An Improved Hashing & Salting Based Password Security Scheme- A Review

Teena Kashyap[1], Er. Divya[2]
Department of Computer Science and Engineering
[1]Research Scholar, RPIIT Bastara, Karnal
[2]Assistant Professor,RPIIT Bastara, Karnal

Abstract-*Password security is a major issue for any authenticating process and different researches in past have proposed different techniques like hashing, salting, honeywords to make the process most secured. The proposed process performs a unique hashing algorithm with very low time complexity as most of the steps will involve simple binary operations. Authentication is a methodology to ascertain that a particular entity is the desired one and not some intruding entity. So, in modern digitized world, it is a major requirement for all applications whether it is a web application, desktop app or a mobile app. So, there are numerous people across globe, who constantly try to break open these securities and they do succeed in many cases. So, safe authentication is a major problem for IT industry. We have multiple authentication schemes like face recognition, voice recognition, retina detection, finger prints and password based. The base research by BinojKoshy [4] describes a salting based hashing process called Chameleon Salting Hashing where he successfully implemented a password security scheme to prevent various attacks. Another research by Imran Erguler[12], implements protection by creating a list of words called honeywords. We propose an advanced hashing technique combined with salting to improve the security of passwords and use the concept of honeywords to form a blacklist in this case. The salting process is followed by a hashing process which is basically a bit-conversion process. This makes this very time efficient in comparison to contemporary technologies. The research is going to be implemented in Java programming and the results will be validated by comparing the time efficiency and security aspect with the base researches.*

Keywords- Password Security, Hashing, Salting, Honeywords.

I. INTRODUCTION

Once a password file has stolen with the help of password cracking techniques it is easy to get most of the plaintext passwords. According to this, there are two issues that should be acknowledged to overcome these security problems: First is passwords must be protected by taking relevant providence and storing with their hash values enumerated through salting mechanisms. Hence, it must be hard to reverse hashes to acquire plaintext passwords. The second point is that a secure system should detect whether a password file is revealed or not to take relevant actions.

Earlier, the researchers used decoy passwords against hashed password databases to detect attacks. In this the real password is stored with several honeywords for each user account in order to sense imitation.

In this research, we work towards creating a technique, where it almost becomes impossible for the intruder to break the security. If intruder has unauthorized access to the datafile, then the system will generate an alarm to identify the attacker. We provide here a technique which creates a list of passwords called as blacklist using different permutations of current password and the correct password in the list is not known to any user.

II. BASE RESEARCH

There have been numerous researches in the recent past for detection of fraudulent transactions. The following researches by BinojKoshy [4] and Imran Erguler[12] use chameleon salting and honeywords generation respectively. We propose a technique that combines the two techniques and provides a more robust way of password protection system.
Chameleon Salting[4] is an innovative initiative by BinojKoshy et. al to enhance security to the end user, where the end user is provided a secure environment for entity authentication even without the user having to implement or change the way the application is being used. Its implementation will lower impact of a loss, thereby providing better protection to the customers. The author introduced the innovative, original, no cost solution to provide a more robust salting technique to ensure high level of security to the end user; in applications that require logging-in by means of 'Username' and 'Password' as part of entity authentication.

The solution proposed by Koshy[4] can be implemented with least effort and with no additional cost; hence the project implementer will be able to convince the management hierarchy to adopt the 'Chameleon Salting'

solution. Further, the solution will only improve security and also ensure better protection to the end user. The concept of honeywords by Erguler[12] provides a secure way of protection but it is vulnerable to brute force attacks making it less robust.

III. LITERATURE REVIEW

Warutet. al in [1] carried on his research to mitigate the issue of weak passwords by proposing a context-based password strength meter. The authors conducted a randomized experiment on Amazon MTurk and observed the change in users' behavior. The results showed that the proposed method was significantly effective. Users exposed to this password strength meter were more likely to change their passwords after seeing the warning message, and those new passwords were stronger. Furthermore, users were willing to invest their time to learn about creating a stronger password, even in a traditional password strength meter setting. The findings suggested that simply incorporating contextual information to password strength meters were an effective method in promoting more secure behaviors among end users. The study showed that providing additional contextual information along with warning messages displayed by password strength meters could enhance understanding among users, resulting in improved password generating behaviors.

Steffen et. al in [2] researched on the concept of graphical passwords. According to authors, Graphical passwords offered a more memorable alternative to traditional, text-based passwords. Among current contenders, cued-recall based click-point or gesture centered authentication systems like Microsoft's picture gesture authentication (PGA) had been commercially more successful than recognition based systems (e.g., PassFaces). One perceived drawback of graphical authentication systems in general and especially recognition based authentication was the assumption that graphical authentication was slower and thus less user-friendly than traditional password entry via keyboard. This paper addressed these concerns and demonstrated a lower limit for recognition-based password entry times achievable with sufficient practice. While slightly slower than traditional keyboard based passwords, the entry speed of often-used graphical passwords was shown to reach 10 bits/s in an optimized configuration, which was sufficient for everyday use (3-6s per authentication sequence @ 36 bits) and exceeded the reported speed of similarly secure text-based passwords on non-traditional devices using virtual keyboards. As the data clearly showed, well-designed recognition based authentication systems had the potential to support fast password entry speeds similar to current keyboard or cue-based password systems.

Bruno et. al in [3] described the progress on building agent-based models of human behavior with passwords, and the authors demonstrated how these models reproduced phenomena shown in the empirical literature. According to authors, Effective reasoning about the impact of security policy decisions required understanding how human users actually behaved, rather than assuming desirable but incorrect behavior. Simulation could help with this reasoning, but it required building computational models of the relevant human behavior and validating that these models matched what humans actually did. The authors discussed the applicability of agent-based simulations to security policy design, expanded the password simulation to more accurately evaluate the security afforded by password policies, and validated this model, to some extent, with empirical data from the password security literature.

Binojet. al in [4] introduced Chameleon Salting, an innovative initiative to enhance security to the end user, where the end user was provided a secure environment for entity authentication even without the user having to implement or change the way the application was being used. Its implementation lowered impact of a loss, thereby providing better protection to the customers. According to authors, Entity authentication by means of keying in Username and Password had been adopted as the de-facto standard in many Internet based and enterprise based applications the world over. The service providers had always been bogged down by Information Security breaches and Cyber Attacks in the recent past, and the industry had time and again stood against odds and had been able to imbibe the best of security practices to beat back security breaches especially in the domain concerning Entity Authentication. Authentication process needed to be made robust and hardened; Salting of password was practiced by the application service provider to ensure security to the end-customer. Technology today offered protection and also provided the tools to the unscrupulous elements by means of cracking Username and Password. No amount of technology and processes was said to be adequate to keep away the perpetrator in this domain.

Katha Chanda in [5] dealt with password security, a close look at what went into making a password strong and the difficulty involved in breaking a password. The first few sections discussed related work and proved graphically and mathematically the different aspects of password securities, overlooked vulnerabilities and the importance of passwords that were widely ignored. This work described tests that were carried out to evaluate the resistance of passwords of varying strength against brute force attacks. It also discussed overlooked parameters such as entropy and how it tied in to password strength. This work also discussed the password

composition enforcement of different popular websites and then presented a system designed to provide an adaptive and effective measure of password strength. This paper contributed toward minimizing the risk posed by those seeking to expose sensitive digital data. It provided solutions for making password breaking more difficult as well as convinced users to choose and set hard-to-break passwords.Erguler [12] scrutinized the honeyword system and presented some remarks to highlight possible weak points. Also, the author suggested an alternative approach that selected the honeywords from existing user passwords in the system in order to provide realistic honeywords – a perfectly flat honeyword generation method – and also to reduce storage cost of the honeyword scheme. honeywords (decoy passwords) to detect attacks against hashed password databases. For each user account, the legitimate password was stored with several honeywords in order to sense impersonation. If honeywords were selected properly, a cyber-attacker who stole a file of hashed passwords cannot be sure if it was the real password or a honeyword for any account. Moreover, entering with a honeyword to login will trigger an alarm notifying the administrator about a password file breach. At the expense of increasing the storage requirement by 20 times, the authors introduced a simple and effective solution to the detection of password file disclosure events.

IV. PROBLEM FORMULATED

Password plays an important role in various applications like internet services, net banking, ATM machines etc. But passwords are not much safe to provide the security to the users because of large no of attacks [3]. Here, we investigate the security of the proposed model against some possible attack which is as follows:

Brute Force Attack

Brute force is a hit and trial method used by attackers to decode encrypted data such as passwords or data encryption standard keys using brute force. It consists of systematically checking all possible combinations of keys or passwords. But in this technique, Brute force attack is not possible due to high complicated hashing. Brute force attacks are very time consuming because searching a hash from all possibilities is a time taking process. Suppose user enters 8 characters password then it converts into 16 bit hexadecimal string. One hexadecimal contains 16 possible characters. So, the 8 characters passwords have 16^{16} combinations which are practically impossible to check.

DoS Attack

Denial- of- service attack is an attempt to make a machine or networkresources unavailable to its intended users. The suggested model mostly focuses on minimizing the DoS vulnerabilities. In this attack the attacker does not have the password files and their contents. His main purpose is to trigger a false alarm and to raise a black list alarm situation that is depending on the scheme some or all parts of the system may be out of service or disabled unnecessarily. Suppose that the attacker has knowledge of m + 1 username and even he has password file then still attacker is not able to get the real password. And if attacker attempt to login with any password from black list then it raises an alarm for the application and the application can block that particulars user or address.

Password Guessing

In this attack, we suppose that the attacker has stolen password files from the main server and also obtained plaintext passwords by reversing the hash values. Extracted password files gives < username; password > pairs to the attacker, but it is not directly connected to a specific username. By just examine this, the attacker cannot exactly determine which password belongs to which user account. And if the attacker randomly takes an account from the password file and then tries to login with a guessed password, then the success will depend on: First, the selected account is not a decoy account that is it does not contain any mask words. Second, guessing the correct password from the black list. Otherwise, the attacker will be caught by the system and it raises an alarm for the application.

V. PROPOSED METHODOLOGY

The proposed methodology is a hybrid concept of salting and a simple but secure hashing technique to finally generate a password protection scheme which is not vulnerable to any of the attacks including brute force attack which the base research could not counter. The process can be divided into following steps.

Step 1. User enters the password into the password field.
Step 2. The salting is done on this password to add some bits to the right and left of the entered password.
Step 3. The salted password is then taken through the 5-step hashing process using multiple hash keys which are generated randomly.
Step 4. The final result generated is the hashed string which is matched with the hashed string blacklist retrieved from the database.

Step 5. If the generated hashed string matches the middle element in the blacklist then, the user is authenticated.
Step 6. If the user enters the password which matches one of the elements in the blacklist, then it denotes an attack and the system generates the alarm.
Step 7. Any unsuccessful match with any element in blacklist causes an unsuccessful login.

The steps involved in the signup process are as follows:
Step 1. The user preferred password is taken through the salting and hashing process detailed below.
Step 2. Different hash keys are used to generate a list of hashed strings which collectively joined to form what is called blacklist. This blacklist contains the middle element as the user password hash string.

This blacklist is stored into the database for the particular user.

VI. CONCLUSION AND FUTURE SCOPE

In modern digitized world, the one thing which has got maximum focus is information security. This is possible only by proper authentication mechanisms and impeccable protection to security credentials.We have multiple authentication schemes like face recognition, voice recognition, retina detection, finger prints and password based. The password based authentication is still the most favored and trusted way of authentication due to its simple implementation. In the past, many different techniques based on salting, hashing, quantum cryptography and split key methods have been proposed. The base research by BinojKoshy [4] describes a salting based hashing process called Chameleon Salting Hashing where he successfully implemented a password security scheme to prevent various attacks. Another research by Imran Erguler[12], implements protection by creating a list of words called honeywords.We propose an advanced hashing technique combined with salting to improve the security of passwords even though the intruder gets an access to the database. The hashing technique is based on some simple bit conversion methods making it highly efficient in terms of time consumption.The effectiveness of proposed technique can be demonstrated by comparing the results with traditional techniques on basis of time complexity and security aspect.

REFERENCES

[1] WarutKhern-am-nuai, Weining Yang and Ninghui Li, "Using Context-Based Password Strength Meter to Nudge Users' Password Generating Behavior: A Randomized Experiment", 50th Hawaii International Conference on System Sciences | ieee 2017
[2] Steffen Werner, Christopher Hauck, and Marshall Masingale, "Password Entry Times for Recognition-based Graphical Passwords", Human Factors and Ergonomics Society 2016 Annual Meeting, ieee 2016
[3] Bruno Korbar, Jim Blythe, Ross Koppel, Vijay Kothari and Sean Smith, "Validating an Agent-Based Model of Human Password Behavior", AAAI Conference on Artificial Intelligence, Artificial Intelligence for Cyber Security: IEEE2016.
[4] BinojKoshy, NilayMistry and Khyati Jain, "Chameleon Salting: The New Concept ofAuthentication Management", IFS annual Journal 2016.
[5] Katha Chanda, "Password Security: An Analysis of Password Strengths and Vulnerabilities", I. J. Computer Network and Information Security, 2016, 7, 23-30
[6] Blasé Ur, "Supporting Password-Security Decisions with Data", PNC Center for Financial Services Innovation and Microsoft Research. 2016.
[7] Blake Ross, Colin Jackson, Nick Miyake, Dan Boneh and John C. Mitchell, "Stronger password authentication using browser extensions", NSF Science Project 2005.
[8] R.V.Sudhakar, A. Mruthyunjayam, D. SugunaKuamari, M. Ravi Kumar and B.V.S. Ramesh Babu, "Improving Login Authorization by Providing Graphical Password (Security)", Int. Journal of Engineering Research and Application, Vol. 3, Issue 6, Nov-Dec 2013, pp.484-489
[9] Joseph Bonneau, Cormac Herley, Paul C. Van Oorschot and Frank Stajano, "Passwords and the Evolution of Imperfect Authentication", Communications of the ACM vol. 58 no. 7, July 2015 pp. 78–87.
[10] Muhammad Adeka, Simon Shepherd and RaedAbd-Alhameed, "Resolving the Password Security Purgatory in the Contexts of Technology, Security and Human Factors", International Conference on Computer Applications Technology (ICCAT),2015.
[11] Blase Ur, Sean M. Segreti, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, SarangaKomanduri, Darya Kurilova, Michelle L. Mazurek, William Melicher and Richard Shay, "Measuring Real-World Accuracies and Biases in Modeling Password Guessability", 24th USENIX Security Symposium August 12–14, 2015
[12] Imran Erguler, "Achieving Flatness: Selecting the Honeywords from Existing User Passwords", IEEE Transactions on Dependable and Secure Computing 2015
[13] Harikrishan T and C Babu, "Cryptanalysis of Hummingbird Algorithm with Improved Security and Throughput", International Conference on VLSI Systems, Architecture, Technology and Applications, IEEE 2015

[14] YevgeniyDodis, SiyaoGuo and Jonathan Katz, "Fixing Cracks in the Concrete: Random Oracles with Auxiliary Input, Revisited", NSF-REU 2015.
[15] Emin ˙Islam Tatlı, "Cracking more Password Hashes with Patterns", IEEE Transactions on Information Forensics and Security 2015.
[16] PoojaKolte, ReshmaGutal and PriyankaBhairat, "An Efficient Password Security Mechanism Using Two ServerAuthentication and Key Exchange", IJARCSMSVolume 3, Issue 1, January 2015
[17] SeyedHasan, MortazaviZarch, Hussein Soltani and madihesadatYazdani, "Enhance The Security Of Password By Fuzzy Controller", IEEE 2014
[18] Florence Mwagwabi, Tanya mcgill and Michael Dixon, "Improving Compliance with Password Guidelines: How User Perceptions of Passwords and Security Threats Affect Compliance with Guidelines", 47th Hawaii International Conference on System Science, IEEE 2014.
[19] Anupam Das, Joseph Bonneau, Matthew Caesar, Nikita Borisov and xiaofengWang,"The Tangled Web of Password Reuse", NDSS 2014,23-26 February 2014,San Diego,CA,USA, Internet Society,ISBN 1-891562-35-5.
[20] Yulong yang, jannelindqvist and anti oulasvirta, "Text Entry Method Affects Password Security", LASER 2014, USENIX Association.
[21] Samuel OjodeOluoch, "Improving password security using location based intelligence", International Journal of Scientific and Research Publications, Volume 4, Issue 2, February 2014
[22] SeyedHasanMortazaviZarch, Hussein Soltani and madihesadatYazdani, "Enhance The Security Of Password By Fuzzy Controller", 978-1-4799-3351-8/14 2014 Ieee.
[23] Ding Wang and Ping Wang, " Offline Dictionary Attack on Password Authentication Schemes using Smart Cards", 16th Information Security Conference (ISC 2013), November 13-15, 2013, Springer--Verlag, pp.1-16.