Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

Analysis of Cryptography Algorithms for Data Security on Cloud

Author

Md. Shahriar Mahmud

Roll No. 143113

Department of Computer Science & Engineering

Rajshahi University of Engineering & Technology

Supervised by

Dr. Boshir Ahmed

Designation: Professor

Department of Computer Science & Engineering

Rajshahi University of Engineering & Technology

ACKNOWLEDGEMENT

At first, I would like to thank the Almighty Allah for giving me the opportunity and enthusiasm along the way for the completion of my thesis work.

I would like to express our sincere appreciation, gratitude, and respect to our supervisor *Dr. Boshir Ahmed, Professor of Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi*. Throughout the year he has not only given technical guidelines, advice and necessary documents to complete the work but also, he has also given continuous encouragement, advice, helps and sympathetic co-operation whenever he deemed necessary. His continuous support was the most successful tool that helped to achieve my result. Whenever I was stuck in any complex problems or situation, he was there for me at any time of the day. Without his sincere care, this work not has been materialized in the final form that it is now at the present.

I am grateful to him as Head of the Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi. For his extending help in many ways from the department and giving some facilities for students in research areas.

I am also grateful to all the respective teachers of Computer Science and Engineering, Rajshahi University of Engineering and Technology, Rajshahi for good & valuable suggestions and inspirations from time to time.

Finally, I convey my thanks to my parents, friends, and well-wishers for their constant inspirations and many helpful aids throughout this work.

Date: August, 2019

RUET, Rajshahi Md. Shahriar Mahmud

Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

CERTIFICATE

This is to certify that this thesis report entitled ***"Analysis of Cryptography Algorithms for Data Security on Cloud"*** *submitted by* ***Md. Shahriar Mahmud, Roll:143113*** *in partial fulfillment of the requirement for the award of the degree of Bachelor of Science in Computer Science & Engineering of Rajshahi University of Engineering & Technology, Bangladesh is a record of the candidate own work carried out by him under my supervision. This thesis has not been submitted for the award of any other degree.*

Supervisor

Dr. Boshir Ahmed
Professor
Department of Computer Science &Engineering
Rajshahi University of Engineering &Technology
Rajshahi-6204

External Examiner

Dr. Md. Shahid Uz Zaman
Professor
Department of Computer Science &Engineering
Rajshahi University of Engineering &Technology
Rajshahi-6204

ABSTRACT

Cloud Computing is an emerging technology. Day by day it increases its area of computing ability and resources at our hand in everywhere without carrying them. But there are few serious security concerns with cloud computing. Security is one of the main obstacles in the development of this emerging technology. Hence it provides some security services, which will be described but in where client requires extreme security, CSP till could not gain enough trust that is why client often interested in building and managing their own Cloud though it can be costly. In this paper, we proposed a model to secure data in cloud environment. We use existing method with a very small change in key access policy for data. Where the client uses key to encrypt their message with existing Cryptography algorithm to send the Ciphertext and the Cloud Server receives that Ciphertext and stores that while client request for the message then it can reply with the Ciphertext and only client can decrypt that using the key that the client has. There is a number of cryptography algorithms, one can not take an algorithm as best one or an algorithm as worst one (few of algorithm has updated and replaced with other) as there is several comparison parameters and several field of application on numerous numbers of device which has variations of computation power. So we took three algorithm to simulate a proposed model in this thesis work.

Based on this, it is simply illustrated a model of Client and Server using Java Socket Programming for three of Cryptography Algorithm (AES, DES, Blowfish) to analyze performance comparison.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CSP	Cloud Service Provider
AA	Attribute Authority
ABE	Attribute Based Encryption
CP	Ciphertext Policy
KP	Key Policy
IaaS	Infrastructure as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
DSA	Digital Signature Authority
SA	Security Alliance
NIST	National Institute of Standards and Technology
AES	Advance Encryption Standard
DES	Data Encryption Standard
NSA	National Security Agency

Chapter 1

Introduction

Chapter 1
Introduction

1.1 Introduction

Cloud Computing is the name given to a recent trend in computing service provision. This trend has seen the technological and cultural shift of computing service provision from being provided locally to being provided remotely, by third-party service providers. These third parties offer consumers an affordable and flexible computing service provision has evolved from and is the culminated of research stemming from distributed and networked systems, utility computing, the web and software services research. This paradigm shift has led to computing being seen as another household activity and has prompted many a business and individual to migrate parts of their IT infrastructure to the cloud and for this data to become managed and hosted by Cloud Service Provider (CSP). However, Cloud Computing is the backbone among tech pundits and has led to the term 'Cloud Computing' as an umbrella term being applied to differing situation and their solutions. As such a good range of definition for cloud computing. Each of which differ depending on the 'originating authors' learning. This chapter produce good introduction about thesis perspective and outcomes.

1.2 Motivation

The use of encryption schemes is often described through an analogy depicting the transmission on a plaintext message M from one entity, Seder to another entity, Receiver. Here Sender wishes to ensure that only Receiver will be able to read M. This analogy has persisted due to its ability to describe a prevalent communication style, that if Unicast communication. However, this simple analogy does not necessarily represent the entire communication styles that are actively used, it does not take into account *Multicast Communication.*

Traditional symmetric and asymmetric Cryptographic algorithm has several strength and weakness on several condition like efficiency in message size or weakness in encryption or decryption time. Cryptographic algorithm like *Blowfish*, *Advance Encryption*

Standard (AES), Data Encryption Standard (DES), Triple Data Encryption Standard (3DES), RSA has several strengths, weakness, avalanche effect. Depending on these criteria several cryptographic algorithms are used in several area. Where money is not a fact and encryption-decryption time is nothing to worry then there might be device with high computation power and if there is require high security for data then the most secured algorithm can be used but often there is several limitations that few system requires faster, few system requires lighter and may several system has several requirements. None of the cryptographic algorithm serves them all alone. So, it cannot simply avoid or accept a cryptographic algorithm. Almost all of the cryptographic algorithm has some usage depending on purposes. So, it has been essential need to analyze cryptographic algorithm on some several parameters like encryption time, decryption time, memory used, avalanche effect, entropy etc. Depending on these parameter usage areas of cryptographic algorithm may define.

Cloud is become the heart bit of all modern devices which has a communicator. A device alone can perform only a few things because of its hardware and software limitations. But when a device connects with cloud then it crosses its limitation by using remote resources. And now-a-days Cloud Service Provider (CSP) provide several services basically it provides Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS). Organizations often choose XaaS because the as-a-service model can cut costs and simplify IT deployments. With every additional cloud service, an organization can shed pieces of its in-house IT infrastructure, leading to fewer servers, hard drives, network switches, software deployments and more. The combination of cloud computing and ubiquitous, high-bandwidth, global internet access provides a fertile environment for XaaS growth.

Some organizations have been tentative to adopt XaaS because of security, compliance and business governance concerns. However, service providers increasingly address these concerns, allowing organizations to bring additional workloads into the cloud.

However, there is also a huge amount of market share all over the world of Cloud Services. If we focus on growing market of cloud then we can see that every year this market is growing very fast and on 2020 only the *Public Cloud Service will be $411.48*

Billion of U.S dollars[1]. And think about other Cloud Services. The total market already takes a huge part of world trade.

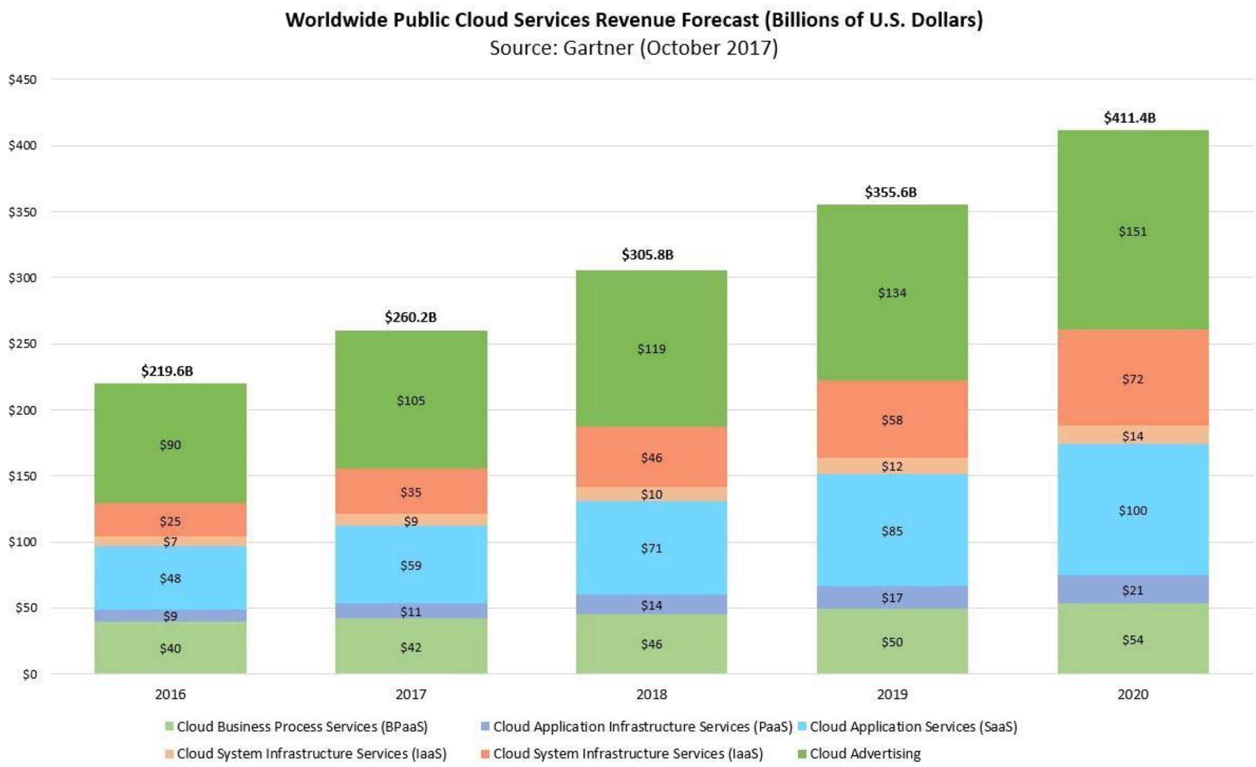


Figure 1.1: Public Cloud Market prepared by Forbes[1]

There are thousands of Cloud Service Provider (CSP) all over the world if it is focused on the market share among themselves then it is seen that alone Amazon, Google and Microsoft Azure take large portion of whole market where the other CSP only occupies **8%** of worldwide market.

Statistics shows that sensitive data like bank and national security related private data often not share even with private cloud, and they used to interest in managing this kind of data by themselves, where managing a whole cloud system is expensive and comes with several managing issues. This is because of lack of trust in CSP. This is the reason of not trusting CSP though they all have Certification by proper authority.

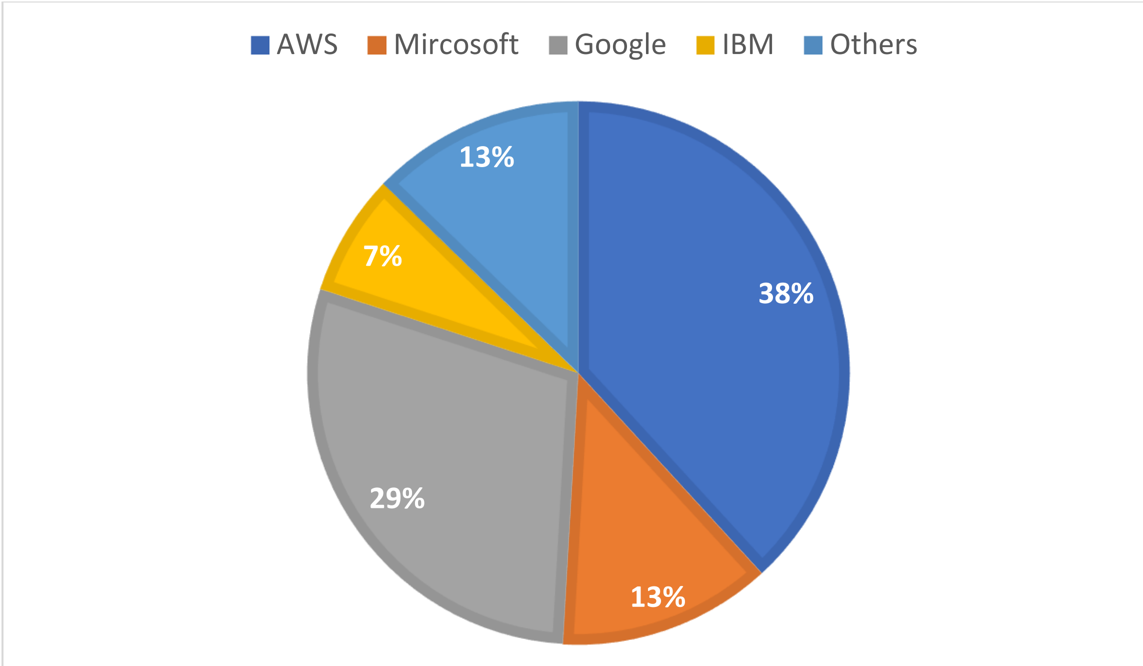


Figure 1.2: CSP Market [1]

1.3 Project and Objectives

This thesis objective is to add an extra layer of security for sensitive data on Cloud Storage. Here it considers few things that help one securing the data onto cloud storage. Develop a Model for this purpose that is intend to secure the data through network channel and data onto network. To get the real time result the proposed model had simulated for getting real-time performance analysis using Socket Programming (Java).

1.4 Research Outcome

In this thesis it will try to understand Cryptography algorithm strength and weakness and working functionalities of these algorithm. And will try to figure out a comparative analysis.

There is several paper review and background study were included on chapter 2, it will discuss about architecture and some issues related with CSP. Hence, it can figure out the problem related with cloud. Cryptography Algorithms, different Cryptography Algorithms strength and weakness, types of Cipher, Stream size, issues relate with data life cycle and overall sort discussion is described on chapter 3. Cloud Computing scenario and difficulties described on chapter 4. Thesis methodology and purposes of thesis will be described in shortly and on chapter 5, Simulation result and comparative analysis of its outcome of this thesis will described shortly in chapter 6. And at the very last on chapter 7, limitations and future work and summarization of thesis is included.

1.5 Conclusions

Cloud Computing is the name given to a recent trend in computing service provision. This trend has seen the technological and cultural shift of computing service provision from being provided locally to being provided remotely, by third-party service providers. This chapter conclude about the cloud market overview and key polices in the cloud. While the key is one of security concern in this area. And In this thesis, we are intendent to apply an issue with key sharing to the service provider.

Chapter 2

Background Study

2.1 Introductions

2.2 Related Work and Their Contribution

2.2.1 Secure Data Access in Cloud Computing.

2.2.2 An Approach towards Data Security in the Cloud Computing Using AES.

2.2.3 Cryptography Algorithms: A Review.

2.2.4 A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish.

2.3 Conclusions

Chapter 2
Background Study

2.1 Introduction

Data privacy and security on cloud is a major concern in everywhere. A distributed system without cloud is impossible to think and apply. A distributed system is totally run from cloud there cloud be several cloud but all of the clouds are connected to each other to serve its purpose in the internet. And if it is discussing about statistics of how cloud market is growing (which will be on Chapter 4) then it is seen that already it took the attention of scientist about privacy. For confidential data privacy is uncompromisable. And the owner of confidential data is so much serious about the security of data and their data must not be accessed by some unauthorized internet user. So, research is done everywhere about how to make it secure. For this securing purpose of data specialist creates three criteria for data privacy and it is CIA (Confidentiality, Integrity, Availability). To serve this purpose of proper privacy of data there has been introduced few of Cryptography Algorithm. Though they have some flaws and strength which need to analyze to find the best way to apply necessary technology in specific field.

2.2 Related Work and Their Contribution

A great research has been done in this area of data privacy about a more secure way of data transition with various tools and techniques. A vast amount of research has been done onto the field of data privacy. From them I had reviewed few papers and from them a few of paper are shortly described.

2.2.1 Secure Data Access in Cloud Computing [2]

Data security and access control is one of the most challenging ongoing research works in cloud computing, because of users outsourcing their sensitive data to cloud providers. Exiting solutions that use pure cryptographic techniques to mitigate these security and access control problems suffer from heavy computational overhead on the data owner as well as the cloud service provide for key distribution and management. This paper address this challenging open problem using capability-based access control technique that ensures only valid users will access the outsourced data. This work also proposes a modified Diffie-Hellman key exchange protocol between cloud service provider and the user for secretly sharing a symmetric key for secure data access that alleviates the problem of key distribution and management at cloud service provider. The simulation run and analysis shows that the proposed approach is highly efficient and secure under existing security models.

If it is discussed about the limitation of this paper then must say that Diffie-Hellman key exchange protocol is already *exploited* successfully against *Man-In-The-Middle* attack but this paper uses Diffie-Hellman key exchange protocol.

2.2.2 An Approach towards Data Security in the Cloud Computing Using AES[3]

With the rapid development of Internet technology, the data of the user ‘s information have raised up largely, so internet storage became more and more important in today ‘s life. The intelligence and networking development of the electronic products, meeting the needs of the public users or the businesses for portable and high capacity has become the most important in development of the information industry. Cloud storage has become the preferred option to provide portable storage service for ordinary users, solve the requirement of large capacity, the difficulty of management and the requirement of high generic extensions. The security mechanism of cloud storage system is also becoming more and more important. By using AES encryption algorithm, the security mechanism of user’s files uploading and downloading has been researched. So, in this paper a new algorithm is introduced, regarding the extent of Cloud Network, the most important feature of the proposed algorithm is its resistivity against the attacks. The algorithm is designed and implemented in java script in *CloudSim* environment. The objective of this paper is the development and creation of a new algorithm by implication of some changes in the initial key of AES encryption algorithm.

2.2.3 Cryptography Algorithms: A Review [4].

Cryptography is derived from Greek word ‘crypto’ means secret ‘grapy’ means writing that is used to conceal the content of message from all except the sender and the receiver and is used to authenticate the correctness of message to the recipient. Today information security is the challenging issue that touches many areas such as computers and communication. Cryptography is such a way that make sure of integrity, availability and identification, confidentiality, authentication of user and as well as security and privacy of data can be provided to the user. In this paper we have defined and analyzed various cryptographic symmetric algorithms like DES, Triple DES, Blowfish, AES and IDEA and asymmetric key cryptographic algorithms like RSA. They have been analyzed on their ability to secure data, key size, block size, features.

2.2.4 A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish [5].

In today's internet era, with online transactions almost every second and terabytes of data being generated every day on the internet, securing information is a challenge. Cryptography is an integral part of modern world information security making the virtual world a safer place. Cryptography is a process of making information unintelligible to an unauthorized person. Hence, providing confidentiality to genuine users. There are various cryptographic algorithms that can be used. Ideally, a user needs a cryptographic algorithm which is of low cost and high performance. However, in reality such algorithm which is a one stop solution does not exist. There are several algorithms with a cost performance trade off. For example, a banking application requires utmost security at high cost and a gaming application sending player pattern for analytics does not bother much about security but needs to be fast and cost effective. Thus, amongst the cryptographic algorithms existing, we choose an algorithm which best fits the user requirements. In, this process of choosing cryptographic algorithms, a study of strengths, weakness, cost and performance of each algorithm will provide valuable insights. In our paper, we have implemented and analyzed in detail cost and performance of popularly used cryptographic algorithms DES, 3DES, AES, RSA and blowfish to show an overall performance analysis, unlike only theoretical comparisons.

2.3 Benefits of Cloud Computing

Hence, we summaries our reviewed papers on benefits of Cloud Computing. Many of the benefits to be had when using Cloud Computing are the lower costs associated. At the infrastructure level, virtual images can be scaled and contracted with complete disregard for any associated hardware costs such as equipment procurement, storage maintenance and use. This is all taken care of by the service provider and will be factored into the payment for the services capital expenditure has been converted into operational expenditure. Resources within the cloud can be treated as a commodity, an ultimate medium. At both the platform and software level similar benefits are seen. Aspects such as software installation, deployment and maintains is virtually non-existent. This is taken care of by the provider within their own infrastructure. The service user only pays technical support. Benefits that offered by Cloud Computing may categories as:

- Flexibility
- Economics of Scale
- Reduce Capital Cost
- Automatic Software Updates
- Pay per Use
- Increased Collaboration
- Work from Anywhere
- Competitiveness
- Environmentally Friendly

2.4 Conclusions

Above all of my reviewed paper describes about security issues and mechanism of security techniques and their performance on different situation. Gathering knowledge from those literature it is tried to insecure the whole architecture in our next few chapters. It will firstly be focused on Cryptography Algorithms and Cloud Computing issues then the architecture of the model will be described.

Chapter 3

Introduction with Cryptography Algorithms

Chapter 3
Introduction with Cryptography Algorithms

3.1 Introduction

In recent years network security has become an important issue. Encryption has come up as a solution, and plays an important role in information security system. Many techniques are needed to protect the shared data. They can be categorized into Symmetric (private) and Asymmetric (public) keys encryption. In Symmetric keys encryption or secret key encryption, only one key is used to encrypt and decrypt data. In Asymmetric keys, two keys are used; private and public keys [6]. Public key is used for encryption and private key is used for decryption (e.g. RSA). Public key encryption is based on mathematical functions, computationally intensive. There are many examples of strong and weak keys of cryptography algorithms like DES, AES. DES uses one 64-bits key while AES uses various 128,192,256 bits keys [7].

3.2 Cryptography

Cryptography is derived from Greek word. It has two parts: 'crypto' means "hidden, secret" and ‘graph’ means “writing". It is the practice and study of techniques for securing communication and data in the presence of adversaries. It is broadly classified into two categories: Symmetric key Cryptography and Asymmetric key Cryptography (popularly known as public key cryptography). Now Symmetric key Cryptography is further categorized as Classical Cryptography and Modern Cryptography. Further drilling down, Classical Cryptography is divided into Transposition Cipher and Substitution Cipher. On the other hand, Modern Cryptography is divided into Stream Cipher and Block Cipher.

3.2.1 Symmetric Key Cryptography

An encryption system in which the sender and receiver of a message share a single, common key that is used to encrypt and decrypt the message.

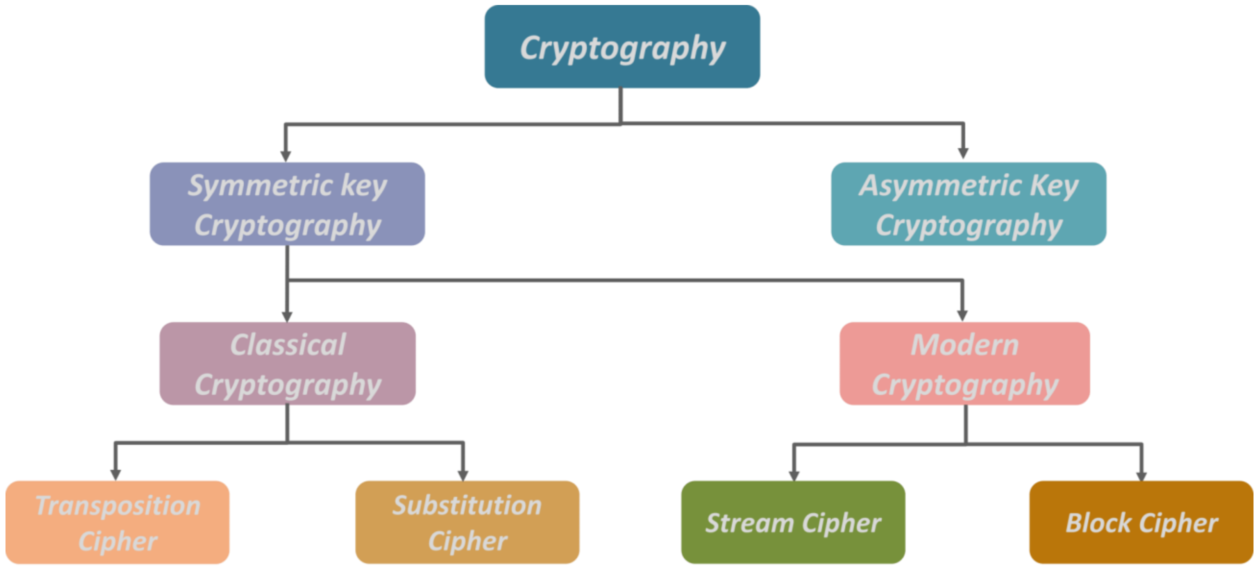


Figure 3.1: Classification of Cryptography [8]

3.2.1.1 Transposition Ciphers

In Cryptography, a transposition cipher is a method of encryption by which the positions held by units of plaintext (which are commonly characters or groups of characters) are shifted according to a regular system, so that the ciphertext constitutes a permutation of the plaintext.

1	2	3	4	5	6
M	E	E	T	M	E
A	F	T	E	R	P
A	R	T	Y		

4	2	1	6	3	5
T	E	M	E	E	M
E	F	A	P	T	R
Y	R	A		T	

Plain Text: MEET ME AFTER PARTY

Key Used: 421635

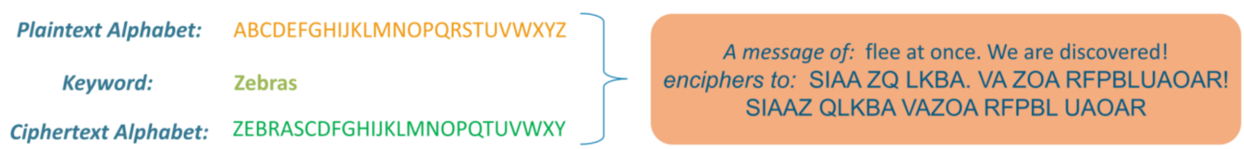
Cipher Text: TEMEEMEFAPTRYRAT

3.2.1.2 Substitution Cipher

Method of encryption by which units of plaintext are replaced with ciphertext, according to a fixed system; the “units” may be single letters (the most common), pairs of letters, triplets of letters, mixtures of the above, and so forth.

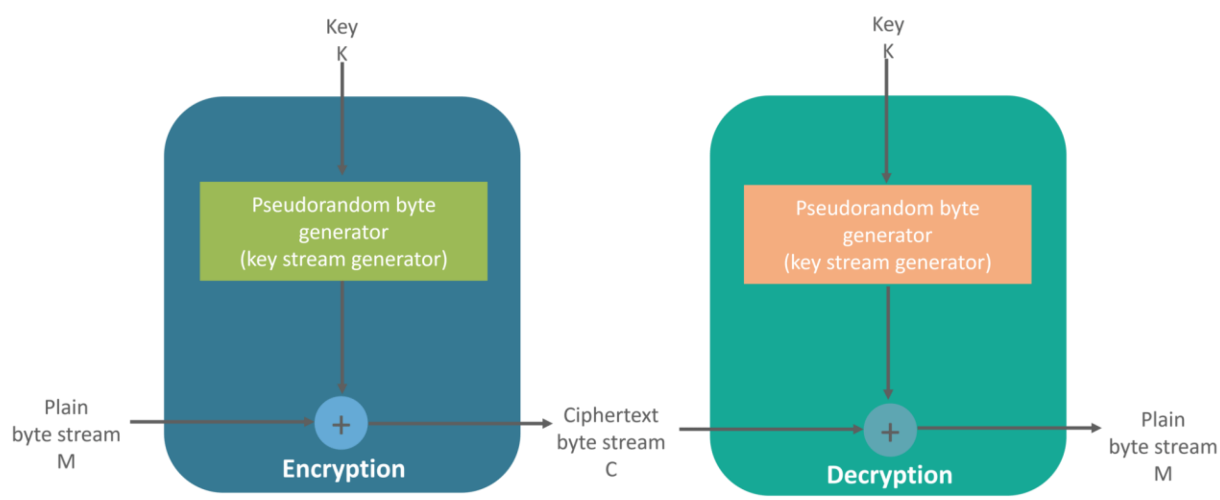
Example:

Consider this example shown on the slide: Using the system just discussed, the keyword “zebras” gives us the following alphabets:



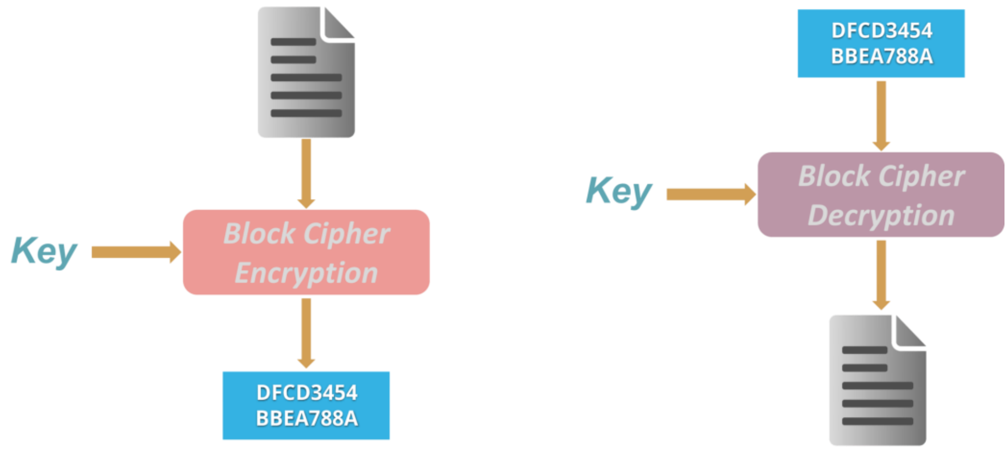
3.2.1.3 Stream Cipher

Symmetric or secret-key encryption algorithm that encrypts a single bit at a time. With a Stream Cipher, the same plaintext bit or byte will encrypt to a different bit or byte every time it is encrypted.



3.2.1.4 Block Cipher

An encryption method that applies a deterministic algorithm along with a symmetric key to encrypt a block of text, rather than encrypting one bit at a time as in stream ciphers.



3.2.2 Asymmetric Key Encryption (or Public Key Cryptography)

The encryption process where different keys are used for encrypting and decrypting the information. Keys are different but are mathematically related, such that retrieving the plain text by decrypting ciphertext is feasible [4].

3.3 Cryptography Goals

There are some goals of cryptography that are given below:

1. **Authentication**: Sender and data receiver must be authenticated before sending and receiving data.
2. **Confidentiality**: The user who is authenticated, can access the messages
3. **Integrity**: Data is free from any kind of modification between sender and receiver.
4. **Non-Repudiation:** The sender the receiver cannot deny that they had sent a message.
5. **Service Reliability:** Attackers can attack on secure systems, which may affect the service of the user.

3.4 Cryptography Algorithms

There is a number of cryptography algorithms, one cannot take an algorithm as best one or an algorithm as worst one (few of algorithm has updated and replaced with other) as there is several comparison parameters and several field of application on numerous numbers of device which has variations of computation power.

1. Data Encryption Standard (DES)

DES is a block encryption algorithm. It was the first encryption standard published by NIST. It is a symmetric algorithm, means same key is used for encryption and decryption. It uses 64-bit key. Out of 64 bits, 56 bits make up the independent key, 8 bits are used for error detection. The main operations are bit permutations and substitution in one round of DES. Six different permutation operations are used both in key expansion part and cipher part. Decryption of DES algorithm is similar to encryption, only the round keys are in reverse order. The output is a 64-bit block. *Many attacks and methods recorded weaknesses of DES, which has made it an insecure block encryption key*.

2. 3DES (Triple DES)

3DES is an enhancement of Data Encryption Standard. It uses 64-bit block size with 192 bits of key size. The encryption method is similar to the original DES but it applied 3 times to increase the safe time and encryption level. Triple DES is *slower* than other block encryption methods. It has the advantage of reliability and *a longer key length that eliminates many shortcut attacks*. 3DES can be used to reduce the amount of time to break DES.

3. AES (Advanced Encryption Standard)

AES also known as the Rijndael's algorithm, is a symmetric block cipher. DES was not secure because of advancement in computer processing power. It has a variable key length of 128, 192 or 256-bits. By default, 256-bit is used. AES encrypts 128 bits data block into 10, 12 and 14 rounds according to the key size. AES can be implemented on various platforms such as small devices encryption of *AES is fast and flexible*. The purpose of NIST was to define a replacement for DES that can be used in nonmilitary information security applications by US government agencies.

4. Blowfish

It is one of the most public domain encryption algorithms. Blowfish was designed in 1993 by Bruce Schneider as a fast alternative to existing encryption algorithms. Blowfish is a symmetric key block cipher that uses a 64-bit block size and variable key length from 32-bits to 448-bits. Blowfish has 16 rounds or less. Blowfish is a *very secure* cipher and to use encryption *free of patents and copyrights*. *No attack is successful against Blowfish*, although *it suffers from weak keys problem*.

5. **IDEA (International Data Encryption Algorithm**)

IDEA is a block cipher algorithm and it operates on 64-bit plaintext blocks. The key size is 128 bits long. The design of algorithm is one of mixing operations from different algebraic groups. Three algebraic groups are mixed, and they are easily implemented in both hardware and software: XOR, Addition modulo 216, Multiplication modulo 216 + 1. All these operations operate on 16-bit sub-blocks. *This algorithm is efficient on 16-bit processors*. IDEA is symmetric key algorithm based on the concept of *Substitution-Permutation Structure*, is a block cipher that uses a 64-bit plain text with 8 rounds and a Key Length of 128-bit permuted into 52 sub-keys each of 128-bits. It does not contain S-boxes and same algorithm is used in reversed for decryption.

6. RC4

RC4 is a stream cipher symmetric key algorithm. As the data stream is simply XOR with generated key sequence. It uses a variable length key 256-bits to initialize a 256-bit state table. A state table is used for generation of pseudo-random bits which is XOR with the plaintext to generate the cipher text.

7. RC6

RC6 is a derivative of RC5. RC6 is designed by Matt Robshaw, Ron Rivest Ray Sidney and is a symmetric key algorithm that is used to congregate the requirements of AES contest. RC6 was also presented to the *CRYPTREC* and *NESSIE* projects. It is patented by RSA Security. RC6 offers *good performance in terms of security and compatibility*. RC6 is a *Feistel Structured private key* algorithm that makes use a 128-bit plain text with 20 rounds and a variable Key Length of 128, 192, and 256-bit. As RC6 works on the principle of RC that can sustain an extensive range of key sizes, word-lengths and number of rounds, RC6 *does not contain S-boxes* and same algorithm is used in reversed for decryption.

8. Serpent

Serpent is an Advanced Encryption Standard (AES) competition, stood 2nd to Rijndael, is a symmetric key block cipher, designed by Eli Biham, Ross Anderson, and Lars Knudsen. Serpent is a symmetric key algorithm that is based on *substitution-permutation network Structure*. It consists of a 128-bit plain text with 32 rounds and a variable Key Length of 128, 192 and 256-bit. It also contains 8 S-boxes and same algorithm is used in reversed for decryption. Security presented by Serpent was based on more conventional approaches than the other AES finalists. *The Serpent is open in the public sphere and not yet patented*.

9. Twofish

Twofish is also a symmetric key algorithm based on the Feistel Structure and was designed by Bruce Schneier along with Doug Whiting, John Kelsey, David Wagner, Niels Ferguson and Chris Hall,. The AES is a block cipher that uses a 128-bit plain text with 16 rounds and a variable Key Length of 128, 192, 256-bit. It makes use of 4 S-boxes (depending on Key) and same algorithm is used in reversed for decryption. The inventors extend the Blowfish team to enhance the earlier block cipher Blowfish to its modified version named Twofish to meet the standards of AES for algorithm designing. It was one of the finalists of the AES, but was not selected for standardization. The Twofish is an open to public sphere and not yet patented.

10. TEA

David Wheeler and Roger Needham (Cambridge Computer Laboratory) in 1994 designed TEA, first presented and published in the proceedings at the Fast Software Encryption workshop. The Tiny Encryption Algorithm (TEA) is known for its simple structure and easy implementation, typically a few lines of code TEA is also a *Feistel Structured symmetric key algorithm*. TEA is a block cipher that uses a 64-bit plain text with 64 rounds and a Key Length of 128-bit with variable rounds having 32 cycles. It does not contain S-boxes and same algorithm is used in reversed for decryption. TEA is designed to maximize speed and minimize memory footprint. Cryptographers have discovered three related-key attacks on TEA. Each TEA key can be found to have three equal keys; thus, it can be used as a hash function. David Wheeler and Roger Needham have proposed extensions of TEA that counter the above attacks.

11. CAST

CAST is symmetric key algorithm based on the backbone concept of Feistel Structure. It is designed by Stafford Taveres and Carlisle Adams, is considered to be a solid algorithm. The CAST is a *block cipher* that uses a 64-bit plain text with 12 or 16 rounds and a variable Key Length of 40 to128-bit. It also contains 4 S-boxes and same algorithm is used in reversed for decryption. Bruce Schneier, John Kelsey, and David Wagner have discovered a related-key attack on the 64 bit of CAST that requires 217 chosen plaintexts, one related query, and 248 offline computations. CAST is patented, which was generously released it for free use.

12. RC2

RC2 is designed by Ron Rivest and a variable-key-size encryption algorithm *from 0 bytes to the maximum string length that the computer system supports*. RC2 is a variable-key-size 64-bit block cipher. It is designed to be a replacement for DES. *RC2 is three times faster than DES* in software implementations. The algorithm encryption speed is independent of key size.

13. RSA

RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman. It was named after the mathematicians who invented it. RSA was first published in 1997. RSA uses variable size key and encryption block. It uses the 2 prime number to generate the public and private key based on mathematical fact and then multiplying large numbers together. It uses the block size data in which plaintext and cipher text are integers between 0 and n1 for some n values. Size of n is considered 1024 bits or 309 decimal digits. In RSA two different keys are used for encryption and decryption purpose. As sender knows encryption key and receiver knows decryption key. Main advantage of RSA algorithm is enhanced security and convenience. Using Public Key Cryptography (PKC) is also an advantage of this algorithm. RSA lacks in encryption speed. RSA may be used to provide both secrecy and digital signature.

14. Diffie-Hellman

This algorithm was introduced in 1976 by Diffie-Hellman. In it, each party generates a key pair and distributes the public key. After obtaining an authentic copy of public keys, then shared secret can be used as the key for a symmetric cipher. The Diffie Hellman algorithm grants two users to establish a shared secret key and to communicate over an insecure communication channel. One-way authentication is free with this type of algorithm. The biggest limitation of this kind of algorithm is communication made using this algorithm is itself vulnerable to man in the middle attack.

15. MD5

MD5's full form is message-digest algorithm. MD5 is derived from MD4 & was designed by Ron Rivest in 1991. MD5 is widely used hash function producing a 128-bit hash value, typically expressed in text format as a 32-digit hexadecimal number. MD5 has been utilized in a wide variety of cryptographic applications, and is also commonly used to verify data integrity.

Table 3.1: Comparison Among Cryptography Algorithms [4]

Algorithm	Created By	Year	Key Size(bits)	Block (bits)	Round	Structure	Flexible	Features
DES	IBM	1975	64	64	16	Feistal	No	Not Strong Enough
3DES	IBM	1978	112 or 168	64	48	Festial	Yes	Adequate Security
AES	Joan Daemen & incent Rijmen	1998	128, 192, 256	128	10, 12, 14	Substitution Permutation	Yes	Replacement for DES, Excelent Security

Blowfish	Bruce Schneier	1993	32-448	64	16	Festial	Yes	Excelent Security
RC4	Ron Rivest	1987	Variable	40-2048	256	Festial Stream	Yes	Fast Cipher in SSL
RC2	Ron Rivest	1987	8-128 64 by default	64	16	Festial	Yes	Stream Cipher
Twofish	Bruce Schneier	1993	128-256	128	16	Festial	Yes	Good Security
Serpent	Anderson, Lars Knudsen	1998	128-256	128	32	Substitution Permutation	Yes	Good Security
IDEA	Ron Rivest, Matt Robshaw	1998	128	64	8.5	Substitution Permutation	No	Not Strong Enough
RC6	Ron Rivest, Matt, Robshaw	1998	128-256	128	20	Festial	Yes	Good Security
RSA	Rivest, Shamir, Adleman	1977	1024-4096	128	1	Public key Algorithm	No	Excellent Security, Low Speed
Deffie Hellman	Whitfield Deffie, Hellman	1976	1024-4096	512	-	Asymmetric Algorithm	Yes	Many Attack
MD5	Ronald Rivest	1992	Series of MD	512	4	Merkle-Damaged Construction		Hash Function

3.5 Conclusion

Internet is mainly used by Individuals, Co-operatives and Governments. They have sent information through internet. But there is a possibility to hack the information. So, to protect information, it is needed to encrypt/decrypt information by using cryptography algorithms. In this research the existing encryption techniques are studied and analyzed to promote the performance of the encryption methods also to ensure the security proceedings. To sum up, all techniques are unique in its own way, which might be suitable for different applications. By Surveying many papers, we had found that throughput value of BLOWFISH is greater than all symmetric algorithms. Power Consumption value of BLOWFISH is least. The experimental results of many papers showed that BLOWFISH has better performance and efficiency than all other block ciphers. The next technique that is widely used to protect our information is RSA. I have read many papers on Cryptography that mainly used RSA algorithm for information security. RSA is the most secure & widely used by researchers. RSA can be used with many techniques like RSA & DES, RSA & AES, RSA & Diffie Hellman, RSA & IDEA, RSA & Blowfish, RSA & Twofish by combining cryptography algorithms to improve security. It had studied many papers on cryptography. Some papers were very good and effective and can be used for future work. In this chapter a detailed analysis of symmetric block encryption algorithms is presented on the basis of different parameters. The main objective was to analyze the performance of the most popular symmetric key algorithms in terms of Authentication, Flexibility, Reliability, Robustness, Scalability, Security, and to highlight the major weakness of the mentioned algorithms, making each algorithm's strength and limitation transparent for application. During this analysis it was observed that AES (Rijndael) was the best among all in terms of Security, Flexibility, Memory usage, and Encryption performance. Although the other algorithms were also competent but most of them have a tradeoff between memory usage and encryption performance with few algorithms been compromised.

Chapter 4

Cloud Computing

Chapter 4
Cloud Computing

4.1 Introduction

This chapter provides an overview of introductory cloud computing topics. It begins with a brief history if cloud computing along with short descriptions of its business and technology drivers. This is followed by definition of basic concepts and terminology, in addition to explanations of the primary benefits and challenged of cloud computing adoption.

4.2 A brief history

The idea of computing in a "Cloud" traces back to the origin of utility computing, a concept that computer scientist John McCarthy publicly proposed in 1961:

"If Computers of the kind I have advocated become the computers of the future, then computing may someday be organized as a public utility just as the telephone system is a public utility. .. The Computer utility cloud become the basis of a new and important industry."

In the late 1990s, Salesforce.com pioneered the notion of bringing remotely provisioned services into the enterprise. A slightly different evocation of the term "Network Cloud" or "Cloud" was introduced in the early 1990s throughout the networking industry.

It was not until 2006 that the term "Cloud Computing" emerged in the commercial arena. It was during this time that amazon launched its Elastic Compute Cloud (EC2) services that enabled organizations to "lease" computing capacity and processing power to run their enterprise application. Google Apps also began providing browser-based enterprise applications in the same year, and three years later, the Google App Engine became another milestone.

4.3 Definition of Cloud Computing

Cloud Computing can be defined as delivering computing power (CPU, RAM, Network Speeds, Storage OS software) a service over a network (usually on the internet) rather than physically having the computing resources at the customer location.

4.4 Why the Name Cloud?

The term "Cloud" came from a network design that was used by network engineers to represent the location of various network devices and their inter-connection. The shape of this network design was like a cloud.

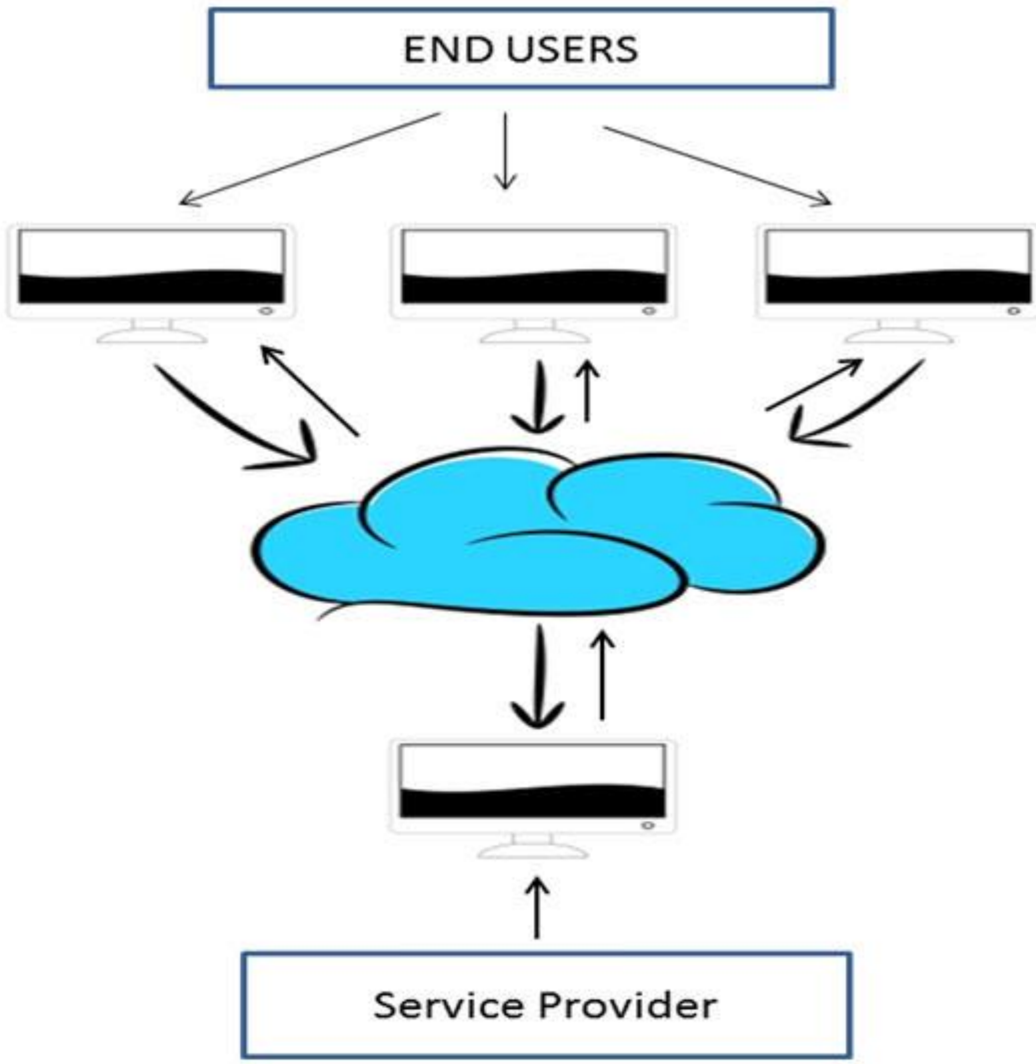


Figure 4.1: Cloud Computing [9]

4.5 Benefits of Cloud Computing

The potential for cost saving is the major reason of cloud services adoption by many organizations. Cloud computing gives the freedom to use services as per the requirement and pay only for what you use. Due to cloud computing it has become possible to run IT operations as a outsourced unit without much in-house resources.

Following are the benefits of cloud computing:

- Lower IT infrastructure and computer costs for users
- Improved performance
- Fewer Maintenance issues
- Instant software updates
- Improved compatibility between Operating systems
- Backup and recovery
- Performance and Scalability
- Increased storage capacity
- Increase data safety

4.6 Characteristics of cloud Computing

1. **Agility** for organizations may be improved, as cloud computing may increase users' flexibility with re-provisioning, adding, or expanding technology infrastructure resources.
2. **Cost** reductions claimed by cloud provides. A public-cloud model converts capital expenditure (e.g., buying servers) to operational expenditure. This purportedly lowers barriers to entry as infrastructure is typically provided by a third party and need not be purchased for one-time or infrequent intensive computing tasks.
3. **Device and location independence** enable users to access systems using a web browser regardless of their location or what device they use (e.g., PC, mobile phone). As infrastructure is off-site (typically provided by a third-party) and accessed via the Internet, Users can connect to it form anywhere.
4. **Maintenance** of cloud competing application is easier, because they do not need to be installed on each user's computer and can be accessed from different places (e.g., different work locations, while travelling, etc.)
5. **Multi-tenancy** enables sharing of resources and costs across a large pool of users thus allowing for:

    - **Centralization** of infrastructure in locations with lower costs (such as real estate, electricity, etc.)

- **Peak-load capacity** increases (users need not engineer and pay for the resources and equipment to meet their highest possible load-levels)
- **Utilization and efficiency** improvements for systems that are often only 10-20 percentages utilized.

6. **Performance** is monitored by IT experts from the service provide and consistent and loosely coupled architectures are constructed using web services as the system interface.
7. **Productivity** may be increased when multiple can work on the same data simultaneously, rather than waiting for it to be saved and emailed. Time may be saved as information does not need to be re-entered when fields are matched, nor do users need to install application software upgrades to their computer.
8. **Reliability** improves with the use of multiple redundant sites, which makes well-designed cloud computing suitable for business continuously and disaster recovery.
9. **Scalability and elasticity** via dynamic("on-demand") provisioning of resources on a fine gained, self-service basis in near real-time (Note the VM startup time varies by VM types, location, Operating Systems and cloud providers)" without users having to engineer for peak loads. This gives the ability to scale up when the usage need increases or down if resources are not being used.
10. **Security** can improve due to centralized of data, increased security-focused resources etc., but concerns can persist about loss of control over certain sensitive data, and the lack of security for stored kernels. Security is often as good as or better than other traditional systems, in part because service providers are able to devote resources to solving security issues that many consumers cannot providers are able to devote resources to solving security issues that many customers cannot afford to tackle or which they lack the technical skills to address. However, the complexity of security is greatly increased when data is distributed over a wider area or over a greater number of devices, as well as in multi-tenant systems shared by unrelated users. In addition, user access to security audit logs may be difficult or impossible. Private cloud installations are in part motivated by user's desire to retain control over the infrastructure and avoid losing control of information security.

4.7 Security issues

Security issues come under many guises both technical and socio- technical in origin. To cover all the security issues possible within the cloud, and in-depth, would be herculean a task not suited even for Heracles himself. Existing efforts look to provide a taxonomy over the issues seen. The cloud Security Alliance is a non-profit organization that seeks to promote the best practices for providing security assurance within the cloud computing landscape. The Cloud Security Alliance (CSA) identify seven threats to cloud competing that can be interpreted as a classification of security issues found within the cloud.

They are –

1. Abuse and Nefarious use of Cloud Computing.
2. Insecure Application Programming Interfaces
3. Malicious Insiders
4. Shared Technology Vulnerabilities
5. Data Loss/Leakage
6. Account, Service and Traffic Hijacking
7. Unknown Risk Profile

There are few other security issues comes with these issues, those are also a serious concern about cloud computing.

A. Cloud Computing Security

Wikipedia [10] defines Cloud Computing Security as “Cloud computing security (sometimes referred to simply as "cloud security") is an evolving sub-domain of computer security, network security, and, more broadly, information security. It refers to a broad set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing.” Note that cloud computing security referred to here is not cloud-based security software products such as cloud-based anti-virus, anti-spam, anti-DDoS, and so on.

B. Security Issues Associated with the Cloud

There are many security issues associated with cloud computing and they can be grouped into any number of dimensions.

According to Gartner [11], before making a choice of cloud vendors, users should ask the vendors for seven specific safety issues: *Privileged user access*, *regulatory compliance, data location*, *data segregation*, *recovery*, *investigative support and long-term viability*. In 2009, Forrester Research Inc. [13] evaluated security and privacy practices of some of the leading cloud providers (such as Salesforce.com, Amazon, Google, and Microsoft) in three major aspects: Security and privacy, compliance, and legal and contractual issues. Cloud Security Alliance (CSA) [11] is gathering solution providers, non-profits and individuals to enter into discussion about the current and future best practices for information assurance in the cloud. The CSA has identified thirteen domains of concerns on cloud computing security [14].

S. Subashini and V. Kavitha made an investigation of cloud computing security issues from the cloud computing service delivery models (SPI model) and give a detailed analysis and assessment method description for each security issue [9]. . Mohamed Al Morsy, John Grundy and Ingo Müller explored the cloud computing security issues from different perspectives, including security issues associated with cloud computing architecture, service delivery models, cloud characteristics and cloud stakeholders [10]. Yanpei Chen, Vern Paxson and Randy H. Katz believed that two aspects are to some degree new and essential to cloud: the complexities of multi-party trust considerations, and the ensuing need for mutual auditability. They also point out some new opportunities in cloud computing security [11].

According to the SPI service delivery models, deployment models and essential characteristics of cloud, there are security issues in all aspects of the infrastructure including network level, host level and application level.

.

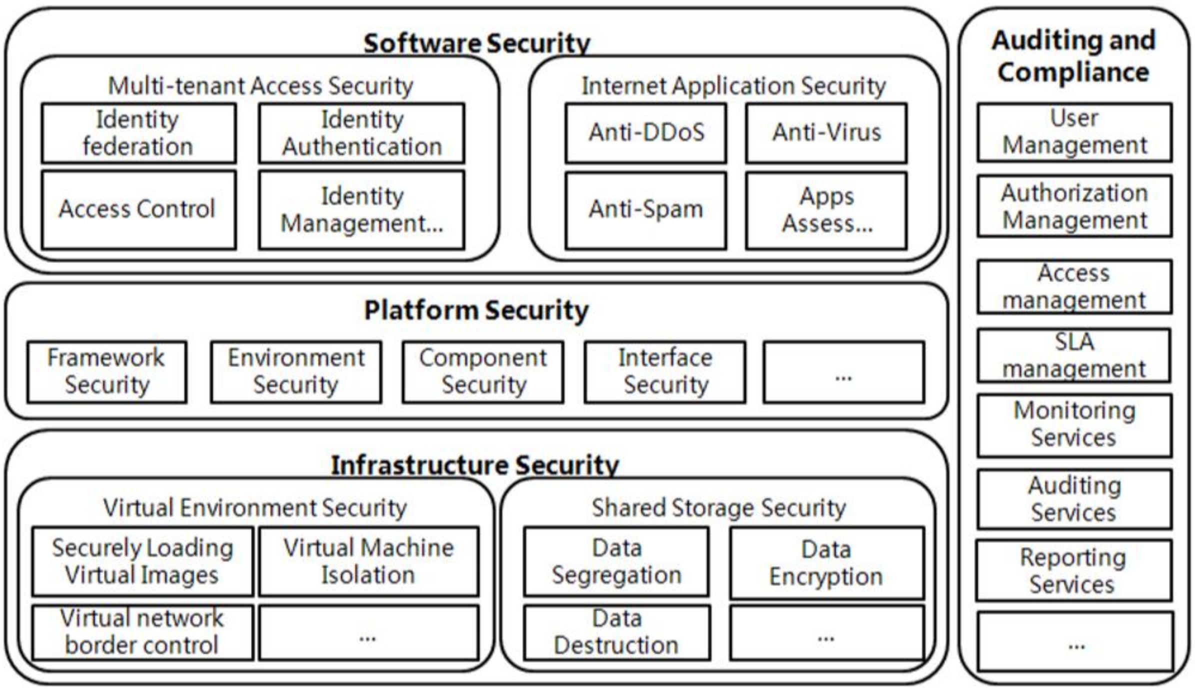


Figure 4.2: Cloud computing security architecture [15]

4.8 Data Security and Privacy Protecting Issues

The content of data security and privacy protection in cloud is similar to that of traditional data security and privacy protection. It is also involved in every stage of the data life cycle. But because of openness and multi-tenant characteristic of the cloud, the content of data security and privacy protection in cloud has its particularities. The concept of privacy is very different in different countries, cultures or jurisdictions. The definition adopted by Organization for Economic Cooperation and Development (OECD) [13] is "*any information relating to an identified or identifiable individual (data subject)*." Another popular definition provided by the American Institute of Certified Public Accountants (AICPA) and the Canadian Institute of Chartered Accountants (CICA) in the Generally Accepted Privacy Principles (GAPP) standard is “The rights and obligations of individuals and organizations with respect to the collection, use, retention, and disclosure of personal information.” Generally speaking, privacy is associated with the collection, use, disclosure, storage, and destruction of personal data (or personally identifiable information, PII).

Identification of private information depends on the specific application scenario and the law, and is the primary task of privacy protection.

The next several sections analyze data security and privacy protection issues in cloud around the data life cycle.

A. Data Life Cycle

Data life cycle refers to the entire process from generation to destruction of the data. The data life cycle is divided into seven stages. See the figure below:

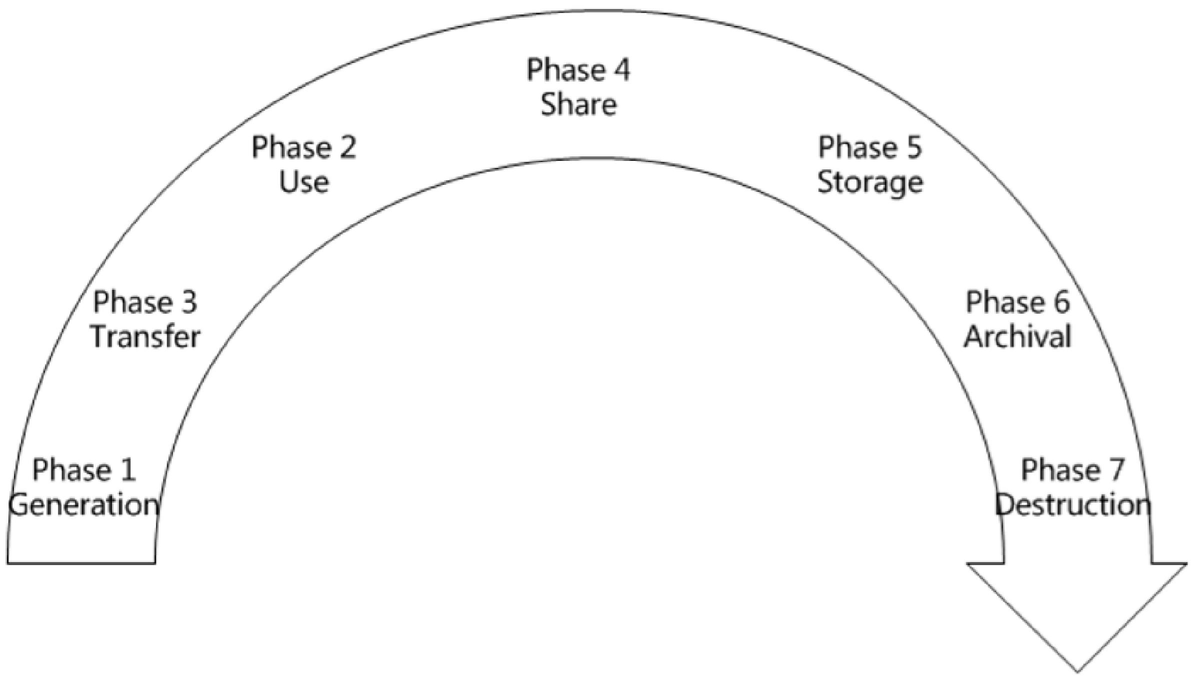


Figure 4.3. Data life cycle [15]

B. Data Generation

Data generation is involved in the data ownership. In the traditional IT environment, usually users or organizations own and manage the data. But if data is to be migrated into cloud, it should be considered that how to maintain the data ownership. For personal private information, data owners are entitled to know what personal information being collected, and in some cases, to stop the collection and use of personal information.

C. Transfer

Within the enterprise boundaries, data transmission usually does not require encryption, or just have a simple data encryption measure. For data transmission across enterprise boundaries, both data confidentiality and integrity should be ensured in order to prevent data from being tapped and tampered with by unauthorized users. In other words, only the data encryption is not enough. Data integrity is also needed to be ensured. Therefore, it should ensure that transport protocols provide both confidentiality and integrity. Confidentiality and integrity of data transmission need to ensure not only between enterprise storage and cloud storage but also between different cloud storage services. In other words, confidentiality and integrity of the entire transfer process of data should be ensured.

D. Use

For the static data using a simple storage service, such as Amazon S3, data encryption is feasible. However, for the static data used by cloud-based applications in PaaS or SaaS model, data encryption in many cases is not feasible. Because data encryption will lead to problems of indexing and query, the static data used by Cloud-based applications is generally not encrypted. Not only in cloud, but also in traditional IT environment, the data being treated is almost not encrypted for any program to deal with. Due to the multi-tenant feature of cloud computing models, the data being processed by cloud-based applications is stored together with the data of other users. Unencrypted data in the process is a serious threat to data security.

Regarding the use of private data, situations are more complicated. The owners of private data need to focus on and ensure whether the use of personal information is consistent with the purposes of information collection and whether personal information is being shared with third parties, for example, cloud service providers.

E. Share

Data sharing is expanding the use range of the data and renders data permissions more complex. The data owners can authorize the data access to one party, and in turn the party can further share the data to another party without the consent of the data owners.

Therefore, during data sharing, especially when shared with a third party, the data owners need to consider whether the third party continues to maintain the original protection measures and usage restrictions.

Regarding sharing of private data, in addition to authorization of data, sharing granularity (all the data or partial data) and data transformation are also need to be concerned about. The sharing granularity depends on the sharing policy and the division granularity of content. The data transformation refers to isolating sensitive information from the original data. This operation makes the data is not relevant with the data owners.

F. Storage

The data in the cloud may be divided into: (1) The data in IaaS environment, such as Amazon's Simple Storage Service; (2) The data in PaaS or SaaS environment related to cloud-based applications.

The data stored in the cloud storages is similar with the ones stored in other places and needs to consider three aspects of information security: confidentiality, integrity and availability.

The common solution for data confidentiality is data encryption. In order to ensure the effective of encryption, there needs to consider the use of both encryption algorithm and key strength. As the cloud computing environment involving large amounts of data transmission, storage and handling, there also needs to consider processing speed and computational efficiency of encrypting large amounts of data. In this case, for example, symmetric encryption algorithm is more suitable than asymmetric encryption algorithm.

Another key problem about data encryption is key management. Is who responsible for key management? Ideally, it’s the data owners. But at present, because the users have not enough expertise to manage the keys, they usually entrust the key management to the cloud service providers. As the cloud providers need to maintain keys for a large number of users, key management will become more complex and difficult.

In addition to data confidentiality, there also needs to be concerned about data integrity. When the users put several GB (or more) data into the cloud storage, they how to check the integrity of the data? As rapid elasticity feature of cloud computing resources, the users don’t know where their data is being stored. To migrate out of or into the cloud storage will consume the user's network utilization (bandwidth) and an amount of time. And some cloud providers, such as Amazon, will require users to pay transfer fees. How to directly verify the integrity of data in cloud storage without having to first download the data and then upload the data is a great challenge. As the data is dynamic in cloud storage, the traditional technologies to ensure data integrity may not be effective.

In the traditional IT environment, the main threat of the data availability comes from external attacks. In the cloud, however, in addition to external attacks, there are several other areas that will threat the data availability:

(1*) The availability of cloud computing services;*

(2) Whether the cloud providers would continue to operate in the future?

(3) Whether the cloud storage services provide backup?

G. Archival

Archiving for data focuses on the storage media, whether to provide off-site storage and storage duration. If the data is stored on portable media and then the media is out of control, the data are likely to take the risk of leakage. If the cloud service providers do not provide off-site archiving, the availability of the data will be threatened. Again, whether storage duration is consistent with archival requirements? Otherwise, this may result in the availability or privacy threats.

H. Destruction

When the data is no longer required, whether it has been completely destroyed? Due to the physical characteristics of storage medium, the data deleted may still exist and can be restored. This may result in inadvertently disclose of sensitive information.

4.9 Abuse and Nefarious Use of Cloud Computing

Legitimate CSPs can be abused for nefarious purposes, supporting criminal or other untoward activities toward customers. For instance, services can be used to host malicious code or used to facilitate communication between remote entities. The emphasis is that legitimate services are used either malicious intent in mind. Other issues seen include the provision of purposefully insecure service used for data capture.

4.10 Insecure Interface

Data placed in the Cloud will be accessed through application programming Interfaces (APIs) and other interfaces. Malfunction and error in interface software, and also the software used to run the Cloud, can lead to the unwanted exposure of the user's data and impugn upon the data integrity. For example, a flaw in Apache, a popular HTTP server, allowed an attacker 7to gain complete control over the web server. Data Exposure can also occur when a software malfunction affects the access policy governing user's data. This has been seen in several Cloud based services in which a software malfunction resulted in which a user privacy setting was overwritten and the user data exposed to non-authorized entities. Threats can also exit as a result of poorly designed or implemented security measures. If these measures can be bypassed, or are non0existent, the software can be easily abuse by malicious entities. Regardless of the thread origin, APIs and other interfaces need to be made secure against accidental and malicious attempts to circumvent the APIs and their security measures.

4.11 Important Security Threads

There is several security Threads which might be depend on the organization architecture and with few other related properties. Those security threads are shortly described below.

A. Data breaches

Cloud environments face many of the same threats as traditional corporate networks, but due to the vast amount of data stored on cloud servers, providers become an attractive target. The severity of potential damage tends to depend on the sensitivity of the data exposed. Exposed personal financial information tends to get the headlines, but breaches involving health information, trade secrets, and intellectual property can be more devastating. When a data breach occurs, companies may incur fines, or they may face lawsuits or criminal charges. Breach investigations and customer notifications can rack up significant costs. Indirect effects, such as brand damage and loss of business, can impact organizations for years.

B. Compromised credentials and broken authentication

Data breaches and other attacks frequently result from lax authentication, weak passwords, and poor key or certificate management. Organizations often struggle with identity management as they try to allocate permissions appropriate to the user's job role. More important, they sometimes forget to remove user access when a job function changes or a user leaves the organization.

Multifactor authentication systems such as one-time passwords, phone-based authentication, and smartcards protect cloud services because they make it harder for attackers to log in with stolen passwords. The Anthem breach, which exposed more than 80 million customer records, was the result of stolen user credentials.

Many developers make the mistake of embedding credentials and cryptographic keys in source code and leaving them in public-facing repositories such as GitHub. Keys need to be appropriately protected, and a well-secured public key infrastructure is necessary. They also need to be rotated periodically to make it harder for attackers to use keys they've obtained without authorization.

Organizations planning to federate identity with a cloud provider need to understand the security measures the provider uses to protect the identity platform. Centralizing identity into a single repository has its risks. Organizations need to weigh the trade-off of the convenience of centralizing identity against the risk of having that repository become an extremely high-value target for attackers.

C. Hacked interfaces and APIs

Practically every cloud service and application now offer APIs. IT teams use interfaces and APIs to manage and interact with cloud services, including those that offer cloud provisioning, management, orchestration, and monitoring.

The security and availability of cloud services - from authentication and access control to encryption and activity monitoring - depend on the security of the API. Risk increases with third parties that rely on APIs and build on these interfaces, as organizations may need to expose more services and credentials.

Weak interfaces and APIs expose organizations to security issues related to confidentiality, integrity, availability, and accountability.

APIs and interfaces tend to be the most exposed part of a system because they're usually accessible from the open Internet. It is recommending adequate controls as the “first line of defense and detection.” Threat modeling applications and systems, including data flows and architecture/design, become important parts of the development lifecycle. It is also recommending security-focused code reviews and rigorous penetration testing.

D. Exploited system vulnerabilities

System vulnerabilities, or exploitable bugs in programs, are not new, but they've become a bigger problem with the advent of multi-tenancy in cloud computing. Organizations share memory, databases, and other resources in close proximity to one another, creating new attack surfaces.

Fortunately, attacks on system vulnerabilities can be mitigated with “basic IT processes”. Best practices include regular vulnerability scanning, prompt patch management, and quick follow-up on reported system threats.

The costs of mitigating system vulnerabilities “are relatively small compared to other IT expenditures” The expense of putting IT processes in place to discover and repair vulnerabilities is small compared to the potential damage. Regulated industries need to patch as quickly as possible, preferably as part of an automated and recurring process. Change control processes that address emergency patching ensure that remediation activities are properly documented and reviewed by technical teams.

E. Account hijacking

Phishing, fraud, and software exploits are still successful, and cloud services add a new dimension to the threat because attackers can eavesdrop on activities, manipulate transactions, and modify data. Attackers may also be able to use the cloud application to launch other attacks.

Common defense-in-depth protection strategies can contain the damage incurred by a breach. Organizations should prohibit the sharing of account credentials between users and services, as well as enable multifactor authentication schemes where available. Accounts, even service accounts, should be monitored so that every transaction can be traced to a human owner. The key is to protect account credentials from being stolen.

F. Malicious insiders

The insider threat has many faces: a current or former employee, a system administrator, a contractor, or a business partner. The malicious agenda ranges from data theft to revenge. In a cloud scenario, a hell-bent insider can destroy whole infrastructures or manipulate data. Systems that depend solely on the cloud service provider for security, such as encryption, are at greatest risk.

It is recommending that organizations control the encryption process and keys, segregating duties and minimizing access given to users. Effective logging, monitoring, and auditing administrator activities are also critical.

It's easy to misconstrue a bungling attempt to perform a routine job as "malicious" insider activity. An example would be an administrator who accidentally copies a sensitive customer database to a publicly accessible server. Proper training and management to prevent such mistakes becomes more critical in the cloud, due to greater potential exposure.

G. The APT parasite

It is advanced persistent threats (APTs) “parasitical” forms of attack. APTs infiltrate systems to establish a foothold, then stealthily exfiltrate data and intellectual property over an extended period of time.

APTs typically move laterally through the network and blend in with normal traffic, so they're difficult to detect. The major cloud providers apply advanced techniques to prevent APTs from infiltrating their infrastructure, but customers need to be as diligent in detecting APT compromises in cloud accounts as they would in on-premises systems.

Common points of entry include spear phishing; direct attacks, USB drives preloaded with malware, and compromised third-party networks. In particular, recommends training users to recognize phishing techniques.

Regularly reinforced awareness programs keep users alert and less likely to be tricked into letting an APT into the network -- and IT departments need to stay informed of the latest advanced attacks. Advanced security controls, process management, incident response plans, and IT staff training all lead to increased security budgets. Organizations should weigh these costs against the potential economic damage inflicted by successful APT attacks.

H. Permanent data loss

As the cloud has matured, reports of permanent data loss due to provider error have become extremely rare. But malicious hackers have been known to permanently delete cloud data to harm businesses, and cloud data centers are as vulnerable to natural disasters as any facility.

Cloud providers recommend distributing data and applications across multiple zones for added protection. Adequate data backup measures are essential, as well as adhering to best practices in business continuity and disaster recovery. Daily data backup and off-site storage remain important with cloud environments.

The burden of preventing data loss is not all on the cloud service provider. If a customer encrypts data before uploading it to the cloud, then that customer must be careful to protect the encryption key. Once the key is lost, so is the data.

Compliance policies often stipulate how long organizations must retain audit records and other documents. Losing such data may have serious regulatory consequences. The new EU data protection rules also treat data destruction and corruption of personal data as data breaches requiring appropriate notification. Know the rules to avoid getting in trouble.

I. Inadequate diligence

Organizations that embrace the cloud without fully understanding the environment and its associated risks may encounter a "myriad of commercial, financial, technical, legal, and compliance risks". Due diligence applies whether the organization is trying to migrate to the cloud or merging (or working) with another company in the cloud. For example, organizations that fail to scrutinize a contract may not be aware of the provider's liability in case of data loss or breach.

Operational and architectural issues arise if a company's development team lacks familiarity with cloud technologies as apps are deployed to a particular cloud. Organizations they must perform extensive due diligence to understand the risks they assume when they subscribe to each cloud service.

J. Cloud service abuses

Cloud services can be commandeered to support nefarious activities, such as using cloud computing resources to break an encryption key in order to launch an attack. Other examples including launching DDoS attacks, sending spam and phishing emails, and hosting malicious content.

Providers need to recognize types of abuse - such as scrutinizing traffic to recognize DDoS attacks - and offer tools for customers to monitor the health of their cloud environments. Customers should make sure providers offer a mechanism for reporting abuse. Although customers may not be direct prey for malicious actions, cloud service abuse can still result in service availability issues and data loss.

K. DoS attacks

DoS attacks have been around for years, but they have gained prominence again thanks to cloud computing because they often affect availability. Systems may slow to a crawl or simply time out. "Experiencing a denial-of-service attack is like being caught in rush-hour traffic gridlock; there is one way to get to your destination and there is nothing you can do about it except sit and wait," the report said.

DoS attacks consume large amounts of processing power, a bill the customer may ultimately have to pay. While high-volume DDoS attacks are very common, organizations should be aware of asymmetric, application-level DoS attacks, which target Web server and database vulnerabilities. Cloud providers tend to be better poised to handle DoS attacks than their customers. The key is to have a plan to mitigate the attack before it occurs, so administrators have access to those resources when they need them.

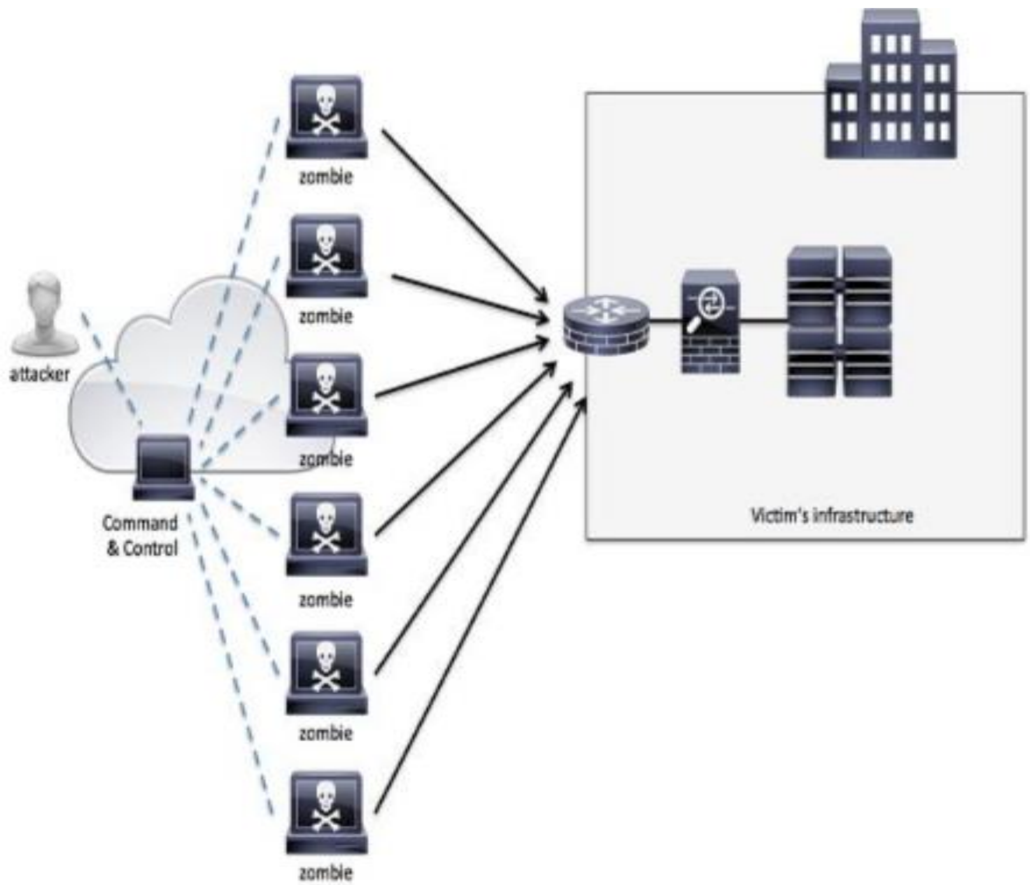


Figure 4.4: DoS attack[20]

L. Shared technology, shared dangers

Vulnerabilities in shared technology pose a significant threat to cloud computing. Cloud service providers share infrastructure, platforms, and applications, and if a vulnerability arises in any of these layers, it affects everyone. "A single vulnerability or miss-configuration can lead to a compromise across an entire provider's cloud".

If an integral component gets compromised - say, a hypervisor, a shared platform component, or an application - it exposes the entire environment to potential compromise and breach. It is recommended a defense-in-depth strategy, including multifactor authentication on all hosts, host-based and network-based intrusion detection systems, applying the concept of least privilege, network segmentation, and patching shared resources.

4.12 Conclusions

In this chapter we have described a brief history and necessity of cloud computing. We also described about the relationship between distrusted system and Cloud Computing. Hence the cloud computing classification and characteristics also described with the security issues relate with each of it. Hence there was also described about the abuse and nefarious Use of Cloud Computing. Cloud Insecure interface also came with its architecture. Overall a Cloud Computing security related issues were described shortly.

Chapter 5

Methodology

Chapter 5
Methodology

5.1 Introduction

Cloud Computing is very fast-growing technology. It has changed look of peoples toward processing technique of data. Because of its flexibility, affectivity, economy of scale its growing like a tree. Privacy is an ancient thought of data security in this growing field. There is already in terms of personal definition security in everywhere assured by some of security provider organization. Meanwhile there is some serious security flaws in transmission channel or for the weakness of Cloud management system. In distributed system like internet there is every source is connected so any malicious activity can be in here. For this reason, those who need an ultimate security like bank or national issues need an ultimate security. But in reality, no organization can provide true privacy and security for the data. As booth the Client and CSP has the key to retrieve data.

5.2 Problem with the Existing Method

Cloud market [17] is a very fast-growing field where we previously described about its field and organization who occupies its total market share. If we analyze that we observe that 96% of total market is occupied by only four of organization. And these four organizations are

i. Amazon Web Services (42%)
ii. Microsoft Azure (32%)
iii. Google (14%)
iv. IBM (8%)

Where there is moreover thousand number of CS all over the world. Hence other than this Four organization occupies only 8% of total market. We can assume that in this 8% organization most of the organization are built to serve their own purposes.

If we focus on Cloud Computing characteristics then there might be different definitions but one of the wide recognized definition is:

“*The National Institute of Standards and Technology (NIST) defines cloud computing as it is known today through five particular characteristics.”*

And these 5 particular characteristics is

- *On-demand self-service. ...*
- *Broad network access. ...*
- *Multi-tenancy and resource pooling. ...*
- *Rapid elasticity and scalability. ...*
- *Measured service*

Hence, it is seen that to provide these services requires huge manpower and it’s really too costly. Where is the necessary to build like this cloud for personal use? This is because of lack of trust in other CSP. Some of issued come with the term of Lack of Trust in Cloud Computing was described previously. Here, in this thesis, it is included some informal issues that come with the trust in CSP. *Firstly, transmitting data through public channel creates more risk than using private channel (Virtual Private Network) for transmitting data. Secondly, if data storage maintaining operation is not enough secure sometime intruder can manage the access through SQLi operations or with some other malicious activity. Thirdly, users claim that a certified CSP analyses data in its storage of Certified users*.

This is the clear indication of security breaches of a client. If the client like bank or other organization who deal with very sensitive information like money-oriented information then the data must not disclose that data anyhow.

5.3 Motivation

Hence, it is taken 2 security risk factors. Firstly, through network channel. Secondly, On the Cloud.

First problem it can mitigate using Cryptography algorithm. In this model it is used for simulation AES, DES and Blowfish these three algorithms which is comparatively strong with encryption and decryption.

Second problem is the most serious problem that a CSP already has a Digital Certification hence it is need to trust that to give our private data. But if this CSP analyze this private data or let others use user's data for their own purposes! This is clearly a security breaches for a user. Recently, it is seen few examples of it such as Facebook recently has penalized by the F.T.C. with a record $5 billion fine for deceiving users about their ability to control the privacy of their personal data.

5.4 Proposed Methodology

In regular scenario it is shared key for both Client and Server while they can manage the access of data. But in this proposed scenario it is intend to share the Key only to the Client where the Server has only Ciphertext but no key it has to decipher the key.

In the client side the total encryption and decryption process are done. Thus, the cloud needs less computational power. And the data is not sent in channel as plaintext. Data and key are not sent in same channel. Thus, analyzing data key is hard to retrieve.

By using this method, we will measure performance of AES, DES and Blowfish algorithm using Java Socket Programming. Where it is considered a static key to simulate the whole process.

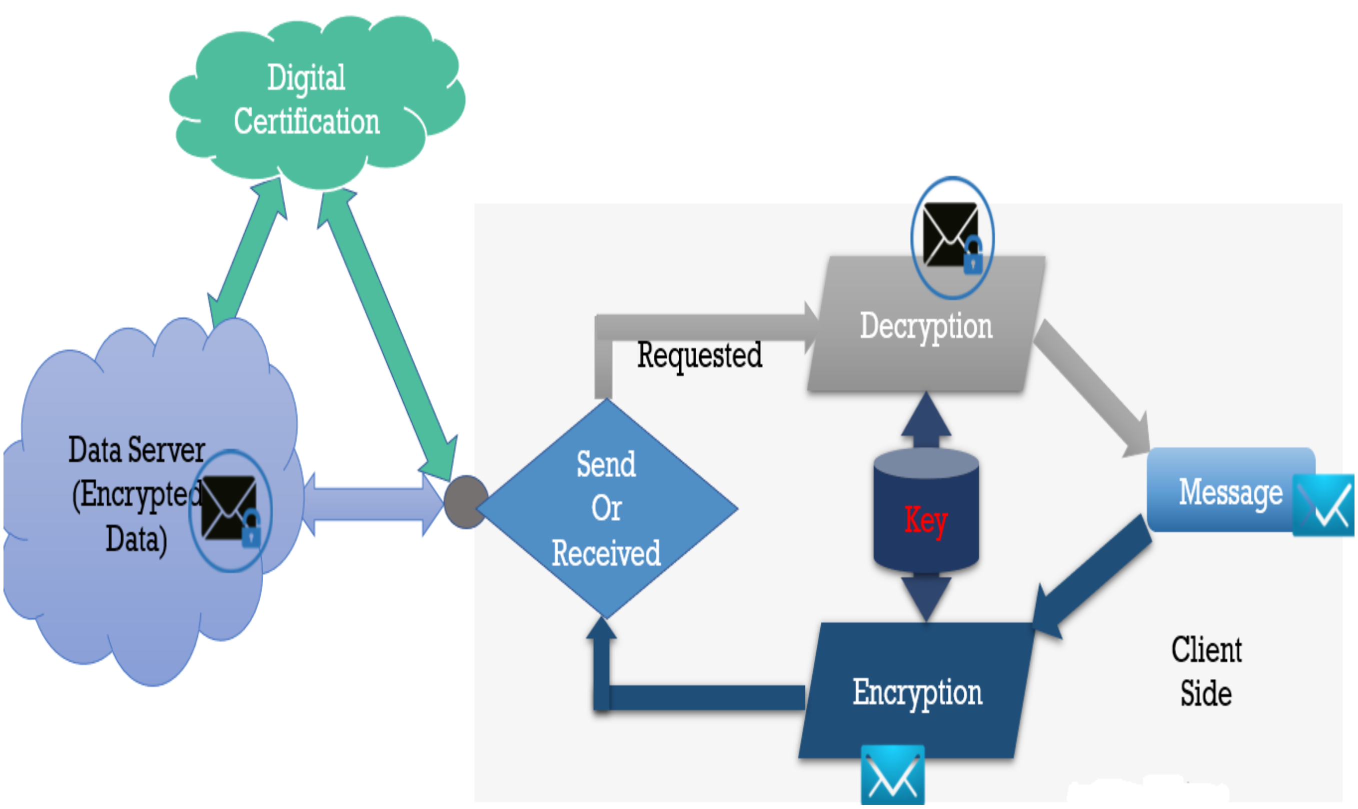


Figure 5.1: Proposed Client Server Model

5.5 Operations of Cryptography Algorithms

Different Cryptography algorithm performs in different manners. Operations of our applied model is described here.

5.5.1 DES

DES (Data Encryption Standard) is a symmetric cryptosystem developed by IBM in the early 1970's. It is based on the LUCIFER system developed earlier by IBM. DES was published in 1975 and was certified as an encryption standard for "unclassified" documents in USA in 1977. After this it has been used a lot in different circumstances, also as the triple system 3-DES.

Defining DES

DES operates with bit symbols, so the residue classes (bits) 0 and 1 of Z2 can be considered as the plaintext and ciphertext symbols. The length of the plaintext block is 64. The key k is 56 bits long. It is used in both encrypting and decrypting. In broad lines DES operates in the following way:

1. The bit sequence x0 is formed of the plaintext x by permutating the bits of x by a certain fixed permutation (the so-called *initial permutation)* π_{ini} then we write

$$X_0 = \pi_{ini}(X) = L_0R_0$$

Where L0 Contains the first 21 bits of X_0 and R_0 the rest.

2. Compute the sequence $L_1R_1, \; L_2R_2, \ldots\ldots L_{16}R_{16}$ by iterating the following procedure 16 times

$$\begin{cases} L_i = R_{i-1} \\ R_i = L_{i-1} \oplus f(R_{i-1}, k_i) \end{cases}$$

Where $\oplus$ is bitwise addition modulo2 (known also by the name XOR), *f* is a function which is given later, and k_i is the key of the i^{th} iteration, obtained from *k* by permuting 48 of its bits into a certain order. An iteration step is depicted on the right.

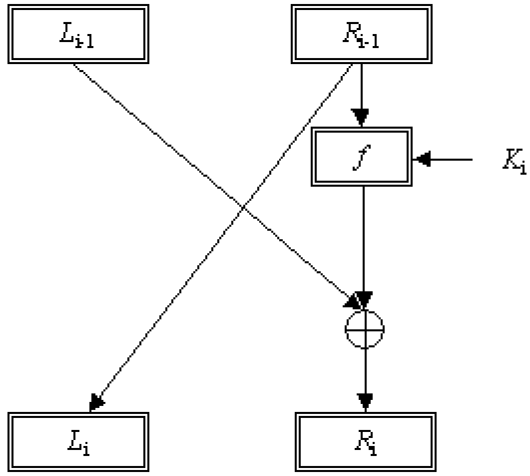


Figure 5.2: Single Round of DES.

Apply the inverse permutation π_{ini}^{-1} (the so-called final permutation) to the bit sequence $R_{16}L_{16}$.

We still need to give the permutation π_{ini}, define the function *f*, and give the key sequence $K1, K2, \ldots\ldots\ K16$ for encrypting to be defined.

First let's see the definition of the function *f*. The first argument R of *f* is a bit sequence of length 32 and the second argument *K* is a bit sequence of length 48. The procedure for computing *f* is the following:

1. The first argument *R is expanded using the expanding function E*. We take the first 32 bits of R into *E(R)*, duplicate half of them and then permute them. Bits are taken according to the table on the right, read from left to right and from top to bottom.

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

2. Compute $E(R) \oplus K = B$ and write the result as a catenation of eight 6-bit bit sequences:

$$B = B_1 B_2 B_3 B_4 B_5 B_6\ B_7 B_8.$$

3. Next it is used eight so-called S-boxes $S_1,...,S_8$. Each S_i is a fixed 4 × 16 table, formed of the numbers 0,1,...,15. When a bit sequence of length of 6

$$B_i = b_1\, b_2\, b_3\, b_4\, b_5\, b_6$$

is obtained, $Si(Bi) = Ci$ is computed in the following way. The bits $b1b6$ give the binary representation of the index r $(r = 0,1,2,3)$ of a certain row. The remaining bits $b_2b_3b_4b_5$ give the binary representation s $(s = 0,1,...,15)$ of a certain column. (The rows and columns of S_i are indexed starting from zero.) Now $S_i(B_i)$ is the binary representation of the number in the intersection of the r_{th} row and the s^{th} column of S_i, initial zeros added if needed to get four bits. The bit sequences C_i are catenated to the bit sequence,

$$C = C_1\, C_2\, C_3\, C_4\, C_5\, C_6\, C_7\, C_8$$

4. The bit sequence C of length of 32 is permuted using the fixed permutation π. The bit sequence $\pi(C)$ obtained this way is then $f(R,K)$.

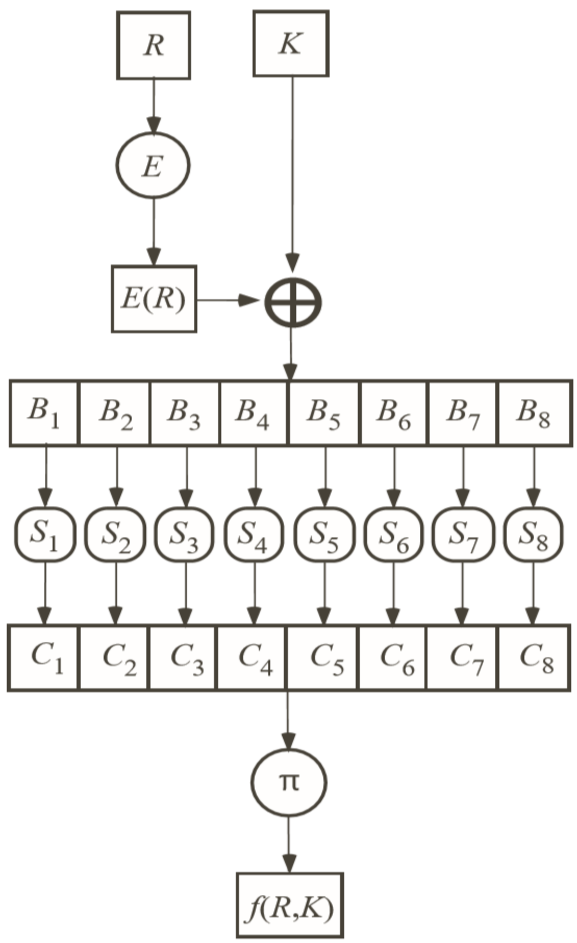


Figure 5.3: Procedure for computing *f*.

The operation is illustrated above. We may note that E and π are linear operations, in other words, they could be replaced by multiplication of a bit vector by a matrix. On the other hand, S-boxes are highly nonlinear.

15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

The definitions of S-boxes can be found in the literature (for example STINSON). On the right is *S2*, given as an example, and below the permutations π_{ini} and π *:*

π_{ini} :

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

π :

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

The key sequence *k1,k2,...,k16* can be computed iteratively in the following way:

1. The key k is given in an expanded form such that every eight bits is a parity-check bit. So, there is always an odd number of 1's in a byte and the length of the key is 64 bits. If the parity check shows that there are errors in the key, it will not be taken into use. Then again, if there are no errors in the key, the parity check bits are removed, and we come to original 56-bit key. First a fixed bit permutation πK1 is applied to the key. Write

$$\pi_{k1}(k) = C_0D_0$$

   where C_o and D_0 are bit sequences of length 28
2. Compute the sequence $C_1\,D_1, C_1\,D_1, \ldots, C_{16}D_{16}$ by iterating the following procedure 16 times:

$$\begin{cases} C_i = \sigma_i(C_{i-1}) \\ D_i = \sigma_i(D_{i-1}) \end{cases}$$

where σ_i is a cyclic shift of the bit sequence by 1 or 2 bits to the left. If $i = 1,2,9,16$ then the shift is 1 bit, otherwise it is 2 bits.

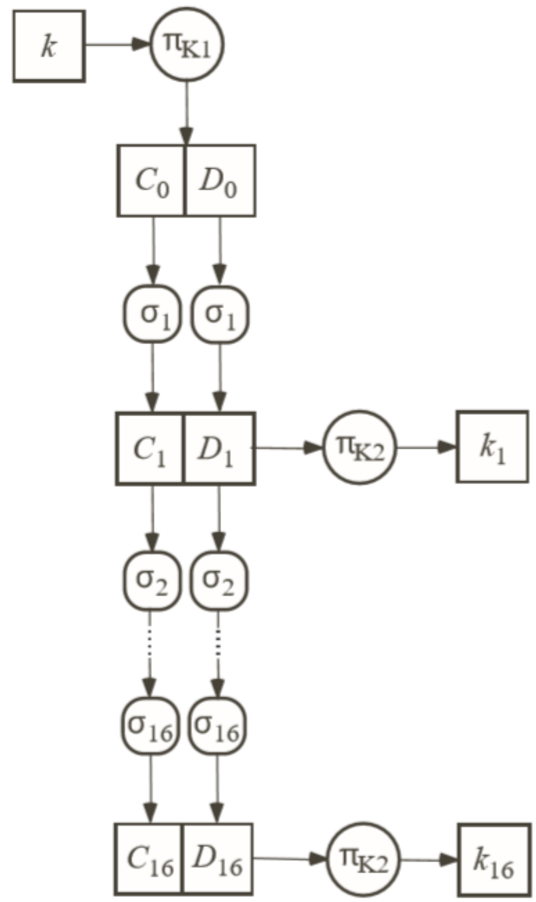
3. Apply the fixed variation π_{k2} of 48 bits to $C_i\, D_i$. In this way we obtain $k_i = \pi_{k2}(C_i\, D_i)$. We must still give the permutation π_{k1} and the variation π_{k2}:

π_{K1} :

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

π_{K2} :

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32



The key generating process is illustrated in the above figure. Decrypting goes essentially by same system but using the key sequence $k_i, k_i, \ldots, k_{16}$ in reverse order and inverting the permutations. Then

$$\begin{cases} L_{i-1} = R_i \oplus f(L_i, k_i) \\ R_{i-1} = L_i. \end{cases}$$

The modes of operation of DES are the same as for AES. Which is described letter.

5.5.2 AES

AES (Advanced Encryption Standard) is a fast-symmetric cryptosystem for mass encryption. It was developed through competition, and is based on the RIJNDAEL system, published in 1999 by Joan Daemen and Vincent Rijmen from Belgium.

AES works on bit symbols, so the residue classes (bits) 0 and 1 of Z_2 can be considered as plaintext and Ciphertext symbols. The workings of RIJNDAEL can be described using the field $F_2{}^8$ and its polynomial ring $F_2{}^8[z]$. To avoid confusion, we use *z* as the dummy variable in the polynomial ring and *x* as the dummy variable for polynomials in Z_2 needed in defining and representing the field $F_2{}^8$. Furthermore, we denote addition and multiplication in $F_2{}^8$ by $\oplus$ and $\odot$, the identity element is denoted by 1 and the zero element by 0. Note that because $1 = -1$ in Z_2, the additional inverse of an element in $Z_2[x]$, $F_2{}^8$ and in $F_2{}^8$ [z] is the element itself. So, subtraction $\ominus$ is the same as addition $\oplus$, in this case.

5.5.2.1 RIJNDAEL

In the RIJNDAEL system the length l_B of the plaintext block and the length l_k of the key are independently either 128, 192 or 256 bits. Dividing by 32 we get the numbers

$$Nk = \frac{lB}{32} \ and \ Nk \frac{lk}{32}$$

Bits are handled as bytes of 8 bits. An 8-bit byte $b_7 \, b_6 \cdots b_0$ can be considered as an element of the finite field $F_2{}^8$, which has the residue representation

$$b_0 + b_1 x + b_2 x^2 + b_3 \, x^3 + b_4 \, x^4 + b_5 \, x^5 + b_6 \, x^6 + b_7 \, x^7$$

The key is usually expressed as a $4 \times N_k$ matrix whose elements are bytes. If the key is, byte by byte,

$$\boldsymbol{K} = k_{00} \, k_{00} \, k_{10} \, k_{20} \, k_{30} \, k_{01} \, k_{11} \, k_{21} \, \ldots\ldots\ldots k_{3,Nk-1}$$

then the corresponding matrix is

$$\mathbf{K} = \begin{pmatrix} k_{00} & k_{01} & k_{02} & \cdots & k_{0,N_\mathrm{K}-1} \\ k_{10} & k_{11} & k_{12} & \cdots & k_{1,N_\mathrm{K}-1} \\ k_{20} & k_{21} & k_{22} & \cdots & k_{2,N_\mathrm{K}-1} \\ k_{30} & k_{31} & k_{32} & \cdots & k_{3,N_\mathrm{K}-1} \end{pmatrix}$$

Note how the elements of the matrix are indexed starting from zero. Similarly, if the input block (plaintext block) is, byte by byte,

$$\mathbf{a} = a_{00}a_{10}a_{20}a_{30}a_{01}a_{11}a_{21}\cdots a_{3,N_\mathrm{B}-1}$$

then the corresponding matrix is

$$\mathbf{A} = \begin{pmatrix} a_{00} & a_{01} & a_{02} & \cdots & a_{0,N_\mathrm{B}-1} \\ a_{10} & a_{11} & a_{12} & \cdots & a_{1,N_\mathrm{B}-1} \\ a_{20} & a_{21} & a_{22} & \cdots & a_{2,N_\mathrm{B}-1} \\ a_{30} & a_{31} & a_{32} & \cdots & a_{3,N_\mathrm{B}-1} \end{pmatrix}.$$

During encryption we are dealing with a bit sequence of length l_B, the so-called state. Like the block, it is also expressed byte by byte in the form of a $4 \times N_B$ matrix:

$$\mathbf{S} = \begin{pmatrix} s_{00} & s_{01} & s_{02} & \cdots & s_{0,N_\mathrm{B}-1} \\ s_{10} & s_{11} & s_{12} & \cdots & s_{1,N_\mathrm{B}-1} \\ s_{20} & s_{21} & s_{22} & \cdots & s_{2,N_\mathrm{B}-1} \\ s_{30} & s_{31} & s_{32} & \cdots & s_{3,N_\mathrm{B}-1} \end{pmatrix}.$$

Elements of the matrices K, A and S are bytes of 8 bits, which can be interpreted as elements of the field F_2^8. In this way these matrices are matrices over this field. Another way to interpret the matrices is to consider their columns as sequences of elements of the field $F_2^8$of length 4. These can be interpreted further, from top to bottom, as coefficients of polynomials with maximum degree 3 from the polynomial ring $F_2^8\ [z]$. So, the state S mentioned above would thus correspond to the polynomial sequence

$$S_{00} \oplus S_{10}\, Z \oplus S_{20}\, Z^2 \oplus S_{30}\, Z^2\ ,\ S_{01} \oplus S_{11}\, Z \oplus S_{21}\, Z^2 \oplus S_{31}\, Z^3\ ,\ \ldots\ ,$$
$$S_{0},N_{B-1} \oplus S_{1},N_{B-1}\, Z \oplus S_{2},\ N_{B-1}\, Z^2 \oplus S_{3},\ N_{B-1}Z^3.$$

For the representation to be unique, a given fixed irreducible polynomial of degree 8 from $Z_2[x]$ mustbeusedintheconstructionofF_2^8. In RIJNDAEL it is the so-called RIJNDAEL polynomial,

$$p(x) = 1 + x + x^3 + x^4 + x^8$$

5.5.2.2 Rounds

There is a certain number N_R of so-called rounds in RIJNDAEL. The number of rounds is given by the following table:

N_R	N_B **= 4**	N_B **= 6**	N_B **= 8**
N_K**= 4**	10	12	14
N_K**= 6**	12	12	14
N_K**= 8**	14	14	14

The i^th^ round receives as its input the current state S and its own so-called round key R_i. In particular, we need the initial round key R_0. In each round, except for the last one, we go through the following sequence of operations:

$$S \leftarrow SubBytes(S)$$
$$S \leftarrow ShiftRows(S)$$
$$S \leftarrow MixColumns(S)$$
$$S \leftarrow AddRoundKey(S,Ri)$$

The last round is the same except that we drop MixColumns. The encrypting key is expanded first and then used to distribute round keys to all rounds. This and the different operations in rounds are discussed one by one in the following sections. Encrypting itself then consists of the following steps:

- Initialize the state: S ← AddRoundKey(A,R_0).
- N_{R-1} ”usual” rounds.
- The last round.

When decrypting we go through the inverse steps in reverse order

5.5.2.3 Transforming Bytes (SubBytes)

In this operation each byte S_{ij} of the state is transformed in the following way:

I. Interpret S_{ij} as an element of the field F28 and compute its inverse S^{-1}_{ij} . It is agreed here that the inverse of the zero element is the element itself.

II. Expanding S^{-1}_{ij} in eight bits $b_7\, b_6\, b_5\, b_4\, b_3\, b_2\, b_1\, b_0$ denote

$b(x) = b_0 + b_1x + b_2x^2 + b_3x^3 + b_4x^4 + b_5x^5 + b_6x^6 + b_7x^7$ (a polynomial in $Z_2[x]$) and compute

$$b'(x) \equiv b(x)(1 + x + x^2 + x^3 + x^4) + (1 + x + x^5 + x^6) \bmod 1+x^8$$

The result

$$b'(x) = b'_0 + b'_1x + b'_2x^2 + b'_3x^3 + b'_4x^4 + b'_5x^5 + b'_6x^6 + b'_7x^7$$

is interpreted as a byte $b'_0b'_1b'_2b'_3b'_4b'_5b'_6b'_7$ or as an element of F_2^8 . By the way, division by $1 + X^8$ in $Z_2[x]$ is easy since

$$x^k \equiv x^{(k \bmod s)} \bmod 1+x^8.$$

The operation may also be done by using matrices. We then apply an affine transformation in Z_2

$$\begin{pmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

Byte transformation is done in reverse order during the decryption. Because in $Z_2[x]$

$$1 = gcd(1 + x + x^2 + x^3 + x^4, 1 + x^8)$$

(easy to verify using the Euclidean algorithm), the polynomial $1 + x + x^2 + x^3 + x^4$ has an inverse modulo $1 + x^8$ and the occurring 8×8 matrix is invertible modulo 2. This inverse is $x + x^3 + x^6$.

5.5.2.4 Shifting Rows (ShiftRows)

In this operation the elements of the rows of the matrix representation of the state are shifted left cyclically in the following way

Shift	Row0	Row 1	Row 2	Row 3
N_B =4	No Shift	1 element	2 elements	3 elements
N_B =6	No Shift	1 element	2 elements	3 elements
N_B =8	No Shift	1 element	3 elements	4 elements

While decrypting rows are correspondingly shifted right cyclically.

5.5.2.4 Mixing Columns (MixColumns)

In this transformation columns of the state matrix are interpreted as polynomials of maximum degree 3 in the polynomial ring F_2^8 [z]. Each column (polynomial) is multiplied by the fixed polynomial

$$c(z) = c_0 \oplus c_1 z \oplus c_2 z^2 \oplus c_3 z^3 \in F_2^8\,[z]$$

modulo $1 \oplus z^{-1}$ where

$$C_0 = x, \qquad C_1 = C_2 = 1 \quad and \quad C_3 = 1+x$$

Dividing by the polynomial $1 \oplus z^4$ in F_2^8 [z] is especially easy since

$$Z^k \equiv Z^{(k \bmod 4)} \bmod 1 \oplus Z^{4.}$$

Alternatively the operation can be considered as a linear transformation of F_2^8 ;

$$\begin{pmatrix} s'_{0i} \\ s'_{1i} \\ s'_{2i} \\ s'_{3i} \end{pmatrix} = \begin{pmatrix} c_0 & c_3 & c_2 & c_1 \\ c_1 & c_0 & c_3 & c_2 \\ c_2 & c_1 & c_0 & c_3 \\ c_3 & c_2 & c_1 & c_0 \end{pmatrix} \begin{pmatrix} s_{0i} \\ s_{1i} \\ s_{2i} \\ s_{3i} \end{pmatrix}.$$

When decrypting we divide by the polynomial c(z) modulo $1 \oplus z^4$. Although $1 \oplus z^4$ is not an irreducible polynomial of F_2^8 *[z]1, c(z)* has an inverse modulo $1 \oplus z^4$, because

$$1 = gcd(c(z), 1 \oplus z^4)$$

The inverse is obtained using the Euclidean algorithm (hard to compute!) and it is

$$d(z) = d_0 \oplus d_1 z \oplus d_2 z^2 \oplus d_3 z^3$$

where

$$d_0 = x + x^2 + x^3\,, d_1 = 1 + x^3\,, d_3 = 1 + x^2 + x^3 \text{ and } d_3 = 1 + x + x^3.$$

So, when decrypting the column (polynomial) is multiplied by d(z) modulo 1 ⊕ z4 and the operation is thus no more complicated than when encrypting. In matrix form in F_2^8.

$$\begin{pmatrix} s_{0i} \\ s_{1i} \\ s_{2i} \\ s_{3i} \end{pmatrix} = \begin{pmatrix} d_0 & d_3 & d_2 & d_1 \\ d_1 & d_0 & d_3 & d_2 \\ d_2 & d_1 & d_0 & d_3 \\ d_3 & d_2 & d_1 & d_0 \end{pmatrix} \begin{pmatrix} s'_{0i} \\ s'_{1i} \\ s'_{2i} \\ s'_{3i} \end{pmatrix}.$$

5.5.2.5 Adding Round Keys (AddRoundKey)

The round key is as long as the state. In this operation the round key is added to the state byte by byte modulo 2. The inverse operation is the same.

5.5.2.6 Expanding the Key

The round keys $R_0, R_1, \ldots, R_{NR}$ are obtained from the encrypting key by expanding it and then choosing from the expanded key certain parts for different rounds. The length of the expanded key in bits is $l_B(N_R + 1)$. Divided into bytes it can be expressed as a $4 \times N_B(N_R+ 1)$ matrix, which has $N_B(N_R + 1)$ columns of length 4:

$$w_0, w_1, \ldots, w_{NB}(N_R+1) - 1.$$

Denote the columns of the key (matrix K) correspondingly:

$k_0, k_1, \ldots, k_{NK-1}$

The expanded key is computed using the following method:

1. Set $w_i \leftarrow k_i \ (i = 0, \ldots, N_K - 1)$.
2. Define the remaining w_i 's recursively by the following rules where addition of vectors in F_2^8 is done elementwise in the usual fashion:

   2.1 If $i \equiv 0 \bmod NK$ then compute $u = x^{i/NK}$ in the field F_2^8 and set

   $$w_i \leftarrow w_i - N_K \oplus SubByte\ (RotByte(w_{i-1})) \oplus \begin{pmatrix} U \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

   Here the operation SubByte means transforming every element (byte) of the column. Operation RotByte does a cyclic shift of one element up in a column.

   2.2 If $N_B = 8$ and $i \equiv 4 \bmod N_K$, set

   $$w_i \leftarrow w_{i-NK} \oplus SubByte\ (w_{i-1})$$

   where the operation SubByte is the same as in #2.1

   2.3 Otherwise simply set

   $$w_i \leftarrow w_{i-NK} \oplus w_{i-1}.$$

Now the round key Ri of the i_{th} round is obtained from the columns $w_iN_B, \ldots, w(i+1)N_{B-1}$ (i = 0,1, ...,N_R). In particular, from the first N_B columns we get the initial round key R_0.

NB*: Expansion of the key can be made in advance, as long as the encrypting key is known. Anyway, the $x^{i/NK}$'s can be computed beforehand in the field F_28 .*

5.5.2.7 A Variant of Decryption

A straight forward procedure for decrypting follows the following chain of operations—they are the inverse operations of the encrypting operations that were introduced before

$S \leftarrow \text{AddRoundKey}(S,R_{NR})$

$S \leftarrow \text{ShiftRows}^{-1}(S)$

$S \leftarrow \text{SubBytes}^{-1}(S)$

$S \leftarrow \text{AddRoundKey}(S,R_{NR-1})$

$S \leftarrow \text{MixColumns}^{-1}(S)$

$S \leftarrow \text{ShiftRows}^{-1}(S)$

$S \leftarrow \text{SubBytes}^{-1}(S)$

…..

$S \leftarrow \text{AddRoundKey}(S,R_1)$

$S \leftarrow \text{MixColumns}^{-1}(S)$

$S \leftarrow \text{ShiftRows}^{-1}(S$

$S \leftarrow \text{SubBytes}^{-1}(S)$

$S \leftarrow \text{AddRoundKey}(S,R_0)$ | can be replaced by the operations

$S \leftarrow \text{MixColumns}^{-1}(S)$

$S \leftarrow \text{AddRoundKey}(S,\text{MixColumns}^{-1}(R_i))$

$S \leftarrow \text{AddRoundKey}(S,R_{NR})$

$S \leftarrow \text{SubBytes}^{-1}(S)$

$S \leftarrow \text{ShiftRows}^{-1}(S)$

$S \leftarrow \text{MixColumns}^{-1}(S)$

$S \leftarrow \text{AddRoundKey}(S,\text{MixColumns}^{-1}(R_{NR-1}))$

$S \leftarrow \text{SubBytes}^{-1}(S)$

$S \leftarrow \text{ShiftRows}^{-1}(S)$

$S \leftarrow \text{MixColumns}^{-1}(S)$

$S \leftarrow AddRoundKey(S, MixColumns^{-1}(R_{NR-2}))$

. . .

$S \leftarrow SubBytes^{-1}(S)$

$S \leftarrow ShiftRows^{-1}(S)$

$S \leftarrow MixColumns^{-1}(S)$

$S \leftarrow AddRoundKey(S, MixColumns^{-1}(R_1))$

$S \leftarrow SubBytes^{-1}(S)$

$S \leftarrow ShiftRows^{-1}(S)$

$S \leftarrow AddRoundKey(S, R_0)$

which reminds us very much of the encrypting process. Hence RIJNDAEL encrypting and decrypting are very similar operations.

5.5.3 Blowfish

Blowfish symmetric block cipher algorithm encrypts block data of 64-bits at a time. The algorithm follows festal network and is divided into 2 main parts:

1. Key-expansion
2. Data Encryption
3. Data Decryption

5.5.3.1 Key Expansion

Prior to any data encryption and decryption, these keys should be computed before-hand.

The p-array consists of 18, 32-bit sub-keys:

P1, P2,., P18

Four 32-bit S-Boxes consist of 256 entries each:

S1, 0, S1, 1,. S1, 255

S2, 0, S2, 1,.. S2, 255

S3, 0, S3, 1,.. S3, 255

S4, 0, S4, 1 ....S4, 255

Generating the Sub-keys: The sub-keys are calculated and generated using the Blowfish algorithm:

1. Initialize first the P-array and then the four S-boxes, in order, with a fixed string. This string consists of the hexadecimal digits of pi (less the initial 3): *P1 = 0x243f6a88, P2 = 0x85a308d3, P3 = 0x13198a2e, P4 = 0x03707344,* etc.
2. . XOR P1 with the first 32 bits of the key, XOR P2 with the second 32-bits of the key, and so on for all bits of the key (possibly up to P14). Repeatedly cycle through the key bits until the entire P-array has been XORed with key bits. (For every short key, there is at least one equivalent longer key; for example, if A is a 64-bit key, then AA, AAA, etc., are equivalent keys.)

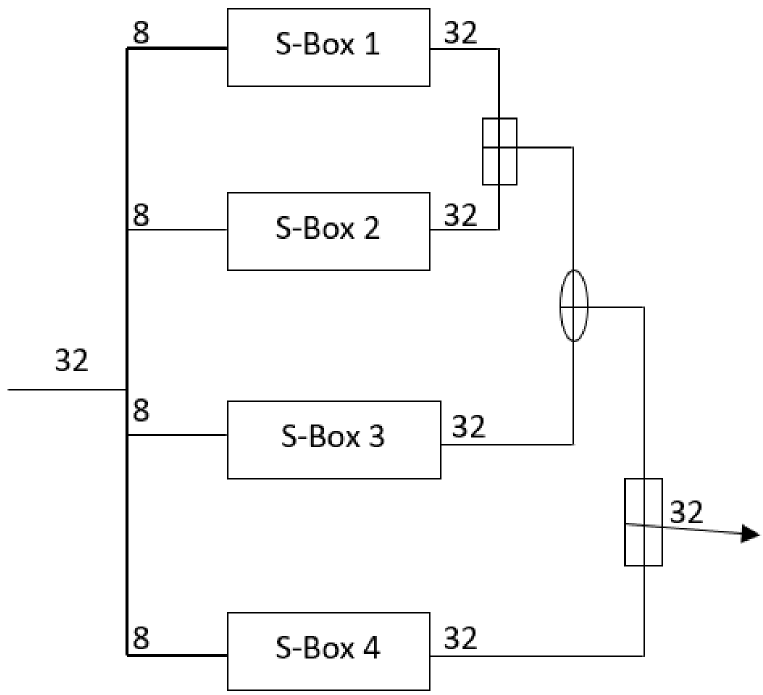


Figure 5.4: S-Box

3. Encrypt the all-zero string with the Blowfish algorithm, using the subkeys described in steps (1) and (2).
4. Replace P1 and P2 with the output of step (3).
5. Encrypt the output of step (3) using the Blowfish algorithm with the modified sub-keys.
6. Replace P3 and P4 with the output of step (5).
7. Continue the process, replacing all entries of the P array, and then all four S-boxes in order, with the output of the continuously changing Blowfish algorithm.

In total, **521** iterations are required to generate all required sub-keys. Applications can store the sub-keys rather than execute this derivation process multiple times.

5.5.3.2 Data Encryption

It is having a function to iterate 16 times of network. Each round consists of key-dependent permutation and a key and data-dependent substitution. All operations are XORs and additions on 32-bit words. The only additional operations are four indexed array data lookup tables for each round.

Algorithm: Blowfish Encryption

Divide x into two 32-bit halves: X_L, X_R

For i = 1 to 16:

$X_L = X_L$ *XOR* P_i

$X_R = F(X_L)$ *XOR*

X_R *Swap* X_L *and*

X_R *Swap* X_L *and*

X_R *(Undo the last swap.)*

$X_R = X_R$ *XOR* P_{17}

$X_L = X_L$ *XOR* P_{18}

Recombine X_L *and* X_R

5.5.3.3 Data Decryption

Decryption is exactly the same as encryption, except that *P1, P2* ..... *P18* are used in the reverse order.

5.6 Conclusions

Data encryption is the method of using algorithmic schemes and mathematical calculations to transform plain text into ciphered text, thus making it non-readable and unusable for unauthorized parties. To decrypt an encrypted message the recipient must use a special key that triggers the mechanism and transforms text back to the original version. In this chapter it is shown that the detailed view of working and functionalities of Cryptography Algorithm (Advance Encryption Standard, Data Encryption Standard, Blowfish). On the next chapter using these methodologies implementation process will be done.

Chapter 6

Result Analysis

Chapter 6
Implementation & Result Analysis

6.1 Introduction

In this chapter it is described comparative result depending on the model and the theory of Cryptography Algorithm (DES, AES, Blowfish) described previously on chapter 5. It has been simulated the result on a single machine and the result can vary depending on machine configuration and its computing capability. For this it will also include some theoretical result in practical purposes. The whole architecture has been simulated using Java Socket programming and we create a prototype for this model a *Data Server (which stores and replies with data) and a Client (which is responsible for creating CipherText and send that to Data Server to store and can request for the CipherText to Decipher into Plaintext)*. And we build our Prototype of this model only for textual data. No other format of data was not considered in the protype.

NB: All Simulation done in a machine which is configures as

- *Processor: Intel® Core™ i5-4210 @ 1.70GHz,2401Mhz, 4 Logical processors*
- *Graphics Memory: 2GB (NVDIA® GEFORCE 840M)*
- *RAM 8GB (DDR3)*
- *SSD 250 GB*
- *OS: Microsoft® Windows 10*

6.2 Implementation

To implement we use java socket programming and creat a Socket between Client and Server. Where in the client end there is Secret key Generator which generates key and share it only to the client thus the client can convert his plaintext into Ciphertext to send it to the Server. As a result both the network channel and server does not have the key to Decrypt the Ciphertext, so cannot analyze the data.

While the data is requested from the Server then the Server replies with the Ciphertext and send it through the transmission media and Client receives it and decrypt it using the key provided by the Secret Key Generator. This process is applying on AES, DES & Blowfish to analyze the performance of the model and a comparative result from this model will be described. The implementation procedure is illustrated in bellow the figure. Where *Set Socket* for establishing a unique port to *Bind* with client and an acknowledgement is done for handshaking among them through *Connect – Listen* port.

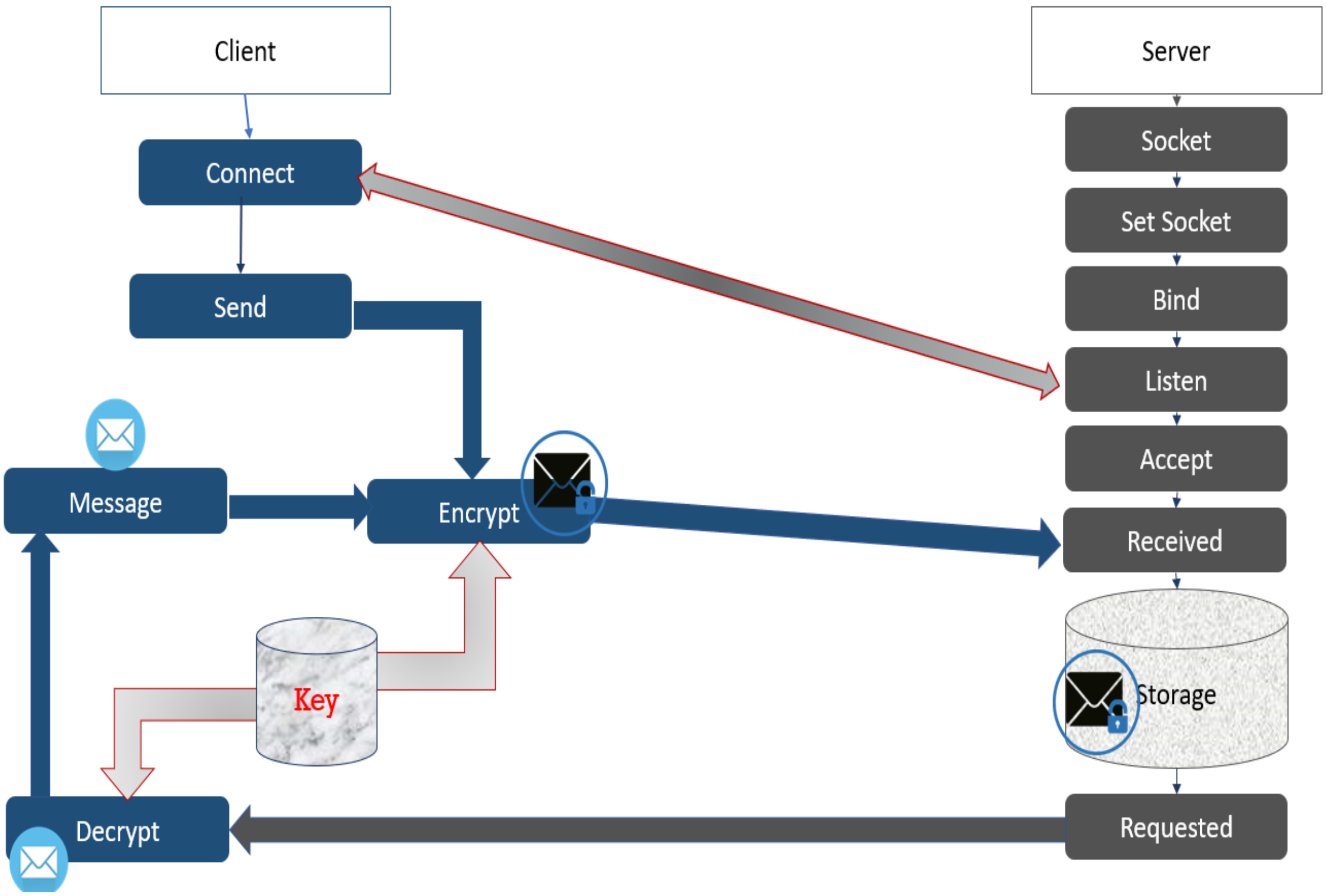


Figure 6.1: Implementation Procedure.

6.3 Result Section

Here it is concluded the outcome of this thesis. Where it is taken 3 input for a single input. Hence, we get several outputs. Due to different *task scheduling* operation is performed by *Operating System* we get different outcome. So, it can be decided to take 3 output and make an average to reduce the error of outcome.

Table 6.1: Result For Data Encryption Standard

Plaintext Size (Bytes)	Ciphertext Size (Bytes)	Encryption		Decryption Time (ns)	Total Time (ns)	Average Time (ns)
		Key Generation(ns)	Encryption Time (ns)			
256	**352**	3690558400	15155700	501600	**3706215700**	**3872334933**
		4936237600	10872500	676800	**3947786900**	
		3948470100	13898600	633501	**3963002201**	
512	**969**	3886509500	12879100	1019300	**3900407900**	**4027990933**
		4057719000	14025801	706800	**4072451601**	
		4093883499	16304301	925500	**4111113300**	
1024	**1376**	3916624100	13059000	653900	**3930337000**	**4173592500**
		4087272601	12149399	558100	**4099980100**	
		4474238600	13791600	2430200	**4490460400**	
2048	**2744**	3945484000	15329200	1337300	**3962150500**	**4167283667**
		4213196201	14617800	1320500	**4229134501**	
		4291407000	17934700	1224300	**4310566000**	
4096	**5472**	3932912900	14375600	996600	**3948285100**	**4211325900**
		4276480701	15794201	1379200	**4293654102**	
		4373245199	17816301	977000	**4392038500**	
8192	**8192**	3937700400	16472600	1716400	**3955889400**	**4110696866**
		4071666300	15557700	1665400	**4088889400**	
		4265652200	19634100	2025500	**4287311800**	

Table 6.2: Result For Advance Encryption Standard

Plaintext Size (Bytes)	Ciphertext Size (Bytes)	Encryption		Decryption Time (ns)	Total Time (ns)	Average Time (ns)
		Key Generation(ns)	Encryption Time (ns)			
256	**364**	3860815700	9973300	381600	**3871170600**	**4096904833**
		4150249700	10857500	258200	**4161365400**	
		4242231100	15627800	319600	**4258178500**	
512	**706**	3955597500	11485700	380100	**3967463300**	**4033160866**
		4096473800	11239300	381900	**4108095000**	
		4012975699	10563900	384700	**4023924299**	
1024	**1388**	3916491100	9965600	690600	**3927147300**	**4005463633**
		4006532300	11571301	704900	**4018808501**	
		4059067700	10865399	502000	**4070435099**	
2048	**2752**	4212977400	13372200	1260100	**4227609700**	**4166011200**
		4071459200	10790200	1280100	**4083085900**	
		4165978801	17265499	1270100	**4187338000**	
4096	**5484**	4350758800	13477400	1578200	**4365814400**	**4205926133**
		4041095700	12181200	2145999	**4055422899**	
		4183547900	11639200	1354000	**4196541100**	
8192	**10944**	4354972900	16201000	2485700	**4373659600**	**4183158732**
		4130447599	10155200	2447199	**4143049998**	
		4016329700	13969700	2467200	**4032766600**	

Table 6.3: Result for Blowfish

Plaintext Size (Bytes)	Ciphertext Size (Bytes)	Encryption		Decryption Time (ns)	Total Time (ns)	Average Time (ns)
		Key Generation(ns)	Encryption Time (ns)			
256	**352**	3959998300	14201300	537800	**3974737400**	**4380443200**
		4052308500	14500301	793800	**4067602601**	
		5081350601	17206200	432800	**5098989601**	
512	**969**	4315550300	13663900	495500	**4329709700**	**4144691000**
		4017261000	14712500	516600	**4032490100**	
		4056587000	14788701	497499	**4071873200**	
1024	**1376**	3960104300	13997700	510700	**3974612700**	**4041796100**
		4045064099	14906100	679501	**4060649700**	
		4072194700	17319600	611600	**4090125900**	
2048	**2744**	4251588100	14383200	904900	**4266876200**	**4138455933**
		4006724300	20546199	723700	**4027994199**	
		4103981600	15598900	916900	**4120497400**	
4096	**5472**	3860427500	15072700	966500	**3876466700**	**4048983066**
		4220940500	14469500	878499	**4236288499**	
		4017897799	15089200	1207000	**4034193999**	
8192	**8192**	3913238200	15330000	1683100	**3930251300**	**4006725300**
		4006344001	17177700	1492300	**4025014001**	
		4042899300	20546100	1465199	**4064910599**	

6.3 Comparison Among Algorithms

Table 6.4: Memory allocation

Plaintext Size (Bytes)	DES	Blowfish	AES
	Ciphertext Size (Bytes)	Ciphertext Size (Bytes)	Ciphertext Size (Bytes)
256	352	352	**364**
512	969	969	**706**
1024	1376	1376	**1388**
2048	2744	2744	**2752**
4096	5472	5472	**5484**
8192	8192	8192	**10944**

Table 6.5: Average Key Generation Time

Plaintext Size (Bytes)	DES	Blowfish	AES
	Key Generation(ns)	Key Generation(ns)	Key Generation(ns)
256	4191755366	3959998300	4084432166
512	4012703999	4315550300	4021682333
1024	4159378433	3960104300	3994030366
2048	4150029067	4251588100	4150138467
4096	4194212933	3860427500	4191800800
8192	4091672966	3913238200	4167250066

Table 6.6 Average Encryption Time

Plaintext Size (Bytes)	DES	Blowfish	AES
	Encryption Time (ns)	Encryption Time (ns)	Encryption Time (ns)
256	13308933	14201300	12152866
512	14403067	14388367	11096300
1024	12999999	15407800	10800766
2048	15960566	16842766	13809299
4096	15995367	14877133	12432600
8192	17221466	17684600	13441966

Table 6.7: Average Decryption Time

Plaintext Size (Bytes)	DES	Blowfish	AES
	Decryption Time (ns)	Decryption Time (ns)	Decryption Time (ns)
256	603967	588133	319800
512	883866	503199	382233
1024	1214066	600600	632500
2048	1294033	848500	1270100
4096	1117600	1017333	1692733
8192	1802433	1546866	2466699

Table 6.8: Average Total time in Cryptography process

Plaintext Size (Bytes)	DES	Blowfish	AES
	Total Time (ns)	Total Time (ns)	Total Time (ns)
256	**3872334933**	**4380443200**	4096904833
512	**4027990933**	**4144691000**	**4033160866**
1024	**4173592500**	**4041796100**	**4005463633**
2048	**4167283667**	**4138455933**	**4166011200**
4096	**4211325900**	**4048983066**	**4205926133**
8192	**4110696866**	**4006725300**	**4183158732**

6.4 Conclusion and Outcome

1. **Memory Allocation:** In term of memory allocation AES got lowest efficiency where DES and Blowfish has equal efficiency.
2. **Key generation** depend on random number generation function, for same size key there is no noticeable difference.
3. **Encryption Time:** All 3 algorithm were incredibly fast and has real life application while in term of encryption speed AES faster than Blowfish and Blowfish was faster than DES.
4. **Decryption Time:** In term of Decryption Time It was faster than encryption process, which is more weighted factor in real world information retrieval. Where AES incredibly was faster in terms of lower data size.

Chapter 7

Limitations and Future Work

Chapter 7
Limitations and Future Work

7.1 Introduction

Previously it was described and compare the outcome among of the Cryptography Algorithms (AES, DES, Blowfish). Hence this proposed model was to secure the key without sharing it to CSP.

7.2 Limitations

If it was discussed about the limitations of this model. Then it must come with the Key Distribution Authority. If the Key Distribution Authority shares the key to the CSP then our model has a serious failure. But booth the CSP and Key Distribution might be separate authority with their own business aspects and reputation. It is hard to compromise themselves with the key.

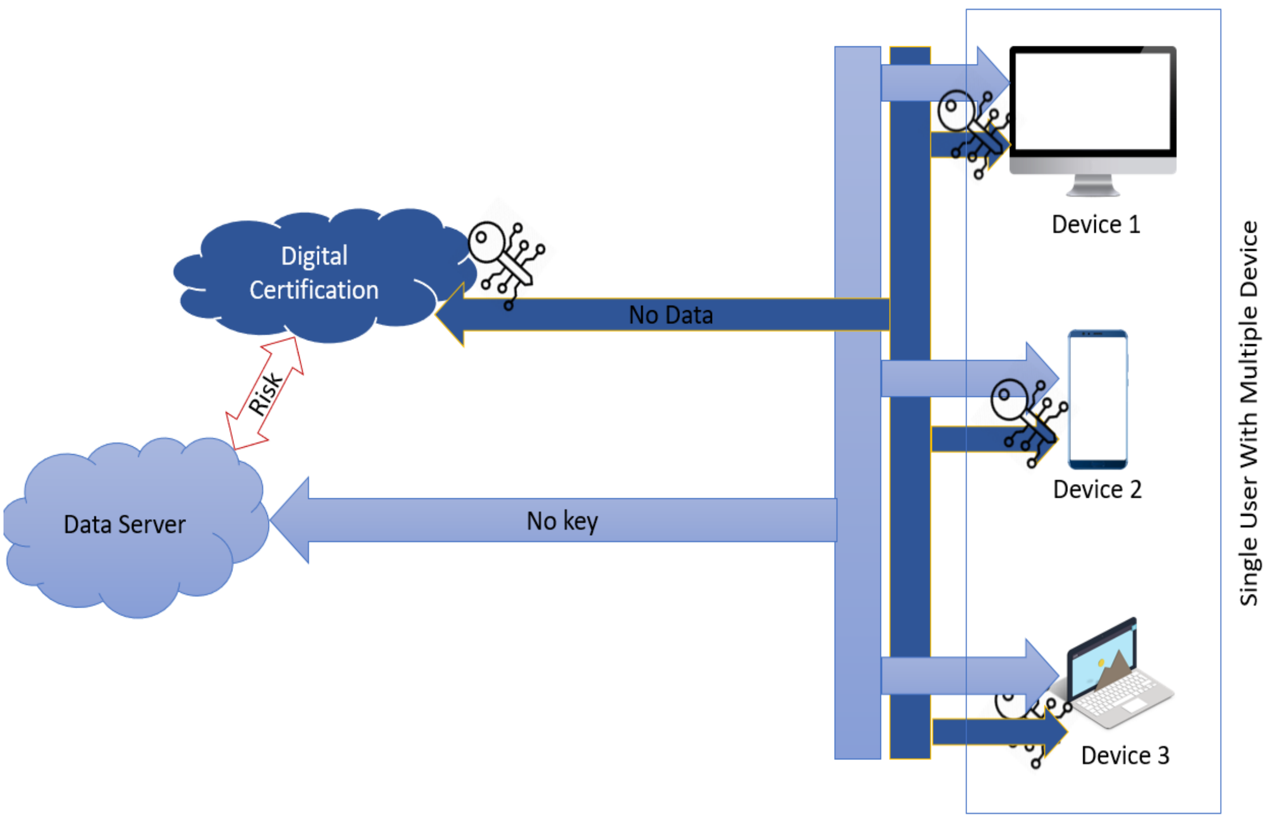


Figure 7.1: Limitations of Proposed Model

7.3 Future Work

Future work for this model is to find out some architecture to eliminate the limitations described earlier. And simulate for more Cryptography for real life implementations regarding computing power of handheld and other devices.

7. 4 Conclusions

Cloud is a modern technology. It makes easy to store data and give an excellent support to the users. Few years ago, people had to think about space for keeping their data. After emerging of computer, the need of space was managed a bit. But modern world needs more data. Cloud overcomes this limitation of storing data. At the same time, the security issue become the headache. It is needed to secure data now. Security is the main concern now-a-days. The algorithms being modified. And many are broken daily. While the major security issues come with the key management. Here is a model for securing data in cloud computing using this proposed model.

.

REFERENCES

[1] Narayan D.G, Meena S.M (2016), “A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish” on Elsevier Volume 78, Pages 617-624.

[2] Sanka S, Hota C, Rajarajan M (2010) ,”Secure data access in cloud computing”, In: IEEE 4th international conference on Internet multimedia services architecture and application (IMSAA), pp 1–6.

[3] Santosh Kumar Singh, Dr. P.K. Manjhi, Dr. R.K.Tiwari, ” An Approach towards Data Security in the Cloud Computing Using AES”,in June 2016 International Journal of Advanced Research in Computer and Communication Engineering , Vol. 5, Issue 6, pp. 22-29.

[4] Anjula Gupta, Navpreet Kaur Wailia (2014),” Cryptography Algorithm: A Review ” ,in IJEDR Volume 2 issue 2 page1670-1671.

[5] Narayan D.G, Meena S.M (2016), “A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish” on Elsevier Volume 78, Pages 617-624.

[6] Idrizi, Florim, Dalipi, Fisnik & Rustemi, Ejup. “Analyzing the speed of combined cryptographic algorithms with secret and public key”. International Journ a l of Engineering Research and Development , e- ISSN : 2278 - 067X, p - ISS N : 2278 - 800X, www.ijerd.com Volume 8, Issue 2 August 2013) , pp. 45.

[7] Abdul.Mina, D.S, Kader, H.M. Abdual & Hadhoud, M.M. “Performance Analysis of Symmetric Cryptography”. pp. 1.

[8] Edureka, What is Cryptography?- an introduction to Cryptography Algorithm, Edureka!, updated May 22,2019, https://d1jnx9ba8s6j9r.cloudfront.net/blog/wp-content/uploads/2018/07/encryption-algorithms-what-is-cryptography-edureka-768x352.png.

[9] Guru99 , Cloud Computing For Beginers, Retrieve From https://www.guru99.com/cloud-computing-for-beginners.html

[10] Edureka, What is Cryptography?- an introduction to Cryptography Algorithm, Edureka!, updated May 22,2019, https://www.edureka.co/blog/what-is-cryptography/

[11] Gartner: Seven cloud-computing security risks. InfoWorld. 2008-07-02. http://www.infoworld.com/d/security-central/gartner-seven-cloudcomputing-security-risks-853.

[12] Cloud computing security, http://en.wikipedia.org/wiki/Cloud_computing_security.

[13] "OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data,"http://www.oecd.org/document/18/0,3343,en_2649_34255_1815186_1_1_1_1,00.html.

[14] Cloud Security Alliance, Security Guidance for Critical Areas of Focus in Cloud Computing, V2.1, http://www.cloudsecurityalliance.org/ guidance/csaguide.v2.1.pdf.

[15] D. Chen and H. Zhao, "Data Security and Privacy Protection Issues in Cloud Computing," in 2012 International Conference on Computer Science and Electronics Engineering (ICCSEE), 2012, vol. 1, pp. 647– 651.

[16] Santosh Kumar Singh, Dr. P.K. Manjhi, Dr. R.K.Tiwari, " An Approach towards Data Security in the Cloud Computing Using AES",in June 2016 International Journal of Advanced Research in Computer and Communication Engineering , Vol. 5, Issue 6, pp. 22-29.