International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 8
August 2017

Security Layer Implementation in EnDeCloudReports Simulator Tool through Modified AES

Shweta Singh
PhD Scholar (CS), IIS University, Jaipur

Dr. Amita Sharma
Asst. Professor (CS), IIS University, Jaipur

***Abstract**: The present research proposes an enhancement in the existing cloud simulator tool CloudReports tool (2014) which is based on cloudsim. CloudReports simulator tool simulates the cloud environment and presents the utilization of hardware resource through graphs and reports. The file comprising the hardware utilization information may be responsible for the development of malwares like Stuxnet for IT industry. In this research paper, the proposed modular architecture of the enhanced simulator tool is presented. The experiments designed and implemented for the selection of best encryption algorithm for implementation of security layer in EnDeCloudReports simulator tool is discussed in the present study. In the present research, improvement and modifications in the existing 128-bit AES algorithm are proposed to achieve a robust and secure encryption algorithm. The results of the experiments are compared on the basis of encryption/decryption time, throughput time, processing time for large file sizes ranging from 10 MB to 120 MB text file sizes.*

Index Terms- Cloudsim, CloudReports, AES, Encryption/decryption time, text file, modified AES, Encryption algorithm, EnDeCloudReports tool, Security layer, throughput time.

I. INTRODUCTION

CloudReports simulator (2014) by Zhaigam Mehmood simulates the cloud environment and presents the utilization of hardware resource through graphs and reports. It is based on the cloudsim simulator tool and the graphs in this tool are generated on the basis raw files restored on server. The raw file consists of hardware resource utilization information of the simulated cloud environment. Such files were found to be responsible for development malwares like Stuxnet as stated in "Lessons from Stuxnet" by Thomas M. Chen, Saeed Abu-Nimeh (2011).

Industrial control systems controlled through software are vulnerable to malware attacks like Stuxnet as such malwares are developed on the basis of hardware resource utilization information. In IT industry such malware attacks can be prevented through encryption of such information kept in raw form (readable) on servers like in cloud environment IT field.

The present research proposes enhancement in the existing CloudReports simulator tool through implementation of security layer for the encryption of raw files kept on server in the tool. The best suitable symmetric block cipher algorithm is selected on the basis of certain parameters to be used for encryption of such hardware resource utilization information in the proposed modular architecture of cloud simulator tool. From computation point of view Symmetric-key algorithms take less time than Asymmetric key algorithms. Often, symmetric key algorithms are a thousand times fast than those of the asymmetric algorithms. [6] Hence, the best suitable method to encrypt the data is, to encrypt it with symmetric key encryption algorithms. The hardware implementation of the symmetric block cipher AES algorithm is very fast. In the next section, the modular architecture of the enhanced simulator tool EnDeCloudReports and the experiments for the best suitable symmetric encryption algorithm for implementation of security layer are discussed.

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 8
August 2017

II. LITERATURE SURVEY

In this section the work published by various researchers in the field of cryptography algorithm and enhancements for cloud simulator tool is discussed. From this survey various gaps have also been identified and defined in section III.

Monika Agrawal et al. 2012 present a detailed study of the well-known symmetric key block cipher algorithms like Blowfish, DES, TRIPLE DES and AES. The speed of symmetric key algorithms is fast when compared with asymmetric key algorithms such as RSA etc. The memory requirement of symmetric key algorithms is lesser than asymmetric encryption algorithms. Further, from security point of view symmetric key encryption is better than asymmetric key encryption.

Gast.M.S (2002),"802.11 Wireless Network: The Definitive Guide, " presented a study which is conducted for various popular secret key algorithms such as RC4, AES, and XOR. They were executed, and their performance was evaluated through encryption of real time video streaming of varying contents. The results showed; AES took less encryption delay overhead when compared with the overhead using RC4 and XOR algorithm. Hence, AES proved itself as the most feasible solution to secure real time video transmissions.

Bala R, Gopalan N (2016) stated in their research paper that information is made illegible through cryptography whereas Steganography ensures that no evidence to determining information exists. In the field of cryptography modifications in AES algorithm are proposed to encrypt data and in Steganography reversible texture synthesis to hide the data is implemented.

Lakshmi R & Mohan HS (2015) stated in their research paper the performance comparison of existing Rijndael AES algorithm with modified AES algorithm through diffusion analysis in terms of First order Strict Avalanche Criteria (SAC) and Higher order SAC. This paper provides the use of dynamic S-Box which is dependent on the key provided by the user. No file size is mentioned for comparison of results.

Ao Zhou, Shangguang Wang*, Qibo Sun, Hua Zou, Fangchun Yang(2013) published that CloudSim 1 is a cloud simulation toolkit developed by the CLOUDS Laboratory of University of Melbourne. CloudSim holds the simulation of a virtualized cloud data center. Experiments on cloud computing infrastructures can be implemented through Cloudsim. CloudSim is an extensible simulation tool. The existing functionalities contributed by CloudSim can be extended by researchers and addition of new features to CloudSim, including CloudAnalyst, CloudSimEx, WorkflowSim, CloudAuction and DynamicCloudSim among others is also possible. However, DynamicCloudSim2 can grant fault -tolerance in some ways, but at present it can only determine whether a task succeeds or fails. At present, none of the available current cloud simulator tools can appropriately simulate Cloud Service Reliability Enhancement Mechanisms.

However, there is a shortage of tools that enable researchers to evaluate their new proposed cloud service reliability enhancement mechanisms. This study is presented to fill this gap. The basic functionalities of CloudSim are extended and FTCloudSim is proposed in this paper. FTCloudSim serves an extensible interface to benefit researchers in implementation of new cloud service reliability enhancement mechanisms. In addition, FTCloudSim can also study the behavior of the new proposed mechanisms.

Thomas M.Chen, Saeed Abu-Nimeh (2011) in “Lessons from Stuxnet” presented the analysis of Stuxnet malware attack. The main reasons for the development of Stuxnet malware was the hardware resource utilization information restored in raw form.

III. GAP IN STUDY

In industrial control systems controlled by software the hardware resource utilization information is not kept in encrypted form. This is one of the main reasons for the development of malwares like Stuxnet, as analyzed by researchers. In IT industry, such malware attacks can be prevented by encrypting such information responsible for malware attack. Cloud environment controlled through software can be secured against such attacks by encrypting the hardware resource utilization information. Also, the existing cloudsim simulator tools are missing the security implementation for security against malware attacks. The existing symmetric ciphers like DES, Triple DES and AES are required to encrypt the hardware resource utilization information.

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 8
August 2017

At present, researchers have proposed modifications in AES symmetric block cipher algorithm for enhancing its efficiency and security. But, the modifications are evaluated for small text file sizes up to a maximum size of 10 MB.

IV. MODULAR ARCHITECTURE OF PROPOSED ENDECLOUDREPORTS SIMULATOR TOOL

The proposed modular architecture of EnDeCloudReports consists of the modules as shown in figure 1. The modules of the EnDeCloudReports shown below are the same as in CloudReports with the only difference of implementation of security layer. The security layer proposed for the simulator tool consists of the existing symmetric encryption algorithms as well the modified version of standard Rijndael algorithm. The symmetric encryption algorithms chosen for implementation of security layer on the basis of literature review are shown in figure 2.

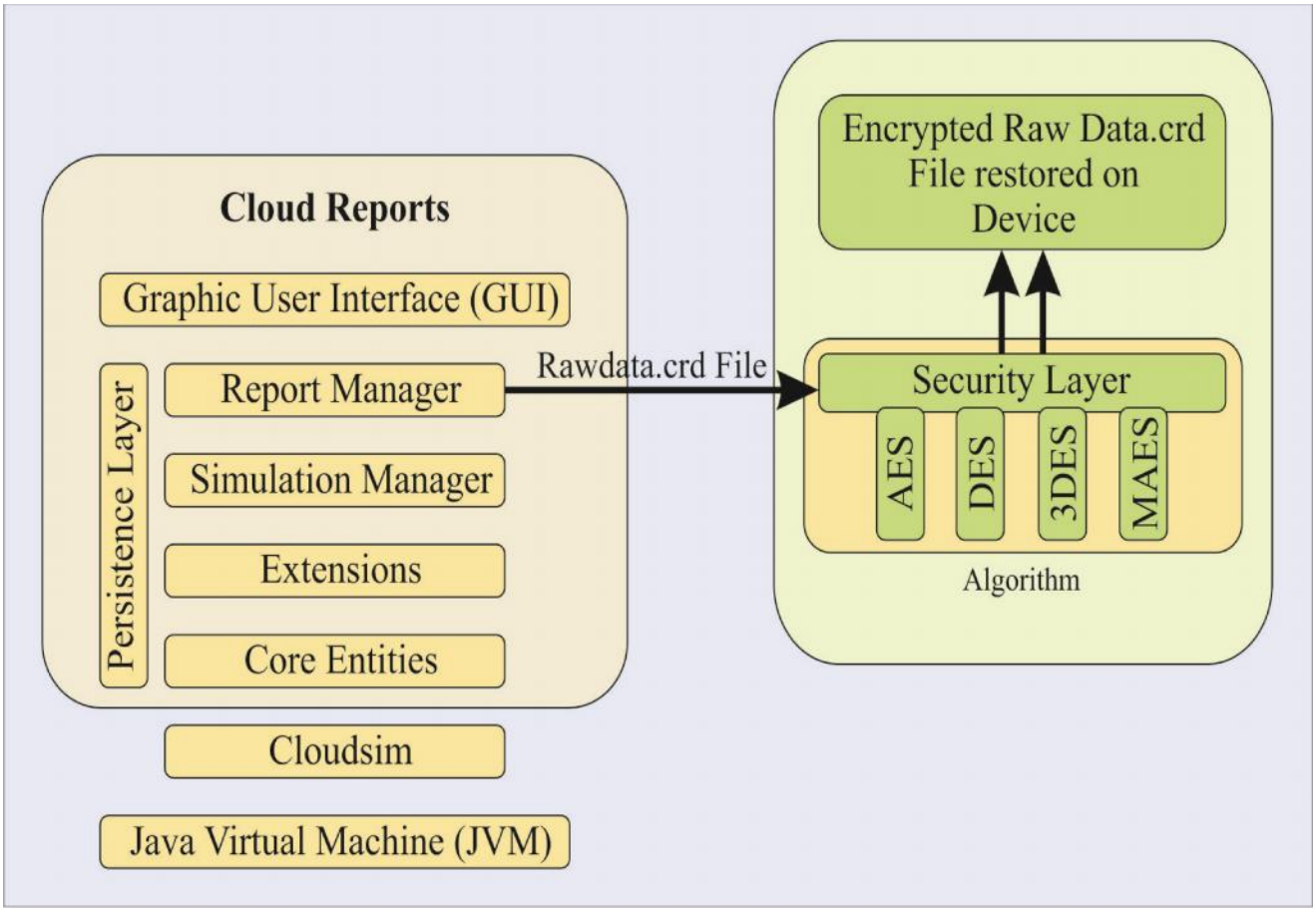

Figure 1: Modular Architecture of EnDeCloudReports simulator tool

Basically the function of security layer in the modular architecture of the enhanced simulator tool is to restore files in encrypted form generated by the Reports manager module. The raw files comprising the hardware resource utilization information responsible for generation of graphs are also responsible for development of malwares like Stuxnet. To prevent such attacks in IT industry, the implementation of security layer is proposed through this enhanced simulator tool EnDeCloudReports.

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 8
August 2017

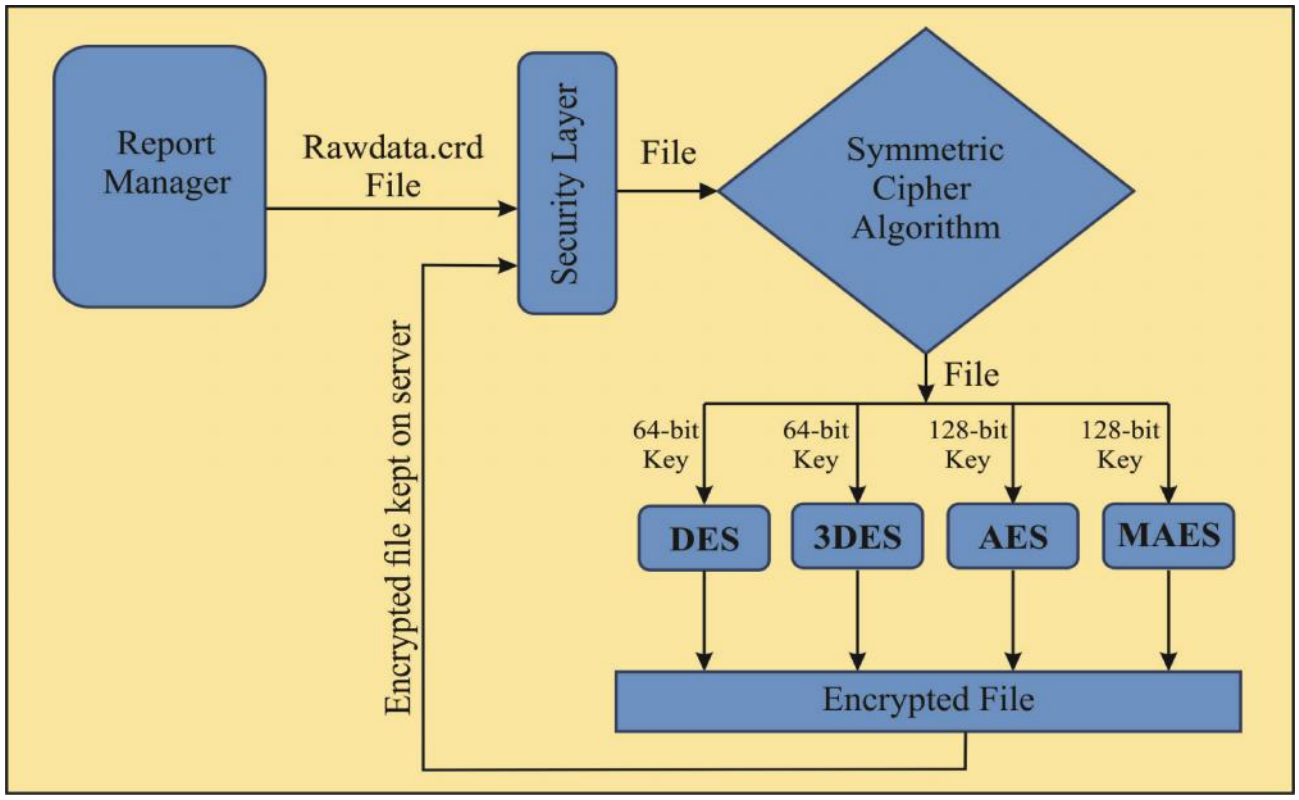

Figure 2: Implementation of Security Layer in modular architecture of EnDeCloudReports

In the next section, the modification proposed for the existing AES algorithm is discussed.

V. PROPOSED MODIFIED AES ALGORITHM

The present research proposes certain modifications in standard Rijndael algorithm for the implementation of security layer in EnDeCloudReports tool. The layers of the modified AES algorithm are discussed below:

1. **Key Addition layer** The 128-bit round key, or sub key, which is being derived from the original key in the key schedule, is used for the function of XOR to the state. In the 128-bit MAES there are two keys generated to increase the complexity of the proposed algorithm.

2. **Byte Substitution layer (S-Box)** Each element of the state is nonlinearly transformed using lookup tables (S-Box) with special mathematical properties. This introduces *confusion* to the data, i.e., it assures that changes in individual state bits propagate quickly across the data path.

3. **Diffusion layer** It provides *diffusion* over all state bits. It comprises of two sub layers, both of which perform linear operations:

3.1 The data in Shift Rows is permuted on a byte level.

3.2 The *MixColumn* layer is a matrix operation which combines (mixes) blocks of four bytes in each column of 4x4 matrixes for alternate rounds i.e. mix column is skipped for odd round number.

The key schedule computes round keys, or sub keys, (k0, k1, . . . , knr) from the original MAES 128-bit key.

In the next section, results after execution of 128-bit MAES are compared with 128-bit AES algorithm.

In the next section of this research paper, the comparison of experiments and results implemented for implementation of security layer are discussed briefly.

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 8
August 2017

VI. RESULTS AND DISCUSSION

The existing symmetric encryption algorithms like DES, Triple DES and AES were executed in Java language. The performance of these block cipher algorithms is compared on the basis of certain parameters viz. encryption/decryption time taken for encrypting raw data files restored on server in simulator. The first experiment takes text file size 10.4 MB as input and gives an encrypted file to restore on the server. The comparison of time taken in encryption/decryption of the file is shown in Table 1 and corresponding graph in Figure 3.

File Size	11,001,387 bytes	10.4 MB
Algorithm	**Encryption (Sec)**	**Decryption (Sec)**
AES	22	37
DES	60	59
3DES	246	247

Table 1: Results of Experiment 1

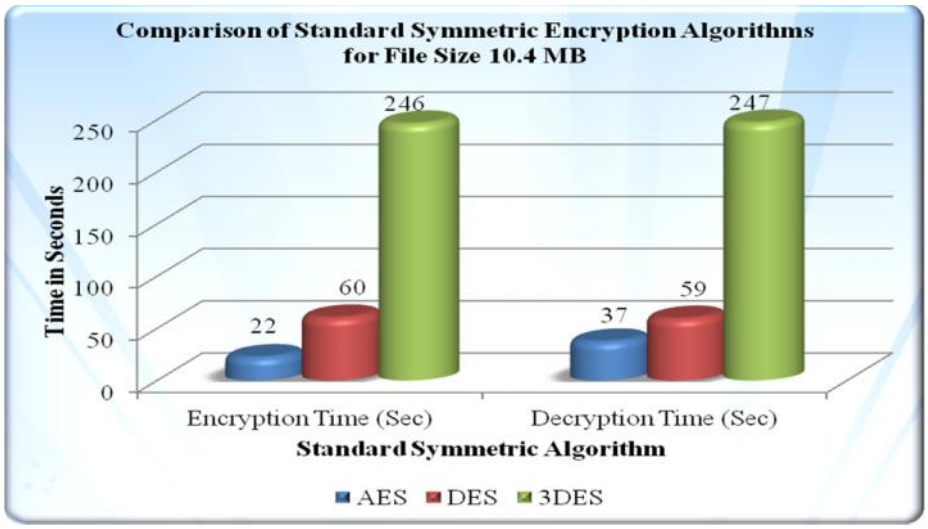

Figure 3 : Comparison of Results of Experiment 1

Table 1 shows the results of Experiment 1in terms of seconds taken for encryption and decryption of file when file size is 10.4 MB. 3DES took 246 seconds for encryption and 247 seconds for decryption. DES took 60 seconds for encryption and 59 seconds for decryption. AES took 22 seconds for encryption and 37 seconds for decryption.

Figure 3 shows the pictorial representation of Experiment1and proves that AES took the minimum the time for encryption and decryption of file size 10.4 MB when compared with DES and TDES on same parameters.

AES is the best algorithm when compared on the basis of time taken for encryption. But, recently researchers have proposed modifications in existing AES to make it more secure and robust. As cryptanalysts are constantly working on techniques to break encryption algorithms, the direction of research is motivated to modification in existing AES.

The results of the proposed modified AES and existing AES are compared on the basis of encryption/decryption time. Both, 128-bit MAES and AES algorithm executed in eclipse environment are used to encrypt raw files of 10.4MB in size. The encryption/decryption time taken by them is shown in Table 2 and corresponding graph in figure 4.

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 8
August 2017

File Size	11,001,387 bytes	10.4 MB
Algorithm(128-bit key)	**Encryption (Sec)**	**Decryption (Sec)**
AES	22	37
MAES	16	19

Table 2: Comparison table of Results of 128-bit Modified AES and AES

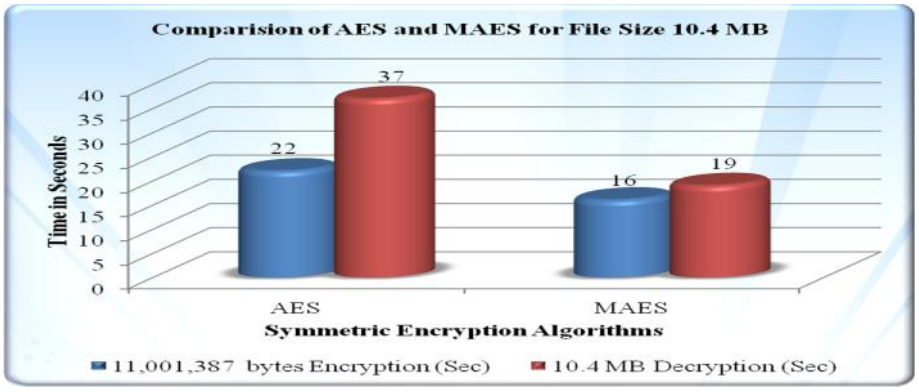

Figure 4: Comparison of performance of MAES and AES

Table 2 shows the results of Experiment 2(a) in terms of time taken for encryption/decryption in seconds for same file size of 10.4 MB by MAES and AES. Figure 4 shows the pictorial representation of Experiment 2(a) and proves MAES to be efficient in terms of encryption/decryption time when compared with existing symmetric encryption algorithm AES for same file size.

VII. FUTURE WORK AND SCOPE

In near future the present study will target to increase the security as well as complexity of the 128-bit MAES. Also, the performance of existing algorithms and the proposed symmetric cipher 128-bit MAES will be compared for different and large raw data files generated in the simulator tool. The parameters of comparison are limited in the present study to encryption/decryption time. In future, the parameters for comparison in this study will be extended to throughput analysis and processing time taken by existing and proposed algorithm.