Chapter 4

User Authentication Building Blocks

4.1. Overview

User authentication is a process to verify the identity of a user who wants to access the system. Different existing user authentication techniques are described in section 2.2 of chapter 2 of this thesis. Some newly developed user authentication schemes based on CAPTCHA (Completely Automated Public Turing-test to tell Computers and Humans Apart) and OTP (One Time Password) have been discussed and also the superiority of newly developed schemes has been elaborated.

In this current work, user authentication 'building blocks' are implemented using CAPTCHA Code based on User Personal information and Likings (CCUPL)[)1], Secret Value based on Randomized One Time Password (SVROTP)[2] and Numeric and biometric Image based One Time Password (NIOTP)[3] schemes. CAPTCHA or OTP generation time, their randomness and how they provide security over different network attacks are considered as standard parameters for assessing their performances.

In this chapter, CAPTCHA Code based on User Personal information and Likings (CCUPL) in section 4.2, Secret Value based on Randomize One Time Password (SVROTP) in section 4.3 and Numeric and biometric Image based One Time Password (NIOTP) in section 4.4 have been discussed. Section 4.5 shows the protection of OTP schemes over different cryptographic attacks. A conclusion has been drawn in section 4.6.

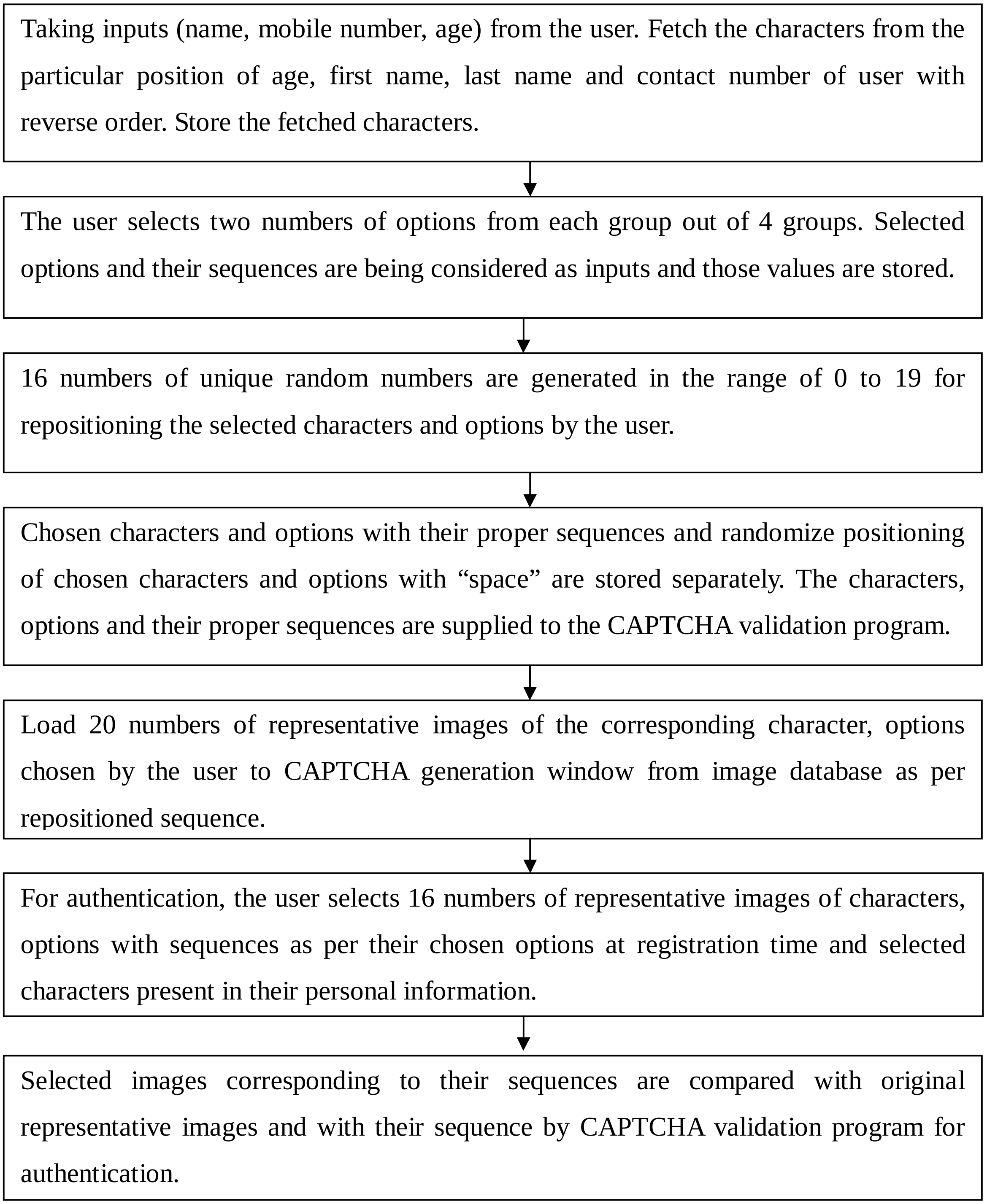
4.2. CAPTCHA Code based on User Personal information and Likings (CCUPL)

CAPTCHA stands for "Completely Automated Public Turing-test to tell Computers and Humans Apart". A test based on challenge & response is carried out to determine if a user is human or not. Automated programs are restricted by CAPTCHA from unauthorized access.

Here in the implemented CCUPL[1] scheme, an image based CAPTCHA is generated based on user personal information and options selected by the user at registration time. The sequence of selected options is also considered as a parameter for generating the CAPTCHA. In this method, user personal information and user likings are taken as inputs at registration time. User likings, personal information and sequence of selected options are represented as images. Those representative images and combination of fake images are loaded into the CAPTCHA generation window from an image database in random manner. For successful authentication, all representative images have to be selected as per user personal information and likings with their sequences as it has been inputted at registration time. Inputted information (images and sequences) are matched with the information provided at registration time by CAPTCHA validation program for authentication.

Selection of proper representative images with the proper sequence as per provided information at registration time is very hard to achieve by an unauthorized user within limited attempts. Thus the implemented CAPTCHA scheme blocks access from unauthorized automated programs, humans. Figure 4.1 represents the entire scheme for CAPTCHA Code based on User Personal information and Likings (CCUPL).

[1] Published in **International Journal of Applied Engineering Research (IJAER), SCOPUS indexed journal, 2017,** Volume 12, Number 16, pp. 5802-5809, with title An Approach to Implement Secured CAPTCHA Code based on Personal Information and Likings of User.



Taking inputs (name, mobile number, age) from the user. Fetch the characters from the particular position of age, first name, last name and contact number of user with reverse order. Store the fetched characters.

The user selects two numbers of options from each group out of 4 groups. Selected options and their sequences are being considered as inputs and those values are stored.

16 numbers of unique random numbers are generated in the range of 0 to 19 for repositioning the selected characters and options by the user.

Chosen characters and options with their proper sequences and randomize positioning of chosen characters and options with “space” are stored separately. The characters, options and their proper sequences are supplied to the CAPTCHA validation program.

Load 20 numbers of representative images of the corresponding character, options chosen by the user to CAPTCHA generation window from image database as per repositioned sequence.

For authentication, the user selects 16 numbers of representative images of characters, options with sequences as per their chosen options at registration time and selected characters present in their personal information.

Selected images corresponding to their sequences are compared with original representative images and with their sequence by CAPTCHA validation program for authentication.

Figure 4.1: Overall Procedure for CCUPL based Authentication Scheme

Section 4.2.1 represents process of CAPTCHA generation. Section 4.2.2 discusses the process of user authentication. Section 4.2.3 describes the implementation with experiment results. Section 4.2.4 shows the security analysis of the implemented CAPTCHA scheme.

4.2.1. Process of CAPTCHA Generation

Following algorithms are executed sequentially for carrying out CAPTCHA generation.

A. User Inputs Algorithm for CAPTCHA

Step 1: Accept contact number, name, age inputted by the user and convert inputs into characters which are kept in arrays named P[], N[] and A[] respectively.

Step 2: Use BufferedReader b1, b2, b3 for accepting inputs of user name, age and contact number respectively. Use String n, a, p for storing the values of user name, age and contact number.

Step 3: Converting the values into CharArray() and store name, age and contact number into N, A, P respectively.

B. Character Sequencing Algorithm

Step1: The value of inputted user age is reversed which is kept in array FIN[].

Step 2: Fetch the last two characters from the user first name in reverse order. Store the value after the position where ‘age’ value is kept in array FIN[].

Step 3: Fetch and store the 5th and 4th character of the inputted mobile number after the position where the value of ‘first name’ is kept in array FIN[].

Step 4: Fetch the first 2 characters of inputted last name in reverse order. Keep those values after the position where ‘mobile number’ is kept in array FIN[].

C. Option Selection Algorithm

Step 1: Each option is represented by a character. Each group has four numbers of options. Four numbers of such groups (selection of hobby, selection of color, selection of playing and selection of festival) are present in the implemented system. As per user priority, two options have to be selected from each group. Chosen options and the sequence of selecting options are considered as inputs.

Step 2: Chosen options from all groups are stored after the position where 'last name' is kept in the array FIN[].

Step 3: An array called SEQ[] holds the value of the sequences of stored characters in FIN[] array. FIN[] array has 16 numbers of characters.

D. Random Sequence Algorithm

Step 1: This algorithm generates sixteen numbers of random values where each value is unique and it belongs in the range from 0 to 19. The generation of random values is done in following manner: ran = randomGenerator.nextInt(20);

Those sixteen unique values are kept sequentially in an array called AA[]. Consider Nth random value (where 1<=N<=16) from array AA[] for repositioning the Nth character (where 1<=N<=16) of array FIN[]. Those repositioned characters are stored in an array called FIN1[].

E. CAPTCHA Generation Algorithm

Step1: Array called FIN1[] of size 20 holds sixteen characters from array FIN[] and four "space" values. Characters of array FIN[] are repositioned and stored into array FIN1[]. Consider Nth random value (where 1<=N<=16) from array AA[] for repositioning the Nth character (where 1<=N<=16) of array FIN[] into the array called FIN1[]. Random value applicable for a particular character defines the position in the array FIN1[] where the character is going to be stored.

Step 2: Fetched characters, chosen options and their sequence are kept in an array called FIN[]. Repositioned characters with ”space” value are kept in another array called FIN1[]. The chosen characters along with their original sequences are supplied to the CAPTCHA validation program to validate user authentication.

Step 3: Upload twenty numbers of representative images of the corresponding character, options chosen by the user to CAPTCHA generation window from image database as per repositioned sequence present in the array FIN1[].

4.2.2. Process of User Authentication

Step 1: For authentication, the user is asked to selects sixteen numbers of representative images of characters, options with their corresponding sequences as per their chosen options at registration time and selected characters present in their personal information.

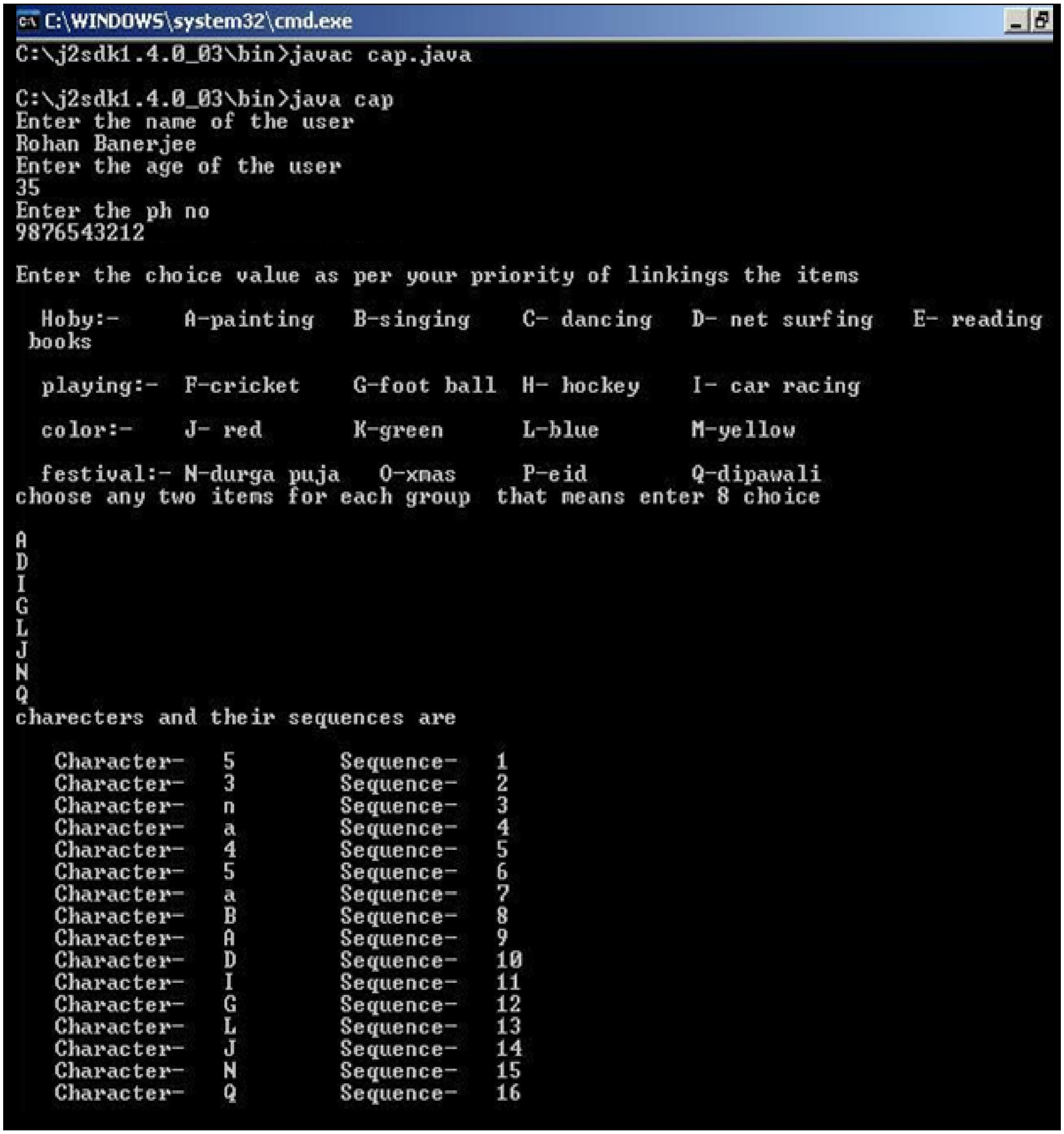
Step 2: Selected images corresponding to their sequences are compared with original representative images, their sequence by CAPTCHA validation program for authentication. Original representative images and their sequence are previously supplied from the array FIN[] to CAPTCHA validation program.

Step 3: The user is an authenticated one if the comparison returns true result.

4.2.3. Implementation with Experiment Results

A. Output of CAPTCHA Generation Process

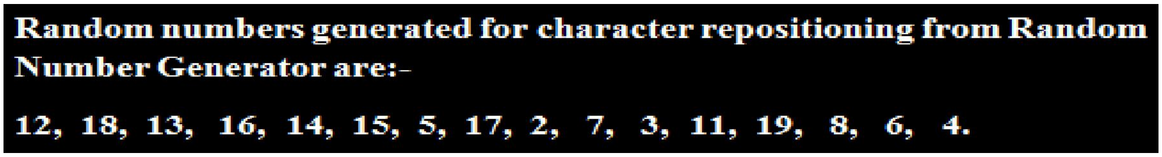
Reverse value of age is stored in the first two characters of outputted string, last two characters from the first name are kept in the next two characters of the outputted string in reverse order, next two characters of the outputted string contains the middle two digits of mobile number, first two characters from the last name are kept in the next two characters of outputted string in reverse order and lastly all user selected options are stored in next 8 characters of the outputted string. This outputted string is used by the CAPTCHA validation program for user authentication. Figure 4.2 represents the entire scheme.



```
C:\j2sdk1.4.0_03\bin>javac cap.java

C:\j2sdk1.4.0_03\bin>java cap
Enter the name of the user
Rohan Banerjee
Enter the age of the user
35
Enter the ph no
9876543212

Enter the choice value as per your priority of linkings the items

  Hoby:-     A-painting   B-singing    C- dancing   D- net surfing   E- reading
 books

  playing:-  F-cricket    G-foot ball  H- hockey    I- car racing

  color:-    J- red       K-green      L-blue       M-yellow

  festival:- N-durga puja   O-xmas     P-eid        Q-dipawali
choose any two items for each group  that means enter 8 choice

A
D
I
G
L
J
N
Q
charecters and their sequences are

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

Figure 4.2: Representation of Character String generated from User Personal Information and Selected Options

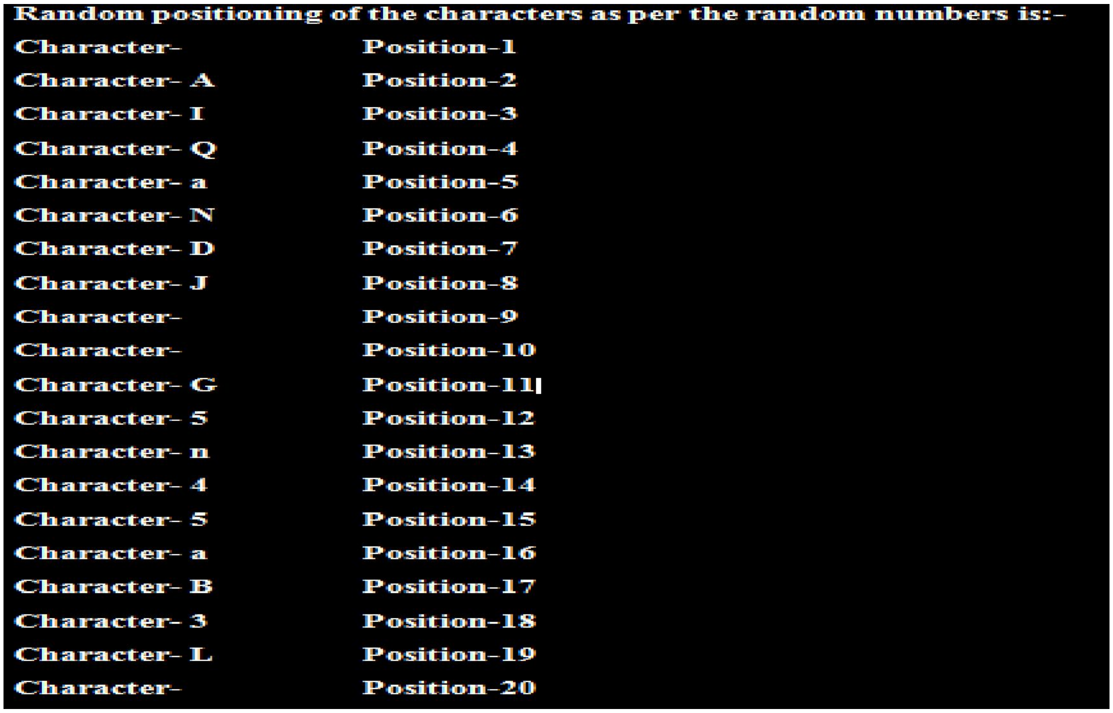
Sixteen Random values are generated in between the range 0 to 20 where ith (i=1 to 16) random value is applied on ith (i=1 to 16) character of outputted string for character repositioning. Figure 4.3 demonstrates the entire scheme.



```
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 4.3: Random Number Generation for Character Repositioning

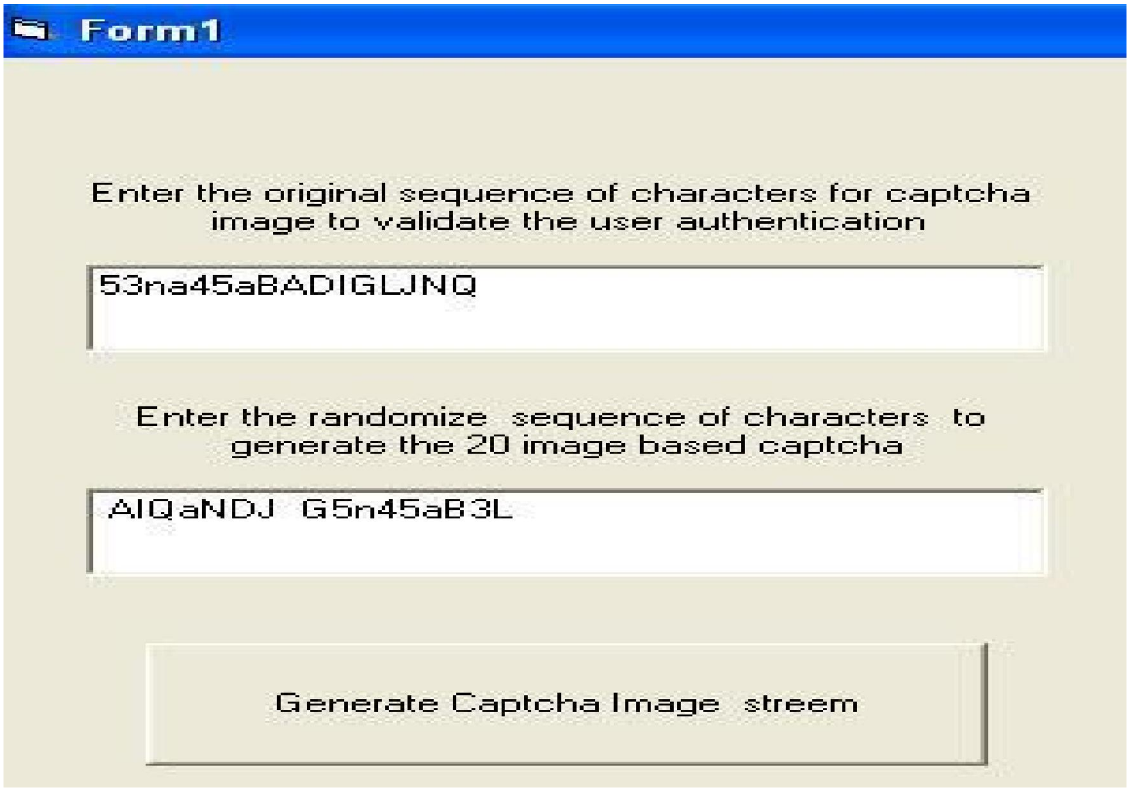
As per the generated random values, all characters from the outputted string are repositioned. The blank value is inserted to those positions (here position 20,9,10 and 1) which are not been mentioned by the random values. Images are loaded into CAPTCHA generation window as per the character sequence in the outputted string. Figure 4.4 represents the entire scheme.



```
Random positioning of the characters as per the random numbers is:-
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 4.4: Representation of Character Repositioning based on Random Sequence

The server automatically inputs the randomize sequence and original sequence of characters to the CAPTCHA generation interface. Original sequence is used for authentication validation and randomize sequence is used for CAPTCHA generation. Image CAPTCHA generation is automatically initiated after supplying the sequences. Figure 4.5 demonstrates the scheme.



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

53na45aBADIGLJNQ

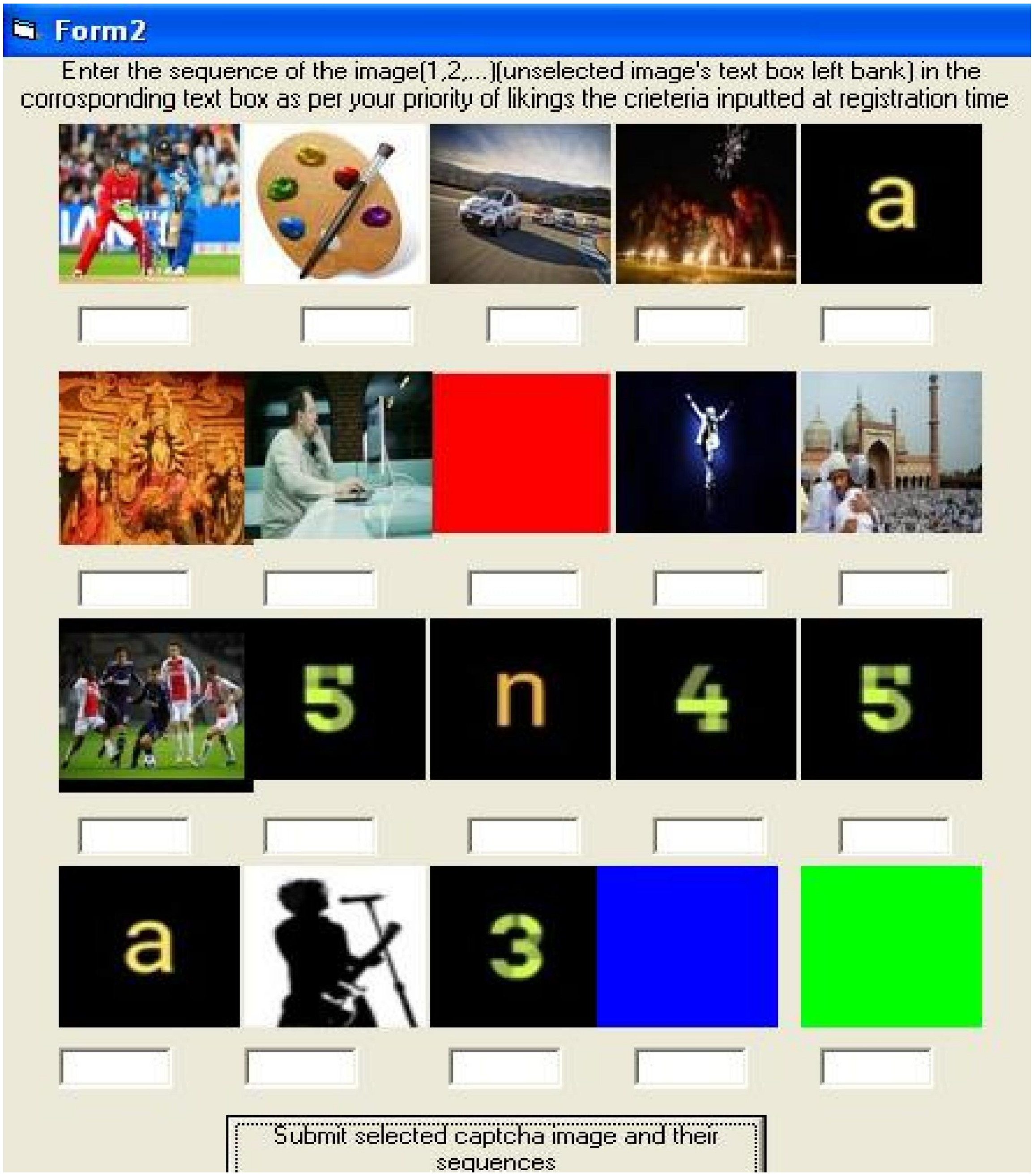
Enter the randomize sequence of characters to generate the 20 image based captcha

AIQaNDJ G5n45aB3L

Figure 4.5: Acceptance of Inputs by CAPTCHA Generation Method

B. Output of User Authentication Process

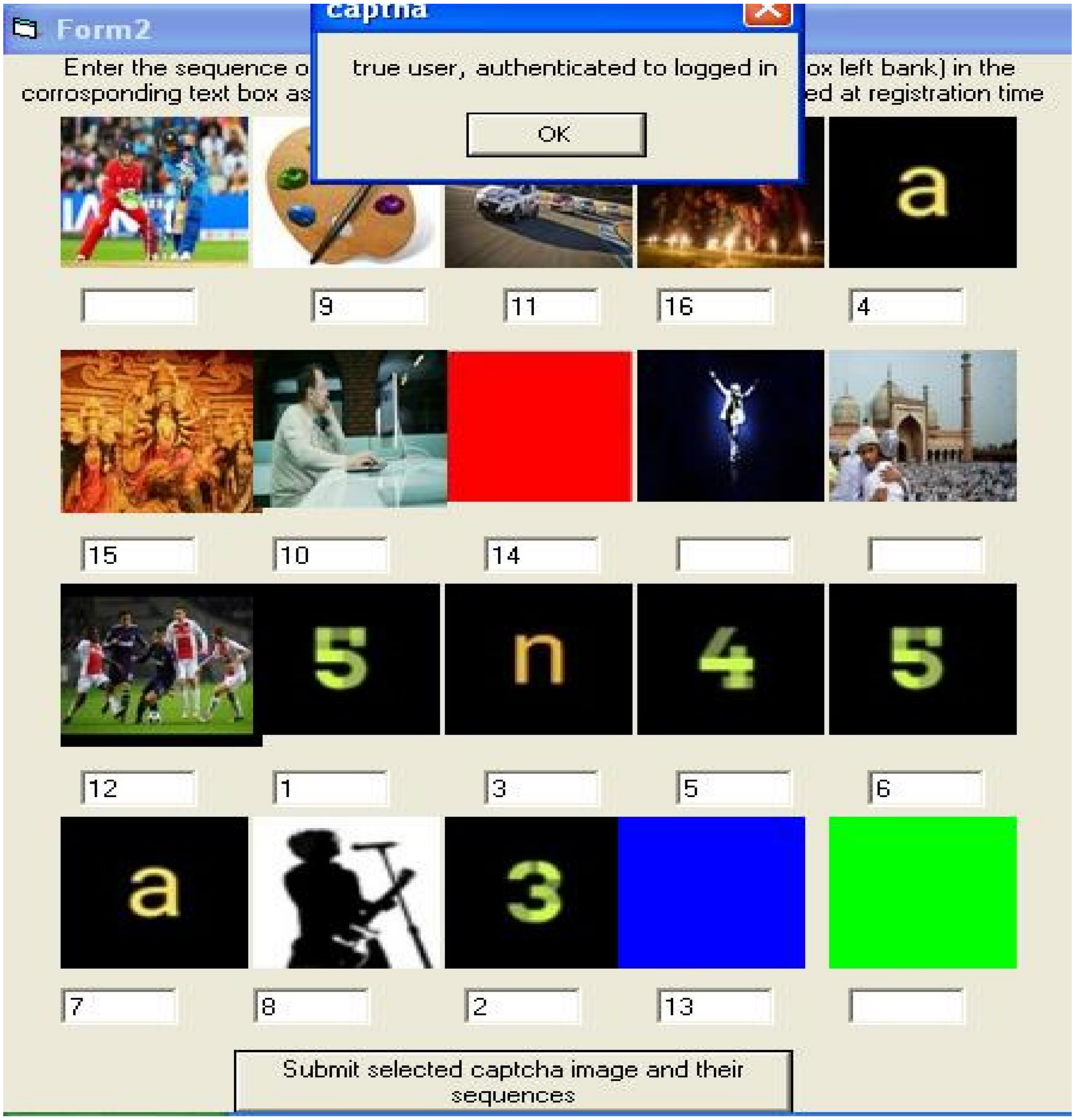
Original images (sixteen numbers) and fake images (four numbers) are loaded into the CAPTCHA generation window based on the inputted randomized character string from the image database. Server randomly selects fake images from an image database which are placed based on the ‘blank’ values present in the randomize character string. Figure 4.6 demonstrates CAPTCHA generation process.



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

Figure 4.6: Image based CAPTCHA Generation for CCUPL Scheme

Selection of actual representative images and their proper sequences are the only criteria for user authentication. If the inputted images and sequences are matched with the defined characters present in personal data and the options chosen by the user previously, then the user is an authenticated one otherwise not. Figure 4.7 represents the entire activity.



Form2

captha

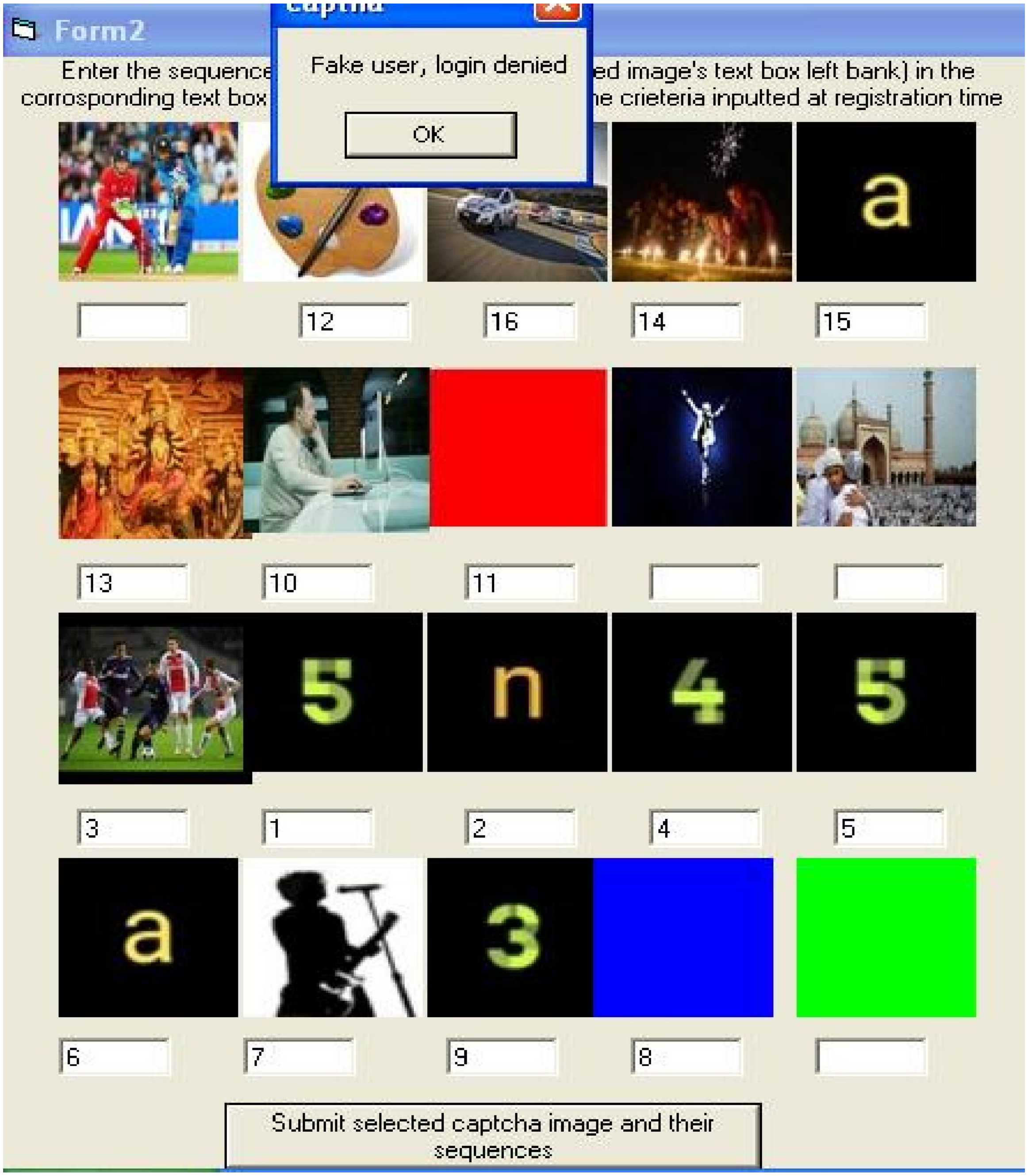
Enter the sequence o ox left bank) in the
corrosponding text box as ed at registration time

true user, authenticated to logged in

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

Figure 4.7: Valid User Authentication based on Selection of Proper Images and Sequences for CCUPL Scheme

Selection of proper images with the wrong sequence blocks the user authentication. Figure 4.8 demonstrates the fact.



Fake user, login denied

Enter the sequence ed image's text box left bank) in the
corrosponding text box e crieteria inputted at registration time

12 16 14 15

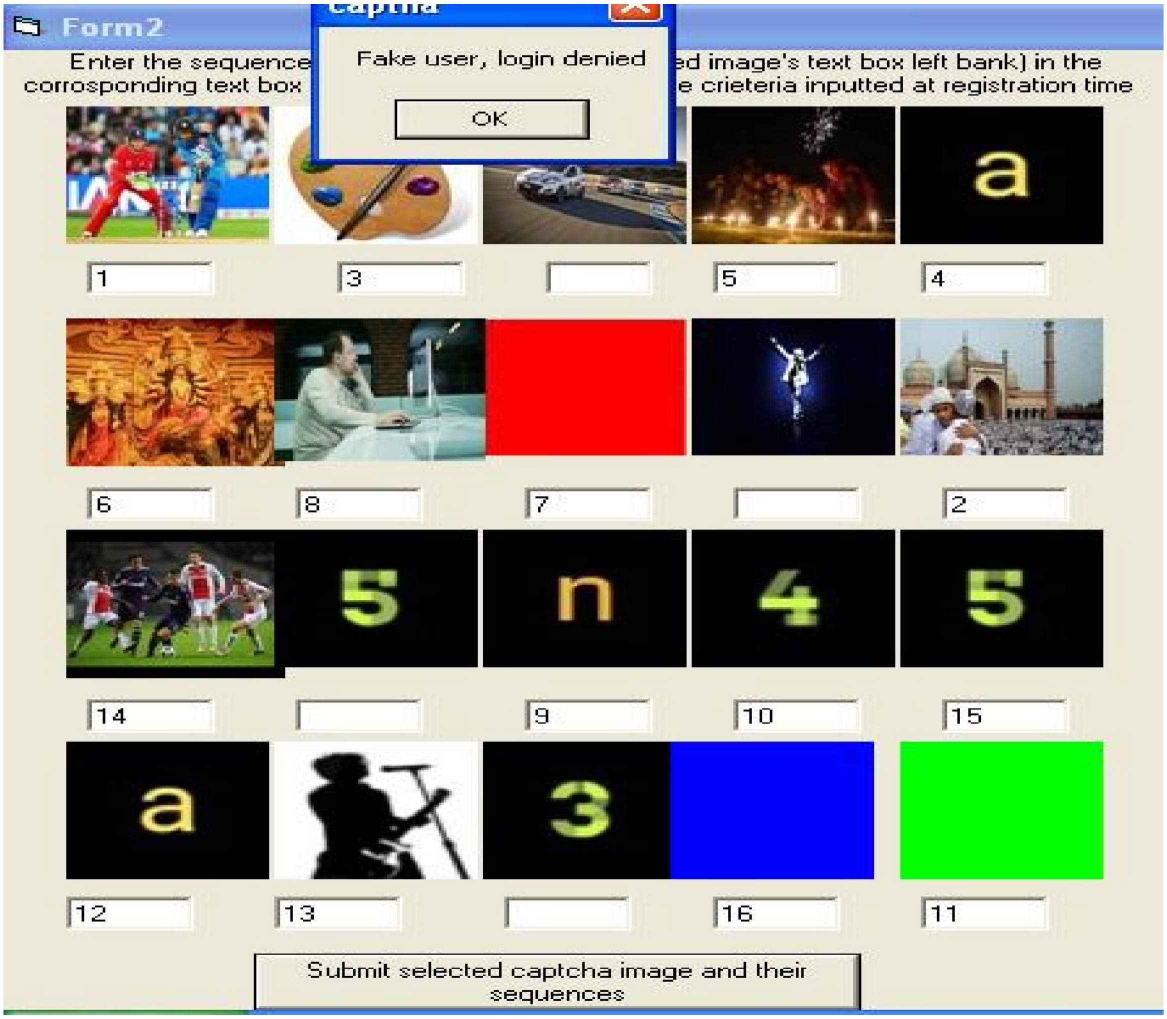
13 10 11

3 1 2 4 5

6 7 9 8

Figure 4.8: Fake Authentication based on Selection of Proper Images but wrong Sequences for CCUPL Scheme

Selection of wrong images with the wrong sequence also blocks the user authentication. Figure 4.9 represents the activities.



Enter the sequence ... ed image's text box left bank) in the corrosponding text box ... e crieteria inputted at registration time

Fake user, login denied

1 3 5 4

6 8 7 2

14 9 10 15

12 13 16 11

Figure 4.9: Fake Authentication based on Selection of Wrong Images and Wrong Sequences for CCUPL Scheme

4.2.4. Security Analysis for CAPTCHA Code based on User Personal information and Likings (CCUPL)

A. Security over Different Cryptographic Attacks

Table 4.1 represents the percentage of security provided by the implemented CAPTCHA scheme over Cryptographic Attacks.

Table 4.1: Security Analysis for CCUPL Scheme over Cryptographic Attacks

Name of The Attack	**Number of Unauthorized Login Attempts**	**Security over Unauthorized Access (in percentage)**	**Remarks**
Brute Force Attack	75	84%	Authentication is based on the proper sequence of characters and images as per the sequence of the selected option at CAPTCHA generation time. So it is very much hard to select actual representative images and insert their proper sequence by random guessing for 16 numbers of images for an unauthenticated user within a limited number of attempts. Thus the system is secured.
Dictionary Attack	68	87%	User selected options are randomly sequenced each login time for generating the images and not been permanently store to the database which blocks dictionary attack as the random selection of characters and options extremely reduces the probability of unauthorized access attempts from the combination of sequences generated from dictionary attacks.
Key Logger Attack	70	81%	As per Key Logger Attack, if all the keystrokes are tracked still the system is secured as the placement of representative images are randomized for each login time. Thus the

			implemented system blocks Key Logger Attack.
Man in the Middle Attack.	64	79%	User selected options are not been permanently stored to the database. The user may change the option selection and also the sequences of selecting options as they logged in every time. Inputs are randomly sequenced for each attempt. Thus that information varies for each log in time. So there is no chance to stealing of that information from the authentication server and system is secured
Phishing Attack, Physical Access to Phone, Mobile Phone Trojans	65	73%	This kind of attack may only access inputted options and personal information but the identification and sequence of representative images corresponding to the inputs are extremely hard to achieve. Thus the implemented system is secured from this kind of attacks.

Figure 4.10 graphically represents the percentage of security measurement over different attacks by the implemented CAPTCHA scheme.

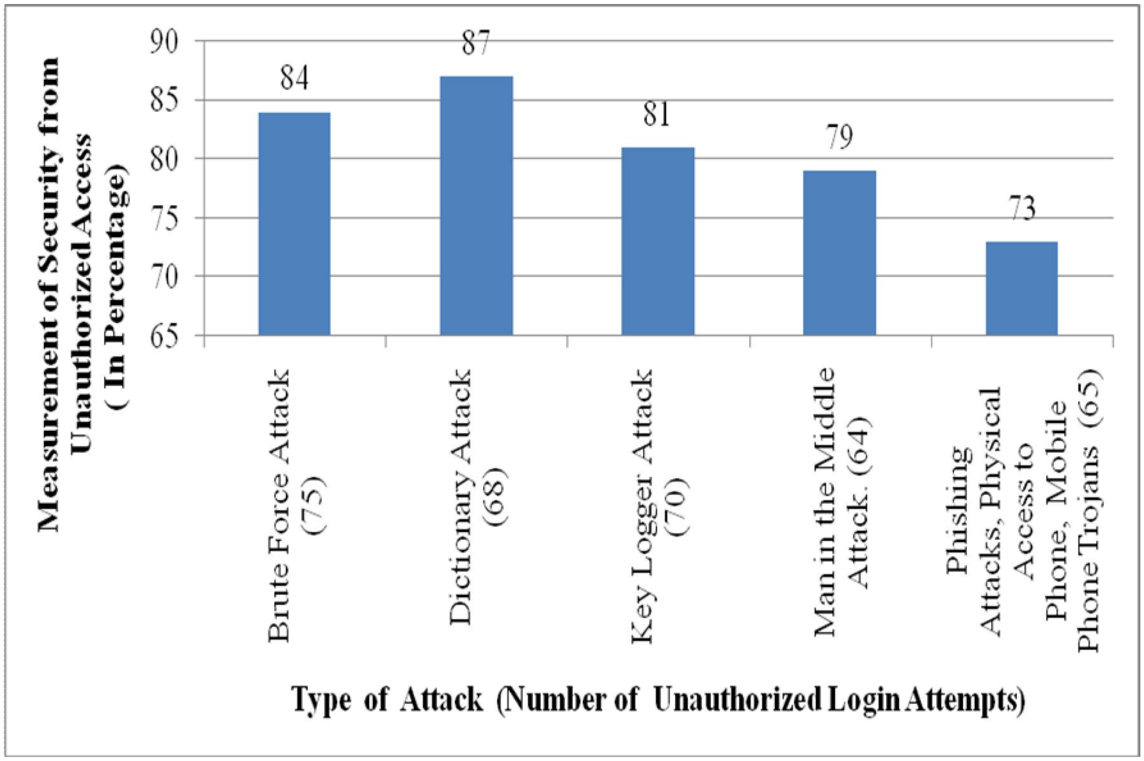


Figure 4.10: Graphical Representation of Security Analysis for CCUPL Scheme

B. Computation Time and Comparison with Existing CAPTCHA Schemes

CAPTCHA generation time is 65881 milliseconds and validation of user authentication needs 5731 milliseconds at server end where the server is installed in a computer with Intel Core i5 3.6 GHz processor and 4.00 GB RAM. Distribution of CAPTCHA is done through the public network and distribution time for sharing CAPTCHA between the server and receiver differs depending upon network congestion, noise, traffic and other obstacles present in the public network. Thus distribution time is not considered for the current scheme. Table 4.2 represents the comparison with existing CAPTCHA schemes with CCUPL scheme based on adopted CAPTCHA techniques.

Table 4.2: Comparison with Existing CAPTCHA Scheme with CCUPL Scheme

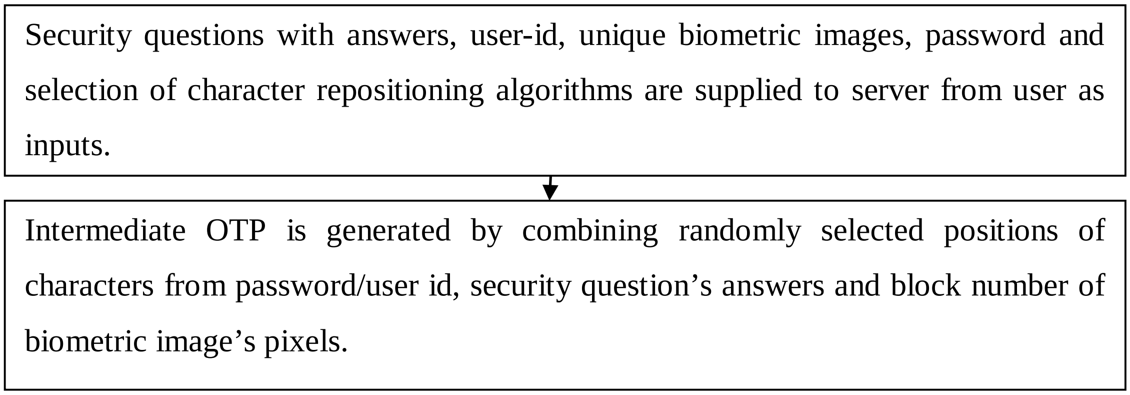
CAPTCHA Scheme	**Adopted CAPTCHA Generation and User Authentication Techniques**			
	Personal Information based CAPTCHA	**CAPTCHA based on Representative Images of user information**	**Recognition and Selection of CAPTCHA Image(s)**	**Sequencing of CAPTCHA Images**
Image Recognition based CAPTCHA [51][67][1]	NO	NO	YES	NO
Image Sequencing based CAPTCHA [3]	NO	NO	NO	YES
Facial Features based Image CAPTCHA[24]	NO	NO	YES	NO
CAPTCHA Code based on User Personal information and Likings (CCUPL)	YES	YES	YES	YES

4.3. User authentication scheme using Secret Value based on Randomize One Time Password (SVROTP)

One Time Password (OTP) contains alphanumeric or numeric character string. The server automatically generates OTP and OTP is valid for a single login session or transaction for a digital device. OTP provides an additional layer of security but distribution channel and device are needed for sharing OTP between server and user.

As the distribution of One Time Password through the public channel is not so much secured so new numeric OTP based SVROTP[2] scheme is introduced. Here secret value is retrieved from the private information (user id, password and security questions) and biometric image of user governed by OTP value. Secret value is used for authentication rather applying OTP value directly. Character positions and the number of pixel blocks are randomly selected by the server from modified private information and biometric image of the user. Positions of characters and pixel's block numbers are combined together and intermediate OTP is generated. Intermediate OTP is converted into final OTP using digit repositioning scheme and final OTP is shared to the user. The secret value which is used for authentication is extracted from private information and biometric image of user governed by intermediate OTP.

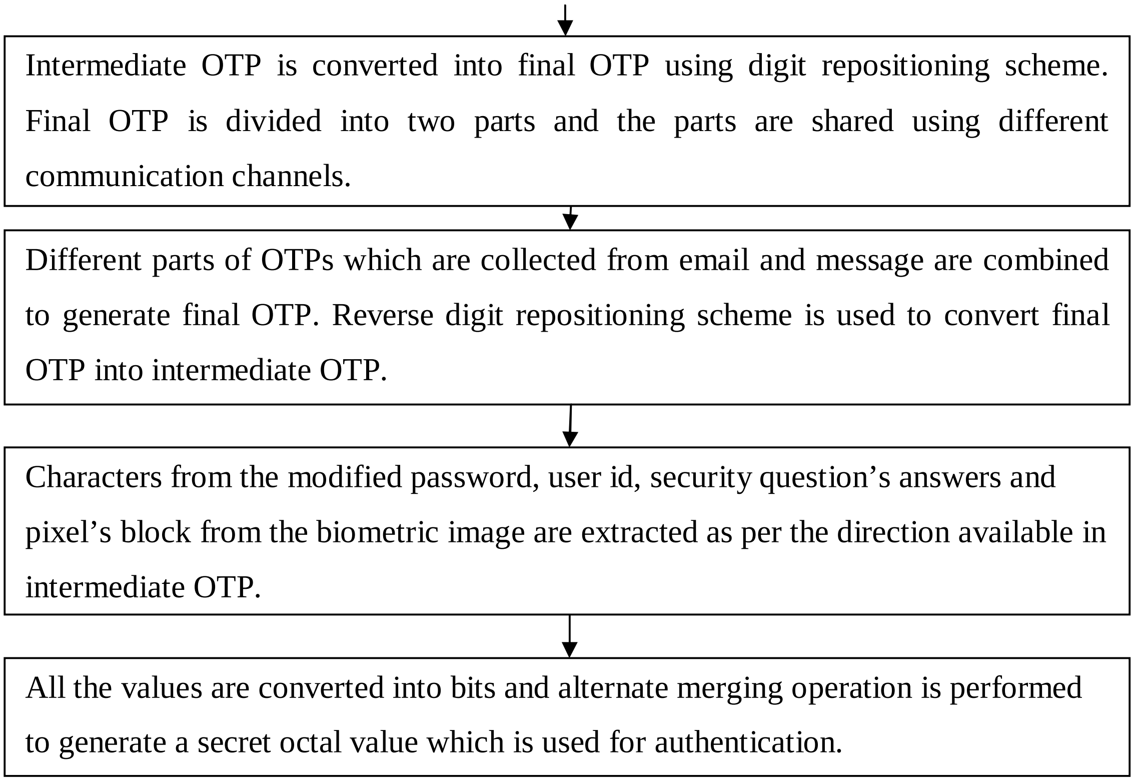
Random selection of information (characters and pixels), distribution of OTPs in the different communication channel, generation of different OTPs for distribution and authentication, retrieval and use of secret values for authentication governed by intermediate OTP provide great security on OTP based authentication system. Figure 4.11 represents the entire scheme for SVROTP based authentication system.



Security questions with answers, user-id, unique biometric images, password and selection of character repositioning algorithms are supplied to server from user as inputs.

Intermediate OTP is generated by combining randomly selected positions of characters from password/user id, security question's answers and block number of biometric image's pixels.

[2] Published in **International Journal of Applied Engineering Research (IJAER), SCOPUS indexed journal, 2017,** Volume 12, Number 19, pp. 8595-8601, with title An Approach to Implement Secure User Authentication Scheme using Secret Values extracted from Private Information and Unique Biometric Images of User directed by Randomize Numeric and Image based OTP.



Intermediate OTP is converted into final OTP using digit repositioning scheme. Final OTP is divided into two parts and the parts are shared using different communication channels.

Different parts of OTPs which are collected from email and message are combined to generate final OTP. Reverse digit repositioning scheme is used to convert final OTP into intermediate OTP.

Characters from the modified password, user id, security question's answers and pixel's block from the biometric image are extracted as per the direction available in intermediate OTP.

All the values are converted into bits and alternate merging operation is performed to generate a secret octal value which is used for authentication.

Figure 4.11: Overall Procedure for SVROTP based Authentication Scheme

Section 4.3.1 represents server side OTP generation. Section 4.3.2 discusses the distribution of the final OTP. Section 4.3.3 describes generation of secret value and authentication. Experimental results are described in section 4.3.4. Section 4.3.5 shows the security analysis of the implemented OTP scheme.

4.3.1. Server Side OTP Formation

A. Character/ Digit Repositioning Algorithms for Password/ User-Id modification in SVROTP scheme

The user selects one of the character repositioning algorithms for repositioning the characters of password / user-id separately. Character repositioning algorithms are described below.

PRONE (Positional Reverse Odd Normal Even)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

PRENO (Positional Reverse Even Normal Odd)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

CRENO (Continuous Reverse Even Normal Odd)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

CRONE (Continuous Reverse Odd Normal Even)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

B. Intermediate OTP Generation Algorithm

Step I: Random selections of positions of characters are carried out from randomly chosen either password or user-id and security question’s answers. Besides this, random selection of pixel block is also carried out from biometric images which are randomly chosen. Intermediate OTP is generated by combining the block number of pixels and positions of characters. Figure 4.12 represents the structure of intermediate OTP.

1st Block		2nd Block		3rd Block		
Character’s position randomly chosen from password/ user-id		Character’s position randomly chosen from security question’s answers		Random selection of pixel’s position and pixel’s block number from user biometric images		
Code for password / user-id selection (1/2)	Randomly chosen character’s value	Code for security question’s answers selection (1/2/3)	Randomly chosen character’s value	Code for biometric image selection (1/2/3)	Code for random pixel selection	Code for random pixel block selection (1/2/3/4)

Figure 4.12: Logical Structure of Intermediate OTP

C. Final OTP generation Algorithm

Intermediate OTP is converted into final OTP with the help of digit repositioning scheme. A single digit from each of the three blocks of intermediate OTP is fetched position wise and stored into an array named F_OTP[]. All the digits of intermediate OTP are fetched in this manner and generate final OTP.

D. Algorithm for Main () Function

Step I: Password, user-id, security questions with answers, user biometric images, selection of character repositioning algorithm for password/user-id modification are supplied as input to the server from the user.

Step II: Call Character Repositioning Algorithms for Password/ User-Id modification.

Step III: Call Intermediate OTP Generation Algorithm.

Step IV: Call Final OTP generation Algorithm.

4.3.2. Final OTP Distribution

Only final OTP is distributed from server to user by text message and email. Final OTP is divided into two sections. Intermediate OTP has not been shared which has to be derived from final OTP using proper repositioning algorithm. Generation of secret value which is used for authentication is governed by intermediated OTP.

4.3.3. Secret Value Generation and User Authentication Algorithm

Step I: Two sections of OTPs available in message and email are combined together to generate final OTP which is converted into intermediate OTP using digit repositioning scheme.

Step II: First four digits from intermediate OTP determines the two characters which have to be fetched from password or user-id and answer of security questions. Value of 5th to the last digit of intermediate OTP defines the block number and pixel number which have to be fetched. Besides this, it also defines the biometric image from where the pixel has to be selected.

Step III: All values are represented into binary form and alternate merging is performed between the bit representation of pixel’s block and characters. Secret value is generated in this manner which is applied for authentication. The server executes the same procedure to generate secret value. The secret value generated at the user and server end must have to be matched for validating the authentication.

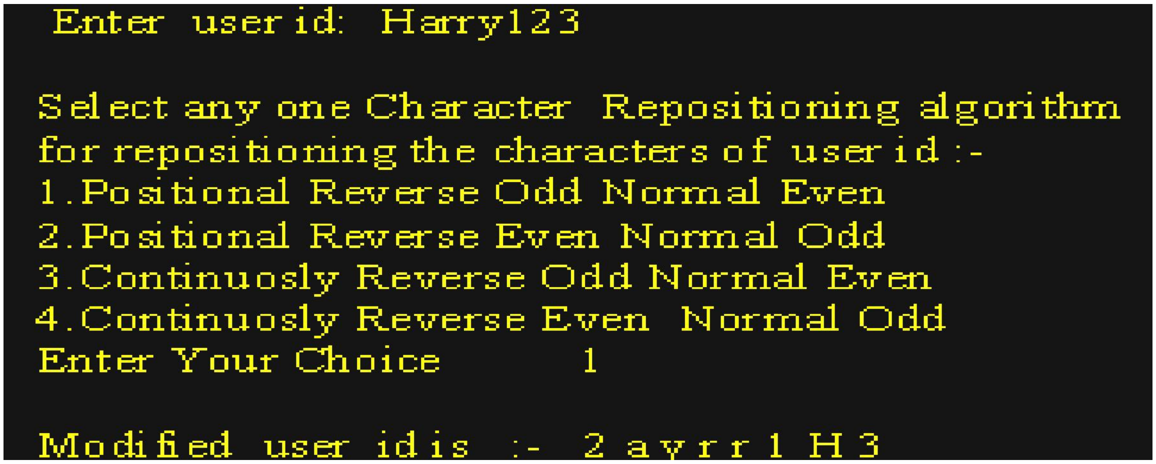
4.3.4. Implementation with Experiment Results

A. User Inputs to Authentication Server

At the time of user registration to the authentication server, a user has to provide password, user-id, selection for character repositioning algorithm and biometric images to the server as inputs.

B. OTP Generation at Server End

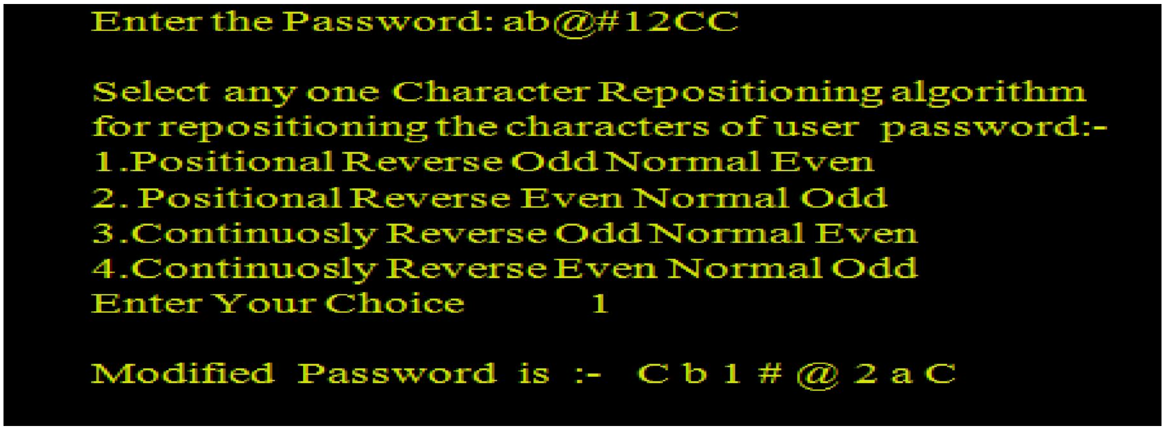
The user chooses a character repositioning algorithm for repositioning the characters of user-id. Figure 4.13 represents the entire activity.



```
 Enter user id:  Harry123

Select any one Character  Repositioning algorithm
for repositioning the characters of user id :-
1.Positional Reverse Odd Normal Even
2.Positional Reverse Even Normal Odd
3.Continuosly Reverse Odd Normal Even
4.Continuosly Reverse Even  Normal Odd
Enter Your Choice        1

Modified  user  id is  :-  2 a y r r 1 H 3
```

Figure 4.13: Generation of Modified User-Id

The User chooses a character repositioning algorithm for repositioning the characters of the password. Figure 4.14 represents the entire activity.

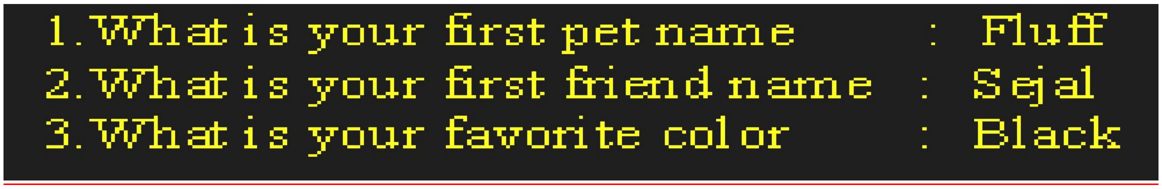


```
Enter the Password: ab@#12CC

Select any one Character Repositioning algorithm
for repositioning the characters of user  password:-
1.Positional Reverse Odd Normal Even
2. Positional Reverse Even Normal Odd
3.Continuosly Reverse Odd Normal Even
4.Continuosly Reverse Even Normal Odd
Enter Your Choice        1

Modified  Password  is  :-   C b 1 # @ 2 a C
```

Figure 4.14: Generation of Modified Password

Normal password and user-id are applicable for authentication whereas modified password and user-id are used to generate the secret value which validates user authentication.

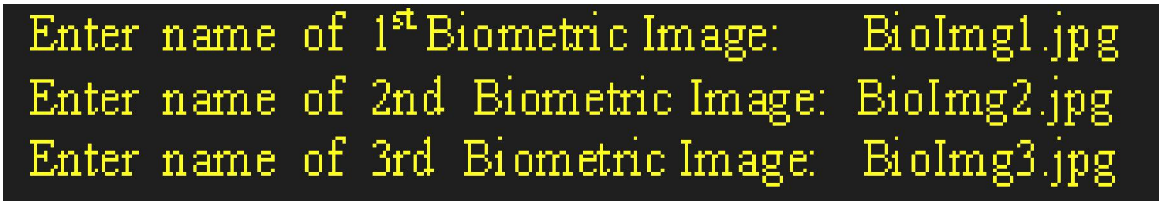
Figure 4.15 shows the security questions with their answers which are inputted by the user.



```
1.What is your first pet name      :  Fluff
2.What is your first friend name   :  Sejal
3.What is your favorite color      :  Black
```

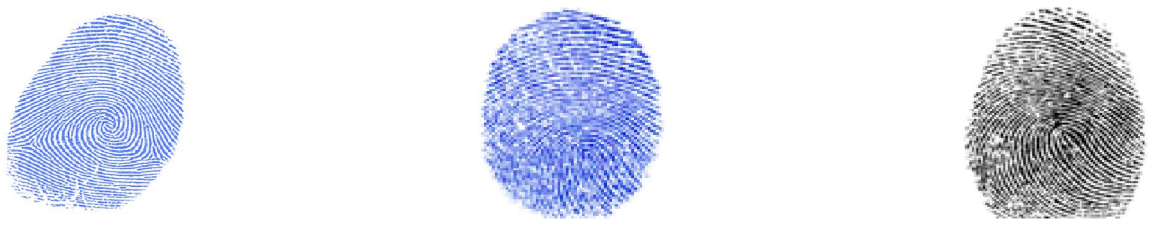
Figure 4.15: Inputs of Security Questions and their Answers

Figure 4.16 shows the biometric images which are inputted by the user.



```
Enter name of 1st Biometric Image:   BioImg1.jpg
Enter name of 2nd Biometric Image:  BioImg2.jpg
Enter name of 3rd Biometric Image:  BioImg3.jpg
```

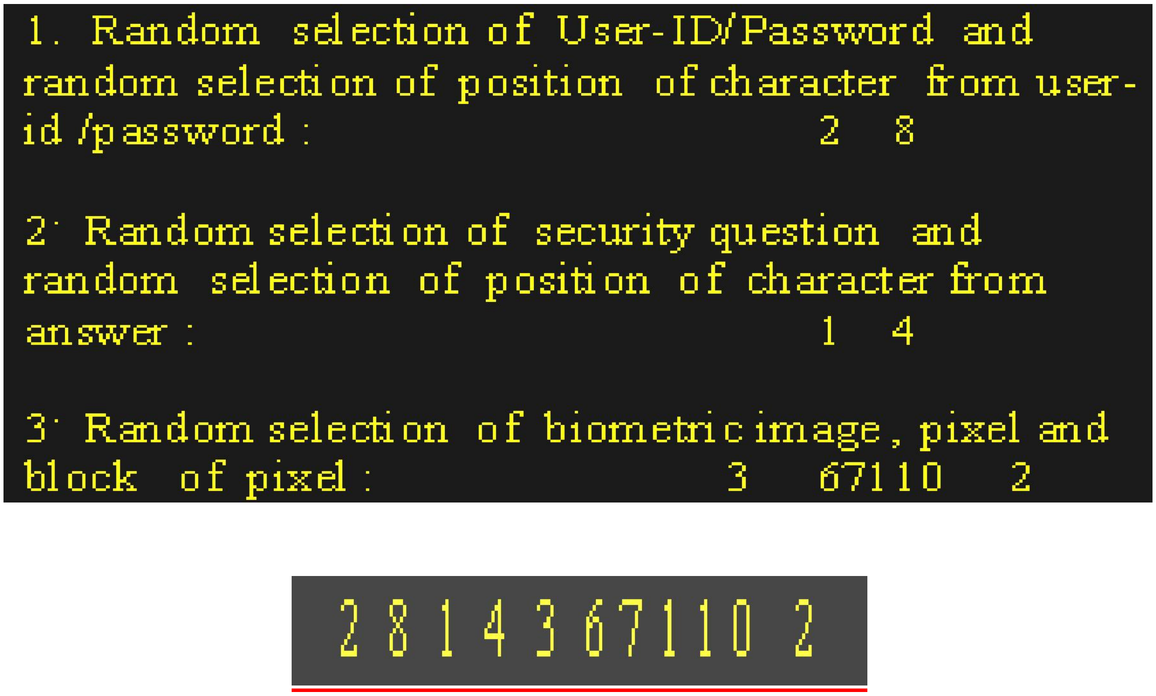
Inputted images are



Biolog1.jpg Biolog2.jpg Biolog3.jpg

Figure 4.16: Inputs of Biometric Image of User

Figure 4.17 represents the intermediate OTP generation by randomizing selections of positions of pixels/characters.

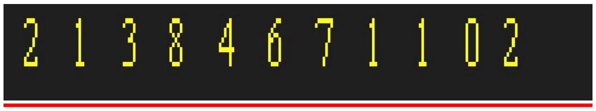


```
1. Random selection of User-ID/Password and
random selection of position of character from user-
id /password :                          2   8

2· Random selection of security question and
random selection of position of character from
answer :                                1   4

3· Random selection of biometric image, pixel and
block of pixel :                    3   67110   2
```

```
2 8 1 4 3 67110 2
```

Figure 4.17: Intermediate OTP Generation

Figure 4.18 shows final OTP generation by repositioning the digits of Intermediate OTP

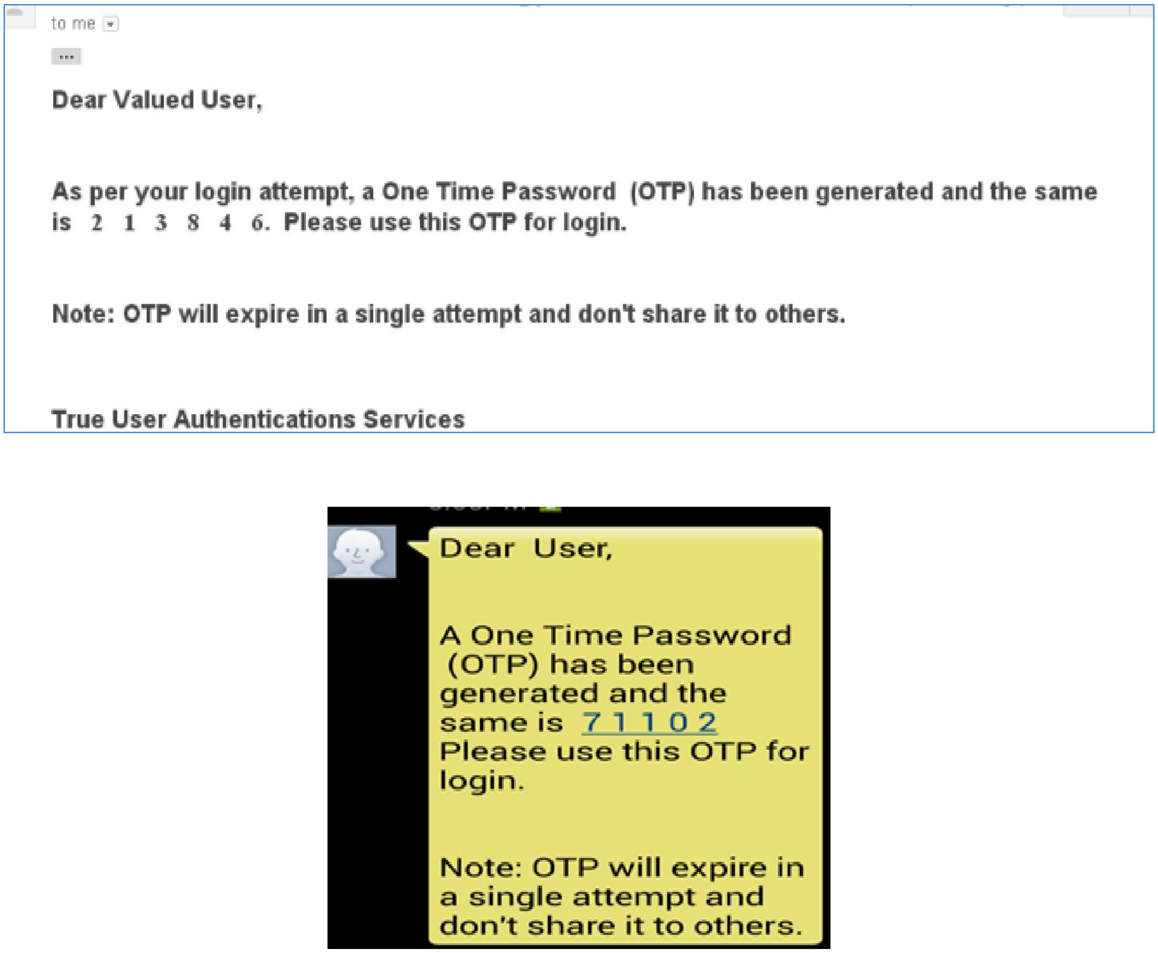


```
2 1 3 8 4 6 7 1 1 0 2
```

Figure 4.18: Final OTP Generation

C. Final OTP Distribution

Final OTP is divided into two segments and shared by using email and text message. Figure 4.19 shows the entire distribution scheme.



Dear Valued User,

As per your login attempt, a One Time Password (OTP) has been generated and the same is 2 1 3 8 4 6. Please use this OTP for login.

Note: OTP will expire in a single attempt and don't share it to others.

True User Authentications Services

Dear User,

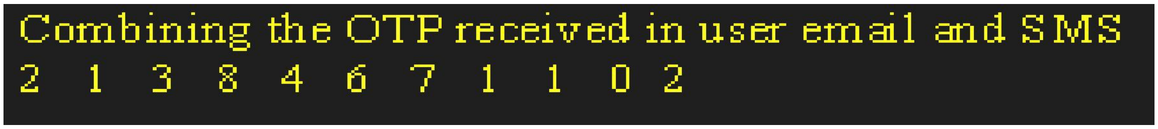
A One Time Password (OTP) has been generated and the same is 7 1 1 0 2 Please use this OTP for login.

Note: OTP will expire in a single attempt and don't share it to others.

Figure 4.19: Distribution of Final OTP through Email and Text Message

D. Final OTP Generation at User side

Figure 4.20 shows final OTP generation at user end by combining the OTPs from email and text message.

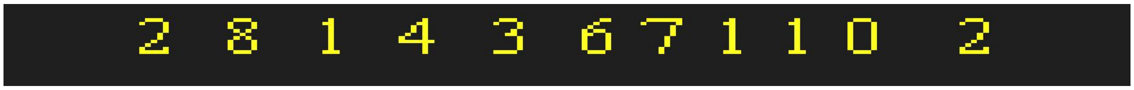


```
Combining the OTP received in user email and SMS
2  1  3  8  4  6  7  1  1  0 2
```

Figure 4.20: Generation of Final OTP at User End

E. Intermediate OTP Generation at User side

Figure 4.21 shows the generation of intermediate OTP by repositioning the digits of final OTP.

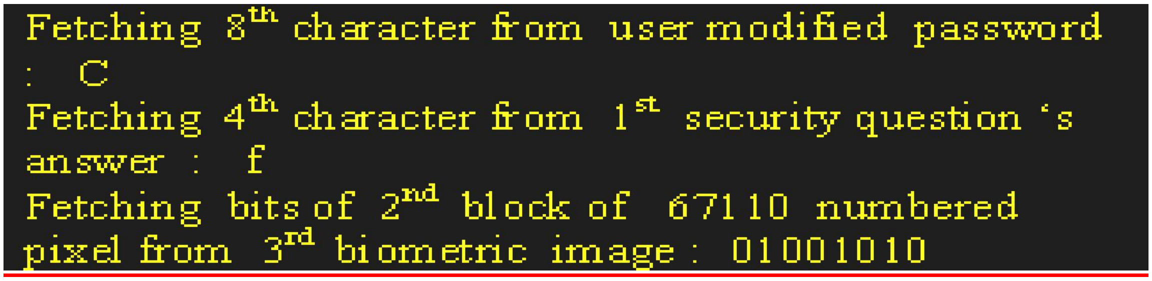


```
2  8  1  4  3  6 7 1 1 0  2
```

Figure 4.21: Generation of Intermediate OTP at User End

F. Formation of Secret Value at User and Server End for Authentication

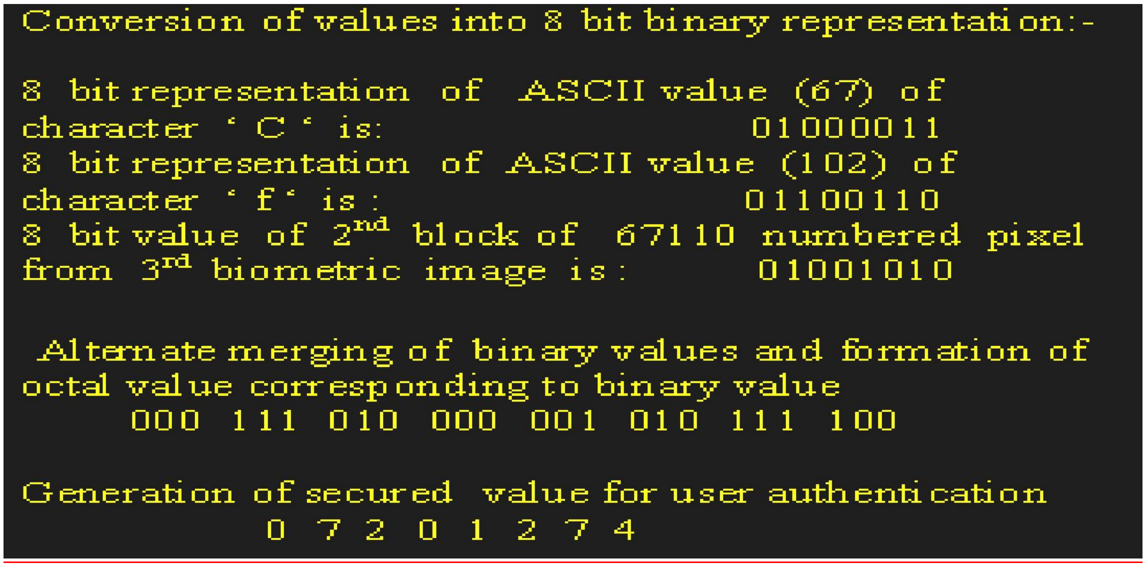
Figure 4.22 shows the characters that are being fetched from the modified password or user-id with the direction of intermediate OTP.



```
Fetching 8th character from user modified password
:  C
Fetching 4th character from 1st security question ‘s
answer :  f
Fetching bits of 2nd block of 67110 numbered
pixel from 3rd biometric image : 01001010
```

Figure 4.22: Value Extraction from Modified User-Id/Password defined by Intermediate OTP

Figure 4.23 shows the generation of secret value using alternate merging between the binary values of fetched characters and block of pixel.

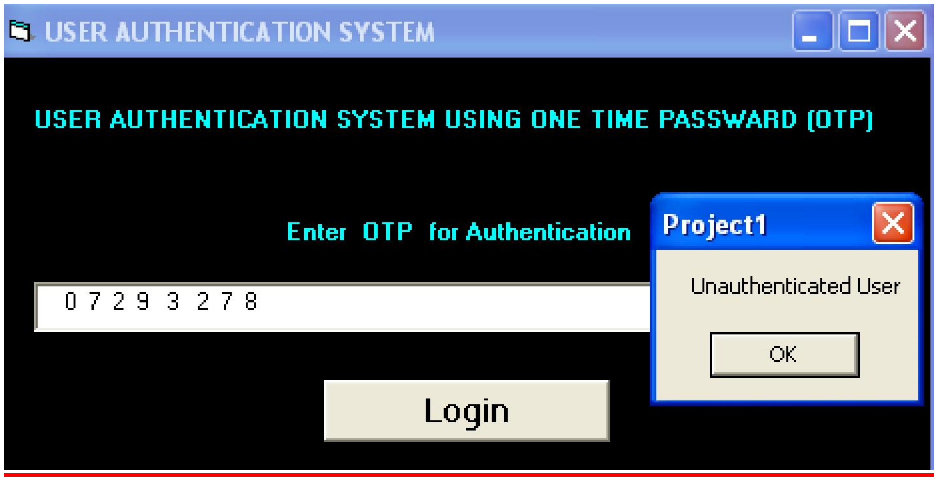


```
Conversion of values into 8 bit binary representation:-

8 bit representation of ASCII value (67) of
character ' C ' is:                        01000011
8 bit representation of ASCII value (102) of
character ' f ' is :                       01100110
8 bit value of 2nd block of 67110 numbered pixel
from 3rd biometric image is :              01001010

 Alternate merging of binary values and formation of
octal value corresponding to binary value
      000 111 010 000 001 010 111 100

Generation of secured  value for user authentication
              0  7  2  0  1  2  7  4
```

Figure 4.23: Generation of Secret Value used for Authentication

G. Validation of User Authentication

Figure 4.24 and figure 4.25 shows authentication attempts made by fake and valid user respectively.



USER AUTHENTICATION SYSTEM

USER AUTHENTICATION SYSTEM USING ONE TIME PASSWARD (OTP)

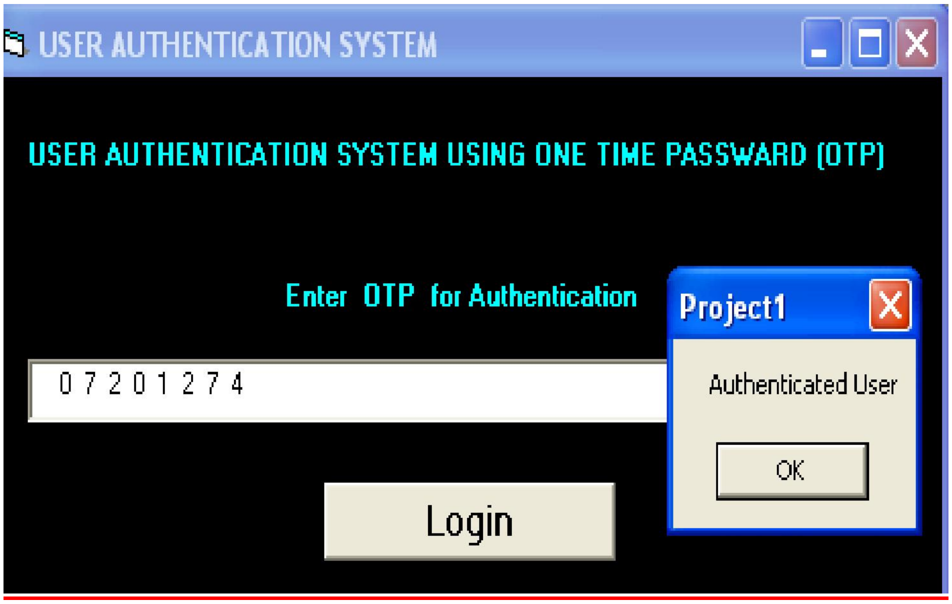
Enter OTP for Authentication

0 7 2 9 3 2 7 8

Project1

Unauthenticated User

Figure 4.24: Login Attempt by Fake User for SVROTP Scheme



USER AUTHENTICATION SYSTEM USING ONE TIME PASSWARD (OTP)

Enter OTP for Authentication

07201274

Authenticated User

Figure 4.25: Login Attempt by Valid User for SVROTP Scheme

4.3.5. Security Analysis for SVROTP Scheme

A. Computation Time

OTP generation algorithm takes 67315 milliseconds and 7123 milliseconds is needed for validating user authentication at server end where the server has Intel Core i5 3.6 GHz processor and 4.00 GB RAM. Distribution of OTP is carried out through public communication mode so that the distribution time is affected by network congestion, noise, traffic and other obstacles present in the public network. Thus only the OTP generation and authentication time are considered.

Table 4.3 represents the security analysis of SVROTP scheme.

Table 4.3: Security Analysis of SVROTP Scheme

Activities	**Size of the parameters on which the security is measured**	**Amount of executions needed for all possible combinations of parameters for OTPs formation**
Generation of Intermediate OTP	User id and password each of 8 characters. Security questions - 3 numbers. Answer of each question has 5 characters. Biometric image-3 numbers. Size of each image is 108*98 pixels (108*98*32 bits).	{ Factorial (8) * [Factorial (8) / (Factorial (1) * Factorial (8-1))] + 1 } * { [Factorial(5) / (Factorial (1) * Factorial (5-1))] + 1} * { [Factorial (108*98*32) / (Factorial (8) * Factorial (108*98*32-8))] + 1 }
Generation of Final OTP	Intermediate OTP is of 11 characters.	Factorial(11)
Generation of secret value used for authentication	User id and password each of 8 characters. Security questions - 3 numbers. Answer of each question has 5 characters. Biometric image-3 numbers. Size of each image is 108*98 pixels (108*98*32 bits).	{ Factorial (8) * [Factorial (8) / (Factorial (1) * Factorial (8-1))] + 1} * { [Factorial (5) / (Factorial (1) * Factorial (5-1))] + 1 }* { [Factorial (108*98*32) / (Factorial (8) * Factorial (108*98*32-8))] + 1}

So total amount of executions needed for all possible combinations of parameters for OTPs formation is T (where T= (Factorial(11) + 2 * [{Factorial(8) * [Factorial(8) / (Factorial(1) * Factorial (8-1))]+1} * {[Factorial(5) /(Factorial (1) * Factorial (5-1))]+1} * {[Factorial (108*98*32) / (Factorial (8) * Factorial (108*98*32-8))]+1}])). As T is extremely higher so these amounts of executions need extreme amount of time still the system is secured as user biometric images and private information are not open to others.

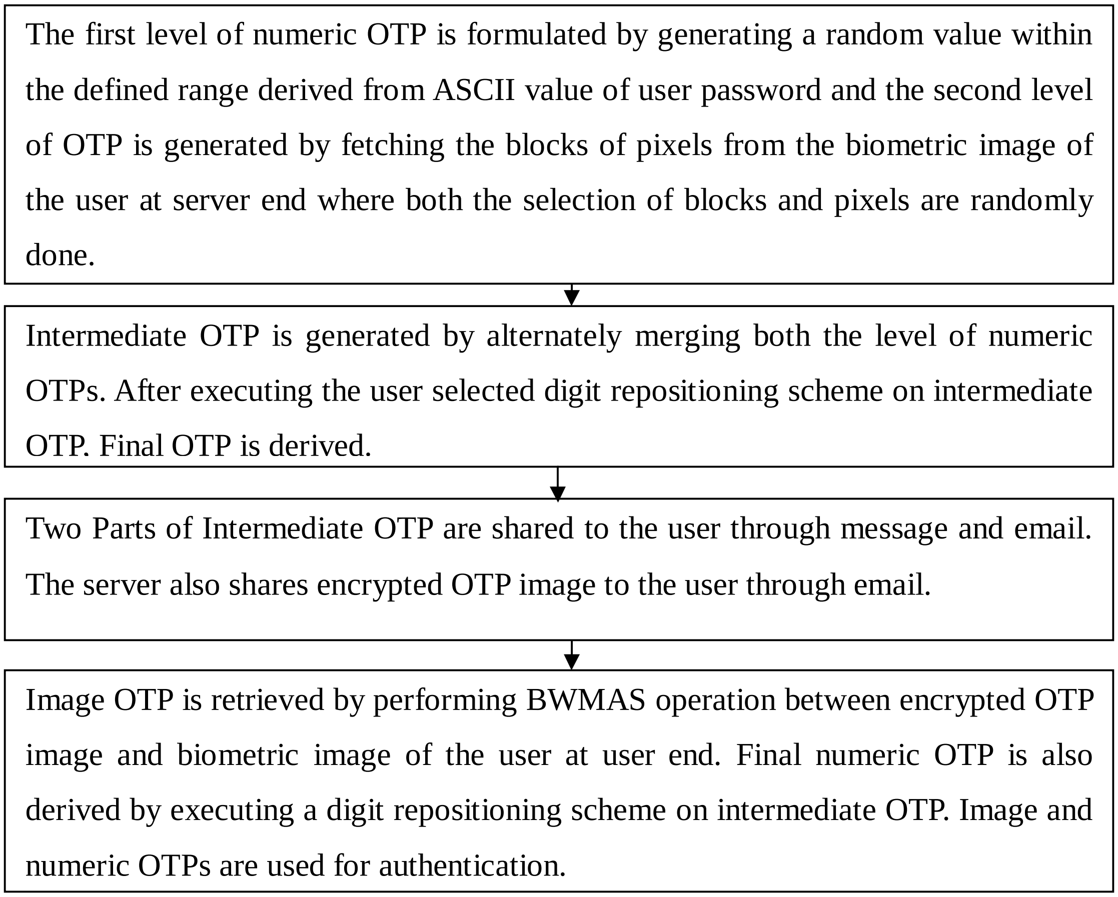
4.4. User authentication scheme using Numeric and biometric Image based One Time Password (NIOTP)

In NIOTP[3] authentication scheme, a combined OTP is introduced. Both Image OTP and final numeric OTP are applied for user authentication. An image is randomly selected as image OTP by the server. Image OTP is encrypted using user biometric image and BWMAS (Bitwise Masking for Alternate Sequence) operation and generates encrypted image OTP which is shared to the user. A large random number is generated for 1st level numeric OTP within a range derived from the user password. The second level of numeric OTP is derived from the blocks of pixels of user biometric image where the selection of blocks and pixels are randomly carried out. Both the OTPs are combined using alternate merging and intermediate numeric OTP is constructed and shared to the user. After executing the user selected digit repositioning scheme on intermediate OTP, Final OTP is derived.

Random selection of image and numeric OTP, distribution of intermediate numeric OTP and encrypted image OTP in different communication modes and generation of final OTP using digit repositioning scheme ensure more protection over existing OTP schemes. Figure 4.26 represents the entire scheme for NIOTP based authentication system.

Password, username and unique biometric image are supplied to the authentication server as input by the user for registration. Image OTP is generated by random selection of image at server end. Besides this, image OTP is also encrypted using a unique biometric image of user with the help of BWMAS (Bitwise Masking for Alternate Sequence) operation at server side.

[3] Published in **Lecture Notes in Electrical Engineering, Springer**, Volume 470, pp. 83 – 93, with title Secure User Authentication System using Image based OTP and Randomize Numeric OTP based on User Unique Biometric Image and Digit Repositioning Scheme.



The first level of numeric OTP is formulated by generating a random value within the defined range derived from ASCII value of user password and the second level of OTP is generated by fetching the blocks of pixels from the biometric image of the user at server end where both the selection of blocks and pixels are randomly done.

Intermediate OTP is generated by alternately merging both the level of numeric OTPs. After executing the user selected digit repositioning scheme on intermediate OTP. Final OTP is derived.

Two Parts of Intermediate OTP are shared to the user through message and email. The server also shares encrypted OTP image to the user through email.

Image OTP is retrieved by performing BWMAS operation between encrypted OTP image and biometric image of the user at user end. Final numeric OTP is also derived by executing a digit repositioning scheme on intermediate OTP. Image and numeric OTPs are used for authentication.

Figure 4.26: Overall Procedure for NIOTP based Authentication Scheme

Section 4.4.1 represents server side OTP generation. Section 4.4.2 discusses the distribution of numeric and image OTPs. Section 4.4.3 describes the extraction of OTP at the user end and authentication. Experimental results are described in section 4.4.4. Section 4.4.5 shows the security analysis of the implemented OTP scheme.

4.4.1. Server Side OTP Generation

A. Algorithm for BWMAS Operation (Bit Wise Masking for Alternate Sequence)

Detailed description is given in section 8.3.1 of chapter 8 (BWMAS Operation (Bit Wise Masking for Alternate Sequence)).

B. Image OTP Generation and Encryption Algorithm for NIOTP Scheme

Step I: An image is randomly selected by the server as image OTP. The server calculates the height (HO) and width (WO) of the image OTP. Height (HI) and width (WI) are also calculated for the biometric image of a user where WI<=WO and HI<=HO. RGB part of 24 bits of user biometric image and image OTP are stored into two arrays namely RGB_IMG[] and KEY_IMG[] respectively where the size of each array is WO*HO*24.

Step II: Server executes BWMAS operation in between RGB_IMG[] and KEY_IMG[] array and the operational result is stored in RE[] array from where the encrypted image OTP is generated. Server shares encrypted image OTP to the user.

C. Numeric OTP Generation Algorithm

Step I: Range value for random number generation is computed by multiplying the ASCII values of each character present in the user password. The first level of numeric OTP is a long random value generated within 0 to range value. Four pixels and four blocks are randomly chosen from the biometric image of the user. Then the bit values of those randomly selected blocks of pixels are retrieved and bit values are converted into the decimal base. This decimal value is considered as the second level of numeric OTP.

Step II: Alternate merging operation is performed between 1st and 2nd level of numeric OTP and intermediate numeric OTP is generated. Final OTP is derived from intermediate OTP by executing a user selected digit repositioning scheme. Intermediate OTP is shared to the user where final OTP is used for user authentication.

D. Algorithm for Digit Repositioning Schemes

The user selects one of the digit repositioning algorithms for repositioning the digits of intermediate OTP. Digit repositioning algorithms are described below.

PRPNNP (Positional Reverse Prime Normal Non-Prime)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

PRNPNP (Positional Reverse Non-Prime Normal Prime)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

CRPNNP (Continuous Reverse Prime Normal Non-Prime)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

CRNPNP (Continuous Reverse Non-Prime Normal Prime)

Detailed description is given in section 8.3.2 of chapter 8 (Cipher & Pixel Block Sequencing Methodologies)

E. Main () Function Algorithm

Step I: Password, username, selection of digit repositioning scheme and biometric image of the user are supplied as input to the server from the user.

Step II: Call Image OTP Generation and Encryption Algorithm

Step III: Call Numeric OTP Generation Algorithm

4.4.2. Numeric OTP and Image OTP Distribution

Two parts of intermediate numeric OTP and encrypted image OTP are shared from server to user through email and text message. Final numeric OTP is derived from intermediate OTP in both server and user end separately.

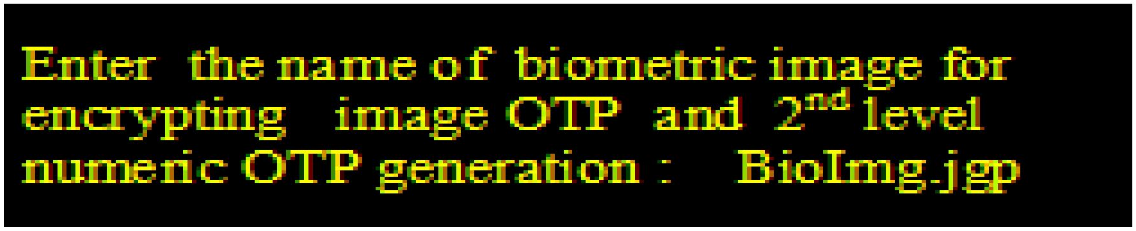
4.4.3. User End OTP Extraction and Authentication

Original image OTP is extracted by performing BWMAS operation between encrypted image OTP and user biometric image. Final numeric OTP is derived from intermediate OTP by executing a user selected digit repositioning scheme. Both the image OTP and final numeric OTP are used for authentication.

4.4.4. Implementation with Experiment Results

A. Inputs for User Registration in Authentication Server

Password, user-id, selection of digit repositioning scheme and biometric image of the user are provided to the server as input from the user. Figure 4.27 and figure 4.28 represent the activities.



```
Enter  the name of  biometric image  for
encrypting   image OTP  and  2nd level
numeric OTP generation :    BioImg.jgp
```

Figure 4.27: Receiving of Biometric Image Input

Figure 4.28: Biometric Image of User (BioImg.jpg)

B. Server Side Image based OTP Generation

The server selects an image (N.jpg here) randomly and considers it as image OTP. Image OTP is encrypted by the biometric image of the user and also shared to the user. Figure 4.29 represent the activities.

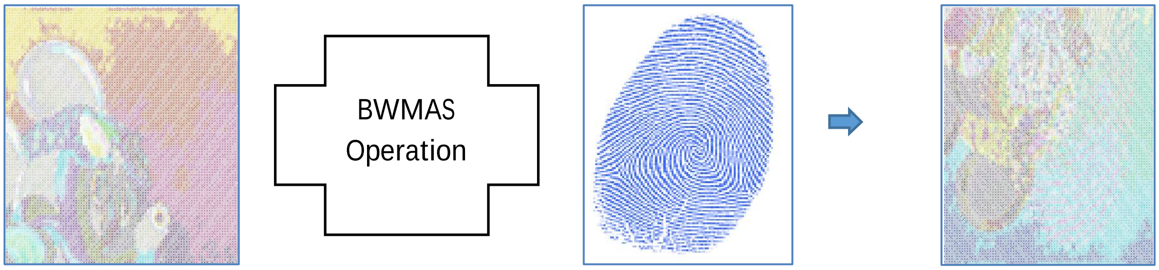


Image OTP (N.jpg)

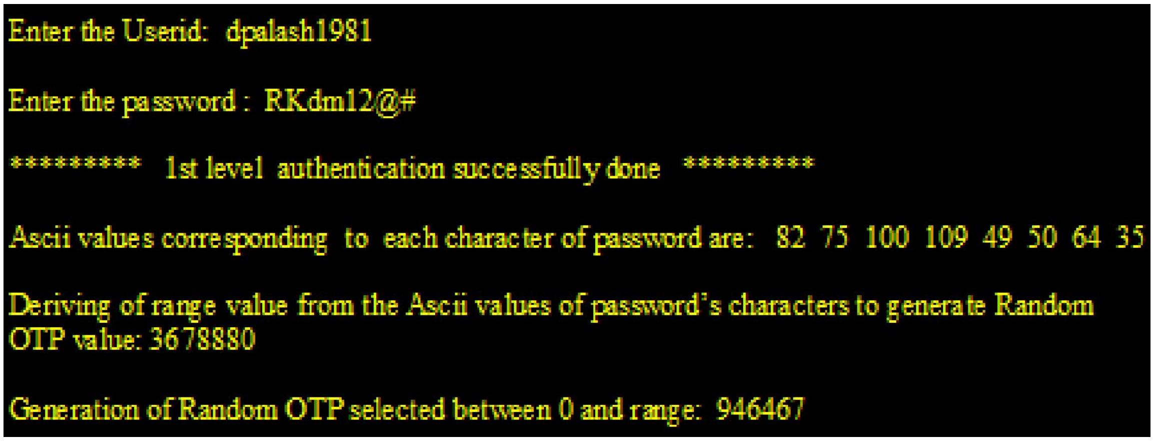
BioImg.jpg

Encrypted image OTP (N_EN.jpg)

Figure 4.29: Generation of Encrypted Image based OTP at Server End

C. Generation of First Level of Numeric OTP

Figure 4.30 represents the entire activities relating to first level numeric OTP generation.

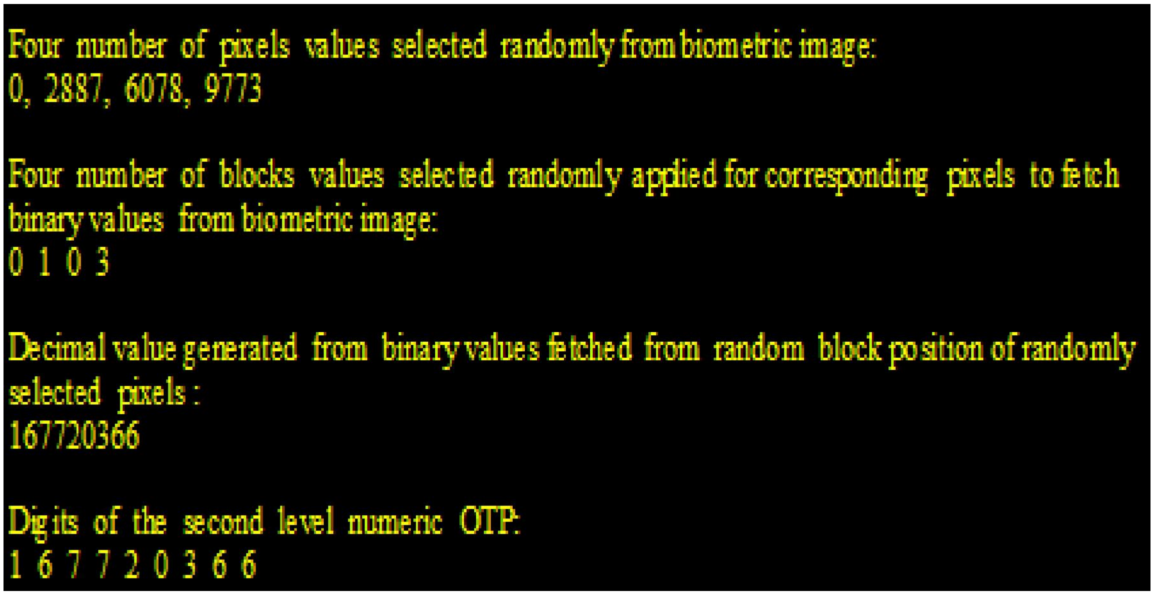


```
Enter the Userid:  dpalash1981

Enter the password :  RKdm12@#

*********   1st level  authentication successfully done   *********

Ascii values corresponding  to  each character of password are:   82  75  100  109  49  50  64  35

Deriving of range value from the Ascii values of password's characters to generate Random
OTP value: 3678880

Generation of Random OTP selected between 0 and range:  946467
```

Figure 4.30: First Level of Numeric OTP Generation

D. Generation of Second Level of Numeric OTP based on Biometric Image

Figure 4.31 represents the entire activities relating to second level numeric OTP generation.

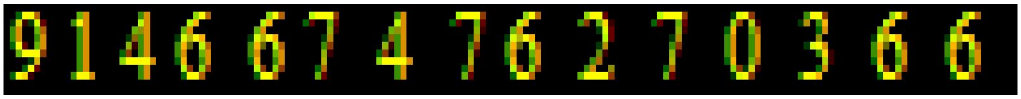


```
Four  number  of  pixels  values  selected  randomly from biometric image:
0, 2887, 6078, 9773

Four  number  of  blocks  values  selected  randomly applied for corresponding  pixels  to fetch
binary values  from biometric image:
0  1  0  3

Decimal value generated  from  binary values fetched  from  random  block position of randomly
selected  pixels :
167720366

Digits  of  the  second  level  numeric  OTP:
1  6  7  7  2  0  3  6  6
```

Figure 4.31: Second Level of Numeric OTP Generation

E. Generation of Intermediate Numeric OTP

Intermediate OTP is generated by performing an alternate merging operation between two levels of OTPs. Figure 4.32 represents the intermediate OTP.

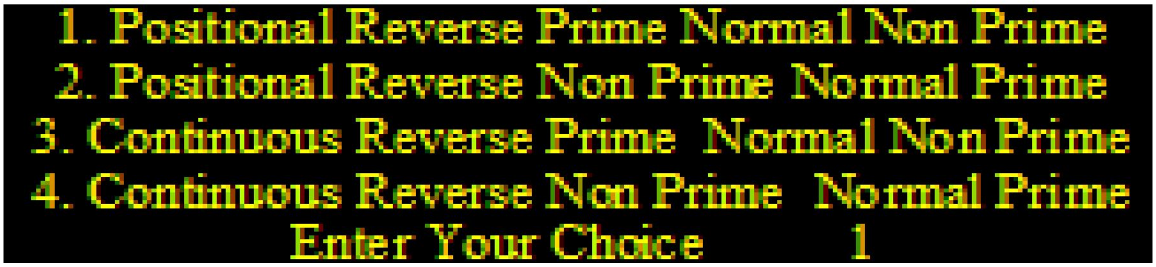


```
9 1 4 6 6 7 4 7 6 2 7 0 3 6 6
```

Figure 4.32: Value of Intermediate Numeric OTP

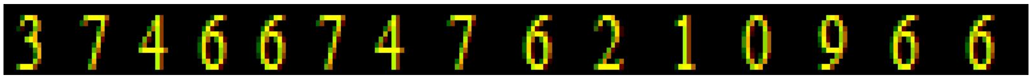
F. Final OTP Generation

User selected OTP repositioning algorithm is applied on intermediate OTP and final numeric OTP is generated at the server end. Figure 4.33 and figure 4.34 represents the selection of a digit repositioning scheme and final numeric OTP respectively.



```
1. Positional Reverse Prime Normal Non Prime
2. Positional Reverse Non Prime Normal Prime
3. Continuous Reverse Prime  Normal Non Prime
4. Continuous Reverse Non Prime  Normal Prime
Enter Your Choice          1
```

Figure 4.33: Selection of Digit Repositioning Scheme for Final Numeric OTP Generation

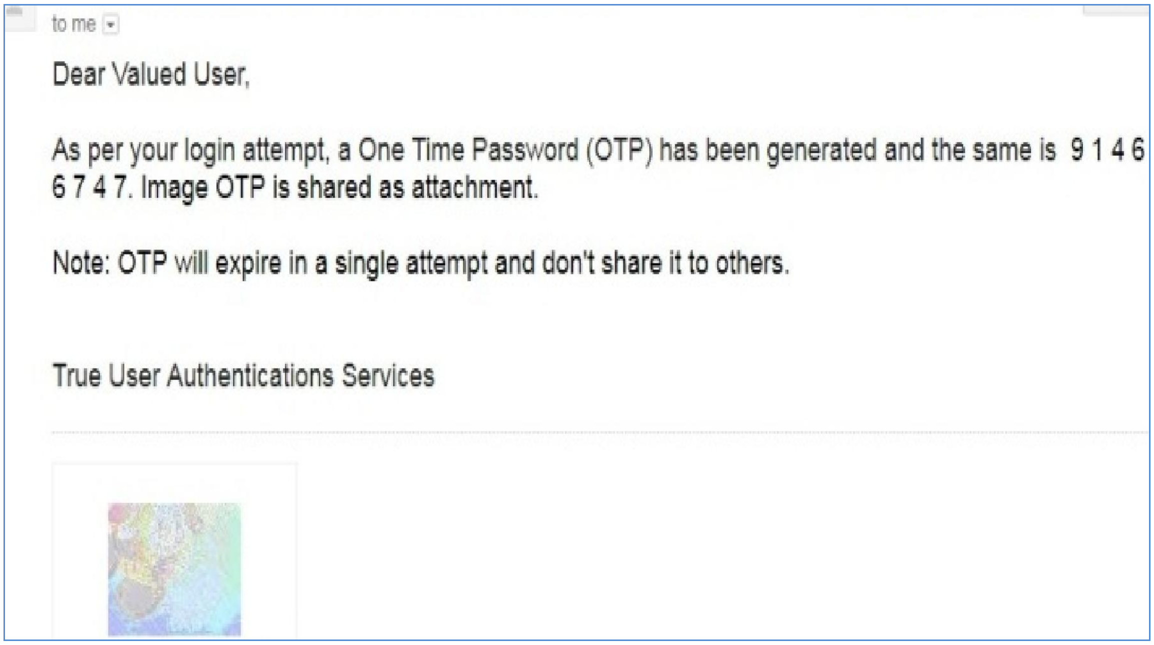


```
3 7 4 6 6 7 4 7 6 2 1 0 9 6 6
```

Figure 4.34: Value of Final Numeric OTP

G. Intermediate Numeric OTP and Encrypted Image OTP Distribution

Intermediate numeric OTP is divided into two parts and they are shared through email and SMS along with encrypted image OTP. Figure 4.35 represents the entire activities.

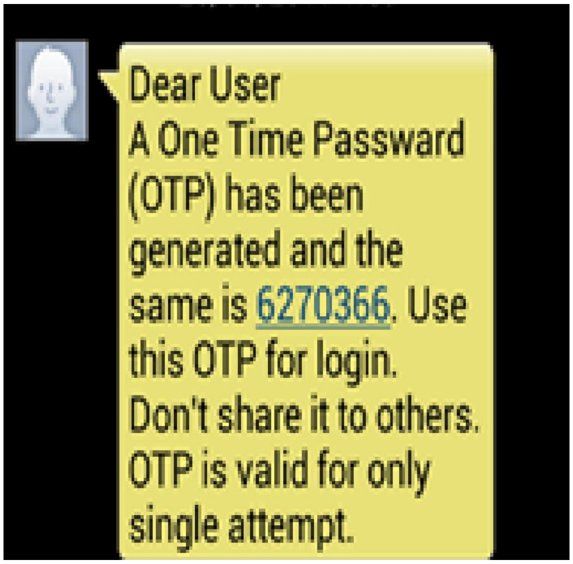


Dear Valued User,

As per your login attempt, a One Time Password (OTP) has been generated and the same is 9 1 4 6 6 7 4 7. Image OTP is shared as attachment.

Note: OTP will expire in a single attempt and don't share it to others.

True User Authentications Services

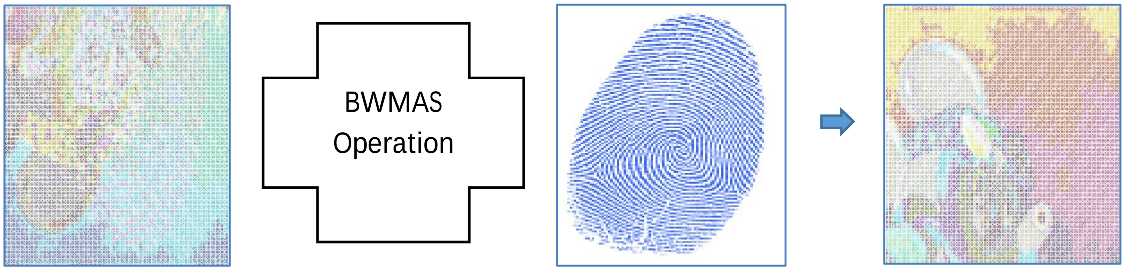


Dear User
A One Time Passward (OTP) has been generated and the same is 6270366. Use this OTP for login. Don't share it to others. OTP is valid for only single attempt.

Figure 4.35: Intermediate Numeric OTP and Encrypted Image OTP Distribution

H. Final Numeric OTP and Image based OTP extraction at User End

Figure 4.36 and figure 4.37 represent the image based OTP and final numeric OTP extraction at the user end.

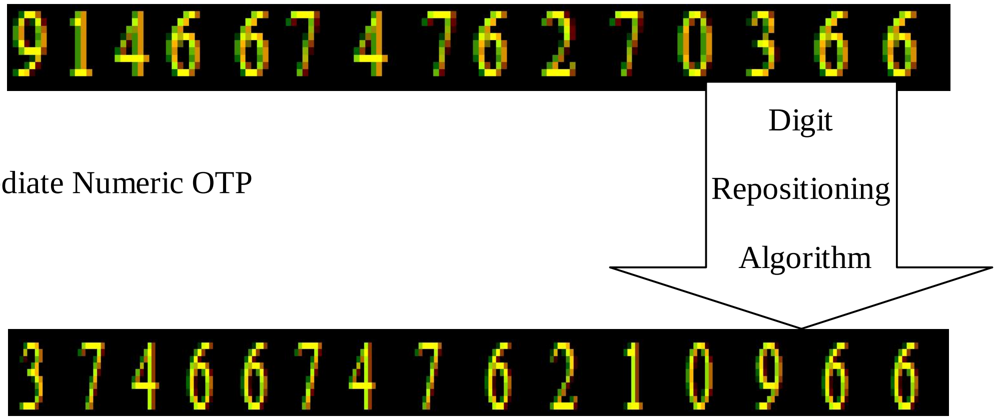


Encrypted image OTP
(N_EN.jpg)

BioImg.jpg

Image OTP
(N.jpg)

Figure 4.36: Image based OTP extraction at User Side



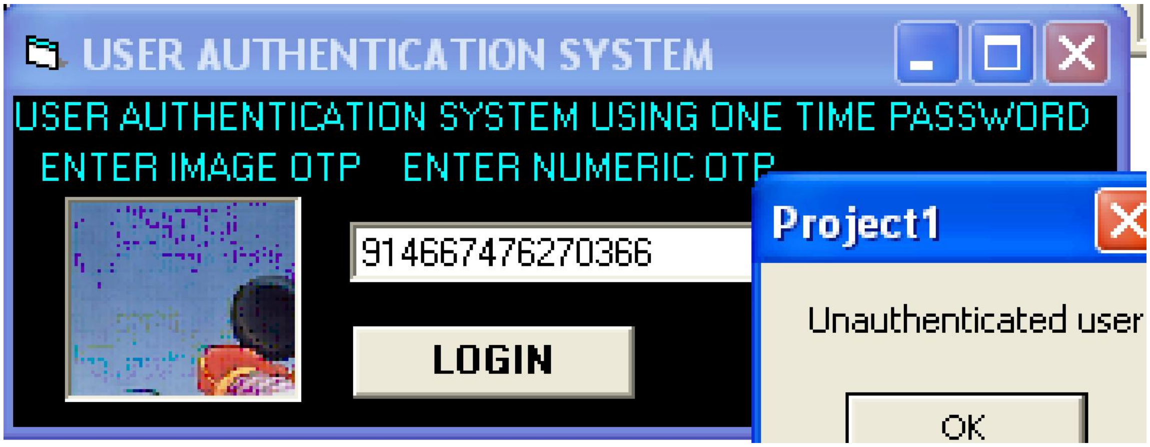
Intermediate Numeric OTP

Final Numeric OTP

Figure 4.37: Final Numeric OTP extraction at User Side

I. User Authentication

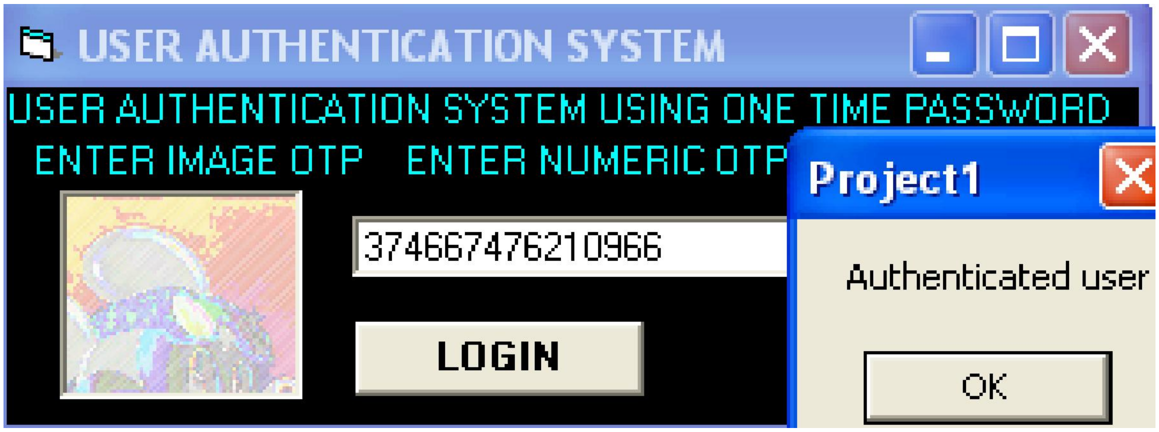
Figure 4.38 and figure 4.39 represent the login attempts by fake and valid user respectively.


USER AUTHENTICATION SYSTEM
USER AUTHENTICATION SYSTEM USING ONE TIME PASSWORD
ENTER IMAGE OTP
ENTER NUMERIC OTP
LOGIN
Project1
OK

914667476270366

Unauthenticated user

Figure 4.38: User Authentication for Fake Login Attempts for NIOTP Scheme


USER AUTHENTICATION SYSTEM
USER AUTHENTICATION SYSTEM USING ONE TIME PASSWORD
ENTER IMAGE OTP
ENTER NUMERIC OTP
LOGIN
Project1
OK

374667476210966

Authenticated user

Figure 4.39: User Authentication for valid Login Attempts for NIOTP Scheme

4.4.5. Security Analysis for NIOTP Scheme

A. Computation Time

Algorithm for OTP generation takes 71894 milliseconds and validation of user authentication method takes 8453 milliseconds at server end where the server is running with Intel Core i5 3.6 GHz processor and 4.00 GB RAM. As OTP distribution time is affected by network congestion, noise, traffic and other obstacles present in the public network so only the OTP generation and authentication time are considered rather than considering the distribution time for the current OTP scheme.

Table 4.4 shows the results of the security analysis of NIOTP scheme

Table 4.4: Results of Security Analysis of NIOTP Scheme

For 1st level numeric OTP generation		**For 2nd level numeric OTP generation**	
Range value for 1st level numeric OTP (product of ASCII values of characters from user password)	**Number of attempts needed for generating all possible random value with the range**	**Size of encrypted image OTP in bits (w(width) *h(height) *32)**	**Amount of executions necessary to generate all possible combinations of bits from user biometric image to produce 2nd level numeric OTP**
367888080000000	367888080000000	1000*2000*32	factorial(64000000) / factorial (32)* factorial (64000000-32)
410653900800000	410653900800000	1000*3000*32	factorial (96000000) / factorial (32)* factorial (96000000-32)
415820673515520	415820673515520	1000*4000*32	factorial (128000000) / factorial (32)* factorial (128000000-32)

After executing extreme amount of executions for 1st and 2nd level OTP generation, the access is still not been validated as hacker access the bit information of encrypted image OTP where intermediate numeric OTP is generated from the biometric image of the user. If the intermediate OTP is hacked still system is safe as final OTP is used for authentication which is derived from intermediate OTP by alternate merging and digit repositioning schemes. If the algorithm of BWMAS operation, digit repositioning scheme and intermediate OTP are compromised, still the authentication system is secured as user biometric image is hidden from hackers so image OTP can't be retrieved. So the security of the system is enhanced.

4.5. Measurement of Security over Cryptographic Attacks by SVROTP and NIOTP Schemes

Table 4.5 represents security measurements over cryptographic attacks by SVROTP and NIOTP schemes.

Table 4.5: Security Analysis for SVROTP and NIOTP Schemes over Cryptographic Attacks

Type of Attack	**Number of Login Attempts by Fake Users**	**Measurement of Security over Unauthorized Access (in percentage)**		**Remarks**
		SVROTP Scheme	**NIOTP Scheme**	
Brute Force Attack	75	86%	89%	As authentication is done using secret value extracted from Private Information and Unique Biometric Images of user directed by numeric and image based OTP rather than using the actual shared OTP value so the

				probability of random guessing and successfully using of OTP is extremely less for the implemented OTP schemes.
Dictionary Attack	68	89%	91%	Random selections of pixels, characters from randomly selected text or biometric image, extraction of secret value from private information directed by intermediate OTP and use it for authentication in state of using direct OTP values blocks dictionary attack as random selection of values extremely reduces the probability of unauthorized access attempts by cycling the combination of words generated from dictionary attacks.
Key Logger Attack	70	88%	86%	As per Key Logger Attack, if all the keystrokes are tracked and final OTP is hacked still the system is secured as intermediate OTP which generates the secret value for authentication can’t be retrieved from the final OTP without the availability of user unique biometric image, knowledge of retrieving procedure, login-id, password and security questions asked to user at the time of authentication. Thus the implemented OTP system blocks Key Logger Attack

Man in the Middle Attack	64	82%	87%	Multiple layers of securities are imposed in implemented systems. Login id and password based authentication, distribution of final OTPs in multiple communication channels, using of separate OTPs for distribution (final OTP) and authentication (intermediate OTP) extraction and use of secret values for authentication directed by intermediate OTP instate of using direct OTP value are implemented in the proposed systems. Multiple numbers of hacking are needed for hacking all values. Thus the systems are secured as user biometric image is unique and only been available to the user. So the proposed system is safe and reduces the probability of Man in the Middle attack.
Phishing Attacks, Physical Access to Phone, Mobile Phone Trojans	65	84%	90%	This kind of attack may only access login id, password, Final OTP. But the generation of secret values which is used for authentication cannot be possible without the knowledge of intermediate OTP, user unique biometric image and answers of the security questions. Thus the implemented OTP system blocks this kind of attacks.

Figure 4.40 graphically represents the percentage of security measurement over different cryptographic attacks by SVROTP and NIOTP schemes.

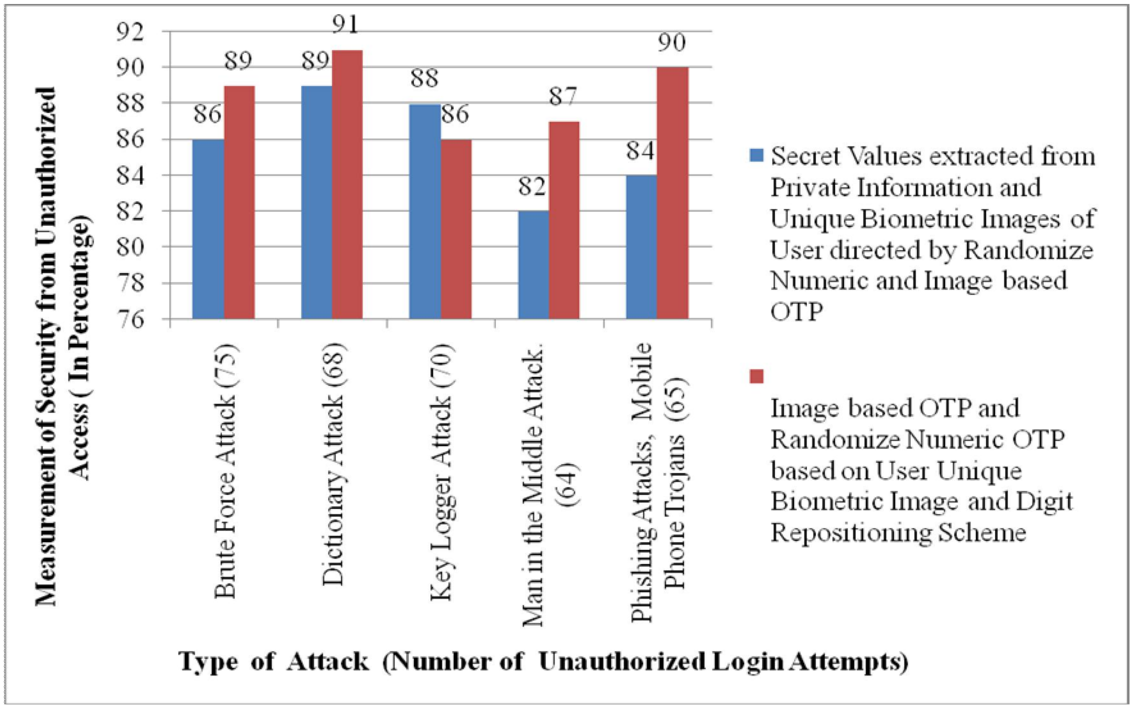


Figure 4.40: Security Measurement over Cryptographic Attacks by SVROTP and NIOTP Schemes

4.6. Conclusion

Authentication is based on the selecting of representative images of user likings as per the sequence of previously selected options for CAPTCHA based authentication scheme. Thus security is enhanced. The user may change the option selection and also the sequences of selecting options if their information is hacked. Thus providing better security compared to existing CAPTCHA schemes.

Multiple layers of securities are present in OTP schemes. User id and password based authentication, random selection of items, distribution of OTPs in multiple modes, the formation of separate OTPs for distribution and authentication, extraction and use of secret value for authentication rather than using OTP values provide great security.

Distribution of OTPs into parts through different communicational channels (email and SMS) increases security level as multiple numbers of hacking is needed to access the entire OTP.

Implemented OTP schemes extract and use secured values for authentication from user private information (biometric image, user id, password and security questions) defined by OTP. So if the OTPs are being hacked still the systems are secured due to the unavailability of user private information. Thus it enhances the security to great extent.

Implemented OTP schemes provide great securities over different attacks compared to existing scheme. So it may be concluded that the newly implemented CAPTCHA and OTP based user authentication schemes may be expected to provide a very satisfactory level of information security.