ISSN-2320-7884 (ONLINE)
ISSN-2321-0257 (PRINT)

A user authentication protocol for Credit Card security in online usage using opass

[1]Mythry Vuyyuru, [2]L. Srikanth
[1]M.Tech Student, [2]Asst Professor
[1,2]Department of Computer Science and Engineering
[1,2]Vignan's Lara Institute of Technology and Science
Guntur, Andhra Pradesh, India- 522213.

Abstract:-Today security concerns are on the rise in all areas such as banks, governmental applications, healthcare industry, military organization, educational institutions, etc. Most systems today rely on static passwords to verify the user's identity. However, such passwords come with major management security concerns. Users tend to use easy-to-guess passwords, use the same password in multiple accounts, write the passwords or store them on their machines, etc. Furthermore, hackers have the option of using many techniques to steal passwords such as shoulder surfing, snooping, sniffing, guessing, etc. Accordingly, new solutions for mobile telephony subscribers have been proposed. One of these utilizes *backward hash chains* to generate an OTP for authentication purposes. Applying the many from one function to a certain seed removes the requirement of sending SMS-based OTPs to users, and reduces the restrictions caused by the SMS system.

Keyword: password, online accounts, two factor authentications

I. INTRODUCTION

Passwords play a large part of the typical web user's experience. They are the near universal means for gaining access to accounts of all kinds. Email, banks, portals, dating and social networking sites all require passwords. The ubiquitous use of textual passwords for user

authentication has a known weakness: users choose passwords with predictable characteristics. This is due to a user tendency to choose passwords that are easy to remember this often means passwords which have "meaning" to the user. Unfortunately, these passwords, Which we will refer to as the probable password space, make up only an insignificant subset of The full password space. It is desirable to have users choose a wide variety of passwords, as this Increases the computational expense for the known threat of the dictionary attack. Graphical password scheme have been proposed as a plausible alternative [12]to text-based schemes, motivated in part by the fact that humans have a remarkable capability to remember pictures. Psychological studies support that people recall pictures with higher probability than words, including those most easily interpreted to have meaning. This motivates password schemes requiring recall of a picture in lieu of a word. If the number of possible pictures is sufficiently large, and the diversity of picture-based passwords can be captured.

Our password study focuses on online accounts. Website authentication scales up a user's password management problem. For real world interactions, users can leverage physical context: they stand at an ATM, they hold a cell phone, or they sit in front of their desktop. For online accounts, users are at the same machine but access many different accounts. Second, real world interactions [15] also have more regularity people may use their voicemail password or their building entry codes almost daily. Online inter actions may be more sporadic, where users visit a specic site rarely. To overcome the restrictions discussed above, this paper will discuss OTP production in the forward direction. This production will completely [1][2] eliminate the mentioned limitations. Our idea is to produce multiple OTPs from an initial seed in a parallel process with the service provider itself, e.g., an online bank, by utilizing two different types of hash functions, which come with a nested chain. The resulting chain provides forwardness and Infiniteness.

The rest of this paper is organized as follows: Section 2 discusses the related work, Section 3 Experimental Method, Section 4 analyzes the security attributes, Section 5 assesses our scheme's performance, and finally Section 6 concludes the paper.

II. RELATED WORK

a. Graphical Passwords:

We study the impact of selected parameters on the size of the password space for "Draw-A Secret" (DAS) graphical passwords. We examine the role of and relationships between the number of composite strokes, grid dimensions, and password[7] length in the DAS password space. We show that a very significant proportion of the DAS password space depends on the assumption that users will choose long passwords with many composite strokes. If users choose [17]passwords having 4 or fewer strokes, with passwords of length 12 or less on a 5 × 5 grid, instead of up to the maximum 12 possible strokes, the size of the DAS password space is reduced from 58 to 40 bits.

Additionally, we found a similar reduction when users choose no strokes of length 1. To strengthen security, we propose a technique and describe a representative system that may gain up to 16 more bits of security with[8] an expected negligible increase in input time. Our results can be directly applied to determine secure design choices, graphical password parameter guidelines, and in deciding which parameters deserve focus in graphical password user studies.

b. Purely Automated Attacks

We introduce and evaluate various methods for purely automated attacks against Pass Points style graphical passwords. For generating these attacks, we introduce a graph-based algorithm to efficiently create dictionaries based on heuristics such as click-order patterns (e.g., 5 points all along a line). Some of our methods combine click-order heuristics with focus of- attention scan-paths generated from a computational model of visual attention, yielding significantly better automated attacks[13][14] than previous work. One resulting automated attack finds 7-16% of passwords for two representative images using dictionaries of approximately 226 entries. Relaxing click-order patterns substantially increased the attack efficacy albeit with larger dictionaries of approximately 235 entries, allowing attacks that guessed 48-54% of passwords (compared to previous results of 1% and 9% on the same dataset for two images with 235 guesses). Our results show that automated attacks, which are easier to arrange than humanseeded attacks and are more scalable to systems that use multiple images, pose a significant threat to basic Pass Points-style graphical passwords.

C. Password Management Strategies

Given the widespread use of password authentication in on- line correspondence, subscription services, and shopping, there is growing concern about identity theft. When people reuse their passwords across multiple accounts, they in- crease their vulnerability; compromising one password[19] can help an attacker take over several accounts. They sometimes failed to realize that personalized passwords such as phone numbers can be cracked given a large enough dictionary and enough tries. We discuss how current systems support poor password practices. We also present potential changes in website authentication systems and password managers.

d. A Large scale Study Of Web Password Habits

We report the results of a large scale study of password use and password re-use habits. The study involved half a million users over a three month period. A client component on users' machines recorded a variety of password strength, usage and frequency metrics. This allows us to measure or estimate such quantities as the average number of pass- words and average number of accounts each user has, how many passwords she types per day, how often passwords[21] are shared among sites, and how often they are forgotten. We get extremely detailed data on password strength, the types and lengths of passwords chosen, and how they vary by site. The data is the large scale study of its kind, and yields numerous other insights into the role the passwords play in users' online experience.

III. EXPERIMENTAL METHOD

We have extended Lamport's idea with some Modifications in order to produce infiniteness and forwardness, avoiding the use of public key cryptography. The shortcoming of those two parameters, *infiniteness* and *forwardness*, cause the several vulnerabilities shown with respect to the related work

a.One Time Password:

One-time Passwords ofter well understood security enhancements over existing password systems. Our proposed scheme gets the excellent protection enjoyed by users of existing OTP systems to all users. We wish to be clear that we will have the same security and usability questions that arise with other OTP systems. Unlike conventional registration, the server requests for the user's account id and phone number, instead of password. After filling out the registration form, the program asks the user to setup a long-term password. This long-term password is used to generate a chain of one-time passwords for further logins on the target server. Then, the program automatically sends a registration SMS message to the server for completing the registration procedure. The context of the registration SMS is encrypted to provide data confidentiality.

b.Registration Phase:

The aim of this phase is to allow a user and a server to negotiate a shared secret to authenticate succeeding logins for this user. The user begins by opening the oPass program installed on her cell phone she enters IDu (account id she prefers) and IDs (usually the website url or domain name) to the program. The mobile program sends account id and url to the telecommunication service provider (TSP) through a 3G connection to make a request of registration. Once the TSP received the account id and the url, it can trace the user's phone number based on user's SIM card. The TSP also plays the role of third-party to distribute a shared key between the user and the server. The shared key is used to encrypt the registration SMS with AES-CBC. The TSP and the server will establish an SSL tunnel to protect the communication. Then the[6] TSP forwards account id, and to the assigned server. Server will generate the corresponding information for this account and reply a response, including server's identity ID, a random seed, and server's phone number. The TSP then forwards id, and a shared key to the user's cell phone. Once reception of the response is finished, the user continues to setup a long-term password with her cell phone.

c.Login phase:

This section will discuss the login and authentication process between the *user* and service provider. The steps below are shown in Fig. 3. The *user* logs in to the service provider's website, e.g., *an online bank*, requesting access. As a response to this access request, a secure session is established, i.e., an *SSL session*, allowing the *user* to enter his authentication privileges, i.e., *user name and password*, the first factor of authentication, *what the user knows*. Also the *user* provides the server with his OTP's current status. The current status allows the server to synchronize his seed with the client's current seed to get the same seed value on both sides before sending a challenge. The server randomly challenges the *user* with new indexes. The *user* enters those indexes, in his OTP generator to get the corresponding OTP. The *user* responds with this corresponding OTP. The server compares the received OTP with the calculated one. According to the server check, done in the previous step, the server will transfer an authorization execution or a communication termination.

e.Recovery Phase:

Recovery phase is designated for some specific conditions; for example, a user may lose her cell phone. The protocol is able to recover oPass setting on her new cell phone assuming she still uses the same phone number (apply a new SIM card with old phone number). Once user installs the oPass This message Procedure of recovery phase. Includes all necessary elements for generating the next one-time passwords to the user . When the mobile program receives the message, like registration, it forces the user to enter her long term password to reproduce the correct one-time password. During the last step, the user's cell phone encrypts the secret credential and server nonce to a cipher text. The recovery SMS message is delivered back to the server for checking. Similarly, the server computers and decrypts this message to ensure that user is already recovered. At this point, her new cell phone is recovered and ready

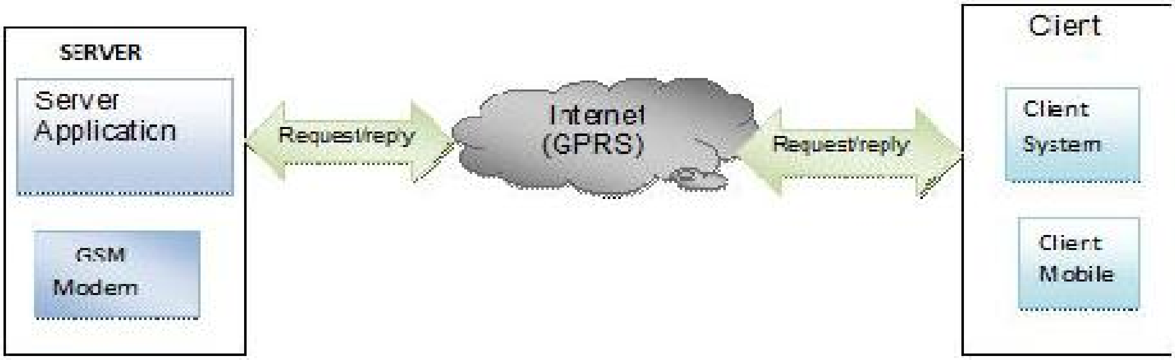


Figure 1. Authentication Protocol Architecture

to perform further logins. For the next login, one-time password will be used for user authentication.

f.Numerical Illustration:

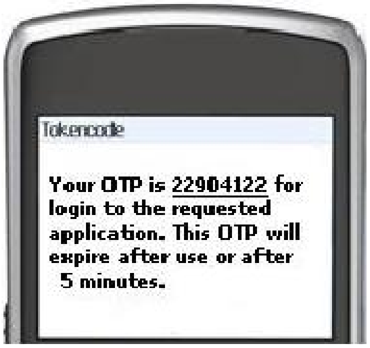
Through the registration process, the *user* gets two different hash functions, e.g., hA , which could be SHA-1, and *hB* , which could be MD5 , along with an initial seed, “sint ,” as the concatenation of the IMEI, IMSI, and registration time, which could be “12345678912345612345678912345070120102 00259” assuming IMEI is “123456789123456,” IMSI is “12345678912345,” and the registration time is “1/1/2013 20:02:59.”

After that the server sends a random challenge value of new indexes, e.g., *x*, *y* = 3, 4 , which means the *user* has to calculate his session OTP using this formula:

$$OTP=h^{4}{}_{B}\left(h^{4}{}_{A}\left(S_{crt}\right)\right)$$

68606061177919188523363813602016333158 .

The server has to calculate the same value in a parallel process, and as soon as the client responds The server will match the two values to give either a yes or no. In this illustration, we did not cover the conversion from digits, *the hashing output*, to characters, the password Format, considering the human interface. The second hash function *B h* allowed us to go in the forward direction by protecting the produced chain by hA. Also as indicated in it is not admissible for *x* nor *y* to be equal to 0.



Tokencode

Your OTP is 22904122 for login to the requested application. This OTP will expire after use or after 5 minutes.

Figure 2. OTP Generation

IV. SECURITY ANALYSIS

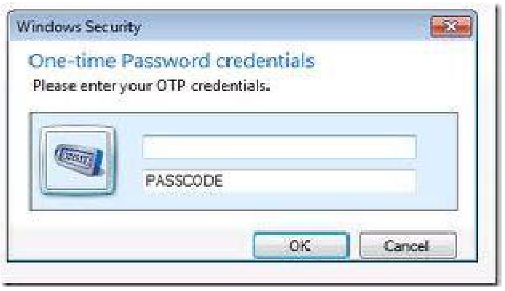
Security Schemes:

It can be said that while designing the GSM system, it had all security measures in mind, but as time passed and algorithms were cracked by the hackers, SMS-OTP based systems were not kept secure. Despite the apparent high costs and the given limitations of password security systems in phone (Fig 1), we believe it is imperative that e-commerce security systems move expeditiously to either augmented password security systems or alternative security schemes such as smart cards or biometrics.

While there are numerous alternatives to password security systems, each involve tradeoffs. Among the considerations are the cost to implement, the time required to use, any special considerations regarding place of use (for example, must it be from a particular computer), ability to[10] change the scheme if it is compromised, physical limitations, health considerations (for example, a fingerprint reader on a public site), non transferability, time stamped, and so on. It is beyond the scope of this article to thoroughly explore each of these alternatives and their limitations. However, the Main classes of alternative technology are discussed to demonstrate their potential.

Pre-Play Attack:

Unless the challenge is protected, a type of "suppress replay attack," known as a "pre-play attack," becomes possible. Consider that an intruder, who is able to predict[11] the next challenge, wishes to impersonate the user to the service provider. The intruder takes the service provider role, by impersonating it to the user, and asks the user to authenticate itself(Fig 3). The intruder chooses the next challenge that will be chosen by the service provider when authenticating the user. The challenge's response sent by the user is memorized by the intruder.



One-time Password credentials

Please enter your OTP credentials.

Figure 3. Verification Application

Then, at some future time, the intruder can impersonate the user to the service provider, using this memorized response. Our proposal allows the service provider to challenge the user with unpredictable uniformly distributed values of x and y . If we suppose that x and y can take one value of forward m values, the probability of successfully guessing a challenge will be the joint probability of x and y , which is equal to 1 m2. We can refer to this property as the ability to resist

Attacks on Registration and login:

To prove the guaranteed secrecy of credential in the registration phase, we translate our desired feature and system settings to the Horn clauses. Then we verify our aim through performing ProVerif with those clauses. The registration phase, which consists of seven messages, requires 26 rules to be defined. ProVerif assumes that an attacker can intercept every message (includes SMS) between the cell phone, the TSP, and the server. Meanwhile, we hypothesize that the attacker does not know the session key of 3G connection; the attacker also does not know the session key of SSL tunnel. The attacker also cannot recover the from the encrypted login SMS. Hence, phishing attacks do not work under oPass. In oPass, users type their accounts into the kiosk and type their longterm passwords into the cell phones. A kiosk that is installed with malwares or key loggers to snatch user passwords is also useless. OPass achieves a one-time password approach to prevent against password reuse attacks.

Predictable attacks:

Furthermore, the produced OTPs cannot help the intruder to calculate further OTPs or to get current or initial of breaking the second hash function.

Reparability:

If the *user* finds or suspects that his seed has been compromised, e.g., *token theft*, he can reregister with the *host* and agree upon new seeds, but this must be done manually.

Forgery Attack:

To mount a forgery attack on the proposed scheme, an adversary must generate an OTP corresponding to a given challenge. Since the adversary doesn't[12] know he can't correctly update the session OTP for acceptance by the *host*. Hence, the proposed scheme can resist the forgery attack. It is also necessary to have tight control over the transition from an old OTP generator to a new one.

Small Challenge Attack:

Attacks based on sending small challenges by intruders who impersonate the communication *host* only affect the backward hash chains' OTPs. Our scheme uses forward hashing techniques, which eliminates this type of attack completely.

V. PERFORMANCE ASSESSMENT

The performance evaluation considers the computational cost from the *user* side. Considering the t *th* authentication login time, the utilization of the S/KEY™ will cost the *user* a number of $N - t$ hash operations, where N is the defined chain length. Bicakci's scheme [8] has the lowest number of steps, utilizing just one chain step; the price of this benefit is the use of *public key cryptography* to produce the signature chain. However time based algorithms have to guarantee a main server synchronized internal clock. Our approach costs the *user* $x + y$ hash operations, which is very cheap compared with the number of $N - t$ hashes. Our approach doesn't involve public key techniques, and has no need of utilizing time synchronization. All participants felt that the registration and login processes in the oPass system were easy. Furthermore, they agreed that oPass was more secure than the original login system. It is quite important to make users feel secure. It also demonstrates that our proposed system was well suited to users regardless of background. Some of the participants prefer oPass to the original login system. Meanwhile, many participants suggested that oPass was better suited for financial websites, for example online banking or online shopping. They believed that general websites do not need such high security level.

VI. CONCLUSIONS

In this paper, we proposed a user authentication protocol named oPass which leverages cell phones and SMS to thwart password stealing and password reuse attacks. We assume that each website possesses a unique phone number. We also assume that a telecommunication service provider participates in the registration and recovery phases. The design principle of oPass is to eliminate the negative influence of human factors as much as possible. Through oPass, each user only needs to remember a long-term password which has been used to protect her cell phone. . For online accounts, users are at the same machine but access many different accounts. Second, real world interactions also have more regularity people may use their voicemail password or their building entry codes almost daily. Our algorithm doesn't require a token embedded server synchronized clock like . Our approach eliminates the problems with utilizing OTPs with an SMS, consisting of the SMS cost and delay, along with international roaming restrictions like.

REFERENCES

[1] B. Pinkas and T. Sander, "Securing passwords against dictionary attacks," in *CCS '02: Proc. 9th ACM Conf. Computer Communication Security*, New York, 2002, pp. 161–170, ACM.

[2] J. A. Halderman, B. Waters, and E. W. Felten, "A convenient method for securely managing passwords," in *WWW '05: Proc. 14th Int. Conf. World Wide Web*, New York, 2005, pp. 471–479, ACM.

[3] K.-P. Yee and K. Sitaker, "Passpet: Convenient password management and phishing protection," in *SOUPS '06: Proc. 2nd Symp. Usable Privacy Security*, New York, 2006, pp. 32–43, ACM.

[4] S. Chiasson, R. Biddle, and P. C. van Oorschot, "A second look at the usability of click-based graphical passwords," in *SOUPS '07: Proc. 3rd Symp. Usable Privacy Security*, New York, 2007, pp. 1–12, ACM.

[5] K. M. Everitt, T. Bragin, J. Fogarty, and T. Kohno, "A comprehensive study of frequency, interference, and training of multiple graphical passwords," in *CHI '09: Proc. 27th Int. Conf. Human Factors Computing Systems*, New York, 2009, pp. 889–898, ACM.

[6] J. Thorpe and P. C. van Oorschot, "Graphical dictionaries and the memorable space of graphical passwords," in *SSYM'04: Proc. 13th Conf. USENIX Security Symp.* Berkeley, CA, 2004, pp. 10–10, USENIX Association.

[7] J. Thorpe and P. C. van Oorschot, "Humanseeded attacks and exploiting hot-spots in graphical passwords," in *SS'07: Proc. 16th USENIX Security Symp. USENIX Security*, Berkeley, CA, 2007, pp. 1–16, USENIX Association.

[8] P. van Oorschot, A. Salehi-Abari, and J. Thorpe, "Purely automated attacks on pass points-style graphical passwords," *IEEE Trans. Information Forensics Security*, vol. 5, no. 3, pp. 393–405, Sep. 2010.

[9] R. Dhamija, J. D. Tygar, and. Hearst, "Why phishing works," in *CHI '06: Proc. SIGCHI Conf. Human Factors Computing Systems*, New York, 2006, pp. 581–590, ACM.

[10] C.Karlof,U. Shankar, J. D.Tygar, andD.Wagner, "Dynamic pharming attacks and locked same-origin policies for web browsers," in *CCS '07: Proc. 14th ACMConf. Computer Communications Security*, New York, 2007, pp. 58–71, ACM.

[11] T. Holz, M. Engelberth, and F. Freiling, "Learning more about the underground economy:Acase-study of keyloggers and dropzones," *Proc. Computer Security ESORICS 2009*, pp. 1–18, 2010.

[12] S. Hallsteinsen, I. Jorstad, D-V., Thanh, "Using the mobile phone as a security token for unified authentication", Systems and Networks

Communication. In: International Conference on Systems and Networks Communications, 2007, pp. 68-74.

[13] T. Laukkanen, S. Sinkkonen, M. Kivijarvi, P. Laukkanen, "Management of Mobile Business", ICMB 2007, International Conference on the Digital Object Identifier, 2007, pp.42-42.

[14] H. Wang, "Research and Design on Identity Authentication System in Mobile-Commerce", In: Beijing Jiaotong University, 2007, pp. 18-50.

[15] S.M. Siddique, M. Amir, "GSM Security Issues and Challenges Software Engineering", Artificial Intelligence, Networking and Parallel/Distributed Computing, 2006. SNPD 2006. 7th ACIS International Conference on Digital Object Identifier, pp. 413-418.

[16] L. Lamport, "Password Authentication with Insecure Communication", In: Comm. ACM, vol. 24, No 11, 1981, pp. 770-772.

[17] N. Haller, "The S/KEY One–Time Password System. In: Proceedings of the ISOC Symposium on Network and Distributed System Security", 1994, pp. 151-157.

[18] A. Chefranov, "One–Time Password Authentication with Infinite Hash Chains. Novel Algorithms and Techniques", In: Telecommunications, Automation and Industrial Electronics, 2008, pp. 283-286.

[19] K. Bicakci N. Baykal, "Infinite length hash chains and their applications" In: Proceedings of 1st IEEE Int. Workshops on Enabling Technologies: Infrastructure for Collaborating Enterprises WETICE'02, 2002, pp. 57-61.

[20] R. Rivest, A. Shamir, L. Adleman, "A method for obtaining digital signatures and public–key cryptosystems", In: Communications of the ACM, 1978.

[21]http://www.rsa.com/node.aspx?id=1156[Accessed: October 04, 2010].