iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 5 **Issue:** IV **Month of publication:** April 2017

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 5 Issue IV, April 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Secure and Recoverable Model to Manage Keys for Cloud Storage

Binay Prakash Tudu[1], Pradeep K.V [2]
[1,2]*The School of Computing Science and Engineering, VIT University, Chennai Campus, India*

Abstract*: Cloud computing is emerging concept which is growing day by day. Whether it is a IT industry or an enterprises or normal user, they are moving for cloud services. There are many cloud services such as computing, storage, web hosting, IOT, analytics, application and many more. The customer need not to have an infrastructure, but as per its need, the client can approach to cloud providers. As one of the service is storage where client can have their data. Usually data can be uploaded through secure connection but it is visible to cloud provide. The main criteria for a cloud consumer is its data security. Consumer need to be assured that their data need to be safe in cloud. Every consumer wants to have control over its data and its security. In this paper, we propose a management of the key such that the data is secure at the cloud service provider and the key is recoverable in case of losing the key. So, consumer is assured that even in the loss of the key, its data is recoverable.
Keywords: Encryption, cryptographic, CSP

I. INTRODUCTION

The amount of internet users, data transaction, and online services is growing day by day. Easy accessibility to the large and small data in this age of fast internet is a growing trend. Such kind of e-services at a large scale is being possible through cloud computing and main uses for cloud is for the storage of data or for computational of data at Cloud service provider (CSP). Another great advantage is its cost benefits with it. Therefore, cloud consumer is increasing day by day and its demands.

Most of the customer data being stored and computed at CSP as cloud ease the burden of data access and storage management. Thus, data become more vulnerable for attacks or chance of being misplaced. This data is sensitive and valuable to customer. Hence, there is rise of data security in cloud as CSP cannot be trusted for data.

For data security in cloud, it usually adapts cryptography approach as a solution. It makes data in cloud secure on CSP side. Having so many secure data at CSP let in the increase of key. Cryptographic keys are used for the data encryption and decryption. Thus keys play vital role for the data security. Various key managements are used to manage these keys. Keys can be either symmetric or asymmetric. There is various cloud platform such as public, private, hybrid and community cloud. Each platform has its own unique problems for managing those keys. Even the location of keys need to be addressed that is whether the key should be managed at client side or cloud side or at third party. Thus the need of managing keys arises.

As there increase in cloud consumer so there will be many keys that need to be managed. So, it is the job of the key management system (KMS) to provide this key and to maintain those keys. In the case of loss of key, it is the job of KMS to recover the key.

Thus KMS is responsible for creating, providing, maintaining, deleting, recovering of the keys. Various key management methods that can be adopted. This key management method is based on placing of the key on different locations. Key can be placed either at client side, CSP side or at third party. Management of the key can be at client, at CSP or both the client and CSP sides, splitting the key, key management at centralized server and group key management. There is various key management method based on various scenarios like public cloud, private cloud, hybrid cloud, home based cloud, online community cloud, outsourced community cloud.

In case of public cloud scenario based, the data is encrypted at the client side and kept in the cloud server. Fig 1 describes user U1, U2, U3, U4 uses the symmetric approach in public cloud scenario. The user takes the data from cloud and decrypt at its side. The key is managed at the client side. Since the keys are managed only at the client side, it is not fault tolerant. It means if the key is lost at client side than the key cannot be recovered.

NIST (National Institute of Standards and Technology) in 1997 wish to have a successor to DES. Thus a community was formed. Five algorithm were selected for the final list. They are Twofish, Serpent, Rijndael, RC6 and Mars [4]. Rijndael was selected as the official AES by NIST. All of them are symmetric-key algorithm that uses same cryptography key to encrypt and decrypt the data. Since AES is the most widely used. It is being used for the proposed work.

This paper proposes an effective solution for the above problem. The data is encrypted at client side and then split the key into

Volume 5 Issue IV, April 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

two sub keys. Each key is again obfuscated with different technique and it is store at the CSP. As the obfuscation method is not revealed to CSP, the data cannot be decrypted at CSP, hence data it secure. Even if the client losses its key. The key can be regenerated from the key present in the CSP, hence it is recoverable. Thus proposed solution maintains the integrity, availability and security of the client data and it is fault tolerant.

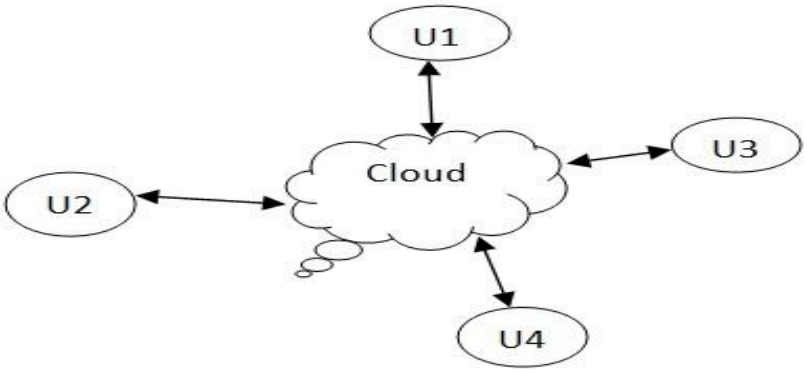


Fig. 1. public cloud scenario

II. METHODOLOGY

The proposed solution can be divided into two steps

A. Encrypting the Data

In this step the data is encrypted using a key at the client side.

B. Securing the Key

In this step the key used for the encryption is split into two sub-keys. Then this two sub keys are obfuscated using different technique and along with the encrypted data is stored at CSP.

C. Decrypting the Data

In this step the encrypted data is downloaded from the cloud. Apply the reverse mechanism of securing the key to get the normal data back.

III. MODULES OF THE PROPOSED WORK

To achieve the proposed method, the proposed work is divided into modules. Individual modules have a specific role to achieve the desired goal Fig 2.

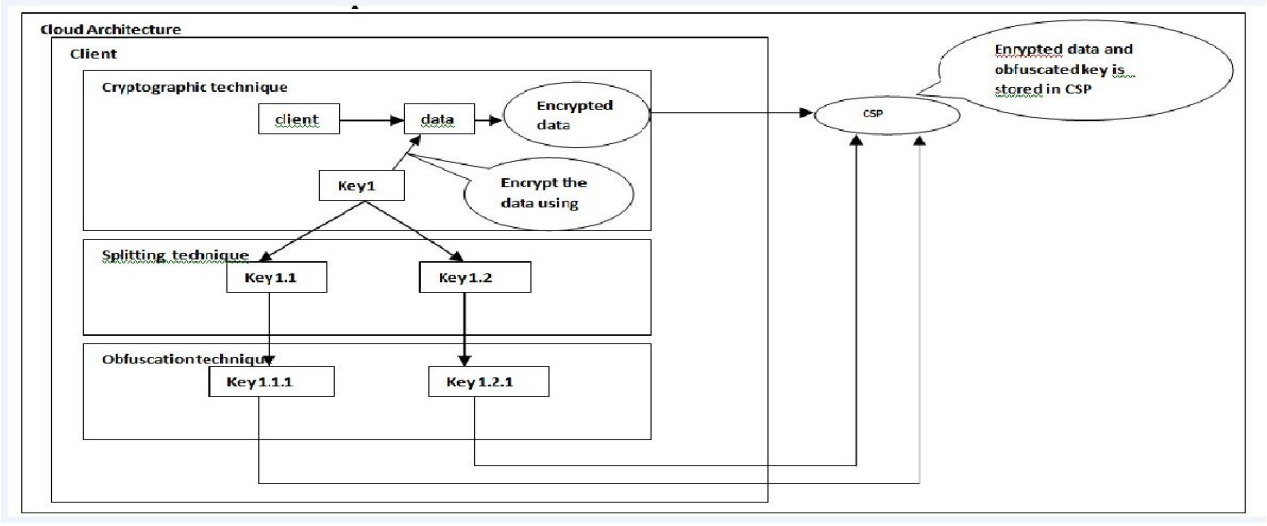


Fig. 2. Modules of proposed work

Volume 5 Issue IV, April 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

A. Cryptographic Module

In this module the data is encrypted using one of the advanced encryption standard technique known as AES [8]. It is a symmetric key algorithm. It is a block cipher with 128-bit block cipher that accepts a variable-length key of 128, 192 and 256 bits. AES main design principle is substitution and permutation network.

B. Splitting Module

In this module the key used for encryption is split into two sub keys. Shamir secret sharing technique used to split the keys into two sub keys. In Shamir secret sharing technique a secret is divided into n pieces and at least k pieces required to be merged to get the secret back. Its uses polynomial interpolation as a recovery. In this proposed method, the key is split into two.

C. Obfuscation Module

The split keys are again obfuscated using data obfuscation.

D. Cloud Module

Public cloud architecture is used to implement the proposed work. Both, the data and the split encrypted key is generated and stored in the public cloud.

IV. IMPLEMENTATION ARCHITECTURE

A. Cloud Architecture

Implementation is done based on public cloud scenario. In this case personal laptop is used as client and Amazon S3 in AWS is used as cloud storage. Internet is used to connect the laptop and Amazon S3. The module implemented is made into an application. This application need to be present in the client side. The encrypted data and the encrypted keys are stored in the cloud Fig 3.

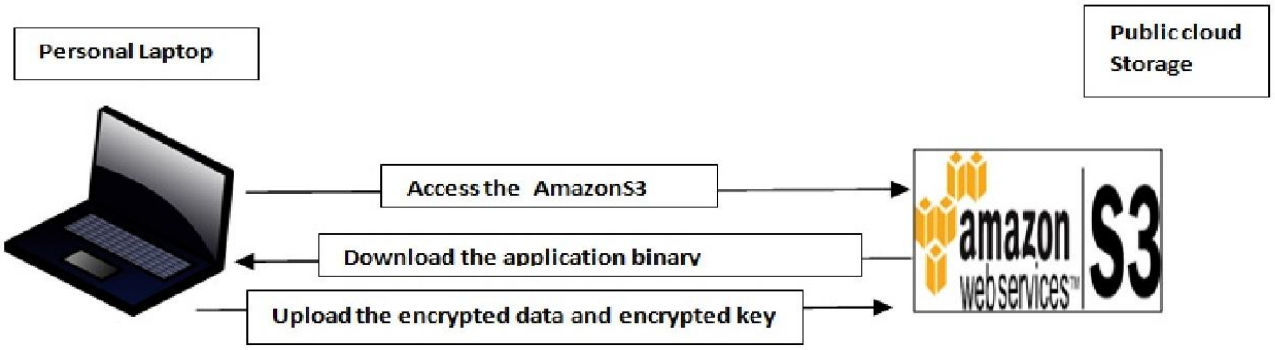


Fig. 3. Architecture

Above mentioned way is the basic way of doing it. Extend of it is that the application binary can be kept in the cloud storage. Amazon S3 is used for it.

The Advantage is that whenever a client wants to use it. Client can download the application binary and can perform the encryption. Once the work is over, delete the application. But, the generated key which should be in the client side must be kept safe. It is at client risk.

There is another need of making the application binary to be encrypted because binary are java classes which can be easily decoded by any java expert. Amazon S3 allows uploading objects (data) to be encrypted. There are two types of encryption provided by Amazon s3 using server side encryption or client side encryption. In this case Server side encryption is applied using the Amazon S3 service master key.

V. IDENTIFIED TOOLS

A. NetBeans

It is a Java-based integrated development environment (IDE), a software tool uses for free to develop applications.

VI. RESULTS

NetBeans tool is used to implement the proposed work. The code is developed in java. The AES encryption is implemented with the help of Bounty Castle a java package. Bounty Castle is collection of cryptographic API helps to implement the various cryptographic encryption. A key is chosen and encrypt the file containing the data. We get an encrypted data in the notepad.

Now the chosen key is split into two sub keys using Shamir secret sharing technique. Key is given as the input and we get two sub keys. The two sub keys are merged together and stored in file.

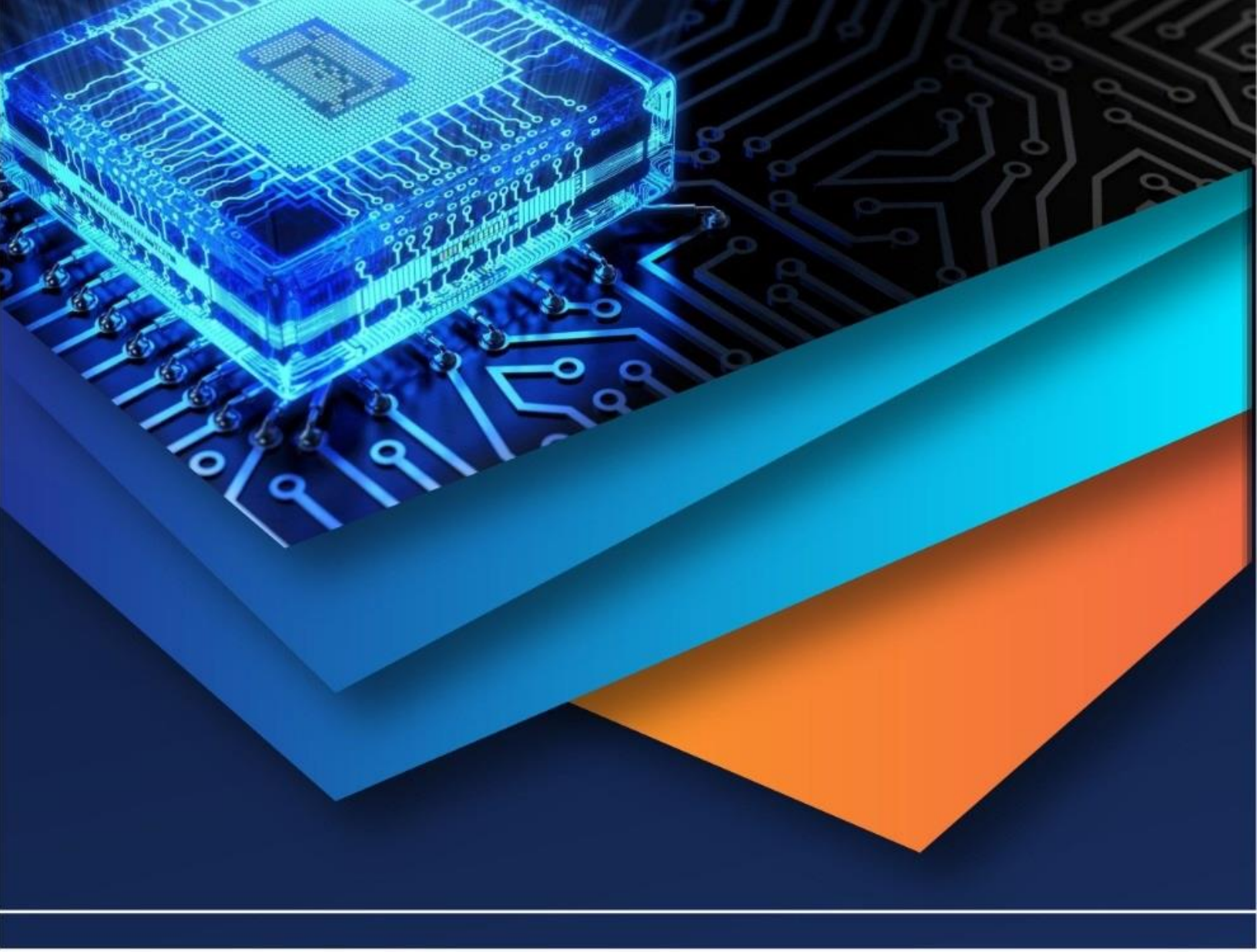
While decrypting in case the client loses its keys, the stored file and the keys are retrieved from amazon s3 and the merged keys are split and again split keys are merged together to get the original key. Now use the original key to decrypt the encrypted data. GUI based java application developed using NetBeans IDE. This application allows selecting a file and then encrypting using a key. Splits the key into two and again apply data obfuscation to it. The encrypted data along with the two keys are stored in the cloud storage. GUI based java application can be used to decrypt the file using those split key automatically.

VII. CONCLUSION

The proposed solution clear shows that the key is maintained at the client side and even if the client losses its key, still the client can retrieve the keys from the CSP. Thus make the proposed solution as fault tolerant. Since the keys at CSP cannot be decrypted makes the key secure, available and maintain the integrity. The proposed method ensures the client has more control on its data.

REFERENCES

[1] Amar R. Buchade and Rajesh Ingle, "Key Management for Cloud Storage: Methods and Comparisons," Fourth International Conference on Advanced Computing and Communication Technologies 2014.
[2] Marius Popa, "Techniques of Program Code Obfuscation for Secure Software," Journal of Mobile, Embedded and Distributed Systems, vol. III, no. 4, 2011.
[3] Bruce Schneider, John Kelsey, Doug Whiting, David Wagner, Chris Hall and Niels Ferguson, The Twofish encryption algorithm: A 128-bit block cipher: ISBN-471-35381-7 208 pages March,1999.
[4] Aarti Singh, "Study of MDS Matrix used in Twofish AES Algorithm and its VHDL Implementation," Central Electronic Engineering Research Institute in Delhi, India, 1997.
[5] Ramaswamy Chandramouli, Michaela Iorga and Santosh Chokhani, "Cryptographic Key Management Issues and Challenges in Cloud Services" Nist September 2013.
[6] Arshad Noor, "Symmetric Key Management Systems," February 2007 by ISSA Journal.
[7] S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, “A novel ultrathin elevated channel low-temperature poly-Si TFT,” IEEE Electron Device Lett., vol. 20, pp. 569–571, Nov. 1999.
[8] https://en.wikipedia.org/wiki/Advanced_Encryption_Standard_process.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY