

Graphical Pattern based Authentication Schemes for Session Passwords

Apeksha Gadkar
Dept. of Computer Engineering
SKNCOE, Pune, India
apekshagadkar@gmail.com

Ritvik Joshi
Dept. of Computer Engineering
SKNCOE, Pune, India
ritvikjoshi1992@gmail.com

Pranav Jain
Dept. of Computer Engineering
SKNCOE, Pune, India
armsrace1945@live.com

Mahendrapratap Singh
Dept. of Computer Engineering
SKNCOE, Pune, India
mahendra289@gmail.com

Pushkar Jaltare
Dept. of Computer Engineering
SKNCOE, Pune, India
pushkar2911@rediffmail.com

Abstract— Despite being the most common authentication scheme, textual passwords face various security threats like dictionary attack, brute force attack, shoulder surfing, eavesdropping and social engineering. These vulnerabilities have inspired numerous individuals and organizations to come up with new and better authentication schemes that aim to provide better security and resistance against the mentioned security threats. But these new schemes, which are mostly graphical in nature, are vulnerable to shoulder surfing. To overcome this problem, we introduce and elaborate the implementation of two session password authentication schemes in this paper. In these schemes, text is combined with colors, to generate a session password. The users follow a definite scheme to extract characters of the session password from grids randomly generated by the authentication system. A session password can be used only once, as, after the end of every session, a new password is generated by the system. These techniques can be applied for authenticating users in Personal Digital Assistants, web applications and for locking bank vaults and lockers.

Keywords—Authentication, Session Passwords, Eavesdropping, Social Engineering, Dictionary Attacks.

I. INTRODUCTION

Textual passwords are the most common authentication schemes used today. But these are vulnerable to various security threats like dictionary attack, shoulder surfing, eavesdropping and social engineering. A possible solution is using longer and more random passwords. But this faces the problem of being harder to remember. Studies show that people tend to pick passwords that are familiar to them and include parts of their names or date of births and thus, are easy to crack or guess [1]. Biometrics based authentications [2,3]

like finger print recognition, facial or iris recognition, face technical challenges like being slow, costly and require a good share of maintenance. So, these are not that widely deployed.

A viable option, that tries to overcome these drawbacks, is to use graphical passwords. But, most of the graphical passwords devised during the last decade, face the problem of shoulder surfing. Some of the newer schemes that claim to be resistant to shoulder surfing, have their own share of drawbacks like usability difficulties and have problems setting the appropriate tolerance levels.

In this paper, we propose two session based authentication schemes [5] that use a combination of text and colors. A session password is generated each time the user tries to log into the system. This password is rendered useless after the session expires. Session passwords provide better security against dictionary and brute force attacks as the password changes for every session.

A user has to enter the password and other details during the registration phase of the authentication scheme. After successful registration, the user can log in using any of the two proposed authentication schemes. Upon successful verification of the session password of the corresponding scheme, the user is granted access to the system.

II. RELATED WORK

Various researchers have come up with diverse graphical authentication schemes, each one trying to cope up with the drawbacks of the others. Some of these schemes are described below. We have categorized the schemes based on the type of actions carried out by the user to login.

A. Those involving a specific sequence of steps by the user

Jermyn, et al. [11] proposed a technique called Draw-a-Secret (DAS) where the user is required to re-draw a pre-defined picture on a 2D grid. The drawing should touch the same grids in order to get the user authenticated. Goldberg [9] designed a technique called “passdoodle”, which uses handwritten design or text drawn on a screen using a stylus. These two schemes are vulnerable to shoulder surfing. Syukri [12] developed a technique where the user is authenticated if he draws his signature on the screen correctly, using a mouse. Various parameters, set during the registration phase, are verified to authenticate the signature. But this scheme is vulnerable to signature forgery.

B. Clicking in a particular region/sequence

Blonder [13] designed a scheme, where, in order to authenticate, the user must click on approximate areas of pre-defined locations. Passlogix [14] modified this scheme by changing the areas into items. Clicking upon these items in correct sequence would grant access to the user. Wiedenback [6] devised a scheme, wherein, the user is required to click inside the convex hull formed by pre-selected objects, to gain entry into the system. All these schemes are vulnerable to either shoulder surfing or guessing.

C. Identifying/choosing the correct sequence/object

Dhamija and Perrig [10] devised a scheme, where, in order to authenticate, a user has to identify a set of pre-defined images, selected during the registration phase. A technique called Passface [15] uses a scheme, where, the user has to select a pre-defined face from a grid of nine faces. As, during the registration, the user has to choose four faces, in order to gain entry into the system, this process has to be repeated correctly four times. Jansen [7] proposed a scheme for mobile devices, where, the user has to set a sequence of pictures as the password and recreate the sequence during the login phase. Correct recreation of the sequence grants access to the user. All the above schemes are vulnerable to shoulder surfing. A hacker can replicate a user’s actions and gain unauthorized entry into the system.

D. Shoulder surfing resistant schemes

Zhao and Li [4] proposed a shoulder surfing resistant scheme “S3PAS” where users have to find their original text passwords in a login image, and click inside an invisible triangle region to gain entry. Man, et al. [8] proposed a scheme, where the user chooses some images as pass-objects, each of which is assigned a unique code. During authentication, the user is required to enter the codes corresponding to the scenes provided by the system. The drawback of this scheme is that the user has to remember the codes of the images.

III. THE PROPOSED SCHEMES

In order to overcome the drawbacks of the currently used authentication schemes, the two proposed schemes use grids with a combination of text and colors. Both the schemes, unlike

most graphical authentication schemes currently used, are resistant to shoulder surfing and various other security threats. The techniques consist of three main phases: registration phase, login phase and the verification phase. In case the user forgets the password, a recovery phase will ensure suitable recovery of the password, using recovery techniques like security question or password reset by sending an email to the user’s account.

Figure 1 shows a sample registration screen for both the schemes combined. As shown, the user has to assign unique color codes to the eight colours given at the top of the figure for the hybrid scheme. The user also enters the textual (alphanumeric) password for second, i.e., the textual scheme. He enters other personal details in the respective fields shown in the figure. The user gets registered successfully once all the constraints on the password and the personal details are satisfied.

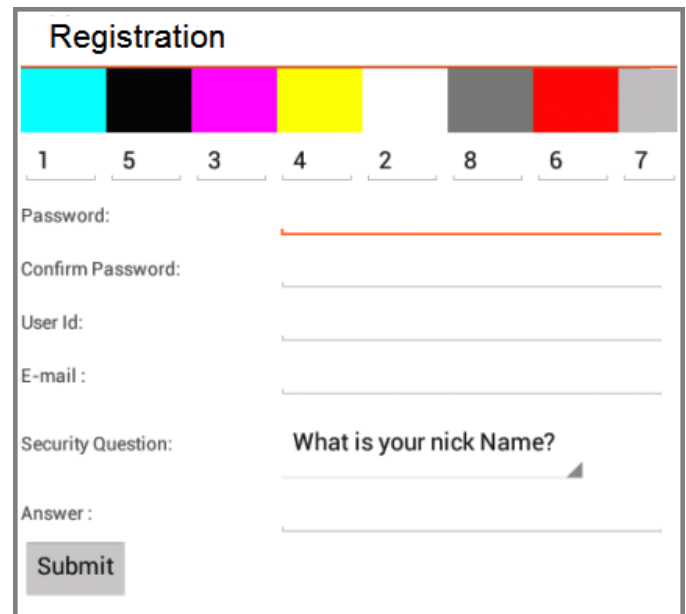


Fig. 1. Combined registration phase for both the schemes

A. Pair-based Authentication Scheme

This scheme requires the user to enter a username and a password during the registration phase. The password should be of minimum 8 characters, with even number of characters and containing a combination of the alphabets (a-z) and the digits (0-9), without repetition. This is the original password and has to be always remembered by the user. The registration phase is complete once the user enters a password that meets the constraints specified.

For the login phase, the system generates an interface consisting of a 6x6 grid of alphabets (a-z) and numbers (0-9), arranged randomly. The interface is different for each session. The user’s job is to calculate the session password in the following way:

1. Consider the original password’s characters as pairs.
2. For each pair, look for the intersection of the first character’s row and the second character’s column.

3. The intersection mentioned in step 2 is a character in the session password for the current session.
4. Obtain the session password characters, as mentioned above, for the original password, left to right, and enter them in the login text box.

Consider an example where the original password is “QWERTYUI”. As per the algorithm given above, the first pair is “QW”. Step 2 instructs to look for the elements in the first character’s row, i.e., Q’s row and the second character’s column, i.e., W’s column. The intersection gives us “B”, which is the first digit in the session password. Obtaining the remaining characters in a similar manner, we obtain the next three characters as “TE1”, completing the session password as “BTE1”.

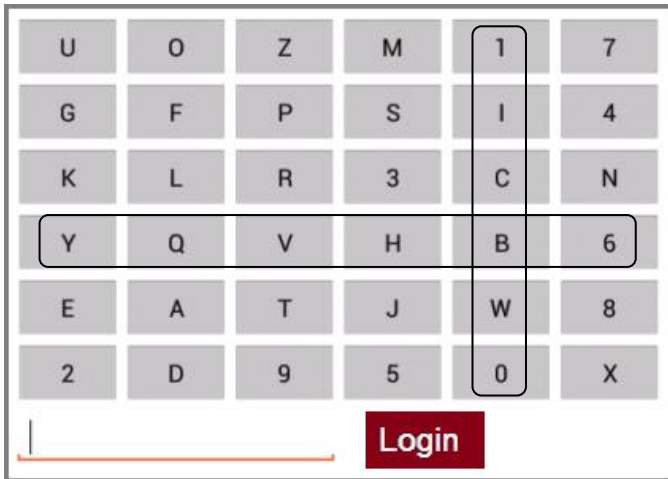


Fig. 2. Login phase of Pair-based Scheme

The system calculates the session password in a similar manner for each session. During the user validation phase, the text entered in the login text box is compared with the session password generated by the system. When both of them match, the user gets system access.

As the interface changes after each session, this scheme is resistant to shoulder surfing and other attacks. Even if a hacker finds out the rows and columns, i.e., a partial key, guessing/finding the right character requires him to choose from a number of characters. The interface changes each time, making it hard to guess the correct characters, let alone the complete password, making this scheme resistant to hackers.

B. Hybrid text Authentication Scheme

This scheme requires the user to provide a username and assign color codes to 8 pre-decided colors during registration. Each color has to be given a color code or rating from 1-8, with no repetitions, as per the user’s wish. Upon satisfaction of this criterion, the user gets registered successfully. During the login phase, after entering the username, the user gets presented with an interface consisting of an 8x8 grid and a color strip with a random sequence of the 8 original colors to which the user assigned codes during registration. The grid consists of the letters 1-8, placed randomly in each cell of the rows and columns of the 8x8 grid. The grid is indexed as 1-8, for both

the rows and the columns. These indexes are used by the user to find the appropriate rows and columns while finding the intersections for session password digits.

The user has to calculate the session password as follows:

1. Consider the colors of the color strip as pairs and recall the codes (ratings) assigned to each of the colors during registration.
2. For each pair of color codes generated, look for the intersection of the first digit’s row and the second digit’s column, each time referring to the corresponding index row and column.
3. Generate the intersection for all the 4 pairs of colors in the color strip. This is the session password for the current session.

Consider an example where the original color codes are: CYAN = 2, BLACK = 5, PINK = 1, YELLOW = 8, WHITE = 7, DARK GRAY = 3, RED = 4, LIGHT GRAY = 6.

For a color strip generated by the system as shown in Fig. 3, following the algorithm given, we recall the color codes, and thus, for the first pair, i.e., RED-LIGHT GRAY, we get the codes as 4 and 6. We now check for the intersection of the 4th row and the 6th column, giving us the digit 8, which is the first digit of the session password. Proceeding in a similar manner, we obtain the session password as “8842”.

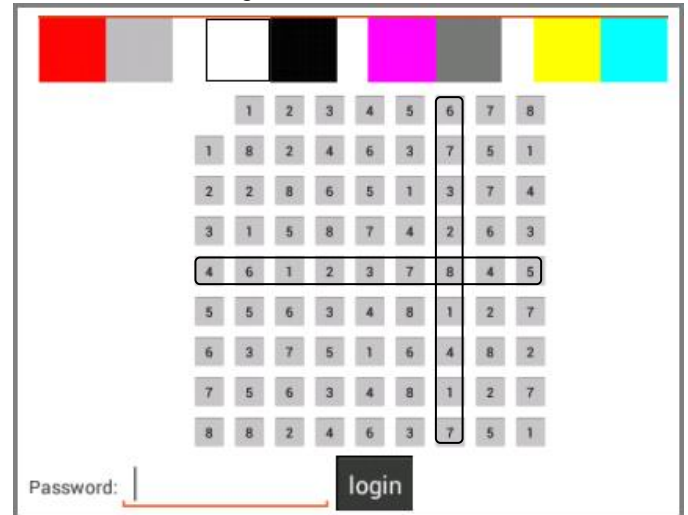


Fig. 3. Login phase for Hybrid authentication scheme

The system generates a session password for the current session in a similar manner. It finds out the intersections as mentioned in the steps above, to find out the session password, then comparing it with that entered by the user, inside the login box. If both of them match, the user gets authenticated, gaining entry into the system.

The system is resistant to shoulder surfing, as a hacker cannot guess the color codes assigned. Even if he gets a partial guess right, the 8x8 grid makes it complex enough to guess the complete password. Also, the interface (color strip and the grid) changes for each new session, thus making it resistant to shoulder surfing and other attacks.

IV. IMPLEMENTATION

A. Agent environment

We designed an android app that uses the two authentication schemes to lock the user's details. The app was developed using Android development toolkit (ADT) with Java programming language. In client devices, the app runs within an instance of the Dalvik Virtual Machine.

B. Algorithm for text based scheme

The text based scheme uses two main algorithms: one to generate the interface and other to calculate the session password.

Algorithm 1 GenerateInterface

1. add A-Z and 0-9 into array list val
2. $i=0$
3. if $i < (\text{pass.length})/2$
4. generate 4 random values from 0 to 5 using math.random and put in row1, row2, col1, col2 respectively
5. if (row1 = row2) and (col1 = col2)
6. go to step 3
7. else
8. $a \leftarrow \text{concatenate}(\text{row1}, \text{col1})$, $b \leftarrow \text{concatenate}(\text{row2}, \text{col2})$
9. if a or b exist in the array M already then
10. go to step 3
11. else
12. add a and b to M
13. add character c from current index of password array into 2D array $\text{Mf}[\text{row1}, \text{col1}]$
14. remove(c) from val
15. increment index of password array
16. add c of from current index of password array into $\text{Mf}[\text{row2}, \text{col2}]$
17. remove(c) from val
18. increment index of password array
19. end if
20. end if
21. increment i
22. go to step 3
23. end if
24. fill the remaining entries from val randomly into Mf

The second algorithm uses the interface created from Algorithm 1 to calculate the session password. Algorithm 2 shows the basic steps in the process.

Algorithm 2 GenerateSessionPassword

1. if $i < (\text{pass.length})/2$
2. retrieve row1 and col2 from M

3. find value at row1,col2 from Mf
4. add value to character array session password
5. increment i
6. go to step 1
7. end if

C. Algorithm for hybrid scheme

The hybrid scheme uses two main algorithms: one to generate the color strips and the interface and other to calculate the session password.

Algorithm 3 GenerateColorStrip

1. add numbers 1-8 uniquely to array list a
2. while a is not empty do
3. $r_i \leftarrow$ random value from a using math.random
4. remove r from a
5. end while
6. $c_j \leftarrow j^{\text{th}}$ color cell
7. for all r compare r_i with predefined color codes
8. assign colors to c_j accordingly

The 8x8 interface is generated such that no element gets repeated in a particular column or row. The session password is generated in a similar way as done in the case of text based scheme.

V. SECURITY ANALYSIS AND USER STUDY

A. Resistance to various attacks

The two proposed schemes are resistant to the various security threats like dictionary attacks, brute force attacks, shoulder surfing, guessing, etc. The concept of session passwords helps randomize the password for each session. The interface changes for each session, changing the session password each time, making it secure to various attacks.

Dictionary attacks are attacks where the hackers use tools that try dictionary words sequentially, to hack into the system. But as our system's session password changes with the interface and the session password is some random alphanumeric value, the dictionary attack is rendered useless. Brute force attacks too, are rendered useless and uncertain, as session passwords are used.

Guessing is out of question in the text based scheme, as it has a complexity of 36^4 and the interface changes after each session. In the hybrid scheme, the only possibility of cracking the system is when the user is careless enough to use the ratings sequentially, numbering the colors 1-8. Otherwise, even after noting down a few session passwords, the system is unbreakable.

Shoulder surfing or hidden camera attacks are ineffective too, as in the case of the pair based scheme, the original password created during the registration remains hidden during the registration phase. Even if the hacker manages to find a session password, it is not enough to trace back to the original

password. In the hybrid scheme, the color codes are assigned during registration. In case a hacker manages to find a session password, it cannot be traced back to the color codes thanks to the complexity being 8^4 .

B. Complexity

For the pair based scheme, the complexity depends on the length of the original password. For a password of length x (x is even), the complexity is 36^x . In case of the hybrid scheme, the complexity is (8^4) .

C. User Study

We conducted a demo for the user-friendliness and usability of the system. The users were briefed about the two schemes using demonstrations. A group of 20 participants was used to test the various possible issues with the system. The registration and login times for various users were tabulated. The user study data is presented in Table I. All the numbers shown in the table are in seconds. The table shows the time taken by the user to register in both the schemes combined and for three sessions which were conducted few hours apart. The best, average and the worst time is tabulated for each scheme separately.

It was observed that, as the users got familiar with the system, they were able to log into the system without any problems. They found the pair based authentication scheme easier than the hybrid scheme. The users, who could devise some personalized method to remember the hybrid scheme's color codes, preferred the method, as they felt it to be safer than the pair based one. The user study indicates that normal users tend to use the pair based scheme as they feel it to be easier, whereas, the more advanced users prefer the hybrid scheme as they feel it to be safer than the pair based scheme.

TABLE I. USER STUDY DATA

Technique	Time taken	Registration	Session 1	Session 2	Session 3
Pair based scheme	Best	85.7	26.9	24.9	21
	Average	96.8	31.8	30.5	28.2
	Worst	113.3	46	44.4	43.4
Hybrid scheme	Best	85.7	32.2	31.2	28
	Average	96	49.4	47.9	44.3
	Worst	113.3	70.3	69.2	66.2

VI. CONCLUSION

The two proposed authentication schemes will provide resistance against various security threats like dictionary attacks, brute force attacks, guessing, shoulder surfing and social engineering. Both the schemes involve grids of characters, which the user uses to obtain session passwords for each session. The pair based scheme requires the user to enter an even character password during registration, which is then used in the login stage to generate the session password by both- the system and the user. In the hybrid scheme, the user assigns color codes to 8 pre-defined colors. To login, he calculates a session password for every session by finding the

intersections in the grid corresponding to the color sequence generated randomly by the system.

ACKNOWLEDGMENT

We would like to thank Prof. Apeksha Gadkar for guiding us throughout the selection and implementation of these schemes and providing the resources for making this paper possible.

REFERENCES

- [1] A. Adams and M. A. Sasse, "Users are not the enemy: why users compromise computer security mechanisms and how to take remedial measures," *Communications of the ACM*, vol. 42, pp. 41-46, 1999.
- [2] K. Gilhooly, "Biometrics: Getting Back to Business," in *Computerworld*, May 09, 2005.
- [3] A. Jain, L. Hong, and S. Pankanti, "Biometric identification," *Communications of the ACM*, vol. 33, pp. 168-176, 2000.
- [4] H. Zhao and X. Li, "S3PAS: A Scalable Shoulder-Surfing Resistant Textual-Graphical Password Authentication Scheme," in *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW 07)*, vol. 2. Canada, 2007, pp. 467-472.
- [5] M. Sreelatha, M. Shashi, M. Anirudh, MD Sultan Ahamer, V Manoj Kumar "Authentication Schemes for Session Passwords using Color and Images", *International Journal of Network Security & Its Applications (IJNSA)*, Vol.3, No.3, May2011.
- [6] S. Wiedenbeck, J. Waters, J.C. Birget, A. Brodskiy, N. Memon, "Design and longitudinal evaluation of a graphical password system". *International J. of Human-Computer Studies* 63 (2005) 102-127.
- [7] W. Jansen, "Authenticating Mobile Device User through Image Selection," in *Data Security*, 2004.
- [8] S. Man, D. Hong, and M. Mathews, "A shoulder surfing resistant graphical password scheme," in *Proceedings of International conference on security and management*. Las Vegas, NV, 2003.
- [9] J. Goldberg, J. Hagman, V. Sazawal, "Doodling Our Way to Better Authentication", *CHI '02 extended abstracts on Human Factors in Computer Systems*, 2002.
- [10] R. Dhamija, and A. Perrig. "Déjà Vu: A User Study Using Images for Authentication". In *9th USENIX Security Symposium*, 2000.
- [11] Jermyn, I., Mayer A., Monroe, F., Reiter, M., and Rubin., "The design and analysis of graphical passwords" in *Proceedings of USENIX Security Symposium*, August 1999.
- [12] A. F. Syukri, E. Okamoto, and M. Mambo, "A User Identification System Using Signature Written with Mouse," in *Third Australasian Conference on Information Security and Privacy (ACISP)*: Springer- Verlag Lecture Notes in Computer Science (1438), 1998, pp. 403-441.
- [13] G. E. Blonder, "Graphical passwords," in *Lucent Technologies, Inc., Murray Hill, NJ, U. S. Patent*, Ed. United States, 1996.
- [14] Passlogix, site <http://www.passlogix.com>.
- [15] Real User Corporation: Passfaces. www.passfaces.com.