A Secure Authentication Protocol to vigilant from Password Stealing and Reuse Attacks by using Opass

Joga Venkata Hari Babu V [1], G. Lavanya [2]

[1]M. Tech Student, Department of CSE, Anurag Group of Institutions, Hyderabad, India

[2]Assistant Professor, Department of CSE, Anurag Group of Institutions, Hyderabad, India

Abstract: ***Now-a-days, most of the users are using the websites; at the time of login session user have to enter the user name and password. Here, text password is the most familiar form of user authentication on websites, due to its well-located and simplicity. However, user's passwords are prone to be stolen and compromised under different coercion and vulnerabilities. Sometimes, user may select the weak password for their remembrance and reuse the same password across the many variant websites. This may leads to the domino effects. Sometimes user may use the passwords in unauthorized computer suffers password thief coercions the password is prone to stealing attacks such as phishing, malware and key loggers etc. In this paper, a user authentication protocol named Opass is designed, that makes use of the customer's cellular phone and short message service to ensure protection against password stealing attacks. Opass requires a unique phone number that will be possessed by each participating website. The registration and the recovery phases involve a telecommunication service provider. The main concept of the project is reducing the password reuse attack. We have implemented the one time password technology, and then reduce the password validity time. The performance had improved the security.***

Keywords: user authentication, hash function, network security, one-time password, password reuse attack, password stealing attack, encryption.

1. Introduction

Many of the past few decades, text password has been adopted as the crucial mean of user authentication for websites. People have to choose their username and text-passwords when registering accounts on a website. Users have to log into the website successfully, users must evoke these passwords. Generally, password based user authentication can defend against brute-force and dictionary attacks if users select strong passwords. However, password based user authentication has a foremost problem that humans are not experts in remembering text strings. Thus, most users prefer to choose easy passwords (i.e., weak passwords) even if they know that passwords might be insecure. Another key problem is that users have a tendency to reuse passwords transversely various websites [1][2]. Password reuse causes users to lose perceptive information stored in different websites if a hacker compromises one of their passwords. This type of attack is called as password reuse attack. The above problems are caused by the harmful persuade of human factors. Therefore, it is important to get human factors into consideration when designing a user authentication protocol.

Many of the Researchers have investigated a array of technology to condense the harmful influence of personal factors in the user authentication procedure. Since humans are more adept in identifying graphical passwords than text passwords many graphical password schemes were designed to address human's password reuse problem[5]-[9]. Using password management tools is an alternative [10]. These tools automatically produce strong passwords for each website, which concentrate on password reuse and recall problems. The advantage is that users have to identify a master password to right to use the management tool.

Here, we are assisting the two technologies—graphical password and password management tool—the user authentication system still suffers from several significant drawbacks.

Even though graphical password is a enormous idea, it is not yet full-grown enough to be broadly implemented in practice [13][14] and it still vulnerable to numerous attacks. A password management tool was works well; however, common users doubt its security and thus its feel uncomfortable about using it. Furthermore, they have trouble to using these tools due to the lack of security knowledge.

Moreover the password reuse attack, it is also important to think about the effects of password stealing attacks. Adversary steal or compromise passwords and unauthorized users' identities to launch malicious attacks, gather sensitive information, perform unauthorized payment actions, or disclose financial secrets. Phishing is one of the most general and efficient password stealing attack. Many of the previous studies would have proposed schemes to protect against password stealing attacks.

Some of the researches focus on the three-factor authentication rather than password-based authentication to afford more consistent user authentication. Three-factor authentication which depends on what would you know (e.g., password), what you have (e.g., token), and who are you (e.g., biometric). To pass the authentication, the user must have enter a password and provide a pass code generated by the, and scan her/his biometric features (e.g., finger-print or pupil). Three-factor authentication is a broad resistance mechanism against password stealing attacks, but it requires proportional high cost.

Thus, two-factor authentication is more eye-catching and sensible than three-factor authentication. Even though, many of the banks sustain two-factor authentication, it still suffers from the negative persuade of personal factors, such as the password reprocess attack. Users have to commit to memory another four-digit PIN code to work together with the token.

A user authentication protocol named Opass which power of influence a user's cellphone and short message service (SMS) to thwart password stealing and password reuse attacks. It is complex to prevent password reuse attacks from any scheme where the users have to remember something(like password). The main reason of stealing password attacks is when users type passwords to untrusted public kiosks.

Therefore, the major perception of Opass is free users from having to remember or type any passwords into conformist computers for authentication. Unlike basic user authentication, Opass engages a new component, the cellphone, which is helpful to generate one-time passwords and a new communication channel, SMS, which is used to broadcast authentication messages.

Opass presents the following advantages:

- **Anti-malware**—Users are capable to log into web services without entering passwords on their kiosk. Thus, malware cannot acquire a user's password from untrusted kiosks.
- **Phishing Protection**—Allows users to effectively log in-to websites without illuminating passwords to kiosks.
- Secure Registration and Recovery—SMS aids Opass in establishing a secure channel for message swap in the registration and recovery phases.
- **Password Reuse Prevention and Weak Password Avoidance**—Users do not need to memorize any password for registering. They only maintain a long- term password for accessing their cellphones, and put down the rest of the work to Opass.
- **Cellphone Protection**—An opponent can steal users' cell phones and try to pass through user authentication. However, the cellphones are guarded by a long-term password. The opponent cannot impersonate a legal user to login without being detected.

2. Problem Definition

Usually deployed web services facilitate and develop several applications like online banking, e-commerce, social networks, and cloud computing. But user authentication is simply handled by text passwords for the bulk websites. Applying text passwords has a number of crucial drawbacks.

First, users generate their passwords by themselves. For easy remembrance, users have a tendency to choose relatively weak passwords for all websites. This behavior causes a risk of a domino effect due to password reuse. To take sensitive information on websites for a specific victim (user), an opponent can extract her/his password through compromising a weak website because she/he probably reused this password for other websites as well.

Second, humans have difficulty to remembering complex or meaningless passwords. Some websites create user passwords as random strings to sustain elevated entropy, although users are still changing their passwords to simple strings to help them recollect it. Phishing attacks and malware are coercions against password shield. Protecting an authorized password on a kiosk is infeasible when key loggers or backdoors are previously set up on it. Considering the existing mechanisms, authenticating users via passwords is not a best solution.

3. Proposed System

We proposed a user authentication, called OPass, to prevent the above attacks. The goal is to avoid users from typing their remembered passwords into kiosks. By adopting one-time passwords, password information is no longer significant. A one-time password has expired when the user completes the present session. Different from using Internet channels, OPass leverages SMS and user's cellphones to keep away from password stealing attacks. SMS is a proper and secure medium to pass on important information between cellphones and websites. Based on SMS, a user uniqueness is authenticated by websites without inputting any passwords to untrusted kiosks. User password is used to restrict access on the user's cellphone. In OPass, each user simply remembers a long-term password for access her/his cellphone. The long-term password is used to defend the information on the cellphone from a thief.

4. Architecture and Assumptions

4.1 Architecture

The Fig. 1 describes the architecture (and environment) of the OPass system. The assumptions in OPass system are as follows:

1) Each web server possesses a unique phone number. Via the mobile number, users can cooperate with each website through an SMS channel.
2) The users' cellphones are malware-free.
3) The telecommunication service provider (TSP) is a bridge between subscribers and web servers. It affords a service for subscribers to make the registration and recovery process with each web service. For example, a subscriber inputs her/his id ID and a web server's id ID to start to execute the registration phase. Then, the TSP promotes the request and the subscriber's phone number to the corresponding web server based on the received ID .
4) The TSP and the web server establish a secure sockets layer (SSL) tunnel. Via SSL protocol, the TSP can be able to verify the server by its certificate to avoid phishing attacks. With the assist of TSP, the server can obtain the exact sent from the subscriber.

5)5)If a user loses her/his cellphone, she/he can notify her/his TSP to immobilize her lost SIM card and apply a new card with the alike phone number. Therefore, the user can achieve the recovery phase using a new cell-phone.

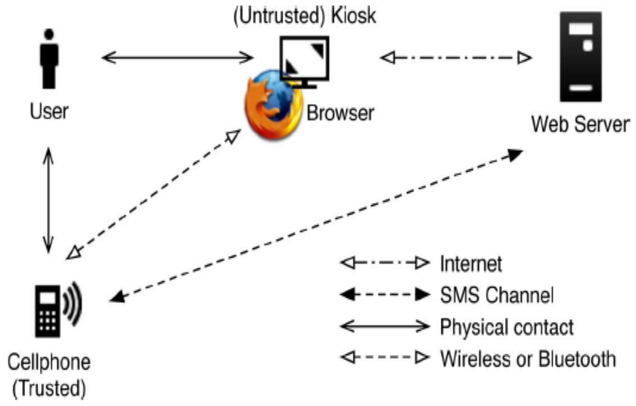

Fig. 1. Architecture of oPass system.

4.2 Overview of OPass

OPass consists of registration, login, and recovery phases.

Fig. 2 describes the operation flows of users during each phase of OPass. OPass make use of a user's cellphone as an authentication token and SMS as a protected channel. OPass and are marked in black rectangles in Fig. 2, to show difference of additional steps with normal login process.

Contrasting with common cases, login method in OPass does not need users to type passwords into an untrusted web browser. The user name is the single information input to the browser. Next, the user have to open the OPass program on her/his phone and enters the long-term password; the program will automatically generate a one-time password and send a login SMS safely to the server. The login SMS is encrypted by the one-time password. Finally, the cellphone obtains a response message from the server and confirms a success message on her/his screen if the server is able to prove her/his identity. The message is used to ensure that the website is a authorized website, and not a phishing one.

4.3 Registration Phase

Fig. 3 represents the Registration phase. This is to allow a user and a server to agree a shared secret to authenticate succeeding logins for the user. The user intiates by opening the OPass program setup on her/his cellphone. She/he enters ID_u (account id she/he prefers) and ID_s (usually the website url or domain name) to the program.

The mobile program launches ID_u and ID_s to the telecommunication service provider (TSP) through a 3G connection to make a request of registration. Once the TSP obtained the ID_u and ID_s, it can hint the user's phone number T_u based on user's SIM card. The TSP also plays the role of thirdparty to distribute a shared key K_{sd} between the user and the server. The shared key K_{sd} is used to encrypt the registration SMS with AES-CBC(algorithm). The TSP and the server S will establish an SSL tunnel to guard the communication. Then the TSP forwards ID_u, T_u, and K_{sd} to the assigned server S.

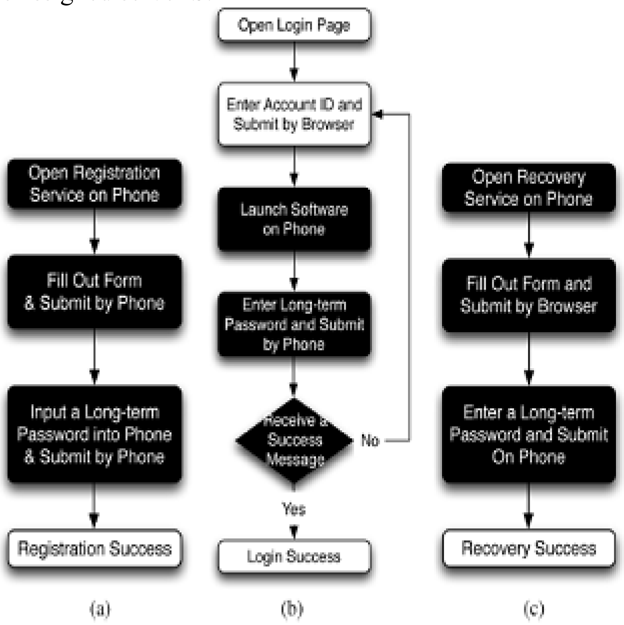
Open Login Page
Enter Account ID and Submit by Browser
Open Registration Service on Phone
Launch Software on Phone
Open Recovery Service on Phone
Fill Out Form & Submit by Phone
Enter Long-term Password and Submit by Phone
Fill Out Form and Submit by Browser
Input a Long-term Password into Phone & Submit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

Figure 2: Operation flows for users in each phase of oPass systems respectively. Black rectangles indicates extra steps contrasted with genetic authentication system (a) registration, (b) login, and (c) recovery

Server will generate the corresponding information for this account and reply a response, including server's identity ID_s, a random seed ϕ, and server's phone number T_s. The TSP then forwards ID_s, ϕ, T_s, and a shared key K_{sd} to the user's cellphone. Once reception of the response is finished, the user maintains to setup a long-term password P_u with her cellphone. The cellphone computes a secret record by the following operation:

c= (P_u||ID_s||ϕ) (5)

To prepare a secure registration SMS, the cellphone encrypts the computed credential c with the key and generates the corresponding MAC, i.e., $HMAC_1$. HMAC-SHA1 takes input user's identity, cipher text, and IV to output the MAC. Then, the cellphone sends an encrypted registration SMS to the server by phone number T_s as follows:

Cellphone ^{sms} S: $ID_{u,}$ $\{c||\phi\}\mathbf{K}_{sd,}$ IV, $HMAC_{1.}$ (6)

Server S can decrypt and verify the authenticity of the registration SMS and then obtain c with the shared key K_{sd}. Server S also compares the source of received SMS with T_u to prevent SMS spoofing attacks. At the end of registration, the cell phone stores all information {ID_s, T_s, ϕ, i}, except for the long-term password P_u and the secret c. Variable i indicates the present index of the one-time password and it is primarily set to 0. With i, the server can authenticate the user device during each login. After receiving the message (6), the server stores {ID_u, T_u,c, ϕ, i} and then completes the registration.

4.4 Login Phase

The login phase initiates when the user sends a request to the server S through an untrusted browser (on a kiosk). The user uses her/his cell phone to generate a one-time password, e.g., δ_i, and deliver necessary information encrypted with δ_i to server S via an SMS message. Based on preshared secret credential *c*, server S can validate and authenticate user based on δ_i. Fig. 4 shows the detail flows of the login phase. The protocol starts when user *u* wishes to log into her favorite web server (already registered). However, *u* begins the login procedure by accessing the desired website via a browser on an untrusted kiosk. The browser sends a request to S with *u*'s account ID$_u$. Next, server S supplies the ID$_s$ and a fresh nonce n to the browser.

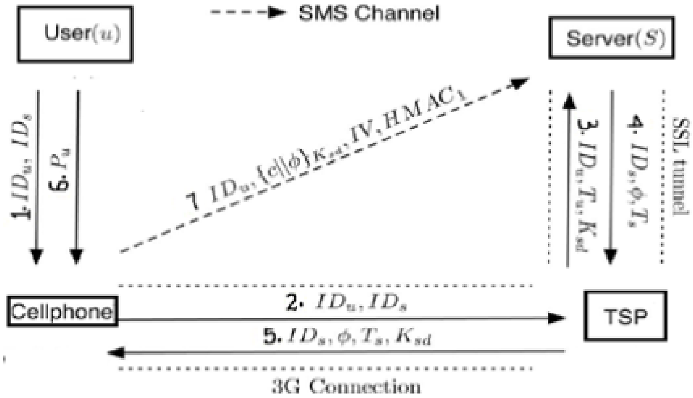

Fig. 3. Procedure of registration phase.

Meanwhile, this message is promoted to the cellphone through bluetooth or wireless interfaces(wifi). After reception of the message, the cellphone findouts related information from its database via ID$_s$, which comprises server's phone number T$_s$ and other parameters{ϕ,*i*}. The next promoting a dialog for her long-term password P$_u$. Secret shared credential *c* can be regenerated by inputting the correct P$_u$ on the cellphone.

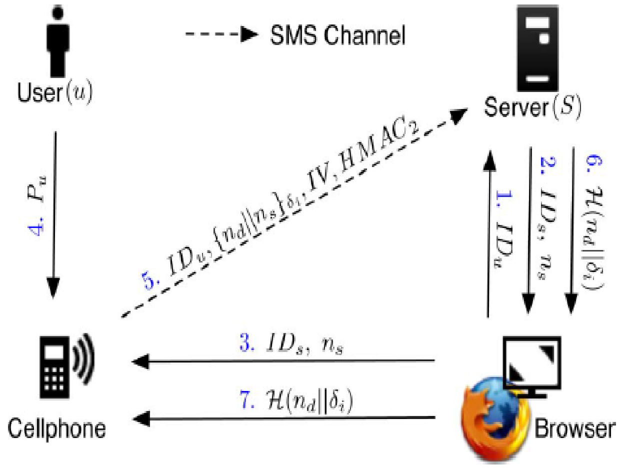
SMS Channel
User(u)
Server(S)
4. P_u
5. ID_u, $\{n_d||n_s\}_{\delta_i}$, IV, $HMAC_2$
1. ID_u
2. ID_s, n_s
6. $\mathcal{H}(n_d||\delta_i)$
3. ID_s, n_s
7. $\mathcal{H}(n_d||\delta_i)$
Cellphone
Browser

Fig. 4. Procedure of login phase.

The one-time password δ_i for current login is recalculated using the following operations:

c = (P$_u$||ID$_s$|| ϕ). (7)

=(c). (8)

δ_i is only used for this login (*i*th login after user registered) and is regarded as a secret key with AES-CBC(algorithm). The cellphone produces a fresh nonce n$_d$. To organize a secure login SMS, the cellphone encrypts n$_d$ and n$_s$ with and generates the corresponding MAC, i.e., HMAC$_2$. The next exploit on the cellphone is sending the following SMS message to server S:

Cellphone }, IV, HMAC^{sms} S : ID$_u$, {$_2$. (9)

After receiving the login SMS, the server recomputes δ_i (i.e., $\delta_{i=}$ H^{N-1} (c))) to decrypt and confirm the authenticity of the login SMS. If the collected n$_s$ equals the previously generated n$_s$, the user is rightful; if not, the server will reject the login request. Upon successful confirmation, the server sends back a success message through the Internet, H(n$_d$||δ_i), to the user device(cellphone). The cellphone will confirm the received message to ensure the completion of the login process. The last verification on the cellphone is used to thwart the phishing attacks and the man-in-the-middle attacks. If the verification is failed, the user knows the failure of login, and the device would not increase the index *i*. If the user is successfully log into the server, the index is able to automatically increased, *i*=*i*+*1*, in both the device and the server for synchronization of one-time pass- word. After N-1 rounds, the user and the server can reset their random seed by the *recovery* phase to refresh the one-time password.

Table 1: shows the Notations of Opass System

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.

4.5 Recovery Phase

Recovery phase is designated for some specific conditions; for example, a user *u* may lose her cellphone. The protocol is able to recover OPass setting on her/his new cellphone assuming she/he still uses the same phone number (apply a new SIM card with old phone number).

Once user set up the OPass program on her/his new cellphone, she/he can initiate the program to send a recovery request with her/his account ID_u and requested server ID_s to predefined TSP through a 3G connection. ID_s can be the domain name or URL link of server S. Once server receives the request, check outs the account information in its database to validate if account is registered or not. If account ID_u exists, the information used to compute the secret credential *c* will be fetched and be sent back to the user. The server generates a fresh nonce and replies a message which consists of ID_s, ϕ, T_s, *i*, and n_s. This message consists of all necessary elements for generating the next one-time passwords to the user *u*.

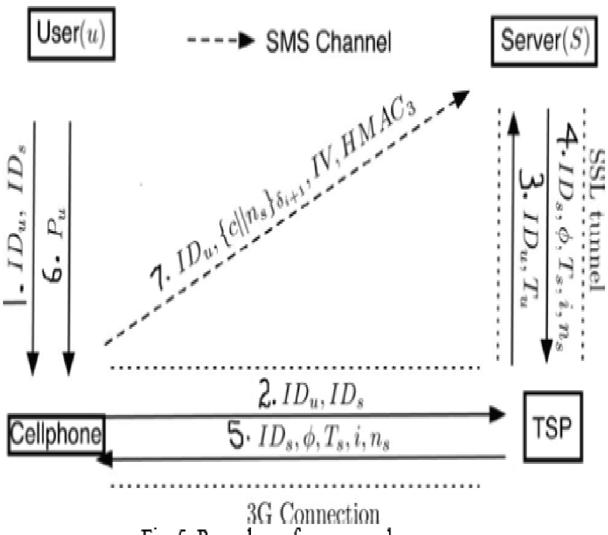

Fig. 5. Procedure of recovery phase.

When the mobile program receives the message, it forces the user *u* to enter her long-term password to reproduce the correct one-time password δ_{i+1}. During the last step, the user's cellphone encrypts the secret credential and server nonce to a cipher text. The recovery SMS message is carried back to the server for checking.

5. Conclusions

OPass which influences cellphones and SMS to thwart password stealing and password reuse attacks. We assume that each website acquires a unique phone number. We also believe that a telecommunication service provider participates in the registration and recovery phases. The design principle of OPass is to eradicate the harmful persuade of human factors as much as possible. Through OPass, each user only needs to memorize a long-term password which has been used to defend her/his cellphone. Users are free from typing any passwords into untrusted computers (kiosks) for login on all websites. Compared with previous schemes, OPass is the initial user authentication protocol to thwart password stealing (i.e., phishing and malware) and password reuse attacks (keylogger or backdoors) instantaneously. The reason is that OPass adopts the one-time password approach to make certain independence between each login. To make OPass completely functional, password recovery is also considered and supported when users lose their cellphones. They can make progress our OPass system with reissued SIM cards and long-term passwords. A prototype of OPass is also executed to measure its performance. The standard time spent on registration and login is 21.8 and 21.6 s, respectively. Besides, the performance of login of OPass is better than graphical password schemes, for example, Pass faces. Therefore, we believe OPass is believable and trustworthy for users.

References

[1] Hung-Min Sun, Yao-Hsin Chen, and Yue-Hsun Lin, "oPass: A User Au-thentication Protocol Resistant to Password Stealing and Password Reuse Attacks" ieee transactions on information forensics and security, vol. 7, no. 2, April 2012 651

[2] 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.

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

[4] T. Holz, M. Engelberth, and F. Freiling, "Learning more about the under-ground economy:Acase-study of keyloggers and dropzones," Proc. Comput-er Security ESORICS 2009, pp. 1–18, 2010.

[5] Phishing Activity Trends Rep., 2nd Quarter/2010 Anti-Phishing Working Group [Online]. Available: http://www.antiphishing.org/

[6] S. Garriss, R. Cáceres, S. Berger, R. Sailer, L. van Doorn, and X. Zhang, "Trustworthy and personalized computing on public kiosks," in Proc. 6th Int. Conf. Mobile Systems, Applications Services, 2008, pp. 199–210, ACM.

[7] M. Weir, S. Aggarwal, M. Collins, and H. Stern, "Testing metrics for pass-word creation policies by attacking large sets of revealed passwords," in Proc. 17th ACM Conf. Computer Communications Security, New York, 2010, pp. 162–175, ACM.

[8] C. Yue and H. Wang, "Session Magnifier: A simple approach to secure and convenient kiosk browsing," in Proc. 11th Int. Conf. Ubiquitous Computing, 2009, pp. 125–134, ACM.

[9] R. Biddle, S. Chiasson, and P. van Oorschot, "Graphical passwords: Learning from the first twelve years," in ACM Computing Surveys, Carleton Univ., 2010.

[10] S. Chiasson, A. Forget, R. Biddle, and P.C. van Oorschot. User Interface Design Affects Security: Patterns in Click-Based Graphical Passwords. In-ternational Journal of Information Security, 8(6):387–398, 2009.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

Author Profile

Joga Venkata Hari Babu V received the B.Tech degree in computer science and Engineering from JNTU Hyderabad in 2011 and pursuing M. Tech degree in Computer science and Engineering from Anurag Group of Institutions (Formerly CVSR College of Engineering) JNTU Hyderabad.

G.Lavanya working as Assistant Professor in Computer Science Engineering from Anurag Group of Institutions (Formerly CVSR College of Engineering) JNTU Hyderabad. She has received the Bachelor of Technology in Computer Science and Engineering from Gokaraju Rangaraju Institute of Engineering and Technology, JNTU Hyderabad and M.Tech in Software Engineering from Sree Nidhi Institute of Science and Technology, JNTU Hyderabad.