DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
PUDUCHERRY
KARAIKAL – 609609

Network Security – CS429

Binary Playfair Encryption

A report submitted in partial fulfilment of the requirements for the award of the degree of
Bachelor of Technology
in
Department of Computer Science and Engineering
by

Name	**Roll No.**
Anurag Kumar	CS19B1006
Rishav Gupta	CS19B1043
Shivraj Singh	CS19B1052
Anil Kumar	CS18B1006

Date of submission:
October 26, 2022

BONAFIDE CERTIFICATE

This is to certify that the project work entitled "**Binary Playfair Encryption**" is a bonafide record of the work done by **Anurag Kumar (CS19B1006)**, **Rishav Gupta (CS19B1043)**, **Shivraj Singh (CS19B1052)**, and **Anil Kumar (CS18B1006)** in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering of the **NATIONAL INSTITUTE OF TECHNOLOGY PUDUCHERRY** during the odd semester of the year **2022 - 2023**.

Faculty Signature

ABSTRACT

The well-known multiple-letter encryption cipher is the Playfair cipher. Here the digraphs in the plaintext are treated as single units and converted into corresponding ciphertext digraphs. However, because of the drawbacks inherent in the 5*5 Playfair cipher which adversely affects security, we proposed a 4*4 Binary Index-based rotation Playfair cipher. The various limitations of Playfair cipher are addressed in this project by developing a novel approach which converts the plaintext into a binary representation, then applies a left shift operator depending on the index and later uses Playfair cipher.

Keywords:

Playfair cipher, matrices, Special symbols, Random number, cryptanalysis, brute force, Polyalphabetic cipher.

ACKNOWLEDGEMENT

We would like to thank our teacher, **Dr R. Maivizhi**, for giving us this opportunity and for her constant motivation and guidance during the project. The knowledge we gained during this project, We are sure it will be helpful for a long time.

We would also like to thank all our sources, mentioned in the references, and our friends who helped us by providing mental and logistical support.

TABLE OF CONTENTS

PROJECT DESCRIPTION

INTRODUCTION:

The Playfair cipher is the earliest and best-known digraph substitution cipher to use symmetry encryption. Charles Wheatstone developed the cipher in 1854, and Lord Playfair, who supported its use, gave it its name.

The Playfair cipher shows a significant advancement over the monoalphabetic ciphers. The identification of diagrams is more difficult than individual letters. In the Monoalphabetic cipher, the attacker searches in 26 letters only. But by using the Playfair cipher, the attacker must search in 26 x 26 = 676 diagrams. The relative frequencies of individual letters exhibit a much greater range than that of diagrams, making frequency analysis much more difficult.

Although Playfair cipher is an upgrade over monoalphabetic ciphers it still has various drawbacks which we intend to address in this project. These limitations are listed below.

LIMITATIONS OF PLAYFAIR CIPHER:

- It supports only 26 letters of the English alphabet (either lowercase or uppercase)
- In 5 x 5 Playfair cipher only 25 letters can be represented in the matrix, so we end up having a conflict in one cell of the matrix (usually i/j conflict)
- The cryptanalysis of the Playfair cipher is aided by the fact that a digraph and its reverse will encrypt in a similar way (If AB encrypts to XY then BA will encrypt to YX)

PROBLEM STATEMENT:

To design and develop an algorithm which overcomes the above-mentioned drawbacks of the Playfair cipher.

METHODOLOGY

1. Proposed Algorithm

Various steps involved in the proposed algorithm:

Algorithm for Encryption:

1. Take **Plaintext** and **Key** as input from the user
2. If the length of the **Plaintext** is odd then add a buffer letter **'X'** at the end of the **Plaintext**.
3. Convert the text into character-wise binary representation using **ASCII** values. Each character should be represented in 8-bit binary form.
4. Now perform a **Left Rotation Operation (ROL)** based on the **index** of character
5. Now split the 8-bit binary representation into two blocks of 4-bits and further represent it in its corresponding hexadecimal form.
6. Create a **4x4 Substitution Matrix** using the **Key** as it is normally done in the classic Playfair algorithm.
7. Now take the first hexadecimal character of a **digraph** and encrypt it using the Substitution matrix. Perform the same operation on the second hexadecimal character as well.
8. Repeat **Step 7** for each **digraph**.
9. Represent the encrypted hexadecimal characters in their corresponding decimal values to get the ASCII value of ciphertext.
10. Use these decimal values and the ASCII table to obtain the final ciphertext.

Algorithm for Decryption:

1. To decrypt, represent ciphertext in the form of corresponding decimal values using the **ASCII table**.
2. Create a **4x4 Substitution Matrix** using the **Key** as it is normally done in the classic Playfair algorithm.
3. Decrypt the first hexadecimal character of a **digraph** using the Playfair matrix. Perform the same operation on the second hexadecimal character as well.
4. Repeat step three for each **digraph**.
5. Convert the newly formed hexadecimal representation of each character in their corresponding 8-bit binary form.

6. **Right Rotation Operation (ROR)** on this binary representation based on the index.
7. Represent this binary form into its corresponding decimal value.
8. Obtain plaintext from this representation using the ASCII table.

Encryption Example:

Plaintext	HELLO
Key	F5D0B2

Substitution Matrix

F	5	D	0
B	2	1	3
4	6	7	8
9	A	C	E

Encryption Table

Index	0	1	2	3	4	5
Token	H	E	L	L	O	X
ASCII	72	101	108	108	111	120
BIN	0100 1000	0110 0101	0110 1100	0110 1100	0110 1111	0111 1000
ROL BIN	0100 1000	0100 1000	0100 1000	0110 0011	1111 0110	0000 1111
HEX	4 8	C A	B 1	6 3	F 6	0 F
ENC HEX	7 6	9 E	2 3	4 B	5 4	F 5
ENC BIN	0111 0110	1001 1110	0010 0011	0100 1011	0101 0100	1111 0101
ASCII	118	158	35	75	84	245

ENC	v		#	K	T	õ

Decryption Example:

Plaintext	v#KTõ
Key	F5D0B2

Substitution Matrix

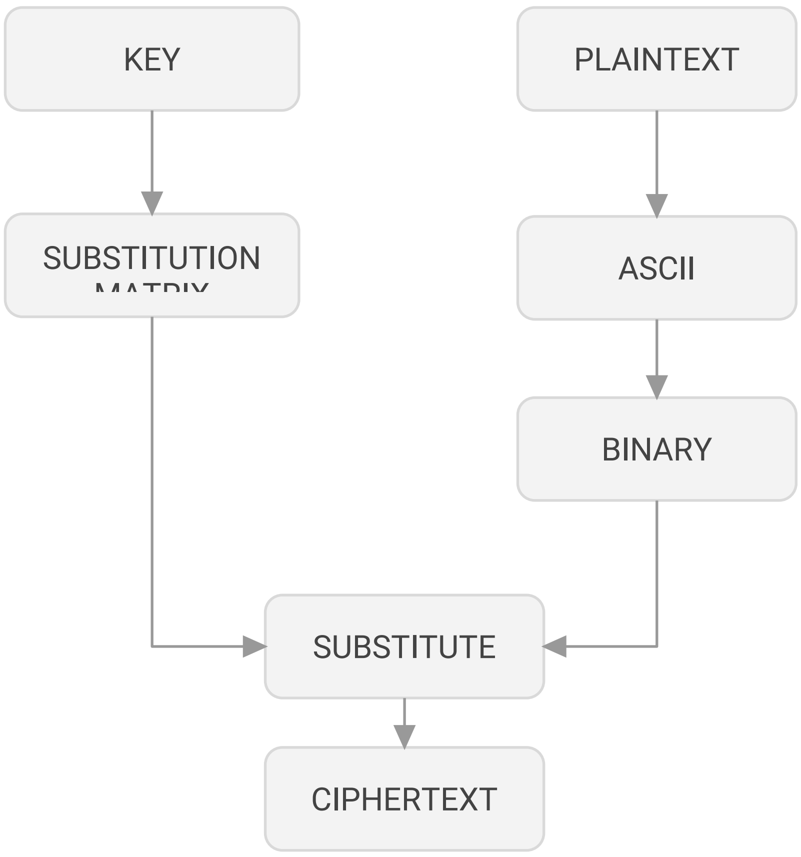
F	5	D	0
B	2	1	3
4	6	7	8
9	A	C	E

Decryption Table

Index	0	1	2	3	4	5
Token	v		#	K	T	õ
ASCII	118	158	35	75	84	245
BIN	0111 0110	1001 1110	0010 0011	0100 1011	0101 0100	1111 0101
HEX	7 6	9 E	2 3	4 B	5 4	F 5
DEC HEX	4 8	C A	B 1	6 3	F 6	0 F
BIN	0100 1000	0100 1000	0100 1000	0110 0011	1111 0110	0000 1111
ROR BIN	0100 1000	0110 0101	0110 1100	0110 1100	0110 1111	0111 1000

ASCII	72	101	108	108	111	120
DEC	H	E	L	L	O	X

2. Architectural Diagram:



3. Technology Used:

Platform used:

To create a web application, we used Reactjs, Nodejs, HTML, and CSS.

Reactjs: It's used for building interactive user interfaces and web applications quickly and efficiently with significantly less code than you would with vanilla JavaScript.

Nodejs: It is used for server-side programming, and primarily deployed for non-blocking, event-driven servers, such as traditional websites and back-end API services, but was originally designed with real-time, push-based architectures in mind.

HTML: It is used to structure a web page and its content.

CSS: It is used for describing the presentation of Web pages, including colours, layout, and fonts.

Software used:

To develop this algorithm, we used Visual Studio Code.

Visual Studio Code: it is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.

We used HTML, CSS and JavaScript to make a visualisation tool.

PERFORMANCE EVALUATION

Cryptanalysis is relatively difficult on our algorithm because it doesn’t have the same representation for each digraph. In classic Playfair cipher the fact that commonly occurring digraphs such as ER will always be encrypted in the same representation, cryptanalysis is easier. Whereas due to added rotation this kind of cryptanalysis cannot be performed on our newly designed algorithm.

RESULTS AND CONCLUSION

In this project we developed a modified Playfair cipher. The modifications firstly, include a 4 x 4 matrix of hexadecimal values. Secondly, it adds a left shift operation based on the index of the character. These modifications help to overcome the problems in traditional Playfair cipher.

Considering the analysis made, we think that the proposed cipher is strong, and it is difficult to break it by any cryptanalytic attack. It would be very effective if it is used properly in the right environment.

REFERENCES

1. Srivastava, S.S. and Gupta, N., 2011. A novel approach to security using extended Playfair cipher. *International Journal of Computer Applications*, *20*(6), pp.0975-8887.
2. https://www.geeksforgeeks.org/playfair-cipher-with-examples/
3. https://intellipaat.com/blog/playfair-cipher/
4. https://en.wikipedia.org/wiki/Playfair_cipher