# Post Quantum Computing

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

***Abstract*—The security implications of quantum computers has seen to the investigation of quantum resistant cryptography. 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 and the predicted affect of quantum on cryptography, we will discuss the need for post-quantum computing. Subsequently, we will delve into the viable solutions. 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 *efficient* solution before a pertinent quantum computer can be built.**

## I. Introduction

The exponential evolution and prosperity of technology has brought... The integrity of any modular cryptographic systems, namely discrete logarithm and factoring based systems, are at risk of a breach.

### A. The Art of Encryption

blah blah blah

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.

### B. Rivest Shamir Adleman

Rivest Shamir Adleman (RSA) 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, $n$. Select a number $pub$ to be the public key, such that $pub$ is not a factor of $(p-1)$ and $(q-1)$. Next, generate the private key $priv$, such that $(priv * pub) mod (p-1) * (q-1) = 1$. The message $M$ is converted to an integer $C$ by multiplying itself to the power of the public key, then employs a wrapping scheme to ensure $C > 0$ and $C < n$. The decryption is a similar process, the cipher $C$ is converted to $M$ by squaring itself to the power of the private key. A wrapping scheme is again implemented, ensuring $M > 0$ and $M < n$. As long as you know the values $p$ and $q$, you can compute a corresponding private key from this public key, 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.

### C. Elliptic Curve Cryptography

Although Elliptic Curve Cryptography (ECC) was originally proposed 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. Small keys are very beneficial nowadays, considering cryptography is often implemented on low powered devices, e.g. a mobile phone. While multiplying two prime numbers together is easier, when the prime numbers start to get very long, even just the multiplication step can take some time on a low powered device.

ECC is based on the algebraic structure of elliptic curves over finite fields [2]. 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); ECDLP in ECC applies to the laborious task of locating the discrete logarithm of random elliptic curve element even with a known point [1]. 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 $d$.

### D. Quantum Computing

blah blah blah

## II. Security Implications of Quantum Computers

blah

Does not have to have a QC but could be relying on a service provided by another party that has a quantum computer.

### A. Shor's Algorithm

blah

### B. Grover's Algorithm

blah

### C. Social Concerns

Hackers.

Governments:

Citizen Privacy - use NSA's leakage of PRISM program as example. Refer to [1].

Increase Global Tensions by increasing a governments ability to spy on another.

## III. Post-Quantum Cryptography

blah

### A. Classical Solutions

blah

*1) Ex 1:*

*2) Ex 2:*

### B. Quantum Solutions

blah

*1) Ex 1:*

*2) Ex 2:*

### C. Integration and Standardization

blah

## IV. Conclusion

blah blah blah

## Appendix A
## Proof of the First Zonklar Equation

Some text for the appendix. This is obvious[3].

## 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=truedb=edselp&AN=S0016328715000294&site=eds-live

[2] 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

[3] J. Doe, *The Book without Title*. Dummy Publisher, 2100.