SECTIGO

The Search for Quantum Resistant Cryptography

A WHITEPAPER FROM SECTIGO

SEPTEMBER 2019

Introduction: The Coming Quantum Apocalypse

Cryptography depends on the ability to create strings of bits that are extremely difficult to guess using brute force methods. Before it can be used, any encrypted data must be decrypted using a digital key, and for our cryptographic systems to work these asymmetric keys must be impractical to discover just by running through combinations until we get one right. We accomplish this goal by using a suitably large "key space" and ensuring that our encryption algorithms are truly unpredictable, rather than pseudo random numbers where a clever decryption program can use predictable patterns of number generation to find results more quickly.

Decryption depends, of course, on how long it takes for a computer program to move through and guess the potential combinations, and cryptographic algorithms use techniques that make it hard for computers to do so. The two common cryptographic foundations for our standardized, ubiquitous PKI systems are RSA encryption and Elliptical Curve Cryptography (ECC).

- RSA depends on prime numbers. Large prime numbers are very difficult for traditional computing platforms to factor, so they are tremendously time-consuming as the computer has no option but to go through all combinations.
- Elliptic Curve Cryptography works by finding two points on an elliptic curve that intersect perfectly to "unlock" an encrypted asset. Solving for two points in this curve is likewise difficult for traditional computers.

Enter quantum computers. Quantum computers take advantage of the very nature of quantum physics to create an entirely new computing paradigm, unlike the traditional 0/1 gated computers we have been using since the 1960s. Instead, they run on quantum bits (known as qubits) which can superpose and entangle themselves in order to perform multiple processes simultaneously.

Quantum computing as we know it was greatly aided and inspired by a piece of mathematics named Shor's Algorithm. Revealed by Peter Shor in 1994, Shor's Algorithm provides a roadmap for using a quantum computer to aid immensely the factorization of numbers. This fact matters because both RSA and ECC depend on the difficultly of factorization to remain secure from brute force attacks. Shor's Algorithm reduces the amount of processing required to crack

either of these cryptographic schemes by a great many orders of magnitude, making it not only practical but inevitable that quantum computers will render both RSA and ECC unusable due to security concerns.

Not all aspects of our standardized cryptographic systems are subject to Shor's Algorithm, including popular hashing algorithms such as AES and SHA-256. The approach to defeating these schemes using quantum computers was described by Lov Grover in 1996 using what is known as Grover's Algorithm. Though Grover's Algorithm does significantly reduce the time required for a quantum computer to defeat these schemes using a brute force attack, the required time to break remains sufficiently immense that these algorithms are not consider realistic attack vectors using foreseen quantum computing architecture.

To put the difference between the two in perspective, the National Academies for Sciences, Engineering, and Medicine provides these estimates of the time required to break the cryptographic primitives that are subject to either Shor's or Grover's Algorithm. As you can see, Shor's algorithms reduces the time to break popular algorithms to hours or days, while Grover's Algorithms reduces them at worst to thousands of years.

Cryptosystem	Key Size	Security Parameter	Quantum Algorithm Expected to Defeat Cryptosystem	Time Required to Break System
AES-GCM	**128 192 256**	**128 192 256**	**Grover's algorithm**	**2.61 × 1012 years 1.97 × 1022 years 2.29 × 1032 years**
RSA	**1024 2048 4096**	**80 112 128**	**Shor's algorithm**	**3.58 hours 28.63 hours 229 hours**
ECC Discrete-log problem	**256 384 521**	**128 192 256**	**Shor's algorithm**	**10.5 hours 37.67 hours 55 hours**
SHA256	**N/A**	**72**	**Grover's algorithm**	**1.8 × 104 years**
PBKDF2 with 10,000 iterations	**N/A**	**66**	**Grover's algorithm**	**2.3 × 107 years**

Source: *Quantum Computing: Progress and Prospects,* edited by Emily Grumbling and Mark Horowitz. The National Academies Press. 2019.

Once quantum computers reach the point where RSA 2048 and ECC 256 are compromised, the foundational security of all our present day digital systems will be invalid. Our modern systems of finance, commerce, communication, transportation, manufacturing, energy, government, and healthcare will for all intents and purposes cease to function. We cannot circumvent the problem simply by increasing key lengths, as the necessary increases would render performance unworkably slow. This eventual outcome is so severe that it is sometimes referred to as the "Quantum Apocalypse." Avoiding the Quantum Apocalypse is of paramount importance.

To have this effect, quantum computing need not be as advanced as you may think. Real-time decryption and encryption are not required for a compromise to be damaging. As quantum computing gets to the point where it can find a private RSA or ECC key in some reasonable period of time, the potential for data breach will be vast. Imagine a scenario where a quantum computer can break a key based on a full day of processing, or a full month for that matter. In this environment, a bad actor could store the encrypted file of a high-value data target and set a quantum computer on the task of breaking its private key. For many types of sensitive information, whether the decrypted form of this information is available today or in a week is of little consequence.

There also exists the possibility that a well-resourced bad actor might simply store high-value encrypted data files for future decryption once the technology has caught up. While certain targets such as currently active credit card numbers are likely to have little value by the time a quantum computer can crack them, many confidential items such as industrial or state secrets or PII/PHI might still be damaging if revealed a decade from today.

Truly Random Numbers Are Harder Than They Appear

Since cryptography depends on unpredictability, encryption and hashing algorithms always must start from some kind of unpredictable "seed" number to generate their output. While it may seem easy enough to generate unpredictable "random" numbers for our own use in day-to-day life, for industrial-strength computing applications, truly random numbers are a nontrivial task.

The first challenge is with scale. While flipping a coin may work for deciding who gets to ride "shotgun" in the front seat, key-generating and hashing activities need to keep pace with the high volume, low latency needs of our contemporary computing environments. Digital transformation projects are bringing all aspects

Truly Random Numbers Are Harder Than They Appear (continued...)

of business and government into the computing world, while new-generation architectures such as containerization and Internet of Things (IoT) expand our certificate needs exponentially. That means our random number generators must produce huge numbers of unpredictable values very rapidly.

Additionally, it's important to obtain seed values that are truly random and not merely pseudo random. Pseudo randomness is apparent randomness to the unaided human brain that is not actually random. It is quite possible for an algorithm or method to put out a sequence of values such that a simple observer would not be able to predict future values with any accuracy. However, there might still be patterns or biases in these numbers that statistical analysis and big data techniques could uncover.

After all, even if a number is not completely determined, it still might not be random. In the event that we are able to rule certain values right out, then the available *number space* decreases, and with it the total set of combinations that must be tried. This total number of combinations is referred to as the amount of *entropy* in a string of values. Note that the total entropy and the total number *size* need not be the same. If some values of particular digits in the string or some combinations of values can be categorically ruled out, then the amount of entropy goes down—even though the number's size does not—as these values or combinations need not be guessed in the effort to find a single correct string. Even if we cannot completely eliminate certain values or combinations, bias in the likelihood of some numbers over others can help guide the combinatorial process to the most likely outcomes first, reducing the average time required for a brute force attack.

Organizations go to great lengths to create random numbers in the volume they require. In fact, some institutions offer *randomness beacons*, providing random numbers for those who need them. Some of the more colorful randomness beacons include these:

- **Cloudflare offers LavaRand, which creates random numbers from the digital feed of a camera pointed at a wall full of lava lamps.**
- **The University Chile's Seismic Girl generates its random numbers from the combination of seismic measurements in Chile, a stream from a local radio station, a selection of Twitter posts, data from the Ethereum block chain, and a Random Number Generator (RNG) card.**
- ***École polytechnique féderale de Lausanne* brings us URand, which collects input behavior such as mouse movements and key strokes from a large number of users to create a continuous random number stream.**

Mosca's Inequality and the "Z Date"

As we seek to understand in more detail quantum computing's threat to our ubiquitous cryptographic systems, one important consideration is exactly how long it will take for quantum computers to reach the danger point. While nobody knows for sure, some thought has gone into estimating how much time we might have.

Analysis of this question was pioneered by the University of Waterloo's Michele Mosca. Mosca postulated the formula that has come to be known as Mosca's Inequality. **This idea can be expressed in a simple equation:**

$$X + Y \stackrel{?}{=} Z$$

X: The amount of time our encrypted data must remain secure before its breach is of no significant value

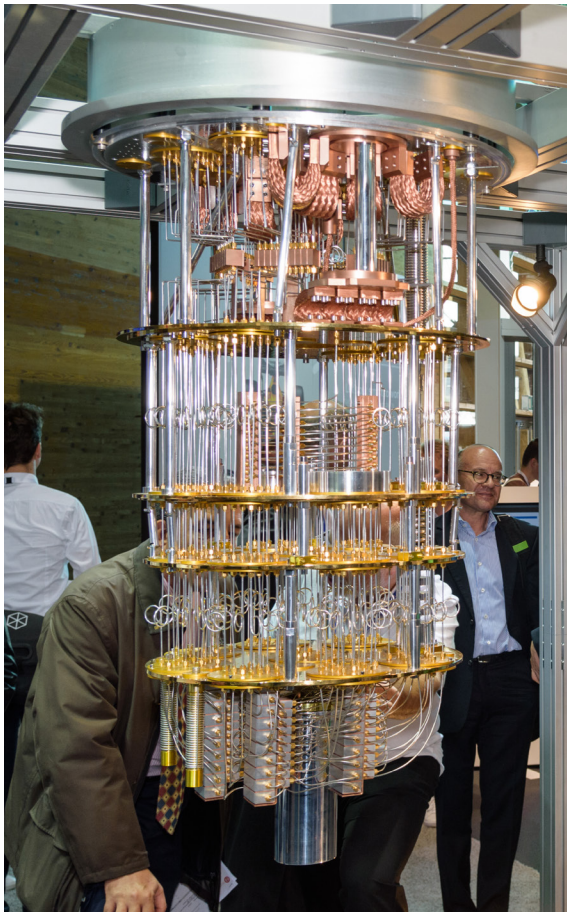
Y: The time we need to find one or more new algorithms to replace RSA and ECC and then to develop, test, deploy, and adopt systems globally that will use the new algorithm(s)

Z: The time until quantum computers will be powerful enough to break RSA and ECC

If X+Y is less than Z, then we are able to find and deploy new cryptography in time to obviate the quantum computing threat. If X+Y is greater than or equal to Z, then the Quantum Apocalypse occurs.

While this framework is widely accepted and has withstood the test of scrutiny for some years, it does present a problem, which is how to precisely peg down the values of X, Y, and Z. In particular, there is a lot of focus on what we have come to call "the Z date." The Z date is the last date on which data encrypted with RSA and ECC at commonly used key lengths remain secure from attack by quantum computers.

To calculate the Z date, we must predict the speed at which quantum computers will advance. Though nobody knows for sure, we do have estimates. Mosca himself estimated in 2015 that there was a 15% chance of breaking RSA-2048 by 2026 and a 50% chance by 2031. In 2019 the National Academies of Sciences, Engineering, and Medicine published a book called *Quantum Computing: Progress and Prospects,* edited by Emily Grumbling and Mark Horowitz.



This consensus study states,

> Given the current state of quantum computing and recent rates of progress, it is highly unexpected that a quantum computer that can compromise RSA 2048 or comparable discrete logarithm-based public key cryptosystems will be built within the next decade.

However, the report goes on to say,

> Even if a quantum computer that can decrypt current cryptographic ciphers is more than a decade off, the hazard of such a machine is high enough—and the time frame for transitioning to a new security protocol is sufficiently long and uncertain—that prioritization of the development, standardization, and deployment of post-quantum cryptography is critical for minimizing the chance of a potential security and privacy disaster.

If we view the Z date to be sometime between ten and fifteen years away, how does that compare to the expected values for X and Y?

The value for X is a matter of debate and to some degree varies based on the type of information we're discussing. While a data type like credit card numbers may have a very short shelf life, it's easy to think of information that owners will want to keep secret for decades to come. One alternative that Mosca offers for consideration is seven years, as that is a very common business record retention requirement.

Y is determined by adding the time it will take to research and settle on new cryptographic algorithms and the time it will take to deploy these algorithms systematically across the ecosystem. In August 2019, the National Institute for Standards and Technology (NIST) suggested that the academic community needs two to three years to vet candidate algorithms to replace RSA and ECC. Once this step is complete, we can begin looking at deployment options. As with the X value, to some degree this idea is an unfair overgeneralization. In reality,

some systems will be upgraded very quickly while others will languish. As a general principle, mission-critical systems and highly sensitive data will probably undergo upgrades more quickly, but there will be plenty of exceptions to this rule.

We have seen precedents that required a great deal of time for phase-overs of this sort. Similar efforts with SHA-1 and MD5 took about ten years for widespread transition to replacement algorithms, and ECC took a similar amount of time from its early days until it became practical for most commercial uses.

These numbers illustrate the potential trouble we may face. If we take an X value of seven years and calculate Y as three years of research followed by ten years of development and deployment, that is a total of twenty years until RSA and ECC are effectively phased out. However, we just saw estimates putting Z ten to fifteen years away. That leaves a potential exposure period, according to these estimates, of five to ten years. That's a lot of exposure.

Furthermore, most projections of the progress in quantum computing assume that research and manufacturing methods remain similar to what they are today. They do not account for the possibility of breakthrough "Eureka" moments in either the production of quantum computers or the methods of using them to attack these cyphers. Should such developments occur, they will shorten the timetable for Z.

Additionally, very heavily resourcing for research could cut by a great deal the time required to achieve a quantum computer capable of defeating current, commonly used, RSA and ECC keys. Oxford University's Simon Benjamin estimates that a state-sponsored, Manhattan-project-style research initiative could cut that timeline to as little as six years. In this scenario, presumably the party in possession of the crypto-breaking machine would keep silent about it, making it unlikely that outside observers would even know that the Z date had arrived.

These examples illustrate the high degree of uncertainty in putting values on Mosca's equation. They also illustrate that we must not in any way be complacent about finding and migrating to quantum-resistant cryptographic standards as soon as we are able.

Does Quantum Annealing Pull Forward the Z Date?

Quantum annealing is a special case of quantum computing for which the engineering challenges are lessened. Rather than attempting to maintain qubits in a static state, as a full-blown quantum computer does, quantum annealers use the natural quantum progression to perform calculations. This fact radically reduces the engineering challenges required to create working computers, and therefore we can expect quantum annealers to achieve stability sooner.

Though quantum annealers are not suited for all computing tasks, the presence of a startup commercial industry in this space indicates that researchers are finding practical applications for these computers. Recent work suggests that factoring prime numbers is one of the tasks for which quantum annealing is suited, which is important because prime number factorization is the foundation of RSA encryption.

Though the world has yet to see a practical demonstration of quantum annealing as the key to unlocking RSA, there exists a real chance that computers of this stripe may bring forward the Z date specifically as it applies to RSA encryption.

NIST Leads the Search for the New Cryptography

To help the industry arrive at one or more legitimate alternatives to RSA and ECC, the NIST has created its Post-Quantum Cryptography project. The purpose of this project is to motivate and coordinate the efforts of academics and industry to identify and vet potential next-generation cryptographic schemes with an eye toward arriving at one or more algorithms that are reliably demonstrated to be safe from defeat by advances in quantum computing.

To be clear, quantum computers will not replace binary computers. Rather, both architectures will go on to live side by side, with traditional binary computing serving most tasks while quantum computers designated for the specific use cases where they offer improved performance. Therefore, the encryption paradigms of the future need to withstand attack not only by quantum computers but by traditional computers as well. For the expected future the platforms on which this encryption operates will mostly be of traditional binary architecture.

Such is the environment in which our new quantum-resistant cryptography will need to succeed. For an algorithm to prove suitable in the post-quantum world, it will have to meet a few criteria:

- Fast to encrypt using traditional computers
- Fast to decrypt (with the private key) using traditional computers
- Impractically slow to decrypt (without the private key) using either traditional or quantum computers
- Able to generate encrypted data of a size that is reasonable for storage and transmission across networks and the internet
- Compatible with the vast range of software, hardware, and services we depend on today
- Well understood and vetted against potential attacks

These last two points are often overlooked in discussion but are extremely important. RSA- or ECC-based cryptography is present in virtually every digital environment on this earth. All commercially available hardware, software, and services use and depend on Public Key Infrastructure (PKI) structures that take the presence of these two algorithms as a given. Likewise, private development efforts from enterprises, governments, schools, and even at-home hobbyists assume that RSA and ECC are available.

It would be impossible to catalog the full set of applications, use cases, SaaS offerings, and devices that would completely fail in the absence of traditional PKI. To the greatest degree possible, any new algorithm must be able to plug and play in our existing environments and applications and just plain work.

Similarly, we cannot overstate the need for our new cryptography to be proven against cryptographic attacks. The RSA and ECC schemes have the advantage that they have protected the world's most valuable information targets for decades without anyone discovering a way to fundamentally undermine them using available computing architectures. In fact, ECC did not gain widespread acceptance until the discovery of a proof of Fermat's last theorem in 1995, as that was required for the community to become convinced that no crafty attack against the algorithm was waiting to be discovered.

Now industry, academics, and government will all be hurrying up to choose and deploy one or more new algorithms to replace these proven, battle-hardened schemes. If it's possible for a math genius at a chalk board to discover a way to fundamentally undermine the security of a chosen algorithm, the effects will be devastating. To prevent that outcome, all candidates must be thoroughly investigated.

NIST has divided its Post-Quantum Cryptography project into phases, or "Rounds:"

Round 1 focused on information gathering. Sort of the equivalent of a community brainstorm, Round 1 collected as many potential cryptographic candidates as the community could muster. Round 1 received eighty-two submissions, which in December 2017 NIST culled down to sixty-nine that it deemed "complete and proper."

NIST divided these candidates into four groups based on their fundamental approach: Lattice-based, code-based, multivariate, symmetric-based, and "other." NIST suggested the merger of similar candidate algorithms where it appeared that teams could profitably work together to integrate the best parts of their approaches, dropping the number of candidates dropped to twenty six. These were announced in January 2019 at the beginning of Round 2.

Round 2 concentrated on testing the Round 1 candidates for cryptographic viability. In particular, candidates had to prove at least as hard to break for both binary and quantum computers as AES128, AES192, and SHA256 are. This phase concluded in August 2019.

NIST has announced the results of Round 2, which is that more than twenty candidates have met the requirements laid out for them. NIST describes these candidates as being "quite good, and quite diverse" and states that there is no obvious best choice.

Based on this status, NIST has suggested that more research and dialog are needed, including an examination of the relative importance of success criteria. The institute predicts that this examination will require two to three years and targets 2022 for publication of a new standard.

NIST's is not the only effort in this regard. The appropriately skilled individuals at universities and think tanks continue to put cycles against these questions, and IT industry providers like Sectigo are placing strong emphasis on the tools and services necessary for rapid adoption of new algorithms when they are ready.

Conclusion

Quantum computers are definitely coming, and they will render the digital world's cryptographic underpinnings insecure. It is not a matter of *if* but only *when*. Though the exact timing of the Quantum Apocalypse is not—and cannot be—known, by any reasonable estimate we need to be focused on finding and deploying new cryptographic alternatives as soon as possible.

Leaders in research, government, and industry are all putting their weight behind the effort to investigate, select, and roll out new cryptographic standards in time to prevent damage to the world's digital economy. It won't be a quick or easy process, but it is a vital one.