Modeling Software Architecture Design on Data Storage Security in Cloud Computing Environments

Srinath Venkatesan
New York university; United States, sv778@nyu.edu

Julio César Moyano Alulema
Escuela Superior Politécnica de Chimborazo. Grupo de Investigación & Vinculación AUTOPRO, Riobamba, Ecuador, j_moyano@espoch.edu.ec

Ángel Geovanny Guamán Lozano
Escuela Superior Politécnica de Chimborazo. Grupo de Investigación & Vinculación AUTOPRO, Riobamba, Ecuador, a_guaman@espoch.edu.ec

Jhonny Marcelo Orozco Ramos
Escuela Superior Politécnica de Chimborazo. Grupo de Investigación & Vinculación AUTOPRO, Riobamba, Ecuador, jhonny.orozco@espoch.edu.ec

Abstract

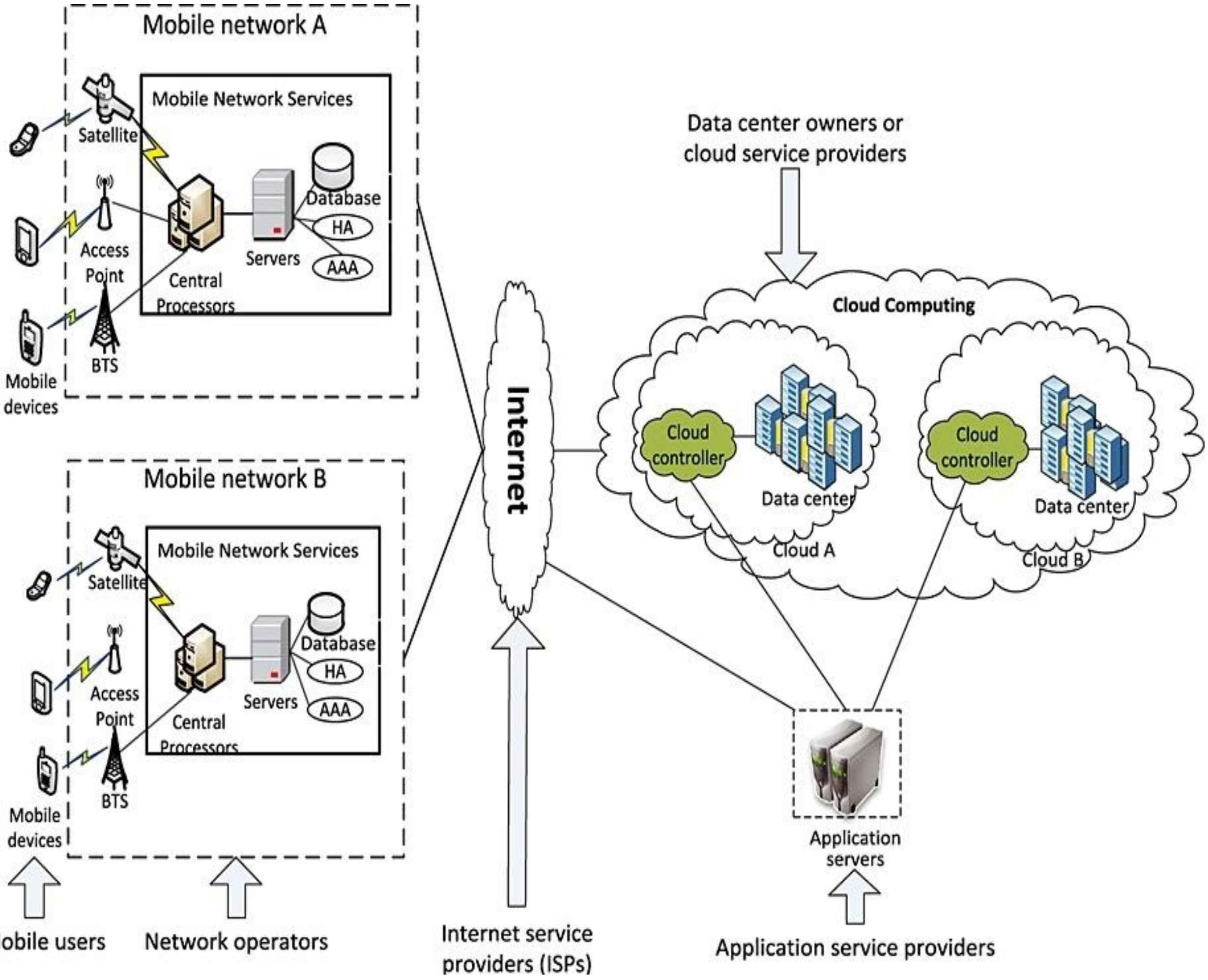
Cloud-based computation is known as the source architecture of the upcoming generation of IT enterprise. In context to up-coming trade solutions, the Information Technology sections are established under logical, personnel, and physical control, it transfers application software and large database to appropriate data centers, where security and management of database with services are not trustworthy fully. So this process may face many challenges towards society and organizations and that not been well understood over a while duration. This becomes one of the major challenges days today. So in this research, it focuses on security-based data storage using cloud, which plays one of the important aspects bases on qualities of services. To assure user data correctness in the cloud system, a flexible and effective distributed technique with two different salient features was examined by utilizing the token called homomorphic with erasure-coded data for distributed verification, based on this technique it achieved error data localization and integration of storage correctness. Also, it identifies server misbehaving, efficient, and security-based dynamic operations on data blocking such as data append, delete, and update methods. Performance analysis and security show the proposed method is more effective resilient and efficient against Byzantine failure, even server colluding attacks and malicious data modification attacks.

Keywords: *Cloud security, Architecture design, Data Storage, Homomorphic token, dynamic operation.*

1. INTRODUCTION

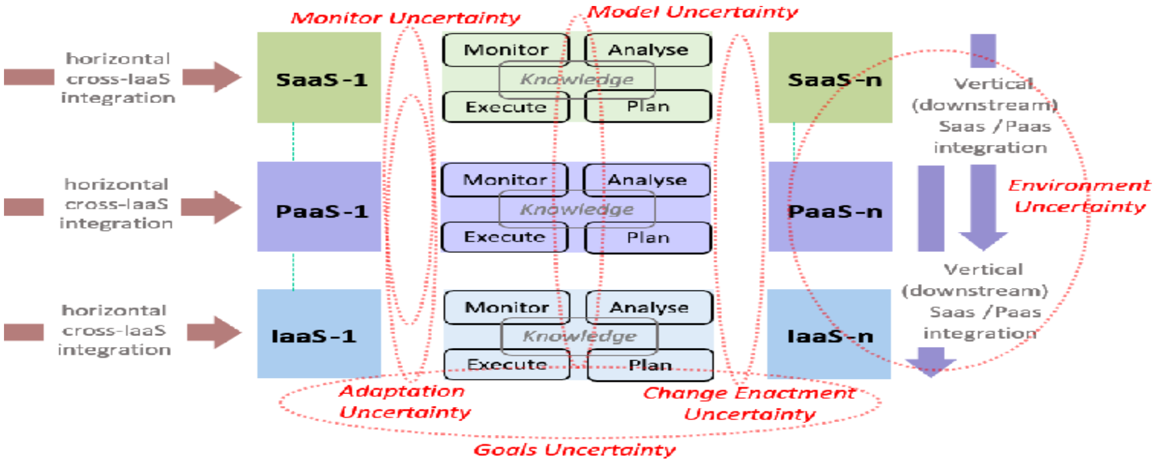
Many trades are established with cloud computing technology in markets, which follow digital and internet-based strategies. By increasing bandwidth range and flexible network connections, it makes it possible for the users to scribe with high-quality services, convenience to transfer information such companies are (EC2) and (S3) [1]. While these online services with internet-facilities provide a larger quantity of data storage area and computing customer resources, cloud- based platform computing moved, however, it is responsible for eliminating data maintenance for the local device at the same interval of time [2, 3]. Many companies are dependent on cloud computing servers for data integrity and availability, due to which new challenges come up across the security of threats for many reasons [4, 5]. Newly cryptographic measures for security reasons for the database cannot be directly adopted due to control losses by user's information under the cloud-based system. For that, correct data storage verifications need to be done, without the knowledge of complete information explicitly. Now consider, many users have stored a different variety of information in the cloud system and they have demanded continuous long term assurance about the safety of information and the problem rises for long term data security become more challengeable. The storage of information by the user needs to update frequently for inserting, deleting, appending, modification of data [6, 7]. To reduce integrity threats information is stored in multiple locations. The importance of data integrity is highlighted by the following works [8, 9, 10]. These techniques ensure the correctness and security of data by focusing on a single server and dynamic data operations.

Fig. 1. Business Oriented Cloud Computing Architecture



Based on the theoretic-control framework a dynamic manageable model with feedback implementation was investigated [11, 12]. Always requirement needs to be implemented in a system that allows the self-adaptation process to communicate directly with the environment through a good decision-making system and understand the problems to overcome security issues [13, 14, 15]. Based on environmental perception, it is very important to adjust the behaviour of systems; this adjustment can be done through the self-adaptive process [16, 17]. To build a good security-based process we need to develop an application architecture model that fulfills all critical aspects and challenges in a distributed environment [18, 19]. Cloud is moving towards a distributed environment with security-based services [20, 21]. Software engineers are working hard to develop a more secure cloud service that works on a security-based client-server model [22, 23].

Fig. 2. System Architecture



Software as a service (SAAS) also called on-demand software or web-based software which is accessed by users by leading enterprise and company. SAAS is a part of PAAS and IAAS. PAAS provides hosts to software and hardware on its infrastructure and IAAS Provides virtualized computing resources. This model creates a platform to develop a secure distributive system in cloud computing. Based on protocol each transmission performs. Due to the largest platform of enterprises and business organizations of cloud computing, it is very important to take care of security-based transmission.

2. Methodology

In this research work, a flexible and effective distributed technique with dynamic information supports has been developed to assure the user data correctness in the model of cloud-based server. Present work helps in correcting code during file distribution and information dependability. Present research work far-reaching effect to reduce storage with communication problem compared to technique during the distribution of file by utilizing the token called “homomorphic” with erasure-coded data for distributed verification. Based on this technique it achieved error data localization and integration of storage correctness. This work provides data error localization, efficient supports and dynamic secure operations based on information blocks, such as append, delete and update and also provide safety against the attack of server colluding, attack of malicious data modification and Byzantine failure.

Fig: 3. Cloud Servers

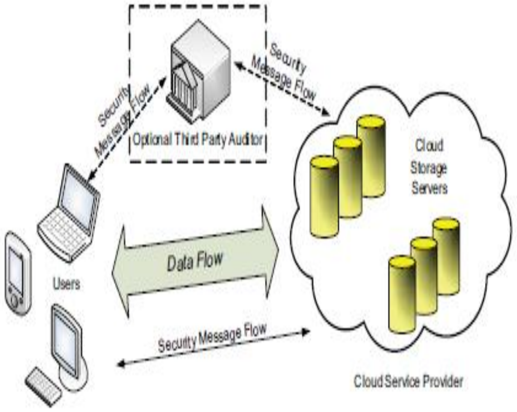
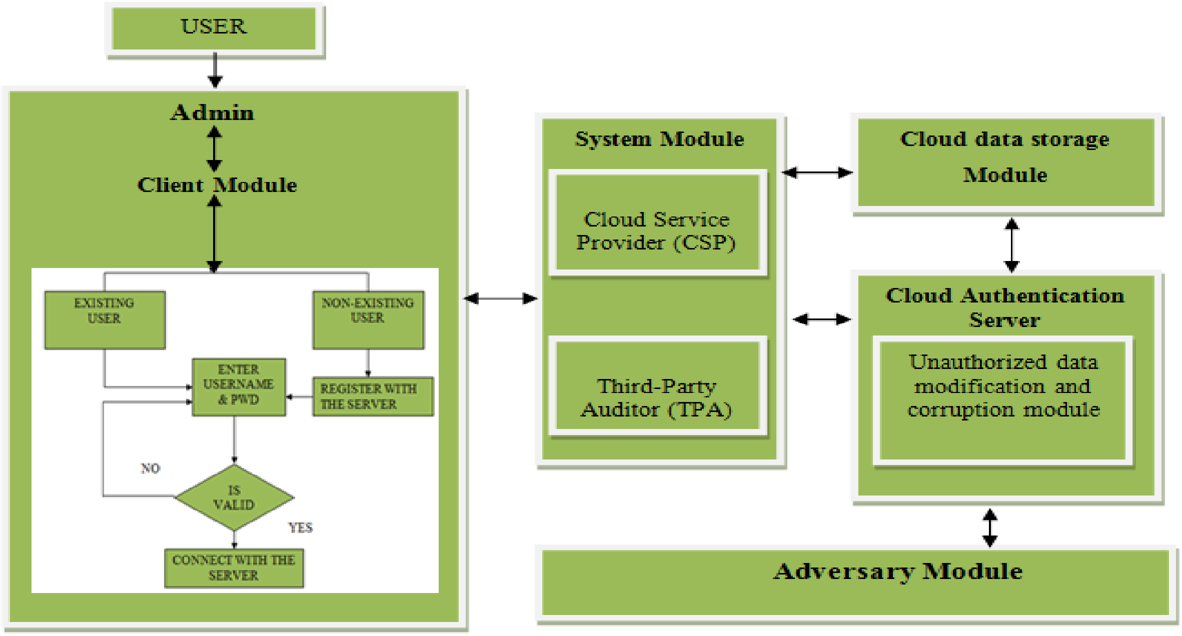


Fig. 4. Process Architecture Model



2.1 Client Module

In this module, initially, client authorization steps involved than a client request file from the server, based upon a request server send the required file to the client. On the other end, the server verifies the client registered password and user name for security reason. Based on verification result, it satisfies then sever will accept the request given by the client then search files corresponding to the client from the stored data center. During the process, if the server finds intruder then it set an alternative pathway for the intruders to overcome issues.

2.2 System based Module

In the system module, three entities identify user, cloud service provider, and third-party auditor. 1) Users whose information that are stored in the server which reply on a server for computation of data it consists of individual organizations and consumer process. 2) A cloud Service Provider which help in managing resources significantly and expertise in cloud computing building management, during the distribution of data and information. 3) In TPA, it helps users to expose the risk and data storage security based on user requests.

2.3 Data Cloud Storage Module

In this module, the client stores his relevant information into cloud servers using CSP, which runs a secure model, the client access and retrieve his data from the cloud server via CSP. In a few cases, on his data user needs to perform block-level operations for security purpose to assured secure storage data without existence copies locally. In this situation, the client does not have much time necessary to monitor their data and information regularly, they delegate their tasks to a trusted operational with TPA based on respective choice, but in the proposed module, channel-based communication with end –to- end between each client and server are reliable and authenticated were designed, which can perform well compared to an existing technique.

2.4 Cloud Server based Authentication

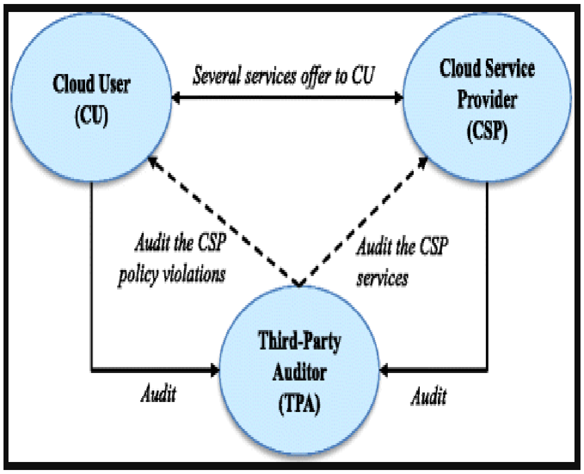
This module works with additional nature which added to the client protocol authentication. Initially, the client will send an authentication report to the router “masquerading”. The server also performs as authority-based ticketing and control access control through application network. Other operations supported by AS are updating the list of clients, authentication time reduction, and removal of the valuable client depending on the order or comment given by clients.

2.5 Corruption and modification of data through unauthorized party module

This module plays an important role to identify the key issues of data corruption and modification based on compromise server failure and Byzantine random failures. This module overcomes the above issues based on successfully detecting inconsistency to find data error lies in which server. This module helps in regular monitoring of the data modification alert and saves the original data before corrupting.

In many cases adversary are interested in corrupt individual data or information from the server, once the server compromised itself, an adversary can modify the original data or pollute important files by continuous modifying or inserting fraudulent data which change the originality of data in such cases weak adversary module will continuously monitor and track the changes over a period of the interval similarly in strong adversary module it check and track the hidden data, corruption, unknown modification, and data loss periodically.

Fig. 5. Protection in Adversary Module


Cloud User (CU)
Several services offer to CU
Cloud Service Provider (CSP)
Audit the CSP policy violations
Audit the CSP services
Third-Party Auditor (TPA)
Audit
Audit

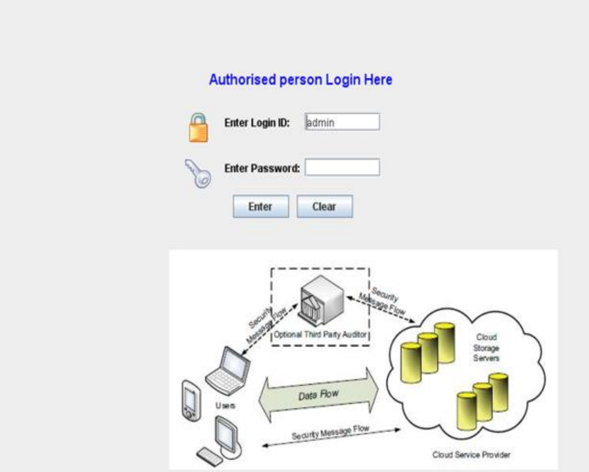
2.6 Adversary Module

In the Adversary module, cloud data storage can face security threats, which comes from two different bases. In the first base, a CSP can be interested, un-trusted, and probably malicious. During the movement of data transform clients may face difficulties of information loss issues based on errors in data management, Failure “Byzantine” and so on. Secondly, for a certain period’s data remain undetected by CSPs during data modification at different time intervals. In this research work, it follows two adversary modules: weak adversary and strong adversary.

3. Results and Discussion

Initially, it follows the authorized person login method using an authorized user name and password (Fig.6).

Fig. 6. Authorized Admin page



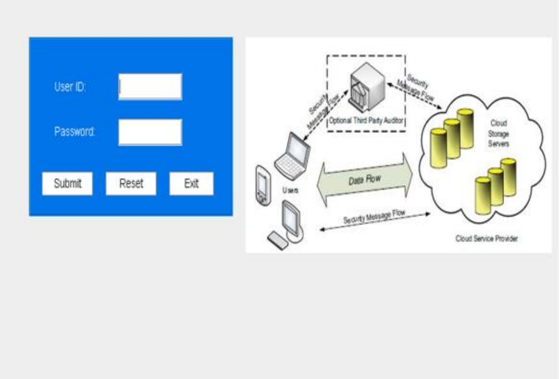
Authorised person Login Here

Enter Login ID: admin

Enter Password:

To assure more secure information stored in the cloud server then again user name and password need to be entered for the authentication process. Then a client requests authentication from the server according to the query, the cloud server sends the approval to the user (Fig. 7).

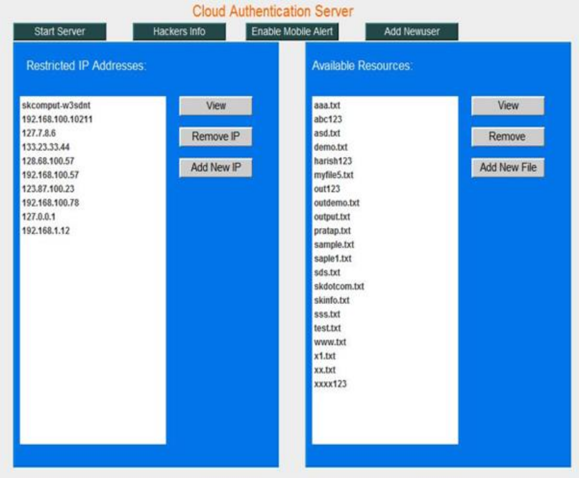
Fig. 7. Secure data storage security Admin page



The client will send authentic data to the router. The server performs control permission on the network. This section client can update the list and removal of valid client requests depending upon the request given by clients.

Based on authentication and secure data transform clients and regularly access, retrieve and append relevant information with ensuring complete security to the files (Fig. 8, 9 & 10).

Fig. 8. Cloud Authentication Server page



Cloud Authentication Server

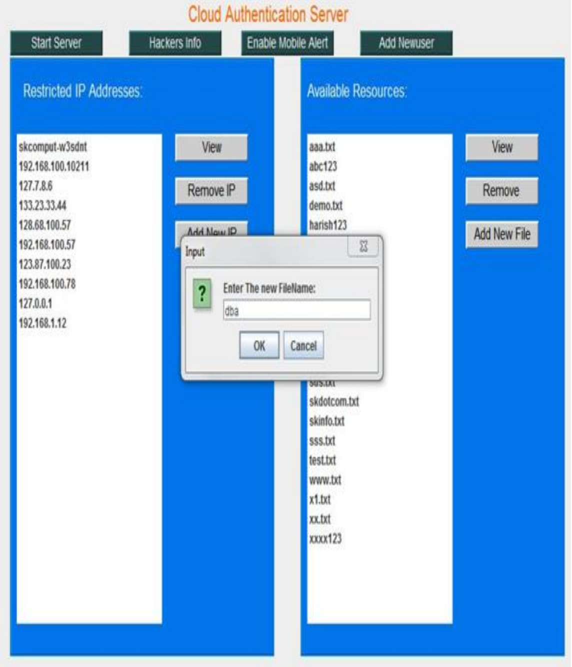
Restricted IP Addresses:

skcomput-w3sdnt
192.168.100.10211
127.7.8.6
133.23.33.44
128.68.100.57
192.168.100.57
123.87.100.23
192.168.100.78
127.0.0.1
192.168.1.12

Available Resources:

aaa.txt
abc123
asd.txt
demo.txt
harish123
myfile5.txt
out123
outdemo.txt
output.txt
pratap.txt
sample.txt
saple1.txt
sds.txt
skdotcom.txt
skinfo.txt
sss.txt
test.txt
www.txt
x1.txt
xx.txt
xxxx123

Fig. 9. Authorized Processing page

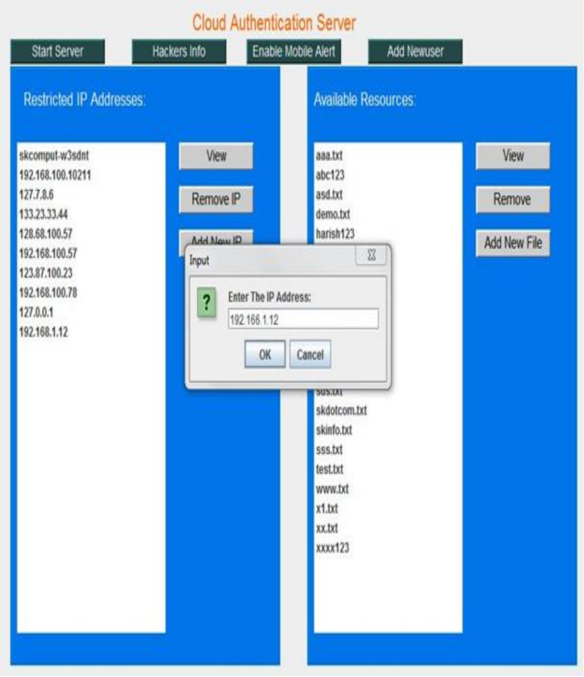


skcomput-w3sdnt
192.168.100.10211
127.7.8.6
133.23.33.44
128.68.100.57
192.168.100.57
123.87.100.23
192.168.100.78
127.0.0.1
192.168.1.12

aaa.txt
abc123
asd.txt
demo.txt
harish123
skdotcom.txt
skinfo.txt
sss.txt
test.txt
www.txt
x1.txt
xx.txt
xxxx123

Enter The new FileName: dba

Fig. 10. Authorized Operational Page



skcomput-w3sdnt
192.168.100.10211
127.7.8.6
133.23.33.44
128.68.100.57
192.168.100.57
123.87.100.23
192.168.100.78
127.0.0.1
192.168.1.12

aaa.txt
abc123
asd.txt
demo.txt
harish123
skdotcom.txt
skinfo.txt
sss.txt
test.txt
www.txt
x1.txt
xx.txt
xxxx123

Enter The IP Address: 192.166.1.12

Table 1: Test cases and Expected Behavior with Present status

Test no.	Input	Expected Behavior	Observed Behavior	Status passed(p)
1	Enter the authorized person login id and password	Enter into cloud authentication server page	Authentication page is open	pass
2	Enter wrong user name and password	Display message to enter correct user name and password	Unauthorized user check database	pass
3	Start server click view	Display registered IP address	List available resources	pass
4	Add new file	Enter the new files name	Saved in database	pass
5	Enter user IP address	User IP address will be saved in an IP database	Saved in database	pass
6.	Enter wrong user IP address	User IP address is wrong to enter correct IP Address	Unauthorized IP check database	Pass
7.	Deleting File	Notification of deleting file YES/NO	If YES accept command/ If No Back to page	Pass
8.	Appending File	Notification of appending file YES/NO	If YES accept command/ If No Back to page	Pass
9.	System crash	Enter into the cloud authentication server page to enter the authorized person id and password	Authentication page is open	Pass
10.	Update File	Notification of updating file YES/NO	If YES accept command/ If No Back to page	Pass

4. Conclusion

This research investigates and assures the security-based problems in data using cloud information storage. It also assures the correctness and security of users' information or data using a cloud server with a flexible distributed technique including block delete, append, and update. This research work, erasure correcting coded file distribution with a guarantee and secure transmission. With the use of a token called "homomorphic" with verification of erasure-coded data, this investigation assures storage correctness integration insurance with localization of information error and it also guarantees the detection of simultaneous servers misbehaving. Through complete performance analysis and security, this investigation is highly reliable, efficient, and resilient to any kind of failure, data modification attack, and server-based colluding attack. As we know in data

Storage security in the cloud area is very challenging and many problems need to be corrected and identified. This research also plans to work with fine-grained data error localization with a secure system using cloud computing.

Reference

[1] L. Ghezzi, Baresi, sms com journey: self-managing Computer Science – Research & Development, conferences with cloud computing structure, 2013.

[2] Poernomo, A MOF for monitoring properties of non-functional. Heidelberg (Springer) with international publication, 2005.

[3] Lemos, self-adaptive systems using SE: On roadmap. Springer, Heidelberg, 2013.

[4] Farokhi, Self-adaptation challenges for CBA with control-theoretic perspective. 10th (Workshop) Intl on Feedback Computing, 2015.

[5] M., Angelopoulos, Filieri, Control theory using software engineering in Symposium (International) for self management and adaptive using Soft Eng., 2015.

[6] Ghezzi, C., Non functional Managing uncertainty using adaptively model driven. Soft. Engineering Conference (International), 2013.

[7] Hoorn Van, M., Gul, A., Resource-efficient enabling framework operation of software in workshop of IEEE ICSE, 2009.

[8] Iftikhar, A self-adaptation assuring active formal model, International (Conf.) ACM on Soft. Engg., 2014.

[9] Jamshidi, A., Pahl, C, Resource used for cloud based software. International (Symp.) in Soft. Engg., SEAMS, 2014.

[10] Pahl, C., Architectures based on clusters cloud – Review based on technology, in Intl. (Conf.) on Future Things on internet and server based cloud, 2015.

[11] P., Bencomo, Finkelstein, A., Aware systems for self adaptive systems management. In Intl. Conference (Engg.), 2010.

[12] K. Angelopoulos N., Control theory on Software Engg., Intl. (Symposium) on Software (Engg.) for self managing systems, 2015.

[13] L. Baresi, Self managing situational computing. Comp. Sci. – Research and Development, 2013.

[14] K. Chan, Framework nonfunctional properties. Soft. Arch. With Software based quality, 2005.

[15] R. De Lemos, Soft. Engg., self adaptive syst., roadmap based research, soft. Engg. Using adaptive syst., 2013.

[16] S. Farokhi, Challenges based cloud applications, (Workshop) Intl. computing, 2015.

[17] C. Ghezzi, G. Tamburrelli. And P. Spoletini, Managing uncertainty non-functional. Intl. Conf. on Software Engineering, 2013.

[18] D. Weyns. And M.U. Iftikhar, Assure system advantages under self adaptation. Intl. Conf. proceeding ACM on Soft. Engg., 2014.

[19] C. Pahl, A. Ahmad, and P. Jamshidi, Provisioning of software using cloud. Intl. (Symposium) on Soft. Engg. SEAMS, Self managing Syst., 2014.

[20] C. Pahl. Containers for edge cloud based architecture - review. International Conf. on Internet and Cloud, 2015.

[21] N. Bencomo, J. Whittle, P. Sawyer, A. Finkelstein and E. Requirements systems for self-adaptive syst., International Engg. Conf., 2010.

[22] M. Rohr, A. van Hoorn, W. Hasselbring and A. Gul, Framework enabling resource- of software systems. CSE Proceedings IEEE, 2009.

[23] Y. Zhang, L. Zhang, L. Xu, C. Pahl and P. Jamshidi. Workload patterns of dynamic self auto-scaling and cloud service configuration. Intl. Conf. on Cloud and Utility Cloud Compt. UCC, 2014.