International Journal of Security and Its Applications
Vol. 11, No. 7 (2017), pp.59-72
http://dx.doi.org/10.14257/ijsia.2017.11.7.05

Reliable and Enhanced Third Party Auditing in Cloud Server Data Storage

Nitin Nagar[1] and Ugrasen Suman[2]

[1] *IIPS, DAVV, Indore, INDIA*
[2] *SCSIT, DAVV, Indore, INDIA*
[1]*nitin28nagar@gmail.com,* [2] *ugrasen123@gmail.com*

Abstract

Cloud computing provides a service based environment for data storage and resource sharing that are available to user through internet with on-demand basis. Thus, users can access their data across any geographical location at any time. Cloud environment also provides better scalability, flexibility, high performance, availability and less storage cost as compared to other physical storage of data. Maintaining data integrity and security in cloud environment is difficult especially, when the stored data is not completely reliable and trustworthy. However, the security of stored data is the major concerned for organizations and individual user to adopt cloud based environment. In this paper, we have proposed and enhanced the functionalities of third party auditor server to protect the availability and integrity of outsourced data in a cloud environment. The proposed approach uses the functionality such as, public verifiability, metadata generation, data dynamics, storage access point, encryption and decryption of data through RSA algorithm and IP range in case of private cloud. The proposed work also focuses on a solution to reliability, availability and integrity of data that are the major issues in the cloud adoption.

Keywords*: TPA, Reliability, Availability, Integrity, MTTF, MTTR and MTBF*

1. Introduction

Cloud computing provides Internet based services (i.e., SaaS, PaaS and IaaS), computing, and data storage to individual users as well as organization such as IT, finance, healthcare and government. The underlying computing infrastructure is applied exclusively when it is required. Cloud computing emphasizes on various service oriented architectures with the help of tools, which reduces the infrastructure cost of users. There are varying range of cloud tools and technology such as, Eucalyptus, OpenNebula, Nimbus and OpenStack etc. [1]. In a cloud computing environment, resources are mutual among servers, users or individuals. As a consequence, files or stored data in the cloud are openly accessible to all. Therefore, data or files of an individual are vulnerable to attack [2]. It is very easy for an attacker to access, use, and destroy the original data from the cloud data server. Attacker may also break off the transmission of data. The other problem with the cloud environment is that an individual is not aware of exact location of data. Therefore, it is extremely important for the cloud data to be safe at cloud storage as well as data transmission site [3] [4].

In a cloud computing environment, data and the application are controlled by the CSP. This leads to a usual concern about data protection from internal and external threats. Usually, in a cloud computing environment, data storage and computation are performed in a single datacenter that generally utilizes the virtualization technology [22]. The online data implicit storage security is more useful in a cloud environment that offers implicit storage security architecture for storing data. It also uses some common partitioning mechanisms [5]. Thus, data partitioning scheme are proposed for online data storage that

ISSN: 1738-9976 IJSIA

involves the finite field polynomial root. The partitioned data are kept on cloud servers that are chosen randomly along the network and these partitions are retained in order to recreate the master copy of the data. Split data is accessible to authorized user. SSL authentication protocol (SAP) is not appropriate to a cloud environment, it is very complex and difficult to deploy. As an alternative of SAP, a new authentication protocol based on identity is proposed with signature and encryption schemes to achieve secure communication [6]. The cloud data server must ensure the security and integrity of stored data. A novel and homogeneous structure are introduced to provide security to different cloud models. To achieve data storage security, BLS (Boneh–Lynn–Shacham) algorithm is introduced for signing the data blocks before outsourcing data into the cloud. BLS algorithm is efficient and reliable than the other proposed algorithms [7].

Data storage in the cloud is not fully trustworthy because the user did not have a local copy of data stored in the cloud. To address these issues, a new system using the data reading protocol and algorithm to verify the data integrity is proposed. Service providers help the users to verify the data security by using the proposed effective automatic data reading algorithm [8]. An effective and secure storage protocol is introduced that provides the data storage confidentiality and integrity. This protocol is uses elliptic curve cryptography and sobol sequence is used to confirm the data integrity [9]. The proposed mechanism allows users to auditing the cloud data storage and this auditing result utilized homomorphic token with reed-solomon algorithm to ensure the correctness in code. The proposed design is extended to support block-level data dynamic operations. If the cloud user is not able to possess information, time and utility then the users can assign their job to an evaluator, i.e., TPA for the auditing process in a safe manner [10]. The system consist the set of master and slave servers. There is no direct communication link between clients and slave servers in the proposed model. The master server is responsible to process the client's requests and at slave server chunking operation is carried out to store copies of files in order to provide data backup for file recovery in future [11].

Cloud computing security is the leading challenge in the field of research. The rest of the paper is organized as follows. The Section 2 states encryption and RSA algorithm along with other security, access and storage algorithms for ensuring the data integrity in cloud. In Section 3, we states performance issues of the propose work to improvement in cloud computing security. In Section 4, we state the conclusion.

2. Encryption Algorithms

Security and privacy of data are prime concerns for the organization to adopt cloud based environment [13]. Triple-DES encryption is used to information block particularly useful for digital signatures and RSA based authentication is useful for good communications links. United Nations Environment (UNE) relies on cloud privacy and protection of sensitive data in a proposed memory database encryption technology [14]. The encryption is applied to the database in the cloud, which uses a new asymmetric encryption system [15] [12]. Other issues related to cloud computing such as penetration, service availability, data integrity are also major concern. Several clouds are useful to assured data integrity or cloud can be used to store information in the database [16]. Three layer data protection technique is proposed; in which first layer is username, password and M-pin authentication server for the cloud user's authenticity [17]. The second layer security and data encryption through RSA algorithm ensure confidentiality of the user's data. The third layer is useful for checking the integrity of stored data, which is performed through different functionalities of TPA.

2.1. Security Algorithm Using TPA

Third party auditor can be defined as a server for system or an environment in a cloud model to track file throughout its entire lifecycle. In our proposed TPA server, it collects

all the information about the file for auditing and control on data. A security audit can be defined as a systematic evaluation of CSP security by measuring how well the CSP conforms to the set of established criteria. It is also important for cloud service user to ensure that the integrity and confidentiality of their data is protected by the CSP. Poor audit-ability means that the system has poorly-maintained data and systems not efficient perform auditing processes within the cloud, therefore, good audit-ability should be maintained to get the better security within the cloud. TPA server only fulfills the request from registered users with valid IP address; otherwise this request goes to un-trusted server [12]. The un-trusted sever also sends a warning mail to actual user through CSP for unauthorized access. Figure 1 shows the flow of secure data transmission and cloud storage through TPA.

2.1.1. Architectural View: The architectural view of secure cloud storage through TPA is shown in Figure 1. The architecture includes additional functionalities to enhance the auditing capabilities of TPA server. These functionalities are public verifiability, registered IP address verifier, encryption and decryption of data through RSA algorithm, metadata generation, data dynamics, and storage access point. These functionalities are only available for authorized user via a CSP. Authorized user enters into CSP based application with proper authentication verification. CSP is connected with TPA. TPA ensures safety of data transmission and data storage. It also maintains the integrity with its functionalities. The various functionalities of TPA are discussed in the subsequent subsections.

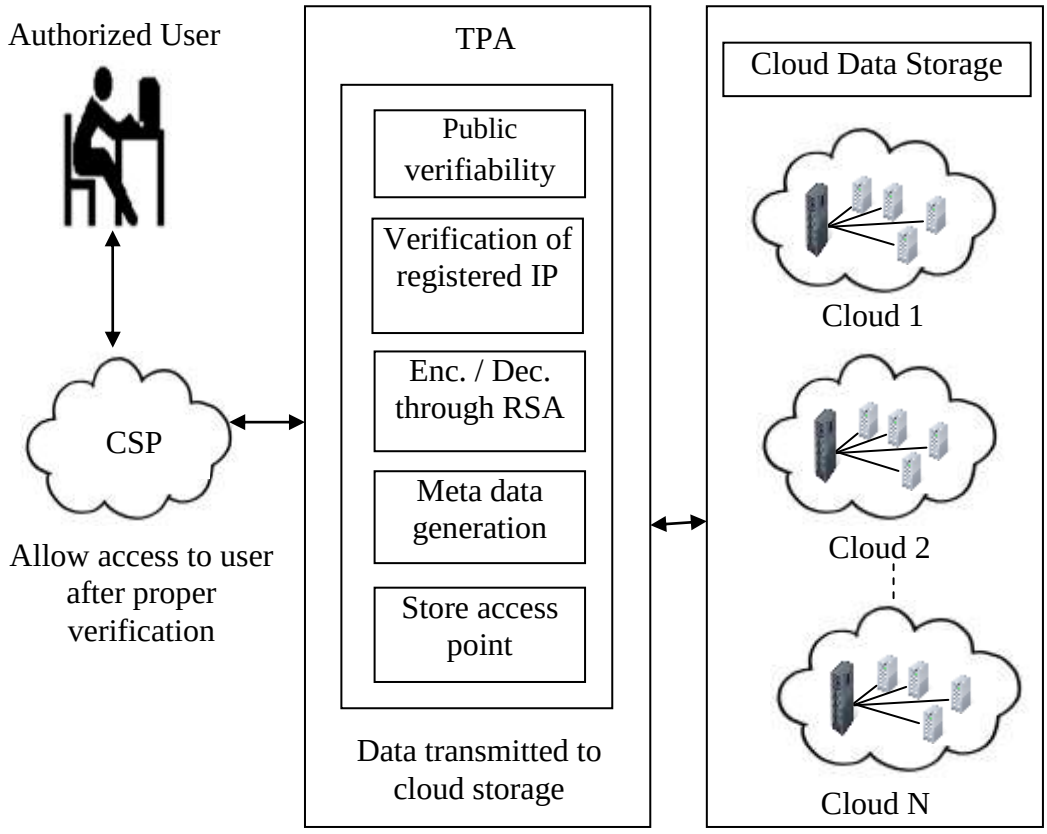

Figure 1. Architectural Representation of Cloud Storage through TPA

2.1.1.1. Public Verifiability: Public verifiability allows anyone apart from user (data owner) to challenge the cloud server for correctness of data storage. Users are able to hand over the auditing of the service performance to an independent TPA. In cloud, users themselves are unreliable or cannot afford the overhead of performing frequent integrity checks. Thus, for practical use, it seems more rational to adopt the verification protocol with public verifiability.

2.1.1.2. Registered IP Address Verifier: Registered IP address verifier uses a protocol through which authorized IP address is verified. This IP address is provided by CSP at the time of initial registration of user. At the same time, user stores identity and other information through CSP and TPA. TPA provides additional layer to built users' information more strong via metadata generation, and encryption and decryption through RSA algorithm.

2.1.1.3. Metadata Key Generation: Suppose verifier wants to store the file F. Let this file consist of n file blocks. TPA initially preprocesses the file and creates metadata to be appended to the file. Each n block consists of b bits of data. Each of metadata from data blocks mi is encrypted by using RSA algorithm to form a new modified metadata. The encryption method can improvised to provide strong protection to store data. All the metadata bit blocks that are concatenated to each other. This concatenated metadata appended to file F before storing it in the cloud server.

2.1.1.4. Encryption and Decryption through RSA: The encryption and decryption through RSA functionality can improvise to provide strong protection while storing the data into cloud. All the metadata bits blocks are concatenated to each other and encrypted through RSA algorithm. This encrypted and concatenated metadata is appended to file before storing it into the cloud server. The reverse process is applied for decryption of file when user wants to access the file.

2.1.1.5 Data Dynamics: In data dynamics, user stores data on cloud server. User can dynamically append, delete, and update stored data. These data dynamics are performed at block level of TPA server with privacy preservation of user's data. Operations performed at block level are block append, block deletion, and block update. These operations are discussed in the following subsections.

Block Appends Operation: In block append operation, suppose 40 GB storage volumes initially are allocated by CSP for user's data storage at block level. These allocated volumes are compared with storage measurement scheme and verified for its correctness through our proposed work by measuring the entire cloud server. If it is confirmed that the allocated volume does not have any data in cloud server then the integrity of data has to be considered strongest one.

Volume (V) can be defined as the volume on the server, which is provided by CSP to user. Data (D) can be defined as the appended data at the later by user. Here, initially we have assumed that $D \in V$. Newly added data (A) is defined as a data, which is added to cloud server after comparing the storage volume before and after. The following equations can be used to explain the storage volume taken into the account:

$$\sum_{i=0}^{n} (V_i - D_i)^{V-1} + \sum_{i=0}^{n} D_i \,^{V-1}$$

$$\sum_{i=0}^{n} V_i \; + \sum_{i=0}^{n} A_i \,^{A-1}$$

Block deletion operation: In block deletion operation, deletion can be performed through checking the availability of data from existing data on cloud server. The equations for checking the availability is defined as follows:

$$\sum_{i=0}^{n} (V_i - D_i)^{V-1} \neq \sum_{i=0}^{n} \emptyset^{[(V\emptyset_i)^{\emptyset}i]}$$

The deletion of data can be expressed by the similar equation. If no deletion operation is performed then it can be represented by the following equation:

$$\sum_{i=0}^{n} \emptyset^{[(V\emptyset_i)^{\emptyset}i]} \neq \emptyset$$

Where, $\emptyset$ = empty volume in cloud server or shows no data initially.

Block update operation: Update operation is performed after completion of needed action taken by user. For new data update, each data block should be updated automatically from already existing volumes to new updated data. If data block considered as an array of formation then the result is stored through the equation $V = (V \pm D_i)$. It depends upon the operation that the user performs on data.

2.1.1.6 Storage Access Point: Storage access point is used to maintain the integrity of user's data. TPA manages every specific time while updating the database from CSP. Therefore, one restore access point is created automatically in every update within cloud database for the future purpose or to recover data from previous stored level. If user performs any operations such as, append, delete, and update on cloud database then all stored access points are automatically stored in cloud database within a specific time.

The algorithm for storing the access point, i.e., STORAGE_ACCESS_POINT is performed to check and store user's operations at a specific time, which is discussed in Algorithm 1.

Algorithm 1: STORAGE_ACCESS_POINT (*D, V,* $\emptyset$, n, S). D is the data stored by user through CSP and TPA. *V* is the initial volume provided by CSP to user. The n is the collection of inserted volume in CSP. $\emptyset$ is used to define the emptiness in the server. S is the data storage servers.
Input: *D, V,* $\emptyset$, n.

Output: Access point, total number of servers, data availability, access points of various operations at specific time

1. Begin

2. If (S≠ V & S = ∅) / If server doesn't have any volume */*

3. No restored point is resulted in search

4. else access point is found and restored

4. then S = V + D

5. for (S = 0; S <= n; S++)

6. S++

7. Return S / Recursive call for each update */*

In TPA, server access point is initially selected as $V = (v_i, v_{i+1}, v_{i+2\ldots} v_{i+n}) \forall V$. The total allocation volume is z in GB from CSP. The algorithm first compares V and $\emptyset$ from the storage server. If server doesn't have any volume and shows the emptiness then the no restore point will result into search. Otherwise, it identifies an access point and restored it. STORAGE_ACCESS_POINT algorithm monitors accessibility of operations that is performed in a specific time. These operations can be defined with the DATA_DYNAMICS_OPERATIONS, which is described as follows:

Algorithm 2: DATA_DYNAMICS_OPERATIONS (*D, V,* ∅, n, b_i). D is the stored data on which user performs data dynamics at later time. V is the initial volume provided by CSP to user. The n is the collection of inserted volume in CSP. ∅ is used to define the emptiness in volume. b_i is defined as a deleted data on storage servers.
Input: *D, V,* ∅, n, b_i.

Output: Availability of total number of servers, data availability, access points of various operations and data dynamics.

1. *Begin*
2. *If* $(D \notin V)$ */* If V is not having any data */*
3. $V = (V_i - D_i)^{V-1}$ */* For total number of server */*
4. *else* $V = (V + D_i)^{V-1}$
5. *If* $(D \notin V) \neq (D \in V)$ */* Compare values */*
6. $[\sum_{i=0}^{n} [(V_i - D_i)^{V-1}] + [\sum_{i=0}^{n} [D_i]^{V-1}]$
7. *else* $\sum_{i=0}^{n} \emptyset^{[(V\emptyset_i)^{\emptyset_i}]} \neq \emptyset$

/ Cloud storage server has no data */*

8. *If* $([\sum_{i=0}^{n} [(V_i - D_i)^{V-1}] \neq \left[\sum_{i=0}^{n} \emptyset^{[(V\emptyset_i)^{\emptyset_i}]}\right])$

   / Deletion of operation */*
9. $V = \emptyset$ */* No data */*
10. *else if* $\left[\sum_{i=0}^{n} \emptyset^{[(V\emptyset_i)^{\emptyset_i}]} \neq \emptyset\right]$

    / ∅ means empty volume and value */*

11. V = remove (b_i) / Data is deleted, b is assign to deletion */*

12. Return (V)

13. Stop

TPA selects multiple servers at the time of data auditing to identify the value or weight on data. We assume that $D \in V$ means that data D is included in sever volume *V*. If condition $D \notin V$ is occurred then the server has $(V_i - D_i)^{V-1}$ volume in its overall storage excluding any addition data in its volume. Otherwise, *V* has additional amount of data from storage area of user, which is defined as $V = (V \pm D_i)$ for entire cloud server. If comparison $(D \notin V) \neq (D \in V)$ is performed then TPA calculates present data at any particular position in cloud server. It is measured from line 5 and 6 of algorithm. Server doesn't contain any data determined by line 7 in algorithm. If the server is not equal to empty volume then it means that it has data to be deleted according to user's update

otherwise it can be expressed by line 8 in algorithm. STORAGE_ACCESS_POINT and DATA_DYNAMICS_OPERATIONS algorithm are used to maintain the cloud storage data integrity. When specific attack occurs and the size of entire cloud data is changed by modification, deletion and append operations. TPA can rectify server access point to previously updated data of user from cloud data storage within a specific time.

3. Process Flow of Proposed Architecture

The process flow of TPA based reliable data transmission and storage is shown in Figure 2. This process requires user's data as input for data dynamics according to previously stored user identification by CSP. Initially, authorized user can access the cloud based application through CSP. In case of unauthorized access, warning email is sent by CSP to authorized user. As a result, it provides reliable and secure data storage capabilities. After complete verification and security checks, user is able to perform data dynamics. CSP can provide data dynamics and other functionalities with proper interaction of TPA. TPA includes various functionalities for secure and reliable data transmission as well as storage. These functionalities are public verifiability, encryption and decryption through RSA algorithm, metadata generation, and storage access point.

4. Performance Issues in Proposed Work

The reliability refers to the ability of a computer products (hardware or software) consistently perform according to its specifications. In theory, a reliable product is totally free of technical errors. In practice, providers commonly express product reliability as a percentage. Cloud reliability measures can be used to improve cloud application through supporting quantitative evaluation of applications, tracking development status, conducting upgrades and perform maintenance activities. The most common abnormal behavior of unreliable storage is that the cloud service providers may discard part of the user's update data, which is hard to be checked. Fork-Join-Causal-Consistency and eventual consistency architecture is introduce for effectively resist problems such as discarding and it can support the implementation of other safety protections in the reliable cloud storage environment such as Amazon S3 [18].

Cloud storage data availability means when accidents such as hard disk damage, IDC fire, power failures, code malfunctioning, system error, VM unavailability and network failures occur, the extent that user's information can be used or reclaimed. Storing the data over trans-border servers is a serious concern of users because the cloud service provider is governed by the local laws and, therefore, the cloud users should be aware of those laws. Moreover, the cloud service provider should ensure the data security, particularly data confidentiality and integrity. The cloud provider should share all such concerns with the user and build trust relationship. Locating data can help users to increase their reliability on the cloud. Cloud storage provides the transparent storage service for users, which can decrease the complexity of cloud, but it also decreases the control ability on data storage of users. The proofs of geographic replication and succeeded in locating the data stored in Amazon cloud is studied in the proposed work [19].

Data integrity in the cloud environment means preserving information integrity. The data should not be lost or modified by unauthorized users. Data integrity is the basis to provide cloud computing service such as SaaS, PaaS, and IaaS. Besides data storage of large-scaled data, cloud computing environment usually provides data processing service.

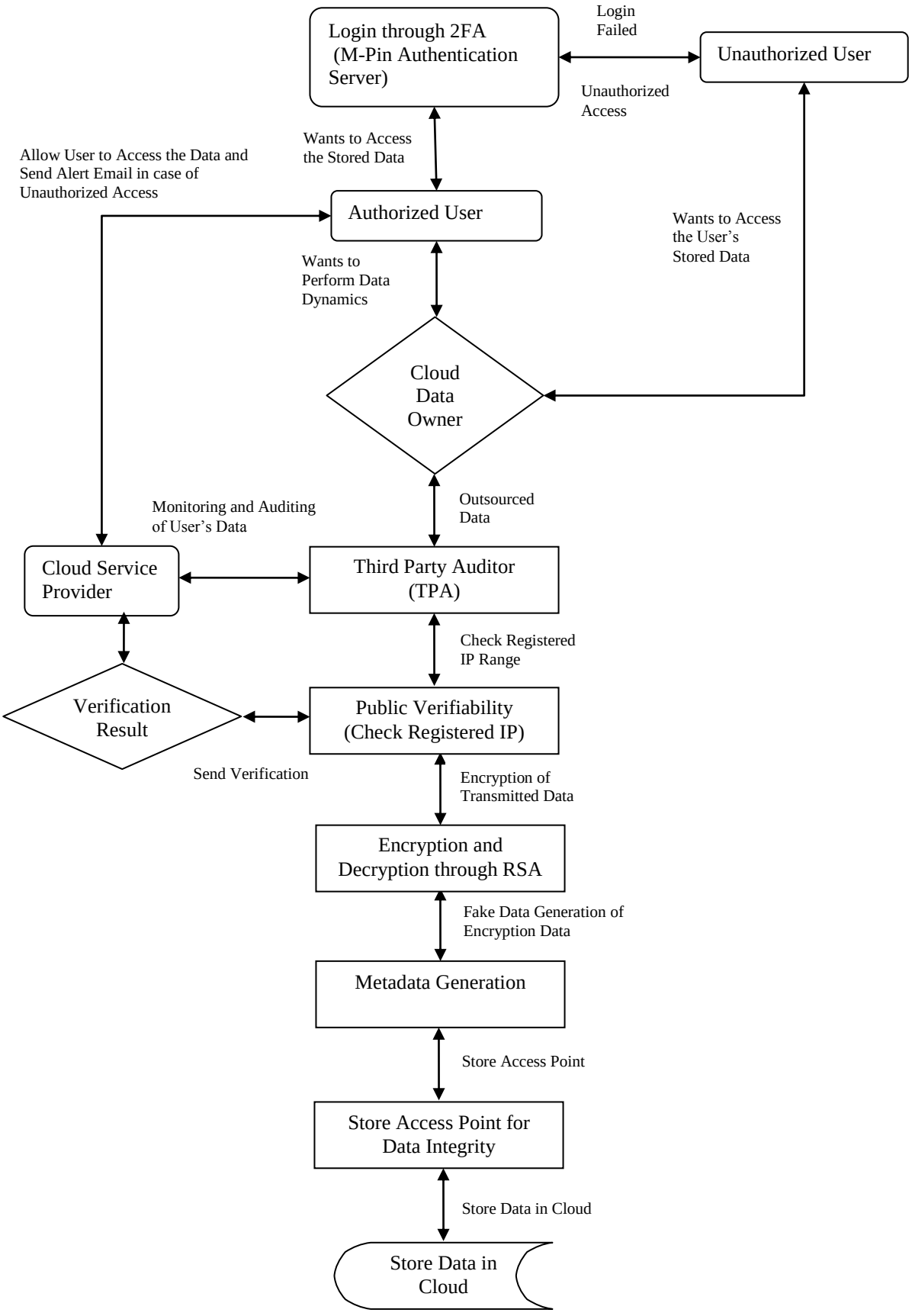

Figure 2. Secure Data Transmission on Cloud Storage

The HAIL system uses POR mechanism to check the storage of data in different clouds, and it can ensure the redundancy of different copies and realize the availability and integrity [20]. Trusted platform module (TPM) is proposed for checking the data integrity remotely [21].

4.1. Cloud Data Storage Reliability Metrics

Reliability metrics in cloud data storage are derived from failure occurrence of expression and data. The common reliability metrics include Probability of Failure on Demand (POFOD), Rate of Occurrence of Failures (ROCOF), Mean Time to Failure (MTTF), availability, Mean Time to Repair (MTTR), and Mean Time between Failures (MTBF). In our proposed work, we have included four metrics such as, availability, MTTF, MTTR, and MTBF. Cloud Storage Server (CSS) availability is the ratio of expected values in uptime of server to the aggregate of the expected values of up and down time of server. This can be represented through following equation.

$$CSS_{Avail} = \frac{\delta\,[\,Server\_Uptime]}{\delta\,[\,Server\_Uptime] + \delta\,[\,Server\_Downtime]}$$

Let n be the total number of server, V is the volume and D is the data. The server composition set can be represented as,

$$CSS_{comp} = [\,(V_i - D_i)^{V-1}]$$

The availability for 'n' participating storage services is computed using the following equation,

$$[\,\sum_{i=0}^{n}\,[(V_i - D_i)^{V-1}]\,\sum_{i=0}^{n}\,CSS_{Avail} = [\frac{\delta\,[\,Server\_Uptime]}{\delta\,[\,Server\_Uptime] + \delta\,[\,Server\_Downtime]}]$$

We define the cloud server monitoring function as M(t), which represents failure occurrence of random process at time t. M(t) can be represented as,

$$M\,(t) = \begin{cases} 1, \text{ Server continual function at the time } t \\ 0, \text{ otherwise.} \end{cases}$$

Therefore, the cloud server availability $CSS_{Avail}\,(t)$ at time t>0 is represented as,

$$CSS_{Avail}\,(t) = Prob[M\,(t) = 1] = \delta\,[M\,(t)]$$

Average availability must be defined on the interval of real line. If we consider arbitrary constant c>0, then average service availability is represented as,

$$Avg\,[CSS_{Avail}]_{\,c} = \frac{1}{c}\int_0^c\,CSS_{Avail}(t)\,dt$$

Constant state (i.e., limited) availability is represented as,

$$CSS_{Avail} = lim_{t\to\infty}\,CSS_{Avail}\,(t)$$

For example, suppose MTBF of 2.30 years and MTTR of 2 hour then MTBF in hours can be calculate as,

$$2.30*365*24 = 20148$$

The availability can be formulized as,

Availability = MTBF/ (MTBF + MTTR)
= 20148/20148+2
= 20148/20150
= 99.99007% and
Unavailability = 0.00993%

The availability can be measured through availability class, which helps to find the availabilities of system types. Various availability classes of cloud data storage are shown in Table 1. Further, these availability measures are used to evaluate various access points of storage server in cloud such as, login, registration, access stored data etc. these access points can be measured through CS_MTBF, CS_MTTR, availability and unavailability. Different reliability evaluation for cloud storage server is shown in Table 2.

Table 1. Cloud Storage Availability Class

Availability classes of cloud storage	Availability (%)	Unavailability (min/year)	System type
1	90.0	92560	Unmanaged
2	99.0	9256	Managed
3	99.9	926	Well -managed
4	99.99	92.6	Fault-tolerance
5	99.999	9.3	Highly available
6	99.9999	0.93	Very high available
7	99.9999	0.00993	Ultra available

Table 2. Reliability Evaluation for Cloud Storage Server

Access points	CS_MTBF (min/year)	CS_MTTR (min/year)	Availability (%)	Unavailability (%)
Login	17520	17522	99.98859	0.01141
Registration	18978	18980	99.98946	0.01054
View	20345	20347	99.99017	0.00983
Hacker info	20909	20911	99.99044	0.00956
Access stored data	78900	78902	99.99747	0.00253
Perform data dynamics	17890	17892	99.98882	0.01118
RSA encryption	78967	78969	99.99747	0.00253
RSA decryption	77778	77780	99.99743	0.00257

Add IP addresses	16789	16791	99.98809	0.01191
Remove IP addresses	16898	16900	99.98817	0.01183
Public verifiability	17890	17892	99.98882	0.01118
Metadata generation	56757	56759	99.99648	0.00352
Store access point	34566	34568	99.99421	0.00579

5. Security Evolution of Cloud Storage Server

Cloud security testing is a process to determine that system protects the cloud storage data and maintains the integrity of system. We include four basic security concepts that need to be covered by security testing such as integrity, authentication, availability and authorization. Integrity measure the system's ability to with stand attacks to its security. In order to measure integrity two additional parameters threat and security are needed. Threat is the probability that an attack of certain type will happen over a period of time. Security is the probability that an attack of certain type will be removed over a period of time. The integrity is formulized as Integrity = $\sum$ [(1 - threat) X (1 - security)]. Where, threat and security are summed over each category of attacks. Figure 3 shows the organization of data security in cloud storage. The equipment used to obtain the measurements are an Intel (R) Pentium (R) Core 2 Duo CPU 2.4 GHz (VT enables machine), 4 GB of RAM, and Ubuntu 12.04 (LTS) runs on Linux. To enhance the security of system, we use JAVA language for development of proposed algorithms. We considered availability, integrity, security, MTTF, MTTR, MTBF, network I/O and network connection for the measurement of performance. Table 3 shows the reliability evaluation for cloud storage server.

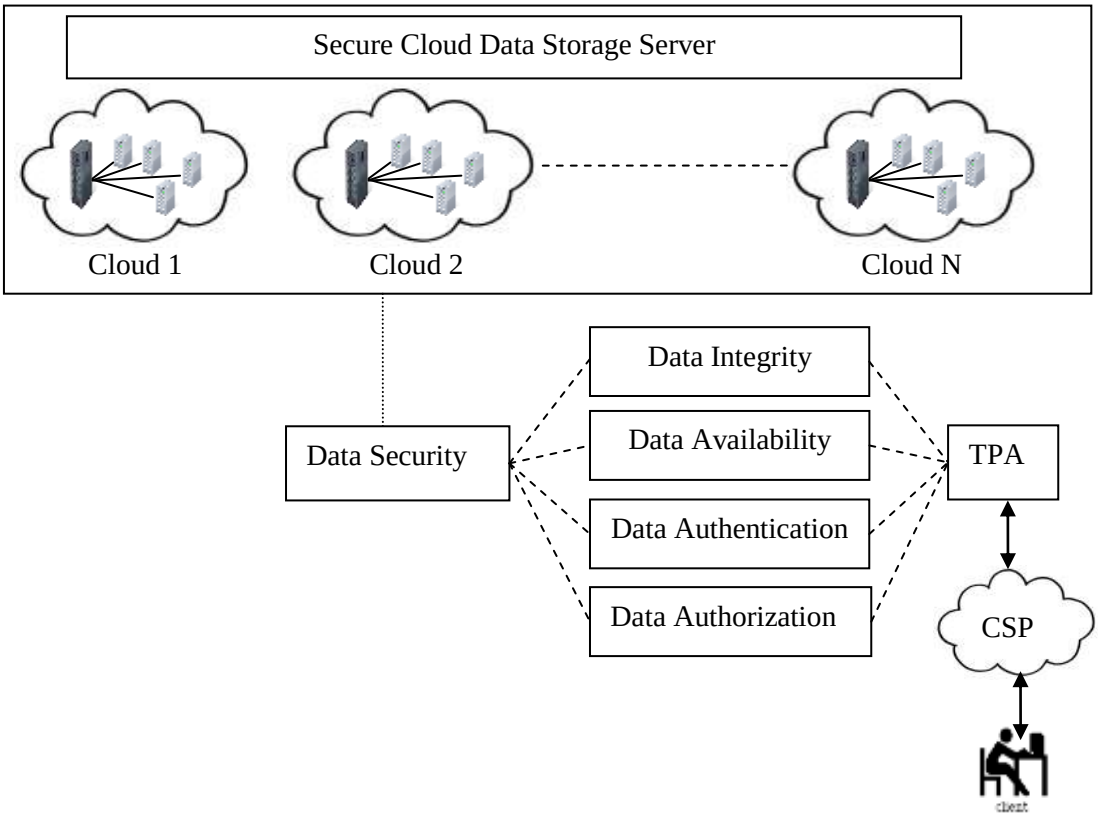

Figure 3. Organization of Data Security in Cloud Storage

6. Conclusion

Cloud server stored data security introduces new challenges that have to be considered by cloud service providers and TPA. The proposed system is suitable for providing reliability, integrity, availability and protection of user's stored data. The proposed system supports data insertion, modification and deletion at the block level in encrypted manner, and also supports public verifiability and metadata generation. In the current work, data level dynamics can be supported by using block level dynamics. The system is proved to be secure against an un-trusted server. It is also private against third party auditors. Analysis and experimental results demonstrate that the proposed system has good efficiency in the aspects of communication, computation and storage data availability. Whenever a piece of data is modified, the corresponding blocks are updated through access point algorithm. The enhancement of TPA functionality in cloud environment will not ensure that the data will more secure although it gives a new perimeter of security level in cloud computing.

References

[1] N. Nagar and U. Suman, "Architectural Comparison and Implementation of Cloud Tools and Technologies", Proc. of 4 th IEEE International Conference on Electronics Computer Technology (ICECT'12), Kanyakumary, **(2014)**, pp. 978-1.
[2] R. Bhadauria, R. Chaki, N. Chaki and S. Sanyal, A survey on security issues in cloud computing, **(2011)**.
[3] M. A Vouk, "Cloud computing–issues, research and implementations", CIT. Journal of Computing and Information Technology, vol. 16, no. 4, **(2008)**, pp. 235-246.
[4] Y. Hu, J. Wong, G. Iszlai and M. Litoiu, "Resource provisioning for cloud computing", Proceedings of the 2009 Conference of the Center for Advanced Studies on Collaborative Research, **(2009)**, pp. 101-111.
[5] A. Parakh and S. Kak, "Online data storage using implicit security", Information Sciences, vol. 179, no. 19, **(2009)**, pp. 3323-3331.
[6] H. Li, Y. Dai, L. Tian and H. Yang, "Identity-based authentication for cloud computing", Cloud computing, **(2009)**, pp. 157-166.
[7] S. Balakrishnan and G. Saranya, "Introducing Effective Third Party Auditing (TPA) for Data Storage Security in Cloud", International Journal of computer science and Technology, vol. 2, no. 2, **(2011)**, pp. 397-400.
[8] C. Dinesh, "Data Integrity and Dynamic Storage Way in Cloud Computing", arXiv preprint arXiv:1111.2418, **(2011)**.
[9] P. Syam Kumar and R. Subramanian, "An efficient and secure protocol for ensuring data storage security in Cloud Computing", IJCSI International Journal of Computer Science Issues, vol. 8, no. 6, **(2011)**.
[10] C. Wang, Q. Wang, K. Ren and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing", INFOCOM, Proceedings IEEE, **(2010)**, pp. 1-9.
[11] P. M. Deshmukh, A. S. Gughane, P. L. Hasija and S. P. Katpale, Maintaining File Storage Security in Cloud Computing, **(2012)**.
[12] N. Nagar, U. Suman, "A Secure Cloud Environment through Location Signature and HTML5 WebDB", 3rd International Conference on Advances in Cloud Computing Pune (MS), **(2014)**, pp. 31-35. CSI.
[13] A. Kaur and M. Bhardwaj, "Hybrid encryption for cloud database security", International Journal of Engineering Science & Advanced Technology, vol. 2, no. 3, **(2012)**, pp. 737-741.
[14] F. Pagano and D. Pagano, "Using in-memory encrypted databases on the cloud", Securing Services on the Cloud (IWSSC), 2011 1st International Workshop, **(2011)**, pp. 30-37.
[15] K. Huang and R. Tso, "A commutative encryption scheme based on ElGamal encryption", Information Security and Intelligence Control (ISIC), 2012 International Conference, **(2012)**, pp. 156-159.
[16] M. AlZain, B. Soh and E. Pardede, "MCDB: Using Multi-clouds to Ensure Security in Cloud Computing", Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference, **(2011)**, pp. 784-791.
[17] N. Nagar and U. Suman, "Two Factor Authentication using M-pin Server for Secure Cloud Computing Environment", International Journal of Cloud Applications and Computing, vol. 4, no. 4, **(2014)**, pp. 42-54.
[18] K. Benson, R. Dowsley and H. Shacham, "Do you know where your cloud files are?", Proceedings of the 3rd ACM workshop on Cloud computing security workshop, **(2011)**, pp. 73-82.
[19] P. Mahajan, S. Setty, S. Lee, A. Clement, L. Alvisi, M. Dahlin and M. Walfish, "Depot: Cloud storage with minimal trust", ACM Transactions on Computer Systems (TOCS), vol. 29, no. 4, **(2011)**, p. 12.

[20] K. D. Bowers, A. Juels and A. Oprea, "HAIL: a high-availability and integrity layer for cloud storage", Proceedings of the 16th ACM conference on Computer and communications security, **(2009)**, pp. 187-198).
[21] J. Schiffman, T. Moyer, H. Vijayakumar, T. Jaeger and P. McDaniel, "Seeding clouds with trust anchors", Proceedings of the 2010 ACM workshop on Cloud computing security workshop, **(2010),** pp. 43-46.
[22] N. Nagar, U. Suman, "Analyzing Virtualization and Design a Secure Cloud Environment to Prevent from XSS Attack", International Journal of Cloud Applications and Computing (IJCAC), IGI Global, vol.6, no.1, **(2016)**, pp. 1-14. DOI: 10.4018/IJCAC.2016010101.