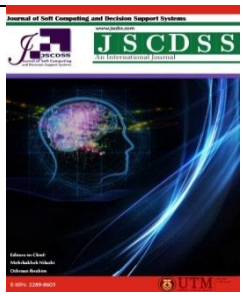
An International Journal
JSCDSS
http://www.jscdss.com
Vol.4 No.1 February 2017: 19-25
Article history:
Accepted 20 February 2017
Published online 25 March 2017

Journal of Soft Computing and Decision Support Systems



E-ISSN: 2289-8603

A Novel Two-Factor Authentication System Robust Against Shoulder Surfing

Mohammadreza Hazhirpasand Barkadehi [a,*], Mehrbakhsh Nilashi [a], Othman Ibrahim [a]
[a] Faculty of Computing, Universiti Technologi Malaysia, Johor, Malaysia

*** Corresponding author email address**: mhhazhirpasand@gmail.com

Abstract

To stop attackers from accessing protected contents of a website or a mobile application, authentication systems with various forms are presented. One of the challenging barriers in nowadays identification systems is unauthorized bystanders. This attack is mostly applicable on many sorts of authentication systems. To fight with unauthorized eyes, many approaches have been proposed. Each one has its own pros and cons. In this paper, the proposed system is a two-factor authentication in conjunction of smart-phone of owner. To disable malicious softwares to key log keystrokes or take screenshot or observers to memorize your hand movement on keyboard or mouse curse on a virtual keyboard, proposed system came up with a novel way to decrease the effect of these attacks.

Keywords: User authentication, Two-step authentication, Shoulder surfing attack

1. Introduction

Authentication systems have been changing through the passage of time. At first, password-based authentications were popular because of its simplicity of remembrance and applicability among people (Shen et al., 2016). Due to the fact of simplicity, valuable pieces of information or assets are protected by simple passwords and this causes insecurity for data owners. The essence of a password is exposed to many easy-to-run attacks such as phishing, social engineering or keylogging technique (Svogor and Kisasondi, 2012). Each of password characteristics such as password length, password composition and password selection reveals different concern which some designers take some or all of them into consideration to force users to choose a strong password (Shen et al., 2016). As a result, authentication systems gradually became more creative. For instance, on-the-fly password policy systems indicate weakness or strength of user chosen password according to the password characteristics as mentioned before.

Later graphical passwords introduced to boost the security strength of systems but still they were vulnerable to shoulder surfing and recording attacks (Prabhu & Shah, 2015). With the advent of new paths in related technologies, mixing password-based systems with other factors such as biometric or ownership increased the assurance of data safety to some extent. Biometric factor has two sub-categories of biometric authentication, physiological and behavioural (Kang et al., 2014). As physiological authentication name implies, it is regarded as measurement and recognition of physical specifications such as fingerprint, iris recognition and on the other hand, behavioural authentication relies on behavioural characteristics of the person such as keystroke dynamics (Bailey, 2014). Physiological examples such as fingerprints are sensitive to spoofing techniques, however, behavioural biometrics are much more suitable for second phase of authentication and need less additional hardware (Chen et al., 2015). Two-factor authentication systems nowadays mostly use a smartphone in industrial scales such as Yahoo or Google because of their availability in our any day-to-day life (Crossman and Liu, 2016; Abdurrahman et al., 2013).

In this study, a two-factor authentication model comprised of a traditional password and a virtual keyboard are presented. The second factor involves a smartphone to cooperate in order to accomplish the process of identification. Often on-screen keyboards are vulnerable to observers and software which are capable of taking screenshot stealthy. The considerable part of this system is neither a real user nor observers can see the on-screen keyboard.

2. Literature review

Current stage of research in this area contains different novel ideas to battle with onlookers and malicious software. As a result, different ideas on different devices and platforms have been implemented so far. For example, Shankar et al. (2016) proposed a system called IPCT which improves mobile authentication systems. In their system, a user must select a series of pictures and for enhancing security aspect, the amount of time that user holds the image button and tapping of fingers on the screen will be

measured. Their scheme may not use password-based technique but still vulnerable to bystanders who can watch the process of authentication for several times. Although tapping multiple times on a black screen, remembering the order of selected pictures within non-randomized matrix is not preventive after several times of observation.

Thinking about something to authorize a client as a valid user for the system might be fanciful. In some new emerging trends, EEG or electroencephalogram device is used to create augmented passwords. For instance (Svogor and Kisasondi, 2012) proposed a system to divide password to smaller sections called PEL. Mixing each PEL of password with attention and/or relaxation signal causes extra security defence mechanism because the order of password is not an issue anymore. It means client can enter the password in any order and this makes the process tough for bystanders. The two main disadvantages of these systems are an additional device is needed and the mental state of clients may vary under different conditions.

Gokhale and Waghmare (2016) presented a shoulder resistant authentication system via graphical password. Their system was a combination of recalled and recognition approach. Users choose some pictures within 25 pictures as their picture password and in the second phase they should select three secret questions and region-of-answer or ROAs in the picture. Each time, question numbers are changed and users must be aware of the questions and answers order. Randomization is the key method to prevent shoulder surfing. This system is vulnerable to malicious software which takes screen shot and repetitive observations.

Another sort of prevention and resistance against bystanders is presented by (De Luca et al.,, 2010) to make the method of entering digits dimmer to bystanders at ATM machines. As authentication takes place, according to three colours associated with letters users enter a series of alphabets which indicates the real numeric passphrase. As the relationship between the entered phrases with the real passphrase is one-to-one relation, the primary weakness can be found via intersection attack after several times of observations but still resistant against one time observation.

In another research which has been conducted by De Luca et al., , 2013), fake cursors used to obscure bystanders sights. According to their findings, the most efficient and effective case is about participation of 16 cursors, no colour and randomized on a virtual keyboard. One of the drawbacks can be found in relative speed of cursors. If active cursors moved faster than others, it would be easier for intruders by looking at video or even onlookers to know which one is an active cursor.

Lee and Nam (2013) presented a challenge and response method to make simple PIN digits more robust to shoulder surfing attack. The challenge is to use random mapping between PIN and alphabets to users. This approach can expose the system to recording and intersection attack after several observations.

For the remainder of this paper, the proposed system is divided into smaller sections to explain its procedure and evaluate its security and usability to achieve the best trade-off. In Section 3, our proposed system is briefly explained and section four describes system implementation. An overview of performance metrics, security analysis and usability study are described in Section 4.

3. Proposed system overview

Due to the installation of malicious software which is capable of accomplishing different complicated tasks such as recording keystrokes or taking screenshots in predefined conditions or CCTV cameras and physical interference by human would make authentication systems less secure. Providing biometric metrics for identifying users' identity is also costly in terms of expenses and usability. In addition, the accuracy of biometric systems in some cases can be challenging due to different environmental or physical condition.

This system is capable of breaking the taboo of keyloggers and physical bystanders. To achieve this goal, a mobile application is needed to be installed on the user's smartphone. This application acts as a mirror to the user while the user is trying to enter his second factor of authentication. In fact, the mobile application does nothing itself, except providing some data which user cannot see in his browsers. In the following sequence diagram (see Fig. 1), it can be seen that how the process of authentication between client–server works.

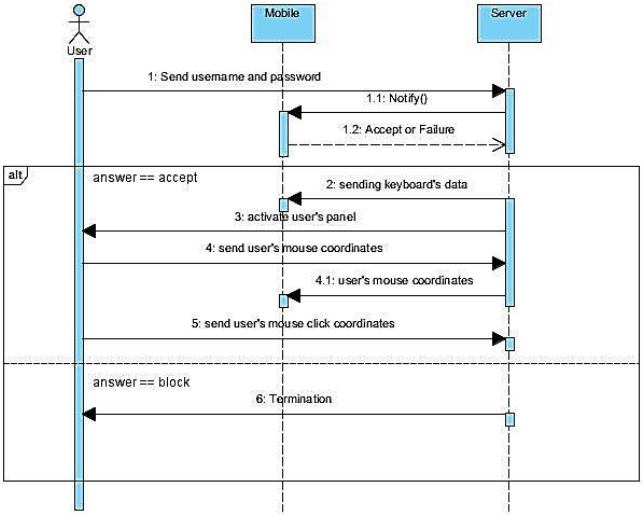


Fig. 1. Authentication process

To make sure that mobile application is functioning effectively without unauthorised installation on other smartphones or cloning, IMEI of the smartphone of the user will be checked every time that application is opened. What is more, every successful try in the first phase will be sent to the user's mobile phone via notification to notify him there is a new and incomplete authentication. Users are capable of terminating the authentication or accepting machine's address to carry on the authentication process within the application. Authentication systems are also vulnerable to man-in-the-middle attack and for this reason, all communications are transmitted via SSL/TLS channel between browser-server and mobile-server. That is to say,

An International Journal JSCDSS E-ISSN: 2289-8603

this paper does not cover approaches concerning man-in-the-middle attack.

The proposed system has built for web authentication purposes and it comprises two steps of entering user's credentials for verification purposes. The system works as follows:

- *Registration Phase*

i. A user chooses his username and password. While password must be according to password policy of the system.
ii. Afterward, the user must select another password which we call it second-pass.
iii. An application should be installed and complete the required steps to be ready for functioning.

- *Login Phase*

i. For the first step, the user should enter his username and password.
ii. If the username and password are correct, a notification will be sent to user's smartphone.
iii. User will decide whether the received Login process is eligible or not. If yes, a panel will be shown to the user and user can see a random keypad including one special character. If the user chooses No, the authentication for that machine will be terminated.
iv. If yes was chosen by the user on his phone, the white box on the webpage will be activated and he can move his mouse cursor and observe his mouse cursor in the application.
v. Then user clicks on the white area for at least 7 times to complete his second-pass.
vi. If user submits and second-pass will be correct, user will be authenticated correctly.

4. System procedure

In this section, two phases of proposed system will be discussed in details. Then how system is implemented is described.

- *Registration Phase*

i. User enters his username and system will check its availability. Then he must choose a password contains alphanumerical and special characters.
ii. User moves to second step, and he chooses second-pass. During registration it is assumed that the machine is secured enough and nothing is stolen or there is no onlooker. Second-pass length must be at least 6 digits plus one special character. Special character in this system is defined to prevent ready-made dictionary attack against numeric password.
iii. User should download and install it on his smartphone. Our default platform for this research is android OS.
iv. After installation, user will enter his username and password and then he will be redirect to the second phase of authentication. User is asked with the second-pass. If both authentication factors are correct, IMEI of the device will be sent to the server and application authenticates the user for further logins via the reverse of his second-pass. He can change the password of mobile application later if he needs.

- *login phase*

i. In the first step, user will surf the web page and enters his username and password. For entering password, virtual keyboard is an optional feature which can be used according to the user's decision. Before submission, browser asks him whether it is allowed to retrieve his location or not.
ii. If both items are correct, the server will send a notification via Firebase Clouding Messaging FCM of Google. The user will be notified by notification and when the application opens up, he will confront with an active login which presents details such as IP address, browser version, Date and Time and geographical position of the machine to the user. In this moment, a user either chooses block or accept. If user selects block, the IP address of that machine will be suspended and the attacker cannot continue authentication process. If he selects accept, a white box will be active in the webpage for selecting second-pass but neither user nor bystanders can see anything inside the white box which was supposed to be a virtual keyboard.
iii. Then user can hold his mobile phone in a way or angle that nobody sees the shuffled keyboard. The randomized keyboard contains 0-9 and one special character.
iv. When user moves his mouse cursor within that area, his mouse location will be streamed to the server via node.js technology. Then he can observe his cursor in the application.
v. Wherever he clicks, in fact, he is selecting one of his second-pass characters. When he submits the second step, application's connection to the server for streaming will be disconnected.
vi. At last, if the second-pass is correct, user will be verified by the system to his control panel.

4.1 Randomness of keyboard

To get over the security flaws, and enhance the system security, virtual keyboards are used to prevent keylogging attack while this method exposes user's credential to bystanders, as the same time, some say a good solution to both of aforementioned threat is to use OTP (one time

An International Journal JSCDSS E-ISSN: 2289-8603

password) (Maheshwari and Mondal, 2016). OTP can have some issues due to hardware dependency and weak signal in some areas might add some barriers. One advantage of the second-pass virtual hidden keyboard is its randomness. Usually, virtual keyboards do not shuffle their keys and this can help attackers to records the clickable spots on the screen or even bystanders can observe what the password is at an ideal distance. In this work, the keyboard cannot be seen by bystanders or even by the client. If so, recording the clickable spots cannot be advantageous due to the randomness of the virtual keyboard. As a result, ten digits and one special character will produce Fact (11) = 39916800 randomness to mitigate any chances of guessing where the keys are.

4.2 implementation phase

For maintaining usernames and passwords, and their login history details MySQL database is responsible for it (see Fig. 2). The database diagram of the proposed work is the following diagram. The system stores every transaction by recording its date, time, latitude and longitude of user's session in user's login_history table. Each user can set one mobile phone as their device and it is assumed that they can change their mobile phone device in their control panel.

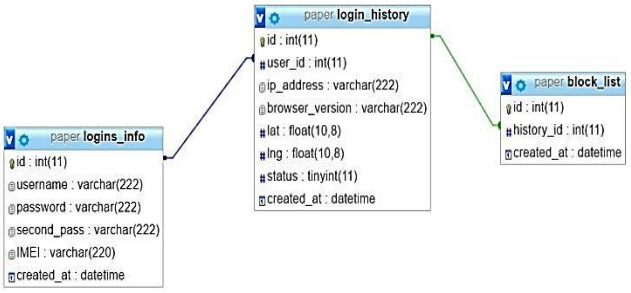
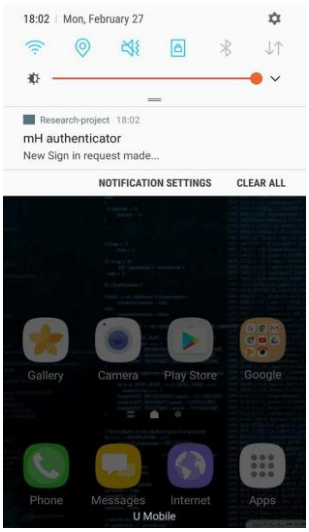
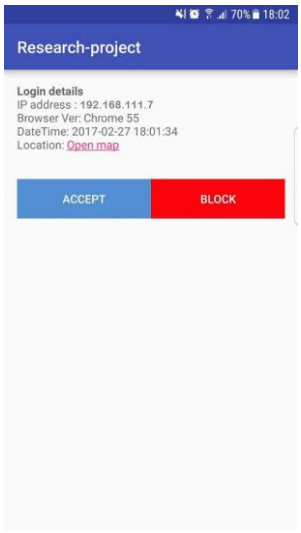


Fig. 2: ER diagram of proposed system's database

Node.js is one of the most widespread and well-accepted programming languages over last years which uses an event-driven, non-blocking I/O model that makes it lightweight and efficient between server and client, for the sake of scalability the back-end of the proposed system has been developed by Express.js framework (Haque et al., 2016). At first, for passing the first step of authentication users are confronted with such a screen. If username and password are correct and validated on the server, then a notification will be sent to the user's smartphone as Fig. 3 presents that user must make decision whether he wants to continue this active authentication or not. The IMEI of client's smartphone is available during the registration period. As long as the user's smartphone does not have any access to the Internet, the next step of authentication wouldn't be active for him.



(a) Notification of new login



(b) informing user of the details

Fig. 3. Implementation phase

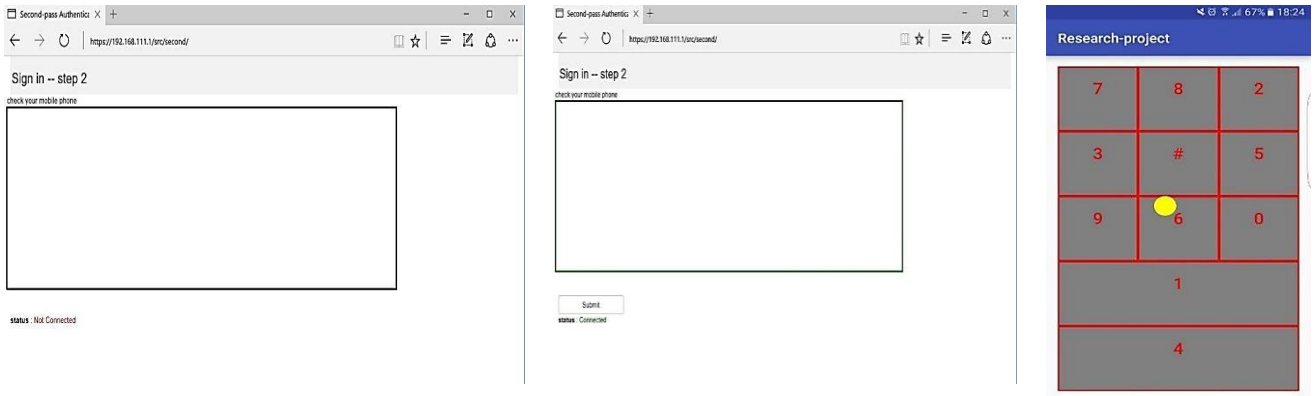
To continue and validate the process of verification, the user must apply the correct decision of him/herself whether to accept or block the pending login request. If the user selects accept button, he will face a shuffled keyboard and if he moves his cursor on web page within the specified box, he is able to observe his cursor in his mobile application by streaming mouse cursor location; it is worth mentioning that in order to achieve better privacy, cursor is invisible within the green box. This task is done by Socket. IO enables real-time bidirectional event-based communication and works on every platform ("Socket.IO," 2017). Fig. 4 shows that user's mouse cursor X and Y will be streamed to the server after user's confirmation on smartphone and then it will appear as a yellow circular indicator on the user's smartphone.

Consequently, the places that the user clicks indicate the real second-pass to the system. In the second step, the only noticeable thing is that the client's computer will send coordination of the user's mouse cursor to the server. Whenever user finishes his input, he can submit and tell the server to verify him. In the case of getting the second-pass and verify it as a correct one, then the server would assign a session to the current user and redirects him to the control panel page.

5. Evaluation of the system

In this section, evaluations are scattered in different forms. First, security analysis and adversary models will be applied to the system. Secondly, timing analysis and usability study will be conducted on 12 participants and consequently results are measured to determine system's applicability.

An International Journal JSCDSS E-ISSN: 2289-8603



(a) **(b)** **(c)**

Fig. 4. Snapshots of (a) before making decision on mobile application, (b) in case of selecting accept button and (c) mobile application side, client can see his cursor movement

5.1 security analysis

There are always some risks which endanger authentication systems. In the following list, we consider discussing these security threats, although we mainly focused on shoulder surfing and malicious software.

Replay attack: the traffic between computer and server can be captured in the first phase of authentication. Meanwhile, if the attacker replies the captured traffic of the second step between computer and server it would not be successful due to wrong hot-spot click points. Each time, the clickable points for the correct second-pass would be different from previous tries because of the randomized virtual keyboard. Moreover, if the user does not accept the first step in his mobile application, sending the second step data does not make sense to the system. Reply attack cannot be done on mobile application data, while each successful login alert contains a security token which is generated randomly.

Stolen phone: a sharp growth in number of mobile users has resulted in more smart phone theft and consequently more investigation to find or retrieve private data (Chakraborty et al., 2016). If the attacker gained access to the phone in any way, and we assumed that he knows first steps credentials, he would get the notification but the application would be PIN protected. If the attacker were sophisticated to figure it out or already knew the PIN, another obstacle called second-pass would exist to slow down the process of a penetration. In that case, he would be able to accept the incoming login, but still, he would need to trial and error the second-pass whose least minimum length is seven characters.

Man-in-the-middle-attacks: to intercept the communication between parties (client and server) and (mobile and server), an attacker must run SSL-strip attack to deceive the victim to go through the attacker's machine. The attacker cannot intercept the notification because it is securely sent through FCM (Firebase Cloud Messaging) (Kwak, Liu, Kim, Nath, & Iftode, 2016) to user's smartphone. In the second phase, the attacker each time in case of successful man-in-the-middle attack only gets different coordinates due to different browser's size and shuffled keyboard. If so, it is infeasible to guess or calculate the second-pass.

Keyloggers/screen shot attack: malicious programs such as those programs which record clients keystroke only can overcome in the first phase. Still getting into the second phase requires confirmation of user on the mobile side. Taking screenshot of the second step of verification does not help attackers to obtain any direct or indirect clues regarding second-pass except the length of second-pass which cannot be assigned less than 7 characters.

In case of knowing second-pass: if the second-pass either has been shoulder surfed or disclosed in any way, an attacker cannot enter into the system. This is because he cannot see what arrangements of keys are for the next time to click on correct spots and apart from this, the first step needs to be verified by the owner of the account through the mobile application.

Online password guessing attack: an attacker can guess the length of second-pass via different ways such as recording the number of the mouse click or listening to the mouse click, still, he must be approved within the mobile application. In the case of wrong second-pass input, the user will be redirected to the first step. In the second step, the keyboard is shuffled and the same coordinates will produce another password while the attacker cannot send them at once because the system will check whether mouse moves and select the received areas or not. In the previous assumption, the attacker has physical access to the mobile application.

To further validate the system, a test designed to evaluate whether the current system is resistant to bystanders or not. The test has been done by six participants and randomly the other six members observed in different angles. Three common angles which usually happen in real life have been selected to measure the resiliency of the system. Afterward, six observers were asked what they could see, the observation results were satisfying enough. Results clearly show that second-pass

An International Journal JSCDSS E-ISSN: 2289-8603

could not be revealed by any bystanders. We leave a deep analysis of more types of experiments of this type as a future work.

Fig.5. Sample of bystander's positions and user

5.2 proposed system performance

A well-designed system is meant to be fast enough, especially authentication system. If the system is slow or requires many steps to be done, users' satisfaction toward the system might decrease due to time-consuming or slow performance. Often complex systems add some overhead to users' perceptions of how a system could be used correctly and consequently this leads to higher false rate results.

The first test was done regarding authentication speed. This test measures the time regardless of whether authentication was successful or not. All second-pass in this test was set to seven-length and with different username and same password length and characters. The equality of length of second-pass and first password among all 12 users was assumed to measure authentication speed life cycle more accurately and typing or clicking more spots would not interfere in results. Our participants have been divided into two different groups and a brief explanation was given to them about how system works. The first group members are not computer science students and do not spend much time with the Internet per day as the second group does and the members are computer science students.

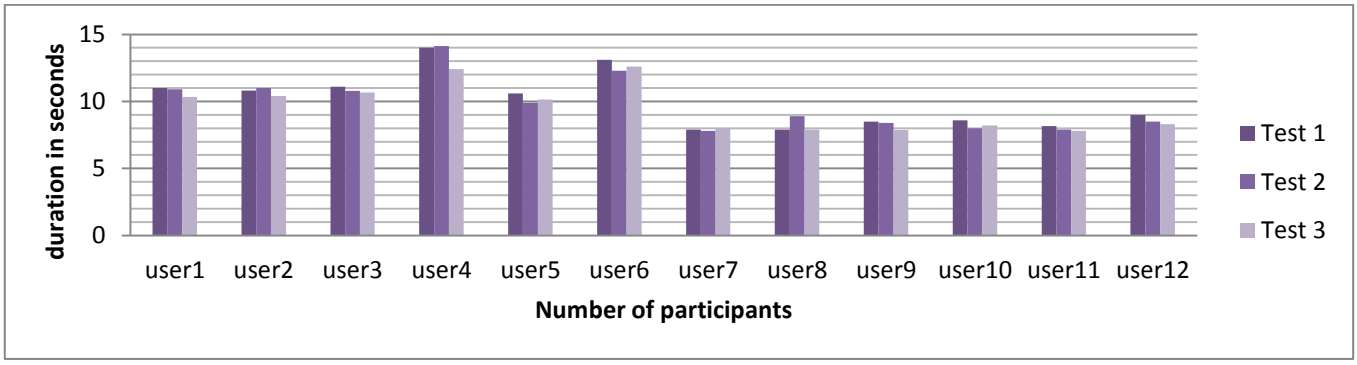


Fig. 6. Authentication speed test

As it can be observed in Fig. 6, each person does the experiment for three times to measure the authentication speed in a more reliable way. First group average authentication time was higher than the second group due to their lower skills related to working with computer and smartphone at the same time. In the third attempt, most members performed it better than their first attempt and the main reason of taking long for users is to move the cursor by their hand and find it on shuffled keyboard on their phone. Because of this randomness and synchronising between computer mouser's cursor and finding it on smartphone screen, perhaps the duration would not decrease significantly after long interaction with the system.

In this regard, out of 36 attempts, 32 attempts were successful and 3 attempts were unsuccessful. An average error rate about 11% achieved and this is supposed to be moderately improved if they continue using the system. Moreover, users were asked to fill an e-questionnaire at the end of their tests. Questions of e-questionnaire were designed to evaluate the system usability from users' perspective. The number of questions was limited to three questions to increase their attention while they want to answer. The questions are as follows:

i. Q1. How easy the system is it to remember?
ii. Q2. Is this system efficient to hide your password from bystanders in real world?
iii. Q3. How applicable the system is in daily use?

The users answer these questions at the end of their tests and give one (disagree) to five (agree) rate.
Their average rate of satisfaction regarding usability and efficiency of the proposed system among all the participants is according the following table.

Table1.
Usability of the proposed system

Questions	**Q1**	**Q2**	**Q3**
Participants AVG rate	3.90	4.30	3.30

Table1 describes that simplicity of the system is about 78 percent and this delivers that participants found this system to be easy and not complicated to work with. The most promising result with just over 4 quarters of participants shows that the efficiency of the system in

An International Journal JSCDSS E-ISSN: 2289-8603

protecting user's credentials against bystanders and keyloggers was quite acceptable. Although, the applicability of the system has the lowest percentage but some modifications could be considered to improve its applicability.

6 Discussion and future work

To pave the way for significantly more secure future regarding authentication systems and defeating possible and common threats, so many suggestions have been proposed in different forms. Each one has its own advantages and disadvantages while achieving a good trade-off between perfect security and usability is always hard. Typically, shoulder surfing attacks are classified into two categories. There is no special equipment most of times in the first type which is called weak shoulder surfing attack while in the second type a strong shoulder surfing attack with the help of equipment like cameras would help attackers to record hands movements or mouse clicks for later use (Wu, Lee, Lin, & Wang, 2014). The proposed system main focus is to battle with malicious software and the two types of shoulder surfing attacks. Experiments have shown that in different scenarios bystanders were not able to grab the second-pass as client click different positions with a hidden cursor. In the future work, experiments can be extended to several bystanders and more complicated scenarios while it perhaps requires some considerations to be applied to the system to make it more robust in those conditions. Performance of system under heavy load, different internet speed at client side must be considered as well. In addition, other factors that may influence the performance should be investigated precisely.

References

Abdurrahman, U. A., Kaiiali, M., & Muhammad, J. (2013). A new mobile-based multi-factor authentication scheme using pre-shared number, GPS location and time stamp. 2013 International Conference on Electronics, Computer and Computation, ICECCO 2013, 293–296. https://doi.org/10.1109/ICECCO.2013.6718286

Chakraborty, N., Randhawa, G. S., Das, K., & Mondal, S. (2016). MobSecure: A Shoulder Surfing Safe Login Approach Implemented on Mobile Device. Procedia Computer Science, 93(September), 854–861. https://doi.org/10.1016/j.procs.2016.07.256

Chen, Y., Sun, J., Zhang, R., & Zhang, Y. (2015). Your Song Your Way: Rhythm-Based Two-Factor Authentication for Multi-Touch Mobile Devices. 2015 IEEE Conference on Computer Communications (INFOCOM), 2686–2694. https://doi.org/10.1109/INFOCOM.2015.7218660

Crossman, M. A., & Liu, H. (2016). Two-factor authentication through near field communication. 2016 IEEE Symposium on Technologies for Homeland Security, HST 2016. https://doi.org/10.1109/THS.2016.7568941

De Luca, A., Hertzschuch, K., & Hussmann, H. (2010). ColorPIN – Securing PIN Entry through Indirect Input. Proceedings of the 28th International Conference on Human Factors in Computing Systems - CHI '10, 1103. https://doi.org/10.1145/1753326.1753490

De Luca, A., von Zezschwitz, E., Pichler, L., & Hussmann, H. (2013). Using fake cursors to secure on-screen password entry. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13, 2399. https://doi.org/10.1145/2470654.2481331

Gokhale, M. A. S., & Waghmare, V. S. (2016). The Shoulder Surfing Resistant Graphical Password Authentication Technique. Procedia Computer Science, 79, 490–498. https://doi.org/10.1016/j.procs.2016.03.063

Haque, S. A., Islam, S., Islam, M. J., & Grégoire, J. C. (2016). An architecture for client virtualization: A case study. Computer Networks, 100, 75–89. https://doi.org/10.1016/j.comnet.2016.02.020

Kang, J., Nyang, D., & Lee, K. (2014). Two-factor face authentication using matrix permutation transformation and a user password. Information Sciences, 269, 1–20. https://doi.org/10.1016/j.ins.2014.02.011

Kwak, D., Liu, R., Kim, D., Nath, B., & Iftode, L. (2016). Seeing Is Believing: Sharing Real-Time Visual Traffic Information via Vehicular Clouds. IEEE Access, 4(8), 3617–3631. https://doi.org/10.1109/ACCESS.2016.2569585

Lee, M. K., & Nam, H. (2013). Secure and Usable PIN-Entry Method with Shoulder-Surfing Resistance. Communications in Computer and Information Science, 374(PART II), 745–748. https://doi.org/10.1007/978-3-642-39476-8_149

Maheshwari, A., & Mondal, S. (2016). SPOSS: Secure Pin-Based-Authentication Obviating Shoulder Surfing. In I. Ray, M. S. Gaur, M. Conti, D. Sanghi, & V. Kamakoti (Eds.), Information Systems Security: 12th International Conference, ICISS 2016, Jaipur, India, December 16-20, 2016, Proceedings (pp. 66–86). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-49806-5_4

Prabhu, S., & Shah, V. (2015). Authentication using session based passwords. Procedia Computer Science, 45(C), 460–464. https://doi.org/10.1016/j.procs.2015.03.079

Shankar, V., Singh, K., & Kumar, A. (2016). IPCT: A scheme for mobile authentication. Perspectives in Science, 8(C), 522–524. https://doi.org/10.1016/j.pisc.2016.06.009

Shen, C., Yu, T., Xu, H., Yang, G., & Guan, X. (2016). User practice in password security: An empirical study of real-life passwords in the wild. Computers and Security, 61, 130–141. https://doi.org/10.1016/j.cose.2016.05.007

Socket.IO. (2017). Retrieved December 3, 2017, from https://socket.io/

Svogor, I., & Kisasondi, T. (2012). Two factor authentication using EEG augmented passwords. Proceedings of the International Conference on Information Technology Interfaces, ITI, 373–378. https://doi.org/10.2498/iti.2012.0441

Wu, T. S., Lee, M. L., Lin, H. Y., & Wang, C. Y. (2014). Shoulder-surfing-proof graphical password authentication scheme. International Journal of Information Security, 13(3), 245–254. https://doi.org/10.1007/s10207-013-0216-7

An International Journal JSCDSS E-ISSN: 2289-8603