Barriers to Quantum-Resistant Digital Infrastructure: A Sociotechnical Analysis

A Research Paper submitted to the Department of Engineering and Society

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Garrett Delaney

Spring 2025

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Advisor

Sean Murray, Department of Engineering and Society

The difficulty lies not so much in developing new ideas as in escaping from old ones.

— John Maynard Keynes, *The General Theory of Employment, Interest and Money,* 1936

I. Introduction

Amidst the recent cultural fixation with artificial intelligence (AI), another emerging technology has flown under the radar: quantum computing. While quantum computing is seldom discussed among the general public compared to AI, this nascent technology has the potential to be just as disruptive to society. Quantum computing increases computational power by leaps and bounds compared to conventional computers. In late 2024, Google's Quantum AI team announced a triumph from their new Willow chip. For perspective, Harmut Neven the head of Google Quantum AI boasted: "Willow performed a standard benchmark computation in under five minutes that would take one of today's fastest supercomputers 10 septillion years—a number that vastly exceeds the age of the Universe" (Valerio, 2024). Quantum computing is projected to revolutionize computational power, but it especially presents an unprecedented threat to global cybersecurity.

Modern encryption methods, including RSA and Elliptic Curve Cryptography (ECC), rely on the difficulty of mathematical problems such as integer factorization and discrete logarithms, problems that classical computers struggle to solve in a timely manner. However, quantum computers can solve these problems exponentially faster, rendering today's encryption methods obsolete (Bhatia & Ramkumar, 2020). In response, researchers have developed post-quantum cryptography (PQC), a suite of cryptographic techniques designed to resist quantum attacks. The National Institute of Standards and Technology (NIST) recently standardized three PQC algorithms after nearly a decade of research (Boutin, 2024). While the technical feasibility

of PQC has been established, widespread implementation remains a challenge. The sociotechnical hurdles associated with deploying a new quantum-resistant cryptographic scheme throughout digital infrastructure are analyzed in this paper.

The transition to quantum-resistant cryptography is not simply a technical upgrade, it is a massive sociotechnical shift that faces significant barriers. Store Now, Decrypt Later (SNDL) attacks highlight the urgency of this transition; adversaries are currently collecting encrypted data with the expectation that they will be able to decrypt it once a cryptanalytically relevant quantum computer (CRQC) becomes available (Joseph et al., 2022). Despite the clear risks, organizations across industries have been slow to adopt PQC. While technical challenges such as increased cryptographic key sizes and performance overhead contribute to this hesitation (Rawal & Curry, 2024), social and governmental factors such as standardization delays, compliance concerns, and a shortage of skilled workers also play a major role (Hekkala et al., 2023; Joseph et al., 2022).

In this paper, I argue that the primary barriers to implementing quantum-resistant digital infrastructure are not merely technical but are deeply rooted in social and governmental factors. The transition to PQC is hindered by three major challenges which I have categorized as: (1) technical constraints, including increased computational requirements and performance trade-offs, (2) social and workforce limitations, particularly the lack of skilled cryptographers and secure implementation practices, and (3) institutional inertia, as regulatory uncertainty and compliance concerns slow adoption. Using the framework of technological momentum, I analyze these barriers as both social and technical phenomena, demonstrating that PQC adoption is not a simple as inventing a new algorithm. Understanding these challenges is crucial for policymakers, cybersecurity professionals, and organizations preparing for the post-quantum era.

II. Problem Definition: The Urgency of Quantum-Resistant Digital Infrastructure

Encryption is the mathematical process of scrambling data to appear random and ensure it cannot be intercepted and read by unauthorized parties (Cloudflare, Inc., n.d.-a), according to a Cloudflare, a prominent cybersecurity and networking company which powers 19.4% of all websites on the internet (W3Techs, n.d.). The original content of the message is known as the plaintext, and the encrypted and unreadable form is known as ciphertext (Cloudflare, Inc., n.d.-a). Shown in Figure 1 is a basic example of encrypting the plaintext element “Hello” into an encrypted ciphertext “SNifgNi+uK0=”.

Figure 1

Encryption Example

Note. Encryption involves transforming a plaintext message into a scrambled ciphertext. From (Cloudflare, Inc., n.d.-a).

An encryption algorithm is used to transform the data from its plaintext form to its ciphertext form as well as a decryption algorithm to go the other way. In addition to the plaintext, a cryptographic key is supplied to the encryption algorithm. If the same key is used to encrypt and decrypt, this is known as symmetric encryption, if different keys are used, it is known as asymmetric encryption (Aydeger et al., 2024). For symmetric encryption, the sender and the recipient of the message must agree on the cryptographic key (Cloudflare, Inc., n.d.-a). Referencing Figure 1, the intended recipient uses the cryptographic key to decrypt the cyphertext message “SNifgNi+uK0=” back to its plaintext form “Hello”. This is a trivial example but in the

real world this can be used to protect sensitive data such as passwords, financial transactions, medical records, national security documents, etc.

Consider if a third-party eavesdropper snoops the ciphertext, much like how an Internet service provider (ISP) can see all the traffic of its customers. A well-designed and secure encryption algorithm makes it extremely difficult for the eavesdropper to decrypt the message without the cryptographic key. Consider one of the simplest encryption algorithms: the Caesar cipher, which involves shifting each letter in the plaintext by a certain number as shown and explained in Figure 2. It is named after Julius Caesar who encrypted military messages with it and his adversaries were unable to decrypt it (Bloomfield, n.d.-a).

Figure 2

Caesar cipher with a cryptographic key of 4

```
Plaintext:   A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Cipher Text: W X Y Z A B C D E F G H I J K L M N O P Q R S T U V
```

Note. With a cryptographic key of 4, to encrypt go 4 letters to the left, to decrypt go 4 letters to the right. For example, the plaintext “DOG” becomes the ciphertext “ZKC”. (Created by author).

The Caesar cipher is insecure because it is not difficult for a third-party eavesdropper to decrypt the message. Assuming English is used, there are only 26 letters; therefore, there are only 26 different key or shift values, only 25 of them actually alter the message, since a shift of 0 or 26 results in no change. Therefore one can easily try and observe all 25 possible shifts until a meaningful message is obtained (Bloomfield, n.d.-a). The Caesar cipher is an example of a substitution cipher, meaning each letter is substituted by some other letter. Other schemes can be used for the substitutions besides the shifting or rotating employed in the Caesar cipher. Any substitution cipher is easily crackable due to letter frequency analysis. For example, in English, ‘E’ is the most common letter at 12.7% and ‘Z’ is the least common letter at 0.1% (Bloomfield,

n.d.-a). Assuming a large enough cipher text is snooped, the most common ciphertext letter likely decrypts to 'E', the least common letter most likely decrypts to 'Z', etc. Much of this information comes from UVA professor Aaron Bloomfield's Intro to Cybersecurity (ICS) course and serves as context and background information for understanding what encryption is and what makes a secure encryption algorithm.

Current encryption algorithms are much more sophisticated than substitution ciphers and rely on certain math problems that are difficult and time-consuming for computers to solve such as integer factorization and discrete logarithms (Gidney & Ekerå, 2021). The integer factorization problem is the easiest to understand conceptually and is utilized by the Rivest–Shamir-Adleman (RSA) algorithm. The most important step of the RSA algorithm to understand is that during encryption two large random prime numbers p and q are selected. From there, the product of the two $n=p*q$ is computed (Bhatia & Ramkumar, 2020; Bloomfield, n.d.-a). Without divulging into the entirety of the RSA algorithm, the security of this algorithm comes from the difficulty of factoring a large product into its prime factors (Bloomfield, n.d.-a). The optimal algorithm for factoring large numbers is the General Number Field Sieve (GNFS), which has big O complexity $O(e^n)$, i.e. the time it takes to factor the number grows exponentially with the size of the number (Bloomfield, n.d.-a). Compared to multiplication, division is a relatively slow operation for a computer to perform. Finding the factors of a number involves repeated division operations. Conventional computers cannot find the factors of an integer any better than brute force, i.e. repeated trial and error (Bhatia & Ramkumar, 2020). For perspective: in 2005, a 193-digit number was factored on 30 2.2 GHz CPUs over 5 months, equivalent to a single CPU running for 12.5 years (Bloomfield, n.d.-a). Considering the fact RSA 2048 uses 2048-bit (617 decimal digits) key sizes and that time to factor grows exponentially with the size of the number,

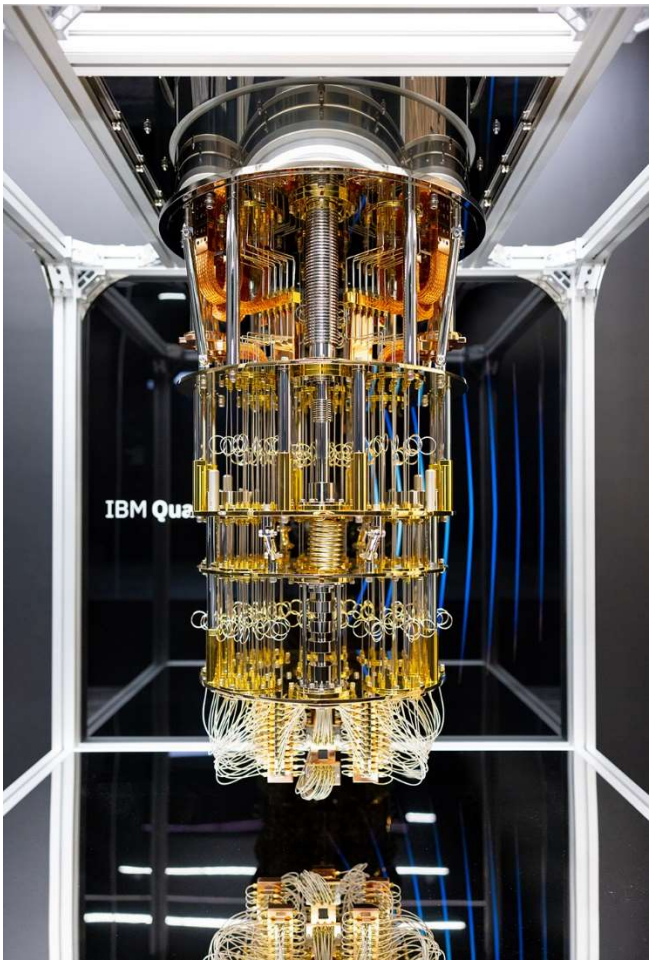
it is estimated that it would take a conventional computer 300 trillion years to crack RSA 2048 (Johnson, 2023). No encryption algorithm is uncrackable, what makes an algorithm secure is the difficulty and thus how long it would take to do so. Since the universe is estimated to be 13.7 billion years old, a conventional computer would take over 21,000 times the age of the universe to crack RSA 2048, rendering it for all intents and purposes secure to attacks by classical computers. The algorithm was published in 1978 (Rivest et al., 1978) and has since been widely utilized across the Internet. If a computer existed that could factor large numbers quickly, then RSA would no longer be secure, which invites the next topic: quantum computers.

A quantum computer is a type of computer that harnesses unique properties of quantum mechanics to solve complex problems faster than classical computers. Classical computing operates on binary bits that can only exist in two states: 0 or 1. A zero bit is electrically represented as 0 volts also known as ground and a one is electrically represented as a higher voltage such as 5 volts. Quantum computing operates on qubits (quantum bits), which can exist in a superposition of states, i.e. simultaneously some combination of 0 and 1 (IBM, 2024; Gidney & Ekerå, 2021; Joseph et al., 2022). Qubits, created from particles like photons or electrons, require extremely precise conditions such as extremely low temperatures to function accurately (IBM, 2024). This creates an exponential increase in information storage capacity and processing power. Four key principles that quantum computing relies on are: superposition, entanglement, decoherence, and interference (IBM, 2024). Superposition allows qubits to exist in a blend of states simultaneously. Quantum entanglement means qubits become intrinsically linked so that the state of one instantly influences another. Decoherence describes the loss of quantum behavior as qubits interact with their environment, collapsing into classical states. Interference enables the constructive and destructive combination of probability amplitudes to

highlight correct computational outcomes (IBM, 2024). Figure 3 shows what a typical quantum computer looks like, kind of like a futuristic golden chandelier.

Figure 3

Quantum Computer



Note. IBM Quantum System One quantum computer. It was created by a partnership with the University of Tokyo and IBM beginning in 2019 (The Government of Japan, n.d.)

Quantum processors operate fundamentally differently from classical computers. Instead of calculating every step sequentially, quantum circuits process enormous datasets

simultaneously. This allows them to explore multiple computational pathways at once and determine the most likely solution probabilistically, rather than delivering a single, deterministic outcome (IBM, 2024). Unlike classical computers that execute one task at a time (assuming a single core CPU, no multithreading), quantum systems can process multiple possibilities in parallel (Bhatia & Ramkumar, 2020; IBM, 2024), potentially solving complex problems like molecular simulations, cryptography, and optimization tasks far more efficiently. Although quantum machines often provide a range of possible answers, which may seem less precise, they can dramatically reduce computation time for extraordinarily complex problems, such as advanced prime factoring or tasks involving massive data sets (IBM, 2024). However, while quantum computing holds great promise for certain specialized applications, it is not the optimal solution for every type of problem. Quantum computing is poised to revolutionize diverse fields by addressing challenges deemed unsolvable by current supercomputing technology.

In 1994, researcher Peter Shor developed a quantum algorithm to find the prime factors of a number exponentially faster than on a classical computer, known as Shor's Algorithm (Bhatia & Ramkumar, 2020; Gidney & Ekerå, 2021; Joseph et al., 2022). A quantum algorithm can only be implemented on a quantum computer; it cannot execute on a classical (non-quantum) computer. In 1996, Lov Grover developed a quantum searching algorithm (Aydeger et al., 2024; Hidary, 2019). Grover's algorithm is a quadratic speed increase (compared to an exponential speedup for Shor's) for the unstructured search problem. It does not break symmetric-key cryptography, however it effectually halves the key length (Aydeger et al., 2024). In summary, Shor's algorithm necessitates a complete replacement for conventional asymmetric algorithms like RSA or Elliptic Curve Cryptography (ECC); while, Grover's algorithm mandates longer keys for symmetric algorithms like the Advanced Encryption Standard (AES).

Why have RSA and other common encryption algorithms remained secure even though Shor's algorithm was invented over three decades ago that could defeat it? The reason is because there has not yet been a cryptanalytically relevant quantum computer (CRQC), which is capable of implementing the algorithm. It is one feat to theorize Shor's algorithm to break RSA or ECC. However, implementing it on a quantum computer is a whole different beast. This gap between theory and engineering implementation is why Shor's algorithm was published in 1994 and as of this writing in 2025, RSA and ECC have not been broken. This is a common occurrence in the progression of technology. After a researcher theorizes that something is possible, scientists and engineers have to actually implement and realize the technology, which can take a while. For example, the fundamental theory that all digital electronics and programming languages are built upon is Boolean algebra, which was first written about by George Boole in 1847 (Boole, 1847). After that, simple computers existed either in mechanical forms or electronic ones made out of vacuum tubes, however they were power-hungry, delicate, and bulky. It was not until one century later with the transistor's invention by AT&T Bell Telephone Laboratories in 1947 (Zorpette, 2022) that marked the dawn of the modern computing revolution.

Quantum computing is a very similar story. Between 1999 and 2001, researcher Yasunobu Nakamura constructed and successfully demonstrated a working, controllable superconducting qubit (Hidary, 2019). In 2001, researchers at IBM Almaden Research Center and Stanford University first demonstrated Shor's Algorithm by factorizing the number 15 into its prime factors 3 and 5 using qubits (Singh & Singh, 2016). The most qubits in a quantum computer was IBM's 433-qubit Osprey machine built in 2022, which was then surpassed by Atom Computing's 1,180 qubit quantum computer in 2023, the first to break a thousand qubits and the current world record (Wilkins, 2023). Higher qubit count does not directly translate to

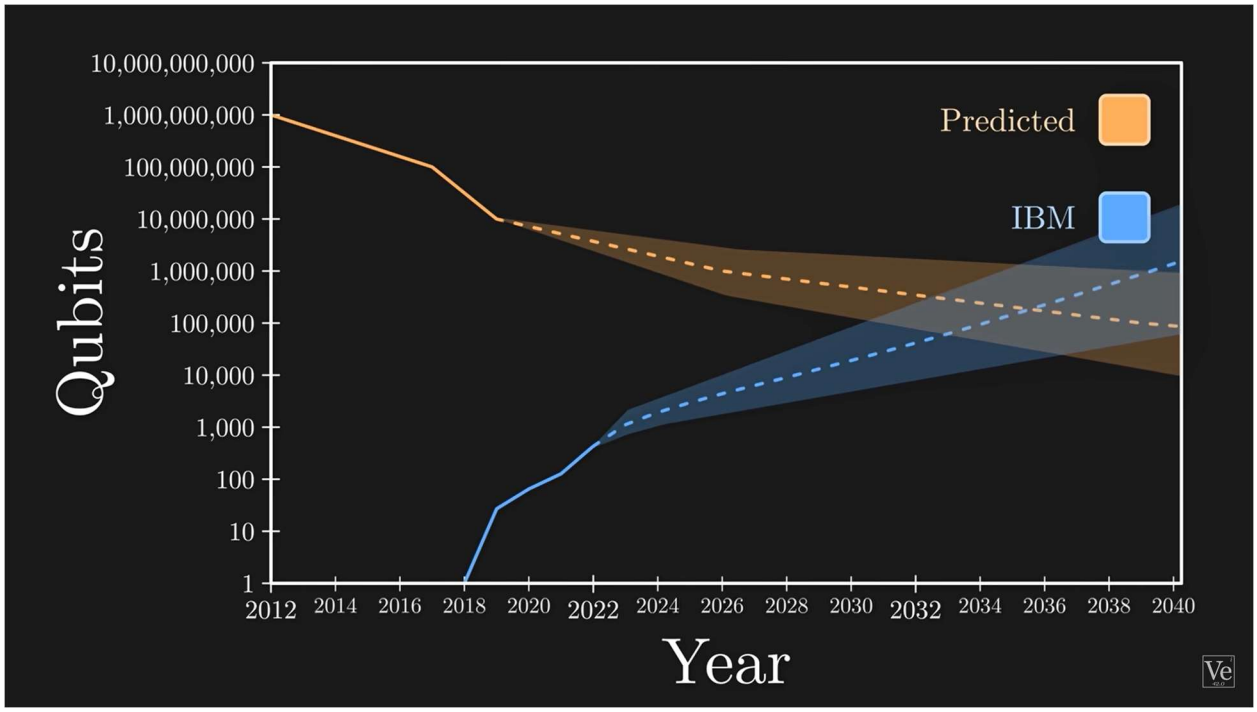
more performance. Running a quantum computer requires highly sophisticated equipment and ideal conditions. Desktop computers, laptops, and servers typically rely on fans for cooling, but quantum processors must be kept near absolute zero (0 Kelvin), at about a 0.01K to minimize noise and prevent decoherence, thereby preserving their quantum states (IBM, 2024). Even at 0.01K, there is still a significant amount of noise. Thus, many extra qubits have to be added for error correction. Microsoft researchers estimate that only about 4000 logical qubits are needed to break 2048-bit RSA and even less, about 2500 logical qubits are needed to crack the Elliptic Curve Discrete Logarithm Problem (ECDLP), a common alternative to RSA (Roetteler et al., 2017). A logical qubit is a perfect qubit operating in ideal conditions with no noise, in practice this translates to many more physical qubits for error correction. It is currently estimated that a quantum computer with twenty million noisy qubits could break RSA in 8 hours (Gidney & Ekerå, 2021). Due to breakthroughs in quantum computing research, this number dropped two orders of magnitude from a billion qubits to twenty million qubits in just 4 years (Gidney & Ekerå, 2021).

As the estimated number of qubits needed to break conventional cryptography continues to drop over time and the qubit count in existing quantum computers continues to rise, there will come a critical point in time when these two curves cross paths as shown in Figure 4. In 2021, over half of the experts surveyed estimated that there was a greater than 50% chance a cryptanalytically relevant quantum computer (CRQC) would be able to break cryptographic systems based on integer factorization and discrete logarithms within the next 15 years (Mosca & Piani, 2021), putting the estimated year at about 2036. Government memorandums from the OMB project the year close by at 2035 (Young, 2022). Then public key encryption such as RSA

and Elliptic Curve Cryptography (ECC) that are ubiquitous throughout digital infrastructure will be insecure.

Figure 4

Projection of when 2048-bit RSA will be insecure



Note. The orange line shows the predicted qubit count to break RSA-2048 dropping over time. The blue line shows the number of qubits in IBM quantum computers increasing over time. Solid lines are historical data, dashed lines represent future projections, and shaded regions around the dashed lines represent the possible uncertainty in predicting the future. From (Veritasium, 2023).

According to a TLS Telemetry Report by F5 Labs, 52% of web servers still allow the use of RSA as of 2021 (Warburton, 2021). After a CRQC is attained, no secret protected by RSA is safe: financial information, medical records, social security numbers, etc. are all vulnerable. In fact, digital thieves steal sensitive data right now in its encrypted form, with the hope they will soon have a CRQC capable of decrypting it, called Store Now Decrypt Later (SNDL) (Joseph et al., 2022). While passwords or bank account info can be changed or updated, there are certain

documents such as matters of national security that need to be protected and are sensitive for decades that these SNDL attacks target. Due to SNDL, the PQC transition is already running behind. There is an ongoing "global geopolitical, military and commercial race" (Cochetti, 2024) to develop the world's first CRQC (Petrova, 2025; Valerio, 2025), much like the space race to land on the moon during the Cold War, or more accurately, the race to develop the first atomic bomb during WWII due to the destructive and powerful implications. Boston Consulting Group estimates that governments around the world have pumped more than $50 billion into advancing quantum technologies (Petrova, 2025). The leaders include China at $15 billion and the U.S. at $5 billion, which excludes the major private-sector investment in America (Cochetti, 2024).

Post-quantum cryptography (PQC) is a relatively new type of cryptography designed to be resistant to both classical attacks as well as to quantum attacks such as Shor's algorithm. (Aydeger et al., 2024; Joseph et al., 2022; Rawal & Curry, 2024). The encryption algorithms in PQC are based on problems that are designed to be difficult and time consuming for quantum computers, much like how prime factorization in RSA is difficult and time consuming for classical computers. PQC algorithms currently exist, however much of the world's digital infrastructure is not using PQC yet (Hekkala et al., 2023).

III. Research Approach: Applying Technological Momentum to PQC Adoption

Technological momentum is a theoretical framework which argues a technology appears more socially constructed in its infancy and seems more deterministic as it becomes embedded in infrastructure (Hughes, 1969). I will use this framework to analyze the evolving social, governmental, and technical dynamics of post-quantum cryptography (PCQ) adoption. Technological momentum is particularly relevant as PQC transitions from an emerging technology to a necessary digital infrastructure.

I examined scholarly research from journals such as *Quantum*, *Nature*, or research databases like *IEEE Xplore*. From these sources I learned about conventional encryption and their vulnerabilities in the face of the quantum threat. I studied the PQC algorithms, the historical timeline as well as future projections regarding PQC integration. These sources also provided great insight into the technical barriers regarding the PQC transition throughout the Internet's infrastructure.

I studied reports from companies that work on quantum computing such as IBM and Microsoft. I likewise researched companies that are working on integrating PQC encryption into their software services such as Microsoft and Apple. Interestingly in the case of Microsoft, sometimes there is overlap, meaning the company is working to accelerate quantum computing technology which will one day break RSA and ECC encryption, but at the same time, the corporation has millions of users which depend on their secure online services, necessitating a PQC transition. From these sources I gained further insight into the technical challenges regarding PQC integration. Additionally, from reading reports from software engineers that are working on PQC incorporation, I was able to glean insight into the social factors that play a role, by studying the people that work in this industry and the challenges they face.

I also analyzed government memorandums, policy documents, and bills. For example, I read the Congressional Bill H.R.7535: Quantum Computing Cybersecurity Preparedness Act (117th U.S. Congress, 2022) signed into law by former President Biden on December 21st, 2022. I also looked at government memorandums such as M-23-02 Memorandum for the Heads of Executive Departments and Agencies: Migrating to Post-Quantum Cryptography (Young, 2022) from the former director of the Office of Management and Budget (OMB) which was issued around the

same time in late 2022. From these sources I gained an understanding about the government's influence in the PQC transition.

A key step in analysis was categorizing the barriers to PQC integration throughout digital infrastructure into three main categories: (1) technical, (2) social, and (3) governmental. I analyzed the three different dimensions of these barriers using the technological momentum framework. The key insight of technological momentum coined by Thomas Hughes is that as a technology becomes deeply entrenched into a society's infrastructure, it gains a certain inertia whereby it is harder to steer and direct. This is especially relevant for cryptographic schemes as their transitions take years or decades, which I explore further in the Results section below. This sociotechnical analysis provides a comprehensive understanding of the PQC adoption landscape beyond purely technical solutions.

IV. Results: Sociotechnical Barriers to PQC Adoption

1. Technical Challenges and Performance Overhead

Widespread cryptographic infrastructure transitions throughout society cannot happen overnight. For instance, Elliptic Curve Cryptography (ECC) was introduced in the 1980s and is significantly more efficient in terms of both space and speed compared to RSA, yet it took more than twenty years to achieve widespread use (Joseph et al., 2022). Similarly, hash functions have also experienced delayed adoption; the National Institute of Standards and Technology (NIST) Secure Hash Algorithm 3 (SHA-3) competition was launched in 2007, its winner was announced in 2012, yet even by 2021, SHA-3 had not been widely embraced (Joseph et al., 2022). Consequently, even relatively simple cryptographic transitions can require years or decades, and the move to post-quantum cryptography is even more challenging due to its novelty and the comparatively lower performance of many proposed algorithms.

When picking an encryption algorithm there are tradeoffs between execution speed, memory size, and security. For example, ECC has widely been used as an alternative to RSA due to being faster and taking up less space in memory. However, ECC is more vulnerable to quantum attacks with only 1000 logical qubits required to break 160-bit ECC compared to double that at 2000 logical qubits needed to factor 1024-bit RSA (Hekkala et al., 2023). Notice there are usually different versions of each algorithm for example 1024-bit RSA or 2048-bit RSA, which has to do with the size of the keys, larger key sizes are more secure, but are slower and take up more space in memory. On the technical side PQC integration involves performance overhead such as larger key sizes (Aydeger et al., 2024), using more network bandwidth and memory. PQC on embedded and Internet of Things (IoT) devices where resources are limited will be a challenge. IoT devices use key sizes of 128-4096 bits whereas PQC key sizes can be from a couple thousand kilobyte to megabyte size (Rawal & Curry, 2024).

PQC algorithms are much more complex than pre-quantum ones. Bugs are hard to find in development because of the use of randomness (Hekkala et al., 2023). Many current cryptographic schemes like AES use hardware acceleration (Aydeger et al., 2024; Hekkala et al., 2023). Hardware acceleration means adding dedicated special-purpose circuitry to the CPU chip to perform parts of the algorithm at the hardware level, rather than the software level, to speed things up. Thus, for PQC to be fast, it is not as simple as just issuing a software update. Rather, entirely new chips need to be designed and manufactured that support PQC hardware acceleration, which takes time.

2. Social Barriers and Lack of Skilled Workers

I examined a guide and memo for developers written in the *SN Computer Science* peer-reviewed journal. Writing a cryptographically secure encryption algorithm is a daunting task and

requires an elite level of mathematical and programming knowledge, which most software developers lack (Hekkala et al., 2023). As such, usually developers rely on a cryptography library, which is a collection of code written by someone else that has been thoroughly tested, rather than implementing it from scratch (Hekkala et al., 2023). Very few open-source libraries have implemented PQC support so far. The creators of OpenSSL, a popular open-source cryptography library, stated they would not add PQC support until standardization, which did not occur until August 2024 (Boutin, 2024). While developing a PQC algorithm takes an advanced amount of rigorous mathematical skill, implementing one in software and hardware is another challenge. If there are bugs in the implementation, it can lead to vulnerabilities, such as the infamous Heartbleed, a severe security flaw in OpenSSL. Heartbleed allowed attackers to read sensitive data from a server's protected memory by exploiting improper bounds checking, exposing private keys, passwords, and other critical information, affecting millions of servers worldwide (Hekkala et al., 2023).

A survey showed that 37% of vulnerabilities in cryptographic software comes from memory and resource management issues (Hekkala et al., 2023). In other words, these are vulnerabilities not related to the algorithm itself, but how it gets translated from math to a programming language like C or C++. A different survey showed only 17% of vulnerabilities came from issues in the library and the other 83% were developers misusing the libraries (Hekkala et al., 2023). Developers make mistakes when they write the libraries and more often when they use them. At the end of the day, human beings like software engineers and system administrators will need to implement the switch to PQC. People make mistakes as evinced in those surveys. Thus, not only will software developers need to be extremely careful when writing cryptographic libraries, but other developers will need to be taught how to use these new

PQC libraries properly. I believe both higher education and industry should bear the responsibility for teaching this. I think anyone with a computer science or computer engineering degree should be taught about the quantum threat and using PCQ libraries, it should not be relegated to a cybersecurity elective class that is completely optional.

3. Government Ineptitude and Bureaucratic Inertia Regarding Cybersecurity

There is a lack of technical expertise and knowledge regarding cybersecurity among political leaders. For example, the proposed Compliance with Court Orders Act (CCOA) of 2016, also known as the Burr-Feinstein Anti-Encryption Bill, required that tech companies decrypt any communication upon court order (Conger, 2016; Pfefferkorn, 2016). Thus, an encryption algorithm would have to be developed such that it can be easily decrypted by someone other than the sending and receiving parties, defeating the entire purpose of encryption. The bill would force companies to weaken their encryption standards by preventing them from designing systems that keep data accessible only to the user. This would undermine modern security features like end-to-end encryption, exposing users to cyber threats. Furthermore, experts contend that criminals and terrorists would still access robust encryption available from overseas, while law-abiding citizens face reduced protection (Pfefferkorn, 2016). The Burr-Feinstein bill was declared dead in May 2016 (Reitman, 2016).

Following the 2015 San Bernadino attack that left 14 innocent people dead and 24 injured, the FBI confiscated an Apple iPhone 5c from one of the perpetrators, both of which were killed by law enforcement. The phone was encrypted and the FBI ordered Apple to write a program to decrypt it for the purposes of criminal investigation (Grossman, 2016). This is commonly known as a “back-door” in cybersecurity, where there is some sort of master key that can decrypt or unlock something and allow one to gain access. Apple refused to help, citing ethical and privacy

concerns that it could be used for mass surveillance (Grossman, 2016). Apple fought back in the court of law, however, eventually the case was dropped after the FBI utilized a third party to unlock the iPhone via an exploit which was later patched in an iOS update (Bloomfield, n.d.-b; Selyukh, 2016). Much like the Burr-Feinstein bill, the Apple-FBI encryption dispute shows that the government has routinely tried to make encryption less secure. A backdoor that could decrypt any iPhone could fall into the wrong hands, say if it were to leak outside of the U.S. control to a foreign adversary or terrorist organization. Consider if U.S. government phones used a similar encryption technique, then this backdoor could be used by a foreign adversary to leak American national security secrets. By setting up a backdoor, the government is shooting itself in the foot by making its own devices less secure. Furthermore, I would consider the U.S. government having a backdoor into cell phones to be dangerous, since the government can and will abuse its power by spying on its law-abiding citizens as evinced by famous whistleblower Edward Snowden.

I have established that the U.S. government has repeatedly tried to weaken encryption, but what does this mean for the transition to post-quantum cryptography (PQC)? I think that if a government, American or foreign, is the first possessor of a CRQC they would try to keep it hidden for as long as possible. No doubt there are military and classified intelligence programs around the world working on quantum computers. Obviously, I cannot prove or cite this due to their clandestine nature, but military technology is often ahead of the curve because of the massive budgets and security concerns driving research and development. Many technologies like GPS, the Internet, drones, and even super glue had military origins before being widely used commercially (NATO, n.d.). These covert military technologies can be decades ahead of

commercial deployment (Cochetti, 2024). This further necessitates a rapid transition to PQC, since we probably will not even know the exact moment RSA and ECC encryption are broken.

The government tends to move at a glacial pace compared to the exponential and explosive rate of growth in the tech industry. Consider the OMB's Policy to Require Secure Connections across Federal Websites and Web Services issued under the Obama administration in June 2015 also known as the HTTPS-Only Standard directive (Scott, 2015). Anyone that has used a web browser has probably seen "http://" or more commonly now "https://" at the beginning of a URL. HTTP stands for hypertext transfer protocol, the application level protocol used by web pages (Cloudflare, Inc., n.d.-b). The S in HTTPS means secure, basically the encrypted version of HTTP, appropriate for sending or receiving sensitive data like bank info, emails, medical documents, etc. HTTPS uses an encryption protocol known as Transport Layer Security (TLS), previously known and sometimes still referred to as Secure Sockets Layer (SSL) (Cloudflare, Inc., n.d.-b). TLS handles three aspects of cybersecurity: encryption to obfuscate the data from third parties, authentication to ensure that the sender or receiver are who they claim to be, and integrity which protects against tampering of the message (Cloudflare, Inc., n.d.-c). It takes only about *18 minutes* to switch an Apache web server to use the encrypted HTTPS instead of the vulnerable plaintext HTTP (Bloomfield, n.d.-c). Yet in the OMB directive, federal government websites were given *18 months* until December 31, 2016 to make the switch (Scott, 2015). There are various different versions of TLS, but it typically uses RSA or ECC for asymmetric public key exchange, which are vulnerable to quantum attacks and need to be replaced in the PQC transition. Whenever a quantum-resistant TLS is available, I believe history shows the government will have an equally slow response in transitioning. I think the best way for the government to accelerate the PQC transition is through private-public partnership. I think the

government should build a task force with the brightest minds from Apple, Google, Microsoft, etc. and ask for their help in transitioning all government services to being quantum-resistant.

On August 13th, 2024, NIST finally standardized three official PQC standards, the culmination of almost a decade of work (Boutin, 2024). Since 2015, NIST assessed over 82 algorithms from cryptography researchers and experts in 25 different countries. The three standards are FIPS 203: Module-Lattice-Based Key-Encapsulation (ML-KEM), FIPS 204: Module-Lattice-Based Digital Signature Algorithm (ML-DSA, and FIPS 205: Stateless Hash-Based Digital Signature Algorithm (SLH-DSA). Two of these 3 algorithms rely on problems related to algebraic lattices which are quantum-resistant (Aydeger et al., 2024; Joseph et al., 2022; Rawal & Curry, 2024) rather than previous methods of integer factorization, discrete logarithms, or elliptic curves.

NIST has historically been the most important government agency in standardizing cyber security algorithms (Joseph et al., 2022). For compliance reasons many organizations must use cryptographic standards in accordance with the Federal Information Processing Standard (FIPS). Before standardization, many companies avoided experimentation with PQC due to the fact they did not want to lose their FIPS compliance. While hybrid algorithms allowed companies to stay compliant by combining the old algorithms like RSA and ECC that were FIPS certified yet quantum-vulnerable with the new PQC methods (Joseph et al., 2022), these were never standardized and were slower due to having to do two separate encryption algorithms. Now that FIPS 203-205 have been standardized, organizations can finally switch to these and not worry about hybrid algorithms or not complying with government standards. Security researchers have known about RSA and ECC vulnerabilities since Peter Shor released his quantum factoring algorithm in 1994; however, NIST did not begin working on standardization until 2015, over two

decades later. It took almost a decade for these standards to be developed and they were released in 2024, just over a decade before RSA and ECC are projected to be broken in about 2035. Seeing as the RSA to ECC transition took two decades to propagate throughout the tech industry and gain widespread use, this does not bode well for the internet being quantum-resistant by 2035.

Conclusion: Towards a Quantum-Resilient Future

The transition to post-quantum cryptography (PQC) is an urgent yet complex challenge, shaped by both technical limitations and sociotechnical barriers. As quantum computing advances toward the capability of breaking conventional encryption, the necessity of implementing PQC becomes increasingly clear. However, as this paper demonstrates, the obstacles to PQC adoption extend far beyond algorithmic complexity. Technical constraints such as increased key sizes, computational overhead, and performance trade-offs pose significant hurdles, particularly for IoT and resource-limited devices (Rawal & Curry, 2024). Additionally, the transition is hampered by a shortage of skilled cryptographers, the slow pace of standardization, and bureaucratic inertia within both public and private sectors (Boutin, 2024; Hekkala et al., 2023).

From a technological momentum perspective, the current state of cryptographic infrastructure reflects the inertia of deeply embedded systems. While encryption is foundational to modern cybersecurity, the widespread adoption of quantum-resistant algorithms requires overcoming resistance from industries, governments, and developers accustomed to pre-quantum cryptographic standards. As history has shown with previous cryptographic transitions, such as the tech industry's gradual shift from RSA to ECC (Joseph et al., 2022) or the government's lethargic switch from HTTP to HTTPS (Scott, 2015), widespread implementation and

government policy does not happen overnight. Without proactive measures, there is a risk that organizations will delay PQC adoption until it is too late, leaving critical systems vulnerable to Store Now, Decrypt Later (SNDL) attacks and future quantum breaches (Joseph et al., 2022).

To mitigate these risks, a coordinated effort is required across multiple domains. Governments must accelerate regulatory frameworks and compliance measures while investing in workforce development to train a new generation of cryptographers and software developers. Companies need to begin the migration process now by integrating PQC schemes into their software systems. Finally, further research is needed to address PQC's practical challenges, particularly in optimizing its performance for constrained environments such as IoT and embedded systems.

The race toward a cryptanalytically relevant quantum computer (CRQC) is already underway. Whether the world's digital infrastructure will be prepared for this paradigm shift remains uncertain. However, what is clear is that the barriers to PQC adoption are not insurmountable. By understanding and addressing these technical, social, and governmental challenges now, we can work toward a quantum-resilient future before the stakes become irreversible.

References

117th U.S. Congress. (2022, December 21). *H.R.7535—Quantum Computing Cybersecurity Preparedness Act* (2022-04-18) [Legislation]. https://www.congress.gov/bill/117th-congress/house-bill/7535/text

Aydeger, A., Zeydan, E., Yadav, A. K., Hemachandra, K. T., & Liyanage, M. (2024). Towards a Quantum-Resilient Future: Strategies for Transitioning to Post-Quantum Cryptography. *15th International Conference on Network of the Future (NoF). IEEE.* https://www.researchgate.net/profile/Madhusanka-Liyanage/publication/382077518_Towards_a_Quantum-Resilient_Future_Strategies_for_Transitioning_to_Post-Quantum_Cryptography/links/668c4711c1cf0d77ffc37c00/Towards-a-Quantum-Resilient-Future-Strategies-for-Transitioning-to-Post-Quantum-Cryptography.pdf

Bhatia, V., & Ramkumar, K. R. (2020). An Efficient Quantum Computing technique for cracking RSA using Shor's Algorithm. *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*, 89–94. https://doi.org/10.1109/ICCCA49541.2020.9250806

Bloomfield, A. (n.d.-a). *Encryption*. Retrieved February 25, 2025, from https://aaronbloomfield.github.io/ics/slides/encryption.html#/title-slide

Bloomfield, A. (n.d.-b). *Ethics*. Retrieved March 30, 2025, from https://aaronbloomfield.github.io/ics/slides/ethics.html#/title-slide

Bloomfield, A. (n.d.-c). *US Cybersecurity Policy*. Retrieved March 3, 2025, from https://aaronbloomfield.github.io/ics/slides/policy.html#/title-slide

Boole, G. (1847). *The Mathematical Analysis of Logic*. https://www.gutenberg.org/ebooks/36884

Boutin, C. (2024). NIST Releases First 3 Finalized Post-Quantum Encryption Standards. *NIST*. https://www.nist.gov/news-events/news/2024/08/nist-releases-first-3-finalized-post-quantum-encryption-standards

Cloudflare, Inc. (n.d.-a). *What is encryption?* Retrieved February 25, 2025, from https://www.cloudflare.com/learning/ssl/what-is-encryption/

Cloudflare, Inc. (n.d.-b). *What is HTTPS?* Retrieved March 3, 2025, from https://www.cloudflare.com/learning/ssl/what-is-https/

Cloudflare, Inc. (n.d.-c). *What is Transport Layer Security (TLS)?* Retrieved March 3, 2025, from https://www.cloudflare.com/learning/ssl/transport-layer-security-tls/

Cochetti, R. (2024, May 5). The quantum computing race is on. *The Hill*. https://thehill.com/opinion/technology/4642324-the-quantum-computing-race-is-on/

Conger, K. (2016, April 14). Burr-Feinstein encryption bill is officially here in all its scary glory. *TechCrunch*. https://techcrunch.com/2016/04/13/burr-feinstein-encryption-bill-is-officially-here-in-all-its-scary-glory/

Gidney, C., & Ekerå, M. (2021). How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. *Quantum*, *5*, 433. https://doi.org/10.22331/q-2021-04-15-433

Grossman, L. (2016, March 17). *Inside Apple CEO Tim Cook's Fight With the FBI: Exclusive*. TIME. https://time.com/4262480/tim-cook-apple-fbi-2/

Hekkala, J., Muurman, M., Halunen, K., & Vallivaara, V. (2023). Implementing Post-quantum Cryptography for Developers. *SN Computer Science*, *4*(4), 365. https://doi.org/10.1007/s42979-023-01724-1

Hidary, J. D. (2019). A Brief History of Quantum Computing. In J. D. Hidary (Ed.), *Quantum Computing: An Applied Approach* (pp. 11–16). Springer International Publishing. https://doi.org/10.1007/978-3-030-23922-0_2

Hughes, T. P. (1969). Technological Momentum In History: Hydrogenation In Germany 1898–1933. *Past and Present*, *44*(1), 106–132. https://doi.org/10.1093/past/44.1.106

IBM. (2024, August 5). *What Is Quantum Computing?* https://www.ibm.com/think/topics/quantum-computing

Johnson, J. (2023). The Vulnerabilities to the RSA Algorithm and Future Alternative Algorithms to Improve Security. *Cybersecurity Undergraduate Research Showcase*, *7*. https://doi.org/10.25776/mgp4-kz08

Joseph, D., Misoczki, R., Manzano, M., Tricot, J., Pinuaga, F. D., Lacombe, O., Leichenauer, S., Hidary, J., Venables, P., & Hansen, R. (2022). Transitioning organizations to post-quantum cryptography. *Nature*, *605*(7909), 237–243. https://doi.org/10.1038/s41586-022-04623-2

Mosca, M., & Piani, M. (2021). Quantum Threat Timeline. *Global Risk Institute*. https://globalriskinstitute.org/publication/2021-quantum-threat-timeline-report-global-risk-institute-global-risk-institute/

NATO. (n.d.). *Military inventions that we use every day*. NATO. Retrieved March 30, 2025, from https://www.nato.int/cps/en/natohq/declassified_215371.htm

Petrova, M. (2025, March 21). *Why startups and tech giants are racing to build a practical quantum computer*. CNBC. https://www.cnbc.com/2025/03/21/the-race-to-build-a-practical-quantum-computer.html

Pfefferkorn, R. (2016, April 15). *Here's What the Burr-Feinstein Anti-Crypto Bill Gets Wrong*. Just Security. https://www.justsecurity.org/30606/burr-feinstein-crypto-bill-terrible/

Rawal, B. S., & Curry, P. J. (2024). Challenges and opportunities on the horizon of post-quantum cryptography. *APL Quantum*, *1*(2), 026110. https://doi.org/10.1063/5.0198344

Reitman, R. (2016, May 27). *Security Win: Burr-Feinstein Proposal Declared "Dead" for This Year*. Electronic Frontier Foundation. https://www.eff.org/deeplinks/2016/05/win-one-security-burr-feinstein-proposal-declared-dead-year

Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, *21*(2), 120–126. https://doi.org/10.1145/359340.359342

Roetteler, M., Naehrig, M., Svore, K. M., & Lauter, K. (2017). *Quantum resource estimates for computing elliptic curve discrete logarithms* (No. arXiv:1706.06752). arXiv. https://doi.org/10.48550/arXiv.1706.06752

Scott, T. (2015). *M-15-13 Policy to Require Secure Connections across Federal Websites and Web Services*. https://obamawhitehouse.archives.gov/sites/default/files/omb/memoranda/2015/m-15-13.pdf

Selyukh, A. (2016, March 28). The FBI Has Successfully Unlocked The iPhone Without Apple's Help. *NPR*. https://www.npr.org/sections/thetwo-way/2016/03/28/472192080/the-fbi-has-successfully-unlocked-the-iphone-without-apples-help

Singh, J., & Singh, M. (2016). Evolution in Quantum Computing. *2016 International Conference System Modeling & Advancement in Research Trends (SMART)*, 267–270. https://doi.org/10.1109/SYSMART.2016.7894533

The Government of Japan. (n.d.). *Pursuing the 100,000-Qubit Quantum Computer Through Japan-U.S. Collaboration*. JapanGov. Retrieved March 1, 2025, from https://www.japan.go.jp/kizuna/2024/03/100000_qubit_quantum_computer.html

Valerio, P. (2024, December 12). A Quantum Leap Forward by Google. *EE Times*. https://www.eetimes.com/a-quantum-leap-forward-by-google/

Valerio, P. (2025, January 14). A Global Race for Supremacy in Quantum Computing. *EE Times*. https://www.eetimes.com/a-global-race-for-supremacy-in-quantum-computing/

Veritasium (Director). (2023, March 20). *How Quantum Computers Break The Internet... Starting Now* [Video recording]. https://youtu.be/-UrdExQW0cs?t=1058

W3Techs. (n.d.). *Usage Statistics and Market Share of Cloudflare, February 2025*. Retrieved February 25, 2025, from https://w3techs.com/technologies/details/cn-cloudflare

Warburton, D. (2021, October 20). *The 2021 TLS Telemetry Report*. F5 Labs. https://www.f5.com/labs/articles/threat-intelligence/the-2021-tls-telemetry-report

Wilkins, A. (2023, October 24). *Record-breaking quantum computer has more than 1000 qubits*. New Scientist. https://www.newscientist.com/article/2399246-record-breaking-quantum-computer-has-more-than-1000-qubits/

Young, S. D. (2022). *M-23-02 Memorandum for the Heads of Executive Departments and Agencies: Migrating to Post-Quantum Cryptography*. https://www.whitehouse.gov/wp-content/uploads/2022/11/M-23-02-M-Memo-on-Migrating-to-Post-Quantum-Cryptography.pdf

Zorpette, G. (2022, November 20). *How the First Transistor Worked*. IEEE Spectrum. https://spectrum.ieee.org/transistor-history