The Impact of Quantum Computing on Cryptography

Tara O'Kelly, *(Hons) Software Development, GMIT*

Abstract—With the arrival of quantum computing, one must acknowledge the security implications of quantum computers. The integrity of many modular cryptographic systems, namely discrete logarithm and factoring based systems [1], are at risk. Examining the pre-quantum state of cryptography, we will discuss the predicted affect of quantum on cryptography and potential collateral ramifications. Although there are quantum computers that exist today, all are far from capable of performing operations complex enough to break cyptographic algorithms widely used today. The attempts of post-quantum cryptography are in contention with the emerging technology, pursuing to implement a *feasible*, *flexible* and *computationally efficient* solution before a pertinent quantum computer can be built.

I. INTRODUCTION

The continuous growth in the industry of technology reaps many upsides. Yet the progression in one area, can lead to the retrogression of another. A prime example being the advancements in the field of quantum computing. Quantum algorithms have been found to solve certain complex problems abundantly faster than that of classical algorithms, benefiting various areas like machine learning [1] and developing new drugs [2]. However, some of these algorithms could also aid the devaluation of cryptographic systems in our digital entrenched world. The suggestion of the succeeding corruption of privacy is distressing, possibly catastrophic.

The following review is not meant to provide in-depth understanding quantum computing. Nor is it intended to be a thorough explanation of current cryptographic algorithms, but rather a basic understanding of their workings and where quantum algorithms compromise them. Subsequently, we will delve into viable post-quantum solutions proposed in attempts to conserve secure communications.

II. INTRODUCTION TO CRYPTOGRAPHY

Cryptography is a method of protecting data from people who are not authorized to access it. Encryption, a significant mechanism in cryptography, is achieved by transforming plain-text into ciphertext, with the intent of rendering it meaningless to those who do not have the classified resources. With these resources, the ciphertext is usually transformed it back to it's original state, a process called decryption.

Symmetric-key encryption utilizes the same key to both encrypt and decrypt, Advanced Encryption Standard (AES) being the prevailing form. Symmetric encryption is more accomplished in achieving a faster outcome than that of asymmetric. Problematically, the sender and the receiver must hold the same key. How do they exchange keys digitally without another party eavesdropping? A typical solution would be to use asymmetric encryption to exchange symmetric keys.

Asymmetric-key encryption (a.k.a. public-key cryptography) practices an encryption technique with two differing, math-ematically linked keys. The first key being a public key and the corresponding key being a private key - e.g. Rivest Shamir Adleman (RSA), Elliptic Curve Cryptography (ECC). As implied in it's name, the public key is to be publicized. The private key is to be kept secret. The public key can be used to encrypt data, whereas the homologous private key can be used to decrypt the data. An ideal public-key cryptographic algorithm should serve as a trapdoor function. A trapdoor function is a function that is easy to perform one way, but has a secret that is required to perform the inverse calculation efficiently. The objective is to decrease the probability that the secret could be identified as much as possible.

So far, we have acknowledged AES, RSA and ECC; all can be adopted in pursual of a **confidential** message. Cryptography is also concerned with proving the **integrity**, **authenticity** and **non-repudiation** of a message e.g. Message Authentication Code (MAC), Digital Signatures. Another method attempting the assurance of integrity could be by hash functions, e.g. Secure Hash Algorithm (SHA).

A. Rivest Shamir Adleman

Rivest Shamir Adleman was first publicized in 1978 by Ron Rivest, Adi Shamir and Leonard Adleman. The RSA algorithm is the most popular and perhaps the best understood public key cryptography system. RSA's security derives from the difficulty of factoring two large integers and the expeditious multiplication to get these large numbers; it is a befitting example of the "trapdoor" methodology.

Two prime numbers, p and q, are generated. The values p and q are multiplied together to get the maximum value, n. A number pub is selected, such that pub is not a factor of $(p-1)$ and $(q-1)$. Born from these values is the number $priv$. The public key (pub, n) can be released, but the private key $(priv, p, q)$ must be kept confidential. As long as you know the values p and q, you can compute the corresponding private key from pub, explaining how factoring relates to breaking RSA. Factoring the maximum number into its component primes allows you to compute someone's private key from the public key and decrypt their private messages.

B. Elliptic Curve Cryptography

Although Elliptic Curve Cryptography was originally pro-posed in 1985 by by Neal Koblitz and Victor S. Miller, it was not widely utilized until the 21st century. It is not as widely understood as RSA, with it's complexity to blame. ECC allows the use of smaller keys than RSA to get the same levels of security.

ECC is based on the algebraic structure of elliptic curves over finite fields [3]. ECC handles the following domain parameters: (p, a, b, G, n, h). To briefly explain the parameters, p is the field that the graph is defined over, the variables a and b are values that define the curve, G is known as the generator point (a.k.a. base point), n is the prime order of G and h is the cofactor of the curve.

The private key d is a randomly selected integer in the interval $[1, n - 1]$. Subsequently, the public key $Q = dG$. The security of ECC is built upon the Elliptic Curve Discrete Logarithm Problem (ECDLP) [3]; ECDLP in ECC applies to the laborious task of locating the discrete logarithm of random elliptic curve element even with a known point. With the given the domain parameters and Q, ECDLP refers to the problem of determining d. In a real world standard application, it would be unfeasible to check all the possibilities of d.

III. QUANTUM COMPUTING

blah blah blah

A. Shor's Algorithm

Shor's Algorithm has a severe impact on cryptography, potentially breaking the public key cryptography systems used today [4]. In 1994, Peter Shor established a quantum algorithm to find the prime factors of an integer in polynomial time ($O(N^c)$ time) as opposed to the exponential time ($O(c^N)$ time) of a classical computer with the same objective. The proficient factorization can be obtained as all of the values of a certain function can be computed by applying quantum parallelism and entanglement [5]. Shor has also released an adapted version that can solve the discrete logarithm problem [6], damaging the impregnability of the discrete logarithm problem and ECDLP.

Public key encryption and digital signatures such as the favored algorithms RSA, ECC, Digital Signature Algorithm and Elliptic Curve Digital Signature Algorithm are vulnerable in this scenario. With the corruption of public key encryption comes the simultaneous weakening of single key encryption by straining the means to disclose symmetric keys.

B. Grover's Algorithm

Grover's Algorithm was formulated by Lov Grover in 1996. This algorithm is a strong foundation on which lot of the superior applications of quantum computing can be built [6]. Harmful affects by the algorithm in question can be seen in symmetric encryption and hash functions, including the ever popular AES and the SHA family. It important to note that Grover's algorithm only decreases the security of these cryptographic methods; it does not break them.

A classical computer can do a search with the speed of $O(N)$ assuming we do not know if the search parameter exists in an unordered domain. Grover's algorithm suggests that the same search can be implemented in $O(\sqrt{N})$ time using quantum queries. Grover's acceleration from $O(N)$ to $O(\sqrt{N})$ is not quite as calamitous as Shor's. However, indicates another significant quantum approach to cryptanalysis [7], the study of code breaking.

C. Security Implications

The exertion of the quantum algorithms above on a capable quantum computer has the power to revolutionize cryptanalysis. Fortunately, no quantum computer is qualified to do so yet. As Bacon and Dam stated, "We can communicate securely, today, given that we cannot build a large scale quantum computer tomorrow" [4]. This extensive advancement in cryptanalysis will presumably open the door to a copious amount of privacy issues. Under the assumption of a relevant quantum computer, an unauthorized third party could eavesdrop on transactions and even manipulate said transactions. Additionally, this third party could easily impersonate others - even trusted sites, with the debilitation of digital certificates. The cryptographic principles confidence, integrity, authentication and non-repudiation would all be put in jeopardy. Many types of digital transactions would be called into question; personal banking, online purchases and private messaging to name a few. The immense volume of private information accumulated in our cloud based world, such as the vast amount of personal information submitted to social networks and medical records, would no longer be secure.

The variety of reasons to motivate malicious hackers is extensive. For example, a company may be seeking to gain an unfair advantage over competitors or a government may wish to monitor transactions to identify and combat threats. A government's acquisition of a quantum cryptanalysis system can be seen as a compelling concern [1]. The exploitation of citizen privacy may occur. The NSA's leaked documents revealed PRISM, a strict surveillance program, that can be referenced in this case. PRISM achieved extreme monitoring with the cooperation of service providers. A program such as PRISM can lead to a harmful surveillance society, possibly resulting in citizen unrest and mistrust of their governing authority. In a drastic instance, global tensions could be escalated by increasing a governments ability to spy on another. Whether it be by discovering another administration's private information, or the detection of another administration observing their state or their civilian's private information. Of course it is not only powerful entities, like a sizable company or a governing body, with the means to develop or obtain a quantum computer that are applicable to be considered a threat. One does not have to own a quantum computer, but could rely on a service provided by another party that possesses a quantum computer.

IV. POST-QUANTUM CRYPTOGRAPHY

The security implications of quantum computers has seen to the investigation of quantum resistant cryptography. Given that no relevant/capable CQ has been made yet - we still got time fam.

MAC still safe wahey

V. CONCLUSION

According to Kudelski there is no need to worry about QC breaking current cryptographic systems for years to come; cryptographers have time to create an optimal solution [8]. However, one must also consider the time costs of integration and standardization. It could take companies years blah blah blah.

APPENDIX A
PROOF OF THE FIRST ZONKLAR EQUATION

Some text for the appendix. This is obvious.

APPENDIX B
PROOF OF THE FIRST ZONKLAR EQUATION

Some text for the appendix.

ACKNOWLEDGMENT

The authors would like to thank...

REFERENCES

[1] A. Majot and R. Yampolskiy, "Global catastrophic risk and security implications of quantum computers." *Futures*, vol. 72, no. Confronting Future Catastrophic Threats To Humanity, pp. 17 – 26, 2015. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=edselp&AN=S0016328715000294&site=eds-live

[2] E. Conover, "Quantum computers get real." *Science News*, vol. 191, no. 13, pp. 28 – 33, 2017. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=ukh&AN=123749408&site=eds-live

[3] D. Verma, R. Jain, and A. Shrivastava, "Performance analysis of cryptographic algorithms rsa and ecc in wireless sensor networks." *IUP Journal of Telecommunications*, vol. 7, no. 3, pp. 51 – 65, 2015. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=108921659&site=eds-live

[4] D. BACON and W. VAN DAM, "Recent progress in quantum algorithms." *Communications of the ACM*, vol. 53, no. 2, pp. 84 – 93, 2010. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=47776287&site=eds-live

[5] M. A. Wright, "Feature: The impact of quantum computing on cryptography." *Network Security*, vol. 2000, pp. 13 – 15, 2000. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=edselp&AN=S1353485800090279&site=eds-live

[6] D. J. Bernstein and T. Lange, "Post-quantum cryptography." *Nature*, vol. 549, no. 7671, pp. 188 – 194, 2017. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=cmedm&AN=28905891&site=eds-live

[7] Q. Zhou, S. Lu, Z. Zhang, and J. Sun, "Quantum differential cryptanalysis." *Quantum Information Processing*, vol. 14, no. 6, pp. 2101 – 2109, 2015. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=a9h&AN=102915247&site=eds-live

[8] J.-P. Aumasson, "Feature: The impact of quantum computing on cryptography." *Computer Fraud Security*, vol. 2017, pp. 8 – 11, 2017. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=edselp&AN=S1361372317300519&site=eds-live