**DATA SECURITY AND PRIVACY FOR**

**END TO END MESSAGE COMMUNICATION**

**A synopsis submitted in partial fulfillment of the requirements for the course of**

**Minor project - I**

**In third year – Fifth semester of**

**B-tech Specialization in**

**Banking, financial services, and insurance.**

**E-commerce, Retail, and Automation**

**Under the guidance of**

**Mr. Amitava Choudhury**

**Assistant Professor, Department of Informatics, School of Computer science, UNIVERSITY OF PETROLEUM & ENERGY STUDIES**

**SUBMITTED BY**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Roll number** | **SAP ID** | **Branch** |
| Anushka Bhardwaj | R114218004 | 500068040 | B.tech CSE ECRA |
| Rima Giri | R114218016 | 500069082 | B.tech CSE ECRA |
| Vaishnavie | R133218029 | 500069265 | B.tech CSE BFSI |

****

**SCHOOL OF COMPUTER SCIENCE**

**UNIVERSITY OF PETROLEUM & ENERGY STUDIES**

**Bidholi Campus, Energy Acres, Dehradun – 248007.**

**July Dec-2020**

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Topic** | **Page** |
| **01** | **Abstract** | **03** |
| **02** | **Introduction** | **03-09** |
| **03** | **Motivation** | **09** |
| **04** | **Problem statement** | **09** |
| **05** | **Literature Review** | **09-14** |
| **06** | **Objective** | **15** |
| **07** | **Methodology** | **15-16** |
| **08** | **System Requirements** | **17** |
| **09** | **Schedule** | **18** |
| **10** | **References** | **18-19** |

**MINOR 1**

**PROJECT TITLE:** Data security and privacy for an end to end message communication.

**ABSTRACT:**

This project aims at taking a step into the field of data security by creating a machine that is capable of data encryption. The input is taken from the user at runtime in the form of a text which is stored in an array. This string array is then passed through the machine which encrypts the data and this encrypted data is then transmitted as a ciphertext. At the receiving end, the data can only be decrypted if the receiving machine has the correct encryption key. If the encryption key matches, the ciphertext undergoes decryption, and the text sent by the sender is displayed at the receiver’s end.

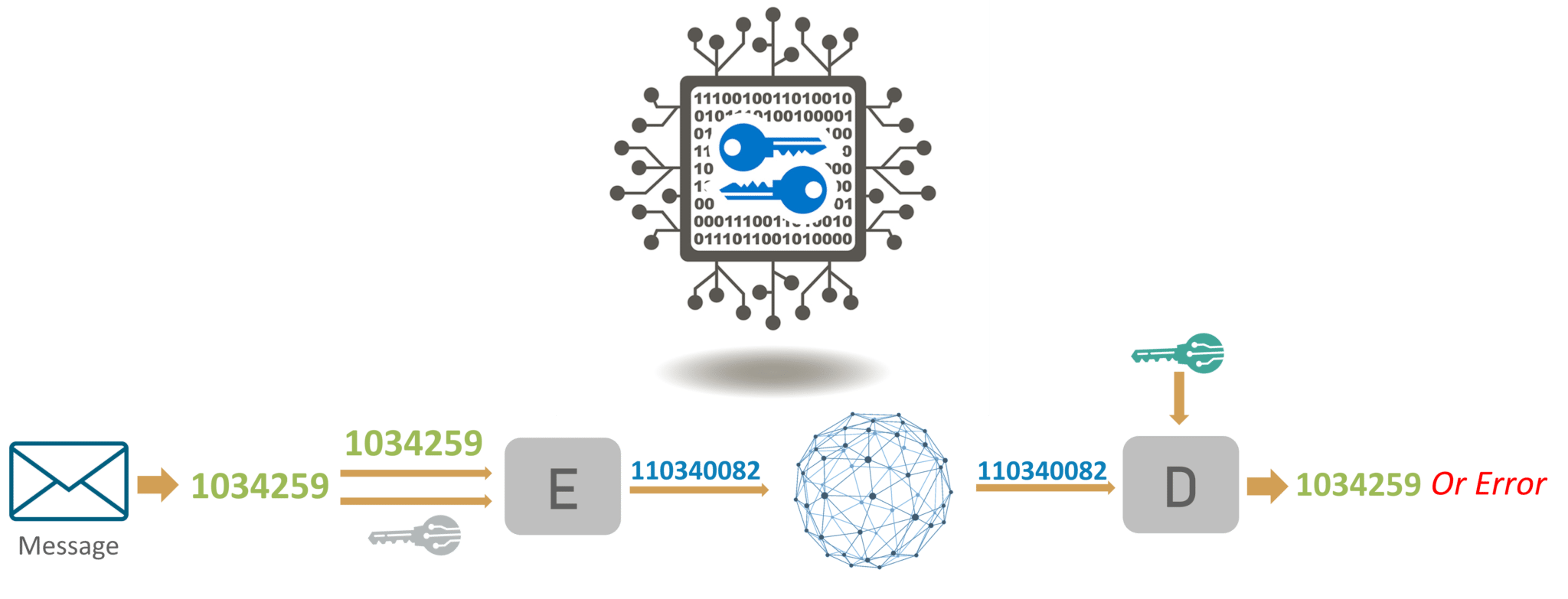
**KEYWORDS:** Data encryption, cryptography, an encryption key, ciphertext

**INTRODUCTION:**

Cryptography is essentially important because it allows you to securely protect data that you don’t want anyone else to have access to. It is used to protect corporate secrets, secure classified information, and to protect personal information to guard against things like identity theft.

Cryptography is the study and practice of a set of techniques used to protect the integrity of networks, programs, and data from attack[1-2], damage, or unauthorized access. The main goal of cryptography is to make communication secure.

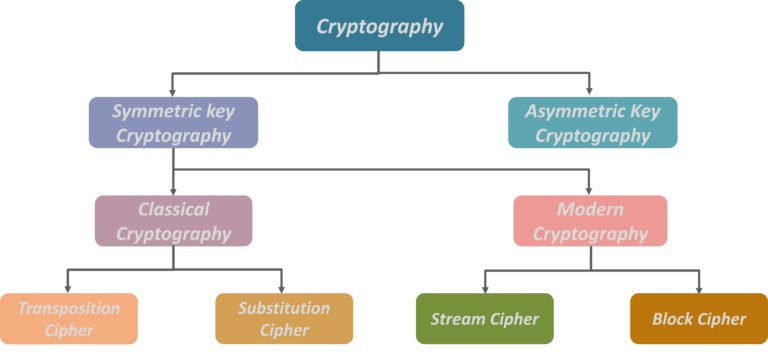
The process of how this works can be understood from the figure 1



**Figure 1:Cryptography**

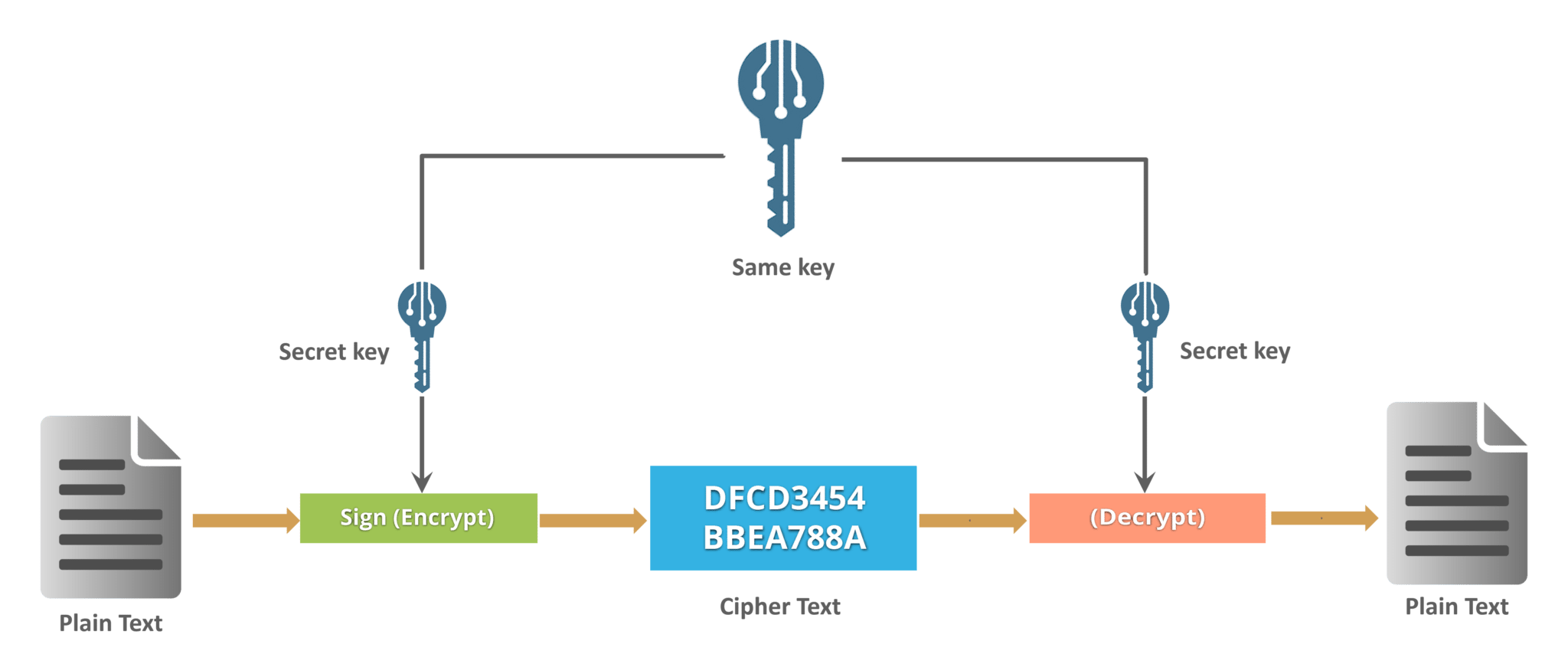
First, we have a message which is converted into a numeric form and then this numeric form is applied with a key, called the encryption key and this key is used in an encryption algorithm. So once the numeric message and the encryption key has been applied to the encryption algorithm[3], we get the ciphertext. This ciphertext is sent over the network to the other side of the world where the other person, for whom the message is intended for, will use the decryption key and use the ciphertext as the parameter for the decryption algorithm, and then he’ll get what we actually sent as a message and if some error had actually occurred, he’ll get an error message[6].

Figure 2: Based on the type of keys and encryption algorithms, cryptography can be classified under the following categories:



**Figure 2 :Types of Cryptography**

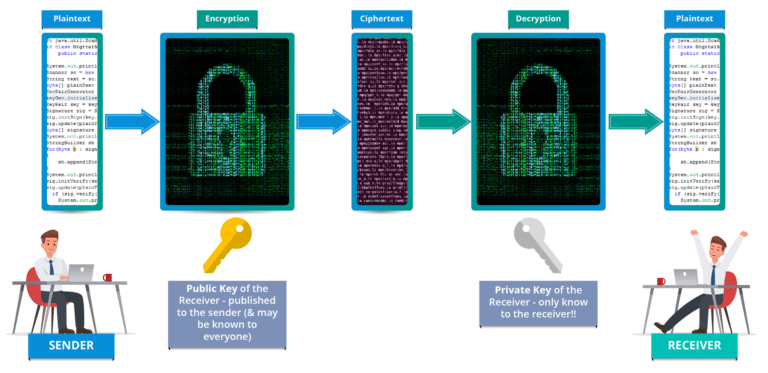
**Symmetric key cryptography:** Symmetric key algorithms are algorithms for cryptography that use the same key for both encryption of plain text and decryption of ciphertext [7]. The keys may be identical or there may be some simple transformation to go between the two keys. The keys in practice represent [3] a shared secret between two or more parties that can be used to maintain a private information link. The requirement that both parties have access to the secret key is not the main drawback of symmetric key encryption in comparison to public encryption, also known as Asymmetric key encryption shown in Figure 3.



**Figure 3:Symmetric key cryptography**

Symmetric key cryptography is also called secret-key cryptography and the most popular symmetric key encryption system is Data Encryption Standard (DES).

**Asymmetric key cryptography:** Asymmetric key cryptography (figure 4), also known as Public-key cryptography is any cryptographic system that uses a pair of keys which is a public that may be emanated widely and private keys which may be known only to the owner. This accomplishes two functions, authentication where the public key verifies that the holder of the paired private key sent the message[3], and encryption where the paired private key holder can decrypt the message, encrypted with the public key [2].



**Figure 4 :Asymmetric Key cryptography**

**RSA ALGORITHM**

**RSA (Rivest–Shamir–Adleman)** is an [algorithm](https://simple.wikipedia.org/wiki/Algorithm) used by modern computers to [encrypt](https://simple.wikipedia.org/wiki/Encryption) and decrypt messages. It is an asymmetric [cryptographic](https://simple.wikipedia.org/wiki/Cryptography) [algorithm](https://simple.wikipedia.org/wiki/Algorithm). Asymmetric means that there are two different [keys](https://simple.wikipedia.org/wiki/Key_(cryptography)). This is also known as [public-key cryptography](https://simple.wikipedia.org/wiki/Public-key_cryptography) because one of the keys can be given to anyone[7]. The other key must be kept private. The algorithm is based on the fact that finding the [factors](https://simple.wikipedia.org/wiki/Factorization) of a large [composite number](https://simple.wikipedia.org/wiki/Composite_number) is difficult: when the factors are [prime numbers](https://simple.wikipedia.org/wiki/Prime_number), the problem is called [prime factorization](https://simple.wikipedia.org/wiki/Prime_factorization). It is also a key pair (public and private key) generator.

Encryption

If the secret was important enough, you wouldn’t risk writing it down normally–spies or a rogue postal employee could be looking through your mail. Likewise, someone could be tapping your phone without your knowledge and logging every single call you make.

One solution to prevent eavesdroppers from accessing message contents is to encrypt it. This basically means to add a code to the message which changes it into a jumbled mess. If your code is sufficiently complex, then the only people who will be able to access the original message are those who have access to the code [1].

If you had a chance to share the code with your friend beforehand, then either of you can send an encrypted message at any time, knowing that you two are the only ones with the ability to read the message contents. But what if you didn’t have a chance to share the code beforehand?

This is one of the fundamental problems of cryptography, which has been addressed by public-key encryption schemes (also known as asymmetric encryption) like RSA.

Under RSA encryption, messages are encrypted with a code called a public key, which can be shared openly. Due to some distinct mathematical properties of the RSA algorithm, once a message has been encrypted with the public key, it can only be decrypted by another key, known as the private key. Each RSA user has a key pair consisting of their public and private keys. As the name suggests, the private key must be kept secret.

Public key encryption schemes differ from symmetric-key encryption, where both the encryption and decryption process uses the same private key. These differences make public-key encryption like RSA useful for communicating in situations where there has been no opportunity to safely distribute keys beforehand.

Symmetric-key algorithms have their own applications, such as encrypting data for personal use, or for when there are secure channels that the private keys can be shared over.

Decryption

1. The Key Distribution Problem

Private-key systems suffer from the key distribution problem. In order for secure communication to occur, the key must first be securely sent to the other party. An unsecured channel such as a data network cannot be used.

Couriers or other secure means are typically used. Public-key systems do not suffer from this problem because of their use of two different keys[6]. Messages are encrypted with a public key and decrypted with a private key. No keys need to be distributed for secure communication to occur.

2. Public-Key Cryptosystems

A user wishing to exchange encrypted messages using a public-key cryptosystem would place their public encryption procedure, E, in a public file. The user's corresponding decryption procedure, D, is kept confidential. Rivest, Shamir, and Adleman provide four properties that the encryption and decryption procedures have three important steps:

Deciphering the enciphered form of a message M yields M. That is, D(E(M)) = M

E and D are easy to compute [4].

Publicly revealing E does not reveal an easy way to compute D. As such, only the user can decrypt messages which were encrypted with E. Likewise, only the user can compute D efficiently.

**RSA security**

RSA security relies on the computational difficulty of factoring large integers. As computing power increases and more efficient factoring algorithms are discovered, the ability to factor larger and larger numbers also increases.

Security of RSA:-

These are explained as follows below.

1. Plain text attacks:

It is classified into 3 subcategories:-

(i) Short message attack:

In this, we assume that the attacker knows some blocks of plain text and tries to decode ciphertext with the help of that. So, to prevent this pad the plain text before encrypting.

(ii) Cycling attack:

This attacker will think that plain text is converted into ciphertext using permutation, and he will apply right for conversion. But the attacker does not write plain text. Hence, will keep doing it.

(iii) Unconcealed Message attack:

Sometimes it happens that plain text is the same as ciphertext after encryption. So it must be checked that it cannot be attacked.

2. Chosen cipher attack:

This attacker is able to find plain text based on ciphertext using the Extended Euclidean Algorithm.

3. Factorization attack:

If an attacker is able to know P and Q using N, then he could find out the value of the private key. This can be failed when N contains at least 300 longer digits in decimal terms, the attacker will not be able to find. Hence, it fails.

**MOTIVATION**

The key motivation behind selecting this project is to involve secure communication, which is possible by using data security. It has great potential in various aspects such as security, privacy.

**PROBLEM STATEMENT**

There are many challenges faced by people while communicating between two devices. There is a chance to steal and manipulate data, so we need data security and privacy for an end to end message communication.

**LITERATURE REVIEW:**

In recent years network security has become an important issue. Encryption has come up as a solution, and plays an important role in the information security system [1]. Many techniques are needed to protect the shared data. The present work focuses on cryptography to secure the data while transmitting in the network. Firstly the data which is to be transmitted from sender to receiver in the network must be encrypted using the encryption algorithm in cryptography [4]. Secondly, by using decryption technique the receiver can view the original data. We implemented three encrypt techniques like AES, DES algorithms and compared their performance of encryption techniques based on the analysis of its stimulated time at the time of encryption and decryption. Experiment results are given to analyze the effectiveness of each algorithm.

Advanced Encryption Standard (AES) :

Advanced Encryption Standard (AES) algorithm (Figure 5) not only for security but also for great speed. Both hardware and software implementation are faster still. New encryption standard recommended by NIST to replace DES. Encrypts data blocks of 128 bits in 10, 12 and 14 rounds depending on key size. It can be implemented on various platforms especially in small devices. It is carefully tested for many security applications.

i. Algorithm Steps :

These steps used to encrypt the 128-bit block:

1. The set of round keys from the cipher key.
2. Initialize the state array and add the initial round key to the starting state array.
3. Perform round = 1 to 9 : Execute Usual Round.
4. Execute Final Round.
5. Corresponding ciphertext chunk output of Final Round Step

ii. Usual Round :

Execute the following operations which are described above.

1. Sub Bytes
2. Shift Rows
3. Mix Columns
4. Add Round Key, using K(round)

iii. Final Round:

Execute the following operations which are described above.

1. Sub Bytes
2. Shift Rows
3. Add Round Key, using K(10)

iv. Encryption :

Each round consists of the following four steps:

1. Sub Bytes : The first transformation, Sub Bytes, is used at the encryption site. To substitute a byte, we interpret the byte as two hexadecimal digits.
2. Shift Rows : In the encryption, the transformation is called Shift Rows.
3. Mix Columns : The Mix Columns transformation operates at the column level; it transforms each column of the state to a new column.
4. Add Round Key : Add Round Key proceeds one column at a time. Add Round Key adds a round key word with each state column matrix; the operation in Add Round Key is matrix addition. The last step consists of XO Ring the output of the previous three steps with four words from the key schedule. And the last round for encryption does not involve the “Mix columns” step. [8]

v. Decryption:

Decryption involves reversing all the steps taken in encryption using inverse functions like

1. Inverse shift rows
2. Inverse substitute bytes
3. Add round key, and
4. Inverse mix columns.

The third step consists of XO Ring the output of the previous two steps with four words from the key schedule. And the last round for decryption does not involve the “Inverse columns” step. AES shown in Figure 9.

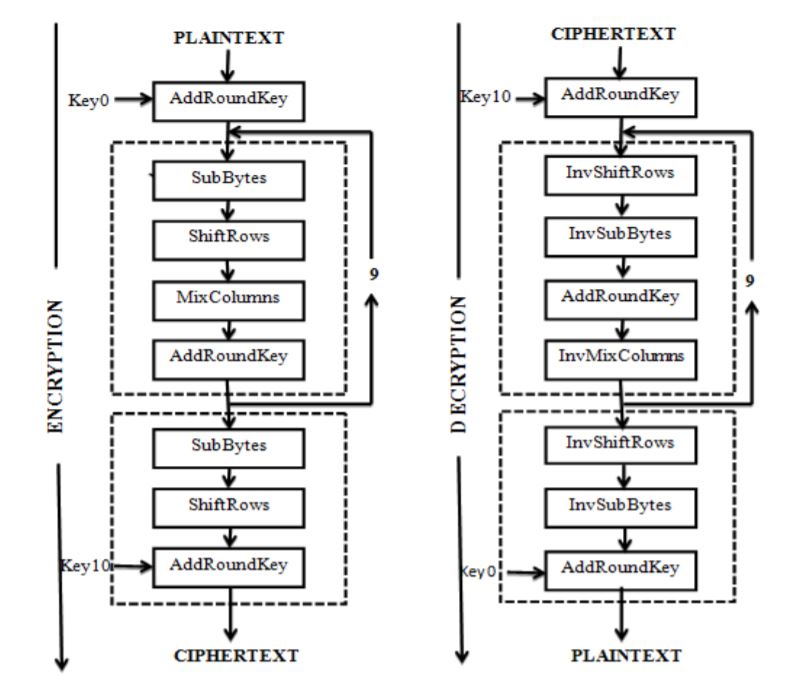


Figure 5: AES

b) Data Encryption Standard (DES)

DES (Data Encryption Standard) algorithm purpose is to provide a standard method for protecting sensitive commercial and unclassified data. In this same key used for encryption and decryption process [7]. DES algorithm consists of the following steps:

i. Encryption

1. DES accepts an input of 64-bit long plaintext and 56-bitkey (8 bits of parity) and produces output of 64 bit block.
2. The plaintext block has to shift the bits around.
3. The 8 parity bits are removed from the key by subjecting the key to its Key Permutation.
4. The plaintext and key will be processed by following

ii. The key is split into two 28 halves

iii. Each half of the key is shifted (rotated) by one or two bits, depending on the round.

iv. The halves are recombined and subject to a compression permutation to reduce the key from 56 bits to 48 bits. These compressed keys were used to encrypt this round’s plaintext block.

v. The rotated key halves from step 2 are used in the next round.

vi. The data block is split into two 32-bit halves.

vii. One half is subject to an expansion permutation to increase its size to 48 bits. viii. Output of step 6 is exclusive-OR’ed with the 48- bit compressed key from step 3.

ix. Output of step 7 is fed into an S-box, which substitutes key bits and reduces the 48-bit block back down to 32-bits.

x. Output of step 8 is subject to a P-box to permute the bits

xi. The output from the P-box is exclusive-OR’ed with other half of the data block. k. The two data halves are swapped and become the next round’s input.

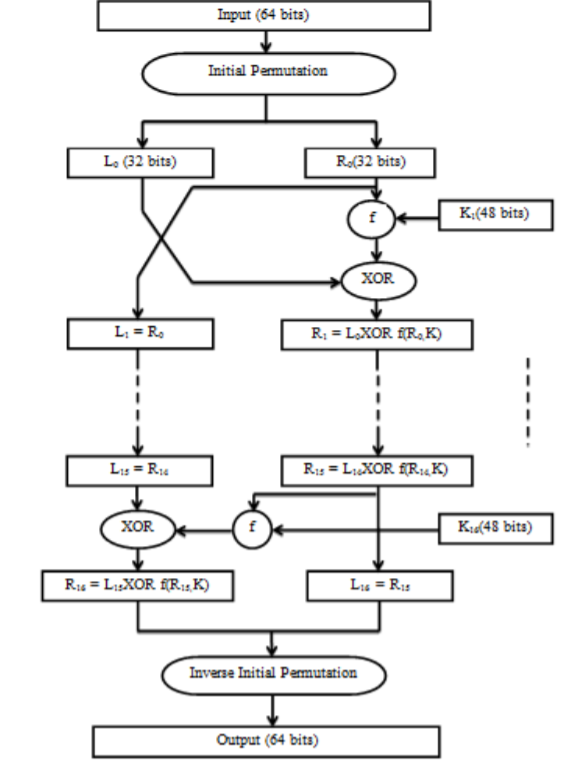


Figure 6 :DES

**OBJECTIVE**

With our project we aim to realize the following objectives:-

1. Data integrity: The receiver of a message should be able to check whether the message was modiﬁed during transmission, either accidentally or deliberately. No one should be able to substitute a false message for the original message, or for parts of it.
2. Authentication: The receiver of a message should be able to verify its origin.
3. Non-repudiation: The sender should not be able to later deny that she sent a message[5].

**METHODOLOGY**

For implementing and achieving data security, we will be using the following algorithms in C language:

RSA algorithm

Counting algorithm

RSA(Figure 7) involves a public key and a private key. The public key can be known to everyone- it is used to encrypt messages. Messages encrypted using the public key can only be decrypted with the private key. The keys for the RSA algorithm are generated in the following way:

1. Choose two different large random [prime numbers](https://simple.wikipedia.org/wiki/Prime_number) p,q
2. Calculate n=p\*q where n is the modulus for the public and private keys.
3. Calculate totient phi(n) =(p-1)\*(q-1)
4. Choose e such that 1<e<phi(n), e is coprime to phi(n) i.e, gcd(e,phi(n)=1) where e is derived number and used as a public key exponent.
5. Compute d to satisfy the congruence relation de **≅** 1 (mod phi (n)) simple way to calculate de\*mod phi(n) =1
6. For encryption C=m^e mod(n) where m is plain text, c is ciphertext
7. For decryption m=c^d mod(n)

Step by step process shown in figure 7

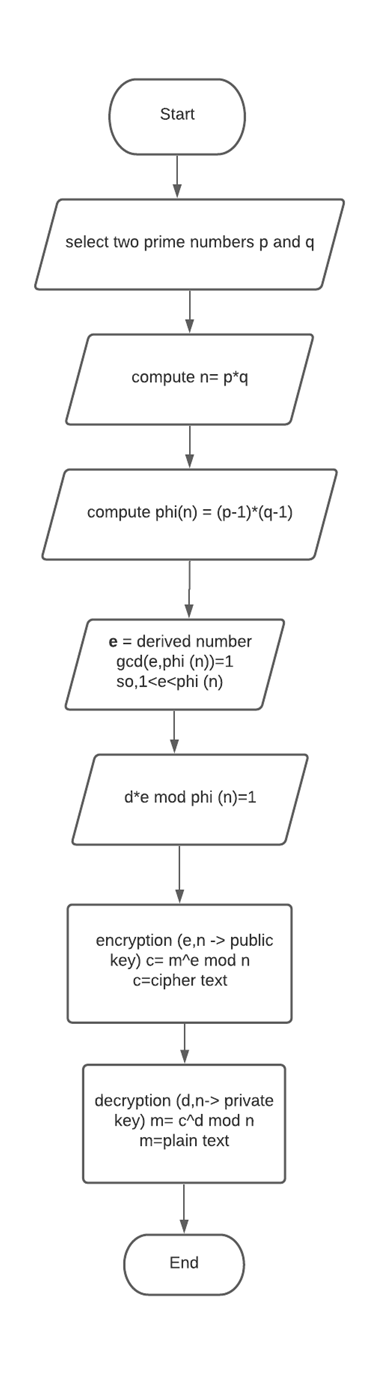


Figure 7:RSA Algorithm

**SYSTEM REQUIREMENTS**

Software Requirements

|  |  |
| --- | --- |
| **Name of component** | **Specification** |
| Operating System | Windows 10 |
| Language | C |

Hardware Requirements

|  |  |
| --- | --- |
| **Name of component** | **Specification** |
| Processor | Intel(R) Core(TM)i5-3210M CPU @ 2.50GHz 2.50 GHz |
| RAM | 4 GB |
| Hard Disk | 1 TB |

**SCHEDULE:**

**(pert chart)**

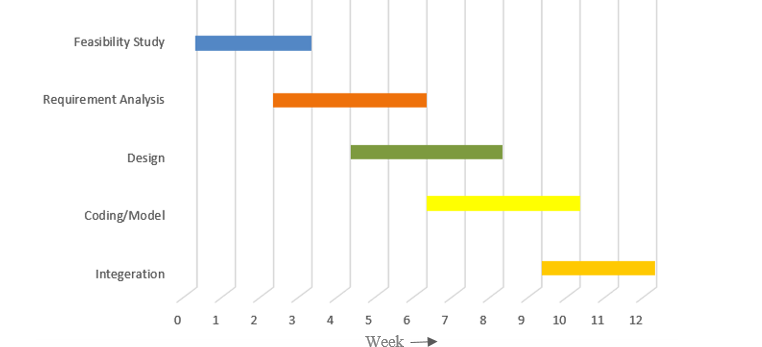


Figure 8:Pert Chart

**REFERENCES:**

[1] Elminaam, Diaa Salama Abd, Abdual Kader, Hatem Mohamed & Hadhoud, Mohiy Mohamed. “Evaluating The Performance of Symmetric Encryption Algorithms”. International Journal of Network Security, Vol.10, No.3, May 2010, pp. 216

[2]Chehal Ritika, Singh Kuldeep. “Efficiency and Security of Data with Symmetric Encryption Algorithms”. International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, Volume 2, Issue 8, August 2012, pp. 1.

[3]Abdullah Al Hasib, Abul Ahsan Md. Mahmudul Haque, “A Comparative Study of the Performance and Security Issues of AES and RSA Cryptography”, Third International Conference on Convergence and Hybrid Information Technology, 2008.

[4]Prashanti.G, Deepthi.S & Sandhya Rani.K. ”A Novel Approach for Data Encryption Standard Algorithm”. International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-5, June 2013, pp. 264.

[5]Li Dongjiang, Wang Yandan, Chen Hong, “The research on key generation in RSA public- key cryptosystem”, Department of Computer Science, North China Electric Power University, Beijing, China, Fourth International Conference on Computational and Information Sciences 2012.

[6]Vishwa Gupta, Gajendra Singh, Ravindra Gupta, “Advance Cryptography algorithm for improving data security”, International Journal of Advanced Research in Computer Science and Software Engineering, Vol 2, Issue 1, Jan 2012.

[7]Das Debasis, Misra Rajiv. “Programmable Cellular Automata Based Efficient Parallel AES Encryption Algorithm”. International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.6, November 2011, pp. 204

[8] Matus Nemec, Marek Sys, Petr Svenda, Dusan Klinec,Vashek Matyas, “3TThe Return of Coppersmith’s Attack:” Practical Factorization of Widely Used RSA Moduli3T, , November 2017.