Variable Word Length:
A Quantum-Proof Encryption Solution

Mitchell Palmer
Cyber Security Engineering
George Mason University
Fairfax, VA
mpalmer7@gmu.edu

Abstract -- **Quantum computing is coming; and, with it, the security of modern cryptography will be compromised. Technologies such as industrial control systems, banking, and smartphones depend on cryptography to ensure the confidentiality and integrity of operations. Any data sent over the internet relies on cryptography to ensure the confidentiality of information. In some cases, information is so sensitive that if the encryption is broken years in the future the results could be catastrophic. A post-quantum cryptographic solution is required today. Variable World Length is a proposed quantum-proof symmetric cryptography algorithm. It is low-labor, low-latency, and low-power; with a keyspace of 10^{511} in its simplest form (for comparison, AES-256 is 2^{256}). Variable Word Length performs bit manipulation to send "words" of varied length. Without the key, an attacker would need to brute force all possible combinations. Due to the design of the algorithm, some of these combinations will decrypt real plaintext that was not the encrypted message. Additionally, the keyspace can be increased without an exponential tax on computing resources.**

Keywords: *variable word length, post-quantum cryptography, quantum-proof cryptography*

1. INTRODUCTION

Quantum computing is coming; and, with it, the security of modern cryptography will be compromised. Everything from smartphones, to banking, to industrial control systems rely on cryptography to ensure the confidentiality and integrity of data. Online banking logins (and communications within/between banks themselves) are encrypted. Communication between a power plant and a substation is - hopefully - encrypted. The lack of secure cryptography would result in bank accounts being drained and a sudden reliance on candles. Hopefully, this fictional power plant is not nuclear. These are two of copious examples of how society relies on cryptography. Wi-Fi and cellular communications, for example, are a form of radio frequency communication. Anyone can, quite easily, tune in and collect data, though most communication is encrypted (e.g. HTTPS). The strength of modern-day cryptography is calculable and secure. In the case of AES-256 [17], the strongest supercomputers today would take longer than the lifetime of the universe to break it[1]. Modern cryptography is secure, for now. A functional quantum computer could utilize quantum-physical properties to provide a vast parallelism of computing power and break modern cryptographic algorithms. This paper makes no attempt to establish a timeline on when a quantum computer powerful enough to do so will exist, merely argue that (1) there is enormous monetary investment and progress in quantum computing, (2) when quantum computing arrives, it will be devastating to society if quantum-secure cryptography does not exist; (3) provide a solution called Variable Word Length (VWL). VWL is an extremely fast to compute, low-labor, low-power symmetric cryptographic algorithm that is quantum proof. The number of possible states of the algorithm at a base level is 10^{511} (for comparison, AES-256 is 2^{256}). VWL performs bit manipulation to send "words" of varied length. Without the key, an attacker would need to brute force all possible combinations. VWL provides (1) a larger keyspace, requiring a much more powerful quantum computer to discover the key; (2) the ability to increase the key size *without* an exponential tax on computing requirements; (3) an encryption design that lends itself to the "infinite monkey theorem" [18]: attackers will decode false, but legitimate, results.

[1] To date, the most powerful supercomputer is IBM's Summit, capable of performing 200 quadrillion Floating-Point Operations Per Second (FLOPS) [1]. With a generous estimate for only 1000 FLOPS required to check one AES-256 key, Summit could check 200 trillion keys/second. It would take roughly 1.8 * 10^{55} years to check every one of the 2^{256} possibilities, or 1.8 * 10^{54} years on average. The lifetime of the universe is estimated to be 13.8 billion years old [2].

2. QUANTUM COMPUTING & CRYPTOGRAPHY

Quantum Computing (QC) is the extension of classical computing to the use of quantum systems. Operating in a new physical realm, certain properties of quantum computers have no classical analogue [3]. These properties can be applied to grant a vast parallelism of computing power or to solve problems previously thought impossible by classical methods [4].

2.1 Quantum Computing Research and Investment

The idea of QC dates to 1980 [5], with the first proposed practical implementation occurring in 1993 [4]. Today, much progress has been made. In October 2019, Nature published a Google venture dubbed "Quantum Supremacy Using a Programmable Superconducting Processor" [6]. The paper claimed a functional quantum computer of 53 qubits. Likewise, many organizations are currently developing hardware for quantum computers, and can be found on *quantumcomputingreport.com* - which lists 79 organizations researching qubit technology, to date. Many prestigious organizations such as Microsoft, Oxford University, and MIT are pursuing this research [7]. A 2017 report lists $278 million USD in venture capital invested into quantum computing firms [8]. Russia has pledged ~$780 million USD in the next 5 years towards quantum research [9]. There is much research and investment across industry developing quantum computers.

In the above articles, journals, and papers (among others not listed) there was consistent contention over the progress and timeline of QC [10] [11] [12]. Unequivocally, quantum computers are coming. The debate is *when* not *if*. Massive investment across industry (corporations, private investors, nation states) has put enormous resources and some of the world's best minds on the problem. Whether the wait for functional QC is five, ten, or thirty years; QC will have a devastating impact. See below.

2.2 How Quantum Computing Breaks Modern Cryptography

First, a differentiation: symmetric and asymmetric cryptography. The basis for asymmetric cryptography (e.g. RSA, Diffie-Hellman) is the *integer factorization problem* which states that it is difficult for a classical computer to factor prime numbers. Shor's algorithm is a quantum algorithm that would make this factorization trivial [13]. Such modern asymmetric cryptography can be broken by Shor's algorithm in the time it would take someone with the key to merely encrypt it: seconds. While avenues are being pursued on asymmetric variations of VWL (as well as Kerberos-esque [14] implementations), the base implementation of VWL is symmetric.

The modern symmetric encryption standard is the Advanced Encryption Standard (AES) [17]. The quantum attack against symmetric key cryptography is Grover's algorithm, which reduces the work required to brute-force a key to be proportional to the square root of the keyspace (i.e. a quadratic speed up) [15]. For example, the keyspace of AES-128 is 2^{128}. A quantum computer that has implemented Grover's algorithm would reduce the key-space to 2^{64}, which is easily breakable by modern standards [16]. The security AES-128 provides is voided by QC. Likewise, AES-256 is reduced to a keyspace of 2^{128}. While this is currently out of reach of classical computing, the brute-force attack against symmetric cryptography is the *worst-case scenario* attack: it tries every possible option. The AES algorithm is incredibly robust against *classical computing* attacks. The Biclique technique is the only known *general attack* against AES, and only reduces the AES-256 key size to $2^{254.4}$ [16]. Comparatively, quantum attacks against AES are not heavily researched. Classical computing power is generally agreed to increase according to Moore's law [20]. If a quantum algorithm is discovered that speeds up an attack against AES, compounded with Grover's algorithm, the margin of safety before AES is broken is arguably low.

2.3 Importance of Cryptography

In 2018, the National Institute of Standards and Technology (NIST) released a study estimating a positive $250 billion economic impact from the development of AES (between 1996-2017). The estimated benefit-to-cost ratio for funding the AES program for the economy was 1,976 to 1 [21]. Cryptography protects everything from bank transactions, to social media, to online shopping. Therefore, there is a need to develop cryptographic systems that are secure against both classical and quantum computers that can operate with existing communication protocols and networks. VWL provides such a solution.

3. VARIABLE WORD LENGTH ALGORITHM DESCRIPTION

Variable Word Length (VWL) is a proposed quantum-proof encryption algorithm first designed by Brian Penny [19]. Its most basic implementation will be described below. The algorithm is simple in its function, which is a key factor in reducing the attack space. Four manipulations are performed in the following order: *split, pad, shuffle* and *multiply*. Each step is done at the bit level. In the base model, the keystore contains 255 words, with each word randomly assigned to be 1 - 22 bits in length. The number of words is called "N" and the max length of each word is called "CP". For simplicity, N=15 and CP=10 will be described here. Below, the first part of the keystore was generated according to the rules. 15 numbers were generated, each within the max bit length.

[7, 8, 6, 4, 4, 10, 4, 8, 6, 3, 5, 3 ,9, 2, 2]

Now, a sample message to encrypt will be chosen. The message is "Bob drove.". In binary, the bit representation of this message is:

```
01000010  01101111  01100010  00100000  01100100  01110010  01101111  01110110   01100101  00101110
```

3.1 Splitting (with simple pad)

The plaintext message is split by bit per the key. In the key, the first "word" is to be 7 bits long. The first 7 bits of the message are split off into their own "word". Then, the next 8 bits are split off into a "word" and so on and so forth. Each "word" is padded from the beginning of the "word" to a length of CP bits. In this case, the CP is 10. Each padded bit is represented as "-".

```
---0100001 --00110111 ----101100 ------0100 ------0100 0000110010 ------0011
--10010011 ----011110 -------111 -----01100 -------110 -010100101 --------11 --------0x
```

This example (by design) had a key perfectly sized to fit the message. For a message longer than a key, the key is simply repeated. For a shorter message, the rest of the key is unused (though in practical implementation this will never occur – see below). If a word-length of a key is misaligned with the end of the plaintext message (as with the above case "x" symbol) the *end* of the "word" is padded. This does not disrupt the encryption algorithm.

The equation to compute the keyspace for splitting the data is $P = CP^N$. This results in $P = 10^{15}$ possible keys, or $P = 22^{255}$ in the base model. Much larger than the 2^{256} possible keys of AES-256, though a large keyspace is useless unless the algorithm itself is secure – which is outlined below.

3.2 Padding

An additional key of N_2 words, each with size CP_2 is generated. For simplicity, N_2 and CP_2 will be set equal to N and CP - but this does not need to be the case. The second key's bits are padded to the start of each word. Each of these padded bits are generated to randomly be "0" or "1". In variable word length, each CP is referred to as a "cut-point" and $CP+CP_2$ is the final "word-length" (WL). The keystore holds the list of cut-point, with each item as $[CP, CP_2]$. Below is the generated second key and padded bits:

[9, 1, 6, 4, 6, 8, 8, 5, 4, 7, 0, 3, 7, 2, 3]

```
+++++++++---0100001   +--00110111   ++++++----101100   ++++------0100   ++++++------0100   +++++++0000110010
++++++++------0011   +++++--10010011   ++++----011110   +++++++-------111   -----01100   +++-------110   +++++++-
010100101   ++--------11   +++--------0x
```

Padded bits filled in:
1010100011111**0100001** 111**00110111** 1110010110**101100**
000111010**0100** 11010100100**0100** 1110100**000110010**
00001101010110**0011** 0101100**10010011** 00001011**011110**
01011010011101**111** 01001**01100** 1010101011**110**
0010011**010100101** 0001011011**11** 11001101110**0**1

The keyspace equation here is P = CP^N x CP_2^{N2}.

3.3 Shuffle

A third key groups the words and rearranges the groups - performing a true data shuffle. For example, the 15 words could be split into 3 equal groups A, B, C and rearranged to form B, C, A. The groups do not need to be equal or match the message size.

3.4 Prime Multiplication

Finally, each shuffled word group is multiplied by a prime number. A prime number allows for the reversibility of this step. The multiplication step prevents known-plaintext attacks. Without multiplication the message is lost within the bits, but unchanged. An attacker that has knowledge of some of the data being sent can locate the bits and work out a partial or full key. Multiplying the bits changes the "1's" and "0's" of the legitimate bits obscuring them and preventing this sort of attack. With Z possible prime numbers to multiply by, the keyspace can now be calculated. For simplicity the shuffle step will be left out of the calculation – each word being multiplied instead. The resulting smaller keyspace is can be calculated by the equation P = Z(CP^N x CP_2^{N2}). For z=10, N,N_2=255, and CP,CP_2=10, the keyspace is 10^{511}.

4. VARIABLE WORD LENGTH AS A QUANTUM-PROOF CRYPTOGRAPHY SOLUTION

As a symmetric algorithm, Grover's algorithm [15] applies; granting a quadratic speedup for a brute-force attack against VWL. The 10^{511} keyspace of VWL negates the effectiveness of Grover's algorithm: after it a classical computer would still need to brute force $10^{255.5}$ possibilities. VWL is well out of reach. Additionally, the infinite monkey theorem states that a monkey hitting typewriter keys at random will eventually produce the complete works of Shakespeare, given infinite time [18]. The message "Bob drove." has roughly a 1 in 10^{17} chance of being produced at a given time. Without knowing how much data is sent, which bits are real, if those bits are flipped, and $10^{255.5}$ possibilities for these bits to arrange - an attacker decoding "Bob drove." can never be sure if the message is correct. The plaintext could have occurred randomly in the data. Maybe the intended message was "Bob rode.". Both cases are entirely plausible and equally likely. Against unknown future machine learning or quantum computing algorithms, encryption scheme that allows for the decoding of fake but legitimate results is imperative.

4.1 How Variable Word Length Mitigates Vulnerabilities

To brute force VWL, an attacker would first need to guess how the encrypted stream is split into words. Then the attacker would have to guess the right set of prime numbers to divide each word by. They would then have to shuffle the words back in the correct order. Next, they would have to guess how many fake bits start each word. This leaves an attacker unsure how much data is legitimate or fabricated. Additionally, these parameters could be changed *in real time* by the defender, further complicating the ability to be confident in a decoded answer. Vulnerabilities in the algorithm have not been discovered, as of now. The modularity of the algorithm allows for a weakness to be patched in the case one is discovered. The "N" value can be raised, the cut-point adjusted larger or smaller, more shuffles and prime multiplications can be added; all with nominal tax on compute time.

4.2 Flaws of Variable Word Length

Two flaws in VWL were identified and outlined here. The first flaw of variable word length is the *increased bandwidth*. Adding bits to a message will, by definition, increase the size of the message (~200% increase in bandwidth, at base[2]). An enterprise implementing VWL to encrypt network traffic would need to support the additional bandwidth. Latency-wise, this is nominal (see next section). Secondly, the algorithm relies on *randomness*. True randomness is impossible to produce. If the randomness generator used in an implementation of VWL is predictable, it is possible that portions of the key could be derived. The algorithm assumes a strong method for producing randomness.

4.3 Implementations of Variable Word Length

A hardware implementation of VWL is currently being tested. The prototype was built on a starter Field Programmable Gate Array (FPGA) board[3] and end-to-end encrypts traffic between hosts over ethernet. Every bit that leaves a host is encrypted with VWL. The hardware implementation is useful because it requires no configuration from the user and no change in how they operate. It is protocol agnostic. If network traffic is already encrypted, VWL can run on top of the current methods to make them quantum-proof. For testing, the packet size was set to 256 bytes. After baseline tests, VWL was switched on and the latency was tested. *The average latency was calculated to be 0.04336ms.* Even as a prototype, the latency is miniscule. Multiple avenues were found to bring the latency down even further. Power analysis was conducted in Quartus[4] that recorded a total thermal power dissipation of 548.08mW. Once off a developer board, the power draw is expected to be small enough to Power over Ethernet (PoE) [22]. An enterprise server implementation (like Kerberos [14]) is also being tested. VWL does not need to be implemented in hardware, as a software version was developed alongside. Additionally, a successful key-change in real time was tested.

5. CONCLUSION

Quantum computing is coming. Google already claims a 53qubit system [6] and there is great investment in the market of quantum computing. Our reliance on technology as a society demands a need for secure cryptography. Everything from internet privacy, secure banking, and industrial control system security rely on cryptography. Quantum computing poses immense risk to the strength of modern cryptography. Shor's algorithm and Grover's algorithm denote quantum methods that could easily break modern cryptography. Quantum-proof cryptographic algorithms need to exist *before* quantum-computing exists, in order to ensure the public is protected. The above research aims to display how VWL is one such algorithm. Its many perks - simplicity, efficiency, strength, modularity - make it a notable algorithm for the future.

ACKNOWLEDGEMENTS

I would like to thank Brian for introducing me to VWL, allowing me to produce a paper on it, and his guidance on the function of the algorithm.

REFERENCES

[1] "New 200-Petaflops System Debuts as America's Top Supercomputer for Science," Oak Ridge National Laboratory, https://www.ornl.gov/news/ornl-launches-summit-supercomputer (2018).

[2] Ade, P. A. R. and Aghanim, N. and Arnaud, M. and Ashdown, M. and Aumont, J. and Baccigalupi, C. and Banday, A. J. and Barreiro, R. B. and Bartlett, J. G. and et al., "Planck 2015 Results," EDP Sciences, https://doi.org/10.1051/0004-6361/201525830 (2016).

[3] Brassard, G., Chuang, I., Lloyd, S., & Monroe, C., "Quantum Computing," Proceedings of the National Academy of Sciences of the United States of America, 95(19), 11032-11033, www.jstor.org/stable/46252 (1998).

[4] Li, S., Long, G., Bai, F., Feng, S., & Zheng, H. , "Quantum Computing," Proceedings of the National Academy of Sciences of the United States of America, 98(21), 11847-11848, www.jstor.org/stable/3056810 (2001).

[5] Dyakonov, Mikhail, "The Case Against Quantum Computing," IEEE Spectrum, https://spectrum.ieee.org/computing/hardware/the-case-against-quantum-computing (2018).

[6] Arute, F., Arya, K., Babbush, R. et al., "Quantum supremacy using a programmable superconducting processor," Nature 574, 505–510. https://doi.org/10.1038/s41586-019-1666-5 (2019).

[7] Finke, Douglas, "Quantum Computing Report," https://quantumcomputingreport.com/scorecards/qubit-technology/ (2020).

[2] A version of VWL with only 130% "bloat" (added bits, on average) is being tested.

[3] Terasic Cyclone V GX Starter Kit

[4] Quartus settings: ambient temp-21C, cooling solution-15mm heat sink, 200LFpM airflow.

[8] Gibney, Elizabeth, “Quantum Gold Rush: The Private Funding Pouring into Quantum Startups,” Nature - News Feature, https://www.nature.com/articles/d41586-019-02935-4 (2019).

[9] Schiermeier, Quirin, “Russia Joins the Race to Make Quantum Dreams a Reality,” Nature - News Feature, https://www.nature.com/articles/d41586-019-03855-z (2019).

[10] Computerphile, “128 Bit or 256 Bit Encryption – Computerphile,” https://www.youtube.com/watch?v=pgzWxOtk1zg (2020).

[11] Aaronson, Scott, “Happy New Year! My Response to M. I. Dyakonov,” https://www.scottaaronson.com/blog/?p=1211 (2019).

[12] Finke, Douglas, “The Case Against Quantum Computing - A Rebuttal,” Quantum Computing Report, https://quantumcomputingreport.com/our-take/the-case-against-quantum-computing-a-rebuttal/ (n.d.).

[13] Shor, P.W., "Algorithms for quantum computation: discrete logarithms and factoring," Proceedings 35th Annual Symposium on Foundations of Computer Science. IEEE Comput. Soc. Press: 124–134. (1994).

[14] “Kerberos Papers and Documentation,” Massachusetts Institute for Technology, https://web.mit.edu/kerberos/papers.html (2018).

[15] Grover, Lov., “A fast quantum mechanical algorithm for database search,” Proceedings, 28th Annual ACM Symposium on the Theory of Computing, (May 1996) p. 212 (1996).

[16] Hawthorne, D. S., “A quantitative study of advanced encryption standard performance as it relates to cryptographic attack feasibility. Available from ProQuest Dissertations & Theses Global,” https://search-proquest-com.mutex.gmu.edu/docview/2189849453 (2018).

[17] National Institute of Standards and Technology, “Announcing the advanced encryption standard (AES),” (FIPS PUB 197), http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf (2001).

[18] Émile Borel, "Mécanique Statistique et Irréversibilité," J. Phys. (Paris). Series 5. 3: 189–196. (1913).

[19] Penny, B., 2016. *Technologies For Enhancing Computer Security*. 9,454,653.

[20] Moore, G., “Progress in Digital Electronic Circuits,” Intel Corporation, http://www.eng.auburn.edu/~agrawvd/COURSE/E7770_Spr07/READ/Gordon_Moore_1975_Speech.pdf (1975).

[21] Leech, D., Ferris, S., Scott, J., “The Economic Impacts of the Advanced Encryption Standard, 1996-2017,” National Institute of Standards and Technology, https://csrc.nist.gov/publications/detail/white-paper/2018/09/07/economic-impacts-of-the-advanced-encryption-standard-1996-2017/final (2018)

[22] “IEEE 802.3-2018 Standard for Ethernet,” https://standards.ieee.org/standard/802_3-2018.html (2018).