User Authentication by Secured Graphical Password Implementation

Samir Kumar Bandyopadhyay
Member *IEEE*, Department of Computer Science and Engineering, University of Calcutta, Senate House, 87 /1 College Street, Kolkata-700073, India,
skb1@vsnl.com

Debnath Bhattacharyya
Computer Science and Engineering Department, Heritage Institute of Technology, Anandapur,
Kolkata – 700107, India,
debnathb@gmail.com

Poulami Das
Computer Science and Engineering Department, Heritage Institute of Technology, Anandapur,
Kolkata–700107, India,
dippolami@yahoo.com

***Abstract*- Most common computer authentication method is traditional 'User Name' and 'password'. There are numerous Biometric authentication methods also proposed. In this paper, we propose a novel and new alternative; Figure as a Password. In this scheme users will get complete freedom to select their Passwords.**

In this paper, we conduct an extensive survey of the existing graphical password schemes and propose an alternate scheme. Entire work can be divided into three phases- a. sampling of users passwords, processing and storage; b. security on transmission; and c. Recognition and authentication. Our scheme supports the application environment and we strongly believe that "User Authentication by Secured Graphical Password Implementation" could be a solid platform for future research and study.

Keywords: Security, authentication, biometric, graphical, challenge set and multigrid.

I. Introduction

A graphical password is a secret that a human user inputs to a computer with the aid of the computers' graphical input (e.g., mouse, stylus, or touch screen) and output devices. Graphical Password can be formed in the combination of Image Icons or Pictures.

Information and computer security is dependent on passwords for the authentication of users and very common practice till date. Password design methods include text passwords, biometrics, etc. Biometric password scheme cannot be used everywhere commonly. Text based password is popular, easy to implement and common. Password scheme, generally with the following features:

1. Passwords should be easy to remember.
2. User authentication protocol should be executed quickly and easily.
3. Passwords should be secure (random, hard to guess and not in plain text).

Text based password scheme is lacking the above points mostly. Graphical passwords may be a solution to the password problem. Greg Blonder pioneered the idea of graphical passwords in 1996. His idea is to let the user click (with a mouse or stylus) on a few chosen (pre-designed) regions in (pre-processed) an image that appears on the screen.

Conventionally, Text based password scheme follows the following guidelines:

1. At least 8 characters long and alphanumeric.
2. Should not be easy to relate to the user (e.g. last name, phone number, birth year).
3. Should not be a word that can be found in a dictionary or public dictionary.
4. Should combine upper and lower case letters and digits.

User may face the following difficulties, like,

1. Watching a user log on as they type their password.
2. Dictionary attacks.
3. User may forget the password if it is too long or complicated or the password remain unused for a long time.

Advantages of using Graphical Passwords:

1. Human brains can process graphical images easily.
2. Examples include places we visited, faces of people and things we have seen.

Difficult to implement automated attacks (such as dictionary attacks) against graphical passwords.

II. Earlier Works

Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy and Nasir Memon, 2005, described PassPoints [1] in their research paper, a new and more secure graphical password system. They reported an empirical study comparing the use of PassPoints to alphanumeric passwords. Participants created and practiced either an alphanumeric or graphical password. The results have shown that the graphical password users created a valid password with fewer difficulties than the alphanumeric users.

Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy and Nasir Memon have expanded their work [2] by studying the effect of tolerance, or margin of error, in clicking on the password point and the effect of the image used in the password system. In their tolerance study, results shown that accurate memory for the password is strongly reduced when using a small tolerance (10 x 10 pixels) around the user's password points.

Julie Thorpe and P.C. van Oorschot noted [2004] that a very significant proportion of the DAS (Draw-A-Secret) password space depended on the assumption when users have chosen long passwords with many composite strokes.

978-4-88552-226-0/08/$25.00 ©2008 IEICE

If users have chosen passwords having 4 or fewer strokes, with passwords of length 12 or less on a 5 x 5 grid, instead of up to the maximum 12 possible strokes, the size of the DAS password space reduced from 58 to 40 bits. Additionally, they marked a similar reduction when users have chosen no strokes of length 1. To strengthen security, they proposed a technique and described a representative system that gained up to 16 more bits of security with an expected negligible increase in input time [3].

Wei-Chi Ku and Maw-Jinn Tsaur, 2005, proposed a scheme [4] for graphical password. This can withstand the replay attack, the password- file compromise attack, the denial-of-service attack, the predictable n attack, and the insider attack. In particular, the proposed scheme is easily reparable.

Xiaoyuan Suo, Ying Zhu and G. Scott. Owen, in 2005, have published a survey paper on graphical password. This survey will be useful for information security researchers and practitioners who are interested in finding an alternative to text-based authentication methods [5].

Kanako Yokota and Tatsuhiro Yonekura, 2005, proposed "COMPASS" (COMmunity Portrait Authentication SyStem) that uses a portrait as the authentication image, to solve the issues of the previous graphical passwords. Moreover, COMPASS has the idea of the "community authentication" that a member of a certain community consisting of a party of human can be authenticated [6]. The validity of the proposed method is confirmed by a set of subject experiments.

Jean-Camille Birget, Dawei Hong and Nasir Memon, 2006, have generalized Blonder's graphical passwords to arbitrary images and solved a robustness problem that this generalization entails. The password consisting of user-chosen click points in a displayed image. In order to store passwords in cryptographically hashed form, they needed to prevent small uncertainties in the click points from having any effect on the password. They achieved this by introducing a robust discretization, based on multigrid discretization [7].

Fawaz A Alsulaiman and Abdulmotaleb El Saddik, 2006, explained a scheme [8], based on a virtual three-dimensional environment. Users navigate through the virtual environment and interact with items inside the virtual three-dimensional environment. The combination of all interactions, actions and inputs towards the items and towards the virtual three-dimensional environment constructs the user's 3D password. The 3D password combines most existing authentication schemes such as textual passwords, graphical passwords, and biometrics into one virtual three-dimensional environment.

Fabian Monrose and Michael K. Reiter, 2005, produced [9] sufficient supporting material for Graphical Password. They analyzed and presented several Graphical Password design and security.

On the simple DAS scheme [10] the user draws a design on a display grid, this design may include block text as well as graphical symbols and shapes. Motivating by the fact that users tend to draw lines and geometrical shapes on specific areas in the grid. In this scheme, the user draws a design on a display grid, which is used as the password. The latter means that strokes can start anywhere and go in any direction, but must occur in the same sequence, as the one in the registration phase. To produce a password, each continuous stroke is mapped to a sequence of coordinate pairs, by listing the cells through which it passes, in the order in which it traverses the cell boundary. Exact repetition allows for the password to be stored as the output of a one-way hash function. To avoid ambiguity in cases of strokes that run along the cell boundaries, the size of each cell must be sufficiently large to provide a degree of tolerance when the user draws a password. Jermyn et al. suggest that the size of the password space for graphical passwords formed using a 5 x←5 grid is larger than that of alphanumeric passwords. Figure-1 illustrates a four-stroke password entry.

D. Nali and J. Thorpe [12] claimed that the password centering effect could make the DAS passwords vulnerable to attacks. The survey conducted by them, showed that 86% of the users drew a centered or approximately centered password. Furthermore, 45% of the passwords were totally symmetric and 29% of the passwords were invalid (strokes cross the cell corners or follow the grid lines). However, Jermyn et al. [10] suggested that the size of the cell space for graphical passwords in the grid must be sufficiently large. Figure-2 illustrates an example of a centered password.

GoldBerg et. al [13] conducted a study in 2002, to compare the text password login success ratio with that of the DAS password login success ratio. The results showed that about 58% of the users recalled their text-passwords correctly and 52% of them successfully recalled their DAS passwords. Another interesting result of this survey was that too many users forgot their stroke order. Consider Figure-3, for DAS Checking.

R. Dhamija and A. Perrig, in 2000, proposed a scheme [11], in this scheme a user selects a set of images that form his portfolio set. Then, at login phase, the user is shown a randomly generated set of images called the challenge set. The challenge contains some images of the user's portfolio set. For a successful login, the user has to select the images that belong to his portfolio. Figure-4 illustrates an example of challenge set.

"Passface" is a technique developed by Real User Corporation [14]. The basic idea is as follows. The user will be asked to choose four images of human faces from a face database as their future password. In the authentication stage, the user sees a grid of nine faces, consisting of one face previously chosen by the user and eight decoy faces (figure-5). The user recognizes and clicks anywhere on the known face. This procedure is repeated for several rounds. The user is authenticated if he/she correctly identifies the four faces. The technique is based on the assumption that people can recall human faces easier than other pictures. Table-1 shows the advantages and disadvantages of various Graphical Password Schemes available currently.

III. Our Work

Patrick, Long and Flinn explained the three major areas of human computer interaction, namely, authentication,

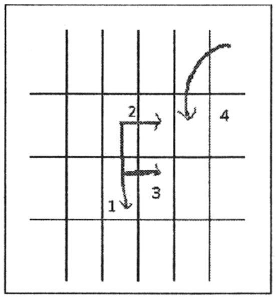
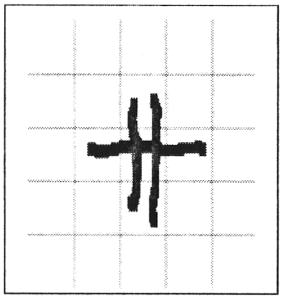


FIGURE-1, 4 x 6, DAS Password



Centered Password
Stroke : 3 Length : 9
FIGURE-2

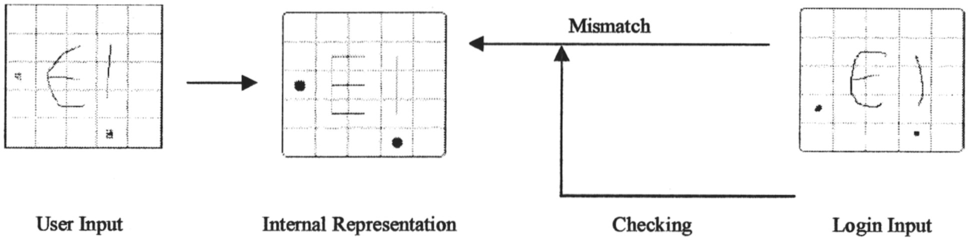
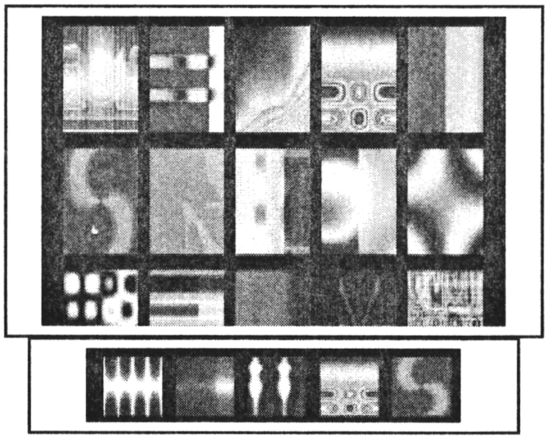
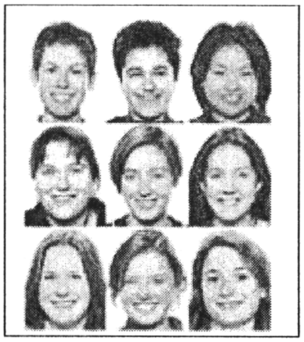


FIGURE-3. DAS Checking Scheme



Random Images and Challenge Set
FIGURE-4



PassFace, an example
(http://www.realuser.com)
FIGURE-5

security operations and developing secure systems [15]. Here we are considering authentication and security only.

For the Graphical Password scheme, space (processing and storage) is the prime factor. In DAS (Draw-A-Secret) scheme, Jermyn et al. [10] suggested that the size of the cell space for graphical passwords in the grid must be sufficiently large and this large space is not fixed.

Considering the use and popularity of Internet in current days, we propose a scheme, which is secured over network and effective in recognition. The entire work is divided into three sub-divisions:

a. Handwritten Password is taken, and then scaled it to (*N* x *M*) size; this size is fixed for any input. Thinning operation is performed after the scaling [16]. (Numerous sampling will be taken for each user and after digitization the corresponding two-dimensional arrays will be stored in the magnetic storage area).

b. Login Handwritten Password Input from the user will be taken during user Login. This input also will be Scaled and thinned, then the scaled and thinned image will be encoded and transmitted over network to the target. Where Scaled and thinned image will be extracted (decoding) [17, 18].

c. The extracted Handwritten Password now digitized to two-dimensional array and matched with previously stored samples [19].

A. *Pseudocode for the Sub-division-1*

Scaling:

```
int iWidth = input_width for scaling;
int iHeight = input_height for scaling;
double iRatio = (double)iWidth / (double)iHeight;
int imageWidth = input_image_width for scaling;
int imageHeight = input_image_height for scaling;
double imageRatio =(double)imageWidth /
                        (double) imageHeight;
if (iRatio < imageRatio)  Then
   iHeight = (int)(iWidth / imageRatio);
else
     iWidth = (int)(iHeight * imageRatio);
```

Thinning Algorithm Part-1:

```
Step 1   Take the surrounding pixels of
         foreground.
Step 2   Foreground points must have at least a
         single background neighbor.
Step 3   Reject points that with more than one
         foreground neighbor.
Step 4   Continue Steps [2 to 4] until locally
         disconnect (divided into 2 parts)
         region with Pixel iterate until
         convergence.
```

Thinning Algorithm Part-2:

```
Step 1   Open image file.
Step 2   Convert to PlanarImage.
Step 3   Two (2) Binary Kernels each of  (3 x 3)
         are taken.
Step 4   Using Gradient operation with the
         created PlanarImage in Step 2, (Create
         another PlanarImage from the step 2
         with the parameters gradientmagnitude,
         PlanarImage (Created in Step 2),
         Kernel Mask 1 and Kernel Mask 2.
         Thinning logic will work here stated in
         the Thinning Algorithm Part-1, before).
Step 5   Create the Rendering Image from ROI
         of the PlanarImage created in Step 4.
Step 6   Store this thinned rendered image into
         BufferedImage.
Step 7   Put BufferedImage into another image
         file. This is actually transformed image.
Step 8   Close file
```

B. *Data Hiding and Extraction Algorithms*

Resultant output will be stored in the storage area by multiple sampling from the users. Login Input now taken and processed by using Scaling and Thinning Algorithms and passed over to target where checking will be taken place. For passing over the Internet we will be using Data Hiding and Extraction Algorithms that are stated below:

```
Step 1    Open Handwritten Password Image file
          (iFile) in Read Mode
Step 2    Open an Output Image file (wFile) in
          Write Mode
Step 3    Initialize: integer i = 0; byte ch1, ch2,
          ch3 = 1
Step 4    While not end of iFile
Step 5    Loop
Step 6       Read byte from iFile and store to ch1
Step 7       Increase i by 1
Step 8       if (i > 54)  Then
Step 9           Take Bitwise complement of ch1
                 and store to ch2
Step 10          add ch3 to ch2
Step 11          write ch2 to wFile
Step 12      else
Step 13          write ch1 to wFile
Step 14   End Loop
Step 15   Close iFile, wFile
```

Algorithm stated below where almost all the Steps are equal to Data Hiding Algorithm stated above, but, remarkably the input file must be the resultant output image of Data Hiding Algorithm. Here, the algorithm(s) are used as the common key to the source and target locations.

```
Step 1    Open Transmitted Image file (iFile) in Read
          Mode
Step 2    Open an Output Image file (wFile) in Write
          Mode
Step 3    Initialize: integer i = 0; byte ch1, ch2, ch3 = 1
Step 4    While not end of iFile
Step 5    Loop
Step 6       Read byte from iFile and store to ch1
Step 7       Increase i by 1
Step 8       if (i > 54)  Then
Step 9           Take Bitwise complement of ch1 and
                 store to ch2
Step 10          add ch3 to ch2
Step 11          write ch2 to wFile
Step 12      else
Step 13          write ch1 to wFile
Step 14   End Loop
Step 15   Close iFile, wFile
```

C. *Pseudocode for Recognition (Matching / Authentication)*

Digitize all the samples of Handwritten Password images, which have been collected and processed by the Sub-division-1. Then Neural Network Learning Process will start. During the training process, the input to the neural network is the input matrix **M**, we defined as follows:

```
If  Image[i, j] =1 Then
    InputMatrix[i, j] =1
Else
    InputMatrix[i, j] = -1
```

The generated InputMatrix **M** is now used as input to the neural network. Typical, weights have to be adjusted during Network Training. In this method of learning, each candidate Image data taught to the network possesses a corresponding weight matrix. For the **k**th image data to be taught to the network, the weight matrix is denoted by $\mathbf{W_k}$. As learning of the image data progresses, it is this weight matrix that is updated. At the commencement of teaching (supervised training), this matrix is initialized to zero. Whenever an image data is to be taught to the network, an input pattern representing that image is submitted to the network. The network is then instructed to identify this pattern as, say, the **k**th character in a knowledge base of images. That means that the pattern is assigned a label **k**. In accordance with this, the weight matrix $\mathbf{W_k}$ is updated in the following technique (pseudo code):

```
for i =1 to x
loop
  for j =1 to y
  loop
    $W_k [i, j] = W_k [i, j] + M[i, j]$
  End loop
End loop
```

Product of corresponding elements of the weight matrix $\mathbf{W_k}$ of the **k**th learnt pattern and an input pattern **I** (Login Handwritten Password) as its candidate. Mathematical Expression is as follow:

$$\theta(k) = \sum_{i=1}^{x} \sum_{j=1}^{y} W_k(i, j) * I(i, j) \quad \text{---} \quad (1)$$

In the training process where **M** was the processed input matrix, in the recognition process, the binary image matrix **I** is directly fed to the system for recognition.

Ideal Weight Score **(μ)**, (which gives the sum total of all the positive elements of the weight matrix of a learnt pattern) can be computed as follow:

```
for i=1 to x
Loop
  for j=1 to y
  Loop
    if ($W_k(i, j) > 0$) then
      $\mu (k) = \mu (k) + W_k(i, j)$
  End loop
End loop
```

Recognition Quotient **(Q(k))** is a statistical value which gives a measure of how well the recognition system identifies an input pattern as a matching candidate for one of its many learnt patterns. It is simply given by:

$$Q(k) = \theta(k) / \boldsymbol{\mu}(k) \quad \text{---} \quad (2)$$

The greater the value of Quotient **(Q)**, the more confidence does the system on the input pattern as being similar to a pattern already known to it. The classification of input patterns now follows the following trivial procedure:

For an input candidate pattern **I**, calculate the recognition quotient **Q(k)** for each learnt pattern **k**. Determine the value of **k** for which **Q(k)** has the value **1**. Value less than **1** not is accepted or Login Input not matched.

IV. Result

Result and output shown in figure-6, here after execution of the Scaling and Thinning Algorithms.

Handwritten Password Image is passed through Data Hiding Algorithm before sending to the target site for security over the network and at target site Extraction Algorithm is used to extract the forwarded Image before Authentication and Recognition.

The process of digitization is important for the neural network used in the system. In this process, the input image is sampled into a binary window, which forms the input to the recognition system. Image is digitized into digital cells (for example, *6* x *8 = 48)*, each having a single color, either black or white. We assign a value +1 to each black pixel and 0 to each white pixel and create the binary image. This conversion is enough for neural networking. Digitization of an image into a binary matrix of specified dimensions makes the input image invariant of its actual dimensions. Hence an image of whatever size gets transformed into a binary matrix of fixed pre-determined dimensions. This establishes uniformity in the dimensions of the input and stored patterns as they move through the identification system.

Figure-7a to figure-7c shows the digitization of three input patterns representing D that are presented to the system for it to learn. Remarkably, Figure–7b is slightly different from 7a and 7c. This is the nature of Handwritten Password. Handwritten Password may vary from person to person as well as time to time.

V. Conclusion

We use the Recognition based technique in Graphical Password Scheme. Our work suggested that password is difficult to break. The entire system is checked and tested repeatedly with the desired output. Processing speed also improved in our scheme, because of digitization of data. Issue relating privacy and accuracy are tested. Technical aspects are analyzed and operations are validated and verified with numerous samples.

References

[1] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy and Nasir Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system", International J. of Human-Computer Studies (Special Issue on HCI Research in Privacy and Security) 63, 102-127, 2005.- Elsevier Ltd., http://www.sciencedirect.com

[2] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy and Nasir Memon, "Authentication Using Graphical Passwords : Effects of Tolerance and Image Choice", SOUPS'05 Conference, July 6–8, 2005, Pittsburgh, PA, USA.

[3] Julie Thorpe and P.C. van Oorschot, "Towards Secure Design Choices for Implementing Graphical Passwords", *IEEE* CS Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC'04).

[4] Wei-Chi Ku and Maw-Jinn Tsaur, "A Remote User Authentication Scheme Using Strong Graphical Passwords", Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05).

[5] Xiaoyuan Suo, Ying Zhu and G. Scott. Owen, "Graphical Passwords: A Survey", *IEEE* CS Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC-2005).

[6] Kanako Yokota and Tatsuhiro Yonekura, "A Proposal of COMPASS (COMmunity Portrait Authentication SyStem)", *IEEE* CS Proceedings of the 2005 International Conference on Cyberworlds (CW'05).

[7] Jean-Camille Birget, Dawei Hong and Nasir Memon, "Graphical Passwords Based on Robust Discretization", IEEE Transactions on Information Forensics and Security, Vol. 1, No. 3, September 2006.

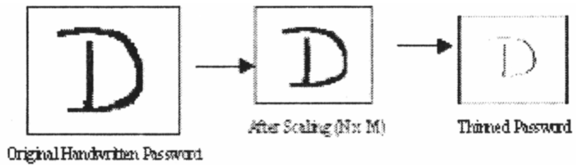


FIGURE-6

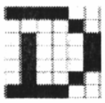


FIGURE-7a

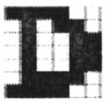


FIGURE-7b

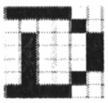


FIGURE-7c

[8] Fawaz A Alsulaiman and Abdulmotaleb El Saddik, "A Novel 3D Graphical Password Schema", VECIMS 2006 – *IEEE* International Conference on Virtual Environments, Human-Computer Interfaces, and Measurement Systems La Coruna - Spain, 10-12 July 2006.

[9] Fabian Monrose and Michael K. Reiter, "Graphical Passwords", August 5, 2005, Chapter-9, Pg. 161-179.

[10] I. Jermyn, A. Mayer, F. Monrose, M. Reiter, A. Rubin, "The design and analysis of graphical passwords", In 8th *USENIX* Security Symposium, 1999.

[11] R. Dhamija and A. Perrig, "Deja Vu: a user study using images for authentication", 9th *USENIX* Security Symposium, 2000.

[12] Nali and J. Thorpe, "Analysing user choice in graphical passwords", Tech. Report TR-04-01, School of Computer Science, Carleton University, Canada, 2004.

[13] J. Goldberg, J. Hagman, V. Sazawal, "Doodling our way for better authentication", CHI '02 extended abstracts on Human Factors in Computer Systems, 2002.-

[14] RealUser, www.realuser.com, last accessed in June, 2005.

[15] A. S. Patrick, A. C. Long, and S. Flinn, "HCI and Security Systems", presented at CHI, Extended Abstracts (Workshops). Ft. Lauderdale, Florida, USA., 2003.

[16] Samir Kumar Bandyopadhyay, Debnath Bhattacharyya and Poulami Das, "Handwritten Signature Verification System using Morphological Image Analysis", *IWAIT-2007* Conference Proceedings, Pg. 110-114, Bangkok, Thailand, 8-9 January, 2007.

[17] Samir Kumar Bandyopadhyay, Debnath Bhattacharyya and Anindya Jyoti Pal, "Secure Delivery of Handwritten Signature", *ACM* Ubiquity, Vol. 7 Issue. 40, October 16, 2006, http://www.acm.org/ubiquity/

[18] Samir Kumar Bandyopadhyay, Debnath Bhattacharyya and Poulami Das, "Handwritten Signature Extraction from Watermarked Images using Genetic Crossover", MUE-2007, *IEEE* CS Proceedings, Seoul, 26-28 April, 2007.

[19] Samir Kumar Bandyopadhyay, Debnath Bhattacharyya and Poulami Das, "A Flexible ANN System for Handwritten Signature Identification", Proceedings of the International MultiConference of Engineers and Computer Scientists 2007 Volume II, IMECS '07, March 21 - 23, 2007, Hong Kong, Lecture Notes in Engineering and Computer Science, pp. 1883-1887, Newswood Limited, 2007.

TABLE-1

Password Scheme	**Password Input**	**Recapitulation Power**	**Processing Speed**	**Authentication**
Text based	Fast	Depend on length and type of characters combination	Fast; Complexity, N	Low
Birget	Fast Input	Low; when large number of objects involved	Slow; Complexity depends on size and type of pictures. Can be given as N!/K!(N-K)! (N is the total number of picture objects; K is the number of pre-registered objects)	High
PassFace	Take Longer than Text based	Easier to remember, but, prediction	N^K (K is the number of rounds of authentication, N is the total number of pictures at each round)	High, but, chance of dictionary attack
Goldberg	Draw with stylus on touch sensitive screen; time taking	Depends on drawing complicacy	High Password Space	Guess dictionary attack
DAS	Depends on type of input; Draw with stylus on touch sensitive screen	Depends on drawing complicacy	Space consuming	Dictionary attack
User Authentication by Secured Graphical Password Implementation	Depends on size of password	Easy to remember	Minimum consumption due to digitization	Totally secured; Handwritten Characters are varied from person to person. Forgery Detection can be incorporated