Quantum-Light:

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

Updated Project Description, Solution Approach, and Work Plan.

by

Nathan Whaley

Supervised by

Dr. Ayesha Khalid

January 12th, 2024

Contents:

Introduction:.....3
Project Description:.....4
Solution Requirements and Success Criteria:.....5
Functional Requirements:.....5
Non-Functional:.....6
Project development plan:.....7
Gantt Chart:.....7
Description of Each Stage of Development:.....7
Risks and Mitigations:.....9
References:.....10

Introduction:

The project “Quantum-Light: A Light-weight Quantum-resistant Public Key Encryption scheme on an ARM processor" is aimed at addressing the impending threat that quantum computers pose to traditional encryption algorithms. Quantum computers, once sufficiently developed, could potentially break widely used encryption methods, compromising data security and privacy on a global scale. [1] Most systems manufactured in today’s “post-pc” era fall under the category of “embedded systems” [2] which, due to their low-resource nature, pose another set of challenges in encrypting data. This project seeks to evaluate the efficacy of a quantum-resistant encryption algorithm on a simple, low-cost and low power device. Specifically, whether the family of hashing algorithms used in the post-quantum encryption algorithm known as “Kyber” can be substituted with a lightweight replacement which would allow it to be effectively implemented on an ARM Cortex M4 chip.

Project Description:

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. [3]

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. [4] 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.

The Kyber algorithm is a Key Encapsulation Mechanism (KEM) [5], 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 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, decoding errors, 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.

Solution Requirements and Success Criteria:

This section outlines the specific criteria that will be used to evaluate the success of the project. These criteria not only serve as benchmarks for assessing the project's progress and outcome but also ensure that the project meets its intended goals, adheres to high standards of quality, and addresses the key challenges in implementing quantum-resistant cryptography on an ARM processor.

The use of traditional testing frameworks like unit tests, integration tests, functional and acceptance isn't as prevalent in this project as the testing framework is largely built in to the PQM4 library which features its own internal unit, integration and functional tests. This allows for quick and simple testing and benchmarking of the project to the standards set by the developers of the PQM4 library.

Each requirement is detailed with its corresponding success criteria below:

Functional Requirements:

Compatibility with NUCLEO Board and ARM Cortex-M4 Chip:

- The system successfully operates on the NUCLEO board with the ARM Cortex-M4 chip without compatibility issues.

Integration of ASCON Algorithm into Kyber:

- Complete replacement of all SHA-3 family hashes in Kyber with the ASCON lightweight hash algorithm.
- Verification of the integration through the built-in PQM4 tests that confirm the cryptographic integrity and operational consistency with the original design.

Key Generation in Quantum Resistant Framework:

- Ability to generate keys that are proven secure within the quantum resistant Kyber framework.
- Confirmation through functional tests that the keys are suitable for both the encryption and decryption processes.

Support for Encryption and Decryption Operations:

- Successful implementation of encryption and decryption operations using the generated keys.

- Validation of these operations using the functional tests that the encryption and decryption (encapsulation and decapsulation) functions remain valid and usable.

Production of Comparative Benchmarks:

- Generation of benchmarks that measure the performance and effectiveness of the new implementation in the following metrics: code/memory/stack sizes, throughput efficiency and decoding errors.
- Comparative analysis demonstrating how the new implementation fares against the reference implementation in terms of the previously defined metrics.

Non-Functional:

Usability:

- The system should be user-friendly, with clear and extensive documentation and guidelines on the structure of the implementation and use of the benchmarking and testing facilities. Any analysis or comparison of implementations should be well documented.

Maintainability:

- Code is well-documented, with clear comments and structuring that facilitate ease of understanding and modification.

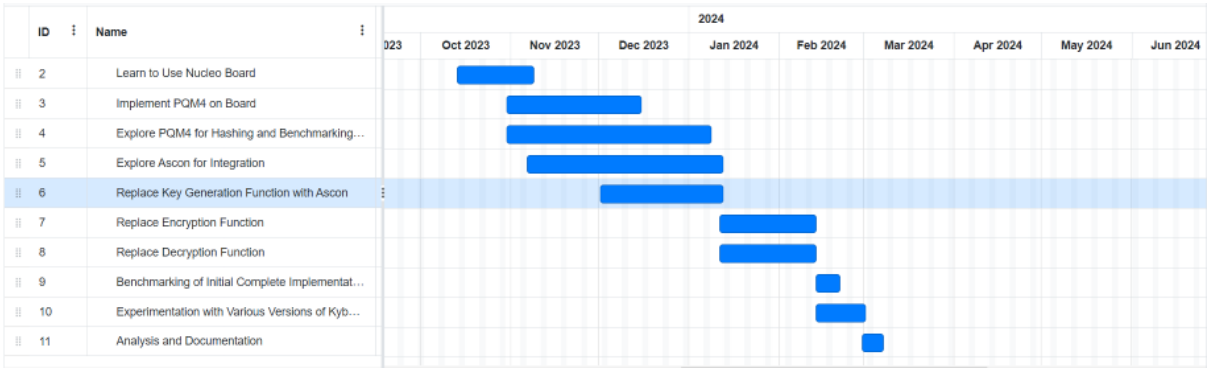
Efficiency:

- The system exhibits a low memory footprint and minimal storage space usage, as measured by the benchmarking tools built into PQM4.

Project development plan:

This section outlines the Project Development, featuring a Gantt chart for a visual timeline and detailed stage descriptions. It also addresses potential risks and their mitigation strategies, ensuring a comprehensive approach to project management.

Gantt Chart:



Description of Each Stage of Development:

Learn to Use Nucleo Board:

- Become familiar with the Nucleo board hardware and development environment.
- Setting up the development environment, install libraries and dependencies.
- Basic programming and testing on the Nucleo board.

Implement PQM4 on Board and Initial Benchmarks:

- Installation and configuration of PQM4 framework on the Nucleo board.
- Running initial benchmarks and tests to confirm Kyber is implemented successfully.

Explore Kyber for Hashing and Benchmarking Framework:

- Deep dive into the code pertaining to the hashing algorithms used in Kyber.
- Understanding the benchmarking, testing and build framework of PQM4.
- Selecting an initial version of Kyber for the project.

Explore Ascon for Integration:

- Studying the structure and format of the Ascon library.
- Identifying relevant Ascon code segments for integration.
- Selecting an initial version of Ascon for the project.

Replace Key Generation Function with Ascon:

- Integrating Ascon into the key generation function of Kyber.
- Testing and validating the new key generation mechanism using in-built tests.

Replace Encryption Function:

- Modifying the Kyber encryption function to incorporate Ascon.
- Ensuring functional integrity post-integration.

Replace Decryption Function:

- Modifying the Kyber decryption function to incorporate Ascon.
- Ensuring functional integrity post-integration.

Benchmarking of Initial Complete Implementation:

- Comprehensive benchmarking of the newly integrated system.
- Analysing performance, efficiency, and security metrics.

Experimentation with Various Versions of Kyber and Ascon

- Implementing different versions of Kyber and Ascon.
- Benchmarking and analysing each combination for comparative study.

Analysis and Documentation

- Detailed analysis of all findings from the experiments.
- Documentation on the integration process, and how to use the new implementation.

The project development should complete approximately March 15th to allow for a 1-month buffer window before submission for anything unplanned to occur and to account for the potential error in estimating the development time of the above stages.

Risks and Mitigations:

This section identifies and analyses potential risks associated with the project and outlines corresponding mitigation strategies. These measures are designed to proactively address challenges, ensuring the project's objectives are met with minimal disruptions.

Difficulty of implementation:

- There could be an issue of development time overrunning if the difficulty in integrating the ASCON algorithm with the Kyber framework proves to be too complex.
- This is mitigated by scheduling the Gantt chart to finish approximately 1 month before the submission date for the project is, allowing for extra time to finish development.

Nucleo board hardware failure:

- The Nucleo board is a fundamental component of this project and relied upon for executing all code. If there was a hardware issue or failure with the board, it has implications on both development time and cost of the project.
- This is mitigated by being able to order a replacement board through the school of EEECS or using an emulator to execute code.

Project delays:

- There could be delays due to unforeseen circumstances, such as technical roadblocks or personal emergencies.
- This is mitigated by scheduling the Gantt chart to finish approximately 1 month before the submission date for the project is, allowing for extra time to finish development.

Corrupted data:

- As the project code and all associated project documentation is being stored on a hard drive, there is the potential for corrupted data if there is an error with the device being used for development.
- This is mitigated by having regular commits to the GitLab repo and having all worked backed up to OneDrive on a regular basis to provide redundancy.

References:

[1] Buchanan, W. and Woodward, A., 2017. Will quantum computers be the end of public key encryption? Journal of Cyber Security Technology, 1(1), pp.1-22.

[2] F. Richter, "The Post-PC Era Has Arrived," statista.com, chart 1, May. 7, 2012. [Online]. Available: https://www.statista.com/chart/276/global-tablet-smartphone-and-pc-shipments-from-2010-to-2016/. [Accessed Sept 20, 2023]

[3] Monz, T., Nigg, D., Martinez, E.A., Brandl, M.F., Schindler, P., Rines, R., Wang, S.X., Chuang, I.L. and Blatt, R., 2016. Realization of a scalable Shor algorithm. *Science*, *351*(6277), pp.1068-1070.

[4] National Institute of Standards and Technology, "Post-Quantum Cryptography" nist.gov, para 1, Jan. 3, 2017 [Online]. Available: https://csrc.nist.gov/projects/post-quantum-cryptography.[Accessed Sept 20, 2023]

[5] PQ-Crystals, "Cryptographic Suite for Algebraic Lattices" pq-crystals.org, para 1, Dec 23, 2020 [Online]. Available: https://pq-crystals.org/kyber/ [Accessed Jan 10, 2024]

[6] National Institute of Standards and Technology, "Lightweight Cryptography" nist.gov, para 1, Jan. 3, 2017 [Online]. Available: https://csrc.nist.gov/projects/lightweight-cryptography.[Accessed Sept 20, 2023]

[7] S. Segars, "Arm Partners Have Shipped 200 Billion Chips," arm.com, para 1, Oct. 18, 2021. [Online]. Available: https://www.arm.com/blogs/blueprint/200bn-arm-chips [Accessed Sept 20, 2023]