14th International Conference on Recent trends in Engineering, Applied Science and Management
Vedant College of Engineering & Technology, Bundi, Rajasthan (India) (ICRTESM-19)
16th February 2019 www.conferenceworld.in ISBN : 978-93-87793-71-2

# SECURE AND EFFICIENT INFORMATION RETRIEVAL FROM CLOUD USING MINING PROCESS

Aditya Sharma[1], Dr. Amit Sharma[2]

[1]*Mtech Scholar Computer Science & Engineering Department Vedant College of Engineering and Technology Bundi, Rajasthan, India*

[2] *Professor, Computer Science & Engineering Department, Vedant College of Engineering and Technology, Bundi, Rajasthan, India*

***Abstract:-*** *In current trade of international as well nation companies using the cloud storage for maintainingthe data. Even though the cloud continues to gain popularity in usability and attraction, the problems lies with data confidentiality, loss of control, lack of trust, data theft and the fact that user data is stored in unencrypted format. So security is always a concern. Lots of works related to save the data to be down but still the doubt is in the environment so still many want more secure and faithful data communication to done through Cloud. In the paper a secret keys to be used with the data into the cloud once the data is encrypted stored it retrieved in encrypted manner when owner accepts a request by an authorized user, and an application server provides an Access key. Using an access key, a user downloads data and uses a secret key to convert cipher text into a plain text. Using this technique end-to-end encryption to be secure which completely hides the data from cloud service providers hence maintain confidentiality. Implementation involved building an encryption application algorithm, for deployment on the user computer, which consists of a single encryption and hybrid encryption modules. It established the trust between data owners and cloud service providers.*

***Keywords : Cloud computing, Cloud security issue, Data mining, Classification,***

## I INTRODUCTION

Cloud computing is a decent stage for information mining. It can counterbalance imperfections of past strategies in breaking down system information. In cloud, assets for capacity and registering were dispersed. In this way, information mining in cloud is led fundamentally not quite the same as the conventional mining worked on neighborhood PCs, and meets the necessities of information mining in Internet.

The framework of distributed computing is built of significant server groups, which supply the cloud with ground-breaking limits of registering, putting away, information examining and information, the executives. These limits give basic preconditions to monstrous information mining on the Internet. What's more, IT assets and applications are given as open offices in cloud. As the manner in which you utilize water, power and gas, you can utilize assets and application in cloud without thinking about where they originate from and how to create them.

A distributed storage framework comprise gathering of capacity servers, through the web it gives various administrations to the seller. The encoded client's information in cloud ought to be secret and verified plan. To ensure secrecy, expected to structures encryption standard, yet in addition limit the usefulness of the capacity plot on the grounds that a specific tasks are ceaseless over encoded information. To develop a secured stockpiling plan that underpins numerous reasons for existing is extreme when the capacity plot is circulated and has no focused expert. The circulated distributed storage plot gives secure and strong information stockpiling, recovery and furthermore allows a customer store and recovers information in the capacity servers to another customer. The point is to decrease the time usage to get to the information to the vendor's, forward the inquiries between the merchant's and the capacity server, in light of the fact that the machine based methodology is utilized to coordinates scrambling, encoding and sending the seller's information. The information are encoded dependent on the machine abilities through the ZIPF computerization instrument and it's put away in the cloud utilizing Generalization plot. Transfer speed dispensed to the information work from example grouping, it enhances the productivity of the information stockpiling. Questions are proficiently handled by Erasure code framework, which naturally evacuate the inquiry ask for put away in the line support in the wake of getting the inquiry reaction.

## II PROPOSED ARCHITECTURE

The System architecture for this model presents a conceptual model while defining views, structure and behavior. In this research, the architecture presents, layout of the model showing how modules interacts, how data flows and hardware and software building blocks composition. Model architecture enables the implementation, understanding, maintenance, repair and further development of the model under study. The System Architecture of the cloud repository system shown in down side Figure describes various components and communication between those components. A user as depicted in the system architecture, shall be authorized to login to the local user computer domain.

The key generation and encryption module is where a selected encryption algorithm generates secret key. This module converts a plain text into cipher text of a file that is be uploaded to the cloud storage. This process automatically generates a secret key. This module resides at the client side of the model. The encrypted file is then uploaded to the cloud repository system residing at the cloud side of the model.

Cloud repository system is a cloud storage infrastructure. It hosts cloud storage servers with the backup infrastructure to ensure continuity. A cloud user rents this space from cloud service provider and pay per use for the storage of uploaded files. This model requires that the uploaded cipher text is stored in an encrypted format. The download module enables an authorized user download an encrypted cipher text from the cloud repository system. Only users with access key can access this module to be able to download files. Access key is controlled and distributed by a system administrator. Decryption module is located at the user end. It is used to decrypt the downloaded cipher text form the user machine. A user must have a secret key to decrypt a cipher text.

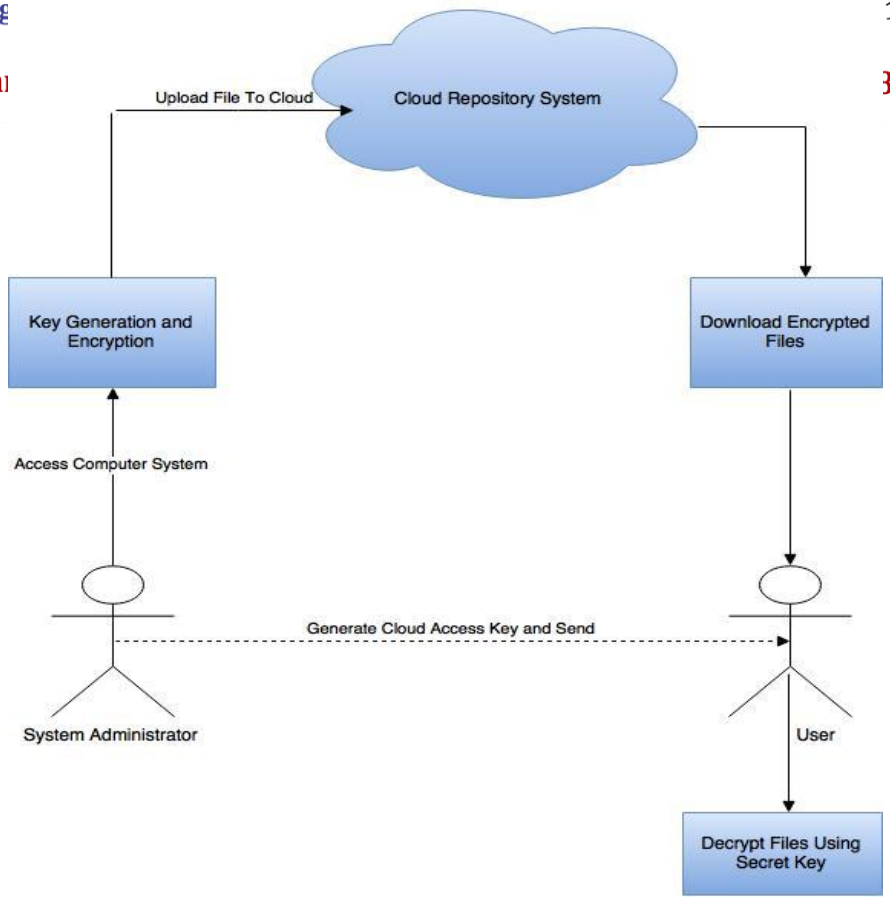


**Figure1: Proposed Model Architecture**

## III EVALUATION OF SYMMETRIC KEY ALGORITHMS

In this model, an analysis tool determines whether an algorithm is suitable for a particular class of data. It measures the performance of an encryption algorithm. The tool is capable of an analysising plaintext inputs. More importantly this tool can be extended to evaluate combined algorithms as well as new symmetric key algorithms. The other idea for incorporating this tool was to give a numerical output to depict the response times for a fixed text sample rather than simulating some known attacks. This reduces the complexity of the tool and makes it more users friendly. If someone introduces a new symmetric key algorithm the tool can be used to analyze its secrecy and performance

### 3.1 Functional Requirements

This refers to what the model can do in fulfilling the user objectives with an improved performance, affordable cost without compromising quality. The main functionalities of this model are listed below

i. User Authentication

ii. Upload encrypt data

iii. Providing confidentiality to the data stored in the cloud

iv. Restricting access control levels

v. Requesting access for encrypt data

vi. Reducing the cloud security as a service cost

vii. Maintaining logs of downloaded files

viii. Non-Functional Requirements

The model is designed with flexibility to enable it being responsive and adaptive to change of user requirements and environment. Scalability; the model provide for future expansion as level of security threats changes. It enables to change encryption algorithms based on threats and performance requirements. On usability, the model is easy to use. The security function of the model is improved in the sense that all encryption and decryption is undertaken at the user side. Key generation and storage is at the user side. Security control is under the data owner. In terms of securing users data stored in the cloud repository system, this model provides a reasonable data confidentiality protection.

### 3.2 CONCEPT

Suppose one want to send a message to the receiver and it wants the integrity protection in the transmission processes. So here we are proposing a novice approach for integrity protection by generating the hash code trough neural network which is quite simpler and more secure technique than existing ones.

- First of all we build the tree parity machine for each of the communicating party.
- We are generating the secure key through the process of mutual learning and neural synchronization instead of manually selected by user which offers great security due to neural synchronization which is much faster than learning. So man-in-the middle attack is not possible.
- After the key generation by tree parity machines this generated key of 128 bit which is further divided in four parts 32 bit each. These four key will be quantized and used to generate all the twelve sub keys which are composed of 224 data pixels for the uses of neural networks in hash code generation. The chaotic map is applied on subparts of keys to generate sub keys.
- Because we are generating the key by TPM, then every time each sub key is different for each layer every time which doubles the security offered.
- After the sub key generation the message is fed into the four layer feed forward neural network for hash code generation. The transfer function is used here is piecewise chaotic linear map which makes it impossible to predict the initial conditions exactly.
- Chaotic map often occurs in the study of dynamical system, small changes in initial conditions yield widely diverging outcomes So long term prediction is impossible. No one can use the statistical properties to infer something about the particular individual.
- After processing of neural network the intermediate hash value is generated for a message of different sizes and after that we have apply the gray code concept for offering great security and error correcting capability to the generated hash as the part of this thesis.

## IV PROPOSED HASH FUNCTION BASED ON KEY GENERATED BY TPM

### 4.1 Program of implement message length v/s time for md5

```
inp=['abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ']
for ii=1:52
inp1=inp(1,1:ii);
tic
digest=md52(inp1);
time(ii)=toc
end
ml=8*[1:52]
plot(ml,time,'-ro','markerfacecolor','g','markersize',4);
ylabel ('time in seconds');
xlabel ('message length in bits');
title('message length v/s time for md5 method')
```

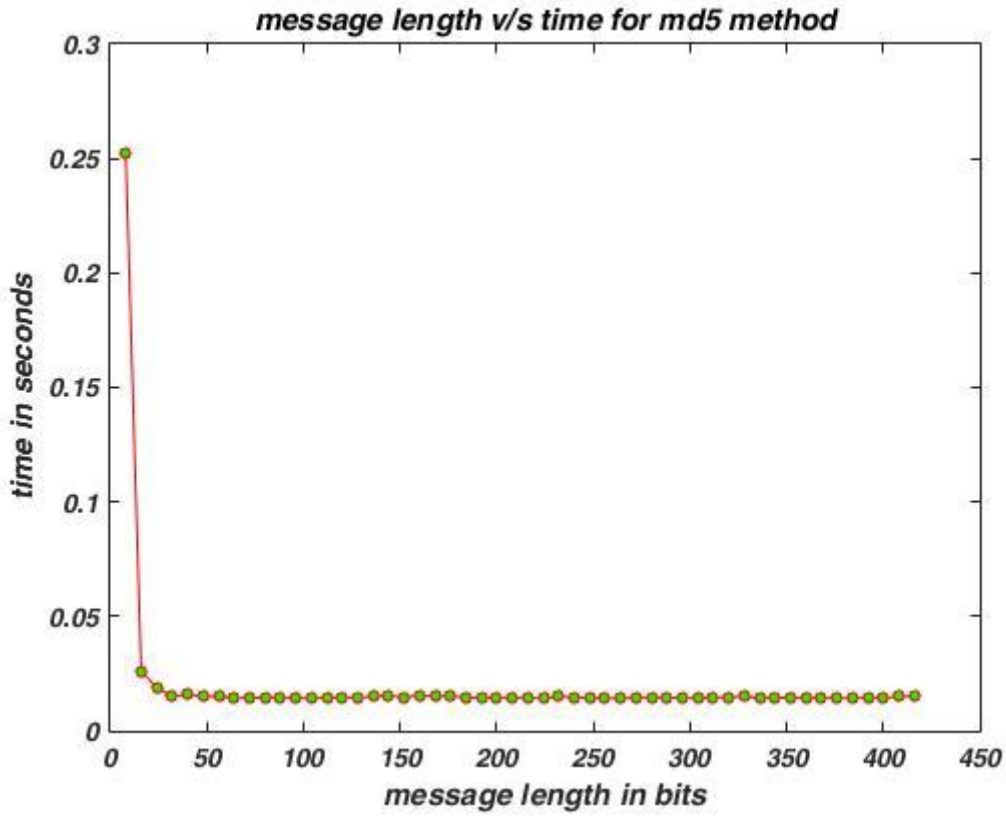


**Figure 2 Time comparision form Message length vs MD5 Algorithms**

**4.2 Program of plain text sensitivity v/s message length in bits'**

```
[fid]=fopen('hasc','r');
[hg,count]=fread(fid,'int8');
status = fclose(fid);
hg1=reshape(hg,128,52);
hg1=hg1';
hg3=zeros(52,128);
s=zeros(52,1);
for i=2:52
for j=1:128
hg3(i,j)=xor(hg1(i,j),hg1(i-1,j));
if hg3(i,j)==1
s(i)=s(i)+1
else
s(i)=s(i)
end
end
end
ml=8*[1:52]
y=s
plot(ml,y,'-ro','markerfacecolor','g','markersize',4);
xlim([16 8*52])
ylim([10 90])
ylabel ('plain text sensitivity');
xlabel ('message length in bits');
```

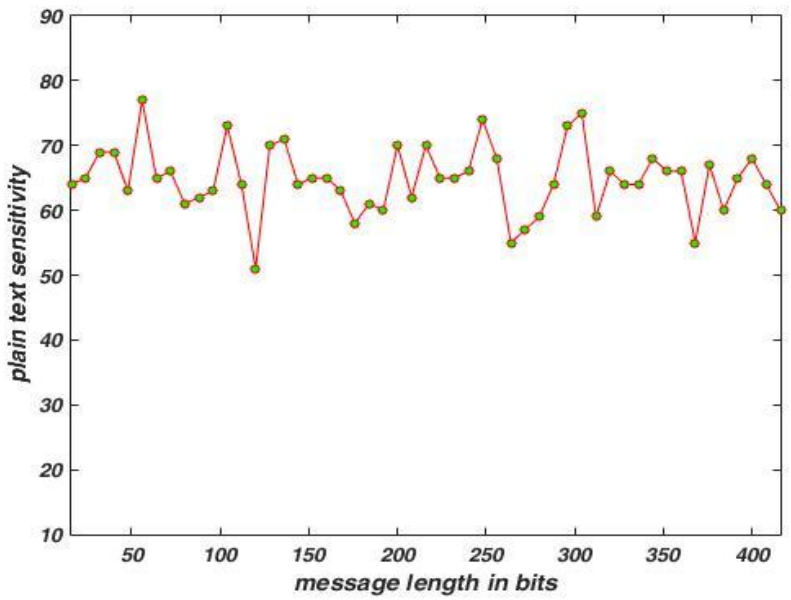


**Figure 3 Plain Text Sensitivity message length in bit**

## V CONCLUSION

Cloud Storage is a cost-effective Information Technology service to the general user or enterprise customer. Because of that many data owners are interested in outsourcing their sensitive data to the cloud storage. However, there exist data confidentiality flaws within cloud computing storage servers. Due to this, data owners lack the courage to strategically use the cloud computing storage as a service. Once the trust issues are addressed through the deployment of this model where data confidentiality protection starts at local user machine, there shall be some attitude change on the side of data owners towards the need to adopt cloud computing because the trust gap between user and service provider will be minimized. In this paper when end-to-end encryption of user data is maintained give the confidence to the person that data to be secure in transaction. To keep the cost low and maintain high sensitive data, it is recommended that users encrypt data at the user end before uploading to cloud data storage servers. This model enables secret keys to be retained by the user.

## VI FUTURE WORK

Easy to implement on small and middle level organization were reluctant to adopt cloud-computing platforms was fear that their data will land into the hands of unauthorized persons.

This papercan be thought as one that adopted software engineering approach by designing and deployment of a secure cloud storage product. Later it should come up with a convenient and secured way to distribute secret keys to users. Also it to be added how users can be able to work on the files in encrypted format in the cloud repository without the need to first download to user computers as this may reduce the overhead associated with Internet latency. Consideration should be given to develop new encryption algorithms with improved security levels and acceptable performance levels respectively as attackers keep on improving attacks such as dictionary attacks and brute force. It is also proposed that future work should focus on how we can have access key used for encryption, decryption and access for encrypted data in the cloud. This will reduce maintenance of secret and private keys.

## VII Reference

[1] Abdullah, A. M., & Aziz, R. H. H. (2016, June). New Approaches to Encrypt and Decrypt Data in Image using Cryptography and Steganography Algorithm., International Journal of Computer Applications, Vol. 143, No.4 (pp. 11-17).

[2] Singh, G. (2013). A study of encryption algorithms (RSA, DES, 3DES and AES) for information security. International Journal of Computer Applications, 67(19).

[3] Gaj, K., & Chodowiec, P. (2001, April). Fast implementation and fair comparison of the final candidates for Advanced Encryption Standard using Field Programmable Gate Arrays. In Cryptographers' Track at the RSA Conference (pp. 84-99). Springer Berlin Heidelberg.

[4] Stallings, W. (2006). Cryptography and network security: principles and practices. Pearson Education India.

[5] Yenuguvanilanka, J., & Elkeelany, O. (2008, April). Performance evaluation of hardware models of Advanced Encryption Standard (AES) algorithm. In Southeastcon, 2008. IEEE (pp. 222-225).

[6] Lu, C. C., & Tseng, S. Y. (2002). Integrated design of AES (Advanced Encryption Standard) encrypter and decrypter. In Application-Specific Systems, Architectures and Processors, 2002. Proceedings. The IEEE International Conference on (pp. 277-285).

[7] Mohamed, A. A., & Madian, A. H. (2010, December). A Modified Rijndael Algorithm and it's Implementation using FPGA. In Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on (pp. 335-338).

[8] Pramstaller, N., Gurkaynak, F. K., Haene, S., Kaeslin, H., Felber, N., & Fichtner, W. (2004, September). Towards an AES crypto-chip resistant to differential power analysis. In Solid-State Circuits Conference, 2004. ESSCIRC 2004. Proceeding of the 30th European IEEE (pp. 307-310).

[9] Deshpande, H. S., Karande, K. J., & Mulani, A. O. (2014, April). Efficient implementation of AES algorithm on FPGA. In Communications and Signal Processing (ICCSP), 2014 IEEE International Conference on (pp. 1895-1899).

[10] Nadeem, H (2006). A performance comparison of data encryption algorithms," IEEE Information and Communication Technologies, (pp. 84-89).

[11] Diaa, S., E, Hatem M. A. K., & Mohiy M. H. (2010, May) Evaluating the Performance of Symmetric Encryption Algorithms. International Journal of Network Security, Vol.10, No.3, (pp.213-219).

[12] Jain, R., Jejurkar, R., Chopade, S., Vaidya, S., & Sanap, M. (2014). AES Algorithm Using 512 Bit Key Implementation for Secure Communication. International journal of innovative Research in Computer and Communication Engineering, 2(3).

[13] Selmane, N., Guilley, S., & Danger, J. L. (2008, May). Practical setup time violation attacks on AES. In Dependable Computing Conference, 2008. EDCC 2008. Seventh European (pp. 91-96). IEEE.

[14] Berent, A. (2013). Advanced Encryption Standard by Example. Document available at URL http://www. networkdls. com/Articles/AESbyExample. pdf (April 1 2007) Accessed: June.

[15] Benvenuto, C. J. (2012). Galois field in cryptography. University of Washington.

[16] Lee, H., Lee, K., & Shin, Y. (2009). Aes implementation and performance evaluation on 8-bit microcontrollers. arXiv preprint arXiv:0911.0482.

[17] Padate, R., & Patel, A. (2014). Encryption and decryption of text using AES algorithm. International Journal of Emerging Technology and Advanced Engineering, 4(5), 54-9.

[18] Reddy, M. S., & Babu, Y. A. (2013). Evaluation of Microblaze and Implementation of AES Algorithm using Spartan-3E. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(7), 3341-3347.