**Abstract**

Quantum computers provide many new opportunities for cryptography, such as key distribution, confidentiality, integrity, and non-repudiation. Because of the destructive and probabilistic nature of quantum measurements and the no-cloning theorem which prevents the duplication of quantum states, quantum cryptographic protocols are dramatically different from classical cryptography. A large number of these protocols have been proposed, but little comparative work has been done. We perform a survey of existing protocols, with a focus on practical applications, in order to provide an overview of the current state of the field. We also present an implementation of a practical quantum cryptographic algorithm capable of providing confidentiality, integrity, and non-repudiation. In addition, we generalize our implementation process in order to provide a template for the translation of algorithms from a research paper into a quantum programming framework.

**Outline**

1. Introduction
   1. General Introduction
   2. Background & Problem Context
   3. Motivation
      1. Gaps that currently exist, how the literature has evolved over time, what we're actually talking about in this paper
      2. Related topics which are out of scope of this paper
   4. Research Questions
      1. **Research Question 1:** What are the impacts of quantum computing on today's commonly used cryptographic protocols for hashing, symmetric, and asymmetric encryption? Which of these protocols or classes of protocols need to be replaced by quantum or post-quantum solutions?
      2. **RQ2:** What are the most promising techniques, either quantum or a hybrid of classical and quantum techniques, for maintaining the confidentiality of data at rest and in motion?
      3. **RQ3:** What are ..., for verifying the integrity of data received?
      4. **RQ4:** What are ..., for ensuring the authenticity and non-repudiation of messages?
      5. **RQ5:** What standard process can be followed to translate a theoretical quantum algorithm into an implementation suitable for hands-on testing?
2. Related Work
   1. (Basically recycling our literature review and chunking out into categories - port over, categorize, and refine)
   2. Impacts of quantum computing on classical cryptography - Answers RQ1
      1. Shor's algorithm impact on RSA and digital signatures
      2. Grover's impact on AES and symmetric algorithms
      3. Impact on hashing functions and authenticated encryption
   3. Foundational papers in the different areas of quantum cryptography (like the original BB84 paper etc)
   4. Attacks against quantum cryptography techniques
      1. Photon splitting attacks
      2. Denial of service
      3. Man in the middle
      4. Any others uncovered in final literature review
   5. Include research methodology background for similar studies comparing different cryptographic algorithms
      1. Found two different papers which fall into this category, can discuss here (Jorstad and Khan)
   6. Post-quantum cryptography
      1. NIST competition
      2. Promising algorithms must not fall into BQP class (Bounded-error Quantum Polynomial)
3. Methodology
   1. Overview of methodology as described in technical plan
      1. How algorithms were selected
      2. How algorithms were classified
         1. Different classification scheme was used for each intended function
         2. Algorithms may appear in multiple categories if they provide multiple attributes e.g. both confidentiality and integrity
         3. Explanation of criteria and why each was selected - TODO for team: Fill these in
            1. QKD criteria - <https://github.com/Vidmaster/cybr8950-quantum/blob/main/QKD%20Protocols.md>
            2. Confidentiality criteria

Application

Target data

Key agility

Resource requirements/limitations

* + - * 1. Integrity criteria

Verifiable

Reliable

One transmission or multiple

Security principle (Hashing, Encryption, Digital Signature, etc.)

Target (classical or quantum data)

* + - * 1. Non-Repudiation criteria

Blind or arbitrated

Reusable

One transmission or multiple

Security principle (rotation, hashing, etc.)

Target (classical or quantum data)

Signature length vs. message length

* + 1. How algorithms were evaluated as implementation candidates
       1. Level of overall complexity
       2. Practical applicability to research questions
  1. Selection of algorithms
     1. QKD - <https://github.com/Vidmaster/cybr8950-quantum/blob/main/QKD%20Protocols.md>
     2. Confidentiality
     3. Integrity - As Quantum computing grows the integrity of classical computing shrinks. RSA public-key asymmetric encryption algorithm is especial vulnerable to Quantum Computing. A strong contender for securing Integrity in Public and Private key sharing in Classical Computing is NTRUEncrypt. NTRUEncrypt is similar to RSA in that it can be used in both encryption and signature. NTRUEncrypt uses a lattice-based public-key and is very resilient against Quantum Computing, at this time. NTRUEncrypt is also less computational intensive as RSA when using 8-bit AVR Micro controllers. Quantum computing can ensure its Integrity by using QKD if transmitting data. To insure integrity on the Quantum Computer Zero Knowledge can be used. Zero Knowledge is Scalable verification of computational integrity over confidential data sets. Zero Knowledge uses a proof and argument system S=(P,V). S is the soundness or correctness if you will. P is prover of athe algorithm and V is verifier. This system allows the Quantum computer to generate its own check for integrity. Zero Knowledge is based of the PCP theorem, constraint satisfaction NP-hard for maximum fraction of constraints within some constant.
     4. Non-Repudiation
        1. Curty 2001
        2. Kang 2015
        3. Nikolopoulos 2020
        4. Hematpour 2020
  2. Classification
     1. Envisioning this section as basically just being several tables accompanied by discussion
  3. Evaluation
     1. Implementation
     2. Applicability to hypothetical use case
     3. Ways to evaluate against possible attacks (if we get that far)
  4. Revisit research questions and relevance to the methodology. If RQs overlap, can keep this structure, otherwise consider having a top level for each RQ and then go from there. Which part of our methodology addresses each question?

1. Analysis & Results
   1. Sample implementation of a selected algorithm
      1. Challenges encountered during implementation
      2. Limitations of our implementation
      3. How this implementation could be used in practice
   2. General formula for implementation of quantum algorithms - Answers RQ5
      1. How to identify an algorithm and turn it into a prototype implementation using Qiskit or similar
      2. Limitations of this approach - what it works for, what it doesn't work for
      3. Here’s a workflow that someone could follow to take a mathematical implementation from a research paper and turn it into an implementation using Qiskit or another framework.
      4. Here’s an example we did following these steps, here are the challenges we encountered, insight into how to translate other frameworks besides Qiskit
   3. Tie back to research questions - what's our answer, and how did we get there?
      1. RQ2 - Confidentiality
         1. Have found a limited number of quantum algorithms which provide confidentiality. Quantum data can be "encrypted" using a one-time pad approach by applying a pre-shared set of random rotations to qubits before transmission, and removing them after. Quantum encryption with a classical key is also possible through quantum teleportation, as entangled qubits can be shared and then an arbitrary state teleported by transmitting two classical bits.
         2. Quantum encryption of data at rest is not a relevant concern. Due to the no-cloning theorem and the destructive nature of quantum measurements, as well as the current short-lived nature of quantum data, qubits can be treated as always being "in motion", and separate encryption schemes are not necessary.
         3. Post-quantum cryptography will be useful in the future, as these algorithms can be applied today with classical computers to protect data.
      2. RQ3 - Integrity
         1. Zero-Knowledge quantum integrity may be most promising
      3. RQ4 - Non-Repudiation
         1. Challenging due to the impossibility of verification without a third party
         2. Signatures are frequently not reusable
         3. Can provide value in some cases, but most promising approach is to use post-quantum signature techniques
2. Conclusion
   1. General conclusions
      1. Revisit research questions and goals
         1. RQ1 - replace vs. strengthen vs. quantum solution
         2. RQ2-4 - most promising protocols if applicable
         3. RQ5 - framework for implementing quantum algorithms
      2. Recommendations for today
         1. Gain an awareness of quantum computing, what it can and can't be used for, and how it threatens current systems
         2. Be prepared to implement post-quantum encryption once it has been approved by NIST
         3. Data transmitted today with insecure encryption could be intercepted and stored until quantum computers are viable, and then decrypted by an adversary.
      3. Recommendations for 10+ years
         1. Expect to see quantum computing become much more widespread, and start to be applied for simple cases like entropy generation, key distribution, and optimization problems
         2. Be prepared to react to new algorithms being discovered which threaten previously-secure cryptosystems.
   2. Future research
      1. Encryption of quantum data has not been well explored
      2. Other gaps we uncover during final preparation