design makes use of cryptographic hash functions and one may refer (Paar and Pelzl, 2010) for theoretical aspects of definitions and properties on hash functions. Basically, for security reason, we require that the hash function is collision-resistant that is, it is hard to find two different inputs of the function that map to a same value of output. In other words, we require that no algorithm could find the collision of the hash function in polynomial time.

The scheme also makes use of a secure channel. This channel assumes no attacker in some ways could steal or modify the smart card. In real life, this is not something impossible to achieve and in fact many forms of secure channel have been introduced and used practically and securely. The new proposed scheme also allows users to freely change their passwords if necessary. The scheme requires an SRC, the **honest-based assumption server** to communicate with the users and complete the required transactions. We now examine and discuss the phases/algorithms in our password authentication scheme. Refer Fig. 1-3 for summary and clear description.

Phase 1: Initialization Phase

1. SRC picks two public secure cryptographic hash functions; $h(\cdot,\cdot)$ and $H(\cdot)$
2. SRC chooses his master secret key, x

3. SRC creates user's identity as ID_i
4. The user with identity ID_i then chooses his password as PWD_i

The hash functions, $h(\cdot,\cdot)$ and $H(\cdot)$ respectively need two and single inputs and output a single number. The secret key, x is kept private by the SRC and only the legitimate user knows his PWD_i whereas his ID_i is publicly known.

Phase 2: Registration Phase

In this phase, a smart card will be issued by the SRC to each registered user, U_i:

1. User U_i submits his identity, ID_i and password, PWD_i to SRC for registration
2. SRC then computes user's secret information, $w = h(ID_i, x)$ using SRC's secret key
3. SRC calculates $b = w \oplus H(PWD_i)$ where $\oplus$ denotes the standard XOR operation
4. SRC embeds the two computed secret values w and b into the smart card
5. SRC issues the smart card to the user U_i via a secure channel

The smart card now contains the values of w and b which uniquely corresponds to the authorized user. The card must be safely stored by the user.

Phase 3: Login Phase

If user wishes to log into the system, he must first insert the smart card into the terminal and insert his identity, ID_i and password, PWD_i. Suppose that T is the current time of the user inserting his identity and password into the smart card. The smart card then digitally performs the following operations:

1. Compute $t = h(T, ID_i)$
2. Compute $m = b \oplus H(PWD_i)$
3. Compute $C = h(m, t)$
4. Send the message $E = (ID_i, C, T)$ to the system

Phase 4: Authentication Phase

Upon receiving the message E, the system authenticates the login user as follows. Suppose that, the system receives the message E at time T'. The system then does the following:

1. Check the validity of ID_i. If the format of ID_i is not correct, then the system rejects the login request
2. Check the validity of the time interval between T and T' to resist replay attacks. If $T' - T \geq \Delta T$, where ΔT denotes the allowable expected valid time interval for transmission delay, then the system rejects the login request
3. Compute $w = h(ID_i, x)$
4. Check whether $C = h(w, t)$. If it holds then the system accepts the login request. Otherwise, it rejects the login request.

Proposition 1

If the above system (Phase 1-4) runs smoothly and the message $E = (ID_i, C, T, t)$ is properly generated, then the login request via $C = h(w, t)$ will always be successful. This can be shown as below:

$$\begin{aligned} C = h(m,t) &= h\left(b \oplus H\left(PWD_i\right),t\right) \\ &= h\left(w \oplus H\left(PWD_i\right) \oplus H\left(PWD_i\right),t\right) \\ &= h(w,t). \end{aligned}$$

If the login request is rejected three times the user's account will be automatically locked. The user then must contact the SRC to unlock the account.

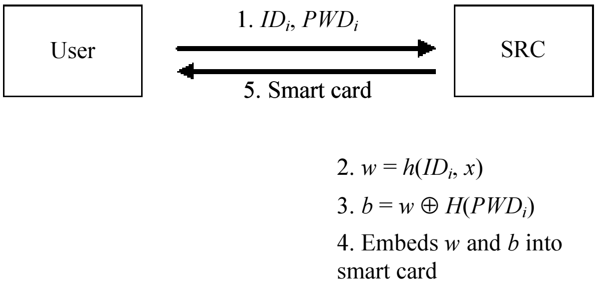
User
1. ID_i, PWD_i
SRC
5. Smart card
2. $w = h(ID_i, x)$
3. $b = w \oplus H(PWD_i)$
4. Embeds w and b into smart card

Fig. 1: The Registration phase

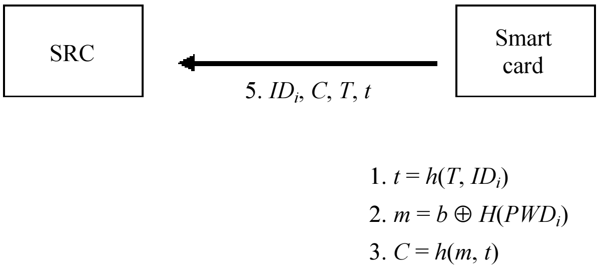

Fig. 2: The Login phase

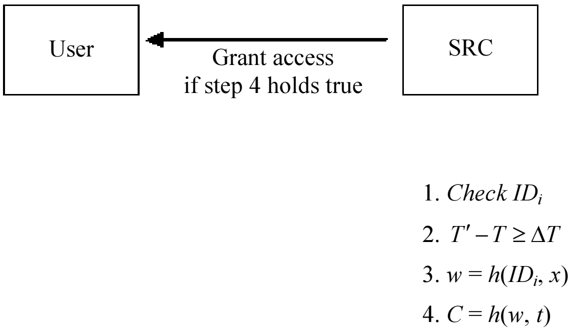

Fig. 3: The Authentication phase

Password-Changing Mechanism

If user wants to change his password from PWD_i to PWD' the following procedure will be performed:

1. User selects and sends PWD_i' to the SRC
2. SRC calculates $b' = b \oplus H\left(PWD_i\right) \oplus H\left(PWD_i'\right)$
3. SRC updates b to b' on the memory of smart card

The new password, PWD_i' is now successfully updated as shown below:

$$\begin{aligned} b' &= b \oplus H\left(PWD_i\right) \oplus H\left(PWD_i'\right) \\ &= w \oplus H\left(PWD_i\right) \oplus H\left(PWD_i\right) \oplus H\left(PWD_i'\right) \\ &= w \oplus H\left(PWD_i'\right). \end{aligned}$$

Security and Performance Analysis

In this section, we define the security requirements an ideal password authentication scheme should satisfy. The definitions are taken from Tsai *et al.* (2006):

Forgery Attacks (Impersonation Attacks)

An attacker attempts to modify intercepted communications to masquerade the legal user and login to the system.

Forward Secrecy

It ensures that the previously generated passwords in the system are secure even if the system's secret key has been revealed in public by accident or is stolen.

Password Guessing Attacks

Most passwords have such low entropy that it is vulnerable to password guessing attacks, where an attacker intercepts authentication messages and stores them locally and then attempts to use a guessed password to verify the correctness of his/her guess using theses authentication messages.

Replay Attacks

An attacker eavesdrops on a secure network communication, intercepts it and then maliciously or fraudulently delays or resends it to misdirect the receiver into doing what the attacker wants.

Smart Card Loss Attacks

When the smart card is lost or stolen, unauthorized users can easily change the password of the smart card or can guess the password of the user by using password guessing attacks or can impersonate the user to login to the system.

Denial of Service Attacks

An attacker can update false verification information of a legal user for the next login phase. Afterwards, the legal user will not be able to login successfully anymore.

Parallel Session Attacks

Without knowing a user's password, an attacker can masquerade as the legal user by creating a valid login message out of some eavesdropped communication between the user and the server.

This section demonstrates that the proposed password authentication scheme is secure against the following cryptographic attacks. A good password authentication scheme provides protection from different possible attacks. The cyber-criminal might perform attacks as follows.

Resistance to Impersonation Attack

We demonstrate that our scheme can resist from impersonation attack. In this case, the adversary/intruder first chooses w' and tries to solve $w' = h\left(ID_i, x\right)$ for SRC's secret key, x. However, this is impossible because of the hardness of inverting the hash function, h. Adversary may also submit the identity ID_i and his choice of password PWD^A to the SRC who then performs the Phase 2. Now the adversary learns that $b^A = w = \oplus H(PWD^A)$. If $b = b^A$, then the adversary successfully impersonates the actual user. However, this is impossible to achieve because $b = b^A$ holds true if and only if $H(PWD_i) = H(PWD^A)$. Note that, selecting PWD^A randomly from the space of password values will give the probability of $H(PWD_i) = H(PWD^A)$. As small as $\frac{1}{2^n}$ where n is the number of possible passwords.

Resistance to Guessing Attacks

Guessing attacks can be classified by off-line and on-line attacks. In our scheme, if adversary intercepts $E = (ID_i, C, T, t)$, he cannot recover PWD_i by playing off-line guessing attacks since the PWD_i is fully protected by the hardness of inverting the one-way hash functions, h and H in C. Now, assuming the adversary passes the login phase, the device then sends $E^A = (ID_i, C^A, T, t)$ where $C^A = h(m^A, t)$ to the system. However, it is clear that $C^A = h(m^A, t) \neq h(w, t)$ since $m^A = b \oplus H(PWD^A) \neq b \oplus H(PWD_i) = w$. where the equality happens only if $PWD^A = PWD_i$ which again occurs with non-negligible probability. Therefore, the system will reject the adversary's login request. The on-line guessing attacks on the other hands can be prevented easily by limiting the number of failed log-in. For example, if the login request is rejected three times then the user's account will be automatically locked by the system.

Resistance to Replay Attack

The replay attack cannot work on our proposed scheme because of timestamp, the SRC is able to identify an intruder who replays a message. Assuming in the login phase intruder is successful send the message $E = (ID_i, C, T, t)$ to the SRC. Even if an intruder eavesdrop message E successfully in login phase and replay message E to confuse the server, he will fail in step 2-check validity of time interval of the authentication phase. Next, to pass the authentication phase the intruder must change T in order to satisfy $T' - T \geq \Delta T$. However, once T is changed, the authentication phase would fail.

Resistance to Man-In-The Middle Attack

In this type of attack, the attacker intercepts the messages send between the user and the SRC, then replays these intercepted messages within the valid time frame. In our proposed scheme, the attacker can intercept the login request message $E = (ID_i, C, T, t)$ from the user to the server. Then he starts a new session with the SRC by sending a login request by replaying the login request message $E = (ID_i, C, T, t)$ within the valid time frame. The attacker can authenticate itself to the SRC as well as to the legitimate user but cannot compute the session $C = h(w, t)$ because the attacker does not know the value of w, b and $\boldsymbol{m}$. Therefore, the proposed scheme is secure against man-in-the-middle attack.

Resistance to Insider Attack

In this attack, a privileged insider of the server can access other servers by stealing the identity. However, in the proposed scheme, it is computationally infeasible for the attacker to derive the password PWD_i from the $b = w \oplus H(PWD_i)$ because of the system is protected by the one-way function. Therefore, the proposed scheme resists the insider attack.

Resistance to Denial of Service Attack

Denial of service attack might be result from the computation consumption. If ID_i is a legal user identity and T is a legitimate timestamp, the server will perform the authentication. They might send the forged login request message to server. The more forged login request messages are sent, the more computational load to server performs. In the proposed scheme, if the login request is rejected three times then the user account is automatically will be locked and he has to contact server to unlock the account.

Resistance to Stolen Smart Card Attack

The same scenario goes if the user loses his smart card. The adversary still needs to insert the actual password before granting access to the system. However, this seems impossible. If adversary tries to update the password, he still needs to supply the actual password to the SRC before the mechanism is done.

For performance consideration, we compare our scheme with twelve other schemes in terms of various operations. We chose the twelve schemes since they are hash function-based and are secure until today. Table 1 shows the comparisons of time complexity, communication and computational costs between the twelve schemes for various phases. We define three mode of operations used in the system as the following: T_{hash} is time for performing hash function operation, T_{mul} is time for performing multiplication operation and $T_{\oplus}$ is time for performing Exclusive-OR (XOR) operation.

From Table 1, discusses the implementation result of proposed scheme and related schemes. Xu *et al.* (2009) needs $1T_{\exp} + 2T_{hash}$ computational cost for registration phase, $3T_{\exp} + 5T_{hash}$ for login phase and $3T_{\exp} + 4T_{hash}$ for authentication phase. Kumar (2010) needs $T_{\exp} + 2T_{hash} + T_{\oplus}$ computational cost for registration phase, $3T_{\exp} + 5T_{hash} + 2T_{\oplus}$ for login phase and $3T_{hash} + 3T_{\oplus}$ for authentication phase. Sood (2012) needs $4T_{\exp} + T_{hash} + T_{\oplus}$ computational cost for registration phase, $6T_{\exp} + 5T_{hash}$ for login phase and $6T_{\exp} + 4T_{hash} + T_{\oplus}$ for authentication phase. Rajaram and Amutha (2012) needs $2T_{\exp} + 3T_{mul}$ computational cost for registration phase, $2T_{\exp} + 3T_{mul}$ for login phase and $3T_{\exp} + 2T_{mul}$ for authentication phase. Thandra *et al.* (2016) needs $2T_{\exp} + 3T_{mul} + 3T_{hash}$ computational cost for registration phase, $4T_{\exp} + 4T_{mul} + 2T_{hash}$ for login phase and $3T_{\exp} + 6T_{mul} + 2T_{hash}$ for authentication phase. Pooja and Pramav (2016) needs $3T_{mul} + 7T_{hash} + 2T_{\oplus}$ computational cost for registration phase, $4T_{mul} + 8T_{hash} + 4T_{\oplus}$ for login phase and $2T_{mul} + 5T_{hash} + 3T_{\oplus}$ for authentication phase. Liu *et al.* (2017) needs $7T_{hash} + 2T_{\oplus}$ computational cost for registration phase, $4T_{hash} + 4T_{\oplus}$ for login phase and $9T_{hash} + 12T_{\oplus}$ for authentication phase. Lee (2013) needs $4T_{hash} + T_{\oplus}$ computational cost for registration phase, $11T_{hash} + 11T_{\oplus}$ for login phase and $4T_{hash} + 4T_{\oplus}$ for authentication phase. Chien *et al.* (2002) needs $T_{hash} + 2T_{\oplus}$ computational cost for registration phase, $T_{hash} + 2T_{\oplus}$ for login phase and $4T_{hash} + 4T_{\oplus}$ for authentication phase. Hwang *et al.* (2002) needs $2T_{hash} + 2T_{\oplus}$ computational cost for registration phase, $T_{hash} + 2T_{\oplus}$ for login phase and $2T_{hash} + 2T_{\oplus}$ for authentication phase. Sun (2000) needs T_{hash} computational cost for registration phase, $T_{hash} + T_{\oplus}$ for login phase and $2T_{hash} + T_{\oplus}$ for authentication phase. Lee *et al.* (2002) needs $2T_{hash} + 2T_{\oplus}$ computational cost for registration phase, $3T_{hash} + T3_{\oplus}$ for login phase and $2T_{hash} + 2T_{\oplus}$ for authentication phase.

We can see that our scheme is more superior compared to the other twelve schemes except that for Login Phase. Note that our scheme also provides most efficient for password–changing. No hashing is required in the mechanism.

Table 1: Comparison between twelve schemes in terms of their time complexity

	Registration Phase	Log in Phase	Authentication Phase	Total	Password Change
The new scheme	$T_{hash} + T_{\oplus}$	$2T_{hash} + T_{\oplus}$	$2T_{hash}$	$5T_{hash} + 2T_{\oplus}$	$2T_{\oplus}$
Xu *et al.* (2009)	$1T_{\exp} + 2T_{hash}$	$3T_{\exp} + 5_{hash}$	$3T_{\exp} + 4T_{hash}$	$7T_{\exp} + 11T_{hash}$	Not supported
Kumar (2010)	$T_{\exp} + 2T_{hash} + T_{\oplus}$	$3T_{\exp} + 5T_{hash} + 2T_{\oplus}$	$T_{hash} + 3T_{\oplus}$	$4T_{\exp} + 10T_{hash} + 6T_{\oplus}$	$2T_{\oplus}$
Sood (2012)	$4T_{\exp} + T_{hash} + T_{\oplus}$	$6T_{\exp} + 5T_{hash}$	$6T_{\exp} + 4T_{hash} + T_{\oplus}$	$16T_{\exp} + 10T_{hash} + 2T_{\oplus}$	$T_{\exp} + T_{hash}$
Rajaram and Amutha (2012)	$2T_{\exp} + 3T_{mul}$	$2T_{\exp} + 3T_{mul}$	$3T_{\exp} + 2T_{mul}$	$7T_{\exp} + 8T_{mul}$	Not supported
Thandra *et al.* (2016)	$2T_{\exp} + 3T_{mul} + 3T_{hash}$	$4T_{\exp} + 4T_{mul} + 2T_{hash}$	$3T_{\exp} + 6T_{mul} + 2T_{hash}$	$9T_{\exp} + 13T_{mul} + 7T_{hash}$	$2T_{\exp} + 3T_{mul} + 2T_{hash}$
Pooja and Pramav (2016)	$3T_{mul} + 7T_{hash} + 2T_{\oplus}$	$4T_{mul} + 8T_{hash} + 4T_{\oplus}$	$2T_{mul} + 5T_{hash} + 3T_{\oplus}$	$9T_{mul} + 20T_{hash} + 9T_{\oplus}$	$2T_{mul} + 3T_{hash}$
Liu *et al.* (2017)	$7T_{hash} + 2T_{\oplus}$	$4T_{hash} + 4T_{\oplus}$	$9T_{hash} + 12T_{\oplus}$	$20T_{hash} + 18T_{\oplus}$	$3T_{hash} + T_{\oplus}$
Lee (2013)	$4T_{hash} +2T_{\oplus}$	$11T_{hash} + 11T_{\oplus}$	$4T_{hash} + 4T_{\oplus}$	$19T_{hash} + 16T_{\oplus}$	$11T_{hash} + 6T_{\oplus}$
Chien *et al.* (2002)	$T_{hash} + 2T_{\oplus}$	$T_{hash} + 2T_{\oplus}$	$4T_{hash} + 4T_{\oplus}$	$6T_{hash} + 8T_{\oplus}$	Not supported
Hwang *et al.* (2002)	$2T_{hash} + 2T_{\oplus}$	$T_{hash} + 2T_{\oplus}$	$2T_{hash} + 2T_{\oplus}$	$5T_{hash} + 6T_{\oplus}$	$T_{hash} + 2T_{\oplus}$
Sun (2000)	T_{hash}	$T_{hash} + T_{\oplus}$	$2T_{hash} + T_{\oplus}$	$4T_{hash} + 2T_{\oplus}$	Not supported
Lee *et al.* (2002)	$2T_{hash} + 2T_{\oplus}$	$3T_{hash} + 3T_{\oplus}$	$2T_{hash} + 2T_{\oplus}$	$7T_{hash} + 7T_{\oplus}$	$2T_{hash} + 2T_{\oplus}$

Conclusion

In this study, we designed a new password authentication scheme based on the hardness of inverting one-way hash function. The proposed scheme requires no verification table in order to authenticate users. The scheme also provides a synchronised system clock (timestamp) that could block intruders from penetrating the verification phase wherein a time-change by an intruder fails the authentication step. Furthermore, analyses showed that the scheme can withstand common possible cryptographic attacks including impersonation attack. The scheme also allows users to freely change their passwords. Finally, we demonstrated lower time complexity, communication and computation costs in each phase of the proposed scheme compared to the chosen four schemes.

Acknowledgment

The author's feel grateful to the anonymous reviewers for their valuable suggestions and comments to improve the quality of the paper and would like to thank the editors of the journal from the core of the heart.

Funding Information

The authors acknowledge the Malaysian Ministry of Higher Education and Universiti Kebangsaan Malaysia for funding this academic research under the grant no. GUP-2017-089.

Author's Contributions

Muhammad Helmi Ali: Designed the algorithm of this password authentication scheme, provided efficiency performance of the algorithm and prepared the preliminary version of this manuscript.

Eddie Shahril Ismail: Prepared security analysis of the algorithm and provided major modification and correction for final preparation of this manuscript.

Firdaus Mohamad Hamzah: Provided thorough revision for an improvement of the manuscript for publication.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Chang, C. C. and J.S. Lee, 2006. An efficient and secure remote authentication scheme using smart cards. Inform. Security Int. J., 18: 122-133. DOI: 10.11610/isij.1807

Chang, C.C. and T.C. Wu, 1991. Remote password authentication with smart cards. IEE Proc. E., 138: 165-168. DOI: 10.1049/ip-e.1991.0022

Chang, C.C. and W.Y. Liao, 1994. A remote password authentication scheme based upon ElGamal's signature scheme. Comput. Security, 13: 137-144. DOI: 10.1016/0167-4048(94)90063-9

Chien, H.Y. J.K. Jan and Y.M. Tseng, 2001. A modified remote login authentication scheme based on geometric approach. J. Syst. Software, 55: 287-290. 10.1016/S0164-1212(00)00077-7

Chien, H.Y., J.K. Jan and Y.M. Tseng, 2002. An efficient and practical solution to remote authentication: Smart card. Comput. Security, 21: 372-375. DOI: 10.1016/S0167-4048(02)00415-7

ElGamal, T., 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inform. Theory, 31: 469-472. DOI: 10.1109/TIT.1985.1057074

Evans, A. Jr., W. Kantrowitz and E. Weiss, 1974. A user authentication scheme not requiring secrecy in the computer. Commun. Assoc. Comput. Mach., 17: 437-442. DOI: 10.1145/361082.361087

Harn, L., D. Huang and C.S. Laih, 1989. Password authentication using public-key cryptography. Comput. Math. Applic., 18: 1001-1017. DOI: 10.1016/0898-1221(89)90028-X

Hwang, M.S., 1999. A remote password authentication scheme based on the digital signature method. Int. J. Comput. Math., 70: 657-666. DOI: 10.1080/00207169908804781

Hwang, M.S., C.C. Lee and Y.L. Tang, 2002. A simple remote user authentication scheme. Math. Comput. Model., 36: 103-107. DOI: 10.1016/S0895-7177(02)00106-1

Hwang, T.Y., 1983. Passwords authentication using public-key encryption. Proceedings of the International Carnahan Conference on Security Technology, (CST' 83), Zurich, Switzerland, pp: 35-38.

Kumar, M., 2010. A new secure remote user authentication scheme with smart card. Int. J. Security, 10: 175-184.

Lamport, L., 1981. Password authentication with insecure communication. Commun. ACM, 24: 770-772. DOI: 10.1145/358790.358797

Lee, C.C., M.S. Hwang and W.P. Yang, 2002. A flexible remote user authentication scheme using smart cards. ACM Operat. Syst. Rev., 36: 46-51. DOI: 10.1145/567331.567335

Lee, Y.C., 2013. Weakness and improvement of the smart card based remote user authentication scheme with anonymity. J. Inform. Sci. Eng. 29: 1121-1134.

Lennon, R.F., S.M. Matyas and C.H. Meyer, 1981. Cryptographic authentication of time-invariant quantities. IEEE Trans. Commun., 29: 773-777. DOI: 10.1109/TCOM.1981.1095067

Li, X., J. Niu, M.K. Khan and J. Liao, 2013. An enhanced smart card based remote user password authentication scheme. J. Netw. Comput. Applic., 36: 1365-1371. DOI: 10.1016/j.jnca.2013.02.034

Liao, I.E., C.C. Lee and M.S. Hwang, 2006. A password authentication scheme over insecure networks. J. Comput. Syst. Sci., 72: 727-740. DOI: 10.1016/j.jcss.2005.10.001

Liu, Y., C.C. Chin and C.C. Shih, 2017. An efficient and secure smart card-based password authentication scheme. Int. J. Netw. Security, 19: 1-10.

Paar, C. and J. Pelzl, 2010. Understanding Cryptography: A Textbook for Students and Practitioners. 1st Edn., Springer-Verlag Berlin Heidelberg, Berlin, ISBN-10: 978-3-642-04101-3, pp: 372.

Pooja, M. and K. Pramav, 2016. Biometric based remote login password authentication scheme using smart card. Int. J. Adv. Eng. Res. Develop., 3: 134-140. DOI: 10.21090/IJAERD.031122

Rajaram, R. and P.M. Amutha, 2012. An efficient password authentication scheme for smart card. Int. J. Netw. Security 14: 180-186.

Sood, S.K., 2012. An improved and secure smart card Based dynamic identity authentication protocol. Int. J. Netw. Security 14: 39-46.

Sun, H.M., 2000. An efficient remote use authentication scheme using smart cards. IEEE Trans. Consumer Electron., 46: 958-961. DOI: 10.1109/30.920446

Thandra, P.K., J. Rajan and S.S.V.S. Murthy, 2016. Crptanalysis of an efficient password authentication scheme. Int. J. Netw. Security, 18: 362-368.

Wang, S.J. and J.F. Chang, 1996. Smart card based secure password authentication scheme. Comput. Security, 15: 231-237. DOI: 10.1016/0167-4048(96)00005-3

Wu, T.C., 1995. Remote log in authentication scheme based on a geometric approach. Comput. Commun., 18: 959-963. DOI: 10.1016/0140-3664(96)81595-7

Xu, J., W.T. Zhu and D.G. Feng, 2009. An improved smart card based password authentication scheme with provable security. Comput. Standards Interfaces, 31: 723-728. DOI: 10.1016/j.csi.2008.09.006

Yoon, E.J., E.K., Ryu and K.Y. Yoo, 2005. An improvement of Hwang-Lee-Tang's simple remote user authentication scheme. Comput. Security, 24: 50-56. DOI: 10.1016/j.cose.2004.06.004