October 2013

GRAPHICAL AUTHENTICATION USING REGION BASED GRAPHICAL PASSWORD

G. NIRANJANA

Dept of computer science and engg, SRM University. Chennai, India, niranjana.g@ktr.srmuniv.ac.in

KUNAL DAWN

Dept of computer science and engg, SRM University. Chennai, India, kunal.dawn@gmail.com

Follow this and additional works at: <https://www.interscience.in/ijcsi>

Part of the [Computer Engineering Commons](#), [Information Security Commons](#), and the [Systems and Communications Commons](#)

Recommended Citation

NIRANJANA, G. and DAWN, KUNAL (2013) "GRAPHICAL AUTHENTICATION USING REGION BASED GRAPHICAL PASSWORD," *International Journal of Computer Science and Informatics*: Vol. 3 : Iss. 2 , Article 5.

DOI: 10.47893/IJCSI.2013.1125

Available at: <https://www.interscience.in/ijcsi/vol3/iss2/5>

This Article is brought to you for free and open access by the Interscience Journals at Interscience Research Network. It has been accepted for inclusion in International Journal of Computer Science and Informatics by an authorized editor of Interscience Research Network. For more information, please contact sritampatnaik@gmail.com.

GRAPHICAL AUTHENTICATION USING REGION BASED GRAPHICAL PASSWORD

G. NIRANJANA¹ & KUNAL DAWN²

^{1,2}Dept of computer science and engg, SRM University, Chennai, India
E-mail : niranjana.g@ktr.srmuniv.ac.in & kunal.dawn@gmail.com

Abstract - Password authentication is failing as an authentication since it increases the user burden to remember the passwords. Graphical authentication is proposed as a alternative for textual passwords since it may be easy for users to remember. In this paper we propose a new image region selection based graphical password scheme. We are going to present a new technique for authentication which is based on the tracking of mouse motions on an image called mouse gestures for selecting regions in the image. In general, a gesture is a sequence of interactions with the application, which represents one of the Specified symbols. A mouse gesture is a continuous, directed sequence of the mouse cursor movements with the clearly distinguished start and end. A set of gestures may be stored in a database called gesture classes for each user. Users are allowed to select a set of random images and a gesture for each image. Some tolerance level is also given for each gesture. When logging in if the user draws the correct gesture using mouse the user will be treated as an authenticated user. Mouse gestures are captured through bounding box and corner detection algorithms. This method provides more security than cued click points where the user is allowed to click on a particular point called pass point for authentication which is more vulnerable to hackers.

Keywords-mouse gesture, gesture class, cued click points, authentication.

I. INTRODUCTION

Passwords are expected to comply with two fundamentally conflicting requirements:

1. Passwords should be easy to remember,
2. Passwords should be secure

Satisfying these requirements is virtually impossible for users[5]. Various graphical password schemes have been proposed as alternatives to text-based passwords. Research and experience have shown that text-based passwords are fraught with both usability and security problems that make them less than desirable solutions[5]. Psychology studies have revealed that the human brain is better at recognizing and recalling images than text[3]; graphical passwords are intended to capitalize on this human characteristic in hopes that by reducing the memory burden on users, coupled with a larger full password space offered by images, more secure passwords can be produced and users will not resort to unsafe practices in order to cope.

Initially Cued Click Points (CCP) was used for authentication where the users are allowed to select a set of images and in each image one particular point is selected. This particular point is called the passpoint through which the user is authenticated [1]. A password consists of one click-point per image for a sequence of images. The next image displayed is based on the previous click point. Graphical authentication using CCP method has the drawback that the user has to remember the particular pass point in the given set of images and it should have high tolerance level. In this paper, we propose a new image region selection based graphical password scheme. It can be viewed as a combination of Pass Points and Cued Click Point (CCP). A password

consists of one region selection per image for a sequence of images. The next image displayed is based on the previous selected region so users receive immediate implicit feedback as to whether they are on the correct path when logging in. This scheme offers both improved usability and security. In the proposed method the user can just remember the region and its shape. No burden to human brains and also a security level may also be given for it.

The rest of the paper is organized in such a way that section 2 contains introduction to mouse gesture. Section 3 carries the region detection algorithm which includes registration process of an user using bound box algorithm for capturing the gesture, Authentication procedure is detailed with corner detection method and region pixel count method to calculate the virtual grid pixels in the region. Section 4 is explained with experimental results and analysis. Conclusion and future work are described in section 6.

II. MOUSE GESTURE

A mouse gesture is a continuous, directed sequence of the mouse cursor movements with the clearly distinguished start and end points. In our work, gestures are marked by pressing the right button. The usability of the assumed notion of gesture was assessed during experiments described in Hofman [2]. For high usability of gestures-based interface, three basic features must be preserved: accuracy, efficiency (of recognition), and adaptability to the possibilities and needs of the individual user[4]. Accuracy is understood as the percentage of properly recognised gestures in relation to the intention of the user performing them. Efficiency should be enough

for the use on an average computer. Adaptability means easy registration of the own classes of gestures of the given user. Some of the simple gestures are

III. REGION DETECTION ALGORITHM:

To detect the user selected region in a very efficient manner we have used three methods that represent the region in a numerical way so that it can be stored in the database in an efficient way and also reduce the network transmission of data.

A. Bounding box method

The first method we will discuss is the bounding box method. This involves drawing a box around the gesture and dividing it up into a grid. The gesture is then defined by the areas that it passes through. The grid would be set up as follows.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

The gesture on the right would then be parsed as 13,9,5,1,2,3,4,8,12,16 if you had drawn it from left to right. There are many advantages to this method. The first is that it's very easy to code. The design is simple and creating and defining gestures is very easy. It would also be pretty accurate when actually interpreting the gestures. It would deal with erratic mouse behavior well because human errors are of no consequence as long as it stays in the area. For this same reason, it handles curved gestures as well as straight ones.

In this method we determine the rectangle that bounds the user selected region. A virtual grid is presented on top of the image. When user draws the region on top of an interested image object the pixels are plotted on top of the grid pixels. The grid pixels are set to value of 1 where the user gave drawn the

region and elsewhere is set to 0. The bounding box can be detected easily by calculating the maxima and minima of the user inputted pixels in the grid.

$$g(x, y) = \begin{cases} 1 & \text{if its a gesture point} \\ 0 & \text{else where} \end{cases}$$

Registration algorithm

```

registration(user_id)
sequence_number:=1;
While sequence_number is less than 4 do
    ➤ Generate a random number between 1 to 203
      (total number of images in the database), let it be
      the image_number;
    ➤ Retrieve the random image with image_number
      from the data base and show it to the user;
    ➤ Draw a virtual grid over the image;
    ➤ Wait for the user to select the region;
    ➤ Calculate the parameters , , ,
      ;
    ➤ Store the parameters with sequence_number
      image_number and user_id in the database;
    ➤ Sequence_number:=sequence_number+1;

```

B. Corner detection method

Another method is corner detection. At first this may seem similar to the change in direction method, but is actually quite different. This involves figuring out which points are the corners, and then looking at the relationship between those corners. The advantage of this is that it would be very accurate, provided the algorithm detects the corners properly. This method also takes into account the proportions of each part of the gesture.

In this method we determine the top left and bottom right corners of the bounding box. We use this method because it uniquely represents every user selected regions in that image, that is, no two different regions on the image will have the same Corner value. This can be done easily by the following set of equations.

(1)

(2)

(3)

(4)

The algorithm is defined as follows

Corner detection algorithm

```

login(user_id)
set sequence_number:=1;
set login_stat:=1;
While sequence_number is less than 4 do
    If login_stat=1 then do
        ➤ Fetch the parameters from the database with
          current sequence_number and user_id;
        ➤ Retrieve the image from the database with
          fetched image_number and show it to the user;
        ➤ Draw a virtual grid over the image.
        ➤ Wait for the user to select the region;
        ➤ Calculate the new parameters , , ,
          ;

```

```

➤ Calculate the difference ;
➤ If the calculated differences are within CT and
  GPT range then
a. sequence_number:=sequence_number+1;
b. login_status:=1;
else
a. sequence_number:=sequence_number+1;
b. login_stat:=0;
else do
➤ Generate a random number between 1 to 203
  (total number of images in the database), let
  image_number;
➤ Retrieve the image with image_number from the
  data base and show it to the user;
➤ Draw a virtual grid on top of the image;
➤ Wait for the user to select the region;
➤ sequence_number:=sequence_number+1;
End
End
if login_stat=1 then do
successful login;
else
login fail;
End
The differences are calculated
using the following equations.

```

$$d_{tx} = |n_{topx} - o_{topx}| \quad (5)$$

$$d_{ty} = |n_{topty} - o_{topty}| \quad (6)$$

$$d_{bx} = |n_{bottomx} - o_{bottomx}| \quad (7)$$

$$d_{by} = |n_{bottomy} - o_{bottomy}| \quad (8)$$

$$d_{gp} = |n_{gesturepixel} - o_{gesturepixels}| \quad (9)$$

Here are the displacement values and is the pixel difference value that are calculated from the new and old saved parameters as described above.

B. Region Pixel Count Method:

In this method we determine how many virtual grid pixels the user had used to select the region. This method is required because two different user may select the same region in the same image but there selection cannot be the same hence they defer on the number of the virtual pixels and also it distinguish the same region selection with different shape because different shape will have different pixel count

(10)

Where is defined as

These three methods together creates the required parameters that can represent the data as a set of numeric values which are very much convent to store

and retrieve and also encryption on this data can be done while transferring over network.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

At the time of registration the user selects a specific region on the given image of his interest which he can remember easily. A set of random and unique images are fetched from the database for registration. For the set of images given to the user, user selects regions in every image and remembers its position and size in that sequence and this information is stored in the database for future login purpose. At the time of login the sequence of images given to the user one by one which was saved at the time of the registration. Now the user selects the region with the gesture that was used at the time of registration. For every image, our algorithm calculates the parameter's and matches with the previously stored parameter's for that image with some tolerance value. If the match is a success then next image is fetched from the database and the process is repeated and if the match is unsuccessful then the user is not notified until the end and at the next sequence a random image is given to the user and login fail flag is activated. This approach increases the total search space of the attacker and also smart users can select complex regions.

In our work, we have used 203 images for selecting random images. The parameters are calculated using the equations (1) – (11). Some tests are carried out based on the following parameters.

Number Of Images	203
Size Of Image	500x312
Size Of Grid	125x78
Size Of Grid Cells	4x4
Number Of Images for Authentication	4
Total Attempts per test	20

Users are allowed to register to the system by creating their own passwords. Then we record the number of Successful Login per 20 attempts to login correctly. It is also possible that the user may fail to login although he or she has entered correct data. This occurs because of the tolerance value we have used for the test. Here we vary the CT and GPT parameters and check how they affect the Success per Correct Attempt.

Figure 1. Registration process

Figure 1. shows sample outputs (a) original images (b) user selected gestures (c) bounding box for gestures.

Table 1 shows the actual gesture calculated values of x_1, y_1, x_2, y_2 using the equations (1) – (4).

Table 2 shows the calculated values of x_1, y_1, x_2, y_2 along with the tolerance when the user logs in.

Table 3 shows the various test cases results by calculating difference between the stored data and login data using the equations (5) – (10). Table 4 shows the results of various tolerance level by calculating coordinate tolerance and grid pixel tolerance using the equations (11) and (12).

Fig 4 shows the graph of the success rate for the proposed method. Users are authenticated by allowing them to select their own gestures so that they can easily remember it. Also hackers are diverted in such a way that they could not identify where they went wrong.

Table 1. Actual data stored for user

Img Id	Seq No	Top X	Top Y	Bottom X	Bottom Y	Grid Pixel Count
84	1	59	3	73	30	71
165	2	85	15	115	59	135
169	3	12	46	79	78	151

Table 2. Data collected at the time of login

Image Id	Sequence	Top X	Top Y	Bottom X	Bottom Y	Grid Pixel Count
84	1	59	6	73	31	74
165	2	86	18	118	60	121
169	3	11	43	81	73	179

Result : Successful Login With

Coordinate Tolerance (C.T): 10

Grid Pixel Tolerance (G.P.T): 30

Table 3. Test Cases:

Calculated Difference among the Stored Data & Login Data

Sequence Number	Top X Difference	Top Y Difference	Bottom X Difference	Bottom Y Difference	Grid Pixel Count Difference	Result
Login Attempt 1 / Authenticated / Tolerance [C.T=10] , [G.P.T=50]						
1	0	3	0	1	3	Pass
2	1	3	3	1	14	Pass
3	1	3	3	1	28	Pass
Login Attempt 2 / Login Fail / Tolerance [C.T=5] , [G.P.T=50]						
1	2	1	2	3	4	Pass
2	6	2	1	0	12	Fail
3	-	-	-	-	-	Fail
Login Attempt 3 / Login Fail / Tolerance [C.T=10] , [G.P.T=10]						
1	2	1	2	3	4	Pass
2	6	2	1	0	12	Fail
3	-	-	-	-	-	Fail
Login Attempt 4 / Success / Tolerance [C.T=8] , [G.P.T=20]						
1	2	1	2	3	4	pass
2	6	2	1	0	12	pass
3	1	7	3	8	18	pass

Table 4. Test Cases result on different tolerance level

No	C.T	G.P.T	Success/Correct Data	Total Attempt	Success Rate
1	2	2	2	20	10%
2	2	4	1	20	5%
3	2	8	2	20	10%
4	2	12	3	20	15%
5	2	16	5	20	25%
6	4	2	9	20	45%
7	4	4	11	20	55%
8	4	8	10	20	50%
9	4	12	13	20	65%
10	4	16	13	20	65%
11	8	2	1	20	10%
12	8	4	3	20	15%
13	8	8	17	20	85%
14	8	12	17	20	85%
15	8	16	18	20	90%
16	12	2	3	20	15%
17	12	4	7	20	35%
18	12	8	15	20	75%
19	12	12	19	20	95%
20	12	16	20	20	100%
21	16	2	4	20	20%
22	16	4	6	20	30%
23	16	8	15	20	75%
24	16	12	18	20	90%
25	16	16	20	20	100%

Our study shows that CT and GPT parameters have greater effect on the security of the system. The low value of the CT and GPT means it will accept more accurate login data only where if we increase the tolerance value then the system will allow more user errors. In case of low value of CT and GPT the user require more attempts to login but it also increases the security to great extent where with high value of CT and GPT the user may login at his first attempt with near correct data only which also decreases the security of the system.

V. CONCLUSION AND FUTURE WORK

The data in the table depends on the parameters we have selected and the capability of the users while we testing the system. Different implementation with different parameters may produce different data. To generalize this approach we define a new parameter called Security level which is based on different requirements of the secure system. A system with low security level allow users to login to their system with maximum error in the login attempt hence it makes easy for the user to login but it also decreases the search space of the attacker per image. Where a system with high security level allow users to login to their system with near accurate data in login attempt hence it makes more difficult for the user to login but also increases the security to maximum level.

The security level parameters have maximum value of 5 which indicated maximum security of the system and minimum value of 1 which indicate minimum security of the system. Some examples of the different systems with different security levels are:

1. Security Level 1 – System with very less security that only requires user authentication but not great security. Examples of this kind of systems may be simple Forum and Blog sites.
2. Security Level 2 – System with less security. Examples of this kind of systems may be Individuals sites with very less valuable contents.
3. Security Level 3 – System with medium security. Examples of this kind of systems may be Social Networking sites.
4. Security Level 4 – System with high security. Examples of this kind of systems may be Private Business sites or corporate login systems.
5. Security Level 5 – System with very high security. Examples of this kind of systems may be Online Banking Transaction gateway sites.

Depending upon this 5 Security levels we categorize our CT and GPT values. With CT and GPT values of 0 means that no tolerance is accepted and the user is required to enter the exact data that was used at the time of registration which is near impossible to remember. Our experiments shows that with CT, GPT value greater than 2 the users are able to login to the system and CT, GPT value greater than 16 gives very less security because it accepts login

data with great error hence every approximate attempt is a successful login.

Our experiments proved that our method provides a good way of authentication using gestures in graphical passwords. Also our authentication method decreases the rate of probability of getting hacked as the options space for authentication is more. Also the hacker is not informed about the error at the instant thereby avoiding unnecessary attempts. Our future work includes capturing the color and texture information of the gesture and using it for authentication.

REFERENCES

- [1] Sonia Chiasson, P.C. van Oorschot¹, and Robert Biddle "Graphical Password Authentication Using Cued Click Points"
- [2] Hofman, P.: Selected Issues of Artificial Intelligence in the Construction of User Interface to a CASE System. MSc Thesis, Wroclaw University of Technology, (2005).
- [3] Nelson, D.L., U.S. Reed, and J.R. Walling. Picture Superiority Effect. *Journal of Experimental Psychology: Human Learning and Memory* 3, 485-497, 1977
- [4] Paweł HOFMAN¹ Maciej PIASECKI¹ "Efficient Recognition of Mouse-based Gestures "
- [5] Blonder, G.E. "Graphical Passwords". United States Patent 5,559,961, 1996.
- [6] Chiasson, S., R. Biddle, R., and P.C. van Oorschot." A Second Look at the Usability of Click-based Graphical Passwords". *ACM SOUPS*, 2007.
- [7] Cranor, L.F., S. Garfinkel. "Security and Usability". O'Reilly Media, 2005.
- [8] Davis, D., F. Monrose, and M.K. Reiter. "On User Choice in Graphical Password Schemes". 13th USENIX Security Symposium, 2004.
- [9] Dirik, A.E., N. Menon, and J.C Birget." Modeling user choice in the PassPoints graphical password scheme". *ACM SOUPS*, 2007. Article in a conference proceedings:
- [10] H.Goto, Y. Hasegawa, and M. Tanaka, "Efficient Scheduling Focusing on the Duality of MPL Representatives," *Proc. IEEE Symp. Computational Intelligence in Scheduling (SCIS 07)*, IEEE Press, Dec. 2007, pp. 57-64, doi:10.1109/SCIS.2007.357670.