ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

Volume 3, Special Issue 3, March 2014

2014 International Conference on Innovations in Engineering and Technology (ICIET'14)
On 21st & 22nd March Organized by

K.L.N. College of Engineering and Technology, Madurai, Tamil Nadu, India

An Efficient Dynamic Data Violation Checking Technique For Data Integrity Assurance In Cloud Computing

P.Premkumar[1], Dr.D.Shanthi[2]

[1]Assstant Professor, Department Of CSE, K.L.N.College of Engineering, Madurai, India.

[2]Professor & Head , Department Of CSE, PSNA College of Engineering & Technology, Madurai, India.

Abstract-Cloud Computing is an Internet based computing which enable Users to share resources in the form of Software, Platform , Infrastructure services across multiple computers. Data Integrity is an important aspect while attain Accuracy, Security, Confidentiality, Reliability of data on cloud storage. It ensure that data retrieved is same as data stored or transmitted. This paper propose a new technique to improve the performance of data integrity checking on cloud storage through detection of data integrity violation and also protection of data with less cost ,Time, memory Space. This method based on determinant approach to detect violation of data dynamically. Each block of data is arranged in a matrix form so that to check the data violations very efficiently and then using Digital Signature method to protect the privacy of data for ensure data integrity in cloud environment. This research paper mainly points out how Third Party Auditors can be avoided due to bandwidth,overheads and cost while data validation. The paper is organized as follows, section1 describes Introduction, section2 describes Literature review, section3 describes Proposed method followed by Conclusion.

Keywords : Cloud Computing, Data Integrity, Cryptography, Digital signature, Encryption/ Decryption algorithms.

I) INTRODUCTION

Cloud Computing :

Cloud computing is a new computing paradigm in which dynamically scalable resources are shared as a service over the

internet.Now a days,it becoming more and more popular, where the data is outsourced through the cloud. Cloud storage services
enable user to enjoy high capacity and high quality storage with less overhead but it has
many potential threats ie.,data integrity,data availability,data privacy and so on. In Cloud Computing, maintain data integrity is one of the major challenge because the user has no control over the security mechanisms that are used to protect the data. These services are broadly divided into three categories:

a) SaaS (Software as a Service)

SaaS is a model of software deployment where consumer use the provider's application running on a cloud infrastructure through client interface.The best SaaS applications are IBM Lotus Live,Google Gmail,Online customer relationship management services from salesforce.com

b) PaaS (Platform as a Service)

The platforms used to develop, build and test applications are provided to the consumer by the cloud.Popular Paas platforms include Google App.Engine,Microsoft windows Azure also supports Paas and SaaS applications.

c) IaaS (Infrastructure as a Service)

This is a model in which service provider owns the equipments used to support operations, including storage,

hardware, servers and networking components. Amazon's Elastic Compute cloud is a good example of IaaS.There are two issues are mainly occurred in Cloud computing. The first issue is data integrity. The second issue is unfaithful cloud server providers (CSP). As data owner outsourced their data to the cloud and do not maintain the local copy so cryptographic measures cannot be used directly to monitor the integrity of data. Also downloading the data for monitoring integrity is not a viable solution as it incurs high cost of input/output and transmission across the network. Therefore, an external third party auditor (TPA) is required. The TPA is an independent authority that has expertise and capabilities to monitor the integrity of cloud data outsourced by the client and inform about data corruption or loss, if any.

Data Integrity:

Data Integrity is the form of protection of data against loss and damage caused by hardware and software failure. It relates to quality of data.Inaccuracy of data can occur either accidently through programming errors or maliciously through breaches or hacks.Data integrity is one of the important aspect among the other cloud storage issues because data integrity ensured that data is of high quality, correct, consistency,accuracy and accessible but maintenance of data integrity in the cloud is a major challenge that is faced by today's cloud users. Data integrity refers to the assurance by the user that the data is not modified or corrupted by the service provider or other users. The data will be stored in the cloud by a user and the integrity of the data will be checked by Auditor. The efficiency of data integrity is measured using the parameters like time for processing the data, storage cost, memory for storage. Database security professionals employ number of practices to assure data integrity which includes

- Data encryption, which locks data by cipher
- Data backup, which stores a copy of data in an alternate location
- Access controls, including assignment of read/write privileges(security mechanism)
- Input validation, to prevent incorrect data entry through interface designing
- Data validation, to certify uncorrupted transmission
- Using Error detection and correction software when transmitting data.

The scope of the Data Integrity assurance mechanism can be classified into two levels: To prevent data corruption, To detect and correct data violation.

Data Integrity Checking Methods:

Mirroring technique (MT)

Mirroring or data replication is a common integrity assurance technique used to check if there is any data violation in the storage devices. The MT maintains two or more copies of the original data. Integrity check is made by comparing these copies; any differences will indicate a possible corrupted data.There are many weaknesses with this technique. If the original and mirrored data have the same modifications, the MT cannot detect the data corruption. If a malicious person inserts the same value into the original and mirrored data, the MT cannot detect the integrity violation and the inserted data is considered as part of the original data. MT cannot also detect integrity violation by user errors.

Checksum technique (CST)

A checksum is an error detection mechanism that is created by ''summing up'' all the bytes or words in a data word to create a checksum value, which is appended to the stored or transmitted data word.The checksum of the retrieved data word is recomputed and compared with the received checksum value. If the computed and received checksum are identical, it is unlikely that the data word suffered an error during transmission or storing. If there is a corruption in retrieved data, the checksum technique does not provide information about which components in the data, series are corrupted. There are many types of checksum approaches. The simplest checksums involve a simple ''sum'' function across all bytes or words in a data word. However, there are some types of error which cannot be detected by this method. The first one when the data is reordered. Second type is when zero values are inserted in the data or deleted from it. The third type of errors occurs if multiples errors amount to zero. Three other commonly used simple ''sum'' functions are XOR two's complement addition and one's complement addition. These checksums provide fairly weak error detection coverage, but have very low computational cost.

Hamming code method (HC)

Hamming code method can be used as an error detection mechanism. It depends on parity bit. A code word is generated by adding the original data bits and the check bits to form Hamming Code word. The check bits are helps in detecting data integrity violation.It is obtained by XORing the weights of bits that having integrity violation in the original data. The new check bits of the retrieved code word are recomputed and XORed with the received check bits. If the result is zero, the data has no corruption during the transmission or storing.HC method has many deficiencies. If the original data bits are reordered or if the check bits are modified in the way to give zero XORing with the new check bits at received stage, this method cannot detect the data corruption. The hamming code is determined in terms of binary word. For 15-bit representation,4-bits of them are reserved for the check bits. This will increase the frame length of data and increase the data size and transmission cost.

Hash function method (HF)

A hash function is a transformation that takes an input or message and returns a fixed-size string,which is called the hash value sometimes called a message digest, a digital fingerprint, a digest or a checksum.The hash functions

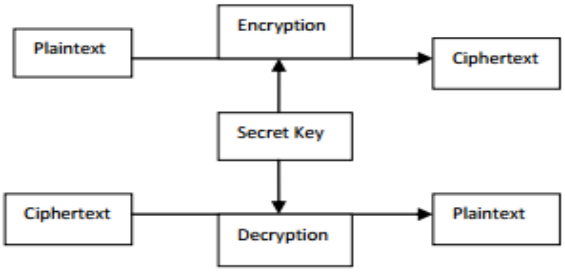
such as encryption file systems provide some degree of integrity assurance but they do not have the capability to detect the data violation. The malicious user can easily modify the hash codes of data result in data violation in the original data. For a large number of data, it is impossible to obtain a unique hash code. Hashing is a form of cryptographic security which differs from encryption,Whereas encryption is a two step process used to first encrypt and then decrypt a message, hashing condenses a message into an irreversible fixed-length value or *hash*. Two of the most common hashing algorithms are seen in networking are MD5
and SHA. Hashing is used only to verify the data; the original message cannot be retrieved from a hash. When used to authenticate secure communications, a hash is typically the result of the original message plus a secret key. Hashing algorithms are also commonly used without a secret key simply for error checking.

Message Authentication Code Method (MAC) :

Assume the outsourced data file F consists of a finite ordered set of blocks m1; m2; . . .; mn. One straightforward way to ensure the data integrity is to precompute MACs for the entire data file. Specifically before data outsourcing, the data owner precomputes MACs of F with a set of secret keys and stores them locally. During the auditing process, the data owner each time reveals a secret key to the cloud server and asks for a fresh keyed MAC for verification. This approach provides the facility for verification of all the data blocks. However, the number of verifications allowed to be performed in this solution is limited by the number of secret keys. Once the keys are exhausted, the data owner has to retrieve the entire file of F from the server in order to compute new MACs which is impractical due to the huge communication overhead.

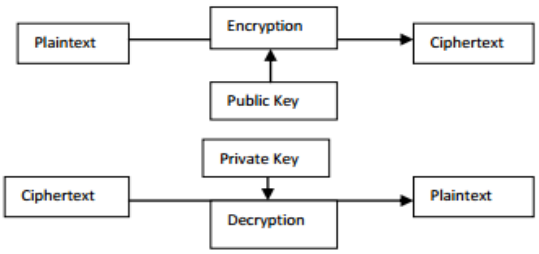
Cryptography:

Cryptography is referred to as " Study of Secrect". It is the mechanism which helps in achieving the following four basic goals. confidentiality, integrity, authentication and non-repudiation. Every algorithm ensures that these four goals are met while transmitting any digital message.The term encryption involves conversion of the plaintext into the cipher text. The plaintext is the readable data and the cipher text is a data which is impossible to read without the key. The key acts as the medium to convert the plain text into cipher text and vice versa. The process of conversion of cipher text into the plain text is called decryption. The key for encryption and decryption is meant to be secret and the strength of key decides the privacy of cipher text.There are two main categories of cryptography depending on the type of secret keys used to encrypt/decrypt the data.There are number of cryptographic algorithms used for encrypt the data and most of all fall into two generic categories –Symmetric key or Private Key, single-key, secret-key, one-key cryptography and Asymmetric or Public Key cryptographic. In symmetric key encryption, both sender and receiver share the same key which is used to both encrypt and decrypt messages. Sender and receiver only have to specify the shared key in the beginning and then they can begin to encrypt and decrypt messages between them using that key.The strength of the key encryption depends on size of key used.The examples are AES,DES, 3DES,RC4,RC5,Blowfish.



Symmetric Key Cryptography Process

In Asymmetric Encryption method use two keys: a public key and a private key. The public key is made publicly available and is used to encrypt messages by anyone who wishes to send a message to the person that the key belongs to. The private key is kept secret and is used to decrypt received messages. The private and public key are mathematically linked. Public key encryption is slow compared to symmetric encryption. Not feasible for use in decrypting bulk messages.Asymmetric encryption techniques are almost 1000 times slower than Symmetric techniques,because they require more computed processing power. The examples are RSA, Diffe-Hellman, ECC, ELGAMAL.



Asymmetric Key Cryptography Process

In this paper, we also provide surveys on various approaches and algorithms used to ensure data integrity in cloud environment at dynamically.

Symmetric Algorithms:

DES(56 bits key size , 64 bits block size) **:**

DES is a block cipher. It encrypts data in blocks of size 64 bits each. 64 bits of plain text goes as the input to DES which produces 64 bits of cipher text.The key length is 64 bits

3DES(64bit blocksize,192bit key size) **:**

3DES is an enhancement of DES; In this standard the encryption method is similar to the one in the original DES but applied 3 times to increase the encryption level

and the average safe time. It is a known fact that 3DES is slower than other block cipher methods .

RC2 (64 bits block cipher with 8 to128 bits key size)**:** RC2 is a block cipher with a variable key size .

RC6 (128 bits block size with 128, 192 and 256 bits Key size)**:**RC6 is block cipher derived from RC5. It was designed to meet the requirements of the AESStandard competition.

AES (128 bits block size with 128,192 or 256 bits key size)**:** AES is a block cipher with variable key length; default 256. It is based on a design principle known as substitution-permutation network. The AES cipher is specified as a number of repetitions of transformation rounds that convert the input plaintext into the final output of cipher text.

It encrypts data blocks of 128 bits in 10, 12 and 14 round depending on the key size. AES encryption is fast and flexible; it can be implemented on various platforms especially in small devices.

BLOWFISH (64-bits symmetric block cipher with 32 bits to 448 bits key size) **:** The algorithm operates with two parts: key expansion part and data- encryption part. The role of key expansion part is to converts a key of at most 448 bits into several sub key arrays totaling 4168 bytes,default 128 bits. The data encryption occurs via a 16-round Feistel network. It is only suitable for application where the key does not change often like communications link or an automatic file encryption. It is significantly faster than most encryption algorithms when implemented on 32-bit microprocessors with large data caches. It can be used as a replacement for the DES algorithm. It takes a variable length key; default 128 bits. Blowfish has variants of 14 rounds or less. It is successor to Twfish.

Algorithm	Block Size (Bits)	Key Size (Bits)	Speed	Security
DES	64	56	Low	Less
3DES	128	112, 168	Low	Less
RC2	64	8-128	Fast	High
RC6	128	128,192	Fast	Secure
AES	128	128,192, 256	Fast	More secure
Blowfish	64	32-448	Fast	More Secure

Asymmetric Encryption Algorithms :

The following are the major asymmetric encryption algorithms used for encrypting or digitally signing data.

Diffie-Hellman key agreement: Diffie-Hellman algorithm is not for encryption or decryption but it enable two parties who are involved in communication to generate a shared secret key for exchanging information confidentially. It is used to generate the symmetric key that is known by both the user and the auditor.

Rivest Shamir Adleman (RSA): RSA is a block cipher mechanism in which the block size is 2k. bits ,where $2^k < n <= 2^{k+1}$. This public key algorithm can be used for encrypting and signing data. The encryption and signing processes are performed through a series of modular multiplications.

EllipticCurve Cryptography (ECC): Elliptic Curve Cryptography (ECC) provides similar functionality to RSA. Elliptic Curve Cryptography (ECC) is being implemented in smaller devices like cell phones. It requires less computing power compared with RSA. ECC encryption systems are based on the idea of using points on public/private key pair.

El Gamal: El Gamal is an algorithm used for transmitting digital signatures and key exchanges.The method is based on calculating logarithms. El Gamal algorithm is based on the characteristics of logarithmic numbers and calculations.The Digital Signature Algorithm (DSA) is based on El Gamal algorithm.

Digital Signature Algorithm (DSA): Digital Signature Algorithm can be used only for signing data and it cannot be used for encryption.The DSA signing process is performed through a series of calculations based on a selected prime number. Although intended to have a maximum key size of 1,024 bits, longer key sizes are now supported. When DSA is used, the process of creating the digital signature is faster than validating it. When RSA is used, the process of validating the digital signature is faster than creating it. A digital signature is an electronic signature that can be used to authenticate the identity of the sender of a message or the signer of a document and possibly to ensure that the original content of the message or document that has been sent is unchanged. Digital signatures are easily transportable, cannot be limitated by someone else, and can be automatically time-stamped. The ability to ensure that the original signed message arrived means that the sender cannot easily repudiate it later. A digital signature can be used with any kind of message, whether it is encrypted or not, so that the receiver can be sure of the sender's identity and that the message arrived intact. A digital certificate contains the digital signature of the certificate-issuing authority so that anyone can verify that the certificate is real. The data owner precomputes the signature of each block and sends both F and the signatures to the cloud server for storage. To verify the correctness of F,the data owner can adopt a spot-checking approach.Notice that the above methods can only support the static data and also a large communication overhead that greatly affects system efficiency.The following Table shows comparison of different methods used for ensure data integrity.

Methods	DES	RSA
Approach	Symmetric	Asymmetric
Encryption	Faster	Slow
Decryption	Faster	Slow
Key Distribution	Difficult	Easy
Security	Moderate	Highest
Secure Services	Confidentially	Confidentially, Integrity, Non-Repudiation

II. LITERATURE REVIEW :

This section discusses the performance of the various methods and algorithms used to ensure data integrity.

In this paper [1] certain degree of Integrity assurance can be provided by RAID technique but it can opertate only on binary data and takes more computation time with high cost.

In this paper [7] consider the performance of encryption algorithm for text files, it uses AES, DES and RSA algorithm and is evaluated through the parameters like Computation time, Memory usage, Output bytes. First, the encryption time is computed. The time is taken to convert plain text to cipher text is known as encryption time. Comparing these three algorithms, RSA takes more time for computation process. The memory usage of each algorithm is considered as memory byte level. RSA takes larger memory than AES and DES. Finally, the output byte is calculated by the size of output byte of each algorithm. The level of output byte is equal for AES and DES, but RSA algorithm produces low level of output byte.

In this paper [8] the selected algorithms are AES, 3DES, Blowfish and DES. By using these algorithms the performance of encryption and decryption process of text files is calculated through the throughput parameter. Encryption time is calculated as the total plaintext in bytes encrypted divided by the encryption time. Decryption time is calculated as the total plaintext in bytes decrypted divided by the decryption time. As a result mentioned in the paper , it is said that Blowfish algorithm gives the better performance than all other algorithms in terms of throughput. The least efficient algorithm is 3DES.In this paper [10] discuss the performance evaluation of AES and BLOWFISH algorithms and the parameters are Time consumption of packet size for 64 bit encodings and hexadecimal encodings, encryption performance of text files and images are compared with these two algorithms and calculate the throughput level,

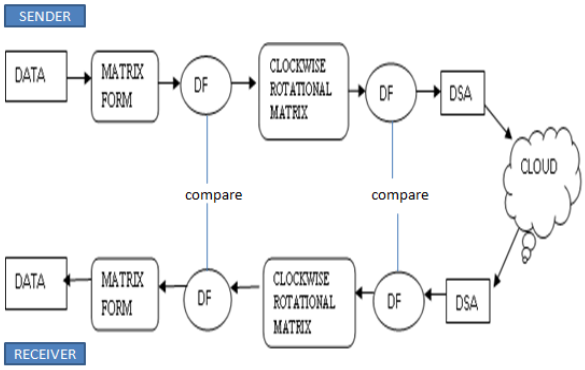
The results shows that Blowfish has better performance than AES in almost all the test cases includes changing the packet size. Based on these work Blowfish is more suitable when focus on encryption method.

III. PROPOSED MODEL:

This proposed technique is based on the determinant factor in measuring the data integrity assurance which involves the following steps: Before transmitting the series of data, it is divided into N-matrices, where N is given by:

$$N = \frac{total\ number\ of\ data}{d \times d}$$

where (d×d) is the number of elements per a matrix. The determinant of each matrix is computed directly from the data matrix and appended with the data. At retrieving stage, it is compared with the determinant of the retrieved data to measure the integrity assurance. But it is observed that there are many defects with this method. The determinant is zero if any row is proportional to another row; the same is true for columns. Also the determinant does not change if some rows or some columns are interchanged. In addition, the determinant is zero if any single row or column has zero values only. Therefore, in this paper an improvement is added to eliminate these avove defects. For each matrix of data, a clockwise-rotated matrix is generated. Then a clockwise rotation for the two halves of the matrix is made to formalize the original data matrix into a new form. The determinants of both; original and clockwise-rotated matrices are computed and appended with each matrix and then add digital signature with encryption for each Determinant Factor before transmission or storing the data. At the retrieving or receiving stage, both determinants are recomputed again in a similar way and compared with the attached values. If there is no difference, the data of that particular matrix has no errors during the transmission or storing. This method overcomes all the deficiencies accompanied with other data integrity checking methods through data can be checked in blockwise as well as send the data with privacy from unauthorized users. .



IV.CONCLUSION:

In this paper discussed theoretical performance analysis of selected symmetric and asymmetric Encryption algorithm. There are two major reasons which made

Public key cryptography algorithms more reliable in the areas of confidentiality key distribution and authentic. These algorithms are based on mathematical calculations rather than substitution and permutations like the symmetric cryptosystem. These algorithms use two keys in contrast to symmetric algorithms which uses only one key. Severa1points are to be concluded. First, despite the key distribution, DES is more suitable to the application which has the decryption as the highest priority. There is no doubt that, an asymmetric key cryptographic system provides high security in all ways. Second, AES algorithm is executed lesser processing time and more throughput level as compared to other algorithms. In this paper we proposed a model for the integrity checking over the cloud computing and utilize digital signature to achieve the data integrity in such a way that helps the user to verify and examine the data from unauthorized people that manipulate with the cloud.This paper presents a new approach for improving the detection of data integrity violations during data storing or transmission. In the proposed technique, the data are divided into blocks; where each block is arranged in square matrix. The new procedure overcomes all the defects that go along with data integrity assurance methods.The performance measures like better encryption time and a quicker detection of compromise of the clients' files in cloud environment is possible with the proposed system.In this method accuracy is to be maintain at satisfied level through checking factor is determined after rearranging the data two times via original matrix and its corresponding clockwise rotational matrix.But it requires more computation time compared to other checking methods.This deficiency is not a major issue due to good level of accuracy of data.

REFERENCES:

[1]. J.A.Ghaeb,M.A.Smadi,J.Chebil , "A High Performance Data Integrity assurance based on the Determinant technique" , ELSEVIER, April 2010

[2]. Shashi Mehrotra Seth, Rajan Mishra, "Comparitive Analysis of Encryption Algorithms For Data Communication", IJCST Vol.2, Issue 2, June 2011

[3]. "Performance Analysis of AES and BLOWFISH Algorithms", National Conference on Computer Communication & Informatics", School of computer science, RVS college of arts and science, March 07, 2012.

[4]. "BLOWFISH algorithm http://pocketbrief.net/related/BlowfishEncryption.pdf

[5]. Atul Kahate, "cryptography and network security", Tata McGraw-Hill publishing company, New Delhi, 2008.

[6]. B. Schneier," Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish) fast Software Encryption", Cambridge Security Workshop Proceedings (December 1993), Springer-Verlag, 1994, pp. 191-204.

[7].Tingyuan Nie Teng Zhang, "A study of DES and Blowfish encryption algorithm", Tencon IEEE Conference, 2009.

[8].William Stallings, "cryptography and network security", pearson prentice hall, 2006, 4th edition.

[9].Gurjeevan Singh, Ashwani Kumar Singla, K.S.Sandha, "Through Put Analysis of Various Encryption Algorithms", IJCST Vol.2, Issue3, September 2011.

[10]."Blowfish Algorithm" Available : http://www.schneier.com/blowfish. html

[11]."DES algorithm" http://orlingrabbe.com/des.htm

[12] Coppersmith D, The Data Encryption Standard (DES) and Its Strength Against Attacks, I BM Journal of Research and Development, May 1994, pp. 243 - 250.

[13] Bruce Schneier, The Blowfish Encryption Algorithm, Retrieved October 25, 2008

[14] W. Diffie, M. E. Hellman, New Directions in Cryptography, IEEE Trans. On Inform. Theory, pp. 644-654, 1976.