ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Master's Degree - 2nd Year
Big Data and Internet of Things
Academic Year 2024-2025

Progress Report

Optimization of IoT Communications through Compression and Lightweight Post-Quantum Cryptography

Student:
Abdessamad JAOUAD
M2 Big Data & IoT

Supervisor:
Prof. Ibrahim GUELZIM
ENSAM Casablanca

December 2025

Contents

1 Introduction

This progress report presents the current state of my research project titled **"Optimization of IoT Communications through Compression and Lightweight Post-Quantum Cryptography"**. The project aims to address one of the most critical challenges in modern IoT security: how to protect constrained devices against future quantum computing threats while maintaining acceptable performance levels.

The Internet of Things (IoT) ecosystem is expanding rapidly, with billions of devices deployed across various sectors including healthcare, smart cities, industrial automation, and agriculture. These devices often have severe resource constraints (limited CPU, RAM, flash memory, and battery) yet need to communicate securely over potentially hostile networks. With the advent of quantum computing, current cryptographic standards like RSA and ECC will become vulnerable, making the migration to Post-Quantum Cryptography (PQC) essential.

My work focuses on understanding, analyzing, and proposing optimizations to make PQC algorithms practical for resource-constrained IoT devices, combining cryptographic security with data compression techniques to reduce the communication overhead.

2 Research Objectives

The main objectives of this research project are:

1. **Understand the quantum threat**: Study how quantum algorithms (Shor's, Grover's) break current cryptographic systems and why PQC is needed.
2. **Master PQC algorithms**: Deep dive into the three main NIST-standardized lattice-based algorithms:
   - ML-KEM (CRYSTALS-Kyber) for key encapsulation
   - ML-DSA (CRYSTALS-Dilithium) for digital signatures
   - Falcon for compact signatures
3. **Analyze IoT constraints**: Identify the specific resource limitations of typical IoT devices (MCUs like ARM Cortex-M, RISC-V) and how they impact PQC deployment.
4. **Propose optimization strategies**: Develop and evaluate techniques to reduce the computational and memory footprint of PQC on constrained devices.
5. **Integrate compression**: Explore how data compression can complement PQC to reduce overall communication overhead in IoT networks.

3 Work Completed So Far

3.1 Literature Review and Documentation

I have conducted an extensive literature review covering the following areas:

3.1.1 Post-Quantum Cryptography Fundamentals

I studied the mathematical foundations of lattice-based cryptography, which forms the basis of the main NIST PQC standards. Key concepts I have understood include:

- **Lattices**: High-dimensional discrete structures defined as integer combinations of basis vectors. The security of PQC relies on hard problems over these structures.
- **Learning With Errors (LWE)**: A problem where we try to recover a secret vector s from noisy linear equations:

  $$b_i = \langle a_i, s \rangle + e_i \pmod q$$

  The noise term e_i makes the problem computationally hard, even for quantum computers.
- **Short Integer Solution (SIS)**: Finding a short nonzero vector x such that $Ax \equiv 0 \pmod q$.
- **Module-LWE and Module-SIS**: Structured variants over polynomial rings $R_q = \mathbb{Z}_q[x]/(x^n + 1)$ that provide better efficiency while maintaining security.

3.1.2 NIST Standardization Process

I reviewed the NIST Post-Quantum Cryptography standardization process and the resulting standards:

Table 1: NIST PQC Standards Overview

Standard	Algorithm	Type	Based On
FIPS 203	ML-KEM (Kyber)	Key Encapsulation	Module-LWE
FIPS 204	ML-DSA (Dilithium)	Digital Signature	Module-LWE/SIS
FIPS 205	SLH-DSA (SPHINCS+)	Digital Signature	Hash-based
(Upcoming)	Falcon	Digital Signature	NTRU lattices

3.1.3 Resources Collected and Studied

I have gathered and studied the following key documents:

1. **NIST FIPS 203** - ML-KEM (Module-Lattice-Based Key-Encapsulation Mechanism) official specification
2. **NIST FIPS 204** - ML-DSA (Module-Lattice-Based Digital Signature Algorithm) official specification
3. **Peikert's "A Decade of Lattice Cryptography"** - Comprehensive survey on lattice-based cryptography foundations
4. **Regev's LWE papers** - Original theoretical foundations of the Learning With Errors problem

5. **NIST PQC Migration Project** - Guidelines for transitioning to post-quantum cryptography
6. **Preskill's "Quantum Computing in the NISQ era"** - Context on quantum computing capabilities and timeline

3.2 Understanding the Three Main PQC Algorithms

3.2.1 ML-KEM (CRYSTALS-Kyber)

ML-KEM is the primary standard for key establishment. I have studied its design and operation:

- **Purpose**: Securely establish a shared secret key between two parties (replaces Diffie-Hellman/RSA key exchange)
- **Key Generation**:
  - Sample small secret vector s and error vector e
  - Compute public key: $t = As + e$ (a noisy linear function)
- **Encapsulation**: Sender creates ciphertext (u, v) using fresh randomness
- **Decapsulation**: Receiver recovers the shared secret using private key s
- **Security**: Based on the hardness of Module-LWE problem

3.2.2 ML-DSA (CRYSTALS-Dilithium)

ML-DSA is the primary standard for digital signatures:

- **Purpose**: Authenticate messages, verify identity, ensure integrity (replaces RSA/ECDSA signatures)
- **Design**: "Fiat-Shamir with aborts" approach
- **Signing Process**:
  - Sample random vector y, compute $w = Ay$
  - Hash message with commitment to get challenge c
  - Compute response $z = y + cs_1$
  - Reject and resample if z is too large (prevents secret leakage)
- **Security**: Based on Module-SIS hardness

3.2.3 Falcon

Falcon offers an alternative signature scheme with different trade-offs:

- **Advantages**: Smaller signatures and faster verification than Dilithium
- **Disadvantages**: More complex implementation, requires discrete Gaussian sampling
- **Based on**: NTRU lattices and GPV framework
- **Best suited for**: Bandwidth-constrained scenarios

3.3 IoT Constraints Analysis

I have analyzed the typical resource constraints of IoT devices and their implications for PQC deployment:

Table 2: Typical IoT Device Constraints

Resource	**Typical Range**
CPU	ARM Cortex-M0/M3/M4, RISC-V (16-168 MHz)
Flash (code storage)	64 KB – 512 KB
RAM (working memory)	8 KB – 64 KB
Network bandwidth	Low (LPWAN, BLE, constrained Wi-Fi)
Power	Battery-operated, energy harvesting

Key challenges identified:

1. PQC algorithms have larger key and signature sizes than classical algorithms
2. Polynomial arithmetic (NTT) requires significant computational resources
3. Memory footprint during cryptographic operations can exceed available RAM
4. Side-channel resistance adds additional overhead

3.4 Optimization Strategies Identified

Based on my research, I have identified several optimization strategies applicable to lightweight PQC:

3.4.1 Algorithm-Level Optimizations

- **Parameter selection**: Use lower NIST security levels (Level 1) when appropriate to reduce sizes and computation
- **Polynomial arithmetic**: Choose between NTT (faster, more code) vs. schoolbook multiplication (slower, less code) based on device constraints

3.4.2 Implementation-Level Optimizations

- **Memory management**: In-place operations, buffer reuse, streaming hashing
- **Code sharing**: Common routines (NTT, sampling, hashing) shared between Kyber and Dilithium
- **Compile-time optimization**: LTO, size optimization flags (-Os), removal of unused code
- **Hardware acceleration**: Leverage AES/SHA accelerators and DSP instructions when available

3.4.3 Protocol-Level Optimizations

- **Role separation**: Devices only verify signatures (signing done by servers)
- **Session resumption**: Avoid repeated key exchanges using tickets
- **Gateway offloading**: Powerful local gateways handle PQC-heavy operations
- **One-time onboarding**: Perform expensive operations only during provisioning

3.4.4 Security Considerations

- **Constant-time implementations**: Remove branches on secret data
- **Masking**: Protect against power analysis attacks
- **Careful RNG**: Ensure high-quality randomness for cryptographic operations

4 Key Findings

4.1 The Quantum Threat is Real and Imminent

- **Shor's algorithm** can break RSA and ECC in polynomial time on a quantum computer
- **"Harvest now, decrypt later"** attacks mean adversaries are already collecting encrypted data to decrypt later
- Migration to PQC must begin now, especially for long-lived IoT deployments

4.2 Lattice-Based Cryptography is the Leading Approach

- Three of the four NIST standards are lattice-based (ML-KEM, ML-DSA, Falcon)
- The underlying problems (LWE, SIS) have been studied extensively and are believed to be quantum-resistant
- Lattice schemes offer good performance compared to other PQC families

4.3 PQC on IoT is Challenging but Feasible

- PQC algorithms have larger footprints than classical algorithms
- With proper optimization, they can run on Cortex-M class devices
- Role-based design (verify-only devices) significantly reduces requirements
- Existing optimized implementations (e.g., pqm4 project) demonstrate feasibility

4.4 Compression Can Help

- Larger PQC keys and signatures increase bandwidth requirements
- Data compression before encryption can offset some of this overhead
- Need to balance compression ratio vs. computational cost on constrained devices

5 Next Steps and Planning

5.1 Short-Term (Next 4-6 weeks)

1. **Benchmark analysis**: Study existing PQC benchmarks on microcontrollers (pqm4 project results)
2. **Implementation setup**: Set up development environment for ARM Cortex-M or RISC-V simulation
3. **Compression research**: Survey lightweight compression algorithms suitable for IoT (LZ4, Snappy, specialized schemes)

5.2 Medium-Term (2-3 months)

1. **Prototype development**: Implement a minimal PQC communication stack for a target IoT platform
2. **Optimization experiments**: Apply identified optimization techniques and measure impact
3. **Compression integration**: Combine compression with PQC and evaluate tradeoffs

5.3 Long-Term (Final phase)

1. **Performance evaluation**: Comprehensive benchmarking of the optimized solution
2. **Security analysis**: Verify that optimizations do not compromise security
3. **Documentation**: Final thesis writing and defense preparation

6 Challenges and Risks

1. **Hardware access**: May need physical IoT devices for realistic benchmarking (currently using simulation/emulation)
2. **Implementation complexity**: PQC implementations require careful attention to avoid security vulnerabilities
3. **Rapidly evolving field**: PQC standards and best practices are still maturing
4. **Trade-off balance**: Finding the right balance between security, performance, and resource usage is non-trivial

7 Conclusion

This progress report summarizes the work I have completed so far on the optimization of IoT communications through compression and lightweight post-quantum cryptography. I have:

- Established a solid understanding of the quantum threat and why PQC is necessary
- Studied the mathematical foundations (lattices, LWE, SIS) underlying the main PQC algorithms
- Analyzed the three key NIST-standardized algorithms: ML-KEM (Kyber), ML-DSA (Dilithium), and Falcon
- Identified the specific constraints of IoT devices and their implications for PQC deployment
- Catalogued multiple optimization strategies at algorithm, implementation, and protocol levels
- Collected and studied key reference documents and specifications

The next phase of my work will focus on practical implementation and experimentation, moving from theoretical understanding to hands-on benchmarking and optimization of PQC for constrained IoT environments.

Submitted by: Abdessamad JAOUAD
Date: December 11, 2025
Supervisor: Prof. Ibrahim GUELZIM

References

1. NIST, "FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard," 2024.

2. NIST, "FIPS 204: Module-Lattice-Based Digital Signature Standard," 2024.

3. C. Peikert, "A Decade of Lattice Cryptography," Foundations and Trends in Theoretical Computer Science, 2016.

4. O. Regev, "On Lattices, Learning with Errors, Random Linear Codes, and Cryptography," Journal of the ACM, 2009.

5. NIST, "Post-Quantum Cryptography Standardization Process," https://csrc.nist.gov/projects/post-quantum-cryptography.

6. J. Preskill, "Quantum Computing in the NISQ era and beyond," Quantum, 2018.

7. CRYSTALS-Kyber Team, "CRYSTALS-Kyber Algorithm Specifications," https://pq-crystals.org/kyber.

8. CRYSTALS-Dilithium Team, "CRYSTALS-Dilithium Algorithm Specifications," https://pq-crystals.org/dilithium.

9. Falcon Team, "Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU," https://falcon-sign.info.

10. pqm4 Project, "Post-quantum crypto library for the ARM Cortex-M4," https://github.com/mupq/pqm4.