Asian Journal of Applied Science and Technology (AJAST)
(Open Access Quarterly International Journal) Volume 2, Issue 1, Pages 275-280, 2018

Cloud Security – Protection for Cloud Data Using 3DES Encryption

Divin Davis[1], Anjana Shaji[2], Edwin Earny[3], R Satheesh Kumar[4]

[1,2,3]*UG Student, Dept. of Computer Science and Engineering, Sahrdaya College of Engineering & Technology, Kerala, India.*
[4]*Associate Professor, Dept. of Computer Science and Engineering, Sahrdaya College of Engineering & Technology, Kerala, India*

Article Received: 27 November 2017 *Article Accepted: 24 January 2018* *Article Published: 30 March 2018*

ABSTRACT

Storage is the most prominent feature of cloud computing, growing rapidly in quality which gives immediate access to information through web service application programming interface (or) web-based content management systems. Cloud storage providers store the data on multiple servers when it is distributed. These servers are maintained by hosting companies for providing immediate access increasing the risk of unauthorized access to the private content of data. The risk of unauthorized access can be reduced by using encryption techniques. In the proposed system user encrypts all the files with distinct keys before uploading them to the cloud. The user can upload the files as private or public. However, public files can be downloaded directly, but to download the private files a user will send a request to the file owner. The user has the flexibility to request single or multiple files at a time. When the file owner accepts the request the application server provides a single Access key extracted from the attributes of the requested files. This Access key is shared with the requesting user which further retrieves the private key of the files. Using the private key cypher text is converted into plain text, and the plain text gets downloaded. This technique increases the flexibility of sharing the files as we are sharing single Access key for multiple files requested.

Key Words: Cloud storage, encryption technique, triple DES.

1. INTRODUCTION

Cloud computing has become an emerging infrastructure for organizations throughout the world. The cloud computing uses specialized connections with a network of servers gathered substantially for data processing across them. Frequently, virtualization techniques are utilized to maximize the power of cloud computing [1]. Through the use of virtualization, it reduces the need for purchasing, maintaining and updating their own networks and computer systems as it uses the computing resources as a service over a network. Cloud storage is the place where digital data is saved in logical pools. Cloud storage spans multiple servers. In favour of that, all the rights for the physical environment belong to the hosting company. Here, clients purchase storage capacity of the providers to host asserts facilitated with them in the remote server.

Individual user and organizations benefit from cloud computing services, which allow permanent online storage of files. The problem occurs when companies store highly confidential documents on cloud servers. It suffers from all inherent issues the earlier networking technologies suffered. Networking between two computers has always been plagued with security and privacy issues. The data and communication channel had to be secured from intrusion; however, there can never be foolproof security. Cloud Computing has to face many issues and challenges as it is still in its infancy. Much research is being done in this field to better the structure and functioning of the cloud.

We propose a system for increasing data security using encryption technique. This paper aims to introduce a backbone structure for a cloud storage system where the security and personal privacy is highly maximized. It is very obvious that cloud computing servers are highly protected against unauthorized access, but in some cases these files stored can be accessible by the maintenance staffs. Fully protection is needed to ensure that the files stored on the server are only accessible to owners.

2. SYSTEM DESIGN

In our work system, the user encrypts all the files with distinct keys before uploading them to the cloud. The user can upload the files as private or public. However, public files can be downloaded directly, but to download the private files a user will send a request to the file owner. The user has the flexibility to request single or multiple files at a time. When the file owner accepts the request the application server provides a single Access key extracted from the attributes of the requested files. This Access key is shared with the requesting user which further retrieves the private key of the files. Using the private key cypher text is converted into plain text, and the plain text gets downloaded. This technique increases the flexibility of sharing the files as we are sharing single Access key for multiple files requested. The primary objective of this project is to maintain security of the data stored in the cloud that runs on the network. To process with the cloud system, the system should have network facility. The data stored will be encrypted by the system using symmetric encryption. This encryption is to prevent the unauthorized access, from intruders including employee of enterprise, which attempts to retrieve data of the cloud storage user while the data is in transmission.

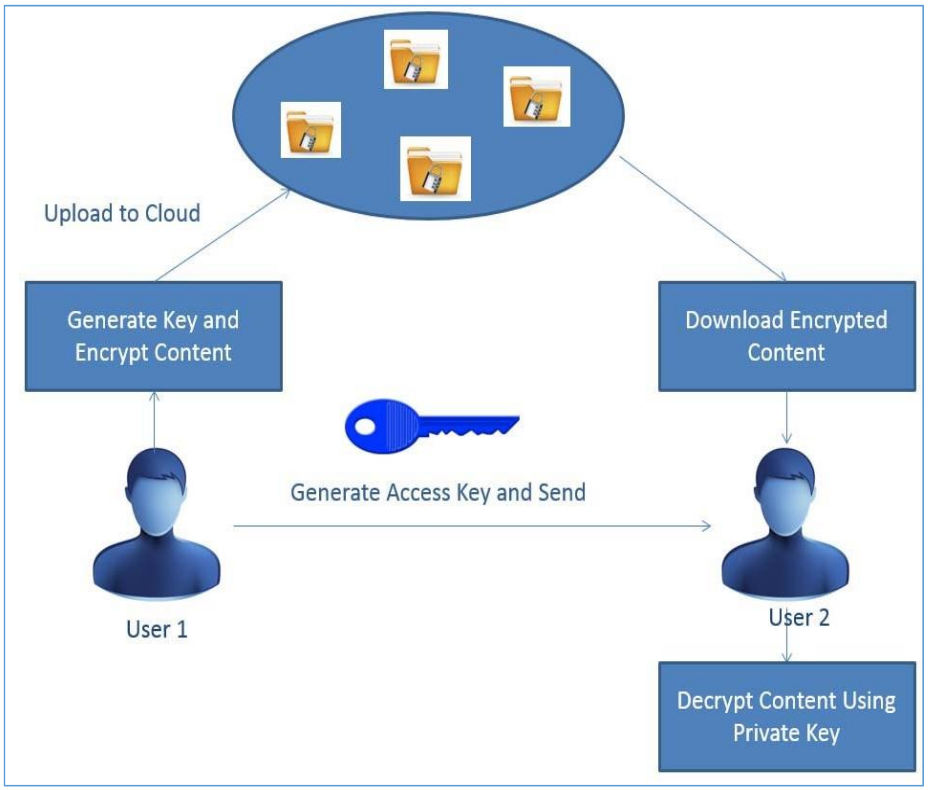

Fig -1: Cloud Repository System Architecture

In this technique, a user will encrypt the data using the private key and converts plain text into cypher text. Extracted cypher text will be stored in the cloud and the private key used for encryption will be stored in the local database. As the data stored is secure, any type of data such as personal or computed or an application data can be stored. To access the files of other users he/she can make a request. Whenever a request is made, the file owner generates an Access key for the requested set of files. The user can retrieve the shared data based upon the user credentials, file attributes and the Access key.

The System Architecture of the cloud repository system shown in Figure 3.1 describes various components and communication between those components. A user as depicted in the system architecture should be authorized to log in to the system. The user will communicate with the application server to store the data in the cloud through a web browser. When the user uploads the data it is encrypted using a key generated and thus uploaded to the cloud. Whenever a user requests for the files stored in the cloud, the file owner shares an Access key for requested files. As soon as the user enters the Access key, it gets the private key used to encrypt that file from the local database and decrypts the file using the private key and gets downloaded.

3. WORKING AND EXPERIMENTATION

Our system uses the cloud to store information about the users, files uploaded by the users, requests made, Access keys generated for the requested files for the requesting user. The login validations check the username and password entered with the username and password in the database and confirms or rejects login accordingly. Upon confirmation, the application server will establish a connection with the cloud repository system. After that, it will pull all the information from the cloud and show it to the user. This application allows the user to store or retrieve data from cloud repository system. Whenever a user tries to upload a file, a private key will be generated and that key will be used to encrypt the file. The key used to encrypt the file is stored in the local database and the encrypted data is stored in the cloud. Whenever a user tries to retrieve the data the public file can be downloaded directly whereas to retrieve the private files the user needs to request for an Access key. Using this Access key and file name, the private key for that particular file can be taken from the local database by the application server and file can be decrypted and downloaded.

4. MODULE DESIGN

There are five modules in this system, which we are again divided into sub- modules. The main modules are:-

1. Registration/login
2. Uploading Files
3. Requesting Files
4. Sharing Files
5. Downloading Files

3.1 Registration/login

In this module for the first time login user needs to register with the system to use the application. In the registration page, a form will be displayed to the user where valid information needs to be filled in the provided fields with a generated unique user id. A unique user id will be generated. All the required fields need to be filled appropriately. Once the user clicks the submit button with valid information it needs to be uploaded to the cloud. If the registration is successful, the user is redirected to the login page prompting successful registration. In the login page, a form will

be displayed to the user to enter his credentials provided during registration. If the login is successful, the user can start managing the files on the cloud server.

3.2 Uploading Files

A user can upload text files and image files. For each uploaded file a unique id is generated by the application server. However, both private files and public files are encrypted and stored using 3DES algorithm. While uploading, the user needs to mention the file name and upload it.

3.3 Requesting files

A user can see the files uploaded by all the users registered into the system. However, files made as the public can be downloaded directly. To download the private files, a user needs to send a request to the file owner to share the private key used for encryption.

3.4 Sharing files

Here the user has the flexibility to accept or reject the requests made. In order to accept/reject the requests made he/she needs to select the requested username. Eventually, all the files requested by the user are displayed where he can accept/reject few or all the files requested.

3.5 Downloading files

A user can download his/her files directly from the "MY FILES" page and the requested files can be downloaded in the "RECEIVED FILES" page. Whenever a request is made key element maintains any one of the status mentioned below:

1. Waiting
2. Accept
3. Reject

First, when the request is made by the user, the status of the key element will be in waiting until file owner accepts or rejects. Second, if the file owner accepts the request made then the status will be changed to the Access key shared. Last, if the file owner rejects the request made, the status will be changed from waiting to reject. Here only the accepted files would be able to download by the user.

In order to download the accepted files, a user needs to navigate to the "RECEIVED FILES SCREEN" and select the accepted file and enter the Access key shared by the file owner. Whenever the Access key is entered, it compares with the requested username and file name associated with it. If it matches the application server will get the private key used to encrypt the file from the local database and decrypts the file using and downloads the file. If the Access key entered does not match, it gives an appropriate error message to the user for resolving.

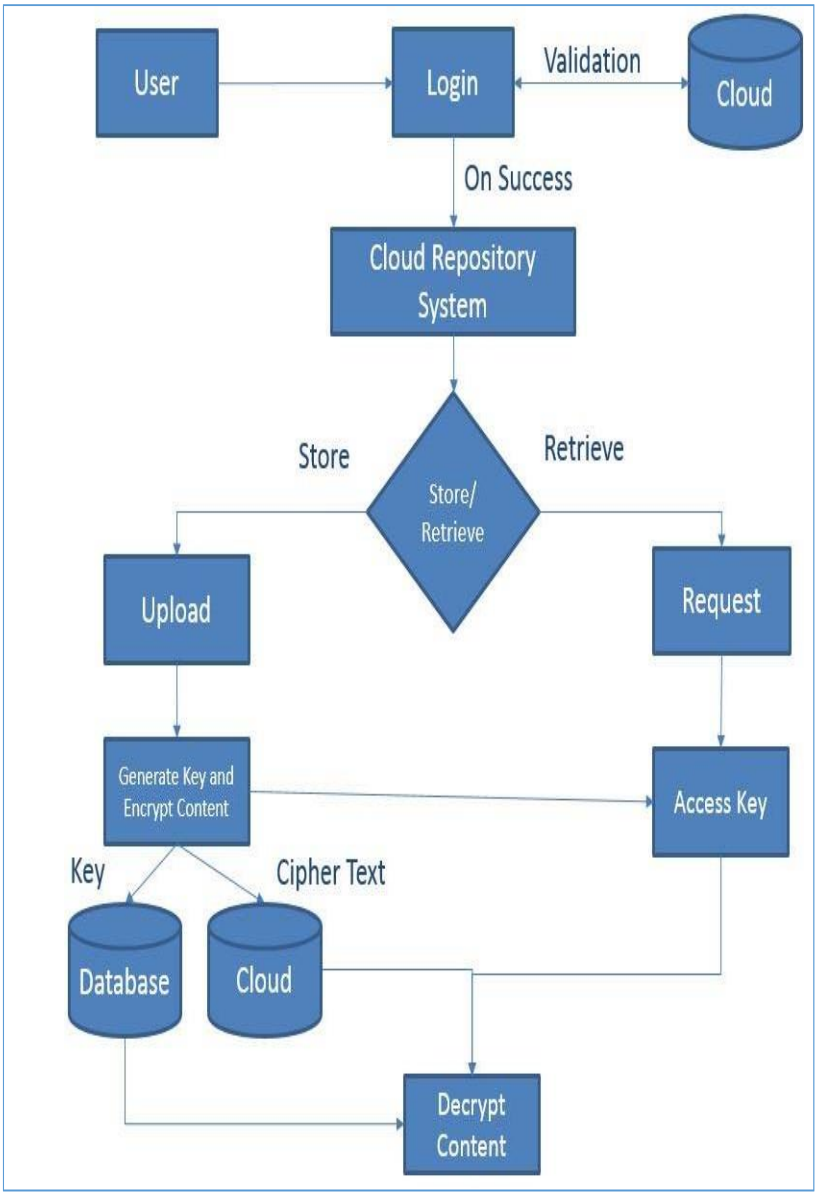

Fig -2: System Design

4. CONCLUSION

This project contributes to provide security to the data stored in the cloud, by encrypting the data before uploading into the cloud. As encryption consumes more processing overhead, many cloud service providers will have basic encryption applied only on few data fields. If cloud service providers can encrypt data, then cloud service can providers can decrypt encrypted data. To keep the cost low and maintain high sensitive data, it would be better to encrypt the data before uploading. In this project, we encrypt data using symmetric key encryption where private keys of the files will be stored in the local database.

REFERENCES

[1] G. T. Mell P, "The NIST definition of cloud computing," National Institute of Standards and Technology, U.S. Department of Commerce., 2012.

[2] "Understanding the cloud computing stack: SaaS, Paas, IaaS" Rackspace support, October 22, 2013

[3] E. Gorelik, "Comparison of Cloud Computing Service and Deployment Models," 2013

[4] C.-K. Chu, S. S. M. Chow, W.-G. Tzeng, J. Zhou and a. R. H. Deng, "Key- Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage," 2014.

[5] M. Stanley, "Cloud Computing Takes Off," Global Technology and, 2011.

[6] T.-S. Chou, "Security Threats on Cloud Computing Vulnerabilities," International Journal

[7] "Data security in cloud computing using encryption and stegnagraphy", Karun Handa et al, International Journal of Computer Science and Mobile Computing, Vol.4 Issue.5, May- 2015, pg. 786-791

[8] "A Survey of the Existing Security Issues in Cloud computing,"Parul Chachra / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (2) , 2014, 1066-1068

[9] "Enhanced Data Security Model for Cloud Computing",

[10] The 8th International Conference on Informatics and

[11] Systems (INFOS2012) - 14-16 May,Cloud and Mobile

[12] Computing Track.

[13] " A Proposed System Concept on Enhancing the Encryption and Decryption Method for Cloud Computing", 2015 17th UKSIM-AMSS International Conference on Modelling and Simulation.