Secured Multi Cloud Storage using CPDP

S. Shanmuga Priya[1] P.K. Sheela Shantha Kumari[2]
[1,2]Assistant Professor
[1,2]Vel Tech Hightech Dr. Rangarajan Dr. Sakunthala Engineering College

Abstract— PDP is a technique designed for CSP to show that complete file of the client is in secured state without downloading the whole file. PDP scheme of multi cloud includes multiple csp's to mutually store and maintain the client's data. Later CPDP was proposed which also has the Homorphic verifiable response and Hash index hierarchy properties, but there is a security flaws. The problem is, when a malicious csp or organizer generates a valid response which also clears the verification process in case of deletion of the stored data. This means, simply, an attacker gets the information without storing the client's data. In this paper, we discuss about the security flaws in CPDP (Cooperative Provable Data Possession) scheme.

Key words: CPDP, PDP, TPA, Zero Knowledge, HIH

I. INTRODUCTION

Generally cloud computing is internet based computing in which large groups of remote servers are networked to allow sharing of data processing, data storage and online access to the resources. Cloud computing provides flexibility, security, cost efficient, documental control and environmental friendly.

Cloud computing has three different deployment models: public, private, hybrid. When multiple cloud services provide a distributed environment, then it is called as multicloud. In multicloud, multiple cloud service providers provide the data integrity and security in data.

In this paper, we discuss about the security and data protection.ie the upload and download of data, also there is technique introduced for the security of storage services. The scenario is, when the user wants to upload a file in the multicloud environment, the file will be encrypted separately according to file size and encrypted file will be storage in different active clouds. The decryption is applied to all separated files when the user wants to download the file. The rest of the paper explains the existing system, proposed system and its modules, experimental result for the proposed system (hacker side from one active server).

II. EXISTING SYSTEM

In existing system the whole file will be stored in cloud for example if file1 to be stored in Google then the same full file will be stored in Amazon, that is the full copy of our file will be stored in different cloud. so if hacker tries to hack our file from database they will get our full file, then if they found the whole file, it will be hacked, the information in the file will be recovered and they use it, another problem faced in the existing system is the cloud admin or the third party who view the file can make changes to the file, updation of file from unknown user is also possible. While user uploading the files strictly to the cloud or server, first to store our file in cloud we need to purchase the space in cloud for that first we need to register in cloud and authentication of the particular user will be provide, then we upload our file to the cloud So, Server or Cloud can easily modify our file content stored in the cloud the file stored in cloud is verified by the third party and admin will view it, sometimes server will update or make changes to our file without our permission, so any cloud service provider cannot assure the security of attacks from outside enterprise cloud.

III. PROPOSED SYSTEM

A. Techniques Used

1) Homorphic Response

Homomorphic encryption is a form of encryption that allows computations to be carried out on cipher text, thus generating an encrypted result which, when decrypted, matches the result of operations performed on the plaintext.

2) Zero Knowledge

Zero knowledge means it differentiate user from hacker, i.e. if user login to the system we first register in cloud for storing our file in provided space, then user login and upload the file in cloud and then any modification user can proceed, but if hacker hacks the file directly hacker will view the file storage phase, then it is zero knowledge proof identified by admin

3) Hash Index Hierarchy

The hash index hierarchy which describes where next part of file is stored in the server, by this we can retrieve our file from different server. Express Layer offers the abstract representation of the stored resources; Service Layer offers and manages cloud storage services; and Storage Layer realizes data storage on many physical.

B. Algorithm Used

1) Keygen

KEYGEN algorithm which determines when we upload our file a key will be generated automatically, for next file another key will generated, for downloading same key should be used.

2) Taggen

TAGGEN algorithm is used which take secret key, file and cloud service provider then it returns combination of the input taken, and allocates the space by providing the hash index table to represent the location where our data is divided and stored.

C. Modules

1) Multi Cloud Storage

Multi cloud means use of multiple cloud computing services in a single architecture the reason for using multicloud is to reduce the risk of data loss or downtime due to failure of component used in cloud computing environment. In our system, user allow their data to be stored in different cloud for that first we need to purchase space in the cloud, then user will upload the file in cloud, using the interface client can easily access the remote data, for that we first provide registration phase which includes all the required fields user name, password server, server size, domain, domain size,

date, plan and once we complete this registration phase it moves to cooperative provable data possession phase which provide the authentication for the user who registered or provided space in the cloud.

2) CPDP and Data Integrity

Automatically after registration phase is completed, we get a authentication for particular user, which provides the detail about user name password, server size, domain, domain size, cloud purchase details and when cloud expires, also selecting the required number of sectors in specified block will minimize the computation cost of the client, in cooperative provable data possession we also have a knowledge proof which specifies the hacker from user, if user login to the system we first register in cloud then login next step is viewing the file but if hacker login, they will directly view or download the file, reliable access to data is pre request for most of the computer system several factor cause modification to our data stored in the cloud hence data integrity is very important because it ensures weather data is consistent and accessible, in our system modification in stored file is not possible all our details are encrypted and stored, we can recover our data without any modification.

3) TPA

Here we use third party auditor to view our file ,verifies and then upload in the cloud, after uploading file to the cloud, if any hacker or unknown person tries to modify the content won't be modified and alert will be sent to trusted third party then they will change the encryption format to the file stored in different cloud, also third party auditor will eliminate the work of client by auditing the file weather it is in secured state and by this management of file becomes easy task.

4) Cloud User

Once Cloud user stored the data in multicloud, it will divided and stored according to the active server available, we can open our file if any modification or updation is needed we can perform in our file and hence if we download the file data from different part will be merged and then it is downloaded as a complete file. for recovering data from different server we use The TAGGEN algorithm is used which take secret key, file and cloud service provider then it returns combination of the input taken ,and allocates the space by providing the hash index table to represent the location where our data is divided and stored.

5) Disaster Recovery

Disaster recovery is providing a backup when primary data is not available a new copy of the application will be available for accessing, if any deletion of data occurs due to sudden power shortage or any disaster occurrence, we can take the backup from cloud which is maintained only in one cloud, in our system once the file is uploaded in cloud we remove or delete our file from the local system hence we can show that when the file is downloaded it is downloaded from the database by this we can safe guard the data for back uping process we use least common denominator cloud interface. There are different back up sites available like hot backup site and warm backup site.

6) Re-encryption

In our proposal we have re-encryption process as when we upload our file to cloud it will be divided according to the active server available and stored in different server as specified, if any one server is hacked, the hacker can view our file but it will be in encrypted format, if they find encryption format used by us, then our file will be viewed information can be recovered but with of no use because a part of our data only available in this file rest part is stored in another cloud, so attacker before reaching our files in different cloud, automatically encryption format for files stored in different server will be changed, we also have the technique for changing encryption format any number of times.

IV. EXPERIMENTAL RESULTS

In our system, first we register in cloud for providing space for our file to be stored in cloud by this we reduce the computational cost of the client and storage area, then the files will be stored in cloud as different parts for example if we upload file1 to the cloud, the file will be divided into different parts according to the active server available ,then each part will be stored in different cloud, this is maintained by the admin using the hash index hierarchy which describes where next part of file is stored in the server, by this we can retrieve our file from different server.

When we download the file or transfer the file, different parts from multicloud are merged and retrieved as a single file, attacker is a major problem in existing system, we control the attacking by when attacker hacks the file they can't view our file it will be in encrypted format, if hacker found the encryption format they can view our file but cannot modify it, even if they gather information it is of no use ,only part of file is available so they can't proceed further and alert is sent to third party auditor if our file is hacked then automatically encryption format in other files stored in different server will be changed.

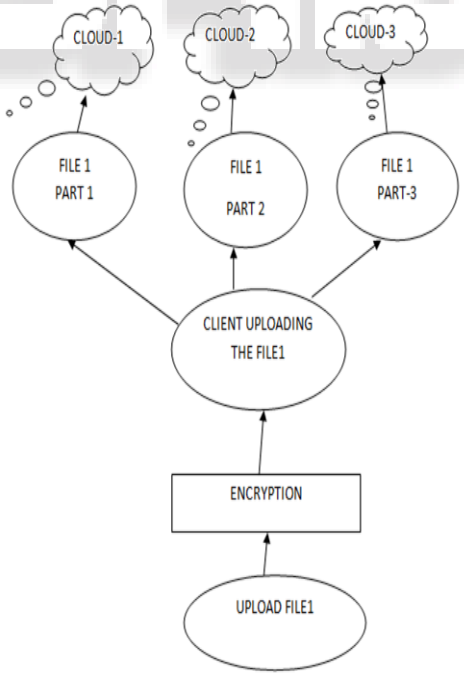


Fig.1: Uploading File

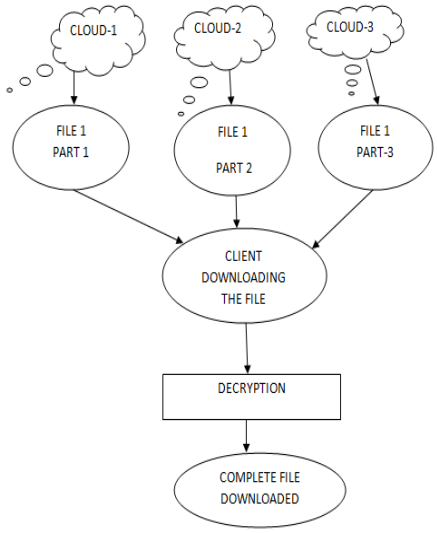
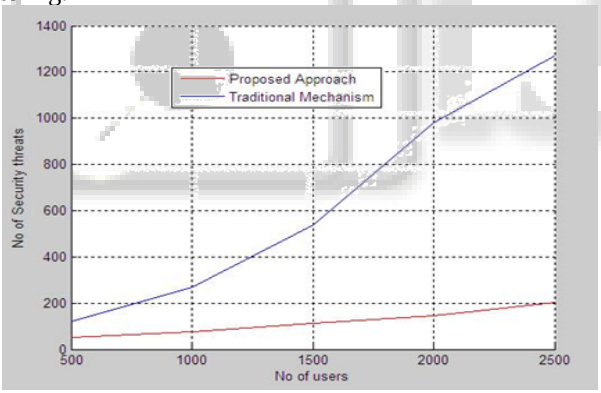


Fig. 2: Download File

V. PERFORMANCE ANALYSIS

We simulated the proposed solution for providing service availability, attack against data intrusion and data integrity. We measured the performance in terms of number of security threats with and without our mechanism. We varied the number of users in the cloud for 4 cloud accounts and 10 storage server in each cloud. From the performance chart that our proposed mechanism is able to reduce the number of security attacks gradually thanks to the adaptive user security profile setting and the reputation based server filtering.



VI. CONCLUSION

Thus we provide a secured multi cloud storage which divides our file according to the available active server and stores accordingly in the multi cloud. Also security is provided in uploading and downloading files, thus when we upload the file, third party will verify and then it will be uploaded to the cloud,so any file corruption can be avoided in first step itself then accordingly file will be divided then it will be stored, while downloading only particular user can download it, hacker cannot do so, complete file from different server will be merged and then downloaded.

REFERENCES

[1] Provable Data Possession at Untrusted Stores, Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 598-609, 2007.
[2] Scalable and Efficient Provable Data Possession, Proc. Fourth Int'l Conf. Security and Privacy in Comm. Networks (Secure Comm '08), 2008.
[3] Dynamic Provable Data Possession, Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), pp. 213-222, 2009.
[4] Cooperative Provable Data Possession for Integrity Verification in MultiCloud Storage, IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231-2244,Dec. 2012.
[5] Efficient Provable Data Possession for Hybrid Clouds, Proc. 17th ACM Conf. Computer and Comm. Security (CCS '10), pp. 756-758, 2010.
[6] Proxy Provable Data Possession in Public Clouds, IEEE Trans. Services Computing, DOI: 10.1109/TSC.2012.35.
[7] Cooperative Provable Data Possession for Integrity Verification in MultiCloud Storage, IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231-2244, Dec. 2012.
[8] On the Power of Multi-Prover Interactive Protocols, Theoretical Computer Science, pp. 156-161, 1988.
[9] Identity-Based Encryption from the Weil Pairing, Proc. 21st Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO '01), pp. 213-229, 2001.
[10] New Explicit Conditions of Elliptic Curve Traces for FR-Reduction, IEICE Trans. Fundamentals, vol. 5, pp. 1234-1243, 2001.

