ENHANCING TRUST IN DATA ENCRYPTION-DECRYPTION USING SIMPLE AND HIGHLY SECURE METHOD

[1]Athira S., [2]Binumon Joseph

[1] PG Scholar of Amal Jyothi College of Engineering, Kanjirappally, Kerala

[2] Asst. Professor of Amal Jyothi College of Engineering, Kanjirappally, Kerala

[1] athiras152@gmail.com, [2] binumonjosephk@amaljyothi.ac.in

Abstract: Data with deferent types are now transferring over the internet and the demands for clouding systems are now increasing. Data generated, captured, and replicated are increasing in size and expanding applications.So the need for data encryption-decryption methods became a vital necessary issue. The used technique here must very simple, effective, and accurate. A simple and highly secure encryption decryption (SHSED) algorithm that can be used for cloud computing-based applications, where introduced. Encrypting data on client machine and then storing the information to cloud storage server, computing hash of the information on client machine and storing hash of data in client machine, client trying out the responsibility of sharing the trick key about encryption with specific band of people. SHSED uses simple and efficient logical operations, such as XORing, addition, and subtraction in addition to byte shifting. The introduced method is also powered by the flexibility of selecting the secret key length and the number of rounds to generate the cypher text. Introduced method were compared with other methods results (LED, AES, DES), and as a result of comparisons the introduced method give a good improvement in encryption-decryption time measurement.

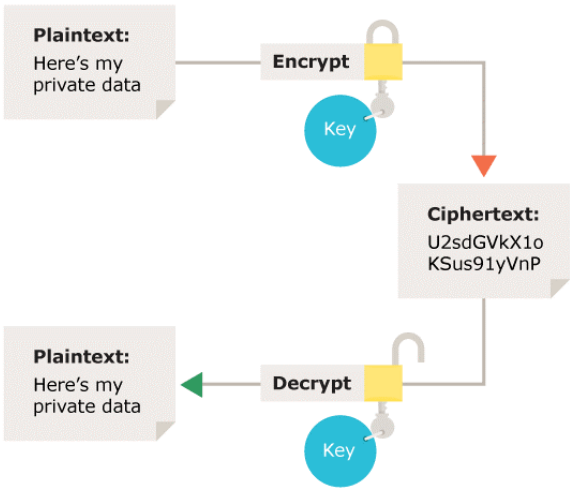
Keywords- AES, DES, LED, encryption time, decryption time.

I. INTRODUCTION

Cryptography is the practice and study of techniques for secure communication in the presence of third parties (called adversaries). More generally, it is about constructing and analyzing protocols that overcome the influence of adversaries and which are related to various aspects in information security such as data confidentiality, data integrity, authentication, and non-repudiation. Cryptography prior to the modern age was effectively synonymous with encryption, the conversion of information from a readable state to apparent nonsense. The originator of an encrypted message shared the decoding technique needed to recover the original information only with intended recipients, thereby precluding unwanted persons to do the same. Since World War I and the advent of the computer, the methods used to carry out cryptology have become increasingly complex and its application more widespread. Cryptography includes the following process:

1.1 Encryption and Decryption:

It is the process of converting ordinary information (called plaintext) into unintelligible text (called ciphertext). Decryption is the reverse, in other words, moving from the unintelligible ciphertext back to plaintext. A cipher is a pair of algorithms that create the encryption and the reversing decryption. The detailed operation of a cipher is controlled both by the algorithm and in each instance by a "key". This is a secret (ideally known only to the communicants), usually a short string of characters, which is needed to decrypt the ciphertext .



Modern Cryptography

2.1 Modern Cryptography

Following are the modern field of cryptography:

a) Symmetric-Key Cryptography Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key. Symmetric key ciphers are implemented as either block ciphers or stream ciphers. A block cipher enciphers input in blocks of plaintext as opposed to individual characters, the input form used by a stream cipher. The Data Encryption Standard (DES) and the Advanced Encryption Standard (AES) are block cipher designs which have been designated cryptography standards by the US government.

b)Public-Key Cryptography Symmetric-key cryptosystems use the same key for encryption and decryption of a message, though a message or group of messages may have a different key than others. A significant disadvantage of symmetric ciphers is the key management necessary to use them securely. Each distinct pair of communicating parties must, ideally, share a different key, and perhaps each ciphertext exchanged as well. The number of keys required increases as the square of the number of network members, which very quickly requires complex key management schemes to keep them all consistent and secret. Diffie and Hellman's publication sparked widespread academic efforts in finding a practical public-key encryption system. This race was finally won in 1978 by Ronald Rivest, Adi Shamir, and Len Adleman, whose solution has since become known as the RSA algorithm.

c) Cryptanalysis The goal of cryptanalysis is to find some weakness or insecurity in a cryptographic scheme, thus permitting its subversion or evasion. It is a common misconception that every encryption method can be broken. In connection with his WWII work at Bell Labs, Claude Shannon proved that the one-time pad cipher is unbreakable, provided the key material is truly random, never reused, kept secret from all possible attackers, and of equal or greater length than the message. Most ciphers, apart from the one-time pad, can be broken with enough computational effort by brute force attack, but the amount of effort needed may be exponentially dependent on the key size, as

compared to the effort needed to make use of the cipher.

II. LITERATURE REVIEW

Suzaki T [1] .al describes that Both colour and black & white image of any size saved in tagged image file format (TIF) can be encrypted & decrypted using blowfish algorithm. Histogram of encrypted image is less dynamic and significantly different from the respective histograms of the original image. Blowfish cannot be broken until an attacker tries 28r+1 combinations where r is the number of rounds. Hence if the number of rounds are been increased then the blowfish algorithm becomes stronger. Since Blowfish has not any known security weak points so far it can be considered as an excellent standard encryption algorithm. [2].

S. Morioka [2] al shows that MREA algorithm is used to encrypt files and transmit encrypted files to other end where it is decrypted. The project works efficiently for small size while it consumes time for large size of files. At a instant only one file can be encrypted and transmitted. As a future work multiple file encryption and decryption can be possible. It has broad development prospects. The project application was designed to take the efficiency and reusability into account. Great level of security is achieved using this algorithm. Modified RSA algorithm for file transmission algorithm can be used where high security file transmission required in public forums.

E. Kobayashi [3] al proposed a new key generation algorithm based on palm print which is used for encryption and decryption of an image. Our scheme allows one party to send a secret image to another party over the open network, even if many eavesdroppers listen. This scheme gives reliable security. They presented an image encryption/decryption scheme based on bit XOR method in this paper. The salient features of the proposed asymmetric image encryption scheme can be summarized as: (a) Lossless encryption of image. (b) Less computational complexity. (c) Convenient realization. (d) Choosing a suitable size of matrix according to the size of image. (e) Encryption/decryption scheme uses integer arithmetic and logic operations.

Jian Guo al [4] .al represents that the proposed algorithm has the following limitations:

1) More Execution time 2) Key Length and length of plain text must be same

In the future wok related to proposed algorithm, the limitations of proposed algorithm are overcome by encrypting and decrypting data may or may not be same key length size in comparison with input size.

Axel Poschmann [5.al describes that the method proposed in this paper has got a lossless encryption of image. This also gives access to variable lengths of the encryption keys. Another main feature of this method is that it satisfies the properties of Confusion and diffusion and also has a perfect guess of encryption key makes decryption impossible. This Encryption uses only integer arithmetic and it can be easily implemented in the hardware.

III. PROBLEM DEFINITION

Steps to access the database of a website

- **STEP 1: Find Vulnerable Website**

We can’t SQLi attack on all websites. The websites need a SQLi vulnerability in order to do this technique. Website URL need a parameter like php?id=4 / php?id= any number to inject. Once you find a website, then you can checkfor SQLi vulnerability.Put an‘(Apostrophe) at the end of the URL parameter If the page remains in same page or showing some other webpages. Then it is not vulnerable If it showing any errors which is

related to sql query or showing that page not found then it is vulnerable .

- **STEP 1:** check the version by using the command sqlsus.

Fig 4: Give the version

- **STEP 3:** Create a configuration file.

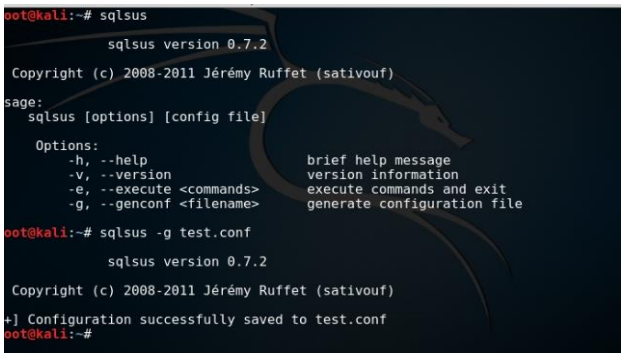


Fig 2: Creating configuration file

- **STEP 3:** Edit the configuration file and add the url of the website.

```
##############################
########## GENERAL ###########

# Start of the url used for the injection
# In inband/union mode, it is generally a good idea to append "AND 0" so that the real
query returns nothing
# Ex : our $url_start = "http://localhost/script.php?id=1'";
our $url_start = "https://www.webscantest.com/datastore/searchget_by_id.php?id=4";

# End of the url used for the injection
# When possible, it is generally a good idea to use "#" here, so that our queries won't
be polluted by the original one
# Ex : our $url_end = "#";
our $url_end = "";

# Use POST instead of GET
our $post = 0;

# Use blind injection ?
# set it to 1 for boolean-based blind injection
# set it to 2 for time-based blind injection (requires MySQL >= 5.0.12)
our $blind = 0;

# In boolean-based blind mode, string to be found in the HTML if the statement is true
our $blind_string = "";
```

Fig 3: adding url of website

- **STEP 4:** Execute the configuration file using the following command

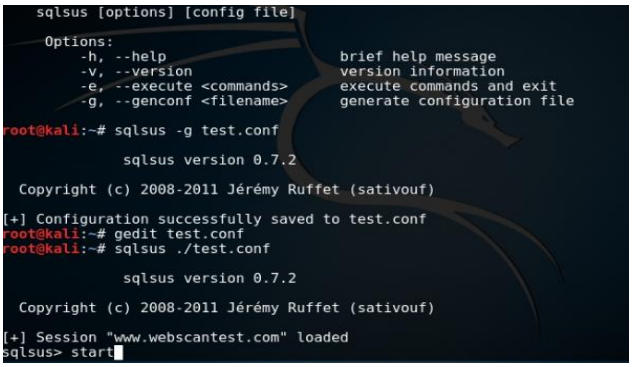


Fig 4: Load the database

- **Step 5**: List the tables.

Fig 5: Capture all tables

- **Step 6**: Use command select * from accounts to get the encrypted password and username

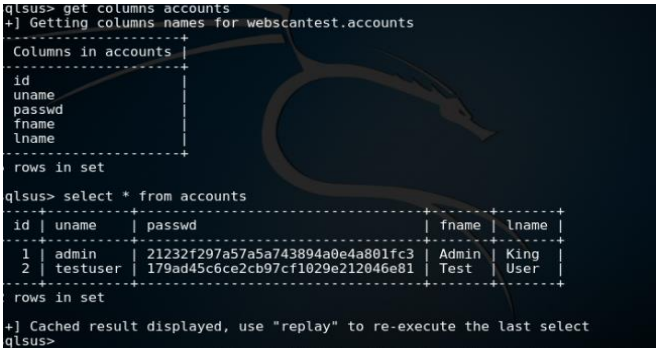


Fig 6: Capture all the values in the columns

- **Step 7:** find the encryption format using hash-identifier

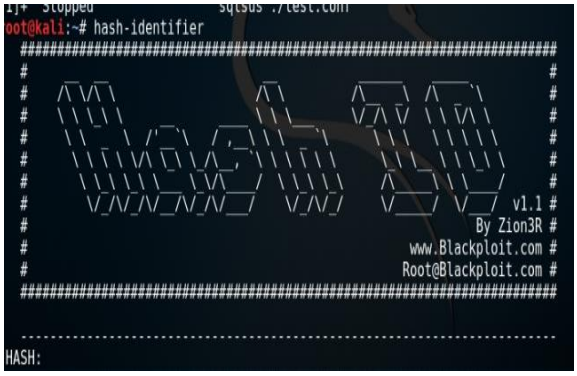


Fig 7: Use hash-identifier

- **Step 7:** Enter the hashed text here.

Fig 7: copy hashtext

- **Step 4:** Save the encrypted password in a text file using this command

Fig 4 Save the password to text file

- **Step 4:** Detect the password using the following command (assume that the hashing technique used was MD5).

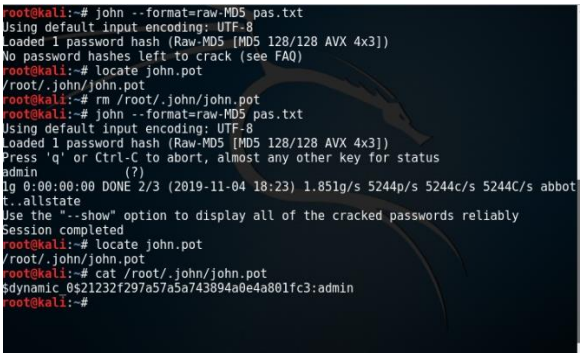


Fig 4 Save the password to text file

PREVENT ATTACK

SQL injection is a hacking that was discovered more than fifteen years ago and is still proving to be devastatingly effective today. SQL is the command and common language for relational databases such as Oracle, and MySQL. In modern web development, these databases are often used on the back end of the web applications and content management systems written in PHP. ASP.NET or other scripting languages.

Steps to prevent SQL injection attacks:

1. Trust no one: Assume that all user input data is evil and use proper validation**.**

2. Don’t use dynamic SQL :Don’t construct queries with user input.

3. Update and patch: Vulnerabilities in application and databases that hackers can exploit using SQL injection are regularly discovered.

4. Firewall: use a web application firewall.

IMPLEMENTATION

SHSED ALGORITHM

A simple and highly secure encryption decryption (SHSED) algorithm that can be used for cloud computing-based

applications, where introduced. It uses simple and efficient logical operations, such as XORing, addition, and subtraction in addition to byte shifting. The introduced, introduced method is also powered by the flexibility of selecting the secret key length and the number of rounds to generate the cypher text. Experimental results for the introduced method were compared with other methods results (LED, AES, DES), and as a result of comparisons the introduced method give a good improvement in encryption-decryption time measurement.

For the implementation and testing of SHSED, the personal computer (PC) is used for installation and execution, with the following technical features:

1. Operating system: Windows 10 home 64-bit.

2. Processor: A10-9600P RADEON R5, 10 COMPUTE CORES 4C+6G GHz.

3. Installed Memory (RAM): 6.00 GB

IV.RESULT

Speed gain comparison for SHSED with respect to AES, DES, and LED

	Speed up	
	Encryption	Decryption
AES	4.4904	6.4151
DES	0.5388	0.7615
LED	1.2310	2.2301

Result analysis of rounds for the Introduced SHSED

Rounds	Encryption time(Seconds)	Decryption time(seconds)
2	0.0710	0.0258
3	0.0706	0.0265
4	0.0778	0.0277
5	0.0730	0.0316
6	0.0747	0.0328
7	0.0715	0.0288
8	0.0732	0.0307
16	0.0762	0.0362
20	0.0787	0.0385
24	0.0793	0.0453
30	0.0820	0.0441

V. CONCLUSION

A simple light weight and highly secure encryption_decryption (SHSED) method was introduced and it can be applicable for various data processing applications.The speed up obtained by the proposed (SHSED) method compared with AES and the lightweight LED algorithms are encouraging for a practical application as the efficiency was 4.4 times for encryption and 6.41 times for decryption in the case of AES algorithm. It is also 1.23 times faster than LED for encryption and 2.23 times for decryption in the case of LED. However, it was slightly slower than DES.

VI. FUTURE WORK

Combinations of algorithms can be used to increase the speed and efficiency of encryption-decryption. So the future work can be done in this context to reduce the time for encrypting and decrypting the passwords.

VII. REFERENCES

1. ziad alqadi, “Analysis of stream cipher security algorithm” Journal of Information and Computing Science, Vol. 2, No. 4, pp 288-298.2007..

2. Jean Raphael NgnieSighom, Pin Zhang and Lin You, “Security Enhancement for Data Migration in the Cloud,” Future Internet, P 1-13, 2017...