**Modified SkipJack Encryption with a new Modular Multiplicative Inverse Technique**

A Mini Project Report Submitted by

Deon Victor Lobo Dhruv Shetty

USN:4NM17CS054 USN:4NM17CS057

Section B Section B



UNDER THE GUIDANCE OF

Mr. Radhakrishna D

Associate Professor

Department of Computer Science and Engineering

in partial fulfilment of the requirements for the award of the Degree of

Bachelor of Engineering in

Computer Science & Engineering

from

Visvesvaraya Technological University, Belgaum



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

N.M.A.M. INSTITUTE OF TECHNOLOGY

(An Autonomous Institution under VTU, Belgaum) (AICTE approved, NBA Accredited, ISO

9001:2008 Certified) NITTE -574 110, Udupi District, KARNATAKA.

**April 2020**



(ISO 9001:2015 Certified), Accredited with ‘A’ Grade by NAAC

: 08258 - 281039 – 281263, Fax: 08258 – 281265

**Department of Computer Science and Engineering**

B.E. CSE Program Accredited by NBA, New Delhi from 1-7-2018 to 30-6-2021

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

# CERTIFICATE

“**Modified SkipJack Encryption with a new Modular Multiplicative Inverse Technique**”

is a bonafide work carried out by

 Deon Victor Lobo Dhruv Shetty

USN:4NM17CS054 USN:4NM17CS057

in partial fulfilment of the requirements for the award of

Bachelor of Engineering Degree in Computer Science and Engineering

prescribed by Visvesvaraya Technological University,

Belgaum during the year 2018-2019.

It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report.

The Mini project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the Bachelor of Engineering Degree.

Signature of Guide Signature of HOD

**ACKNOWLEDGEMENT**

We believe that our project will be complete only after we thank the people who have contributed to make this project successful.

First and foremost, our sincere thanks to our beloved principal, **Dr. Niranjan N. Chiplunkar** for giving us an opportunity to carry out our project work at our college and providing us with all the needed facilities.

We express our deep sense of gratitude and indebtedness to our guide **Mr. Radhakrishna D**, Associate Professor, Department of Computer Science and Engineering, for his inspiring guidance, constant encouragement, support and suggestions for improvement during the course of our project.

We sincerely thank **Dr. K.R. Udaya Kumar Reddy**, Head of Department of Computer Science and Engineering, Nitte Mahalinga Adyantaya Memorial Institute of Technology, Nitte.

We also thank all those who have supported us throughout the entire duration of our project.

Finally, we thank the staff members of the Department of Computer Science and Engineering and all our friends for their honest opinions and suggestions throughout the course of our project.

Deon Victor Lobo (4NM17CS054)

Dhruv Shetty(4NM17CS057)

**Abstract:**

Skipjack is a block cipher that supports a 64-bit block size and a 80-bit key. The block is internally divided into four 16-bit words, where each round applies a keyed non-linear permutation to one word from the block.

We have made some modifications to the code in order to achieve better security and greater frequency distribution in the cipher .

Our modification to skipjack includes increase in the block size of skpjack to 80 bits. The block is divided into 20 bit words.

We have created a new mod function which works on the principle of modular multiplicative inverse ,which takes a 80 bit block and and provides four 20 bit enciphered text as output. This 20 bits blocks are sent to the skipjack algorithm to improve security of the plaintext.

By this we are adding more layers to the encryption making it difficult to decrypt for the attacker and also to improve the entropy , and thereby increasing the security.

**INDEX**

**Chapter 1: Introduction……………………………………………….6**

**Chapter 2: Literature Survey………………………………………….7**

**Chapter 3: Problem Definition………………………………………..9**

**Chapter 4: Methodology……………………………………………....10**

**Chapter 5: Advantages, Disadvantages and Applications…………….15**

**Chapter 6: Result and Discussion……………………………………..17**

**Chapter 7: Reference………………………………………………….19**

**Chapter 1: INTRODUCTION**

In [cryptography](https://en.wikipedia.org/wiki/Cryptography), Skipjack is a [block cipher](https://en.wikipedia.org/wiki/Block_cipher) an [algorithm](https://en.wikipedia.org/wiki/Algorithm) for encryption developed by the [U.S.](https://en.wikipedia.org/wiki/United_States) [National Security Agency](https://en.wikipedia.org/wiki/National_Security_Agency) (NSA). Initially [classified](https://en.wikipedia.org/wiki/Classified_information), it was originally intended for use in the controversial [Clipper chip](https://en.wikipedia.org/wiki/Clipper_chip). Subsequently, the algorithm was declassified.

The algorithm was initially secret, and was regarded with considerable suspicion by many for that reason. It was [declassified](https://en.wikipedia.org/wiki/Classified_information) on 24 June 1998, shortly after its basic design principle had been discovered independently by the public cryptography community.

Skipjack uses an [80-bit](https://en.wikipedia.org/wiki/Key_size) [key](https://en.wikipedia.org/wiki/Key_(cryptography)) to encrypt or decrypt [64-bit](https://en.wikipedia.org/wiki/Block_size_(cryptography)) data blocks. It is an [unbalanced Feistel network](https://en.wikipedia.org/wiki/Feistel_cipher#Unbalanced_Feistel_cipher) with 32 rounds. It was designed to be used in secured phones.

It has two types of rounds, called Rule A and Rule B. Each round is described in the form of a linear feedbackshift register with an additional nonlinear keyed G permutation. Skipjack applies eight rounds of Rule A, followed by eight rounds of Rule B, followed by another eight rounds of Rule A, followed by another eight rounds of Rule B.

[Eli Biham](https://en.wikipedia.org/wiki/Eli_Biham) and [Adi Shamir](https://en.wikipedia.org/wiki/Adi_Shamir) discovered an attack against 16 of the 32 rounds within one day of declassification, and (with [Alex Biryukov](https://en.wikipedia.org/wiki/Alex_Biryukov)) extended this to 31 of the 32 rounds (but with an attack only slightly faster than exhaustive search) within months using [impossible differential cryptanalysis](https://en.wikipedia.org/wiki/Impossible_differential_cryptanalysis). A truncated differential attack was also published against 28 rounds of Skipjack cipher. A claimed attack against the full cipher was published in 2002, but a later paper with attack designer as a co-author clarified in 2009 that no attack on the full 32 round cipher was then known.

Although the algorithm has not fully been attacked and deciphered ,theres a high possibility that it could happen in the future. Hence the introduction of a new layer “**Modular Multiplicative Inverse Technique**” at the beginning of the skipjack the algorithm can hopefully be made immune to any attacks.

**Chapter 2: Literature Survey**

Skipjack is the secret (symmetric) key algorithm encrypts and decrypts data in 64-bit blocks, using an 80-bit key called crypto variable. It takes a 64-bits block of plaintext as input and outputs a 64-bits block of cipher text. The change involves performing 32 steps or iterations of a complex, nonlinear function. As the number of rounds growths, the security of the algorithm growths exponentially. Skipjack is an iterated block cipher with 32 rounds of two kinds, called Rule A and Rule B. Each round is defined in the form of a linear feedback shift register with additional non-linear keyed G permutation. Rule B is fundamentally the opposite of Rule A with minor positioning differences. Skipjack applies eight rounds of Rule A, followed by eight rounds of Rule B, followed by another eight rounds of Rule A and followed by another eight rounds of Rule B. The original meanings of Rule A and Rule B are given in where counter is the round number (in the range 1-32) and where G is a four rounds Feistel permutation whose F function is defined as an 8×8bits S box (called the F table) and each round of G is keyed by eight bits of the key.

Step Rule A does the following:

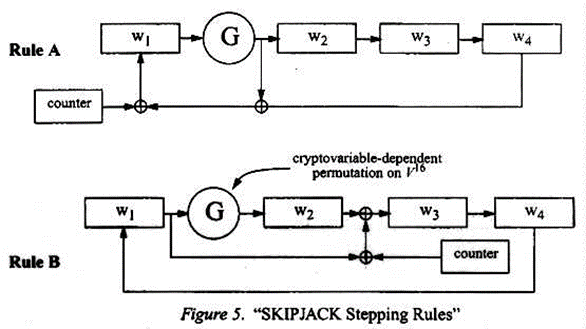
• G permutes w1

• The new w1 is the XOR or of the G output, the counter and w4

• Words w2 and w3 shift one register to the right, that is they become w3 and w4 respectively

• The new w2 is the G output

• The counter is incremented by one



Rule B workings similarly: To encrypt, the algorithm starts the counter at 1. The algorithm phases Rule A for 8 steps, then changes to Rule B for 8 more times. The algorithm again changes to Rule A for next 8 steps and finishes the encryption with 8 steps in Rule B. The counter is incremented by one after each step. To decrypt, the algorithm starts the counter at 32. The algorithm steps into Rule B-1 for 8 steps, Rule A-1 for 8 steps, Rule B-1 for 8 more steps and lastly steps Rule A-1for another 8 steps. The counter is decremented by one after each step.

To understand skipjack algorithm more we referred several websites and YouTube videos which are been added to our reference section.

Our guide Mr.Radhakrishna sir also provided us with his research work to understand the forward encryption and reverse decryption process for our modification .

**Chapter 3: Problem Definition**

The main goal of the modifying the design of skipjack algorithm is that it needs to be secured against unauthorized attacks. For all applied applications, performance and the cost of implementations that is it should be efficient. The modification should secure enough but not slow in performance because it needs to be used in the daily life applications such as e-commerce, banking and online transaction processing applications.

**Chapter 4: METHODOLOGY**

**Working principle of skipjack:**

Skipjack is a block cipher that supports a 64-bit block size and a 80-bit key. The block is internally divided into four 16-bit words, where each round applies a keyed non-linear permutation to one word from the block.

Design specifications:

Designers :NSA

Key sizes : 80 bits

Block sizes : 64 bits

Structure : unbalanced Feistel network[1]

Rounds: 32

SKIPJACK encrypts 4-word (i.e., 8-byte) data blocks by alternating between the two stepping rules (A and B) shown below. A step of rule A does the following:

a. G permutes w1,

b. the new w1 is the xor of the G output, the counter, and w4,

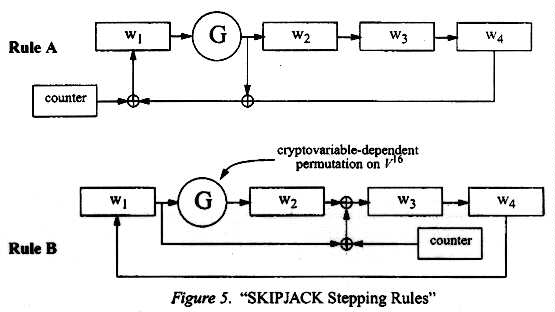
c. words w2 and w3 shift one register to the right; i.e., become w3, and w4 respectively,

d. the new w2 is the G output,

e. the counter is incremented by one.

Rule B works similarly.

It has two types of rounds, called Rule A and Rule B. Each round is described in the form of a linear feedbackshift register with an additional nonlinear keyed G permutation. Skipjack applies eight rounds of Rule A, followed by eight rounds of Rule B, followed by another eight rounds of Rule A, followed by another eight rounds of Rule B.



**Our Modification to skipjack:**

The modified algorithm supports 80 bit block size . The block is internally divided into four 20-bit words .Skipjack applies 4 rounds of rule A and 4 rounds of rule B alternatively until 32 rounds .

Design specifications:

Key sizes : 80 bits

Block sizes : 80 bits

Structure : unbalanced Feistel network[1]

Rounds: 32

**Modular Multiplicative Inverse Function :**

This technique is added at the beginning i.e. before applying skipjack .

**Encryption:**

**Steps Applied for the encipher mod function :**

**Step1:** The plain text is 80 bit blocks divided into four 20 bit words say P1,P2,P3,P4

**Step2:** Find m1,m2,m3,m4 such that :

* (mi\*mj\*mk)≈
* m1,m2,m3,m4 are pairwise co-prime
* Difference between each pair m1,m2,m3,m4 is minimum

**Step 3:** M=(m1\*m2\*m3\*m4)

Mi = M/mi where (i=1,2,3,4)

**Step 4:** Xi = (mi\*) ≡ 1 mod Mi

i.e. Xi is the modular multiplicative inverse of mi wrt Mi

**Step 5:** xi = (Pi\*Xi) mod Mi

**Step 6:** wi = (xi\*k) mod Mi where (k\*) ≡1 mod Mi

**Step 7:** Send w1,w2,w3,w4 to the skipjack algorithm .

w1,w2,w3,w4 are 20 bit integers

**Step 8:** The ciphertext produced is of 80 bits

**Decryption:**

**Step 1:** Apply Skipjack Algorithm to decipher and get w1,w2,w3,w4

**Step 2 :** Apply decipher Mod function on w1,w2,w3,w4

M=(m1\*m2\*m3\*m4)

Mi = M/mi where (i=1,2,3,4)

**Step 3 :** xi = (wi\*) mod Mi where is the inverse of k wrt Mi

**Step 4:** Pi = (xi\*mi) mod Mi

**Flowchart:**



**Modular Multiplicative Inverse Function Solved eg:**

**Encryption:**

Step1:P1=106107 , P2=108102 , P3=105103 , P4=111110

Step2:Consider m1=99 ,m2=100 , m3=101 , m4=103

Step3: Mi = M/mi

M=m1\*m2\*m3\*m4= 102989700

M1 =M/m1= 1040300

M2 =M/m2= 1029897

M3 =M/m3= 1019700

M4 =M/m4= 999900

Step4: Xi = (mi\*) ≡ 1 mod Mi

X1= 388799

X2= 690031

X3= 252401

X4= 708667

Step 5: xi = (Pi\*Xi) mod Mi

x1=P1\*X1 mod M1= 158693

x2=P2\*X2 mod M2= 351246

x3=P3\*X3 mod M3= 606803

x4=P4\*X4 mod M4= 865070

Step 6**:** wi = (xi\*k) mod Mi where (k\*) ≡1 mod Mi

K=223

w1=x1\*k mod M1=18339

w2=x2\*k mod M2=55686

w3=x3\*k mod M3=716669

w4=x4\*k mod M4=929810

**Decryption:**

Step 1: w1=18339 ,w2=55686 , w3=716669 , w4=929810

Step 2: Mi = M/mi

M=m1\*m2\*m3\*m4= 102989700

M1 =M/m1= 1040300

M2 =M/m2= 1029897

M3 =M/m3= 1019700

M4 =M/m4= 999900

Step 3: xi = (wi\*) mod Mi where is the inverse of k wrt Mi

=415187, = 831307 ,=219487 , = 31387

x1 = w1\* mod M1=158693

x2 = w2\* mod M2=351246

x3 = w3\* mod M3=606803

x4 = w4\* mod M4=865070

Step 4: Pi = (xi\*mi) mod Mi

P1=x1\*m1 mod M1=106107

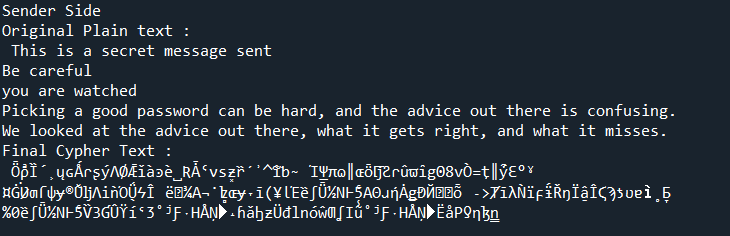
P2=x2\*m2 mod M2=108102

P3=x3\*m3 mod M3=105103

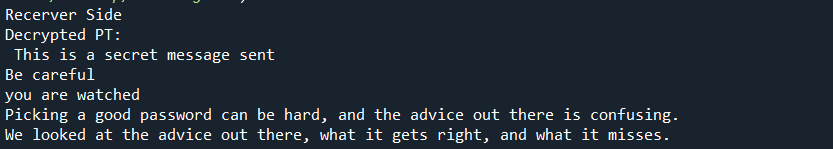
P4=x4\*m4 mod M4=111110

**Our Program Output:**

Encryption :



Decryption:



**Chapter 5: ADVANTAGES, DISADVANTAGES AND APPLICATIONS**

**Advantages:**

* Since no attack on the full 32 round cipher is known till date which makes it very secure.
* The following processes makes the cipher secure:

1. It avoids too much symmetry. Symmetry sometimes allows for clever cryptanalytic attacks. In Skipjack, symmetry is broken with round counters, and this helps avoid, e.g., complementation properties and slide attacks.
2. Round counter an important part of the cipher as eliminating the round counter from Skipjack introduces complementation properties that can be used to speed up exhaustive key search(Brute-force attack).
3. This ordering of round A before round B makes it much harder to peek deep inside the cipher. With B-rounds before A-rounds, one may mount a 7-R attack and look as far as 7 rounds, into the cipher, without too much effort; but applying A-rounds first makes it very expensive to look more than one round deep (i.e., 2-R attacks are too costly). Also, the (8B, 8A)\* structure is weaker against truncated differential cryptanalysis than the real (8A, 8B)∗ cipher.
4. We use a 80-bit key as with a longer key, differential-style attacks would have a lower complexity than exhaustive key search, so it is natural to choose 80 bits as an upper bound on the effective key length of the algorithm. At the same time, keys much shorter than 80 bits are too weak to resist exhaustive key search for long.

* The modification made to the existing algorithm is increases the entropy which leads to a more secure algorithm.

**Disadvantages:**

* It encrypts and decrypts the data slower as compared to algorithms like blowfish.
* Bad interactions between the round-types. The A- and B-rounds interact poorly where they are applied in consecutive rounds. In general, transitions between round-types appear to reduce security. Moreover, if µ is not chosen carefully, this bad effect can be much worse than necessary.

**Applications:**

* The algorithm can be used in any one of the four streaming styles

1. Electronic Code Book
2. Output Feedback
3. Cipher feedback
4. Cipher Chaining

* Previously it used to be secret NSA encryption algorithm and they used to use in clipper chip.

**Chapter 6 : Result and Discussion**

**Skipjack is symmetric:** A cipher is symmetrical if encryption and decryption have the same structure. Symmetry is a beneficial implementation property which simplifies the task of implementation and which may reduce resource requirements, since one can support both encryption and decryption at once with just one cryptographic engine. Also, a symmetrical cipher has identical resistance to chosen-plaintext and chosen-ciphertext attacks, which is a desirable security property.

**Minimize bad interactions between the round-types.** We have seen considerable evidence that the A- and B-rounds interact poorly where they are applied in consecutive rounds. In general, transitions between round-types appear to reduce security, applying 4 rounds of rule A then 4 rounds of rule B alternatively increases

Security.

**Shannon Entropy for cyphertext:**

The graph shows the Shannon entropies for random plaintexts of different sizes .

From this we can understand that the entropies of ciphertext of our modified algorithm are greater than the ciphertext of the original algorithm by an average of 0.3 points.

**Time complexity analysis of Skipjack and Modified skipjack:**

|  |  |  |
| --- | --- | --- |
| Plaintext size | Time Complexity of skipjack in secs | Time Complexity of Modified skipjack in secs |
| 250kb | 7.9 | 8.01 |
| 500kb | 22.56 | 22.7 |
| 1mb | 42.48 | 42.9 |
| 10mb | 421.89 | 422.08 |

From this we see that the time complexities don’t very by a huge margin ,but can be negligible .

**Result:**

Thus our modified program can be said to provide more security with negligible difference with time complexicities.

We were successful in creating a new modular arithmetic technique that helped the skipjack algorithm to be more secure. The modifications made improves the entropy and makes it nearly impossible for the attacker to decipher the message.

**Chapter 7: References**

1. <https://csrc.nist.gov/csrc/media/projects/cryptographic-algorithm-validation-program/documents/skipjack/skipjack.pdf>
2. <https://www.sciencedirect.com/science/article/pii/S0166218X00003474>
3. <https://link.springer.com/chapter/10.1007/3-540-48892-8_27>
4. Eli Biham, Alex Biryukov, Orr Dunkelman, Eran Richardson, Adi Shamir , Initial Observations on the Skipjack Encryption Algorithm, June 25, 1998, available at <http://www.cs.technion.ac.il/~biham/Reports/SkipJack/>.
5. <https://link.springer.com/chapter/10.1007/3-540-48910-X_2>