Web Based Security Analysis of oPass Authentication Scheme Using Mobile Application

[1]P. Vaisagan, M. [2]Nasreen Fathima, [3]P.Elenthendral

[1] Assistant Professor, IT Department, Jeppiaar Engineering College,
Chennai, Tamil Nadu, India

[2,3] B.Tech IT, III Year, Jeppiaar Engineering College,
Chennai, Tamil Nadu, India

Abstract

Text password is the most common form of user authentication on websites due to its simplicity and convenience. Users passwords are easy to be stolen and compromised under different threats. Two types of mistakes are commonly done by the users. Firstly, users select weak passwords and reuse the same passwords across different websites. Secondly typing the passwords into un trusted computers leads to password threat and theft. In this paper, we design a user authentication protocol named Opass which uses a user's cell phone and short message service to thwart password stealing and reuse attacks . Opass only requires each participating website to possess a unique phone number and involves a (TSP) telecommunication service provider in registration and recovery phases. Through Opass users only need to remember a long term password to login on all websites.

Keywords: Network security, password stealing, password reuse attack , User authentication

1. Introduction

Over the past few decades text password has been adopted as the primary means of user authentication for websites. People select the username and text password when registering accounts on a Website. In order to login into the website successfully, user must recall the registered password. Generally password based user authentication can resist brute-force and dictionary attacks if user select strong passwords. Thus most users would choose an easy to remember password. Even if they know the password might be unsafe. Another crucial problem is the user tends to reuse the password across on various websites. Password reuse causes users to lose sensitive information stored in different websites if the hacker compromises one of the passwords. This attack is referred to as a password reuse attack. The advantage is those users only have to remember Master password to access the management tool. Adversaries can steal or compromise passwords and impersonate users identities to launch malicious attacks, collect sensitive information and performs unauthorized payments actions or leak financial secrets. The main cause of password stealing and reuse attack is when user types the passwords into un trusted public computers. The main concept of Opass is free users from having to remember or type any passwords into conventional computers for authentication. Opass involves new component a cell phone which is used to generate one-time password and a new communication channel, SMS which is used to transmit authentication messages.

2. Proposed System

Opass utilizes a user's cell phone as an authentication token, SMS as a secure channel. In the registration phase the users start the pass program to register a new account on the website she wishes to visit in the future. The server requests for the users account Id and phone number instead of the users password. After filling out the registration form the program asks the users to setup a long-term password. This long-term password is used to generate a chain of one-time password 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. Pass also designed a recovery phase to fix problems in conditions such as losing one's cell phone. Login procedure in Opass does not require user to type password into a un trusted web browser. The user opens the Opass program on her phone and enters the long-term

password. The program will generate a one-time password and send a login SMS securely to the server. The login SMS is encrypted by the one-time password. Finally the cell phone receives the response message from the server and shows the success message on the screen to verify her identity.

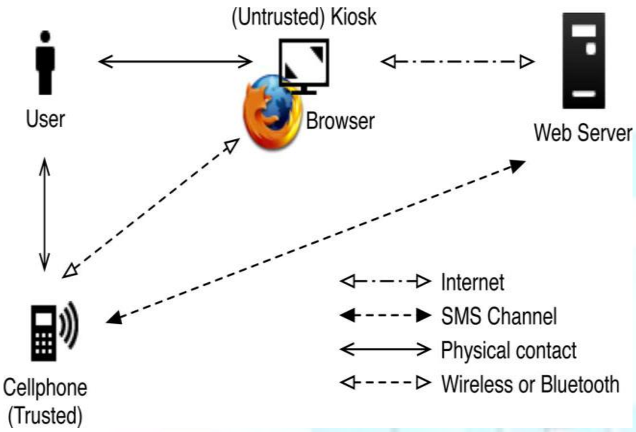


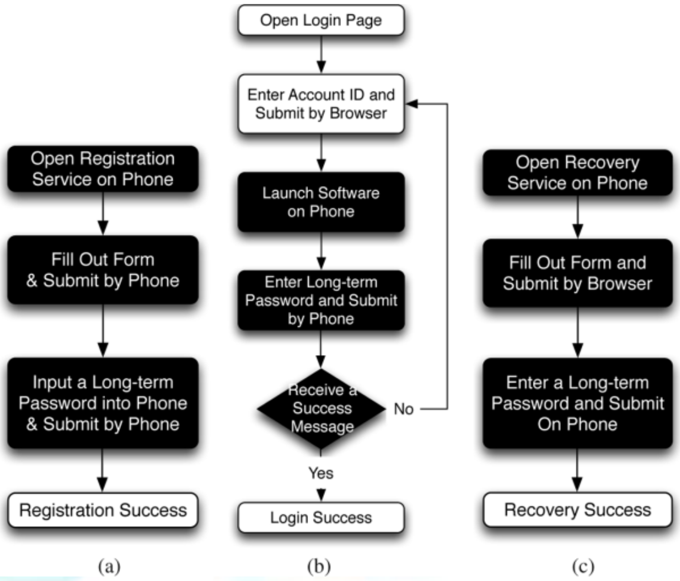
Fig. 1. Procedure of login phase.

Opass has the following advantages:

I. Anti-Malware: malware that collects the sensitive information from users including their passwords. In Opass users are able to login into the web services without entering the passwords on their computers. Thus malware cannot obtain user passwords from un trusted computers
II. Phishing Protection: Adversaries often launch phishing attacks to steal users password by cheating users when they connect to forged websites. Users adopt Opass to withstand phishing attacks.
III. Secure Registration and Recovery in Opass SMS is an out of band communication interface. Opass cooperates with TSP in order to obtain the correct phone number of users and websites. SMS aids Opass in establishing a secure channel for message exchange in registration and recovery phase. Recovery phase is designed to deal with the case when users lost their cell phone with the help of new sim cards; Opass still works on new cell phone.
IV. Password Reuse Prevention and Weak Password Avoidance:
Opass follows one-time password approach. The cell phone automatically derives different passwords for each logins. Under this approach users do not need to remember any password for login, they only need to remember a long time password for accessing their accounts
V. Cell phone Protection:
An adversary can steal user's cell phone and try to pass through user authentication. However the users are protected by the long-term password.

3. Implementation

Opass consists of Registration, Login and Recovery Phases.


Open Registration Service on Phone
Fill Out Form & Submit by Phone
Input a Long-term Password into Phone & Submit by Phone
Registration Success
Open Login Page
Enter Account ID and Submit by Browser
Launch Software on Phone
Enter Long-term Password and Submit by Phone
Receive a Success Message
No
Yes
Login Success
Open Recovery Service on Phone
Fill Out Form and Submit by Browser
Enter a Long-term Password and Submit On Phone
Recovery Success

(a) (b) (c)

Fig. 2. Three Phases Of Opass.

3.1 Registration

The aim of this phase is to allow the user and server to negotiate the shared secret to authenticate the succeeding logins for the user. The user begins by opening the Opass program installed on her cell phone. She enters IDu (account ID) and IDs (website URL or domain name) to the program. The mobile program sends IDu and IDs to TSP through a 3G connection to make a request of registration. Once the TSP received the IDu and IDs, it can trace the user's phone number (Tu) based on users sim card. The TSP also plays the role of third party to distribute the shared key Ksd between the user and the server. The shared key Ksd is used to encrypt the registration SMS with AES-CBC. The TSP and the server S will establish a SSL tunnel to protect the communication. Then the TSP forward the IDs, Tu ,Ksd to the assigned server S. server S will generate the corresponding information for this account and reply a response including the server's identity IDs and random seed ϕ and the server's phone number Ts. The TSP then forwards the IDs, ϕ, Ts, and the shared key Ksd to the users cell phone. Once reception of the response is finished the user continues to setup a long-term password (Pu) with her cell phone. The cell phone computes a secret credential C by the following operation

C=H(Pu || IDs || ɸ).

To prepare a secure registration SMS, the cell phone encrypts the computed credential c with the key Ksd and generates the corresponding MAC. I.e. HMAC1.HMAC-SHA1 takes inputs users identity ,cipher text and IV to the output MAC. Then the cell phone Sends the encrypted registration SMS to the server by phone number Ts has follows

Cell phone → S: IDs. {C || ɸ} Ksd, IV, HMAC1.

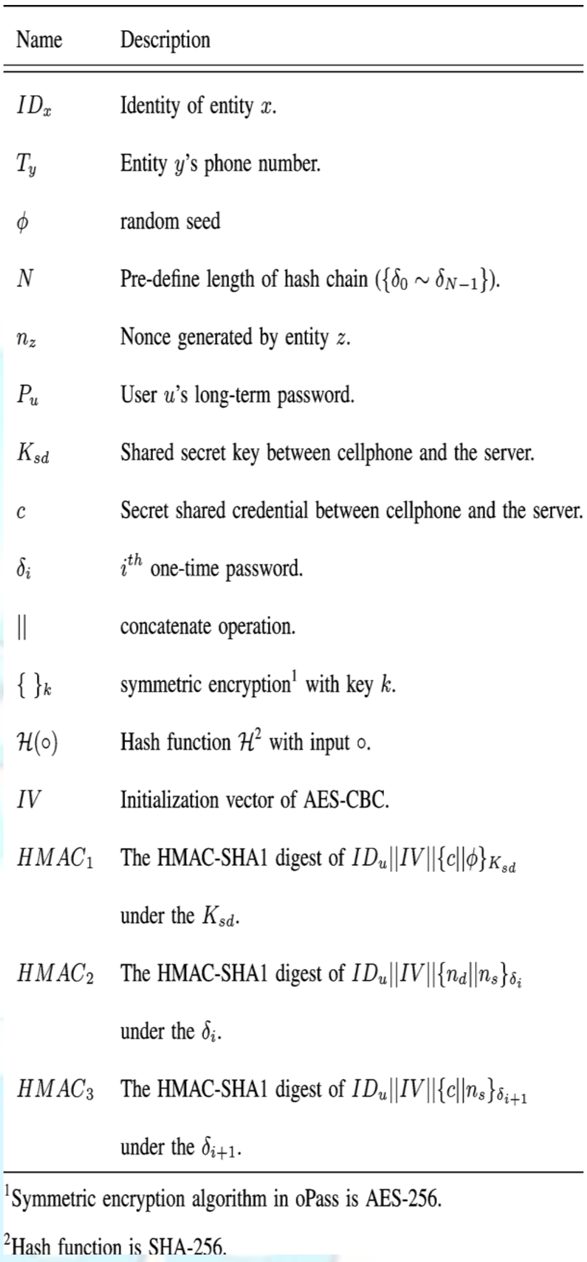
Server S can decrypt and verify the authenticity of the registration SMS and then obtain c with the shared key Ksd. Server S also compared the source of received SMS with Tu to prevent SMS Spoofing attacks. At the end of registration the cell phone stores all the information (Ids,Ts,ɸ,i) ,except for the long-term password Pu and the secret s. variable I indicates the current index of the one-time password and is initially set to 0. With i the server can authenticate user's device during each login. After receiving the message (6) the server stores (IDs, Tu, c, ɸ, i) and completes the registration.

3.2 Login:

The login phase begins when u sends the request to the server s through an un trusted browser (on a kiosk). The user uses a cell phone to produce an one-time password e.g□i and the deliver necessary information encrypted with □i to the server S via an SMS message. Based on the pres hared secret credential c, server S can verify an authenticated user u based on □i. the protocols starts when user u wishes to login to her favourite web server S. However u begins the login procedure by accessing the desired website via a browser on an un trusted kiosk. The browser sends a request to S with u's account IDs. Next server S supplies the IDs and fresh nonce ns to the browser. After reception of the message the cell phone inquiries related information from its database via IDs, which includes the servers phone number Ts and other parameters {ɸ,i}. Secret shared credential c can be regenerated by inputting the correct long-term password pu on the cell phone. The one-time password □i for current login is recomputed using the following operations C=H (Pu || IDs || ɸ)

□i = H N-i(c).

Table 1:Notations



Name	Description
ID_x	Identity of entity x.
T_y	Entity y's phone number.
ϕ	random seed
N	Pre-define length of hash chain ($\{\delta_0 \sim \delta_{N-1}\}$).
n_z	Nonce generated by entity z.
P_u	User u's long-term password.
K_{sd}	Shared secret key between cellphone and the server.
c	Secret shared credential between cellphone and the server.
δ_i	i^{th} one-time password.
$\|\|$	concatenate operation.
$\{\ \}_k$	symmetric encryption[1] with key k.
$\mathcal{H}(\circ)$	Hash function $\mathcal{H}$[2] with input $\circ$.
IV	Initialization vector of AES-CBC.
$HMAC_1$	The HMAC-SHA1 digest of $ID_u\|\|IV\|\|\{c\|\|\phi\}_{K_{sd}}$ under the K_{sd}.
$HMAC_2$	The HMAC-SHA1 digest of $ID_u\|\|IV\|\|\{n_d\|\|n_s\}_{\delta_i}$ under the δ_i.
$HMAC_3$	The HMAC-SHA1 digest of $ID_u\|\|IV\|\|\{c\|\|n_s\}_{\delta_{i+1}}$ under the δ_{i+1}.

[1] Symmetric encryption algorithm in oPass is AES-256.

[2] Hash function is SHA-256.

The cell phone generates a fresh nonce and. To prepare a secure login SMS, the cell phone SMS the cell phone encrypts and and ns with □i and generates MAC.i.e HMAC 2. The cell phone sends the following SMS to the server S.

Cell phone → S:IDs, {nd, ɸ, ns}.

After receiving the login SMS the server computes □i to decrypt and verify the authenticity of the login SMS. If the received SMS equals the previously generated SMS, then the user is said to be legitimate. Otherwise the server will reject the login request. After successful verification the server sends a success message through the internet. H {nd || □i} to the user device . Thecell phone will verify the received message to ensure the completion of the login procedure. The last verification on the cell phoneis used to prevent the phishing attacks and Man-in-Middle attacks. If the verification failed the user knows the failure of login and the device would not increase the index i. if the user successfully logged into the server means then the index i is automatically increased i.e. i=i+1.

3.3 RECOVERY:

Recovery phase is designed because of some specific conditions. for example the protocol is able to recover Opass setting on a new cell phone assuming she still uses same phone number.i.e Applying the new cell phone with the same phone number. Once user u installs the Opass program on a new cell phone she can launch the program to send the recovery request with the account and the requested server IDs to TSP through a 3G connection.

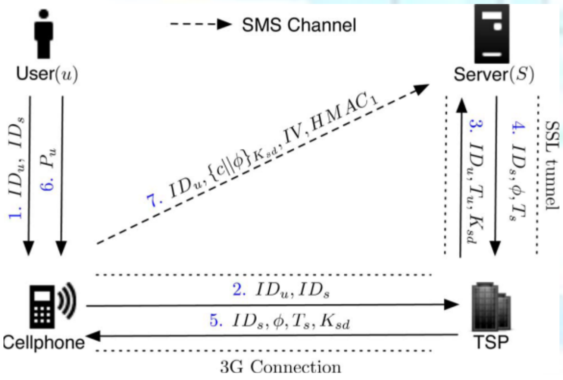


Fig 3:Recovery phase

4. Experimental Setup

:

We implemented a prototype of Opass according to the above three phases. It consists of three components mobile program running on the Android smart phones, an extension of fire fox browser and a web server. The server provides a web service by an Apache server running on a workstation with Windows. Therefore the SMS service with GSM Modem is connected to itself.3G connection is established between the cell phone and TSP to provide data confidentiality. SSL tunnel is used to provide secure communication between TSP and Web Browser. Bluetooth and cable line can also be substituted for the communication interface.

5. Results:

About 50% of users in different websites reuse the same password. The data shows half of the participant's passwords are weak passwords. All participants have never adopted any password management tool to protect their accounts. Therefore the login success rate is over 90% except for few typing errors. The average time of login is 21.62 s and SMS delay time is 8.9s.

6. Experimental design:

We implemented a prototype of Opass and conducted a user study to analyze the performance and usability.

A. Prototype Implementation:

We implemented a prototype of Opass according to its three phases. The prototype consists of three components: a mobile program running on Android smart phones (Android OS v2.1); an extension on Firefox browser; and a web server. The server offers a web service by an Apache server running on a workstation with Windows XP, and SMS service with a GSM modem connected to itself. The communication interface between the phone and the browser extension is based on a client/server model over the TCP/IP network. Phones utilize their Wi-Fi or 3G to connect the TCP server built by the extension. Other
Mediums, such as Bluetooth and cable line, can substitute for current communication interface. Moreover, we reduced the amount of user interactions and optimize the whole performance in all components. Detail implementations of these components are depicted as follows. We developed the client program on Android OS due to its popularity and generality. The program has been established and conducted on a Motorola Milestone phone. For safety operations, fundamental information of Opass is kept safe in an encrypted SQLite database with as an encryption key. After installing the program, a user creates an account to a website via the registration procedure. Upon successful registration, the user can log into the website. To make the progress smooth, the user only has to key in her long-term password and select a website. Then the remaining operations would perform by the program through clicking a button. All required interactions are eliminated to ensure Opass efficiency. Currently, a fully functional extension has been implemented on Firefox browsers. Users installed the extension on the browser in a kiosk. The major purpose of the browser extension on kiosks is allowing forwarding data from the web server tothe userscell phone during the *login* phase. While the user attempts to login on a predefined website, the extension automatically sets up a TCP server socket. After the user clicks the login button on her cell phone, a connection could be built and forward data from the

website to the cell phone and vice versa. In the web server implementation, we developed a server program which consists of main server codes (PHP) and setup scripts for database (MYSQL). Server program can be installed and performed on an Apache HTTP server. On the other hand ,capacity of sending/receiving SMS via a GSM modem relies on an open source library SMS Lib . For simulating TSP, partial PHP codes and related information were also established by The database.

B. User Study:

To analyze the effectiveness and usability of Opass, we conducted a user study with 24 participants, of which eight are female. The average age of the participants was 22. All were university students from distinct departments. Half were computer science students who have knowledge about security and half were not. All were regular computer users, and the average computer experience was 11.9 years. However, most of them were not familiar with the use of a smart phone, especially typing on phones. Participants completed individual tests which consisted of three processes that included setting up, registering, logging in. Before starting the study, participants were first asked to complete a demographics questionnaire. They were then introduced to the Opass system. They were told that they would be setting up the system, registering an account, and logging in via a cell phone. Further, they were instructed to choose a strong long-term password that should be at least eight digits long. Participants Completed one practice test (not included in the analysis data) to ensure that they understood how to operate the system. They then proceeded to complete a formal test which consisted of the following steps.

1) Setting up the system: Different from the ordinary user authentication system, users should install a cell phone software and a browser extension to setup the Opass system.

2) Registering for an account: Users first open the registration software on the cell phone. Users then fill out a form, which includes an account id, a website's id, and a long-term password, and submit it to the website.

3) Logging into the website: Users first enter their account id into the browser on the kiosk and submit it to the server. Users then type their long-term password into the cell phone and submit to the server. The login succeeds if a success Message is shown on the screen of cell phone. If login fails, participants should try again until they are successful.

After the test, the participants also completed a post-test questionnaire in order to collect their opinions.

7. Collected Results:

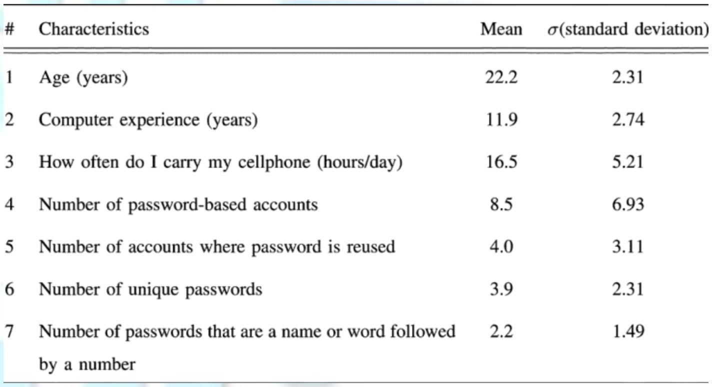
Our data analysis is used to show the usability of the Opass system and to estimate its performance.

A. Usability Evaluation

As shown in Table II, over 50% of accounts, which are in different websites, reuse the same password. Furthermore, the data shows that half of participants' passwords are weak passwords.

Despite these security risks, all participants had never adopted any password management tool to protect their accounts. This fact appears to be consistent with our observation about password reuses and weak password.

As opposed to computer science participants, others (12 participants) did not have experience with browser extensions. After introducing the installation steps, they succeeded in setting up the system. The login success rate is over 90%, except for a few typing errors. Table III shows the result of post-test questionnaire. The scores are out of ten and higher scores indicate that the statement accepts more users' approvals. Most participants can easily install the extension on the Firefox



#	Characteristics	Mean	σ(standard deviation)
1	Age (years)	22.2	2.31
2	Computer experience (years)	11.9	2.74
3	How often do I carry my cellphone (hours/day)	16.5	5.21
4	Number of password-based accounts	8.5	6.93
5	Number of accounts where password is reused	4.0	3.11
6	Number of unique passwords	3.9	2.31
7	Number of passwords that are a name or word followed by a number	2.2	1.49

TABLE 2:DEMOGRAPHIC CHARACTERISTICS

8. Conclusion:

In this paper we proposed a user authentication protocol named Opass which leverages a users cellphone and SMS to thwart password stealing and reuse attacks.

9. REFERENCES:

[1}B. Ives, K. R. Walsh, and H. Schneider, "The domino effect of pass- word reuse," Commun. ACM, vol. 47, no. 4, pp. 75–78, 2004.

[2] S. Gaw and E. W. Felten, “Password management strategies for online accounts,” in SOUPS ’06: Proc. 2nd Symp. Usable Privacy . Security, New York, 2006, pp. 44–55, ACM.

[3] D. Florencio and C. Herley, “A large-scale study of web password habits,” in WWW ’07: Proc. 16th Int. Conf. World Wide Web., New York, 2007, pp. 657–666, ACM.

[4] S. Chiasson, A. Forget, E. Stobert, P. C. van Oorschot, and R. Biddle, “Multiple password interference in text passwords and click-based graphical passwords,” in CCS ’09: Proc. 16th ACM Conf. Computer Communications Security, New York, 2009, pp. 500–511, ACM.

[5] I. Jermyn, A. Mayer, F. Monrose, M. K. Reiter, and A. D. Rubin, “The design and analysis of graphical passwords,” in SSYM’99: Proc. 8th Conf. USENIX Security Symp., Berkeley, CA, 1999, pp. 1–1, USENIX Association.

[6] A. Perrig and D. Song, “Hash visualization: A new technique to Improvereal-worldsecurity,”inProc.Int.WorkshopCryptographic Techniques E-Commerce, Citeseer, 1999, pp. 131–138.

[7]J.ThorpeandP.vanOorschot,“Towardssecuredesignchoicesforimplementing graphical passwords,” presented at the 20th. Annu.Computer Security Applicat.Conf., 2004.

[8] S. Wiedenbeck, J. Waters, J.-C.Birget, A. Brodskiy, and N. Memon, “Passpoints: Design and longitudinal evaluation of a graphical pass- word system,” Int. J. Human-Computer Studies, vol. 63, no. 1–2, pp. 102–127, 2005.

[9] S. Wiedenbeck, J. Waters, L. Sobrado, and J.-C.Birget, “Design and evaluationofashoulder-surfingresistantgraphicalpasswordscheme,” in AVI ’06: Proc. Working Conf. Advanced Visual Interfaces, New York, 2006, pp. 177–184, ACM.

[10] B. Pinkas and T. Sander, “Securing passwords against dictionary at- tacks,” in CCS ’02: Proc. 9th ACM Conf. Computer Communications Security, New York, 2002, pp. 161–170, ACM