

An Approach to Implement Secured CAPTCHA Code Based on Personal Information and Likings of User

Ramkrishna Das¹, Bipasha Biswas Mallick², Aditi Chakraborty³, Santosh Nandi⁴, Saurabh Dutta⁵

^{1,2}Department of Computer Applications, Haldia Institute of Technology, Haldia-721602, WB, India.

^{3,4}Department of Computer Science, Panskura Banamali College, Panskura, Paschim Medinipur, W.B. India.

⁵Department of Computer Applications, Dr. B. C. Roy Engineering College, Durgapur-713206, WB, India.

ORCID: ¹0000-0003-2354-9904, ²0000-0001-9156-0588, ³0000-0001-9654-4552, ⁴0000-0001-6822-1158, ⁵0000-0001-5892-0094

Abstract

The objective of this paper is to propose a CAPTCHA system based on personal information and likings of user which restricts unauthorized access with better security compare to existing systems. We have proposed a system where the CAPTCHA code will be generated automatically depending upon the user's personal information and on some chosen options by the user as per their likings. All the characters of the personal information and the chosen options of the user with their own sequences are being represented by the images automatically loaded from image database with random manner. Selection of actual representative images corresponding to the characters of personal information and the chosen options by the user with their proper sequences is the criteria for authentication. It is very much hard to select actual representative images with proper predefined sequences by an unauthenticated programs, robots or humans within limited attempts. Thus security is increased.

Keywords: image based CAPTCHA; random sequence; input based CAPTCHA generation; sequence based authentication.

INTRODUCTION

CAPTCHA restrict automated program from unauthorized access by taking some test. The CAPTCHA tests includes codes in the form of images, letters, alphabets and numbers in intersected or overlapped manner[1][2]. Users have to read and rewrite the code in specified pattern to gain the access of the system. These codes are easy to understand and rewrite for any human being but not for the automated program. So in that way CAPTCHA restricts various automated programs to sign-up E-mail accounts, cracking passwords, spam sending, privacy violation etc.

A. Background Study

Origin of image based CAPTCHA is being motivated from

graphical password which has been introduced by R. Biddle [3] in early of 1999. This knowledge based authentication approach provides great security compare to existing text based authentication approach. A new idea has been introduced by U. Padhye [4] based on image CAPTCHA used as graphical password. Image CAPTCHA is being introduced by Chew and Tygar [5] for the first time. They uses images or perhaps photographs to generate Image CAPTCHA. Their focus on image recognition CAPTCHA where the user have to recognize the image which he/she have selected earlier. Image recognition CAPTCHAs are usually database-based. Kittenauth[6] introduced image recognition CAPTCHA where a varieties of images of same object are shown to the user. Asirra[7] represents a system where it uses a closed database to source the images. Users have to select the proper image from the closed image pool. These systems show a common weakness that the numbers of solutions are so small that a random guessing can have a high probability of success. Aditya Raj [8] incorporated the concept of sequencing in image CAPTCHA. This idea is known as SPC. In this approach there are two levels of security, first the user has to recognize the image and the second is to determine the corresponding logical sequence of the recognized image. Gaurav Goswami.[9] introduce a new image based CAPTCHA based on facial features, facial expressions and face recognition. The user has to identify only two similar faces matching each other with the expressions and facial features. Most of the existing CAPTCHA system focus on the server generated codes which restrict the access from the automated programs but not from unauthorized access from a human[10][11].

B. Objective of the Article

Here we have proposed a system where image based CAPTCHA[12][13] code will be generated automatically depending upon the personal information of the user and on some options chosen by the user as per their likings. The sequence of chosen options will also be a parameter for

CAPTCHA generation. In this proposed method, we first take the input of user personal information and the likings of the user as per their priorities at the user registration time. That information (personal information, likings of user and the sequence of chosen options) are being represented as images which will be loaded from the image data base to the CAPTCHA generation window in a random manner with combination of multiple fake images [14]. A user has to select all the images which represent their personal information and the likings by following the sequence in which they have input their information at registration time. CAPTCHA validation program match the inputted information (images and sequences) with the original information provided at the time of registration and thus the authentication is being done.

It is very much hard for an unauthenticated automated program or a human to select multiple numbers of images with proper sequence as per the information provided at registration time within a very small numbers of attempts. So an attempt is made to enhance the security of CAPTCHA system from unauthenticated access by the automated program as well as from the humans.

The concept is based on image recognition based CAPTCHA with logical sequencing. In the proposed system, we generate the representative images from the image database as per the inputted personal information and the options chosen by the user at registration time. Besides this, user has to input the sequence number for each of the representative image. Selection of representative images with proper sequences is the only criteria for authentication. So the concept of user input based image CAPTCHA has been introduced for the first time which impose a great security on the existing CAPTCHA system.

C. Structure of the Article

In this paper, Section-II discusses the preliminaries; Section-III describes the overall procedure. Section-IV and Section-V represent CAPTCHA generation process and user authentication process respectively. Experimental results are being described in Section-VI. Section-VII shows the comparison between different CAPTCHA process and Section-VIII draws conclusions.

PRELIMINARIES

A. CAPTCHA

CAPTCHA is an abbreviation which stands for Completely Automated Public Turing-test to tell Computers and Humans Apart. CAPTCHA restricts various automated programs to sign-up E-mail accounts, cracking passwords. CAPTCHA is simply some visual test or some simple puzzle. Any mentally sound human can crack this challenge very easily, but an

automated program will not be able to crack the challenge and it will not be able to gain access.



This CAPTCHA of "smwm" obscures its message from computer interpretation by twisting the letters and adding slight background color gradient.

OVERALL PROCEDURE

- Step 1: Take name, age, mobile number of the user as input.
- Step2: Reverse the value of the age , last two characters of the first name, the character 5 and character 4 of the mobile number and first two characters of the last name inputted by the user in reverse order and store those values.
- Step 3: There are four groups each of having four numbers of options represented by a character. A user has to select two numbers of options from each group as per his/her priority. The sequence of choosing two options in each group is being taken as an input also. The characters and the sequences of inputting characters will be stored.
- Step 4: 16 numbers of unique random numbers will be generated in the range of 0 to 19 for repositioning the characters and those numbers will be stored.
- Step5: Create an array to store 16 numbers of characters and 4 numbers of "spaces".
- Step 6: Characters retrieved from personal information and choose characters (as options) are being stored in an array with their proper sequence and randomize positioning of characters with "space" value is being present in another array .The original characters and the sequence of characters will be supplied to the CAPTCHA validation program to compare with user selected CAPTCHA code inserted at the login time. The basic structure of the image CAPTCHA is given in the figure 1.

Image of 2 chars in reverse order from age	Image of last 2 chars in reverse order from first name	Image of middle 2 chars in reverse order from mobile no	Image of first 2 chars in reverse order from last name	Image of 8 items/events corresponding to their selected options represented by characters
2 images	2 images	2 images	2 images	8 images

Figure 1: Basic structure of image CAPTCHA.

Step 7: load the 20 numbers of images of the corresponding character and the items to the CAPTCHA generation window from the image database as per the randomize character sequence.

Step 8: User is asked to select the 16 image of those items and specific positioned character as per their option selection and the characters present in their inputted information which he/she has provided earlier. User must have provide the sequence number for each image for validating the sequence of selected options and characters which he/she provided earlier.

Step 9: The selected images and their inputted sequences are being compared with the original images and with their sequence which is previously supplied to the CAPTCHA validation program to authenticate the user.

CAPTCHA GENERATION PROCESS

The CAPTCHA generation process is being carried but by executing the following algorithm sequentially

A. Algorithm for Taking User Inputs

Step 1: Take name, age, mobile number of the user as input. Convert those values in respect of characters and store them into arrays named N[], A[] and P[] respectively.

The algorithm is as follows:

```
BufferedReader bu3=new BufferedReader(new InputStreamReader(System.in)); System.out.println("Enter the name of the user"); String nnn=bu3.readLine();
```

```
BufferedReader bu4=new BufferedReader(new InputStreamReader(System.in)); System.out.println("Enter the age of the user"); String aaa=bu4.readLine();
```

```
BufferedReader bu5=new BufferedReader(new InputStreamReader(System.in)); System.out.println("Enter the ph no");
```

```
String ppp=bu5.readLine(); N = nnn.toCharArray();
```

```
A = aaa.toCharArray(); P = ppp.toCharArray();
```

B. Algorithm for Character Sequencing

Step1: Reverse the value of the age inputted by the user and stores it into the array named FIN[];

Step 2: Read the last two characters of the first name inputted by the user in reverse order and store it into the array FIN[] just after the position where the pervious age value has been stored.

Step 3: Read the character 5 and character 4 of the mobile number inputted by the user and store it into the array FIN[] just after the position where the pervious first name value

has been stored.

Step 4: Read the first two characters of the last name inputted by the user in reverse order and store it into the array FIN[] just after the position where the pervious mobile no value has been stored. So there will be 8 numbers of characters in the FIN [] array. Algorithm is as follows:

```
//storing age // l=nnn.length(); t=-1; FIN[++t]=a[1]; FIN[++t]=a[0]; //storing first name // for(i=1;i<=l;i++) { if (n[i]==' ') {pos=i; break; }} i=i-1; FIN[++t]=n[i]; i=i-1; FIN[++t]=n[i]; // storing mobile number // FIN[++t]=p[5]; FIN[++t]=p[4]; //storing last name //pos=pos+2; FIN[++t]=n[pos]; pos=pos-1; FIN[++t]=n[pos];
```

C. Algorithm for Taking Inputs for Option Selection

Step 1: There are four groups each of having four numbers of options represented by a character. The groups are hobby selection, colour selection, playing selection and festival selection. A user has to select two numbers of options from each group as per his/her priority. The sequence of choosing two options in each group is being taken as an input also.

Step 2: Those selected options in each group will be stored in arrays named FIN[] just after the position where the previous last name value has been stored.

Step 3: Corresponding sequence of each of the stored characters in the FIN [] array will be stored in the array named SEQ[] sequentially. So there will be 16 numbers of characters in the FIN [] array. The algorithm is as follows:

```
System.out.println("Enter the choice as per your priority of linkings the items"); System.out.println("\n Hoby:- A-painting B-singing C-dancing D-net surfing"); System.out.println("\n playing:- F-cricket G-football H- hockey I-car racing"); System.out.println("\n color:- J- red K-green L-blue M-yellow"); System.out.println("\n festival:- N-durga puja O-xmas P-eid Q-dipawali"); System.out.println("choose any two items for each group that means enter 8 choice\n");
```

```
for(i=8;i<=15;i++){ bop = (char) new InputStreamReader(System.in).read (); FIN[i]=bop; } int SEQ[]= new int[17];
```

```
System.out.println("charecters and their sequences are");
```

```
for(i=0;i<=15;i++){ System.out.print(FIN[i]+" "+SEQ[i]);}
```

D. Algorithm for Random Sequence Generation

Step 1: 16 numbers of unique random number will be generated in the range of 0 to 19 by using the rand function in the following way: ran = randomGenerator.nextInt(20);

We will store those 16 unique random values in the array named AA[] sequentially. The Nth (where N=1 to 16) random value from the array AA[] will be applicable for the Nth

(where N=1 to 16) character of the array FIN[] for repositioning the character into the array FIN1[].The algorithm is as follows:

```
int AA[] = new int[17]; int ran,y,flag; i=0; Random random
Generator = new Random(); while(i<=15) {flag=0; ran=
randomGenerator.nextInt(20); if(i==0) AA[++i]=ran; else
{ for( y=1;y<=i; y++) { if(AA[y]==ran){flag=1;break;}} if
(flag==0) AA[++i]=ran;}}System.out.print ("rand sequence
");for(i=1;i<=16;i++) System.out.print(AA[i]);
```

E. Algorithm for CAPTCHA Generation

Step1: Create an array named FIN1 [] of size 20. This array will contain 4 numbers of "space" value and 16 numbers of characters from the FIN [] array as those characters will be repositioned here as per their random position value. The Nth (where N=1 to 16) random value from the array AA [] will be applicable for the Nth (where N=1 to 16) character of the array FIN [] for repositioning the character into the array FIN1[]. Characters will be stored into that position of FIN1 [] array which will be defined by the random value applicable for that character. Algorithm is as follows:

```
//implementing sequence for characters
char FIN1[]={ ' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' '};
int zz,gg; char cc; for(gg=1;gg<=16;gg++) {cc=FIN[gg];
zz=aa[gg]; FIN1[zz]=cc; }System.out.println(" rand
orientation of the characters are");for(gg=1;gg<=20;gg++)
System.out.print(" "+FIN1[gg]+"| pos: "+gg);
```

Step 2: Original sequence of inserted and choose characters (as options) are being stored in FIN [] array and randomize positioning of characters with "space" value is being present in FIN1 [] array .The original sequence of characters will be supplied to the CAPTCHA validation program to compare with user selected CAPTCHA code inserted at the login time.

Step 3: load the 20 numbers of images of the corresponding character and the items to the CAPTCHA generation window from the image database as per the randomize character and their sequence present in the FIN1 [] array. The algorithm is as follows.

```
For ( i=1 to 20) Image i. Picture = LoadPicture (App.Path &
"\ picture name ".jpg") End for.
```

Here App. Path provides the path of the image database, picture name is the name of the picture which will be loaded in the ith picture box defined from the randomize repositioned characters present in the FIN1 [] array. This activity will be repeated for 20 times and the CAPTCHA image array will be generated in the window.

USER AUTHENTICATION PROCESS

Step 1: User is asked to select the 16 image of those items and specific positioned character as per their option selection and the characters present in their inputted information which he/she has provided earlier. User must have provide the sequence number for each image for validating the sequence of selected options and characters which he/she provided earlier.

Step 2: The selected images and their inputted sequences are being compared with the original images and with their sequence which is previously supplied to the CAPTCHA validation program from the FIN[] array.

Step 3: If the comparison gives true result then the user will be authenticate to access the system otherwise his/her access will be blocked.

RESULTS AND DISCUSSIONS

A. Result of CAPTCHA Generation Process



Figure 2: Personal information and chosen options based character string.

In the figure-2, first two characters holding the reverse value of age, next two characters holding reverse value of the last two character of the first name, next two characters holding two middle digits of the user mobile no, next two characters holding the reverse value of the first two characters of the last name and the next 8 characters holding the options as the user

have inserted. This string will be supplied to CAPTCHA validation program to authenticate user selected CAPTCHA image.

Random numbers generated for character repositioning from Random number Generator are:-
 12, 18, 13, 16, 14, 15, 5, 17, 2, 7, 3, 11, 19, 8, 6, 4.

Figure 3: Generation of random numbers.

In the figure-3 sixteen numbers of random numbers are being generated where the 1st random number will be applicable on the first character to repositioning it and so on.

Random positioning of the characters as per the random numbers are:-

Character-		position- 1
Character-	A	position- 2
Character-	I	position- 3
Character-	Q	position- 4
Character-	a	position- 5
Character-	N	position- 6
Character-	D	position- 7
Character-	J	position- 8
Character-		position- 9
Character-		position- 10
Character-	G	position- 11
Character-	5	position- 12
Character-	n	position- 13
Character-	4	position- 14
Character-	5	position- 15
Character-	a	position- 16
Character-	B	position- 17
Character-	3	position- 18
Character-	L	position- 19
Character-		position- 20

Figure 4: Repositioning of characters as per random sequence.

In the figure-4, All the characters are repositioned as per the corresponding random number. We provide the blank for the position 20,9,10 and 1 as these positions are not present in the random number list. Now as per this sequence of character CAPTCHA image will be loaded from the image database.

Form1

Enter the original sequence of characters for captcha image to validate the user authentication

53na45aBADIGLJNQ

Enter the randomize sequence of characters to generate the 20 image based captcha

AIQaNDJ G5n45aB3L

Generate Captcha Image stream

Figure 5: Input of original sequence and randomize sequence of characters and options.

Original sequence of characters for validating the user authentication and the randomize sequence of characters to generate the CAPTCHA image stream will be supplied automatically by the server as inputs to the interface to generate the image CAPTCHA. A timer function will initiate the execution of image CAPTCHA generation process automatically after 2 second of supplying the values of two sequences. Figure 5 represents the insertion of randomize and original sequences.

B. Result of User Authentication Process

Form2

Enter the sequence of the image[1,2,...](unselected image's text box left bank) in the corresponding text box as per your priority of likings the criteria inputted at registration time

Grid of images: Soccer player, Artist palette, Car, Fire, 'a', Buddha, Man at computer, Red square, Angel, Mosque, Soccer player, '5', 'n', '4', '5', 'a', Silhouette, '3', Blue square, Green square.

Submit selected captcha image and their sequences

Figure 6: Generation of image CAPTCHA.

Figure-6 represents the generation of image CAPTCHA. 16 numbers of original image and 4 numbers of fake image will be loaded on the CAPTCHA generation window from the image database automatically as per the randomize character string has been enter to this system. Fake images has been placed corresponding to that position of the randomize character string where the 'blank' values are present. Fake images are randomly chosen by server.

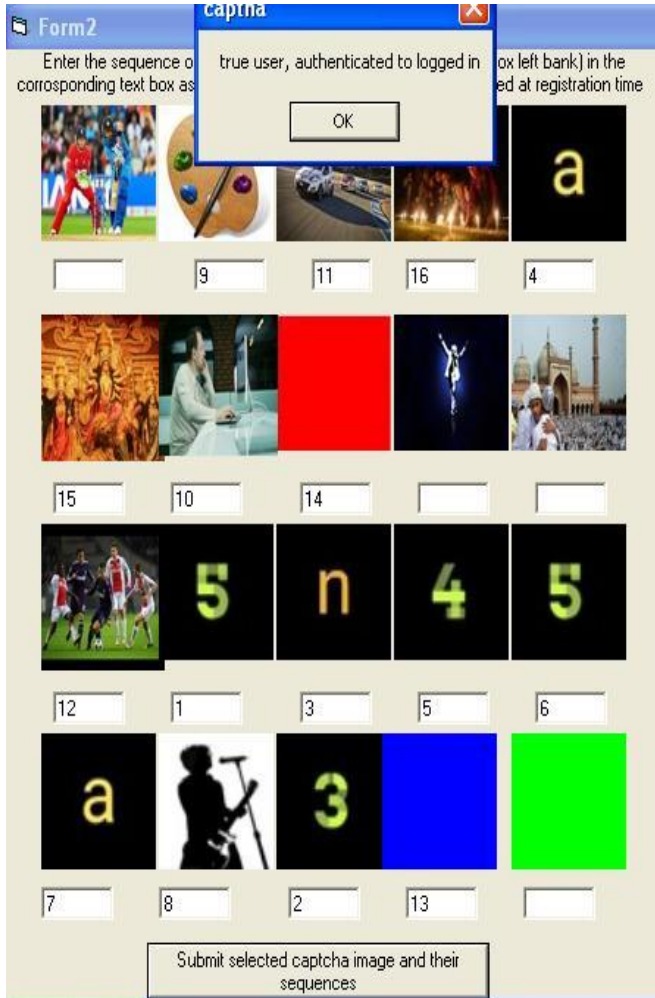


Figure 7: True user authentication by original images with proper sequences.

If a user select the proper images along with their proper sequence as per his/her choose options and the specified character present in his personal data which he/she has inputted earlier, then only he/she will be authenticate to access the system. Otherwise access will be denied. This procedure is being represented in figure-7.

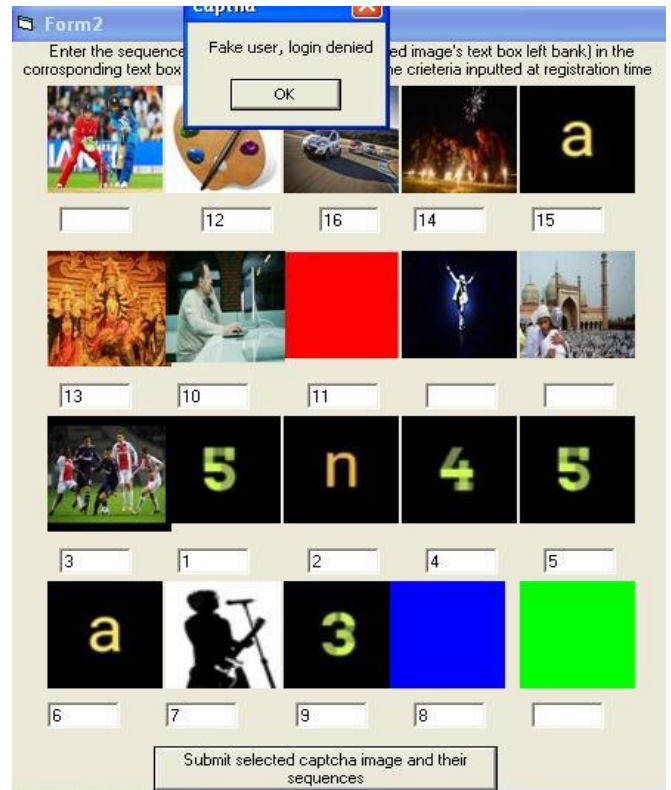


Figure 8: Fake user authentication by original images but with wrong sequences.

If a user select proper images with wrong sequence then his/her access will be denied. Figure-8 represents the procedure.

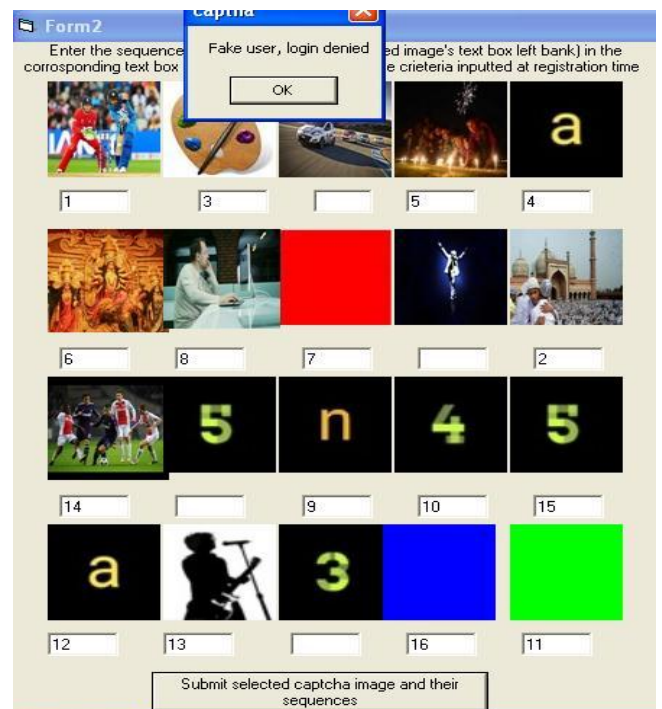


Figure 9: Fake user authentication by fake images with wrong sequences.

If a user selects wrong images with wrong sequence then the access will be denied also. Proper number of images with proper sequence will lead a user for successful authentication. Figure-9 represents the procedure. Thus the security of the system has increased.

COMPARISON BETWEEN CAPTCHA SYSTEMS

Table- I compares image based CAPTCHA systems.

Table I: Comparison Between Different Captcha Systems

Type of CAPTCHA	Core Idea
CAPTCHA as Graphical Passwords by R. Biddle [1]	Knowledge based Image selection approach.
Image recognition based CAPTCHA(from a set of different image) by Chew and Tygar [7]	Focus to select validate images from a list of different images.
Image recognition based CAPTCHA(from a set of varieties image of same object) by Kittenauth[12]	Focus to identify similar type of images of the same object.
Image CAPTCHA based on sequences in images by Aditya Raj [3]	Focus to recognize the image with the corresponding sequence of that image.
Facial features based image CAPTCHA by Gaurav Goswami.[4]	Focus to identify two similar faces based on some facial features.
Secured CAPTCHA code based on personal information and likings of user.	Based on recognition of images corresponding to inputs, selection of validated images with their corresponding sequences.

CONCLUSIONS

In this proposed CAPTCHA authentication system, user are being authenticated with the character selection from their personal data and the image selection corresponding to their choose options with proper sequence of selection. As all this information are private and variable to the user so it will be harder to select proper images for a unauthenticated user .So the security of the system will increase.

Authentication is also based on the proper sequence of characters and images as per the sequence of selected option at the CAPTCHA generation time. So it will be very much hard to insert proper sequence for 16 numbers of images for a

unauthenticated user, thus the security will be enhanced.

A user can make only 3 number of attempt to select the images with proper sequence. Otherwise the system will be blocked. So it will be very tuff to select 16 proper pictures with their corresponding sequences in 3 attempts for a unauthenticated user. Thus an attempt is made to increase the security of the system.

User selected options are not been permanently store to the database. So there is no chance to steeling of that information and security will be increase.

User may change the option selection and also the sequences of selecting options as they logged in every time. Thus that information will varies for each log in time. So the security will be increased.

As CAPTCHA code is used to block the other computer automated program which is repeatedly making fake access request to a server, so this proposed system will restrict those program in the following manner;-

Automated program may not be able to enter a set of inputs (personal data, option selection) by reading the instruction stepwise.

Similarly automated program may not be able to select proper images and their sequences at the authentication validation time as it does not have the information(as it does not insert any data at CAPTCHA generation time so no information will be present in its database) to do so.

Automated program may not be able to select the proper image corresponding to the selected option. Such amount of intelligence may not be present in it.

So based on the discussed points, the security of the proposed system will be increased as per the accessed by the automated programs. Thus an attempt is made to provide strong security to the proposed system against the unauthenticated access.

REFERENCES

- [1] Biddle, R., Chiasson, S., Van Oorschot, P.C., 2012, "Graphical passwords: Learning from the first twelve years", ACM Computer. Surveys, vol. 44, issue. 4.
- [2] Padhye, U., Kansare, P., Chavan, K., Shinde, D., Mangale, D., April, 2017, "Image based CAPTCHA as a Graphical Password", International Journal of Innovative Research in Computer and Communication Engineering, vol. 4, issue. 4, pp. 6652-6659.
- [3] Raj, A., Jain, A., Pahwa, T., Jain, A., 2010, " Picture CAPTCHAs with Sequencing: Their Types and Analysis", International Journal of Digital Society, vol. 1, issue. 3, pp. 208-220.
- [4] Goswami, G., Powell, B., Vatsa, M., Singh, R., Noore, A.,

- August. 2012, "FaceDCAPTCHA: Face detection based color image CAPTCHA", Future Generation Computer Systems, www.elsevier.com/locate/fgcs, doi:10.1016/j.future.2012.08.013.
- [5] Angre, A., Kapadia, M., Ugale, M., March. 2015, "PiCAPTION: Picture CAPTCHAs for Internet Authentication", International Journal of Computer Applications, (0975 – 8887), vol. 114, issue. 10, pp. 6-9.
- [6] Singh, N., Jain, S., January. 2015, "Dynamically Generated CAPTCHA: Selection of Random Number Generated CAPTCHA for Enhancing Security". International Journal of Application or Innovation in Engineering & Management (IJAIEM), www.ijaiem.org, vol. 4, issue. 1, pp. 126-131.
- [7] Chew, M., Tygar, J.D., Zhang, K., Zheng, Y., 2004, "Image recognition CAPTCHAs". in ISC 2004, LNCS, Springer, Heidelberg, pp. 268–279.
- [8] Elson, J., Douceur, J.R., Howell, J., Saul, J., Asirra, A, 2007, "CAPTCHA that exploits interest aligned manual image categorization", 14th ACM Conference on Computer and Communications. CCS '07.
- [9] Gao, H., Cao, F., Zhang, P., 2016, "Annulus: A novel image-based CAPTCHA scheme", Region 10 Conference (TENCON), IEEE, Singapore, 2016, ISSN: 2159-3450, DOI: 10.1109 /TENCON. 2016. 7848042.
- [10] The official CAPTCHA site. [Online]. Available: <http://www.captcha.net/>.
- [11] Drupal Projects.[Online].Available <https://www.drupal.org/project/confidentcaptcha/>.
- [12] Warner, O., The cutest human-test: Kittenauth, [Online]. Available: http://thepepspy.com/read/the_cutest_humantest_kittenauth/.
- [13] Berkely Projects,(2011), [Online]. Available: <https://www.ischool.berkeley.edu/projects/2011/picatcha-image-based-captcha/>.
- [14] Code Projects. [Online]. Available: <https://www.codeproject.com/Articles/18860/The-image-based-CAPTCHA/>.