International Journal of Advance Foundation and Research in Computer (IJAFRC)
Volume 2, Special Issue (NCRTIT 2015), January 2015. ISSN 2348 - 4853

Review On Incremental Encrypted Backup For Cloud

Rohini Ghenand, Pooja Kute, Swapnil Shinde, Amit Shinde, Mahesh Pavaskar, Shitalkumar Jain
Department of Computer Engineering MIT AOE
rohinighenand26@gmail.com

A B S T R A C T

With the development of science and technology the capacity of hard disk and quantity of data are continuously increasing. Backup has become an important mechanism for prevention of data loss. When the amount of data is small, full backup is appropriate. However, if the amount of data is considerably large, taking full backup every time becomes time consuming. In this case, incremental backup is used for saving time. Incremental backup do not create multiple copies of data; so, incremental backup is faster than full backup. To solve the related security problem, encryption is necessary when the backup data are stored in the storage server. Though, most incremental backup cannot be encrypted. Thus, this paper presents a method to take encrypted incremental backup with the help of iscsi protocol. It will require for additional storage space when amount of hard disk full. The encrypted incremental backup collect information from every file and stores it into a single file called the checksum file. This information contains filename, last modified time, file size, and delete stamp, checksum, and encryption key. When the backup begins, the client collects the filename, checksum, last modified time, and file size. Then, the client gets another checksum file from the storage server. By comparing these two checksum files, the system will know which files have been changed and should be transmitted in the backup this time. Before these files are sent to the storage server, the client will generate random keys to encrypt the files and store the encrypted keys in the checksum file. The system will use another key (key encryption key; KEK) to encrypt the checksum file; the administrator password is used for encrypting/decrypting this KEK.

Index Terms -Incremental Backup, Encrypted Backup, iscsi, Digital Signature, Raid.

I. INTRODUCTION

Extensive network coverage and mobile devices have made information and files readily available. The security of business information and user's data has become pivotal for enterprises Therefore; there is a need for security technology that can be used for protecting backup data in a storage server. The encrypted backup to the backup data that have been encrypted to keep data secure. If such a technology is used, then even if other users get these encrypted data, they will not be able to retrieve the data content easily. On the other hand, because the amount of data used in day-to-day life has increased considerably in the recent years, incremental backup is has become increasingly important now. The main concept of incremental backup is to back up only the difference between two file versions as doing

so will reduce the transmitting time and the data transfer. Unfortunately, as encryption protects the file content, it may not allow the incremental backup algorithm to detect the parts of a file that have been changed since the last backup. Therefore, most incremental backup methods do not support encryption.

II. GOAL AND OBJECTIVES

A. Support Incremental Backup

We take backup on timely basis, so that it will reduce the transmitting time and the data transfer speed. That means we will transmit only changed data from last modified time.

B. Support Encrypted Backup

By using encryption algorithm we make our system to support encrypted backup.

C. Support Different Storage Servers

In this section, the system already has the transmitting file list and encrypts all the files in the list. Now, we have to decide the storage server to store the encrypted data in. Considering that customers have difference storage servers that they want to backup to, this backup system must have the ability to store data in different storage servers.

III. PROPOSED SYSTEM

In our proposed system, for logging and monitoring required check files. For security of data we required encryption algorithm. Check files, encryption and storage server described as follows.

A. Check Files

To determine what parts of a file have been changed since the last backup and need to be transmitted to the storage server in the current backup. After the data are encrypted, the system cannot compare the original file with the encrypted file for determining the modified parts of the file. Therefore, the system must have the file information from the last backup for determining what parts of the file have been changed. In this proposed method, the system will collect file information before the backup; this file information includes the full path of the file, last modified time, file size, checksum, delete stamp, and encryption key. Therefore the system can compare the two checksum lists to determine which part of the file has been changed from the last backup.

B. Encryption

After the system generates the transmitted file list, all files in this list will be encrypted before sending files to the storage server. Encryption is the process of transforming data using an algorithm to make the data unreadable to anyone except those possessing a special key. Using encryption to protect certain information and data from other people is a very common procedure.

C. Storage Server Switcher

Because of different users have different storage server requirements, therefore a storage server switcher will transmit data through a different application interface on the basis of the storage server that the user chooses. After the encryption of a file, the encrypted data will be treated as normal data in the storage server, and the storage server should have the ability to perform only a few basic functions such as move, copy, and transmit file. When a user wants to original file, the system will use the checksum file to retrieve the encrypted file from the server and decrypt it on the client side. Such way that, it becomes a simple storage space because most of the computing is performed on the client side. It is extremely easy to implement this system. We need not to implement split file, checksum, and encryption ourselves. We can use well-designed open-source software to help us complete this system.

IV. MODULES OF PROJECT

A. Module 1: Data security

1. **Symmetric-key algorithms** are a class of algorithms for cryptography that use the same encryption keys for both encryption of plaintext and decryption of ciphertext. The keys may be identical or there may be a simple transformation to go between the two keys. The Advanced Encryption Standard (AES) algorithm.

2. **Asymmetric-key algorithms**, is a class of algorithms which requires two separate keys, one of which is secret (or private) and one of which is public. Although different, the two parts of this key pair are mathematically linked. The public key is used to encrypt plaintext or to verify a digital signature;

B. Module 2: Backup Technique

1. RAID 1

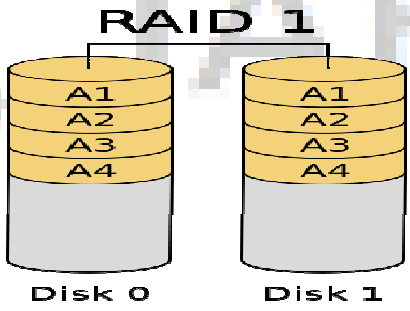

Figure 1. Raid 1

An exact copy (or mirror) of a set of data on two disks. This is useful when performance read or reliability is more important than data storage capacity. Such an array can only be as big as the smallest member disk. A classic RAID 1 mirrored pair contains two disks.

C. Module 3: Monitoring

1. DSA (Digital signature Algorithm)

Digital signatures are essential in today's modern world to verify the sender of a document's identity. Digital signature algorithm that would need to assure the integrity and originality of data.

2. ISCSI

iSCSI is an Internet Protocol based storage networking standard for linking data storage facilities. By carrying SCSI commands over Internet Protocol networks, iSCSI can facilitate data transfers over local area networks (LANs), wide area networks (WANs), or the Internet. Through iSCSI, the space on server will be treated as local disks by client's operation system. But actually, all data transferred to the disk are actually transferred over network to the storage server.

3. User Quota

There are two type of disk management:-

1. Soft Quota – It gives warning message when user access more space than allocated. But it will not blocked write operation.

2. Hard Quota- It will blocked write operation, when user access more space.

D. Module 4: Incremental backup Technique

For incremental backup technique, we are generating log file .This file contain file name, last modified time, file size, checksum, delete stamp, encryption key.

By using this file only modified data will be reflecting on storage server.

V. SYSTEM ARCHITECTURE

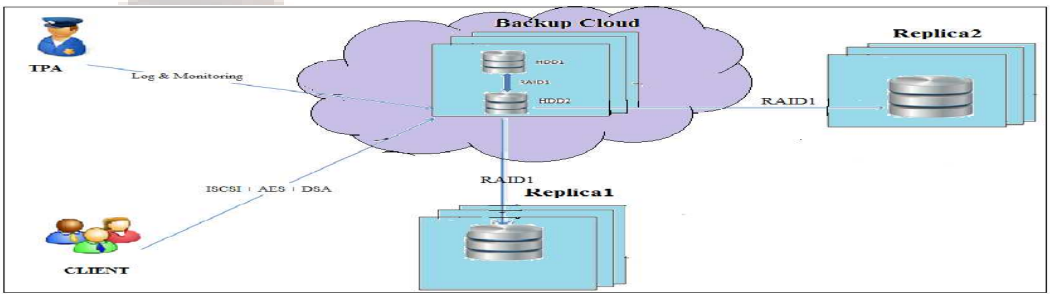

Figure 2. Architecture Of Incremental Encrypted Backup For Cloud

VI. CONCLUSION

The goal of the project was to design and implement a cloud application for an online backup and for the restoring of clients' data. A successful move to cloud backup requires ensuring data protection requirements are met while overcoming or mitigating the issues associated with traditional data protection methodologies. Incremental backup save the disk space and time as compare to traditional backup methods.

VII. REFERENCES

[1] Microsoft Corporation. Description of full, incremental, and differential backups [online]. Microsoft support; 15 November 2006.

[2] URL:http://support.microsoft.com/kb/136621

[3] F. Hou, N. Xiao, F. Liu, H. He, "Secure Disk with Authenticated Encryption and IV Verification", Fifth International Conference on Information Assurance and Security, pp. 41-44, 209.

[4] J. Li, H. Yu, "Trusted full disk encryption model based on TPM", 2nd International Conference on Information Science and Engineering, pp. 1-4, 2010.

