Indian Journal of Science and Technology, *Vol 11(5), DOI: 10.17485/ijst/2018/v11i5/121107,* February *2018*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Secure Annihilation of Out-of-Band Authorization for Online Transactions

Sabahat Hussain, Burhan Ul Islam Khan*, Farhat Anwar and Rashidah Funke Olanrewaju

Department of Electrical and Computer Engineering, Kulliyyah of Engineering, International Islamic University Malaysia, Gombak, Kuala Lumpur, Malaysia; burhan.iium@gmail.com

Abstract

Objectives: In this paper, an approach to online banking authorization using one-time passwords has been illustrated. **Methods/Statistical Analysis:** The algorithm presented in this paper provides an infinite as well as forward One-Time-Password (OTP) generation mechanism employing two Secure Hash Algorithms viz. SHA3 and SHA2, followed by dynamic truncation to produce human-readable OTP. An inimitable authentication scheme has been presented in which a unique initial seed is used for generating a series of OTPs on the user's handheld gadget (i.e. a mobile phone). **Findings:** The proposed scheme demonstrated better results than the previous schemes of authorization after a security analysis was conducted on it. This is attributed to the eradication of cellular network within the authorization process. A high level of performance and efficiency in authentication and authorization was evident from the results that are vital for transacting online. **Applications/Improvements:** In the proposed system, the inherent features of the user's device (mobile phone) are utilized to form the initial seed. The application of hash functions to that seed eliminates the necessity to send one time passwords to the users via Short Message Service and decreases the limitations posed by out-of-band systems, thus making it suitable to be employed in online banking and other financial organizations.

Keywords: Authentication, Authorization, Out-of-band Authorization, Hash Functions

1. Introduction

The Internet has altogether transformed the daily lives of people since it is the standard medium chosen for all sorts of communication with online services and internet business. People make use of the internet for doing all kinds of jobs, such as sharing their private and personal information in various social networks. They conduct transactions online for online shopping, ticket booking, fund transfer, etc[1]. Consequently, privacy, as well as security, becomes the chief issue in such transactions. This is because there is every apprehension that an adversary may attempt to obtain the private information of some user by eavesdropping or actively corrupting the confidential data or information or rendering the system unavailable for the intended users[2].

The effect of growing technology has increased the reliance of organizations on information systems. Though the applications based on the web can provide efficiency and convenience, a range of new security threats may, however, pose severe security challenges to the IT infrastructure of organizations if mishandled. For about tens of years, organizations have been relying on the security policies offered on the network boundary for protecting their IT infrastructure[3]. Nevertheless, the conventional security technologies and policies may not be able to secure the web applications from emerging threats due to the targeting of security flaws in web application designs by the attackers[4]. Thus, new security policies – administrative as well as technical – must be implemented together with web application development. Also, the common vulnerabilities in web applications have to be understood for tackling the threats to new application services[5].

As per Internet Trends Report by Kleiner Perkins Caufield and Byers (KPCB)[6] published in 2013, mobile

*Author for correspondence

devices contribute more than 15 percent of the traffic on the internet. This depicts that a large population of Internet users make use of resource-constrained mobile phones to perform the various jobs online including banking transactions as well. Nevertheless, such users are still vulnerable to phishing attacks just like other users[7].

People are now getting more and more cautious about the appropriate use of information, specifically personal data. Besides, the assaults conducted by terrorists and criminals on the information systems are growing in number day by day. These assaults bring to light the inadequacies of safety efforts of multifactor authentication and secure socket layer protocol that numerous financial organizations have embraced. These safety efforts are constrained on the grounds that they just require that the bank and client believe each other and don't give the additional necessary confirmations to defeat Man-In-The-Middle (MITM) or related plans. It follows that the most common requirement of people all through the world is information security. Cryptography plays a major role in information security. Different cryptographic algorithms are used to make the data secured in web[8]. Cryptographic algorithms provide security to the web by maintaining data integrity. Encryption techniques, attack recognition, access control list, network security architecture, vulnerability and protocol analysis are present in cryptography[9]. A wide variety of cryptographic algorithms are employed to safeguard networks and researches on newer algorithms are being conducted on a regular basis to evolve enhanced techniques for securing communication[5,10]. In the following, we shall analyze different mechanisms used for authentication and authorization on the web.

1.1. Available Methods of Authentication and Authorization

Today, everyone is concerned that their identity should not be stolen or used by other people and therefore such security systems should be set up that prevent unauthorized data access. In particular, individuals working in organizations that require access to confidential and sensitive data need an authentication mechanism that is the strongest of all and that can only be provided by stand-alone OTP solution. Various methods are available for authorizing the users into any system. Besides the traditionally used combination of username/password, user authenticity can be verified using some additional methods as well. The username/password combination can be employed while authentication is performed before-hand while other modes are also available that monitor a user while the system is being used[11].

Authenticating transactions needs additional data in comparison to the users' actions at that instant and may lead to the generation of false positives. This method is utilized by banks for tracking transactions, e.g., if the credit card of an individual is used at some remote location other than the place of his residence, those transactions shall be monitored by the bank, and the card owner shall be contacted to ensure there is no misuse of card[8,11].

Biometric authentication involves the usage of retina scans, face recognition, fingerprints and voice recognition for authenticating users. Voice recognition is the simplest method that provisions the user to be authenticated remotely with no requirement of any additional hardware though it has the limitation to be counterfeited if a recorded voice is used and may suffer due to the poor quality of phone line[11].

Tokens are devices that can be used for authentication in combination with username/password or by themselves. They are mostly used in buildings for allowing people into areas with restricted access. They may even be electronic devices, like OTP generators. An example of a simple token includes the keys to one's car or house[11].

Multi-factor authentication involves combining more than one authentication techniques. Generally, such combinations lead to enhanced security. The various categories included in multi-factor authentication are something known to the user (i.e., PIN, password, etc.), something possessed by the user (i.e., token, smart card, etc.) and something inherent to the user (i.e., fingerprint, iris, face, etc.)[11].

In out-of-band authentication, an entirely different system is employed for authenticating the user, such as the transmission of verification SMS or call to the user while being logged in using the internet[11,12].

The different critical issues related to authentication, authorization and security of private and profoundly classified data have been analyzed by numerous specialists. Research work presented here features various procedures that have been taken up in the past to alleviate different sorts of assaults on the system of authentication and authorization of the client and takes care of issues related to securing entities. During the time spent investigating different methods taken up in the past and even in the present-day framework, it was discovered that the use of OTP appears to ensure enhanced security in access

management in private as well as public system[13]. OTP is legal for just a single try of access while attempting to make a unit of exchanges. One of the undeniable focal points of utilizing OTP is its profound security towards replay assault[14] which implies that unique password once produced will never be rehashed for the second time, and henceforth if the secret key is in control of the attacker, it will be of no utilization. In this manner, usage of OTP has been examined to consider a superior likelihood of making further upgrades in authentication of the clients[15,16].

Numerous authentication schemes have been put forward by researchers, but those based on OTPs have been found to be the strongest among all. A mobile/web-based authentication scheme for improving multi-factor authentication has been given[17] which is compatible and secure. OTP keys have been generated using Ping Pong 128 stream ciphers that behave just like one-time code. The dual communication channel, i.e., Global System for Mobile (GSM) and TCP/ IP are used in this authentication scheme which is burdensome. A fuzzy vault scheme has been used by researchers[18] for securing biometric data. A biometric authentication system based on speech recognition has been demonstrated in the paper[19], but used of a single biometric can be compromised by pre-recording the authenticated user's voice. An easy-to-implement framework for up-gradation of two-factor authentication to three-factor authentication is proposed[20]. The system makes use of three factors for user authentication, i.e., password, smart card and facial recognition. However, the system employs GSM besides being vulnerable to man-in-middle and imitation attacks. The security vulnerabilities of two-factor authentication in ATM system have been explored in a research[21], and a three-factor authentication scheme is proposed for providing effective security to ATM banking transactions. However, the system uses a single biometric, i.e., fingerprint information in addition to user PIN and smart-card. Different OTP innovations[17,22] are additionally observed to be patented, yet standardization of the OTP procedure is still considered to be a testing venture because of its various forms of utilization and architecture proposed by numerous past scientists and protocol producers.

Currently, OTPs based on SMS are generally used for authenticating and authorizing users for a wide variety of applications. However, it has been observed that SMS based OTPs are under massive attack, particularly in smartphones[23]. Recently, the National Institute of Standards and Technology (NIST) has denounced the use of two-factor authentication based on SMS or phone and shall not be considered advisable or secure in future[12,24].

The remaining paper is organized as follows: Section 2 presents various cryptographic hash functions and our proposed authorization protocol. In section 3, the results have been discussed, and section 4 concludes the paper.

2. Research Method

The goal of this study is to devise an authorization system for online transactions that is devoid of the vulnerabilities of out-of-band authorization systems. In spite of the fact that the use of OTP guarantees security in client validation yet the process of OTP generation from a server that is based on GSM in the present-day mobile based confirmation framework is liable to the danger of being compromised. Accordingly, a proper financially savvy convention and a secure authorization mechanism which caters the much-needed convenience of customers without compromising on security aspect are required. To overcome all the shortfalls, device details based two-way authentication mechanism needs to be developed; further improvements need to be achieved employing Secure Hash 'Algorithms - SHA3' and SHA2 in place of SHA1 and MD5 and facilitating human ease of short data entries as opposed to large length data. The reasons for not opting other cryptographic algorithms have also been explained in this section.

The proposed work makes use of SHA3 and SHA2 as standard algorithms for generation of One-Time-Passwords (OTPs) from an initial germ with no dependence on GSM network. The hardware profiles (International Mobile Equipment Identity-IMEI, International Mobile Subscriber Identity-IMSI and device OS timestamp) and software profile (index number) form the initial seed. Two random numbers 'N' and 'M' are generated which specify the number of SHA3 and SHA2 iterations respectively. RIPEMD128 results in 128-bit data which is later collapsed into a 64-bit result. The 64-bit key may be converted into six words in a user readable format using FRC 1751.

The proposed system is much more secure than the before mentioned approaches since it provides security and supports the performance with continued existence. The proposed system is cost-effective and ensures privacy, confidentiality, and non-repudiation as it employs secure

hash algorithms (SHA3 and SHA2) that has the capability of generating One Time Passwords on Android environment and thus ensures enhanced security with respect to the security of passwords required for authorizing a user.

2.1. The Concept of Cryptographic Hash Functions

A hash function is a function that generates fixed length output mapped from a variable length input. Thus, the output of hash function shall represent the fingerprint of the data that is input to it[25]. Cryptographic hash functions are found to play an important role in message integrity, user authentication, digital signatures, password protection and generating pseudo-random numbers. Authentication protocols like Kerberos make use of hash functions. It provides authentication, the integrity of data and eavesdropping prevention in client-server architecture. Hash functions also offer secure communication protocols like Secure Socket Layer (SSL) and Internet Protocol Security (IPSec)[26]. In SSL, the handshaking protocol employs a hash function for creating a message authentication code. To ensure the integrity of e-mail messages in Secure/Multipurpose Internet Mail Extensions (S/MIME) and Pretty Good Privacy (PGP), hash functions are widely used[2].

In mathematical notation, a hash function can be defined as MD = HF(M), HF being a hash function and MD being the message digest[27]. An ideal cryptographic hash function is characterized by the following chief features[7]:

i. Feasible computationally to generate a hash value for a given message.
ii. Infeasible to obtain a message from a given hash.
iii. Infeasible to change a message unless the hash is changed.
iv. Infeasible to have two distinct messages with identical hash values.

The best-known algorithms[2] to generate message digest are SHA3, SHA2, SHA1, SHA0, MD5, MD4 and MD2. A brief description of these has been presented below:

MD2 is a hash function that produces a 128-bit message digest in 18 rounds of compression function[27]. Researchers[28] demonstrated that the compression function of MD2 is vulnerable to collisions.

MD4 is a fast hash function that produces a 128-bit message digest with the compression function of 48 rounds. In the research[29], the cryptanalysis of MD4 has been demonstrated by finding a collision within a minute.

MD5 generates a message digest of 128-bit output with 64 rounds of its compression function. The research[30] presented the cryptanalysis of the MD5 compression function.

SHA0 generates 160-bit message digest by taking 80 rounds. The demonstration of collision attacks to SHA0 in 1995 was given in the study[31].

SHA1 is the most commonly used hash algorithm for integrity which generates a 160-bit message digest with 80 rounds. A research[32] demonstrated how to produce collisions on SHA1.

SHA2: NIST has published different secure hash functions SHA224, 256, 384 and 512 between 2001 and 2004. These are generally more robust than SHA1 but not time efficient than SHA1[27].

SHA3: In 2012, Keccak was declared the winner among a multitude of public entries in a competition conducted by NIST. It generates similar hash lengths as that of SHA2, but it differs in its internal structure from the remaining SHA family. SHA3 is an efficient hash function in hardware but not time efficient which takes a quarter of the time to run in hardware and comparing with SHA2 takes double the time to run in software[2].

NIST has listed specific algorithms which should not or should be used, how frequently the keys should be altered, their relative strengths based on their key size and other relevant information[33]. Furthermore, European Union Agency for Network and Information Security (ENISA) has also presented similar recommendations regarding the use of key sizes and cryptographic algorithms. In a report given in 2014, SHA2 family was recommended for use in the future but only till 256-bit version is employed. Since SHA224 lacks the minimum 128 bits recommended for security, this version is not suggested[34]. Other algorithms suggested include Whirlpool and SHA3 (with a minimum of 256 bits)[12,34].

2.2. Implementing the Authorization Protocol

In this section, the implementation details of the proposed authorization protocol have been highlighted including the installations. The Android Software Development Kit (SDK) includes all APIs and tools required for writing

Android applications. So, before delving into the depths of Android application development, all that is necessary to be done is setting everything up correctly. The primary requirements for Android development include Android SDK, an Integrated Development Environment (IDE) or an Editor and the Java Development Kit (JDK). IDE like Eclipse can be used for making the development comfortable and easier. Besides, a plug-in for Eclipse, Android Development Tools (ADT) is also required for adding support to Android development. Furthermore, the list of One-Time-Passwords has been computed before-hand to enhance the efficiency and performance. This is done using SQLyog as the database management tool owing to the ease of use and automaticity it provides.

The process of authorization of the user (as shown in Figure 1) has been explained below in a stepwise manner:

2.2.1. Employee Login

At the server (bank) site, the employee who is privileged can log in to the system. The employees and their details, i.e. employee id and passkey are stored on the server. It is only the registered employee who can manage the customers/users who want to avail the online transaction services. After logging in successfully, the employee can open an account for the user or can credit amount.

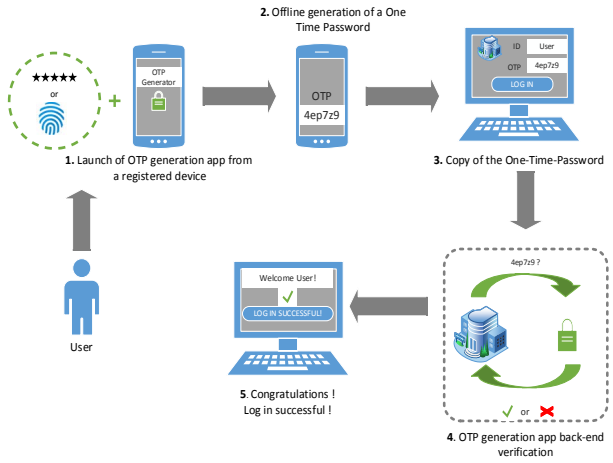

Figure 1. Methodology

2.2.2. Open new Account for User

Only the registered employees can add the user's account after obtaining the hardware/software profiles and other details of the user. The authentication details of the user have to be submitted manually by the users to the server (bank) which is considered as the most secure form of passing on the details. The hardware profiles constitute the IMEI (International Mobile Equipment Identity), the IMSI (International Mobile Subscriber Identity), the Device OS Timestamp and the software profile comprises of the Index number that uniquely identifies the user's application. These details are taken from the user's device and are placed under System Info. Even if the application is uninstalled on the device, the index number remains there with the server. It shall be flushed only if the user removes the application file (.apk) from his/her device. In that case, the user will have to request for the application from the server (bank) again and re-register his/her details manually. After registering, the user is provided with an account number to be used for accessing the online services. The details of all the users registered are stored at the server side.

2.2.3. User Login Page

When provided with an account by the server (bank), a new user submits his/her login id and password to account for the first layer of authentication. The user has to activate his/her account by logging in to his/her account at first.

The server (bank) stores the login details of the users which comprises of the user id and the password. It is worth mentioning that it is the hash value of the password that is stored rather than the password itself. This protects the system from various attacks such as stolen-verifier attack where it may be possible that the intruder gets access to the password file stored on the server.

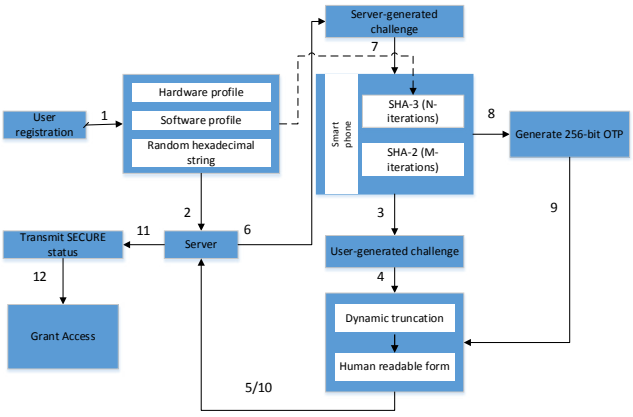

Figure 2. User authorization process

2.2.4. Sending Current OTP to Server

As soon as the user is logged in, the server prompts the user (shown in Figure 2) for the current OTP, the coordinate N (number of iterations of SHA3) and the coordinate

M (number of iterations of SHA2). The current OTP and the values of M, N are generated at the user device. The OTP generated after dynamic truncation comprises of 6 human-readable words which provide increased user convenience.

2.2.5.Server Challenge

The next step in the authentication process consists of a challenge sent by the server to the user to validate the user by his/her response. In the challenge, the server sends the coordinates M and N to the user and asks it to generate the final OTP on his/her handheld device. When the server receives the OTP from the client, it matches this OTP with the one produced on the server side. Only when the two matches, the server grants access privileges to the user, i.e. only when the user shall be allowed to transact from his account.

The user, when authenticated, is allowed to access his/her transaction services online. A user may transfer funds, view his/her account summary or change his/her password very conveniently. The account summary includes the date and time of transactions made from and to the user account. The server keeps a record of the transaction details of all the customers in its database.

3. Results and Analysis

The authorization mechanism implemented makes use of OTP which is valid only for a single session and is generated with the help of strong mechanism implementing concatenated cryptographic functions (SHA3 and SHA2), which results in a system impossible to break until the quantum computing comes in picture. In the proposed system, the initial seed is formed by the concatenation of IMEI, IMSI, device OS timestamp and index number. The values of SHA3 from 0-99 are pre-calculated to enhance efficiency. In this way, strong, efficient and stochastic One-Time-Passwords are generated without the use of an out-of-band system which has been shown in Figure 3.

The proposed scheme has been observed to ward off off-line guessing attack since it employs strong passwords obtained from strong nested hash functions. The generated OTP proved beneficial in avoiding identity theft. In addition, it can protect the online transactions from shoulder surfing attacks, key-loggers, replay attacks, etc. Furthermore, it restricts replay of reusable passwords since passwords are encoded such that they can be used only once. The results from the security analysis have been given below:

i. Our proposal allows the service provider as well as the user to utilize the challenge-response mechanism to cater pre-play attacks. This is because the next challenge is not predictable. Furthermore, the produced OTPs cannot help the intruder to calculate further OTPs or to get current or initial seeds because the intruder will be faced with the necessity of breaking two hash functions and counter dynamic truncation.

	n_value	in_value	out_value
	0	A12344354354354352121210090 1DFG	265335B1B60C04084708E0A187AA9B0079F6707D3BEB9D48FDA8820A67EC3D6FE01
	1	265335B1B60C04084708E0A187AA9B0079F6707D3BEB9D48FDA8820A67EC3D6FE01	894B1CE96DB5E5C1490BBE59E141300F1E0FCB0594763D492361E20081DD080BB28
	2	894B1CE96DB5E5C1490BBE59E141300F1E0FCB0594763D492361E20081DD080BB28	CFE9CE35DAE907B2FF19A8736412E88A11936CF08FB3CB52EAC2C89AD87C1581818
	3	CFE9CE35DAE907B2FF19A8736412E88A11936CF08FB3CB52EAC2C89AD87C1581818	7FECFA556C73E8ACEF98C032AA908EB54EBFA4BDDBB6AB16FDCA997A40F329E01D9
	4	7FECFA556C73E8ACEF98C032AA908EB54EBFA4BDDBB6AB16FDCA997A40F329E01D9	3F8BB9A59FE353BD38B22773DD2D17409D476A97D6C30C3DEFDB758DF3D2F73AE3F
	5	3F8BB9A59FE353BD38B22773DD2D17409D476A97D6C30C3DEFDB758DF3D2F73AE3F	6356F9CAF86B52279423D6E0891724D99FA48EFD0F1F837629A4772CADBCD5FE12D
	6	6356F9CAF86B52279423D6E0891724D99FA48EFD0F1F837629A4772CADBCD5FE12D	F0B610901CAA2CC548E9F67D0AB5EDD28C024E3316F6B046EB982059B5BFA1B9BDF
	7	F0B610901CAA2CC548E9F67D0AB5EDD28C024E3316F6B046EB982059B5BFA1B9BDF	F066D85D79FF9C4BFCF4AB54D0445E851624D27F93B599CAA2539D90DA44306CD17
	8	F066D85D79FF9C4BFCF4AB54D0445E851624D27F93B599CAA2539D90DA44306CD17	D1267AD632BC9F02699009AE8E97BDEF7D82A2DB83CE57B99BCA522E83A23A69E26
	9	D1267AD632BC9F02699009AE8E97BDEF7D82A2DB83CE57B99BCA522E83A23A69E26	C6E9A6340CFFD0E7625A51E8AE3CF88ACBD0A4CDC68F472BC944155A3176061AC3D
	10	C6E9A6340CFFD0E7625A51E8AE3CF88ACBD0A4CDC68F472BC944155A3176061AC3D	E5CF34302D9619112EB14503E2CD019FC0B3D47B7F9398C3E37A8EB8C8515CC2F4A
	11	E5CF34302D9619112EB14503E2CD019FC0B3D47B7F9398C3E37A8EB8C8515CC2F4A	48FD4EC8EF31CCF37D4DF3F223483B9D1EC5B7EDE60194ECB5E3FB2BD79AB4686A1
	12	48FD4EC8EF31CCF37D4DF3F223483B9D1EC5B7EDE60194ECB5E3FB2BD79AB4686A1	ACCF931DF8DB755DC2CCAF1875B19C1A2BD27D3FE90D36C6678B4534E11DEADC13E
	13	ACCF931DF8DB755DC2CCAF1875B19C1A2BD27D3FE90D36C6678B4534E11DEADC13E	07989A02F795A0885A220B1D2BB7E67D88235F98923F476AE6ABBB3C20E55740592
	14	07989A02F795A0885A220B1D2BB7E67D88235F98923F476AE6ABBB3C20E55740592	E6698C5FBBAB67752D9E6B9D7670BE2C90DBA1BB414DECA11418E009B8435AA62B4
	15	E6698C5FBBAB67752D9E6B9D7670BE2C90DBA1BB414DECA11418E009B8435AA62B4	F65D35C88ACE760EF5E013F41098847F41D64F02906C95AE4255A4E529AEE445E3F

Figure 3. List of generated OTPs

ii. To mount a forgery attack on the proposed scheme, an adversary must generate an OTP corresponding to a given challenge which is impossible to generate as the system makes use of strong concatenated cryptographic functions SHA3, and SHA2 and which results in a system impossible to break.
iii. The replay attack is a specific category of MITM attack which is absolutely intentional. This is warded off by making both parties exchange random number (M and N) and use it for all transactions between them.
iv. Since the challenge-response mechanism implemented by the system leaves the intruder crippled as the intruder won't be able to generate an acceptable challenge for the user so that authentication takes place, thus it also prevents phishing attacks.
v. The countermeasure to shoulder-surfing attack is that the proposed system implements OTP which changes every time the user makes a transaction.
vi. In the proposed system, brute force attack fails as the system employs OTP (valid for short time generated) from the strong combination of two cryptographic hash functions which can't be reversed until quantum computing is introduced.
vii. Key-logger captures the keystrokes of the user for stealing passwords. But the proposed algorithm counters it by simple OTP usage. OTP generated in the system caters this problem as every time a new OTP is generated.

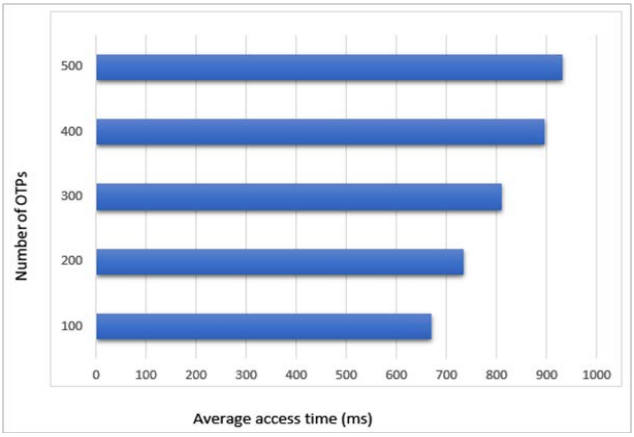

Figure 4. Average access time vs. Number of OTPs

Although performance was a chief concern from the beginning, the proposed algorithm could access 100 OTPs in few seconds with 50 iterations of hash for each password, as shown in Figure 4. This was too fast for normal usage, and quicker than expected, thus no optimization was required for the OTP generation in general.

4. Conclusion

Nowadays, authentication and authorization mechanisms based on a single factor such as a password is no more secure for banking and transactions made online. Automatic collection of passwords using some software is ubiquitous to retrieve passwords that are easy to guess like age, name, etc. In order to meet the requirement of providing robust authorization mechanism to the users, two-factor authentication was introduced. Most often, hardware tokens are supplied to users for their accounts. But the cost of maintenance and manufacturing of these tokens becomes a burden for the client as well as the organization. Moreover, the existing authentication and authorization mechanisms employ the services of GSM network for transmitting the one-time verification code which has been found to be vulnerable. Thus, a new password authorization mechanism is proposed where the OTPs are generated on the user's gadget (in this case, mobile phone) using the hardware and software profiles of the device. For OTP generation, two hash functions, i.e. SHA3 and SHA2 are used which are concatenated with each other. Then, the final OTP is generated by dynamic truncation that results in human readable one time password easy and convenient to be used by the user. However, the major limitation associated with the proposed scheme is that mobile phone used for authenticating the user becomes the only point of failure. In future, the proposed algorithm may be extended to be used with Windows, Palm and Blackberry phones.

5. Acknowledgment

This work was partially supported by Ministry of Higher Education Malaysia (Kementerian Pendidikan Tinggi) under the Research Initiative Grant Scheme number RIGS16-084-0248.

6. References

1. Olanrewaju RF, Khan BUI, Matto MMUI, Anwar F, Mir RN, Nordin ANB. Securing electronic transactions via payment gateways a systematic review. International Journal

of Internet Technology and Secured Transactions. 2017; 7(3):245–69. Crossref.
2. Madhuravani B, Murthy DSR, Reddy PB. Novel authentication protocol using multi cryptographic hashfunctions and steganography. International Journal of Advanced Computing (IJAC). 2015; 1-5.
3. Mir MS, Suhaimi AB, Khan BUI, Mattoo MMUI, Olanrewaju RF. Critical security challenges in cloud computing environment: An appraisal. Journal of Theoretical and Applied Information Technology. 2017; 95(10):1-15.
4. Masihuddin M, Khan BU, Mattoo MM, Olanrewaju RF. A Survey on E-Payment Systems: Elements, Adoption, Architecture, Challenges and Security Concepts. Indian Journal of Science and Technology. 2017 May 25;10(20).
5. Pradhan S, Giri CK. Role of different cryptographic algorithms in information security on web. International Journal of Engineering and Management Research (IJEMR). 2016; 6(5):339–45.
6. Internet trends. 2017 June 31. Available from: http://www.kpcb.com/internet-trends.
7. Goyal T, Vakil A, Jain R, Jinwala D. Preventing phishing attacks: a novel approach. International Journal of Computer Applications. 2015; 121(14): 8–12. Crossref.
8. Khan BUI, Olanrewaju RF, Baba AM, Langoo AA, Assad S. A compendious study of online payment systems past developments present impact, and future considerations. International Journal Of Advanced Computer Science And Applications. 2017; 8(5):256-71.
9. Gupta G, Chawla R. Review on encryption ciphers of cryptography in network security. International Journal of Advanced Research in Computer Science and Software Engineering. 2012; 2(7): 211–3.
10. Khan BU, Olanrewaju RF, Mir RN, Yusoff SH, Sanni ML. Trust and Resource Oriented Communication Scheme in Mobile Ad Hoc Networks. Proceedings of SAI Intelligent Systems Conference; Springer, Cham. 2016 Sep 21. p. 414-30.
11. Authentication. 2018 January 17. Available from: https://en.wikipedia.org/wiki/Authentication.
12. Kuhmonen S. One-Time Password Implementation for Two-Factor Authentication [Bachelor of Engineering thesis]. 2017; 1–53.
13. Mehraj T, Rasool B, Khan BUI, Baba A, Lone AG. Contemplation of effective security measures in access management from adoptability perspective. International Journal of Advanced Computer Science and Applications. 2015; 6(8):188–200. Crossref.
14. Mobile operating system. 2016 August 23. Available from: https://www.uswitch.com/mobiles/guides/mobile-operating-systems/.
15. Duan X, Niu B. A change password attack resistant scheme for remote user authentication using smart card. IEEE International Conference of Online Analysis and Computing Science, Chongqing, China. 2016. P.269–72. Crossref.
16. Deore U, Waghmare V. Cyber security automation for controlling distributed data. 2016 International Conference on Information Communication and Embedded Systems (ICICES), Chennai, India. 2016. p.1–4. Crossref.
17. Davaanaym B, Lee YS, Lee H, Lim H. A ping-pong based one-time-passwords authentication system. 5th International Joint Conference on INC, IMS and ID, Seoul, South Korea. 2009. p.574–9. Crossref.
18. Moon KY, Moon D, Yoo JH, Cho HS. Biometrics information protection using fuzzy vault scheme. 8th International Conference on Signal Image Technology and Internet Based Systems (SITIS), Naples, Italy. 2012. p.124–8.
19. Ma H, Yan S, Bai X, Zhu Y. The research and design of identity authentication based on speech feature. International Conference on Sensor Network Security Technology and Privacy Communication System (SNS and PCS), Nangang, China. 2013. p.166–9.
20. Avhad PR, Satyanarayana R. A three-factor authentication scheme in ATM. International Journal of Science and Research (IJSR). 2014; 3(4): 656–9.
21. Oruh JN. Three-factor authentication for automated teller machine system. IRACST - International Journal of Computer Science and Information Technology and Security (IJCSITS). 2014; 4(6):160–6.
22. Shivraj VL, Rajan MA, Singh M, Balamuralidhar P. One time password authentication scheme based on elliptic curves for Internet of Things (IoT). 5th National Symposium on Information Technology: Towards New Smart World (NSITNSW), Riyadh, Saudi Arabia. 2015. p.1–6.
23. Singh B, Jasmine KS. Secure end-to-end authentication for mobile banking. Software Engineering in Intelligent Systems. Springer International Publishing; 2015. p.223–32.
24. NIST Special Publication 800-63B. 2017 June. Available from: https://www.bing.com/cr?IG=7FB848AA70884B518B95E6D7B783495A&CID=05230D5E64F76C4F304106DF65586D32&rd=1&h=E8AvgWD5pB3wWjSaJuYixKkA1mBb_Vh2H7B_IwBE0TA&v=1&r=https%3a%2f%2fpages.nist.gov%2f800-63-3%2fsp800-63b.html&p=DevEx,5069.1.
25. Matusiewicz K. Analysis of modern dedicated cryptographic hash functions [PhD thesis]. Macquarie University, 2007. p.135–47.
26. Masihuddin M, Khan BUI, Mattoo MMUI, Olanrewaju RF. A survey on e-payment systems elements adoption architecture challenges and security concepts. Indian Journal of Science and Technology. 2017, 10 (20), pp. 1-19. Crossref, Crossref.
27. Stallings W. Cryptography and network security principles and practice. 3rd edition. Prentice Hall. 2002. p.1–900.

28. Rogier N, Chauvaud P. MD2 is not secure without the checksum byte. Designs Codes and Cryptography. 1997;12(3):245–51. Crossref.
29. Dobbertin H. Cryptanalysis of MD4. In Fast Software Encryption. Springer Berlin/Heidelberg. 1996. p.53–69. Crossref.
30. Dobbertin H. Cryptanalysis of MD5 Compress. Rump session of Eurocrypt. 1996.
31. Wang X, Yu H, Yin YL. Efficient collision search attacks on SHA-0. Annual International Cryptology Conference Springer Berlin, Heidelberg. 2005. p.1–16. Crossref.
32. Stevens M, Bursztein E, Albertini A, Markov Y, Karpman P. The first collision for full SHA-1. HashClash-Framework for MD5 & SHA-1 Differential Path Construction. 2009. P.1–23.
33. Barker EB, Barker WC, Burr WE, Polk WT, Smid ME. Recommendation for key management part 1: General (revision 3). NIST special publication. 2012; 800(57):1–47.
34. ENISA: Algorithms, key size and parameters report. 2014 November. Available from: https://www.enisa.europa.eu/publications/algorithms-key-size-and-parameters-report-2014/at_download/fullReport.