International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Lightweight Security Framework for Data Outsourcing and Storage in Mobile Cloud Computing

Manzamasso Kpelou, Keshav Kishore

Abstract: ***Mobile Cloud computing is define as the new paradigm, and in this new paradigm computational power is on demand service and accessible through mobiles and tablets. When user decides to go towards cloud computing, there is a risk of losing control on his own data. So providing security of the data during transmission and when saved on the cloud storage is a major issue. Any application relying upon an emerging technology should consider the different possible threats. The principal reason why people are afraid of using Cloud technology is that data security and integrity among all other issues involved in Cloud technology constitute a great challenge. As data security issues, access control as well as key management are also a reason why Cloud technology is not widely adopted. In mobile cloud computing, the security related to data involving data integrity, confidentiality, availability and traceability is the most critical concern of cloud users. As now mobile cloud is changing how users used to work over the network, by helping them in terms of cost and reduction of tasks complexity. Hence, it is very important for the user to be ensured that the transmission of the data is secure, that the consistency as well as the storage of the data on the cloud provider side is secured. Hence the necessity for developing trust security model is demanding.***

Keywords **: Mobile Cloud Computing, Cloud Computing, Data Encryption, Mobile Cloud Security.**

I. INTRODUCTION

1.1 Mobile Cloud Computing (MCC)

MCC is a method, also an environment in which applications especially mobile application are created and managed using cloud technology.

With the mobile cloud perspective, developers can design applications for users with unlimited computing power and memory capacity which lack in smartphone. MCC services are widely accessed via a mobile browser from a remote web server, and this is done without the need for installing a user's side application.

Multiple mobile applications need a lot of computational power and runtime machine for application execution. Various Mobile phones are not predisposed to support this new kind of applications. Since the computational power, unlimited storage and platform support needed to execute these application are brought by MCC that makes a high number of mobiles devices to be supported.

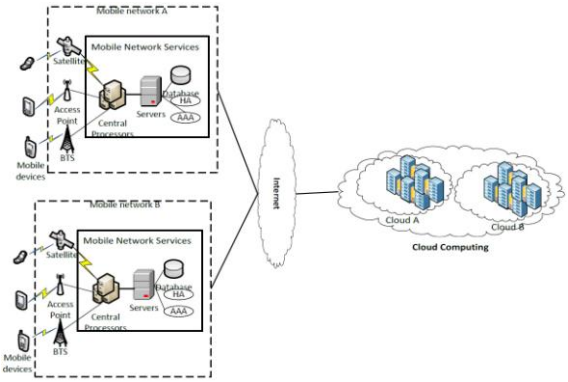

Fig.1 Mobile cloud architecture [1]

1.2 Difference between cloud computing and mobile computing

Cloud computing refers to a new technologies which allow its users to access, execute and manage application stored in remote location [2]. Cloud service vendors can serve multiple clients in the same time. They allow their clients access to their data from a private network, to the cloud storage space and data backup systems. Hence the cloud service provider can intake files sent by users and save them securely, while providing back cloud services to users by internet connections.

Mobile computing refers to the introduction of new lightweight devices and systems. Mobile devices such as tablets or smartphones can now perform complex tasks and operations almost at the same level as traditional desktop. Mobile computing functions comprise:

1- Supporting various software applications,
2- Accessing the Internet through browsers,
3- And sending and receiving various data types [2].

The mobile operating system furnish, a familiar and common search technologies and easy touch-screen commands, with easy to use interface and intuitive icons for users supports.

While mobile computing is a new technology with reticent users due to various security issues, cloud computing is used by many businesses and companies.

Revised Manuscript Received on 30 July 2019.

***** Correspondence Author

Manzamasso KPELOU, is currently student of Computer Science & Engineering at Alakh Prakash Goyal Shimla University, Shimla.

Keshav Kishore is currently working as Faculty of Computer Science & Engineering at Alakh Prakash Goyal Shimla University, Shimla.

Cloud computing may be beneficial to individuals, however various and complex cloud computing services are designed for businesses. For instance, Large enterprises as well as smaller ones operations use specific cloud based tasks or services to make various processes like inventory handling, customer relationships and moreover the overall production more efficient. MCC comes along with the emergence of smart phones and mobile tablet which support new operation system as well as new networking services that are used both with MCC and Cloud computing.

1.3 Data security challenge in MCC

The most serious concern of mobile cloud users is how to ensure their data security and privacy when transferring and saving them in the cloud environment. Data security is then one of the principal challenges in MCC environment. Here are some issues related to security in mobile cloud computing: data privacy, data ownership and other security issues.

1.3.1 Data privacy

Data privacy also called information privacy is state as one of the major challenge in MCC environment. It tells about how data is collected, shared and used. It states if or how the data can be shared with third parties, if data can be legally collected or stored. Many questions are raised by users:

- How files are created and the back-up is done
- What happen when user delete his files
- Who can access the data
- Where the data are located

1.3.2 Confidentiality of data

Data confidentiality is related to data privacy and ensures data is visible to only authorized user. It refers to the protection of information from being accessed by unauthorized persons. Access to data must be restricted to only those authorized to view the data. Many questions are raised by user:

- Where the encryption and decryption processes are taking place?
- confusion because equations do not balance dimensionally. If you What are the menaces when transferring data client to cloud?
- How to perform operations like search on an encrypted form

1.3.3 Data integrity

Data integrity involves maintaining the accuracy, consistency, and trustworthiness of data its life cycle. Integrity of data means data must not change while being transferred and must not be altered by unauthorized persons. From the time the data is uploaded to time the data is download through the time the data spent on cloud storage, there must not be any change or alteration of the data .Data integrity can be guaranteed with access controls and permission on data.

1.3.4 Data ownership

Despite the advantages of cloud services, a person must answer the most important question when going to upload his data on cloud, which is “who owns the data”. Sometime when a user decide to download his data or delete them, is that mean the cloud hosted service doesn’t make a copy of those data? The situation in which the hosted service keep some information without the permission of the owner, the user are not any more the data owner.

1.3.5 Other security issues

Here is other security issues like that affect data security in MCC: Denial of Service, Side Channel, Man-in-the-middle and Authentication attacks.

It is insecure to transfer important data to the cloud because there are common concerns such like [3]:

- ✓ Handling of encryption and decryption keys
- ✓ Violation of privacy rights
- ✓ Risk of data theft
- ✓ Absence of a data integrity standard
- ✓ Presence of different vendors leading to services incompatibility.

To incent and attract more users to adopt Mobile Cloud Computing, it is important to protect the data from the various security concerns.

1.4 Data security requirement in MCC

Data encryption in the cloud consist of transforming or encoding data before being uploaded to cloud storage. With cloud services, there are two forms of encryption: “Transit” encryption and “resting” encryption. We talk about “Transit” encryption when data uploaded are being transferred between both the user and the cloud service using Secure Sockets Layer (SSL). When talking about “Resting” encryption, we refer to the encryption form in which data are stored in their encrypted form on the cloud storage. Most of the time cloud service vendors offer various encryption services – encryption keys to decrypt the data as needed – and an encrypted connection to limited encryption of sensitive data [4].

But researcher in their work in order to provide data security and privacy have proposed various framework in which the encryption of data no longer needed to be provided by a third party.Encryption of the data is the top priority for Mobile Cloud Computing in term of data security and privacy. And there is some minimum requirement that involved in the process on encryption.

1.4.1 Data must be encrypted before being upload to the cloud

Encryption of data before being uploaded to the cloud is one the best practice to execute which ensure the data security. Data needs to be protected from unauthorized access and transmitted to the intended receiver with confidentiality and integrity [5]. Data sent in clear through the transmission media can be intercept by adversaries. Sensitive data intercept by a malicious person with attacks such as Man-In-The-Middle are avoid using encryption techniques on data before sending it to the cloud storage.

1.4.2 The choice of encryption algorithm for mcc

Many algorithms such like DES, AES, Blowfish, RC2, 3DES, and RC6 can be used to ensure data encryption in mobile cloud computing. Those algorithms can be from Traditional symmetric or asymmetric encryption algorithms. There are many disadvantages for symmetric encryption algorithms like key maintenance and there are also many drawbacks of asymmetric encryption algorithms regarding the consumption of computing resources such as CPU time, memory, and battery. The choice of the encryption algorithm must be suitable for the type of data and also based on how faster and secure it can be.

1.4.3 A secure communication channel

After the data being encrypted using an encryption algorithm and ready to be sent on the cloud provider storage space, a secure channel will ensure data security during the transmission. A secure communication channel appear then as a reliable requirement to ensure data security in Mobile Cloud Environment. Transport Layer Security (TLS) and Secure Sockets Layer (SSL) which is its predecessor, are both cryptographic protocols that furnish secure communications on the Internet for e-mail, instant messaging, Internet faxing, web browsing and other data transfers [6].

1.5 Mobile Cloud Computing Security Models

This are the three categories for mobile cloud computing security on which the existing models are based on:

- Authentication based models

Also called Identity Base Encryption (IBE) it is a public-key cryptosystem where any string can be chosen as a valid public key. In particular, email addresses and dates can be public keys [7].

- Data Access Models

This model secure data access cryptographic schemes, secure resource and allocation methods, data privacy preserving approach, secure network channels.

- Location based security models

Data security is mapped to a specific location so that data can't be accessed from another area.

II. Related Work

Mohd Rizuan Baharon et al. [8] Proposed a New Lightweight Homomorphic Encryption (LHE) which reduces the overall process time and power applied for encryption and key generation. In the paper, the authors deeply look at homomorphic encryption efficiency and proposed the LHE since the primitive encryption schemas can't be used to ensure a better data security in cloud because they don't support operations on encrypted data and such encryption schemas will expose data to cloud. According to the authors, computation on encoded data can be performed by Homomorphic Encryption which make it most likely to be adopted by cloud users. But Homomorphic Encryption is not widely implement in cloud architecture due to its low performance.

Syam Kumar Pasupuleti et al. [9] Use probabilistic public key encoding technique for data Encryption. For data retrieving from the cloud, a keyword search algorithm is applied on the encrypted data. Unlike several keyword searchable encryption schema, that approach is used to encrypt data efficiently as well as ensuring the data privacy. Data is protected against privacy violations, and its integrity is also verified.

Yinghui Zhang et al. [10] proposed a new security schema: match-then-decrypt that enhance the attribute-based Encryption (ABE), in which a particular data requested by a user is first found before the decryption process is applied. In that security schema, some security keywords are embedded in the cyphertexts and when there is a match when a request is made, the specific cyphertext is decrypted and retrieved. This technique provide a faster decryption and its applications in mobile cloud computing will benefit users.

Jiang Zhang et al. [11] proposed an efficient data distribution methodology applicable in MCC which ensure a secure data storage and data sharing for cloud users. That methodology leverage many primitive encoding schemas to create a better, efficient and secure data distribution system in mobile cloud. They also use the BLS signature to ensure data authentication and integrity. The combination of BLS signature and Merkle hash tree (MHT) help the data owner to perform various operations such as data modification and data deletion dynamically on his data and help data users to verify the data owner's identity. The distribution system required no trusted third party and this give a total control to users on their data.

Mehdi Bahrami et al. [12] suggested a new light-weight technique for mobile cloud that allow clients, with the derived pseudo-random permutation from chaos system, to save files on one or more cloud storage platforms. The presented method is evaluated with a case study using JPEG image format. The proposed method is compared to other encryption schemas like AES, pixel and colour encryptions based on their performance parameter. The computation is done on the user's side to avoid a trusted third party and to maintain user's privacy.

Table-I: Literature Review inference

PAPER TITLE / PARAMETERS	symetric	Asymetric	Key lenght	Block size	Number of roundS	Complexity	performance	Type of attacks	Encryption on the user side	Computation on encrypted data	Data compression	Data fragmentation	Algorithm used
a new lightweight homomorphic encription scheme for mobile cloud compution	**no**	**yes**	**-**	**-**	**-**	**-**	**-**	**-**	**yes**	**yes**	**no**	**no**	**RSA**
An efficient and secure privacy-preserving approach for outsourced data of resource constrained mobile devices in cloud computing	**no**	**yes**	**-**	**-**	**-**	**-**	**-**	**-**	**yes**	**no**	**no**	**no**	**RSA, SHA-1**
Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing	**no**	**yes**	**-**	**-**	**-**	**-**	**-**	**-**	**Yes**	**yes**	**no**	**no**	**ABE,AES**
Towards Secure Data Distribution Systems in Mobile Cloud Computing	**no**	**yes**	**128**	**128**	**-**	**-**	**-**	**-**	**yes**	**no**	**no**	**no**	**AES, SHA-256**
A Light-Weight Permutation Based Method for Data Privacy in Mobile Cloud Computing	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**brute Force**	**yes**	**no**	**no**	**yes**	**chaos system**
Encryption as a service for securing data in mobile cloud computing	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**no**	**no**	**no**	**no**	**RSA, ElGamal Cryptosystem, homomorphic encryption, Elliptic Curve**
Intelligent cryptography approach for secure distributed big data storage in cloud computing	**yes**	**no**	**-**	**-**	**-**	**-**	**-**	**-**	**yes**	**no**	**yes**	**yes**	**Alternative Data Distribution, Secure Efficient Data Distributions, Efficient Data Conflation**
Fine-grained data sharing in cloud computing for mobile devices	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**yes**	**no**	**no**	**no**	**Attribute-based encryption, ElGamal Cryptosystem**
Engineering searchable encryption of mobile cloud networks: when QoE meets QoP	**yes**	**no**	**-**	**-**	**-**	**-**	**-**	**-**	**yes**	**yes**	**no**	**no**	**-**
Secure framework for data access using Location based service in Mobile Cloud Computing	**yes**	**yes**	**-**	**-**	**-**	**-**	**-**	**-**	**no**	**no**	**no**	**no**	**AES, RSA**
Homomorphic Encryption in Mobile Multi Cloud Computing	**no**	**yes**	**-**	**-**	**-**	**-**	**-**	**-**	**yes**	**yes**	**no**	**no**	**homomorphic encryption,**

III. PROPOSED METHODOLOGY

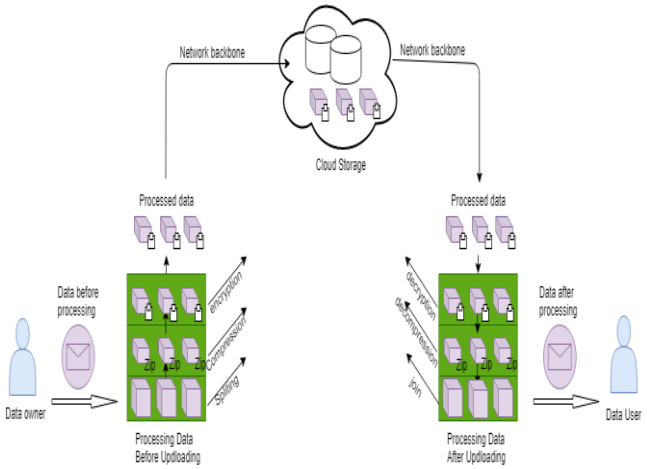

Fig.2 Proposed security system architecture

The experiment is divided into 3 parts:

- Splitting of data to be transferred into chunks
- Applying a compression algorithm on each chunks
- Encryption of each compressed chunk

A. Algorithm Used

Following are the algorithms that are used in the proposed work.

✓ **Split and join algorithm**

Data splitting is a technique of protecting sensitive data from non-authorized persons. It consist of dividing the data into smaller parts which can be store on different servers. A non-authorized person would need to know the different pieces that compose the original data and how to combine them.

Various data type can split and joined. The file types can be Image, Text and PDF, Video. The algorithms used to perform these operations are depicted as follows:

SPLIT-

1. Enter the file path
2. Enter the number of bytes per chunk (chunksize)
3. Calculate the number of chunks to be split in (numsplit)
4. Set the maximum buffer size (maxbufsize)
5. Create a new list of chunk (chunklist)
6. Open buffered output stream
7. Set temp = 0
8. For 1 to total number of chunks

Read chunksize in original file from temp
Create new chunk
Add chunk to chunklist
Temp = temp + chunksize

9. Close original file.

MERGE-

1. Enter chunks directory path
2. Enter the chunklist
3. Create a bufferedwriter specifying the name of the final file
4. For all the chunk in chunklist

File = file + chunk

5. Close writer.

✓ **GZIP Algorithm**

GZIP is a file format, but it is also considered as a compression algorithm. It was created by Phil Katz for the PK zip archiving tool and is used for various file types such as PDF, HTTP, PNG and other files.

So, GZIP is an association of two algorithms, LZ77 and Huffman coding. The working process of GZIP is explained below:

- **LZ77 Compression Algorithm**

The LZ77 encoding Algorithm is used to reduce the input data size by replacing redundant information with metadata based on the information obtain from the analysis of the input data [27]. Data blocks are encoded and when in the encoding process some data blocks are similar to the already encoded data blocks, they are replaced by a few quantity of metadata that shows how to convert them back in their original form.

To use the LZ77 Compression Algorithm:

1. Set the coding position in the input stream beginning.
2. Search for the longest match in the define buffer for the lookahead buffer (LK) which is the byte sequence from the coding position to the end of the input stream.
3. If there is match, write the pointer P and move the coding place with the window L bytes forward.
4. If there is match, throw a null pointer and add the initial byte in the LK. Move the coding place with the window one byte forward.
5. If the LK is not empty then restart from step 2.

- **Huffman Encoding**

Huffman encoding is a greedy algorithm that is used for lossless compression data and most suitable for text compression. It uses variable length encoding in which variable length codes are affected to all the characters based on how frequently they appear in the given text file [28]. The most frequent character to occur is assigned the smallest code and the less frequent is assigned the largest code.

Huffman encoding, to be sure that the code affected to any character is not used for any other character prefix implements a rule called: Prefix Rule.

There exist two important steps in Huffman coding-

1. Creating a Huffman tree based on the input characters
2. Affecting codes, by traversing the Huffman tree, to the characters

Creating a Huffman Tree-

1) Generate a leaf node for all unique characters with the appearing number of each character.
2) Based on the frequency value of each node, order all the nodes in the growing order
3) Select the first and the second nodes with the less frequency and generate a new internal node and get it frequency by summing the frequencies of both nodes. Set the first chosen node as the left child and make the second node as the right child of the node which was created earlier.
4) Repeat Step-2 and Step-3 until it remains only one node at the end. The last node is the root node and the

Huffman tree is then complete.

Steps to assign code to Huffman tree-

1) The Huffman tree is traversed and weights are assign to all the edges.
2) Weight '0' is assign to the left edges and weight '1' to the right edges.

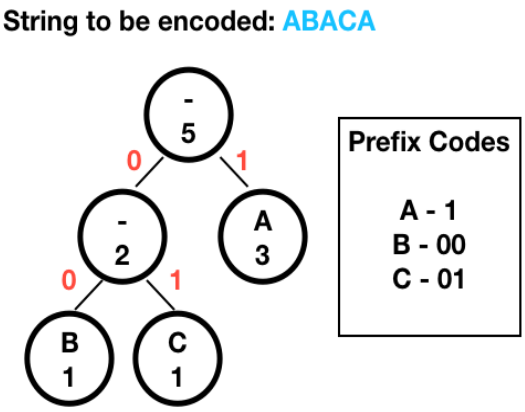

Fig.3 Huffman Encoding Tree [25]

✓ **AES with RSA**

RSA for file encryption can only support a limited buffer size which is set for a maximum of 245 bytes.

To overcome this buffer size constraint, it is possible to involve a symmetric algorithm for the encryption part while the symmetric algorithm secret key is encrypted by RSA. In this case, the symmetric algorithm used is AES. Using AES encryption involve sending the initialization vector (IV) to the receiver which is indispensable for the message decryption [28].

1. initiate the AES Key
2. Instantiate the RSA Private Key
3. store the AES Key
4. Write the Initialization Vector
5. Encoding the file data using the AES Key
6. Decoding the file data with the RSA Public Key
7. Load the RSA Public Key from File
8. Load the AES Secret Key
9. Read the Initialization Vector
10. Decrypt the file Contents

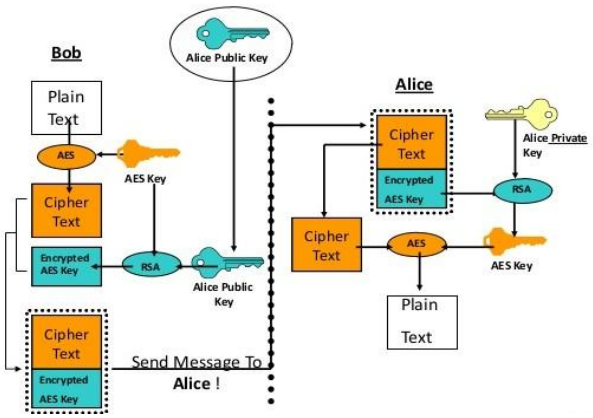

Fig.4 RSA with AES [29]

B. Proposed Approach Flow chart

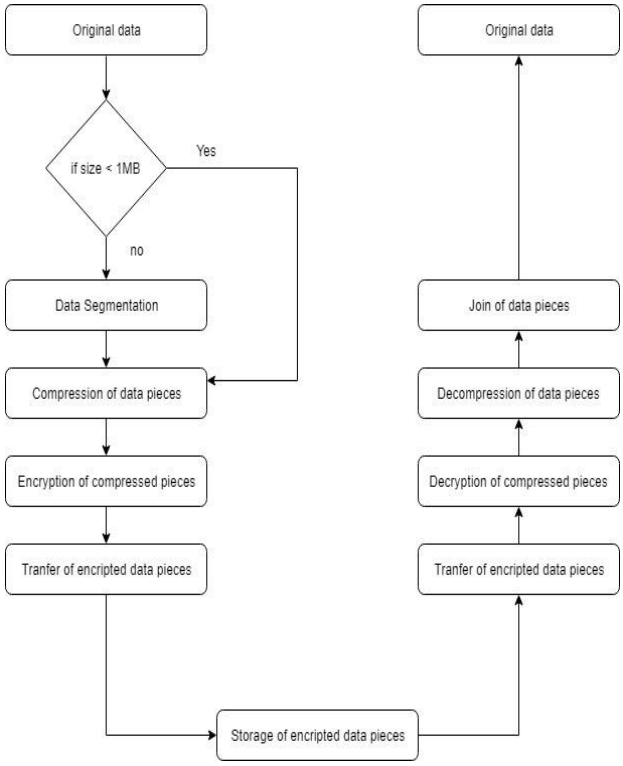

Fig.5 Proposed Methodology flow chart

IV. EXPERIMENT RESULT AND DISCUSSION

For our experiment, we apply our algorithm on a video "test.AVI" of size 350 Mb. The execution time was 35s 851ms and the output was 350 data chunks (compressed and encrypted) with individual size less than 1 MB.

A. Server Information Screen Shot

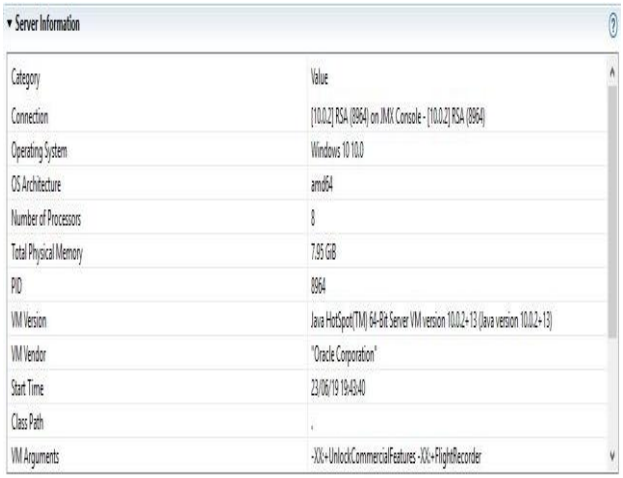

Fig.6 Server Information

Fig.6.1 shows the configuration of the Server on which the experiment is run. The algorithm applied on the video was run on a server with the above configuration.

B. Execution Time Screen Shot

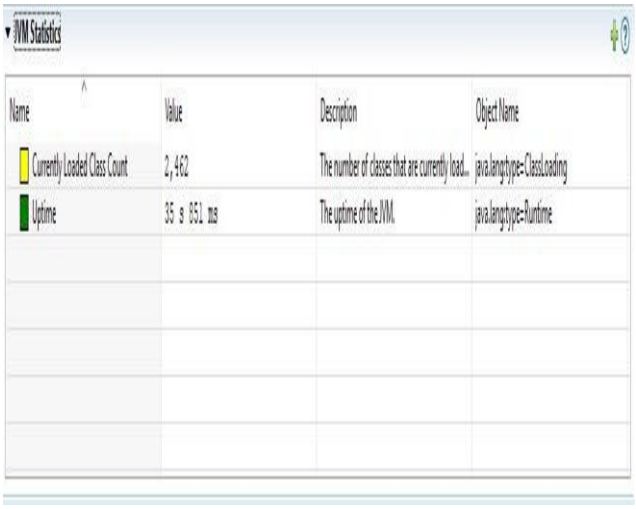

Fig.7 Execution Time

Fig.6.2 shows the required time needed to apply the algorithm on the test video and get the output. The overall execution time as mentioned in the second line table shown in the figure is 35s 851ms.

CPU Usage Screen Shot

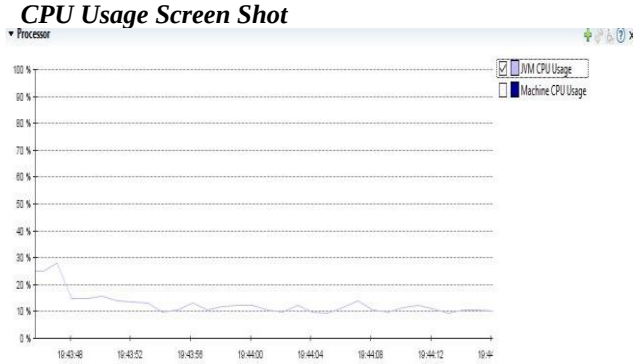

Fig.8 CPU Usage

Fig.6.3 shows the percentage of CPU Used to run the algorithm. The average CPU usage during the execution time is 10%.

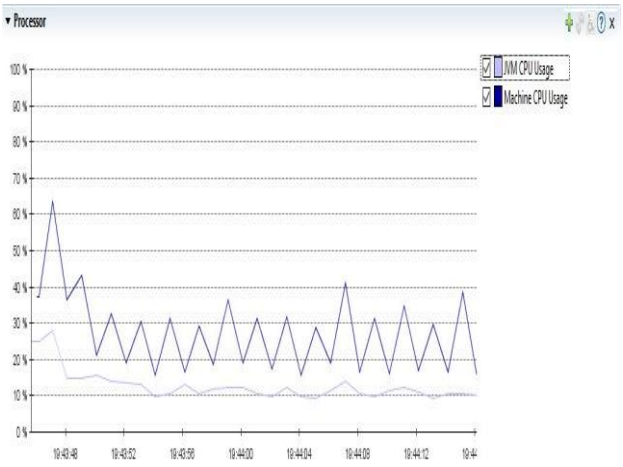

Fig.9 JVM CPU Usage Vs Machine CPU Usage

Fig.6.4 shows the percentage of CPU Used to run the algorithm and the percentage of CPU Used by the Machine.

C. Memory Usage Screen Shot

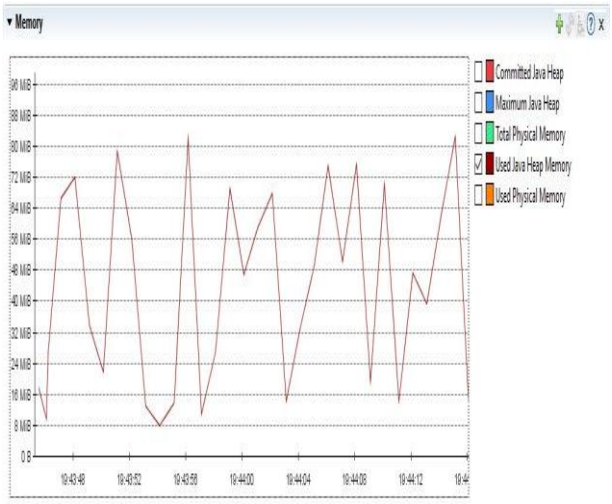

Fig.10 Memory Usage

Fig.6.5 shows the amount of memory used by the algorithm which is maximum around 80 MB.

V. CONCLUSION AND FUTUREWORK

In this thesis we discussed various schemas used for data security and privacy in mobile cloud Environment. Those security schemas are meant to protect users' data privacy, integrity and confidentiality. As we studied various papers, we tend to highlight the advantages and weaknesses of each security schema. And based on our analysis of this related works, we proposed a novel security framework which ensure a better security and privacy of user's data in MCC environment. The data in the process is fragmented, compressed, encrypted and the output is uploaded and each step consist of a security level as a split data can't be read nor a compressed data as well as an encrypted data. All this processes are operated on the user side, Each level of security required a specific algorithm, and this require no third party to ensure the data security and give the user a total control on his data.

In future we can enhance this work by:

- Create a Cloud based application for data sharing between users.
- Create a Microservices or APIs which can be integrated to existing data sharing model or applications.
- Implementing it in a popular and standard transfer protocol.

REFERENCES

1. https://en.wikipedia.org/wiki/Mobile_cloud_computing, last access on 31/05/2019
2. https://www.techopedia.com/7/29697/technology-trends/what-is-the-difference-between-cloud-computing-and-mobile-computing, last access on 31/05/2019
3. https://blog.appknox.com/security-challenges-in-mobile-cloud-computing/, last access on 31/05/2019
4. https://www.agileit.com/news/data-encryption-methods-secure-cloud/, last access on 31/05/2019
5. Zoran Hercigonja et al, "Comparative Analysis of Cryptographic Algorithms." International Journal of DIGITAL TECHNOLOGY & ECONOMY Volume 1, Number 2, 2016.

6. https://en.wikipedia.org/wiki/Secure_transmission, last access on 31/05/2019
7. https://crypto.stanford.edu/ibe/ , last access on 31/05/2019
8. Baharon M. R., Shi Q. et al. “A New Lightweight Homomorphic Encryption Scheme for Mobile Cloud Computing.” IEEE, 2015 International Conference on Computer and Information Technology. IEEE, 2015.
9. Pasupuleti S. K., Ramalingam S. et al. “An efficient and secure privacy-preserving approach for outsourced data of resource constrained mobile devices in cloud computing.” Journal of Network and Computer Applications 64 (2016): 12–22.
10. Zhang, Y., Chen, X., Li, J., et al. “Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing.” Information Sciences 379 (2017): 42–61.
11. Zhang J., Zhang Z. et al. “Towards Secure Data Distribution Systems in Mobile Cloud Computing.” IEEE Transactions on Mobile Computing 16(11): 3222–3235.
12. Bahrami M. and Singhal M. “A Light-Weight Permutation Based Method for Data Privacy in Mobile Cloud Computing.” 2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering. IEEE, 2015.
13. Mouhib I., Driss E. O. et al. “Encryption as a service for securing data in mobile cloud computing.” 2015 15th International Conference on Intelligent Systems Design and Applications (ISDA).
14. Li Y., Gai K. et al. “Intelligent cryptography approach for secure distributed big data storage in cloud computing.” Information Sciences 387 (2017): 103–115.
15. Shao J., Lu R. and Lin X. “Fine-grained data sharing in cloud computing for mobile devices.” 2015 IEEE Conference on Computer Communications (INFOCOM). IEEE, 2015.
16. Li H., Liu D., Dai Y. and al. “Engineering searchable encryption of mobile cloud networks: when QoE meets QoP.” IEEE Wireless Communications 22(2015): 74–80.
17. Goyal D. and Krishna, M. B. “Secure framework for data access using Location based service in Mobile Cloud Computing.” 2015 Annual IEEE India Conference (INDICON). IEEE, 2015.
18. Louk, M., and Lim, H. “Homomorphic encryption in mobile multi cloud computing.” 2015 International Conference on Information Networking (ICOIN).
19. Stergiou, Christos and al. "Secure integration of IoT and cloud computing." Future Generation Computer Systems 78 (2018): 964-975.
20. Malik, Ahmad Kamran, et al. "Rule Adaptation in Collaborative Working Environments using RBAC Model." International Journal of AdvancedComputer Science and Applications 8 (2017): 452-457.
21. Premarathne, Uthpala Subodhani, et al. "Hybrid Cryptographic Access Control for Cloud based Electronic Health Records Systems." IEEE CLOUD COMPUTING (2017): 1-7.
22. [22] Zhang, Peng, and al. "A Survey on Access Control in Fog Computing." IEEE Communications Magazine 56 (2018): 144-149.
23. https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-wusp/fb98aa28-5cd7-407f-8869-a6cef1ff1ccb, last access on 31/05/2019
24. https://www.gatevidyalay.com/huffman-coding-huffman-encoding/ (40)
25. https://cdn.journaldev.com/wp-content/uploads/2018/09/huffman-coding-algorithm-prefix-codes.png, last access on 31/05/2019
26. https://www.ijana.in/Special%20Issue/C36.pdf , last access on 31/05/2019
27. https://www.researchgate.net/profile/Ahmed_Wadday2/publication/324796235/figure/fig1/AS:619919657926656@1524811772120/Block-diagram-for-AES-encryption-and-decryption.png , last access on 31/05/2019
28. https://www.novixys.com/blog/using-aes-rsa-file-encryption-decryption-java/#3_Generating_the_AES_Key , last access on 31/05/2019
29. https://image.slidesharecdn.comcissp-d5-cryptographyv2012-minicoursev2-140127114251-phpapp0195cissp-d5cryptography-v2012mini-coursev2-79-638.jpgcb=1390823113, last access on 31/05/2019
30. https://blog.algorithmia.com/introduction-to-microservices/, last access on 31/05/2019
31. https://www.techopedia.com/definition/5594/java-development-kit-jdk, last access on 31/05/2019
32. https://en.wikipedia.org/wiki/Software_development_kit, last access on 31/05/2019
33. https://www.goodfirms.co/glossary/software-development-kit-sdk/, last access on 31/05/2019
34. https://www.superwits.com/library/cloudsim-simulation-framework, last access on 31/05/2019
35. https://slogix.in/what-are-the-basic-components-and-features-of-cloudsim, last access on 31/05/2019
36. https://www.techpaste.com/2011/08/advanced-java-program-split-join-file-upload-size-slices/, last access on 31/05/2019
37. https://www.journaldev.com/966/java-gzip-example-compress-decompress-file, last access on 31/05/2019
38. https://github.com/jaysridhar/java-stuff/blob/master/source/rsa-encryption/src/main/java/sample/sample1.java, last access on 31/05/2019
39. Shakeeba S. Khan et al. International Journal of Computer Science and Mobile Computing 3 (2014): 517-525.
40. David, S., Xavier, B., and al. “A panoramic overview on fast encryption techniques for outsourced data in mobile cloud computing environment.” 2017 International Conference on Inventive Computing and Informatics (ICICI).
41. Gai K., Qiu M., Thuraisingham B. and Tao, L. “Proactive Attribute-based Secure Data Schema for Mobile Cloud in Financial Industry.” 2015 IEEE 17th International Conference on High Performance Computing and Communications. IEEE, 2015.
42. https://www.enlume.com/mobile-data-security/, last access on 31/05/2019
43. https://greengarageblog.org/8-pros-and-cons-of-asymmetric-encryption, last access on 31/05/2019
44. https://whatis.techtarget.com/definition/Confidentiality-integrity-and-availability-CIA, last access on 31/05/2019
http://www.(URL)

AUTHORS PROFILE

Manzamasso KPELOU, is currently student of Computer Science & Engineering at Alakh Prakash Goyal Shimla University, Shimla. He is having keen interests in Mobile Cloud Computing and Data security.

Keshav Kishore is currently working as Faculty of Computer Science & Engineering at Alakh Prakash Goyal Shimla University, Shimla. He is having more than 9 years of Industry and Academics in the fiend of Computer Science and Information Technologies. He also worked as LAMP Developer in Computer Ware India Pvt. Ltd- New Delhi, India. He is pursuing Ph. D (Computer Sciecne & Applications) from Magadh University, Bihar, India. He is an MCA and alumnus of SRM University, Chennai, India under the program of M. Tech (Computer Science & Engineering). He has published more than 15 papers in national and International Journals of repute.