Cryptanalysis of oPass

K.Marimuthu[1],D.Ganesh Gopal[2],Shivam Aditya3,Varun Mittal[4]

[1,2,3,4]School of Computing Science and Engineering,VIT University, Vellore-632014, Tamil Nadu, India

[1]k.marimuthu@vit.ac.in;[2]ganeshgopal@vit.ac.in;[3]varun.mittal2011@vit.ac.in; sa.[4]shivam.aditya@gmail.com

***Abstract*--The security of oPass: A User Authentication Protocol Resistant to Password Stealing and Password Reuse Attacks proposed by H.M.Sun *et al.* in IEEE Transactions on Information Forensics and Security, Vol.7, No.2, April 2012 is analyzed. Upon completion of the analysis of the paper, four kinds of attacks SMS service, attacks on oPass communication links, unauthorised intruder access using the master password and Network attacks on untrusted web browser are identified in different scenarios. Thus, we proved that oPass proposed by H.M.Sun *et al.* is not suitable for practical application.**

Keywords--password reuse attack, password stealing attack, Nonce, Intruder attack, SMS

I INTRODUCTION

Text password is the most widely used form for user authentication on websites due to it being convenient, appropriate and less complex. However, passwords of users are inclined to be pilfered and compromised under different threats and vulnerabilities. Like, users often choose some weak password known to them and use them frequently in various websites leading to password reuse attacks. Regularly reusing passwords causes a chain effect, when an attacker compromises one password, he/she will exploit it to gain access to more websites. Secondly, when a user types a password into an untrusted website, it can also lead to password thief threats. An adversary can launch a variety of password stealing attacks to snatch passwords, such as phishing, key-loggers and malware etc.

oPass scheme [1] involves 6 assumptions for evolving a protocol resistant to password stealing and password reuse attacks. They also proposed a user authentication protocol named oPass which leveraged a users cell phone and short message service (SMS) to thwart password stealing and password reuse attacks. oPass only requires each participating website possesses a unique phone number and involves a telecommunication service provider (TSP) in registration and recovery phases. Through oPass, users only need to remember a long-term password for login on all websites. It claimed that it satisfied all the requirements of the user and is immune to various attacks. In this paper, however, some security loopholes of their scheme will be pointed out and the corresponding attacks will be described.

The rest of paper is organised as follows. The Section 2.contains the review of oPass scheme with all the diagrams, working, and formulae used in the various phases. A brief description about each phase is also included in it. The cryptanalysis of oPass is presented in Section 3. Section 4.contains the conclusion in which it is proved that oPass scheme has technical faults, bugs and other defects which need to be rectified to make it more efficient and reliable for usage.

II REVIEW OF OPASS SCHEME

In this section, oPass scheme will be briefly reviewed. Some notations will be given. Then, the registration phase, the login phase, and the recovery phase of this scheme will be described in turn.

A. 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 with key k
h(.)	Cryptographic hash function
IV	Initialisation 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 $ID_u\|\|IV\|\|\{n_d\|\|n_s\}$ δ_i under the δ_i
$HMAC_3$	The HMAC-SHA1 digest of $ID_u\|\|IV\|\|\{c\|\|n_s\}$ δ_{i+1} under the δ_{i+1}

B. Registration Phase

The aim of this phase is to allow a user and a server to negotiate a shared secret in the form of OTP (One Time Password) to authenticate succeeding logins for this user.

ISBN No. 978-1-4799-3914-5/14/$31.00 ©2014 IEEE

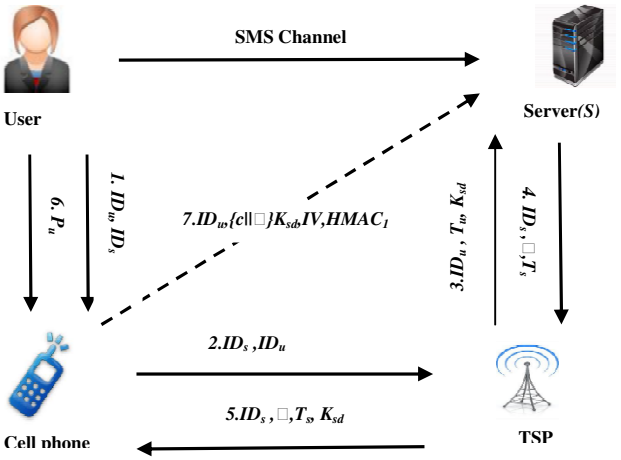

Fig 1: Registration Phase

The detailed steps of this phase are revealed as follows and depicted in Fig. 1.

1. The user open the oPass program on her cell phone. He/She enters account id ID_u and the website URL or domain name ID_s to the program.

2. The oPass program sends ID_u and ID_s to the TSP through a 3G connection to make a request for registering.

3. The TSP and the server will produce an SSL tunnel to protect the communication.
Then the TSP forwards { $ID_{u,,}$ T_u ,K_{sd} } to the assigned server S.

4. Server *S* will generate the corresponding information for this account and reply with a response message, including server's identity ID_s, a random seed ϕ, and server's phone number T_s.

5. Once response message is received, the user types a long-term password P_u into his/her cell phone. The cell phone calculates a secret credential by the following operation:

$$c = h(P_u \| ID_s \| \phi)$$

6. Then, the cell phone sends an encrypted registration SMS to the server by using phone number T_s as follows:

$$Cellphone \xrightarrow{SMS} S : ID_u, \{c \| \phi\} K_{sd}, IV, HMAC_1$$

7. After receiving the message (6), the server stores $\{ID_u, T_u, c, \phi, i\}$ and then completes the registration phase. Here, variable i indicates the current index of the one-time password and is initially set to 0.

C. *Login Phase*

The login phase begins when the user sends a request to the server through an untrusted browser by using a kiosk.

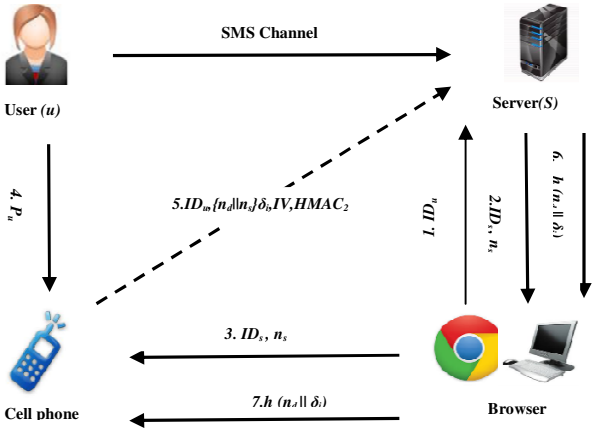

Fig 2: Login Phase

The detailed steps of this phase are revealed as follows and depicted in Fig. 2.

1. The user begins the login procedure by accessing the required website through a browser on an untrusted kiosk.

2. The browser sends a request to *S* with user's account id ID_u.

3. Then, server *S* supplies the ID_s and a fresh nonce n_s to the browser. Meanwhile, this message is forwarded to the cell phone via Bluetooth or Wireless interfaces.

4. After receiving the message, the cell phone inquiries related information from its database via ID_s which includes server's phone number T_s and other parameters $\{\phi, i\}$.

5. After this, a dialog is prompted for his/her long-term password P_u. Secret shared credential c can be regenerated by inserting the correct password P_u on the cell phone. The one-time password for current login is again recomputed using the following operations:

$$c = h(P_u \| ID_s \| \phi)$$
$$\delta_i = h^{N-i}(c)$$

6. The next action on the cell phone is sending the following SMS message to server *S*:

$$Cellphone \xrightarrow{SMS} S : ID_s, \{n_d \| n_s\} \delta_i, IV, HMAC_2$$

7. After receiving the login SMS, the server again calculates $\delta_i = h^{N-i}(c)$ to decrypt and verify the authenticity of the login SMS. If the received n_s equals the previously

generated n_s , the user is legitimate; otherwise, the server will not accept this login request.

8. Upon successful verification, the server sends back a success message $h(n_d \| \delta_i)$ through the Internet to the user's cell phone. The cell phone will verify the received message to guarantee the completion of the login process.

D. *Recovery Phase*

Recovery phase is designated for some specific conditions; for example, a user may lose his/her cell phone. The protocol is able to recover oPass setting on his/her new cell phone assuming he/she still uses the same phone number by applying for a new SIM card with old phone number. The detailed steps of this phase are revealed as follows and depicted in Fig. 3.

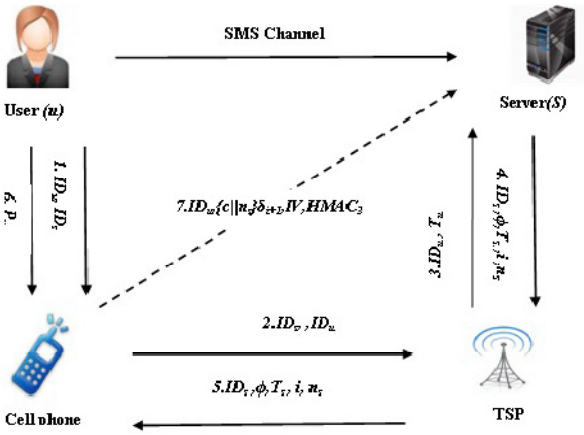

Fig 3: Recovery Phase

1. Once user *u* installs the oPass program on his/her new cell phone, she can instigate the program to convey a recovery request with his/her account ID_u and requested server ID_s to predefined TSP through a 3G connection.

2. TSP can trace his/her phone number T_u based on her SIM card and send his/her account ID_u and the T_u to server *S* through an SSL tunnel.

3. Once server *S* receives the request, it searches the account information in its database to confirm if account is registered or not. If account ID_u exists, the information used to work out the secret credential *c* will be fetched and be sent back to the user.

4. The server *S* generates a fresh nonce n_s and replies with a message consisting of{ ID_s, ϕ, T_s, i, n_s }. This message includes all compulsory essentials for generating the next one-time passwords to the user *u.*

5. 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 δ_{i+1}. During the last step, the user's cell phone encrypts the secret credential *c* and server nonce n_s to a cipher text. The recovery SMS message is delivered back to the server *S* for checking as follows

$$Cellphone \xrightarrow{SMS} S : ID_u, \{c \| n_s\}\delta_{i+1}, IV, HMAC_3$$

Similarly, the server *S* computes δ_{i+1} and decrypts this message to certify that user is already recovered. At this point, his/her new cell phone is recovered and ready to perform more logins.

III CRYPTANALYSIS OF OPASS

In oPass introduced in the previous section, the OTP is eliminated after completion of operation by user. This elimination not only enhances the security of the scheme, but also alleviates the overhead of computation and storage for the entire process. However, some security loopholes still exist and will be described in this section.

A. *Use of SMS Service*

SMS Service is slow and consumes approximately 41% time of whole operation of oPass. It also has low reliability. Messages are often delivered late and many are not even delivered due to congestion and many other reasons. It does not give an acknowledgement that the message has been received or not. Thus, the use of SMS makes oPass inefficient and sluggish to users. Keynote is a provider of on-demand test and measurement products for mobile communications and internet performance. It announced that in its two-week test period conducted by it using Keynote Mobile Interactive Testing Environment (MITE), 26,000 messages were sent and 7.5% of its text messages never reached their destinations [2-3]

B. *Attacks on oPass Communication Links*

1) *Attacks on Bluetooth Links:* In the Login phase in step 3, oPass uses Bluetooth to send the data {IDs, n_s, *h(.)* } between browser and cell phone. These are some of the essential information required for the user to log in especially the n_s. This data can be intercepted by the following attacks

- *BluePrinting Attack:* The first step to hack data from a cell phone is to gather information. The intruder can then use it to break into target cell phone using oPass. Bluetooth devices can be fingerprinted or investigated for information gathering using this technique known as

BluePrinting [4]. Using this the hackers can know various information about the device like manufacturer, model, version, etc. for the target Bluetooth enabled device. This enables the intruder to know more about the device in order to attack it using further attacks. Tools like BluePrint [4] and BTScanner [4] are used in it.

- *BlueJack Attack:* It is the process of sending an anonymous message from one Bluetooth enabled phone to another without the recipient knowing the exact source of the received message. However it has to be used within a particular range. In the oPass, the intruder can use this method to send worms and viruses into the victim's cell phone, thus enabling him to easily hack into the victim's cell phone and shut down the security programs of the cell phone. Tools like FreeJack [4] and CIHWB[4] are used in it.
- *BlueSnarf Attack*: It is the process of connecting vulnerable mobile phones through Bluetooth, without knowing the victim[4]. It involves using the OBEX protocol [4] with which we can forcibly push/pull sensitive data in/out of the victim's mobile phone. Also called OBEX pull attack, this attack is the principal method used by the intruder to bypass the security features of oPass. However, this attack requires J2ME enabled mobile phones (as a tool for the intruder). With a J2ME enabled phone, just by using Bluesnarfing tools like Blooover, Redsnarf, Bluesnarf, etc. an intruder can break into victim's mobile phone for stealing sensitive data such as address book, photos, mp3, videos, SMS etc. Using these tools, the intruder can extract vital information like { ID_u ,ID_s, n_s, ϕ, i,, Pu,} from the user. These data enables the intruder to fake as the user of oPass. The intruder can compute the one time password using

$$c = h(P_u \| ID_s \| \phi)$$

$$\delta_i = h^{N-i}(c)$$

After this the intruder can send the following SMS message to server *S* and use the account of the oPass user.

$$Cellphone \xrightarrow{SMS} S : ID_s, \{n_d \| n_s\}\delta_i, IV, HMAC_2$$

So, the server after receiving the login SMS, calculates $\delta_i = h^{N-i}(c)$ to decrypt and verify the authenticity of the login SMS. As the received n_s equals the previously generated n_s which was copied by the intruder the server thinks user to be legitimate and grants access to the intruder.

2) *Attacks on Wireless Interfaces*

In the Login phase, we use Wi-Fi to send the data {ID_u , n_s ,k } between browser and cell phone. These are some of the essential info required for the user to log in especially the n_s. This data can be intercepted by using the tools Airoway [5] and Wifislax[5].The software selects the wireless channel and wait for packets to accumulate**.** The more packets are accumulated the greater the chance that WEP (Wired Equivalent Privacy)[5] of the Wi-Fi can be hacked. After a successful attack, the key will be displayed in the bottom-right window. The attacker can then easily get into the network and steal {$i,$, Pu ,ID_u ,ID_s, n_s, ϕ } from the user's cell phone. This data is vital to the functioning of oPass and can be used by the intruder to impersonate the user during the login phase of oPass. Thus, he/she can gain control over user's accounts on the internet.

3) *Attacks on GSM modem*

Since the oPass uses a cheap GSM modem in the SMS service in registration phase to send the info between cell phone and server depicted by

$$Cellphone \xrightarrow{SMS} S : ID_u, \{c \| \phi\}K_{sd}, IV, HMAC_1$$

Thus the following attacks can happen on GSM modem since it is poor quality:

- *Man-in-the-middle attack:* This is the capability whereby the intruder puts itself in between the target user and a genuine network and has the ability to eavesdrop, modify, delete, re-order, replay, and spoof signaling and user data messages exchanged between the two parties.

In the oPass system, the use of SMS is done in various phases consisting of Registration phase, Login phase and Recovery phase. In the registration phase of oPass, an encrypted registration SMS is sent from the cell phone to the server using the following formula

$$Cellphone \xrightarrow{SMS} S : ID_u, \{c \| \phi\}K_{sd}, IV, HMAC_1$$

In the login phase of oPass, an SMS is sent from the cell phone to the server using the following formula

$$Cellphone \xrightarrow{SMS} S : ID_s, \{n_d \| n_s\}\delta_i, IV, HMAC_2$$

In the recovery phase of oPass, an SMS is sent from the cell phone to the server using the following formula

$$Cellphone \xrightarrow{SMS} S : ID_u, \{c \| n_s\}\delta_{i+1}, IV, HMAC_3$$

Thus using the SMS the server authenticates the user and completes the procedure. However, if the intruder uses Man-in-the-middle attack he can disrupt the successful transmission of the SMS and this leads to process failure thereby causing a hindrance to the operation flow of oPass system. Additionally, the trespasser can also gain access to various vital information like { $ID_{u,}$ $ID_{s,}$ $K_{sd,}$ $IV,$ δ_i,

$HMAC_1, HMAC_2, HMAC_3$ }.The required equipment is modified Base Transreceiver Station in conjunction with a modified Mobile station.

- *Impersonation of a user:* This is the capability whereby the intruder sends signaling and/or user data to the network, in an attempt to make the network believe they originate from the target user. Now, let us assume that the attacker gains access to the long-term password P_u. Using the data obtained above the intruder can compute the secret shared credential c and one- time password δ_i.

$$c = h(P_u \| ID_s \| \phi)$$

$$\delta_i = h^{N-i}(c)$$

Then the intruder can impersonate the user and send the SMS to the server using the following formula

$$Cellphone \xrightarrow{SMS} S : ID_s, \{n_d \| n_s\}\delta_i, IV, HMAC_2$$

The required equipment is again a modified Mobile station.

C. Unauthorized Intruder Access using the Master Password

In the case of phone being lost or stolen, if the snooper gets the master password by any chance, then he can use the phone indefinetly due to there being a lack of any security mechanisms except the master password. Once knowing the master password, the intruder can perform the login phase of oPass using the following operations:

i. Compute the secret shared credential between secret and server.
$c = h(P_u \| ID_s \| \phi)$

ii. Compute the OTP.
$\delta_i = h^{N-i}(c)$

iii. Compute and send the SMS from cell phone to server.

$$Cellphone \xrightarrow{SMS} S : ID_s, \{n_d \| n_s\}\delta_i, IV, HMAC_2$$

D. Network Attacks on Untrusted Web Browser

It is established that the browser is often unprotected and can be compromised. Thus data can be stolen from it. Even so, various data items like { ID_u, n_s } which is very important and used to identify phishing attacks is transferred through the browser. Thus the following incidents could happen violating an explicit or implicit security policy of the web browser.

1) Trojan horse programs

Intruders commonly trick people using Trojan Horse programs in order to install "back door" programs. These back door programs (or remote administration programs) allow other people to access and control the victim's computer without the knowledge of the victim. The intruder can then use it to change the victim's system configurations, or infect his computer with a computer virus.On Windows computers, three tools commonly used by intruders to gain remote access to the victim's computer. They are BackOrifice [6], Netbus [7], and SubSeven [8].

In oPass, the user initiates the login process by gaining access to the desired website through a browser on an untrusted kiosk. Therefore if the trespasser can hack the kiosk and install Trojan horse programs to either steal the user's secret information like ID_u and later use it to impersonate as the user and gain access to user's private accounts provided he has the long term password P_u. He/she can also get info like ID_s to shut down the server and thereby preventing transactions from taking place in the oPass system.

2) Denial of service

Another type of attack is called a denial-of-service (DoS) attack. This attack causes the victim's network to crash or become so busy processing data that he is unable to use it. It should be noted that in addition to being the target of a DoS attack, it is also possible for the victim's computer to be used as a participant in a denial-of-service attack on another system. It often causes a more serious network security incident. However, in the login phase of oPass, the attacker can use the Denial of Service (DoS) attack to crash the system of the user. Thus, the user cannot send the SMS to the server and is thus prevented from accessing the account for further operations.

3) Mobile code (Java/JavaScript/ActiveX)

There have been many cases when "mobile code" such as Java, JavaScript, and ActiveX has caused problems for the users. These are programming languages that let web developers develop code which can be executed by your web browser. Although the code is generally useful, it can be misused by intruders to gather information about the user (such as which web sites you visit) or to run malicious code on the user's computer. Thus, the attacker can use these "mobile code" to run malicious code on the user's system and gain various important data related to the oPass like $\{ID_u, ID_s, n_s, h(n_d||\delta_i)\}$ which can be later used to impersonate the user and gain unauthorised access to the user accounts.

4) Cross-site scripting

A malicious web developer may attach a script to anything sent to a web site, such as a URL, an element in a

form, or a database inquiry. Later on, when the user opens the web site, the malicious script is transferred to the user's browser. Thus, it enables the attackers of oPass to infect the system with Trojan Horses and extract data from the system.

5) Packet sniffing

A packet sniffer is a program that captures data from information packets as they travel over the network. That data may include user names, passwords, and other proprietary information that travels over the network in clear text. Thus, in oPass, the attacker can use the packet sniffing attack to capture data from the oPass transmission like { ID_u, ID_s, n_s, $h(n_d||\delta_i)$ } and use it in further attacks.

IV CONCLUSION

oPass proposed a user authentication protocol resistant to password stealing and password reuse attacks and demonstrated its immunity against various attacks. However, after review of their scheme and analysis of its security, four kinds of attacks, SMS service, Attacks on oPass communication links, unauthorised intruder access using the master password and Network attacks on untrusted web browser, are presented in different scenarios. The analyses show that the scheme is insecure for practical application.

REFERENCES

[1] Hung-Min Sun, Yao-Hsin Chen, and Yue-Hsun Lin (2012), oPass:A User Authentication Protocol Resistant to Password Stealing and Password Reuse Attacks, IEEE Transactions on Information Forensics and Security 7(2) 651 - 663
[2] http://tech-beta.slashdot.org/story/03/01/15/2154225/ sms-messaging-unreliable
[3] http://www.keynote.com/company/press room/releases 2009/04.20.09. html
[4] http://backer-angkyai.blogspot.in/2013/03/bluetooth-hacking-tools.html
[5] http://www.wikihow.com/Crack-a-Wep-Protected-Wi-Fi-With-Airoway-and-Wifislax
[6] http://en.wikipedia.org/wiki/Back Orifice
[7] http://en.wikipedia.org/wiki/NetBus
[8] http://en.wikipedia.org/wiki/Sub7