MTR170487
MITRE TECHNICAL REPORT

MITRE

Blockchain and Quantum Computing

Project No.: 25SPI050-12

Princeton, NJ

Author(s): Brandon Rodenburg, PhD
Stephen P. Pappas, PhD

June 2017

This page intentionally left blank.

Abstract

The novel computational data structure known as a blockchain provides an open, public, distributed ledger that has many promising applications. However, any new cryptographic application should take into account anticipated technological development expected to occur within the lifespan of any potentially deployed systems, many of which will be in operation for decades. This document examines vulnerabilities of blockchain technology manifested by the development of quantum computers and gives general recommendations on how to make blockchain more resistant to such technological advances.

This page intentionally left blank.

Executive Summary

Technological advancements promise the development of computers that process information not according to the rules of classical physics and probability, but according to the rules of quantum mechanics. This portends a dramatic increase in computational capacity for specific problems, such as function inversion via Grover's algorithm, and factoring large numbers into prime factors via Shor's algorithm.

The computational data structure known as a blockchain provides an open, public, distributed ledger that has many interesting applications, including digital currencies. The security of this ledger depends on the difficulty of solving certain cryptographic problems which are undermined by the potential of quantum computation. Specifically, hashes as used in signing the blocks of the ledger can be compromised, as can any public/private key system which relies on the so called hidden subgroup problem.

The principal threat is Grover's algorithm, which can dramatically speed up function inversion. This allows the generation of a modified pre-image from a given hash (a hash collision) allowing a signed data block to be modified. This voids guarantees of authenticity of the ledger entries undermining the entire blockchain. The speed-up due to Grover's algorithm is a factor of the square root of the number of possible hashes, meaning that a hash subjected to quantum attack would only be as secure as one with half as many bits subjected to classical attack.

The second threat is Shor's algorithm, which applies to any aspect of blockchain that relies on asymmetric key cryptography. The most commonly referenced problem is that of breaking RSA encryption. RSA relies on the ease of multiplying prime numbers in contrast to the difficulty of factoring large numbers into prime factors. Shor's algorithm speeds-up this process exponentially, effectively breaking RSA encryption. Variants of Shor's algorithm do the same for other asymmetric key cryptosystems.

To counter these threats work has commenced on developing quantum-resistant (a.k.a. post-quantum) cryptographic tools. Currently, the National Institute of Standards and Technology is responsible for navigating this threat landscape. Under the American Innovation and Competitiveness Act of 2017, Congress has tasked NIST with researching and developing cryptographic standards and tools to counter the threat of quantum computation. Post-quantum cryptography is rapidly expanding but has a great deal of uncertainty and no developed standards yet.

Additional research is needed to develop quantum informational versions of systems like blockchain. The most established quantum application is Quantum Key Distribution (a.k.a. Quantum Cryptography), which promises guaranteed secrecy of a given degree for cryptography, despite potential eavesdropping even if the eavesdropper is equipped with a quantum computer. More exotic developments involve using quantum states to represent information, such as quantum currencies, and would require development of easily used quantum state storage.

While quantum information itself is not yet developed to a high technology readiness level, neither are the defenses against the algorithms that quantum computation promises. Both are subjects of active research and may show interesting developments in the next decade, though it is unlikely that we will see significant advances in the very near term.

This page intentionally left blank.

1 Introduction: Classical and Quantum Computation

Computing has revolutionized information processing and management. As far back as Charles Babbage, it was recognized that information could be processed with physical systems, and more recently (e.g. the work of Rolf Landauer) that information must be represented in physical form and is thus subject to physical laws. The physical laws relevant to the information processing system are important to understanding the limitations of computation. Traditional computing devices adhere to the laws of classical mechanics and are thus referred to as "classical computers." Proposed "quantum computers" are governed by the laws of quantum mechanics, leading to a dramatic difference in the computational capacity.

Fundamentally, Quantum Mechanics adds features that are absent in classical mechanics. To begin, physical quantities are "quantized," i.e. cannot be subdivided. For example, light is quantized: the fundamental quantum of light is called the photon and cannot be subdivided into two photons. Quantum mechanics further requires physical states to evolve in such a way that cloning an arbitrary, unknown state into an independent copy is not possible. This is used in quantum cryptography to prevent information copying. Furthermore, quantum mechanics describes systems in terms of superpositions that allow multiple distinguishable inputs to be processed simultaneously, though only one can be observed at the end of processing, and the outcome is generally probabilistic in nature. Finally, quantum mechanics allows for correlations that are not possible to obtain in classical physics. Such correlations include what is called entanglement.

Many useful computational algorithms and data structures have been developed for use on classical computers. Many of these algorithms have parallels on quantum computers but due to the quantum mechanical nature of the information processing could have far greater power. The simplest example of this is called Deutsch's Problem, which demonstrates that quantum computation can be significantly faster than classical computation. We are given a function that is either balanced (equal number of outputs 0 and 1 for full set of inputs) or constant (returns same value regardless of input), and we want to determine which is the case. For the classical case, we need to do two calculations, one for each input value. For the quantum case, we only need to do one calculation and the result is then known to be balanced or constant, though the output values are not identified by the algorithm.

The general promise of quantum computation is that such speedups for specific difficult problems can be obtained more broadly. Clearly this affects many areas of information science and computation, where the functionality of a system is predicated on the difficulty of some calculation. In such cases, a significant speedup can cause a system to break down. This is especially true for cryptographic systems that rely on an asymmetry in computational effort to evaluate a function and to evaluate its inverse. RSA encryption relies on the fact that multiplication of large primes is easy and thus fast, but factoring large composite numbers into two prime factors is very difficult and thus slow. Hash functions have the important property of being easy to calculate but difficult to invert. They provide a quasi-unique fingerprint precisely because it is very difficult to take a given hash value and find a chosen pre-image that yields that hash.

The general threat of quantum computation is that such algorithms become unviable because the premise of asymmetric effort of computation is invalidated. Quantum computing provides potential attacks on many cryptographic systems and algorithms. As of now, no functional quantum computer exists that is sophisticated enough to perform such computations, though there is no doubt as to the efficacy of the algorithms themselves and of their threat to cryptographic systems. Such a quantum computer would need to have at least as many logical, error corrected quantum bits or qbits as the output of the computations, e.g. 256 logical qbits to encode a hashing function with a 256-bit output. Each logical Qbit will likely need to be composed of some as yet unknown large number of physical/noisy qbits, and the current state of the art quantum computer in 2017 only contains on the order of 10 physical qbits. Despite the technical hurdles still to be overcome, the first practical quantum computer will have a tremendous impact on information security and cryptography.

2 Blockchain

In this paper, we are interested in those data structures and algorithms related to blockchain. The blockchain structure was initially developed in the context of the digital currency Bitcoin (Ref. 1) to solve the problem of multiple spending.

The core component implements an open, distributed, cryptographically signed digital ledger that is secure against modification and verifiable by anyone. To prevent bulk rewriting of an entire sequence of blocks from some point in the past as well as attacks to deny service or grow the chain faster than legitimate sources can, a work requirement is added to make rewriting long chains prohibitive. For our purposes here, the relevant structure amounts to the following description:

1) The blockchain consists of a sequence of blocks that are stored on and copied between publicly accessible servers.
2) Each block consists of four fundamental elements:
    a) the hash of the preceding block;
    b) the data content of the block (i.e. the ledger entries);
    c) the nonce that is used to give a particular form to the hash;
    d) the hash of the block.

By including the hash of the preceding block, each successive block strengthens the authenticity claim for the preceding block. Blocks early in the chain cannot be modified without modifying all subsequent blocks or the modification will appear as an inconsistency in the hashes. Similarly, adding the data to the hash makes the data unmodifiable without breaking the consistency of the block sequence. Adding a nonce that is used to impose a signature structure to the hash requires significant work to be performed to generate a new block. This implements the work requirement, thereby preventing the wholesale recreation of a long chain of blocks to supersede the existing chain with modified data.

3 Quantum Computation Algorithms

To understand blockchain in the context of quantum computing and quantum enhanced attacks, we must understand two fundamental algorithms: Grover's Algorithm (Ref. 2) and Shor's Algorithm (Ref. 3). The former is an input search algorithm to find a unique input to a black box function which operates significantly faster than brute force search, thus severely compromising hash functions of insufficient length. The latter provides an exponential speed increase in factoring integers when compared to the general number field sieve (the best-known factoring algorithm) and also can be applied to the hidden subgroup and discrete logarithm problems. These problems are at the heart of breaking many known asymmetric ciphers, and thus are relevant to breaking things like public key cryptography and digital signatures. Taken together, the two quantum algorithms present a significant danger to systems implementing blockchain.

4 Grover's Algorithm

Blockchain relies on the computation of hashes to provide security against modification of the past blocks. The chain is secure against extended revision by both its distributed nature and the computational effort required to re-compute a chain of blocks. Modification of a single block is secured by the difficulty of finding a hash collision with the existing hash, which amounts to the problem of inverting the hash function.

Grover's algorithm is specifically a solution to the problem of finding a pre-image of a value of a function that is difficult to invert. If we are given a signature that is the hash value of some data $s = H(d)$, and the function $H(d)$ can be implemented on a quantum computer, then Grover's algorithm allows us to find d for a given s in time of order $O(\sqrt{n})$ where n is the size of the space of valid hashes. In other words, it allows us to generate hash collisions more efficiently than brute force search, which would be $O(n)$.

For a hash of length k bits this means that we have a significant speedup by a factor of $2^{k/2}$. This can be very large even for small values of k.

5 Shor's Algorithm

Shor's Algorithm provides a dramatic improvement in the efficiency of factoring large numbers. Thus, Shor's algorithm can be used to attack RSA encryption and related problems. The complexity of the general number field sieve (the most efficient known algorithm to factor numbers) is super-polynomial (run time longer than any polynomial in the input length) but sub-exponential (shorter than exponential in the input length), while Shor's algorithm is polynomial in the input length, making the gain in speed roughly exponential. In practical terms, this makes RSA keys of 4096 bits in practice unbreakable with classical computation, but breakable with quantum computation. The consequence is that any aspect of a blockchain implementation that relies on RSA or similar algorithms would be vulnerable to quantum computational attack.

The first target of Shor's algorithm was the factoring of large composite integers consisting of a product of two large primes. However, factoring is a specific case of the more general hidden

subgroup problem, and modifications of Shor's algorithm can solve all such problems. This allows solution of problems such as the discrete logarithm problem, which in turn makes such cryptographic algorithms as ElGamal encryption, Diffie-Helman key exchange, the Digital Signature Algorithm, and elliptic curve cryptography insecure. The existence of Shor's algorithm demonstrates that a quantum computer opens vulnerabilities beyond that of just hash collision generation or function inversion by Grover's algorithm.

6 Threat to Blockchain

In the context of quantum computing, we are confronted with two aspects of invalidating the promises of blockchain. First, the inversion of hashes is assumed to be computationally difficult. If this can be dramatically simplified by a quantum computer, the authenticity of the upstream blockchain can no longer be guaranteed and the authenticity of entries in the blockchain is compromised. As stated above, Grover's algorithm seeks the pre-image to a function value, and can do so significantly faster than the classical brute force search of generating each output and comparing it to isolate the generating input.

Grover's algorithm can be used in two ways to attack the blockchain. The first, and most obvious, is that it can be used to search for hash collisions which can be used to replace blocks in situ without disturbing the integrity of the blockchain. The second is that it can speed up the generation of nonces, potentially to the point that entire chains of records can be recreated with consistent modified hashes sufficiently quickly to undermine the integrity of the chain. In both cases the algorithm is used to find the pre-image of a given value under a difficult to invert function.

As a secondary threat, in any aspect of a blockchain implementation that uses public/private key cryptography, whether it be in information exchange between parties or in digital signatures, a quantum computer may be able to break the security of the encryption.

7 Grover's Algorithm Attack: Full Collision

If full collisions of hash values can be generated, it is possible to take a modified block content and a given hash, and add trivial data to the content to make the given hash consistent with the block content. In general, this problem is computationally difficult. The general case assumes that it requires a brute force search through the possible source data with sufficient additional bits to exhaust the hash space until a case is found that matches the known hash value. For an ideal hash, this requires linear time in the size of the hash space. Known weaknesses in the hash function can reduce this time, but generally the reduction is not large. We expect a run time of order $O(n)$ for this classical attack.

Grover's algorithm runs in time $O(\sqrt{n})$, and so would give a speedup of $O(\sqrt{n})$ compared to classical collision search algorithm. This potentially makes it viable to insert a modified block into the chain without compromising the sequential consistency of the blocks. This speed increase is equivalent to finding a hash collision by brute force with half as many bits in the hash. Since this attack is only moderately fast, one could consider increasing (doubling) the hash length, but the

computational effort to calculate the nonce with longer hashes would tend to limit the ability to generate the chain, and would possibly make the blockchain not viable.

We consider this asymmetric case, attacker with a quantum computing and defender with only classical computing, as a worst scenario. A slightly better scenario is when both parties have the same computational capability because then there is hope that the balance of time to generate hashes and to invert hashes remains similar to the classical case. If this is true, the operational consequence is that whoever gets quantum computational capacity first has an advantage, but only until the defending parties develop the capacity themselves. At that point, we expect that either the system is again viable, or the system is broken beyond repair and must be discarded.

8 Grover's Algorithm Attack: Mining Time

When we consider the mining step of the blockchain growth, we run into another problem: the calculation of the nonce. This calculation adds computational cost to re-writing the chain, and amounts to finding a pre-image to a partially defined hash. Grover's algorithm could speed up the generation of nonces, making the reconstruction of the chain from a modified block forward much faster, thereby opening the attack of regenerating the chain by undermining the computational effort of extension.

It becomes feasible for a party with a quantum computer to rapidly outstrip competitors, who have only classical computing capacity, in generating additional blocks on the chain. In crypto-currency applications this means that the mining step becomes much shorter and thus allows individuals to obtain more currency than others by mining faster. In the case of a consensus blockchain for other ledger applications, the fastest miners will dominate the generation of new blocks and thus can take control of the content of the blockchain.

Of course, if the generation of nonces is even faster, there is nothing to prevent a wholesale re-creation of an entire blockchain in negligible time, and then substituting that history by growing faster than others can grow the true chain. Since the longest chain is conventionally chosen as the accepted truth, the faster growing chain will come to dominate the blockchain, effectively re-writing history.

9 Threats Beyond Hashes

Hashes are for the most part only known to be susceptible to Grover's algorithm for finding function pre-images. Shor's algorithm, on the other hand, is highly effective at factoring integers, or more completely, solving the hidden subgroup problem. Any aspect of blockchain that implements commonly used public/private key algorithms is susceptible to attack with Shor's algorithm. From the simplest point of view the algorithm serves to find the two prime factors of a composite integer used as a public key in an algorithm like RSA. Being able to factor the integer, which is computationally challenging on classical computers, yields to the attacker the private key of the public/private pair. That makes it possible for the attacker to forge messages, signatures, etc.

While this is not a threat to the blockchain structure of linked hashes, nor to the generation or re-generation of nonces, it means that, for example, any content that is signed may be forged by a

suitably equipped intermediary in the process, passing the forged content on to the blockchain where it gets incorporated and thereby gains the legitimacy of being part of the publicly readable and verifiable record.

Furthermore, any encrypted communications used in the infrastructure upon which a blockchain is constructed are vulnerable to an attacker who can break the cryptographic security of the communications. While this is slightly removed from the core features of a blockchain implementation, it is nonetheless of importance in considering implementations that might be of critical importance.

10 Quantum-Resistant Cryptography

As described in the previous sections, the advent of quantum computers will have a major impact on how we think about algorithms used for cryptographic applications. According to the Information Assurance Directorate (IAD) of the NSA (Ref. 5), algorithms used in national security systems require twenty years for full deployment and should be designed to protect information for at least thirty years. We cannot predict if or when a large-scale quantum computer will ever be manufactured, however many experts anticipate such a device within these timescales. Therefore, the development of cryptographic algorithms which are "quantum resistant" has been determined to be a national priority.

Quantum resistant cryptography, also known as post-quantum cryptography, is a field that includes potential attacks using a quantum computer as part of the analysis of (classical) cryptographic algorithms (Ref. 4 and Ref. 6). Although there are some insights in this area, as mentioned above, it is still a very new area with a good deal of uncertainty and no accepted standards. To remedy this problem Congress enacted a law in 2017 known as "The American Innovation and Competitiveness Act," which has tasked the National Institute of Standards and Technology (NIST) to "research and identify, or if necessary, develop cryptography standards and guidelines for future cybersecurity needs, including quantum-resistant cryptography standards." NIST has already initiated this process, and public updates to this process are posted to <http://www.nist.gov/pqcrypto>. The IAD has stated in relation to this process that the "NSA believes that the external cryptographic community can develop quantum resistant algorithms and reach broad agreement for standardization within a few years."

11 Post-quantum Cryptography for Blockchain Protocols Involving Hash Functions

Despite the fact that standards are still being developed for quantum resistant cryptography, we can make some general statements of what aspects will be important for designing systems that involve blockchain based technologies. The first aspect is related to the hashing function itself. As described in the previous sections, Grover's algorithm provides a quadradic speedup over classical algorithms for evaluating hash functions. Since this speedup is not an exponential speedup like Shor's algorithm, this means that the desired computational complexity of a function that is desired for secure applications can be restored simply by increasing the number of bits used in the

calculation. At most one needs only twice the number of bits due to the quadradic speedup of the algorithm.

As previously described, there are two aspects in which hash functions are used to protect a blockchain. The primary method relies on the fact that inverting a hash or finding a collision is computationally difficult. The difficulty in finding a different data block with the same hash grows with the length of the hash. The complexity is $O(n)$ classically, but $O(\sqrt{n})$ *via* Grover's algorithm for a hash space of size n. So, if a certain level of difficulty is required for security, a quantum-resistant standard will require twice the hash length of a similar requirement that considers only classical algorithms.

The second way in which blockchain may utilize hash functions for security is by signing a block. This is done for instance by finding a nonce such that the first m bits of the block's hash are zero. This is equivalent to computing a partial collision of the hash function, and is computationally difficult. This difficulty is precisely the 'proof of work' that a signature is designed to require. Just as the hash length k can be increased in order to maintain a desired level of protection against a quantum attack, so also the desired length m required for signing the block can be increased to ensure a minimal 'proof of work'. However, this comes at the expense of making the required work computationally twice as hard per additional bit, or equivalently take twice as long, for the classical devices that are used to sign a data block. Therefore, there will be an inherent trade-off between the system requirements necessary for implementing any blockchain protocol that uses hash based block signatures and protecting against a spoofing attack from a quantum device.

12 Post-quantum Cryptography for Protocols Beyond the Blockchain Hash Function

We have implied above that in addition to hash functions in blockchain based technology, there are likely to be serious concerns about quantum threats to other aspects beyond just the blockchain itself. If the blockchain ledger needs to be distributed, then encryption schemes will be required. There will also likely need to be various protocols in place defining what entities are, for instance, allowed to expand the blockchain, in which case identity verifications or digital signatures might be utilized. In many of these cases, current standard cryptographic algorithms are generally insufficient to protect against the threat of quantum computing.

As stated earlier, a key issue with many current cryptographic algorithms is that security relies on the difficulty of a mathematical problem. The asymmetric key encryption scheme of RSA relies on the difficulty of prime factorization of large numbers; the Digital Signature Algorithm (DSA), the standard for digital signatures, is based on the problem of computing discrete logarithms; and the Elliptic Curve Digital Signature Algorithm (ECDSA) is a promising variant of DSA and example of elliptic curve cryptography (ECC). All three types of problems of factorization, discrete logarithms, and ECC can easily be solved by Shor's algorithm on a sufficiently powerful quantum computer (Ref. 6).

Although NIST has yet to define quantum-resistant cryptographic standards, there are a number of promising classes of cryptographic systems that are believed to be robust to attacks from either classical or quantum devices. As listed in reference 6, some of the most promising areas include:

- Hash-based cryptography. The classic example is Merkle's hash-tree public-key signature system (1979), building upon a one-message-signature idea of Lamport and Diffie.
- Code-based cryptography. The classic example is McEliece's hidden Goppa-code public-key encryption system (1978).
- Lattice-based cryptography. The example that has perhaps attracted the most interest, not the first example historically, is the Hoffstein–Pipher–Silverman "NTRU" public-key-encryption system (1998).
- Multivariate-quadratic-equations cryptography. One of many interesting examples is Patarin's Hidden Field Equations (minus variant) public-key-signature system (1996), generalizing a proposal by Matsumoto and Imai.

13 Quantum Based Cryptography

An additional strategy for future crypto-systems involves exploiting quantum features in new technology. This field of quantum cryptography is distinct from post-quantum cryptography which relies purely on classical methods and present-day technologies to protect against potential future quantum attacks. Instead, quantum cryptography is itself part of quantum information science and looks for how quantum effects can create fundamentally new ways of doing cryptography.

The primary and most mature technology that has come out of quantum cryptography is quantum key distribution (QKD). QKD is a protocol by which a random bitstream can be generated between parties. Once established, this random message is used as a one-time pad (OTP) or Vernam-cipher to encrypt a secret message. This method of distributing a secret shared key is not secured by mathematical complexity like normal methods of distributing cryptographic keys (*e.g.* Diffie–Hellman), but instead is based on the laws of quantum physics itself. This security specifically comes from the Quantum No-cloning theorem, a consequence of the Heisenberg uncertainty principle which states a signal made of individual quantum particles cannot be copied without introducing observable errors, preventing any eavesdropper from avoiding detection. Once a random key has been established between two parties *via* a QKD protocol, the encrypted message is considered cryptographically or unconditionally secure.

QKD is the most mature technology within the field of quantum information science. Commercial companies exist that will sell transmitters and receivers, and such systems have been used in both the private and public sectors. The technology currently requires private networks (*e.g.* dark fibers), cannot be repeated or routed and is currently limited to city scale networks. Although these current limitations place severe limits on QKD's practicality for most applications, the technology is still developing at a rapid pace and so will likely become more widespread in the near-future.

In addition to QKD, there is a wide variety of ideas that are currently being researched that could make a significant impact to blockchain based systems. For instance, information can be encoded and transmitted directly into a quantum stream (rather than just using a quantum channel to distribute a key). There has also been a proposal for a "Quantum Bitcoin," which uses a classical blockchain ledger but uses quantum methods to mine and verify a block. There are also protocols to encode and store information such as a ledger in a quantum system making the information tamper-proof. There are also quantum bit commitment protocols which may be seen as a type of

alternative to digital signature schemes. Many of these ideas show promise, however all of these technologies are currently at very low technology readiness levels with many of the technologies being at least as difficult to implement as quantum computing itself.

References

1) S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” bitcoin.org/bitcoin.pdf
2) Lov K. Grover, “A fast quantum mechanical algorithm for database search,” Proceedings of the twenty-eighth annual ACM symposium on Theory of computing 212–219. ACM. doi:10.1145/237814.237866 arXiv:quant-ph/9605043
3) Peter W. Shor, “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer,” SIAM J. Comput., 26 (5): 1484–1509. doi:10.1137/S0036144598347011
4) “Report on Post-Quantum Cryptography,” National Institute of Standards and Technology, L. Chen et al., NISTIR 8105. doi:10.6028/NIST.IR.8105
5) “CNSA Suite and Quantum Computing FAQ,” Information Assurance Directorate of the NSA, Document MFQ-U-OO-815099-15 (2016).
6) Post-Quantum Cryptography, DJ Bernstein, J Buchmann, and E Dahmen, eds. (Springer Berlin Heidelberg, 2009) doi:10.1007/978-3-540-88702-7