www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 4 Issue 11 Nov 2015, Page No. 15049-15054

Secured Data in Cloud by Using LFSR

Sumalatha Potteti[1], Ravi Kumar Goura[2]
[1]Assistant Professor, Department of CSE,BRECW,Hyderabad,India,sumalatha.po@gmail.com
[2]Assistant Professor, Department of CSE,MEDHA,Hyderabad,India,ravisslm@gmail.com

Abstract:

As an emerging technology and business paradigm, Cloud computing platforms provide easy access to a company's high-performance, computing and storage infrastructure through web services. Cloud computing gives numerous profits to the clients, for example approachability and accessibility. As the information is accessible over the cloud, it might be gained entrance to by diverse clients. There may be delicate information of association. This is the one issue to give access to validated clients only. But the data can be accessed by the owner of the cloud. So to abstain from getting information being gained entrance to by the cloud manager, we will utilize the different encryption algorithms to encrypt the data available over the cloud which in turn will give security to the information and only authenticated users will get access to the secure data over the cloud. Duplication of Data is a technique for reducing the amount of storage space, an organization needs to save its data. The proposed system supports authorized duplicate check in hybrid cloud architecture. Proposed authorized duplicate check scheme acquires minimal overhead compared to normal operations. To reduce the weaknesses of convergent encryption, we are proposing LFSR (Linear Feedback Shift Register) encryption technique. Security analysis determines that this system is secure in terms of the definitions specified in the proposed security model.

Key words: Cloud Computing, Cloud storage, Encryption, Duplication, Convergent Encryption, Proof of ownership, Authorized Duplicate Check, Differential Authorization.

1. INTRODUCTION

The enterprise data which transforms, store extremely large volumes information processing in a local area network are to a high degree expensive. For to continue volume's information required data storage devices of the high capacity such as memory net of fastening memory (NAS), storage area nets, and memory servers supply high speed, high availability data storage, which is accessible over interface standard. Beyond that data storage devices have many weaknesses, including are they very expensively to be brought to have a limited lifetime its require your support and recovery systems, and storage system required ecological conditions, requires personnel, to handle and use considerable quantities of energy for energy and cooling systems. Bewölken your data storage offers, like Google, Microsoft, Amazon, IBM, make yourself available very cheaply, practically unlimited data storage in the remote facilities. The data, which are stored with these offers, are accessible over the Internet. Costs on the scale make possible for offers to supply the data

storage which is to a high degree cheaper as the equivalent data storage systems. Cloud data storage has much reconciliation. It is, requires no installation of the system (server), does not need a maintenance and not replacing, has supported and recovery systems, has no committed personnel, requires no ecological conditions, requires no personnel and does not require energy not for energy or the cooling cheaply. Cloud data storage has however some important disadvantages, including security questions, achievement, availability, integrity, incompatible interfaces and lack of standards. In this paper address, this paper the problem of the safe and reliable data outsourcing in a cloud environment investigates these difficult questions.

Cloud computing provides infinite virtualized resources to users as services across the entire Internet, while hiding platform and implementation details. CSP (cloud service providers) deals at relatively low costs with both highly available storage and especially parallel computing resources. One major challenge of cloud storage services is the management of the ever-growing volume of data as cloud computing becomes universal; an amount of data is being stored and common by users with specified privilege, which define the access rights of the stored data. The rapid implementation of Cloud services is convoyed by increasing volumes of data stored at remote servers hence techniques for saving disk space and network bandwidth are needed. A central up and coming concept is reduplication where the server stores a single copy of each file, in spite of how many clients asked to store that file. All clients that store the file simply use links to the single copy of the file stored at the server. Furthermore, if the server already has a copy of the file then clients do not even need to upload it again to the server, thus saving bandwidth as well as storage. In a usual storage system with reduplication, a client first sends to the

server only a hash of the file and the server checks if that hash value already exists in its database. If the hash is not in the database then the server scans for the entire file. Otherwise, since the file already exists at the server, it tells the client that there is no need to send the file itself. Either way the server marks the client as an owner of that file, or from that point on there is no difference between the client and the original party who has uploaded the file. The client can therefore ask to restore the file, regardless of whether he was asked to upload the file or not. Data reduplication has certain benefits to Eliminating redundant data can extensively shrink storage requirements and increase bandwidth efficiency. Since primary storage has gotten cheaper over time, typically store many versions of the same information so that new workers can reuse earlier work done. Some operations like backup store extremely redundant information. Data reduplication is data compression technique for eliminating duplicate copies of repeating data in storage. This technique is used to improve storage utilization and can also be applied to network data transfers to decrease the number of bytes that must be sent. Reduplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy instead of keeping multiple data copies with the same content. Reduplication can take place at the file level and the block level. It eliminates duplicate copies of the same file at file level and eliminates duplicate blocks of data that occur in non-identical files at the block level.

2. THE CLOUD DATA STORAGE SYSTEM MODEL

The figure 1 shows a systematic model looking at the cloud data storage service which makes available for share data separating services as well as efficient data recovery and repair service including four different entities: Data owner, data user, cloud server, and third server. The data owner springs the encrypted fragments of the file m to N as a storage server to indicate cloud servers.

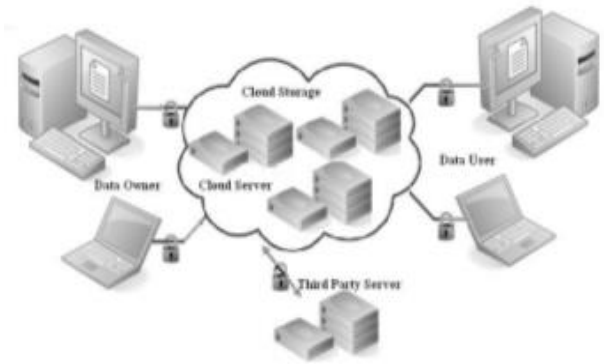

Fig 1: Cloud data storage system Model

If the data owner the holds of the data contents confidentially which require file can be M first encrypted before the encoding. Separated data are added by metadata like verification tags to make available integrity control-ability. After the data separating a data user some k storage server can select to retrieve coded segments, and to regain the file m which can be further deciphered, in case of that the file encrypted is. In the meantime, the third server checks regularly the integrity of data supplied in cloud servers. Fruitless cloud servers can be repaired with the help of other healthy cloud servers. In this available model many threats have, the cloud server is looked as "curious and vulnerable". Specifically the cloud server is vulnerable to Byzantine failures and outside attacks. While Byzantine failures can be done by hardware mistake or the clouds maintenance staff, outside attacks could be in the interval of physical disasters, how fire and earthquake to the willful chopping of opponents. After the opponent wins the control of the cloud data server, it can seize the soiling offensive or the replay-attack which has to the purpose to break the tongue independence under coded data, the data supplies on the spoil cloud server with old coded data substituting. If the cloud server is not spoiled, it follows properly the called protocol specification, but it will try, to derive and to analyze data in his storage and interaction during the protocol execution to learn additional information. This represents a threat against the security of cloud user data supplied in the server. Our suggested new model of the system conquered by a lot of screenplay like to make available sure and reliable clouds data storage services should reach our design at the same time achievement guarantees during the data recovery and repair.

3. LUBY TRANSFORMS COMPUTING SYSTEMS

The figure 2 shows that Luby transform system. Luby change codes the codes are correct classical by fountain codes which are close-optimum Ausradierung. Luby figures encoding around in particular come round this problem, a disposable protocol basically accepting. The sender encodes and sends a packet after the packet of the information. The receiver values every packet as it will receive. If there is a mistake, the wrong packet is

Rejected but the packet is saved as a piece of the news. In the end, the receiver has enough valid packets to rebuild the complete news. When the complete news has been received successfully, the receiver gives signs that the transference is concluded. The figure 2 shows, that to Luby at the cloud storage surroundings system being based Coding system reshaping. In this system is by many scales reliably as the reply-funded system. Data users can regain complete M of original packets, while they retrieve the same number of

DOI: 10.18535/Ijecs/v4i11.33

code packets of every K combination from n to servers, and, therefore, every server must supply only the coded packets from m / k which are looking at the property of the optimum trade of the profusion dependability of it. However, his square deciphering complexity does it very ineffectively for data user attains data during the data recovery again.

Fig 2:Luby Transform Coding System – Encoding and decoding

In addition, are the news costs to repair a fruitless storage server, the size of the complete original data in the optimized code-founded cloud data distributed storage system immediately? The encoding process begins the uncoding message into N blocks roughly of the same length sharing. Then coded packets

are generated with the help of a pseudo-accidental number generator. The degree d, $1 \leq d \leq n$, the following packet is chosen in the random. Exactly d blocks of the news are chosen by chance. If Wednesday is the block of the news, the data part of the following packet is estimated at equation 1

$$M_{i_1} \oplus M_{i_2} \oplus \cdots \oplus M_{i_d} \text{ --------1}$$

Where {i1, i2, ..., id} they by chance elective indications are for the D blocks enclosed in this packet. A prefix will have in the encoded packet tag on, defining how many blocks and are in the news how many blocks been real impossibly - in the data part of this packet, and the list of indications {i1, i2, ..., id}. In the end, a form is applied by mistake recognition code on the packet, and the packet is sent. This process continues, until of the receiver signal gives that receive the message and successfully decoded has become. The deciphering process uses " exclusively or " operation to retrieve the coded message. If the present packet is not clean, or if it repeats a packet which has already been worked on, the present packet is rejected. If the stream received cleanly packet, is from the degree d> 1, it is worked on first against all completely decoded blocks in the news which stands area of Queue, Stored in a buffer area when his diminished degree is bigger than 1. If a new, clean packet of the degree d = 1 (block Mi), will receive it is moved to the news queuing area, and then is compared against all packets of the degree d> 1 living in the buffer. This is exclusive - order in the data part of every buffered packet which was encoded, using Wednesday, there is the degree of this fitting packet decremented, and the list of indications for this packet is adjusted to reflect the application of Mi If this process unlocks a block of the degree d = 2 in the buffer, this block is reduced to the degree 1 and moves on his part to the message queuing area, and then worked on against the packet which remain in the buffer. When all N blocks of the message have been moved to the news queuing area, the receiver gives signal to the transmitter that the message has become successfully decoded.

4. FLOW CHART

The proposed methodology will comprise of different clouds which will be available to use by the user. But before using any service of the cloud computing the user will need to register. While registering the user will need to choose the encryption algorithm then the user will enter the security key to be used for encrypting the data of this user only. After the registration process will be completed the user will get to know the different keys which can be used to encrypt/decrypt the data which the user will store over the cloud.

There will be multiple clouds available where the client's information will be available in the encrypted form. The same information will be available in the number of different clouds, so this will help the authenticated clients to get access of the information effectively and permits the cloud to get the cloud for load balancing while information being retrieved by the number of different authenticated customers because the decryption technique is known to the authenticated users only. Assuming that the other client tries to gain entrance to the information from the cloud, the client will have to pass from the different security focuses where the validation of the client is carried out. Provided that the client appears to be valid client then only the client will get access to the data.

The validation of the client will be done by using the intrusion detection layers. When the client wants the access of the data, it will need to log into the account. With this all,

the security of the data is maintained with availability of the integrated data within the durable time to access/ retrieve the data by the user.

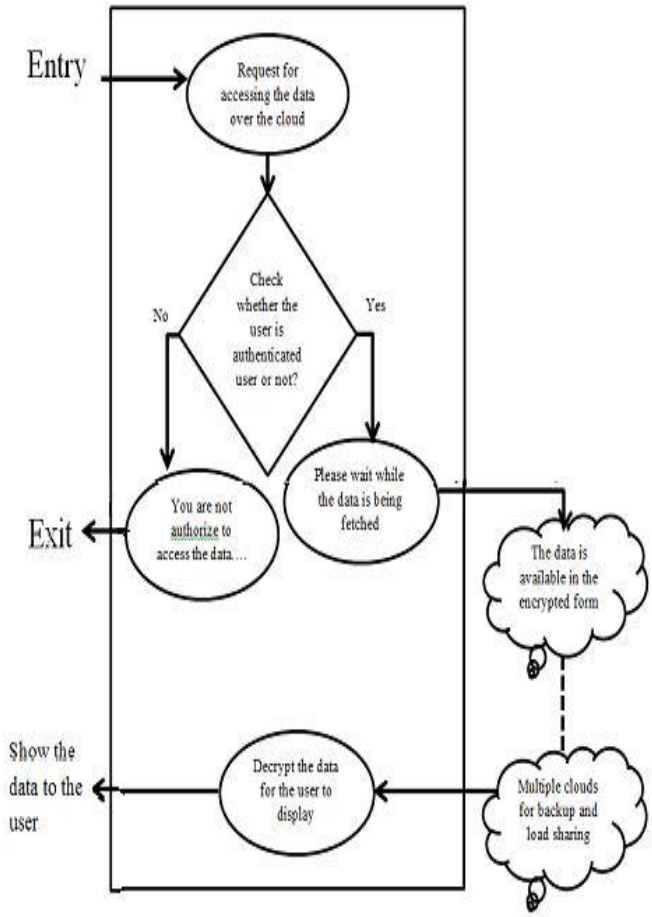

Fig 3: Flow chart

5. PROPOSED ALGORITHM

To cope with the convergent encryption weaknesses by using LFSR (Linear Feedback Shift Register) encryption technique including block level reduplication which preserves confidentiality and privacy even against potentially malicious cloud storage providers.

A. Goals and objectives:

The below objectives listed to solve the problem of privacy preserving reduplication in cloud computing and propose a new reduplication system.

1) Covergent Key Issue:

To overcome convergent encryption issue the LFSR technique for Encryption which preserves the combined advantages of block level reduplication and convergent encryption? It assures block level reduplication and data confidentiality while coping with weaknesses raised by convergent encryption. Block-level reduplication renders the system more flexible and efficient and preserves confidentiality and privacy even against potentially malicious cloud storage providers.

2) User Authorization:

To perform duplicate check based on his privileges, each authorized user is able to get his/her individual token of his file. Other user will not allow generating a token for duplicate check out of his privileges or without the aid from the private cloud server.

3) Authorized Duplicate Check:

To create query for certain file and the privileges he/she owned with the help of private cloud , authorized user is able to use his/her individual private keys, while the public cloud performs duplicate check directly and tells the user if there is any duplicate.

B. System Architecture:

The proposed system is divided into three sections: Client, Public cloud where the client likes to outsource the data and the Private cloud where the token generation will be performed for each file. Before uploading the data or file in public cloud, the client sends the file to private cloud for token generation which is unique for each file. Private cloud then generates a hash and token and sends the token to client. Token and hash keep in the private cloud itself so that whenever next file comes for token generation, the private clod can refer the same token. Once Client gets token for a particular file, public cloud search for the similar token if it exists or not. If the token exist public cloud will return a pointer of the already existing file otherwise it will send a message to upload a file. Public cloud breaks the file into blocks and generates the key with linear feedback Shift Register (LFSR) technique. The token and tag generate on public cloud which will then send to private cloud to update that the token has been generated for the particular file.

DOI: 10.18535/Ijecs/v4i11.33

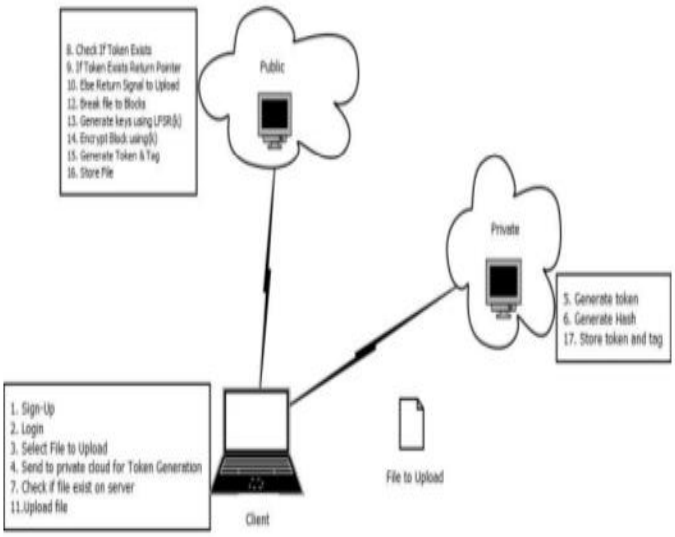

Fig 4: Architecture

C. Algorithms:

Algorithm 1: Advance Encryption Standard with LFSR technique.

Cipher(byte in[16], byte out[16], key array round key[Nr+1])

begin

byte state[16]; state = in;

LFSR(round key, tap);

AddRoundKey(state, round key[0]);

for i = 1 to Nr-1 stepsize 1

do

SubBytes(state);

ShiftRows(state);

MixColumns(state);

AddRoundKey(state, round key[i]);

end for

SubBytes(state);

ShiftRows(state);

AddRoundKey(state, round key[Nr]);

end

LFSR function :

uint16 t start state = 0xACE1u; uint16 t lfsr = start state; unsigned bit;

unsigned period = 0; do

{

bit = ((lfsr 0)^(lfsr 2)^(lfsr 3)^(lfsr 5) lfsr = (lfsr 1) j (bit 15);

++period;

} while (lfsr 6= start state); return 0;

Algorithm 2: MD5

Step 1: Appending Padding Bits. The original message is "padded" (extended) so that its length (in bits) is congruent to 448, modulo 512.

Step 2: Appending Length. 64 bits are appended to the end of the padded message to indicate the length of the original message in bytes.

Step 3: Initializing MD Buffer. MD5 algorithm requires a 128-bit buffer with a specific initial value.

Step 4: Processing Message in 512-bit Blocks which loops through the padded and appended message in blocks of 512 bits each. For each input block, 4 rounds of operations are performed with 16 operations in each round.

Step 5: Output. The contents in buffer words A, B, C, D are returned in sequence with low-order byte first.

CONCLUSION & FUTURE WORK:

Encryption algorithms play an important role in data security on cloud and by comparison of different parameters used in algorithms, it has been found that AES algorithm uses least time to execute cloud data. Blowfish algorithm has least memory requirement. RSA consumes longest memory size and encryption time. Proposed algorithms are helpful for today's requirement. In future several comparisons with different approaches and results to show effectiveness of proposed framework can be provided.

ACKNOWLEDGMENT:

It is not only customary but necessary for a researcher to mention his/her indebtedness to those who had helped in carrying out and enhance the research work.
I pay my deep regards to God, my Parents and my loving Friends for their support and wishes which made this tedious work easy and successful.
Finally, I would like to extend my thanks to all those who have contributed, directly or indirectly to make this project successful.

REFERENCES:

[1] M. Hogan, F. Liu, A. Sokol and J. Tong, "NIST Cloud computing Standards Roadmap, NIST Special Publication 500-291 (SP500-291)," Gaithersburg, July 2011.

[2] P. Mell and T. Grance, "The NIST Definition of Cloud Computing, NIST Special Publication 800-145 (SP800-145)," National Institute of Standards and Technology, Gaithersburg, September 2011.

[3] K. Scarfone and P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," NIST Special Publication 800-94 (SP800-94)," Gaithersburg, February 2007.

[4] G. Tyler, "Information Assurance Tools Report Intrusion Detection Systems," Information Assurance Technology Analysis Center (IATAC), September 2009.

[5] R. Robbins, "Distributed Intrusion Detection Systems: An Introduction and review, SANS Institute Information Technology Reading Room, GSEC Practical Assignment, version 1.4b, Option 1,

January 2002.

[6] William Stallings, ―Cryptography and Network Security Principles and Practices,‖ Prentice Hall, New Delhi.

[7] http://en.wikipedia.org/wiki/Google_App_Engine [8] Wayne A. Jansen, ―Cloud Hooks: Security and Privacy Issues in Cloud Computing‖, 44th Hawaii International Conference on System Sciences 2011.

[9] J. Baek, R. Safavi-Naini, and W. Susilo. Public key encryption with keyword search revisited. In International conference on Computational Science and Its Applications, pages 1249-1259. Springer-Verlag, 2008.