International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

Securing Data and Providing Integrity over Cloud Storage

Chetan Patil[1], Rowan Arland[2], Aditya Kotian[3], Akash Panda[4]
[1]*Assistant Professor,*
[2, 3, 4] *Student, Computer Engineering Department, ST John College of Engineering and Management, Palghar, Maharahstra, India*

Abstract: ***Cloud computing is an important application for storage of data on cloud servers. In cloud storage data is moved to a remotely located cloud server over which users do not have any control. Users can store their data from distant places and have rights to cloud storage on demand. This project provides a scheme, which gives an assurance of data integrity, and security in the cloud storage, this data can be checked for correctness by comparing data with the original meta data.***
Keyword: Cloud Computing, Data integrity, AES, MD5.

I. INTRODUCTION

Data integrity means we can check accuracy and consistency of data. In general term data integrity means we can check if the data is illegally modified or not. In recent days data integrity is a term which is used with cloud storage. When we upload any file to cloud, at that time of uploading the file, if the file is hacked or modified then we use this technique because data integrity provides 100% assurance of our data. Cloud storage is visualized pools where data and applications are stored which are hosted by the third party. The main advantage of cloud storage is that reduction in storage cost. It means that we can avoid our local storage area like hard disk drive. Instead of loading our file in local storage we can store our file into the cloud storage area. After moving the data to the cloud, owner hopes that their data and applications are in secured manner.

But that hope may fail sometimes when the owner's data may be altered or deleted. This essentially means that the owner of the data moves its data to a third-party cloud storage server which is supposed to presumably for a free faithfully store the data with it and provide it back to the owner whenever required. In this paper we deal with the problem of data retrievability sometimes it is called as proof of retrievability. This problem tries to verify that the data which stored by user at remote data storage is not modified by archive in this way integrity of user's data is assured. As data generation is far outpacing data storage it proves costly for small firms to frequently update their hardware whenever additional data is created.

II. DRAWBACKS OF EXISTING SYSTEM

A. Large overhead on the server side because encryption of the entire block is done.
B. A TPA (Third Party Auditor) is involved who can alter the data that the user has uploaded.
C. Existing system uses random sampling sobol sequence method which gives results based on the probability of the given inputs.

III. PROPOSED SYSTEM

The proposed system will have encryption and decryption of the given file with the help of AES algorithm. To create a hash value of the file MD5 algorithm is used. MD5 creates a hash value of 32 characters. It is very useful for small scale project like this. In this project the whole file is not encrypted instead the file is break down into smaller blocks and one of that block is encrypted so this process minimizes the overhead on the server issue.

The architecture of the proposed system is explained below. In this architecture, user uploads the file F in the system.

The key generator produces a key 'k' and the file which is uploaded by the user F is encrypted using this key and the result is F'. The metadata of the encrypted file and the encrypted file is appended and encrypted. One copy of generated key is provided to the user for verification.

The final encrypted product is stored in archive or database of cloud server. This uploaded file can be further challenged to maintain security and integrity.

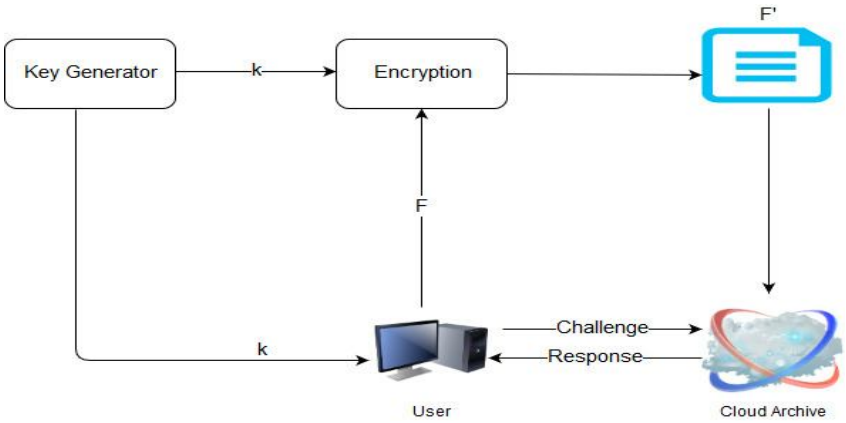

Figure 1: Architecture Diagram

IV. ALGORITHMS USED

This project focuses mainly towards integrity and security of data. For security purpose, AES algorithm is used. The file is broken down and encrypted. This encrypted file is appended with the meta data. Hashing function MD5 is used to generate hash value in the appended data. For integrity purpose, MD5 algorithm is used. It generates hash value which helps user to check whether the integrity of the uploaded file is compromised or not.

A. MD5 [5]:

Step 1: Append padded bits.
Step 2: Append length.
Step 3: Divide the input into 512-bit block.
Step 4: Initialize chaining variables.
Step 5: Output/Process blocks.

B. Aes [6]

1) *Key Expansions:* Round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate128-bit round key block for each round plus one more.
2) *Initial Round*: Add Round Key: each byte of the state is combined with a block of the round key using bitwise XOR.
3) *Rounds*
a) Sub Bytes: anon-linear substitution step where each byte is replaced with another according to a look up table
b) Shift Rows: a transposition step where the last three rows of the state are shifted cyclically a certain number of steps
c) Mix Columns: a mixing operation which operates on the columns of the state combining the four bytes in each column.

V. DESIGN

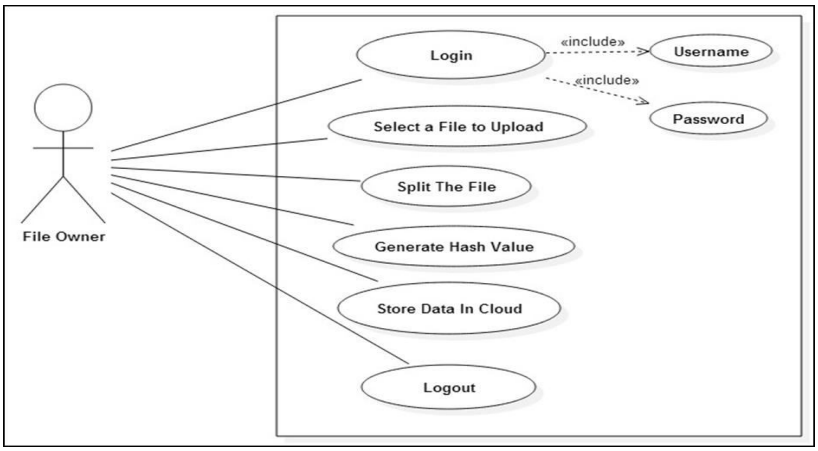

Figure 2: Use case for file owner.

The above figure shows that the file owner will select the file to be uploaded. Then the system will perform splitting of file in number of blocks specified by the owner. After that the hash value with the help of hash function would be calculated and stored along with the file on the cloud server.

The below figure shows that the user can download the file, if the user is an authenticated user. Non-authenticated user do not have access to the files on cloud.

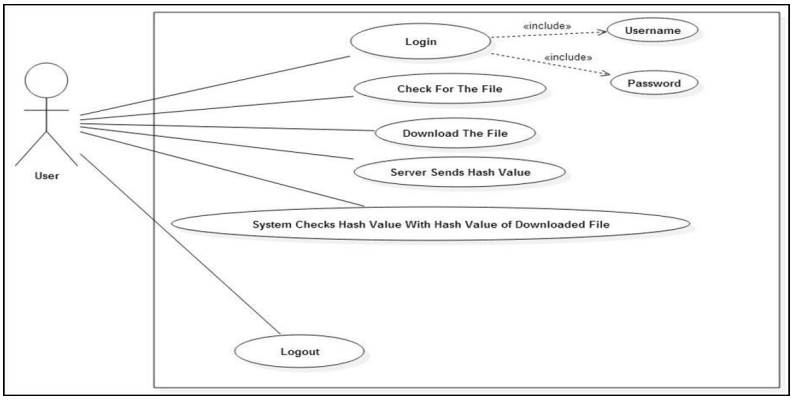

Figure 3: Use case for user.

VI. CONCLUSION

A. *In This Project we Have Observed*

1) The input file is broken down into two parts and each part is encrypted and stored on the cloud.

2) A secret key is generated during the encryption process performed by the AES algorithm and the hash value is generated using the MD5 algorithm.

3) As MD5 is used the hash value which is created is of 32 characters.

4) Data integrity and security is provided to static as well as dynamic data.

REFERENCES

[1] Dynamic Solutions Inc, 'different types of cloud computing, services offered by cloud computing' 2017. [Online]. Available: http://www.dynamixsolutions.com/what-are-the-different-types-of-services-offered-by-cloud-computing/. [Accessed: 29- Oct- 2017].

[2] S. Wang, D. Agrawal, A.E. Abbadi: A Comprehensive Framework for Secure Query Processing on Relational Data in the Cloud. Secure Data Management 2017.

[3] tutorialspoint, 'ASP.NET Tutorials' 2017. [Online]. Available: https://www.tutorialspoint.com/asp.net/. [Accessed: 01-Nov- 2017].

[4] Vic (J.R.) Winkler,"Securing the Cloud, Cloud Computer Security, Tech-niques and Tactics", Elsevier, 2016.

[5] Atul. Kahate, CRYPTOGRAPHY and NETWORK SECURITY. New Delhi: McGraw Hill Education, 2016.

[6] Behrouz A. Forouzan and Debdeep. Mukhopadhyay, CRYPTOGRAPHY AND NETWORK SECURITY. New Delhi: McGraw Hill Education, 2015.

[7] Sravan Kumar R, Ashutosh Saxena, "Data Integrity Proofs in Cloud Storage", 2015 Third International Conference on Communication Systems and Networks (COMSNETS 2015).

[8] Kartik Sharma, Gitesh Kumar, Parul Saluja, Prashant Dalal, Kapil Narwal, "A Secure Method of Dynamic Data Operation in cloud computing", International Journal of Advancements in Research & Technology, Volume 2, Issue 5, May-2013.

[9] MargaretRouse, 'cloudcomputing' [Online]. Available: http://searchcloudcomputing.techtarget.com/definition/cloud-computing [Accessed: 29- Oct- 2017].

[10] E. Mykletun, M. Narasimha, and G. Tsudik, "Authentication and integrity in outsourced databases," Trans. Storage, vol. 2, no. 2, pp. 107–138, 2006.

[11] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in SP '00: Proceedings of the 2000 IEEE Symposium on Security and Privacy. Washington, DC, USA: IEEE Computer Society, 2000, p. 44.

[12] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in CCS '07: Proceedings of the 14th ACM conference on Computer and communications security. New York, NY, USA: ACM, 2007, pp. 598–609.