Security in Cloud Computing using Cryptographic Algorithms

Project report submitted in fulfillment of the requirement for the degree of
Bachelor of Technology

In

Computer Science and Engineering

By

Shobhankar Rajvanshi (131212)
Himanshu Bansal(131217)

Under the supervision of

(Dr. Amit Kumar Singh)

To

Department of Computer Science & Engineering and Information Technology

**Jaypee University of Information Technology Waknaghat, Solan-173234,
Himachal Pradesh**

CERTIFICATE

Candidate’s Declaration

This is to certify that the work which is being presented in the report entitled **“Security in Cloud Computing using Cryptographic Algorithms”** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of our own work carried out over a period from August 2016 to May 2017 under the supervision of **Dr. Amit Kumar Singh** (Assistant Professor, Senior Grade, Computer Science & Engineering Department).
The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Shobhankar Rajvanshi, 131212 **Himanshu Bansal, 131217**

This is to certify that the above statement made by the candidates is true to the best of my knowledge.

Dr. Amit Kumar Singh
Assistant Professor(Senior Grade)
Computer Science & Engineering Department
Dated:

ACKNOWLEDGEMENT

We owe our profound gratitude to our project supervisor **Dr. Amit Kumar Singh**, who took keen interest and guided us all along in my project work titled **—Security in Cloud Computing using Cryptographic Algorithms**, till the completion of our project by providing all the necessary information for developing the project. The project development helped us in research and we got to know a lot of new things in our domain. We are really thankful to him.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF GRAPHS

ABSTRACT

The cloud is a next generation platform that provides dynamic resource pools, virtualization and high availability. Since cloud computing rests on internet, security issues like privacy, data security, confidentiality and authentication is encountered. In order to get rid of the same, a variety of encryption algorithms and mechanisms are used in different combinations. On the similar terms, we chose to make use of multilevel encryption with the use of hybrid cryptographic algorithms to enhance the security of data on cloud. We aimed at analyzing different combinations of encryption algorithms, on the basis of different performance parameters to deduce a hybrid algorithm which can secure data more efficiently on cloud.

CHAPTER 1
INTRODUCTION

Nowadays, information is one of the most valuable possessions of companies, organizations and individuals. From the beginning of time, people try to secure information saved on various kinds of storage.[8] Cloud computing is rapidly emerging due to the provisioning of elastic, flexible, and on-demand storage and computing services for customers.[14] Cloud computing is a kind of Internet-based computing that provides shared processing resources and data to computers and other devices on demand. A cloud refers to a distinct IT environment that is designed for the purpose of remotely provisioning scalable and measured IT resources.[12] The term originated as a metaphor for the Internet which is, in essence, a network of networks providing remote access to a set of decentralized IT resources. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services), which can be rapidly provisioned and released with minimal management effort.[16] Companies are now switching from traditional databases to cloud computing to curb the cost of maintenance of databases.[9] The mitigation from traditional methods to cloud has some set of problems like reliability and security of data.

The proposed project is an aim at providing better security solutions to the data stored in cloud and thus make it a more reliable source.[2] Any application or data stored on cloud or any communication between a client and a server can be made more secure by the use of an amalgamation of high end, powerful cryptographic algorithms.[15]

The project aims at choosing the best alternative among the various cryptographic techniques available. The given document discusses the uses, advantages and disadvantages of various cryptographic algorithms and provides a comparative analysis for the same.

1.1) PROBLEM STATEMENT

With the increasing need and dependence on cloud as a medium of storage and communication and a similar increase in the advancement of cyber crimes, the earlier measures and methods of ensuring secure and reliable transfer of data have found to be insufficient and unreliable. The numbers of cryptanalytic attacks have increased manifolds because of the faster technology available which poses a serious threat on the integrity of user and user information addressed. Also there is an increasing need for the development of reliable methods for hiding of information so that everything is not visible to everyone.

In particular this cloud application explicitly addresses the issue of security and strives to find better and more dependable ways of providing data security by means of using advanced multilevel cryptographic algorithms and test them on an application using cloud as a medium of data storage.

1.2) OBJECTIVE

The main objective of this project is to study cloud computing as a part and necessity in today's world, its importance and why it has become such a big topic for discussion. And therefore the project aims at understanding the potential risks and threats present along with it and why ensuring security of data is of primary concern. The objective of this project is to study the various used and potential algorithms available for security provision and finally finding the most suitable and effective combination for the same.

The emphasis is on finding the practical implications of the results proposed and not only focusing on the theoretical concepts.

The motive is to become completely aware and familiar with the technology used for the implementation of the project and make the best use of it for our project completion. Thus we aim at drawing out results which could be used in future in real-time projects by means of collective learning, problem solving and collaborative research work through proper coordination and cooperation.

1.3) METHODOLOGY

- **Hybrid Algorithm:** This step involves the development of a suitable combination of cryptographic algorithms that best serves our purpose. The combination can be then tested on various parameters such as encryption speed, throughput etc.

- **Application Design:** We then create an application which acts as an interface between the user and the cloud. The application reads a text file from the user considering the business point of view and stores it on cloud after encrypting the data using the developed hybrid cryptographic algorithm.

- **Integration of security in cloud API:** In this step we add the various security features in our application to ensure user authentication and use of hybrid cryptographic algorithms for encryption of data that would be stored in a database on cloud.

- **Deployment on cloud:** In this step we finally deploy our test application on cloud.

CHAPTER 2

LITERATURE SURVEY

Cloud Computing has emerged as the fastest growing and easy to use platform which provides resource sharing and pay for use services. The following papers serve to define the objectives of cloud computing and also give its benefits and challenges. In the following analysis security has emerged as the biggest concern for shift to cloud computing, and hence we seek various cryptographic encryption algorithms to analyze the best single level encryption algorithms and come up with a hybrid algorithm to improve the security of cloud computing.

2.1) Title: “Data Security Challenges and Its Solutions in Cloud Computing” (2015) [4]

The authors give us a comprehensive view about the major inhibitors to cloud adoption which is shown in Figure 1

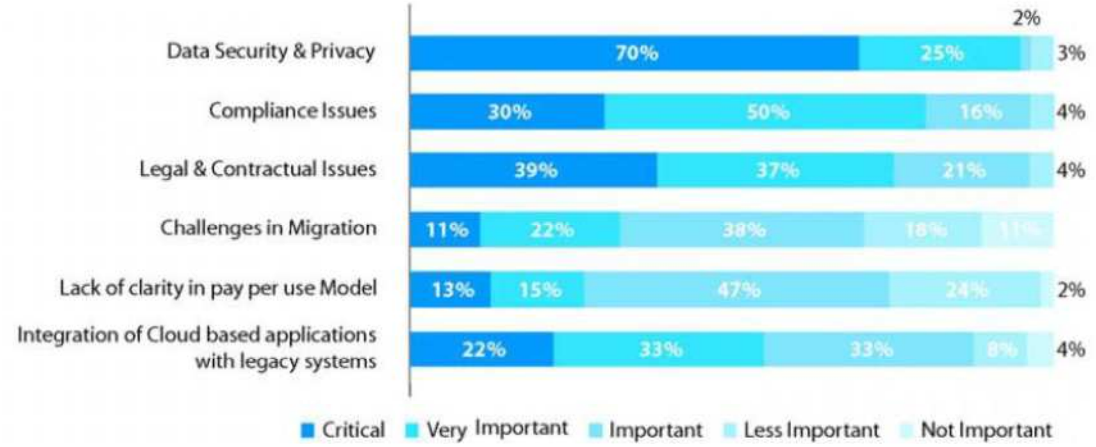


Figure 1 – Main inhibitors to cloud [4]

It can be easily concluded that privacy and security of data is the most crucial aspect of cloud computing that needs to be addressed.

2.1.1) Data Security Challenges:

With the advent of cloud computing platforms fully functional on the internet backbone privacy and security of data on cloud is a major concern. Losses and breaches of data can severely harm the reputation and relevance of the users.[4] Data security challenges are summarized in Figure 2.

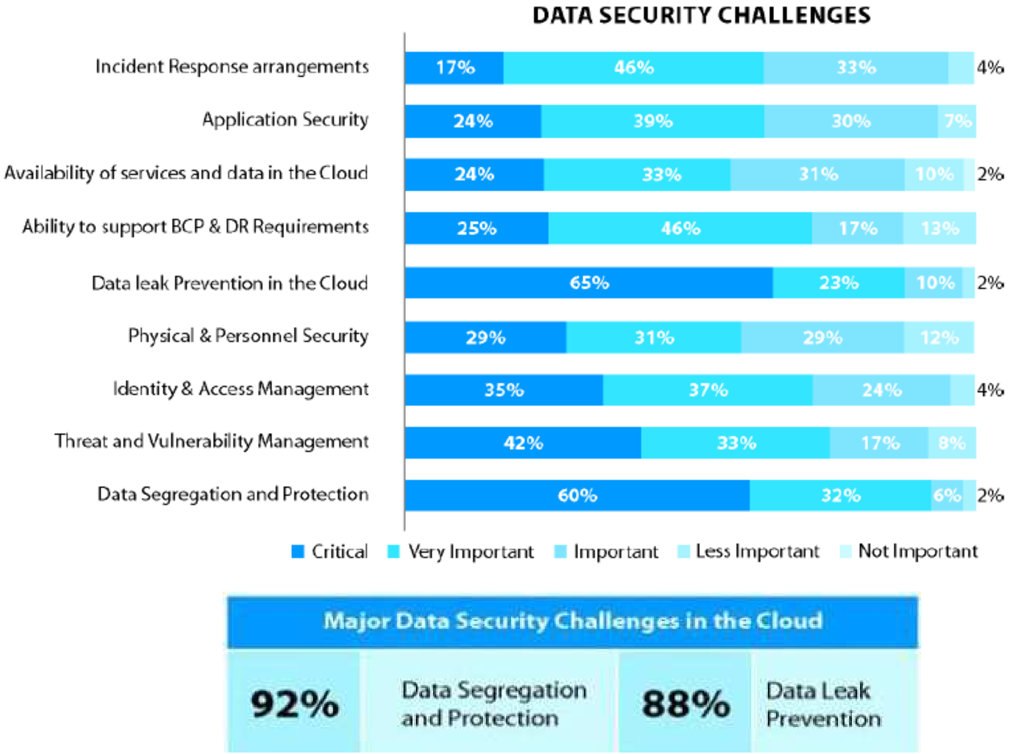


Figure 2 - Challenges in Data Security [4]

2.1.1.1) Security:

In the cloud computing paradigm where the same resource or data is used by a variety of users whether individual or organisations, the risk of security breach in the form of data mishandling and mismanagement cannot be ignored. Hence, in order to do away with this risk, repositories of data need to be secured. Moreover, security of channel can also ensure security of data in transit, operations and storage. Security concepts of authorization, authentication and access control are necessary to be implemented to improve the security of the data that is rendered on cloud [4]. The three main areas in data security are depicted in the figure 3 below -

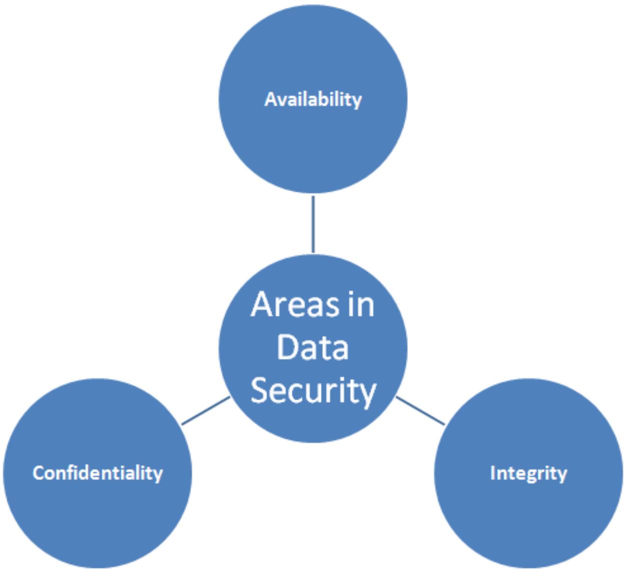


Figure 3 – Areas in Data Security

2.1.1) Conclusion:

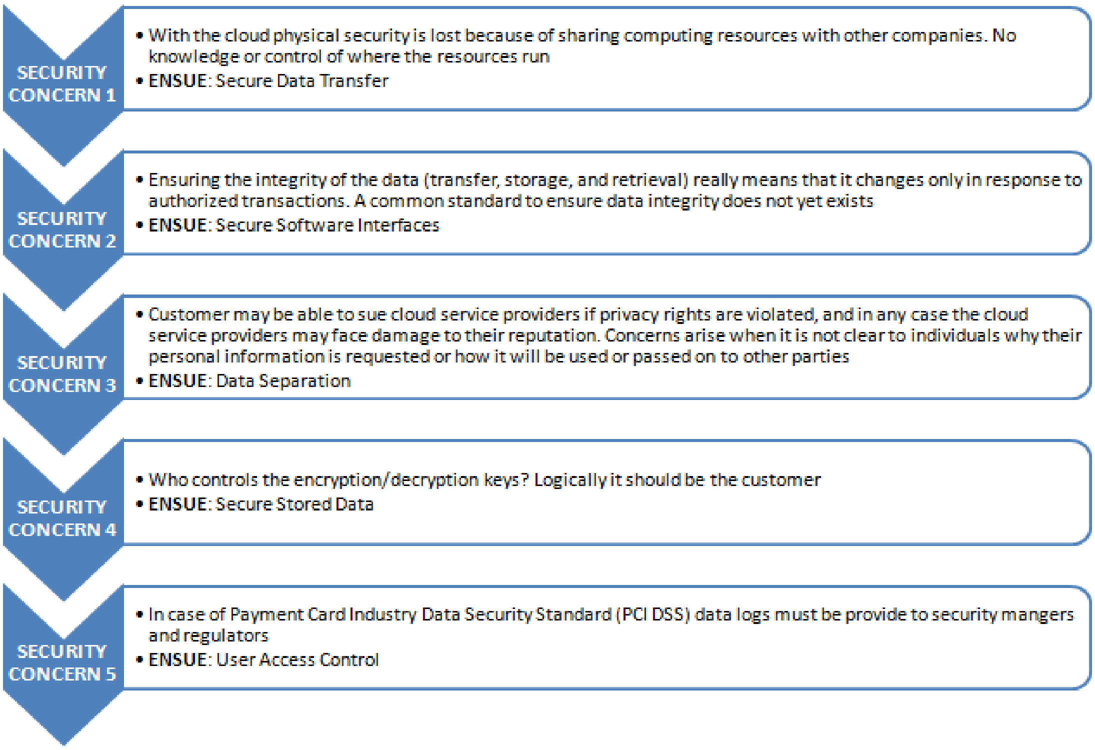
This paper addressed the main concerns about the adoption of cloud computing and pointed out that security and privacy of data on the cloud is the major concern that needs to be tackled. Further the paper also concluded that data leak prevention is considered as most important factor with 88% of Critical and Very important challenges. Similarly Data Segregation and Protection has 92% impact on security challenges.[4] The paper suggested the use of encryption to secure information and to use RSA algorithm for this.

2.2) Title: “Secure User Data in Cloud Computing Using Encryption Algorithms” (2013) [6]

The authors of this paper gives us a brief idea about the various cloud computing issues and challenges and provides a emblematic description of various cryptographic security algorithms like DES, RSA, Blowfish and AES. They compare the Scalability, Security, Data Encryption on Capacity, and Execution Time on these algorithms.

2.2.1) Security Issues and Challenges of Cloud Computing:

With the ever increasing technological growth, security remains one of the most crucial parameter in the world of computing and with the advent of cloud technologies; it has gained ground owing to the relevance of data rendered on the cloud. As cloud is in its initial form it makes use of a variety of new and unsecure technologies, hence it becomes necessary to evaluate the platform on parameters of privacy and security in order to ensure user reliability.[6] The various security concerns that are prevalent in the cloud computing platform are given below in Figure 4.



SECURITY CONCERN 1

- With the cloud physical security is lost because of sharing computing resources with other companies. No knowledge or control of where the resources run
- **ENSUE**: Secure Data Transfer

SECURITY CONCERN 2

- Ensuring the integrity of the data (transfer, storage, and retrieval) really means that it changes only in response to authorized transactions. A common standard to ensure data integrity does not yet exists
- **ENSUE**: Secure Software Interfaces

SECURITY CONCERN 3

- Customer may be able to sue cloud service providers if privacy rights are violated, and in any case the cloud service providers may face damage to their reputation. Concerns arise when it is not clear to individuals why their personal information is requested or how it will be used or passed on to other parties
- **ENSUE**: Data Separation

SECURITY CONCERN 4

- Who controls the encryption/decryption keys? Logically it should be the customer
- **ENSUE**: Secure Stored Data

SECURITY CONCERN 5

- In case of Payment Card Industry Data Security Standard (PCI DSS) data logs must be provide to security mangers and regulators
- **ENSUE**: User Access Control

Figure 4 – Security Concerns

2.2.3) Security Algorithms used in Cloud Computing:

Following are the description of various algorithms that were used in the analysis.

2.2.3.1) Rivest, Shamir, and Adleman (RSA)

It is a public key asymmetric block cipher algorithm. The features are in Figure 5 -

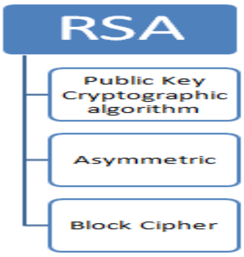


Figure 5 – RSA Features

2.2.3.2) Advanced Encryption Standard (AES)

It is a symmetric block cipher algorithm, used as a NIST standard. The features are in Figure 6 -

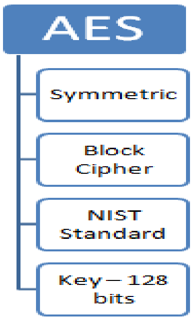


Figure 6 – AES Features

2.2.3.3) Data Encryption Standard (DES)

It is a symmetric block cipher. The features are in Figure 7 -

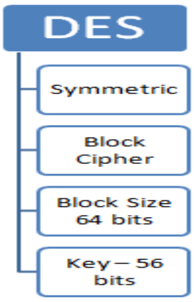


Figure 7 – DES Features

2.2.3.4) BLOWFISH

It is a symmetric block cipher. The features are in Figure 8 -

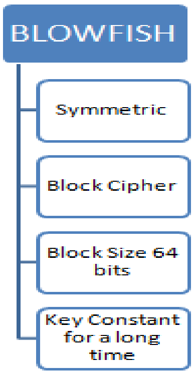


Figure 8 – BLOWFISH Features

2.2.4) Conclusion:

This paper addressed the main security concerns of cloud computing. The paper also compared some of the widely used encryption algorithm and stated that AES algorithm has the fastest execution time, DES algorithm has least encryption time, Blowfish algorithm has least memory requirement, and RSA consumes longest memory size and encryption time.[6]

2.3) Title: “Security in Cloud Computing Using Cryptographic Algorithms”(2015) [5]

In this research paper, first of all the authors describe what is cloud computing and then they define the advantages that it provides to the users, which are given in Figure 9.

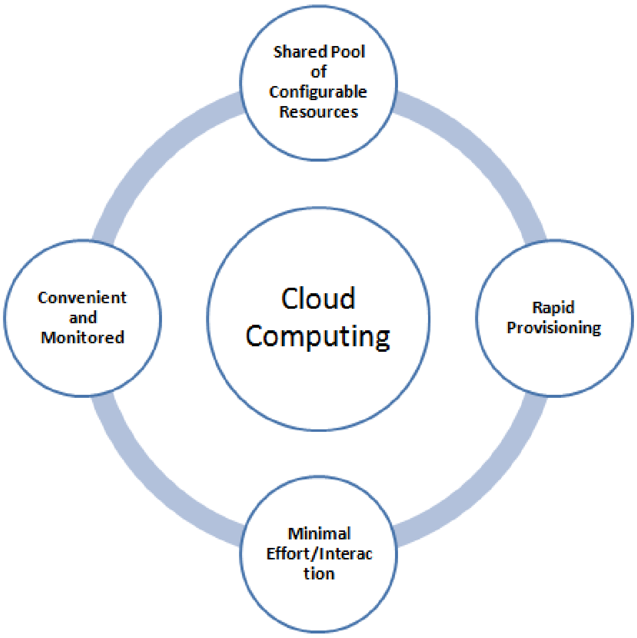

Shared Pool of Configurable Resources
Convenient and Monitored
Cloud Computing
Rapid Provisioning
Minimal Effort/Interac tion

Figure 9 – Features of Cloud Computing [5]

It is due advantages like these that businesses are shifting to cloud. Hence it becomes necessary to secure this data and this security can be achieved by securing the data in treatments (calculations) and storage (databases). The security goals are in Figure 10 -

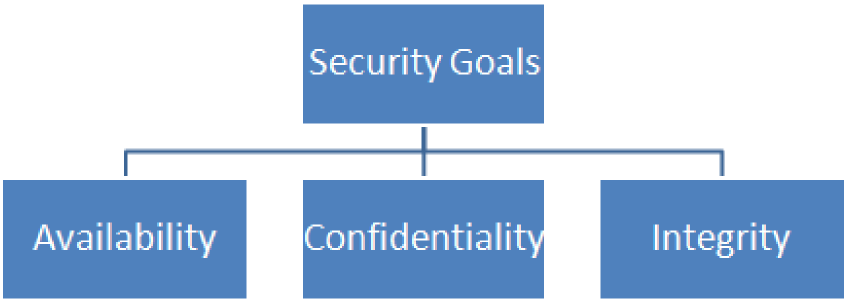


Figure 10 – Security Goals [5]

Confidentiality of data in the cloud is accomplished by cryptography. The cryptographic algorithms are in Figure 11 -

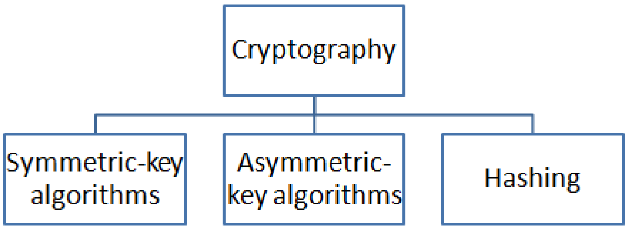


Figure 11 – Types of Cryptographic Algorithms [5]

Integrity of data is ensured by hashing algorithms. Cryptography is the process of transforming the plain text into an unreadable form using the key to make it secure against invaders. The reverse process is decryption. The paper illustrates on two algorithms namely, DES and RSA and proposes a system of multilevel encryption.

2.3.1) Existing Algorithms:

The very availability of cloud means that the data is available throughout and can be accessed from multiple resources that form the platform of the cloud. To provide secure communication over distributed and connected resources, encryption algorithm plays a vital role. It helps in safeguarding the data by converting it to a unreadable form using a key and user needs to have the key to decrypt the data. The authors propose use of both symmetric and asymmetric algorithms simultaneously.[5] The algorithms used are DES and RSA.

2.3.2) Proposed System:

In cloud storage systems, the service provider is entrusted with the security of the data that the user puts on the cloud. Hence it becomes very necessary to make cloud storage secure from a personal point of view.

1. Proposed System Design:

The proposed system is designed to secure the text files with the use of DES and RSA algorithms.

1) Encryption - The encryption process is given as follows:

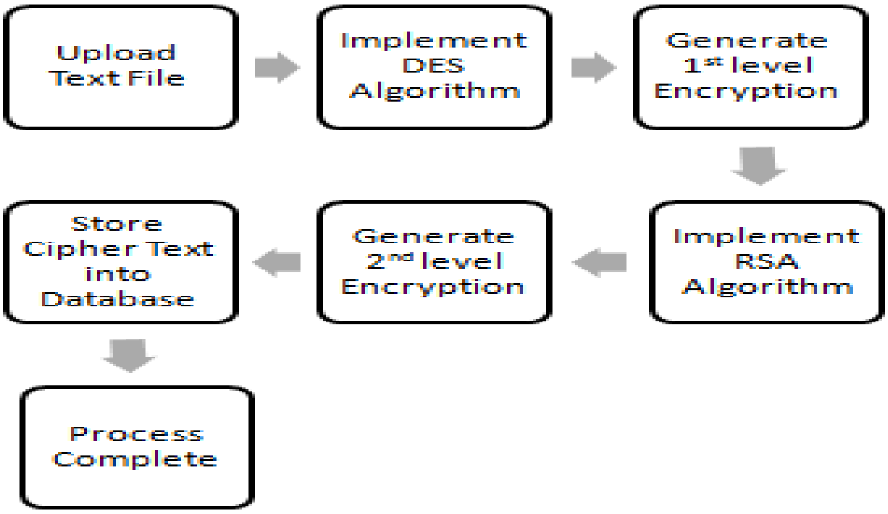

Upload Text File
Implement DES Algorithm
Generate 1st level Encryption
Implement RSA Algorithm
Generate 2nd level Encryption
Store Cipher Text into Database
Process Complete

Figure 12 – Encryption Process [5]

2) Decryption - The decryption process is given as follows:

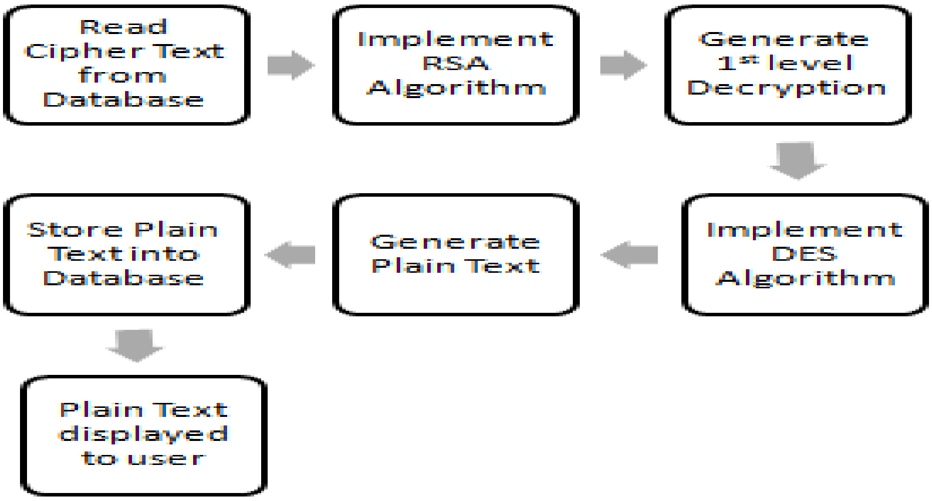

Read Cipher Text from Database
Implement RSA Algorithm
Generate 1st level Decryption
Implement DES Algorithm
Generate Plain Text
Store Plain Text into Database
Plain Text displayed to user

Figure 13 – Decryption Process [5]

2. Proposed Algorithm:

In this paper, the authors have suggested the use of multilevel encryption in place of single level encryption. They have proposed a system in which user uploads his/her text file onto the cloud, where multilevel encryption takes place. First of all, the plain text is subjected to DES encryption to generate first level encryption. Then this cipher text is subjected to RSA encryption to get the final cipher which is stored on the database. For decryption, the cipher text is taken from the database and is subjected to inverse RSA decryption and then with inverse DES decryption to get the final plain text.[5]

2.3.3) Conclusion:

The paper suggests that cyber criminals can easily crack single level encryption. Hence it proposes a system which uses multilevel encryption and decryption to provide more security for Cloud Storage. As the proposed algorithm is a Multilevel Encryption and Decryption algorithm. Even if some intruder (unauthorized user) gets the data accidentally or intentionally, he must have to decrypt the data at each level which is a very difficult task without a valid key.[5] It is expected that using multilevel encryption will provide more security for Cloud Storage than using single level encryption.

2.4) Title: “Analysis of Security Algorithm in Cloud Computing” (2014) [7]

The authors of this paper gives us a brief idea about what cloud computing is and its characteristics and also provides the various advantage of using cloud computing. It also provide us with various challenges of cloud computing. The paper also provides us with various data security issues in cloud computing. The paper also compares various encryption algorithms like AES, Blowfish, DES, RC5 and 3-DES.[7]

2.4.1) Cloud Computing:

Cloud computing is the combination of distributed and utility computing. It is distributed computing in the sense that many nodes/resources are connected to a bigger network and they work to achieve distributed processing and shared pool of resources in a load balancing way. It is utility computing in the sense that the user doesn’t need to physically purchase the resources, it is the service provider who does that and offers them to use for the users in a pay as you use fashion.[7] Cloud computing is the amalgamation of mature web resources and the growing interoperability of systems. Characteristics of cloud from a technical aspect are in Figure 14 -

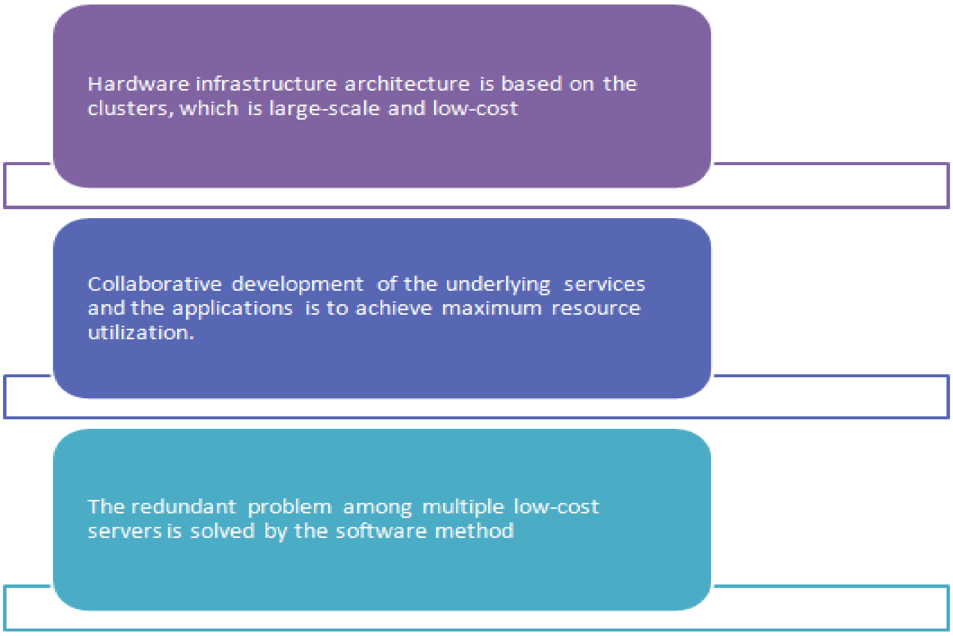

Hardware infrastructure architecture is based on the clusters, which is large-scale and low-cost
Collaborative development of the underlying services and the applications is to achieve maximum resource utilization.
The redundant problem among multiple low-cost servers is solved by the software method

Figure 14 – Technical Aspects of Cloud Computing [7]

2.4.1.1) Benefits:

Following are the advantages of cloud computing in Figure 15 -

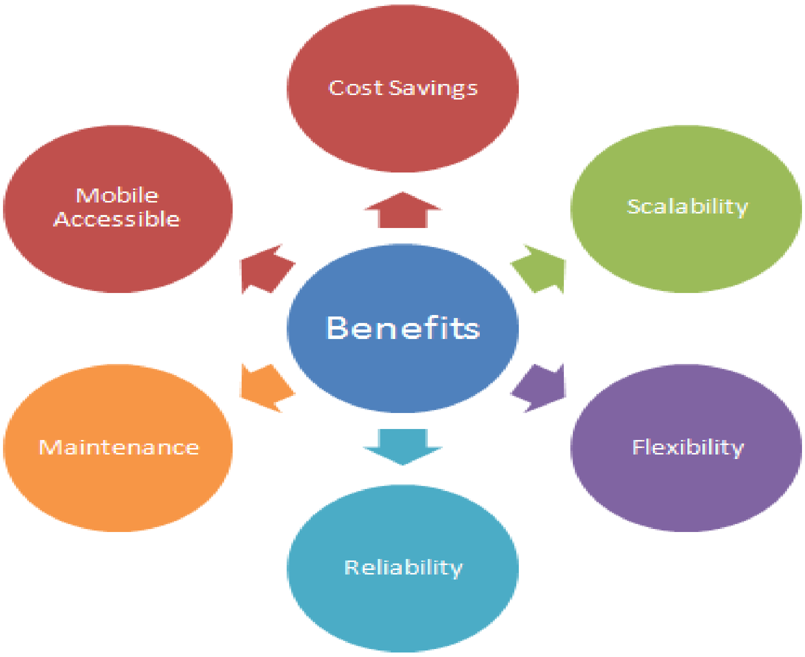


Figure 15 – Benefits of Cloud Computing [7]

2.4.1.2) Challenges:

Following are the challenges that stand in front of cloud computing in Figure 16 -

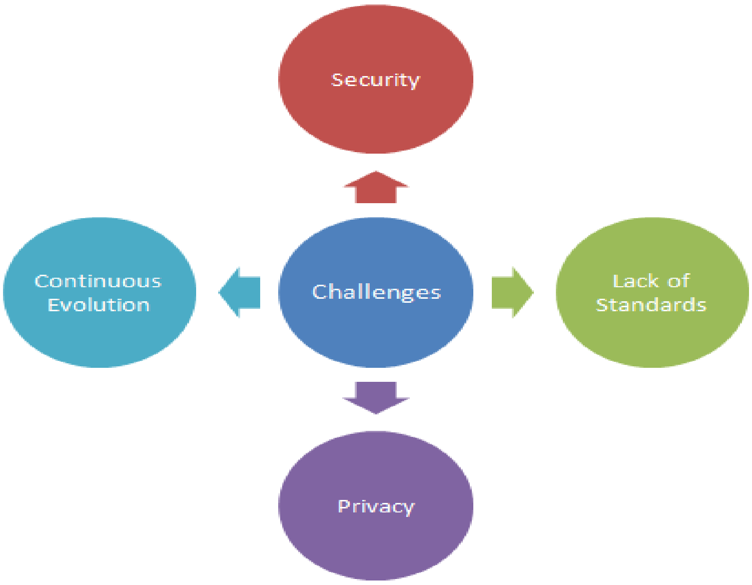
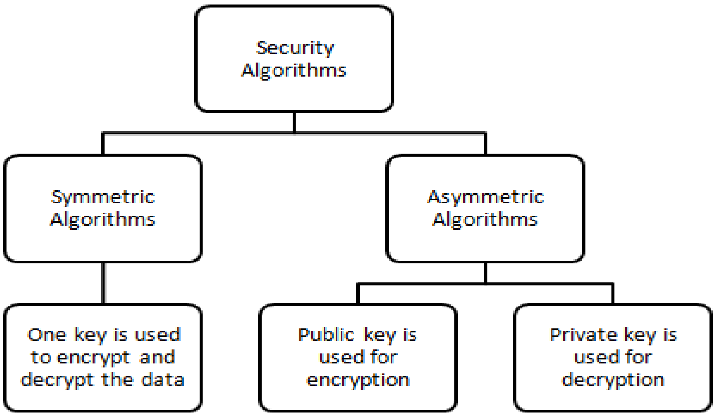


Figure 16 – Challenges to Cloud Computing [7]

2.4.2) Existing Algorithm for Security:

Cryptographic encryption is very important to ensure security of data. Cryptographic encryption algorithms make use of a key to transform the data into an unreadable mixed form and again use the key for its transformation into the plain text. The algorithms are given in Figure 17 and Figure 18 -


Security Algorithms
Symmetric Algorithms
Asymmetric Algorithms
One key is used to encrypt and decrypt the data
Public key is used for encryption
Private key is used for decryption

-

Figure 17 – Cryptographic Security Algorithms

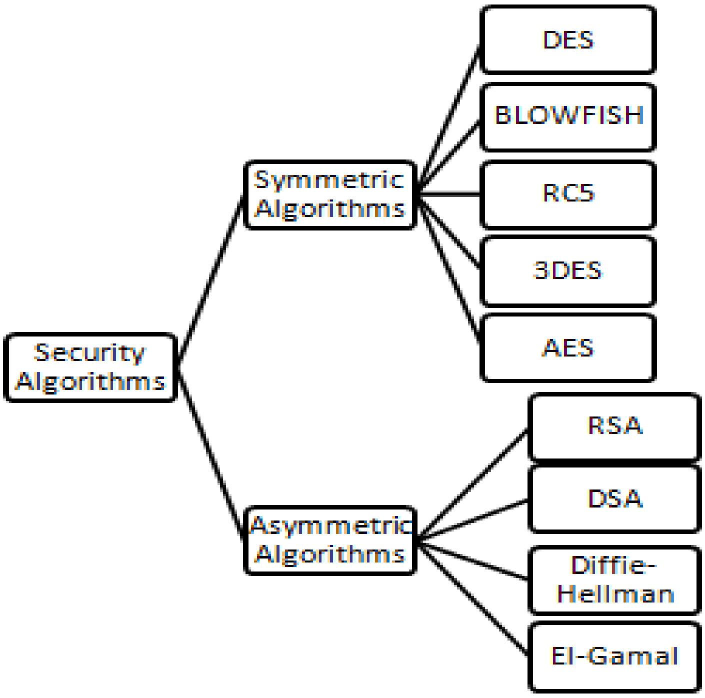


Figure 18 – Different Cryptographic Algorithms [7]

2.4.3) Conclusion:

The paper talked about benefits and challenges in cloud computing. It also talked about various encryption algorithms that are used for security. The paper concluded that among DES, RC5, AES, 3-DES and Blowfish, DES is proven inadequate in security. It also concluded that AES is the fastest algorithm among the pool of algorithms that have been considered.

2.5) Title: “Advantages and challenges of adopting cloud computing from an enterprise perspective” (2014) [3]

With the advent of technology and widening of internet penetration and its greater adoption, cloud computing has come into view as a new age fast-paced and easy to use technology. As it doesn’t require major capital investment in the form of resource purchase and provisioning, it is becoming a good choice for businesses but this process is still in a novel state. Computing as we know today reflects a paradox — on one hand, computers continue to become exponentially more powerful and the per-unit cost of computing continues to fall rapidly, so much so that computing power per sec is nowadays considered to be largely a commodity. On the contrary, as computing is becoming the backbone of the industry, the complexity in terms of resource, protocol, standards and infrastructure management has increased making this management process of computing costly for the organization. The areas under cloud computing are given in Figure 19 -

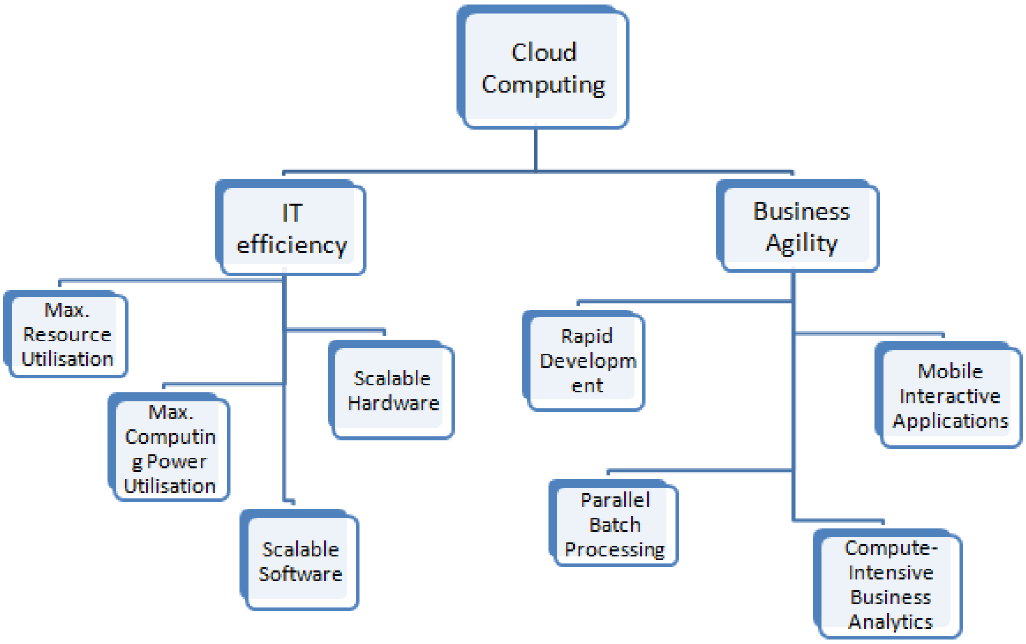

Cloud Computing
IT efficiency
Business Agility
Max. Resource Utilisation
Max. Computin g Power Utilisation
Scalable Hardware
Scalable Software
Rapid Developm ent
Parallel Batch Processing
Mobile Interactive Applications
Compute-Intensive Business Analytics

Figure 19 – Areas under Cloud Computing [3]

2.5.1) Advantages of Cloud Computing

The advantages of shifting to cloud computing are as follows in Figure 20 -

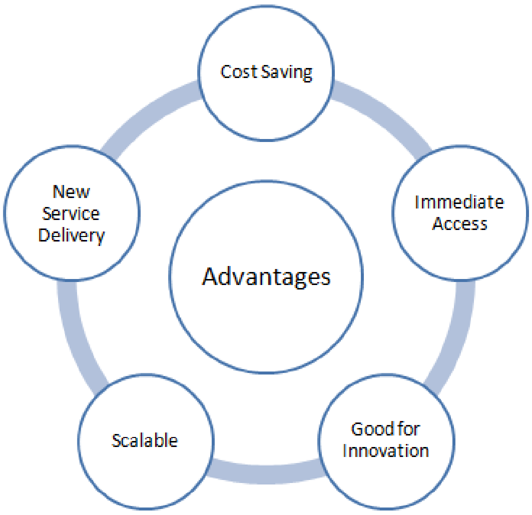


Figure 20 – Advantages of Cloud Computing [3]

2.5.2) Barriers to Cloud Computing Adoption in the Enterprise

The barriers which hamper the adoption of cloud are given in Figure 21-

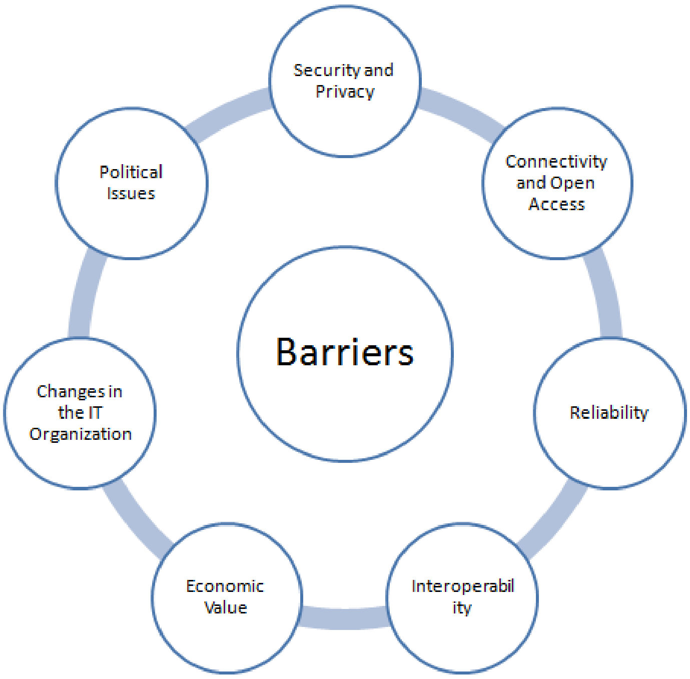


Figure 21 – Barriers to Cloud Computing [3]

2.5.3) Conclusion

Cloud Computing is the new age technology and due its popularity and different interpretations in terms of its structure, use and delivery; it becomes difficult to limit the term. This hurdle is further amplified by the urgency created by vendors when everyone wants to step in the business and make their product as appealing to the users as possible. Before rushing to transform the whole enterprise structure, it becomes necessary to evaluate the economic significance and legitimacy of this transformation from one technological backbone to another. No business today can run without technology, hence in order to determine the profitability of a business IT also needs to justify its value for that business. Prior to rushing into the cloud, it becomes mandatory for enterprises to analyze the advantages and disadvantages of movement to cloud and then plan the migration accordingly.

2.6) Title: “A Review of Cryptographic Algorithms in Network Security” (2016) [10]

When a computer is connected to a network the connected Systems must meet many threats from the hackers. They effect the data transmission over the network. Security mechanisms should be given the information which goes through the network. This mechanism is called cryptography. Cryptography allows the data to be sent in an unidentifiable format. This can’t be read by intruder. Only the sender of the information can understand the message, and the intended receiver can understand the message by applying the key given by the sender. Cryptography has two concepts Encryption and Decryption. Encryption is changing the plain text into unreadable form using key, called cipher text. The cipher text and key will be sent to the receiver. At the receiving end receiver will apply the key on cipher text and gets the actual information.[10]

2.6.1) Types of Cryptography

- **Symmetric Key Cryptography:** If there is only one key used for both encryption and decryption. This key is private key. The algorithms which came under this category are DES, TDES, AES, RC4.[10] The process is in Figure 22 -

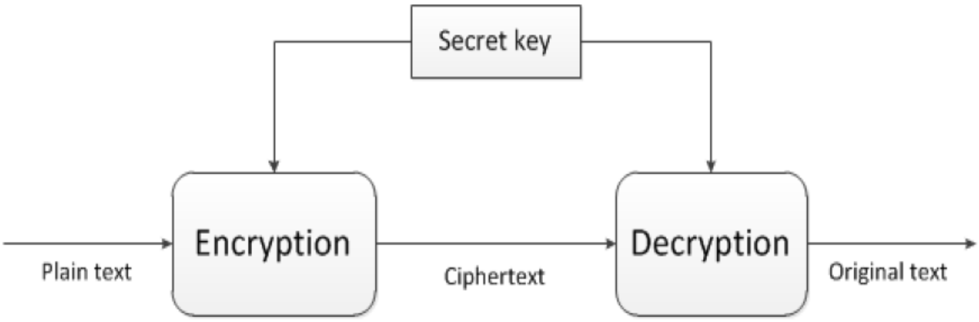


Figure 22 - Symmetric Key Cryptography Process

- **Asymmetric Key Cryptography:** If two keys are used one for encryption and one for decryption. The algorithms which came under this category are RSA, MD5, ECC.[10] The process is in Figure 23

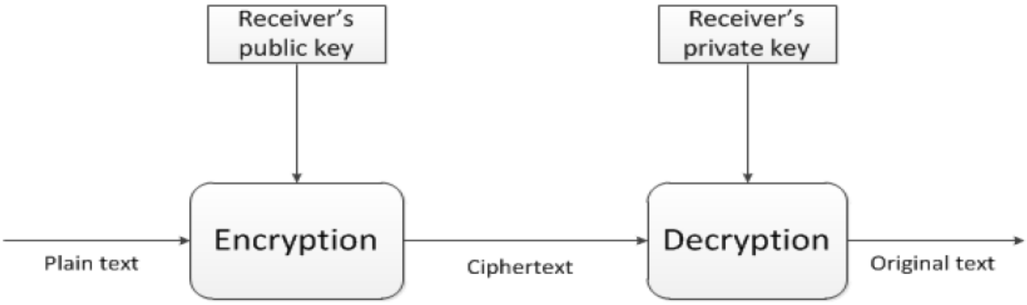


Figure 23 - Asymmetric Key Cryptography Process

2.6.2) Existing Algorithms

The existing ciphers can be classified in Figure 24 -

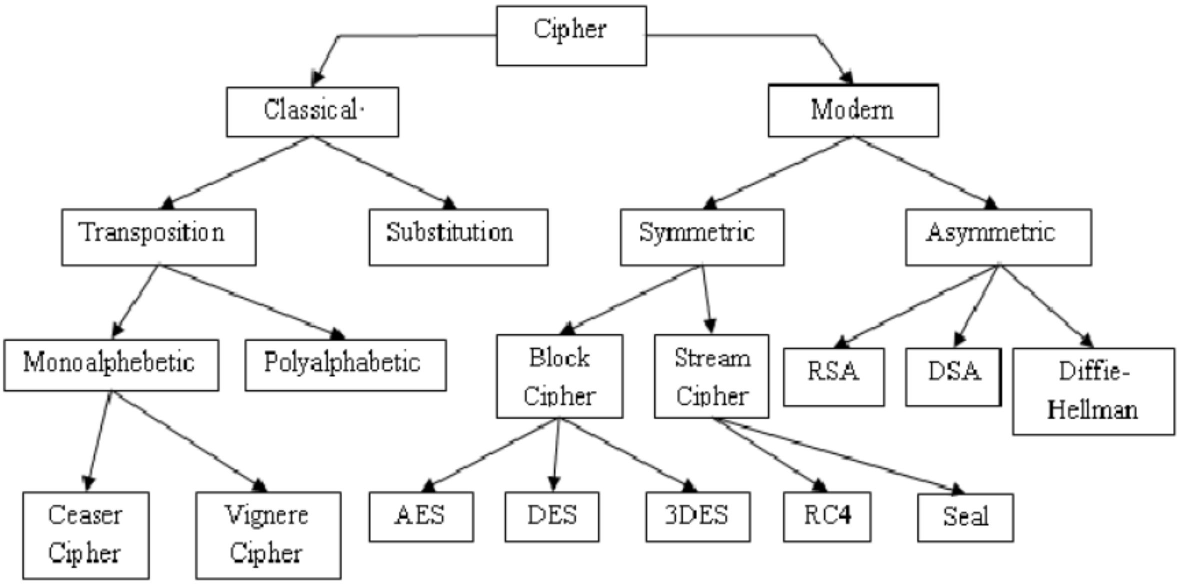


Figure 24 - Classification of Ciphers

- **DES:** DES is developed in 1970s and it uses the Fiestel Structure. It is a symmetric and block cipher algorithm that is DES uses the same key for both encryption and decryption. So the sender and receiver must know the private key. The key length is 64 bits, where 8 bits are taken for parity check. It has 16 rounds of permutation process to encrypt a message. Almost the encryption and decryption process is same except, the decryption is done in reverse order. The possible attack to DES is brute-force attack.[10] Also there are three fast attack is possible to DES algorithm. Those are,

a) Differential Cryptanalysis

b) Linear Cryptanalysis

c) Davies Attack

DES is considered as less secure, and this algorithm is not used much since this has been broken very easily.

- **Triple DES:** Triple data encryption standard is the next level of DES it was designed to break the attacks that DES met. To enhance the security, it processes DES in three times. 48 rounds are needed for TDES process and it has key length of 168 bits. By using this longer key, it applies to each block and encrypts the original text. The TDES is also known as TDEA (Triple DES). There are three keying options. Keying option 1 is strongest and the three keys: K1, K2 and K3 are independent. In keying option 2, the two keys K1 and K2 are independent, and in keying option 3 all the three keys K1, K2 and K3 are identical.[10]

- **AES:** It overcomes the drawback of DES algorithm, AES is also a symmetric and block cipher algorithm. The original name of AES is Rijindeal and published in 1977. It has 128 bit block size and key sizes are 128 (10 rounds), 192 (12 rounds) and 256 bits (14 rounds).[10] The AES permutation process has four stages of substitute bytes, shift rows, mix columns and add round key.

  a) Substitution bytes – In this step, each byte (ai,j) of matrix is replaced with a sub byte (si,j), that is Rijindeal S-Box. At the decryption end, the sub bytes are inversed to reach the original state.

  b) Shift Rows - The shift rows operation, shift each rows with a certain constraint. That is first row of matrix is left same, the second, third and forth rows are shifted to one place left.

  c) Mix Columns – In this step, the each column is multiplied with a fixed polynomial and the new value of the columns is placed.

  d) Add Round Key – This sub key is derived from the main key and the sub key is added into this step by applying XOR to the matrix.

- **RSA :** RSA algorithm is a public-key encryption method of having two keys called private and public keys. It is a block cipher encryption scheme and the key length of 1024 bits. RSA uses two prime numbers to generate public and private keys. These two prime numbers should be chosen randomly.[10] The possible attacks on RSA are,

a) The exponent of small number can be broken easily.

b) If more receivers are getting encrypted message with same exponential, can be decrypted. Also the chosen-cipher text is possible.

- **RC4** : RC4 is developed by Ron Rivest also known as Rivest Cipher 4. Here the stream cipher is used for encryption of the plain text. Pseudorandom stream of bits (key stream) are generated by the RC4 algorithm, and bit-wise encryption/decryption has been performed. The generation key system involves two stages, One is the permutation of all 256 bytes Another is two 8-bit index-pointers. The key length for this RC4 is between 40-128 bits.[10] If the common block ciphers are not used MAC strongly, bit-flapping attack is possible and the stream-cipher attack is also vulnerable if they are not correctly implemented.

- **MD5** : The MD5 Message Digest algorithm is a cryptographic hash function used in many areas. Previous versions of MD5 are MD2 and MD4, and the next version to the MD5 is MD6. Here in MD5 the 128 bits that is 16 bytes of hash function is applied for encryption and decryption. In software field, MD5 is used to give assurance of the downloading files those are not met any intruder.[10] That is the file servers provide a MD5 checksum, the user may compare this MD5 checksum with the downloading file, which confirms the file security.

- **SHA** : SHA is a set of cryptographic hash functions, have SHA-0, SHA-1, SHA-2, SHA-3. The hashing algorithms are most widely trusted and used in many applications for security. The usage of hash function is to provide index to the hash table. It is developed by NIST and published in 1993. The SHA-0 and SHA-1 are moreover same in block size (160 bits) and rounds (80). The SHA-2 has different block sizes of 224, 256, 384, 512 are denoted as SHA-224, SHA-256, SHA-384, SHA-512. The SHA-3 also has different block sizes of 224, 256, 384, 512 can be denoted as SHA3-224, SHA3-256, SHA3-984, SHA3-512.[10]

- **ECC** : It is a public-key cryptography system, that is a pair of keys, one is public-key and another one is private-key. The public-key is a point (x,y) in the curve and the private-key is a random number chosen by user The advantages of ECC algorithm is, it uses shorter key length, CPU consumption is low and memory usage is also very less.[10]

2.6.3) Conclusion

The day to day improving internet technology needs more and fast security for the communication channel, through which the information is passing. Even many algorithms are there to provide security to the network, almost of authors have studied and compared repeatedly the symmetric algorithms. The future work can be done in comparative study of symmetric algorithms on the specific or different parameters like flexibility, speed, encryption time, scalability and memory usage. This will lead to find which one is best in all the parameters to reach better security to the complicated or unsecured network.

2.7) Title: “Next Generation of Computing through Cloud Computing Technology” 2012 [1]

Cloud Computing is the latest developments of computing models after distributed computing, parallel processing and grid computing. Cloud computing achieve multi-level virtualization and abstraction through effective integration of variety of computing, storage, data, applications and other resources, users can be easy to use powerful computing and storage capacity of cloud computing only need to connect to the network. Google, Amazon, Yahoo and other Internet Service Provider, IBM, Microsoft and other IT vendors have put forward their own cloud computing strategy, various telecom operators are also have put great deal of attention on cloud computing, the very low cost of cloud computing platforms becomes the focus of the industry.[16] It has been envisioned as the next generation computing model because of its advantages in on-demand self-service, ubiquitous network access and, location independent resource pooling and transfer of risk.[1] The services of cloud computing is broadly divided into three categories: IaaS, PaaS, SaaS.

2.7.1) Cloud Computing Service Models

The services offered by cloud can be categorized under three headings in Figure 25 -

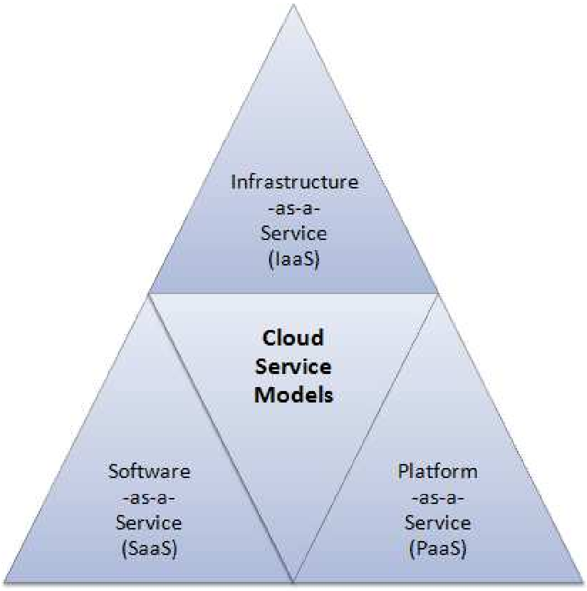


Figure 25 – Cloud Service Models [1]

2.7.2) Cloud Computing Platforms

The commonly prevalent cloud platforms are in Figure 26 -

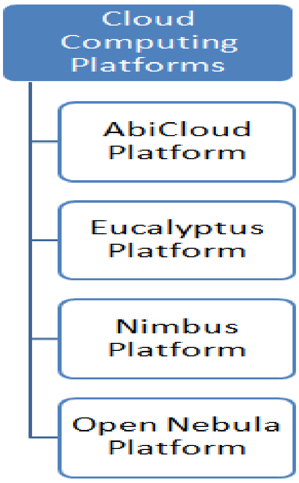


Figure 26 – Cloud Computing Platforms [1]

2.7.3) Conclusion

Cloud computing is a new internet based technology widely used and studied in recent memories. Currently, there are lots of cloud computing platform with varying level of application, architecture, characteristic etc. Now, the difference in the platform is becoming an issue in terms of understanding and usage. In this paper a detailed comparison of four major cloud computing platform has been presented. Based on the analysis, users now have the opportunity to understand the features and be able to make choices of cloud computing platform in respect to cloud computing module, services, development supports, cloud interfaces, deployment, OS supports and compatibility.

CHAPTER 3

SYSTEM DEVELOPMENT

3.1) SOFTWARE REQUIREMENTS:

- Netbeans IDE 8 or above
- JDK(Java Development Kit) 1.7 or above
- bcprov-ext-jdk15on-155.jar
- commons-codec-1.7.jar

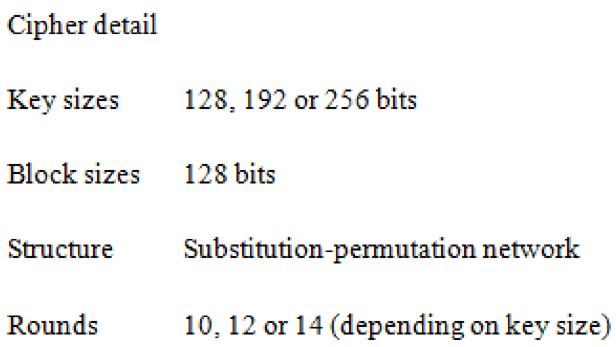
3.2) HARDWARE REQUIREMENTS:

- **System Requirements:**
  - CPU: 1.7 GHz Processor and above
  - RAM: 1 GB or above
  - OS: Windows 8 and above

3.3) RESEARCH ALGORITHMS:

3.3.1) AES

The Advanced Encryption Standard (AES), also known as Rijndael, was established by the U.S. National Institute of Standards and Technology (NIST) in 2001. Rijndael is a family of ciphers with different key and block sizes. It is a symmetric block cipher, to be used until about 2030 and provides data protection until 2100. It is free of licences.



Cipher detail	
Key sizes	128, 192 or 256 bits
Block sizes	128 bits
Structure	Substitution-permutation network
Rounds	10, 12 or 14 (depending on key size)

For instance, if there are 16 bytes, these bytes are represented as this matrix:

$$\begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix}$$

The cycles of repetition vary for key sizes, 10 for 128 bit, 12 for 192 bit and 14 for 256 bit. The round structure is in Figure 27 -

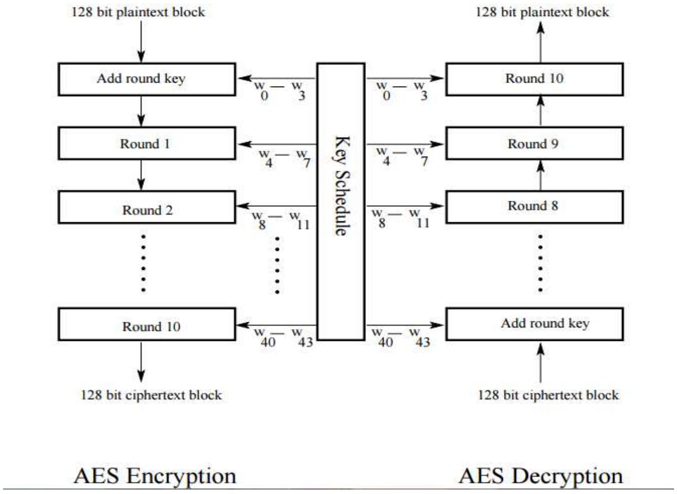


Figure 27 - AES Round Structure

- InitialRound
    - AddRoundKey
- Rounds
    - SubBytes in Figure 28 -

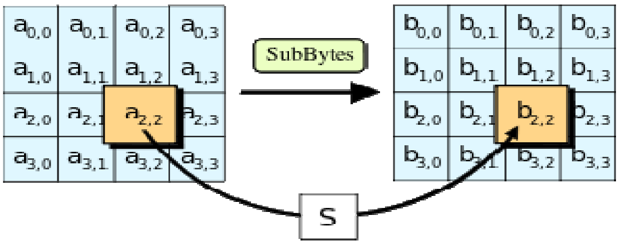

$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
SubBytes
S
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$

Figure 28 – SubBytes Function

- ShiftRows in Figure 29 -

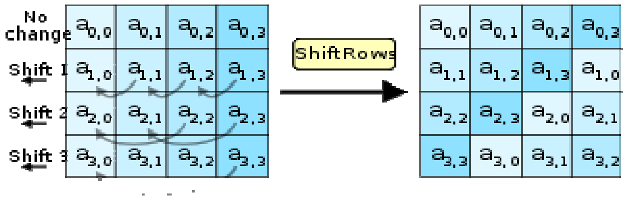

No change
Shift 1
Shift 2
Shift 3
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
ShiftRows
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,1}$ $a_{1,2}$ $a_{1,3}$ $a_{1,0}$
$a_{2,2}$ $a_{2,3}$ $a_{2,0}$ $a_{2,1}$
$a_{3,3}$ $a_{3,0}$ $a_{3,1}$ $a_{3,2}$

Figure 29 – ShiftRows Function

- MixColumns in Figure 30 -

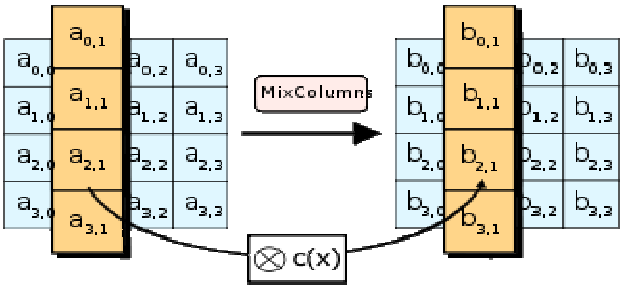

$a_{0,1}$ $a_{1,1}$ $a_{2,1}$ $a_{3,1}$
MixColumns
⊗ c(x)
$b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{3,1}$

Figure 30 – MixColumns Function

- AddRoundKey in Figure 31 -

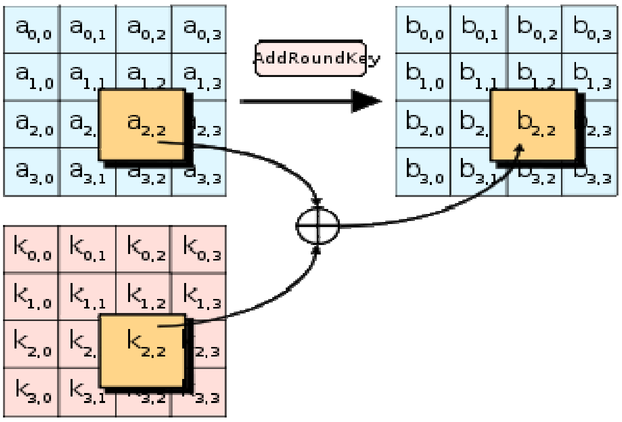

$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
AddRoundKey
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
$k_{0,0}$ $k_{0,1}$ $k_{0,2}$ $k_{0,3}$
$k_{1,0}$ $k_{1,1}$ $k_{1,2}$ $k_{1,3}$
$k_{2,0}$ $k_{2,2}$ $k_{2,3}$
$k_{3,0}$ $k_{3,1}$ $k_{3,2}$ $k_{3,3}$

Figure 31 – AddRoundKey Function

- Final Round – It has all the steps except MixColumns

The process is given in Figure 32 -

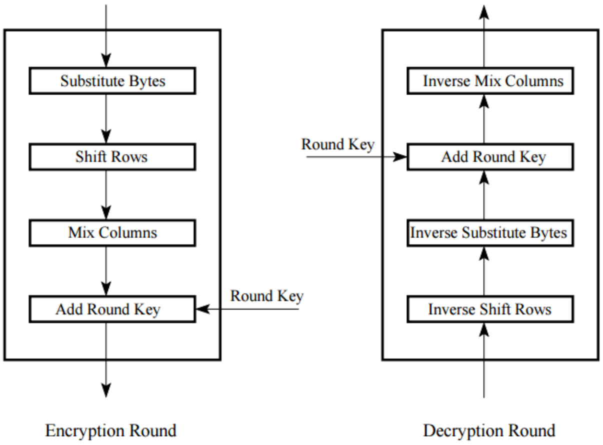
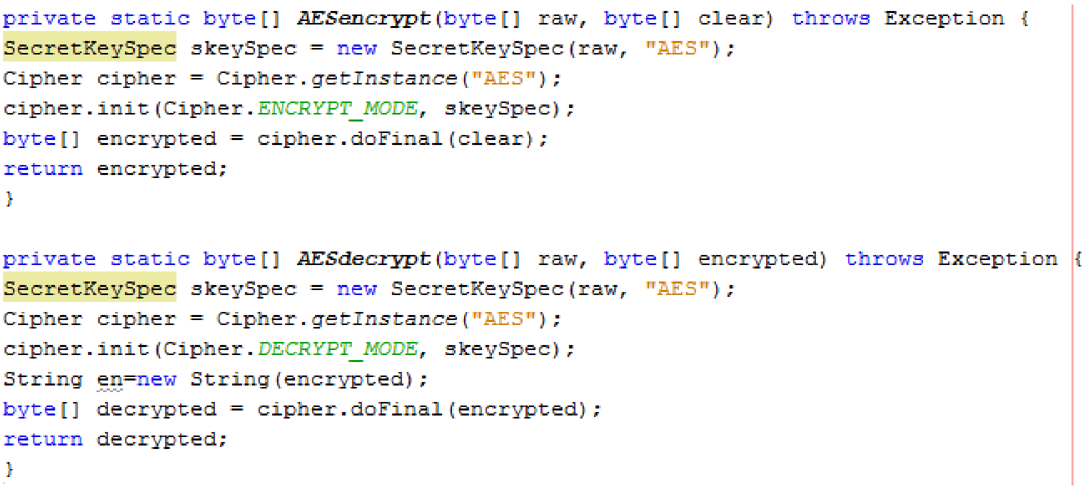


Figure 32 - AES Encryption Decryption Process



```
private static byte[] AESencrypt(byte[] raw, byte[] clear) throws Exception {
SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");
Cipher cipher = Cipher.getInstance("AES");
cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
byte[] encrypted = cipher.doFinal(clear);
return encrypted;
}

private static byte[] AESdecrypt(byte[] raw, byte[] encrypted) throws Exception {
SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");
Cipher cipher = Cipher.getInstance("AES");
cipher.init(Cipher.DECRYPT_MODE, skeySpec);
String en=new String(encrypted);
byte[] decrypted = cipher.doFinal(encrypted);
return decrypted;
}
```

Figure 33 – AES Code

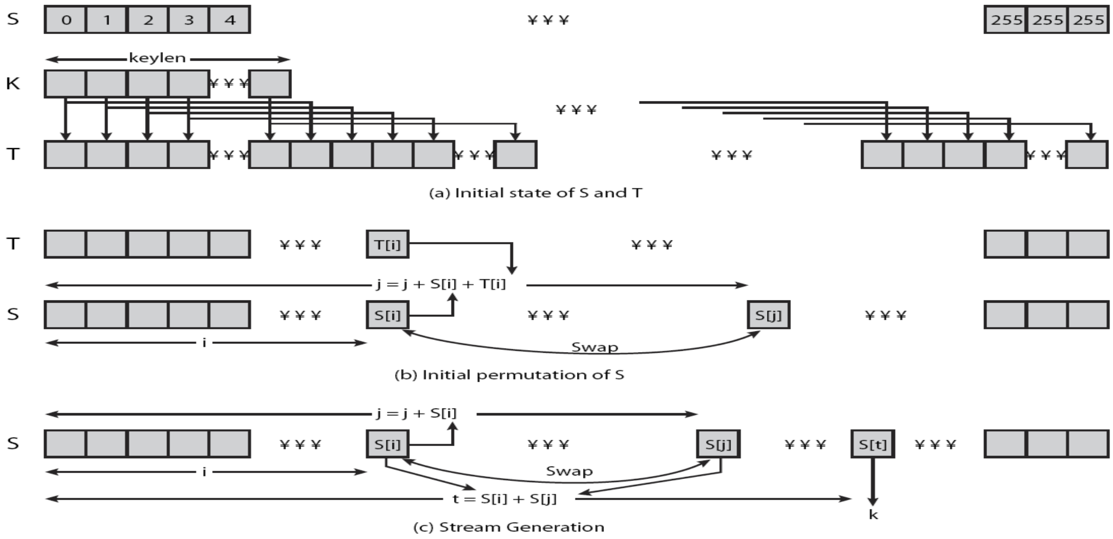
3.3.2) RC4

Cipher detail

Type	Shared Key Stream, Symmetric

Key sizes	40–2048 bits
State size	2064 bits (1684 effective)
Rounds	1

RC4 generates a pseudorandom stream of bits. As with any stream cipher, these can be used for encryption by combining it with the plaintext using bit-wise exclusive-or; decryption is performed the same way. The RC4 process is given in Figure 34 -

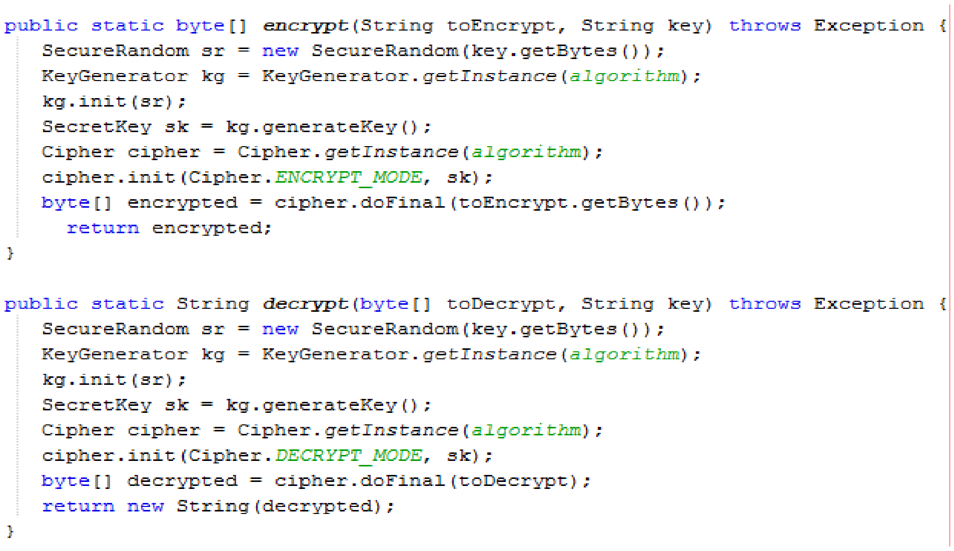


(a) Initial state of S and T

(b) Initial permutation of S

(c) Stream Generation

Figure 34 - RC4 Process



```
public static byte[] encrypt(String toEncrypt, String key) throws Exception {
   SecureRandom sr = new SecureRandom(key.getBytes());
   KeyGenerator kg = KeyGenerator.getInstance(algorithm);
   kg.init(sr);
   SecretKey sk = kg.generateKey();
   Cipher cipher = Cipher.getInstance(algorithm);
   cipher.init(Cipher.ENCRYPT_MODE, sk);
   byte[] encrypted = cipher.doFinal(toEncrypt.getBytes());
     return encrypted;
}

public static String decrypt(byte[] toDecrypt, String key) throws Exception {
   SecureRandom sr = new SecureRandom(key.getBytes());
   KeyGenerator kg = KeyGenerator.getInstance(algorithm);
   kg.init(sr);
   SecretKey sk = kg.generateKey();
   Cipher cipher = Cipher.getInstance(algorithm);
   cipher.init(Cipher.DECRYPT_MODE, sk);
   byte[] decrypted = cipher.doFinal(toDecrypt);
   return new String(decrypted);
}
```

Figure 35 – RC4 Code

3.3.3) RC6

It uses data-dependent rotations to achieve a high level of security.

Type	Symmetric
Key sizes	128, 192, or 256 bits
Block sizes	128 bits
Structure	Feistel
Rounds	20

The RC6 structure is given in Figure 36 -

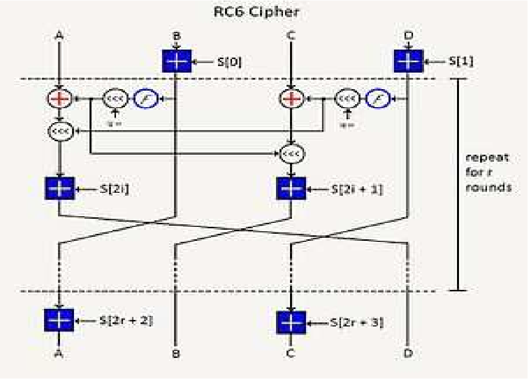
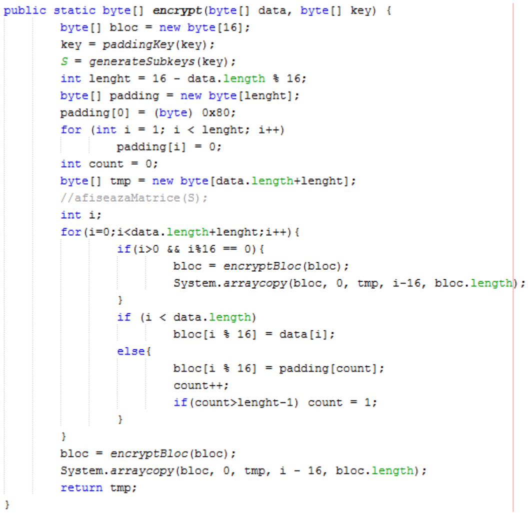
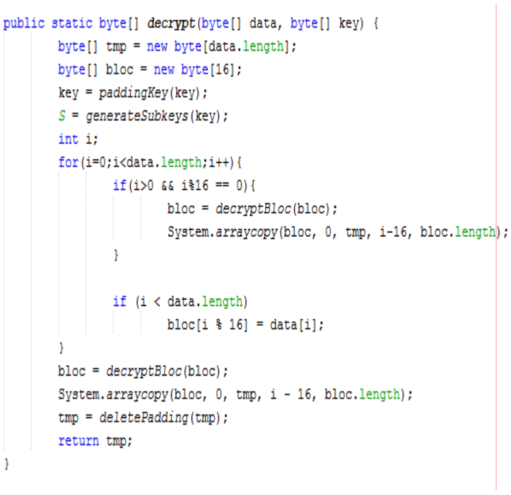


Figure 36 - RC6 Process



```
public static byte[] encrypt(byte[] data, byte[] key) {
        byte[] bloc = new byte[16];
        key = paddingKey(key);
        S = generateSubkeys(key);
        int lenght = 16 - data.length % 16;
        byte[] padding = new byte[lenght];
        padding[0] = (byte) 0x80;
        for (int i = 1; i < lenght; i++)
                padding[i] = 0;
        int count = 0;
        byte[] tmp = new byte[data.length+lenght];
        //afiseazaMatrice(S);
        int i;
        for(i=0;i<data.length+lenght;i++){
                if(i>0 && i%16 == 0){
                        bloc = encryptBloc(bloc);
                        System.arraycopy(bloc, 0, tmp, i-16, bloc.length);
                }
                if (i < data.length)
                        bloc[i % 16] = data[i];
                else{
                        bloc[i % 16] = padding[count];
                        count++;
                        if(count>lenght-1) count = 1;
                }
        }
        bloc = encryptBloc(bloc);
        System.arraycopy(bloc, 0, tmp, i - 16, bloc.length);
        return tmp;
}
```



```
public static byte[] decrypt(byte[] data, byte[] key) {
        byte[] tmp = new byte[data.length];
        byte[] bloc = new byte[16];
        key = paddingKey(key);
        S = generateSubkeys(key);
        int i;
        for(i=0;i<data.length;i++){
                if(i>0 && i%16 == 0){
                        bloc = decryptBloc(bloc);
                        System.arraycopy(bloc, 0, tmp, i-16, bloc.length);
                }

                if (i < data.length)
                        bloc[i % 16] = data[i];
        }
        bloc = decryptBloc(bloc);
        System.arraycopy(bloc, 0, tmp, i - 16, bloc.length);
        tmp = deletePadding(tmp);
        return tmp;
}
```

Figure 37 – RC6 Code

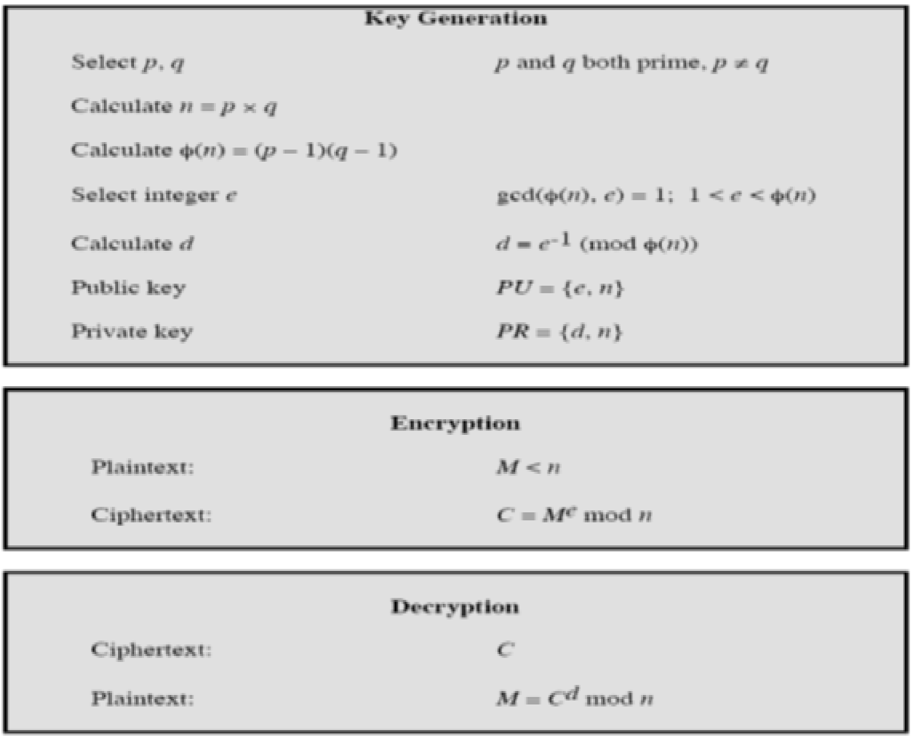
3.3.4) RSA

Use of two large prime numbers.

Cipher detail

Type	Asymmetric
Key sizes	1,024 to 4,096 bit typical
Rounds	1
Issue	Slow, hence used to encrypt keys
Security	Due to cost of factoring large prime numbers O(e log n log log n)

The details of RSA are given in Figure 38 -



Key Generation

Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$d = e^{-1} \pmod{\phi(n)}$
Public key	$PU = \{e, n\}$
Private key	$PR = \{d, n\}$

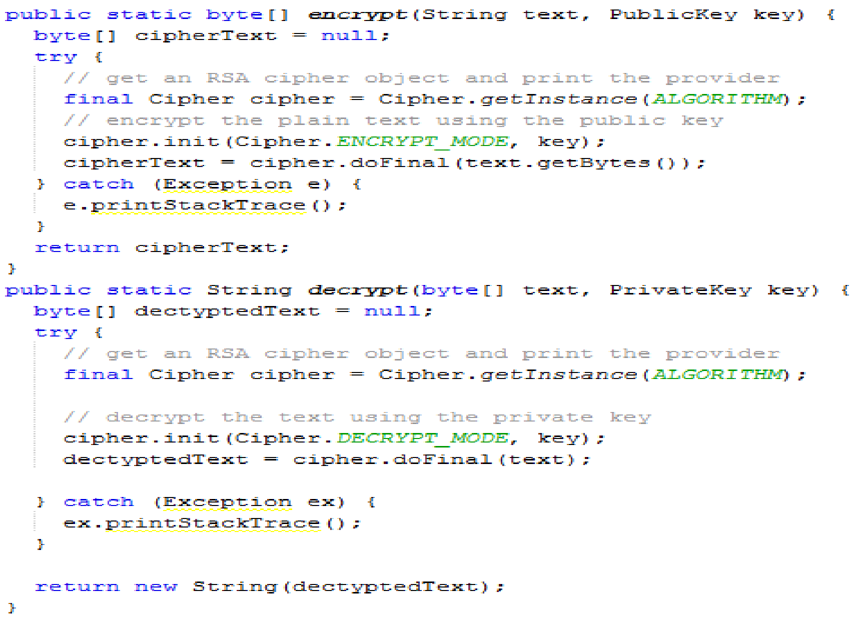
Encryption

Plaintext:	$M < n$
Ciphertext:	$C = M^e \bmod n$

Decryption

Ciphertext:	C
Plaintext:	$M = C^d \bmod n$

Figure 38 – RSA Features



```
public static byte[] encrypt(String text, PublicKey key) {
  byte[] cipherText = null;
  try {
    // get an RSA cipher object and print the provider
    final Cipher cipher = Cipher.getInstance(ALGORITHM);
    // encrypt the plain text using the public key
    cipher.init(Cipher.ENCRYPT_MODE, key);
    cipherText = cipher.doFinal(text.getBytes());
  } catch (Exception e) {
    e.printStackTrace();
  }
  return cipherText;
}
public static String decrypt(byte[] text, PrivateKey key) {
  byte[] dectyptedText = null;
  try {
    // get an RSA cipher object and print the provider
    final Cipher cipher = Cipher.getInstance(ALGORITHM);

    // decrypt the text using the private key
    cipher.init(Cipher.DECRYPT_MODE, key);
    dectyptedText = cipher.doFinal(text);

  } catch (Exception ex) {
    ex.printStackTrace();
  }

  return new String(dectyptedText);
}
```

Figure 39 – RSA Code

3.3.5) ECC

An elliptic curve will simply be the set of points described by the equation:

$m^2=n^3+qn+w$

Where $4q^3+27w^2\neq0$ (this is required to exclude singular curves). The equation above is what is called *Weierstrass normal form* for elliptic curves.

Depending on the value of q and w, elliptic curves may assume different shapes on the plane. As it can be easily seen and verified, elliptic curves are symmetric about the x-axis.

The group law for elliptic curves -

We can define a group over elliptic curves. Specifically:

- The elements of the group are the points of an elliptic curve;
- The identity element is the point at infinity 0;
- The inverse of a point P is the one symmetric about the xx-axis;
- Addition is given by the following rule: given three aligned, non-zero points P, Q and R, their sum is P+Q+R=0.

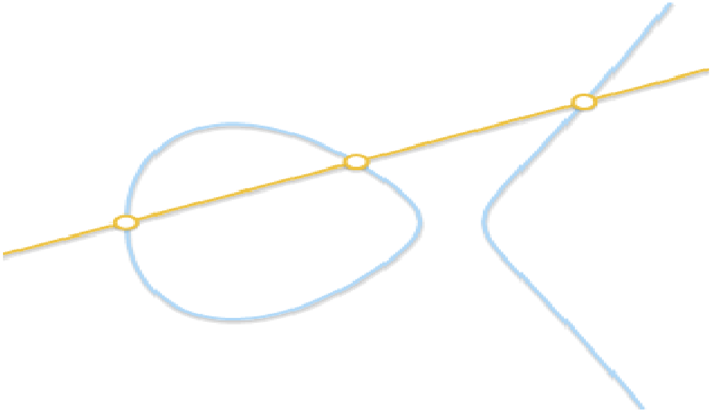
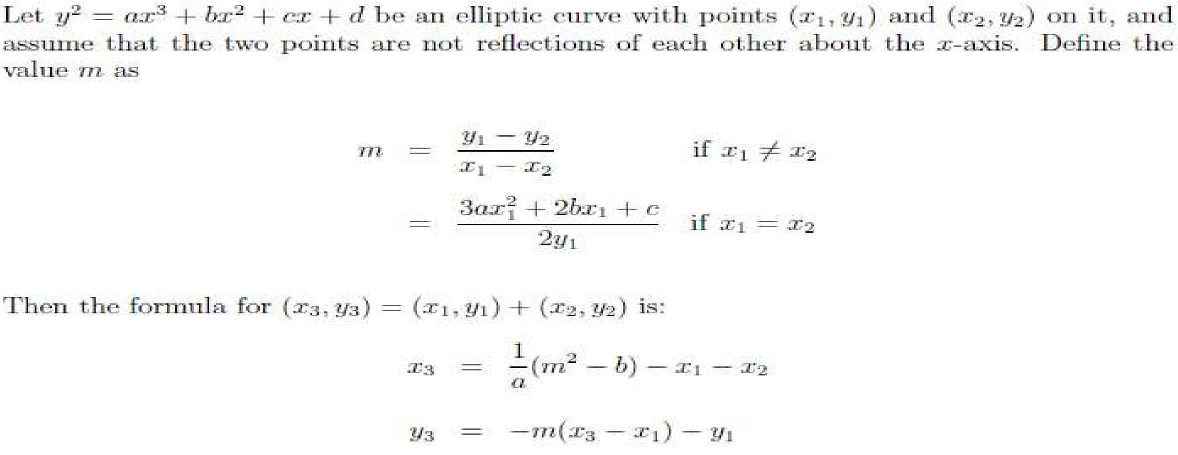


Figure 40 - Sum of three aligned point is 0

Equation for Addition



Let $y^2 = ax^3 + bx^2 + cx + d$ be an elliptic curve with points (x_1, y_1) and (x_2, y_2) on it, and assume that the two points are not reflections of each other about the x-axis. Define the value m as

$$\begin{aligned} m &= \frac{y_1 - y_2}{x_1 - x_2} && \text{if } x_1 \neq x_2 \\ &= \frac{3ax_1^2 + 2bx_1 + c}{2y_1} && \text{if } x_1 = x_2 \end{aligned}$$

Then the formula for $(x_3, y_3) = (x_1, y_1) + (x_2, y_2)$ is:

$$\begin{aligned} x_3 &= \frac{1}{a}(m^2 - b) - x_1 - x_2 \\ y_3 &= -m(x_3 - x_1) - y_1 \end{aligned}$$

Figure 41 – Equation of ECC

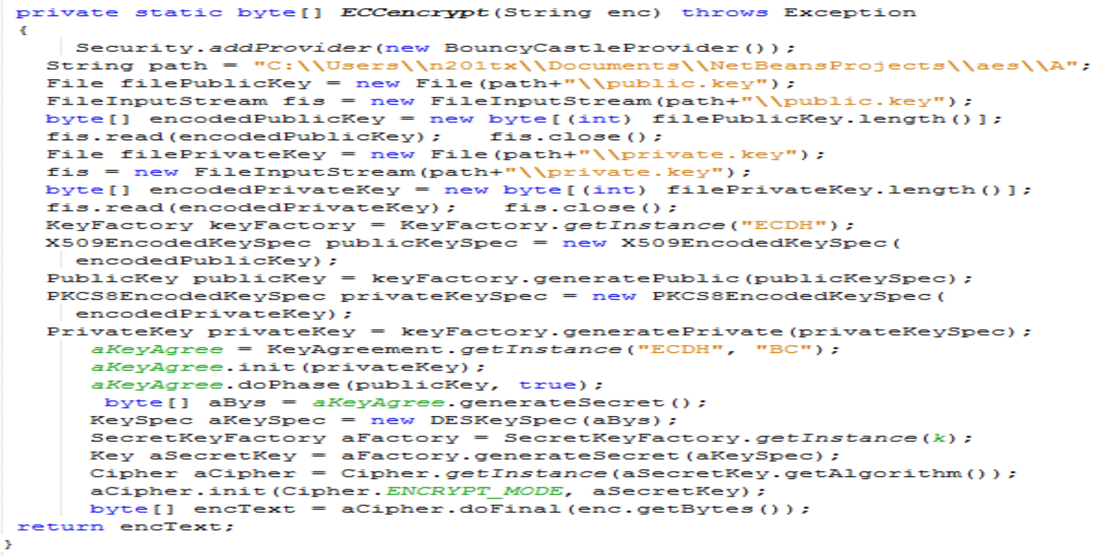
This is how elliptic curve public key cryptography works. For Alice and Bob to communicate securely over an insecure network they can exchange a private key over this network in the following way:

1. A particular rational base point P is published in a public domain for use with a particular elliptic curve E (F_q) also published in a public domain.

2. Alice and Bob choose random integers k_A and k_B respectively, which they use as private keys.

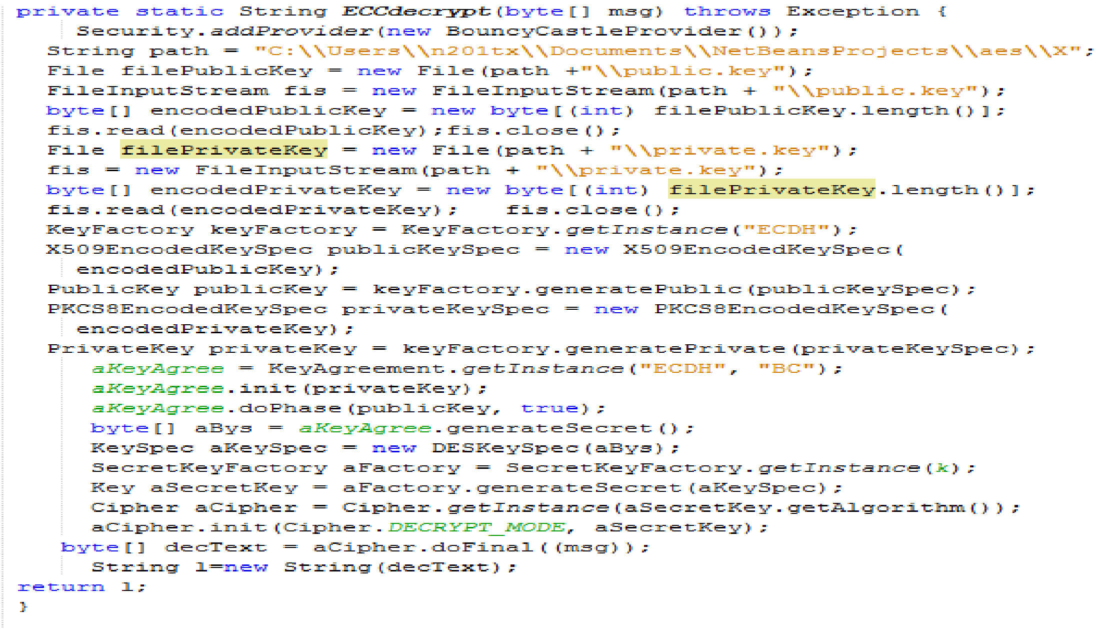
3. Alice computes k_A*P, Bob computes k_B*P and they exchange these values over an insecure network.

4. Using the information they received from each other and their private keys, both Alice and Bob compute (k_A*k_B)*P = k_A*(k_B*P) = k_B*(k_A*P). This value is then the shared secret that only Alice and Bob possess. Note that the difficulty of the ECDLP ensures that the private keys kA and kB and the shared secretly (k_A*k_B)*P are difficult to compute given k_A*P and k_B*P. Thus, Alice and Bob do not compromise their private keys or their shared secret in the exchange.



```
private static byte[] ECCencrypt(String enc) throws Exception
{
     Security.addProvider(new BouncyCastleProvider());
   String path = "C:\\Users\\n201tx\\Documents\\NetBeansProjects\\aes\\A";
   File filePublicKey = new File(path+"\\public.key");
   FileInputStream fis = new FileInputStream(path+"\\public.key");
   byte[] encodedPublicKey = new byte[(int) filePublicKey.length()];
   fis.read(encodedPublicKey);    fis.close();
   File filePrivateKey = new File(path+"\\private.key");
   fis = new FileInputStream(path+"\\private.key");
   byte[] encodedPrivateKey = new byte[(int) filePrivateKey.length()];
   fis.read(encodedPrivateKey);    fis.close();
   KeyFactory keyFactory = KeyFactory.getInstance("ECDH");
   X509EncodedKeySpec publicKeySpec = new X509EncodedKeySpec(
     encodedPublicKey);
   PublicKey publicKey = keyFactory.generatePublic(publicKeySpec);
   PKCS8EncodedKeySpec privateKeySpec = new PKCS8EncodedKeySpec(
     encodedPrivateKey);
   PrivateKey privateKey = keyFactory.generatePrivate(privateKeySpec);
      aKeyAgree = KeyAgreement.getInstance("ECDH", "BC");
      aKeyAgree.init(privateKey);
      aKeyAgree.doPhase(publicKey, true);
       byte[] aBys = aKeyAgree.generateSecret();
      KeySpec aKeySpec = new DESKeySpec(aBys);
      SecretKeyFactory aFactory = SecretKeyFactory.getInstance(k);
      Key aSecretKey = aFactory.generateSecret(aKeySpec);
      Cipher aCipher = Cipher.getInstance(aSecretKey.getAlgorithm());
      aCipher.init(Cipher.ENCRYPT_MODE, aSecretKey);
      byte[] encText = aCipher.doFinal(enc.getBytes());
 return encText;
}
```

Figure 42 – ECC Code 1



```
private static String ECCdecrypt(byte[] msg) throws Exception {
    Security.addProvider(new BouncyCastleProvider());
  String path = "C:\\Users\\n201tx\\Documents\\NetBeansProjects\\aes\\X";
  File filePublicKey = new File(path +"\\public.key");
  FileInputStream fis = new FileInputStream(path + "\\public.key");
  byte[] encodedPublicKey = new byte[(int) filePublicKey.length()];
  fis.read(encodedPublicKey);fis.close();
  File filePrivateKey = new File(path + "\\private.key");
  fis = new FileInputStream(path + "\\private.key");
  byte[] encodedPrivateKey = new byte[(int) filePrivateKey.length()];
  fis.read(encodedPrivateKey);    fis.close();
  KeyFactory keyFactory = KeyFactory.getInstance("ECDH");
  X509EncodedKeySpec publicKeySpec = new X509EncodedKeySpec(
    encodedPublicKey);
  PublicKey publicKey = keyFactory.generatePublic(publicKeySpec);
  PKCS8EncodedKeySpec privateKeySpec = new PKCS8EncodedKeySpec(
    encodedPrivateKey);
  PrivateKey privateKey = keyFactory.generatePrivate(privateKeySpec);
     aKeyAgree = KeyAgreement.getInstance("ECDH", "BC");
     aKeyAgree.init(privateKey);
     aKeyAgree.doPhase(publicKey, true);
     byte[] aBys = aKeyAgree.generateSecret();
     KeySpec aKeySpec = new DESKeySpec(aBys);
     SecretKeyFactory aFactory = SecretKeyFactory.getInstance(k);
     Key aSecretKey = aFactory.generateSecret(aKeySpec);
     Cipher aCipher = Cipher.getInstance(aSecretKey.getAlgorithm());
     aCipher.init(Cipher.DECRYPT_MODE, aSecretKey);
   byte[] decText = aCipher.doFinal((msg));
     String l=new String(decText);
return l;
}
```

Figure 43 – ECC Code 2

3.4) SYSTEM DESIGN:

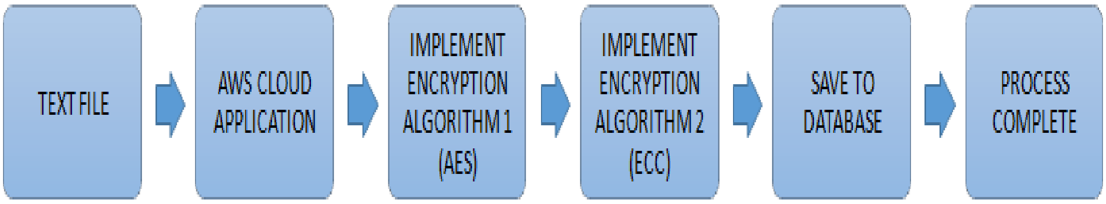

TEXT FILE
AWS CLOUD APPLICATION
IMPLEMENT ENCRYPTION ALGORITHM 1 (AES)
IMPLEMENT ENCRYPTION ALGORITHM 2 (ECC)
SAVE TO DATABASE
PROCESS COMPLETE

Figure 44 – Proposed Encryption

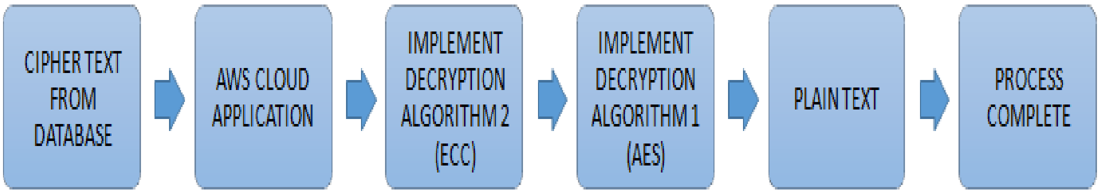

CIPHER TEXT FROM DATABASE
AWS CLOUD APPLICATION
IMPLEMENT DECRYPTION ALGORITHM 2 (ECC)
IMPLEMENT DECRYPTION ALGORITHM 1 (AES)
PLAIN TEXT
PROCESS COMPLETE

Figure 45 – Proposed Decryption

.

Our system reads a text file from the user, through an application on the cloud which acts as an interface between the user and the cloud. Then it implements multilevel encryption algorithm on it, first encrypting using first encryption algorithm and then with the second and finally stores the encrypted files in the cloud database.

During decryption, the application reads the encrypted cipher text file from the cloud database and implements decryption algorithms in the reverse order of encryption i.e. decryption algorithm second followed by decryption algorithm first, and finally renders the file in plain text to the user.

CHAPTER 4

PERFORMANCE ANALYSIS

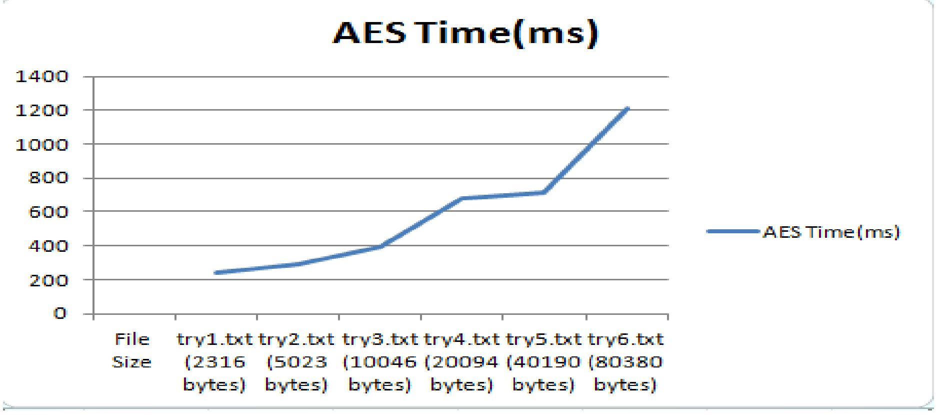
4.1) SINGLE LEVEL ALGORITHM TESTING

This testing seeks to run an algorithm on files of different sizes in order to record the performance of the algorithm in relation to increasing file size. The files used in the testing were text files of sizes varying from 2,316 bytes to 80,380 bytes. Graphs plotted for various algorithms have been depicted.

4.1.1) ADVANCED ENCRYPTION STANDARD (AES):-

Table 1 – AES runtime with increasing file size

FILE SIZE	RUNTIME(ms)
try1.txt (2316 bytes)	244
try2.txt (5023 bytes)	290
try3.txt (10046 bytes)	392
try4.txt (20094 bytes)	676
try5.txt (40190 bytes)	711
try6.txt (80380 bytes)	1212

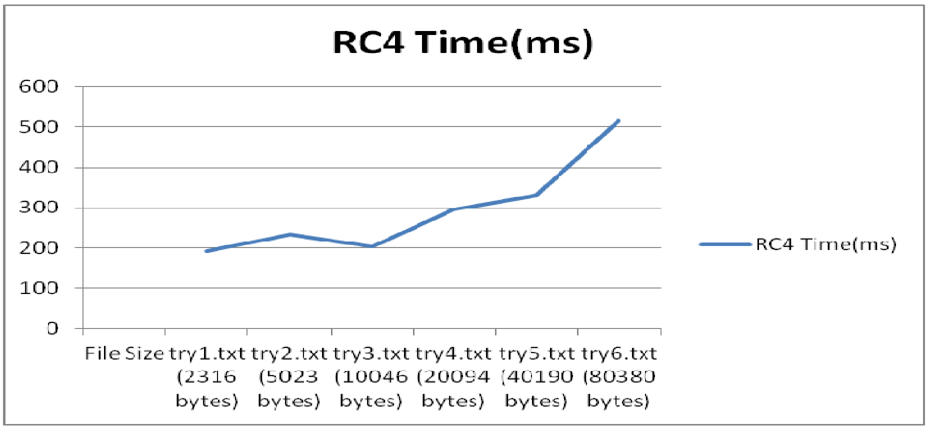


Graphical Figure 1 - AES runtime with increasing file size

4.1.2) RC4 :-

Table 2 – RC4 runtime with increasing file size

FILE SIZE	RUNTIME(ms)
try1.txt (2316 bytes)	192
try2.txt (5023 bytes)	233
try3.txt (10046 bytes)	203
try4.txt (20094 bytes)	296
try5.txt (40190 bytes)	331
try6.txt (80380 bytes)	517

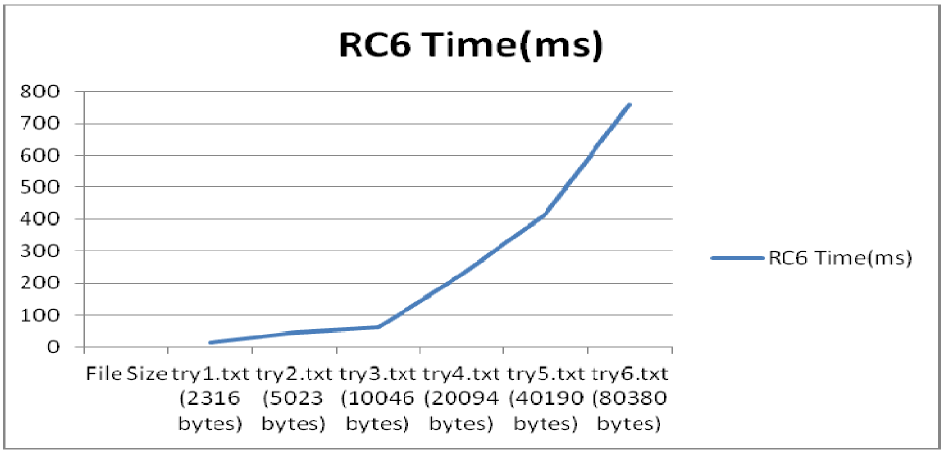


Graphical Figure 2 - RC4 runtime with increasing file size

4.1.3) RC6 :-

Table 3 – Runtime of RC6 with increasing file size

FILE SIZE	RUNTIME(ms)
try1.txt (2316 bytes)	15
try2.txt (5023 bytes)	47
try3.txt (10046 bytes)	63
try4.txt (20094 bytes)	227
try5.txt (40190 bytes)	416
try6.txt (80380 bytes)	760

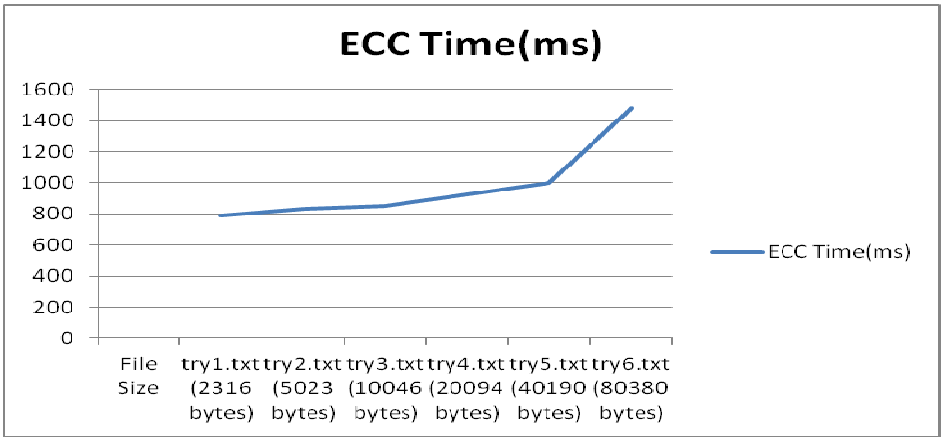


Graphical Figure 3 - Runtime of RC6 with increasing file size

4.1.4) ECC :-

Table 4 – Runtime of ECC with increasing file size

FILE SIZE	RUNTIME(ms)
try1.txt (2316 bytes)	788
try2.txt (5023 bytes)	832
try3.txt (10046 bytes)	851
try4.txt (20094 bytes)	924
try5.txt (40190 bytes)	1001
try6.txt (80380 bytes)	1479

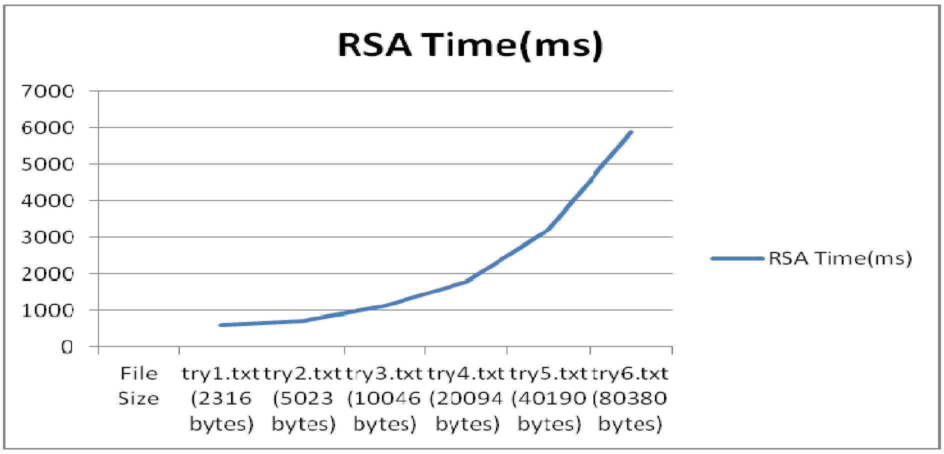


Graphical Figure 4 - Runtime of ECC with increasing file size

4.1.5) RSA :-

Table 5 – Runtime of RSA with increasing file size

FILE SIZE	RUNTIME(ms)
try1.txt (2316 bytes)	625
try2.txt (5023 bytes)	734
try3.txt (10046 bytes)	1125
try4.txt (20094 bytes)	1812
try5.txt (40190 bytes)	3235
try6.txt (80380 bytes)	5876

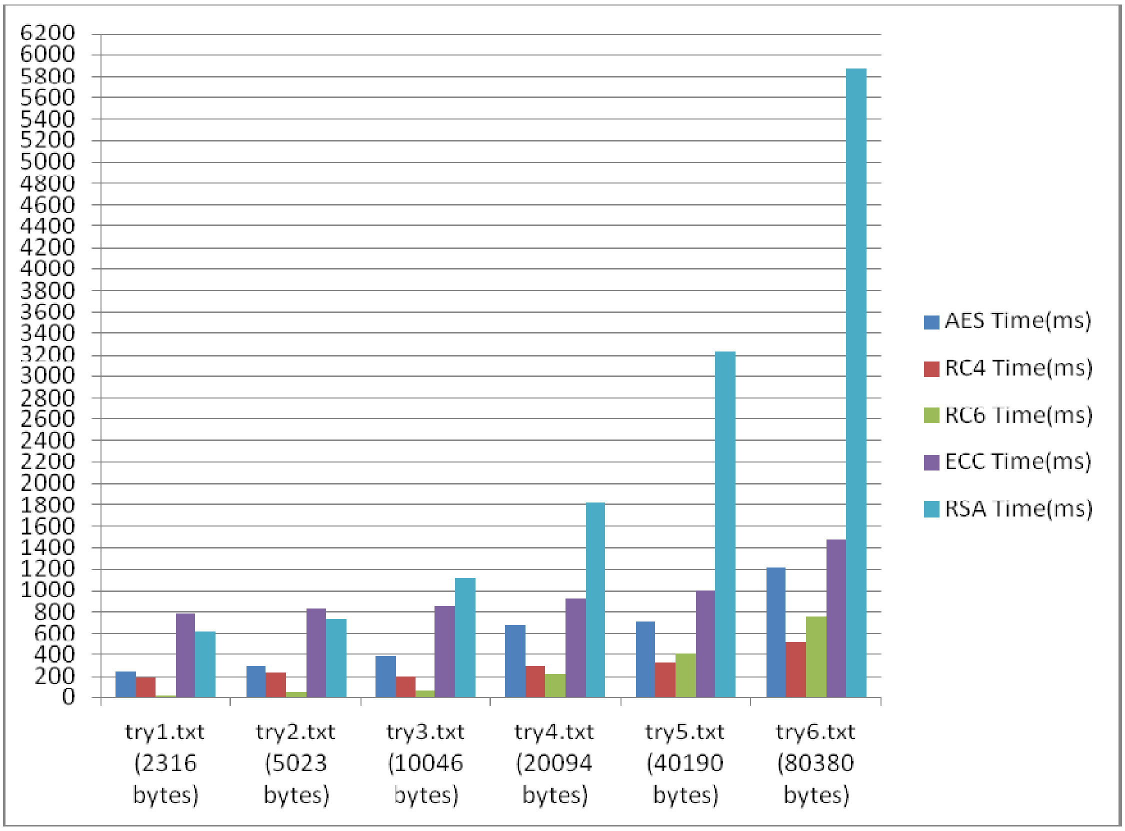


Graphical Figure 5 - Runtime of RSA with increasing file size

4.1.5) COMPARISION OF ALGORITHMS:-

Table 6 – Comparison of runtimes of single level encryption algorithms

File Size	AES	RC4	RC6	ECC	RSA
	Time(ms)	Time(ms)	Time(ms)	Time(ms)	Time(ms)
try1.txt (2316 bytes)	244	192	15	788	625
try2.txt (5023 bytes)	290	233	47	832	734
try3.txt (10046 bytes)	392	203	63	851	1125
try4.txt (20094 bytes)	676	296	227	924	1812
try5.txt (40190 bytes)	711	331	416	1001	3235
try6.txt (80380 bytes)	1212	517	760	1479	5876



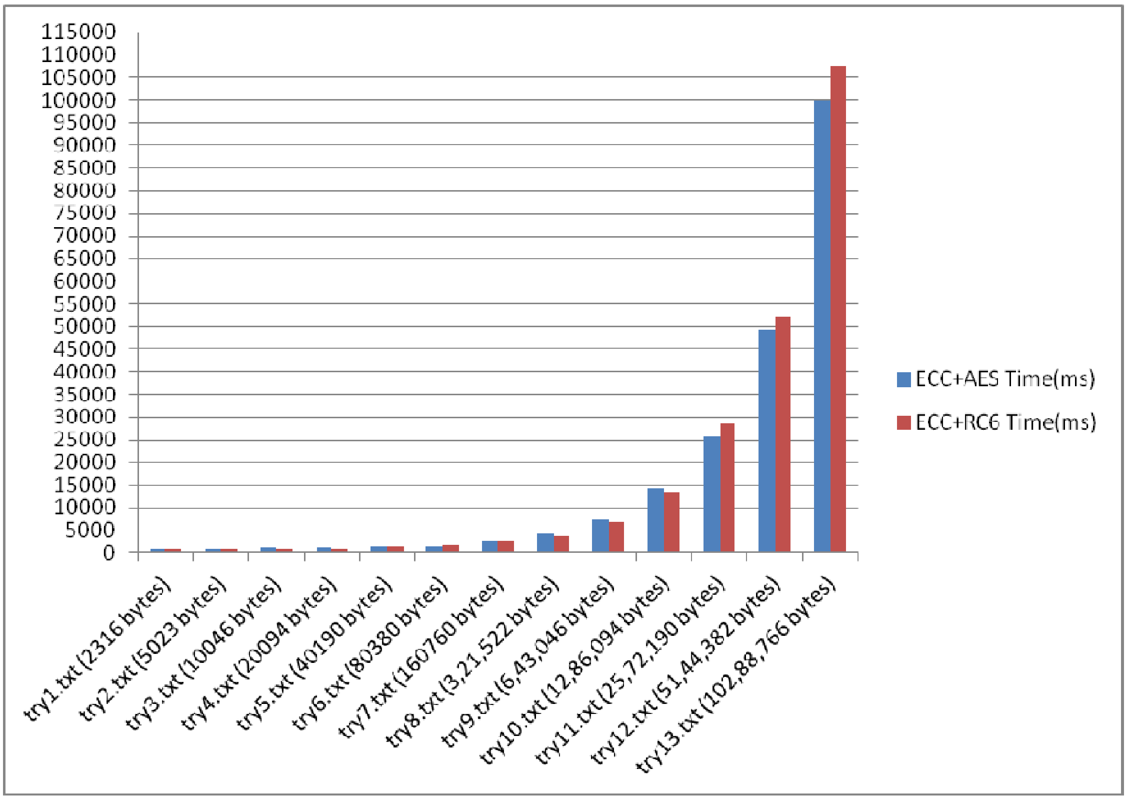
Graphical Figure 6 - Comparison of runtimes of single level encryption algorithms

4.2) MULTI LEVEL ALGORITHM TESTING

This testing seeks to run an algorithm combination on files of different sizes in order to record the performance of the combination in relation to increasing file size. On the basis of the individual runtimes of the algorithms, it was concluded that AES, RC6 and ECC were substantially effective in execution time. Hence, the combinations of these algorithms are selected for testing. The files used in the testing were text files of sizes varying from 2,316 bytes to 102,88,766 bytes. Graphs plotted for various algorithms have been depicted.

Table 7 – Comparison of runtimes of multilevel encryption algorithms

FILE SIZE	**ECC+AES** Time(ms)	**ECC+RC6** Time(ms)
try1.txt (2316 bytes)	985	922
try2.txt (5023 bytes)	1032	953
try3.txt (10046 bytes)	1109	1031
try4.txt (20094 bytes)	1188	1093
try5.txt (40190 bytes)	1495	1359
try6.txt (80380 bytes)	1594	1703
try7.txt (160760 bytes)	2469	2488
try8.txt (3,21,522 bytes)	4266	3844
try9.txt (6,43,046 bytes)	7207	6767
try10.txt (12,86,094 bytes)	14159	13411
try11.txt (25,72,190 bytes)	25729	28573
try12.txt (51,44,382 bytes)	49340	52161
try13.txt (102,88,766 bytes)	99874	107536

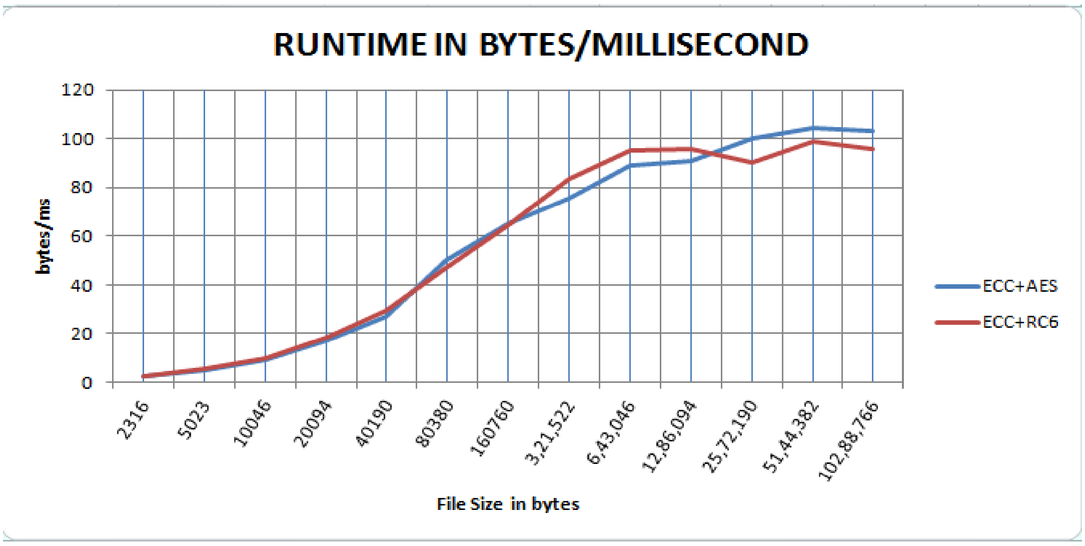


Graphical Figure 7 - Comparison of runtimes of multilevel encryption algorithms

4.3) TESTING ANALYSIS

Table 8 – Throughput of multilevel encryption algorithms

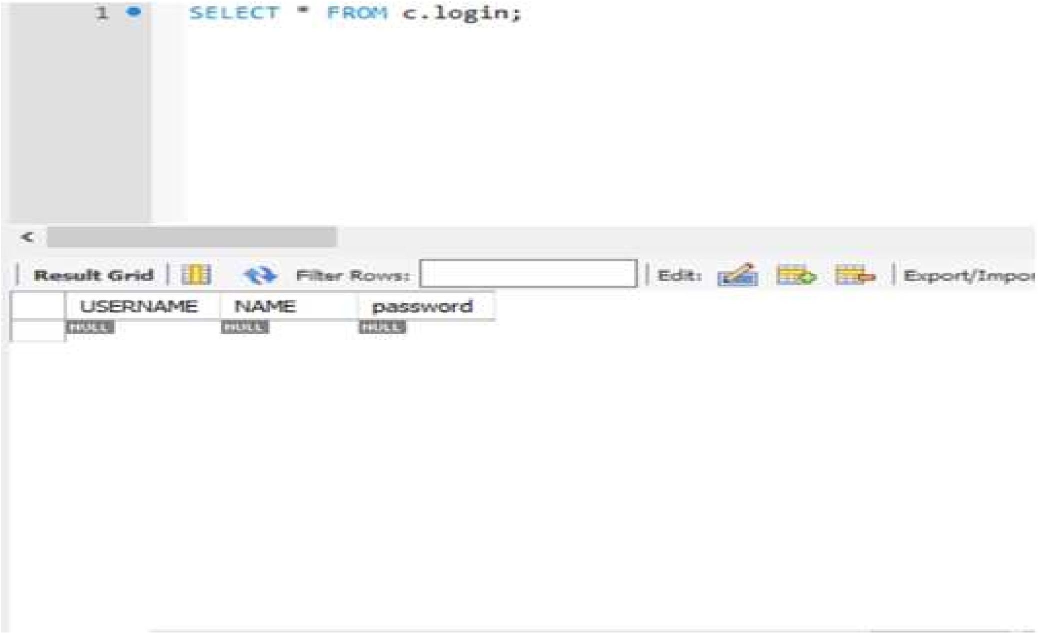
FILE SIZE(bytes)	ECC+AES (bytes/ms)	ECC+RC6 (bytes/ms)
2316	2.351269	2.511931
5023	4.867248	5.270724
10046	9.058611	9.743938
20094	16.91414	18.38426
40190	26.88294	29.57322
80380	50.4266	47.19906
160760	65.11138	64.61415
3,21,522	75.3685	83.64256
6,43,046	89.2252	95.02675
12,86,094	90.83226	95.89844
25,72,190	99.9724	90.0217
51,44,382	104.2639	98.62506
102,88,766	103.0175	95.67741



Graphical Figure 8 - Throughput of multilevel encryption algorithms

4.4) Embedding Algorithm in Application

* Database for user registration

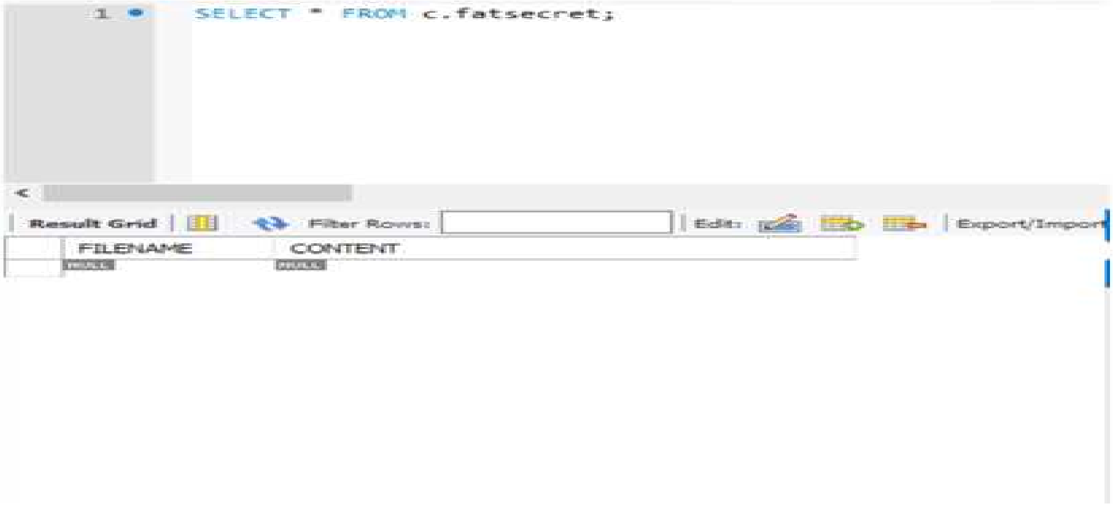


```
SELECT * FROM c.login;
```

USERNAME	NAME	password
NULL	NULL	NULL

Figure 46 – User Database

* Database for file encryption



```
SELECT * FROM c.fatsecret;
```

FILENAME	CONTENT
NULL	NULL

Figure 47 – File Database

* Application Interface

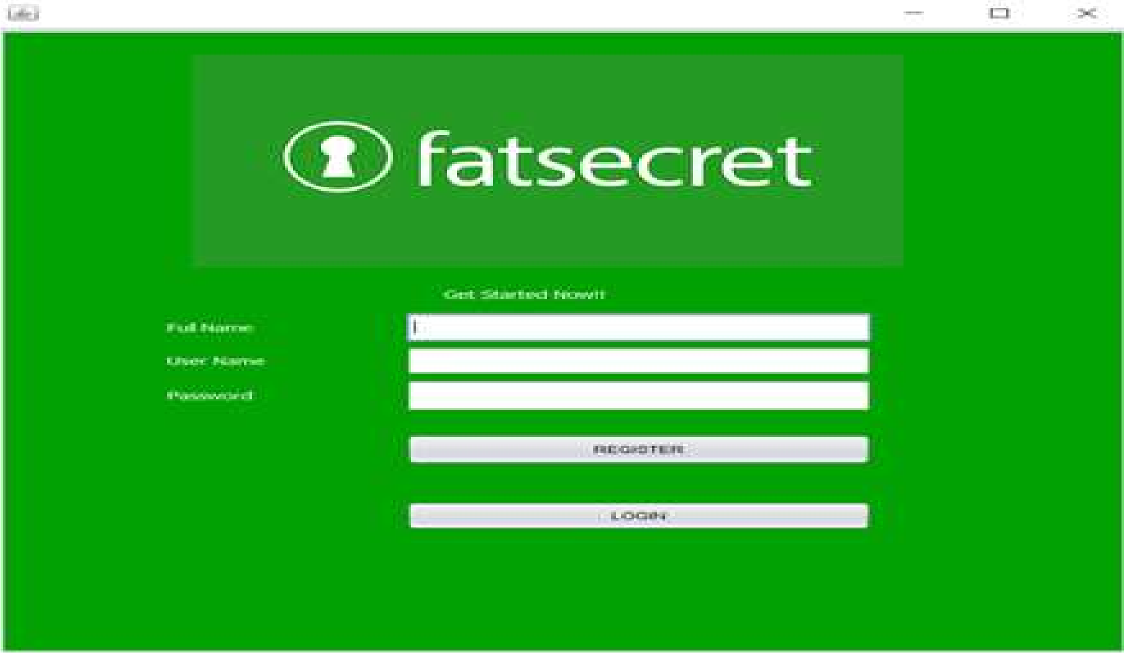
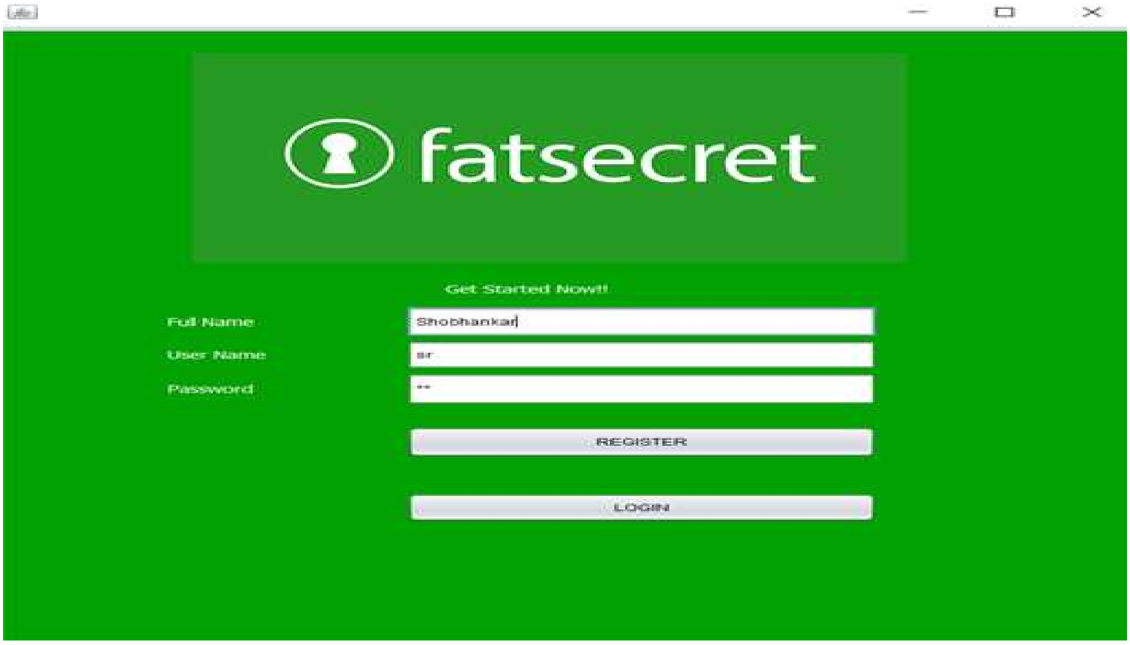


Figure 48 - Application

* User Registration



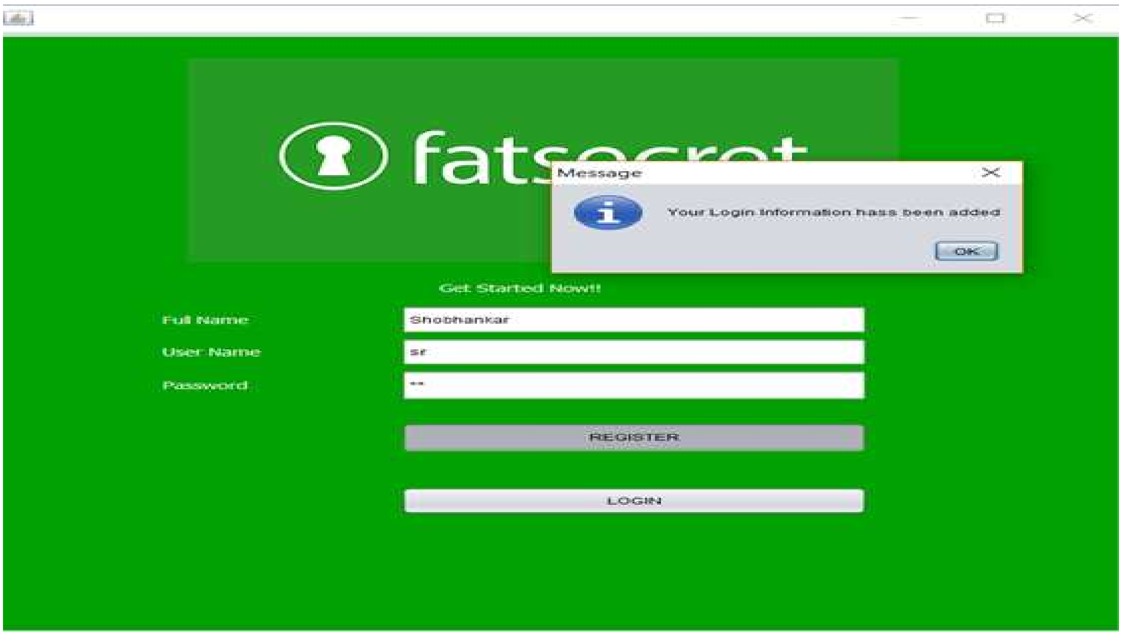
Full Name: Shobhankar

User Name: sr

Password: **

Figure 49 – User Registration

* Successful Registration



Your Login Information hass been added

Full Name: Shobhankar

User Name: sr

Password: **

Figure 50 – Successful Registration

* User Login Page

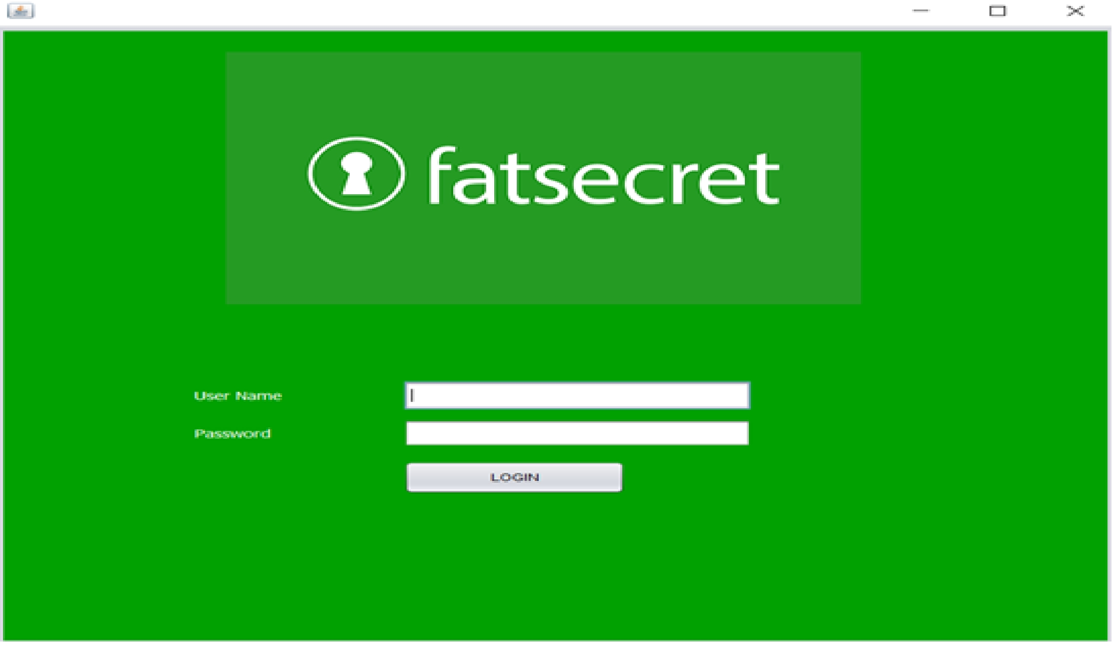
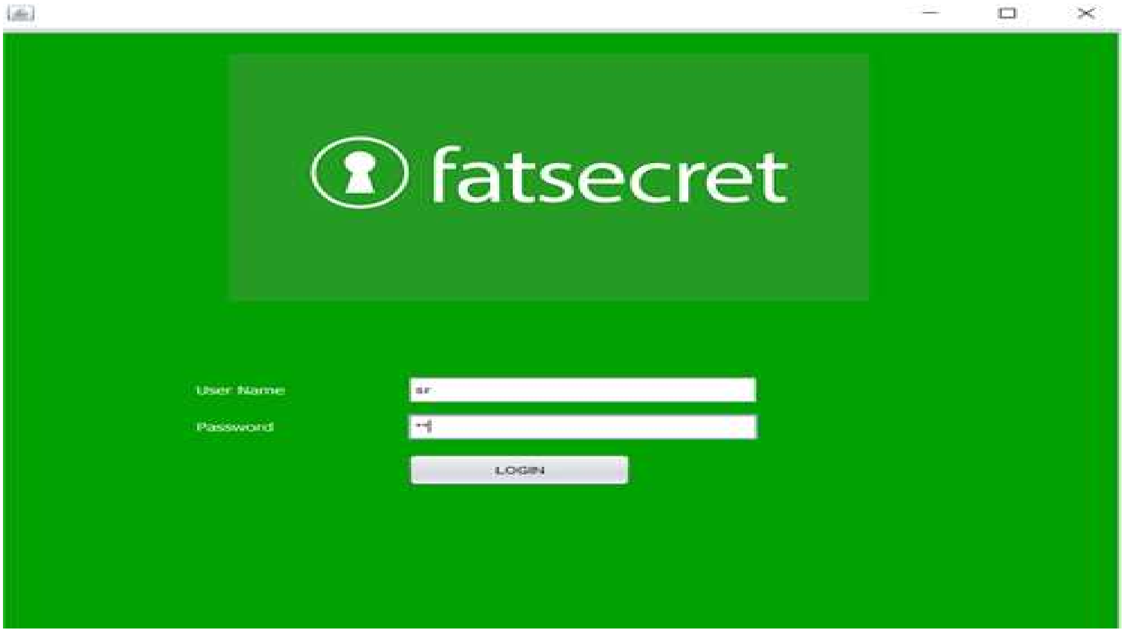


Figure 51 – Login Page

* Login Info

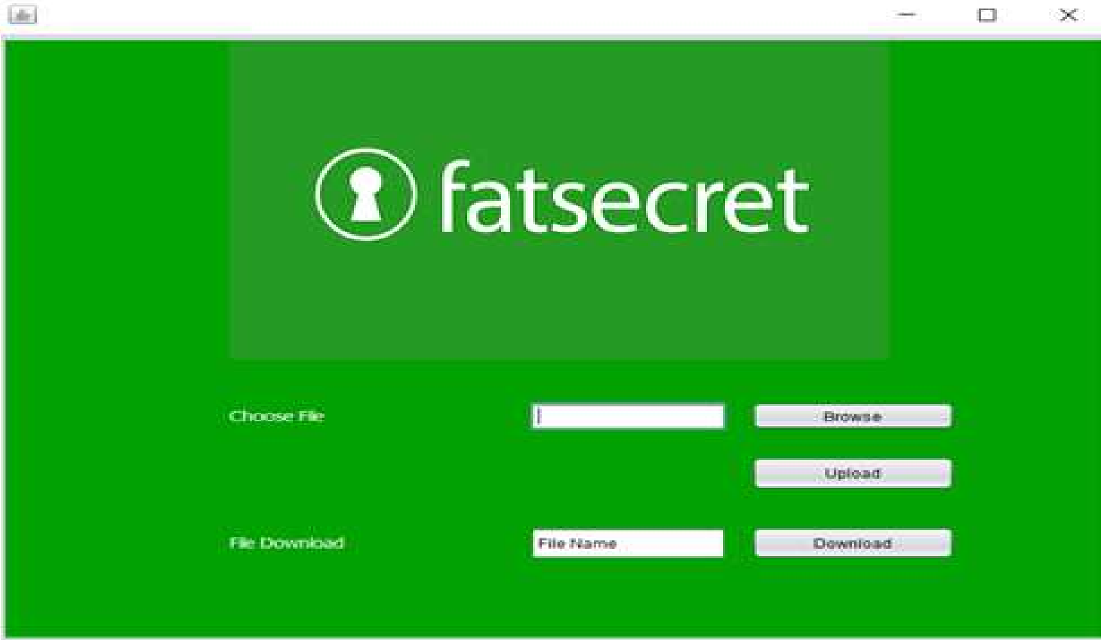


User Name sr

Password

Figure 52 – User Login

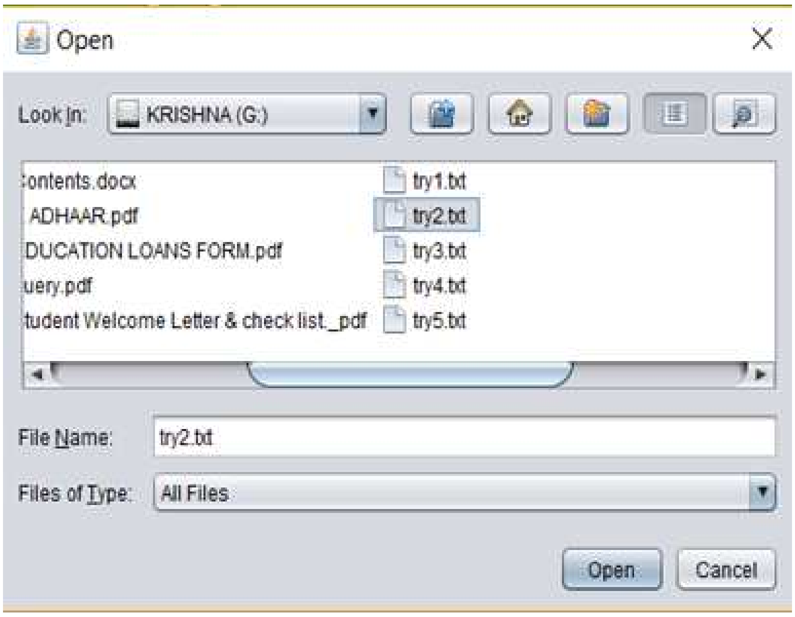
* User Encryption Interface



File Download File Name

Figure 53 – Encryption Interface

* Select file for encryption



Look In: KRISHNA (G:)

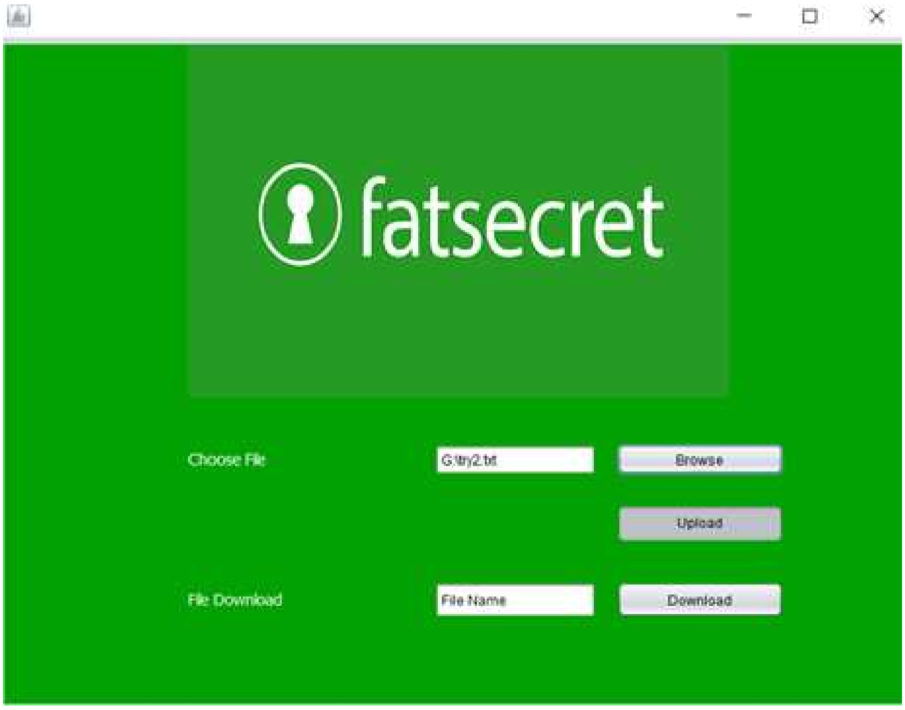
:ontents.docx
ADHAAR.pdf
DUCATION LOANS FORM.pdf
uery.pdf
tudent Welcome Letter & check list._pdf
try1.txt
try2.txt
try3.txt
try4.txt
try5.txt

File Name: try2.txt

Files of Type: All Files

Figure 54 – File Selection

* File Selected

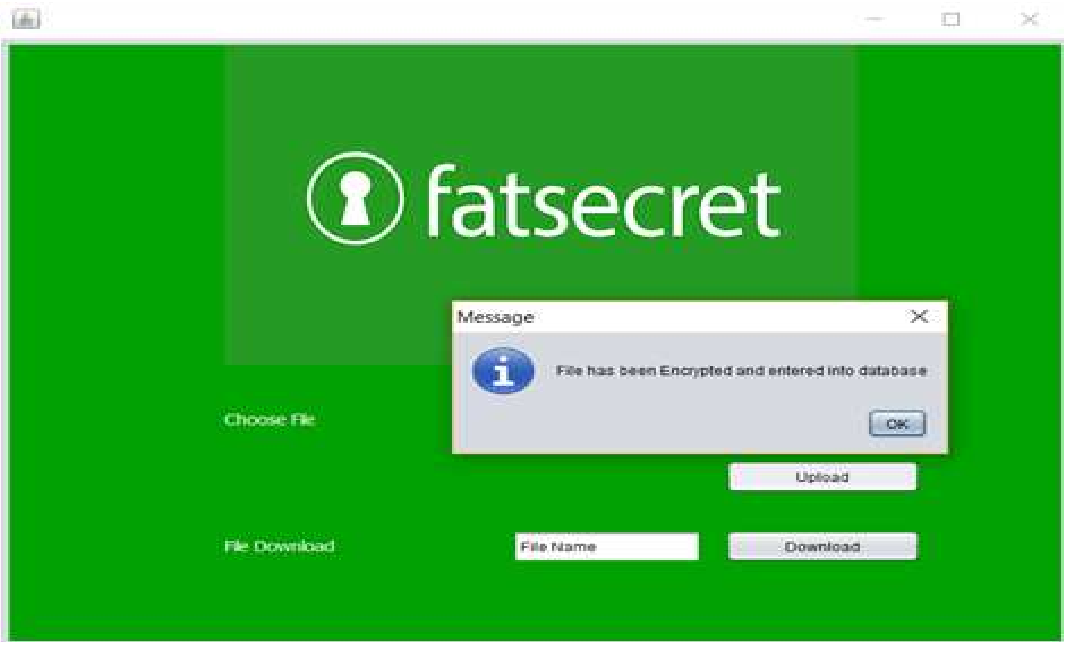


Choose File G:\try2.txt

File Download File Name

Figure 55 – File Selected

* File Encrypted Successfully



Message

File has been Encrypted and entered into database

Figure 56 –Encryption Successful

* Decryption Initiation

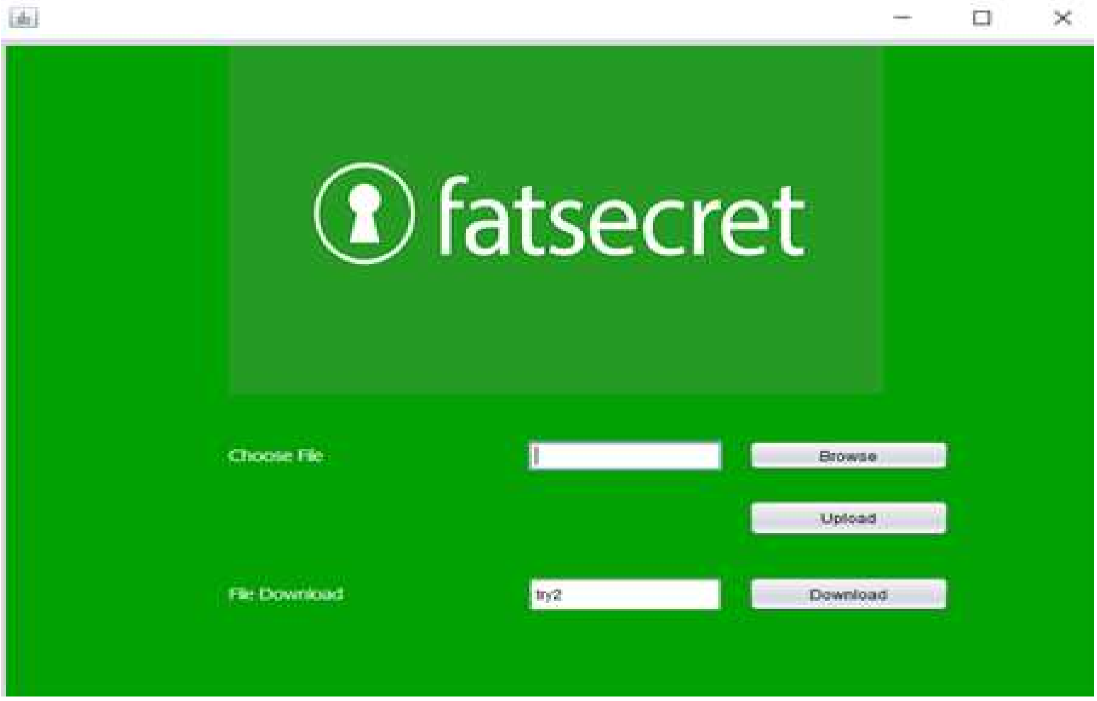
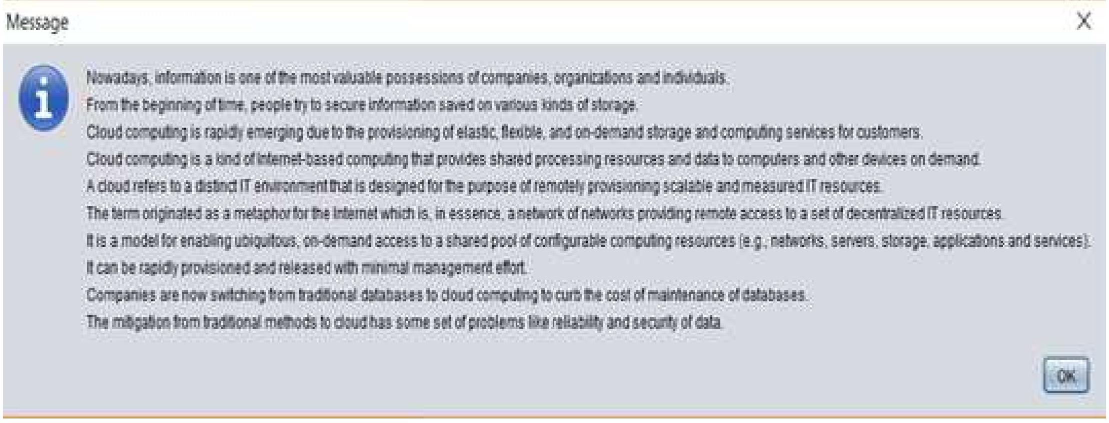


Figure 57 – Decryption Initiation

* Decryption Successful



Nowadays, information is one of the most valuable possessions of companies, organizations and individuals.
From the beginning of time, people try to secure information saved on various kinds of storage.
Cloud computing is rapidly emerging due to the provisioning of elastic, flexible, and on-demand storage and computing services for customers.
Cloud computing is a kind of Internet-based computing that provides shared processing resources and data to computers and other devices on demand.
A cloud refers to a distinct IT environment that is designed for the purpose of remotely provisioning scalable and measured IT resources.
The term originated as a metaphor for the Internet which is, in essence, a network of networks providing remote access to a set of decentralized IT resources.
It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services).
It can be rapidly provisioned and released with minimal management effort.
Companies are now switching from traditional databases to cloud computing to curb the cost of maintenance of databases.
The mitigation from traditional methods to cloud has some set of problems like reliability and security of data.

Figure 58 – Decryption Successful

CHAPTER 5
CONCLUSION

By the analysis of the following results, we were able to conclude that the hybrid algorithm of hybrid ECC and AES provided better execution time in comparison to that of hybrid ECC andRC6. We observed that for relatively small file sizes hybrid EC and RC6 provided better throughput in bytes per millisecond as compared to that of hybrid ECC and AES, but as the size of file increases hybrid ECC and AES showed better results. Taking into consideration, the large amount of data that business applications tend to store on the cloud, file sizes can vary to very large numbers, hence the use of ECC and AES hybrid algorithm is suggested to implement multilevel security on cloud data storage.

5.1) Future scope

On the basis of the analysis of performance of single level encryption algorithms and multi level encryption algorithms, we were able to conclude on an hybrid algorithm to implement on cloud.

In the future part of the project, we aim to combine the algorithm chosen with the application to be deployed on the cloud, which acts as an interface between the user and the cloud. On the cloud we plan to test the hybrid algorithm on various performance parameters in real time usage and against various cryptanalytic attacks in order to check the robustness and reliability of the proposed system.

This project then intends to provide better and enhanced real time security solutions in order to provide more efficient and enhanced user experience with the services utilized on cloud so that the problems of data security, vulnerability and non-repudiation can be solved.

REFERENCES

[1] Muhammad Baqer, et.al , Next Generation of Computing through cloud computing technology. 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), 2012, pp. 1-6.

[2] Heap-Yih Chong, et.al , An explanatory case study on Cloud computing applications in the built environment, Automation in Construction, 44 (2014) 152-162.

[3] Maricela-Georgiana Avram (Olaru) , Advantages and challenges of adopting cloud computing from an enterprise perspective, Procedia technology,12 (2014) 529-534.

[4] R. Velumadhava Rao, et.al, Data security challenges and its solution in cloud computing, International Conference on Intelligent Computing, Communication & Convergence (ICCC), 2015, pp. 204-209.

[5] Shakeeba S. Khan, et.al, Security in Cloud Computing using Cryptographic Algorithms, International Journal of Innovative Research in Computing and Communication Engineering, 3 (2015) 148-154.

[6] Rachna Arora, et.al, Secure User Data in Cloud Computing Using Encryption Algorithm, International Journal of Engineering Research and Applications (IJERA), 3 (2013) 1922-1926.

[7] Randeep Kaur, et.al, Analysis of security algorithm in cloud computing, International Journal of Application or Innovation in Engineering & Management (IJAIEM), 3 (2014) 171-176.

[8] Manpreet Kaur, et.al, Implementing encryption algorithms to enhance data security in cloud computing, International Journal of Computer Applications, *70 (2013) 16-21*.

[9] Mandeep Kaur, et.al, Using Encryption Algorithms to Enhance the Data Security in Cloud Computing, International Journal of Communication and Computer Technologies, 1 (2013) 56-59.

[10] B. Nithya, et.al, A Review of Cryptographic Algorithm in Network Security, International Journal of Engineering and Technology, Vol 8 No 1 Feb-Mar 2016, 324-331

[11] Jagroop Kaur, et.al, Cloud Computing Security Issues and its Solution : A Review, 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)

[12] Patil Madhubala R., et.al, Survey on Security Concerns in Cloud Computing, 2015 International Conference on Green Computing and Internet of Things (ICGCIoT)

[13] S. Subasree, et.al, Design of a New Security Protocol Using Hybrid Cryptography Algorithms, IJRRAS 2 (2) February 2010

[14] Rewagad, et.al, Use of Digital Signature with Diffie Hellman Key Exchange and AES Encryption Algorithm to Enhance Data Security in Cloud Computing, International Conference on Communication Systems and Network Technologies, 2013

[15] AL.Jeeva, et.al, Comparative analysis of Performance Efficiency and Security Measures of Some Encryption Algorithms, International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622

[16] William Stallings, Cryptography and Network Security, Principles and Practice, Sixth Edition, Pearson Publication 2014