CRYPTOGRAPHIC TECHNIQUE FOR COMMUNICATION SYSTEM

by

Abhinav Dwivedi (1901430100004)

Aditya Bhardwaj (1901430100010)

Amish Jha (1901430100022)

Anmol Saroha (1901430100030)

Aradhy Mishra (1901430100041)

Under the Supervision of

Mr. Aditya Sam Koshy

Assistant Professor

Department of Computer Science and Engineering

IMS Engineering College

NH-09, Adhyatmik Nagar, Near Dasna,

Distt: Ghaziabad, UP

June, 2023

VISION OF THE INSTITUTE

To make IMSEC an Institution of Excellence for empowering students through technical education coupled with incorporating values and developing engineering acumen for innovations and leadership skills for the betterment of society.

MISSION OF THE INSTITUTE

- To promote academic excellence by continuous learning in core and emerging Engineering areas using innovative teaching and learning methodologies.
- To inculcate values and ethics among the learners.
- To promote industry interactions and produce young entrepreneurs.
- To create a conducive learning and research environment for life-longlearning to develop the students as technology leaders and entrepreneursfor addressing societal needs.

VISION OF THE DEPARTMENT

To provide globally competent professionals in the field of Computer Science & Engineering embedded with sound technical knowledge, aptitude for research and innovation with ethical values to cater to the industrial & societal needs.

MISSION OF THE DEPARTMENT

M1: To provide quality undergraduate education in both the theoretical& applied foundations of Computer Science Engineering.

M2: Conduct research to advance the state of the art in Computer Science &Engineering and integrate the research results as innovations.

M3: To inculcate team building skills and promote life-long learning with ahigh societal and ethical values.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

Graduate Will:

PEO1: Possess knowledge to enable continued professional development.

PEO2: Engage in life-long learning to foster personal & organization growth.

PEO3: Work productively as successful professionals in diverse career paths.

PEO4: Effectively communicate ideas to promote collaboration in accordance with societal standards & ethical practices.

PROGRAMME SPECIFIC OUTCOME (PSOs)

PSO1: To apply standard software engineering practices & strategies in real-timesoftware project development.

PSO2: To apply latest programming languages in creating innovative career opportunities.

PROGRAM OUTCOMES

Engineering Graduates will be able to:

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. Problem analysis: Identity, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and cultural, societal, and environmental considerations.
4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activitieswith the engineering community and with society at large, such as, beingable to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. Project management and finance: Demonstrate knowledge and understandingof the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

CO-PO-PSO MAPPING FOR ACADEMIC SESSION 2022-23

Course Name: Project AKTU Course Code: KCS851

Semester / Year: VIII/ 4th NBA Code:C411

Course Outcomes:

CO. No.	DESCRIPTION	COGNITIVE LEVEL (BLOOMS TAXONOMY)
C411.1	Analyze and understand the real-life problem and apply theirknowledge to get programming solution.	**K4 , K5**
C411.2	Engage in the creative design process through the integration and application of diverse technical knowledge and expertise to meet customer needs and address social issues.	**K4 , K5**
C411.3	Use the various tools and techniques, coding practices fordeveloping real life solution to the problem.	**K5 , K6**
C411.4	Find out the errors in software solutions and establishing theprocess to design maintainable software applications	**K4 , K5**
C411.5	Write the report about what they are doing in project and learningthe team working skills	**K5, K6**

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
C411.1	3	3	3	3	3	1	1	2	2	3	3	3	3	3
C411.2	3	3	3	3	3	1	1	2	2	3	3	3	3	3
C411.3	3	3	3	3	3	1	1	2	2	3	3	3	3	3
C411.4	3	3	3	3	3	1	1	2	2	3	3	3	3	3
C411.5	3	3	3	3	3	1	1	3	3	3	3	3	3	3
Avg.	3	3	3	3	3	1	1	2.2	2.2	3	3	3	3	3

DECLARATION

I hereby declare that the work, which is being presented in the Project, entitled **"Cryptography Technique For Communication System"** in partial fulfillment for the award of Degree of "Bachelor of Technology" in the Department of Computer Science & Engineering, and **submitted to the Department of Computer Science & Engineering**, IMS Engineering College, Ghaziabad, affiliated to **Dr. A.P.J Abdul Kalam Technical University, Lucknow, Uttar Pradesh** is a record of my own investigations carried out under the Guidance of **Mr. Aditya Sam Koshy, Assistant Professor**, IMS Engineering College, Ghaziabad.

I have not submitted the matter presented in this Project anywhere for the award of any other Degree.

Signature:

Name: Abhinav Dwivedi

Roll No: 1901430100004

Date:

Signature:

Name: Aditya Bhardwaj

Roll No: 1901430100010

Date:

Signature:

Name: Amish Jha

Roll No: 1901430100022

Date:

Signature:

Name: Anmol Saroha

Roll No: 1901430100030

Date:

Signature:

Name: Aradhy Mishra

Roll No: 1901430100041

Date:

CERTIFICATE

I hereby certify that the work which is being presented in the project report entitled "Cryptography Technique For Communication System" by "**Abhinav Dwivedi, Aditya Bhardwaj, Amish Jha, Anmol Saroha, and Aradhy Mishra**" in partial fulfillment of requirements for the award of the degree of B.Tech. (CSE) submitted in the Department of CSE at "IMS Engineering College" under **A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW, UTTAR PRADESH** is an authentic record of my own work carried out under the supervision of **Mr. Aditya Sam Koshy (Assistant Professor).**

Mr. Aditya Sam Koshy
Assistant Professor
IMS Engineering College, Ghaziabad

ACKNOWLEDGEMENT

I would like to place on record my deep sense of gratitude to **Mr. Aditya Sam Koshy** Dept. of Computer Science & Engineering, IMSEC, Ghaziabad, India for his generous guidance, help and useful suggestions.

I express my sincere gratitude to **Prof. Dr. Sonali Mathur, HOD** in Department of Computer Science & Engineering, IMSEC, Ghaziabad, for her stimulating guidance, continuous encouragement, and supervision throughout the course of present work.

I am extremely thankful to **Prof. Dr. Vikram Bali, Director**, IMSEC, Ghaziabad, for providing me infrastructural facilities to work in, without which this work would not have been possible.

Signature:

Name: Abhinav Dwivedi

Roll No: 1901430100004

Date:

Signature:

Name: Aditya Bhardwaj

Roll No: 1901430100010

Date:

Signature:

Name: Amish Jha

Roll No: 1901430100022

Date:

Signature:

Name: Anmol Saroha

Roll No: 1901430100030

Date:

Signature:

Name: Aradhy Mishra

Roll No: 1901430100041

Date:

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Secure Communication of message from sender to receiver is one of the main security concern of Internet users across world. It is because of the regular attacks and threats and most Important Data Privacy. In order to sort out these issues, we use cryptographic algorithm which encrypts data in some cipher and transfers it over the internet and again decrypted to original data.

Thus, lightweight cryptography methods are proposed to overcome many of the problems of conventional cryptography. Ciphers act as encapsulating system for message. Hybrid Algorithm will be formed from use of different types of ciphers. The cryptosystem performs its encryption by encrypting the plaintext using Vigenere Cipher and further again processing though Polybius Cipher.

CHAPTER 1

INTRODUCTION

Information security can be summed up to info, a group of steps, procedures, and strategies that are used to stop and observe illegal access, trouble- shooting, revelation, perturbation and adjustment of computer network sources. Enhancing the privacy, eligibility and reliability of the work requires a lot work to strengthen the current methods from constant trials to break them and to improve new ways that are resistant to most kinds of attacks if not all. Accordingly, it was proven that encoding is one of the most reliable strategies used to secure information since the ancient days of the Romans who used similar methods to enable security on their valued information and documents.

Cryptography is the art of creating written or generated codes that allow information to be kept secret. Cryptography converts data into a format that is unreadable for an unauthorized user, allowing it to be transmitted without unauthorized entities decoding it back into a readable format, thus compromising the data. Information security uses cryptography on several levels. The information cannot be read without a key to decrypt it.

Cryptography is technique of securing information and communications through use of codes so that only those people for whom the information is intended can understand it and process it. Thus preventing un authorized access to information. The prefix "crypt" means "hidden" and suffix graphy means "writing".

1.1-Overall Description:

In Cryptography the techniques which are use to protect information are obtained from mathematical concepts and a set of rule based calculations known as algorithms to convert messages in ways that make it hard to decode it. These algorithms are used for cryptographic key generation, digital signing, and verification to protect data privacy, web browsing on internet and to protect confidential transactions such as credit card and debit card transactions.

Techniques used For Cryptography: In today's age of computers, cryptography is often associated with the process where an ordinary plain text is converted to cipher text which is the text made such that intended receiver of the text can only decode it and hence this process is known as encryption. The process of conversion of cipher text to plain text this is known as decryption.

1.2-Features of Cryptography: These are mentioned below:

Confidentiality:

Information can only be accessed by the person for whom it is intended and no other person except him can access it.

Integrity:

Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.

Non-repudiation:

The creator/sender of information cannot deny his or her intention to send information at later stage.

Authentication:

The identities of sender and receiver are confirmed. As well as destination/origin of information is confirmed.

1.3-Types of Cryptography:

In general, there are three types of cryptography which are explained as follows.

Symmetric Key Cryptography:

It is an encryption system where the sender and receiver of message use a single common key to encrypt and decrypt messages. Symmetric Key Systems are faster and simpler but the problem is that sender and receiver have to somehow exchange key in a secure manner. The most popular symmetric key cryptography system is Data Encryption System (DES).

Hash Functions:

There is no usage of any key in this algorithm. A hash value with fixed length is calculated as per the plain text which makes it impossible for contents of plain text to be recovered. Many operating systems use hash functions to encrypt passwords.

Asymmetric Key Cryptography:

Under this system a pair of keys is used to encrypt and decrypt information. A public key is used for encryption and a private key is used for decryption. Public key and Private Key are different.

Even if the public key is known by everyone the intended receiver can only decode it because he alone knows the private key.

In figure 1.1, Cryptographic design for the process is shown.

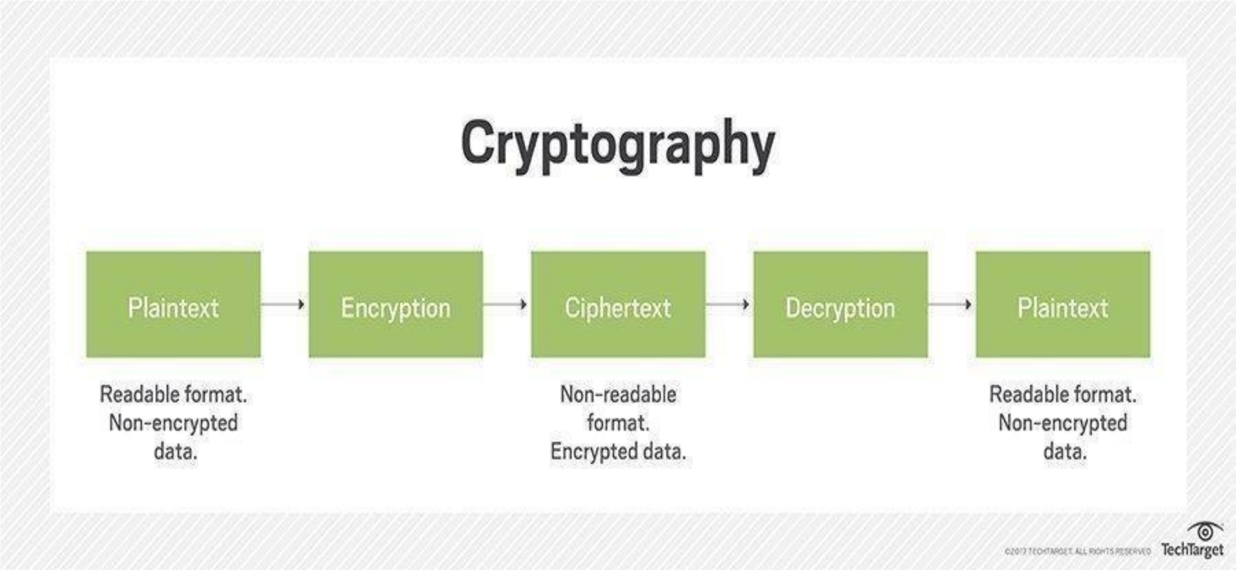

Cryptography
Plaintext
Encryption
Ciphertext
Decryption
Plaintext
Readable format. Non-encrypted data.
Non-readable format. Encrypted data.
Readable format. Non-encrypted data.
©2017 TECHTARGET. ALL RIGHTS RESERVED
TechTarget

Figure 1.1: Cryptographic Design[1]

1.4-Components of a Cryptosystem:

The various components of a basic cryptosystem are as follows:

Plaintext:

It is the original data to be protected during transmission.

Cipher text:

It is the scrambled version of the plaintext produced by the encryption algorithm using a specific the encryption key. The ciphertext is not guarded. It flows on public channel. It can be intercepted or compromised by anyone who has access to the communication channel.

Encryption Algorithm:

It is a mathematical process that produces a ciphertext for any given plaintext and encryption key. It is a cryptographic algorithm that takes plaintext and an encryption key as input and produces a ciphertext.

Decryption Algorithm:

It is a mathematical process, that produces a unique plaintext for any given ciphertext and decryption key. It is a cryptographic algorithm that takes a ciphertext and a decryption key as input, and outputs a plaintext. The decryption algorithm essentially reverses the encryption algorithm and is thus closely related to it.

Encryption Key:

It is a value that is known to the sender. The sender inputs the encryption key into the encryption algorithm along with the plaintext in order to compute the cipher text.

Decryption Key:

It is a value that is known to the receiver. The decryption key is related to the encryption key, but is not always identical to it. The receiver inputs the decryption key into the decryption algorithm along with the cipher text in order to compute the plaintext.

For a given cryptosystem, a collection of all possible decryption keys is called a **key space**. An **interceptor** (an attacker) is an unauthorized entity who attempts to determine the plaintext. He can see the ciphertext and may know the decryption algorithm. He, however, must never know the decryption key.

Purpose:

In digital world cyber-attacks are very frequent. Any social networking site, web application, etc are more prone to attacks. To counter this Study and analysis of attacks is usually important s this guide to solve major problem and make system anti attacks. Some of the attacks are given as follows.

1.5 Cryptographic Attacks:

The basic intention of an attacker is to break a cryptosystem and to find the plaintext from the ciphertext. To obtain the plaintext, the attacker only needs to find out the secret decryption key, as the algorithm is already in public domain.

Hence, he applies maximum effort towards finding out the secret key used in the cryptosystem. Once the attacker is able to determine the key, the attacked system is considered as broken or compromised.

Based on the methodology used, attacks on cryptosystems are categorized as follows −

Ciphertext Only Attacks (COA):

In this method, the attacker has access to a set of ciphertext(s). He does not have access to corresponding plaintext. COA is said to be successful when the corresponding plaintext can be determined from a given set of ciphertext. Occasionally, the encryption key can be determined from this attack. Modern cryptosystems are guarded against ciphertext-only attacks.

Known Plaintext Attack (KPA):

In this method, the attacker knows the plaintext for some parts of the ciphertext. The task is to decrypt the rest of the ciphertext using this information. This may be done by determining the key or via some other method. The best example of this attack is linear cryptanalysis against block ciphers.

Chosen Plaintext Attack (CPA):

In this method, the attacker has the text of his choice encrypted. So he has the ciphertext-plaintext pair of his choice. This simplifies his task of determining the encryption key. An example of this attack is differential cryptanalysis applied against block ciphers as well as hash functions. A popular public key cryptosystem, RSA is also vulnerable to chosen-plaintext attacks.

Dictionary Attack:

This attack has many variants, all of which involve compiling a ‘dictionary’. In simplest method of this attack, attacker builds a dictionary of ciphertexts and corresponding plaintexts that he has learnt over a period of time. In future, when an attacker gets the ciphertext, he refers the dictionary to find the corresponding plaintext.

Brute Force Attack (BFA):

In this method, the attacker tries to determine the key by attempting all possible keys. If the key is 8 bits long, then the number of possible keys is $2^8 = 256$. The attacker knows the ciphertext and the algorithm, now he attempts all the 256 keys one by one for decryption. The time to complete the attack would be very high if the key is long.

Man in Middle Attack (MIM):

The targets of this attack are mostly public key cryptosystems where key exchange is involved before communication takes place.

Host A wants to communicate to host B, hence requests public key of B.

An attacker intercepts this request and sends his public key instead. Thus, whatever hosts A sends to host B, the attacker is able to read.

In order to maintain communication, the attacker re-encrypts the data after reading with his public key and sends to B.

The attacker sends his public key as A’s public key so that B takes it as if it is taking it from A.

Side Channel Attack (SCA):

This type of attack is not against any particular type of cryptosystem or algorithm. Instead, it is launched to exploit the weakness in physical implementation of the cryptosystem.

Timing Attacks:

They exploit the fact that different computations take different times to compute on processor. By measuring such timings, it

is be possible to know about a particular computation the processor is carrying out. For example, if the encryption takes a longer time, it indicates that the secret key is long.

Power Analysis Attacks:

These attacks are similar to timing attacks except that the amount of power consumption is used to obtain information about the nature of the underlying computations.

Fault analysis Attacks:

In these attacks, errors are induced in the cryptosystem and the attacker studies the resulting output for useful information.

In Figure 1.2 the basic security attack is shown.

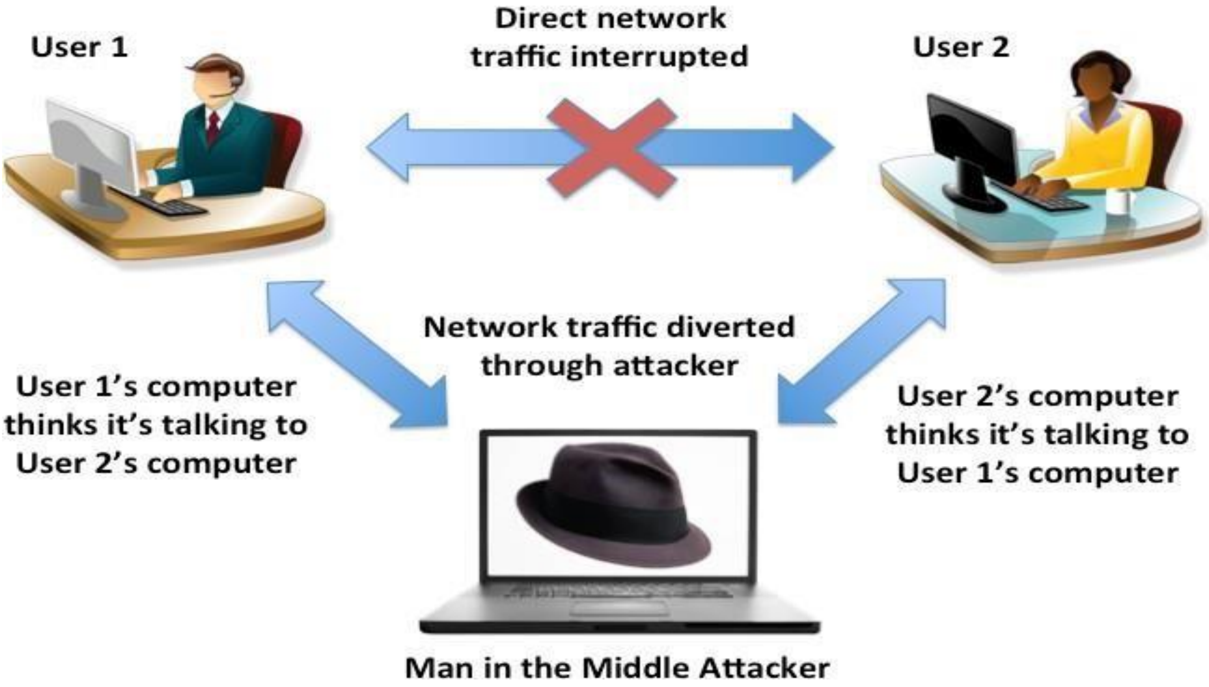

User 1
Direct network traffic interrupted
User 2
Network traffic diverted through attacker
User 1's computer thinks it's talking to User 2's computer
User 2's computer thinks it's talking to User 1's computer
Man in the Middle Attacker

Figure 1.2 : Security Attack[2]

1.6-Cipher

In cryptography, a cipher (or cypher) is an algorithm for performing encryption or decryption—a series of well-defined steps that can be followed as a procedure. An alternative, less common term is encipherment. To encipher or encode is to convert information from plain text into cipher or code. In nontechnical usage, a 'cipher' is the same thing as a 'code'; however, the concepts are distinct in cryptography. In traditional cryptography, ciphers were distinguished from codes. Codes commonly substitute diverse length series of characters in the yield, while ciphers commonly substitute indistinguishable number of characters from are input. There are special cases and some cipher frameworks may utilize marginally more, or less, characters when output versus the number that were input.

There are mainly Two Traditional Ciphers such as:-

Substitution Cipher Technique:

In Substitution Cipher Technique plain text characters are replaced with other characters, numbers and symbols as well as in substitution Cipher Technique, character's identity is changed while its position remains unchanged.

Example – Caesar Cipher, Polybius Cipher, Vigenere Cipher

Transposition Cipher Technique:

Transposition Cipher Technique rearranges the position of the plain text's characters. In transposition Cipher Technique, The position of the character is changed but character's identity is not changed.

Example – Rail fence Cipher

CHAPTER 2

LITERATURE SURVEY

2.1-NEED OF CRYPTOGRAPHY

Cryptography is a type of technology used to secure communication between two or more parties. The use of cryptography is to protect data from unauthorized access, modify, and/or destruction. The main goal of cryptography is to ensure privacy and confidentiality of the transmitted data. Cryptographic techniques can be used for various applications such as message authentication, digital signatures, secure key exchange, and data encryption.

The security related algorithm paper authored by Puneet Kumar highlighted [5] the security for web keeping money, account passwords, messages accounts secret word, etc requires content protection in mechanized media. It shows the security besides; pressure for the information with the move encryption standard. The age of key has been done with the assistance of the Polybius square. The extension in number of rounds it will require increasingly computational speculation and will end up irksome for the software engineer to break the system

Caesar cipher, otherwise called the shift cipher, is one of the least complex and most generally known old style encryption systems. It is a kind of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the letters in order. For example, with a shift of 3, A would be replaced by D, B would become E, etc. The encryption step performed by a Caesar cipher is regularly joined as a component of progressively complex plans, for example, the Vigenère cipher, and still has present day application in the ROT13 framework. Similarly as with all single letters in order substitution ciphers, the Caesar cipher is effortlessly broken and in present day practice offers basically no correspondence security.

[6] In cryptography, a transposition cipher is a process of encryption by which the positions held by units of plaintext are shifted by a customary framework or example, so that the ciphertext comprises a stage of the plaintext. That is, the request for the units is changed toward the finish of the shifting process. Mathematically, a bijective function is utilized on the characters' positions to encode and an inverse function to decrypt. The letters themselves are kept unaltered, which suggests that the impact is just on their positions just, making their request inside the message mixed by a few all around characterized scheme. Numerous transposition ciphers are done as per a geometric design.[7][8]

In [9] changed variant of vigenere algorithm was proposed in which dispersion is given by adding an arbitrary piece to every byte before the message is scrambled utilizing Vigenere. This strategy falls flat kasiski assault to discover the length of key on the grounds that the cushioning of message with irregular bits. The fundamental downside of this system is that the size of the scrambled message will be expanded by around 56%.

In [10] another method for executing Vigenere algorithm was presented via naturally changing the cipher key after every encryption step. In this technique progressive keys were utilized that were reliant on the underlying key an incentive during the encryption process.

In [11] adjustment of Vigenere cipher by irregular numbers, punctuations and scientific images was introduced. In proposed technique numbers, punctuations furthermore, scientific images were utilized for key instead of characters to make it increasingly hard for animal power assault. It was inferred that if irregular numbers are utilized for key what's more, to spread the range then just skilled people can recognize the message recognize the message.

Another algorithm [12] by combining Vigenere substitution cipher with Stream cipher was proposed in which repeated bits of plaintext consistently encrypted with the diverse segment of the catchphrase or binary key. The letters in odd location were encoded with stream cipher and the letters in even location with Vigenere cipher. It was inferred that proposed algorithm conceals the connection between cipher content and plain content that makes cryptanalysis much troublesome.

Tianfu [13] address that internet is one of the most unsafe communication medium due to huge connection and public network. Information protection is one the of essential requirement. At present various security algorithms are proposed to achieve security during communication. All of them have certain good point and certain bad point. To improve the strength of encryption algorithm they proposed a hybrid model. Proposed model is combination of AES and DES. Both algorithms are symmetric key technique and itself they are very much capable for encryption. Integration of AES and DES would give a strong level of security at encryption end. A significant improvement in results has been observed with proposed solution.

Jakimoski et al. [14] analyzed and evaluated the most important security techniques for data protection that are already accepted from the cloud computing providers. They classified them in four sections according to the security mechanisms that they provide: authentication, confidentiality, access control and authorization. They conclude that if all recommended measures are taken into account providing authentication, confidentiality, access control and authorization, then the cloud computing can be trusted in data protection. They focused on the security issues that should be taken into account in depth in order to have proper data security in the cloud. They recommended important security measures relating to data protection in the cloud that must be taken into account.

2.2-Security Challenges

What is the CIA triad?

Confidentiality, integrity and availability, also known as the CIA triad, is a model designed to guide policies for information security within an organization. The model is also sometimes referred to as the AIC triad (availability, integrity and confidentiality) to avoid confusion with the Central Intelligence Agency. Although elements of the triad are three of the most foundational and crucial cybersecurity needs, experts believe the CIA triad needs an upgrade to stay effective.[14]

In this context, confidentiality is a set of rules that limits access to information, integrity is the assurance that the information is trustworthy and accurate, and availability is a guarantee of reliable access to the information by authorized people.[14]

Confidentiality:

Sometimes safeguarding data confidentiality involves special training for those privy to sensitive documents. Training can help familiarize authorized people with risk factors and how to guard against them. Further aspects of training may include strong passwords and password-related best practices and information about social engineering methods to prevent users from bending data-handling rules with good intentions and potentially disastrous results.[2]

A good example of methods used to ensure confidentiality is requiring an account number or routing number when banking online. Data encryption is another common method of ensuring confidentiality. User IDs and passwords constitute a standard procedure; two-factor authentication (2FA) is becoming the norm. Other options include Biometric verification and security tokens, key fobs or soft tokens. In addition, users can take precautions to minimize the number of places where information appears and the number of times it is actually transmitted to complete a required transaction. Extra measures might be taken in the case of extremely sensitive documents, such as storing only on air-gapped computers, disconnected storage devices or, for highly sensitive information, in hard-copy form only.[2]

Integrity:

These measures include file permissions and user access controls. Version control may be used to prevent erroneous changes or accidental deletion by authorized users from becoming a problem. In addition, organizations must put in some means to detect any changes in data that might occur as a result of non-human-caused events such as an electromagnetic pulse (EMP) or server crash.[2]

Data might include checksums, even cryptographic checksums, for verification of integrity. Backups or redundancies must be available to restore the affected data to its correct state. Furthermore, digital signatures can be used to provide effective nonrepudiation measures, meaning evidence of logins, messages sent, electronic document viewing and sending cannot be denied.[2]

Availability:

This is best ensured by rigorously maintaining all hardware, performing hardware repairs immediately when needed and maintaining a properly functioning operating system (OS) environment that is free of software conflicts. It's also important to keep current with all necessary system upgrades. Providing adequate communication bandwidth and preventing the occurrence of bottlenecks are equally important tactics. Redundancy, failover, RAID -- even high-availability clusters -- can mitigate serious consequences when hardware issues do occur.

Fast and adaptive disaster recovery is essential for the worst-case scenarios; that capacity relies on the existence of a comprehensive DR plan. Safeguards against data loss or interruptions in connections must include unpredictable events such as natural disasters and fire. To prevent data loss from such occurrences, a backup copy may be stored in a geographically isolated location, perhaps even in a fireproof, waterproof safe. Extra security equipment or software such as firewalls and proxy servers can guard against downtime and unreachable data blocked by malicious denial-of-service (DoS) attacks and network intrusions.[14]

Best Practices for implementing CIA triad:

In implementing the CIA triad, an organization should follow a general set of best practices. Some best practices, divided by each of the three subjects, include:[14]

Confidentiality, Data should be handled based on the organization's required privacy. Data should be encrypted using 2FA.Keep access control lists and other file permissions up to date.

Integrity, ensure employees are knowledgeable about compliance and regulatory requirements to minimize human error.[14]

Use backup and recovery software. To ensure integrity, use version control, access control, security control, data logs and checksums. Availability, use preventive measures such as redundancy, failover and RAID. Ensure systems and applications stay updated. Use network or server monitoring systems. Ensure a data recovery and business continuity (BC) plan is in place in case of data loss. Encrypting Communications Cloud, disk and file encryption deal with data that is either being stored or in transit to and from a storage device. Business is more and more about exchanging data and collaborating. Sensitive data has to be communicated securely among employees, contractors, partners and customers.

A white paper published at MimeCast looks at content encryption with specific references to email. Here the issues with secure encryption are compounded by the fact that you don’t know the security status of many of the endpoints of your communications. If you send sensitive data to a consultant for analysis and your system is perfectly secured, he might be sitting in a coffee shop looking at his email on an unsecured laptop. The communication will be in plain text before it reaches the consultant and a data breach can result.[14]

Breaches caused by human error:

Here is a brief review of seven well-known breaches caused by human error.

1. Equifax — Expired certificates delayed breach detection

In the spring of 2017, the U.S. Department of Homeland Security's Computer Emergency Readiness Team (CERT) sent consumer credit reporting agency Equifax a notice about a vulnerability affecting certain versions of Apache Struts. According to former CEO Richard Smith, Equifax sent out a mass internal email about the flaw. The company’s IT security team should have used this email to fix the vulnerability, according to Smith’s testimony before the House Energy and Commerce Committee. But that didn’t happen. An automatic scan several days later also failed to identify the vulnerable version of Apache Struts. Plus, the device inspecting encrypted traffic was misconfigured because of a digital certificate that had expired ten months previously. Together, these oversights enabled a digital attacker to crack into Equifax’s system in mid-May and maintain their access until the end of July.[14]

How encryption may become a factor in scenarios like this: Once attackers have access to a network, they can install rogue or stolen certificates that allow them to hide exfiltration in encrypted traffic. Unless HTTPS inspection solutions are available and have full access to all keys and certificates, rogue certificates will remain undetected.

Impact: The bad actor is thought to have exposed the personal information of 145 million people in the United States and more than 10 million UK citizens. In September 2018, the Information Commissioner's Office issued Equifax a fine of £500,000, the maximum penalty amount allowed under the Data Protection Act 1998, for failing to protect the personal information of up to 15 million UK citizens during the data breach.[9][14]

2. Ericsson — mobile services go dark when the certificate expires

At the beginning of December 2018, a digital certificate used by Swedish multinational networking and telecommunications company Ericsson for its SGSN–MME (Serving GPRS Support Node—Mobility Management Entity) software expired. This incident caused outages for customers of various UK mobile carriers including O2, GiffGaff, and Lyca Mobile. As a result, a total of 32 million people in the United Kingdom alone lost access to 4G and SMS on 6 December. Beyond the United Kingdom, the outage reached 11 countries including Japan.

How encryption may become a factor in scenarios like this: Expired certificates do not only cause high-impact downtime; they can also leave critical systems without protection. If a security system experiences a certificate outage, cybercriminals can take advantage of the temporary lack of availability to bypass the safeguards.[9]

Impact: Ericsson restored the most affected customer services over the course of 6 December. The company also noted in a blog post that "The faulty software [for two versions of SGSN–MME] that has caused these issues is being decommissioned."[9]

3. LinkedIn—Millions miss connections when the certificate expires

On 30 November, a certificate used by business social networking giant LinkedIn for its country subdomains expired. As reported by The Register, the incident did not affect www.linkedin.com, as LinkedIn uses a separate certificate for that particular domain. But the event, which involved a certificate issued by DigiCert SHA2 Secure Server CA, did invalidate us.linkedin.com along with the social media giant's other subdomains. As a result, millions of users were unable to log into LinkedIn for several hours.[14]

How encryption may become a factor in scenarios like this: Whenever certificates expire, it may indicate that overall protection for machine identities is not up to par. Uncontrolled certificates are a prime target for cybercriminals who can use them to impersonate the company or gain illicit access.

Impact: Later in the afternoon on 30 November, LinkedIn deployed a new certificate that helped bring its subdomains back online, thereby restoring all users' access to the site.[14]

4. Strathmore College — Student records not adequately protected

In August 2018, it appears that an employee at Strathmore secondary college accidentally published more than 300 students' records on the school's intranet. These records included students' medical and mental health conditions such as Asperger's, autism and ADHD. According to The Guardian, they also listed the exposed students' medications along with any learning and behavioral difficulties. Overall, the records remained on Strathmore's intranet for about a day. During that time, students and parents could have viewed and/or downloaded the information.[13]

How encryption may become a factor in scenarios like this: Encrypting access to student records makes it difficult for anyone who doesn't have the proper credentials to access them. Any information left unprotected by encryption can be accessed by any cybercriminals who penetrate your perimeter.

Impact: Strathmore's principal said he had arranged professional development training for his staff to ensure they're following best security practices. Meanwhile, Australia's Department of Education announced that it would investigate what had caused the breach.[13]

5. Veeam — Customer records compromised by unprotected database

Near the end of August 2018, the Shodan search engine indexed an Amazon-hosted IP. Bob Diachenko, director of cyber risk research at Hacken.io, came across the IP on 5 September and quickly determined that the IP resolved to a database left unprotected by the lack of a password. The exposed database contained 200 gigabytes worth of data belonging to Veeam, a backup and data recovery company. Among that data were customer records including names, email addresses and some IP addresses.[9]

How encryption may become a factor in scenarios like this: User names and passwords are a relatively weak way of securing private access. Plus, if an organization does not maintain complete control of the private keys that govern access for internal systems, attackers have a better chance of gaining access.[9]

Impact: Within three hours of learning about the exposure, Veeam took the server offline. The company also reassured TechCrunch that it would "conduct a deeper investigation and… take appropriate actions based on our findings."[9]

6. Marine Corps — unencrypted email misfires

At the beginning of 2018, the Defense Travel System (DTS) of the United States Department of Defense (DOD) sent out an unencrypted email with an attachment to the wrong distribution list. The email, which the DTS sent within the usmc.mil official unclassified Marine domain but also to some civilian accounts, exposed the personal information of approximately 21,500 Marines, sailors and civilians. Per Marine Corp Times, the data included victims' bank account numbers, truncated Social Security Numbers and emergency contact information.[14]

How encryption may become a factor in scenarios like this: If organizations are not using proper encryption, cybercriminals can insert themselves between two email servers to intercept and read the email. Sending private personal identity information over unencrypted channels essentially becomes an open invitation to cybercriminals.[14]

Impact: Upon learning of the breach, the Marines implemented email recall procedures to limit the number of email accounts that would receive the email. They also expressed their intention to implement additional security measures going forward.[9]

7. Pennsylvania Department of Education — misassigned permissions

In February 2018, an employee in Pennsylvania's Office of Administration committed an error that subsequently affected the state's Teacher Information Management System (TIMS). As reported by PennLive, the incident temporarily enabled individuals who logged into TIMS to access personal information belonging to other users including teachers, school districts and Department of Education staff. In all, the security event is believed to have affected as many as 360,000 current and retired teachers.[13]

How encryption may become a factor in scenarios like this: If you do not know who's accessing your organization's information, then you'll never know if it's being accessed by cybercriminals. Encrypting access to vital information and carefully managing the identities of the machines that house it will help you control access.[13]

Impact: Pennsylvania's Department of Education subsequently sent out notice letters informing victims that the incident might have exposed their personal information including their Social

Security Numbers. It also offered a free one-year subscription for credit monitoring and identity protection services to affected individuals.

2.3-Hybrid Ciphers

Cryptography is the generally utilized technique for the security of data. Gronsfeld Cipher And Vigenere cipher are one of the cryptographic techniques that are viewed as least complex and most vulnerable because of numerous impediments. To conquer the impediments of Gronsfeld Cipher And Vigenere cipher we proposed an upgraded variant as Combination of Polybius cipher that is a lot of secure against Kasiski and Friedman assaults. Cryptanalysis, frequency analysis, pattern prediction and brute attack on proposed technique are also much difficult due to use of multiples tables for encryption. The modified hybrid is now a high percentage of Diffusion and Confusion in the algorithm that generates them making it a very strong cipher and difficult to break.[8]

CHAPTER 3

PROPOSED SYSTEM

The method employs use of Gronsfeld Cipher,Vegenere Cipher and Polybius Square Cipher in its encryption process.

The ciphertext will first be dealt with Gronsfeld. The resulting ciphertext will then be used for further process.

The ciphertext will then be operated on using Vegenere. A chosen key out of random will initiate the process.

At the end of the process, the resulting ciphertext then becomes a message as Input for the Polybius Square Cipher process.

This process will end up making the final ciphertext more difficult to be broken using existing cryptanalysis processes.

This process is shown in figure 3.1.

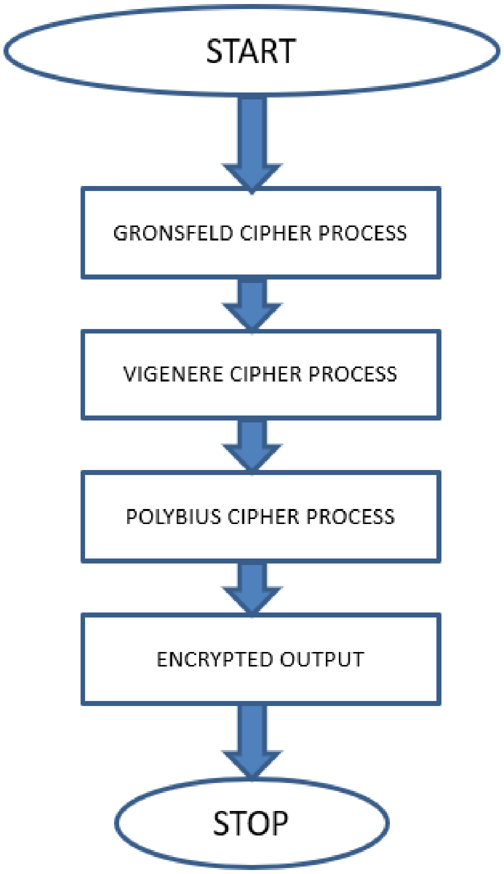


Figure 3.1 : Proposed System[7]

3.1 GRONSFELD CIPHER

Gronsfeld Cipher is a cryptography method that works like a Vigenere Cipher. Gronsfeld Cipher uses keys from decimal numbers instead of letters, but sometimes it can uses ASCII as the key substitustion. The key will be repeated periodically with the intention that each plaintext has a key. It has the same length as plaintext. When the user enters a key that is smaller in length than plaintext, the key will automatically be repeated from the beginning for the next plaintext .

There are two models of use of characters in the Gronsfeld algorithm. This algorithm can use 256 ASCII characters or use only 26 alphabetical characters.

The following equations are the formulas used to implement the Gronsfeld algorithm.

Encryption:

$$E\ (x) = (P\ (x) + K\ (x)) \bmod 26$$

$$E\ (x) = (P\ (x) + K\ (x)) \bmod 256$$

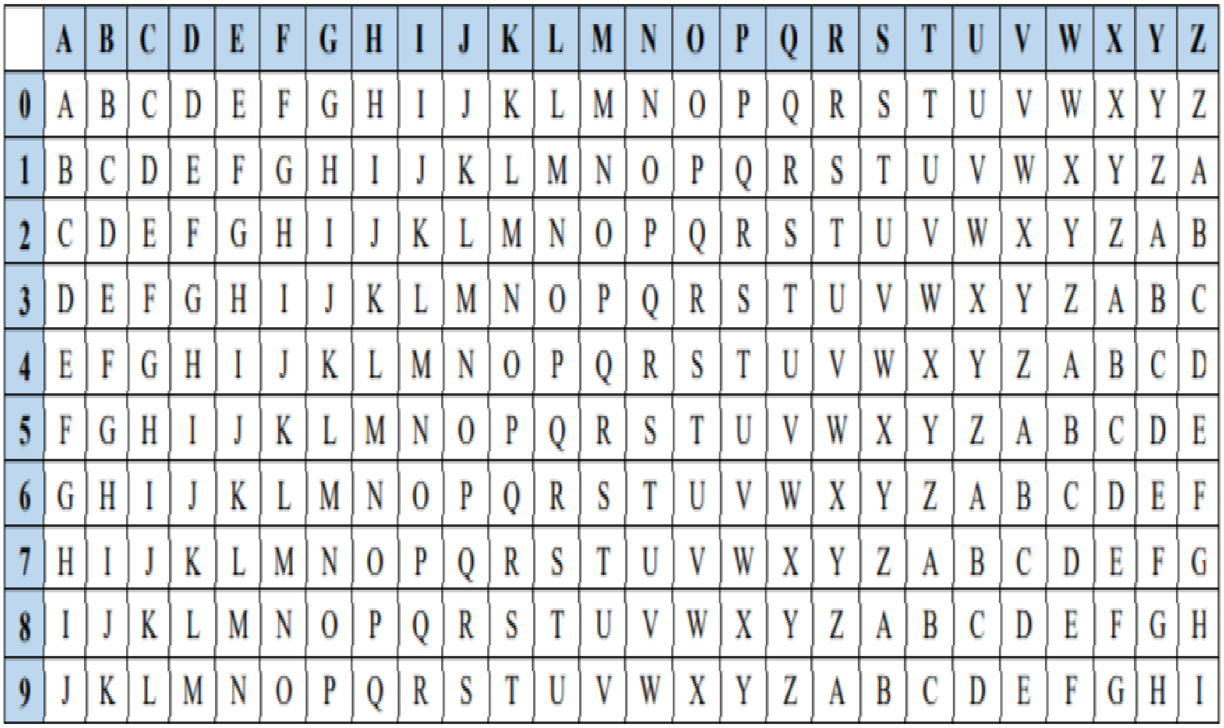
Decryption:

$$D\ (x) = (P\ (x) - K\ (x)) \bmod 26$$

$$D\ (x) = (P\ (x) - K\ (x)) \bmod 256$$

In performing the text encryption process, specify the plaintext to be encrypted and then change it in capital/uppercase form if necessary. Determine the key in the form of numbers.

If the key length is not the same as the length of the plaintext, then the key is repeated periodically so that the number of key characters is the same as the number of plaintexts. The plaintext will be changed to decimal. The ASCII code will be added with a decimal value to the key. The result of the sum if it exceeds 256 will experience a modulo process. The final result is changed back to the character form [7] as shown in Table 3.1.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
4	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I

Table 3.1 : Gronsfeld Cipher[8]

Working

This section explains the Gronsfeld algorithm testing. This algorithm is one of the classic cryptographic algorithms that use symmetrical keys. This method is straightforward to implement because this algorithm has an easy calculation. Even though the key is simple, this algorithm has a high complexity to solve if using a long key because the key loop is unpredictable in which part. The following is an illustration of the Gronsfeld algorithm calculation.

In Table 3.2 and Table 3.3, the key values and Plaintext ASCII value are given.

Plaintext = UNIVERSITY

Key=6

K1	K2	K3	K4	K5	K6
1	2	3	4	5	6

Table 3.2 : Keys [8]

Plaintext ASCII =

P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
U	N	I	V	E	R	S	I	T	Y
85	78	73	86	69	82	83	73	84	89

Table 3.3 : Input[8]

Encryption-

C1=P1+K1=85+1=86

C2=P2+K2=78+2=80

C3=P3+K3=73+3=76

C4=P4+K4=86+4=90

C5=P5+K5=69+5=74

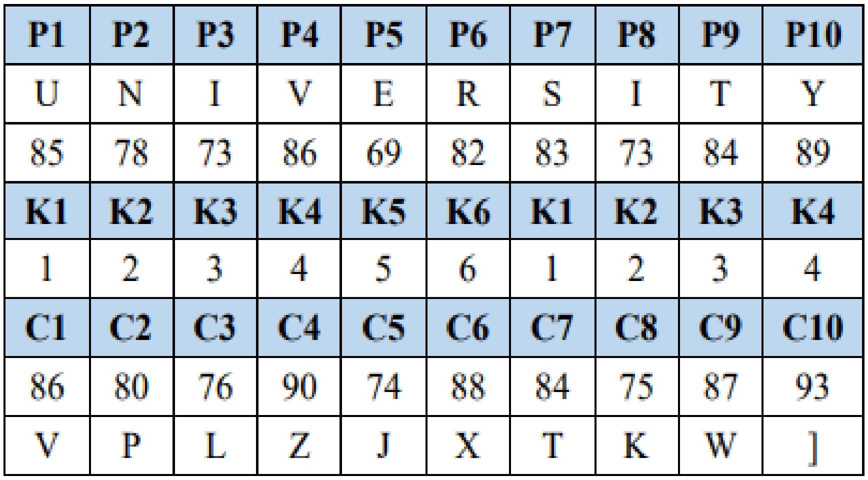
C6=P6+K6=82+6=88

C7=P7+K1=83+1=84

C8=P8+K2=73+2=75

C9=P9+K3=84+3=87

C10=P10+K4=89+4=93



P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
U	N	I	V	E	R	S	I	T	Y
85	78	73	86	69	82	83	73	84	89
K1	K2	K3	K4	K5	K6	K1	K2	K3	K4
1	2	3	4	5	6	1	2	3	4
C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
86	80	76	90	74	88	84	75	87	93
V	P	L	Z	J	X	T	K	W	]

Table 3.4 : Encryption Result[8]

Decryption-

P1=C1-K1=86-1=85

P2=C2-K2=80-2=78

P3=C3-K3=76-3=73

P4=C4-K4=90-4=86

P5=C5-K5=74-5=69

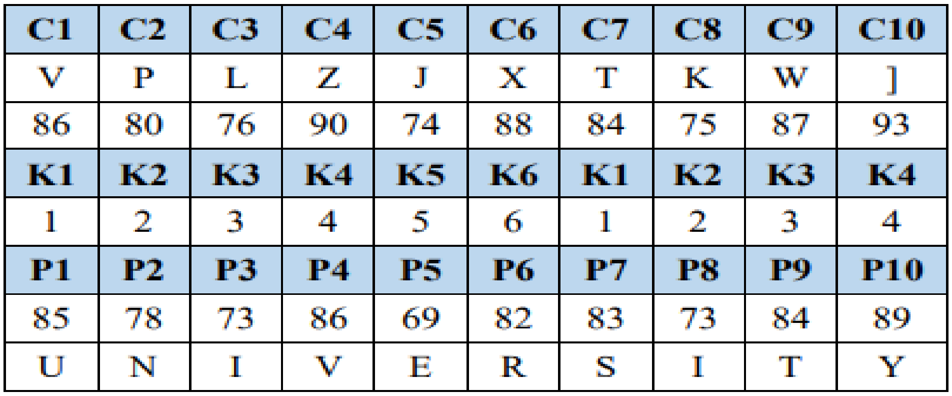
P6=C6-K6=88-6=82

P7=C7-K1=84-1=83

P8=C8-K2=75-2=73

P9=C9-K3=87-3=84

P10=C-0+K4=93-4=89



C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
V	P	L	Z	J	X	T	K	W	]
86	80	76	90	74	88	84	75	87	93
K1	K2	K3	K4	K5	K6	K1	K2	K3	K4
1	2	3	4	5	6	1	2	3	4
P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
85	78	73	86	69	82	83	73	84	89
U	N	I	V	E	R	S	I	T	Y

Table 3.5 : Decryption Result[8]

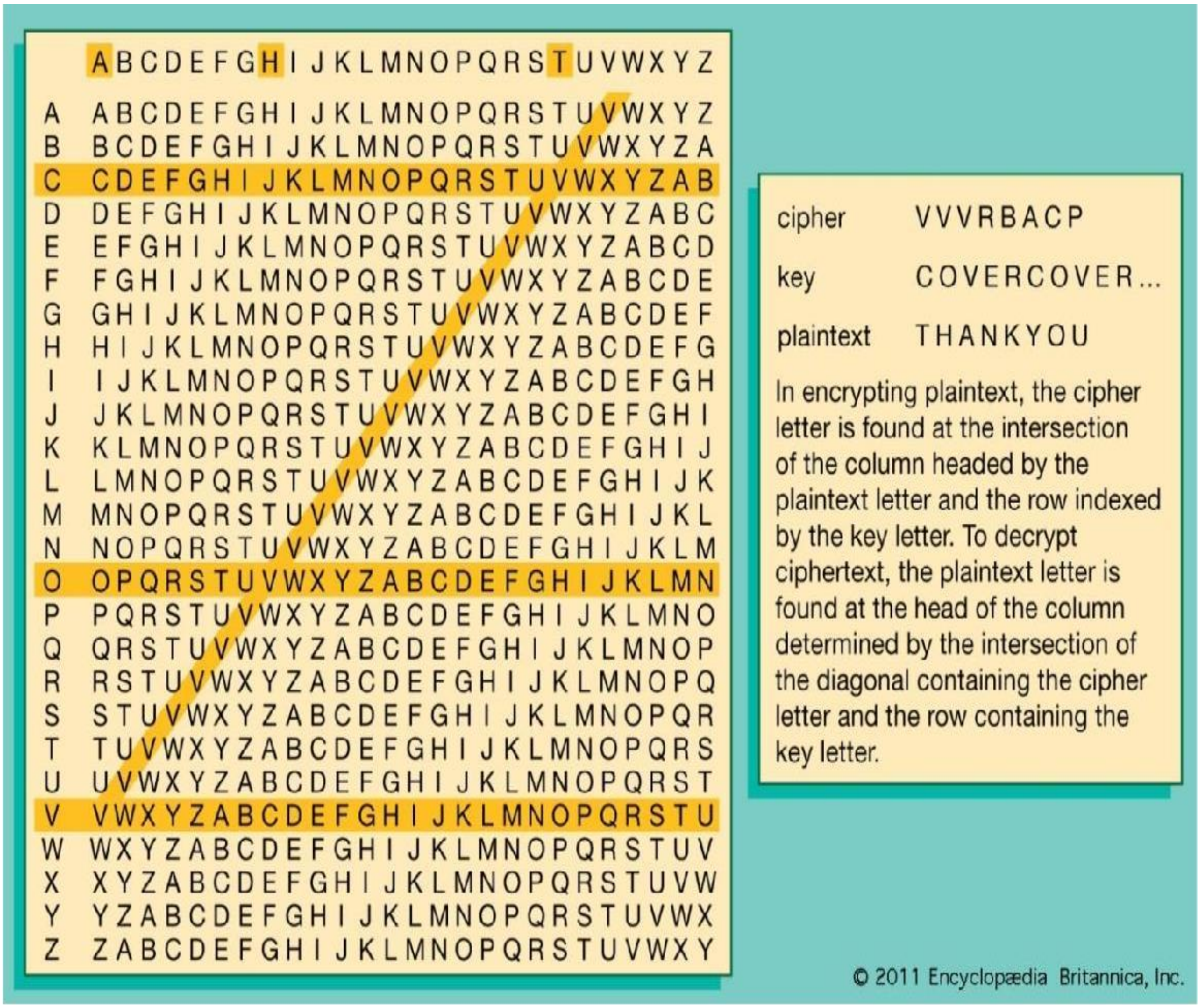
In Table 3.4 and Table 3.5, the encryption and decryption results is shown. Gronsfeld Cipher is a development of Vigenere Cipher. Gronsfeld Cipher is named after its inventor such as Johann Franz Graf Gronsfeld-Bronkhorst. He was the imperial commander in the Bavarian national revolt of 1705-1706. Gronsfeld is identical to Vigenere Cipher, but the difference is that this cipher key uses numbers while Vigenere uses the alphabet. The advantage of Gronsfeld Cipher is that the key is not a word, but the weakness of the cipher is the same as that of Vigenere such as the key can be rotated to produce plaintext. The power of Gronsfeld Cipher is on a long key. This algorithm is straightforward to implement and has a high speed.[7]

3.2 Vigenere Cipher

Vigenere Cipher is a method of encrypting alphabetic text. It uses a simple form of polyalphabetic substitution. A polyalphabetic cipher is any cipher based on substitution, using multiple substitution alphabets .The encryption of the original text is done using the Vigenère square or Vigenère table This makes the cipher less vulnerable to cryptanalysis using letter frequencies. [3]

Blaise de Vigenère developed what is now called the Vigenère cipher in 1585.

He used a table known as the Vigenère square, to encipher messages as shown in Fig 3.2.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

cipher	VVVRBACP
key	COVERCOVER...
plaintext	THANKYOU

In encrypting plaintext, the cipher letter is found at the intersection of the column headed by the plaintext letter and the row indexed by the key letter. To decrypt ciphertext, the plaintext letter is found at the head of the column determined by the intersection of the diagonal containing the cipher letter and the row containing the key letter.

© 2011 Encyclopædia Britannica, Inc.

Figure 3.2 : Vigenere Cipher[3]

Example :

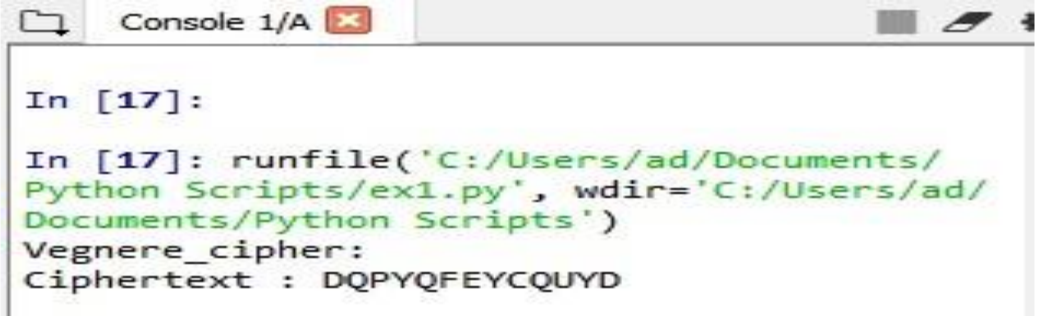
Input: Plaintext : INDIA

Key: AYUSH

Output: ILXAH

In Fig 3.3 and Fig 3.4, we perform the cipher algorithm in our system.

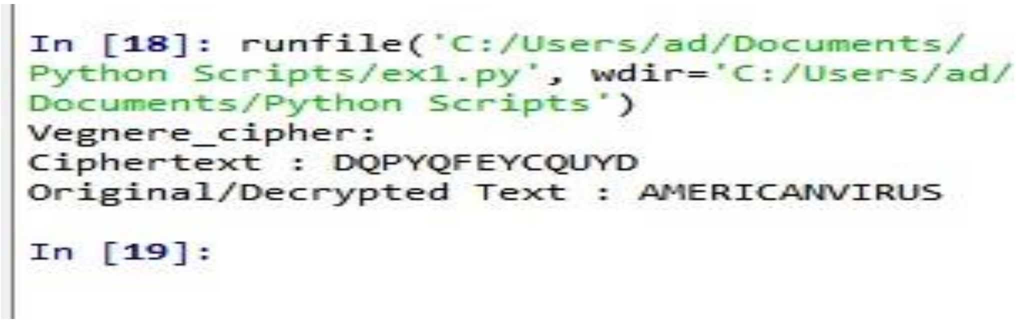
Encryption:



```
In [17]:

In [17]: runfile('C:/Users/ad/Documents/
Python Scripts/ex1.py', wdir='C:/Users/ad/
Documents/Python Scripts')
Vegnere_cipher:
Ciphertext  :  DQPYQFEYCQUYD
```

Figure 3.3 : Vigenere Cipher Encryption

Decryption:

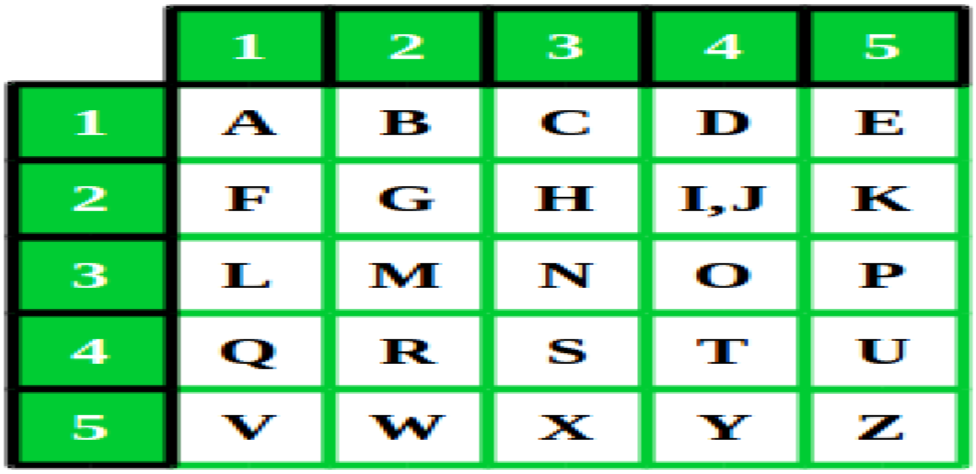


```
In [18]: runfile('C:/Users/ad/Documents/
Python Scripts/ex1.py', wdir='C:/Users/ad/
Documents/Python Scripts')
Vegnere_cipher:
Ciphertext  :  DQPYQFEYCQUYD
Original/Decrypted Text  :  AMERICANVIRUS

In [19]:
```

Figure 3.4 : Vigenere Cipher Decryption

3.3-Polybius Square Cipher

A Polybius Square is a table that allows someone to convert letters into numbers. To make the encryption little harder, this table can be randomized and shared with the recipient. In order to fit the 26 letters of the alphabet into the 25 cells created by the table, the letters ‘i’ and ‘j’ are usually combined into a single cell. Originally there was no such problem because the ancient greek alphabet has 24 letters. A table of bigger size could be used if a language contain large number of alphabets.[4]



	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I,J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Table 3.9 : Polybius Cipher[4]

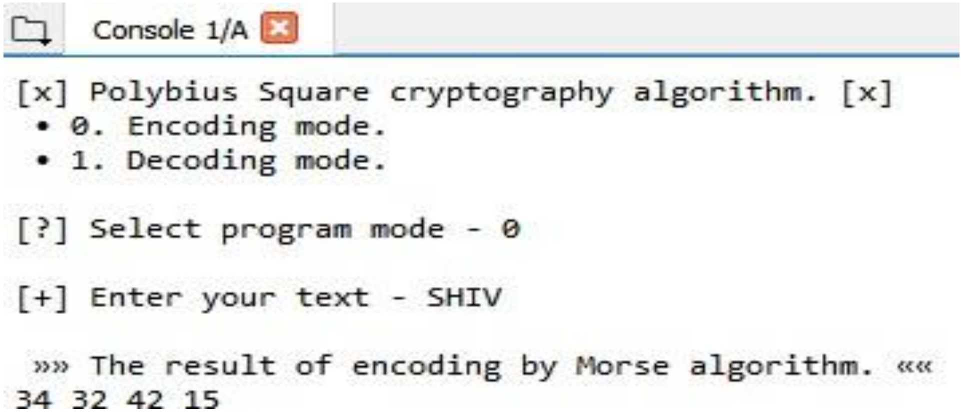
Example**:**

Input: BUS

Output: 124543

In Fig 3.5, we perform the Polybius cipher algorithm.

Encryption/Decryption:



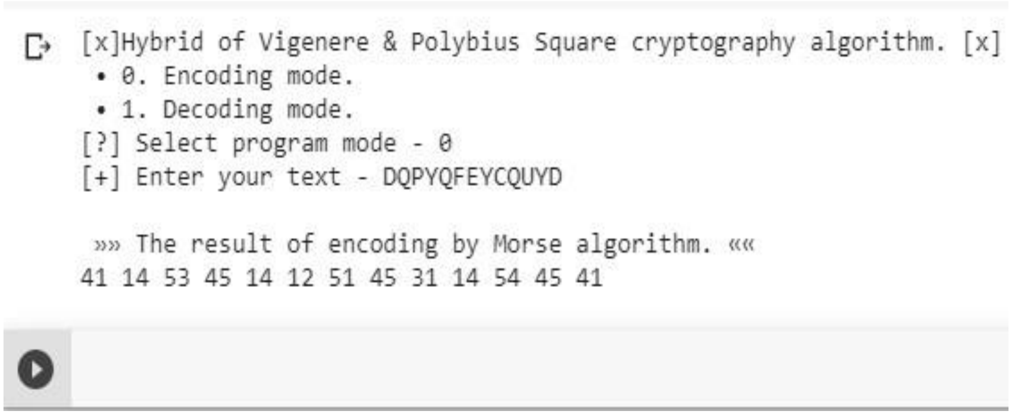
```
[x] Polybius Square cryptography algorithm. [x]
 • 0. Encoding mode.
 • 1. Decoding mode.

[?] Select program mode - 0

[+] Enter your text - SHIV

 »» The result of encoding by Morse algorithm. ««
34 32 42 15
```

Figure 3.5: Polybius Cipher Encryption/Decryption

Now, Gronsfeld, Vigenere and Polybius cipher will be done summation and combination for formation of hybrid cipher system. This combination makes use of alphabetic substitution that is gronsfeld cipher, vigenere Cipher and polyalphabetic Numerical Cipher that is Polybius Square

Cipher which make the message and plain text to encrypted message which is very confusing, unstructured and diffused that cannot be easier to break.

In Fig 3.6, we perform the Hybrid algorithm for cipher.



```
[x]Hybrid of Vigenere & Polybius Square cryptography algorithm. [x]
 • 0. Encoding mode.
 • 1. Decoding mode.
[?] Select program mode - 0
[+] Enter your text - DQPYQFEYCQUYD

 »» The result of encoding by Morse algorithm. ««
41 14 53 45 14 12 51 45 31 14 54 45 41
```

Figure 3.6 :Hybrid Model

CHAPTER 4

METHODOLOGY

Cryptography is the generally utilized technique for the security of data. Gronsfeld Cipher And Vigenere cipher are one of the cryptographic techniques that are viewed as least complex and most vulnerable because of numerous impediments. To conquer the impediments of Gronsfeld Cipher And Vigenere cipher we proposed an upgraded variant as Combination of Polybius cipher that is a lot of secure against Kasiski and Friedman assaults. Cryptanalysis, frequency analysis, pattern prediction and brute attack on proposed technique are also much difficult due to use of multiples tables for encryption. The modified hybrid is now a high percentage of Diffusion and Confusion in the algorithm that generates them making it a very strong cipher and difficult to break.

The message as plaintext and Key is send through sender in three phase for execution and working of System as in first phase it will proceed through Gronsfeld Cipher in the first phase and the resulting cipher text is forwarded to Vigenere cipher as input. The Vigenere Cipher further on works on the cipher text and then the new instructed and disputed encrypted cipher comes and then in third phase it became the input of Polybius cipher which result as output as Numerical encrypted Cipher that is confusing and scrambled mix numerical.

This Output from Polybius at last phase is numerical and the Input that proceed in first phase was alphabetic letters this all confuses and doesn’t allow the intruders, detectors, thefts, hackers and cyber crime to commit any assaults and attacks on system and doesn’t allow them to steal Information.

A python programming is written and executed for the working of System. Google Colab as Online and Sypder IDE on Independent System are taken for execution of process.

In Fig. 4.1, the flow of the algorithm is shown.

Flowchart of Hybrid Algorithm-

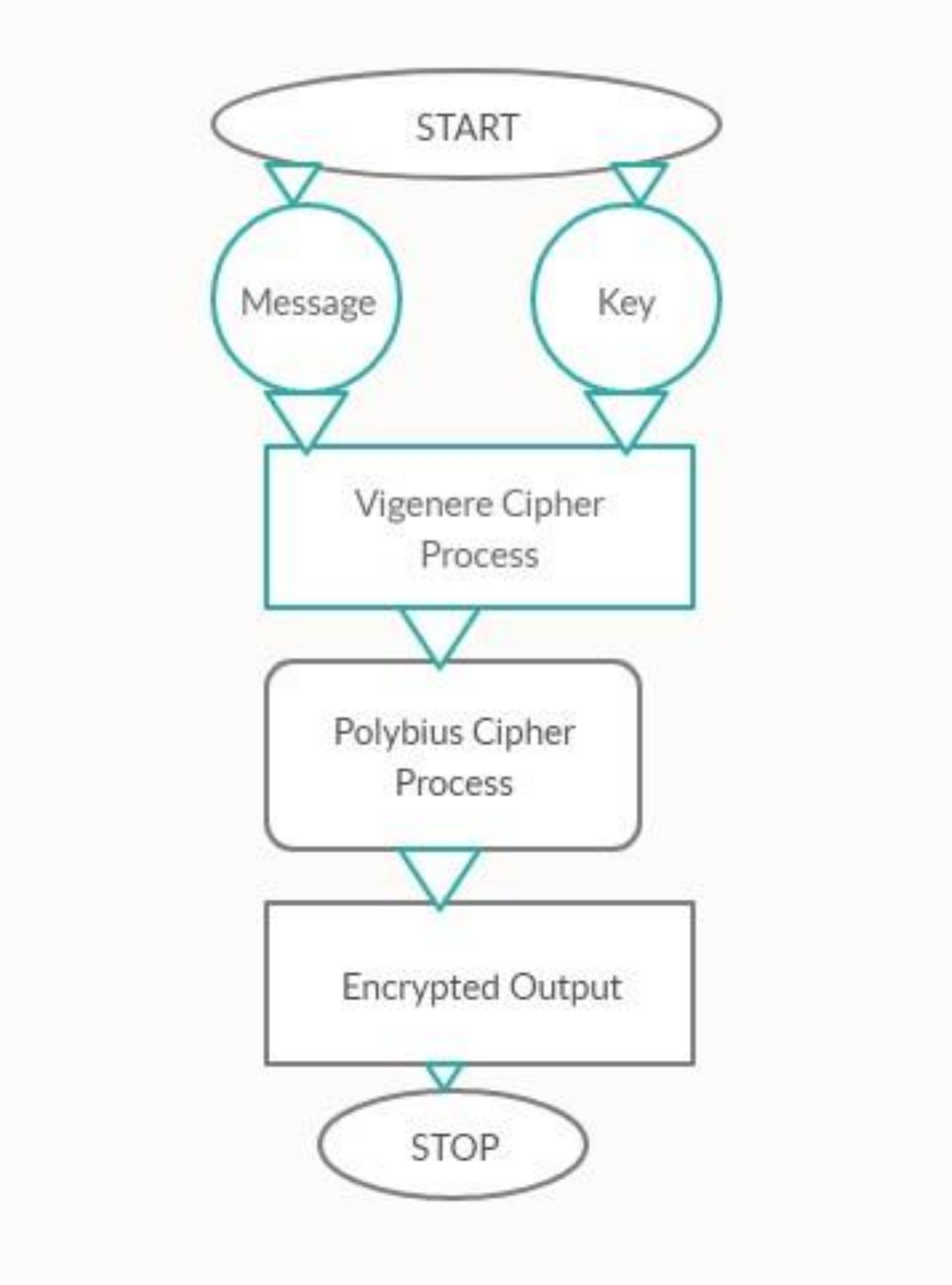


Figure 4.1: Flowchart of Hybrid Algorithm

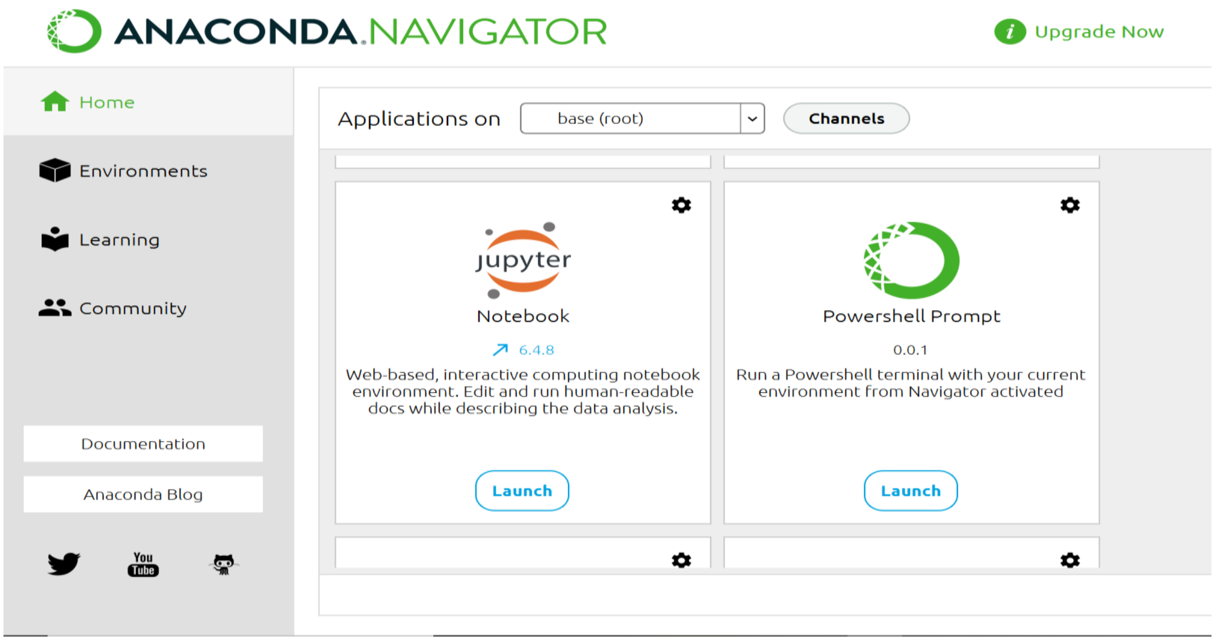
4.1 TECHNOLOGIES USED

Jupyter Notebook:

Jupyter notebooks are an open source web-based application that allow users to create and share documents containing live code, equations, visualizations and narrative text. It is a powerful tool for data scientists and developers as it allows them to quickly and easily develop and share their work. Jupiter notebooks are used for data cleaning, data analysis, and machine learning. They provide an easy-to-use platform for data scientists to interact with their datasets, visualize the results and document the process. Jupiter notebooks are used in many industries and can help companies improve their data analysis capabilities.

The Jupyter Notebook is an open source web application that we can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter. In Fig 4.2, the process to open jupyter notebook using Anaconda Navigator.

Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows us to write programs in Python, but there are currently over 100 other kernels that we can also use.



Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Run a Powershell terminal with your current environment from Navigator activated

Figure 4.2 : Jupyter notebook launch

Jupyter Notebooks Requirements:

- recommendations are based on user count.
- With PySpark (Team Studio version 6.2 and later)
- Memory and disk space required per user: 1GB RAM + 1GB of disk + .5 CPU core.
- Server overhead: 2-4GB or 10% system overhead (whatever is larger), .5 CPU cores, 10GB disk space.
- Port requirements: Port 8000 plus 5 unique, random ports per notebook.
- Without PySpark (Team Studio version 6.0 or 6.1)
- Memory and disk space required per user: 512MB RAM + 1GB of disk + .5 CPU core.
- Server overhead: 2-4GB or 10% system overhead (whatever is larger), .5 CPU cores, 10GB disk space.
- Port requirements: Port 8000.

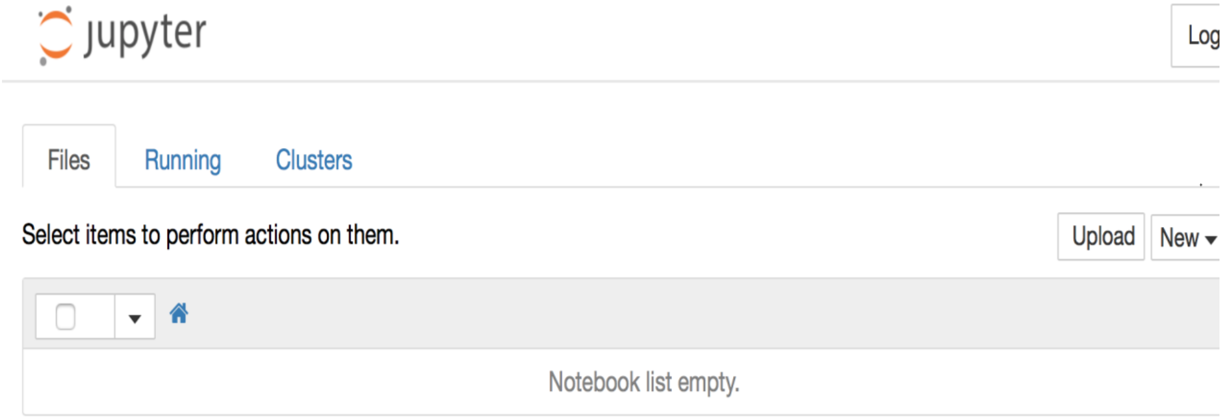
Installation:

we can use a handy tool that comes with Python called pip to install Jupyter Notebook

```
$ pip install jupyter
```

Starting the Jupyter Notebook Server

we need to open up your terminal application and go to a folder of your choice and run the following as shown in Fig 4.3.



Select items to perform actions on them.

Notebook list empty.

Figure 4.3 : Jupyter notebook server from our system

Creating a Notebook:

Click on the New button. It will open up a list of choices. Choose Python 3.

In fig 4.4, we are creating the jupiter notebook.

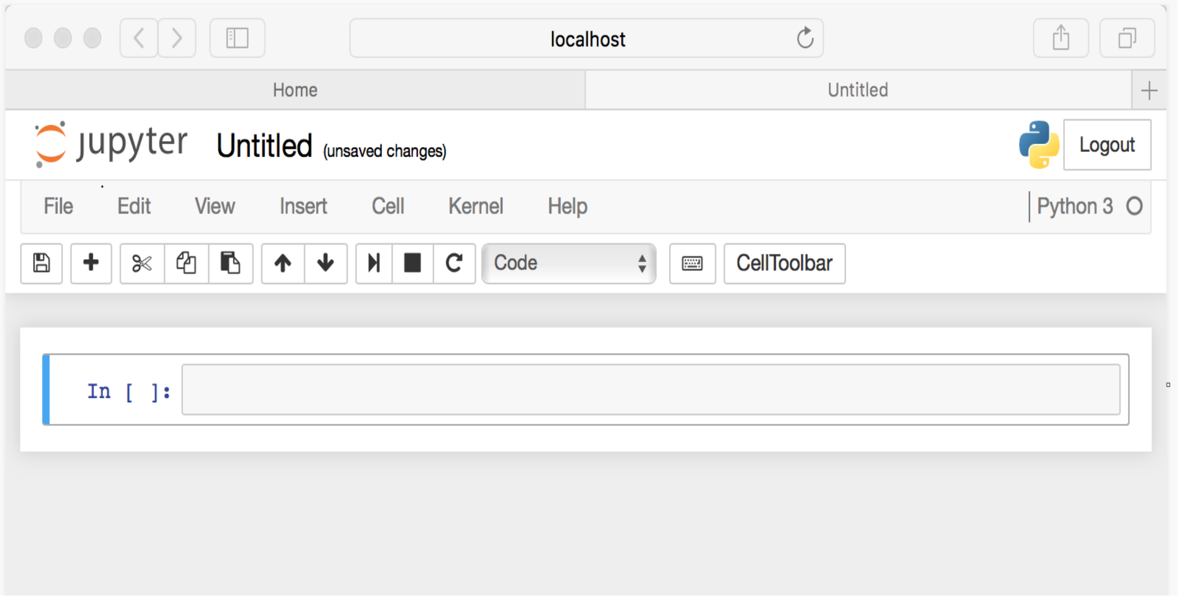


Figure 4.4 : Creating a notebook in our system

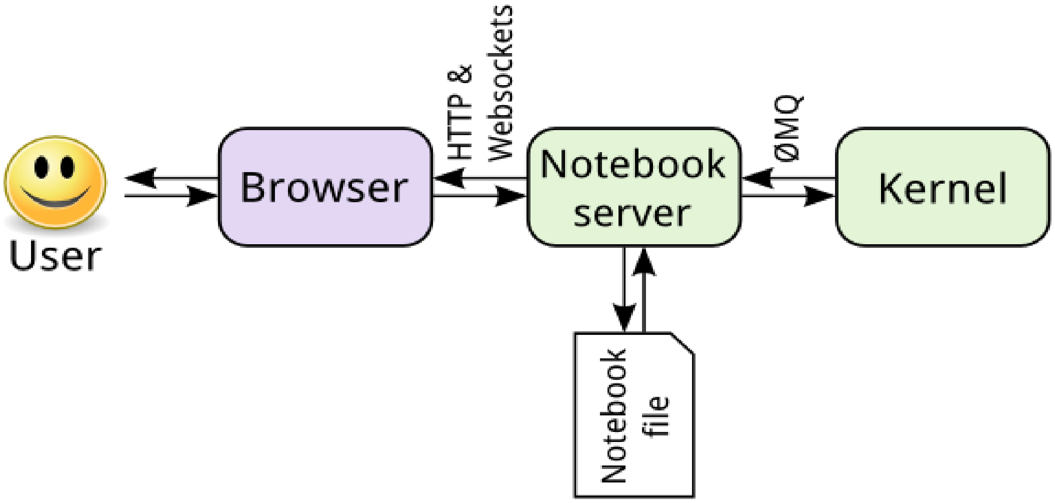


Figure 4.5 : Jupyter Notebook Structure

In Fig 4.5, the structure for the working of the notebook from the user end using browser is shown.

4.2-HYBRID ALGORITHM

Using an algorithm that combines symmetric encryption with an asymmetric encryption algorithm can provide better security than either algorithm used alone. The result of such a combination may also be faster execution times, as the encryption and decryption processes can be distributed among multiple algorithms. Additionally, hybrid algorithms are often more resilient to attack, as different algorithms may be used to protect different parts of the data. .In terms of execution, hybrid cryptographic algorithms can be computationally intensive and require specialized hardware to optimize their performance. Additionally, the amount of time it takes to complete the encryption and decryption processes may vary depending on the algorithms used and the size of the data being encrypted. For example, the execution time for a hybrid algorithm that combines symmetric encryption with an asymmetric encryption algorithm may be significantly longer than for either algorithm used alone. However, the added security provided by the combination of these algorithms may outweigh the additional execution time. Overall, hybrid cryptographic algorithms are an important tool for ensuring the security of data in a variety of contexts. By combining multiple algorithms, hybrid cryptographic algorithms can provide enhanced security while also providing faster execution times and greater resilience to attack.

In terms of execution, hybrid cryptographic algorithms can be computationally intensive and require specialized hardware to optimize their performance. Additionally, the amount of time it takes to complete the encryption and decryption processes may vary depending on the algorithms used and the size of the data being encrypted. For example, the execution time for a hybrid algorithm that combines symmetric encryption with an asymmetric encryption algorithm may be significantly longer than for either algorithm used alone. However, the added security provided by the combination of these algorithms may outweigh the additional execution time.

CHAPTER 5

ANALYSIS AND IMPLEMENTATION

5.1-ANALYSIS

Hybrid Algorithm is a convenient technique for the encryption of the transmission of data. The biggest challenge in communication devices is the level of security. The main goal of the different algorithms in cryptographic techniques is to enhance the capability of security to communication devices. The lack of basic security awareness among phone users and the general lack of understanding of Bluetooth technology is certainly an advantage for hackers.

After the problem analysis, we discovered that we can design this hybrid algorithm using the Vigenere cipher, Polybius cipher, and Gronsfed cipher. Information security in the network has been a challenge, which demands urgent attention.

Notably, with the rapid development of computer technology, several issues arose at the surface of the Information Security field such as User Authentication, data encryption, data integrity, and access control. For example, Bluetooth is a short-range radio communication standard, that enables electronic devices to be connected and communicated wirelessly.

The original message remains safe as long as the encryption key that is being used remains secret. Even if the data sent using the hybrid encryption algorithm is tracked, the complex message is organized in such a way that the tracker will not understand which part of the complex message contains the AES encrypted key and the ciphertext.

Also, the private key of the receiver will not be known and hence the AES key cannot be decrypted ensuring the data in transit remains safe. This algorithm will add the quality of extra filter of security in the communication devices.

Security analysis for communication devices should focus on areas such as authentication, encryption, and software integrity. Authentication is key to ensuring that only authorized users can access the device. Encryption should be used to protect the data in transit, and software integrity should be ensured to protect against malicious code. Additional measures such as two-factor authentication and secure remote access should also be considered. Finally, security policies should be established and enforced to ensure that users are aware of their responsibilities when using the device. Security analysis for communication devices is the process of evaluating the security of a device or system that is used to communicate. This evaluation includes assessing the device's ability to protect itself from malicious attack and to protect the data that is sent and received over the device. It also includes assessing the device's ability to detect, respond to, and recover from security incidents. Security analysis can help identify any weaknesses in the device or system that could be exploited by attackers. This analysis is typically conducted by an experienced security expert. Cryptography is the practice of using cryptography to protect data from unauthorized access or manipulation. Cryptographic techniques are used to ensure the privacy, integrity and authenticity of data transmitted over a network.

5.2-IMPLEMENTATION

The implementation of hybrid algorithms using different ciphers can be done in the following steps:

1. Create a new project: Create a new project in Jupyter Notebook and name it appropriately.

2. The authentication and authorization part can be implemented using Notebook Authentication. Firebase Authentication is an authentication service provided by Google that allows users to sign in using their email address or phone number. Algorithms Authentication also provides various methods of authentication such as OAuth, SMS, and social media providers. The user can be authenticated and authorized using Firebase Authentication.

4. Add data validation: Add validation to the form fields such as checking for valid ciphers as we are using Polybius cipher.

7. Test the algorithm: Test the application to make sure everything is working as expected.

8. Deploy the algorithm: Deploy the algorithm to the GitHub and cryptographic algorithms listings.

To make the proposed hybrid algorithms more secure and anonymous, the following measures can be taken:

1. Implement two-factor authentication: Two-factor authentication is a security measure that requires a user to provide two pieces of evidence to gain access to an account. This can be done by using a mobile phone number or email address for authentication.

2. Encrypt data: Encrypt the data that is being transferred between the client and the server.

3. Use a secure connection: Use a secure SSL connection for all communication between the client and the server.

4. Implement a privacy policy: Implement a privacy policy that clearly outlines how user data is used, stored, and shared.

5. Periodically review security measures: Periodically review the security measures to make sure they are up-to-date.

5.3-IMPLEMENTATION OF ALGORITHM WITH CODE

A hybrid algorithm of cryptographic techniques combines multiple techniques to create a more secure encryption scheme. Examples of techniques that can be combined include public-key encryption, symmetric encryption, hashing algorithms, and digital signature algorithms. The combination of techniques creates a more secure system by providing multiple layers of security and reducing the chance of a successful attack. Additionally, hybrid algorithms provide flexibility by allowing the use of different techniques and strengths of encryption depending on the application. Some considerations for creating a hybrid algorithm include selecting the appropriate techniques for the application, combining techniques in a secure manner, and evaluating the overall strength of the resulting algorithm.

Generate key: Using vigenere cipher we create a function that generates the key in a cyclic manner until its length isn't equal to the length of the original text.

```
# Python code to implement Vigenere Cipher
# This function generates the key in a cyclic manner until
# its length isn't equal to the length of the original text
def generateKey(string, key):
    key = list(key)
    if len(string) == len(key):
        return (key)
    else:
        for i in range(len(string) - len(key)):
            key.append(key[i % len(key)])
        return ("".join(key))
```

PyCharm Community edition supports Jupyter notebooks in read-only mode, to get full support for local notebooks download and try PyCharm Professional now!

Try DataSpell — a dedicated IDE for data science, with full support for local and remote notebooks.

Try Datalore — an online environment for Jupyter notebooks in the browser.

Encryption: Alice transmits her public key (n, e) to Bob and keeps the private key secret. Bob then wishes to send message M to Alice. He then computes the cipher text c corresponding to Bob and then transmits c to Alice.

The cipher and mode that is used are randomly selected among the ciphers that are common between the two servers. Make sure that all servers and client computers that participate in encrypted communication have ciphers and modes in common. Encryption is more secure if you include more ciphers and modes that the database server can switch between.

```
# This function returns the
# encrypted text generated # with the help of the key
def cipherText(string, key):
    cipher_text = []



for i in range(len(string)):
    x = (ord(string[i]) + ord(key[i])) % 26
    x += ord('A')
    cipher_text.append(chr(x))
    return ("".join(cipher_text))
```

Decryption: This function here uses the decryption of the encrypted text and returns the original text. The receiver system decrypts the received encrypted data by applying the private key and session key.

To decrypt the ciphertext, apply the receiver's session key on this ciphertext data.

The receiver system decrypts the received encrypted data by applying the private key and session key. To decrypt the ciphertext, apply the receiver's session key on this ciphertext data.

```
 # This function decrypts the encrypted text and returns the original text
def originalText(cipher_text, key):
    orig_text = []

for i in range(len(cipher_text)):
    x = (ord(cipher_text[i]) - ord(key[i]) + 26) % 26
    x += ord('A')
    orig_text.append(chr(x))
    return ("".join(orig_text))
```

Driver Code: main driver function is described below in which we pass two parameters. The first one is the string and the second is the keyword. To test the hybrid algorithm input is given below. Therefore the results are discussed in the further results column.

```
# Driver code
if name == " main ":
    string  = "FILE IS ENCRYPTION"
    keyword = "ENCRYPTED"
key = generateKey(string, keyword)
```

```
cipher_text = cipherText(string, key)
print("Ciphertext :", cipher_text)
print("Original/Decrypted Text :",
    originalText(cipher_text, key))
```

Implementation Code in Python:

```
# Python code to implement
# Vigenere Cipher

# This function generates the
# key in a cyclic manner until
# it's length isn't equal to
# the length of original text
def generateKey(string, key):
  key = list(key)
  if len(string) == len(key):
    return(key)
  else:
    for i in range(len(string) -
          len(key)):
      key.append(key[i % len(key)])
  return("" . join(key))

# This function returns the
# encrypted text generated
# with the help of the key
def cipherText(string, key):
  cipher_text = []
  for i in range(len(string)):
    x = (ord(string[i]) +
      ord(key[i])) % 26
    x += ord('A')
```

```
        cipher_text.append(chr(x))
    return("" . join(cipher_text))
```

```
# This function decrypts the
# encrypted text and returns
# the original text
def originalText(cipher_text, key):
    orig_text = []
    for i in range(len(cipher_text)):
        x = (ord(cipher_text[i]) -
            ord(key[i]) + 26) % 26
        x += ord('A')
        orig_text.append(chr(x))
    return("" . join(orig_text))

# Driver code
if __name__ == "__main__":
    string = "AMERICAN VIRUS"
    keyword = "DELHI"
    key = generateKey(string, keyword)
    cipher_text = cipherText(string,key)
    print("Ciphertext :", cipher_text)
    print("Original/Decrypted Text :",
        originalText(cipher_text, key))
```

```
Ciphertext : DQPYQFEYADLVFZ
Original/Decrypted Text : AMERICANTVIRUS
```

```
from string import ascii_uppercase as alphabet
def codes_table(char):
    table = {
        "A": 11, "B": 21, "C": 31, "D": 41, "E": 51,
        "F": 12, "G": 22, "H": 32, "I": 42, "K": 52,
        "L": 13, "M": 23, "N": 33, "O": 43, "P": 53,
        "Q": 14, "R": 24, "S": 34, "T": 44, "U": 54,
        "V": 15, "W": 25, "X": 35, "Y": 45, "Z": 55, "J": 0,

        11: "A", 21: "B", 31: "C", 41: "D", 51: "E",
        12: "F", 22: "G", 32: "H", 42: "I", 52: "K",
        13: "L", 23: "M", 33: "N", 43: "O", 53: "P",
        14: "Q", 24: "R", 34: "S", 44: "T", 54: "U",
        15: "V", 25: "W", 35: "X", 45: "Y", 55: "Z", 0: "J"
    }
    return table[char]
def encoding(text):
    text, finished_text = text.upper(), ""
    for symbol in text:
        if symbol in alphabet:
            finished_text += str(codes_table(symbol)) + " "
    return finished_text
def decoding(text):
    text, finished_text = text.upper(), ""
    for symbol in list(map(int, text.split())):
        finished_text += codes_table(symbol)
    return finished_text
```

```
def assembly(mode):
    text = str(input("[+] Enter your text - "))

    if mode == 0:
        finished_text = encoding(text)
    else:
        finished_text = decoding(text)

    print("\n »» The result of encoding . ««")
    print(finished_text)



def main():
    print("[x]Hybrid of Vigenere & Polybius Square cryptography algorithm. [x]")
    print(" • 0. Encoding mode.\n • 1. Decoding mode.")

    mode = int(input("[?] Select program mode - "))
    assembly(mode)



if __name__ == '__main__':
    main()
```

```
[x]Hybrid of Vigenere & Polybius Square cryptography algorithm. [x]
 • 0. Encoding mode.
 • 1. Decoding mode.
[?] Select program mode - 0
[+] Enter your text - DQPYQFEYADLVFZ

 »» The result of encoding . ««
41 14 53 45 14 12 51 45 11 41 13 15 12 55
```

CHAPTER 6

RESULT

6.1-FINAL RESULT

The output will be Encrypted text as Cipher text will be generated from the system. This two combinations of cipher programs will be executed back to back to get cipher text. It can be implemented on any System, IDE, Interpreter, and Compiler or on Cloud System such as Jupyter, Anaconda, Google collaboratory, etc.This Output from Polybius at last phase is numerical and the Input that proceed in first phase was alphabetic letters this all confuses and doesn't allow the intruders, detectors, thefts, hackers and cyber crime to commit any assaults and attacks on system and doesn't allow them to steal Information. Hybrid cryptographic algorithms are algorithms that incorporate elements of both symmetric and asymmetric cryptography. The most common example of a hybrid algorithm is Diffie-Hellman key exchange, which uses a combination of public-key and secret-key cryptography. The result of a hybrid cryptographic algorithm is a shared secret key that can be used to encrypt and decrypt messages. Additionally, because the key is generated and exchanged using both public-key and secret-key cryptography, it is less vulnerable to attack than either type of cryptography alone. Hybrid cryptographic algorithms are algorithms that combine two or more separate cryptographic algorithms for enhanced security. . The result of such a combination may also be faster execution times, as the encryption and decryption processes can be distributed among multiple algorithms. Additionally, hybrid algorithms are often more resilient to attack, as different algorithms may be used to protect different parts of the data. Hybrid cryptographic algorithms are algorithms that combine two or more separate cryptographic algorithms for enhanced security. The results of the execution of such algorithms may vary depending on the algorithms used, but in general hybrid cryptographic algorithms are designed to increase the security of the data being encrypted or decrypted. For example, using an algorithm that combines symmetric encryption with an asymmetric encryption algorithm can provide better security than either algorithm used alone. The result of such a combination may also be faster execution times, as the encryption and decryption processes can be distributed among multiple algorithms. Additionally, hybrid algorithms are often more resilient to attack, as different algorithms may be used to protect different parts of the data.

Overall, hybrid cryptographic algorithms are an important tool for ensuring the security of data in a variety of contexts. By combining multiple algorithms, hybrid cryptographic algorithms can provide enhanced security while also providing faster execution times and greater resilience to attack.

The results of the execution of such algorithms may vary depending on the algorithms used, but in general hybrid cryptographic algorithms are designed to increase the security of the data being encrypted or decrypted.

6.2-DISCUSSION

Using an algorithm that combines symmetric encryption with an asymmetric encryption algorithm can provide better security than either algorithm used alone. The result of such a combination may also be faster execution times, as the encryption and decryption processes can be distributed among multiple algorithms. Additionally, hybrid algorithms are often more resilient to attack, as different algorithms may be used to protect different parts of the data. .In terms of execution, hybrid cryptographic algorithms can be computationally intensive and require specialized hardware to optimize their performance. Additionally, the amount of time it takes to complete the encryption and decryption processes may vary depending on the algorithms used and the size of the data being encrypted. For example, the execution time for a hybrid algorithm that combines symmetric encryption with an asymmetric encryption algorithm may be significantly longer than for either algorithm used alone. However, the added security provided by the combination of these algorithms may outweigh the additional execution time. Overall, hybrid cryptographic algorithms are an important tool for ensuring the security of data in a variety of contexts. By combining multiple algorithms, hybrid cryptographic algorithms can provide enhanced security while also providing faster execution times and greater resilience to attack.

In terms of execution, hybrid cryptographic algorithms can be computationally intensive and require specialized hardware to optimize their performance. Additionally, the amount of time it takes to complete the encryption and decryption processes may vary depending on the algorithms used and the size of the data being encrypted. For example, the execution time for a hybrid algorithm that combines symmetric encryption with an asymmetric encryption algorithm may be significantly longer than for either algorithm used alone. However, the added security provided by the combination of these algorithms may outweigh the additional execution time.

CHAPTER 7

CONCLUSION & FUTURE PROSPECTIVE

7.1-CONCLUSION

Cryptography is the generally utilized technique for the security of data. Vigenere cipher is one of the cryptographic techniques that are viewed as least complex and most vulnerable because of numerous impediments. To conquer the impediments of Vigenere cipher we proposed an upgraded variant as Combination of Polybius cipher that is a lot of secure against Kasiski and Friedman assaults. Cryptanalysis, frequency analysis, pattern prediction and brute attack on proposed technique are also much difficult due to use of multiples tables for encryption. The modified hybrid of both thE Caesar Cipher and Vigenere Cipher, there is now a high percentage of Diffusion and Confusion in the algorithm that generates them making it a very strong cipher and difficult to break. In spite of the fact that there are numerous cryptographic strategies yet this space still requires genuine consideration of research network for the improvement of data security. In future our point is to give approval of proposed approach by performing security and performance analysis.

The conclusion of hybrid cryptographic algorithms execution is that they can provide a secure communication channel with enhanced security features. Hybrid cryptographic algorithms are able to combine the strengths of two or more algorithms to provide improved security as compared to using a single algorithm. By combining the strengths of multiple algorithms, hybrid algorithms can offer a more robust security solution for securing sensitive data. They are also able to reduce the amount of computing resources required, which can lead to improved performanceOverall, hybrid cryptographic algorithms are an important tool for ensuring the security of data in a variety of contexts. By combining multiple algorithms, hybrid cryptographic algorithms can provide enhanced security while also providing faster execution times and greater resilience to attack.

The conclusion of hybrid cryptographic algorithms execution is that they can provide a secure communication channel with enhanced security features. Hybrid cryptographic algorithms are able to combine the strengths of two or more algorithms to provide improved security as compared to using a single algorithm. By combining the strengths

of multiple algorithms, hybrid algorithms can offer a more robust security solution for securing sensitive data. They are also able to reduce the amount of computing resources required, which can lead to improved performance.

7.2-FUTURE PROSPECTIVE

The future of hybrid proposed cryptographic algorithms looks promising. Hybrid cryptography combines the best aspects of various types of cryptography, such as symmetric, public-key and hash functions, to create a more secure and efficient encryption system. As technology continues to advance, hybrid cryptography will become more secure and efficient. It will also become increasingly popular as organizations and individuals seek more secure ways to protect their data. Additionally, hybrid cryptography will become an increasingly important tool for protecting data in the Internet of Things (IoT) era, where data is being transmitted and stored in a variety of different devices and systems. Finally, hybrid cryptography is expected to become an important part of quantum computing, since it can be used to protect data from quantum attacks.

The future prospects of hybrid proposed cryptographic algorithms are very promising. Hybrid algorithms are becoming increasingly popular as they offer a higher level of security than traditional algorithms. These algorithms are more resilient to attacks and offer better protection of sensitive data. As new technologies emerge, hybrid algorithms will become more advanced and secure, giving users more protection and control over their data. Additionally, hybrid algorithms can be used to create new protocols and applications, such as blockchain and cryptocurrency, which have the potential to revolutionize the way we interact with digital data. With the increasing demand for secure and efficient digital systems, the development of hybrid algorithms will only continue to grow.

REFERENCES

[1] Chaudhari, Swapnil. (2018). A Research Paper on New Hybrid Cryptography Algorithm.

[2] Jakimoski, Kire, "Security Techniques for Data Protection in Cloud Computing." International Journal of Grid and Distributed Computing 9.1 (2016): 49-56.

[3] https://en.wikipedia.org/wiki/Vigenere_cipher

[4] https://en.wikipedia.org/wiki/Polybius_square

[5] Puneet Kumar, Shashi B. Rana, Development of modified AES algorithm for data security, Optik - International Journal for Light and Electron Optics, Volume 127, Issue 4, 2016, Pages 2341-2345, ISSN 0030-4026, http://dx.doi.org/10.1016/j.ijleo.2015.11.188.

(http://www.sciencedirect.com/science/article/pii/S0030402615018215)

[6] Encryption. Wellesley college Computer Science Department lecture note retrieved from : http://cs110.wellesley.edu/lectures/L18-encryption/.

[7] Classical cipher, Transposition ciphers: rom http://en.wikipedia.org/wiki/Classical_cipher

[8]Transposition ciphers, columnar transposition, Gronsfeld cipher.

[9] C. Sanchez-Avila and R. Sanchez-Reillo, "The Rijndael block cipher (AES proposal): a comparison with DES," in Security Technology, 2001 IEEE 35th International Carnahan Conference on, 2001, pp. 229-234.

[10] Q.-A. Kester, "A cryptosystem based on Vigenère cipher with varying key," International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), vol. 1, pp. pp: 108-113, 2012.

[11] C. Bhardwaj, "Modification of Vigenère Cipher by Random Numbers, Punctuations & Mathematical Symbols," Journal of Computer Engineering

(IOSRJCE) ISSN, pp. 2278-0661, 2012

[12] F. H. S. Fairouz Mushtaq Sher Ali, "Enhancing Security of Vigenere Cipher by Stream Cipher," International Journal of Computer Applications, vol.

100, pp. 1-4, 2014

[13] P. Gutmann, ―Cryptographic Security Architecture: Design and Verification‖. Springer-Verlag,2004.

[14] Jakimoski, Kire, "Security Techniques for Data Protection in Cloud Computing." International Journal of Grid and Distributed Computing 9.1 (2016): 49-56.

[15] M. Abror, "Pengertian dan Aspek-Aspek Keamanan Komputer," 2018. [Daring]. Tersedia pada:https://www.ayoksinau.com/pengertian-dan-aspek- aspek-keamanan-komputer-lengkap/. [Diakses: 01-Okt-2018].

[16]V.Beal.(2009,Encryption.Available:Http://www.webopedia.com/TERM/E/encryption.ht ml