ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

REMOTE DATA BACK-UP AND PRIVACY PRESERVING DATA DISTRIBUTION IN THE CLOUD

Ruchira. H. Titare*, Prof. Pravin Kulurkar
* Mtech CSE, Vidarbha Institute of Technology, Nagpur, India.
Assistant Professor, Vidarbha Institute of Technology, Nagpur, India.

ABSTRACT

Cloud computing is the very hot topic in recent years. It is said to be the subsequent immeasurable thing in the world of computer after the internet. Cloud computing is the type of Internet-based computing which provides the online storage. No matter where the users are or which computers the users use, it enables the cloud users to put their data into the cloud. User can store and get the data placed in the cloud provided that it connects to the Internet. In cloud computing, very large amount of data is generated in electronic form. The cloud storage providers are responsible for keeping the data available and accessible to the user at any cost. When data is distributed, it is stored online at different locations increasing the risk of unauthorised physical access to the data. The service providers must provide security to the user's data. Also there is risk of data losses. So there is necessity of data recovery services in the cloud. To maintain the data integrity, we need to provide security. In this paper, we focus on cloud data storage security, which has always been a key feature of quality of service. We propose a secured data back-up technique for cloud computing in this paper. The two objectives are achieved by proposed technique, first it helps the users to recover their data files if the cloud is destroyed due to any reason and second is to provide the security to user data during storage on main cloud by using RC6 encryption algorithm. We have also provide the compression technique to store the back-up of data. Thus the time and storage related issues are also being solved by proposed techniques.

KEYWORDS: cloud computing, data integrity, data back-up, data recovery, RC6 encryption, compression.

INTRODUCTION

The data files or information concerning clients which is stored in any devices is lost due to hardware problem like if the system gets physically crashed or data gets corrupted then there is no other source to recover it. It is very complex job to manage various client records since work is done manually. There are lots of chances that the errors can occur in maintaining the various users and also there is large data storage problem in centralized system.

Cloud storage provides the online storage where data stored in form of virtualized pool that is usually hosted by third parties. Cloud computing lets you access all your application and document from anywhere in the world. The hosting company called CSP (Cloud service provider) operates large data on data center and according to the requirements of the customer these data center virtualized the resources and expose them as the storage pools that help user to store files or data objects. Either the human error, faulty equipment's, network connectivity, a bug or any criminal intent may put our cloud storage on the risk and danger.

Cloud Computing describes applications that are extended to be accessible through the internet and for this purpose large data centers and powerful servers are used to host the web applications and web services. As the sizes of IT infrastructure continue to grow, cloud computing is a natural extension of virtualization technologies that enable scalable management of virtual machines over a plethora of physically connected systems. The cloud computing has rapidly grown in recent years due to the advantages of greater flexibility and availability of computing resources at lower cost. Cloud service providers offer users efficient and scalable data storage services with a much lower marginal cost than traditional approaches. Cloud computing gives flexibility to the user, when users put their data in the cloud, they need not manage the information stored in cloud storage. Data sharing becomes a standard feature in most cloud storage offerings, including Dropbox, iCloud and Google Drive. As number of user shares the storage and other resources, it is possible that other customers can access your data.

The information and data that is stored on the Cloud is important to people with harmful intent so security is very important in cloud environment. Online backup, also known as remote backup, is a method of offsite data storage in which files, folders, or the entire contents of a hard drive are regularly backed up on a remote server or computer with a network connection.

In literature many techniques have been proposed SBA[1], PCS[2], secure data storage[3] E-health care[4], CyberGuarder[5], ERGOT[6] etc. that, discussed the data recovery process.The data integrity in cloud storage, however, is subject to timidity as data stored in the cloud can easily be lost or infected due to the inevitable hardware/ software failures and human errors. To make this matter even worse, hosting company needs to notify users about these data errors in order to maintain the status of their services and avoid losing profits. This application provides a feasible solution if the data is lost from main server and there is no other backup facility to restore this data.

To overcome these issues, we propose a new method in which we create the back-up of data in the remote cloud. If the data on main cloud gets unavailable then the remote cloud will offer the back-up of that data to the client. The admin panel will provide a secure ID to each client which will preserves the privacy of the clients. Also the data is store on cloud by enciphering original data with RC6 cipher which provides safety to the user's data. We are providing the compression technique to reduce the storage at remote cloud.

RELATED LITERATURE

In literature, we study most of the recent back-up and recovery techniques that have been developed in cloud computing area. The SBA technique help the users to collect information from any remote location in the absence of network connectivity and to recover the files in case of the file deletion or if the cloud gets destroyed due to any reason [1].

The PCS(Parity Cloud Service) is extremely simple, can completely relieve users of their concern about privacy protection, easy to use, requires a reasonable server-side cost, and can recover user data with sufficiently high probability. It generates virtual disk in user system for private data backup, makes parity group across virtual disks of multiple users, and stores the parity data of the parity group in the cloud storage[2]. In the technique "Secure Data Storage in cloud", the homomorphic token is introduced. The token computation function we are considering belongs to a family of universal hash function, chosen to preserve the homomorphic properties, which can be perfectly integrated with the verification of erasure-coded data [3].

The objective of Remote Data Collection Server: E-Health Care collects data and send to a centralized repository in a platform independent format without any network consideration [4].

All these existing methods tried to cover different issues maintaining the cost of implementation but it creates the large amount of data and requires more storage as the size of data is not reduced. Also there is lack of privacy and security in these techniques.

PROBLEM DEFINITION

In the cloud, huge amount of the user's personal data are stored but due to the failure of server or deletion of file, the secret data of the user will be lost. This paper will explain the process of cloud creation and data storage on cloud. The user is allowed for data storage only when authentication is made by admin panel. To provide the privacy to the user we provide ID to each user through the admin panel to authenticate the user.

We are also providing security to the user's data by using RC6 encryption algorithm. We are also creating the backup at the remote server so as to avoid data loss. We are providing compression technique to reduce the space at the remote server. Fig 1 shows the architecture of proposed technique.

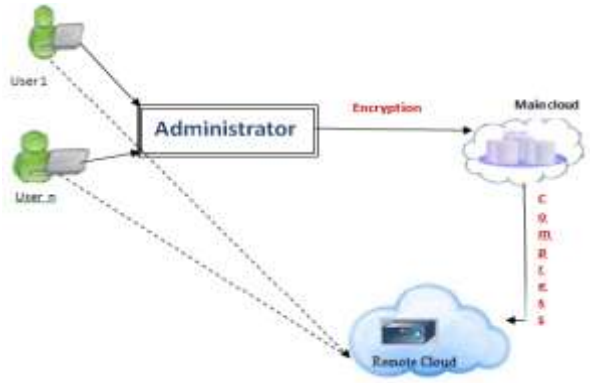

Fig. 1. Architecture of Privacy preserving data distribution and remote data back-up in cloud

PROJECT OBJECTIVES

The objective of proposed techniques is :

- To help the users to store the files on cloud.
- To recover files if the cloud gets destroyed due to any reason
- To provide user friendly and secure data storage on cloud by RC6 encryption.
- Privacy of user is maintained by providing a key to the user through admin panel.
- User data is compressed and store at remote server to minimize the space.

PROPOSED APPROACH

As discussed in literature, many techniques have been proposed for recovery and backup but implementation complexity, low cost, security and time related issues are still challenging in the field of cloud computing. So in this technique we are providing the secure data storage facility to each user on the cloud and also provide the recovery facility in case of data loss.

Overview

To achieve the objective of proposed technique, we have created a web application. In this we are providing the registration form to the user. After user is registered successfully, the data is send to the admin.

The admin then monitors the data and provide the authentication to the user. The admin send the key to the registered email ID of the user. After entering this key during login, the user's account will be activated. Then only user is allowed to upload data on the cloud.

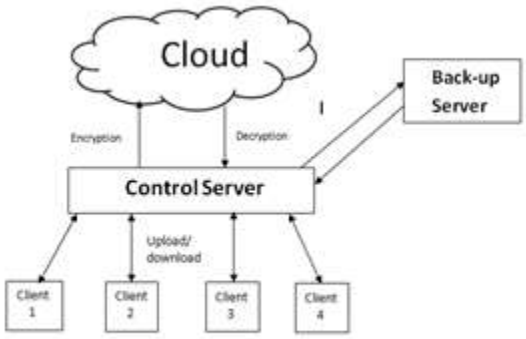

Fig 2. Secure data storage on cloud

Algorithm

Initialization: Main cloud: M_c

Client on Main cloud: C_i

Client ID: C_id_i, Password: psw_i

Admin: A, Client's Key: K_i
File to upload: F_i
Encrypted File: E_i
Decrypted file : D_i
Compressed file : CM_i
Decompressed file : DC_i

***Step 1*:** Create Client Registration Form containing all the fields of client details.
***Step 2*:** After the successful registration of Client C_i , the Admin panel A provides the key K_i for activation of Client's C_i account.
***Step 3*:** If the client is Authenticated then only client can login by entering Client *ID* C_id_i , Password psw_i and Key K_i .
***Step 4*:** The client *Ci* can upload any file F_i so as to access anywhere.
***Step 5*:** The Main Cloud M_c, performs the encryption on file F_i , and store encrypted file E_i on main cloud.
***Step 6*:** File F_i is compressed and then stored on remote cloud CM_i
Step 7 : While downloading the file F_i , user have to provide key K_i , If key matches, the file is decrypted D_i , and downloaded. Original file F_i = Decrypted file D_i
Step 8 : If server is crashed, then request is sent to remote server for backup. The file is decompressed DC_i , then downloaded from remote cloud.

Flow Diagram

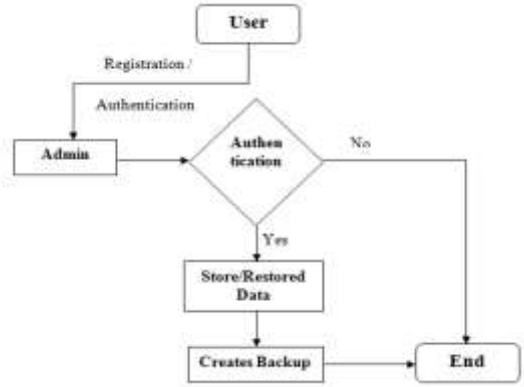

Fig 3. Phase 1:Create Back-up

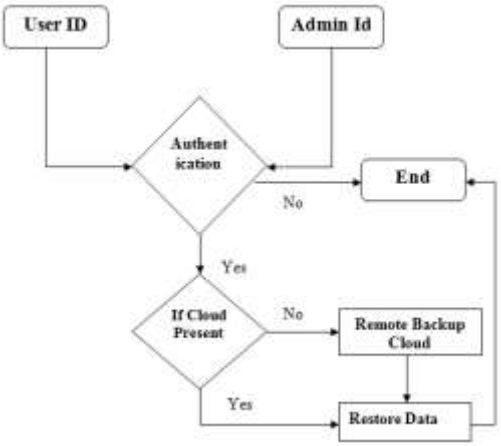

Fig 4. Phase 2: Restore data when cloud crashes

RC6 Cipher Algorithm

For encrypting and decrypting the file we are using the RC-6 algorithm. RC6 is a symmetric key block cipher derived from RC5. RC6 has a block size of 128 bits and supports key sizes of 128, 192 and 256 bits. It is very similar to RC5 in structure, using data-dependent rotations, modular addition and XOR operations; which could be viewed as interweaving two parallel RC5 encryption processes. However, RC6 does use an extra multiplication operation not present in RC5 in order to make the rotation dependent on every bit in a word, and not just the least significant few bits. In this algorithm, 128 bit plaintext is divided into four 32-bit blocks and then manipulated with the keys to generated cipher text.

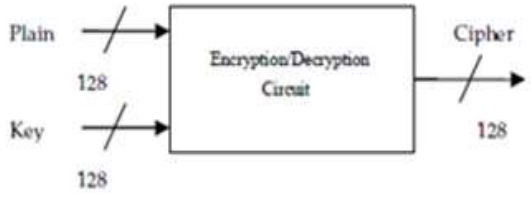

Fig. 5 Block diagram of RC 6 cipher

The user supplies a key of 'b' bytes and the number of rounds 'r'. From this, (2r+4) words (w bits each) are derived and stored in the array S [0… 2r+3]. This array is used in both encryption and decryption.

Encryption

Input:

- Plain text stored in four w-bit input registers A, B, C, D
- Number of rounds 'r'
- w-bit round keys S[0, … ,2r + 3]

Output:

- Cipher text stored in A, B, C, D

Procedure:

B = B + S [0]
D = D + S [1]
for i = 1 to r do
{
t = (B * (2B + 1)) <<< log w
u = (D * (2D + 1)) <<< log w
A = ((A $\oplus$) <<< u) + S [2i]
C = ((C $\oplus$) <<< t) + S [2i+ 1]
(A, B, C, D) = (B, C, D, A)
}
A = A + S [2r + 2]
C = C + S [2r + 3]

Decryption

Input:

- Cipher text stored in four w-bit input registers A, B, C, D
- Number of rounds 'r'

w-bit round keys S[0, … 2r + 3]

Output:

- Plaintext stored in A, B, C, D

Procedure:

C = C – S [2r + 3]

```
A = A – S [2r + 2]
for i = r downto 1 do
{
 (A, B, C, D) = (D, A, B, C)
 u = (D * (2D + 1)) <<< log w
 t = (B * (2B + 1)) <<< log w
 C = ((C – S [2i + 1]) >>> t)    u  ⊕
 A = ((A – S [2i]) >>> u)     t ⊕
}
D = D – S [1]
B = B – S [0]
```

Compression Techniques

The process of reducing the size of data is known as "data compression". In digital signal processing, data compression, source encoding and low bit rate reduction involves encoding information using fewer bits than the original representation. Compression is useful because it helps reduce resource usage, such as data storage space or transmission capacity.

Gzip compression

The deflation algorithm used by gzip (also zip and zlib) is a variation of LZ77 (Lempel-Ziv 1977, see reference below). It finds duplicated strings in the input data. The second occurrence of a string is replaced by a pointer to the previous string, in the form of a pair (distance, length). Distances are limited to 32K bytes, and lengths are limited to 258 bytes. When a string does not occur anywhere in the previous 32K bytes, it is emitted as a sequence of literal bytes. (In this description, `string' must be taken as an arbitrary sequence of bytes, and is not restricted to printable characters.) Literals or match lengths are compressed with one Huffman tree, and match distances are compressed with another tree. The trees are stored n a compact form at the start of each block.

JPEG compression

The acronym JPEG stands for the Joint Photographic Experts Group, a standards committee that had its origins within the International Standard Organization (ISO). JPEG, is a lossy compression algorithm for images.The algorithm behind JPEG is relatively straightforward and can be explained through the following
steps:

1. Take an image and divide it up into 8-pixel by 8-pixel blocks. If the image cannot be divided into 8-by-8 blocks, then you can add in empty pixels around the edges, essentially zero-padding the image.
2. For each 8-by-8 block, get image data such that you have values to represent the color at each pixel.
3. Take the Discrete Cosine Transform (DCT) of each 8-by-8 block.
4. After taking the DCT of a block, matrix multiply the block by a mask that will zero out certain values from the DCT matrix.
5. Finally, to get the data for the compressed image, take the inverse DCT of each block. All these blocks are combined back into an image of the same size as the original.

As it may be unclear why these steps result in a compressed image.

EXPERIMENTAL RESULTS AND ANALYSIS

We have created the web application which provides the facility of file uploading to user only when it is authenticated.

File uploading

Fig. 5 File upload

Original file

User can upload any text or image file. The following fig. shows the contents of original text file.

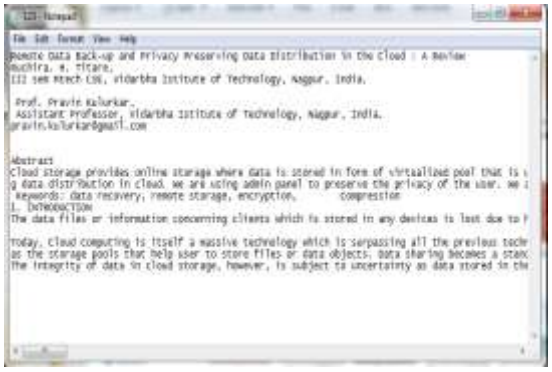

Fig. 6 Original File

Encrypted file

when the file is uploaded by user. It will be encrypted before it is stored on main cloud. The following fig. shows the contents of encrypted text file.

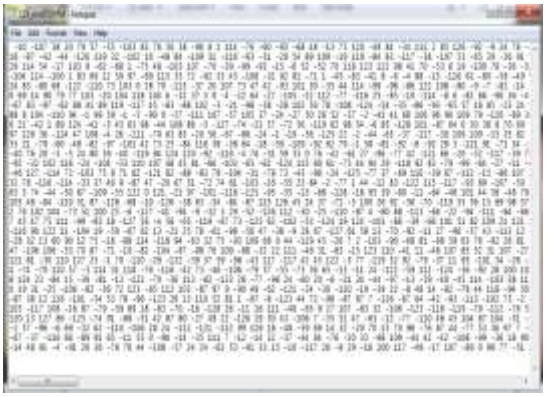

Fig. 7 Encrypted file

The following graph shows the storage required by main cloud and remote cloud. The file is compressed before storing on the remote cloud which decreases the space at the remote server.

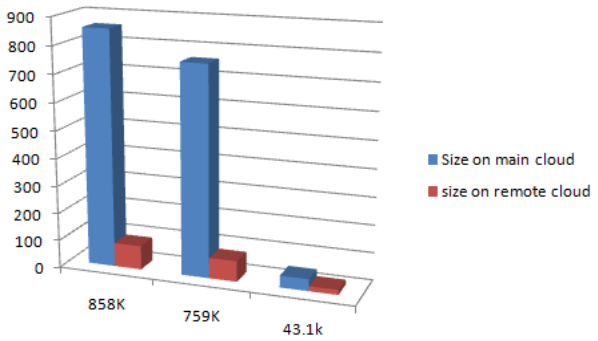

Fig. 8 storage on main and remote cloud

In above fig. the first to comparisons are of two different images in which the compression ratio of main and remote cloud is 10:1. The third bar shows the compression ratio of text file which is 3:1. Thus it shows that we can reduce large amount of storage on remote cloud.

CONCLUSION AND FUTURE SCOPE

This paper proposes a smart remote data backup technique. In this paper we have proposed the algorithm for registration and authentication of user. It also explains cloud creation on central server which provides the facility to store user's data. This technique provides the security to the user and user's data is stored on central server by using RC6 block cipher algorithm. The data is compressed and store at remote server which reduces the storage at remote server to large extent. Remote server provides the backup of users' data in case of cloud crash.

This application provide the facility of dat security on private cloud. In future this technique can be applied for public or hybrid cloud

REFERENCE

[1] Ms. Kruti Sharma, Prof. Kavita R Singh, 2013, "Seed Block Algorithm: A Remote Smart Data Back-up Technique for Cloud Computing", 2013 IEEE international conference on communication system and network technology, 6-8 April 2013, ISBN 978-1-4673-5603-9

[2] Chi-won Song, Sungmin Park, Dong-wook Kim, Sooyong Kang, 2011, "Parity Cloud Service: A Privacy-Protected Personal Data Recovery Service," International Joint Conference of IEEE TrustCom-11/IEEE ICESS-11/FCST-1

[3] B. Shwetha Bindu, B. Yadaiah, "Secure Data Storage In cloud computing", International Journal of Research in Computer Science ISSN 2249-8257 Volume 1 Issue 1 (2011) pp. 63-73

[4] Kalyani Bangale, Nivedita Gupta, Swati Singh Parihar, "Remote Data Collection Server : E-Health Care" International Journal of Innovative Research in Computer and Communication Engineering, An ISO 3297: 2007 Certified Organization, Vol. 2, Issue 2, February 2014.

[5] Milind Mathur, Ayush Keasarwani, "Comparison between DES, 3DES, RC2, RC6, BLOWFISH and AES", Proceedings of National Conference on New Horizons in IT - NCNHIT 2013

[6] Fouad Ramia, Hunar Qadir, "RC6 Implementation including key scheduling using FPGA", ECE 646 Project, December 2006

[7] Boyang Wang, Baochun Li, Hui Li have presented a new technology "Oruta: Privacy-Preserving Public Auditing for Shared Data in Cloud", IEEE transactions on cloud computing, vol. 2, no. 1, january-march 2014.

[8] S.Ezhil Arasu, B.Gowri, S.Ananthi presented "Privacy-Preserving Public Auditing In Cloud Using HMAC Algorithm" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-2, Issue-1, March 2013.

[9] Giuseppe Pirr´o, Paolo Trunfio, Domenico Talia, Paolo Missier and Carole Goble, 2010, "ERGOT: A Semantic-based System for Service Discovery in Distributed Infrastructures," 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing.

[10] [Kruti Sharma, Kavita R Singh, "Online Data Back-up and Disaster Recovery Techniques in Cloud Computing: A Review" ISSN: 2277-3754 ISO 9001:2008 Certified International Journal of Engineering and Innovative Technology (IJEIT) Volume 2, Issue 5, November 2012

[11] Lili Sun, Jianwei An, Yang Yang, Ming Zeng, 2011, "Recovery Strategies for Service Composition in Dynamic Network," International Conference on Cloud and Service Computing.
[12] D. Srinivas, "Privacy-Preserving Public Auditing In Cloud Storage Security", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (6) , 2011, 2691-2693.
[13] Tejashree Paigude, Prof. T. A. Chavan, "A survey on Privacy Preserving Public Auditing for Data Storage Security", International Journal of Computer Trends and Technology- volume4Issue3- 2013
[14] Cong Wang, Qian Wang, and Kui Ren, "Ensuring Data Storage Security in Cloud Computing"