A Survey and Simulation of Three Quantum Key Distribution Protocols

Conner Taylor
cotaylor@mines.edu

***Abstract*—Private-key ciphers such as the One Time Pad are the only cryptographic systems with mathematically proven security, even against an adversary using a quantum computer. However, the One Time Pad is rarely used in practice due to the difficulty of secretly generating and distributing the long, random keys it requires. Quantum key distribution algorithms exploit the physical properties of quantum bits to provide a method for two parties to establish a shared key with guaranteed security. This paper will examine three of the most common protocols for quantum key distribution and provide a basic simulation and analysis of each.**

I. Introduction

The infeasibility of large-scale key distribution for private-key cryptography has led to the widespread adoption of public-key ciphers such as RSA, which are not quantum-safe. The most secure public-key ciphers today can be broken in polynomial time by an opponent with sufficient quantum computing ability. Quantum key distribution (hence QKD) protocols attempt to improve the usability of private-key schemes such as the One Time Pad or AES by providing a secure method for generating shared keys over public channels. After the two communicating parties perform classical techniques such as error correction and privacy amplification, the key material can be used in any private-key cipher.

QKD protocols exploit three physical laws governing quantum bits such as polarized photons or spin-$\frac{1}{2}$ particles to achieve guaranteed security:

1) It is impossible to duplicate an unknown quantum state without measuring it
2) Measuring a quantum bit necessarily disturbs its state
3) It is impossible to measure a quantum state in non-compatible bases simultaneously

These properties make it impossible for an eavesdropper to gain information about a quantum key without changing it. We assume two parties, Alice and Bob, wish to establish a shared key while minimizing their mutual information with an eavesdropper, Eve. Even if Eve is allowed to intercept and modify Alice's and Bob's communication over the quantum channel, she will be unable to conceal her measurements and will be detected through classical error analysis. We assume that Eve is also capable of intercepting the classical communication between Alice and Bob, but that she is unable to alter their messages in any way. Therefore, as long as Alice and Bob use an authenticated classical channel and perform error correction and privacy amplification on their keys, Eve will be unable to gain any useful information.

This report will focus on three major protocols for quantum key distribution, namely the BB84 protocol proposed by Bennett and Brassard[1], the B92 protocol proposed by Bennett[2], and the E91 protocol proposed by Ekert[3]. Section II will contain background material on classical cryptography and quantum bits as well as the motivation of this project. Section III will describe the three chosen protocols and compare their strengths and weaknesses. Section IV contains a description of the three simulations, their corresponding unit tests, and results. Section V will discuss the current state of QKD and an alternative six-state protocol. Solutions to selected exercises from Nielsen & Chuang's *Quantum Computation and Quantum Information* as well as the source code for the simulations will be included in the appendix.

II. Background

A. Classical Cryptography

Most electronic communications today are encrypted using a public-key cipher such as RSA[4]. Public-key algorithms are convenient for everyday use because, unlike private-key systems, they do not require a unique shared key for each pair of users who wish to communicate. Instead, each user possesses a private key, which is used for authentication and digital signing, and a public key, which provides confidentiality. As a result, for a system with n users, any pair of whom wish to communicate, a public-key system only requires $n(n-1)$ total keys, $2n$ of which are unique. For comparison, a similar system using private-key cryptography would require $n(n-1)/2$ unique keys (see Appendix A1). For example, a system with 100 users would require 200 unique keys using RSA and 4950 unique keys using a private-key cipher.

However, the limitations of public-key cryptography lie in its reliance on problems which are difficult but not computationally infeasible. The most commonly used public-key cipher, RSA, relies on the difficulty of factoring composite integers[4]. Another type of public-key system, the elliptic curve cipher, relies on the difficulty of computing discrete logarithms[5]. These systems are only considered secure because no polynomial-time algorithm has been

found to solve either of these problems. It has been shown that a network of computers can solve RSA classically in sub-exponential time[6], and the lower bound on time to solve these problems has not been proven. Additionally, n attacker with a sufficiently complex quantum computer can break RSA and elliptic curve ciphers in polynomial time[7]. Although a quantum computer capable of performing Shor's algorithm on 2048-bit RSA keys is not yet practical, advances such as topological quantum computing may render public-key cryptography obsolete in the near future.

Unlike public-key schemes such as RSA, private-key ciphers require the generation of a new key for every pair of communicators. While this requires a much larger amount of key material, the resulting ciphertext is more secure as it does not rely on the infeasibility of solving difficult problems. Instead, the One Time Pad cipher guarantees that the ciphertext is unbreakable through either brute force or cryptanalysis. The One Time Pad is one of the simplest examples of a private-key cipher. It requires a truly random key at least as long as the message to be encrypted. To encrypt, the sender adds the key (modulo 2) bitwise with the message, and as long as the sender and recipient have the same key the recipient can perform the same operation to decrypt. This cipher is perfectly resilient to brute force attacks, since any number of valid plaintext messages can map to the same ciphertext. Unless the attacker learns the key, it is impossible to determine which plaintext is the original message. Other private-key ciphers such as Triple DES and AES use key expansion to reduce the amount of key material required while maintaining nearly perfect security.

In practice, private-key systems are seldom used due to limitations of classical key distribution:

- Keys must be truly random, as defects in pseudo-random number generators can result in low key entropy[9].
- Keys must be exchanged in secret, classically requiring a face-to-face meeting.
- Keys must be guarded until use and destroyed afterwards.

In order for Alice and Bob to regularly exchange encrypted messages, they would have to meet in secret and generate terabytes of key material each meeting. Alternatively, they could rely on a trusted third party to generate the key material and distribute it to both of them. However, this would mean that the key is not known only to Alice and Bob, intoducing a new potential attack vector and making the key unusable for digital signing. QKD protocols address these limitations of private-key cryptography by providing a method for Alice and Bob to generate truly random, shared key material over long distances, even in the presence of eavesdroppers.

B. Quantum Encoding

Part of the security of QKD arises from the fact that neither party intends to use any specific key at the outset. Instead, the key is generated truly randomly from quantum mechanical phenomena such as thermal noise[8]. In most QKD protocols, we assume Alice begins by generating a long string of truly random classical bits. Her goal is to encode this classical information in the states of quantum bits which are subject to the physical laws discussed in Section I.

C. Motivation

The BB84 protocol for quantum key distribution was originally only proven to be secure against specific classical attacks. However, the algorithm has since been proven to be secure against all possible eavesdropping attacks, including those utilizing the properties of quantum computers[7]. The widespread implementation of a secure QKD protocol such as BB84 would greatly enhance the security of private-key cryptosystems, as such a protocol resolves the major weakness of private-key systems: secure key distribution.

The use of quantum key distribution over conventional key distribution will play an important role in future cryptosystems. Currently, conventional key distributions relies on the trust of third parties to securely distribute keys and ciphers to adequately encrypt messages that are assumed to be infeasible to decrypt with brute-force. Quantum key distribution solves this problem by allowing the generation of key material independently and has the equivalent security of One Time Pad. Research in this area could lead to a larger shift towards this technique with timeless security.

D. Goals

We approach this study with three objectives:

1) Examine, compare, and explain current QKD systems.
2) Compare advantages and disvantages of QKD to conventional key distribution.
3) Show what is required to make QKD practical and more widely used.

III. Approach

A. Survey

In order to show that quantum key distribution systems will provide more secure encryption than conventional key distributions we will need to perform extensive and in-depth research into both quantum key distribution and conventional key distribution methods. We will explore the weaknesses inherit in the conventional key distributions and attempt to address ways that quantum key distribution will mitigate those weaknesses. In order to best show that quantum key distribution will play a huge and important role in future cryptography we will need to examine the strengths present in this method and compare them to the strengths that are present with conventional key distributions. We will also need to show that the strengths of quantum key distribution will more than make up for any weaknesses we find in quantum key distribution during our research.

IV. Deliverables

A full report containing an in depth analysis of QKD will be provided along with a brief presentation of the research. These will contain an overview of the most common QKD systems and the math and physics behind their uses, a comparison to conventional key distribution systems and where QKD could be used in their place will also be provided, and an explanation on where current and future research in QKD will be heading.

V. Conclusion

Classical private-key encryption systems are only secure as long as the keys used are exchanged in advance, protected until they are needed, and then destroyed. These limitations are the reason why the key distribution problem is one of the biggest drawbacks of private-key cryptosystems. Quantum Key Distribution protocols provide a possible solution to this problem by exploiting the physical properties of small particles, which comprise the 'qubits' of a quantum computer.

Because the quantum state represented by a qubit cannot be observed or copied without disturbing the state, an eavesdropper cannot intercept a key transmitted through a quantum channel without modifying the keystring. By performing a survey of several implementations of quantum key distribution protocols, we will attempt to determine the security advantages and disadvantages QKD offers compared to classical key distribution techniques.

References

[1] C. H. Bennett and G. Brassard, "Quantum Cryptography: Public Key Distribution and Coin Tossing," Intl. Conf. on Computers, Systems, & Signal Processing, 12 Dec. 1984.

[2] C. H. Bennett, "Quantum Cryptography Using Any Two Nonorthogonal States," Phys. Rev. Lett., Volume 68, Number 21, 25 May 1992.

[3] A. K. Ekert, "Quantum Cryptography Based on Bell's Theorem," Phys. Rev. Lett., Volume 67, Number 6, 5 August 1991.

[4] R. Rivest, A. Shamir, and L. Adleman, "On Digital Signatures and Public Key Cryptosystems," Commun. Ass. Comp. Mach., Volume 21 (1978) pp. 120-126.

[5] M Rosing, "Implementing Elliptic Curve Cryptography," Manning Publications, Greenwich (1999) ISBN 1-884777-69-4.

[6] E. W. Weisstein, "RSA-640 Factored," MathWorld Headline News, 8th November (2005), http://mathworld.wolfram.com/news/2005-11-08/rsa-640/.

[7] P. Shor, "Polynomial-time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," Siam Journal on Computing, Volume 26, Issue 5 (1997) pp. 1484-1509.

[8] N. Heninger, Z. Durumeric, E. Wustrow, J. A. Halderman, "Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices," Proc. 21st USENIX Security Symposium, Aug. 2012.

[9] B. Jun, P. Kocher, "The Intel Random Number Generator," Cryptography Research, Inc. white paper prepared for Intel Corp., 22 Apr. 1999.

[10] M. Nielsen and I. Chuang, "Quantum Computation and Quantum Information," Cambridge UP, 2015. Print.