Indian Journal of Science and Technology, *Vol 9(7), DOI: 10.17485/ijst/2016/v9i7/85017, February 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

EAM: Architecting Efficient Authentication Model for Internet Security using Image-Based One Time Password Technique

A. Jesudoss[1*] and N. P. Subramaniam[2]

[1]Faculty of Computer Science and Engineering, Sathyabama University, Chennai – 600119, Tamil Nadu, India; jesudossas@gmail.com, jesudoss.mca@sathyabamauniversity.ac.in
[2]Department of EEE, Pondicherry Engineering College, Puducherry – 605014, Tamil Nadu, India; npsubbu@yahoo.com

Abstract

In the era of Internet world, passwords are essential to protect our data and the application. Relying on a simple plaintext password would lead to vulnerability. Apart from choosing strong passwords, the authentication model plays a crucial role in Internet Security by protecting the web applications efficiently from various security attacks. **Objectives:** The aim of the paper is to provide an Efficient Authentication Model and adopt a new technique in generating a dynamic salt from the client. Hence it protects Internet applications from five types of security attacks, namely password-guessing attack, keylogger attack, replay attack, streaming bots, and screen-capture attacks. **Methods:** The One Time Password (OTP) based image-selection enables the user to protect the Internet application from streaming bots, keyloggers and screen-capture attacks. **Findings:** The architecture has been designed efficiently to minimize the number of transactions between the client and the server. The dependency on hardware devices for authentication can be completely eradicated by using Efficient Authentication Model (EAM). Hence, the authentication is well-suited for Internet applications requiring higher levels of security. **Application:** It is a single solution for multiple security problems with minimal cost and highly secured with improved performance. Hence, it can be implemented by banks, financial organization, etc. where security is very significant.

Keywords: Authentication – Keylogger – Replay Attack – Password-Guessing Attack –Streaming Bots – Screen-Capture Attack

1. Introduction

When advanced technologies or complex algorithms are used for an authentication model, it adds more overhead to the model[1]. Therefore usability and security should be balanced efficiently. An efficient authentication model is not the one that uses latest technologies or the one that uses expensive hardware devices or complex algorithms. It is the one that provides utmost security in a simple usable way with a less time, resource and effort. For instance, nowadays banks are using mobile One Time Password (OTP) for secondary authentication. The idea here is simple; it is just sharing a secret over the mobile. In fact, there is no complex algorithm involved or expensive hardware devices used but it could eliminate the attacker even if the primary password is stolen. Though the idea is simple, it is very effective and efficient.

In the Internet World, there are millions of tools available for breaching the security. It is essential to know how these tools and concepts are used in breaching the Internet Security. For Instance, Screen Scraper is a software that takes the screenshot of the victim computer periodically[2]. Keyloggers are common among Internet users for share the computer resource such as memory and processor with other running programs. It does not attract user's attention as it remains silent and invisible. It is clearly

*Author for correspondence

shown[3] how unprivileged processes run keyloggers and is designed completely as a black-box model. By correlating the input and the output of the keylogger, the behavior of the keyloggers is well-studied. Keylogger has become very common these days. Most of the attacks are keylogger based. A Keylogger helps to impersonate a legitimate user. To mitigate the keylogger attack, the keyboard Interrupt Vector Table (IVT) has to be modified[4]. The authors opine that designing an authentication protocol is quite challenging. Further, the author says that when the human beings are involved in authentication process, it affects the usability of the authentication. In fact, they have their limitations in computation and in memorizing the information.

Graphical passwords are alternatives to textual ones. It is easy to remember graphical passwords. The vulnerabilities that exist for the textual password are also applicable to graphical passwords[5]. Therefore, we need the combination of image-based and textual information for providing the password. The proposed work focuses on this aspect. In Passfaces scheme, a recognition-based graphical password scheme, the user has to select a set of faces for each round[6]. The set of faces selected for each round will be used for creating a password. In order to login successfully, the user should select correct faces in every round. The set of images are the same but displayed in different positions. It is prone to password-guessing attacks. In recall-based scheme, the user has to draw the password. These graphical password schemes are vulnerable to shoulder-surfing attack.

Honey Encryption technique is a technique that generates some message even though the key or the password entered is wrong[7]. Therefore, it misleads the attacker and prevents them from attempting further. The methodology[8] indicates how the password-guessing attack can be handled. They also says that the IP addresses should not blacklisted, for it will consume huge memory or sometimes even legitimate users may get caught in the blacklist. The authentication scheme[9] points out how the password can be obtained using offline password guessing attack. In this case, the password is simply hashed and hence it is easy to run brute-force attack on it. The proposed work shows that the salt should be appended to the password to make it very complex.

The three-factor security protocol[10] explained about securing the Universal Serial Bus (USB) devices. It is based on Elliptic Curve Cryptography (ECC) smart card and biometric system. Elliptic Curve Cryptography (ECC) has proved to be better than public key cryptography as it has smaller key size. It showed that the system has vulnerability to password-guessing attack, the Denial of Service (DoS) attack, and the replay attack. It also shows that the hardware devices are not a solution for avoiding attacks. The biometric information can be captured at the fake banking terminals[11]. Therefore, Biometric authentication has been proved to be vulnerable as the biometric information is used for impersonation. The author insists the need for hiding information or credentials. A part of the authentication process must be hidden from the prying eyes. The hardware devices cannot be trusted fully for the purpose of authentication or cryptography. Hardware Trojan Attacks[12] showed that implementation of hardware devices does not ensure security. Therefore, the above points about the biometric authentication and hardware devices for authentication insist the facts that the hardware devices are also vulnerable to various attacks. Investing money on hardware devices will not ensure security at any cost. This motto had been the basic logic for developing the proposed work based on human intellectualness.

A CopyCat attack is an attack which simply captures the packet and forwards it with or without minor changes in the packet[13]. It showed how the copycat attack can be handled. It just sends the replicas of already transmitted packets so the router will be confused with the replicas. Elastic Password Authentication Scheme[14] introduced a new concept called Wheel Lock which protects against shoulder surfing attack, brute force attack, etc. The security functions are based on the technology as well as the human factors[15]. Once again, authors have insisted that human factors must have higher priority than the technological factors.

Noisy Password Security Technique[16] indicated how different noise patterns can be inserted into One Time Password (OTP) scheme and makes it more effective than the traditional One Time Password (OTP). This alleviates shoulder surfing attack and the password is dynamic and unique per transaction. OTP-Based Two Factor Authentication Scheme[17] provided an alternative method for providing One Time Password (OTP). The One Time Password (OTP) is generated from the initial seed value and is dynamic every time. Two-Factor Smart Card-Based Authentication Scheme[18] suggested another multifactor authentication model that uses smart card. The authentication scheme for remote access in intelligent home networks[19] proposed a One Time Password-based

(OTP-based) user authentication scheme for home networks. In this scheme, two-factor authentication is provided and key distribution is made without any shared secret. Integrated User Authentication Method[20] provides a solution for managing authentication information over multiple clouds while balancing the safety and convenience of the user. The latest Freak Attack[21] is an attack on Secure Sockets Layer /Transport Layer Security (SSL/TLS). It enables an attacker to intercept Hypertext Transfer Protocol Secure (HTTPS) traffic. This showed that there is no end to wrestling between safeguard and attack. Security is a never-ending and ever-emerging process. The authentication scheme for wireless networks[22] provides method for protecting MITM attacks by verifying physical location of the device and by using radio time arrival and ultrasonic range.

2. Motivation

While developing a security authentication model, many crucial factors should be considered and incorporated into the system. The following factors are considered in this security model. They are listed below.

1. To provide single solution for various security problems.
2. To provide cost effective solution by eradication of hardware devices for authentication
3. To improve the performance of the security model without reducing the level of security.

A Salt is just a random data that is appended to the password to make the password unique for every transaction. The salt is usually generated on the Server. Hence, the salt generated should be communicated to the client for sending its user credentials. This involves an additional transaction between the server and the client. Therefore in the proposed model, the salt is generated on the client and helps to reduce the transactions between the server and the client. The salt is generated per-user, per-password and per-transaction basis. Hence, it is unique for every transaction and helps to avoid replay attacks. Moreover, a good security system is not the one which uses complex technologies and hardware resources. But it is the one which makes things very simple, cost-effective, usable, and secured as well. Therefore, it is necessary to design a security authentication model with the above-mentioned features.

3. Proposed Work

First, the client and the server should be synchronized with time using Network Time Protocol. The client enters the Uniform Resource Locator (URL) on the browser and the appropriate web application is displayed on the browser. While rendering the web page, the server displays five images on the web page. Any number of images can be chosen to increase the level of complexity. In order to simplify it, only five images are considered for the proposed work. The user has to enter the username and the password and also has to choose an image from the pool of images. This image selection process serves like a One Time Password (OTP) from the client and the image selected is used for the generation of the salt on the client. Each time, different images are chosen by the client. Hence, salt will be unique for every transaction per user. Depending upon the image being selected, the server will be able to validate the user. The password is static but the image chosen is dynamic as it is different every time. The server will not know in advance or will not be able to predict what the user would have chosen. But still the server is capable of validating the user. The client sends the sequence of images being displayed along with the user credentials to the server. But it need not specify the image being selected by the user. The client indicates it to the server by means of the salt value added to the password. The server on seeing the image sequence given by the client knows what should be the correct image used for the salt based on the information available against the userkey in the database, i.e. secret image and secret value information. Therefore, the client does not include the image selected by the user. The client does not validate the information entered by the user. It simply passes it to the server for validation. So if the client-side code reveals the information about the image being selected, it will not be helpful for the hacker. Because immediately after selecting the image, the user will click submit button. Therefore there is no use for the hacker in seeing the image being selected by the user. The server, on receiving the request from the client, will decrypt the message and fetch the hash value. Then the server will use image sequence to validate the user.

The Efficient Authentication Model (EAM) in Figure 1 depicts the security architecture of our proposed work. The transaction between the server and the client is reduced. Therefore, authentication is made quickly.

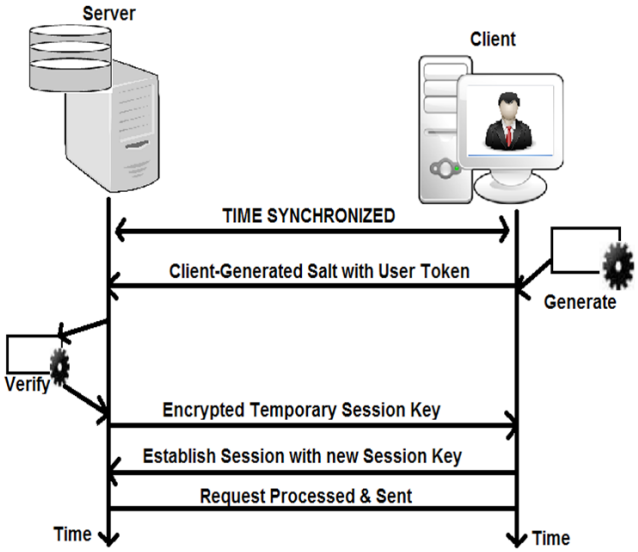

Figure 1 Architecture of Efficient Authentication Model (EAM)

3.1 Generation of Salt and Verification of Salt

The salt can be dynamic when it is generated from the server. If it is generated from the client, it is usually static. The proposed work generates the salt on the client side to reduce the number of transactions between the server and the client. It attempts to generate the salt using a very unique method based on the image selection by the user. Let us assume that the five images to be displayed are any five flowers. The user has to choose one of these flowers. Whatever the image is chosen by the user, it is used for salt generation on the client. At the time of registration, each user has to specify a secret image from the pool of given images and a secret value in the range of 1 to n, where n is the total number of secret images. The number of secret images can be increased in order to raise the level of complexity. For making the concept simple, five images and secret value in the range 1 to 5 are selected. When the images are displayed in a shuffled manner, the user has to choose an image that is "x" steps away from the secret image of the user. For this purpose, the formula adopted in Circular Queue can be used, i.e. (CURRENT NODE + MOVE) % NO.OF_NODES = NEW NODE. CURRENT NODE is the secret image of the user. MOVE is the secret value chosen by the user so that the user has to make those many steps move forward to select the image for any transaction. NO.OF_NODES specifies the total number of images displayed to the user. NEW_NODE is the image selected by the user. If the secret flower is Hibiscus, then Hibsicus may be displayed either at the beginning, middle or at the end. The secret flower of the user has five probabilities of locations to appear for the given example. Therefore, the image chosen by the user will vary every time. So it will protect against screen-capture attack. By monitoring the user's screen, it would not be possible to find the secret image of the user.

4. Implementation

The proposed strategy has been implemented using Java version 7.0 (i.e.jdk -7u45-windows-i586) and Tomcat version 7.0.47. Using Java Cryptography Extension (JCE), Advanced Encryption Standard (AES) 256 bit has been implemented. Secure Hash Algorithm (SHA) 256 is used for the Cryptographic Hash Function. The steps given below demonstrate the implementation of the work:

Step 1:The server and the client are time-synchronized using Network Time Protocol (NTP).

Step 2: The client requests for the login page to the server.

Step 3: The server loads the login page with five images, username and password fields.

Step 4: Based on the image being selected, an appropriate 256-bit salt will be generated by client.

The salt generated by the client serves like a manual One Time Password (OTP) and it need not be entered by the client.

Step 5: The salt is appended to the password and it is hashed using SHA256 algorithm.

Step 6: The timestamp t1, hash value h1 (i.e. hash of password plus salt), image sequence, network address , and the userkey are encrypted using Advanced Encryption Standard (AES) 256 bit algorithm. Like Kerberos protocol, we assume that keys have already been exchanged.

Step 7: The client sends the user token (step 6) along with username to the server.

Step 8: The server checks whether the username is available in the database.

Step 9: If the username is available in the database, then the server decrypts the user token using the appropriate key stored against the username and it validates the user token.

Step 10: The server checks whether the userkey in the user token against the username and userkey in the database.

Step 11: The server checks whether the network address is valid for the given username

Step 12: The server compares the timestamp received with the current time stamp and if it is less than the threshold value, then it is processed further.

Step 13: The server computes the appropriate salt based on the image sequence sent by the server.

Step 14: The password is retrieved from the database and the salt is appended to the password and it is hashed to compare whether the received hash value is equal to the computed one.

Step 15: If they are equal, the user is authenticated successfully and an encrypted session key will be sent to the client.

Step 16: The client will communicate to the server with the newly generated session key.

This proposed work is a generic authentication model which can be implemented in any application that requires authentication. The application implemented for the proposed work is an online banking application.

5.1 Analysis of Attacks and Proposed Countermeasures by Efficient Authentication Model

The Efficient Authentication Model (EAM) proposed model provides an efficient authentication and protects against five different security attacks such as password-guessing attack, keylogger attack, replay attack, streaming bots and screenshot attack. In this section, we study and demonstrate how the various security attacks are handled by Efficient Authentication Model (EAM).

4.1.1 Password-Guessing Attack

Password-guessing attack is an attack in which the attacker performs offline password-guessing. For instance Kerberos protocol obtains the Ticket Granting Ticket (TGT) from the server without providing its identity. Therefore a packet is given to an illegitimate user. In the proposed work, the client or any attacker cannot obtain anything from the server without proving their identity. After proving the identity only, one will be able to receive encrypted data from the server. Unlike Kerberos, password-guessing attack is not possible.

4.1.2 Keylogger Attack

The keylogger program is usually hidden from the running processes and applications list. Therefore, it is difficult to find whether a keylogger is in process. If the secret password between the parties is completely entered as keystrokes, then it may be tracked using keyloggers. Therefore in the proposed work, a part of the secret is not provided via keyboard. The image selection process escapes from the keyloggers and is efficiently utilized for the generation of the salt.

4.1.3 Replay Attack

Replay attack is an attack in which the packet is captured and replayed by the attacker immediately or some time later. Such captured packets may be modified by the attacker. The user credentials, the network address or the timestamp may be modified by the nefarious user. Therefore in the proposed work, the user credentials provided by the client particularly the image chosen varies every time. Moreover, the packet from the client has the timestamp and network information also. Therefore it is not possible to replay the packet.

4.1.4 Streaming Bots Attack

Streaming Bots attack is an attempt to hack using automated bots, where in the bots check for the vulnerability in the application and exploit the vulnerability. In the proposed work, it is not possible to attack the application using streaming bots since the user has to select the appropriate image from the given images. The bots obviously would not know what image is to be selected. Therefore, it fails during authentication. The proposed authentication model requires some human intellectualness to prove its identity by choosing the correct image.

4.1.5 Screen-Capture Attack

A screen-capture attack is a type of attack in which captures the user's screen periodically and uses such screenshots for launching an attack. A screen-capture attack may show what image the user has selected. Since the image selection is dynamic, it will be of no use to anyone who performs screen-capture attack. He/She may know what image has been used for the generation of salt. But before they performs the attack, the user will submit the credentials and finish the transaction. So for the current transaction, the hacker may see what is being selected by the user but cannot use it.

The above countermeasures clearly show how the proposed work is capable of providing security against various security attacks.

5. Results & Discussion

The Login screen of Efficient Authentication Model (EAM) authentication model is shown in Figure 2 given below. It displays the username and the password textboxes with five images of flowers. The user activities recorded by HT Employee Monitor are clearly shown in Figure 3 given below. The Brute Force Attack initiated using Cain & Abel has been shown in Figure 4 and the Figure 5 shows that the brute force attack is unable to crack the given cipher. The brute-force attacks implemented using Cain & Abel software. Win-RT-Gen software Figure 6 & Figure 7 has been used for creating rainbow tables which are used for performing such brute-force attacks.

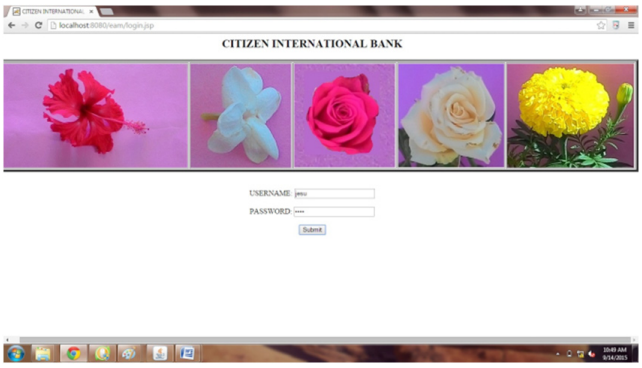

Figure 2 The Efficient Authentication Model Login Screen

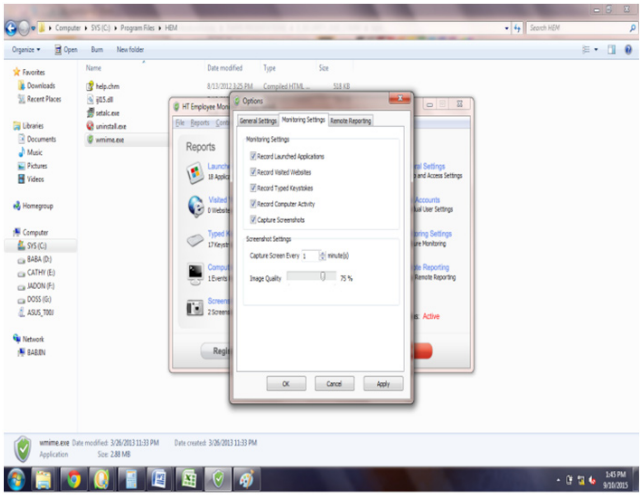

Figure 3 User Activity Recorded by HT Employee Monitor

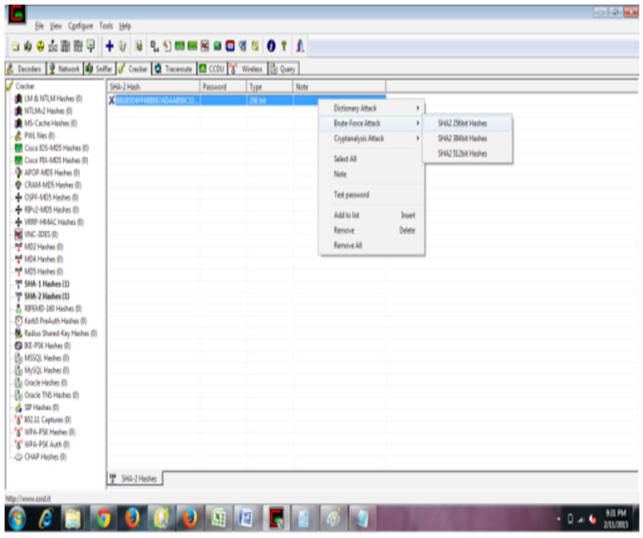

Figure 4 Brute Force Attack by Cain & Abel

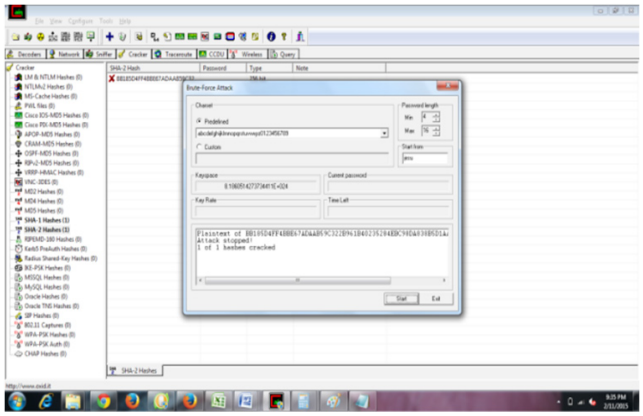

Figure 5 Brute Force Attack Unable to Crack

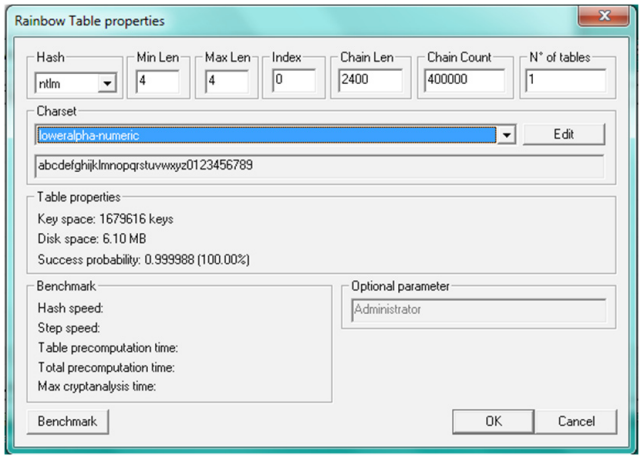

Figure 6 Rainbow Table Creation with 4 characters consisting of alphanumeric

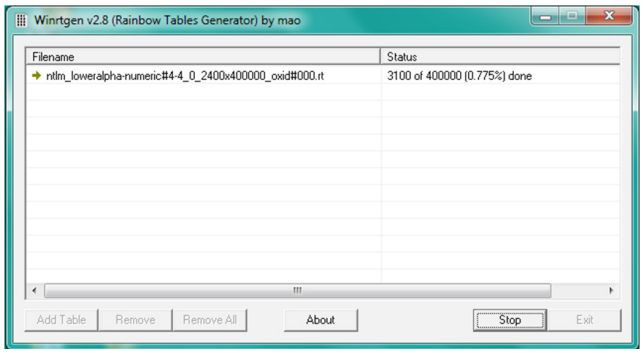

Figure 7 Processing of Rainbow Table

Table 1. Comparison of Efficient Authentication Model with Prevailing Technologies

Types of attack	Kerberos	SSL (TLS)	EAM - Proposed Work	Software Used for Implementation of Attack	Counter measures
Password-Guessing Attack	Vulnerable	Resistant	Resistant	Cain & Abel	Client-Generated Salt
Keylogger Attack	Vulnerable	Vulnerable	Resistant	Refog	Partial Password Entry
Replay Attack	Vulnerable	Resistant	Resistant	Colasoft Packet Builder	dynamic OTP based on Image
Streaming Bots Attack	Resistant	Resistant	Resistant	Not Attempted	Image-Based Question
Screen Capture Attack	Resistant	Vulnerable	Resistant	HT Employee Monitor & Refog	Dynamic Image Selection

5.1 Comparison of Efficient Authentication Model with Prevailing Technologies

The comparison table given in Table 1 shows that Efficient Authentication Model (EAM) Authentication Model is resistant to various types of security attacks. The table also indicates that the proposed work has many advantages over the other authentication models such as Kerberos and Secure Sockets Layer (SSL).

6. Conclusion and Future Work

A good security system is not the merely one which uses complex technologies and hardware resources. But it is the one which makes things very simple, secured and cost-effective as well. Therefore, it is better to rely on simple and effective concepts rather than putting our trust in hardware devices for authentication. It will be simple and cost-effective. Our proposed work proves that the Efficient Authentication Model (EAM) authentication model is simple, secured and cost-effective. It eliminates the need for technological dependency or dependency on hardware resources. This work can be extended by including attacks against database and Structured Query Language (SQL) Injection attacks.

7. References

1. Jesudoss A, Subramaniam NP. Enhanced Kerberos Authentication for Distributed Environment. Journal of Theoretical and Applied Information Technology. 2014; 69:368–74.
2. Sagiroglu S, Canbek G, KEYLOGGERS: Increasing Threats to Computer Security and Privacy. IEEE Technology and Society Magazine. 2009; 28(3):10–7.
3. Ortolani S, Giuffrida C, Crispo B. Unprivileged black-box detection of user-space keyloggers. IEEE Transactions on Dependable and Secure Computing. 2013; 10(1):40–52.
4. Nyang DH, Mohaisen A, Kang J. Keylogging-Resistant Visual Authentication Protocols. IEEE Transactions on Mobile Computing. 2014; 13(11):2566–79.
5. Sadovnik A, Chen T. A Visual Dictionary Attack on Picture Passwords. Proceedings of ICIP '13, Melbourne, VIC. 2013 Sep 15-18; 4447–51.
6. Zhu BB, Yan J, Bao G, Yang M, Xu N. Captcha as Graphical Passwords—A New Security Primitive Based on Hard AI Problems. IEEE Transactions on Information Forensics and Security. 2014; 9(6):891–904.
7. Juels A, Ristenpart T. Honey Encryption – Encryption Beyond the Brute-Force Barrier. IEEE Security and Privacy. 2014 May 11-15; 8441:293–310.
8. Alsaleh M, Mannan M, Van Oorschot P. Revisiting Defenses against Large-Scale Online Password Guessing Attacks. IEEE Transactions on Dependable and Secure Computing. 2012 Jan-Feb; 9(1):128–41.
9. Lee CC, Li CT, Chen CL, Chang RX. On Security of a more Efficient and Secure Dynamic ID-based Remote User Authentication Scheme. IET Information Security. 2014 Mar; 8(2):104–13.
10. He D, Kumar N, Lee JH, Sherratt RS. Enhanced Three-factor Security Protocol for Consumer USB Mass Storage Devices. IEEE Transactions on Consumer Electronics. 2014; 60(1):30–7.
11. Sasamoto H, Christin N, Hayashi E. Undercover: Authentication Usable in Front of Prying Eyes. Proceedings of CHI '08. 2008; 183–92.

12. Bhunia S, Hsiao MS, Banga M, Narasimhan S. Hardware Trojan Attacks: Threat Analysis and Countermeasures. Proceedings of IEEE '14. 2014 Aug; 102(8):1229–47.
13. Feng Z, Ning J, Broustis I, Pelechrinis K, Krishnamurthy SV, Michalis F. Coping with Packet Replay Attacks in Wireless Networks. Proceedings of SECON '11. 2011; 368–76.
14. Yi H, Kim S, Ma G, Yi JH. Elastic password authentication scheme using the Passcell-based virtual scroll wheel. International Journal of Computer Mathematics. 2013; 90(12):2530–40.
15. Adeka M, Shepherd S, Abd-Alhameed R. Resolving the password security purgatory in the contexts of technology, security and human factors. Proceedings of ICCAT '13. 2013 Jan 20-22; 1–7.
16. Alghathbar K, Mahmoud H. Noisy Password Security Technique. Proceedings of ICITST '09. 2009 Nov 9-12; 1–5.
17. Eldefrawy MH, Alghathbar K, Khan MK. OTP-Based Two-Factor Authentication using Mobile Phones. Proceedings of ITNG '11. 2011 Apr 11-13; 327–31.
18. Yang G, Wong DS, Wang H, Deng X. Two-factor mutual authentication based on smart cards and passwords. Journal of Computer and System Sciences. 2008 Nov; 74(7):1160–72.
19. You I. A One-Time Password Authentication Scheme for Secure Remote Access in Intelligent Home Networks. Lecture Notes in Computer Science. 2006; 4252:785–92.
20. Choi JH, Lee1 SH, Kim MK. Integrated User Authentication Method using BAC(Brokerage Authentication Center) in Multi-clouds. Indian Journal of Science and Technology. 2015; 8(25):1–7.
21. Tracking the freak attack website [Online]. Available from: https://freakattack.com/. 03/03/2015
22. Barzegar N, Aminian E. Authentication through Presence in Wireless Networks. Indian Journal of Science and Technology. 2015; 8(25):1–15.