GAS: A Novel Grid Based Authentication System

Narayan Gowraj[1], Srinivas Avireddy[1], and Sruthi Prabhu[2]

[1] Department of Information Technology,
[2] Department of Computer Technology,
Anna University, Chennai-600044, India
{ngowraj,tholi1033,shruthe92}@yahoo.com

Abstract. With the evolving trends in technology, providing security for the users is an essential goal of the application. Authentication is one such important aspect of security which provides access control for the users of an application. The common method to provide authentication is by using a username/password pair .Graphical password authentication has proved to be more powerful and useful when compared to traditional textual password authentication. In this paper we propose a novel graphical password based authentication system called GAS(Grid based Authentication System). We focus our attention on the epigram, "It is easy to remember what we see rather than what we hear". The methodology involves choosing a pattern called Auth Pattern which is formed by placing images in a given grid. We have chosen an optimal size for the grid as 8*8.Our proposed system considers very important parameters such as user memory and length of the password. Considering these parameters we compare our system with the state of the art authentication systems to prove our methodology's efficiency.

Keywords: GAS, Grid system, authentication, dynamic, auth-pattern (AP).

1 Introduction

The increase in the usage of web applications has explicitly called for a secure authentication system which provides security for the users' credentials against unauthorized access. Current authentication methods can be divided into three main areas: Knowledge based authentication, token based authentication and biometric based authentication [1] [2]. Knowledge based authentication techniques are most widely used and include text-based passwords either in the clear text format or cipher text format. Token based authentication involves issuing of a token to authorize a user. Credit card is an example for token based authentication technique. Fingerprints, iris scan, or facial recognition are examples of biometric based authentication. The major drawback of token based and the biometric based authentication methods are that these systems are expensive and require special devices which make them unfeasible for web application.

K. Singh, A.K. Awasthi, and R. Mishra (Eds.): QSHINE 2013, LNICST 115, pp. 578–590, 2013.

Textual passwords are the first choice for authentication by humans for any web application and hence it is essential to have strong and secure authentication processes. The traditional password schemes are vulnerable to many attacks. So in this paper we present a grid based authentication system which makes use of a pattern called Auth-Pattern (AP) to authenticate a user. This Auth-Pattern is created by using images from the 64 images provided to the user. In this case we use an 8*8 grid and hence 64 images. The size of the grid can be varying and hence the application manager can decide the size of the grid depending on the application and its purposes. The number of images in an AP is a variable and not a constant. The user can use a password which has a minimum length of 6 and maximum length of 32. If a user wants to register into the GAS (Grid Bases Authentication) system then the user has to enter a unique username which is in the plain text format and select an Auth-Pattern (AP) as the password. The Auth-Pattern may consist of images placed in various grid boxes thus forming a pattern for the user. The order of arrangement of the images in the Auth-Pattern formed by the user is considered as a metric so as to increase the security.

2 Related Work

Many related graphical authentication processes have been developed till date. Any graphical authentication system falls under these three broad categories: 1. Pure Recall based authentication, 2. Cued Recall based authentication and the 3. Recognition based authentication system. Pure Recall based authentication system requires the user to replicate their user credentials without any help or any reminder. Some of the pure recall based systems are DAS, Passdoodle and Qualitative DAS [3]. The Cued Recall based authentication system is very similar to the pure recall based authentication system except that it provides the user with a framework of hints, reminders and challenge/response questions to reproduce the password. Some of the cued recall based authentication systems are PassPoint [4], Pass-Go [5]. The Recognition based authentication system requires the user to reciprocate the pattern, select the images, spot the symbols etc which they used while registration and some of the powerful unique recognition based systems are WIW, Story etc [6] [7] [8].

Pure-Recall methods such as DAS (Draw A Secret) first proposed by Jermin et al, requires a user to draw the password on a 2D grid. The coordinates of this drawing on the grid are stored in order. During authentication user must redraw the picture to declare as an authenticated user. The user is authenticated if the drawing touches the grid in the same order. The major drawback of DAS is that diagonal lines are difficult to draw and difficulties might arise when the user chooses a drawing that contains strokes that pass too close to a grid-line. Users have to draw their input sufficiently away from the grid lines and intersections in order to enter the password correctly. Another important method in this model is the Passdoodle [9] [10] which is also a pattern and a doodle is considered as a full match if it is drawn in exactly the same order as when the user initially drew the passdoodle, and is considered as a visual match if it is not a full match due to

stroke order, stroke direction, or number of strokes. They found that the order in which a password is drawn introduced much complexity to graphical passwords and suggested to neglect the order. Cued-Recall based authentication systems such as PassPoints proposed by Wiedenbeck et al in which passwords could be composed of several points on an image and the user is required to reciprocate the point correctly on the image given to authenticate as a valid user. Some of the other Cued-Recall methods include Pass-Go and Passmap etc.

The Snake and Ladder authentication system [8] is very similar to our proposed system. Any authentication system consists of three steps: Password registration, password entry and password verification. In the Snake and Ladder authentication method the user registers by remembering a sequence of grid cells or ladders or snakes for their password but the password being generated for the sequence is always static and not dynamic and the order of arrangement does not matter in the snake and ladder system which makes it less secure than the GAS system.

The paper is organized as follows. Section 3 deals with the proposed system architecture. Section 4 deals with the Methodology of the proposed system. Section 5 deals with the mathematical analysis. Section 6 and 7 deals with the evaluation of attacks and experimental results respectively. Finally section 8 has the conclusion.

3 System Architecture

The architecture of the authentication system consists of three tiers as shown in Fig.1. The three logical tiers are the user interface tier, application server tier

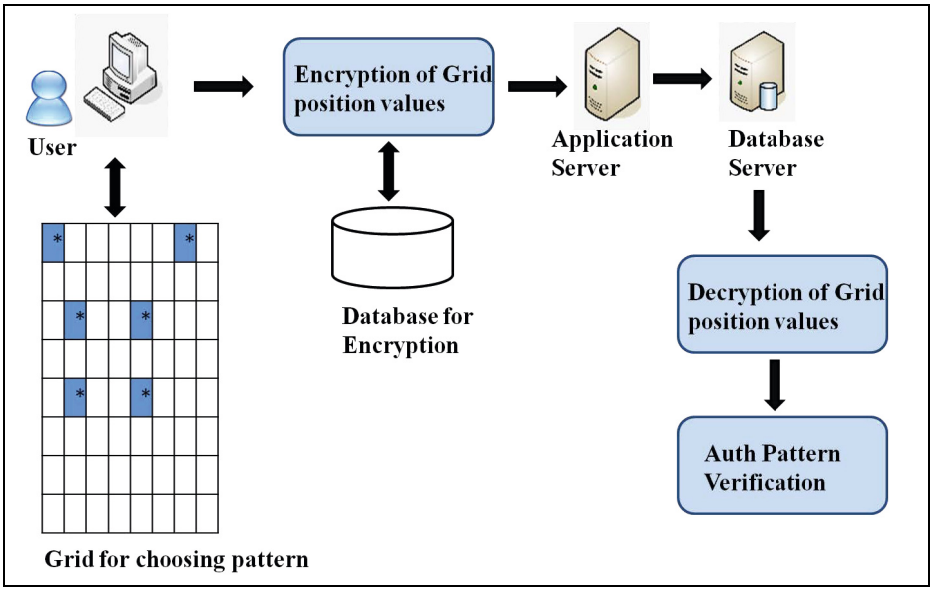
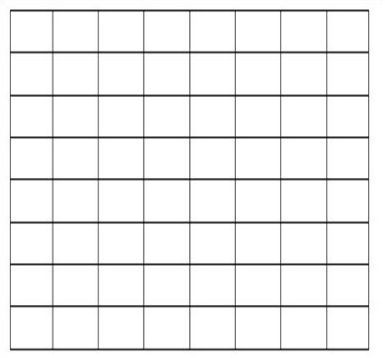


Fig. 1. System Architecture

and the database server tier. The user interface tier provides the user with the Graphical user interface (GUI) where the user interacts with the web browser to register or to validate his credentials to authenticate as a valid user which is the first step of the authentication system as discussed earlier. The user interface is the layer which allows the user to provide their user credentials. In this case, the users' username is in the plain text format and the Auth-Pattern is the password.

The user interface in the GAS system is a grid consisting of 64 boxes which is shown in Fig.3 and the user is provided with 64 images to create an Auth-Pattern as the password which is shown in Fig.2. The user has to insert the images into random locations in the grid and create a unique Auth-Pattern (AP) with the arrangement of the images as a metric in his password. The user interface is mainly created using web development languages like html, xml etc and the validation is provided using JavaScript. The user interface then communicates with the application server tier. This is accomplished by means

Fig. 2. Images for the Grid



	id	gid	loc	text
	1	1	00	YCkv
	2	1	01	fl1k
	3	1	02	HHcQ
	4	1	03	62r0
	5	1	04	raDw
	6	1	05	CTiK
	7	1	06	swD0
	8	1	07	yWZa
	9	1	10	sYq1
	10	1	11	BAnG
	11	1	12	vhPX
	12	1	13	m7Y2
	13	1	14	e8ps
	14	1	15	PJy2
	15	1	16	zpT1
	16	1	17	Dqil

Fig. 3. Grid on the left and Cipher text for locations in the grid on the right

of ordinary WWW technologies, HTTP redirects [16], URL query strings, and cookies [17].

The next layer is the database server tier which has the application server which performs the authentication of a user. Based on the result of the authentication the application server grants or rejects the user to have privileges to access the application. The database server tier consists of multiple databases, each database consists of the encrypted grid table which is shown in the Fig.3 and the user credentials stored as per the Auth-Pattern (AP) chose by the user at the time of registration. Fig.3 show the cipher text of 16 boxes with id as the primary key, gid as the grid id, loc as the location of the grid location. The grid location consists of two numbers where the first number represents the row and the second number represents the column. The text represents the cipher text of each location. Fig.3 is just a sample of the database showing only 16 locations out of the 64 locations in the case of an 8*8 grid.

4 Methodology

When the user tries to login into an application, the first step involved in the authentication procedure is the dynamic generation of a 8x8 grid with random face values for each grid position which is not seen by the user and each grid position is given a ID .The generated grid is sent to the client GUI .The client provides his username which is in the plain text format and has to place the images in the grid positions which he chose while registration. Once the images are placed in the respective positions, the encrypted random face values of each grid position in the pattern are concatenated in the order the user placed. This random string along with the grid ID is sent to the Application Server .The application server then forwards it to the Database Server for further verification to provide authentication of the user.

The database server performs two functions.The first function is to decrypt the encrypted random face values and the second function is to perform the verification of the Auth Pattern.During the first step, random string gets mapped to the respective positions and the sequence of positions separated by a delimiter is concatenated so as to form the Auth-Pattern. During the second step,this auth-pattern is checked for correctness with the Auth-Pattern stored in the persistent database during registration. If the auth pattern, is matched the user is authenticated. The grid is now removed from the database. When the user tries to login again, a new dynamic grid gets generated at the server and stored in the database. So at any point in time only one grid is being stored in the database and when the authentication is over it is removed. This minimizes the spatial overhead. Fig.4 explains our novelty using a simple 3x3 grid to authenticate users who are already registered. We have considered a 3x3 grid as an example. The application manager can decide upon the size of the grid depending on the application.

The grid is then given to the client GUI for the user to select the correct pattern to login. Let us consider the correct pattern to be the positions 00, 01

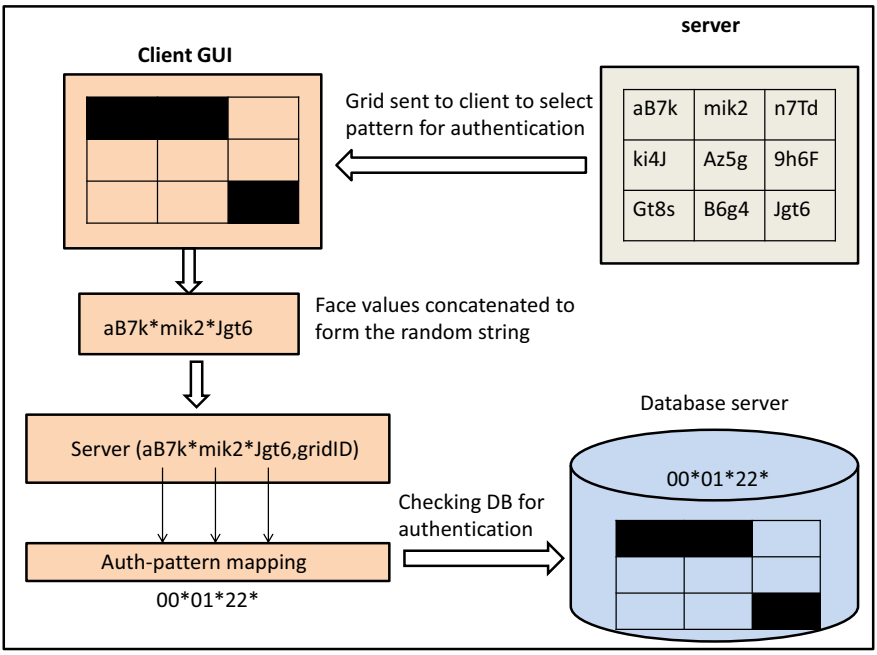


Fig. 4. Methodology of the GAS System

and 22 chosen in order. Now, when the user selects these positions 00, 01 and 22 the face value of these positions aB7k, mik2 and Jgt6 gets concatenated as aB7k* mik2* Jgt6 .Here each grid position is associated with a randomly generated four character string. So even when someone in the middle between the client and the server gets access to the random string he may not be able to crack the password which is a sequence of positions and not the random string associated with it. In the server side, the random string sent gets mapped to the respective grid positions and the Auth-Pattern 00*01*22* is generated. The auth-pattern is checked with the database. If the pattern is correct the user logs into the database, else he needs to retry. During the retry, a new grid gets generated and the old grid used will be deleted from the server, thereby improving the overhead in space complexity. The user credentials are stored in the database where the username exists in the clear text format and the password is based on the pattern the user chooses which is understandable only by the database administrator. This form of cryptic representation is useful for password recovery in case the user forgets his or her password. In systems like DAS, Passpoint and Passdoodles password recovery is not an option whereas in GAS password recovery is possible. Fig.5 shows the username and password for a few users which are stored in the database.

USERNAME	PASSWORD
User1	00*01*35*36*41*07*03
User2	07*16*13*16
User3	00*01*35*36*41
User4	15*43*32*56*71
User5	01*03*70*43*67*73
User6	11*34*25*23*21*17
User7	22*17*54*73*26*51

Fig. 5. Encryption of the Password in the Database Server

The database contents are accessed and are understood only by the database administrator and the contents such as the field names are encrypted using the encryption algorithm MD5 [18] for security concerns. Once the user is registered the user can get access into the GAS system using his legal credentials. The credentials submitted by the user at the time of login are transferred to the application tier which has the application server through http requests. The password transferred is a dynamic password which gets changed everytime the user logins into the GUI. Even if a malicious user gets illegal access to the password by using the man in the middle attacks [19] [20] or the replay attacks [21] he will get access only to the dynamic password and not the Auth-Pattern (AP).

5 Mathematical Analysis

The mathematical analysis is based on 2 parameters. They are the length of the Auth Pattern and size of the grid. The Auth pattern length is constrained by user memory. Users cant remember lengthy pattern. On the other hand, short patterns compromises security.The size of the grid is constrained by the complexity involved in the storage and retrieval. To solve this issue, we formulate a optimization problem.

Optimization problems are of three categories. They are continuous, combinatorial and NP optimization problems [22]. We choose the continuous optimization problem because we have a basic objective function followed by a set of inequality constraints. We formulate the objective function F(x), so as to maximize the security of the proposed system considering the parameters discussed above. The 2 parameters are formulated as constraints.

$$i \leq G(x) \geq j \tag{1}$$

$$k \leq H(x) \geq l \tag{2}$$

RANGE	VALUE OF r	POSSIBLE COMBINATIONS
1-3	2	2016
4-12	4	635376
4-12	6	74974368
4-12	10	151473214816
50-64	62	2016
N=64 (as it is a 8*8 grid)		

Fig. 6. Possible Combinations of Various Ranges of Auth-Pattern

The constraints G(x) and H(x) represent the length of Auth Pattern and Size of the Grid. The values i and j are determined keeping user memory and security in mind. Hence we choose optimal length of the auth pattern between 4 and 12. So the value of i is 4 and j is 12. This is based on the following analysis. Using 50 images for your Auth-Pattern is an overhead and on the other hand using very few images like 2-3 compromises security. So, it would be feasible to choose 4 to 12 images for your Auth-Pattern. Using less number of images (lesser than 4) will make the Auth-Pattern less secure and using large number of images (ranging from 30 to 64) can make the Auth-Pattern difficult to memorize. This can be mathematically proved using the combinations formula quoted in the equation 3.

$$C(n, r) = n!/r!(n - r)! \tag{3}$$

Now Fig.6 represents the various combinations for an 8*8 grid pattern where n has a constant value of 64. It is very evident from the result given in the Fig.6 that Auth-Pattern with number of images ranging from 4 to 12 is very secure that those which has ranges between 1-3 and 50-64. Fig. 7 shows the graph

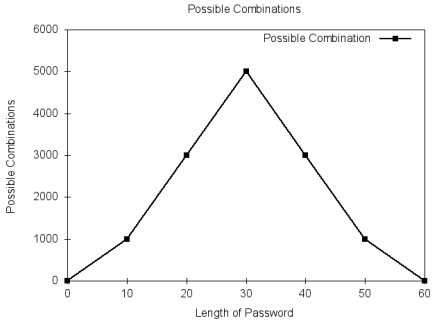


Fig. 7. Possible Combinations

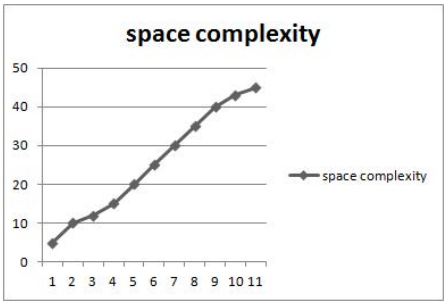


Fig. 8. Space Complexity of Various Grid Sizes

obtained from the results shown in Fig.6 which clearly indicates that any Auth-Pattern with the number of images ranging from 4 to 12 is very secure, easy to remember and incontrovertible towards password breaching. Auth-Patterns which has its length in the range of 20-40 are also very secure but very difficult to memorize, hence we go for small Auth-Patterns which are very secure and easy to remember.Since the complexity of storing retrieving the grid becomes difficult as we increase the grid size we choose values of k as 8 and l as 10 for improved security and less complexity. Fig.8 depicts how the space complexity as the size of the grid increases.

6 Evaluation against Attacks

To prove the efficiency of the proposed system under graphical password attacks, we tested our system under six most common graphical password attacks and compared it with the other state of the art authentication systems. The attacks which we consider are brute force attack, dictionary attack, spyware attack, shoulder surfing attack, social engineering and physical attack [11] [12] [13] [14] [15].

Brute force attack uses an algorithm which makes use of all possible combinations to breach the password. This is used in situations when the malicious user has no clue or hints to breach the password. Generally time consuming. Dictionary attack method uses all the words in the dictionary to check if the passwords used by the user matches any of the words in the dictionary. It is

	ATTACKS					
ALGORITHM	BRUTE FORCE	DICTIONARY	SPYWARE	SHOULDER SURFING	SOCIAL ENGINEERING	PHYSICAL
DAS	S	S	N	N	W	N
PassDoodle	W	S	N	N	S	N
PassPoints	W	S	W	W	S	W
PassGo	W	N	N	W	S	W
WIW	W	N	N	X	X	N
Story	X	X	W	N	X	N
Snake & Ladder	S	S	N	W	W	W
GAS	S	S	W	W	S	S

KEY:
1. S-Offers Strong Resistance
2. W-offers Weak Resistance
3. N-Offers No Resistance
4. X-No Research

Fig. 9. Resistance Provided by different GUA techniques

not feasible for passwords with alphanumeric keys and using dictionary attack on GUA (Graphical User Authentication) is just a waste of time. Spyware attack uses applications and tools to record sensitive data movement such as mouse movement, mouse clicks and key press incurred on the keyboard. Shoulder attack majorly depends on the window size and resolution. Key Loggers are included in this type of attack. This attack usually involves surreptitiously looking at the users credentials without the users notice. As the name suggests, it involves looking over a persons shoulder. Social engineering attack happens when a non authorized personal (malicious user) manages to impersonate sensitive data such as user credentials, codes etc from authorized employees. The attacker interacts with unsuspecting employees and gathers as much information they can to gain access to the protected data. Physical attack happens when hackers or malicious users get direct access to the data present in the server or to the database contents. This type of attack usually involves bypassing of user credentials to get unauthorized access.

Thus these are the major attacks which can compromise the authentication system present in a web application. Fig.9 shows a comparative study of the various attacks on the various methodologies already present and our novel algorithm which is the GAS system.

The results thus obtained are plotted in the form of graph as shown in the Fig.10. Out of six attacks considered for evaluation of our system it is found that our system is efficient enough to strongly resist four of them. The x-axis of the graph refers to the algorithm considered and y-axis refers to the number of attacks resisted categorically such as strongly resisted or weakly resisted or not

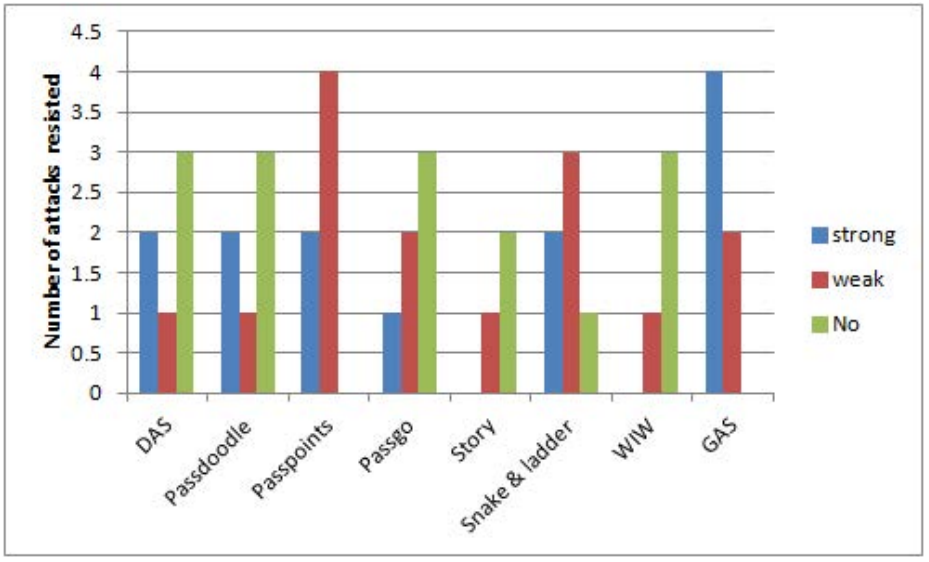


Fig. 10. Resistance Againt various Attacks

resisted. It is clearly found that our system has resisted better than the other existing algorithms.

7 Experimental Results

The experiment to test the usability of the proposed system was done by selecting 100 students. The students were exposed to use the Snake and ladder system and our proposed GAS system. They were asked to set passwords of length 6,8 and 10. The memorability of these passwords were tested in both the systems after the first week of setting the password and tabulated in Fig.12 and plotted in Fig.11. It is found that, it is easy to remember passwords set using our system than the snake and ladder system. Another important finding is that as the length of the password increases the memorability of the password decreases in case of both the systems.

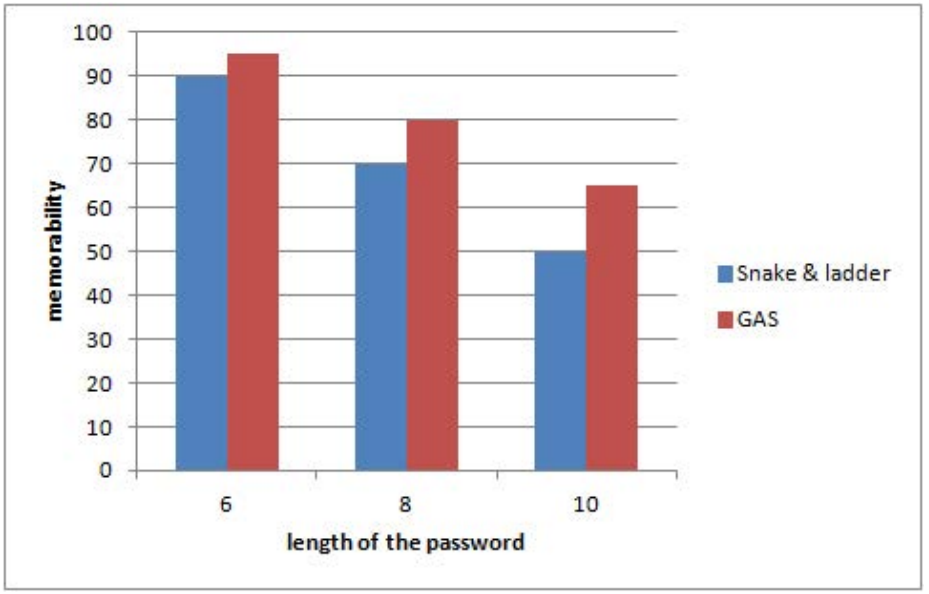


Fig. 11. Comparison of Memorability

Algorithm	Length of Password and Memorability(%)		
	6	8	10
Snake and Ladder	90	70	50
GAS	95	80	65

Fig. 12. Table for Comparison of Memorability

8 Conclusion

The research problem was to design and implement an authentication system which satisfies three goals. 1. Easy memorization of passwords: It has extremely become very difficult to memorize large number of usernames and passwords, so we decided to use a system where the password is given in the form of an image pattern which we call as Auth-Pattern. The system is primarily based on the Mnemonic principle [15] which states that "It is easy to remember what we see rather than what we hear". 2. Secure transmission of passwords: For this purpose we have designed the GAS authentication system is such a way that it produces random and encrypted passwords for every login the user makes and this random passwords gets transferred through the http requests . 3. Password Recovery in graphical authentication methods to make the authentication process more efficient and flexible to the users using it.

References

1. Chiasson, S., Stobert, E., Forget, A., Biddle, R., van Oorschot, P.C.: Persuasive Cued Click-Points: Design,Implementation, and Evaluation of Knowledge-Based Authentication Mechanism. IEEE Transactions on Dependable and Secure Computing 9(2) (March-April 2012)
2. Dass, S.C., Zhu, Y., Jain, A.K.: Validating a Biometric Authentication System: Sample Size Requirements. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(12) (December 2006)
3. Almuairfi, S.: IPAS: Implicit Password Authentication System. In: 2011 Workshops of International Conference on Advanced Information Networking and Applications
4. Salehi-Abari, A., Thorpe, J., van Oorschot, P.C.: On Purely Automated Attacks and Click-Based Graphical Passwords. In: 2008 Annual Computer Security Applications Conference
5. Tao, H.: Pass-Go, a New Graphical Password Scheme
6. Man, S., Hong, D., Matthews, M.: A Shoulder-Surfing Resistant Graphical Password Scheme WIW
7. Martinez-Diaz, M., Martin-Diaz, C., Galbally, J., Fierrez, J.: A Comparative Evaluation of Finger-Drawn Graphical Password Verification Methods. In: 2010 12th International Conference on Frontiers in Handwriting Recognition (2010)
8. Ma, Y., Feng, J.: Evaluating Usability of Three Authentication Methods in Web-Based Application. In: 2011 Ninth International Conference on Software Engineering Research, Management and Applications
9. Bicaki, K.: Towards Usable Solutions to Graphical Password Hotspot Problem. In: 2009 33rd Annual IEEE International Computer Software and Applications Conference
10. Malempati, S., Mogalla, S.: A Well Known Tool Based Graphical Authentication Technique. In: CCSEA 2011, pp. 97–104 (2011)
11. Doja, M.N., Kumar, N.: Image Authentication Schemesagainst Key-Logger Spyware. In: Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing
12. Hu, W., Wu, X., Wei, G.: The Security Analysis of Graphical Passwords. In: 2010 International Conference on Communications and Intelligence Information Security

13. Zhao, H., Li, X.: S3PAS:A Scalable Shoulder-Surfing Resistant Textual-Graphical Password Authentication Scheme. In: 21st International Conference on Advanced Information Networking and Applications Workshops, AINAW 2007 (2007)
14. Gao, H., Ren, Z., Chang, X., Liu, X.: A New Graphical Password Scheme Resistant to Shoulder-Surfing. In: 2010 International Conference on Cyberworlds
15. Zhao, S., Aggarwal, A., Kent, R.D.: PKI-Based Authentication Mechanisms in Grid Systems. In: International Conference on Networking, Architecture, and Storage, NAS 2007 (2007)
16. Tappenden, A., Miller, J.: A Three-Tiered Testing Strategy for Cookies. In: 2008 International Conference on Software Testing, Verification, and Validation
17. Juels, A., Jakobsson, M., Jakobsson, M.: Cache Cookies for Browser Authentication. In: Proceedings of the 2006 IEEE Symposium on Security and Privacy, S and P 2006 (2006)
18. Jrvinen, K., Tommiska, M., Skytt, J.: Hardware Implementation Analysis of the MD5 Hash Algorithm. In: Proceedings of the 38th Hawaii International Conference on System Sciences (2005)
19. Chen, Z., Guo, S., Duan, R., Wang, S.: Security Analysis on Mutual Authentication against Man-in-the-Middle Attack. In: The 1st International Conference on Information Science and Engineering, ICISE 2009 (2009)
20. Alicherry, M., Keromytis, A.D.: DoubleCheck: Multi-path Verification Against Man-in-the-Middle Attacks. IEEE (2009)
21. Rahman, K.A., Balagani, K.S., Phoha, V.V.: Making Impostor Pass Rates Meaningless: A Case of Snoop-Forge-Replay Attack on Continuous Cyber-behavioral Verification with Keystrokes. IEEE (2009)
22. http://en.wikipedia.org/wiki/Optimizationproblem