Password Security Using Bcrypt with AES Encryption Algorithm

Narander Kumar and Priyanka Chaudhary

Abstract With the advancement of technology, the Internet has become a widely used tool of communication. Million numbers of individual all over in the world can get the utilization of technology. Novel issues like cyber stalking have been increasing worldwide global attention. Cyber stalking can be explained as threatening behavior or undesirable advances intended for another using the Internet and other way of online communications, so client authentication in computer systems is an essential feature in the present time for avoiding cyber stalking. In this paper, we have scheduled a technique utilizing Bcrypt hashing technique with AES encryption for securing an online account and reducing cyber criminal activity.

Keywords Cyber crime · Hashing · AES encryption algorithm
Bcrypt hashing · Cyber criminal · Cyber stalking

1 Introduction

In the evaluation of new technologies and advancement has enhanced our lives incalculable manner. Likewise, expand dependency on IT and communication approach for dynamic and active field arrangements has its own particular reaction. The impact of computerized data advancements upon the world surely postures unlimited advantages for the nationals of the developing worldwide town. The dark side of it, as anyone might expect, is misappropriating of Information Technology for criminal exercises. Cyber stalking is another sort of violation that existed subsequent to the late 1990s that raised a significant universal criminological type of problem in 2004. Generally, cyber stalking depicts the utilization of ICT keeping

N. Kumar (✉) · P. Chaudhary
Department of Computer Science, B.B. Ambedkar University (A Central University),
Lucknow, Uttar Pradesh, India
e-mail: nk_iet@yahoo.co.in

P. Chaudhary
e-mail: cpriyanka22@gmail.com

S. C. Satapathy et al. (eds.), *Smart Computing and Informatics*, Smart Innovation,
Systems and Technologies 77, https://doi.org/10.1007/978-981-10-5544-7_37

in mind the end goal to bug one or more martyr [1]. Furthermore, molestation implies any conduct that causes the martyr trouble whether deliberate or not. Cyber stalking frequently discovers their martyr online [2] as they utilize computer and network systems for criminal exercises, as these advances can without much of a stretch be misusage to startle, threaten, force, annoy, and deceive clueless clients. Password-based authentication protocols cannot depend with respect to persevering put away data on the customer side. Rather, they depend on clients' capacity of exact review of a secret key data. It is primarily because of this exact review prerequisite that clients regularly pick basic and low entropy passwords that are anything but difficult to recollect [3]. The vulnerability of passwords turns into the feeble connection of the system, which assailants misuse by dispatching offline or online dictionary attack.

The password hashing strategy is superior than encryption of secret key since hashing has contained single functionality we cannot find plain text content from its hash implies the plain password that builds hash cannot be reconstructed from its containing hash value. The method of hashing is delicate to dictionary assault. Dictionary assault is a strategy for recouping password from known password. So it is conceivable to split hash password by utilizing pre-computed hash value. A hashing algorithm is exceptionally deterministic as they deliver same hash value for same inputted content. Raw hashes have likewise helpless against rainbow tables, a technique for adjusting a requirement for a pre-calculation of hashes and the clearly substantial function is important to keep a whole dictionary containing hashes [4]. For averting from this type of issues, salting acts as a rescue. Salt guarantees that assailants cannot utilize specific assaults like lookup tables and rainbow tables to break huge collection of hashes rapidly, yet it does not keep them from running a dictionary assault or brute force attack on each hash independently. High-end graphic cards and custom equipment could be processed on billions of hashes each second, so these assaults are still extremely powerful. To make these assaults less powerful, they can utilize a method known as key stretching for whatever length of time that an assailant can utilize a hash to check whether a password is correct or wrong, they could be run a dictionary or brute force attack on the hash. The objective of the proposed work is to fortify existing password-based verification conventions against brute force attack for securing client accounts that are the best approach for the user side to memorize and securing password utilizing hashing technique with AES for online account protection. Thus, it is more secure to assault from instance hacking, phishing, fraud.

2 Review of Work

P. Sriramya and R. A. Karthika implemented an algorithm which is used in Bcrypt algorithm with salt hash technique provides more security to users for online shopping [5]. Yu-Chi Chen et al. formulize the blind decoding schema, i.e., planned as technique being secured end client protection in online looking for electronic proof [6]. Salmin Sultana et al. studied on the security of wireless sensor system, wireless sensor network quick created and utilized as a part of numerous divisions. Subsequently, the need for security turns out to be an exceptionally vital component. There are numerous advances being accessible to give security against the assailants, one of the best innovations is cryptography [7]. Halderman et al. [8] studied, an approach, cryptographic hash function have utilized to prepare secure passwords for subjectively various records, and customer requires to hold short secret passwords [9]. These types of procedure perform task absolutely on the user side, server side adjustment is not required. Khiyal et al. [10] planned a system of password hashing, i.e., process for safe passwords. A recent approach taking into the advanced encryption standard for securing a password is designed in [11]. A symmetric cipher technique that depends on the Rijndael technique is designed in [12] yet this technique begins with 200 bits. M.S.H. Khiyal et al. planned an instrument for securing passwords, i.e., using MD5 and SHA1 cryptographic hash function [13]. The function is utilized to infer one or more secret keys from a secret string. It depends on memory-hard function which offers some additional insurance against assaults utilizing custom hardware [14]. The SHA-192 can be utilized as a part of numerous functions such as an open key cryptosystem, advanced signcryption, content verification, approach random generator and in security engineering of forthcoming remote gadgets like programming characterized radio, and so forth [15]. SSH protocol is planned as a substitution for the current rsh, rlogin, rcp, rdist, and telnet protocol [16]. A paper is proposed to deal with security against phishing assaults with “BogusBiter” is discussed in [17]. F. Mwagwabi et al. examination approaches are scheduled during this study have additionally appeared to be a valuable approach to clarifying IS security behavioral intentions [18]. S. Farmand et al. proposed exploration are a way to deal with upgrade the current graphical password strategies and oppose against assaults like shoulder surfing [19].

From an existing review of the work, we have found some vulnerabilities salt guarantees that intruders cannot use specific attacks like lookup tables and rainbow tables to break huge collections of hashes rapidly; however, it does not keep them from running dictionary or brute force attacks on each and every hash independently.

2.1 Brute Force Attack

At the point, when a brute force attack to each possible set of characters up to a given length. This kind of assault is computationally excessive, the minimum expert with respect to as hashes cracked per processor time; however, they will dependably and consequently perform the password.

2.2 Salt Collision

Salt collision happens when two passwords encrypted utilizing with related same salt value. For developing dictionary attacks, an intruder can be grouped with ciphered passwords through the salt and hash each and every applicant password from a dictionary just per salt. The outcome speedup can be determined as follows:

$$\frac{\text{Number of Password}}{\text{Number of Different Salt}} \tag{1}$$

In the event that salts are produced by a random number generator, the accepted number of various salts for n password entries with s salts is

$$\mathrm{EV}(n,s) = \sum_{i=0}^{n-1} \left(\frac{s-1}{s}\right)^{i} = s - (s-1)^{n} s^{1-n} \tag{2}$$

3 Proposed Method

In this proposed method, we use Bcrypt hashing algorithm using the AES encryption technique if an attacker can utilize a hash to check whether a password guessing theory is correct or incorrect, they can run a word reference table or brute force attack on the hash. The hash function that will be decided by this value.

In proposed model, defined in Fig. 1, user opens a login form. Store user information in a list including with user name and password, as well as the necessary information with a password. We apply Bcrypt hashing algorithm. After applying Bcrypt hashing, we use AES encryption algorithm. The list of data would be saved in the format of windows registry. A user could be open his list by entering a valid password and then applying decryption techniques, and also, check the user is valid or not and provides access to the list of data.

Bcrypt algorithm that is discussed in [6] in which initial key is a key stretching as a password then aggressor will be first to try each word in the dictionary or common password list and then try to check all possible character combinations for longer password. In our proposed method, this problem can be reduced using with AES encryption technique.

4 Implementation

We designed our proposed algorithm using with NetBean IDE 8.0 software. The below figures define the implementation of the proposed algorithm by using different numbers of text data values and sizes of a wide range. Figure 2 is defined the

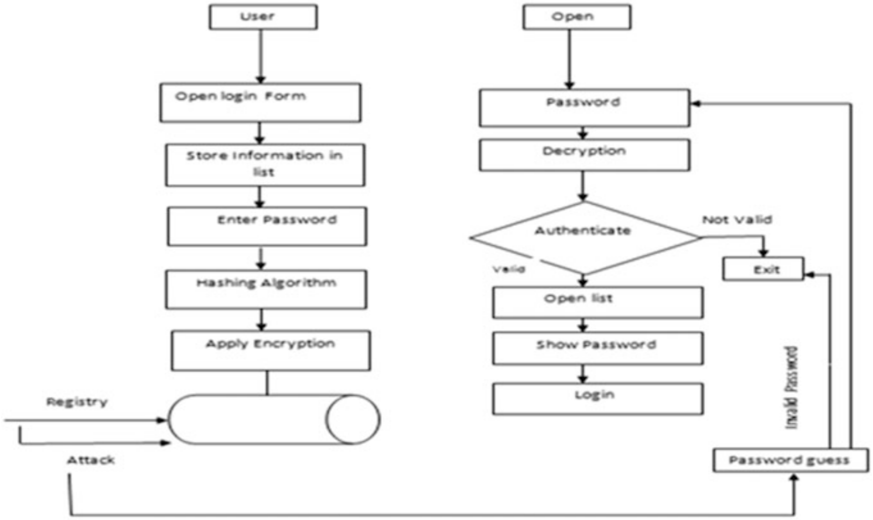


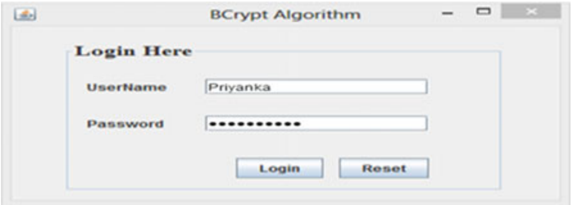
Fig. 1 Flowchart of proposed algorithm

login page of the user. Figure 3 demonstrates that generating salt with a hash of the corresponding user password with the help of Bcrypt hashing algorithm. Figures 4 and 5 demonstrates that application of AES encryption technique to a generating of hash corresponding user password.

5 Result and Discussion

From the outcome, demonstrated database password security utilizing hashing and salting pattern give a stronger security with the end goal that the original password is never put away. Regardless of the possibilities that the password store is compromised, just the hashes get to be a public. The password length is not stored and could not be estimated, making password cracking that much harder. There is no requirement for a secret, as none is utilized to hash the password.

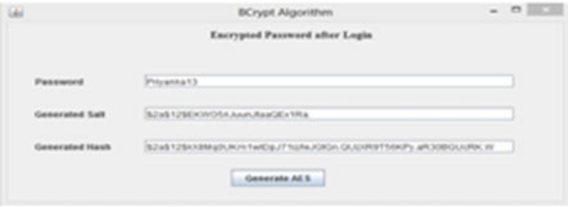
Fig. 2 Login page


BCrypt Algorithm

Login Here

UserName: Priyanka

Password: ●●●●●●●●●●●



BCrypt Algorithm

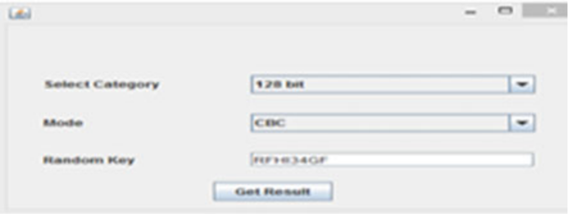
Encrypted Password after Login

Password: Priyanka13

Generated Salt: $2a$12$EKWO5A.Auun.RaaQEx1Ra

Generated Hash: $2a$12$kXBMqDUKm1wIDpJ71iz/ta.iQXGn.QUjXR9T56KPy.aR30BQUrRK.W

Fig. 3 Generated hash using Bcrypt hashing algorithm

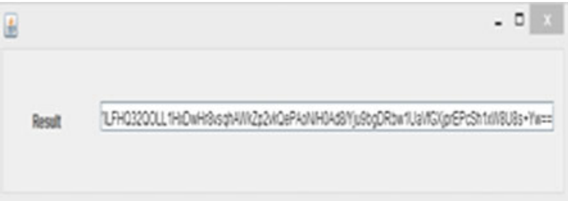


Select Category: 128 bit

Mode: CBC

Random Key: RFHE34GF

Fig. 4 Category and mode selection of AES technique



Result: 'LFHQ32OOLL1HiDwHi9xsqhAWZp2vKQePAoNH0Ad8YjuSbgDRbw1UaMGXprEPcSh1xW8U8s+Yw==

Fig. 5 Encrypted password using AES and Bcrypt algorithm

The performance matrices are defined encryption time and throughput time. The encryption time is defined as, the time is taken for generating a cipher text from plaintext, and throughput time is defined as (Figs. 6 and 7; Tables 1 and 2).

$$\text{Throughput} = \frac{\text{Size of Encrypted Text in MB}}{\text{Time Required for Encryption in Seconds}} \tag{3}$$

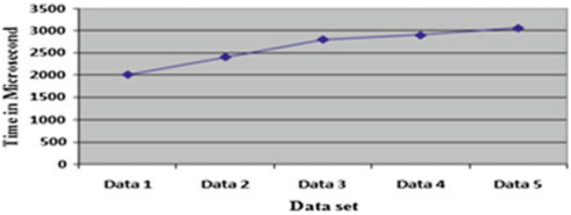


Fig. 6 Execution time for proposed algorithm

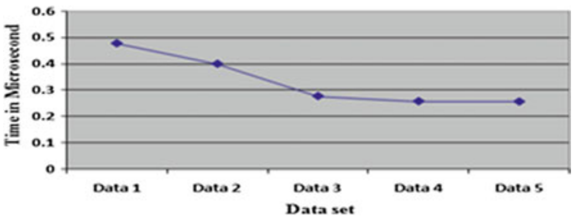


Fig. 7 Throughput time of proposed mechanism

Table 1 Execution time of proposed algorithm

Data text (character)	Proposed algorithm execution time (microsecond)
Data 1	2200
Data 2	2436
Data 3	2661
Data 4	3084
Data 5	3572

Table 2 Throughput of proposed algorithm

Plain text (in terms of character)	Data set (size in MB)	Execution time (seconds)	Throughput
Data 1	0.000967	0.002	0.4835
Data 2	0.000984	0.00244	0.4032
Data 3	0.000965	0.00267	0.3711
Data 4	0.000990	0.00309	0.3203
Data 5	0.001090	0.00350	0.3114

6 Conclusions

Protection and accuracy are the two major fields of overall system customers. This type of problems is illuminated by the study of cryptographic methods. The storage of password security is an essential part of information protection, as most systems nowadays require a validation technique utilizing passwords. The technique of hashing is usually utilized for transforming plain content passwords into set of character probably it cannot be decrypted by intruder due to the way of their one-sided cipher technique. Be that as it may, with time, the attacks got to be possible through the using of word reference tables and rainbow tables. In this paper, we have intended to the utilizing Bcrypt hashing algorithm with AES for providing the user's account protection. Our future work would explore this concept, and a combination of algorithms will be applied either sequentially or parallel, to set up a more secure environment for data storage and retrieval.

References

1. Bocjj, P.: The Dark Side of the Internet: Protecting Yourself and Your Family from Online Criminals, 2nd edn, pp. 159–161. Greenwood Publishing Group, Westport, CT (2006)
2. Morley, D.: Understanding Computers in a Changing Society, 3rd edn, pp. 196–199. Course Technology Cengage Learning, Boston, MA (2008)
3. Buxton P.: Egg rails at password security, Netimperative, June, 24, (2002)
4. Zombie PCs for Rent Information Security&p%5Bne%wsletterId%5D=609, September 2004
5. Dorrans, B.: ASP.NET Security, Wiley (John Wiley & Sons, Ltd), ISBN:978-0-470-74365-2, 2010
6. Sriramya, P., Karthika, R.A.: Providing password security by salted password hashing using Bcrypt algorithm. J. Eng. Appl. Sci. **10**(13), 5551–5556 (2015)
7. Chen, Y.C., Horng, G., Huang, C.C.: Privacy protection in on-line shopping for electronic documents. In: 5th International Conference on Information Assurance and Security, pp. 105–108 (2009)
8. Sultana, S., Ghinita, G. et. al.: A lightweight secure scheme for detecting provenance forgery and packet drop attacks in wireless sensor networks. IEEE Trans. Dependable. Secure Comput. **12**(3), 256–269 (2015)
9. Halderman, J.A., Waters, B., Felten, E.: A convenient method for securely managing passwords. In: Proceeding of the 14th International World Wide Web Conference, pp. 471–479 (2005)
10. Khiyal, M.S.H., Khan, A., Bibi, N., Ashraf, T.: Analysis of password login phishing based protocols for security improvements. In: Proceeding of IEEE 5th International Conference on Emerging Technologies (ICET 2009), pp. 376–379 (2009)
11. Stallings, W.: Data and computer communications, Pearson Education, Inc., Eighth Edition, ISBN: 0-13-243310-9, (2007)
12. Islam, M.N., Mia, M.M.H., Chowdhury, M.F.I., Matin, M.A.: Effect of security increment to symmetric data encryption through AES methodology. In: Nineth ACIS International Conference on Software Engineering. Artificial Intelligence. Networking and Parallel/Distributed Computing, pp. 291–294 (2008)
13. Zhao, Z., Dong, Z., Wang, Y.: Security analysis of a password-based authentication protocol proposed to IEEE 1363. Theor. Comput. Sci. 352, 280–287 (2006)
14. Khiyal, M.S.H., Khan, A., Bibi, Ashraf, N.T.: Analysis of password login phishing based protocols for security improvement. In: Proceeding of IEEE 5th International Conference on Emerging Technologies (ICET 2009), pp 376–379 (2009)
15. Lakshmanan, T., Muthusamy, M.: A novel secure hash algorithm for public key digital signature schemes. Int. Arab J. Inf. Technol. 262–267 (2012)
16. Ylonen, T.: SSH secure login connections over the internet. In: Proceedings of the USENIX Web Security, Privacy & Commerce, 2nd edn
17. Yue, C., Wang, H.: Anti-phishing in offense and defense. In: Proceedings of the 24th Annual Computer Security Applications Conference (AC-SAC'08), pp. 345–354 (2008)
18. Mwagwabi, F., McGill, T., Dixon, M.: Improving compliance with password guidelines: how user perceptions of passwords and security threats affect compliance with guidelines. In: 47th Hawaii International Conference on System Sciences, pp. 3188–3197 (2014)
19. Farmand, S., Zakaria, O.B.: Improving graphical password resistant to shoulder-surfing using 4-way recognition-based sequence reproduction (RBSR4). In: 2nd IEEE International Conference on Information Management and Engineering (ICIME), pp. 644–650 (2010)