Minor Quantum Safe Cryptography — Software project

Introduction

In recent years, there has been rapid growth in technologies based on quantum principles. While many applications are not yet mature enough for industrial implementation, we are currently witnessing the emergence of the second quantum revolution. In the near future, quantum computers, quantum communication and quantum sensors have the potential to greatly expand our technological capabilities.

The new technologies will also pose a threat to data security. The cryptographic protocols that are currently used to secure our data, are designed to be robust against attacks by classical computers. However, once a substantial quantum computer has been developed, it can be used effectively to gain access to most encrypted data. Although such a quantum computer does not exist yet, the security thread is very relevant today. For example, one could store sensitive encrypted data now to unlock it at a later moment. Also, products developed today, might still be in service in the 'quantum future' and should therefore be quantum safe.

To oppose this thread to data security, two technologies can be implemented: post-quantum cryptography (PQC) and quantum key distribution (QKD). PQC is a set of new (classical) mathematical cryptographic algorithms that encrypt/decrypt data in a quantum safe way. QKD is used to share a symmetric key between parties, that can be used in a symmetric cryptographic algorithm. The key will be shared by sending quantum information. When using QKD, the interference of any eavesdropper trying to steal the key, can never be hided for the users, even in theory.

In this project, you will explore two PQC algorithms, evaluate their performance in terms of speed and memory usage, and determine their suitability for various scenarios. Additionally, for those aiming for extra credit, there is an opportunity to develop strategies to optimize these algorithms.

Assignment

The primary objective of this project is to provide hands-on experience with PQC algorithms, allowing you to understand the practical implications and performance characteristics.

In this six-week project, students will work in pairs to explore and evaluate Post-Quantum Cryptography (PQC) algorithms. Each duo will implement the Ring-LWE algorithm in Python. Afterwards, the duo improves the algorithm by implementing the NTT method. An analysis has to be performed on the performance of both implementations in terms of speed and memory usage. Make sure to think about how you implement the algorithms: security should not be compromised! Additionally, students can earn one extra bonus point (when at least for the rest of the project they score a final grade ≥ 5.5) by developing and implementing optimization strategies to enhance the algorithms' efficiency (again, without compromising the security). Deliverables include a project proposal, Python code, performance evaluation reports, and a final presentation summarizing their findings and conclusions.

By the end of the project, students will:

- Understand the fundamentals of post-quantum cryptography.

- Have practical cryptographic skills through Python implementation.
- Know how to evaluate and analyze the performance of cryptographic algorithms.
- Have enhanced collaboration, project management, and technical communication skills.

Submission

The completion of this assignment comprises three parts:

1. Before the end of week 1, you submit a brief proposal outlining a plan for implementation and evaluation of maximally one (substantive) page. This project proposal needs to be approved by the teacher in order to continue.
2. In week 5 you present your results during a 20 minute presentation. This will involve fellow students and Fontys teachers involved in this project. The presentation counts for 25% of the final grade. Make sure that you explain:
    - An explanation on how you implemented the algorithms
    - How you evaluated their performances and how you made sure the performances of the two algorithms were comparable
    - Conclusion based on your findings
    - Optionally: how one could further improve the performance of the algorithm(s)
3. Before the end of week 6, you submit your final report of maximally 5 (substantive) pages. The report counts for 75% of the final grade. The report should consist of the following parts:
    - Introduction: Make sure the introduction includes an assignment description that clearly articulates what you think the questions and expectations are
    - Explanation of the chosen algorithms: describe briefly the algorithm you have implemented (and what the NTT method adds to it) and mention the most relevant theoretical aspects of them
    - Results: Explain how your code works. How did you implement the PQC algorithms and make evaluation possible? Show the results of the evaluation
    - Conclusion: try to put your work in practice and explain how this is relevant in terms of context!
    - Optionally: how could one further improve the performance of the algorithm, without compromising the security?

Assessment

The assessment of this assignment is described in the assessment form.