IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A CLOUD SECURE STORAGE MECHANISM BASED ON DATA DISPERSION AND ENCRYPTION

K. Devendra Reddy[1], Dr . K. Venkataramana[2],

[1]*PG Students, Dept. of MCA, Vemu Institute of Technology, P. Kothakota,* devareddy077@gmail.com

[2]Assosciate Professor Dept. *of MCA, Vemu Institute of Technology*, P. Kothakota, venkatvemucse1@gmail.com

ABSTRACT:

The large-scale sharing needs of many enterprises promote the development of cloud storage. While the cloud computing stores the shared files outside the trust domain of the owner, the demands and concerns for file security is arising. In this, a Group Key Management Protocol for file sharing on cloud storage is proposed. Faced with network attacks from public channel, a group key generation scheme based on mixed encryption technology is proposed. And a verification scheme is used to prevent shared files from being attacked by the collusion attack of cloud providers and group members. Security and performance analyses indicate that the proposed protocol is both secure and efficient for data sharing in cloud computing.

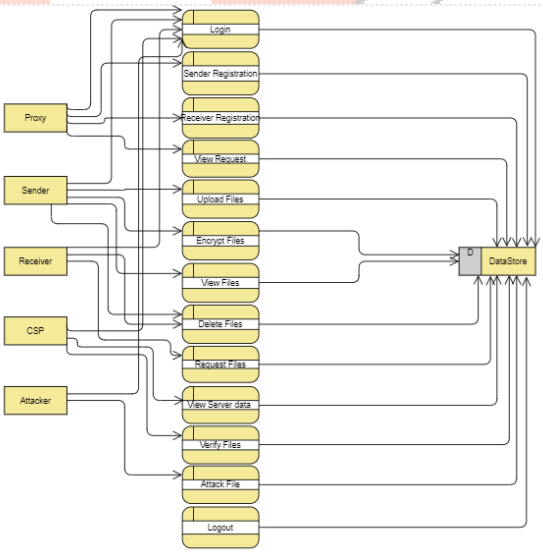
KEY WORDS: Cloud computing, data dispersion, data encryption, key management, storage security.

I. INTRODUCTION:

Cloud computing has shown remarkable development in recent decades. When the storage as a service, it occupies the center stage and backbone for many applications, such as pattern recognition image forensic and forgery detection. As a result, larger volumes of data will be a part of the cloud area. In the cloud industry, Amazon Web Service (AWS) has become the de facto standard. As the core component of the Opens tack that follows this standard, Swift has become one of the most popular cloud storage mechanisms. However, Opens tack Swift mechanism still faces many real security threats while providing convenient services. According to Cloud Security Alliance's top threat case analysis report released in 2018, two thirds of the cases will cause user data leakage, mainly due to management negligence and malicious attacks. For instance, under default configuration, OpenStack Swift mechanism typically stores data in plaintext for the sake of performance.

1.1 OBJECTIVES:

The main aim of this project is to achieve the authentication scheme is used to prevent shared files from being attacked by



2.BLOCK DIAGRAM

PROXY:

Proxy can login with valid credentials; proxy is responsible to add Sender/Receivers Information and send login information to the sender/receiver through email. Proxy also views requests from the receiver

SENDER:

Sender will login with the credentials which are sent by the proxy sender is responsible to upload files, encrypt files and to generate trapdoor view Files and Delete those files.

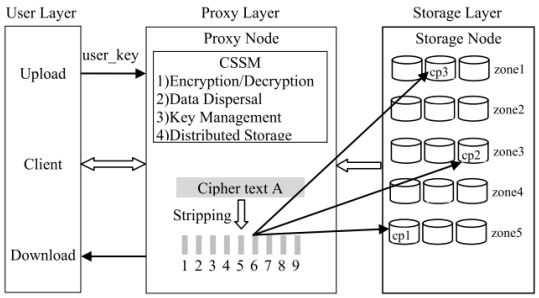
RECEIVER:

Receiver also login as Sender but performs different operations like view all files which are uploaded by the sender and send's request to the proxy to view the file data and view files which are accepted by the proxy. And finally, logout from the site.

CLOUD SERVICE PROVIDER:

Cloud service provider login with valid credentials and view all the server's data and verifies files are attacked by the attacker and provide some security to the files and logout from the system.

ARCHITECTURE:



ARCHITECTURE OVERVIEW

To realize primary object and properties above, this paper presents CSSM, a cloud secure storage mechanism. As shown in Figure 1, CSSM could be divided into three layers: The user layer, the proxy layer, and the storage layer. Specifically, the main functions of each layer are as follows:

1) User Layer: This layer is deployed on the user's machine, and the user operates (upload, download, etc.) cloud data through the client.

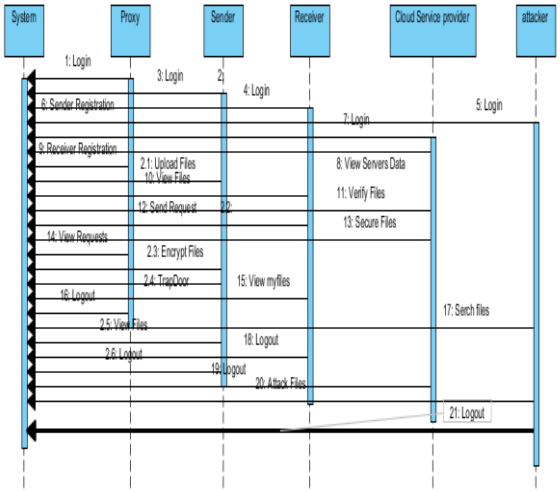
2) Proxy Layer: This layer is deployed in the cloud and composed of proxy nodes with trusted execution environments, such as Intel SGX technology and ARM Trust Zone technology. In trusted execution environment, CSSM programs could perform as expected. CSSM in proxy layer includes four modules: data encryption/decryption, data dispersal, key management and distributed storage.

1. **Encryption/Decryption:** This module is used to encrypt user uploaded data and decrypt user downloaded data.
2. **Data Dispersal:** According to the data dispersal model, the cipher texts is divided into several small blocks.
3. **Key Management:** This module is not only responsible for the generation and maintenance of the key, but also uses the hierarchical key management approach to protect the key.
4. **Distributed storage:** This module distributes chunked and encrypted data to storage layer.

3) Storage Layer: This layer consists of a number of storage nodes that are used to store chunked and encrypted data. Considering data loss or unavailability caused by accident like equipment damage or natural disasters, cloud service providers divide large number of storage nodes into several zones, each of which acts as a failure boundary between multiple copies of the same data.

ATTACKER:

Attacker login with malicious content and attacks the files to destroy or to theft files. Logout from the site.



3 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order.

ER DIAGRAM:

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let's have a look at a simple ER diagram to understand this concept

4. CONCLUSION:

For the issue of cloud data leakage caused by management negligence and malicious attack at storage layer, we proposed CSSM, a cloud secure storage mechanism. CSSM adopted a combined approach of data dispersal and encryption technologies, which can improve the data security and prevent attackers from stealing user data. The experimental results show that CSSM can effectively prevent user data leakage at cloud storage layer. In terms of performance, the increased time overhead of CSSM is acceptable to users. This paper provides a feasible approach to solve the storage security problem, especially prevention from user data leakage at cloud storage

layer. CSSM could also effectively protect cryptographic materials from storage perspective.

5.REFERENCES:

[1] A. Bhardwaj, F. Al-Turjman, M. Kumar, T. Stephan, and L. Mostarda, "Capturing-the-invisible (CTI): Behavior-based attacks recognition in IoT-oriented industrial control systems," IEEE Access, vol. 8, pp. 104956–104966, 2020.

[2] M. Kumar, A. Rani, and S. Srivastava, "Image forensics based on lighting estimation," Int. J. Image Graph., vol. 19, no. 3, Jul. 2019, Art. no. 1950014.

[3] M. Kumar, S. Srivastava, and N. Uddin, "Image forensic based on lighting estimation," Austral. J. Forensic Sci., vol. 51, no. 3, pp. 243–250, Aug. 2017.

[4] J. Li, Y. Zhang, X. Chen, and Y. Xiang, "Secure attribute-based data sharing for resource-limited users in cloud computing," Comput. Secur., vol. 72, pp. 1–12, Jan. 2018.

[5] Y. Zhang, X. Chen, J. Li, D. S. Wong, H. Li, and I. You, "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing," Inf. Sci., vol. 379, pp. 42–61, Feb. 2017.

[6] The OpenStack Project. OSSA-2015-006: Unauthorized Delete of Versioned Swift Object. Accessed: Apr. 14, 2015. [Online]. Available: https://security.openstack.org/ossa/OSSA-2015-006.html

[7] The OpenStack Project. OSSA-2015-016: Information Leak Via Swift Tempurls. Accessed: Aug. 26, 2015. [Online]. Available: https://security. openstack.org/ossa/OSSA-2015-016.html

[8] The OpenStack Project. Possible Glance Image Exposure Via Swift. Accessed: Feb. 23, 2015. [Online]. Available: https://wiki. openstack.org/wiki/OSSN/OSSN-0025

[9] Cloud Security Alliance. Top Threats to Cloud Computing: Deep Dive. Accessed: Aug. 8, 2018. [Online]. Available: https://downloads. cloudsecurityalliance.org/assets/research/top-threats/top-threats-to-cloudcomputing-deep-dive.pdf

[10] The OpenStack Project. OpenStack Security Advisories. Accessed: Feb. 2, 2015. [Online]. Available: https://security.openstack.org/ossalist. html

[11] Common Vulnerabilities and Exposures. CVE-2015-5223. Accessed: Jul. 1, 2015. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename. cgi?name=CVE-2015-5223

[12] Common Vulnerabilities and Exposures. CVE-2016-9590. Accessed: Nov. 23, 2016. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename. cgi?name=CVE-2016-9590

[13] S. Y. Shah, B. Paulovicks, and P. Zerfos, "Data-at-rest security for spark," in Proc. IEEE Int. Conf. Big Data (Big Data), Washington DC, USA, Dec. 2016, pp. 1464–1473.

[14] Z. Liu, Y. Huang, J. Li, X. Cheng, and C. Shen, "DivORAM: Towards a practical oblivious RAM with variable block size," Inf. Sci., vol. 447, pp. 1–11, Jun. 2018.

[15] X. Zhang, X. Chen, J. Wang, Z. Zhan, and J. Li, "Verifiable privacypreserving single-layer perceptron training scheme in cloud computing," Soft Comput., vol. 22, no. 23, pp. 7719–7732, Dec. 2018.