NOVATEUR PUBLICATIONS
JournalNX- A Multidisciplinary Peer Reviewed Journal
ISSN No: 2581 - 4230
VOLUME 7, ISSUE 5, May. -2021

MULTI-FACTOR AUTHENTICATION SYSTEM BASED ON TEMPLATE

O'rinov Nodirbek Toxirjonovich,
Teacher, Department of Information Technology, Andijan State University
E-mail:nodirbekurinov1@gmail.com

Arabbayev Arobidin Xusnidinovich
Teacher of the Department of Informatics Teaching Methods
E-mail: arobijon@mail.ru

Alisher Usmanov Yusupjan o'g'li
Teacher of the Department of Informatics Teaching Methods
E-mail: alisherusmonov_91@mail.ru

ABSTRACT:

User authentication is an integral part of a secure system. Traditional authentication methods have been proven to be vulnerable to various types of security attacks. Artificial intelligence is used to crack plain text passwords, and even CAPTCHAs are removed in a few tries. Using a picture password as an alternative to plain text passwords to authenticate users can be an effective strategy. However, they have been proven to be susceptible to shoulder attacks when surfing. Advanced authentication systems such as biometrics are secure, but they require additional infrastructure to be implemented effectively. This document proposes a new template-based multi-factor authentication scheme that uses a combination of text and images to identify legitimate users. The proposed system was mathematically analyzed and found to provide much more space for passwords than plain text passwords. This makes the proposed system safe from brute force and other dictionary attacks. Plus, using text along with images also reduces the risk of shoulder surfing.

Keywords: security, password, user authentication, template-based multi-factor AMS.

INTRODUCTION:

Recent advances in technology have led to the development of sophisticated IT-based systems to provide value-added services to users. These systems may store users' personal data in order to provide users with personalized services. The rapid growth in demand for personalized services will eventually transform these systems into repositories for various types of users' personal information. This requires more reliable and secure access mechanisms to reduce various security risks associated with unauthorized access to these IT systems [1, 2, 3]. This makes user authentication the most important and indispensable component of such systems. There are three main authentication methods based on token, biometric data and knowledge [4]. Smart cards are a token- based authentication system that implements knowledge-based methods to enhance security, as is the case with PIN-based ATM cards. Biometric authentication methods such as fingerprints, iris scans or facial recognition provide the highest level of security, but are still expensive, slow, unreliable and therefore not yet widely adopted [5]. Knowledge-based methods are the most widely used authentication methods, which include both text and picture passwords [6].

The simplest user authentication mechanism is the use of passwords [7]. The password concept is an efficient and cost effective solution for user authentication. The main requirement for any password is that it must be easy to remember and reasonably secure. In other words, the authentication process must be efficient and the password difficult to guess. A plain text password is still the most widely used form of authentication methods due to a number of factors, such as ease of remembering, difficulty in guessing, and the short time it takes to complete the process. Research has shown that users often choose a short password so they can be easily remembered, but unfortunately, these passwords are easy to crack. This can be explained by the fact that plain text passwords are simply a series of characters (numeric, alphanumeric, and special characters) and are usually based on Latin or other well-known scripts supported by input devices. This makes plain text passwords vulnerable to various security attacks. It was observed that 4.66% of accounts on rockyou.com were hacked through social engineering, dictionary attacks and brute force [8]. According to the Open Security Foundation, millions of credit card records have been compromised by hackers from large organizations such as TRW, Sears Roebuck, Sony Corporation, etc. [9]. According to an article in the news of the computer world, a security group of a large company launched a network password cracker and within 30 seconds identified about 80% of passwords [10].

In light of the above requirements, this article proposes a new pattern-based multi-factor recognition engine for user authentication. The proposed mechanism requires the user to enter a text key along with clicks in certain areas on multiple graphics. The combination of text and graphics increases the space for the password, thereby making the authentication mechanism more secure and more secure against various types of security threats. This article further analyzes the storage requirements and robustness of the proposed password scheme to various possible combinations of images and text in the proposed mechanism.

The rest of the document is organized as follows: Section 2 presents highlights of the existing literature relevant to the proposed work; In Section 3 we discuss the proposed system; Section 4 presents a mathematical analysis of the proposed system; and finally Section 5 concludes the article along with a brief discussion of future work.

LITERATURE REVIEW:

It is well known that humans can remember images better than text, making graphical password schemes a better alternative to text-based schemes [11]. Moreover, if the number of images is large enough, the possible password space of the graphic password scheme exceeds the space of the text schemes, thereby providing better resistance to dictionary attacks. In pattern-based recognition methods, the user is presented with a set of images and is authenticated by recognizing the images he or she selected during the registration step, but in recall-based methods, the user is asked to reproduce something he or she created. or selected earlier at the registration stage. Password retention was measured longitudinally three times: at the end of the first session (R1), after a week (R2) and after four weeks (R3), which revealed the following statistical data presented in Table 2.1 [12].

Table 2.1. Reaction to text and graphic password scheme [12]

	Mode	Average R1	Average R2	Average R3
Misfeed No.	Alphanumeric	1.61	2.82	1.43
	Graphic	0.28	2.44	1.20
Correct feed time (sec)	Alphanumeric	9.01	22.53	20.76
	Graphic	5.28	9.87	8.99

Table 2.1 clearly shows that picture passwords are easy and efficient to implement, and therefore the probability of missending and the time it takes to send correctly is always less than that of alphanumeric passwords. Unlike picture passwords, alphanumeric passwords are more difficult to remember, but easier to implement for a secure system. Tables 2.2 and 2.3 list some important statistics related to plain text passwords.

Table 2.2 The most common text passwords [13]

10 most popular passwords	Number of users	Percentage of use
123456	1666 g.	0.38
Password	780	0.18
Welcome	436	0.1
Ninja	333	0.08
abc123	250	0.06
123456789	222	0.05
12345678	208	0.05
sunlight	205	0.05
princes	202	0.05
Qwerty	172	0.04

From Tables 2.2 and 2.3, we can conclude that a dictionary and brute-force attack can easily decode alphanumeric passwords. A picture password has been used to implement users of a personal handheld device as the password decoder and the user are prompted to enter the image password with several prompts [15] [7]. People have an exceptional ability to recognize images, so the Pass Face scheme was implemented with cognometric or search metrics systems that require the user to memorize a set of images both at the stage of creating a password and at the stage of authentication [16] [17]. Figure 2.1 shows the grid layout of the passages.

Table 2.3. Most popular text password structure [14]

Prevailing password	Number of users	Percentage of use
One to six characters	88164	19.91
One to eight characters	272885	61.63
More than eight characters	169888	38.37
Lowercase alpha only	146486	38.37
Uppercase alpha only	1778	0.4
Alpha only	148264	33.49
Numeric only	26077	5.89
First uppercase last character	1259	0.28
First capital last digit	17464	3.94

Fig. 2.1. Mesh Pass Faces [18]

Hollingworth et al. [19] showed that humans can retain accurate, detailed, visual memories of previously visited sites. They suggested that a user can more accurately memorize certain parts of the image as a password if they focus on what is shown in Figure 2.2.

Ideally, the hint is always useful to the legitimate user during authentication. The callback system requires users to remember certain places in the image, which is easier than just remembering. These systems can also be called locimetric systems because of their dependence on the determination of a specific location. Sobrado et al. [20] developed a picture password technique that solves the problem of

surfing on the shoulder when the user needs to identify their preselected walk-throughs among the objects. The user authenticates himself by clicking inside the convex hull formed by these objects 2.3.

Sabzevar and Stavrow [15] proposed a methodology in which users need to move the frame along with the objects until the passed object in that frame aligns with the other two passed objects. This process can be repeated to reduce the likelihood of accidental authentication, but this slows down the entire process. Jansen et al. [6] proposed a graphical password mechanism for mobile devices in which a user logs in by selecting a theme composed of thumbnails and then registers a sequence of images as a password. During authentication, the user must enter the registered images in the correct sequence. The main disadvantage of this method is that the number of thumbnails is limited to 30 and therefore the space for the password is small. Figure 2.4 shows an example of their second algorithm, in which the user is required to move the frame until the passed objects are aligned with the other two are passing objects. The main disadvantage of this is the slow authentication process.

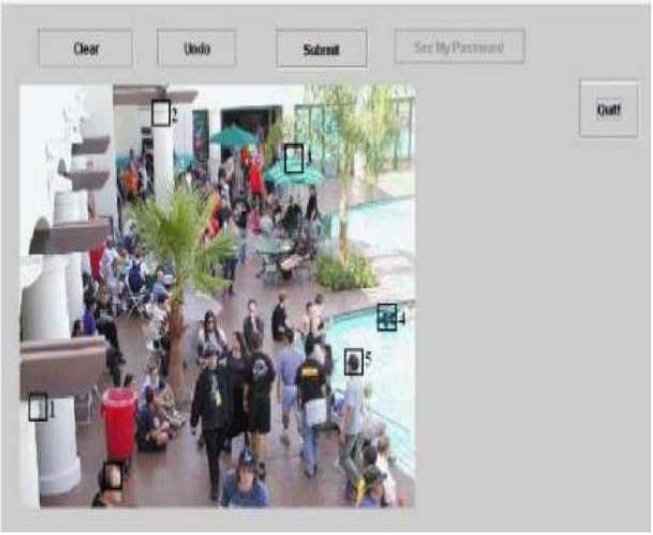

Fig. 2.2. Request revocation system [12]

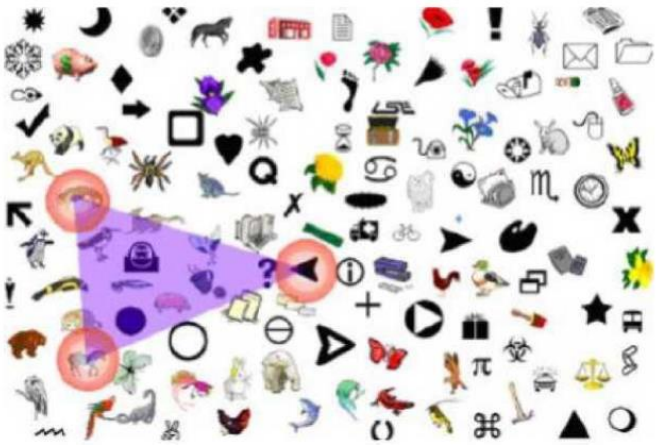

Fig. 2.3. Convex hull of preselected objects [20]

Jansen et. [6] proposed a graphical password mechanism for mobile devices. At the registration stage, the user selects a theme, which consists of thumbnails, and then registers the sequence of images as a password. During authentication, the user must enter the registered images in the correct sequence. One of the weaknesses of this technique is that the number of thumbnails is limited to 30, as shown in Figure 2.5. This results in less password space.

PROPOSED METHODOLOGY:

This article implements a template-based multi-factor authentication scheme in which a series of images are displayed for a fixed interval of time. The user must click in a predefined area on a specific image to authenticate. As the number of clicks increases, the security of the system also increases, but at the same time it becomes more complex, so the number of clicks required for authentication may vary according to the requirements of the system. The proposed system also requires the user to enter a key along with clicking in certain areas on the corresponding images in the slideshow in order to increase the level of security. Images viewed in the proposed system are 1360 x 660 pixels in size, each of which is displayed for 2.5 seconds during a

slideshow. The images selected for authentication are preferably grayscale images to reduce storage space and speed up the process. During authentication, the user must enter

Fig. 2.4. Align a walk-through to a line [20]

Fig. 2.5. Grid of 30 miniatures [6]

a key together with a click in a 30 x 30 pixel area around specific points in the respective images. If the region is too small, it will be more secure, but at the same time, the false rejection will be high for the real user. If this area is too large, it will be easier for attackers to guess the area that can be clicked. Picture passwords are easy to remember, however, they can also be associated with shoulder surfing, so the user has to enter some keystrokes along with a mouse click, which would be difficult for an attacker to notice. The length of time each image is displayed is another important factor. The longer the display time interval, the easier it is for the user to authenticate. But the authentication process will take a long time, and at the same time, the attacker will have more time to guess the password. The principle of operation of the proposed system is shown in Figure 3.1.

Studies of the Pass face technique have shown that people often choose weak and predictable picture passwords [21]. More research is needed to understand the nature of real-world picture passwords. The password scheme has a shorter password space than the system discussed in this article. Davis et al . implemented the Pass Face technique and discovered some obvious patterns among passwords [21]. For example, most users tend to pick the faces of people of the same race, which makes the Pass Face password predictable. This is not the case with this system, since different people select different click points on the same image containing many objects. With the exception of a few exceptions and mouse spyware, log spyware, or key eavesdropping, it is very difficult to crack picture passwords. Mouse movement related to the position, size and time of the window must be controlled to break the graphical password system.

Like plain text passwords, most picture passwords, including this system, are vulnerable to shoulder surfing [21]. However, adding enter keys to this system can help prevent this. Revocation password

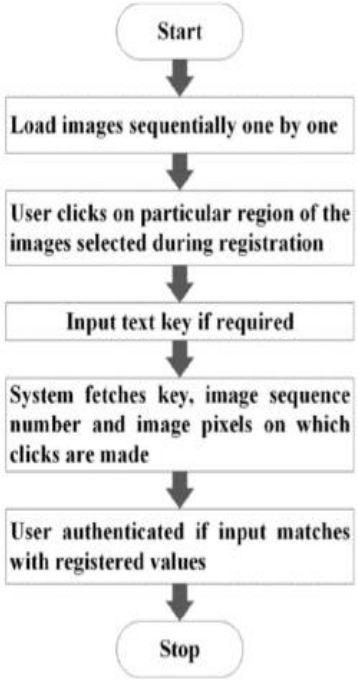

Fig. 3.1. Block diagram of the proposed system.

the system is more prone to shoulder surfing than the graphical password system. When a drawing is entered on the screen, an attacker only needs to see the login process once in order to obtain a password, and re-invoking is not always difficult depending on prompts or memory cues. Passwords based on recognition techniques are remembered for a longer period of time. The system discussed in this article provides greater resistance to shoulder surfing and is more efficient than Jansen et. Algorithm [6], which is based on the correct sequence of clicks on thumbnails of images. The proposed system introduces a key that would be difficult for an attacker to notice together with a correct click. The system discussed here is less confusing than the system used by Sobrado and Brjet to prevent shoulder surfing, as it contains thousands of walk-through objects on the screen, from which the user must select some objects that are selected at the registration stage [6]. Thus, introducing a keystroke along with a click provides better protection against shoulder movement compared to other algorithms. The formal specification of the proposed system is shown in Algorithm 1. The algorithm considers that the user should click on 5 images (image1, image2, image3, image4, and image7) out of n images. In addition, the user also enters an additional text key along with clicking on the image.

* The password string in question is {a, 001, (615, 335), (555, 320), (1052, 335), (115,160), (327, 695), 1111001}. Here a is the key during the first click, and (615, 335), (553, 320), (1052, 335), (115, 160), (327, 695) are the coordinates of the position of the clicks in the first, second, third, the fourth and up to the seventh corresponding images at the registration stage. The user has no particular problem remembering the click points as they are clearly defined locations within the image displayed on the screen.

Algorithm 1: Authenticating a user with graphical passwords

Input:

1. mouseX=x-coordinate of mouse when it is being clicked
2. mouseY=y-coordinate of mouse when it is being clicked
3. ch=the key entered by the user
4. mouseClick=Boolean variable which returns 1 if mouse is clicked, else 0

```
1.  Create an array img of n images;
2.  while all images not loaded do
3.  |  track status;
4.  end
5.  i ← 1
6.  while i <= n do
7.  |  display img[i] for 2.5 sec;
8.  |  if i=1 and mouseClick=yes and mouseX ≥ (615-15) and mouseX≤(615+15) and
    |  mouseY≥(335-15) and mouseY≤(335+15) and ch=a then
9.  |  |  click1 ← 1
10. |  end
11. |  if i=2 and mouse Click=yes and click1=1 and mouseX≥(555-15) and
    |  mouseX≤(555+15) and mouseY≥(320-15) and mouseY≥(320+15) then
12. |  |  click2 ← 1
13. |  end
14. |  if i=3 and mouseClick=yes and click2=1 and mouseX≥(1052-15) and
    |  mouseX≤(1052+15) and mouseY≥(335-15) and mouseY≤(335+15) then
15. |  |  click3 ← 1
16. |  end
17. |  if i=4 and mouseClick=yes and click3=1 and mouseX≥(115-15) and
    |  mouseX≤(115+15) and mouseY≥(160-15) and mouseY≥(160+15) then
18. |  |  click4 ← 1
19. |  end
20. |  if i=7 and mouseClick=yes and click4=1 and mouseX≥(327-15) and
    |  mouseX≤(327+15) and mouseY≥(695-15) and mouseY≤695+15 then
21. |  |  print "USER AUTHENTICATED";
22. |  end
23. |  i ← i + 1
24. end
```

DISCUSSION AND ANALYSIS:

This section provides a mathematical analysis of the proposed system in terms of password space, memory requirements, and security clearance. The discussion is supported by a number of case studies to substantiate the effectiveness of the proposed work.

Password Space;

User needs to input an ASCII character as a key. Each key can take 2^7 different values.
Size of key = 1 Character
Password space for the keys= 2^7
Size of each image= 1360 x 660 pixels
Tolerance= 30 x 30 pixels

Hence, number of square grids for clicking = (1360 x 660) / (30 x 30) = 997
Consider the total number of images = 7
However, only 5 images have to be selected out of 7.
Number of possible ways in which this can be done is $^7C_5 = 21$
Total number of clicks required = 5 Password space for this = $21 \times (997)^5 = 2^{52.61}$
If key is input with the first click, password space of the system = $(2^7) \times (2^{52\,61}) = 2^{59.61}$
However, the key can be input with any of the 5 clicks.
So, total Password space of the system = 5 x $(2^{59.61}) = 2^{61.93}$
This is greater than that of password space of an 8-character (ASCII) alphanumeric password, which is $(2^7)^8 = 2^{56}$.
A super computer may test 100 million passwords every second. Therefore, time required to test 10^7 passwords = 1s
Time required to test $2^{23.25}$ passwords = 1s
Time required to test $2^{61.93}$ passwords = $(2^{61.93}) / (2^{23.25}) = 2^{38.68}$s = 13964.76 years = 14 thousand years (approx.)
Time required to test 2^{56} passwords = $(2^{56}) / (2^{23.25}) = 2^{32\,75}$ seconds = 229 years

Hence, it can be clearly deduced that it would be infeasible to use even a supercomputer for a brute force attack on the proposed system. Table 4.1 shows that there is no much difference in password space, however, the time required to brute force graphical password is 60 times greater than that of textual password system. Hence, graphical password with 7 images and key of length 1 byte is 60 times more secure than textual password.

TABLE 4.1 Comparison of text and graphical password systems

Password system	Space for password	The time it takes for brute force
8-character text password	2^{56}	229 years
Picture password with 7 images in slideshow and 1 byte key	$2^{61.93}$	13964 years (almost 14 thousand years)

4.1. Practical example.

The following subsection presents various case studies concerning the size of the enter key, the number of images, and other important parameters of the proposed system.

Case 1:
Size of input key = 4 characters
Number of images = 5
Number of input keys required = 5
Password space for keys = $(2^7)^4 = 2^{28}$
Password space for the clicks = $^7C_5 \times (997)^5 = 2^{52.61}$
Total Password space of the system = $2^{28} \times 2^{52.61} = 2^{80.61}$
Time required to test $2^{85.78}$ passwords = $(2^{80.61})/(2^{23.25}) = 2^{57\,36}$ s = $5.86 * 10^9$ years = $5.86 * 10^3$ million years (approx.)

Case 2:
Number of images in slide show =10 Size of input key = 4 Characters Number of input keys

required = 5 Password space for clicks = $({}^{10}C_5)$ x $(997)^5$ = $2^{57.78}$ Total password space of the system = 2^{28} x $2^{57.78}$ = $2^{85.78}$

Time required to test $2^{85.78}$ passwords = $(2^{85.78})/(2^{23.25})$ = $2^{62.53}$ s = 2.11 * 10^{11} years = 2.11 * 10^5 million years (approx.)

Case 3:

Size of input key = 1 Character

Number of images = 6

Total number of clicks required = 5

Password space for the clicks = $^6C_5$ x $(997)^5$ = $2^{52.39}$

The key can be with any of the 5 clicks.

Total password space of the system = 5x(2^7) x ($2^{52.39}$) = $2^{61.71}$

Time required to test $2^{61.71}$ passwords = $(2^{61.71})/(2^{23.25})$ s = $2^{38.46}$ s = 11989 years

The memory is still larger than the 8-digit ASCII text password, but it can be reduced if an 8-bit grayscale image is displayed. Hence the total space required to store the slideshow = 6 x 1360 x 660 x 8 bits = 5.14 MB.

Case 4

Let the number of images = 5 Total number of clicks required = 5 Therefore, password space for the clicks = $(997)^5$ = $2^{49.8}$

The key can be input with any of the 5 clicks. Therefore, total password space of the system = 5 x (2^7) x ($2^{49.8}$) = 259.12

Total space required to store the slide show with gray scale image = 5 x 1360 x 660 x 8 bits = 4.28 MB

Time required to test $2^{59.12}$ passwords = ($2^{59.12}$) / ($2^{23.25}$)s = $2^{35.87}$s = 1991 years

Table 4.2 depicts that even if memory storage required for graphical password is much higher than that of textual password but still graphical password is more secure than textual password.

TABLE 4.2. Comparison of text and graphical password systems in terms of memory size and time required to brute force passwords.

Password system	Memory required for storage	The time it takes for brute force
8-character text password	56 bit	229 years
8-bit six-halftone picture password	5.14 MB	12 thousand years
8-bit 5 halftone picture password	4.28 MB	2 thousand years

Case 5:

Let the number of images = 4

Number of clicks required = 4

Password space for clicks = $(997)^4$

Total password space of the system = 4 x 2^7 x 997^4 = 2^{48}

Time required to test $2^{29.4}$ passwords = $(2^{48})/(2^{23.25})$s = $2^{24.75}$s = 326 days

TABLE 4.3 Influence of images and key size on password size and time required to brute force passwords

Number of images in slide show	Size of keys in characters	Image selected	Password space	Time required to brute force
7	1	5	$2^{61.93}$	14 thousand years
5	4	5	$2^{80.61}$	5.86 x 10^3 million Years
10	4	5	$2^{85.78}$	2.11 x 10^5 million Years
6	1	5	$2^{61.71}$	11989 years
5	1	5	$2^{59.21}$	1991 years
4	1	4	2^{48}	326 days

From Table 4.3, we can conclude that even if the text key size remains the same, but the number of images doubles during the slideshow, the time it takes for brute force is also doubled. Increasing the size of the keys is not as important as increasing the number of images during a slideshow. If we reduce the number of images to 4, the space for the password will be less than that of an 8-bit ASCII

(text) password. Therefore, this case is optimal with minimal password space.

The likelihood of an attacker determining the correct password:

Probability of identifying the correct key = $1/(2^7)$

Probability of identifying the correct click with which the key has to be input = 1/5

Probability of identifying the correct region of first click in an image = 1 / ((1360 x 660) px / (30 x 30)px) = 3/2992

Probability of choosing the correct 5 images on which to click out of the 7 images = $1/ {}^7C_5 = 1/21$

Probability of identifying the correct password of the user = (1/5) x (1/128) x (1/21) x $(3/2992)^5 = 7.54 \times (10)^{-20}$

This is too small. Hence, our graphical authentication system is very less vulnerable to password guessing.

It is clear from Table 4.4 that chances of guessing graphical password will always be less than that of alphanumeric password.

TABLE 4.4 Probability of identification of different password systems

Password system	Identification probability
Alphanumeric password	1.3×10^{-17}
Picture password	7.54×10^{-20}

Storage capacity:

Graphical passwords require more space for storage compared to a text password, but in the era of techno - logical progress, the storage place to improve security can not be a problem.

Size of each image = 1360 x 660 pixels.

Space occupied to store 1 pixel = 32 bits on a 32-bit display

Not at all. images in our system = 7

Hence the total space required = 7 x 1360 x 660 x 32 bits = 23.96 MB, which is a lot.

Case 1: binary image:

Space required to store each pixel = 1 bit

Total space required for storing slideshows = 7 x 1360 x 660 x 1 bit = 767 KB

This is much less than what is required on a 32-bit display. Although the number of colors that can be displayed in an image does not affect the proposed system, nevertheless, some users may not prefer using binary images, as it may cause them some inconvenience.

Case 2: 8-bit grayscale images:

Total space required for storing slideshows = 7 x 1360 x 660 x 8 bits = 6 MB

Table 4.5 shows the storage requirements for various password systems.

Storage space for the proposed system:

The password string consists of one key and coordinates (x, y) for each of 5 clicks.

Space required to store 1 key character = 1 byte

Space required to store 5 (click points) x 2 (coordinates of each click), i.e. 10 integers = 10 x 2 bytes = 20 bytes.

Number of bits required to store input key associated with one of the 7 image = 3 bits

(010 indicate the key has to be entered with 2nd image.) Number of bits required to represent the selected image = 7 bits

(1111001 indicates clicks in 1st, 2nd, 3rd, 4th and 7th images are required respectively)

TABLE 4.5 Total memory capacity of different password systems

Password system	Storage space required
Textual password length = 8 characters	56 bits
Graphical password Binary image	767 KB
Graphical password 8-Bit gray scale image	6 MB
Graphical password Colored image	23.96 MB

Therefore, the total amount of memory required to store each user's password string = 3 bits + 1 byte + 20 bytes + 7 bits = 178 bits.

The password string will look like: key, click, (X1, Y1), (X2, Y2), (X3, Y3), (X4, Y4), (X5, Y5), select Here, click refers to the sequence number of the click, to which the key is associated, and the selection refers to the sequential number of the images (in 7 bits) that were selected by the user. Table 4.6 shows that a single 8-character password string requires 56 bits of memory, while the proposed password scheme requires 178 bits of memory.

TABLE 4.6 Comparison of memory space requirements

Password system	Space required to store the password string
Textual password	56 bit
Proposed scheme of password	178 bts

CONCLUSIONS AND FUTURE WORK:

User authentication is one of the most important components of a secure system. Even after the development of advanced authentication mechanisms such as biometrics, the traditional concept of passwords is still the most widely used means of authenticating users. From - the limitation and deficiencies textual passwords, such as password smaller space susceptibility to brute force attacks and shoulder surfing, etc., this document proposes a new pattern on the basis of multifactorial authentication scheme, which comprises using a combination of text and graphics passwords. The proposed system has more space for passwords and is protected from dictionary attacks as it includes additional mouse input along with keyboard input. Moreover, a brute force attack would require the automatic creation of all possible combinations of mouse click and text to crack the actual password. This makes a Bruce-force attack impossible for the proposed system.

Viewing multiple graphics while logging in can be tedious and time-consuming. It also requires storing tens of thousands of images in a centralized database, and therefore optimal storage space is also an issue. In the future, research may be conducted on optimal image storage combined with minimizing network latency. In addition, one of the main design problems of the proposed system is related to the accuracy and reliability of the data entered by the user. High robustness can lead to many false positives, while low tolerances can lead to many false negatives. This requires an optimal error tolerance strategy to improve the accuracy of the system.

REFERENCES:

1) HUNT, H. C., & SHEA, A. (2018). Enhanced user authentication. U.S. Patent Application No. 10/078,783.
2) KHARI, M., SHRIVASTAVA, G., GUPTA, S., AND GUPTA, R. (2017). Role of Cyber Security in Today's Scenario. In R. Kumar, P. Pattnaik, and P. Pandey (Eds.), Detecting and Mitigating Robotic Cyber Security Risks (pp. 177-191). Hershey, PA: IGI Global.
3) SAXENA, A., SHRIVASTAVA, G., & SHARMA, K. (2012). Forensic investigation in cloud computing environment. The International Journal of forensic computer science, 2, 64-74.
4) VELSQUEZ, I., CARO, A., & RODRGUEZ, A. (2018). Authentication schemes and methods. Information and Software Technology, 94(C), 30-37.
5) AWAD, A., & LIU, Y. (2019). Cognitive Biometrics for User Authentication. In Biometric-Based Physical and Cybersecurity Systems (pp. 387-399). Springer, Cham.
6) JANSEN, W., GAVRILA, S. I., KOROLEV, V., AYERS, R. P., & SWANSTROM, R. (2003). Picture password: a visual login technique for mobile devices. UMBC Student Collection.

7) ABHISHEK, K., ROSHAN, S., KUMAR, P., & RANJAN, R. (2013). A comprehensive study on multifactor authentication schemes. In Advances in Computing and Information Technology (pp. 561-568). Springer, Berlin, Heidelberg.
8) FRANCHI, E., POGGI, A., & TOMAIUOLO, M. (2015). Information and Password Attacks on Social Networks: An Argument for Cryptography. Journal of Information Technology Research (JITR), 8(1), 25-42.
9) CNN Business (2013). 5 of the biggest-ever credit card hacks. https://money.cnn.com/gallery/technology/security/2013/12/19/ biggest-credit-card-hacks [Accessed on Jan. 31, 2019]
10) GILHOOLY, K. (2005). Biometrics: Getting back to business. Computerworld, May, 9, 2005
11) Amit, E., Rim, s., Halbeisen, G., Priva, U. C., Stephan, E., & Trope, Y. (2019). Distance-dependent memory for pictures and words. Journal of Memory and Language, 105, 119-130.
12) AGARWAL, G., SINGH, S., & SHUKLA, R. S. (2010). Security analysis of graphical passwords over the alphanumeric passwords. International Journal of Pure and Applied Sciences and Technology, 1(2), 60-66.
13) EMIL PROTALINSKI (2012). The top 10 passwords from the Yahoo hack: Is yours one of them?. https://www.zdnet.com/article/the-top-10-passwords-from-the-yahoo-hack-is-yours-one-of-them [Accessed on Jan. 31, 2019]
14) AHITAGNI (2012). 453,000 Yahoo voice, username and password leaked. http://www.ahitagni.com/?p=422 [Accessed on Jan.31, 2019]
15) SABZEVAR, A. P., & STAVROU, A. (2008). Universal multi-factor authentication using graphical passwords. In Signal Image Technology and Internet Based Systems, 2008. SITIS'08. IEEE International Conference on (pp. 625-632). IEEE.
16) DE ANGELI, A., COVENTRY, L., JOHNSON, G., & RENAUD, K. (2005). Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems. International Journal of Human-Computer Studies, 63(1-2), 128-152.
17) RENAUD, K. V. (2009). Guidelines for designing graphical authentication mechanism interfaces. International Journal of Information and Computer Security, 3(1), 60-85.
18) Passfaces: Two Factor Authentication for the Enterprise. http://www.passfaces.com/ [Accessed on Jan. 31, 2019]
19) HOLLINGWORTH, A., & HENDERSON, J. M. (2002). Accurate visual memory for previously attended objects in natural scenes. Journal of Experimental Psychology: Human Perception and Performance, 28(1), 113-136.
20) SOBRADO, L., & BIRGET, J. (2002). Graphical passwords, The Rutgers Scholar, An electronic bulletin of undergraduate research. Rutgers University, Camden New Jersey, 4,12-18.
21) ZHENG, Z., LIU, X., YIN, L., & LIU, Z. (2010). A Hybrid Password Authentication Scheme Based on Shape and Text. Journal of Computers, 5(5), 765-772.