International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

Graphical Password Authentication for Web-Based System

Abdul Rehman Tahami, Ankita Arya, Abhishek Sharma, Alokit Sharma
CSE Department, Acropolis Institute of Technology And Research
Bypass Road, Square, Mangalya Sadak, Indore, Madhya Pradesh 453771
ankitaarya20550@acropolis.in
abdulrehmantahmani20168@acropolis.in
abhisheksharma20465@acropolis.in
alokitsharma20401@acropolis.in

ABSTRACT

Graphical password authentication is a method that uses passwords in the form of images or symbols. There can be various implementations based on its usage.

Digitization and automation across all industries have resulted in improvements in the efficiencies and effectiveness of systems and processes, and the tech sector is not immune to attacks through hackers.

Graphical Password authentication offers a systematic and secure alternative to the old text-based password designed for the authentication and security of the user.

KEYWORDS

Hashing, Encryption, Authentication, Graphical Passwords, Image Password Authentication

INTRODUCTION

Authentication is the process of determining that the person requesting a resource is the right person. Most of the authentication systems nowadays use an integration of username and password. The problem with the password is that it requires the user to remember it and it should be kept secret. Each authentication system has its guidelines and limitations like password length, passwords must contain alphanumeric and special characters. These passwords are mostly text-based. Either users use passwords that are easy to remember like license plate number, parent name, phone number, and sometimes their name which are very much predictable or complex passwords that they overlook so they might use the same password for different accounts or jot down their password somewhere. Moreover, users are vulnerable to various attacks. Text-based passwords faces security and usability matters.

Graphical passwords consist of choosing images or drawing symbols rather than entering textual characters. It was first described by Greg Blonder in 1996. The human brain is capable of processing and storing large volumes of graphical information with easiness. While it is very tough to recall a string of fifty characters, humans are capable of easily recalling the faces of people, places we visited, and things. These graphical records characterize millions of bytes of facts and thus make them available to big password spaces. Thus, graphical password schemes deliver a way of creating more human-friendly passwords while growing the level of security.

METHODOLOGY

The program will work as a graphical password authentication system that will store different images, and use them as a way of authentication for the convenience of the users.

The basic idea of the proposed system is that at the time of registration, the user enters his/her credentials(name, email, mobile number), after he fills in this information he is redirected to the next page where he is shown a set

of 16 images(4*4 matrix) from which he has to select at least 4 images (user must remember the pattern in which images were selected for the login) on the next page the same set is given to the user so to confirm the password by selecting the previously selected images. If the user selects the correct pattern then he is successfully registered and his password has been stored(hashed password). If both the fields don't match then the user is asked to create the password again.

pattern again. After 3 unsuccessful attempts, an email is sent to the user to change the password.

Users can also create their own set of images which can be used for their authentication. This feature can be accessed only in the change password section after login. The user provides 16 images which are first of all added to the database and then the user is asked to select the password pattern.

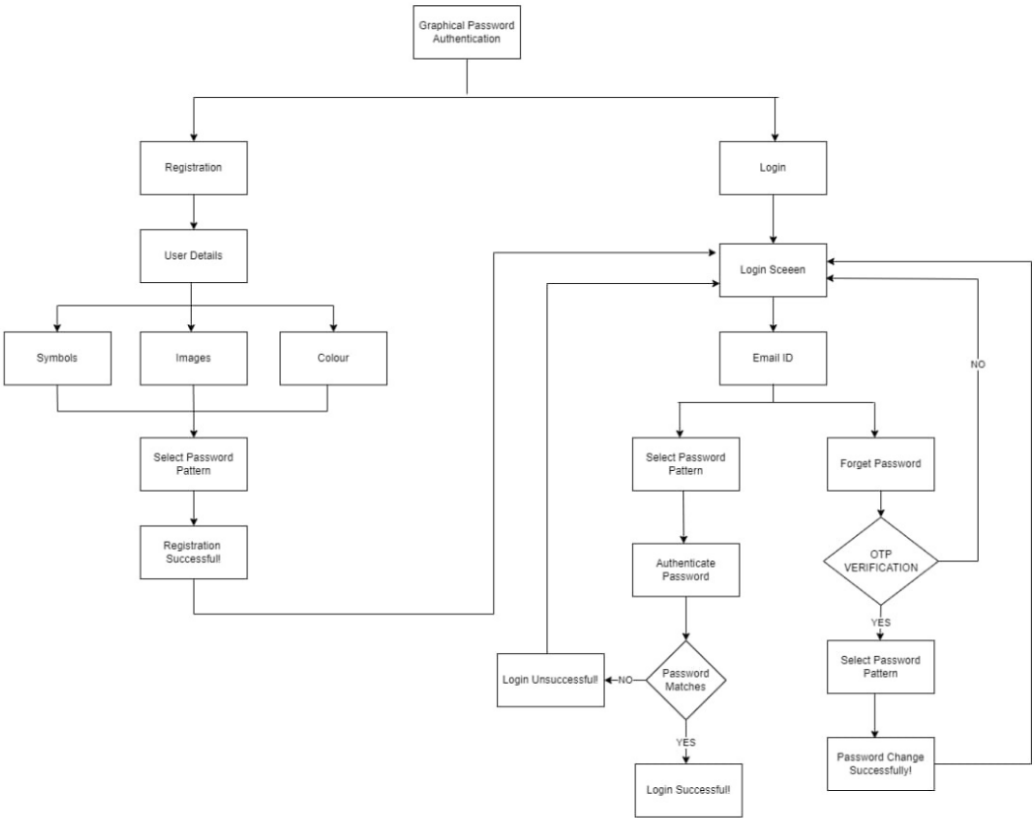

figure 1 : Flow of the System

After successful registration, a user can log in simply by providing his email id and then this id is sent back to the backend and the respective set of images are shown to the user for the password (these images get shuffled in order). The user selects the images in the same pattern he used at the time of registration and if the pattern is the same then the user is logged in else the user is asked to select his

HASHING

Our system uses hashing as a way of transforming the ids of different images into an encrypted value that can be sent into the database for verification.

The algorithm used in this project is very simple but unique. At the time of registration, the user is allotted an

image set id and only those images are displayed to him while creating the password. The user selects images in a specific pattern but the same image pattern is not stored in the database as the user's password rather we create a hash pattern and store it. The hash pattern is generated by creating the hash of the image id of the first image of the selected pattern and then this hashcode is hashed with the image_id of 2nd image of the pattern, and then the hashcode generated is hashed with the image_id of the next image in the pattern and this process is repeated until all the images in the pattern are hashed and this final hashcode is hashed with the user's email id so that a unique hashcode is always maintained. This hash is stored in the database as a password. When a user tries to log in he first provides his email id using which he has displayed the set of images associated with that particular email id. The user selects the correct password pattern, again we use the same procedure to find the final hashcode which is compared with the user hashcode stored on the database if it matches then the user gets logged in else the user is allowed another attempt. There are many hashing algorithms available like MD5, SHA-128, SHA-256, NTLM, and LANMAN.

The hashing algorithm we are using for encryption in our system is SHA 256 algorithm. SHA 256 is a part of the SHA 2 family of algorithms, where SHA stands for Secure Hash Algorithm. Published in 2001, it was a joint effort between the NSA and NIST to introduce a successor to the SHA 1 family, which was slowly losing strength against brute force attacks.

The significance of the 256 in the name stands for the final hash digest value, i.e. irrespective of the size of plaintext/cleartext, the hash value will always be 256 bits.

ADMINISTRATOR,

The administrator has a major role in the functioning of the system. The admin has to verify the “Proof of Identification”. Based on the verification, the admin can approve a user or reject the user’s registration request. The admin also has to notify the users about the various updates related to the system and also about the important notifications. The admin is also responsible for maintaining the credentials of the user in the database. The admin is also responsible for inserting the default image sets in the database.

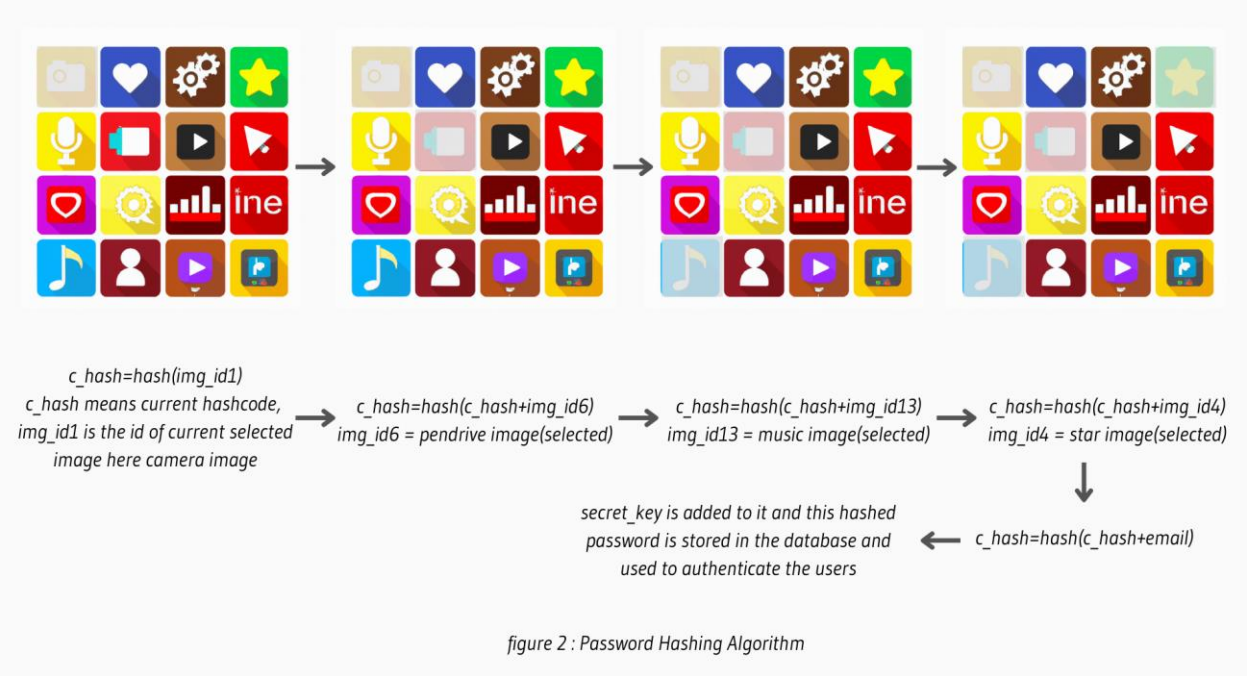

figure 2 : Password Hashing Algorithm

USER

The user in the system can register, login or change the password of the system. To register in a web system the user has to input his username, email id, phone number and has to select 4 images in a specific order to form a password, he also has to confirm it on the next page.

DEVELOPMENT

Being an important phase of software development, the design of the application's architecture was highly considered.

The application has a front end that acts as an interface to the users of the application and a backend that aids in the management of the application. The features of both are listed below.

Front-End

- The Front End of the Application provides several functionalities to the users:
- Users can login with a secure system.
- They can change the password if they wish.
- It allows users to register themselves on the website.

Back-End

The Back End of the Application is used by the administrator:

- Administrators can add/delete/edit the user credentials.
- He can also change the launch status of Courses that will be affected in the Front End.
- Administrators can update the database.
- He can change the status of a User.

The programming languages used in the development of the software were as follows:

Html was used in making the front end of the software. Javascript was used in making the logic of the software.CSS was used to design and beautify the external look of the software.ExpressJS was used due to its various features. These Languages were used due to their popularity, how easy they were to learn and use and their compatibility with each other. Microsoft Visual Studio Code was used as the code editor to program these scripts. MongoDB and Node.js were used to develop the server and the database for the website. During implementation, most of the functionalities were developed as small chunks of code.

We also did intensive testing and debugging to make sure our program was working correctly.

RELATED WORK

A Graphical Password Authentication System

The proposed authentication system works as follows. At the time of registration, a user creates a graphical password by first entering a picture he or she chooses. The user then chooses several point-of-interest (POI) regions in the picture. Each POI is described by a circle (centre and radius). For every POI, the user types a word or phrase that would be associated with that POI. If the user does not type
any text after selecting a POI, then that POI is associated with an empty string. The user can choose either to enforce
the order of selecting POIs (stronger passwords) or to make
The order is insignificant.

Graphical Password Authentication

In the proposed system, a user freely chooses a picture, POIs and corresponding words. The order and number of POIs can be enforced for stronger authentication. Together, these parameters allow for a very large password space In this extended abstract, the creator proposed a simple graphical password authentication system. The system combines graphical and text-based passwords trying to achieve the best of both worlds. It also provides multi-factor authentication in a friendly intuitive system. We described the system operation with some examples and highlighted important aspects of the system

Graphical Password Authentication System

In this extended abstract we are trying to make our authentication system more user friendly and also we have tried to implement a mature & fast Shoulder Surfing Resistant Mechanism. We have considered both methods: text-based and graphical-based systems and tried to reduce the efforts required by the end user to remember passwords. A look at the advancement in technology over the past few years tells us that the next era will have system security at its core. Thus Graphical Password may be adapted in future as a major authentication system.

PassPoints: Design and longitudinal evaluation of a graphical password system

In this paper, it is extended Blonder's idea by eliminating the predefined boundaries and allowing arbitrary images to be used. The image could be any natural picture or painting then it contains several possible click points. As a result, a user can picture or paint and then it contains several possible click points.

EXISTING SYSTEM DRAWBACKS

The main drawbacks of the existing systems are that they may be vulnerable because the set of images is predefined and is publicly accessible by the attacker and another main drawback is the story-based image selection which might be difficult for a user to memorize the order of story and story can be guessed by seeing the sequences of images. So our solution is that we are not storing the pattern or images selected by the user as a password, we are storing the hash which is generated by each image by using some hashing algorithm (SHA256, MD5 etc.) and this hash will be saved and stored in the database. So if the hackers hack the database then also the hacker is not getting the password or pattern directly to crack the password the hacker has to crack the hash code which is difficult to crack. We also proposed the solution for the shoulder surfing attack, the solution is that we are shuffling the set of images in each login so that if the attacker tries to make the possible pattern then the pattern will not match because all the images are shuffled.

RESULT AND DISCUSSION

This system aids in increasing the security and protection required in this system. It also gives users many different advantages in many different aspects In addition, it ensures that the authentication is user-friendly and authentic.

CONCLUSION

This system is designed for the actual implementation of Graphical Password Authentication to allow users to easily access their account just by remembering an image pattern.

REFERENCE

Akshay KPathik Nandi, Dr Preeti Savant, Graphical Password Authentication System

Ahmad Almulhem, A graphical password authentication system

Towseef Akram, Vakeel Ahmad, Israrul Haq, Monisa Nazir, Graphical Password Authentication

Arash Habibi Lashkari, Abdullah Gani, Leila Ghasemi Sabet and Samaneh Farmand, A new algorithm on Graphical User Authentication(GUA) based on multi-line grids

arode, Sanket Mistry and Saurabh Chavan, Graphical Password Authentication System

Geeta M. Rane, Graphical Password Authentication: Methods and Schemes

Wiedenbeck, S., Waters, J., Birget, J.-C., Brodskiy, A., & Memon, N., PassPoints: Design and longitudinal evaluation of a graphical password system

Vishal Kolhe, Vipul Gunjal, Sayali Kalasakar, Pranjal Rathod, Secure Authentication with 3D Password

ACKNOWLEDGEMENT

We would like to acknowledge the students and faculties of Acropolis Institute of technology and research for their efforts in making of this work