

**Quantum-Light:**

A Light-weight Quantum-resistant Public Key Encryption scheme on an ARM processor.

A dissertation submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF SCIENCE in Computer Science

In

The Queen’s University of Belfast

by

Nathan Whaley

April 15th, 2024

SCHOOL OF ELECTRONICS, ELECTRICAL ENGINEERING and COMPUTER SCIENCE

CSC3002 – COMPUTER SCIENCE PROJECT

Dissertation Cover Sheet

A signed and completed cover sheet must accompany the submission of the Software Engineering dissertation submitted for assessment.

Work submitted without a cover sheet will **NOT** be marked.

|  |  |  |  |
| --- | --- | --- | --- |
| Student Name: | Nathan Whaley | Student Number: | 40284751 |
| Project Title: | Quantum-Light: A Light-weight Quantum-resistant Public Key Encryption scheme on an ARM processor. | | |
| Supervisor: | Dr Ayesha Khalid | | |

Declaration of Academic Integrity

Before submitting your dissertation please check that the submission:

1. Has a full bibliography attached laid out according to the guidelines specified in the Student Project Handbook
2. Contains full acknowledgement of all secondary sources used (paper-based and electronic)
3. Does not exceed the specified page limit
4. Is clearly presented and proof-read
5. Is submitted on, or before, the specified or agreed due date. Late submissions will only be accepted in exceptional circumstances or where a deferment has been granted in advance.

**By submitting your dissertation you declare that you have completed the tutorial on plagiarism at** [**http://www.qub.ac.uk/cite2write/introduction5.html**](http://www.qub.ac.uk/cite2write/introduction5.html) **and are aware that it is an academic offence to plagiarise. You declare that the submission is your own original work. No part of it has been submitted for any other assignment and you have acknowledged all written and electronic sources used.**

1. If selected as an exemplar, I agree to allow my dissertation to be used as a sample for future students.

|  |  |  |  |
| --- | --- | --- | --- |
| *Student’s signature* | Nathan Whaley | *Date of submission* 15/04/2024 |  |

**Acknowledgements**

First and foremost, I would like to thank God. He has given me strength and encouragement throughout all the ups and downs of completing this project and dissertation. I am truly grateful for His unconditional and endless love, mercy, and grace.

**Abstract**

This dissertation explores the implementation of lightweight quantum-resistant encryption algorithms on ARM processors, addressing the threat posed by quantum computing to traditional encryption methods. Modifications integrating variants of the lightweight Ascon hash into the Kyber-512 algorithm, when ran on a Nucleo L4R5ZI Arm M4 chip; demonstrate a significant 20% improvement in clock cycle usage alongside reduced memory requirements. The findings illustrate the feasibility of lightweight, efficient encryption solutions for resource-constrained embedded systems, strengthening data security in the face of advancing technology. This research contributes insights into the optimization of encryption algorithms, highlighting the potential for practical deployment in low-power computing environments.

**Contents**

[**1.** **Introduction and Problem Area** 6](#_Toc162175815)

[**2.** **Related Work** 7](#_Toc162175816)

[**3.** **Solution Description and System Requirements** 9](#_Toc162175817)

[**4.** **Design** 10](#_Toc162175818)

[**5.** **Implementation** 11](#_Toc162175819)

[**6.** **Testing** 12](#_Toc162175820)

[**7.** **System Evaluation and Experimental Results** 13](#_Toc162175821)

[**8.** **Conclusions and Future Work** 14](#_Toc162175822)

[**9.** **References** 15](#_Toc162175823)

[**10.** **Appendices** 16](#_Toc162175824)

# **1. Introduction and Problem Area**

The coming issue of the realisation of Quantum computers to a sufficient degree to pose a significant  
threat to all classical forms of encryption is a major threat in the field of cybersecurity. Traditionally,  
data was encrypted with algorithms that posed sufficient challenge that it would take a classical  
computer an extremely large amount of time to decipher encrypted data when the encryption method being used is not known. Quantum computers have the potential to break many of the most popular encryption methods used to secure data today. This is due to their ability to perform calculations differently and exponentially faster than classical computers. One example of this would be Shore’s algorithm. [1]  
To combat this future threat, a number of “post-quantum” encryption algorithms have been developed to be secure against both classical and quantum computers. The National Institute of Standards and Technology (NIST) held a post-quantum cryptography competition to standardise a number of quantum resistant algorithms. [2] The winner of this competition was announced in 2022 as “CRYSTALS-Kyber” and its open-source implementation made available on the NIST website in C.  
While Kyber may be fit for larger devices with a greater number of system resources available, the  
SHA-3 family of hashing algorithms it uses are too resource intensive to be suitable for many smaller  
devices such as embedded systems. This poses a problem as the current shift in the technological  
landscape away from PCs to smaller devices like smartphones and embedded systems demands the  
need for post-quantum encryption algorithms that are lightweight enough to be run on low-resource  
and low-power installations. [3]  
The Kyber algorithm is a Key Encapsulation Mechanism (KEM) [4], comprising of 3 main components:  
key generation, encryption and decryption. Each aspect of the KEM is heavily reliant upon several of  
the SHA-3 family of hashing algorithms, namely SHA3-256, SHA3-512, SHAKE128 and SHAKE256. The  
aim of the project will be to determine if it is possible to implement the recently standardised  
CRYSTALS-Kyber post-quantum encryption algorithm on a lightweight device, specifically a NUCLEO L4R5ZI board with an ARM Cortex-M4 chip, by substituting the above hashing algorithms for the more lightweight ASCON [5] algorithm. The ASCON algorithm from the NIST Lightweight cryptography competition called the NIST LWC [6] is better suited for lower-resource applications such as embedded systems and IoT applications. The consequences of this replacement in terms of code/memory/stack sizes, throughput efficiency, security levels, etc. will be thoroughly investigated in this project. A fully working implementation of the Kyber encryption system with Key generation, encryption and decryption operations using a lightweight hashing algorithm on an ARM Cortex-M4 chip will be the goal. The ARM Cortex-M4 is an extremely widely used chip [7] so will be effective at demonstrating suitability of a lightweight post-quantum encryption algorithm for a wide range of embedded devices.

# **2. Related Work**

The following section investigates the existing literature and research efforts in the field of lightweight post-quantum cryptography with a focus on cryptographic systems tailored for IOT devices.

Exploring NIST LWC/PQC Synergy with R5Sneik **[8]**

This study notes that to use any given NIST PQC algorithm; its internal symmetric components must also be implemented on all target platforms. This can be problematic for lightweight, embedded, and hardware implementations. The study details that it has been widely observed that current NIST-approved symmetric components (AES, GCM, SHA, SHAKE) form a major bottleneck on embedded and hardware implementation footprint and performance for many of the most efficient NIST PQC proposals. It then goes on to discuss R5Sneik, a variant of Round5 that internally uses SNEIK 1.1 permutation-based primitives instead of SHAKE and AES-GCM. They found that R5Sneik is up to 40% faster than Round5 for some parameter sets on ARM Cortex M4 and has substantially smaller implementation footprint. This study shows that there is lots of potential that exists for the optimisation of algorithms in the NIST PQC competition for more lightweight implementations.

Ring-LWE Ciphertext Compression and Error Correction **[9]**

The study explores lattice-based public key cryptography algorithms’ ability to transform ciphertext between different lattice or ring representations efficiently, without needing knowledge of public and private keys. In the study it used ciphertext transformation to compress RingLWE ciphertexts, enabling lightweight implementations in IoT devices, Smart Cards, and RFID systems without modifying encryption procedures or security parameters. Through experimentation, it was demonstrated that this compression technique reduces ciphertext size by over 40% while maintaining equivalent security, enabling public key cryptography on previously inaccessible lightweight platforms, and outperforming RSA-based approaches at similar security levels. This study further shows that the potential for modifying the PQC schemes that already exist to make them more suitable for lightweight implementations is very real and doable via a number of approaches.

A post-quantum lattice based lightweight authentication and code-based hybrid encryption scheme for IoT devices **[10]**

This study introduces a lightweight post-quantum lattice-based authentication and code-based hybrid encryption scheme tailored for resource-constrained IoT devices. Similar to the previous study it utilised Ring-Learning with Errors (Ring-LWE) based authentication in addition to Bernstein reconstruction in polynomial multiplication, with the scheme achieving minimal computation cost on IoT devices. The total authentication delay of the proposed authentication scheme was 23% less than the authentication scheme that is considered conventional polynomial multiplication. In summary this study showed a further avenue of modification in PQC schemes, illustrating that there exists a wide variety of potential optimisations that can be done for lightweight PQC on IoT devices.

Summary:

In summary the above studies have shown that there exists a great potential for optimisation in many areas of PQC algorithms for lightweight implementations. One difference between this project and those that currently are published is the use of Kyber, one of the algorithms selected for standardisation in the NIST PQC competition. In the first study detailed: “Exploring NIST LWC/PQC Synergy with R5Sneik”, A NIST round 2 candidate “ROUND5” was used, however this scheme was not selected to proceed to round 3 of the competition. [11] The aim of this project it to determine whether the Kyber scheme can be modified to utilise ASCON, another NIST standard scheme to be more suitable for lightweight implementations, creating a scheme utilising only NIST-standardised algorithms.

# **3. Solution Description and System Requirements**

**3.1 Introduction**

To address the problem area stated before; the impending threat of quantum computers breaking most current forms of encryption and the unsuitability of most post-quantum encryption algorithms for smaller, lower resource applications, it was essential to set out the system requirements with clarity and precision. This section will detail the requirements for the system, both functional and non-functional, all objectives that need to be met and how each of them is to be met, with verifiable success criteria accompanying each objective. Further, this section will discuss the users of the new system being developed as well as any assumptions made and constraints that occurred.

With the problem area in mind, the aim of this project from the outset was to create a post-quantum encryption scheme that is less resource intensive than current schemes that exist, meaning that it will be better suited for running on lightweight IoT devices where resources are far more limited. As seen in Section 2, there are many possible methods of optimising PQC schemes for more lightweight targets. The angle taken on this project to attempt to solve the problem was to modify the hashing scheme used inside an existing PQC scheme. This was favoured over other approaches for a number of reasons, with the main being it allows for a more straightforward solution given the time constraints of this project. Modifying other aspects of a PQC scheme to improve efficiency on lightweight devices might involve complex changes to the algorithm itself necessitating a large time-sink to become familiar with the mathematics involved. Substituting the hash function for a more lightweight one is a more straightforward procedure, reducing development time and system complexity. Further, lightweight hashing algorithms that are part of established standards, such as those endorsed by NIST, ensures there is compatibility with existing cryptographic frameworks and libraries. This simplifies integration efforts and promotes interoperability across different platforms and implementations.

With the method of modification selected, the next stage in tackling the problem was to select a host scheme for the PQC algorithm and a suitable lightweight hashing algorithm to be used as a replacement for the internal hashing algorithm. It was decided to use two NIST standardised algorithms: CRYSTALS-Kyber and ASCON. The choice of both algorithms was strategic, Kyber was selected as the winner of the NIST post-quantum cryptography competition, gaining recognition and validation from the cryptographic community. Using a NIST-standardised algorithm ensures that the new scheme will adhere to cryptographic standards and is suitable for resistance against quantum computer-based attacks. The choice of ASCON as the hashing algorithm is also nuanced. ASCON is specifically designed for low-resource environments such as embedded systems and IoT devices and was also selected to be standardised in the NIST lightweight cryptography competition. The lightweight design of ASCON is complemented by the robust security of Kyber. With this choice of algorithms, it is possible to achieve a balance between security and efficiency by integrating Kyber with ASCON, guaranteeing that the final encryption system remains reliable while consuming the fewest resources possible.

With the schemes to be utilised selected, the next step in development was to select a target platform to use as a test bed for all implementations. The ARM Cortex-M4 chip was selected as the chip of choice due to its widespread adoption and popularity within the embedded systems industry. The ARM Cortex-M4 is widely recognised for its widespread integration into a wide range of embedded devices in multiple industries, as mentioned in reference [7]. Making use of this widespread popularity, the ARM Cortex-M4 is the processor of choice for development of a lightweight post-quantum encryption algorithm as it would reflect the applicability and scalability of the proposed scheme across a multitude of embedded devices.

Detailed below is the section on the updated requirements definition. This section provides a thorough understanding of the evolution of the system requirements over the course of the project. Any changes and improvements that have taken place are detailed, with justification and clarification on all changes made based on gained knowledge and understanding. In addition, the system's overall high-level requirements are carefully outlined. These requirements form the basis of the functionality of the system. Each requirement detailed in this section is elaborated on in section 3.3 with a comprehensive explanation, where each component is covered in detail.

The aim of the project is to attempt to merge 2 NIST standardised algorithms Kyber and Ascon and run it on a nucleo board to onbtain benchmarks to see if the merger of these 2 produces results that are better suited to lightweight implementations. The merger of these will then be tested to ensure that each scheme works as expected. After its correctness has been verified a number of benchmarks will be run, speed size hash proportion and memory usage. To enable ease of comparison we also need to develop an interface that can display charts comparing the various implementations benchmarks.

**3.2 Updated Requirements Definition**

Description of how the system requirements have evolved throughout the project.

Clarification on the changes made based on gained knowledge and understanding.

*High level requirements – system as a whole*

What requirements have changed as the system went on.

Throughout the course of the project the requirements have undergone several iterations. The initial requirements are the same with it being compatibility with the arm chip, encapsulation, decapsulation, key generation and the production of comparative benchmarks. However after working on the project it became clear that there should also be a graphical interface as part of the finished system. This would facilitate the ease of viewing of results and comparison of benchmarks and aid in report writing as graphs and charts can be produced and benchmarks once kept separate could be combined together for ease of comparison.

**3.3 Software Requirements Specification**

***Specific requitments***

**3.3.1 Functional Requirements**

Complete set of function definitions:

Use cases or user stories (if preferred) written to be testable.

Explanation of each functional requirement and its significance.

**3.3.2 Non-functional Requirements**

Measurable and testable non-functional requirements:

Performance, reliability, usability, etc.

Clear description of each non-functional requirement.

**3.4 System Assumptions and Constraints**

List of any assumptions made about the problem.

Identification of system constraints that may impact the development or functionality of the system.

**3.5 Interface Requirements**

Description of interfaces required:

Interactions with other software or systems.

Explanation of how these interfaces facilitate system functionality.

**3.6 User Interface Requirements**

Specific requirements for the user interface:

Layout, design, navigation, etc.

Justification for each UI requirement.

**3.7 User Characteristics**

Detailed overview of user characteristics:

User demographics, technical proficiency, etc.

How these characteristics influence system design and functionality.

**3.8 Summary**

Recapitulation of key points discussed in this section.

Emphasis on the importance of accurate and comprehensive system requirements and specifications.

# **4. Design**

# **5. Implementation**

# **6. Testing**

# **7. System Evaluation and Experimental Results**

# **8. Conclusions and Future Work**

# **9. References**

**https://www.mdpi.com/journal/mathematics/special\_issues/Post-quantum\_Lightweight\_Cryptography**

[**https://csrc.nist.gov/CSRC/media/Events/Second-PQC-Standardization-Conference/documents/accepted-papers/saarinen\_r5sneiktxt.pdf**](https://csrc.nist.gov/CSRC/media/Events/Second-PQC-Standardization-Conference/documents/accepted-papers/saarinen_r5sneiktxt.pdf)

[**http://library.usc.edu.ph/ACM/SIGSAC%202017/iotpts/p15.pdf**](http://library.usc.edu.ph/ACM/SIGSAC%202017/iotpts/p15.pdf)

[**https://www.sciencedirect.com/science/article/abs/pii/S138912862200367X**](https://www.sciencedirect.com/science/article/abs/pii/S138912862200367X)

**https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8309.pdf**

**page 17 ^**

**https://www.nist.gov/publications/advanced-encryption-standard-aes-0**

# **10. Appendices**