International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT206475

Secure Banking Transaction using Encryption Based Negative Password Scheme

Abinaya R
Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamil Nadu, India

Article Info
Volume 6, Issue 4
Page Number: 402-410
Publication Issue :
July-August-2020

Article History
Accepted : 05 Aug 2020
Published : 10 Aug 2020

ABSTRACT

Password authentication is the most widely used authentication technique, for it is available at a low cost and easy to deploy. Many users usually set their passwords using familiar vocabulary for its convenience to remember. Passwords may be leaked from weak systems. Vulnerabilities are continuously being determined, and no longer all systems may be well timed patched to resist attacks, which give adversaries an opportunity to illegally access vulnerable systems. To overcome the vulnerabilities of password attacks, here propose a password authentication framework that is designed for secure password storage and could be easily integrated into existing authentication systems. In our framework, first, the received plain password from a client is hashed through a cryptographic hash function (e.g., SHA-512). Then, the hashed password is randomly shuffled to get a negative password. Finally, the negative password is encrypted into an Encrypted Negative Password using a symmetric-key algorithm RC5, to further improve security. The proposed hash function and encryption methodologies make it difficult to break passwords from ENPs. This proposed ENP system will be implemented for banking environment to improve security of password storage and transaction details.

Keywords : Password Creation, Hash Generation, Create Negative Values, RC5 Encryption, Negative Password Storage.

I. INTRODUCTION

A key area in protection studies is authentication, the dedication of whether a user need to be allowed access to a given system or resource. The crucial element of authentication is confidentiality and integrity. Also, for protective any resource used for authentication is the primary line of protection. Here, for protection of resource use authentication as a service. It is essential that the equal authentication approach have to not be used in every scenario. A hardship is that customers can also have many

passwords for Bank, community and web websites. The large variety of passwords will increase interference and it is cause forgetting or confusing passwords. The acceptability of any authentication scheme substantially relies upon on its robustness against attacks as well as its aid requirement both at the consumer and on the server end. It means authentication scheme require processing at consumer and sever side. Due to the proliferation of mobile and hand held devices the resource requirement has become a chief component. The implicit passwords major software is the protection of vital assets and structures. Nowadays customers can get access to any data which include banking and corporate database with the use of mobiles or smart devices. However, inspiration of this system is also can be used in different situation wherein confidentiality and integrity are the primary security requirements. Proposed Authentication System implement for banking with the use of Implicit Password. In which the scheme permits any image to be used and it does not want predefined click on areas with well-marked boundaries– a password may be any arbitrarily chosen series of points in the image with a few differences. In IPAS, the server has the piece of records i.e. Password on the time of authentication and at the time of registration, the person supply this records to the server in an implicit shape. Implicit password is in particular applicable for mobile phones and portable computers, even though it can be implemented for any computer system.

To put it surely, authentication is the procedure that confirms a consumer's identification. Traditionally, that is performed through a username and password. The user enters their username, which lets in the device to verify their identification; this gadget is predicated on the fact that best the consumer and the website online's server realize the password. The website authentication technique works by way of comparing the consumer's credentials with the ones on report. The authentication process is complete, when match is found.

1.1 Authentication

- ✓ Authentication is utilized by a server while the server needs to understand exactly who is getting access to their facts or web page.
- ✓ Authentication is utilized by a client while the client desires to recognise that the server is the system it claims to be.
- ✓ In authentication, the person or computer has to prove its identification to the server or client.
- ✓ Usually, authentication by a server involves the use of a user name and password. Other methods to authenticate can be through cards, retina scans, voice recognition, and fingerprints.
- ✓ Authentication through a client commonly includes the server giving a certificate to the customer in which a trusted third party together with Verisign or Thawte states that the server belongs to the entity (consisting of a financial institution) that the client expects it to.
- ✓ Authentication does now not decide what obligations the individual can do or what documents the individual can see. Authentication merely identifies and verifies who the person or device is.

1.1.1 Password Authentication

While password authentication is the most common way to confirm a user's identity, it isn't even close to the most effective or secures method. Anyone with your credentials can easily access account without your permission, and the system wouldn't prevent them. Most passwords are vulnerable, and hacking strategies can crack them in less and much less time.

1.1.2 Email Authentication:

Email authentication is a password less option that allows users to securely log in using just an email

address. The process is very similar to signing in with a social network account, but this method offers a universal approach.

- ✓ **The user clicks the login button.** This opens a mail to link that directs the person to pre-written email that includes an encrypted token.
- ✓ **The consumer sends the email.** The consumer sends the e-mail. The message already comes with a recipient identity so the user doesn't need to go into any records.
- ✓ **The server verifies the request. Using a combination of token-based security check process, the consumer's identity is proven.**

1.1.3 Biometric Authentication:

Biometric authentication includes any form of authentication technique that uses a user's biology factors. While this may seem like new-age technology, you're probably already using it to unlock the screen on your smart phone. Fingerprint scanning is the most popular form of biometric verification, however face identification tools are an increasing number of famous choice for developers.

The hackers have a more hard time replicating a users' biological factors, however it's is crucial to observe that these authentication procedures are frequently much less secure than you'd initially assume. Small fingerprint scanners on smart phones only capture small portions of image of your fingerprint, for instance. Multiple images of part of a fingerprint are provides less secure than a single, clear image. Remember, too, that biometric authentication can't be changed or altered if a user's fingerprints have been compromised. While biometric authentication holds a lot of promise, it's now most useful as an additional login tool to bolster another system.

II. RELATED WORK

Li, et al., [1] introduce a new metric called Coverage to quantify the correlation between passwords and personal information. Afterward, the proposed analysis, improves the Probabilistic Context-Free Grammars (PCFG) technique to overcome the problems of crack passwords by implementing personalized guesses. Since it considers the duration and size of private data, Coverage is a useful metric to identify the strength password. Our quantification results using the Coverage metric confirm our direct measurement results on the dataset, showing the efficacy of Coverage. Moreover, Coverage is easy to combine with current tools, consisting of password strength meters. To display the vulnerability present in the usage of personal data in passwords, here advice a semantics-rich Probabilistic Context- Free Grammars (PCFG) approach called Personal-PCFG, which extends PCFG by thinking about personal information symbols in password systems. Personal-PCFG takes a person's personal records as another input parameter and generates highly personalized guesses for a selected account. With the assistance of such knowledge, Personal-PCFG is able to crack passwords faster than authentic PCFG.

Zhao, et al., [2] The proposed scheme provides two-layer authentication data protection to enhance the security without increasing much computational cost. The scheme proposed in this paper is based on the information known by the user (i.e., the password). Moreover, it can be extended to a scheme with smart cards. That is to say, the smart card can be embedded as a device that stores the private data of the user and executes all the procedures on the client side. Overall, the proposed method can be adopted in following applications such as business management process, network-based consumer electronics, and intelligent household systems. The NDB has several promising properties. Such properties could be used to extend the proposed scheme. Furthermore, following the method used in the proposed scheme, the NDB can

be added to other authentication schemes as an extra layer to further improve security. Propose the first one-time password authentication scheme based on the NDB. Analyze the security of the proposed scheme and show that it can resist replay attacks, password leakage, guessing attacks, and exhaustive attacks. Moreover, we show that our scheme is robust to message blocking, and can be extended to be secure against a man-in-the-middle attack. Also analyze the computational complexity of the proposed scheme and show that it is more efficient than those schemes based on the encryption algorithms or discrete logarithm problem.

Zhao, et al., [3] Negative iris recognition aims to protect the privacy of iris data while attempting to retain the accuracy of the recognition. Negative iris recognition could be integrated into any traditional iris recognition scheme that uses the Hamming distance for matching and thus enhance the security of the iris data. Negative iris recognition contains two parts: enrollment and recognition. Assume that a server or device Sis used for controlling ac-cess to critical resources. A user or client A who wants to use those resources should enroll his iris data with S to be a valid user. After S receives the iris data of A, it generates a p-hidden-NDB and deletes the original iris data. There-fore, S does not keep the plain-text of the iris data, and the iris data thus are relatively secure against even inside attackers. After the enrollment phase, Scan use the iris database P to control the access of all of the users to the critical resources. Once a user A wants to use the resources, he will submit his iris data to S. Next, S checks whether this iris data correspond to a valid user and, then, determines whether A can access the resources. In addition to the Hamming distance, we can also attempt to extend our scheme to support other types of distance measurement, e.g., Euclidean distance. In this case, the iris data are converted to m-dimensional vectors of real-valued features. Real-valued NDB can be employed in our scheme. With a given threshold C, if the iris data are **x**, then entries in an NDB should not be covered by the m-ball E that is centered at **x** with the radius C. This NDB should be as complete as possible, and its entries can also be m-balls that have various radiuses to adaptively cover the complementary set. Thus, the NDB does not match any iris data that have a Euclidean distance of less than C to **x**, and valid users can successfully pass the recognition phase by submitting valid iris data.

Najjar, et al., [4] most common strategies for password storage use storing of hash code of the password with a salt calculated by means of any kind of Message Authentication Code (MAC) tools. These MAC equipment like HMAC or Block cipher, wherein the password is used as MAC key and the salt as the message, are provide more secure than previous approach of H(salt||p). However, the main weakness for such password hashing is brute force attack, which still now not resolved. To overcome this issue, present day algorithms used for password storage depend upon slowing the whole technique to make it extra immune to brute force attack. Such algorithms like BCrypt, PBKDF2 and SCrypt use a method called key stretching. Most of the current password secure algorithms think about the manage of the calculation time for generating hashed password to require extra Central Processing Unit CPU energy. Such solutions take into consideration Moore's Law. Hence, imply while computers get quicker, such algorithms can be changed to request extra CPU power. In this manner, the attacker should invest more time so as to calculate the password. BCrypt is one of the most known modern-day algorithms used in password storage. It is based on using Blowfish block cipher as a characteristic of key derivation for passwords. BCrypt makes use of salt to protect against rainbow desk attacks and it makes use of a function, which controls time of calculation. This feature relies upon on controlling the iteration be counted to make the

calculation of hashed password slower, a good way to be extra proof against brute-force attacks.

Jose, et al., [5] The proposed system does not allow even using a pattern based password which includes a dictionary word. Considering the researches on a pattern based dictionary attack, a password obeying the existing password policy which contains a dictionary word can never be accepted as safe. Hence, here when a user submits a password with certain pattern, it is subjected to pattern based computations to detect any dictionary word in it. If any dictionary word is found in the input, it will be shown as a weak password and the user is therefore not allowed to use it. Here, the complexity rules are much stronger so as to insist users to submit passwords containing words or characters other than dictionary words. Even when a pattern based dictionary attack is made, none of the passwords registered according to this protocol will be cracked. A character-tree structure is used here for storing passwords in a dictionary file. Most commonly used pattern types are identified from the input passwords and it is checked whether it is a dictionary word or not. A search in the character tree finds a word if the input password contains any dictionary words. The character tree gets updated by adding words that are newly recognized. This tree storage gives same performance as that of a tree data structure for storing words.

III. EXISTING METHODOLOGIES

Despite the achievements on password protection, passwords are nonetheless cracked considering the fact that customers' careless behaviors. For instance, many customers regularly pick out weak passwords; they have a tendency to reuse same passwords in special structures; they usually set their passwords using acquainted vocabulary for its convenience to recall. In addition, machine troubles might also purpose password compromises. It is very difficult to attain passwords from excessive safety systems. On the only hand, stealing authentication information tables (containing usernames and passwords) in high protection systems is difficult. On the other hand, while sporting out an online guessing attack, there is mostly a restriction to the number of login attempts. However, passwords may be leaked from weak systems. Vulnerabilities are constantly being observed, and no longer all systems may be well timed patched to resist assaults, which provide adversaries an possibility to illegally access weak structures. In truth, a few old systems are extra vulnerable due to their lack of renovation. Finally, in view those passwords are often reused, adversaries might also log into high protection structures via cracked passwords from systems of low safety.

IV. METHODOLOGY

ENPs could be obtained through the following steps.

Step 1:

The received plain password (i.e., a sequence of characters) from a client is first hashed using a cryptographic hash function.

Step 2:

Next, the hashed password is converted into a negative password using an NDB generation algorithm.

Step 3:

Then, the negative password is encrypted using a symmetric-key algorithm. Thus, the ENP is obtained. The solution of the negative password is the hash value of the received plain password. In the above processing, each component (i.e., the cryptographic hash function, the symmetric-key algorithm, and the NDB generation algorithm) is indispensable.

Step 4:

The cryptographic hash function converts plain passwords to hashed passwords; the fixed length property of resulting hashed passwords offers convenience for the subsequent encryption, since the length requirement for the secret key in the symmetric key algorithm.

The conversion from a hashed password to a negative password is not irreversible; therefore, if no encryption, when an adversary obtains a negative password, the adversary immediately obtains the corresponding hashed password, which makes the strength of the ENP equivalent to that of the hashed password essentially.

However, when adopting encryption, the adversary does not know the key (i.e., the hashed password converted from the original plain password), so the adversary could not decrypt the ENP to get the negative password.

V. SECURE PASSWORD STORAGE USING NEGATIVE DATABASE CREATION

A password safeguard scheme called Encrypted Negative Password (ENP) is proposed, which is based on the Negative Database (NDB), cryptographic hash function and symmetric encryption. The NDB is a new security technique that is inspired by biological immune systems and has a wide range of applications. Symmetric encryption is usually deemed inappropriate for password protection. Because the secret key of the all encrypted passwords and stored combined with the authentication data table, once the authentication data table is stolen by attacker, the shared key may be stolen at the same time. Therefore, these passwords are immediately compromised easily. However, in the ENP, the secret key is the hash value of the password of each user, so it is almost always different and does not need to be specially generated and stored. Consequently, the ENP enables symmetric encryption to be used for password protection. As an implementation of key stretching, multi-iteration encryption is introduced to further improve the strength of ENPs. Compared with the salted password scheme and key stretching, the ENP guarantees the diversity of passwords by itself without introducing extra elements.

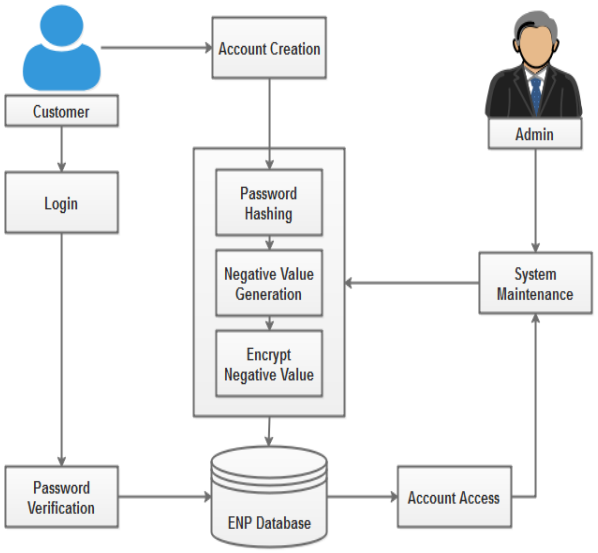

Fig. 3 Illustration of Proposed Work

VI. IMPLEMENTATION

Bank interface creation

Online transaction is thus changing the way people shop and how retailers operate. There is a steep decline in conventional payment methods which includes cash and cheque and people are choosing the digital payment technologies as they render effective flexible and convenient methods for engaging in cashless financial transactions. However, this technology and digital convergence has also attracted the threat of cyber-attacks and made banks and financial institutions more vulnerable to fraud. It has brought about a new breed of fraud perpetrators that use sophisticated technologies to hack into personal devices and corporate networks. Traditional strategies together with password or tokens are no fit to their attacks. To overcome, these attacks, we can design the interface for online transactions in banking system. In this module, admin and user interface created. Admin can be view the details of users, accounts details and so on. The user can be performing various operations such as net banking, credit card transactions, and debit card transactions and so on.

User Enrolment

This module explains about the user process. User has to create account to access online transaction application. User should enter the required fields for registration such as first name, address, account details, PIN number, user name & password. The secret password will transfer into negative value generation process.

ENP Creation

The details for account creation include the customer id, customer name and password. The received password is hashed using the selected cryptographic hash function using SHA 512. The hashed output is then passing to the shuffling process. Here performing conversion of hash into shuffled values. The negative password is encrypted to an ENP using the selected symmetric-key algorithm called RC5, where the key is the hash value of the plain password. The username and the resulting ENP are stored in the authentication data table and "Registration success" is returned, which means that the server has accepted the registration request.

User Verification using ENP

In password authentication users are verified using their unique login constraints. On the client side, a user enters his/her username and password. Then, the username and plain password are transmitted to the server through a secure channel. The details are converted into encrypted negative values and then compared with ENP database. If the received username does not present in the authentication database, then "Incorrect password!" is returned, which means that the server has rejected the authentication request, and the authentication phase is terminated. Otherwise application allows the user to make transaction.

Transaction Making

The user can transfer funds using the transaction password. The user should select the receiver name and the account number. Then, the amount to be transferred should be entered. User gets transaction password verification to make sure the authentication of user. The transaction details will be reflected in the corresponding accounts. The logout is used to exit from the application. After closing the session using logout option, the home page will show to the user.

ALGORITHM

SHA 512:

SHA 512 hash function is a feature generates message diggest 512-bit length and 1024 bit block length. The cryptographic algorithm works SHA 512 is to just accept input inside the form of a message with any duration or length and could generate a message digest that has a constant length of 512 bits.

SHA 512 algorithms is follows:

1. The addition of bits

The initial process is to add a number of bit wedges with message (i.e) the message length (in bits) are congruent with 890 mod1024. The aspect to bear in mind is that the 1024number seems because of the SHA 512algorithm methods messages in blocks of1024 sizes. If there is a message with a 24-bitlength, then the message will still be added with the bundle bits. The message will be added with 896- (24 + 1) = 871 bits. So the length of the wedge bits is between 1 and896.

2. Adding Long Message Redemption Value

Then the next process is the message added again with 128 bits stating the length of the original message. If the message length is largest than 128 bits then the length is taken in modulo of 128.

3. Initialize Hash Value

In SHA 512, the H (0) hash value consists of eight words with sixty four bits within the hexadecimal notation.

RC5 Encryption

- In RC-five, the word size (i.e. Enter plaintext block length), range of rounds and range of keys aren't fixed i.e. All can be of variable length.
- Once w, r, k (phrase size, range of rounds, range of keys) is finalized then they stay same for all of the rounds.
- Plain text can be 32 bits, 64 bits or 128 bits
- Number of rounds can be between 0-255
- Key size can be between 0 to 255 bytes

Encryption involved several rounds of a simple function. In average recommended 12 or 20 rounds, it is depending on security needs and time considerations.

Here initialize the counter to 1 and perform some permutation and combination using addition and XOR

The algorithm works into two phases:

a. First it starts with phase one

b. Output of process of phase one become input of phase two process

We divide the plaintext block into two equal parts A and B

Then they are XOR with two subkeys S{0} and S{1}

C=A+S[0]
D=B+S[1]
for i = 1 to r do:

1. C ⊕ D = E
2. Perform circular left shift on E by D bits
3. Add E and S [2 * i] and store the result in F which is input for step 4
4. D ⊕ F = G
5. Perform circular left shift on G by F bits
6. Add G and S[2 * i + 1] and store the result in H
7. If i< r

F was called as C and H was called as D and also repeats the steps from 1 to 7 else stop

Once both the phases are get completed the counter is incriminated and check the output is greater than the number of rounds, if yes then the algorithm get terminals and if no means then the algorithm iterates further rounds.

Decryption:

Decryption is a straightforward reversal of the encryption process.

VII.CONCLUSION

Encrypted Negative Password is a more secure compared with the other negative password system. This proposed method may be carried out in places where safety is low or extra security is needed. This idea can be used extensively in the area of banking. This method utilizes hash generation process, Negative Value generation and RC5 encryption. Finally the encrypted password would store on server. The password used is safe and no one can ever try to break the password. Instead of only apply hashing here converting the hash value into negative values and encrypting. The results will show that the ENP could resist password guessing attack and provide stronger password protection under dictionary attack.

VIII. REFERENCES

[1]. Li, Yue, Haining Wang, and Kun Sun. "Personal information in passwords and its security implications." IEEE Transactions on Information

Forensics and Security 12, no. 10 (2017): 2320-2333.

[2]. Zhao, Dongdong, and Wenjian Luo. "One-time password authentication scheme based on the negative database." Engineering Applications of Artificial Intelligence 62 (2017): 396-404.

[3]. Zhao, Dongdong, Wenjian Luo, Ran Liu, and Lihua Yue. "Negative iris recognition." IEEE Transactions on Dependable and Secure Computing 15, no. 1 (2015): 112-125.

[4]. Najjar, Mohannad. "Using Improved d-HMAC for Password Storage." Computer and Information Science 10, no. 3 (2017): 1-9.

[5]. Jose, Jacob, Tibin T. Tomy, Vibin Karunakaran, Anoop Varkey, and C. A. Nisha. "Securing passwords from dictionary attack with character-tree." In 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), pp. 2301-2307. IEEE, 2016.

[6]. Zhao, Dongdong, Wenjian Luo, Ran Liu, and Lihua Yue. "A fine-grained algorithm for generating hard-toreverse negative databases." In 2015 International Workshop on Artificial Immune Systems (AIS), pp. 1-8. IEEE, 2015.

[7]. Boonkrong, Sirapat, and Chaowalit Somboonpattanakit. "Dynamic salt generation and placement for secure password storing." IAENG International Journal of Computer Science 43, no. 1 (2016): 27-36.

[8]. Biryukov, Alex, Daniel Dinu, and Dmitry Khovratovich. "Argon2: new generation of memory-hard functions for password hashing and other applications." In 2016 IEEE European Symposium on Security and Privacy (EuroS&P), pp. 292-302. IEEE, 2016.

[9]. Wang, Ding, Debiao He, Haibo Cheng, and Ping Wang. "fuzzyPSM: A new password strength meter using fuzzy probabilistic context-free grammars." In 2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 595-606. IEEE, 2016.

[10]. Sun, Hung-Min, Yao-Hsin Chen, and Yue-Hsun Lin. "oPass: A user authentication protocol resistant to password stealing and password reuse attacks." IEEE Transactions on Information Forensics and Security 7, no. 2 (2011): 651-663.

Cite this article as :

Abinaya R, "Secure Banking Transaction using Encryption Based Negative Password Scheme", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 4, pp. 402-410, July-August 2020. Available at doi : https://doi.org/10.32628/CSEIT206475
Journal URL : http://ijsrcseit.com/CSEIT206475