IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 2, Issue 2, Apr-May, 2014
ISSN: 2320 – 8791 (Impact Factor: 1.479)
www.ijreat.org

Cloud Computing with Increased Performance using Key Distribution Center

A .Azhar Mohammad Yousuf[1], J.Godwin Manickaraj[2], G.Praveen Kumar[3]

[1,2]UG Student, Department of Information Technology, Anand Institute of Higher Technology, Chennai.

[3]Asistant Professor, Department of Information Technology, Anand Institute of Higher Technology, Chennai.

ABSTRACT

Cloud computing plays amajor role in data sharing. Ever since cloud computing was invented there were always been threats related to security and efficiency. In order to increase both security and performance of a cloud server a new concept called Key Distribution Center (KDC) is to be created. These KDC is provided for each and every Organization accessing a particular cloud server. The KDC acts as a sub-server to the Organization. By using KDC every organization gets the right to modify the terms related to data sharing in their corresponding KDC. Since every Organization is provided with separate KDC unauthorized access is restricted. To increase security in data sharing each file is encrypted and converted to jar extension formatted files. The jar files compresses the original file size and thus helps in reducing load in cloud server which leads to increase in performance of a server.

1. INTRODUCTION

Ever since cloud computing has been introduced there is always threats related to security. The security is a major concern in cloud computing since data is always stored in outside sources[1] . Most of the sources is not fully authenticated and chance of security breach is at high rate. Most of the present personal computers were not fully protected and thus they go under major threats and attacks. Some cloud servers are protected by the particular organization resulting in setting up their own cloud Environment.

We introduce a proposed system that lets the user to handle and maintain their own cloud service system which is called as Key Distribution Center(KDC). The KDC acts as a sub-server to the main centralized server. The organization or user is initially requested to register with the cloud server in order to access the KDC. After initialization the user is provided with a separate key distribution center. The user gets the entire authentication to access or maintain their particular key distribution center. The organization declares particular criteria for its user to access the key distribution center. Similarly for each and every organization a separate set of key distribution center is provided . The organization decides whether or not to share their information with other Organization.

In order to increase security some limitation were used inside the key distribution center. The limitation is applied only to the data storing methods. The security is applied to the data storage since it is essential to make sure that data is transferred between users at fully efficient and secured manner. For this security factor the data are stored in the key distribution center in a compressed as well as encrypted manner. For this encryption method an Identity Based Encryption [2] is used .This encryption provides encrypted data to be stored in the server on the basis of converted files. Thus only converted files were stored in the server and the security is increased using this method. Since the files are encrypted the files have to be decrypted in order to get back the original data. Thus the key used for decryption is essential in order to convert the files back to its original form.

The purpose of this project is to increase security along with the performance. The cloud server responds at a slow rate at some situations due to increased in traffic or when the load in the server is increased. The server load can be contained if the

data stored in the server is at compressed rate. For this compressed set of data we use a module called jar creation and security. After encrypting the data they are then stored in the cloud server in form of compressed jar file. Since the file is stored in compressed format the server load is reduced at high rate.

2. PORTFOLIO

Rick Stiggins (1994) defines a portfolio as a collection of student work that demonstrates achievement or improvement. The material to be collected and the story to be told can vary greatly as a function of the assessment context. The Northwest Evaluation Association offers a similar definition: A purposeful collection of student work that illustrates efforts, progress, and achievement in one or more areas [over time]. The collection must include: student participation in selecting contents, the criteria for selection, the criteria for judging merit, and evidence of student self-reflection.

2.1 Cloud Computing

The cloud computing is the process of providing data and authentication in terms of a service. The cloud computing has increased the data sharing at very high rate. The major purpose of cloud computing has used in order to access data at anytime and at anywhere. The cloud computing has involved at major services. It is mainly used for provide data as a service, platform as a service and software as a service.

2.2 Motivation of Research Problem

1. What is the purpose of using Key Distribution center?
2. What are the benefits of uploading encrypted files to the server?
3. What are the benefits of storing compressed jar files in the server?

3. KEY DISTRIBUTION CENTER

The Key Distribution Center used in order to provide the organization with their own control in managing the particular type of contents and limitations. However the key distribution center belongs to the centralized server. The key distribution center provides the organization the full control that is needed to configure their own set of cloud server information. The main objective is to ensure that data shared inside the key distribution center belongs to the particular organization. The centralized server provides the organization the full control that is required in order to maintain and control is corresponding key distribution center. In order to obtain a key distribution center from the server it is necessary for the organization to initially register with a centralized server. It is more often like registering an account for an email ID. The same type of concept is implemented with the organization to the centralized server.

After registering, the control of the key distribution center passes to the organization similar to the control passes to the user after registering the email ID. The key distribution center is then managed by the organization and the organization decides the what type of data that has to stored and who gets the authenticated rights to access those data.

The sharing of information between two organization is done only on the basis of those particular organization wishes to accept the limitations empowered by both these organization with another [4]. The major concern involved in this sharing is to ensure that different organization cannot access data directly from one another. The data is accessed by two organizations with their corresponding key distribution center. Thus when an organization wants to share its data to another organization it should ensure whether the other organization contains a key distribution center.

With the corresponding increase of data sharing in cloud it also comes under the fact as of the importance in cloud computing.

4. CLOUD STORAGE

Application developers seeking easy, cloud-based storage and access for their data will find all Cloud[3] Storage a great match. Google Cloud Storage also helps businesses and individuals with many other tasks[9]

• Archive or back up data Google Cloud Storage provides a high-reliability, high-availability data-backup solution that is easy to maintain.

• Store application data Google Cloud Storage provides fast access to application data, such as images for a photo editing app.
• Share data with colleagues and partners Google Cloud Storage lets data owners quickly create and manage Access Control Lists (ACLs) to their data, which is especially helpful if the data has a dynamic user base.

• Analyze large amounts of data Google Cloud Storage supports Google's analytic tools, including the Google Prediction API and Google Big Query Service, which lets the data owners analyze terabytes of data for powerful business insights.

• Serve static data for websites High availability and performance make Google Cloud Storage a great choice for storing and serving static data (including user-generated content) for websites.

The easy to interface option thus helps in increasing the system modular functions and also ensures that the data are shared at efficient rate and at a perfect manner. The data shared should be secure and informative and thus there should be no loss in data during its transmission or after the transmission of data. Since the cloud storage and accessing is easy to use and thus it increases efficient usage of the cloud server.

The Key distribution center manages all these credential information and also it can be modified as per the organization request since it comes under direct control of the particular developed organization.

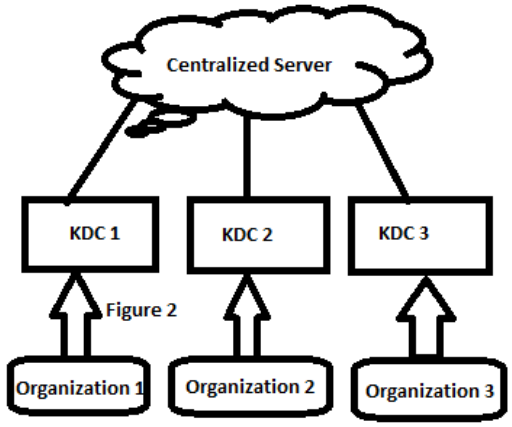

Figure : 1 Key distribution Center

The key distribution centers act as server for the particular sets of organizations and all the key distribution centers are controlled by the centralized server.

The key distribution center is distributed based on Identity based entry scheme to the centralized server. After the creation and initialization of the key distribution center the organization gets the right to restrict the centralized server access to its particular key distribution center.

The centralized server is restricted at some cases due to security factors. However the key distribution center resides only inside the centralized server but it is not allowed to modify or view the contents of the center since they are controlled by the Organization. This results in increase in high level of security to the data that are stored in the server.

The purpose of this method helps in understanding the system effectively. Since the control is passed to the Organization the security factors to the key distribution center is also modified by the Organization. The organization should alter the system in such a way so that the security level will be at a very high rate.

5. AES ENCRYPTION

The data stored in cloud server is often outsourced since they are stored in form of direct set of data. Since they are stored without any modification or

content thus it is easy for hackers to access data without any defects [12].

Encryption, by itself can protect the confidentiality of messages, but other techniques are still needed to protect the integrity and authenticity of a message. For example, verification of a message authentication code (MAC) or a digital signature.

Standards and cryptographic software and hardware to perform encryption are widely available, but successfully using encryption to ensure security may be a challenging problem. A single slip-up in system design or execution can allow successful attacks. Sometimes an adversary can obtain unencrypted information without directly undoing the encryption.

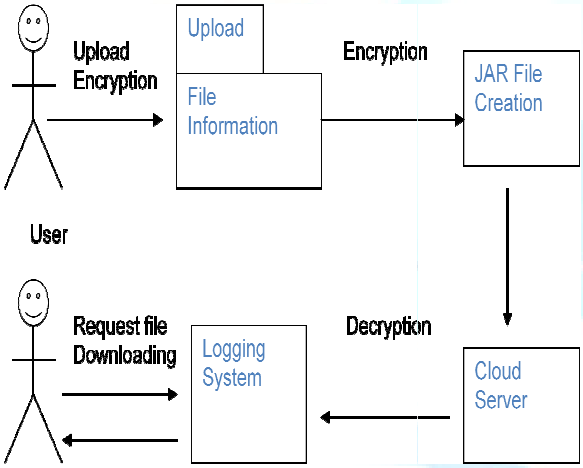

Figure :2 The file which is uploaded is converted in form of Jar format to increase efficiency and reduce server load . The file is then again converted to de-jar format during encryption.

The above diagram displays the functional working of the data storage inside the cloud server system. The data is first encrypted and then the encrypted data is then compressed and then stored inside the cloud server in form of encrypted data file. If an user wants to get back the data then the first step done is the process of de-jar the stored data. Then decryption technique is done in order to retrieve the original set of data to the user. For this decryption the key used for encryption is used in order to retrieve the data. The key distribution center is often acted as a database since the data stored are completely configured, managed by their particular organization. The data are shared within organization using their corresponding key distribution center. The key needed for decryption is shared only by manual methods due to security factors.

In this system the data is stored inside the cloud server only after encrypting the original set of data. The identity based encryption standard is used in order to convert the original set of data. Thus he server is loaded only with the converted encrypted data and it requires the key that is used for conversion in order to retrieve back the original data. Thus decryption is done in order to get back the original set of data.

The key is exchanged manually due to increase in security factors [15]. The key is usually exchanged between multiple systems along with the encrypted files. But this enables the attacker to gain access to the encrypted data since the key is transmitted along with the encrypted files in the server. Thus in this model the key is transferred manually between one another thus increasing security. Even if the attacker gains access to the server the attacker is still restricted to get back the original set of data. This is due to encrypted file and the key responsible for encryption is passed manually among different users. The security factor is increased and the chance for security breach inside this system reduced at high rate.

5.1 Cryptographic Key Management

Prior to any secured communication, users must set up the details of the cryptography. In some instances this may require exchanging identical keys (in the case of a symmetric key system). In others it may require possessing the other party's public key. While public keys can be openly exchanged (their corresponding private key is kept secret), symmetric keys must be exchanged over a secure communication channel. Formerly, exchange of such a key was extremely troublesome, and was greatly eased by access to secure channels such as a diplomatic bag. Clear textexchange of symmetric keys would enable any interceptor to immediately learn the key, and any encrypted data.

The advance of public key cryptography in the 1970s has made the exchange of keys less troublesome. Since the Diffie-Hellman key exchange protocol was published in 1975, it has become possible to exchange a key over an insecure communications channel, which has substantially reduced the risk of key disclosure during distribution. It is possible, using something akin to abook code, to include key indicators as clear text attached to an encrypted message. The encryption technique used by Richard Sorge's code clerk was of this type, referring to a

page in a statistical manual, though it was in fact a code. The German Army Enigma symmetric encryption key was a mixed type early in its use; the key was a combination of secretly distributed key schedules and a user chosen session key component for each message.

6. JAR FILE CREATION

The JAR file is created and only the encrypted JAR file is stored inside the key distribution center [8]. The jar is created automatically with a prescribed mechanism. Any access to the file will automatically trigger a jar file creation [10]. The jar files are stored inside the key distribution center and can be accessed only by the particular organization that stored the corresponding file in the system.

The main responsibility of this concept is to handle authentication of entities which want to access the data stored in the JAR file. In our context, the data owners know the exact servers that are going to handle the data. Hence, authentication is specified according to the servers functionality (which we assume to be known through a lookup service), rather than the server's URL or identity. For example, a policy may state that Server X is allowed to download the data if it is a storage server. As discussed, the JAR may also have the access control functionality to enforce the data owner's requirements, specified as Java policies, on the usage of the data [13]. A Java policy specifies which permissions are available for a particular piece of code in a Java application environment. The permissions expressed in the Java policy are in terms of File System Permissions [14]. However, the data owner can specify the permissions in user-centric terms as opposed to the usual code-centric security offered by Java, using Java Authentication and Authorization Services. Moreover, the JAR is also in charge of selecting the correct data according to the identity of the entity who requests the data

7. CONCLUSION

The developed system enables faster processing of information which leads to increase in performance of the system. Since the server load is reduced due to encrypted jar files the performance level of the key distribution center is normally increased. The security factor is increased and multiple threats have been avoided with the encrypted data being stored in the server. Overall the system is implemented is responsible for high level processing of information stored inside the cloud server.

REFERENCES

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z.Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. ACM Conf. Computer and Comm. Security, pp. 598-609, 2007.

[2]D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," Proc. Int'l Cryptology Conf. Advances in Cryptology,pp. 213-229, 2001.

[3] P. Buneman, A. Chapman, and J. Cheney, "Provenance Management in Curated Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '06), pp. 539-550, 2006.

[4] B. Chun and A.C. Bavier, "Decentralized Trust Management and Accountability in Federated Systems," Proc. Ann. Hawaii Int'l Conf.System Sciences (HICSS), 2004.

s[5] B. Crispo and G. Ruffo, "Reasoning about Accountability within Delegation," Proc. Third Int'l Conf. Information and Comm. Security (ICICS), pp. 251-260, 2001.

[6] Y. Chen et al., "Oblivious Hashing: A Stealthy Software Integrity Verification Primitive," Proc. Int'l Workshop Information Hiding, F. Petitcolas, ed., pp. 400-414, 2003.

[7] X. Feng, Z. Ni, Z. Shao, and Y. Guo, "An Open Framework for Foundational Proof-Carrying Code," Proc. ACM SIGPLAN Int'l Workshop Types in Languages Design and Implementation, pp. 67-78,
2007.

[8]J.W. Holford, W.J. Caelli, and A.W. Rhodes, "Using Self-Defending Objects to Develop Security Aware Applications in Java," Proc. 27th Australasian Conf. Computer Science, vol. 26, pp. 341-349, 2004.

[9] P.T. Jaeger, J. Lin, and J.M. Grimes, "Cloud Computing and Information Policy: Computing in a Policy Cloud?," J. InformationTechnology and Politics, vol. 5, no. 3, pp. 269-283, 2009.

[10] J.H. Lin, R.L. Geiger, R.R. Smith, A.W. Chan, and S. Wanchoo, Method for Authenticating a Java Archive (jar) for Portable Devices, US Patent 6,766,353, July 2004.

[11] T. Mather, S. Kumaraswamy, and S. Latif, Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice), first ed. O' Reilly, 2009.

[12] J. Park and R. Sandhu, "Towards Usage Control Models: Beyond Traditional Access Control," SACMAT '02: Proc. Seventh ACMSymp. Access Control Models and Technologies, pp. 57-64, 2002.

[13] J. Park and R. Sandhu, "Towards Usage Control Models: Beyond Traditional Access Control," SACMAT '02: Proc. Seventh ACM Symp. Access Control Models and Technologies, pp. 57-64, 2002.

[14] S. Pearson and A. Charlesworth, "Accountability as a Way Forward for Privacy Protection in the Cloud," Proc. First Int'l Conf. Cloud Computing, 2009.

[15] S. Pearson, Y. Shen, and M. Mowbray, "A Privacy Manager for Cloud Computing," Proc. Int'l Conf. Cloud Computing (CloudCom), pp. 90-106, 2009.

[16]A. Pretschner, M. Hilty, and D. Basin, "Distributed Usage Control," Comm. ACM, vol. 49, no. 9, pp. 39-44, Sept. 2006.

[17] A. Pretschner, F. Schuo¨ tz, C. Schaefer, and T. Walter, "Policy Evolution in Distributed Usage Control," Electronic Notes TheoreticalComputer Science, vol. 244, pp. 109-123, 2009.

[18] A. Squicciarini, S. Sundareswaran, and D. Lin, "Preventing Information Leakage from Indexing in the Cloud," Proc. IEEE Int'l Conf. Cloud Computing, 2010.

[19] D.J. Weitzner, H. Abelson, T. Berners-Lee, J. Feigen-baum, J.Hendler, and G.J. Sussman, "Information Accountability," Comm.ACM, vol. 51, no. 6, pp. 82-87, 2008.

[20] E. Barka and A. Lakas, "Integrating Usage Control with SIP-Based Communications," J. Computer Systems, Networks, and Comm., vol. 2008, pp. 1-8, 2008.