# Control Techniques for the Common Password Attacks

## CSE 543 Information Assurance & Security


### List of Group Members (Group 14)

| | | |
|---|---|---|
| Agarwal, Mridul | - | 1207054910 |
| Deshpande, Omkar | - | 1206303419 |
| Prakash, Shivam | - | 1207685709 |
| SunkesulaBhaskar, Nikhil | - | 1206286818 |
| Talasila, Madhu Meghana | - | 1207740881 |

# Contents

# 1. Introduction

## 1.1 Summary

Passwords are most popular and form the first line of defense in computer based systems. We should maintain sufficient rules to maintain sufficient security. Our survey covers approaches to overcome the below 3 attacks which cause password attacks and compromise security.

- Hacking attack
- Phishing attack
- Weak passwords

We present highlights of each approach, and compare the strengths and weakness of all approaches.

## 1.2 Background and Motivation

An astute and decisive information system is the need of the hour. In the recent past, some of the IT giants have faced the issue of password leak because of social engineering. Passwords are the basic means of authentication and authorization. It is not necessary that a technically advanced system is fully secured. Typically the passwords are alphanumeric i.e. combination of letters and digits. These types of passwords are difficult to remember. So we land up selecting weak passwords which are vulnerable to dictionary attacks and brute force attacks. To overcome such type of problem, the researchers have proposed a graphical password mechanism. The next big thing related to the password security are the phishing attacks. PhishTank(2009) provided a list of active phishing sites which show that 70% of the sites are designed to retrieve username and passwords. If the phisher obtains this data he can easily login and can further obtain sensitive information which could cause irreparable harm. A solution to this type of attack could be by performing the user authentication via one-time passwords (OTP) which is delivered through a reliable secondary communication channel on demand. The next and popular cause related to the password security breach is the hacking attack. And one of the solutions which is robust against any hacking attack is the noisy password technique. The noisy password consists of several parts - the actual password and the noisy part. It has been proven that the noisy parts

have been robust against all type of hacking attacks. This all motivated us to survey, identify and outline the solutions for avoiding the password attacks.

### 1.3 Goal and scope of study

The scope of our study involves survey of different papers that propose solutions for the following causes of password security breach.

Cause 1: Hacking attack

Cause 2: Phishing attack

Cause 3: Weak passwords

And then present a thorough analysis of present and new techniques/solutions to prevent these attacks. Thereby evaluating the identified techniques to control these causes.

### 1.4 Overview

Proposed solutions for Hacking, Phishing and Weak password for security breaches of password. The work is divided among the group members is as follows:

**Agarwal, Mridul**

1. Strengths and Weakness of different methods to deliver one-time passwords (OTP).
2. Analyzed the OTP via Instant Messaging (IM) to reduce the number of password phishing attacks.
3. Comprehensive and detailed security analysis of the OTP mechanism.
4. Examined the security related strengths and weaknesses of the given mechanism.
5. Identified possible improvements for the identified weaknesses.
6. Prepared a detailed collaborated report.

**Deshpande, Omkar**

1. What is Hacking Attack?
2. Security techniques against Hacking attack.
3. Understanding the Noisy Password Technique to make system robust.
4. Understanding the Noisy Password with Patterns for better results against hacking attacks like shoulder surfing.

5. Security Analysis of Noisy password scheme and Noisy Password scheme with Patterns.
6. Strengths and Weaknesses of Noisy Password and Noisy Password with Patterns.

**Prakash, Shivam**

1. Analyzed various password phishing techniques.
2. Studied toolbars, techniques and other countermeasures to prevent phishing attack.
3. Advantage and disadvantages of each technique.
4. Security analysis and effectiveness of each techniques.
5. Analyzed and compared each technique and identified the best among them.
6. Suggested improvement in the existing technique.

**SunkesulaBhaskar, Nikhil**

1. Brief study on need for password security, causes and existing security techniques.
2. Study of Dynamic password authentication and security system using grid analysis.
3. Analyzing different techniques like dynamic passwords, OTP and PolyPassHash scheme.
4. Examined the security related strengths and weaknesses of the above mechanisms.
5. Prepare a detailed collaborative report.

**Talasila, Madhu Meghana**

1. Password Problem.
2. Solution to Password Problem - Graphical Passwords.
3. Types of Graphical Passwords, usability and disadvantages.
4. Hybrid Method of Graphical Password.
5. Security Analysis of the Hybrid Method.
6. Advantages and Disadvantages of the Hybrid Method.

## 2. Detailed Results

### 2.1 Hacking attacks on password

What are hacking attacks?

The term hacking refers to as “bypassing” a computer system security in order to gain unauthorized access to data. There are many types of hacking attacks on password namely hijacking or impersonation, password theft, dictionary attack, shoulder surfing, video recording etc.

Existing security techniques-

Password based user authentication can counter dictionary attacks if one opts for strong password (Using symbols and numbers along with characters). However the major issue with this is that humans are not experts in memorizing complex strings. This is the reason why most of the users choose easy to remember or weak password, thus giving a way for brute force attack. The biometric technique for authentication is considered to be highly secured for authentication. These techniques use human parts like iris, face, finger prints etc. for authentication. However such techniques are costly and can result in a lot of false rejections. For example, the system may not allow access to a legitimate user because of a cut on his/her fingers. Token based authentication techniques like using ATM cards, are prone to theft attacks. One time password technique (OTP) uses different password every time the user tries to sign in. OTP mainly uses the mathematical algorithms or time synchronization for generating one time passwords. The One time passwords that use time synchronization are expensive for large system. And the one time passwords, that are not time synchronized, are highly prone to hacking attacks. Therefore authors have proposed a new technique which combines Noisy password technique with OTP to make system robust against hacking attacks.

#### 2.1.1 Noisy Password Scheme

Noisy password comprises of actual password along with few extra characters or noise. As oppose to normal static password, it cannot be used directly for authentication. Instead it has to go through the software which extracts the actual password and forwards it for regular authentication. Noisy password can be divided into

four parts- a fixed alphanumeric F, a variable alphanumeric V, a terminator X and a safeguard S. It can be defined in ordered quadruplets as follows-**$P=<T_s, V, X, F>$**

F is defined by the user and it's an alphanumeric text with length of 4 to 8 alphanumeric characters. It forms the fixed part of the noisy password and it represents the actual password. It can be represented as follows-**$F= \{f_1, f_2, f_3 \ldots f_i$ where $4<=i<=8\}$**

V is the variable length of the alphanumeric text that user enters between each of the character of the actual password F until hit by a terminator. It can be represented as follows-**$V= \{v_1, v_3, v_3 \ldots v_i$ where $0<=i<= n_j\}$**

There are few restrictions applied on V. The character that $v_k$ that belongs to V should not be equal to the any character $x_i$ that belongs to X. This restriction can be stated by the following equation-**$(v_k$ *belongs to* $V_i) != (x_i$ *belongs to* $X)$**

X is a character set that acts as a terminator. X can be represented as follows-

**$X= \{x_1, x_2, x_3 \ldots x_i$ $4<=i<=8\}$**

User is requested to enter an alphanumeric text $T_s$ of length S, after each $f_i$, where S is called as safeguard number. As oppose to the normal static password, user has to remember two things- fixed part of the noisy password and the terminator.

#### 2.1.1.1 Mechanism

Example of noisy password-

F = 2, 5, 7, 9

S = 5

X = 1, 3, 6, 8

V = any subset of alphanumeric character set.

User can enter the following variable length password-245921254923594567542 89

The system will now look for the terminators-24592**1**25492**3**5945**6**7542**8**9

The system will now extract the characters that are located immediately after the terminators-24592**12**5492**35**945**67**542**89**

#### 2.1.1.2 Security Analysis of the technique

This algorithm is robust to hacking attacks as compared to other algorithms, if it used by experts who happen to have basic knowledge about the algorithm. The variable length of the password can be as large as required. This will eliminate the use of weak

passwords and hence shield the password against dictionary attacks. Shoulder surfing will be extremely difficult because user will enter different password every time with only the fixed part F and terminator X being the same. However, its strong point can prove its weak point. This technique will prove useless if F and X are comprised and hence it is extremely important to protect F and X. This technique can also prove robust against video recording attack because every time user will enter a long and different password making it extremely difficult, almost impossible, to study the typing pattern.

However, this technique was found tedious by some users, for authentication, as compared to the simple static password authentication technique. According to the experiment carried by the authors on 12 subjects, ranging from 35 years old to 40 years old, participants had difficulty in learning the passwords and it posed a challenge to some. Due to lack of experience in using the noisy password techniques, participants were also slow in inputting the password. Choosing a password took time because of the strict conditions imposed, as stated above. Also there was significant error rate in choosing the valid password. This all factors make this technique more usable in an environment where security is more important than time. The major issue associated with this technique is implementing it on regular basis with the users that have low or no prior knowledge about this algorithm.

### 2.1.2 Noisy password with patterns

A system needs to verify the user mainly via its password in order to authenticate it. This is the main reason why most of the users tend to use simple and constant passwords. However, these users can be victims of password theft. To avoid such threats, authors have proposed noisy password technique which requires software to extract actual password from alphanumeric characters and send it to the server authentication. Noisy password P can be defined as quadruplets with four parts- a fixed part F, a variable part V and two patterns T1 and T2.

$P=<F, V, T1, T2>$ …….. (1)

F and V are same as described in the previous paper. F and V can represented as follows-

$F= \{f_1, f_2, f_3 \ldots f_i \text{ where } 4<=i<=8\}$ …….. (2)

$V= \{v_1, v_3, v_3 \ldots v_i \text{ where } 0<=i<= n_j\}$ …….. (3)

T1 and T2 are the key stroke patterns and they are going to be used for defining F and V. They can be represented as follows-

T1= {A1, D1} …….. (4)

T2= {A2, D2} …….. (5)

In the above equation, A1 represents the choice of the part that it will represent from F or V and D1 corresponds to the duration of key strokes. A2 represents the other choice (F or V) depending on A1's representation and D2 corresponds to the duration of key strokes.

The main concept behind noisy password technique is to allow user to change his/her password every time he enters by adding variable part as noise to the fixed password. Also in addition to the fixed password, user needs to remember the keystroke patterns.

#### 2.1.2.1 Mechanism

Example-F= {2, 4, 6, 8}; T1= {F, 5}; T2= {V, 1}; V = any subset of alphanumeric characters; The valid user input with variable length password and keystrokes with specified duration will be-

1(1)3(1)5(1)7(1)3(1)**2**(1)2(1)**4**(5)4(1)2(1)**6**(5)9(5)0(5)

0(1)**8**(5)0(5)6(5)0(5)7(5)7(1)7(1)9(1)9(1)9(1)3(5)1(5)2(1)3(1)4(1)3(1)4(1)4(1)

In the above input the numbers without the bracket are the key value while the numbers within the brackets are the key durations. As T1 has specified the duration of 5, system will extract the password before the duration 5. It will count for the successive keys till the duration is equal to 5 and it gets the alphanumeric character given in the F. This will give one character of the password. The system will interpret the above example as-

NNNNNN**2**N**4**NN**6**NNNNNNNNNNNN**8**NNNNNNN, where N is the noise in the password and the numbers are the actual password. System will thus extract 2, 4, 6, and 8 which is the actual password.

#### 2.1.2.2 Providing Feedback

It is important to provide user with the feedback that a key pressed has been registered. This can be done playing an audio or flashing the screen after successful key press registration. However providing feedback might leak the timing information and

information regarding the length of the password. The system can provide feedback event in only multiple of 100ms in order to reduce the amount of the timing information leaked. However in any case the feedback provided will leak some information regarding the length of the password. Therefore, it is necessary to provide additional feedback for all the noisy part after the fixed part to deceive the shoulder surfer.

#### 2.1.2.3 Security Analysis

This approach can be used for both digit-based and character-based passwords. This technique eliminates the dictionary attack. This is because user will enter different password of variable length every time and dictionary attack will not be possible because of the noisy part of the password. This technique is more resilient to shoulder surfing. It will be almost impossible for the person who is observing, the user inputting the password, to guess the actual password because he/she will not have any idea about the fixed part and the variable part of the password. It is more robust than the previous technique because here the password depends more on A1, D1, A2, D2 and not the terminators. The error rate observed with the use of feedback was less than the error rate observed without the use of feedback.

However this technique can prove tedious for some users. It requires users to have knowledge about this technique in order to use it successfully. The average time required for the user to enter the password is also comparatively more. This is mainly because user needs to remember T1 and T2 in addition to fixed part F. Also, because all of the above conditions, the error rate in password input is more as compared to the previous technique. If the feedback mechanism is not properly designed, then it will leak considerable amount of timing information and password length information to the shoulder surfer. The technique relies on database. And as with majority of the authentication techniques, this technique can prove incompetent if the database is compromised.

### 2.1.3 Strength and weaknesses

Noisy password technique with patterns requires user to remember T1, T2 along with fixed part of the password F whereas basic noisy password scheme requires user to remember F and terminator X only. As a result of this, user takes more time to input

valid password in Noisy password with pattern in comparison to the basic noisy password scheme. However, this same point makes noisy password technique with patterns more robust against shoulder surfing in comparison to the basic noisy password scheme. Both the techniques are incompetent if the database storing the user credentials is compromised. Both the techniques prove to be robust against the dictionary attack. Since both the techniques require considerable amount of time from user side for inputting the password, they should be used in a scenario where security is important than time. Given the fact that both techniques require significant amount of time, Noisy password technique with pattern is comparatively competent than basic noisy password technique because it gives less error rate for user input.

### 2.1.4 Dynamic Password Authentication and Security System using Grid Analysis (DPASS)

We researched various papers and one of the paper which had less drawbacks was "DPASS – Dynamic Password Authentication and Security System using Grid Analysis" by Balaji R and Roopak V. In this paper they discuss the conventional password system and its security problems. They reference a technique called One Time Password, where the user enters new password every time to login to the system. They propose a new technique which is dynamic and also uses the feature of OTP. The proposed system makes use of a grid which is generated dynamically every time.

#### 2.1.4.1 Implementation

The grid is made up of alphabets, numbers and special characters included multiple times. User tries to enter his password/pattern by selecting the cells of the grid. The user enters a complex character at i'th position to make the pattern complex.

The Grid Generation is based on the following algorithm[20]

Start

Set of useable characters retrieved 40 characters are selected using random and stored in new array - chars

An empty grid is created and all cells are assigned null

Create int array - cells used with data from 1 to 81

For i = 1 to 40

Select ith character k from chars

Select 2 numbers using random from array cells and fill corresponding cells in grid with k

Delete the 2 numbers from array cells

End for

Store array in database

The grids are generated according to capacity of server if the server is high end it is generated dynamically for user authentication, if the server is low end then some defined no of grids are stored in the server database and a random grid is selected for user authentication.

#### 2.1.4.2 Security Analysis

The grid generation is dynamic and the pattern entered by user also varies, this counters the replay attack. This method also counters Brute force attack efficiently as cracking the password takes years even for a 6 character password in a 9*9 Grid.

For a grid of size 9*9 grid and a 6 character user password the number of combinations would be ((9)^2)*6 = 9^12

Using Dpass user can select one character as his complex character, this will increase the no of combinations exponentially to

(9^12)*5*81 =114383962274805

Suppose a password cracking machine cracks 1000 passwords/sec even then it takes 3627.0916 years for it to crack making it secure. This increases by a factor of 9^2 when we increase as the length of password by 1 character.

### 2.1.5 PolyPasswordHasher

PolyPassword Hasher is another new method introduced by Justin Cappos which makes cracking password difficult for an attacker. This scheme offers protection in cases of hacking of password files in database.

The author assumes threat model for attackers and presents his paper based on those assumptions. This Scheme uses SHA 256 with salt for password storage and Shamir secret sharing scheme for dividing the hashed passwords and storing the individual shares separately. The author and professor mentions in the paper this combination works well with the password files and even if the attacker manages to

break into the systems and gains access to password he cannot access the system until he manages to crack a large number of password shares correctly. This makes the system robust to insider and outsider attacks on password files.

#### 2.1.5.1 Implementation

Shamir Secret share creates a random number that can be used as a key in symmetric cryptography. Shamir secret share a secret is divided into different shares and each secret share is unique and threshold shares are necessary to regenerate the secret. The hashes are generally salted and stored in database instead of that here the information in database is used to encode a share. The salted hashes are encrypted using this key. In this scheme the key is not stored as a whole and it is shared with n threshold accounts. Therefore even if attacker can gain access to n-1 threshold accounts he cannot crack the individual password even if the passwords are weak.

Traditional systems store passwords as

Username, Salt, Hash(password+salt)

Polypasswordhash stores differently as Username, Salt, (share(ShareNumber) XOR hash(salt+password))

So assuming the attacker manages to get a password for system the attacker cannot validate the correctness of password. Only way for him to know if the password is valid is to obtain threshold of shares. This increases the search space of the attacker exponentially.

Going by this author argues that if the threshold is relatively high it is impossible for current generation machines to crack the password.

#### 2.1.5.2 Security Analysis

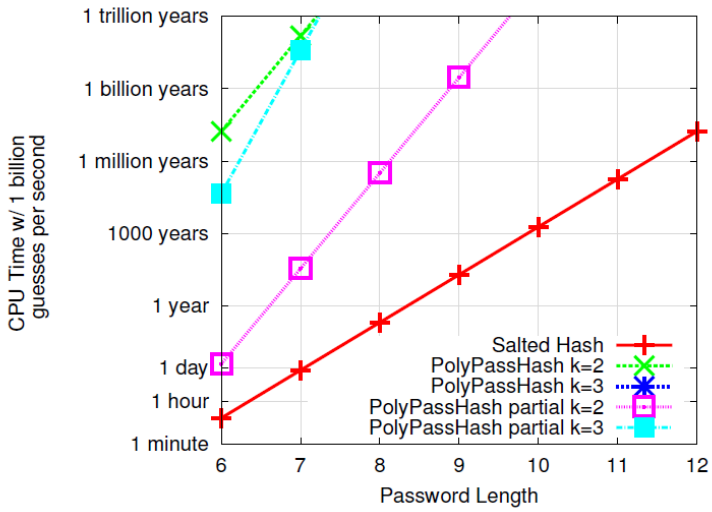


*Fig. 1 Time to crack a password given 1 billion attempts per second [22]*

We can see that when K=1 poly hash scheme performs same as traditional schemes but when threshold is 3 the guessing time increases exponentially (O(V^P)). This makes attacker impossible to guess.

**Advantages**

This scheme is efficient in terms of memory, disk space and CPU time when compared to existing techniques. Has both advantages of salted SHA256 and Shamir secret sharing scheme.

**Disadvantages**

It does not provide strong protection if weak passwords are chosen for all the threshold accounts. For this scheme to be strong enough it requires a certain threshold users. And if the threshold users forget the password then we are not able to create new shares using this scheme. The threat model of author assumes that attacker cannot read arbitrary memory on server. Suppose attacker creates arbitrary n accounts in order & obtain the threshold shares he can successfully break into system, the model does not do anything to prevent this scenario.

### 2.2 Phishing Attack

Phishing is a malicious activity whereby an attacker (phisher) tries to trick Internet users into providing confidential information (Dhamija et al., 2006). With the drastic increase in

number of internet users for performing financial transactions, phishing for password and other crucial information has grown to prominence. PhishTank (2009) provided a list of active phishing sites which show that 70% of the sites are designed to retrieve username and passwords. If the phisher obtains this data he can easily login and can further obtain sensitive information which could cause irreparable harm. Some of the trivial attacks against password are Brute-Force attack, Dictionary attack and Guessing attack. These attacks can be deterred by using strong password.

Passwords also become vulnerable when user uses a single password for every website. Hacker can break into one of those vulnerable sites and can use the retrieved password on some other important websites. Moreover, adversaries now try to phish user directly by falsely claiming to be some legitimate enterprise and trying to allure user to enter their password and other credentials on their fraudulent website.

There are several anti-phishing techniques available-

- Spam Filters.
- Anti-Phishing Tools.
- Password Protection Mechanism.
- One Time Password (OTP).

### 2.2.1Spam Filter

The most fundamental step to control Phishing is that user does not opens the fraudulent website. This is tried to be achieved by Spam filters which filters out the spam mail consisting links of fraudulent website. This is achieved by training filters. The main disadvantage of this technique is that heavily relies on user training the classifier. Also if somehow, a fraudulent mail is not detected then it will be delivered to user and user may use it.

### 2.2.2 Anti-Phishing Toolbars

Anti-Phishing toolbars predicts whether a website is fraudulent or genuine. These toolbars rely on Blacklist (List of fraudulent websites), Whitelist (List of genuine websites) and heuristics to see the difference between the URL used and to the most similar genuine URL. Some of the worth mentioning examples of such toolbars are Google Safe Browsing, McAfee Site Advisor, eBay Toolbar, Netcraft Anti-phishing

Toolbar, Spoofguard, Antiphish etc. Out of these, only Spoofguard relied on heuristics, rest were more or less dependent upon Whitelist/blacklist.

Spoofguard works by calculating score on the basis of the sum of weight given by the heuristics for each webpage. It displays red signal when the score crosses the threshold. The main concern with Spoofguard was its false rate, i.e. it sometimes returned a genuine website ass fraudulent. Also, sometime some other sites used to redirect to a genuine website[11]. In this case, Spoofguard may result in stating the genuine website as fraud. For this reason, an idea was proposed to combine this heuristics with the whitelist[11]. This method can be very effective when focusing on sites which are limited in number (for ex. Banking, Social Media, Email client). Further, Levenshtein distance (edit distance) to find the difference between user's selected URL with Whitelisted URL. As mentioned in Wikipedia,"Levenshtein distance is a string metric for measuring the difference between two sequences" [19].

#### 2.2.2.1 Analysis of Spoofguard:

Though this method is efficient when focusing on small number of websites, but this cannot be applied in general, since no of whitelisted site will be very large in number and not possible to maintain.

### 2.2.3 Password Protection Mechanism:

To deal with this security issue, many researcher and developers tried to find a solution. They basic idea was-Hashing the master password using the domain names as keys. In this way, user may use same password for every website but since it is hashed with the domain name of the website, a unique password will be generated for each site. So even if a hacker creates a web page and somehow tries to misguide user to enter the password then also he will receive a password hashed with its domain name. It is very difficult to decrypt using common attacks like Brute force, Dictionary attack etc. Some of the examples of such applications are- Password Composer, Magic Password Generator, Password Hasher, Password Generator, PwdHash.

#### 2.2.3.1 PwdHash++:

V. Reddy, V. Radha and M. Jindal in their paper, 'Client Side protection from Phishing Attack', compared these various anti-phishing tools. They evaluated these tools on the

basis of certain factors like application type, hashing algorithm, password strength, spoof proof, visibility to web page, visibility to user, and concluded Pwdhash as the most secure tool for passwords protection. Though it was a good tool which provided a decent way to secure password, they identified some shortcomings like invisibility of running app to user, visibility of the activation of app to webpage, password filling as plaintext, easily spoof-able, effects on JavaScript's of other web pages.

To overcome these shortcomings, they provided a solution which addressed to these problems and named their new tool as PwdHash++ [11].It followed a sequence of steps which laid the authenticity of website and the browser plug-in. The extension was activated when user hits special prefix (@@) on the password field of the web page. This popup window contains the identity key (asked at the time of installation from user) which makes sure that the extension popped up is genuine. Apart from this it also have a background, which is a site specific custom image which user assigns to trusted website. So if user tries to open a trusted web site and this image is not displayed then it can be figured out that the website which user is working on is a malicious one. These measure validated the authenticity of the website. After this user is supposed to enter the password through this extension. This password is hashed with the domain name of the site. The algorithm used for hashing is MD5 which is non reversible. This solution improved the efficiency of Pwdhash and made it able to counter JavaScript Attack.

#### 2.2.3.2 Hashing with a Salt value:

Another defense mechanism to protect password from phishing was proposed by researchers S. Khayal, A. Khan, N. Bibi and T. Ashraf [12]. In this paper they suggested an improvement for password hashing technique by using a salt value for hashing, which were basically current parameters of the system like Date, Time and some special characters. The algorithm used here for hashing was SHA-1. The advantage of using salt value will be that even if the hacker populates a table of hashes corresponding to each value, then also he will not be able to get to know the password of the user since the hash value is calculated after including the salt in the user's password. So even he decodes the hashed password then also he will not come to know the master password.

#### 2.2.3.3 MD5 VS SHA1

For the purpose of selection of hashing algorithm, conducted test on different data and files to find out a better hashing algorithm among MD5 and SHA-1. The comparison led to conclusion that though MD5 is faster but SHA-1 produces a larger message digest making it more secure.SHA1 are more secure than MD5 algorithms due to the following reasons-

- SHA-1 has more rounds which derives message of 80 words which makes it difficult to break than MD5 which has only one bit rotation.
- Collision attack algorithm both for MD5 and SHA-1 has been developed but it is much serious for MD5 as compared to SHA-1.

#### 2.2.3.4 Analysis of PwdHash++ and Hashing with Salt technique

Hashing with Salt value will be more effective than the previous approach of hashing the domain name with the password, since hacker knows the domain of the website but doesn't knows the salt value which Is purely based on client side. So it would be more difficult for the hacker to decode the hash code produced using this algorithm.

### 2.2.4 Moody Keyboard - Other tools to counter Phishing attack

MoodyBoard is essentially a keyboard with multi-color LED lights and Help button which warns the user through ambient security notifications [14]. When user uses a sensitive data on an unencrypted website then red light blinks on keyboard and when the mouse was hovered over the submit button alarm was triggered. Also, keyboard vibrated when entering credit card details and password over unsecure network. The advantages of using this is that user's screen will not be blocked (as in case of other toolbars) and simultaneously give warning. But according to an experiment [5], users were not able to efficiently verify the cause of the problem on the basis of the alert notification. Also only 38% of the total phishing site were detected by MoodyBoard.

### 2.2.5 One Time Password (OTP)

Another solution to Password Phishing attack could be by performing the user authentication via one-time passwords (OTP) which is delivered through a reliable secondary communication channel on demand. There are a number of methods to

deliver one-time password. These methods include text messaging, mobile phones, proprietary tokens, web based methods and paper [3].

The *text messaging* service is a ubiquitous communication channel which requires a low cost to be implemented and to make it reachable to the customers. Still there are some disadvantages related to it, the cost might not be acceptable to few of the customers. OTP via text messaging service may not be encrypted. And even if it is encrypted, the hackers can easily decrypt it. In addition to the threats from the hackers, the mobile phone operators also play a very significant role in this state of affairs. The OTP while being delivered might not be encrypted by the service providers. And moreover in case of roaming, we tend to use the services of different service providers. In such a situation more than one operator needs to be trusted. If anyone happens to get this information can easily do a man-in-the-middle attack.

Delivering OTP via *mobile phone* is quite popular as these days many of us already own a mobile phone. And also the more the cost for storing and managing one-time passwords in not that too high. This makes it more popular. But the threat related to using mobile phone is that it can be lost, damaged or stolen.

A variant of the *proprietary token* was proposed by RSA in 2006 and was described as "ubiquitous authentication", in which RSA would partner with manufacturers to add physical Secure ID chips to devices such as mobile phones. Recently, it has become possible to take the electronic components associated with regular key for OTP tokens and embed them in a credit card form factor. However, the thinness of the cards, at 0.79mm to 0.84mm thick, prevents standard components or batteries from being used [3].

The instant messaging service is a *web based* method for delivering one-time password. It utilizes the existing communication infrastructure which is already available on the internet. The advantages of using IM service are: delivers messages in real time, they are almost ubiquitous, cost effective, infrastructure is already available on the internet, and downloading the client side program and obtaining a user account are free of charge. The disadvantage is that it is not fully secured but with proper configuration and design, it can be utilized efficiently.

Another method which is used by few of the banks to deliver one-time password is the paper method. As per the paper method of delivering OTP, the bank sends the user with a list of one-time passwords which are printed on a paper. For any sort of transaction that a user does, he/she has to enter an OTP from the list being sent from the bank. These passwords are often termed as transaction authentication number (TAN). Few of the banks also get those authentication numbers delivered to user's mobile phones, in such cases it is called as mobile transaction authentication number. Though being the cheapest form of delivering the OTP, this is the most insecure method.

### 2.2.6. OTP using Instant Messaging (IM)

After analyzing all the above mentioned methods for delivering one-time passwords, the instant messaging service seems to be the most promising one. As already mentioned it delivers messages in real time, they are almost ubiquitous, cost effective, infrastructure is already available on the internet, and downloading the client side program and obtaining a user account are free of charge. Though this mechanism is also not fully secure but can be made with proper configuration and design. In the later section we will be studying the delivery of OTP using the IM mechanism and its thorough security analysis.

#### 2.2.6.1 Mechanism

Here in the given mechanism [1], the author considers an instant messaging(IM) service as the secondary communication channel. And also the primary channel, i.e. the HTTP protocol is secure. The advantages of using IM service are: delivers messages in real time, they are almost ubiquitous, cost effective, infrastructure is already available on the internet, and downloading the client side program and obtaining a user account are free of charge. The disadvantage is that it is not fully secured but with proper configuration and design, it can be utilized efficiently.

This mechanism involves two processes:

I. a registration process, and
II. a login process.

We use OTP in both of these processes, and the life span and the no. of attempts to enter the password is limited to reduce the chance that it could be guessed. The steps in the registration process are identical to the conventional registration except that the users are not asked to choose a login password. Instead, the user is required to complete an extra IM account registration. After the successful registration, user can login to the system using the OTP assigned by the website.

#### 2.2.6.2 Security Analysis for OTP

Here, we briefly describe the strengths and weaknesses pertaining to security. As far as *mutual authentication* is concerned, by the proposed mechanism website (server) authenticates user (client) by asking to enter the OTP which is already assigned by the website. As only the user knows about the IM account which is used to obtain the OTP, other users cannot guess the correct OTP. And also the user checks the sender of the authentication message through which he authenticates the website. Hence, mutual authentication is very well addressed through this mechanism.

There are three components related to the proposed solution, they are - user's website account, the website's IM account and the user's IM account. Hence, to compromise the proposed solution the attacker needs to get hold of the unrevealed relationship between these three components. It might be a possibility that a phisher might acquire the user's account name, but to compromise the given solution he needs to get hold of the IM account details as well. But we can enhance the security of the IM services by: *encrypting the instant messages*, by having a *customized interpretation of authentication messages* or we can simply *block messages from unknown users*.

Currently, among all the IM systems available only Skype offers an end to end message encryption support. So, by using an IM client which supports a built-in message encryption mechanism would guarantee user's privacy. In the proposed mechanism the authentication messages being sent are in plain text. To ensure a higher security structure the IM service should not be able to understand the message being transmitted. Having a customized interpretation of authentication message aims for every IM client to have a customized plug-ins for interpreting messages. This plug-in support will enable the website to send obfuscated authentication messages which could be only recovered with the help of the plug-in. In most of the IM clients, disabling

the messages from being received from an unauthorized users is possible. So the users must disable the feature if not by default.

This mechanism also provides security when we are accessing the website in an *untrustworthy environment*. In such cases user must have some trusted secondary device which is connected to the internet. In such a situation he might access the IM account through the trusted device to obtain the OTP and could use a publicly shared computer to log in to the website by entering the OTP received via the trusted device.

Now let's analyze the proposed solution with respect to the *Man In The Middle* attack. We have already made an assumption before that the primary channel used to deliver user's login name is secured(encrypted); therefore, OTP cannot be associated with the user's web account name by any eavesdropper. If at all the OTP is discovered then also it will be of no use as it will not be sent via the user's machine, which is identified by the IP address. OTP becomes invalid as soon as it is used by the user and the website. And more over the lifespan of OTP is too short, so even if an attacker comes to know about the username and the OTP it will be of no use owing to such a short time frame validity. Thus, by MITM attack an attacker can only retrieve optional security question and its answer. For strengthening the security, re-authentication of the user in case of a critical transaction seems to be a good solution.

Other kind of popular attack is the *IP-spoofing* attack, but is difficult to occur with the given mechanism, since primary communication channel is encrypted. So, to perform such an attack the attacker will have to first breakdown the primary communication channel. And secondly either the attacker has to be someone using the same IP or some sort of man in the middle. We have already discussed about MITM, as far as attacker using the same IP concurrently is concerned, it can be detected easily. And more over the lifespan of the OTP and the session token is of few seconds. Therefore this type of attack is very difficult to perform.

### 2.3 Weak Passwords

#### 2.3.1 Password Problem

Password are the easiest way for a hacker to get control of the system as they depend on the weakest link of the security chain “Human”. The “password problem”, as given by

Birget in [24], arises because passwords are expected to comply with two conflicting requirements. They are:

1. Passwords should be easy to remember, and the user authentication protocol should be executable quickly and easily by humans.
2. Passwords should be secure, i.e., they should look random and should be hard to guess; they should be changed frequently, and should be different on different accounts of the same user; they should not be written down or stored in plain text.

While choosing alpha numeric passwords (min. of 8 characters and combination of lowercase, uppercase, digits and special characters) user tends to compromise one of the requirements which results in “weak passwords”, prone to dictionary attack, brute force attack and soon. Some of the reasons why users tend to compromise are limitations to human’s long term memory, users might be having a number of accounts, user might not be using account from long time and some accounts need password to be changed periodically for the sake of security. One of the solution to password problem is “Graphical Passwords”.

### 2.3.2 Graphical Passwords

In graphical passwords user selects a series of images from the images displayed on series of screens in Graphical User Interface (GUI). This is also known as Graphical based user authentication. The displayed images might be most common objects like fruits, places, animals, human faces etc. Graphical passwords are easier and more secure than text passwords because they are easy to memorize and difficult for the hacker to hack them. Graphical passwords are vulnerable to Brute force attack, Dictionary attacks, Guessing, Spy-ware, Shoulder surfing and social engineering. Graphical Passwords are broadly classified into 4 categories:

1) **Recognition based Systems** – In this type of systems, user first selects a sequence of images during registration phase. In authentication phase user has to recognize the selected set of images in the same order.

2) **Pure Recall based Systems** – In this type of systems, user has to draw an image or a sketch on a grid. Grid points are stored into the database during registration. User has to redraw the image or sketch to get authenticated

3) **Cued Recall based Systems** – Similar to Pure recall based systems but a clue can be selected during registration and can be used during authentication.

4) **Hybrid Systems** – A combination of one or more of the above systems.

#### 2.3.2.1 Recognition based Graphical Password

| Recognition based | Usability | Disadvantage |
|---|---|---|
| Déjà vu (Hash Visualization Technique) | During authentication, user has to select previously seen images | Server needs to store a large amount of pictures, delaying the authentication process |
| Pass-Objects (Deals with shoulder surfing) | To be authenticated, user has to recognize pass-objects and click inside the convex hull formed by pass-objects | It makes the login process slow. |
| Passfaces | To be authenticated user has to recognize human faces | Users tend to choose people faces from the same race, makes it predictable. Also cannot be used by people who are face blind |

*Table 1 Recognition based Graphical Password*

#### 2.3.2.2 Recall based Graphical Password

| Pure Recall Based | Usability | Disadvantage |
|---|---|---|
| Passlogix | During authentication, user has to draw password in the same grid and in same sequence | Hard to remember and easy to guess. Password space is small. Difficult for people with poor vision |

| | | |
|---|---|---|
| Draw-A-Secret (DAS) | To be authenticated, user has to draw the sketch in the grid as one did during registration phase in the same order. | It is vulnerable to shoulder surfing, brute force attack. Problem might arise because of grid coordinates. |
| PassPoints | To be authenticated user has to click on the region and in the same order as one did during registration. | It takes time for user to choose the exact region where to click. Difficult to operate with standard input device keyboard. Difficult for people with poor vision |

*Table 2 Recall based Graphical Password*

#### 2.3.2.3 Hybrid Systems

Hybrid system is a combination recognition based system and recall based system. According to [30] hybrid system is an approach towards more reliable, secure, user-friendly, and robust authentication.

##### 2.3.2.3.1 Working

This system comprises of 9 steps. Steps1-3 are registration steps and steps 4-9 are the authentication steps. The first step is to type the user name and a textual password which are stored in the database. During authentication user has to give that specific user name and textual password in order to log in. In second step objects are displayed to user and he selects minimum of three objects from the set and there is no limit for maximum number of objects, by using one of the recognition based schemes. The selected objects are then drawn by user, which are stored in database with the specific username. Objects may be symbols, characters, auto shapes, simple daily seen objects etc. In third step during authentication, user draws pre-selected objects as his password on a touch sensitive screen with a mouse or a stylus. This will be done using the pure recall based methods. In fourth step, the system performs pre-processing. In fifth step, the system gets the input from user and merges strokes in the user drawn sketch. In sixth step, after stroke merging, system constructs the hierarchy. Seventh step is sketch simplification. In the eighth step three types of features are extracted from the sketch

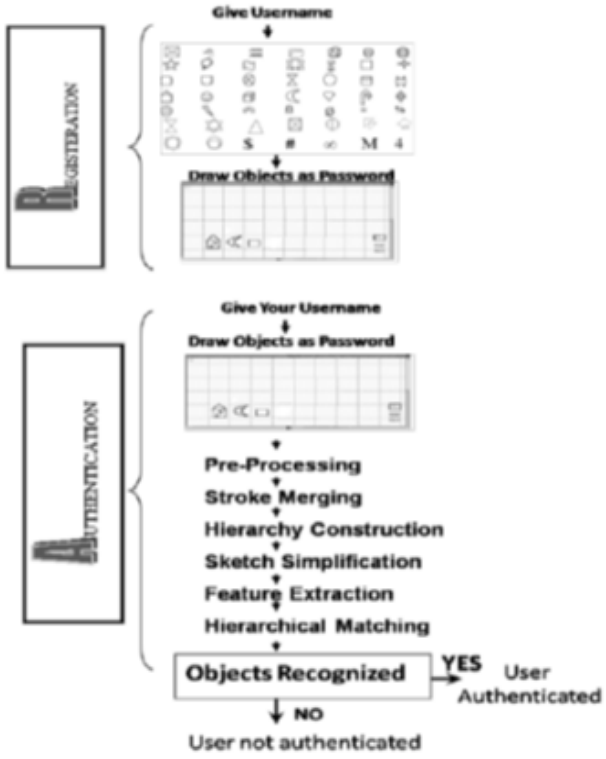


*Fig. 2 Graphical Representation of Hybrid System [30]*

drawn by the user. The last step is called hierarchical matching. As shown in Figure 1 [30].

#### 2.3.2.3.2 Security Analysis

The hybrid system is less vulnerable to brute force attack as the password space is large. It is also less vulnerable to dictionary attack, as if user tends to maintain weak text password, the graphical password makes the password strong and cannot be guessed. It overcomes the problem of shoulder surfing as a combination of techniques are used, even one is compromised it is hard for the hacker to guess other one. Social Engineering also does not work with this method as images are used. User cannot note down the password or even cannot not share the password.

#### 2.3.2.3.3 Advantages and Disadvantages

Advantages: Hybrid system is resistant to all the possible attacks on graphical passwords. People who are face-blind can also use this method as objects are used as images.

Disadvantages: This system also suffers from inherent weakness of graphical passwords i.e., people with poor vision, poor motor control, and color blindness cannot

use graphical passwords effectively. As other graphic password schemes, this method is also slow as it takes more time for normalization and matching.

## 3. Conclusion

Authentication schemes that use passwords for authentication of the user are prone to hacking attacks. Hacking refers to as “bypassing” a computer system security in order to gain unauthorized access to data. There are many types of hacking attacks on password namely hijacking or impersonation, password theft, dictionary attack, shoulder surfing, video recording etc. There are many techniques available for countering hacking attacks. In our survey we focused mainly on two techniques- Noisy Password Scheme and DPASS – Dynamic Password Authentication and Security System using Grid Analysis.

The basic Noisy password scheme made use of variable password along with fixed password to make the scheme dynamic. Terminators were used for extracting the actual password. This technique proved to be robust against hacking attacks like shoulder surfing, dictionary attack, brute force attack etc. The variable part of the password made it extremely difficult for malicious users to attack passwords protected by noisy password mechanism. However, this technique was not found easy by some of the users. Users were required to have some basic knowledge of the technique before inputting the password. The time required for users to enter the password was comparatively high with respect to simple static password authentication. Also the error rate of the users while inputting the password was high. The Noisy password scheme with pattern made use of key strokes and durations to differentiate variable part from the fix part. This technique proved to be more robust than the previous technique while tackling hacking attacks like shoulder surfing. User had to remember T1 and T2 along with fixed part. It was almost impossible for attacker, observing the user inputting the password, to guess the password. However, similar to the basic noisy password scheme this technique also required users to have basic knowledge of the technique before using it. Hence the time required for user to input valid password while registering was also high similar to basic noisy password scheme.

Both the techniques proved robust against hacking attacks but lacked in terms of ease of utilization. Hence they should be implemented in a scenario where security is more important than time required for authentication.

Poly Password Hasher scheme is efficient in terms of memory, disk space and CPU time when compared to existing techniques. This scheme uses SHA256 and Shamir secret value concepts in sharing of password file so it has advantages of both salted SHA256 and Shamir secret sharing scheme. However it does not provide strong protection if weak passwords are chosen for all the threshold accounts. For this scheme to be strong enough it requires a certain threshold users. And if the threshold users forget the password then we are not able to create new shares using this scheme. The threat model of author assumes that attacker cannot read arbitrary memory on server. Suppose attacker creates arbitrary n accounts in order to obtain the threshold shares he can successfully break into system, the model does not do anything to prevent this scenario.

DPass uses a dynamic grid similar to OTP and shares the advantages of OTP. Dpass is dynamic and generates a new grid for every time a user tries to logs in. The author also says that remembering a pattern or shape is easy for the user. Even if an attacker gain unauthorized access to the system he only has access to system until the password is changed. But the paper says that the chances of this case happening is very low. The author says that for a low end server defined number of grids are generated and stored. In that case grids must be repeated used. However suppose the attacker gets the grid id he can sniff large amount of authentication packets for same Grid id from the user and guess the pattern as there is only change of complex character.

Another attack on password which we discussed was Phishing attack. Although there are a number of methods of detecting and protecting from this attack but it is not possible to do so for all phishing sites. These days spam filters are used by all major email-clients to remove the suspected phishing mails. Still there is huge possibility that the user might land up on a fraudulent page and enter his/her password. In this case,

both Anti-Phishing tools and Password protection mechanism may be helpful. One such Anti-phishing tool was an upgraded version of Spoofguard used to tell user if a site is safe or not. One major limitation with this tool was that, it is heavily dependent on whitelist to improve its prediction of phishing site apart from heuristics and therefore this cannot be applied in general. This limitation was removed by Hashing with Salt technique because even if a user enters a password on fraudulent website, phisher gets a hashed password which is very difficult to decode. Although this mechanism is prone to shoulder surfing and toolbar needs to be installed by the user on every system he uses for accessing the websites for which hashed password was used.

On the other hand, the OTP mechanism reduces the number of password phishing attacks by utilizing the mechanism of one-time passwords which is far more advantageous than the traditional static passwords. The key attribute for using the OTP is its dynamic nature. Thus, any website can take advantage of this methodology to lessen such type of attacks. All we need to do is installing an IM bots at the server side only which makes it really cost effective. In this way user don't need to install a toolbar for Password hashing and problem of shoulder surfing is also eliminated. The only drawback with this solution is the attackers targeting the instant messaging accounts, but it is relatively easy to detect such types of attacks with the existing anti-phishing techniques. As the user will no longer be required to enter any website through their static passwords, the websites will become less prone to password phishing attacks. Therefore this technique can be regarded as more effective than the rest against phishing attacks for password.

Now - a - days with the advancements in technology, dependency on the web applications is increasing drastically. From grocery to bank transactions everyone is almost depending on the web applications. Each applications needs user to login to maintain an account. Most of the users with the intention to make their job easier, either using same password for all the accounts or using passwords which are easy to remember. This results in weak passwords. It takes very less time for hackers to know the password using dictionary attack. This forms the basis for having strong passwords

which are easy to remember and hard to crack and less vulnerable to shoulder surfing and social engineering. One such mechanism is “Graphical Passwords”.

Graphical Passwords uses images to authenticate the user with the system. This kind of mechanism is developed based on a fact that images can be remembered easily and for a long period when compared to text. Also, images are difficult to crack using social engineering. A number of graphical password mechanisms are proposed based on recognition and recall.

In recognition based schemes (using human faces), a lot of overhead is on the server to save the images, to map them during authentication. Humans tends to choose faces of their own ethnicity which gives hacker a chance to guess the password in less number of attempts. Also people with face blind disability cannot use this system. Recall/ Cued recall based systems are prone to shoulder surfing. Moreover, studies proved that users are feeling equally difficult to remember the pattern when compared to text. To overcome these disadvantages, hybrid system has been proposed.

Hybrid System uses a combination of recognition and recall based techniques or alphanumeric and graphical passwords. As combination of text and images are used, it is difficult for the user to hack. Even if the user tends to give weak text password, user has to user one kind of graphical password scheme to complete the process. So system becomes highly secure. Dictionary attack is almost impossible as images are used in this method. Also, images we use in this system are most common objects like books, flowers, animals and soon. This makes hacker hard to guess and people who are face blind can use this system, which solves the problem with recognition based scheme. As images and text are used together, shoulder surfing also becomes difficult. Which solves the problem of recall based schemes.

Hybrid systems suffers with the inherent weakness of Graphical Passwords. Although system becomes highly secure and easy for user to access, but people with

poor vision and less machine control (stylus) cannot effectively use this system. Also, this system works slowly as it takes time for normalization.

In conclusion, all the surveyed techniques have their own advantages and few disadvantages. On comparing all these algorithms with respect to countering attacks like Dictionary attack, brute force attack, shoulder surfing, social engineering as well as taking into consideration the human factors like error rates, ease of utilization, authentication time, also considering technical factors like storage space, query retrieval time and accuracy of the system, OTP proves to be comparatively more competent and efficient. The only drawback with this solution is the attackers targeting the instant messaging accounts, but it is relatively easy to detect such types of attacks with the existing anti-phishing techniques. As the user will no longer be required to enter any website through their static passwords, the websites will become less prone to password attacks.

## 4. References

[1] Chun-Ying Huang; Shang-PinMa; Kuan-TaChen, Using one-time passwords to prevent password phishing attacks. Journal of Network and Computer Applications 34(2011)1292–1301

[2] Ahmad Alamgir Khan, Preventing Phishing Attacks using One Time Password and User Machine Identification. International Journal of Computer Applications (0975–8887) Volume 68– No.3, April 2013

[3] K.Aravindhan; R.R.Karthiga, One-time Password: A Survey. International Journal of Emerging Trends in Engineering and Development. Issue 3, Vol.1 (January 2013). ISSN 2249-6149

[4] PhishTank.PhishTank:jointhefightagainstphishing,2009.[online] /http://www.phishtank.com/S. URL:/http://www.phishtank.com/S.

[5] DhamijaR,TygarJD,Hearst M. Why phishing works.In:CHI'06:proceedings of the SIGCHI conference on Human factors in computing systems. NewYork,NY, USA: ACM;2006.p.581–90.

[6] Khaled Alghathbar (ghathbar@coeia.edu.sa),Hanan A. Mahmoud (hanan.hosni@coeia.edu.sa) , Noisy Password Scheme: A New One Time Password System, Center of Excellence in Information Assurance, King Saud University, Riyadh, Kingdom of Saudi Arabia [2010].

[7] Minakshi Bhardwaj, G.P Singh, Types of Hacking Attack and their Counter Measures, International Journal of Education Planing& Administration, Volume 1, Number 1 (2011), pp. 43-53, Research India publications, http://www/ripublication.com/ijepa.htm

[8] Adeka, M.; Shepherd, S.; Abd-Alhameed, R., "Resolving the password security purgatory in thecontexts of technology, security and human factors", Computer Applications Technology (ICCAT), 2013International Conference on DOI: 10.1109/ICCAT.2013.6522044, Publication Year: 2013, Page(s): 1-7

[9] Khaled Alghathbar (ghathbar@coeia.edu.sa), Hanan A. Mahmoud (hanan.hosni@coeia.edu.sa), Noisy Password Security Technique, Center of Excellence in Information Assurance, King Saud University, Riyadh, Kingdom of Saudi Arabia [2010].

[10] Manu Kumar, Tal Garfinkel, Dan Boneh, Terry Winograd “Reducing Shoulder-surfing by Using Gaze-based Password Entry,” Symposium On Usable Privacy and Security (SOUPS) 2007, July 18-20, 2007, Pittsburgh, PA, USA.
[11] Reddy, V.P.; Radha, V.; Jindal, M. Client Side Protection from Phishing attack. International Journal for Advanced Engineering Sciences and Technologies 3.1 (2011):39-45
[12]Khayal, S.H.; Khan, A; Bibi, N.; Ashraf, T.Analysis of password login phishing based protocols for security improvements. Emerging Technologies, 2009. ICET 2009. International Conference, Serbia, 19-20 Oct. 2009.
[13] Ross, Blake, Collin Jackson, Nick Miyake, Dan Boneh, and John C. Mitchell. "Stronger Password Authentication Using Browser Extensions." 14th USENIX Security Symposium — Technical Paper. N.p., n.d. Web. 13 Oct. 2014.
[14]Luca A. D., Frauendienst B., Maurer M., Seifert J., Hausen D., Kammerer N., Hussmann H,. 2011. Does MoodyBoard make internet use more secure?Evaluating an ambient security visualization tool. In Proceedings of the 2011 annual conference on Human factors in computing systems (CHI '11). ACM, New York, NY, USA, 887-890.
[15] Agarwal, V. K.; Bharti, B. T.; Parihar, B. Password Authentication with Secured Login Interface at Application Layer. International Journal of Computer Science and Network Security 14.9(2014):82-85.
[16] Zeydan, H. Z.; Selamat, A.; Salleh, M. Study on Protection against Password Phishing. World Applied Sciences Journal 35:5 (2014):797-801.
[17] Gouda, M.G., Liu, A.X., Leung, L.M., Alam, M.A.; SPP: An anti-phishing Singlepassword protocol. Computer Networks 51(13), 3715–3726 (2007).
[18] Wu, Y.; Zhao, Z. “Enhancing the security of Online Transaction with CAPTCHA Keyboard”. Information Security and Privacy Research 376(2012):531-536.
[19] Levenshtein Distance (n.d.). In Wikipedia. Retrieved November 5, 2014, from http://en.wikipedia.org/wiki/Levenshtein_distance.
[20] Balaji. R, Roopak. V, "DPASS: Dynamic password authentication and security system using grid analysis", Electronics Computer Technology (ICECT), 2011 3rd International Conference on, vol.2, no., pp.250-253, 8-10 April 2011

[21] http://cacm.acm.org/news/177328-new-protection-scheme-makes-weak-passwords-virtually-uncrackable/fulltext
[22] Justin Cappos. PolyPasswordHasher: Protecting Passwords In The Event Of A Password File Disclosure http://polypasswordhasher.github.io/PolyPasswordHasher/
[23] http://engineering.nyu.edu/press-release/2014/07/29/crack-password-not-billion-years
[24] Wiedenbeck, S., Waters, J., Birget, J.C., Brodskiy, A. and Memon, N. Authentication using graphical passwords: Basic Results. Proc. Human-Computer Interaction International 2011, in press.
[25] http://clam.rutgers.edu/~birget/grPssw/
[26] Ahmad Almulhem, “A Graphical Password Authentication System”, World Congress on Internet Security (WorldCIS-2011), London, UK, February 21-23, 2011.
[27] Haichang Gao, Zhongjie Ren, Xiuling Chang, Xiyang Liu, Uwe Aickelin, “A New Graphical Password Scheme Resistant to Shoulder-Surfing”, 2010 International Conference on CyberWorlds, Singapore, 20-22 October 2010.
[28]Wei Hu,Xiaoping Wu, GuohengWei, “The Security Analysis of Graphical Passwords” International Conference on Communications & Intelligence Information Security, 2010.
[29] Elizabeth Stobert, Robert Biddle, “The Password Life Cycle:User Behavior in Managing Passwords”.
[30] Wazir Zada Khan, Mohammed Y Aalsalemand Yang Xiang. “A Graphical Password Based System for Small Mobile Devices”IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 2, September 2011.
[31] MohdJali, Steven Furnell, and Paul Dowland, “Quantifying the Effect of Graphical Password Guidelinesfor Better Security”