A Free Drawing Graphical Password Scheme

Alice J. Lin and Fuhua (Frank) Cheng

University of Kentucky, ajlin0@cs.uky.edu, cheng@cs.uky.edu

ABSTRACT

This paper presents a method for freely drawing a graphical password. The new method achieves better security than conventional textual passwords and other graphical password schemes. With this method it is also easier for a user to remember the password. The basic idea of the new method is to use a number of the user's representative sample drawings to predict the user's future drawing prediction interval. The predicted values are obtained by conducting the least squares method to the polynomial regression model. Based on the predicted values and deviation of the user's sample drawings, a prediction interval for the signature/picture is generated. This prediction interval is used as the password and, subsequently, if the signature/picture drawn by a user lies within the prediction interval, the user is authenticated into the application.

Keywords:, graphical password, security, free drawing, signature, prediction.
DOI: 10.3722/cadaps.2009.xxx-yyy

1. INTRODUCTION

Authenticating users in network-based and Internet-based environments has been a challenge for network administrators and end users. The most popular computer authentication method is for a user to submit a user name and a textual password. The vulnerabilities of this method are well known. One of the main problems is the difficulty of remembering passwords. Studies show that users tend to pick short passwords or passwords that are easy to remember [1]. Unfortunately, these passwords can also be easily figured out or broken.

Despite their vulnerabilities, textual passwords are still the most commonly used authentication mechanism. Although organizations may adopt "strong" password policies [4] that encourage or require users to select passwords less susceptible to discovery, such policies typically increase the burden on the users' ability to remember those passwords [14]. Users tend to type such passwords (considered a non-dictionary word) about 40 percent slower than dictionary words [9], making the data entry process for such authentication more vulnerable to shoulder-surfing attacks. Alternative authentication solutions, such as token-based or biometric authentication, do not rely on the users' memory and introduce an increased level of security at the expense of increased hardware and software costs and usability, and are therefore not used as frequent means of user authentication [2], [5], [12]. Since a user's biometrics are fundamental parts of his/her identity, and may also be used for many other purposes, the risks from this information being stolen or captured are extremely high. Once compromised, biometrics cannot be changed. That is another reason that biometric information is not suitable for authentication.

Graphical password schemes have been proposed as a possible alternative to text-based schemes, motivated partially by the fact that humans can remember pictures better than texts; psychological studies support such assumption [7]. Pictures are generally easier to remember or recognize than texts. In addition, if the number of possible pictures is large enough, the possible password space of a graphical password scheme may exceed that of text-based schemes and thus presumably offer better resistance to dictionary attacks. Because of these advantages, there is a growing interest in graphical passwords. In addition to workstation and web log-in applications, graphical passwords have also been applied to ATM machines and mobile devices.

2. RELATED WORK

Among existing graphical password schemes, there are *recognition-based* and *recall-based* graphical techniques. Using recognition-based techniques, a user is presented with a set of images and the user passes the authentication by recognizing and identifying the images he or she selected during the registration stage. Using recall-based techniques, a user is asked to reproduce something that he or she created or selected earlier during the registration stage.

As shown in the studies by Davis et al. [3], for recognition-based techniques, users' choices of picture passwords are often predictable. Allowing users to use their own pictures would make the password even more predictable, especially if the attacker is familiar with the user.

For the recall-based category, Jermyn et al. [6] proposed a scheme, called "Draw-a-secret (DAS)". A user is asked to draw a simple picture on a 2D grid. During authentication, the user is asked to re-draw the picture. If the drawing touches the same grids in the same sequence, the user is authenticated. Thorpe and van Oorschot [10] analyzed the memorable password space of the DAS scheme. They introduced the concept of *graphical dictionaries* and studied the possibility of a brute-force attack using such dictionaries. They defined a length parameter for the DAS type graphical passwords and showed that DAS passwords of length 8 or larger on a 5 x 5 grid may be less susceptible to dictionary attack than textual passwords. They also showed that the space of mirror symmetric graphical passwords is significantly smaller than the full DAS password space. Since people recall symmetric images better than non-asymmetric images, it is expected that a significant fraction of users will choose mirror symmetric passwords. If so, then the security of the DAS scheme may be substantially lower than originally believed.

Van Oorschot and Thorpe [11] further quantitatively analyzed the size of these classes for DAS under convenient parameter choices and showed that the popular subspace is a surprisingly small proportion of the full password space.

Syukri et al. [8] proposed a system where authentication is conducted by having the user drawing his/her signature using a mouse. Their technique includes two stages, *registration* and *verification*. During the registration stage: the user will first be asked to draw their signature with a mouse, and the system will extract the signature area. The information will later be saved into the database. The verification stage first takes the user input and extracts parameters of the signature. After that, the system conducts verification using geometric average means and a dynamic update of the database. According to a survey performed by Zhu et al. [15], the signature recognition program is not reliable.

3. OUR APPROACH

The expressive movement in handwriting or drawing is made chiefly in a state of unawareness, automatically and impulsively. In his/her handwriting or artistic expression a person not only communicates his/her conscious thought but also his/her underlying thought, by which graphic movement becomes a "diagram of the unconscious". Graphic movements reveal certain consistencies which cannot be explained by chance, nor by learning, nor by imitation of a set pattern [13]. These unconscious movements are unique for everyone.

However, one should also understand that human actions can not be precisely repeated, especially unconscious actions. If one asks a person to draw a signature/picture several times, the results will be close but can never reach the "exactly the same" level. Actually the degree of closeness varies from time to time, depending not only on the person's mental and physical conditions, but also on the external environment. Therefore, using only one sample drawing of a user to predict the user's future drawing pattern obviously is not a reliable approach. One needs to use a set of sample drawings to do the prediction. The best scenario is to collect as many variations of the user's drawing as possible so that future drawings of the user can all be covered by this set of sample drawings. This certainly is not possible. An alternative is to collect enough representative sample drawings so that future drawings can all be covered by combinations of those representative sample drawings.

In this paper, to build a set of representative sample drawings, a user is asked to draw a signature/picture certain times first. This set of sample drawings is then used to create combinations of the collected sample drawings in the form of a banded signature/picture (see Figure 6), called a *prediction interval*. This prediction interval is set as the password for the application. When a user needs to access this application, the user is prompted to draw his/her signature/picture. If the signature/picture lies inside the prediction interval, the drawing is accepted. Details of our approach are described below.

Assume the user draws six times to set up a password (Fig. 1, Fig 2). First we normalize the drawings so that they are all of the same size. Whatever input device the user uses: stylus, touchpad, touch screen or mouse, pixels of the drawings are then converted into point coordinates in the order of user's input. Each drawing is then segmented at significant points. Segments at the same location are considered as the same *part*. For each part of the curve we assume there are n points (Fig. 3 shows three parts). $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$.

Fig. 1: Six times of free draw using mouse.

Fig. 2: Six times of free draw using stylus.

3.1 Polynomial regression model

We use a polynomial regression model to predict the user's future drawing values in a part of the curve if user drawing does not contain too much fluctuation.

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_p x_i^p + \varepsilon_i \tag{3.1}$$

$i = 1,2,\ldots,n.$ β_j, $j = 0,1,2,\ldots,p.$ are unknown parameter. ε_i is the error. We define the best estimate of β_j as the one which minimizes the sum of squared errors. The method of *least squares* is used to choose the values for polynomial parameters that minimize the sum of the squares of the errors ε_i.

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_i^j + \varepsilon_i \tag{3.2}$$

The least squares function is

$$L = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_i^j)^2 \tag{3.3}$$

The function L is minimized with respect to $\beta_0, \beta_1, \ldots, \beta_p$. So the predicted parameter $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$, must satisfy

$$\frac{\partial L}{\partial \beta_0}\Big|_{\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p} = -2\sum_{i=1}^{n} (y_i - \hat{\beta}_0 - \sum_{j=1}^{p} \hat{\beta}_j x_i^j) = 0 \tag{3.4}$$

$$\text{and } \frac{\partial L}{\partial \beta_j}|_{\hat{\beta}_0,\hat{\beta}_1,\ldots,\hat{\beta}_p} = -2\sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \sum_{j=1}^{p}\hat{\beta}_j x_i^{\,j}) = 0 \tag{3.5}$$

$j = 1,2,\ldots,p.$ The predicted polynomial model is

$$\hat{y}_i = \hat{\beta}_0 + \sum_{j=1}^{p}\hat{\beta}_j x_i^{\,j} \tag{3.6}$$

In Fig.3, the red curves are the predicted values and blue curves are 95% prediction interval for three segments of the user's drawing in Fig. 2.

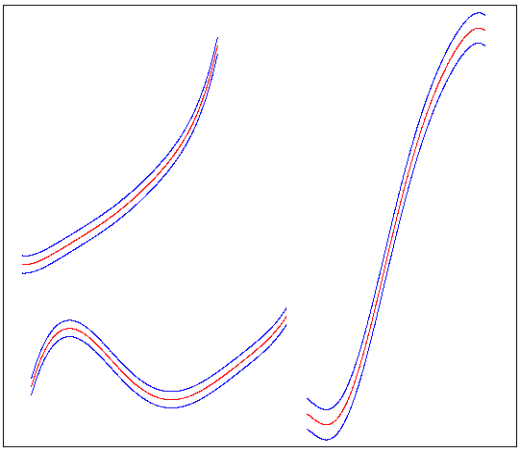

Fig. 3: Predicted values and prediction intervals generated by polynomial regression model.

3.2 The B-spline regression model

We use B-spline regression model to predict the user's future drawing values if the user's drawing contains lots of fluctuation. B-spline basis functions combine the Polynomial smoothness and broken stick local influence.

For the illustration, we decompose drawing to several smaller parts, and show the differences of predicted values and intervals generated from Fig. 2 by polynomial regression model (see Fig. 3) and B-spline regression model (see Fig. 4).

Given n+1 control points, $P_0, P_1, \ldots, P_n$. u_i is a knot. The total sequence is a knot vector and $u_i \le u_{i+1}$ The B-spline curve of degree d is defined by these control points and knot vector is

$C(u) = \sum_{i=0}^{n} N_{i,d}(u)P_i$, where $\sum_{i=0}^{n} N_{i,d}(u)$ are the B-spline basis functions of degree d.

The model to predict the user's future drawing values is $Y_k = \sum_{j=0}^{n} N_{j,d}(u_k)P_j + \varepsilon_k$

Y_k is the sample data. $k = 0,1,2,\ldots m$. The specified set of control points P_j are unknown parameter. ε_k is the error. We use the same method as 3.1 to best estimate of P_j.

$$L = \sum_{k=0}^{m}\varepsilon_k^2 = \sum_{k=0}^{m}(Y_k - \sum_{j=0}^{n} N_{j,d}(u_k)P_j)^2 \tag{3.7}$$

$$\frac{\partial L}{\partial P_i} = -2\sum_{k=0}^{m}(\hat{Y}_k - \sum_{j=0}^{n} N_{j,d}(u_k)\hat{P}_j)N_{i,d}(u_k) = 0 \ , \tag{3.8}$$

$$\sum_{k=0}^{m}\sum_{j=0}^{n} N_{i,d}(u_k)N_{j,d}(u_k)\hat{P}_j \ - \sum_{k=0}^{m} N_{i,d}(u_k)\ \hat{Y}_k = A^T A\hat{P} - A^T\hat{Y} = 0 \tag{3.9}$$

where A is a matrix with n+1 rows and m+1 columns. $\hat{P} = (\ \hat{P}_0, \hat{P}_1, \ldots, \hat{P}_n)^T$, $\hat{Y} = (\ \hat{Y}_0, \hat{Y}_1, \ldots, \hat{Y}_m)^T$

$\hat{P} = (A^T A)^{-1} A^T\ \hat{Y}$. The predicted B-spline model is $\hat{Y}_k = \sum_{j=0}^{n} N_{j,d}(u_k)\ \hat{P}_j$

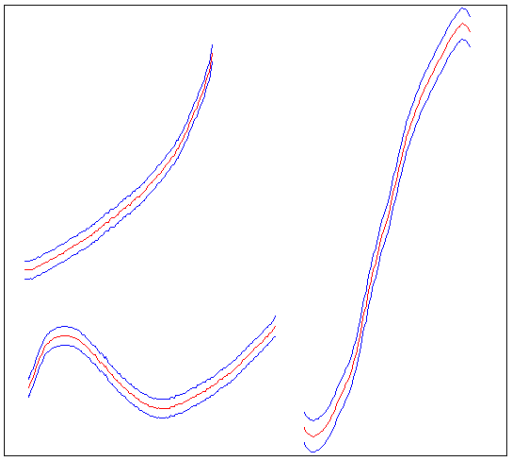

Fig. 4: Predicted values and prediction intervals generated by B-spline regression model

3.3 Prediction interval

The probability density function of t distribution (Fig. 5) is

$$f(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\,\Gamma(\frac{\nu}{2})}\left(1+\frac{t^2}{\nu}\right)^{-(\frac{\nu+1}{2})} \tag{3.10}$$

ν is the number of *degrees of freedom* and Γ is the Gamma function. $-\infty < t < \infty$. The area under curve $f(t)$ is associated with a given interval $[\ -t^*,\ t^*\ ]$ which represents probability. Probability value is

$P = \int_{-t}^{t} f(t)d(t)$, $\int_{-\infty}^{\infty} f(t)d(t) = 1$

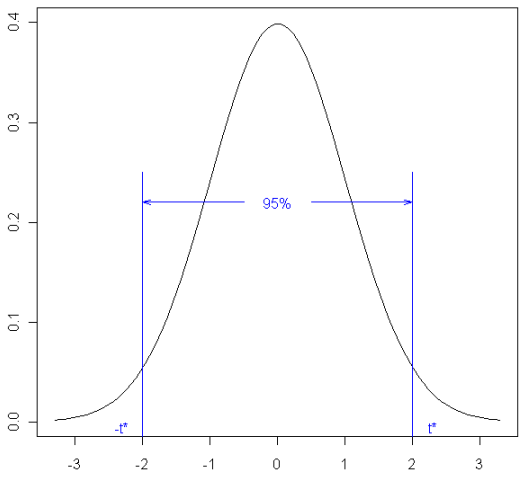

Fig. 5: The probability density function.

The model equation $y = X\beta + \varepsilon$, where

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix},\ X = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix}\ \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix},\ \text{and}\ \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \tag{3.11}$$

The predicted regression model is $\hat{y} = X\hat{\beta}$

We need to assess the uncertainty in prediction. Decision makers need more than just a point estimate to make rational choices. Given a set of particular values of the variables, $x_{i1}, x_{i2}, \ldots, x_{ik}$, $\mathbf{x}_i^T = (1, x_{i1}, x_{i2}, \ldots, x_{ik})$ the predicted value is $\hat{y}_i = \mathbf{x}_i^T\hat{\beta}$. But in assessing the variance of prediction, we must include the variance of ε. We have $\mathrm{var}(\mathbf{x}_i^T\hat{\beta}) = \mathbf{x}_i^T(X^TX)^{-1}\mathbf{x}_i\sigma^2$, σ^2 is variance. A future predicted value is predicted to be $\mathbf{x}_i^T\hat{\beta} + \varepsilon$, but we do not know what the future ε will be. So a confidence level C (such as C=95%) prediction interval for this future predicted value is $\hat{y}_i \pm t^* SE_{\hat{y}_i}$.

t^* is the value for the t density curve with area probability value (such as 95%) between $-t^*$ and t^*. The standard error $SE_{\hat{y}_i}$ is for predicting an individual $\hat{y}$ and is basic measure of the variability of the user input y about the predicted $\hat{y}$. The quantity of $SE_{\hat{y}_i}$ is the estimated standard deviation of $\hat{y} - y$.

$SE_{\hat{y}_i} = \sqrt{(1 + \mathbf{x}_i^T(X^TX)^{-1}\mathbf{x}_i)\hat{\sigma}^2}$, $\hat{\sigma}^2$ is an unbiased estimate of σ^2.

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n - p}, \tag{3.12}$$

The numerator is the error and the denominator is degrees of freedom. $p = k + 1$. The margin of error $t^* SE_{\hat{y}_i}$ decreases as the confidence level C and the standard deviation $\hat{\sigma}$ decreases, and the sample size n increases.

The values of $\hat{\sigma}^2$ will be large if the input y_i are widely spread around their predicted values, and $\hat{\sigma}^2$ s will be small if the y_i are all close to the predicted values.

Prediction interval for future user drawing is:

$$\hat{y}_i - t^* \sqrt{(1 + \mathbf{x}_i^T (X^T X)^{-1} \mathbf{x}_i) \hat{\sigma}^2} \leq y \leq \hat{y}_i + t^* \sqrt{(1 + \mathbf{x}_i^T (X^T X)^{-1} \mathbf{x}_i) \hat{\sigma}^2} \tag{3.13}$$

In Figure 6 the green curves define a 95% prediction interval; the blue lines define a 99% prediction interval. We are 95% confident that in the future if the user tries to draw the same picture, each point will lie within the region bounded by the green curves.

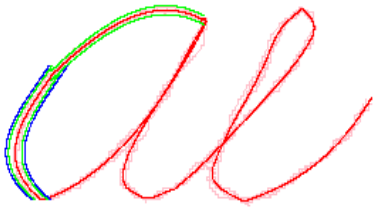

Fig. 6: The green curves define a 95% prediction interval and the blue lines define a 99% prediction interval. The red curves are the predicted values. The pink lines in the background are user's drawing.

3.4 Success rate

If the sample size n is large, the standard error will be small (see Eqn. 3.14). Based on our trial, the more times a user draws the same picture, the more accurate the prediction calculation. Tab. 1 shows the success rate for users using different pictures as password when he/she tried to login. The success rate is the successful times divided by total trial times. Number of draws is that a user draws a certain times of a picture to set a password. We can see that success rate goes up with the number of draws increasing.

Tab. 2 shows the successful login rate that a user uses different pictures or letters as password. Using the pictures created by user or letters that he/she usually writes has higher successful rate while copying pictures/letters created by others has much lower successful rate. Copying pictures/letters created by others as user's password is not a good idea. Since it is not out of your nature, the copied pictures/letters are easily to be forgotten. This may be the reason for low successful rate in login. This result may support that one's signature is best candidate used for password.

Pictures / Number Of draws					
2	0.75	0.63	0.65	0.70	0.65
4	0.80	0.78	0.77	0.83	0.76

6	0.92	0.88	0.89	0.93	0.87
8	0.97	0.92	0.95	0.93	0.90
10	0.99	0.96	0.97	0.99	0.95

Tab. 1: Number of draws in password setting vs. successful login rate for different pictures

Number of draw	Created a picture	Copy a picture	Created letters	Copy letters
5	0.87	0.75	0.90	0.76
8	0.96	0.80	0.97	0.70
11	0.97	0.87	0.99	0.88

Tab. 2: Successful login rate using different pictures/letters as password

4. CONCLUSION

A new graphical password scheme is presented in this paper. In addition to the intrinsic advantage of being less vulnerable to brute force and dictionary attacks, the new method also has the following advantages. First, the new method gives a user the freedom of drawing anything he/she wants to draw and the ease in remembering his/her password. Consequently, the new method is especially efficient for input devices such as stylus or touch screen. Second, the new method achieves better security than conventional textual password and other graphical password schemes. This follows from the fact that the new method has an infinite password space, therefore the password is more secure. Besides, since the signature/picture is difficult to reproduce by others, it is relative secure to shoulder-surfing attacks. Even if a person knows what you drew, it will have very little chance for him/her to draw the same thing as you did. Since the signatures/pictures are usually not symmetric, the password space will not be compromised by the symmetric issue [11].

Future research directions include studying the best size and better sampling techniques for the sample drawing set and better techniques in generating the prediction interval.

5. REFERENCES

[1] Adams, A.; Sasse, M. A.: Users are not the enemy: why users compromise computer security mechanisms and how to take remedial measures, Communications of the ACM, vol. 42, 1999, 41-46.

[2] Anderson, R. J.: Why Cryptosystems Fail, Communications of the ACM, vol. 37, 1994, 32-40.

[3] Davis, D.; Monrose, F.; Reiter, M. K.: On user choice in graphical password schemes, in Proceedings of the 13th Usenix Security Symposium, San Diego, CA, 2004.

[4] Department of Defense Computer Security Center: Department of Defense Password Management Guideline, Department of Defense, Washington, DC, CSC-STD-002-85, April 12, 1985.

[5] Doyle, P.; Hanna, S.: Analysis of June 2003 Survey on Obstacles to PKI Deployment and Usage, Organization for the Advancement of Structured Information Standards, Billerica, MA, August 8, 2003.

[6] Jermyn, I.; Mayer, A.; Monrose, F.; Reiter, M. K.; Rubin, A. D.: The Design and Analysis of Graphical Passwords, in Proceedings of the 8th USENIX Security Symposium, 1999.

[7] Shepard, R. N.: Recognition memory for words, sentences, and pictures, Journal of Verbal Learning and Verbal Behavior, vol. 6, 1967, 156-163.

[8] Syukri, A. F.; Okamoto, E.; Mambo, M. A.: User Identification System Using Signature Written with Mouse, in Third Australasian Conference on Information Security and Privacy (ACISP): Springer-Verlag Lecture Notes in Computer Science (1438), 1998, 403-441.

[9] Thomas, R. C.; Karahasanovic, A.; Kennedy, G. E.: An Investigation into Keystroke Latency Metrics as an Indicator of Programming Performance, presented at Australasian Computing Education Conference 2005, Newcastle, Australia, 2005.
[10] Thorpe, J.; Oorschot, P. C. V.: Graphical Dictionaries and the Memorable Space of Graphical Passwords, in Proceedings of the 13th USENIX Security Symposium, San Deigo, USA: USENIX, 2004.
[11] Van Oorschot, P. C.; Thorpe, J.: On Predictive Models and User-Drawn Graphical Passwords, ACM Transactions on Information and System Security, vol. 10 (4), Article 17, 2008, 1-33
[12] Whitten, A.; Tygar, J. D.: Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0, presented at 8th Usenix Security Symposium, Washington, DC, 1999.
[13] Wolff, W.: Diagrams of the unconscious; handwriting and personality in measurement, experiment and analysis New York, Grune & Stratton, 1948.
[14] Yan, J.; Blackwell, A.; Anderson, R.; Grant, A.: Password Memorability and Security: Empirical Results, IEEE Privacy & Security, vol. 2, 2004, 25-31.
[15] Zhu, X.; Suo, Y.; Owen, G. S.: Graphical Passwords: A Survey In proceedings of ACSAC, IEEE Computer Society, 2005, 463-472.