IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 5, May 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.165

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005166|

Robust System with Secured Password Generator

Prof. Megha Dhotay, Aditya Nevase, Shubham Mehta, Rohan Aware, Siddharth Mahankal

Department of Computer Engineering, MAEER's MIT Polytechnic, Pune, Maharashtra, India

Department of Computer Engineering, MAEER's MIT Polytechnic, Pune, Maharashtra, India

Department of Computer Engineering, MAEER's MIT Polytechnic, Pune, Maharashtra, India

Department of Computer Engineering, MAEER's MIT Polytechnic, Pune, Maharashtra, India

Department of Computer Engineering, MAEER's MIT Polytechnic, Pune, Maharashtra, India

ABSTRACT: Everyone who uses various online services is concerned about security and privacy in order to protect personal information from outsiders. The password authentication system is one of many authentication systems available for the security of an individual's data. Password security and authenticity have become crucial and necessary themes as a result of the increase in information sharing, internet popularity, electronic commerce transactions, and data transfer. However, ensuring the password's strength is also required.Thissystemwill allow our users to generate distinct passwords in favor of their individual conditions by simply going through 3 steps our very first step is to gather non-confidential information/details of our users through which our idea behind this is to put these data into the logic of creating password the next step is to display several conditions, criteria's as the user has to select some according to their needs and in the final step ofthe process, we'd display passwords based on their favorable consideration in case the user does not find the password equitable we also offer an option to refresh the suggestion and display many others. Also, we provide a facility for storing passwords in our database particularly for each and every user individually so that they don't have to go through the need of resetting them. In our project, we are adding one more feature 'Strength Meter'. Using this feature users can test their personal passwords if they are strong enough or not and if they are not as strong as per the strength meter they can generate one.

KEYWORDS: authentication, protection, distinct, non-confidential, criteria, favourable, equitable, strength meter.

I. INTRODUCTION

Password security and authenticity have become crucial and necessary themes as a result of the increase in information sharing, internet popularity, electronic commerce transactions, and data transfer. However, ensuring the password's strength is also required. A number of technologies and approaches have been proposed in recent years to make authentication more usable, convenient, and safe, including smart cards, RFID cards, USB tokens, and graphical passwords. While each of these technologies has its own set of benefits and may be better suited for use in a certain environment or for a specific application, text-based passwords are still the most often used authentication method. This is due in part to the fact that text-based passwords do not require any specific hardware and are simple to type for both end users and system developers[5]. For decades, researchers have documented the multiple issues with text passwords - passwords are easy to guess, difficult to remember, and difficult to change.It's easy to steal and vulnerable to replay and observation assaults. Alternatives such as biometrics, graphical passwords, hardware tokens, and federated identity have received significant academic attention; but, text passwords remain the most common method of authenticating users to computers, and are expected to stay so for the foreseeable future. As a result, a better knowledge of text passwords is still necessary[6]. The compromise between establishing passwords that are both difficult to hack and useable complicates secure password generation. Users may find it difficult to remember truly random passwords, and user-chosen passwords may be highly predictable. Password policies aim to strike a balance between these two purposes by requiring users to include additional complexity in their passwords, such as requiring users to include an odd character or to use passwords that are longer than eight characters. hampered by an ill-defined understanding of their actual effectiveness against real-world attack approaches, as well as the users' circumvention strategies A policy requiring at least three digits in a password, for example, will frequently result in the user just appending "123" to the end of an unsafe password. These tactics will be known to a sufficiently proficient password cracker[7].Despite widespread worries regarding the level of security provided by passwords, they remain a widely used means of user authentication. There are a number of potential successor technologies, including biometrics and

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005166|

trusted personal devices, but it appears unlikely that passwords will be phased out very soon..Finding solutions to improve the usage and management of passwords remains a critical topic, given their current and probable future widespread use. We'll concentrate on a practical issue here: how to make password-based user authentication on a website more secure and convenient. [8]. Passwords are just as secure as the process of creating them. Random passwords Only random password generators can generate them. However, the produced strings may be difficult to remember, particularly if the user has multiple accounts and thus multiple passwords. Humans can readily construct memorable passwords, but this also raises the problem that the method of creating them can be guessed, such as by following certain language structures[10] or themes [11].. Human-created passwords are thus less safe than random passwords. There are password development guidelines to assist users in creating strong passwords. However, given the range of memorization approaches, such guidelines, as those provided in [12], may not be useful for all users. Instead of prescribing a fixed number of characters from some character set, experts generally propose [13] a system for analysing each password against some measure and rejecting the weak ones. [9]

III. MOTIVATION & OBJECTIVES

- Whenever we have any important work to do on the website our main aim is to completethat work as soon as possible.
- In this urgent work we face one issue of setting passwords which sometimes getsmore hectic as every website have there different types of conditions to set a password.
- In day-to-day life browsing has become a daily job so people concern more aboutthe security of their passwords to keep their data on the site safe and secure.
- So in concern of security and time saver, we came up with an idea of creating an auto-generator that is flexible, reliable, and user-friendly application so that users won'thesitate in regards of sharing their data shared along through our application andwould generate and store various passwords with us.
- This system will allow our users to generate distinct passwords in favor of their individual conditions by simply going through 3 steps our very first step is to gather non-confidential information/details of our users through which our idea behind this is to put these data into the logic of creating password the next step is to display several conditions, criteria as the user has to select some according to their needs and in the final step or process, we'd display passwords based on their favourable consideration in case the user does not find the password equitable we also offer an option to refresh the suggestion and display many others. Also, we provide a facility of storing passwords to our database particularly for each and every user individually so that they don't have to go through the need of resetting it. In our project, we are adding one more feature 'Strength Meter'.

IV. BLOCK DIAGRAM

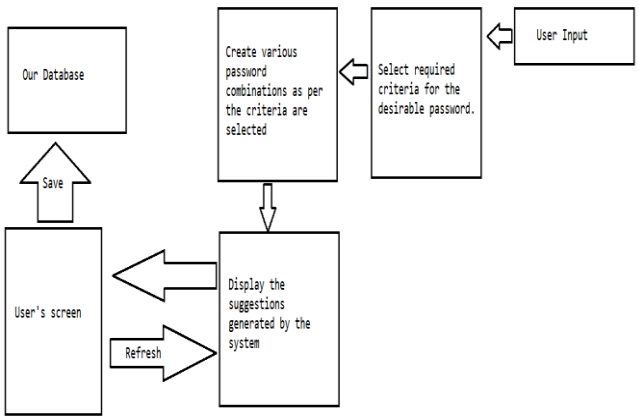

Fig.3.1 Password generator through user inputs.

Fig.3.1 represents our password generator using user inputs.
User Input: In this very first step, we accept inputs such as full name, dob, birthplace, etc.
Criteria: Next, we present criteria to be selected by the needs of the user.
User Display:Next, we display several combinations according to the selected criteria.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005166|

Cloud Database: The combination that the user likes they can opt to save it on our database, the user can keep refreshing the suggestions until they find a satisfying one.

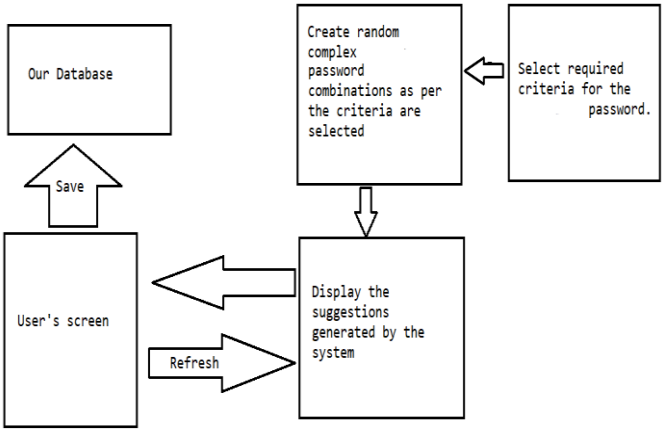

Fig.3.2 Random password generator.

Fig.3.2 represents our random password generator.
RandomPass: Users which are not in favor of providing their personal details may use our random password generator. In this system, we generate regex combinations according to the provided criteria.
Cloud Database: Also, here they are able to save the password in our encrypted database.

V. EXISTING SYSTEM

Password structure plans are used to help users to create passwords. Various password generators have been developed to date and some of them are listed below;
[1]The text password is a popular method of user authentication. Users confront a significant challenge in managing a large number of site-specific and strong (i.e. non-guessable) passwords. A password generator, or client-side technique that generates (and regenerates) site-specific strong passwords on demand with minimal user input, is one solution to address issue.AutoPass, a revolutionary password generating system, is described and analysed in detail in this study. AutoPass was created to address flaws that had been noticed in earlier suggested password generators, and it uses unique approaches to do so. Unlike almost all other schemes, AutoPass allows for the creation of passwords that meet critical real-world needs such as required password updates, the usage of pre-specified passwords, and passwords that meet site-specific requirements. [2]An growing number of passwords are being requested of computer users for use with host accounts, email servers, e-commerce sites, and online banking services. Unfortunately, the password entropy that people can easily memorise appears insufficient to keep unique, safe passwords for all of these accounts, and this trend is expected to continue as the quantity of passwords (and the adversary's computational capability) grows.In this research, we present a method for computing secure passwords for an unlimited number of accounts using a stronger cryptographic hash function and only needing the user to remember a single short password. This technique is fully client-side; no changes to the server are required. Our solution is both highly resistant to brute force attacks and essentially stateless, allowing users to retrieve their passwords from any place as long as they can run our application and remember a short secret.We believe that the mix of security and ease will persuade users to adopt our plan. We go over the details of our algorithm's creation, compare its merits and drawbacks to those of previous approaches, and present Password Multiplier as an extension for the Mozilla Firefox web browser. [3]Everyone who uses various online services is concerned about security and privacy in order to protect personal information from intruders. The password authentication system is one of many authentication systems available for the protection of individual data. Because of the rise in information sharing, internet use, electronic commerce transactions, and data transfer, both password security and authenticity have become critical issues. However, ensuring the password's strength is also required. As a result, all cyber experts advise using complex password patterns. However, because of the intricate patterns, individuals frequently forget their passwords. Unlike other random password generators, we propose a unique approach in this paper that will generate a strong password. This password will be created using the information provided by the users (some words and numbers) so that they do not have to remember it. Various experiments using simulated input data were used to test our system. We also tested the strength of the passwords generated by our generator using four popular online password checkers.The trustworthiness of our created passwords is completely good, according to our

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005166|

tests. We also tested if our generated passwords can withstand the "Dictionary assault" and the "Brute Force attack," two password cracking techniques. Our system was built using the Python programming language. We intend to expand our activities in the near future. developing an online free-to-useuser interface. Our method generates passwords that are not only user-friendly, but also feature the majority of the qualities of strong and non-crackable passwords. [4] Most alphanumeric password generators join string tokens without addressing semantics, which is important because semantics help people remember password strings..Using a context-free grammar, we present a formal technique for generating passwords with semantic annotations. A context-free grammar is a method for producing an endless number of strings or objects. Users can indicate their interests (for example, music or movies), and phrases from these interests are combined with three numbers and one unique character to create memorable passwords..PassGen is a software tool that implements this concept. PassGen takes the user's preferences as input and creates n passwords for them to pick from.

MODEL ALGORITHM

STEP 1: The userneeds to login into our system for custom passwords. If he/she does not have an account they have a choice to create a new account.
STEP 2: If the user does not want a custom password they might opt for the random password option.
STEP 3: The user who opts for a custom password now has to fill in the credentials of the personal details.
STEP 4: After that, they have to choose the desired criteria for their password.
STEP 5: Using those credentials we display suggestions of password combinations. The user has a choice to refresh the suggestion if not satisfied. They can check the strength of the password too using our strength meter.
STEP 6: After the user decides upon a satisfactory password, he can save it into our database so that if even they forget the password they can fetch it from our database. Access to the database is secured with user verification through OTP.

VI. PROPOSED METHODOLOGY

A password generator is a feature on an end-user platform that allows users to authenticate to a remote server using their password (assumed to be a web site). It produces a location password for authentication on demand.Obviously, the web site that authenticates the user must have access to this password.[8] Our system proposes some features that are not present in the existing systems.

A user-specific password generation: This provides the user with a custom password by using their credential inputs that are convenient and desired.

Inbuilt Strength Meter: This allows users to check the strength of password security without any need to access other websites for this particular task.

Password Retention:In case users forget their passwords they can log in and access their site-specific passwords which are stored in our database securely.

The entire application will be built on the Angular Framework. HTML, SCSS, and JavaScript will be used to design the Frontend. We will cover the whole Backend by Node JS and MongoDB will be used as a database.

VII. ACKNOWLEDGEMENT

We wholeheartedly convey our profound appreciation with regard to our respected guide Prof. Megha Dhotay for hervaluable contribution, recommendations, and counseling which helped us numerous times to complete the task in a flourishing way. Her advice has helped us on every step while completing this task which boosted our confidence.

VIII. CONCLUSION

This technique was proposed to provide convenience and password security. Users gain from this strategy in terms of security and secrecy. The password generated using the suggested technique is more secure than user-created passwords since it is chosen from a wide pool of passwords.

REFERENCES

[1] IEEE Xplore, 23-26 Oct 2017- AutoPass: An automatic password generator by Fatma Al Maqbali , Chris J Mitchell
[2] Association for Computing Machinery, 10 May 2005- A convenient method for securely managing passwords by J. Alex Halderman, Brent Waters, Edward W. Felten

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005166|

[3] IEEE Xplore, 17-19Oct 2019-Strong Password Generation Based On User Inputs by Farhana Zaman Glory, Atif UI Aftab, Olivier Tremblay-Savard, Noman Mohammed
[4] IEEE Xplore, 21-22Nov 2019-Syntactic Generation of Memorable Passwords by Abejide Ade-Ibijola, Blessing Ogbuokiri
[5] Richard Shay, Saranga Komanduri, Patrick Gage Kelley, Pedro Giovanni Leon, Michelle L Mazurek, Lujo Bauer, Nicolas Christin, and Larrie Faith Cranor. Encountering stronger password requirements: user attitudes and behaviors. In Proceedings o f the Sixth Symposium on Usable Privacy and Security, ACM, 2010
[6] Michelle L Mazurek, Saranga Komanduri, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Patrick Gage Kelley, Richard Shay, and Blase Ur. Measuring password guessability for an entire university. In Proceedings o f the 2013 ACM SIGSAC conference on Computer & communications security, ACM, 2013.
[7]Matt Weir, Sudhir Aggarwal, Michael Collins, and Henry Stem. Testing metrics for password creation policies by attacking large sets of revealed passwords. In Proceedings o f the 17th ACM conference on Computer and communications security, ACM, 2010.
[8] A. H. Karp, "Site-specific passwords," HP Laboratories, Palo Alto, Tech. Rep. HPL-2002-39 (R.1), May 2003.
[9] Kirsi Helkala and Einar Snekkenes. Password generation and search space reduction. 2009.
[10] A. Narayanan and V. Shmatikov, "Fast dictionary attackson passwords using time-space tradeoff," in Proc. of 12thACM conference on Computer and communications secu-rity, 2005, pp. 364–372.
[11] F. Monrose and M. K. Reiter, "Graphical passwords," inSecurity and Usability: Designing Secure Systems thatPeople Can Use. O'Reilly, 2005, ch. 9, pp. 161–179.
[12] S. Gaw and E. W. Felten, "Password management strate-gies for online accounts," in Proc. of SymbosiumOnUsable Privacy and Security, 2006, pp. 44–55.
[13] E. Gehringer, "Choosing passwords: security and humanfactors," in Proc. of International Symposium on Technol-ogy and Society, 2002, pp. 369–373.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.165

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details