

بسم الله الرحمن الرحيم





شبكة المعلومات الجامعية التوثيق الالكتروني والميكروفيلم



جامعة عين شمس

التوثيق الإلكتروني والميكروفيلم

قسم

نقسم بالله العظيم أن المادة التي تم توثيقها وتسجيلها
علي هذه الأقراص المدمجة قد أعدت دون أية تغيرات



يجب أن

تحفظ هذه الأقراص المدمجة بعيدا عن الغبار





Ain Shams University
Faculty of Engineering
Computers and Systems Department

A Quantum Attack-Immune Public Key Cipher

by

Ayman Wagih Mohsen Ahmed Mohamed
Bachelor Degree in Computers and Systems Engineering
Ain Shams University 2014

A THESIS

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF (MASTERS OF ELECTRICAL ENGINEERING,
COMPUTERS AND SYSTEMS)

DEPARTMENT OF (COMPUTERS AND SYSTEMS)

Supervised by:

Prof. Dr. Ayman Mohamed Bahaa Eldin
Dr. Mohamed Ali Ali Sobh

Cairo, Egypt
December 2019



Faculty of Engineering
Ain Shams University
Computers and Systems

Examiners Committee

Name : Ayman Wagih Mohsen Ahmed Mohamed
Thesis : A Quantum Attack-Immune Public Key Cipher
Degree : Master's of Computer Engineering - Computers and Systems

Name, Title, and Affiliate	Signature
Prof. Dr. Amr Galal El-Din Ahmed Wassal Professor of Computer Engineering, Cairo University	
Prof. Dr. Mohamed Ali Kamel Watheq El Kharashi Professor of Computer Engineering, Ain Shams University	
Prof. Dr. Ayman Mohamed Bahaa Eldin Sadek Professor of Computer Engineering, Ain Shams University	
Dr. Mohamed Ali Ali Mustafa Sobh Professor of Computer Engineering, Ain Shams University	

Date: 2019-12-14

Abstract

Ayman Wagih Mohsen Ahmed Mohamed
A Quantum Attack-Immune Public Key Cipher
Master's in Electrical Engineering, Computers and Systems
Ain Shams University 2019

In this work we discuss the history of lattice-based cryptography, study the recently developed lattice-based cryptosystems, and compare the performance of the HewHope, Kyber, Saber and Round5 CPA public key cryptosystems and CCA key encapsulation mechanisms. These cryptosystems are among the candidates of the second round of the NIST post-quantum cryptography standardization competition. We concentrate on the performance of these cryptosystems. And the main factors affecting the performance are: polynomial multiplication and random buffer generation.

There are several methods to perform polynomial multiplication such as Karatsuba, Toom-Cook, index-based and NTT methods. The NTT method is the fastest, but it limits the choice of the cryptosystem parameters.

Random buffer generation can be sped up by using AES128 in counter mode or any fast stream cipher instead of the SHA3 function shake128. High performance can be achieved on modern processors by using the new AES instructions AES-NI.

We also profile the Kyber CPA cryptosystem to show the impact of random buffer generation using extendable output functions on the performance of such cryptosystems. We make all our code available at <http://github.com/a1024/pqc>.

Keywords:

Post-quantum, lattice-based crypto, asymmetric crypto, key encapsulation mechanisms

Summary

A Quantum Attack-Immune Public Key Cipher

Ayman Wagih Mohsen Ahmed Mohamed

Masters of Science in Electrical Engineering (Computers and Systems Engineering)

Keywords – Post-quantum, lattice-based crypto, asymmetric crypto, key encapsulation mechanisms

In chapter 1 we present the general idea of the field of lattice-based cryptography and the motivation behind this work. Followed by the preliminaries explaining the notations used in the literature of lattice-based cryptography. Then some of the lattice-based problems are explained.

Chapter 2 discusses various examples of lattice-based cryptosystems. Beginning with the old lattice-based cryptosystems that are now depreciated. Then we discuss the cryptosystems based on the learning with errors (LWE) problem. Then the ring-LWE cryptosystem, and the underlying problem: learning with errors over polynomial rings. Then we discuss the modern ring-LWE based key encapsulation mechanisms (KEMs) that appeared in 2012 and afterwards. Leading to the newest KEMs NewHope, Kyber, Saber, and Round5.

In chapters 3 we discuss the underlying operations such as polynomial multiplication and modular reduction methods, and other implementation details.

In chapter 4 we present our implementations of various functions for lattice-based cryptography including the NTT polynomial multiplication method.

In chapter 5 we present and compare the results of benchmarking of different cryptosystems on different SIMD architectures. We also profiled

the Kyber cryptosystem as an example to show what operations have the major effect on performance. Then we present other methods for random buffer generation and compare them for speed. These methods include: SHA3 functions, AES128 in counter mode, and some of the fastest stream ciphers available as of time of writing. Then we present the results of randomness tests applied for these methods of random buffer generation.

In chapter 6 we make conclusions and discuss future work.

Thesis supervisors:

Prof. Dr. Ayman Bahaa El-Din

Dr. Mohamed Ali Sobh

Acknowledgements

I would like to thank God for His blessings and that I'm able to finish this thesis.

Also I'd like to thank mom and dad for their continuous encouragement.

I also would like to thank my supervisors Dr Ayman Bahaa and Dr Mohamed Sobh for their advice, guidance and inspiration.

Statement

This dissertation is submitted to Ain Shams University for the degree of Master's in Electrical Engineering in Computers and Systems department.

The work included in this thesis was out by the author at Computers and Systems Department, Ain Shams University.

No part of this thesis has been submitted for a degree of qualification at other university or institution.

Date : 2019-12-14

Signature :

Name : Ayman Wagih Mohsen Ahmed Mohamed

Table of Contents

Abstract.....	3
Summary.....	4
Acknowledgements.....	6
Table of Contents.....	8
List of Figures.....	10
List of Tables.....	12
1. Introduction.....	13
1.1. Preliminaries.....	15
1.2. Lattices and lattice-based problems.....	16
1.3. Outline.....	18
2. Background on Lattice-based public key cryptosystems (PKCs).....	20
2.1. Early works.....	20
2.1.1. The Ajtai cryptosystem.....	20
2.1.2. The GGH cryptosystem.....	21
2.2. LWE based cryptosystems.....	22
2.2.1. NTRU cryptosystem.....	22
2.2.2. Regev 2005 LWE cryptosystem.....	22
2.2.3. Micciancio-Regev 2008 cryptosystem.....	24
2.2.4. Linder-Peikert 2011 (LP11) cryptosystem.....	25
2.2.5. Compact-LWE.....	27
2.3. Ring-LWE based cryptosystems.....	28
2.3.1. LPR10 Ring-LWE cryptosystem:.....	28
2.4. Modern Key Encapsulation Mechanisms (KEMs).....	31
2.4.1. Ding12 KEM.....	31
2.4.2. Peikert's modification to Ding's KEM in 2014 / BCNS15 KEM....	33
2.4.3. NewHope [ADPS16] KEM.....	35
2.4.4. Kyber KEM and PKC.....	38
2.4.5. Saber KEM and PKC.....	40
2.4.6. Round5 KEM and PKC.....	43
3. Implementation Details.....	44
3.1. Polynomial Multiplication.....	44
3.2. Number Theoretic Transform (NTT).....	47

3.3. The Cooley-Tukey FFT Algorithm.....	49
3.4. Single Instruction Multiple Data (SIMD).....	52
3.5. Other methods for polynomial multiplication: Karatsuba method.....	53
3.6. Other methods for polynomial multiplication: Toom-Cook Method....	54
3.7. Modular Reduction.....	55
Barrett reduction.....	56
Montgomery reduction.....	58
4. Proposed library for the NTT and other algorithms.....	60
4.1. Proposed function for schoolbook multiplication using SSE2.....	65
4.2. Advice for fast deterministic random bitstring generation.....	67
4.3. Proposed library for extendable output function using AES-128.....	67
4.4. Other proposed functions.....	70
5. Benchmarks and tests.....	72
5.1. Polynomial multiplication.....	72
5.2. Benchmarks of modern lattice-based public key cryptosystems and key encapsulation mechanisms.....	75
5.3. Profiling of the NewHope, Kyber, Saber, and Round5 cryptosystems	78
5.4. Randomness tests.....	92
6. Conclusion and future work.....	97
6.1. Future work.....	97
References.....	100
شكر.....	103
الملخص.....	104
مستخلص.....	105

List of Figures

Figure 1: Lattice problem notations.	18
Figure 2: Ding12 Hint function $S(K)$	32
Figure 3: Peikert14 / BCNS functions.	34
Figure 4: D4 lattice described by basis B in 2 dimensions and 3 dimensions	36
Figure 5: Polynomial multiplication using discrete Fourier transform (DFT).	47
Figure 6: DFT matrices of size $n=4$	48
Figure 7: NTT matrices of size $n=4$ with modulus $q=5$	48
Figure 8: NTT matrices of size $n=4$ with modulus $q=17$	49
Figure 9: FFT stage matrices for size $n=4$	50
Figure 10: The FFT stages form the bit-reverse permuted DFT matrix.	50
Figure 11: Bit-reverse permutation of size $n=4$	51
Figure 12: Decimation-in-frequency (DIF) FFT stage matrices for size $n=4$	51
Figure 13: Performance results of multiplication of polynomials with 1024 coefficients, in thousand CPU cycles (average time).	74
Figure 14: Performance results of multiplication of polynomials with 1024 coefficients, without the naive method, in thousand CPU cycles (average time).	74
Figure 15: Performance results of our implementation of NewHope, Kyber and Saber CPA PKCs, minimum time in million CPU cycles. (-A) indicates AES-NI.	76
Error: Reference source not foundError: Reference source not found	Error:
Reference source not found	
Figure 17: Performance results of CCA KEMs on Core i7 4770K Haswell (1), and i7 6600U Skylake (2). These results are from [16] and [17].	77
Figure 18: Performance results on Cortex-M4 from [21].	78
Figure 19: Time for deterministic random buffer generation.	79
Figure 20: Output cycles per byte (CPB) of different methods for generation of deterministic pseudo-random buffer.	79

Figure 24: Profile of the Kyber CPA cryptosystem in thousand CPU cycles. Using AVX2, and libkeccak-tiny.	83
Figure 25: Profile of the Kyber CPA cryptosystem in thousand CPU cycles. Using AVX2, and AES in counter mode, with AES-NI instructions..	84
Figure 26: Profile of the Kyber CPA cryptosystem in thousand CPU cycles. Using AVX2, and libkeccak-tiny (blue) vs AES-NI (red).. . . .	85
Figure 27: Profile of the Saber CPA cryptosystem in thousand CPU cycles. Using SSE2, and libkeccak-tiny.	86
Figure 28: Profile of the Saber CPA cryptosystem in thousand CPU cycles. Using SSE2, and AES in counter mode, with AES-NI instructions.	87
Figure 29: Profile of the Saber CPA cryptosystem in thousand CPU cycles. Using SSE2, and libkeccak-tiny (blue) vs AES-NI (red).. . . .	88
Figure 30: Profile of the Round5 CPA cryptosystem in thousand CPU cycles. Using AVX2, and libkeccak-tiny.	89
Figure 31: Profile of the Round5 CPA cryptosystem in thousand CPU cycles. Using AVX2, and AES in counter mode, with AES-NI instructions.. . . .	90
Figure 32: Profile of the Round5 CPA cryptosystem in thousand CPU cycles. Using AVX2, and libkeccak-tiny (blue) vs AES-NI (red).. . . .	91

List of Tables

Table 1: Examples for algebraically acceptable sizes n and modulus q for NTT polynomial multiplication.	53
Table 2: Benchmark results of polynomial multiplication.	73
Table 3: Randomness evaluation of the ISAAC cryptographically-secure pseudo-random number generator (CSPRNG).	92
Table 4: Randomness evaluation of the Rabbit stream cipher.. . . .	92
Table 5: Randomness evaluation of the HC-128 stream cipher.	93
Table 6: Randomness evaluation of the ChaCha20 stream cipher.	94
Table 7: Randomness evaluation of the AES-128 block cipher in counter (CTR) mode.	94
Table 8: Randomness evaluation of the Keccak hash function. Namely, the Shake128 extendable output function (XOF).	95