Post-Quantum Cryptography: Delving into Lattice-Based Methods

The rise of quantum computers poses a significant threat to current public-key cryptography. Shor's algorithm can efficiently break widely used schemes like RSA that rely on the difficulty of factoring large numbers or solving the discrete logarithm problem. Post-quantum cryptography (PQC) emerges as a critical response, offering new algorithms resistant to both classical and quantum attacks.

Lattice-Based Cryptography: A Mathematical Stronghold

One of the most promising approaches in PQC is lattice-based cryptography (LBC). LBC leverages the mathematical properties of lattices, which are discrete, point-like structures in high-dimensional space. These lattices offer a rich playground for constructing secure cryptosystems due to the inherent difficulty of certain lattice problems.

Core Lattice Problems and their Power

- Shortest Vector Problem (SVP): Given a lattice, finding the shortest non-zero vector within it is believed to be computationally intractable, even for quantum computers. LWE (Learning With Errors) is a fundamental LBC scheme that relies on the hardness of the SVP. In LWE, small random errors are added to lattice points, making it challenging to distinguish them from the actual lattice vectors. This forms the basis for secure encryption and key exchange.
- Closest Vector Problem (CVP): Similar to SVP, CVP seeks the closest lattice vector to a given target point. Code-based cryptography (another PQC approach) utilizes the hardness of CVP to design secure cryptosystems.

Lattice-Based Cryptosystems in Action

Several promising LBC cryptosystems are emerging from ongoing research. Here are a couple of notable examples:

- CRYSTALS-KYBER: This is a finalist in the NIST PQC standardization process. It offers a key encapsulation mechanism (KEM) for secure key exchange, relying on the Ring-LWE variant. Ring-LWE leverages polynomial rings over finite fields, making it more efficient than traditional LWE.
- CRYSTALS-Dilithium: Another NIST finalist, Dilithium is a digital signature scheme based on LWE. It provides strong security guarantees and achieves high efficiency compared to other lattice-based signature schemes.

Advantages and Considerations for LBC

LBC offers several advantages:

- Provable Security: The security of LBC schemes can be formally reduced to the hardness of well-studied lattice problems like SVP and CVP.
- Flexibility: LBC can be used to construct various cryptographic primitives like encryption, key exchange, and digital signatures.
- Standardization Efforts: The inclusion of LBC schemes in the NIST PQC standardization process paves the way for wider adoption

NEXT STEPS:

- We seek to implement Super Singular Isogeny key exchange(SIKE) one of the initial methods used in post quantum cryptography
- Following that we seek to implement the GGH algorithm, explore its primary failures and understand/implement the current improvements that the lattice based methods employ in the NIST organised competition.
- We then seek to understand and implement Learning with errors.
- After this we seek to explore other methods of post-quantum cryptography and report on them depending on availability of time. The idea is to compare and contrast other ways of creating np hard problems for quantum computers and also understanding non-lattice techniques like singular-isogeny, code-based, hash based or multivariate .
- Following the report we seek to either propose a possible improvement on one of the lattice or non lattice based technique’s algorithms or alternatively pick a novel domain of mathematically complex problems and use them to make a simple encryption algorithm that confirms to standards of post-quantum cryptography, a possible domain that we liked and found interesting was chaos modelling and chaos based cryptography.
- The implementation of some lattice based techniques and the algorithmic understanding of other domains and possible implementation should give us a comprehensive understanding of the subject of post-quantum cryptography.