Republic of Iraq
Ministry of Higher Education
and Scientific Research
University of Diyala
College of Science

Improvement of Chacha Algorithm in Mobile Communication

A Thesis

Submitted to the Computer Science Department \College of Science \University of Diyala
In a Partial Fulfillment of the Requirements for The Degree of Master of Science in Computer

By
Mustafa H. Taha

Supervised By

Assistant Prof. Dr. Jamal M. Abbass

2020 A.D. 1442 A.H.

﴿ قَالَ لَهُ مُوسَى هَلْ أَتَّبِعُكَ عَلَى أَنْ تُعَلِّمَنِ مِمَّا عُلِّمْتَ رُشْدًا ﴾

صدق الله العظيم

سورة الكهف : من الآية 66

ACKNOWLEDGMENTS

First of all, Praise is to God, Lord of the worlds, for a blessing that helped me in achieving this research until the end of it, I would like to express my thanks and gratitude to my supervisor ***Assist Prof. Dr. Jamal Mustafa Abbass*** *for supervising this research and for the bounty, patience and continued guidance throughout the work.*

My thanks to all academics and administrative staff at the Department of computer science.

ⓜⓤⓢⓣⓐⓕⓐ

Dedication.

I would like to dedicate this Work To:

The soul of both my father and my brother, and also dedicate to The rest of my family

mustafa

Supervisor's Certification

We certify that this thesis entitled "***Improvement of Chacha Algorithm in Mobile Communication***" was prepared by "***Mustafa H. Taha***" Under our supervisions at the University of Diyala, Faculty of Science Department of Computer Science, as partial fulfillment of the requirement needed to award the degree of Master of Science in Computer Science.

Signature:

Name: Assistant Prof. Dr. Jamal Mustafa Al-Tuwaijari

Date: 30 / 6 /2020

Approved by the University of Diyala Faculty of Science Department of Computer Science.

Signature:

Name: Assistant Prof. Dr. Taha Mohammed Hassan

Date: 30 /6 /2020

(Head of Computer Science Department)

Linguistic Certification

This is to certify that this thesis entitled *" **Improvement of Chacha Algorithm in Mobile Communication** "* was prepared by *" **Mustafa Hussein Taha"*** at the University of Diyala/ Department of Computer Science, is reviewed linguistically. Its language was amended to meet the style of the English language.

Signature:

Name :

Date : / / 2020

Examination Committee Certification

We certify that we have read the thesis entitled "***Improvement of Chacha Algorithm in Mobile Communication***" and an examination committee, examined the student "***Mustafa Hussein Taha***" in the thesis content and that in our opinion, it is adequate as fulfill the requirement for the Degree of Master in Computer Science at the Computer Science Department, University of Diyala.

(Chairman)
Signature:
Name**: Prof. Dr. Ziyad Tariq Mustafa**
Date: / / **2020**

Signature:
Name: **Assist. Prof. Dr. Salah Awad Salman** **(Member)**
Date: / / **2020**

Signature:
Name: **Assist. Prof. Dr. Adulbasit Kadhim Shukur** **(Member)**
Date: / / **2020**

Signature:
Name: **Asst. Prof. Dr. Jamal Mustafa AL-Tuwaijari** **(Supervisor)**
Date: / / **2020**

Approved by the **Dean** of College of Science, University of Diyala

(The Dean)
Signature:
Name: **Prof. Dr. Tahseen H. Mubarak**

Date: / / **2020**

Abstract

Security plays a vital role in digital information and data protection. To provide security reasonably and appropriately, encryption algorithms must be used. One of the most advanced algorithms in data security is Chacha20. It is a lightweight-stream cipher algorithm.

The proposed system was built based on the Chacha20 algorithm. A proposed system consists of a three-stage and NIST test: Chaotic maps stage,chaotic keystream generation stage, and IChacha20 encryption /decryption Operation Stage. The first stage included the creation of an initial matrix with size [16*16] with a random selection of elements from the initial matrix. The second stage included creating input to matrix called Chacha x-matrix With the implementation of the so-called quarter-round that represents the mathematical model of the algorithm. Finally, The third stage included the IChacha20 encryption /decryption operation. National Institute of Standards and Technology (NIST) Package tests prove that the keys (which are generated by the IChacha20 algorithm) are random and unpredictable, so they are robustness against the attacks. IChacha20 keystream is passed most of the NIST tests with high success rates.

The proposed IChacha20 algorithm was applied to multi-media, including text, We used the correlation coefficient metric, the best results have been obtained. Also, the results showed values of metrics ciphering images using the proposed algorithm and they were better than the values obtained using the original algorithm . As well as, the audio signal, the results had provided values of metrics that evaluating the ciphering of audio samples by using IChacha20,

where showed the results best values for the metrics both of correlation coefficient and signal to noise ratio.

The proposed IChacha20 has obtained faster execution time than the original Chacha20, where the execution time of the original Chacha20 algorithm was= 02:07:1sec while the execution time of the proposed algorithm was =01:26:4 sec.

List of Contents

List of Figures

List of Tables

Abbreviations

1D	One-dimensional
AES	Advanced Encryption Standard
IChacha20	Improvement Chacha20
IV	Initial Vector
log	Logarithm
MSE	Mean Square Error1
n	degree of Chebyshev polynomial
NCC	Normalized Cross-Correlation
NIST	National Institute of Standards and Technology
PNB	Probabilistic Neutral Bits
PSNR	Peak Signal to Noise Ratio
Q.R	Quarter-Round
SNR	Signal to Noise Ratio
SOAP	Simple Object Access Protocol
TLS	Transport Layer Security
UDDI	Universal Description Discovery and Integration
UQI	Universal Quality Index
W.S	Web Service
WSA	Web Services architecture
WSDL	Web Services Description Language
X0	initial value of Chebyshev function
XML	Extensible Markup Language

List of Algorithms

Chapter One

Introduction

1.1 Overview

Network security is a general term that incorporates a plurality of technologies, devices (hardware), and processes (software). In its simplest term, it is a collection of rules and components engineered to protect the integrity, accessibility as well as the confidentiality of communications systems and information using both software and hardware technology. Each institution, regardless of size, infrastructure or industry, required a degree of security of the network solutions to protect it from the ever–growing digital threat landscape that exists today [1]

The rapid development of modern web technology and information technology causes individuals, businesses, and government departments to join the Internet, which creates more illegitimate users to attack and devastate the network by imitating internet sites, Trojan horses, fake mail, and backdoor viruses at the same time. The aim of the intruding and attacks on the network are computer devices, therefore, as soon as the intruders succeed, they will cause thousands of computers on the network in a crippled state. Besides, some intruders with ulterior motives perceive the government and the military as the goal that poses enormous threats to national and social security [2].

Cryptography is used for information safety of digital reality for today. It means hidden secrets are concerned with encryption. Cryptography is the science of hiding information. More broadly, it is about designing and analyzing protocols that obstruct the enemy[3]. Various aspects of information security, such as data

integrity, confidentiality, authentication, and non-repudiation, are central to modern cryptography [4].

The Cryptography system can be divided into two parts: first, is symmetric–key cryptography and the second is public–key cryptography. symmetric–key cryptography: In a symmetric–key cryptography system dispatcherand recipient share one key which is utilized to encrypt and decrypt a message. It is also named secret–key cryptography. The algorithms used for symmetric key cryptography are named symmetric–key algorithms. There are two kinds of symmetrical algorithms, like block cipher & stream cipher. Stream ciphers encrypt bits of information one at a time but block ciphers encrypt information by fractioned it into blocks [5].

Algorithms in Cryptographic take an important role in providing the data security against malignant attacks. The competence of the cryptographic algorithm does not only rely on its time taken for encryption and decryption, and it also accounts for the number of the stages utilized to obtain the cipher-text from an original–text[1]. Chacha20 algorithm is one of the common encryption algorithms. Chacha20 is a high-speed stream cipher based on the Salsa20 cipher designed by Daniel J. Bernstein. It aims to improve performance by increasing the diffusion in each round [6]. Although it is a powerful and modern algorithm, it is also may be threatened in penetration, therefore in this thesis, chaos functions were used to generate a random key used with the rounds to increase its strength to IChacha20 based on chaotic functions (Chebyshev and Tent maps).

1.2 Related Works

Many researchers and mathematicians have suggested many works about the Improved Chacha20 algorithm. The following are some studies and discussions that can so far associate works to the suggested work in this thesis:

1. **Daniel J. Bernstein in (2008)** [7]**:** In this article, the author introduced the Chacha family of stream ciphers, a separate version of the Salsa20 family. Chacha implements the same general style principles as Salsa 20 but has modified some of the details. More specifically, there is an improved amount of diffusion each round. He is conceived that the minimal number of secure rounds for Chacha is smaller (and not larger!) than the minimal number of secure rounds for Salsa20. So presents the Chacha family and describes the variation between Salsa 20 and Chacha. The results of the comparison of the Chacha 20 algorithm with the Salsa algorithm showed the following: the same speed on a 64-bit amd64-architecture Core 2 (6f6), the same speed on a 64-bit Sparc-architecture Ultra SPARC IV, 5% faster on a 64-bit amd64-architecture Athlon 64 (15,75,2), 5% faster on a 32-bit x86-architecture Pentium D (f47), 5% faster on a 64-bit ppc64-architecture PowerPC G5 (750), 6% faster on a 32-bit ppc32-architecture PowerPC G4 (7410), 8% faster on a 32-bit x86-architecture Pentium M (695), and 28% faster on a 64-bit amd64-architecture Pentium D (f64).
2. **S.Maitra in (2016)** [8]: In this paper, he revisits the work of Aumasson et.al, whose analyzed the Salsa and Chacha 20 code, to provide a clearer insight of the existing attack (2^{248} complexity for ChaCha7, i.e., 7 rounds) and show certain improvements (complexity around 2^{243}) by exploiting additional Probabilistic Neutral Bits(PNB). More importantly, he described a novel idea that explores the proper choice of the initial vector (IVs) corresponding to the

keys, for which the complexity can be improved further (2^{239}). The choice of IVs corresponding to the keys is the prime observation of this work. he systematically shows how a single difference propagates after one round and how the differences can be reduced with proper choices of IVs. For Salsa too (Salsa20/8, i.e., 8 rounds), he got an improvement in complexity, reducing it to ($2^{245.5}$) from ($2^{247.2}$) reported by Aumasson et.al.

3. **Ganesan et.al** (2016)[9]**:** In this paper, they analyzed and interpreted the property of the diffusion of Quarter Round (QR) of both the Salsa20 algorithm and the Chacha algorithm, in addition to a proposed alternative design called Modified Chacha Core (MCC). They compared the Quarter round (QR) functions of all these three algorithms using the diffusion matrices that reflect a change in output words with a small change in input words. They generated more than a million diffusion matrices for each algorithm depending on the possible permutations of rotation constants used in the QR. They also proved that, for the Salsa algorithm and Chacha algorithm core, there are a high number of alternative rotation constants that generate more diffusion than the original rotation constants. So, they suggested using the MCC core to generate a collision–resistant function of compression for the encoded hash algorithm.
4. **A. R. Choudhuri & S. Maitra in (2016)** [10]: In this paper, they considered the biases in the forward rounds and estimate an upper bound on the number of rounds till such biases can be observed. For this, they proposed a hybrid model (under certain assumptions), where initially the nonlinear rounds as proposed by the designer are considered, and then they employed their linearized counterpart. The effect of reverting the rounds with the idea of Probabilistic Neutral Bits(PNB)are also considered. Based on the assumptions and analysis, they concluded that (12) rounds of Salsa and ChaCha should be considered

sufficient for 256–bit keys under the current best-known attack models, and they recommended that this model may have potential applications in other ARX based ciphers.

5. **S.Dey and S.Sarkar in (2017)**[11]: They gave a new algorithm to design PNBs "Probabilistic Neutral Bits". They have been using this algorithm to enhance existing attacks to decrease both SalSa and ChaCha rounds. These assaults on SalSa and ChaCha are consecutively around 2.27 and 5.39 times Quickly than the existing works of Maitra and Choudhuri (justifiable in FSE 2017), Where Maitra has enhanced the attack to complication $2^{238.9}$ selecting the (IVs) properly to achieve better outcomes, While Choudhuri et.al proposed to use multi-bit outputs rather than single–bit output, They enhanced the complexity to$2^{237.6}$.
6. **P. A. Babu and J. Thomas in (2018)**[12]**:** In this paper, they introduced Freestyle, a randomized, and variable round version of the ChaCha cipher. Freestyle demonstrated the concept of hash-based halting conditions, where a decryption attempt with an incorrect key is likely to take a longer time to halt. This makes it resistant to key-guessing attacks i.e. brute-force and dictionary-based attacks. Freestyle used a novel approach for ciphertext randomization by using a random number of rounds for each block of message, where the exact number of rounds is unknown to the receiver in advance. Due to its inherent random behavior, Freestyle provided the possibility of generating up to(2^{256}) different ciphertext for a given key, nonce, and message; thus resisting key and nonce reuse attacks.
7. **A. Miyaji and Y. Matsuoka in (2018)** [13]: They described the existing security analysis, firstly, they described the stream ciphers Chacha and Salsa, then, the existing security analysis because Chacha needs more analysis of

security because it has been suggested more newly so they suggested compared with the AES algorithm. Moreover, Chacha is an enhancement of Salsa from the diffusion and analysis of the diffusion of Chacha and Salsa. It is important to understand the design of security criteria. In this study, the diffusion analysis is reviewed and weak bits and weak columns of Chacha and Salsa are investigated. To the knowledge of the authors, so they considered, this is the first thorough explanation of the diffusion of Salsa and Chacha.

8. **P. McLaren et.al in (2019)**[14]: In this paper, they explained identifies a significant vulnerability within OpenSSH and OpenSSL and which involves the discovery of cryptographic artifact used within the ChaCha20 cipher. This can allow for the cracking of tunneled data using a single targeted memory extraction. With this, law enforcement agencies and/or malicious agents could use the vulnerability to make copies of the encryption keys used for each tunneled connection. The user of a virtual machine would not be alerted to the capturing of the encryption key, as the method runs from an extraction of the running memory. Methods of mitigation include making cryptographic artifacts difficult to discover and limiting memory access.
9. **S. Dey et.al in (2019)**[15]**:** In this paper, they revisit the existing attacks on Chacha and Salsa ciphers. Firstly, they applied an accurate computation of the attacks complexities of the existing technique instead of the estimation used in previous works. This improves the complexity of something. The differential attacks utilize PNB'S against Salsa and Chacha involve two probability biases: forward probability bias (∈d) and backward probability bias (∈a). In the second part of this paper, a way to growing the backward probability bias is suggested, which helps to decrease the complexity of the attack. Lastly, they concentrate on the principles of the design of Chacha. They suggested a slight medication in the design of this cipher as a countermeasure attack

against differential. They offered that the key recovery attacks suggested against Chacha will not be strong on this improved version.

10. **S. Dey and S.Sarkar in (2020)**[16]**:** In this research, they presented a total theoretical confirmation of both the SalSa and ChaCha differentiators observed. The notion of a probabilistically neutral bit often takes on a crucial role in the main recovery attack. Here, too, they theoretically show the reason for a special key bit of Salsa to be neutral. So this is the first attempt to give a theoretical justification for the recovery of differential key attacks versus these ciphers. And also, this work gives a fascinating insight into the notion of a differential assault against SalSa and ChaCha. Such research finds an accurate explanation for the noticed forward biases for both ciphers, and that was the fundamental tool for each distinguishing differentials as well as attacks of key recovery for decades. The researchers' goal has forever been to grow the noticed bias and, extend it to the next round. This abstract description can assist in getting a good vision from the beginning of propagation of the bias, which will help to find a better distinguisher through some effective tool. This study has also established the theoretic basis for the Probabilistic Neutral Bits(PNB). However, this principle can also help to boost the backward bias by applying some suitable techniques.

1.3 Problem Statement:

In the last few years attempts to crack the Chacha 20 rounds were attempted by attackers by directing several types of attacks such as key recovery attacks, differential attacks, collision attacks, and others. The attackers managed to break rounds of the ChaCha 20 algorithm arrived 7 and more rounds. The first problem not to allow attackers to break any round of the Chacha20 algorithm. The second

problem must be generated a key that has enough randomness cannot be broken, to add power the algorithm and the attackers cannot access in easily.

1.4 Aims of The Thesis

The aims of this thesis are :

1. Improvement Chacha20 algorithm based on chaotic maps functions.
2. Increasing the keyspace to expand the probabilities of the key to be difficult to guess, break, or access.
3. transfers secure data (text, images, and audio) in mobile communication.

1.5 Contribution

The main contribution of this thesis is to add a layer of security to the Chacha algorithm based on the Chaotic maps (Chebyshev and Tent map function), thus this new contribution will provide communication channels for the transmission of data such (text and images & audio).

1.6 Thesis Outlines

The remaining chapters are:

Chapter two which is entitled theoretical background: presents An introduction to the importance of security over networks and mobile phone communications, the Chacha 20 algorithm and its details, and its mathematical model with Chaotic map, Web service, android platform, and some metrics for text, image, and audio.

Chapter three which is entitled The Proposed System: presents the main proposed system, design objectives, and covers the steps of the proposed IChacha20 algorithm using a multi-level of a chaotic map with several algorithms that represent these steps.

Chapter four which is entitled The Results: This presents the results and tests of the proposed system.

Chapter five which is entitled Conclusions, and Suggestions for Future Work: presents the conclusions for the proposed systems, and suggestions for future work.