Quantum Safe Cryptography

A Report Submitted
in Partial Fulfillment of the Requirements
for the Degree of
Bachelor of Technology
in
Computer Science & Engineering

by
Golu Kumar (20174024)
Adarsh Awasthi (20174015)
Ashish Patel (20174006)

to the
COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
MOTILAL NEHRU NATIONAL INSTITUTE OF TECHNOLOGY
ALLAHABAD PRAYAGRAJ
May, 2021

UNDERTAKING

I declare that the work presented in this report titled "*Quantum Safe Cryptography*", submitted to the Computer Science and Engineering Department, Motilal Nehru National Institute of Technology Allahabad, Prayagraj, for the award of the ***Bachelor of Technology*** degree in ***Computer Science & Engineering***, is my original work. I have not plagiarized or submitted the same work for the award of any other degree. In case this undertaking is found incorrect, I accept that my degree may be unconditionally withdrawn.

May, 2021
Allahabad

Golu Kumar (20174024)
Adarsh Awasthi (20174015)
Ashish Patel (20174006)

CERTIFICATE

Certified that the work contained in the report titled "*Quantum Safe Cryptography*", by *Golu Kumar (20174024)*
Adarsh Awasthi (20174015)
Ashish Patel (20174006), has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

(Dr. Anoj Kumar)
Computer Science and Engineering Dept.
M.N.N.I.T, Allahabad

May, 2021

Preface

Quantum computers are computers that leverage phenomena from quantum physics in order to run different kinds of algorithms than the ones we're used to.

Quantum computers don't exist yet and look very hard to build, but if they do exist one day, then they'll have the potential to break RSA, Diffie–Hellman, and elliptic curve cryptography—that is, all the public-key crypto deployed or standardized till today.

This project thrives to find a possible solution to the potential security issues that may be faced by software industry in future by proposing a paradigm for quantum safe cryptography algorithms.

Acknowledgements

We would like to take this opportunity to express a deep gratitude towards our mentor and guide Dr. Anoj sir for his thorough and continuous guidance during the development phase of this project.

We are also obliged to our institute for this opportunity to explore the core subject of computer science which will help us in getting more exposure in future endeavours.

Contents

Chapter 1

Introduction

This section is devoted to explore how we got the idea of working on future generation version of quantum safe algorithms and why it is the need of time to move to more rigorous development of quantum resisting cryptography.

1.1 Motivation

To insure against the risk posed by quantum computers, cryptography researchers have developed alternative public-key crypto algorithms called post quantum algorithms that would resist quantum computers.

In 2015, the NSA called for a transition to quantum-resistant algorithms designed to be safe even in the face of quantum computers, and in 2017 the US standardization agency NIST began a process that will eventually standardize post-quantum algorithms.

Some of the finalists of NIST competition used a similar symmetric cipher as AES-256 but they were using techniques as lattice-based and hash-based algorithms.From this, we got the idea that the use of code-based correction can also be employed here, as many communication technologies use this technique to verify the message codes which has travelled through a noisy medium.

1.2 Challenges

- **Basically, all the quantum computing challenges puts forward two most important requirements:**

  1. **The law of diffusion** - Every bit of cipher should depend on every bit of plain text and every bit of the key.
  2. **The law of confusion** - There should be minimal obvious relationship between plain text, key and cipher.

  These two requirements must be fulfilled as much as possible as a complete concretization is not ideal in real world situation.

- **The problem of handling text data of different languages other than plain English, for e.g. Hebrew, Mandarin etc.**

  The problem of conversion of plain text for different linguistics to binary can be solved by accepting one of the standard encoding-decoding paradigm.

- **The problem of information loss due to limited key size**

  All the plain text messages need not to be the multiple of size of secret keys, this is why we need to put padding at the end of encoded text before sending it for encryption.

Chapter 2

Related Work

2.1 Hash Based cryptography

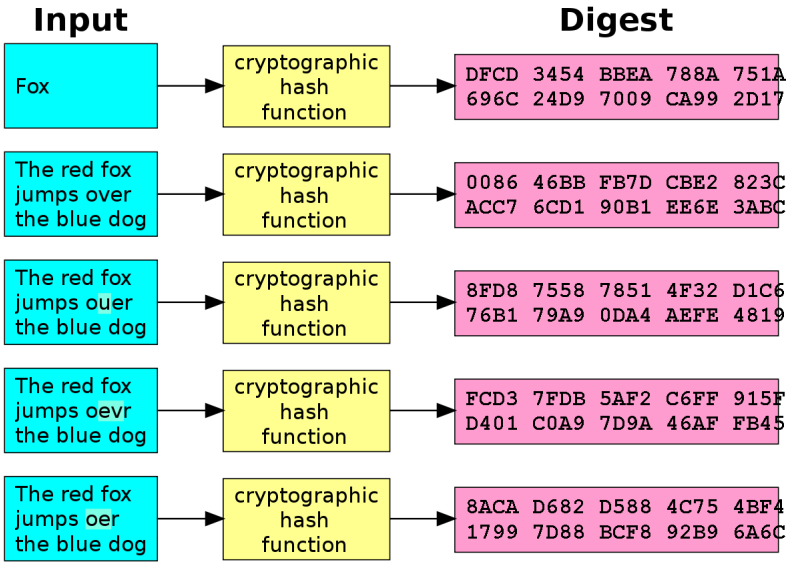

Figure 1: Hash Based Cryptography

As it is evident from above figure that a slightest change in the plain text can bring a huge difference between consecutive ciphers. When used with the substitution process, this intermediate hash value will get sufficiently mixed up with keys and it would become a really tough task to launch a brute force attack.

2.2 Lattice Based Cryptography

Lattice Based cryptography

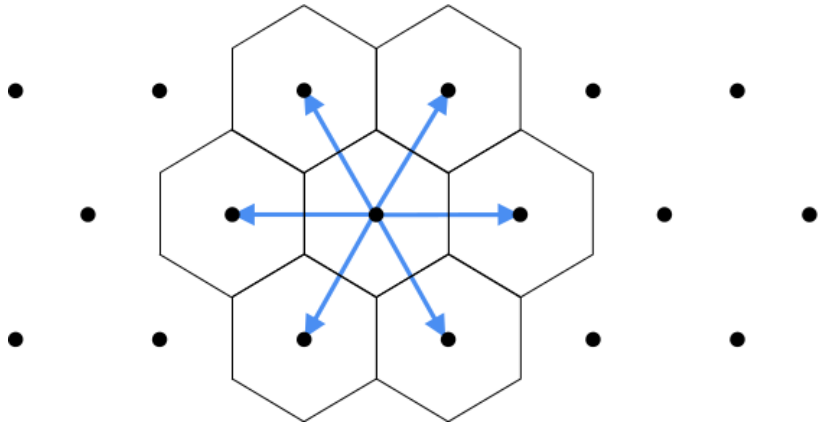

Figure 2: Lattice Based Cryptography

Lattices are mathematical structures that essentially consist of a set of points in an n-dimensional space, with some periodic structure.

1. A first hard problem found in lattice-based crypto is known as short integer solution (SIS). SIS consists of finding the secret vector s of n numbers given (A, b) such that b = As mod q, where A is a random m × n matrix and q is a prime number.

2. The second hard problem in lattice-based cryptography is called learning with errors (LWE). LWE consists of finding the secret vector s of n numbers given (A, b), where b = As + e mod q, with A being a random m × n matrix, e a random vector of noise, and q a prime number. This problem looks a lot like noisy decoding in code-based cryptography.

Both of the above suggest that there is a lot of opportunities and potential solutions available in form of communication technologies which use different innovative algorithms for ensuring the safe transmission in spite of noisy channels.

Chapter 3

Proposed Work

3.1 Code Based Cryptography

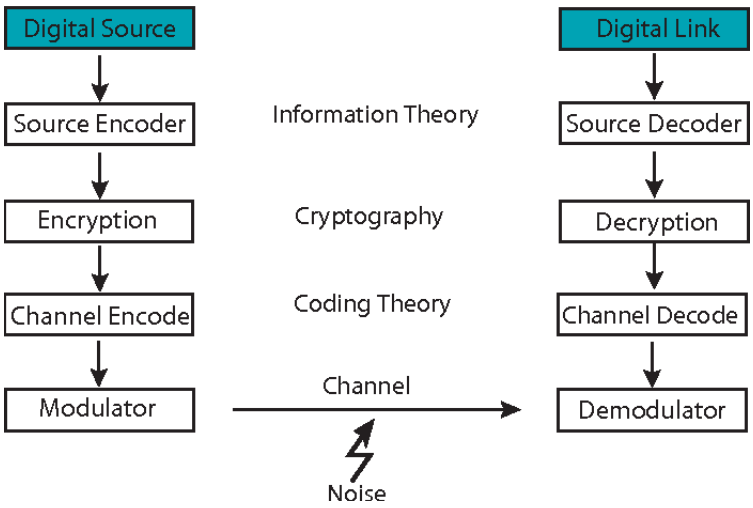

Figure 3: Coding Theory Processing

Code-based post-quantum cryptographic algorithms are based on error-correcting codes, which are techniques designed to transmit bits over a noisy channel.

The first code based encryption scheme (the McEliece cryptosystem) was developed in 1978 and is still unbroken. Code-based crypto schemes can be used for both encryption and signatures.

To transmit a sequence of bits as a sequence of (say) 3-bit words, but the transmission is unreliable and you're concerned that 1 or more bits may be incorrectly transmitted: you send 010, but the receiver gets 011.

One simple way to address this would be to use a very basic error-correction code: instead of transmitting 010 you would transmit 000111000 (repeating each bit three times), and the receiver would decode the received word by taking the majority value for each of the three bits. For example, 100110111 would be decoded to 011 because that pattern appears twice. But as you can see, this particular error-correcting code would allow a receiver to correct only up to one error per 3-bit chunk, because if two errors occur in the same 3-bit chunk, the majority value would be the wrong one.

Linear codes are an example of less trivial error-correcting codes. In the case of linear codes, a word to encode is seen as an n-bit vector v, and encoding consists of multiplying v with an m $\times$ n matrix G to compute the code word w = vG. (In this example, m is greater than n, meaning that the code word is longer than the original word.) The value G can be structured such that for a given number t, any t-bit error in w allows the recipient to recover the correct v. In other words, t is the maximum number of errors that can be corrected.

3.2 Nested substitution according to random permutations

While performing the substitution, one layer of substitution will not be enough, so we have taken idea of performing nested substitution which is based on already defined matrix of key-value pairs.

Chapter 4

Experimental Setup and Results Analysis

4.1 Secret Keys generation and keys for whitening

One very interesting part of the algorithm we have worked upon is that the keys are also computed according to the input/output whitening provided as the only raw material.

4.2 Substitution Boxes

The substitution values are computed in the beginning and they remain same throughout the process of the encryption and decryption. It is a map of key-value pairs consisting of 265 entries, because up to 8-bit nos, we will need not more than keys of size 256.

4.3 The structure and flow of the Algorithm

4.3.1 Encoding and Padding

Conversion of plain text from different linguistics to universal Unicode variables and adding the suitable padding so that the key size can divide encoded plain text in uniform chunks of variables.

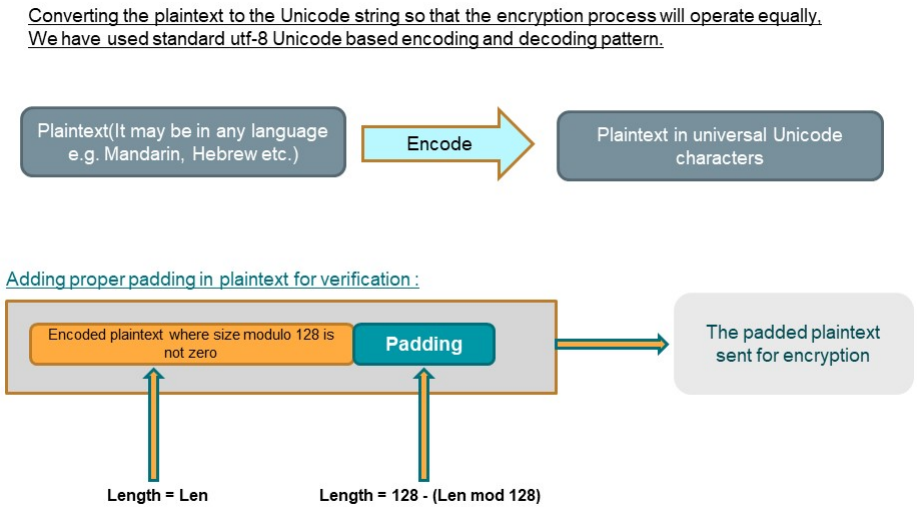

Figure 4: Encoding and padding

4.3.2 Fiestel Cipher Structure

Feistel Cipher model is a structure or a design used to develop many block ciphers such as DES. Feistel cipher may have invertible, non-invertible and self invertible components in its design.

We have changed the basic structure in which this fiestel cipher is further jumbled but is reversible as the cipher is overall symmetric.

4.3.3 The Round Function

The round function makes the input to go through 16 rounds of cyclic encryption process as the modified fiestel cipher as skeleton.

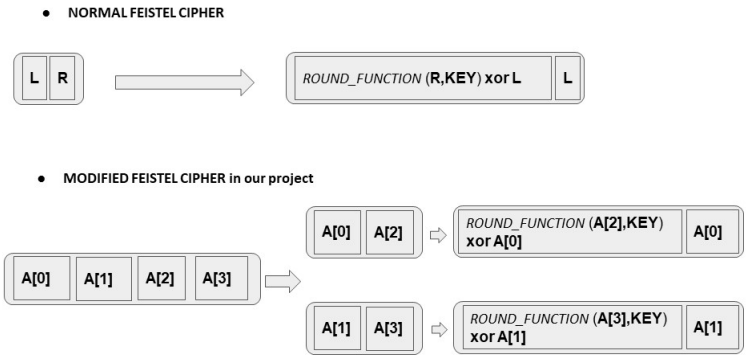

Figure 5: Modified Fiestel Cipher Application For Our Interest

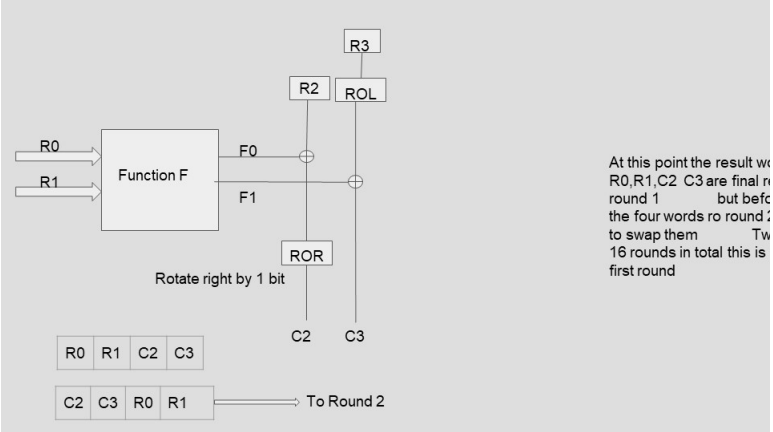

Figure 6: Round Function methodology for iterative encryption

4.3.4 Substitution and applying MDS Matrix(For Linear transformation)

Maximum Distance separable matrix is used for implementing the diffusion part of algorithm, which transforms the plain bits-string into a code-corrected pattern.

The substitution boxes are governed by already determined mapping which remains constant during the whole life of cipher. This should be applied in a random permutation, without repeating the same pattern in two subsequent plain-text processing.

The matrix multiplication here is not a normal multiplication, it is governed

by the polynomial arithmetic rule of multiplication modulo. Polynomial modulo multiplication is one of the most famous aspect of abstract algebra.

A word about no. of rounds used:- Sixteen rounds corresponds to 8 cycles, which seems to be the norm for block ciphers. DES, IDEA, and Skipjack all have 8 cycles. Two

sh was needed to have 16 rounds primarily out of pessimism. Although our best non-related-key attack only breaks 5 rounds of the cipher, we cannot be sure that undiscovered cryptanalysis techniques do not exist that can do better. Hence, we consider 16 rounds to be a good balance between our natural skepticism and our desire to optimize performance.

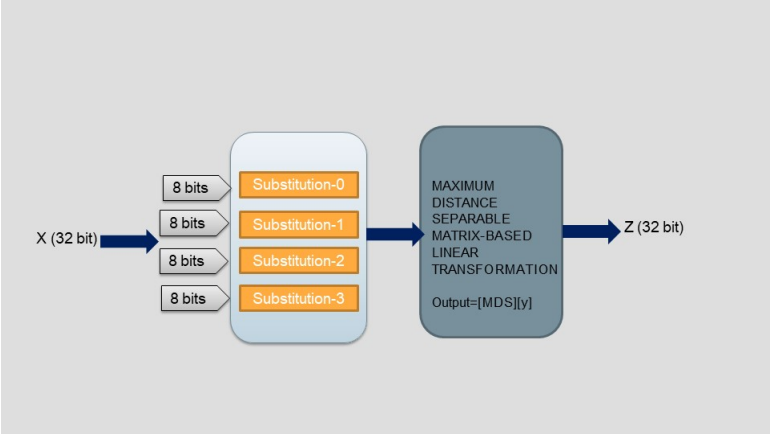

Figure 7: Substitution and Linear transformation

4.3.5 The final output of the program as run in python environment

As it can be easily seen that we can convert the messages of different liguistic into unicode variables and at the same time recover the original message as it is!

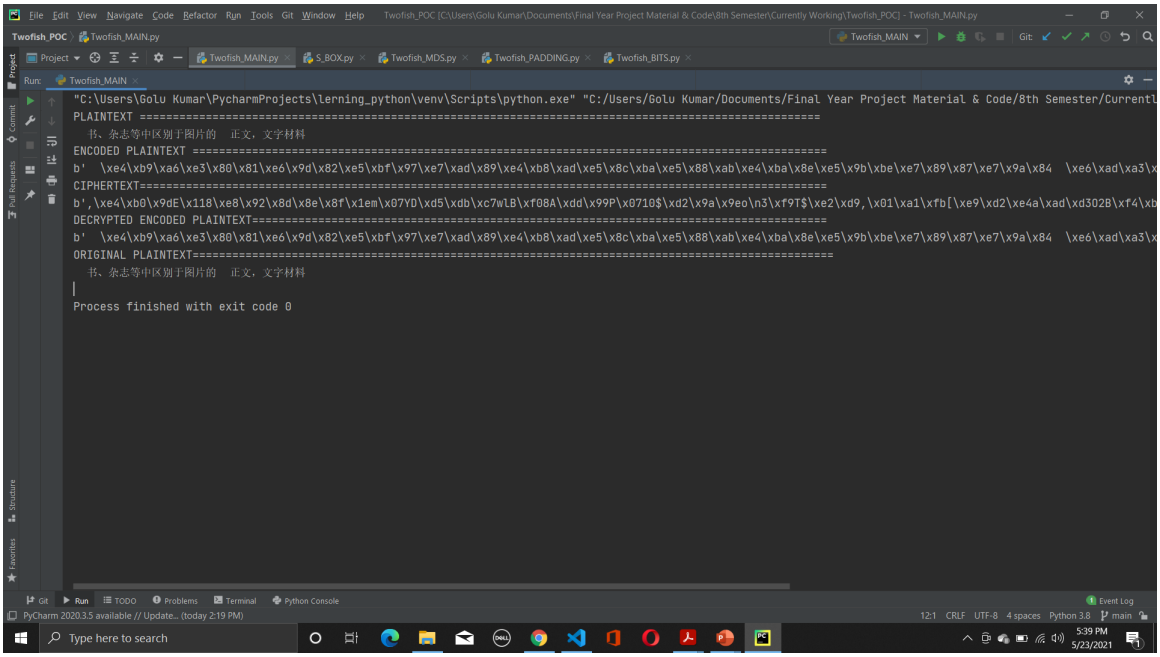

Figure 8: The screenshot of output as executed in PyCharm

Chapter 5

Conclusion and Future Work

5.0.1 Conclusion

We studied a variation of algorithms which participated in the NIST standard Cipher competition in 2019 and tried to re-invent the algorithm with the help of an emerging branch of problem solving mathematics i.e. coding theory.We completed the algorithm design and are progressing towards the efficient implementation of the algorithm.

5.0.2 Future work

Today most of the implemented algorithms are lightweight cryptography algorithms because they are developed on IoT. After discussion with teammates and our mentor , we have decided to extend this project and make this algorithm suitable for IoT devices.

Thank You for Reading !!!

Appendix A

Some Complex Proofs and simple Results

A.0.1 Computing final substitution key-value pairs

$$y_0 = q_0[q_0[q_0[y_{2,0}\,]xorl_{1,0}\,]xorl_{0,0}\,]$$

$$y_1 = q_1[q_1[q_1[y_{2,1}\,]xorl_{1,1}\,]xorl_{0,1}\,]$$

$$y_2 = q_2[q_2[q_2[y_{2,2}\,]xorl_{1,2}\,]xorl_{0,2}\,]$$

$$y_3 = q_3[q_3[q_3[y_{2,3}\,]xorl_{1,3}\,]xorl_{0,3}\,]$$

This is the overall logic which is employed in substitution box initially to make our mapping for whole process in beginning

References

[1] Bruce Schneier, *Applied Cryptography*,June 1993.

[2] : Paul van Oorschot, Alfred Menezes, Scott Vanstone, *Handbook of Applied Cryptography*

[3] Jeffrey Hoffstein *An introduction to mathematical cryptography*,1999.

[4] , Andre Neubauer *Coding Theory* October 2007,