Volume 5, Issue 10, October-2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Maintaining Data Security Integrity and Availability in Cloud Using Seed Block and Advanced Encryption Standard (AES)

[1]Mayuri H. Molawade, [2]Suhas H. Patil
[1] M.Tech student, Computer Engineering Department, BVCOE, Pune, Maharashtra, India
[2] Professor, BVUCOE, Pune, Maharashtra, India

Abstract— We are living in the information age. Data coming from sources is large like in petabytes or in terabytes, so we require large amount of data storage. Cloud is one example of online large data storage. But security is main issue in cloud. Yet for security professionals, the cloud presents huge dilemma: how do you embrace the benefits of the cloud while maintaining security controls over your organization assets? It become a question of balance to determine whether the increased risks are truly worth the agility and economic benefits. In this efficient security is given than previous work. More than one server is given for storing user's data. Before storing that information third party auditor divide data by using seed block algorithm, divided blocks encrypt using encryption algorithm.

Keywords— AES algorithm, seed block algorithm, encryption, data block, backup cloud.

I. INTRODUCTION

Cloud computing can not define in one word there are many statement maintain for cloud computing. But still we can define it as network of computers connected via internet and we use this network for file sharing as well as file storing and retrieving that data at any place at any time. For any cloud there some policies are maintain so that it can provide security, reliability, as well as availability, the most important issues in cloud is security how the provider give it to the user. As said encryption is process of converting normal text into some symbolic form, that is plain text into cipher text. But it is impossible to convert hug data and also it has very complex mathematical computation.

There are three important things related to cloud computing security are: Confidentiality, Integrity and Availability, known as the ACI.

Availability is the testimony that data will be available to the user whenever and wherever user want. It is possible because of fault tolerance, network security and authentication.

Integrity is the assurance that the data is not alter in between process and change made in data is only by authorized user. Integrity is infringed if the transmitted message is not same as received one. It is ensured by: Firewalls and intrusion detection.

Confidentiality is avoidance of unauthorized expose of user data. It is ensured by: security protocols, authentication services and data encryption services.

II. PROBLEM STATEMENT

As we know data coming from user is very large in amount. To improve the security to data store by user in cloud. To secure this amount of data we need powerful system. For efficient data security we need to do a works that provides secure data encryption as well as secure shield against data larceny. With many new technologies and services, information security and data protection issues are intensely debated, and examined far more critically than is the case with offerings that have been around for a while. Many surveys and studies open that efficacious customers have concerns about information security and data protection which stand in the way of a wider deployment. To implement data security, integrity as well as availability in cloud computing systems by using seed block algorithm for dividing data stream and RSA algorithm for encryption decryption.

III. ADVANCED ENCRYPTION STANDARD

AES Algorithm uses 128 bit block for encryption. Encryption consists of 10 rounds of processing for 128-bit keys. All 10 rounds same but only last 10^{th} round is different. AES algorithm also uses 128,192 and 256 bit blocks also. But all our discussion depend upon 128 bits.

As we are using AES algorithm have 10 round of encryption so that more possibilities of secure encryption.Each round of processing includes one single-byte based substitution step, a row-wise permutation step, a column-wise mixing step, and the addition of the round key. Depend upon encryption as well as decryption these steps differ.

To appreciate the processing steps used in a single round, it is best to think of a 128-bit block as consisting of a 4 × 4 matrix of bytes, arranged as follows:

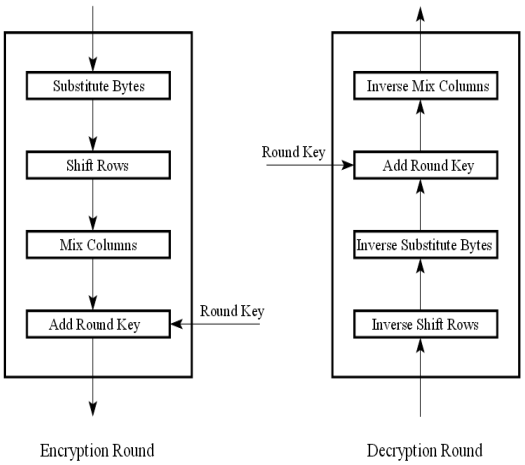

Fig. 1 AES encryption and decryption process

One round of encryption and one round of decryption on left and right hand respectively.

IV. SEED BLOCK ALGORITHM

Backup server is copy of main server. So that whenever the main server is down or not working will get information from backup server. Same as main server backup server is also remotely located. Main cloud is called as central repository.

It uses process of Exclusive-OR i.e. XOR operation. It has done like if two files are there a and b, its result will be d then d=a⊕ b . It uses for backup like if a file is lost then d will XOR with b and will get result as a file.

SEED BLOCK ALGORITHM

Initialization: Cloud: C; Remote Server: S;

Clients of Cloud: Cl; Files: f1 and f'1;

Seed block: Sb; Random Number: r; Client's ID: Cl_Idi

Input: f1 created by Cl; r generated at C;

Output: Recovered file f1 after deletion at C

Authenticated clients could allow performing operations such as downloading, uploading and doing modification on their own the files.

Step1: Generate a random number.

int r = rndno ();

step2: create a seed block Sb for each Cl and

store Sb at RSb =r⊕Cl_Idi (Repeat Step2 for all clients)

Step3: if Cl / Admin creates/modifies a f1 and stores at C, then f'1 create s as f'1= f1 ⊕Sb

Step4: store f'1 at S.

Step5: If server crashes or f1 deleted from C, Then, we do EXOR to retrieve the original f1 as:f1=f'1⊕Sb

Step 6: return f1 to Cl.

Step7: END.

V. PROPOSED SYSTEM

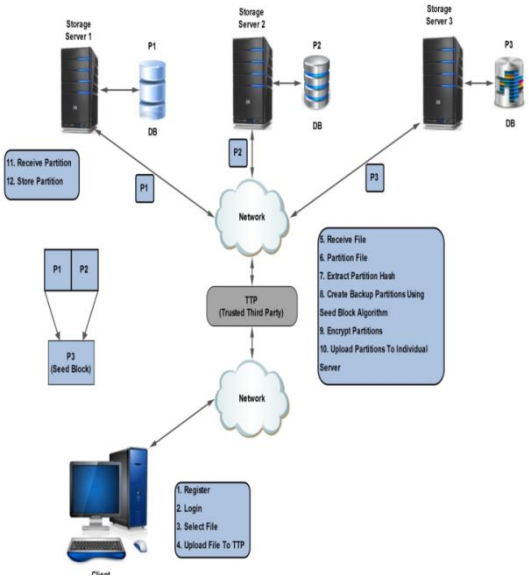

Fig. 2 proposed system block diagram

In first module we collect data from multiple user or client and then it given to the third party auditor (TPA).
In second module first we divide data into 3 blocks by using seed block algorithm. In cloud computing, data whichever send by user is in electronic form are in large amount. To maintain this data efficiently, there is a necessity of data recovery services. The objective of algorithm is divided into two parts, firstly it collect information from remote location user and secondly it recover that store file whichever store at server and incase of accidentally deleted from store location. Time relate issue is also solved by SBA such that whichever time taken for encryption as well as decryption is very less. Security is main concern in proposed system the back-up files stored at remote server, without using any of the existing encryption techniques.
Divided blocks are then encrypted by using asymmetric algorithm. The setting of public-key cryptography is also called the "asymmetric" setting due to the asymmetry in key information held by the parties. Namely one party has a secret key while another has the public key that matches this secret key.
After encryption of data, divided two blocks are store in two server that is server 1 and server 2. These two block of information are XOR with each other then result of XOR are store in third server so that by using third server I can easily retrieve information from first and second server.

VI. RESULT ANALYSIS OF PROPOSED SYSTEM

Table 1. Performance of System

	Precision	Recall
client 1	0.888888889	0.8
client 2	0.941176471	0.8
Total	0.91503268	0.8

Accuracy Percentage	0.8

Data Set Name	Actual	Total	Correct
client 1	20	18	16
client 2	20	17	16

Confusion Matrix			
		client 1	client 2
	client 1	16	1
	client 2	2	16

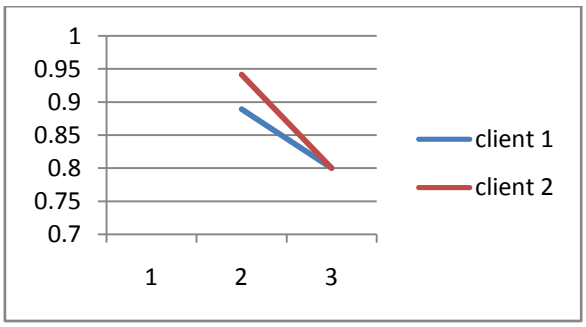

Chart 1. Analysis of Table Value

VII. CONCLUSION AND FUTURE SCOPE

In this paper, we have proposed that efficient encryption of data coming from client and also store backup file. Using backup file we can retrieve any data store in backup server. By using seed block algorithm firstly divide data in to two parts then XOR of that two file are store in backup server, and divided files are store in two server. In future we can implement these algorithm for more bits encryption.

REFERENCES

[1] "Cloud Net Directory. Retrieved 2010-03-01" (http://www.cloudbook.net/directories/research-clouds). Cloudbook.net. . Retrieved 2010-08-22.

[2] nsf.gov – National Science Foundation (NSF) News – National Science Foundation Awards Millions to Fourteen Universities for Cloud Computing Research – US National Science Foun... (http://www.nsf.gov/news/news_summ.jsp?cntn_id=114686).

[3] "IBM, Google Team on an Enterprise Cloud." May 2008. Rich Miller Retrieved 2010-04-01" (http://www.datacenterknowledge.com/archives/ 2008/05/02/ibm-google-team-on-an-enterprise-cloud/). DataCenterKnowledge.com. 2008-05-02. . Retrieved 2010-08-22.

[4] Open Networking Foundation News Release. March 21,2011. (http://www.openflow.org/wp/2011/03/open-networking-foundation-formed-to-speed-network-innovation/)

[5] Prakash, G. L., Prateek, M., & Singh, I. Data Encryption and Decryption Algorithms using Key Rotations for Data Security in Cloud System.

[6] Tirthani, N., & Ganesan, R. (2014). Data Security in Cloud Architecture Based on Diffie Hellman and Elliptical Curve Cryptography. IACR Cryptology ePrint Archive, 2014, 49.

[7] Khan, J. I., & Tahboub, O. Y. (2011, April). Peer-to-Peer Enterprise Data Backup over a Ren Cloud. In Information Technology: New Generations (ITNG), 2011 Eighth International Conference on (pp. 959-964). IEEE.

[8] Khanna, N., Nath, J., James, J., Chakraborty, S., Chakrabarti, A., & Nath, A. (2011, June). New Symmetric key Cryptographic algorithm using combined bit manipulation and MSA encryption algorithm: NJJSAA symmetric key algorithm. In Communication Systems and Network Technologies (CSNT), 2011 International Conference on (pp. 125-130). IEEE.

[9] Almutairi, A., Sarfraz, M., Basalamah, S., Aref, W., & Ghafoor, A. (2011). A distributed access control architecture for cloud computing.

[10] Wee, S., & Liu, H. (2010, March). Client-side load balancer using cloud. In Proceedings of the 2010 ACM Symposium on Applied Computing (pp. 399-405). ACM

