Efficacy for User Authentication and Data Security using Cryptographic Algorithm

Rishabh Adey[1], Gireesh Kumar Chandrakar[2], Jyotsna Sahu[3]

[1]*Department of Computer Science and Engineering, Government Engineering College Raipur, Chhattisgarh, India*
[2]*Department of Computer Science and Engineering, Government Engineering College Raipur, Chhattisgarh, India*
[3]*Department of Computer Science and Engineering, Government Engineering College Raipur, Chhattisgarh, India*

---***---

Abstract – *In this paper we will discuss about various strategies which will be used to accomplish the productive client or user authentication procedure and data security model alongside execution and streamlining of client experience. Here we are confronting the principle issue in authentication of client and improvement of security of database. The past work done have utilized the conventional text based secret passwords for client authentication process. Clients will in general pick passwords that are anything but difficult to recall, making them vulnerable to different assaults that have developed throughout the years. In this paper we will discuss two systems which are utilized to for the client authentication and data security model. This paper basically examines structure utilized for client confirmation known as Object Hash-based Contrivance, where the client machine figures the hash of the article to be utilized as content secret password and for data security known as Encrypted Storage Model which encodes the information before embedding it into database and unscrambles it while recovering back the information from database.*

***Key Words*: Object Hash based Contrivance, password, authentication, data security, hash, Encrypted Storage Model**

1. INTRODUCTION

Clients give inadequate consideration to admirably pick a secret password. This propensity has been misused utilizing straightforward assaults like secret phrase speculating to progressively specific techniques, for example, word reference assaults. The escape clauses of content based passwords are all around archived. It is additionally hard for clients to produce and remember solid or high-entropy passwords. Further, these solid passwords are commonly utilized just in the event that they are much of the time utilized. Passwords for once in a while utilized administrations are difficult to repeat at a later point in time. What's more, when we watch for database secure socket security layer is generally used. It shields information making a trip from the customer to the server. It doesn't ensure tenacious information put away in a database. It is an on-the-wire convention once an aggressor accesses your database straightforwardly, delicate information might be uncovered or abused. Encoding the information is a decent method to relieve this risk.

2. LITERATURE SURVEY

Text based password express structures that are the affirmation segment most conventionally used for user authentication, concerning user security, is just the demonstration of perceiving in order to get access to information or resources. By a long shot a large portion of user approval is worked on using text based password frameworks. In text based password structures, the user is required to introduce a puzzle mystery, which just they should know, in order to affirm their character to a figuring system. Immaculate passwords would be those that are basic for user to remember, improving the strategy of approval, anyway difficult for aggressors to figure, rendering the system security. In trying flawless passwords, we are familiar with the security/convenience tradeoff. Strong (secure) passwords are difficult to remember, and passwords that are definitely not hard to review are usually feeble.

The motivation behind each database is to store data, writings, pictures, even media documents. All unique current sites depend on at least one database for putting away articles and other distributed substances, data about the clients, contact data, associations with different sites, advertisements, and so on. What's more, much the same as whatever else, putting away data will require a space to be put away in.

Database security alludes to the different estimates associations take to guarantee their databases are shielded from interior and outside dangers. Database security incorporates ensuring the database itself, the information it contains its database the executive's framework and the different applications that get to it. Associations must make sure about databases from intentional assaults, for example, cyber security dangers, just as the abuse of information and databases from the individuals who can get to them. Physical database security: It's basic to not neglect the physical equipment on which the information is put away, kept up, and controlled. Physical database security incorporates locking the rooms that databases and their servers are in—regardless of whether they are on premise resources or got to through the cloud.

The utilization of web applications and firewalls is a database security best practice at the edge layer. Firewalls keep interlopers from getting to an association's IT organize through the web; they're an urgent essential for cyber security concerns. Web applications that communicate with databases can be secured by application to get to the executive's programming. This database security measure is like access control records and figures out who can get to web applications and how they can do as such.

3. RELATED WORK

Ensuring efficient client authentication process and providing best security to database is one of the interesting research areas in the field of computer science. Some already developed systems in this problem area are explained below:

In paper [1] Singh, Prabhsimran & Kaur, Kuljit. (2015). Database security using encryption. 2015 1st International Conference on Futuristic Trends in Computational Analysis and Knowledge Management, ABLAZE 2015. 353-358. 10.1109/ABLAZE.2015.7155019. Security of Data is the most important task in today's world. Over the years various encryption schemes have been developed in order to protect the database from various attacks by the intruders. This paper discuss the importance of database encryption and makes an in depth review of various database encryption techniques and compare them on basis of their merits and demerits.

In paper [2] Rao, Sandeep & Mahto, Dindayal & Khan, Danish. (2017). A Survey on Advanced Encryption Standard. International Journal of Science and Research (IJSR). 391. 10.21275/ART20164149. Rijndael's Advanced Encryption Standard (AES) is the block cipher based symmetric-key cryptography to protect the sensitive information. The key sizes of AES are 128, 192, 256 bits. AES is based on substitution-permutation strategy. It is accepted by NIST in 2001 after the five year of security evaluation. It is highly secured and efficient than Data Encryption Standard (DES) and other symmetric-key cryptographic algorithms. This paper depicts all the valuable work done on the Advanced Encryption Standard since it is accepted by National Institute of Standards and Technology (NIST).

In paper [3] Hameed, Sufian & Qaizar, Lamak & Khatri, Shankar. (2017). Efficacy of Object-Based Passwords for User Authentication. Traditional text-based password schemes are inherently weak. Users tend to choose passwords that are easy to remember, making them susceptible to various attacks that have matured over the years. ObPwd [5] has tried to address these issues by converting user-selected digital objects to high-entropy text passwords for user authentication. In this paper, we extend the ObPwd scheme with a new object based password scheme that performs majority of the computation at the server side. This paper essentially discusses two frameworks for object password schemes, an object hash-based scheme (where the client machine computes the hash of the object to be used as text password) and an object-based scheme (where the object is directly transmitted to the server as password). We also evaluate the performance of both the object password schemes against conventional text-based password schemes using prototypes of each of the frameworks. Implications with respect to ease of use, sharing and security are also discussed.

In paper [4] Appel, Andrew. (2015). Verification of a Cryptographic Primitive: SHA-256. ACM Transactions on Programming Languages and Systems. 37. 1-31. 10.1145/2701415. This article presents a full formal machine-checked verification of a C program: theOpenSSL implementation of SHA-256. This is an interactive proof of functional correctness in the Coq proof assistant, using the Verifiable C program logic. Verifiable C is a separation logic for the C language, proved sound with respect to the operational semantics for C, connected to the CompCert verified optimizing C compiler.

In paper [5] Chhillar, Krishan. (2019). Database Security Using Encrypted Private Key. In Database Security Using Encrypted Private Key, we propose a secure data handling scheme for dynamic members of an organization. Firstly, we create a secure way for a private key distribution with a secure communication channel, so that the users can securely obtain their private keys from the organization admin. Secondly, our scheme can achieve fine-grained access control, any user in the group can use the source in the database and revoked users cannot access the database again after they are revoked. Thirdly, we can protect the scheme from collusion attack, which means that revoked users cannot get the original data file even if they conspire with the untrusted database. In our approach, by leveraging polynomial function, we can achieve a secure user revocation scheme. Finally, our scheme can achieve fine efficiency, which means previous users need not to update their private keys for the situation either a new user joins in the group or a user is revoked from the group.

4. OBJECTIVE OF WORK

4.1 Existing Method

Existing method of text based password is generally utilized for client confirmation in the various field wherever any place the client approval is required and with regards to ensuring data security over the database either premises or server farm is bolted genuinely or third party firewall service providers are utilized for making sure about data security which is as yet helpless and relies upon the unwavering quality of the third party security service providers.

4.2 Proposed Work

Numerous efforts have been made over the years by the scientific community to strengthen passwords and to enhance their usability. In this paper we have primarily focus on the most relevant Object Hash based Password Contrivance. It allows client to generate passwords from digital content that may range from a personal collection of photographs to static content from the web. And when it comes for enhancing security of database in this paper we discusses the concept of Encrypted Storage Model which first encrypt sensitive user data that should be decrypted sometimes in a future. Encryption and Decryption are helpful if sensitive data is exchanged between two different applications. The AES (Advance Encryption Standards) with SHA (Secured Hashing Algorithm) provides a good solution to it.

5. METHODOLOGY

5.1 Object Hash based Contrivance

The Object-hash based Password Contrivance utilizes media objects as the client's passwords and performs a greater part of the calculation on the client utilizing client-side contents. The server-side usefulness is steady with that of the content-based secret word plot. The server receives the user-ID and a text-based version of the user password (hash of the media object), brings the secret phrase hash and salt from the database and assesses whether they got accreditations are substantial.

The server gets the client ID and text-based version of the media object as opposed to the password in plaintext from the text-based scheme. Nonetheless, since them two are strings (character successions), the ensuing calculation is the equivalent. The customer, then again, requires some extra calculation.

A hash value (utilizing SHA-256) is figured for the media object utilizing a customer side content written in JavaScript. We utilize a more grounded hash work, for example, SHA-256, rather than SHA-1 in Object Hash based Password Contrivance. Further, Object Hash based Password Contrivance use Password Hash for diminishing the hash esteems into a fixed-sized character long alphanumeric secret key and confines the item size somewhere in the range of 30 and 100000 bytes. This method doesn't make a difference any such limitations.

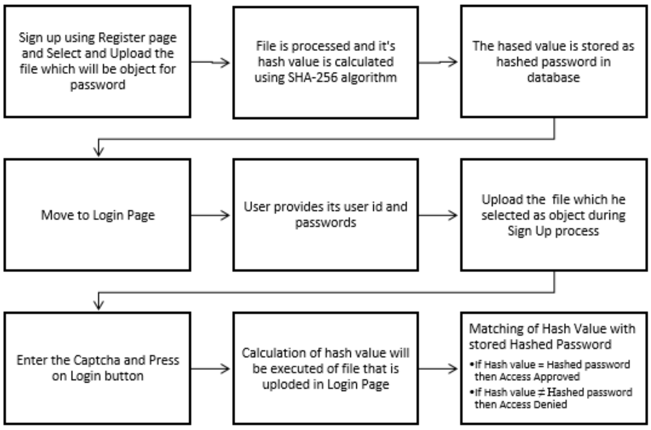

Sign up using Register page and Select and Upload the file which will be object for password
File is processed and it's hash value is calculated using SHA-256 algorithm
The hased value is stored as hashed password in database
Move to Login Page
User provides its user id and passwords
Upload the file which he selected as object during Sign Up process
Enter the Captcha and Press on Login button
Calculation of hash value will be executed of file that is uploded in Login Page
Matching of Hash Value with stored Hashed Password
•If Hash value = Hashed password then Access Approved
•If Hash value ≠ Hashed password then Access Denied

Fig -1: Working of Object Hash based Password Contrivance

5.2 Encrypted Storage Model

First encryption need to done which is the first station is encryption algorithm also called Cipher. In general working and functionality of all algorithms are known and publicly available and they are not hidden. The only thing that is hidden is key which is under our control. Here we will be using Advanced Encryption Standard (AES), also known by its original name Rijndael.

Advanced Encryption Standard (AES) is an iterative rather than Feistel cipher. It is based on 'substitution–permutation network'. This consists of a series of connected operations, some involving replacing inputs with different outputs (substitutions) and others involving shuffling bits (permutations) around them. Ironically, Advanced Encryption Standard (AES) performs all of its computations on bytes instead of bits. Advanced Encryption Standard (AES) handles the 128 bits of a block of plaintext as 16 bytes. The number of rounds in AES is inconsistent and depends on the size or length of the key.

While encrypting, we confine to the depiction of a run of the mill round of AES encryption. Each round include four sub-forms. Byte Substitution (Sub Bytes) the 16 info bytes are subbed by looking into a fixed table (S-box) given in structure. The outcome is in a grid of four lines and four sections. After this shifting of rows is performed. Every one of the four lines of the network is moved to one side. Any passages that 'fall off' are re-embedded on the correct side of the line.

The implementation is done as follows – the first line isn't moved. At that point, the subsequent line is moved one (byte) position to one side. After that third line is moved two positions to one side. The fourth line is moved three positions to one side. The outcome is another network comprising of a similar 16 bytes yet moved regarding one another. Then Mixing of Columns is done. Every line of four bytes is currently changed utilizing a unique mathematical function. This function takes as information from the four bytes of one segment and yields four totally new bytes, which supplant the first section. The outcome is another new lattice comprising of 16 new bytes. It ought to be noticed that this progression isn't acted in the last round.

Lastly addition of round key is performed. The 16 bytes of the grid is currently considered as 128 bits and are XORed to the 128 bits of the round key. On the off chance that this is the last round, at that point, the yield is the cipher text. Something else, the subsequent 128 bits are deciphered as 16 bytes and we start another comparable round.

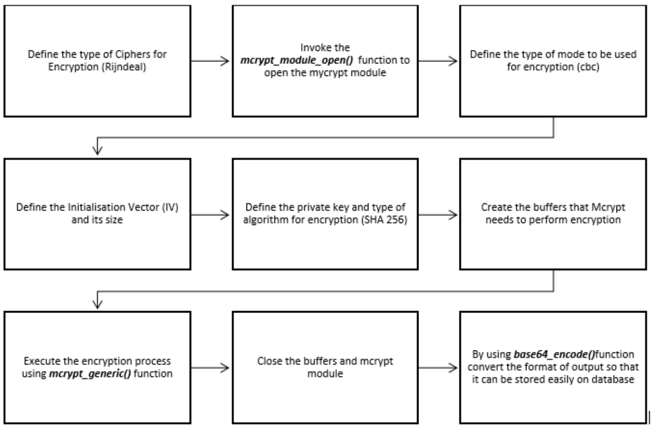

Define the type of Ciphers for Encryption (Rijndeal)
Invoke the mcrypt_module_open() function to open the mycrypt module
Define the type of mode to be used for encryption (cbc)
Define the Initialisation Vector (IV) and its size
Define the private key and type of algorithm for encryption (SHA 256)
Create the buffers that Mcrypt needs to perform encryption
Execute the encryption process using mcrypt_generic() function
Close the buffers and mcrypt module
By using base64_encode()function convert the format of output so that it can be stored easily on database

Fig -2: Encryption of Data in Database

The procedure of decryption of an AES cipher text is like the encryption procedure but in the reverse order. Each round comprises of the four procedures directed in the converse request – Include round key, Blend Column, Move rows and Byte substitution. Since sub-forms in each round are backward way, dissimilar to for a Feistel Cipher, the encryption and decryption calculations should be independently actualized, in spite of the fact that they are firmly related.

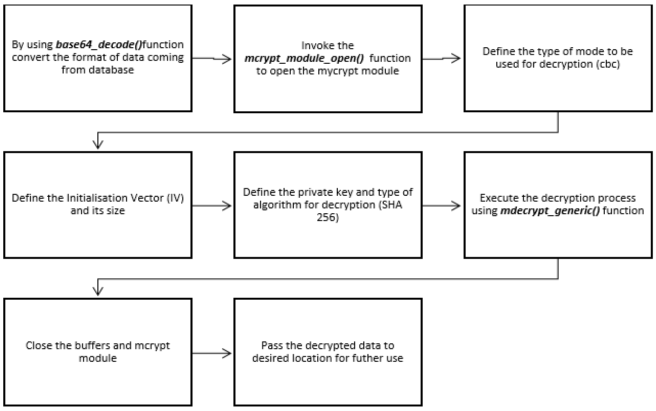
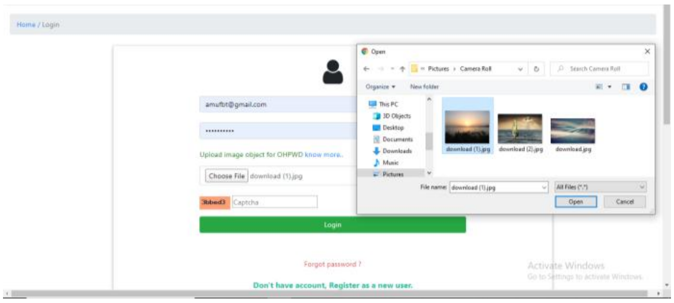

By using *base64_decode()*function convert the format of data coming from database
Invoke the *mcrypt_module_open()* function to open the mycrypt module
Define the type of mode to be used for decryption (cbc)
Define the Initialisation Vector (IV) and its size
Define the private key and type of algorithm for decryption (SHA 256)
Execute the decryption process using *mdecrypt_generic()* function
Close the buffers and mcrypt module
Pass the decrypted data to desired location for futher use

Fig -3: Decryption of Data in Database

6. RESULT

The flow of Object Hash based Password Contrivance and Encrypted Storage Model is illustrated in this section using screen captures. The prototype comprising both the method allows the user to login with credentials which he submits during initial registration or sign up process. We have deployed this prototype on a web-based platform over a local server to get down to the results.

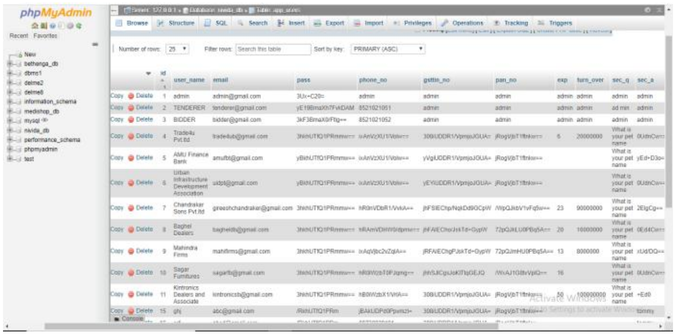
The client will initially enroll in the web-based platform and keeping in mind that enlisting he needs to choose an ideal object which he needs to use as a secret password, and its hash value would be determined and uploaded away in the database for that specific client. After this client will log in to the web-based platform utilizing the object he chose for client verification while enlisting into a web-based platform after that he needs to enter the ordinary password and proceed onward by clicking on the login button. A solicitation would be sent to the server and a hash value of the item so transferred would be coordinated with the recently put away hashed value and as needs be access would be conceded.



amufbt@gmail.com
••••••••••••
Upload image object for OHPWD know more..
Choose File download (1).jpg
3bbed3 Captcha
File name: download (1).jpg
All Files (*.*)

Fig -4: Uploading of Object (Image) for Object Hash based Contrivance

After the arrangement of a complete database of a web-based portal, for the improvement of the security of the database Encrypted Storage Model is actualized portrayed over the normal database and henceforth all the information stored in database gets converted into coded form which is decrypted at whatever point it is extracted from the database and encrypted while storing in the database effectively.



Number of rows: 25 Filter rows: Search this table Sort by key: PRIMARY (ASC)

id	user_name	email	pass	phone_no	gsttin_no	pan_no	exp	turn_over	sec_q	sec_a
1	admin	admin@gmail.com	3Ux+C20=	admin	admin	admin	admin	admin	admin	admin
2	TENDERER	tenderer@gmail.com	yE19BmaXh7FvkDAM	8521021051	admin	admin	admin	admin	ad min	admin
3	BIDDER	bidder@gmail.com	3kF3BmaX0rFftg==	8521021052	admin	admin	admin	admin	admin	admin
4	Trade4u Pvt.ltd	trade4ub@gmail.com	3hkhUTfQ1PRmmw==	ixAnVzXU1Volw==	30BUDDR1/VpmjoJGUA=	jRogVjbT1ftnkw==	6	20000000	What is your pet name	0UdnCw==
5	AMU Finance Bank	amufbt@gmail.com	yBkhUTfQ1PRmmw==	ixAnVzXU1Volw==	yVgUDDR1/VpmjoJGUA=	jRogVjbT1ftnkw==			What is your pet name	yEd+D3o=
6	Urban Infrastructure Development Association	uidpt@gmail.com	yBkhUTfQ1PRmmw==	ixAnVzXU1Volw==	yEYUDDR1/VpmjoJGUA=	jRogVjbT1ftnkw==			What is your pet name	0UdnCw==
7	Chandraker Sons Pvt.ltd	gireeshchandraker@gmail.com	3hkhUTfQ1PRmmw==	hR0nVDbR1/VvkA==	jhFSIEChp/NqkDd9GCpW	/WpQJkbV1vFq5w==	23	90000000	What is your pet name	2ElgCg==
8	Baghel Dealers	bagheldh@gmail.com	3hkhUTfQ1PRmmw==	hRAmVDHWSitpmw==	jhFAIEChp/JskTd+GypW	72pQJkLU0PBq5A==	20	10000000	What is your pet name	0Ed4Cw==
9	Mahindra Firms	mahifirms@gmail.com	3hkhUTfQ1PRmmw==	ixAqVjbc2vZqlA==	jRFAIEChgPJskTd+GypW	72pQJmHU0PBq5A==	13	8000000	What is your pet name	xUd/DQ==
10	Sagar Furnitures	sagarfb@gmail.com	3hkhUTfQ1PRmmw==	hR0iWzbT0PJqmg==	jhh5JlCgsJoKlTlqGEJQ	/WxAJ1GBvVplQ==	16		What is your pet name	0UdnCw==
11	Kintronics Dealers and Associate	kintronicsb@gmail.com	3hkhUTfQ1PRmmw==	hB0iWzbX1/VvIA==	30BUDDR1/VpmjoJGUA=	jRogVjbT1ftnkw==	50	100000000	What is your pet name	+Ed0
15	ghj	abc@gmail.com	/RkhUTfQ1PRm	jBAkUDPz0Ppwnzl=	30BUDDR1/VpmjoJGUA=	jRogVjbT1ftnkw==				tbnmy

Fig -5: Database after implementing Encrypted Storage Model

7. CONCLUSION

After analyzing the result diversified techniques are utilized to achieve the proficient client authentication procedure and data security alongside usage which in results to enhancement of user experience. Usage of hashed password with help of Object Hashed based Contrivance and encrypting them in database using Encrypted Storage Model together in combination diminishes the chance of getting hacked of database and revealing of private data to suspicious unauthorized people or organization. The object hash based contrivance offer higher security benefits over text-based passwords in the event that they are utilized effectively. The object hash-based password contrivance frameworks may also be used in future for interchangeably or in concert with one another. Smartphones can stack the object hash based contrivance creation structure just as progressively ground-breaking frameworks that could use for the equivalent purposes. As far as Encrypted Storage Model is concerned it provides a cost effective way to enhance the security of database and reduces the reliability over less trusted third-party security service providers.

REFERENCES

[1] Singh, Prabhsimran & Kaur, Kuljit. (2015). Database security using encryption. 2015 1st International Conference on Futuristic Trends in Computational Analysis and Knowledge Management, ABLAZE 2015. 353-358. 10.1109/ABLAZE.2015.7155019. M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[2] "Rao, Sandeep & Mahto, Dindayal & Khan, Danish. (2017). A Survey on Advanced Encryption Standard. International Journal of Science and Research (IJSR). 391. 10.21275/ART20164149.

[3] Hameed, Sufian & Qaizar, Lamak & Khatri, Shankar. (2017). Efficacy of Object-Based Passwords for User Authentication.

[4] Appel, Andrew. (2015). Verification of a Cryptographic Primitive: SHA-256. ACM Transactions on Programming Languages and Systems. 37. 1-31. 10.1145/2701415.

[5] Chhillar, Krishan. (2019). Database Security Using Encrypted Private Key.

BIOGRAPHIES

Mr. Rishabh Adey,
Student of 2016-2020 batch GEC Raipur

Mr. Gireesh Kumar Chandrakar,
Student of 2016-2020 batch GEC Raipur

Ms. Jyotsna Sahu,
Student of 2016-2020 batch GEC Raipur