Advanced Data Integrity Checking Mechanism in Cloud

Devika Shetty M
Dept. of Computer Science and Engineering
Alva's Institute of Engineering and Technology
Moodbidri, India

Ashwini P Shetty
Dept. of Computer Science and Engineering
Alva's Institute of Engineering and Technology
Moodbidri, India

Deeksha
Dept. of Computer Science and Engineering
Alva's Institute of Engineering and Technology
Moodbidri, India

Deeksha Shetty
Dept. of Computer Science and Engineering
Alva's Institute of Engineering and Technology
Moodbidri, India

Chanchal Antony
Prof. Dept. of Computer Science and Engineering
Alva's Institute of Engineering and Technology
Moodbidri, India

***Abstract*—Cloud computing is one of the important area, which includes the delivery of hosted services over the Internet. Both time and money can be saved by using the cloud services. The major application of cloud computing is cloud storage, which offers user scalability, flexibility, high quality storage and computation services. Files are outsourced to the cloud by the data owner for the long term storage. Cloud storage servers are not fully trustable; data owner requires dependable means to check the integrity of the files which is outsourced to the remote cloud servers. To solve this problem, data integrity checking mechanism has been presented. Many of the existing approaches have susceptibility in efficiency or data dynamics. However, there are some security issues which are to be solved for end users and enterprises for the storage of data in the cloud. The fact is that users will lose the physical control of their data after outsourcing. The data stored in the cloud can be hacked by the hacker, so the cloud user is worried about the integrity of the data stored in the cloud. The main goal of the project is to suggest an efficient public auditing technique using Third Party Auditor (TPA) to verify the integrity of data stored in the cloud. In the proposed system encryption and decryption can be performed by using AES algorithm and Secure Hash Algorithm (SHA-1) algorithm is used to generate verification metadata to check the integrity of the data.**

***Keywords*—Cloud Computing, Cloud Service Provider (CSP), Data Integrity, Third Party Auditor (TPA)**

I. INTRODUCTION

Cloud computing is a computing model in which resources are shared as a service over the Internet. Cloud computing provides many services to the users. Data storage is one among them. Data storage in cloud provides the facility of maintaining the data, managing the data, backing up the data and making the data available to the cloud users over the Internet. The advantage to the users of cloud is that they can easily move their data or information into the cloud and the users need not to worry about the complexities of hardware handling. Amazon Simple Storage Service and Amazon Elastic Compute Cloud are well known examples of cloud computing vendors. These Internet based online services provides large amount of space for the storage and resources. This helps the users to save the storage space locally and data can be maintained easily.

Even though these benefits are provided by the cloud, users are unwilling to use this computing method because of the security issues [4]. The data or the information in the cloud can be hacked from an outsider or an insider so the users are reluctant to adopt this technology. Once the file is uploaded to the cloud the users will not have any information about the integrity of the file, that is whether the data has been altered or not. To overcome this we introduce integrity checking mechanism through which user can verify the integrity of the data easily [8].

The user sends the request to the Third Party Auditor (TPA) to verify the data that has been uploaded by the user to the cloud [5], [7]. The TPA receives the request and further it will be sent to the cloud from the TPA, this process of verification is known as data auditing [6]. TPA does not contain any copy of client's data, the TPA just acts as a mediator between the user and cloud. Dynamic operation is implemented in this system where the user can modify the data directly in the cloud without downloading the file.

II. PROBLEM STATEMENT

The cloud storage system includes two participants: the CSS and the data owner. The CSS has powerful storage ability and computation resources, it accepts the data owner's requests to store the outsourced data files and supplies access service. The client starts moving all his files into the cloud storage without maintaining the copy of that file. As the cloud storage can be hacked by the insider or an outsider and cloud storage is not trust worthy and it can misbehave [2].

In the existing system, data owner do not have any option to check the data integrity of the information or the files stored in cloud storage. Data owner was not able to detect the

modified data if it is modified due to unintentional catastrophic circumstances. At that time, data owner will consider received cloud data as original data although it is altered by an attacker. There is no system to verify the downloaded data for its integrity.

III. PROPOSED SYSTEM

The proposed system helps the data owner to verify the integrity of the data stored in the cloud. Data owner can verify whether data is modified by cloud or not. This provides both privacy and integrity to the cloud data.

Using proposed system, data owner can also perform dynamic data operations with cloud. This increases the throughput of the system since there is no need to download and then upload the data by the data owner when the data file is to be modified or altered.

Figure 1 depicts the architecture of the proposed system. The proposed system is comprised of three units the cloud user, the Third Party Auditor (TPA) and the cloud server.

First the owner of the data or the user will authenticate himself to the cloud by using his password and secret key provided to him during the registration process. After logging in to the cloud, the user can perform the following operations: uploading a new file to the cloud, view the uploaded files, perform dynamic operation on the file stored in the cloud and can also download the file whenever required. When the user selects the file to upload on to the cloud, first the selected file will be partitioned into three parts and the three parts of the file will be encrypted using AES algorithm [1], followed by generating message digest using Secure Hash Algorithm-1 [3]. The encrypted version of the file will be sent to cloud for storage and the generated hash code will be sent to the TPA.

The main task of the TPA is to perform data auditing only on demand by the client. When the TPA receives the request for audit from the data owner, he requests the cloud for the encrypted file. After receiving the file, the TPA generates the metadata for the file using the SHA-1 algorithm. TPA makes use of the hash code also known as metadata for verifying the integrity of the file uploaded to the cloud. As only the message digest is sent to the TPA during the file upload, the TPA will not have access to the actual data stored by the user.

The TPA compares the newly generated message digest value with the earlier message digest sent by client. If both the hash values are equal then it indicates that data integrity of the file is maintained. If the metadata generated by the TPA and the metadata sent by the client are not equal then it can be inferred that the data has been modified by an attacker and integrity of the file is not maintained. The results of the data auditing or proof verification will be sent to the data owner.

Fig 2 and Fig 3 shows the operations performed by the user during file upload and the overall working of TPA. When the cloud gets a request for the file from the TPA, the cloud will send the blocks of the requested file. This method helps the cloud users to upload files to the cloud server and rely on TPA for verifying the integrity of the file stored in cloud server. Fig 4 illustrates the flowchart for verification .The TPA checks if there are any pending requests from the client and if any it requests the cloud for the file and verifies the integrity.

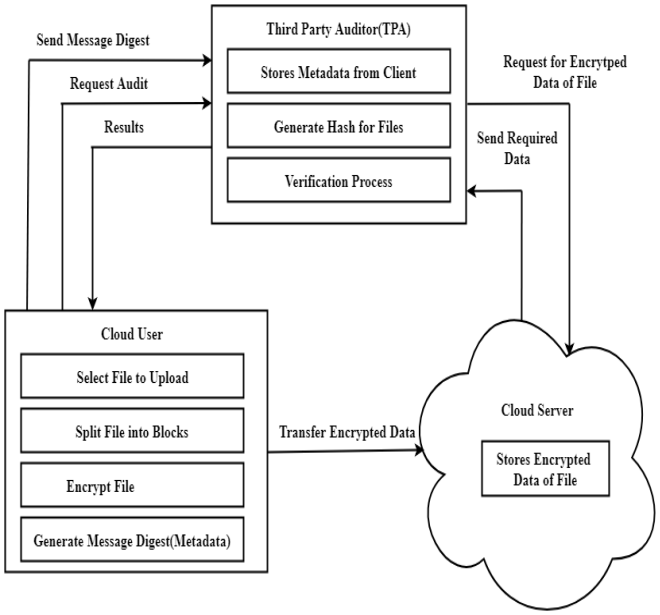

Fig 1: Architecture of Proposed System

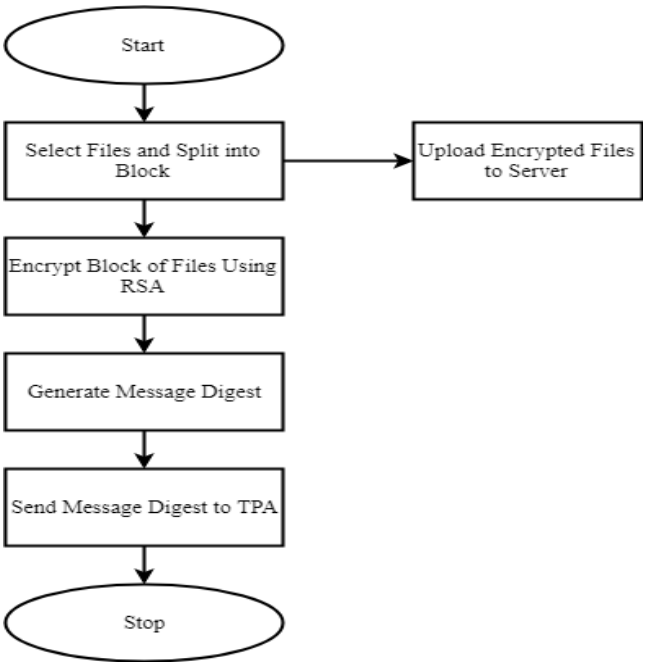

Fig 2: Working of Cloud User

IV. IMPLEMENTATION AND RESULTS

The proposed system is implemented using Java programming on a system with Intel core i3 processor running at 2.33 GHz and 3GB RAM. JSP and CSS3 are used to develop the front end. SQLyog is used as a backend tool. To encrypt the files the proposed system will use the Advanced Encryption Standard (128bit encryption)

algorithm. Secure Hash Algorithm (SHA-160 bit) is also implemented in java for generating the hash keys for the encrypted file.

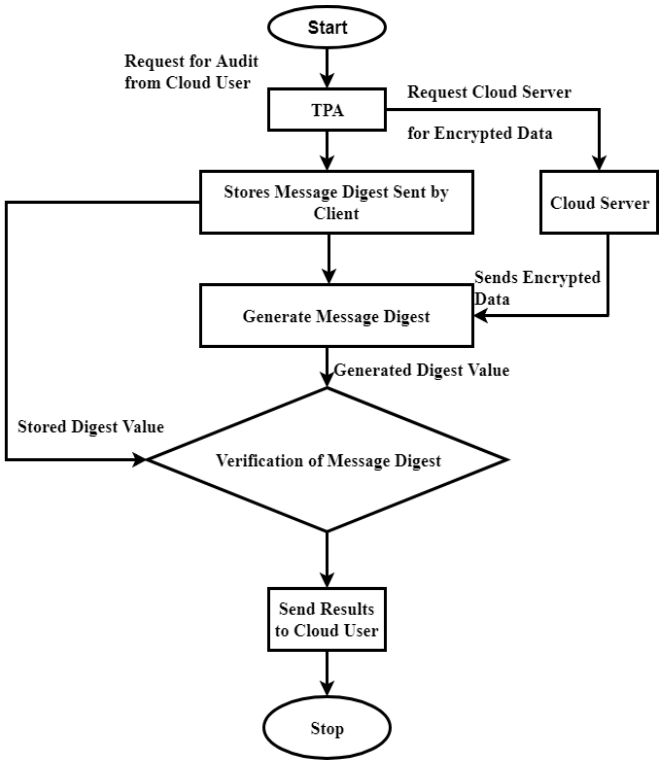

Fig. 3: Working of TPA

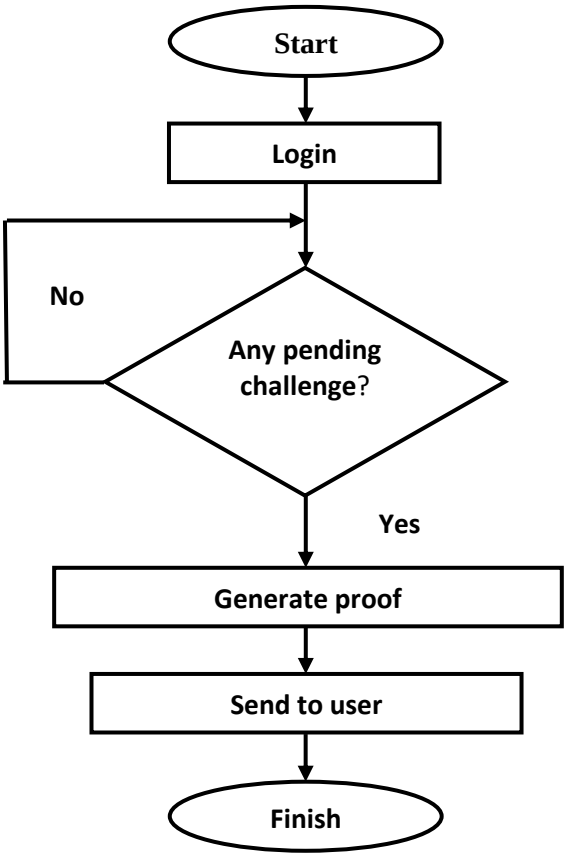

Fig 4: Flow Chart for Verification

SHA 1 is used to generate a message digest. It generates an almost-unique 160-bit cryptographic hash or message digest for the text. The following are some of the 160 bits digest we obtained during the analysis of our proposed scheme:

1. 2c25874babc190d51c5a67d3d25633f8eb0cd5db
2. f73e5683da44ddd71e2d7fe90fad3ca5064b0aa6
3. b8508b6d27dd0053ccbace7561d62dc17f6344a4

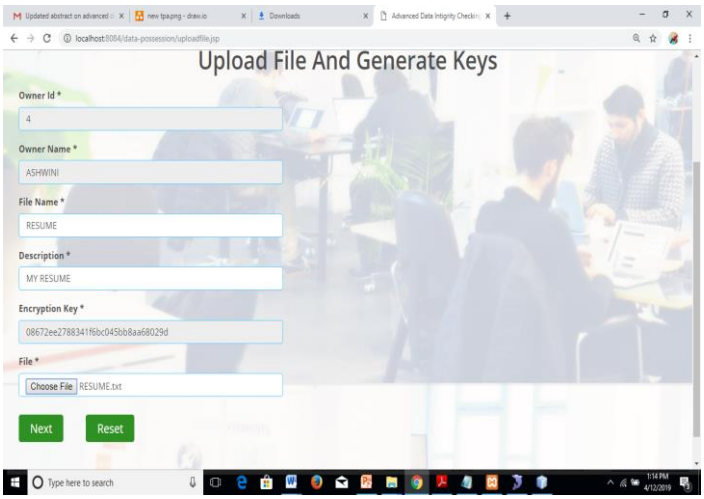

Fig 5: File upload by data owner

The data owner has to register him to the cloud before performing any operation. Only authorized users can upload files to the cloud. Fig.5 shows the snapshot where the registered data owner uploads a file in to the cloud. When the data owner uploads the file, first the file will be divided into 3 blocks and then the AES algorithm will be used to encrypt the file. The owner of the data can see the hash key generated for his file. By requesting the TPA the data owner can verify the integrity of the files which are stored in the cloud. The TPA acts as a mediator between the data owner and the cloud. The TPA doesn't store the actual copy of the file uploaded by the data owner. When the data owner wants to check the integrity of the file he will send a request to TPA, then the TPA sends a request to the cloud. The cloud will generate and send the proof to the Third Party Auditor and then the Third Party Auditor verifies whether the hash key of the files is same during upload and during the verification. If both the hash keys are same then the data owner will be notified that the data is not changed otherwise a mail will be sent to the registered email id that the data stored is altered as shown in Fig.6.

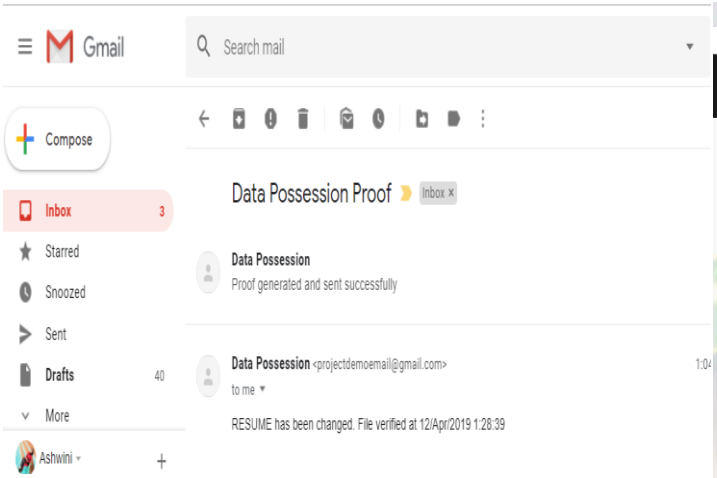

Fig 6: Notification to the data owner when the data is modified.

In cloud storage if the file needs to be modified it has to be downloaded, the update operation has to be performed and then the file is uploaded to the cloud. There is no option to update the content of the file in the cloud, so dynamic operation has been implemented. The main aim of dynamic operation is to allow data owner to modify the file in the cloud without downloading the file. Dynamic operation is implemented using Operation Research Table. The data owner has to update the data on the cloud and a request will be sent to the cloud as shown in Fig. 7. Fig. 8 shows that the cloud has to confirm the request of the owner of the data so that the hash keys of the file can be updated and after the acceptance from the cloud, the contents of the file and the hash keys will be updated.

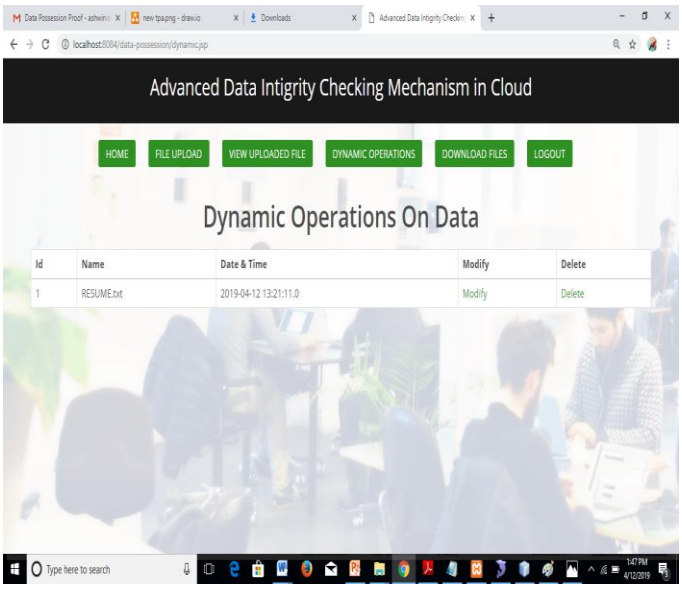

Fig 7: Dynamic operation on data.

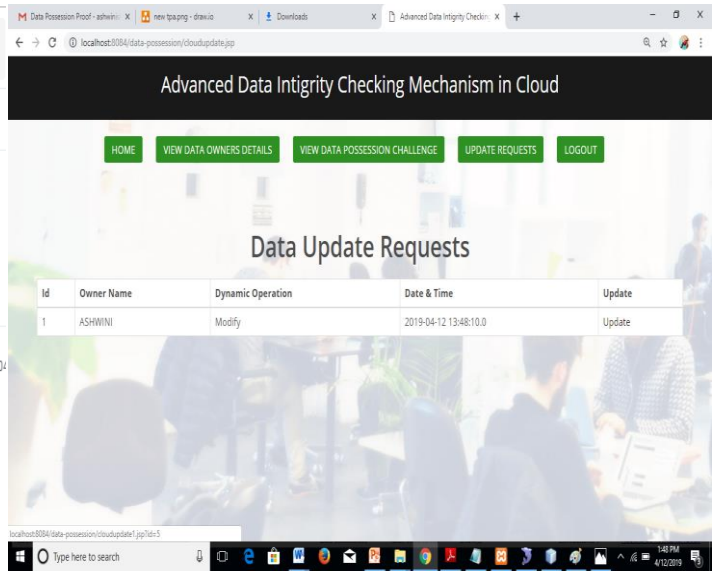

Fig 8: Dynamic operation request sent to the cloud.

IV. CONCLUSION

The users will place their data in the cloud and no longer retain their data locally. The main issue here is to detect the integrity of the stored data in the cloud. In this paper, we implemented a privacy preserving public auditing system for data storage security in cloud. TPA will perform auditing task without downloading the data copy of a cloud user, thus achieves privacy preserving. The user's data is encrypted first before uploading any data to the cloud, thus achieving the privacy of the data. The TPA will check the integrity of the data by verifying both the message digest. In the proposed system, TPA checks whether the data in the cloud is altered and later intimates the same to the cloud user. All the modules in the system are implemented to develop an effective auditing scheme. The dynamic operations are performed to the files using the ORT table, thus ensuring that the data owner need not download the file and then upload the updated file.

REFERENCES

[1] Shivarajkumar Hiremath, Sanjeev Kunte: A Novel Data Auditing Approach to Achieve Data Privacy and Data Integrity in Cloud Computing, 08 February 2018.
[2] Hao Yan, Jiguo Li, Jinguang Han, Yichen Zhang: A Novel Efficient Remote Data Possession Checking Protocol in Cloud Storage, VOL.12, NO. 1, January 2017.
[3] Sandhya Verma and G.S.Prajapati: Robustness and Security Enhancement of SHA with Modified Message Digest and Larger Bit Difference 19 March 2016.
[4] Preeti Sirohi, Amit Agarwal:Cloud Computing Data Storage Security framework relating to Data Integrity, Privacy and Trust,11 January 2016.
[5] S. Chakraborty, S. Singh and S. Thokchom:Third-party auditing for cloud service providers in multicloud environment,12 November 2018.
[6] Yindong Chen,Liping Li,Ziran Chen: An Approach to Verifying Data Integrity for Cloud Storage.
[7] Bambang Leo Handoko, Ridang Widuri and Haryadi Sarjono:Effect of Third Party Auditor and Quality of Service through Cloud Storage Security to Cloud User Trust,21 December 2017.
[8] Andrey N. Rukavitsyn, Konstantin A. Borisenko, Ivan I. Holod, Andrey V. Shorov: The Method of Ensuring Confidentiality and Integrity Data in Cloud Computing,07 July 2018.