Advances in Computing 2017, 7(2): 23-27
DOI: 10.5923/j.ac.20170702.03

Protection of Password using Distributed Clustered Division and Encryption

Aminath Haneena*, Supreetha R.

Department of Computer Science & Engineering, St Joseph Engineering College, Mangaluru, India

Abstract This paper presents a secure technique for the password protection which make use of distributed database. Security to the password has become a crucial problem in this generation as more and more attacks are focused on the password to get access to user's confidential data. Here the password is split into parts and is stored in different database instead of storing the entire password string in one database. The password is stored after encryption, the encryption algorithm used is itself improvised by using a pseudorandom key generator. The encrypted string is split into three, the split is reversed and stored into separate databases. The paper shows the comparison of the improved algorithm made use. The technique of distributed storing enhances the security by making it difficult for the hacker to get the user password easily.

Keywords Distributed partitioning, Encryption, Password, Improved AES algorithm, Cryptography

1. Introduction

Security of the data is one big issue in the development of the communication network. The data stored in the server database and propagated in the network might contain individual privacy information. The password which the user uses in the day to day life is mostly compromised to the hackers trap. Password is being most common authentication technique which provides the access to system resources. Password is the simplest form of authentication technique used, the probability of cracking the password using different combinations is considerably high. Password encryption provides security to a considerable level.

In this a technique is proposed to increase the protection of the user password from unauthorised users. The password of the user is stored in the server database using distributed partitioning of the data where the parts of user password is stored in different database. This is done focusing on the security of the data so that it does not be prone to attackers easily. Earlier the password is stored as it is in plain text format into the server database. The strength of the password does not give any security to the user account. The password constituting of large complex string or a weak password is all the same for the attacker if he gets access to the database where the password is stored.

Along with partitioning of the user password, it is encrypted using an enhanced AES (Advanced Encryption Standard) algorithm with gives a stronger encrypted cipher text. This enhanced algorithm is known as improvised AES algorithm. Encryption is that secret key and plaintext get through complex arithmetical operation to form cryptograph [11]. All plaintext are hidden in cryptograph. Plaintext is the data which is protected. Secret key is generated through encryption algorithm. Cryptograph is transmitted to receiver through channel after encryption success. Decryption is that receiver computes plaintext using cryptograph and secret key. A safe encryption algorithm can be described as follows. Although attacker captures parts of cryptograph or all cryptograph, attacker cannot restore plaintext in limited time and limited resource [12].

* Corresponding author:
haneenaidrise@gmail.com (Aminath Haneena)
Published online at http://journal.sapub.org/ac

2. Related Work

Clustering of the data based on the frequency of accessing the information minimizes and enhances the efficiency of the database. Clustering technique overcome the drawbacks of the database systems which are centralized and the size of the information stored each day increases drastically [1].

An improved AES (Advanced Encryption Standard) algorithm which gives an improved encryption algorithm to increase the security. There were lots of security issues with the AES encryption since it make use of the secret which is public. The new encryption algorithm is based on chaos theory which makes use of a pseudorandom key instead of previously used public key in AES encryption [2].

An improvised approach for plain text password encryption in the server's database. One of the major aspect of password protection issue is to secure it by means of encryption process. A new approach for improvising the scheme of password encryption is using the process of

Jumbling-Salting (JS). The jumbling process selects characters from pre-defined character set and adding them into the plain password using mathematical modulus (%) function; salting comprises of adding a random string into jumbled password [3].

Proposed method is implemented on different cryptograph that is symmetric, asymmetric and hash function. Comparison of performance of different encryption algorithms. The main idea behind every encryption standard is to give security to the information from breaches. In practical this is determined by the applications performance, speed and the cost. An encryption algorithm which provides security but is very slow in case of performance cannot be considered to be much efficient to use in practical. One major reason for the degradation of performance by the encryption system is due to the embedding it to applications [4].

Presents a software application for studying of the most used methods for text encryption/decryption, with utilization in teaching activities and didactical laboratories. The applications is realized using virtual instrumentation software and was designed as a didactical software platform for studying and analyze the two of the most used encryption techniques, the symmetrical and asymmetrical encryption algorithms. These encryption and decryption techniques are applied to encrypt and then decrypt a given text, for an easily understanding and verification of the encrypting/decrypting techniques implemented. [5].

Data Security has become a crucial issue in electronic communication. Secret writing has come up as a solution, and plays a vital role in data security system. It uses some algorithms to scramble data into unreadable text which might be only being decrypted by party those having the associated key. These algorithms consume a major amount of computing resources such as memory and battery power and computation time. This paper accomplishes comparative analysis of encryption standards DES, AES and RSA (Rivest, Shamir and Adelman) considering various parameters such as computation time, memory usages. A cryptographic tool is used for performing experiments. Experiments results are given to analyses the effectiveness of symmetric and asymmetric algorithms [6].

The hardware implementation of AES Rijndael Encryption and Decryption Algorithm by using Xilinx Virtex-7 FPGA. The hardware design approach is entirely based on pre-calculated look-up tables (LUTs) which results in less complex architecture, thereby providing high throughput and low latency. There are basically three different formats in AES. They are AES-128, AES-192 and AES-256. The encryption and decryption blocks of all the three formats are efficiently designed by using Verilog-HDL and are synthesized on the proposed architecture is found to be having good efficiency in terms of latency, throughput, speed/delay, area and power [7].

A constructive theory of randomness for functions, based on computational complexity, is developed, and a pseudorandom function generator is presented. This generator is a deterministic polynomial-time algorithm that transforms pairs (g, r), where g is any one-way function and r is a random k-bit string, to polynomial-time computable functions f. The result has applications in cryptography, random constructions, and complexity theory [8].

A high throughput modified Advanced Encryption Standard (AES)-128 bit algorithm is implemented. A new increased parallelism technique is introduced in modified AES architecture in Mix Column round which increases the overall throughput of AES algorithm [9].

3. Existing System

The application which makes use of password for the authentication of user privacy data usually makes use of two types of method to store the password of the user in the server.

In first method, the user password is stored directly in the plain text format into the database, as entered by the user. Storing in this fashion does not give any security to the user password or the user account which might contain confidential information. Once the hacker get access to the database which contain the user password, the password which contain numerous character and the password which has less character is one and the same for the hacker if its directly stored as plain text to the database. The password is straight away available to the hacker and he can misuse it.

The second method is where the user password is stored into the database after encryption. For the encryption different cryptographic algorithms can be made use. Encryption makes it harder for the attacker to get the user password easily even though he force into the database. But encryption does not safeguard the password all the time. If the attacker identifies the key used to encrypt the password, then it will take only minutes to decrypt the password into plain text by using different combination of the keys.

3.1. AES Encryption

AES is based on Rijndael Cipher. Rijndael is a family of ciphers with different key and block sizes in multiple of 32 bits. AES is a symmetric-key algorithm, because the same key is used for both encrypting and decrypting the data. AES operates on a 4×4 array of bytes, called the state. In encryption, each round consists of four stages except the last round which excludes mix column round.

Sub Bytes - a non-linear substitution step where each byte is replaced with another according to a lookup table (known as S Box).

Shift Rows - a transposition step where each row of the state is shifted cyclically a certain number of steps.

Mix Columns - a mixing operation operates on the columns of the state, using a linear transformation.

Add Round Key - each byte of the state is combined with the round key; each round key is derived from the cipher key using a key schedule [9]. The flow chart for AES encryption is shown in Figure 1.

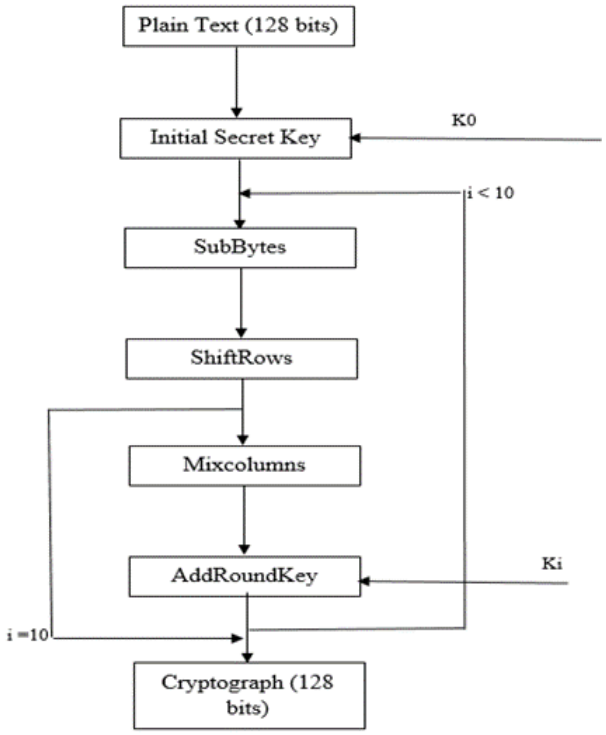

Figure 1. Flow chart for AES encryption

4. Proposed Methodology

This work consist of three different modules (Figure 2):

1. **Clustered Database:** The main reason behind the clustered database is to give protection to the data. When the password is stored in multiple database, it increases the hardness in accessing the password by the unauthorized person. The unauthorized person may not be aware about the different database in which the password is stored. These database might be located in a same location area or in different cloud based servers.
2. **Encryption/Decryption of data**: Encryption is a process where the plain text is converted into a cipher text using the secret key. In this work, an improved AES algorithm is made use to encrypt the password string given by the user. This algorithm is proved to be better than the AES algorithm as it make use of random key each time the encryption takes place.
3. **Splitting and reversing of data:** Splitting and reversing of the password text is done in order to increase the security of the data. This module makes it difficult for the hacker to get the plain user password easily as the user encrypted password is split into different parts. The characters from the split part is reversed and these are stored in a database. In the same way, other remaining splits are also reversed and stored into n different database.

During the retrieving of the data, all the split is merged and rearranged in order to get the user password. This is decrypted and compared with the original password.

4.1. Improved AES Encryption

The Encryption generally used is public as it makes use of public secret keys. Attackers can get secret key according to secret key expansion algorithm if they get one round secret key. Thus it makes easier for the attacker to compute the plain text. Hence, standard AES encryption algorithm has safety problems.

For improving the safety of AES encryption algorithm. This work put forth an improved/ehanced AES encryption algorithm which use pseudorandom sequences which have enough key length generated by Logistic mapping. Improved AES is an asymmetric encryption standard which is used to encrypt a plain text of 128 bits into cipher text to give more security to the data using random logistic cipher keys.

The bits generated from pseudorandom logistic generator is cut down into 128 bits to match with the 128 bit plain text which is mapped together to give a cipher text. The working of improved AES algorithm is shown in the Figure 3 [2].

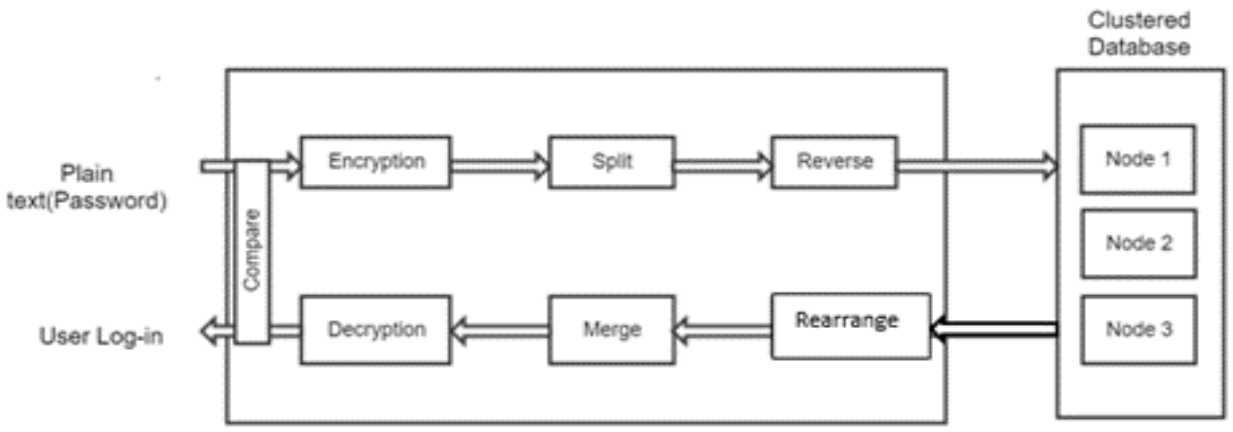

Figure 2. System modules

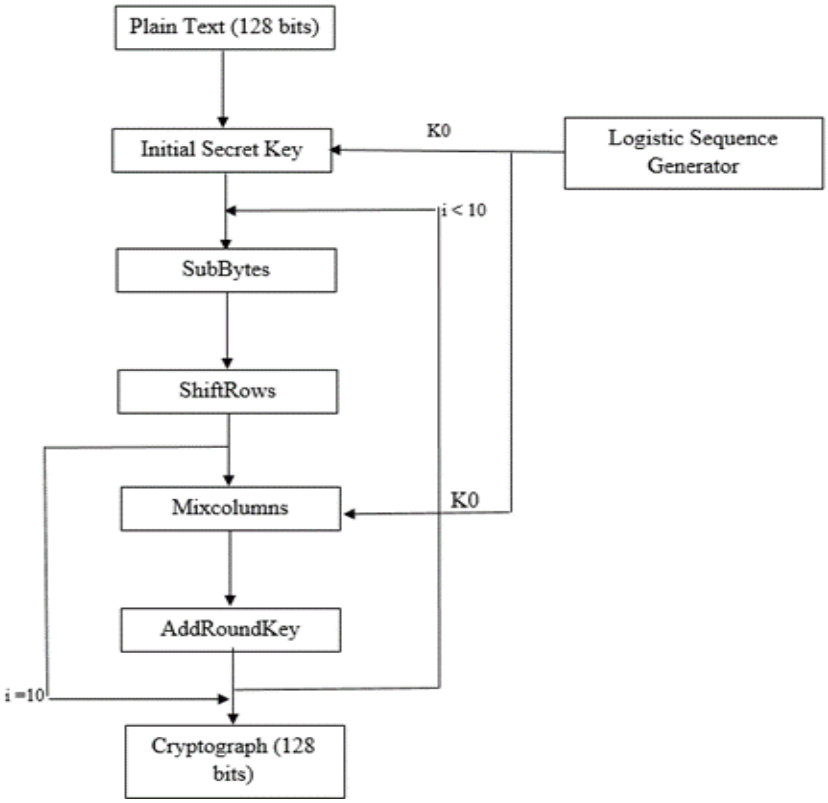

Figure 3. Improved AES Encryption

Decryption is the process of getting back the plain text from the cipher text making use of the secret key encapsulated within the text. Here, for the purpose of decryption it uses the same pseudorandom key which is used to encrypt the plain text.

5. Implementation and Performance Analysis

Improved algorithm used in this work makes use of different keys for each time the encryption takes place. Each round makes use of unique keys. The unique is obtained from the sequence logistic key generator which generates keys randomly for each rounds in the encryption process.

The experiments conducted shows that the enhanced algorithm produces a cipher text longer than the standard algorithm usually made use of. From the analysis it's concluded that the password with shorter length, more special character took more time to encrypt than for the password which consist of more character string. The enhanced AES algorithm was proved to consume less time to encrypt than the AES algorithm which was taken for comparison in this work.

Overall, improved algorithm took minimum time to encrypt and produce a cipher text which is represented in Table 1 and the same is plotted in the Figure 4.

Table 1. Encryption time

Password Text Size	AES	Improved AES
Encryption Time (ns)		
3 (P1)	163	132
4 (P2)	133	115
5 (P3)	138	82
6 (P4)	111	75
8 (P5)	123	125
10 (P6)	162	112
12 (P7)	100	95
14 (P8)	130	122

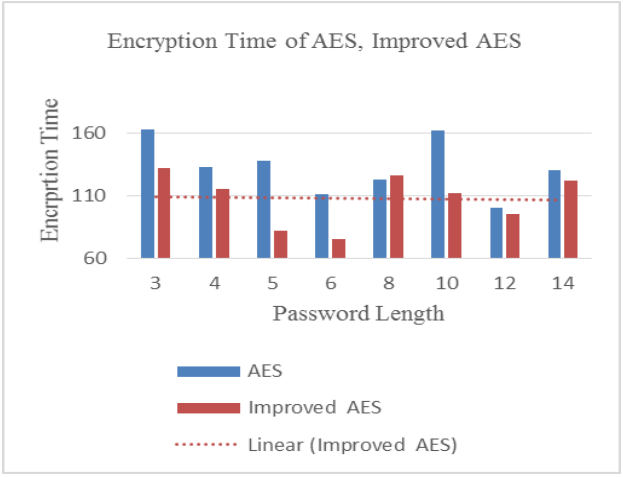

Figure 4. Encryption performance

5.1. Throughput

The throughput is obtained by taking the encryption time and the various password string sizes which is used in the experiment. From analysis it was found that the average throughput for AES was 0.0433 Kb/s whereas the average throughput for improved algorithm was 0.0755 Kb/s. The throughput for the algorithms increases if the encryption time increases. Table 2 and Figure 5 shows the throughput of two algorithms.

The throughput of encryption is calculated by, R = N/T
R = the processing rate
N = the password text size and T = time consumed

Table 2. Throughput rate

Password Text Size	AES	Improved AES
Throughput rate (Kb/s)		
(P1)	0.00184 Kb/s	0.0227 Kb/s
(P2)	0.0300 Kb/s	0.0347 Kb/s
(P3)	0.0362 Kb/s	0.0609 Kb/s
(P4)	0.054 Kb/s	0.08 Kb/s
(P6)	0.0617 Kb/s	0.0894 Kb/s
(P7)	.012 Kb/s	0.1263 Kb/s
(P8)	0.1076 Kb/s	0.1147 Kb/s

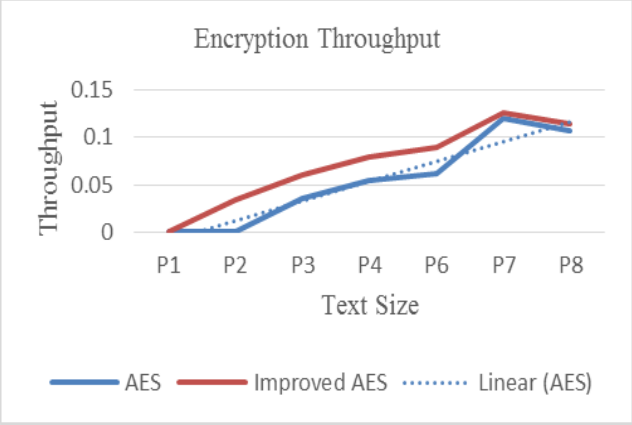

Figure 5. Throughput rate

6. Conclusions

Storing the password using the clustered database enhances the protection of the password from unethical people who are interested in misusing the confidential information of other individuals. This technique can be used in the area where the security of the password is a major concern.

The new enhanced algorithm shown in this work helps in giving a high level of security to the data by using random key generator which produce unique keys alongside reducing the encryption time. The throughput ratio of encryption in case of this algorithm is linear. The size of the encrypted password is larger than the other algorithms.

REFERENCES

[1] Md. Hafizur Rahman, Faisal Bin Al Abid, M. N. Zaman and Md. Nasim Akhtar, "*Optimizing and Enhancing Performance of Database Engine Using Data Clustering Technique*" International Conference on Advances in Electrical Engineering, December, 2015.

[2] Zi-Heng Yang, Ao-Han Li, Ling-Ling Yu, Shi-JunKang and Meng-Jiang Han, "*An Improved AES Encryption Algorithm Based on Chaos Theory in Wireless Communication Networks*", Third International Conference on Robot, Vision and Signal Processing, 2015.

[3] Prathamesh v Churi and Anchor Kutchhi, "*Jumbling- Salting: An Improvised Approach for Password Encryption*", International Conference on Science and Technology, 2015.

[4] Aamer Nadeem and Dr M Yonus Javed, "*Performance Comparison of Data Encryption Algorithms*", IEEE, September-2005.

[5] R.M. Teodorescu, I. Lita, I.B. Cioc and D.A Visan, "*Virtual Instrumentation Application for Symmetrical and Asymmetrical Text Encryption/Decryption Studying*", International Conference, 7th Edition Electronics, Computers and Artificial Intelligence, June-2015.

[6] Priteshkumar Prajapati, Nehal Patel, Robinson Macwan, Nisarg Kachhiya and Parth Shah, "*Comparative Analysis of DES, AES, RSA Encryption Algorithms*", International Journal of Engineering and Management Research, February-2014.

[7] N.S. Sai Srinivas and MD Akramudin, "*FPGA Based Hardware Implementation of AES Rijndael Algorithm for Encryption and Decryption*", International Conference on Electrical, Electronics, and Optimization Techniques, 2016.

[8] Goldreich Oded, Goldwasser S and Micali S, "*How to construct random functions*", Laboratory for Computer Science, MIT, Cambridge, November-2015.

[9] S.Sridevi Sathya Priya, P.Karthigai Kumar, N.M. SivaMangai and V. Rejula, "*FPGA Implementation of Efficient AES Encryption*", IEEE Sponsored 2nd International Conference on Innovations in Information Embedded and Communication Systems, March-2014.

[10] B. Hari Krishna and Raja Shekar Reddy, "*Multiple text encryption, Key entrenched, distributed cipher using pairing functions and transposition ciphers*", IEEE, 2016.

[11] A. A. Hasib, "A comparative Study of the Performance and Security Issues of AES and RSA Cryptography," Third 2008 International Conference on Conver and Hybrid Information Technology, 2008.

[12] R. Matthews, "A High through Put Low-Cost AES Processor," in Cryptologic, 1984.