Password Protection Using Cryptographic Hash Technique

S.Preethika
Research Scholar, PG and Research Department
Quaid-E-Millath Government College for Women (Autonomous), Chennai, India.

Dr.G.Velmayil
Assistant Professor, PG and Research Department
Quaid-E-Millath Government College for Women (Autonomous), Chennai, India.

Abstract – Internet has changed much in two decades since it came in to an existence. It conceived in the era of time sharing, survived in to the era of personal computing, client-server, peer-to-peer computing and networking is now squeezed between attacks as it moved to web application. Numerous folks use web for illegal activity. Security depends mostly on passwords to legitimize users from attackers. The foremost common authentication technique is to use alphanumeric usernames and passwords. The offenders use numerous attacks like SQL injection, DOS, Phishing, Spoofing, etc., to fetch the information from the web. SQL injection attack happens terribly to attack the database by attackers through injecting malicious query. Numerous algorithm were developed using cryptographic concepts to secure data from attacks, however salt might prove to be powerful than others. In this paper, we tend to gift the solution for securing the positive identification and sensitive information from SQL injection carried through cryptographic salt. The salt is random alphanumeric string concatenated with secret data in different databases and use hash technique to entrust the protection.

Index Terms – SQL injection, salted password, password protection, salt, authenticated system, hashed password.

1. INTRODUCTION

The use of passwords is thought to be ancient. Centuries would challenge those desire to enter an area or approaching it to provide a password or watchword, and would solely permit an individual or group to pass if they knew the password. In present, user names and passwords are usually employed by folks during a log in process that controls access to protected computer operating systems, smart phones, cable TV decoders and cash machine (ATMs), etc. A typical computer user has passwords for several purposes: logging into accounts, retrieving e-mail, accessing applications, databases, networks, web sites, and even reading the morning newspaper on-line.

Despite the name, there's no need for passwords to be actual words, so passwords that aren't actual words may be tougher to guess, a fascinating property. Some passwords are formed from multiple words and should more accurately be known as a passphrase. The terms passcode and passkey are typically used once the key data is solely numeric, like the non-public positive identification (PIN) usually used for ATM access. Passwords are usually short enough to be simply memorized and typed.

Even though it is prominent, the vulnerabilities related to the implementation of Username and Password authentication system in internet applications include the threat of password guessing and SQL Injection attacks. Numerous studies have shown that the explanation for SQLI is basically poorly modifying input from the users. Password guessing attacks on the opposite hand are sometimes successful against weak passwords, poor password enforcement policies, and poor style of the authentication functionality similarly as its implementation.

To address the matter of SQL Injection attacks researchers have advanced many techniques starting from defensive coding best practices to machine-controlled frameworks for detection and bar of these sorts of attacks in the same approach, techniques like account lock out, fully machine-controlled.

2. RELATED WORK

In 2009, Shaukat Ali, et.al [10], proposed a way that forestalls the user information from the SQL Injection attack by using the Hashing Techniques. It stores the user details with the hash value for each username and password within the back end. Once user login with user name and password, it generates the hash value and compare it with the backend. Though hash techniques give a lot of advantages it has some problems with implementing the good technique.

In 2012, Deevi Radha Rani, et.al [1], proposed a method that checks user credentials against encrypted values against placeholders in stored procedures. This system is efficient in attacking code injection, but the technique relies on the assumption that the developer is approaching the development during a specific means particularly within the coding of stored procedures, if they're written insecurely, then the approach are

ineffective, once more it offers flexibility to the attacker to launch another attack vector like password guessing attacks.

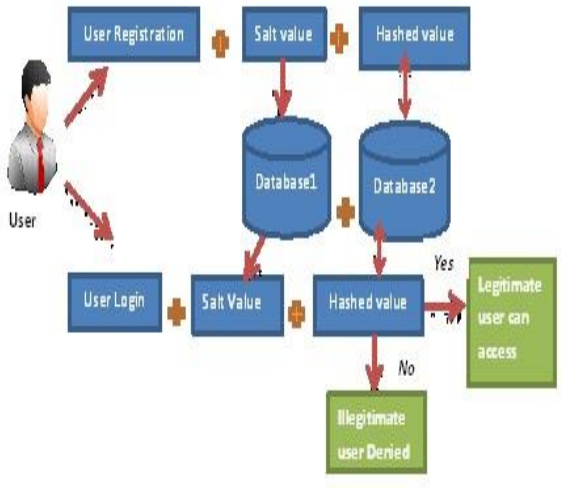
In 2013, Pritesh N. Patel and et.al [8], proposed a cryptography application using salt hash technique that generate salt for every password that is stored within the database. This method indeed of securing the password as a result of implementing the small salt. Thus it results in hacked by attackers using different techniques.

In 2014, T.Rajesh and et.al [13], proposed a password management system using cryptographic salt generated for every Username and Password. And also generate the web concern username/ password using ASCII code for the registered password. It finds the ASCII code of each and every character of the username/password. However this approach fails because of repetition of username and password may results in generating the same ASCII code.

3. PROPOSED SYSTEM

In the Password protection using cryptographic hash technique system, the user information is safeguarded by implementing the cryptographic salt to the username and password that is hold on within the back end. The password security is upgraded by generating the random cryptographic salt for each password that is stored in different databases by using the *RNG Crypto Service Provider* in ASP.net. The small salt results in, attacked by numerous hacking techniques like rainbow attacks, dictionary attacks, due to its size. That's the reason for implementing the large salt during this proposed system. The salt is then kept within the database. Then the salt and the password are passed to the hash technique to come up with the hash value. The hash value is stored once more in the database. In this system *SHA512* hash technique applied to encrypt the salted password.

Architecture Diagram



When the user login with the username and the password, the cryptographic salt retrieved from the backend. Then it is send to the hash technique and generates the hash value. That value is compared with the backend hash value. The legitimate user can access their account. It's common for a web application to store the hash value of a user's password in the backend. Whereas not a salt, an unbowed SQL injection attack may yield easily breakable passwords as a result of several users re-use passwords for multiple sites. The utilization of a salt could be a crucial a part of overall web application security. A salted hash defeats rainbow table attack, easily by exponentially increasing the size of the rainbow table needed to with success of finding a collision. The entire process of the proposed system is explained in the architecture diagram.

4. METHODOLOGY

In this approach, we tend to produce a secure authentication system based on username and password. The Password protection using cryptographic hash technique system has been enforced using the front-end as a Microsoft Visual Studio 2010 and Microsoft SQL server 2012 as a back-end (database). Two stored procedures namely, Create_User_Details and Create_Salted_Password are used.

Create_User_Details store procedure is used once a new user account is made for the first time. It generates the salt value for each user password and keep within the Salted_Password. Then it calculates the hash values of Salt and Password and stores into the User_Details. It retrieve the data from database on every occasion once a user needs to login into database using user name and password.

4.1. Steps to generate salt:

1. Get Password.
2. Generate a salt using random function for the Password entered.
3. Store the password and salt in a database.
4. Concatenate the salt and the password, pass the value to the hash algorithm.
5. Generate the SaltedPassword using hash algorithm.
6. Store the HashedSaltedPassword in a different database.

4.2. Steps to Login:

1. Enter username and password.
2. Retrieve the salt value of entered password by comparing the user-id and username from the database.
3. Concatenate the entered password and the retrieved salt.

4. Pass the value to hash technique.
5. Generate the SaltedPassword using Hash technique.
6. Compare the new HashedSaltedPassword with the registered hashed password in the database.
7. If it is legitimate user, the access allowed or else denies the access.

5. RESULTS

The SHA-512 algorithm takes an input message with a maximum length of less than 2^{128} bits and produces as output a 512-bit message digest. The output of the hash function does not reveal the information of the input. And also the hash functions are durable to search out a collision.

5.1. SQL Injection Attack in Existing System:

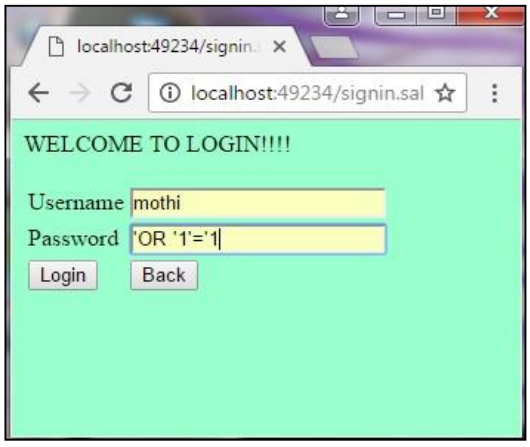


Fig 1: SQL Injection Attack

In the *Fig 1*, the SQL Injection attack occurred in the existing system by entering the password as *'OR '1'='1.*

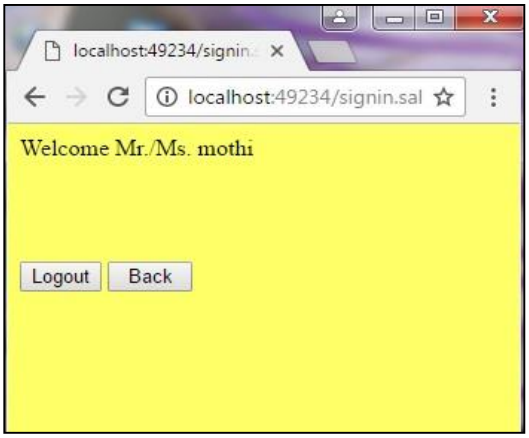


Fig 2: SQL Injection attack successfully executed

In *Fig 2*, the SQL Injection attack executed and entered into the User account page.

5.2. SQL Injection attack in Proposed System

To solve the issues in existing system, the password protection using cryptographic hash technique system is implemented and the screenshot shown in *Fig 3* and *Fig 4.*

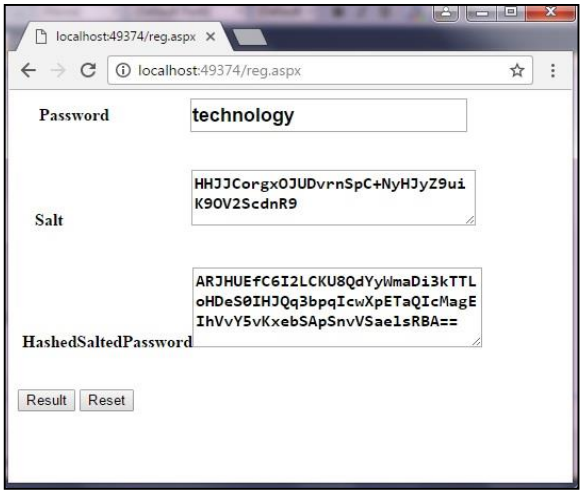


Fig 3: Generating the Salt and Hashed Salted Password

In *Fig 3*, the word "technology" is entered as password. The Salt value and the HashedSaltedPassword value of the entered password are generated.

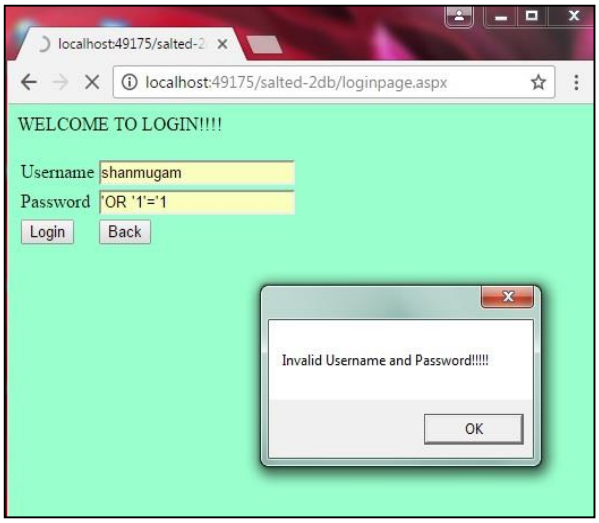


Fig 4: Failed SQL Injection Attack

In *Fig 4*, the SQL Injection attack injected by entering the password as "'OR'1'='1". That fails as a result of implementing the password protection using cryptographic hash technique system.

The Password protection using cryptographic hash technique system sends to entropy test with the existing system of Text

Password and the Hashed Password by using the Shannon Entropy.

It is named after Shannon the American Claude Shannon wrote "A Mathematical Theory of Communication" in 1948, an article that created information theory, although its origin goes back to Pauli and von Neumann.

Shannon entropy, *H* is given by the formula,

$$H = -\sum_{i} p_i \log_b p_i \qquad (1)$$

Formula1: Shannon Entropy

Where *pi* is the probability of character number, *i* showing up in a stream of characters of the given "string".

The entropy of the string is calculated using on top of formula. The *Fig 5* shows the average entropy of the Password, Hashed password and Salted Hashed Password.

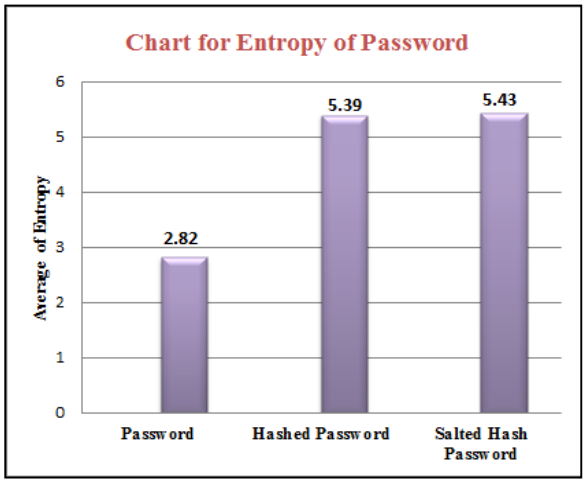


Fig 5: Entropy of Password

The Password holds the password which is registered by the user. Then the same password is passed to the hashed technique and the value holds by the Hashed Password. Finally, the same password passed to the enhanced authenticated system and password protection using cryptographic salt and hash technique system.

Thus, the result implies that the Salted Hashed Password entropy is on top of the other alternative two sorts.

6. CONCLUSION

In this the password protection using cryptographic hash technique system, securing the user data is achieved by implementing the salt and hash technique. The plain text password can attack easily. In Salted Password, the SQL Injection can take place but it takes some time and need computational effort. But when the salt is stored in different database, SQL Injection is tough to crop up. And also this approach can implement to any real time applications which require user authentication to access it. Hash technique provides a more reliable and flexible method of message digest. The salt is stored in the different database which gives twofold security to the system. It prevents the system from various attacks.

7. FUTURE WORK

The cryptographic salt is generated to the text password during this proposed system. The proposed effect will be enhancing by the password security using the graphical password with the salt and hash technique in my future project.

REFERENCES

[1] Deevi Radha Rani, B. Siva Kumar, L.Taraka Rama Rao, V. T. Sai Jagadish, M. Pradeep (2012), "Web security by preventing sql injection using encryption in stored procedures", IJCSIT, Volume 3(2), 2012.

[2] Diksha G. Kumar , Madhumita Chatterjee (2014), "Detection block model for sql injection attacks" , I.J. Computer Network and Information Security, 2014, 11, 56-63.

[3] Kanchan Choudhary, Anuj Kumar Singh, Rashmi Gupta (2016), "A modified scheme for preventing web application against sql injection attack", International Journal of Computer Applications (0975 – 8887) Volume 141 – No.10, May 2016.

[4] Mary Cindy Ah Kioon, ZhaoShun Wang and Shubra Deb Das (2013), "Security analysis of md5 algorithm in password storage", Atlantis Press, Paris, France, 2013.

[5] Ms. Vidya Vijayan, Ms. Josna P Joy, Mrs. Suchithra M S (2014)," A review on password cracking strategies", IJRCCT, 2014.

[6] Nikita Gupta, Lalit Sen Sharma (2016), " A study on sql injection attack and its prevention measures at database management level ", International Journal of Modern Computer Science (IJMCS), Volume 4, Issue 3, June, 2016.

[7] P. Sriramya and R. A. Karthika (2015), "Providing password security by salted password hashing using bcrypt algorithm", ARPN Journal of Engineering and Applied Sciences, VOL. 10, NO. 13, JULY 2015.

[8] Pritesh N. Patel, Jigisha K. Patel and Paresh V. Virparia (2013), "A cryptography application using salt hash technique", IJAIEM, Volume 2, Issue 6, JUN 2013.

[9] Punit Mehta, Jigar Sharda, and Manik Lal Das (2015)" SQLshield: Preventing SQL injection attacks by modifying user input data", Springer International Publishing Switzerland 2015, ICISS 2015, LNCS 9478, pp. 192–206, 2015.

[10] Shaukat Ali, Azhar Rauf, and Huma Javed (2009), "SQLIPA: An authentication mechanism against sql injection", European Journal of Scientific Research, Volume 38, No. 4, 2009.

[11] Sruthy Manmadhan and Manesh (2012), "A method of detecting sql injection attack to secure web applications", International Journal of Distributed and Parallel Systems (IJDPS) Vol.3, No.6, November 2012.

[12] Surya Pratap Singh, Upendra Nath Tripathi, Manish Mishra (2014), "Detection and prevention of sql injection attack using hashing technique", IJMCTR, Volume 2, Issue 9, Sep 2014.

[13] T.Rajesh, P.Shyamala Madhuri, V.Venkanna (2014), "Password management system using cryptography using salt technique", IJETTCS, Volume 3, Issue 1, Jan-Feb 2014.

[14] T.S.Thangavel and K.S.Rangasamy (2010)," Provable secured hash password authentication", International Journal of Computer Applications (0975 – 8887), 2010, Volume 1 – No. 19.

[15] Tivkaa, M.L., Choji, D. N., Agaji, I., Atsa"am, D. (2016), "An enhanced password-username authentication system using cryptographic hashing and recognition based graphical password", IOSR-JCE, Volume 8, Issue 4, Ver-1, Jul-Aug. 2016.

Authors

S.Preethika received the Master degree in Computer Applications from Valliammal College for Women, Chennai, Tamilnadu in 2014. She is a research student of Quaid-E-Millath Government College for Women (Autonomous), Chennai, Tamilnadu. She is pursuing her M.Phil degree in Computer Science in the field of SQL Injection in Computer networks. Her research interest includes authenticating the password, attacks in network and network security.

Dr.G.Velmayil obtained his bachelor degree in Mathematics from M.K University, Madurai in1992, The M.C.A degree from Avinashilingam Deemed University, Coimbatore in 1995, M.Phil degree in Computer Science from Bharathidasan University in 2005 and P.hD degree in Computer Applicaton from Manonmanium Sundaranar University, Thirunelveli in 2014. She is currently an assistant professor in the PG & Research Department of Computer Science, Quaid-E-Millath Government College for Women (Autonomous), Chennai having 20 years of teaching experience. She has organized various workshops, seminars and conferences. Her research interests include Network Security, Cryptography, DDoS attacks, IP Spoofing and especially attack areas in Network .She has published more than 15 papers in various international and national conferences and journals.