SCTE CABLE-TEC EXPO FAST FORWARD 2021
VIRTUAL EXPERIENCE
OCTOBER 11-14

2021 Fall Technical Forum
SCTE • NCTA • CABLELABS

Navigating the Transition to a Post-Quantum World

A Technical Paper prepared for SCTE by

Chujiao Ma
Senior Security R&D Engineer
Comcast Cable Communications, LLC
Philadelphia, PA, USA
Chujiao_ma@comcast.com

Vaibhav Garg
Sr. Director Cybersecurity Research & Public Policy
Comcast Cable
Blacksburg VA
Vaibhav_garg@comcast.com

Table of Contents

Title **Page Number**

1. Introduction ... 3
2. Post-Quantum Cryptography ... 4
   - 2.1. Quantum-Safe Algorithms ... 4
   - 2.2. NIST PQC Standardization ... 5
   - 2.3. CACR Competition ... 8
   - 2.4. Europe ... 9
3. PQC Implementations ... 9
   - 3.1. Libraries ... 10
   - 3.2. Hybrid ... 10
4. PQC Infrastructure ... 11
   - 4.1. Certificates and PKIs ... 11
   - 4.2. Protocols ... 12
   - 4.3. Cloud Solutions ... 13
5. Crypto Agility ... 13
6. Conclusion ... 15

Abbreviations ... 15

Bibliography & References ... 16

List of Figures

Title **Page Number**

Figure 1. Ciphertext and public key sizes for NIST 3rd round key exchange + classical PKE [20] ... 7
Figure 2. Signature and public key sizes for NIST 3rd round and classical digital signature [20] ... 8

List of Tables

Title **Page Number**

Table 1. NIST Finalists: Public-key Encryption ... 5
Table 2. NIST Finalists: Digital Signature Algorithms ... 6
Table 3. CACR Competition Finalists ... 9

1. Introduction

Quantum computing is an emerging technology that can dramatically change the security landscape. Unlike traditional computation that processes information in binary bits, 0 or 1, the information in quantum computing is stored in a particle in a quantum state called a "qubit." Qubits exist in a superposition, which means they can be 0 or 1 or everything in between, until measured. This allows quantum computers to simultaneously perform computations for a range of inputs. In practice, this reduces the computational complexity of certain algorithms from exponential to polynomial time -- from $O(e^n)$ to $O(n)$. This means that to solve certain classes of problems that would take classical computers hundreds of years, quantum computers may only take days or even hours.

For example, Grover's algorithm provides a quadratic speedup on unstructured search problems [1], whereas Shor's algorithm can be used to factor the product of two large prime numbers in polynomial time [2]. This has an impact on the security of current crypto systems. Grover's algorithm halves the security of current symmetric keys and Shor renders all public key cryptosystems insecure [3]. These classical public key algorithms are used ubiquitously in security protocols for digital signatures, authentications, key transport and authorization. These algorithms secure cyber infrastructure, from software distributions to virtual private networks. Thus, the construction of a large enough quantum computer will require a transition to quantum-safe alternatives to ensure the continued security of these systems.

It is unclear whether such computing capacity will be available in the near future -- but it's a matter of when, not if. Many experts posit that there is a substantial probability of this happening in the next 20 years [4]. This may create the impression that the threats associated with quantum computing have a long-time horizon. However, it is important to consider the challenges in crypto transitions. For example, the transition from Secure Hash Algorithm 1 (SHA1) to Secure Hash Algorithm 2 (SHA2) took over 10 years and cost organizations $5M on average [5]. In that light, a transition across all cryptography can seem overwhelming. However, there is much cause to keep calm and carry on.

In the U.S., the National Institute of Standards and Technology (NIST) has a process underway to determine a list of post-quantum cryptography (PQC) algorithms to replace current public-key cryptography [6]. Open Quantum Safe has open-source implementations of many PQCs for benchmarking and exploration [7]. Additional support is available from European Union projects PQCrypto and SAFEcrypto, as well as CREST Crypto-Math project in Japan [41]. Cloud computing providers, such as Amazon Web Services (AWS), already provide the option of using hybrid cryptography, which combines PQC within a classical cryptography wrapper [8].

There are three components of a quantum transition. First is the choice of crypto algorithms themselves. Second is the implementations of these algorithms in core technologies such as FPGAs. The third component is the evolution of supporting infrastructure to support PQC. For best results, these three components get combined in a broader crypto agility strategy. This sets the stage for a smoother transition, within a time period commensurate with an organization's risk tolerance.

In this paper we help you navigate a transition to the post-quantum world. We begin by providing an overview of post-quantum cryptography, including the various standardization efforts. Next, we introduce the different implementations that currently exist for incorporating PQC algorithms into your infrastructure. Then we discuss the state of complementary solutions -- specifically certificates, protocols, and cloud computing. We also describe crypto agility frameworks that can be used to develop a transition strategy and identify potential gaps. Finally, we close with a roadmap to help you move forward efficiently according to your organizational needs.

2. Post-Quantum Cryptography

All current public-key crypto-systems assume that certain mathematical problems are hard to solve, i.e. the time to solve them on classical computers increases exponentially in proportion to the size of the input. For example, RSA assumes that factoring the product of two very large prime numbers is difficult to do with current computing technology. Thus, the security of RSA is predicated on this assumption staying true. A large enough quantum computer may render these assumptions false [3]. For example, Shor's algorithm can factor RSA keys in polynomial time [2]. Consequently, any transition to a quantum safe future requires new classes of algorithms with hardness assumptions that will not be impinged upon by quantum computers [13].

The impact of quantum computing will be different based on the type of algorithms. According to NIST [41], larger key sizes for symmetric key and larger output for hash functions may be needed to ensure security in the post-quantum world, while public key cryptography such as RSA (Rivest Shamir Adelman), ECDSA (Elliptic Curve Digital Signal Algorithm), ECDH (Elliptic Curve Diffie-Hellman) and DSA will no longer be secure. Thus, the development of new classes of crypto algorithms focuses on public key cryptography used for key exchange and digital signature schemes. In this section we provide an overview of these efforts.

2.1. Quantum-Safe Algorithms

Quantum-resistant cryptography is primarily based on one of six different mathematical problems. Each problem has distinct hardness assumptions as well as pros and cons in terms of performance. These are listed below:

1) **Lattice-based cryptography** is based on the hardness of well-studied lattice problems in the construction itself or in the security proof. Two popular sub-categories of it are NTRU signature and ring-LWE (Learning With Errors.) These algorithms are simple, efficient, and parallelizable. However, they have larger public key sizes than RSA. Additionally, it is difficult to give precise estimates of the security using known cryptanalysis techniques [14].
2) **Code-based cryptography** relies on error-correcting codes. Examples include McEliece encryption algorithm and CFS (Courtois-Finiasz-Sendrier) signatures. They have large key sizes and attempts to reduce them so far all resulted in compromised security. There has been more
3) with implementing it for encryption than for signatures [4].
4) **Hash-based signatures** are digital signatures constructed using hash functions such as Merkle signature scheme and XMSS (Extended Merkle Signature Schemes.) The security of hash functions is well studied. However, corresponding schemes can only produce a limited number of signatures, and many require a secure record of the exact number of previously signed messages. Together with the much larger signature, the drawbacks make it tricky to implement for large-scale environments [43].
5) **Multivariate cryptography** is based on the difficulty of solving systems of multivariate polynomial equations over a single finite field. The multivariate encryption schemes are not very efficient, due to large public keys and long decryption times. However, they are more successful for building signature schemes because they provide some of the shortest signatures among the post-quantum algorithms [4].
6) **Isogeny** uses mathematics of super-singular elliptic curves and super-singular isogeny graphs to create a Diffie-Hellman-like key exchange. This mathematical problem is the most recent basis for any post-quantum candidates and is therefore less studied. However, it has one of the smallest key sizes [48].

7) **Zero knowledge proof (ZKP)** proves validity without revealing underlying information. It is currently only used by one PQC, Picnic, where it was made non-interactive and turned into a signature scheme using the traditional Fiat-Shamir transform [15].

   The mathematical structure for different PQC algorithms varies widely. A detailed discussion is beyond the scope of this paper and is available elsewhere [16].

2.2. NIST PQC Standardization

In 2017, NIST started a post-quantum cryptography standardization effort to select algorithms that will supplement or replace existing public key cryptography. There were 82 submissions received in the 1st round, and 69 accepted, with a focus on the security analysis. Round 2 was started in 2019 with 26 candidate algorithms, and a focus on the hardware and software performance as well as security. To keep the diversity but reduce the numbers, NIST encouraged mergers of similar submissions. In July 2020, NIST announced the candidates for the third round, which included 7 primary and 8 alternate candidates. At the time of this writing (summer 2021), the finalists are still being reviewed for standardization at what is the conclusion of the third round. Algorithms with structured lattice schemes appear to be the most promising general-purpose algorithms for public key encryption and digital signature schemes. Several of the alternate candidates have worse performance than the finalists but might be selected for standardization if there's high confidence in their security. Others have acceptable performance but require additional analysis to inspire sufficient confidence in their security [6]. NIST will select which alternates to keep studying in a 4th round and expect the finalized standard to be ready around 2024 [40].

NIST's standardization effort focuses on two categories of PQC algorithms: 1) public key encryption/key establishment and 2) digital signatures. The current candidates all offer a range of security based on the parameter set, that range from two to eighteen. The key sizes and ciphertext sizes differ based on the parameter set selected, as shown in Table I and Table II. The * denotes an alternate candidate.

Table 1. NIST Finalists: Public-key Encryption

Name	Type	Public Key (bytes)	Private Key (bytes)	Ciphertext Size (bytes)
Classic McEliece [9]	Code-based	261120 - 1357824	6492 - 14120	128 - 240
Crystals-Kyber [10]	Lattice	800 - 1568	1632 - 3168	768 - 1568
NTRU [11]	Lattice	699 - 1230	935 - 1590	699 - 1230
Saber [11]	Lattice	672 - 1312	1568 - 3040	736 - 1472
*BIKE [11]	Code-based	2542 - 6206	3110 - 13236	2542 - 6206
*FrodoKEM [11]	Lattice	9616 - 21520	19888 - 43088	9729 - 21632
*HQC [11]	Code-based	2249 - 7245	2289 - 7285	4481 - 14469
*NTRU Prime [11]	Lattice	897 - 1322	1125 - 1999	1025 - 1184
*SIKE [11]	Isogeny	197 - 564	28 - 644	197 - 596

Table 2. NIST Finalists: Digital Signature Algorithms

Name	Type	Public Key	Private Key	Signature
Crystals-Dilithium [11]	Lattice	1312 - 2592	2544 - 4880	2420 - 4595
Falcon [11]	Lattice	897 - 1793	1281 - 2305	690 - 1330
Rainbow [11]	Multivariate	60192 - 1930600	64 - 1408736	66 - 212
*GeMSS [12]	Multivariate	352000 - 10400000	13100 - 12300	240000 - 600000
*Picnic [11]	ZKP	33 - 65	49 - 97	14612 - 209510
*SPHINCS+ [11]	Hash based	32-64	64-128	8080 - 49216

The appropriate PQC algorithm may depend both on the security and the asset constraints. Consider the public key encryption candidates. Lattice-based algorithms such as NTRU and Saber have a much smaller public/private key size than the code-based Classic McEliece. However, Classic McEliece has a smaller ciphertext size than the lattice-based algorithms. Similarly for digital signature algorithms multivariate-based Rainbow has a much larger key size, but much smaller signature size, than the lattice-based Falcon and Crystals-Dilithium.

PQC generally has larger key sizes, but some algorithms at lower security levels have a comparable size to classical algorithms at a higher security level. The five security levels are denoted as:

- Level 1: At least as hard to break as AES-128 using exhaustive key search.
- Level 2: At least as hard to break as SHA-256 using collision search.
- Level 3: At least as hard to break as AES-192 using exhaustive key search
- Level 4: At least as hard to break as SHA-384 using collision search.
- Level 5: At least as hard to break as AES-256 using exhaustive key search.

The focus of NIST competition is on security level 1, 2, and 3. Take the key exchange algorithms, for example, as shown in Figure 1. SIKE, SABER, Crystal-Kyber and NTRU all have comparable key and ciphertext sizes with RSA/DH. For the digital signature candidates, shown in Figure 2, Falcon and Crystals-Dilithium have similar signature and public key sizes as RSA.

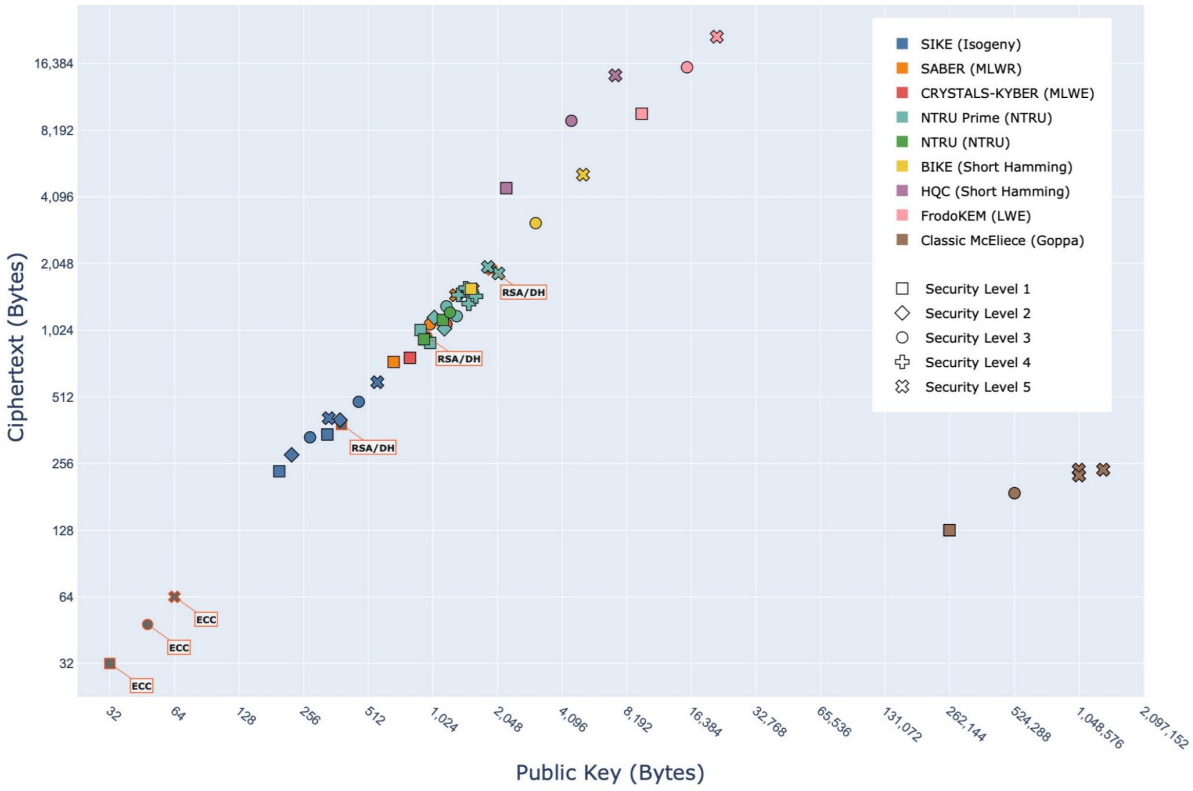

Figure 1. Ciphertext and public key sizes for NIST 3rd round key exchange + classical PKE [20]

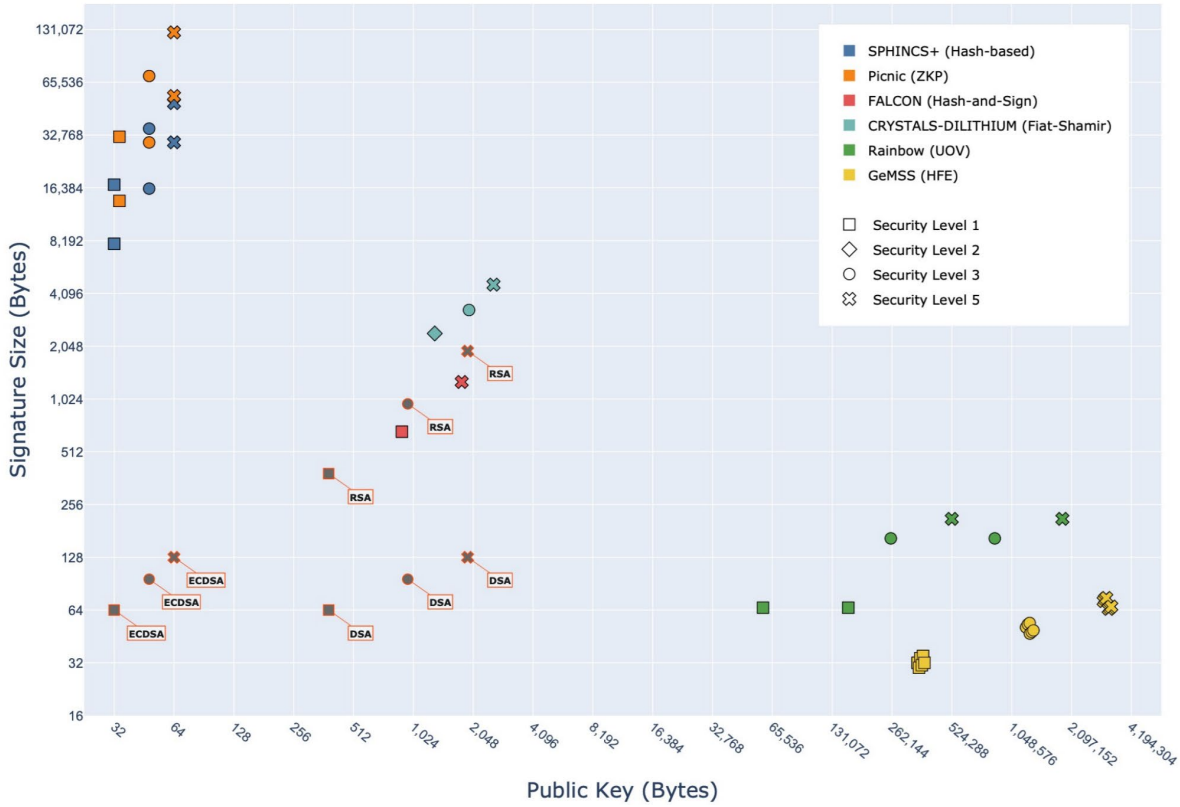

Figure 2. Signature and public key sizes for NIST 3rd round and classical digital signature [20]

2.3. CACR Competition

Aside from NIST, China also held their own post-quantum cryptography competition. The Chinese Association for Cryptologic Research (CACR) issued a notice of algorithm competition in August of 2018. The competition focused on functionalities, security, and performance. While NIST separated the competition into two tracks (digital signature and public key crypto/key encapsulation), China separated the competition into three tracks: digital signature, public key cryptography and key exchange. The first round received 36 submissions, with the majority being lattice-based algorithms. Unlike NIST, which included multiple rounds, the CACR competition only had one round and concluded in December 2019. The results of round 1 contained 14 finalists with 1st, 2nd, and 3rd place winners across the three categories [21]. The list of finalists is provided in Table III and the details are from the Announcement of the Results of the National Cryptographic Algorithm Design Competition Algorithm Selection, a technical report published by the CACR in 2020 [22].

Table 3. CACR Competition Finalists

Rank	Name	Category	Type
1st place	Aigis-sig	Signatures	Lattice
1st place	LAC.PKE	KEM	Lattice
1st place	Aigis-enc	KEM	Lattice
2nd place	LAC.KEX	KEX	Lattice
2nd place	SIAKE	KEX	Isogeny
2nd place	SCloud	KEM	Lattice
2nd place	AKCN	KEM	Lattice
3rd place	OKCN (SKCN-MLWE)	KEX	Lattice
3rd place	Fatseal	Signature	Lattice
3rd place	Mulan	Signatures	Lattice
3rd place	AKCN-E8	KEM	Lattice
3rd place	TALE	KEM	Lattice
3rd place	PKP-DSS	Signature	PKP
3rd place	Piglet-1	KEM	Code-based

2.4. Europe

While Europe did not hold its own post-quantum competition, there have been multiple initiatives exploring the current solutions. The European Technology and Standards Institute (ETSI) is a recognized regional standards body dealing with telecommunications, broadcasting, and other networks and services. It recognized that cryptanalysis and the standardization of algorithms require significant time and effort for their security to be trusted by governments and industry. Thus, ETSI is taking a proactive approach and formed the Cyber Quantum Safe Cryptography (QSC) Working Group to assess and make recommendations for quantum-safe cryptographic primitives, protocols, and implementation considerations. The focus is on practical implementation with consideration of performance, capabilities, benchmarking, architecture, and protocols, instead of development of the cryptographic primitives [14]. The European Union Agency for Cybersecurity (ENISA) also published a study that provides an overview of NIST finalists as well as proposals that system owners can implement now in order to protect the confidentiality of their data against a quantum-capable attacker [4].

3. PQC Implementations

The performance of PQC algorithms will depend on their implementation and deployment environments. Most of the published benchmarks are based on algorithms from round 2 of NIST's competition. They include benchmarking of seven lattice-based algorithms [17]; benchmarking of algorithms in hardware [18]; benchmarking using FPGA [20] and benchmarking in TLS [19]. Because these algorithms are currently evolving, their actual performance may differ from published benchmarks, which is also informed by implementation platforms. The most comprehensive benchmarking of the round 3 algorithms, as of this publication, is from the Open Quantum Source (OQS) profiling project and includes runtime, memory use and performance on x86 [44]. The code for OQS profiling is open source and geared toward collecting information across the algorithms at different levels of the software and network stack. It does not provide testing at the raw algorithm level, which can be done using SUPERCOP, a

toolkit that measures crypto primitives according to length of the key/message and time to generate, encrypt or authenticate [45]. Because each PQC algorithm has its own limitations and might be appropriate for distinct platform or assets, it's best to test and explore the implementation challenges on your target device.

3.1. Libraries

There are several libraries that implement PQC for distinct systems and corresponding requirements. One of the earliest is libpqcrypto, a cryptographic software library produced by the PQCRYPTO project that includes software for 77 cryptographic systems. It includes AES-256 and Salsa20 for symmetric/secret-key cryptography, McBits for public-key cryptography and SPHINCS+ for signatures. PQCRYPTO can be used with OpenSSH and OpenIKED. The project includes a benchmarking and testing framework as well as libraries for ARM Cortex-M4 and FPGAs. However, the library is research-oriented and not ready for use in production environments [23].

Another library that is more widely used is liboqs from Open Quantum Safe (OQS) [7]. It is an open-source library written in C and has well developed support for many platform and languages. The library can run on Linux, Mac and Windows. It supports x86 and ARM architectures, as well as compilers from Clang, GCC (GNU Compiler Collection) and Microsoft. It also contains language wrappers for C++, Go, Java, .Net, Python and Rust. OQS provides code for integration into TLS, SSH, x.509, CMS and S/MIME via OpenSSL and OpenSSH. The library has been used by many external projects, including Microsoft Post-Quantum Cryptography VPN, Mullvad VPN, Thales eSecurity Go wrapper, Liesware Coherence Cryptographic Server, IBM Cloud and others [39]. However, it is built as a library for testing and not commercial deployment, so some components are more mature than others. Some algorithms implemented have large stack usage and may cause failures when run on threads or in constrained environments [11].

In addition to general libraries, Cloudflare released the source code of CIRCL, a cryptographic library written in Go, in 2019. It contains a package that combines an implementation of Diffie-Hellman with SIKE (Super-singular Isogeny Key Encapsulation), allowing developers to experiment with post-quantum key exchange schemes for TLS 1.3. They are currently looking to add lattice-based algorithms such as NTRU and Crystals-Kyber, and post-quantum signature algorithms to CIRCL [24].

Aside from open-source libraries, there are commercial alternatives. ISARA's Radiate Quantum Safe Toolkit supports system Android, iOS, Linux, macOS, Windows 10 and FreeBSD Crypto library and integration tools. It also includes a hybrid mode that has PQC but is backward compatible and maintains the current security measures [25]. Then there's also PQshield, which helps customer transition to quantum-secure standards with a post-quantum cryptography library, hardware for embedded devices, an SDK for mobile and server, as well as a solution for messaging platforms and apps [26].

Other efforts in post-quantum cryptography are happening within the United Arab Emirates. The Technology Innovation Institute (TII), part of Abu Dhabi's Advanced Technology Research Council (ATRC), made available its first PQC software library for the nation in early 2021. Not much is known about the algorithms used, but the library is written in C and supports a wide variety of architectures and OS. A hardware (FPGA-based) implementation has also been developed. The first release of the library has already been integrated in several secure communication products [27].

3.2. Hybrid

There are currently two options to implement post-quantum cryptography, either by replacing current public key algorithms such as RSA and ECDHE, or as part of hybrid cryptosystems. Post-quantum cryptography

has not yet been tested with a real quantum computer, so there is a risk of security or implementation flaws. Thus, many current solutions are exploring the option of hybrid cryptography.

Hybrid cryptosystems combine two or more different cryptographic techniques to perform the same function. There are three different types of hybrids: classical/quantum-safe hybrids, which secure against classical attacks at least; quantum-safe/quantum-safe hybrids, which are good for future attacks, but the security of quantum-safe algorithms remain uncertain; and classical/QKD hybrids, which combine the security of classical algorithms with the only one known quantum-safe method. Hybrid cryptosystems will in theory remain secure if at least one of the underlying cryptographic schemes remains unbroken. However, they can be slower, have a larger footprint for key storage, and be less efficient [8].

4. PQC Infrastructure

The first step to a PQC transition is to identify which algorithm is appropriate for your asset and explore the implementations of these algorithms in core technology. However, because these algorithms are drastically different from classical algorithms, from mathematical foundation to key sizes to performance overhead, there may be changes needed in the infrastructure to support both them and the transition to them. The infrastructures most affected are those that use public-key cryptography, such as certificates and PKI, protocols and cloud solutions. In this section we'll explore some of the changes needed in the infrastructures when transitioning to PQC.

4.1. Certificates and PKIs

The most common certificate sizes on the internet today vary between 500-1500 bytes. Most of the proposed post-quantum schemes have public key and signature sizes of 10-200 kilobytes (kb), which is significantly bigger and can pose challenges for the infrastructures that would use them in X.509 certificates. These challenges include transmission overhead, IP fragmentation and wasted bandwidth for connections. To support new proposed post-quantum signature schemes in X.509, new algorithm identifiers that correspond to certain post-quantum signature scheme parameters and structures will need to be defined [28]. While the x.509 data format allows for long public keys and signatures, some applications may put size limits on the x.509 fields. Also, the cost and performance overhead will differ depending on the implementation and usage of the certificates. Some devices or system may or may not be fully upgradeable due to software or hardware limitations.

The transition to PQC can be a huge undertaking that takes a long time. To ensure a smooth transition, there will be a need for a certificate that can work with both PQC-enabled systems and non-upgraded systems, or hybrid certificate. A hybrid certificate is an X.509 certificate with additional quantum safe components, so you only need to support one certificate instead of two no matter the system. The hybrid certificate would contain extra X.509 certificate fields for quantum-safe keys and signatures as well as encoding for a quantum safe algorithm. NIST has updated the guidance on transition in SP800-56C Rev. 2 to permit the use of hybrid mode. In hybrid mode, an unapproved (i.e. PQC) algorithm can be combined with a NIST-approved algorithm and still receive FIPS validation [42].

One way to implement a hybrid certificate is to offer the option of choosing to use the classical or post-quantum algorithm. This allows the relying parties that cannot update their cryptographic suites to be in the same infrastructure as other relying parties that use a stronger validation algorithm. Another way is to implement the certificate such that it encrypts/decrypts using both classical and post-quantum algorithms. Instead of choosing one or the other, the server/client now needs to use both. This way the certificate is

protected against classical and quantum attacks. However, this does present an implementation challenge, because you need to upgrade the PKI system, and maybe the servers and the clients as well. The signing and validation might also need to be upgraded [29]. There are currently multiple collaborative efforts on new digital certificate formats that can work with both classic and post-quantum algorithms from ISARA, Cisco, CableLabs, DigiCert and Entrust. For more details, refer to [46] for a method of embedding alternative sets of cryptographic materials into digital certificates as well as how creation, verification, signing and revocation would work in such cases.

4.2. Protocols

Once successfully adopted, security protocols tend to be long lived in products and networks. Thus, protocols typically allow for some elements of flexibility in changes to the key sizes and cryptographic parameters in case of algorithm degradation. However, protection against quantum attacks may require more drastic changes, where the cryptographic primitives may need to be replaced entirely or protocol-level changes may be needed. This can be an easy or difficult process depending on how crypto-agile the protocols are. An overview is provided here, and more details can be found in the white paper from ETSI [14].

Internet Key Exchange (IKEv2) is a protocol used mainly for setting up VPNs, using three exchanges to set up a security association. First, a common key is derived using the Diffie-Hellman key agreement algorithm. Second, the key is authenticated using certified digital signatures or pre-shared authentication key. Thirdly, the key agreement is conducted again to generate new ephemeral keys for the IP packet. The protocol standard is rigid and only offers a small set of cryptographic algorithms. Making IKE quantum-safe will require replacing the algorithms used in all three exchanges.

The Transport Layer Security (TLS) protocol, previously SSL, establishes a protected tunnel between a client and server for transmission of application data. It starts with a handshake sub-protocol that authenticates server and client, then establishes shared secret keys for transmission of application data. The shared secret keys are then used in the record layer subprotocol to encrypt and authenticate application data. The handshake uses public key cryptography and will have to be replaced with quantum-safe alternatives. The subsequent record sub-protocol uses symmetric key cryptography and just needs to increase the key sizes. The design of TLS is largely independent of cryptographic algorithms and allows the parties to negotiate the cipher suites to be used. While quantum-safe algorithms with large public keys or signatures may require additional changes to the standard, there are currently libraries available to help test the post-quantum algorithm implementations and identify implementation challenges.

Secure/Multipurpose Internet Mail Extension (S/MIME) is used to securely send email messages. It allows email to remain encrypted during the entire path from sender to recipient, preserving end-to-end confidentiality and data integrity. Content encryption in S/MIME relies upon symmetric ciphers and is believed to be quantum-safe. However, the digital signatures for authentication and integrity use DSA or RSA, which will need to be replaced with quantum-safe alternatives. S/MIME does support extended key size and encryption methods, so it is possible to upgrade signature and key-establishment algorithms without replacing the entire protocol.

Secure Shell (SSH) is used to encrypt information sent over an insecure network and allows remote login, file transfer or operations without compromising data integrity or confidentiality. The SSH protocol involves three major sub-protocols: 1) The transport layer protocol that creates a secure channel and runs over top of TCP/IP; 2) The user authentication protocol that authenticates the client to the server; and 3) The connection protocol that takes the encrypted tunnel generated by transport layer and multiplexes it into

several channels for login, proxy forwarding and accessing secure subsystems etc. The SSH protocol includes a high level of cryptographic agility and allows servers and clients to negotiate the algorithms, so the addition of quantum-safe controls should not require significant changes to the base SSH protocol.

4.3. Cloud Solutions

At its most basic level, quantum-safe solutions involve using quantum-safe cryptography, supported by certificates and protocols that accept the quantum-safe option. At a higher level, quantum-safe cloud computing means quantum-safe server, endpoint, and network infrastructure. The Cloud Security Alliance has published a note on cryptanalytic and mathematical research that builds meaningful confidence in the algorithms' security [30]. It's not an analysis on implementation, performance or application to protocol. However, many companies have already taken steps to explore performances and integrate some post-quantum cryptography into their offerings.

Google took a first step towards post-quantum cryptography by researching and prototyping lattice-based public-key cryptography. In 2016, Google launched an experiment to incorporate the lattice-based algorithm into its Chrome browser in developer mode. It is implemented for OpenSSL and designed to provide post-quantum security for TLS [31]. They also explored the performance of Apache HTTP server using post-quantum key exchange algorithms BCNS, NewHope, NTRU and Frodo (only NTRU and FrodoKEM remained as finalists in NIST's competition.) By looking at throughput, connection time and handshake size, they have concluded that the additional overhead in serving typical webpages (between 10KB and 100KB) with a post-quantum cipher suite will only decrease server throughput by less than a factor of two [32].

IBM researchers developed lattice cryptography suites which include the NIST finalists crystals-Kyber and Crystals-Dilithium, and is working with open-source community to develop open standards implementations as part of Open Quantum Safe. Currently, Kyber has been integrated as part of IBM Key Protect for IBM Cloud, a full-service encryption solution that leverages cloud-based hardware security modules. The algorithm performance may be affected by network profile, CPU speed and API call rates. The quantum-safe TLS is currently supported through the Key Protect software development kit and available both in hybrid mode and quantum-safe mode. It is currently only available in Linux, but support for additional operating systems is anticipated [33].

AWS started incorporating PQC since round 2 and now supports post-quantum TLS in AWS KMS. It supports ECDHE with BIKE and ECDHE with SIKE [8]. For the two hybrid algorithms tested, ECDHE with BIKE have a larger size than ECDHE with SIKE. However, ECDHE with SIKE is slower than ECDHE with BIKE. Which algorithm to use would depend on the constraints of the asset, and whether memory or computational speed is more of a priority.

Microsoft is working with academia and industry on four candidates for cryptography systems: Rooke, SIKE, Picnic and qTESLA. Each algorithm may be appropriate for different scenarios where different trade-offs regarding performance and key size are preferred. In addition to working with Open Quantum Safe to develop a post-quantum branch of TLS and SSH, Microsoft also worked on a fork of OpenVPN integrated with post-quantum cryptography to enable testing and experimentation [34].

5. Crypto Agility

According to NIST, "continued progress in the development of quantum computing foreshadows a particularly disruptive cryptographic transition." Once quantum computers and exploitation of such attacks becomes practical, protecting stored keys and data will require re-encrypting them with a quantum-resistant

algorithm and deleting or physically securing backups. The integrity and sources of information will become unreliable unless they are processed or encapsulated with quantum-resistant mechanisms. In the best case, 5-15 or more years will elapse after the publication of the standards before a full implementation of those standards is complete. Without proper planning, it may take decades to replace most of the vulnerable public-key systems currently in use. Thus, NIST encourages enterprises to identify where and for what it is employing public-key cryptography and all the use characteristics, as well as developing a playbook for crypto agility [13].

While there are many libraries and solutions available to help with the quantum transition, many information systems are not designed to encourage support of rapid adaptations of new cryptographic primitives and algorithms. Cryptographic algorithms cannot be replaced until all components of a system are prepared to process the replacement. This may require not only the replacement of cryptographic algorithms, but also updates to the protocols, hardware, dependent operating systems and procedures, especially in the case of post-quantum cryptography where the parameter and structures can be very different. In addition to replacing algorithms, other non-security issues, such as adoption rates, backward compatibility and performance must also be considered. The Crypto Agility Risk Assessment Framework (CARAF) can be used to combine all the factors into a broader crypto agility strategy that allows for a smoother transition within a time period commensurate with an organization's risk tolerance [36]. The risk framework consists of five phases:

1. **Identify threat** – the threat in this case is security risk attributable to quantum computing, and more specifically the challenges and risk of migrating to PQC. The assets that will be phased out before quantum computers become practical, or NIST publishes a definitive standard for PQC, can be eliminated from the risk assessment. Meanwhile, NIST has published transition guidance on the recommended algorithms and key lengths [35].
2. **Inventory of assets** - will give an overview of how crypto-agile assets are and help in identifying which assets should be prioritized in any migration. In this case, we need to identify where, how and for what public-key cryptography is employed, as well as use characteristics. NIST has published a draft on how to identify and what information should be recorded for post-quantum migration based on 5 scenarios: FIPS-140 validated hardware and software modules, cryptographic libraries, cryptographic applications, embedded code in computing platforms and communication protocols [47].
3. **Estimate risk** – given the lack of information on attack vectors, the risk estimation for crypto agility is based on a combination of timeline, to realize the threat, and cost to mitigate.
4. **Secure asset** - based on an evaluation of the risk and the resources available, the organization may prioritize the assets and decide to accept the risk, mitigate it, or phase out the asset. While it is important to assess how crypto-agile your assets are when assessing the risk, it's also important to keep crypto agility in mind when choosing a new solution to implement, especially since PQC has not been tested against quantum computers and may still undergo changes in the future. Some of the properties to keep in mind for the new solutions are extensibility, removability, interoperability, compatibility, flexibility, and updatability [37].
5. **Create roadmap** - the solutions will likely go through a few iterations and will affect different assets differently. In addition to performing benchmark testing to evaluate the performance and impact on your assets, it's important to make sure that all teams and vendors are on the same page and take future changes into account in policies and guidelines. What the roadmap should consists of depends on the mitigation methods and the organization. Security standards, best practice documentations, installation and administration documentations may all need to be changed or replaced.

The transition to PQC and making assets crypto-agile requires cooperation and collaboration between different teams and vendors. It can be overwhelming at first, but the more elements of crypto agility are implemented, the easier the next step will be. By preparing now for the upcoming transition, we can ensure a more orderly, less costly, and minimally disruptive changeover [38]. Because many of the PQC solutions have not been rigorously tested yet, they may have systemic weakness and go through several iterations before they become secure. Crypto agility will provide a practical framework to address updates to crypto threats in a quick and efficient manner.

6. Conclusion

Transitioning to post-quantum algorithms is a big undertaking. Different algorithms have different key lengths, performance, and operational constraints. There is no one size fits all solution. Benchmarking of the algorithm and crypto agility assessment of the target assets will help determine what algorithm is appropriate, as well as the potential overhead. Even if your asset implements post-quantum cryptography, there will be backward compatibility problems if others don't. Thus, it is important to plan and create a roadmap for your PQC transition.

The first step in transitioning to a post-quantum world is assessing which assets are capable of post-quantum transition using a crypto agility risk assessment. The more elements of crypto agility your assets implement, the easier the transition will be. Based on the risk assessment, action plan options range from phasing out the asset before quantum computing becomes available, accepting the risk, or securing it.

The true security of post-quantum cryptography won't be testable or tested until a practical quantum computer becomes available. For that reason, and in the meantime, a more proactive approach is to focus on the implementation and performance of hybrid cryptography.

Picking the appropriate quantum algorithms is a decision that is tied to security requirements and asset/system constraints. As with all new technological environments, testing how post-quantum algorithms work with your assets will be informative; different implementations may provide distinct benefits. Also important: Transitioning to support a quantum-based crypto environment, and the identifying which phases happen when, based on existing or desired risk tolerance levels.

Abbreviations

AES	Advanced Encryption Standard
API	Application Program Interface
ATRC	Abu Dhabi's Advanced Technology Research Council
AWS	Amazon Web Service
CACR	Chinese Association for Cryptologic Research
CARAF	Crypto Agility Risk Assessment Framework
CPU	Central Processing Unit
DSA	Digital Signature Algorithm
ECDH	Elliptic-Curve Diffie–Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
ENISA	European Union Agency for Cybersecurity
ETSI	European Technology and Standards Institute

HTTP	Hypertext Transfer Protocol
IBM	International Business Machines
IKE	Internet Key Exchange
IP	Internet Protocol
KMS	Key Management Service
NIST	National Institute of Standards and Technology
OQS	Open Quantum Safe
PKI	Public Key Infrastructure
PQC	Post-Quantum Cryptography
QKD	Quantum Key Distribution
QSC	Cyber Quantum Safe Cryptography
RSA	Rivest–Shamir–Adleman
SSH	Secure Shell
S/MIME	Secure/Multipurpose Internet Mail Extension
TCP	Transmission Control Protocol
TII	Technology Innovation Institute
TLS	Transport Layer Security
VPN	Virtual Private Network
XMSS	eXtended Merkle Signature Scheme
ZKP	Zero Knowledge Proof

Bibliography & References

[1] Post Quantum Cryptography Team. A Quantum World and How NIST is Preparing for Future Crypto, 2014.
[2] Peter Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM J. Comput., 26(5):1484– 1509, October 1997.
[3] Tyson Macaulay. Comments in Response to NIST Cyber Security Framework Draft 1.1 and NIST Roadmap for Improving Critical Infrastructure Cybersecurity Draft 1.1. Technical report, National Institute of Standards & Technology, 2018.
[4] Ward Beullens, Jan-Pieter D'Anvers, Andreas Hulsing, Tanja Lange, Lorenz Panny, Cyprien de Saint Guilhem, and Nigel P. Smart. Post-Quantum Cryptography: Current State and Quantum Mitigation. Technical report, ENISA.
[5] Chris Hickman. CSO Insights: 4 Reasons Why You Can't Ignore Crypto Agility, 2019.
[6] PQC Standardization Process: Third Round Candidate Announcement. Technical report, National Institute of Standards & Technology, 2020.
[7] "liboqs," Open Quantum Safe. 2021. https://openquantumsafe.org/liboqs/.
[8] Andrew Hopkins. Post-Quantum TLS Now Supported in AWS KMS. https://aws.amazon.com/blogs/security/post-quantum-tls-now-supported-in-aws-kms/.
[9] Classic McEliece: Conservative Code-Based Cryptography. Technical report, 2020. https://classic.mceliece.org/nist/mceliece-20201010.pdf.
[10] "Kyber," CRYSTALS: Cryptographic Suite for Algebraic Lattices. 2020. https://pq-crystals.org/kyber/
[11] "Algorithms in liboqs," Open Quantum Safe. 2021. https://openquantumsafe.org/liboqs/algorithms/.

[12] GeMSS: A Great Multivariate Short Signature. 2020. https://www-polsys.lip6.fr/Links/NIST/GeMSS_specification.pdf
[13] William Barker, William Polk, and Murugiah Souppaya. Getting Ready for Post-Quantum Cryptography: Exploring Challenges Associated with Adopting and Using Post-Quantum Cryptographic Algorithms, 2021.
[14] Quantum Safe Cryptography and Security: An Introduction, Benefits, Enablers and Challenges. Technical report, ETSI.
[15] Daniel Kales and Greg Zaverucha. Improving the Performance of the Picnic Signature Scheme. 2020.
[16] Olaf Grote, Andreas Ahrens, and Cesar Benavente-Peces. Paradigm of Post-Quantum Cryptography and Crypto-agility: Strategy Approach of Quantum-safe Techniques. 2019.
[17] Farnoud Farahmand, Viet Ba Dang, Michal Andrzejczak, and Kris Gaj. Implementing and Benchmarking Seven Round 2 Lattice-Based Key Encapsulation Mechanisms Using a Software/Hardware Codesign Approach. In NIST Second PQC Standardization Conference. National Institute of Standards & Technology, 2019.
[18] Kris Gaj. Challenges and Rewards of Implementing and Benchmarking Post-Quantum Cryptography in Hardware. In Proceedings of the 2018 on Great Lakes Symposium on VLSI, pages 359–364, 2018.
[19] Christian Paquin, Douglas Stebila, and Goutam Tamvada. Benchmarking Post-Quantum Cryptography in TLS. 2019.
[20] Kris Gaj. Implementation and Benchmarking of Round 2 Candidates in the NIST Post-Quantum Cryptography Standardization Process Using FPGAs. 2020.
[21] The Latest Progress of PQC Competition in China. Technical report, 7th ETSI QSC/IQC Workshop, 2019.
[22] Announcement of the Results of the National Cryptographic Algorithm Design Competition Algorithm Selection. Technical report, CACR, 2020.
[23] PQCrypto Usage and Deployment. IANIX. 2021. https://ianix.com/pqcrypto/pqcrypto-deployment.html
[24] Kris Kwiatkowski and Armando Faz-Hernandez. Introducing CIRCL: An Advanced Cryptographic Library, 2019.
[25] ISARA Radiate. https://www.isara.com/products/isara-radiate.htm
[26] Post-Quantum Cryptography Hardware, Firmware, SDK, Toolkits - PQShield. https://pqshield.com/
[27] Nitin Dahad. UAE Building First Quantum Computing and Cryptography Library. EE Times Asia, 2021.
[28] Panos Kampanakis, Peter Panburana, Ellie Daw, and Daniel Van Geest. The Viability of Post-Quantum X.509 Certificates. Technical report, 2018.
[29] Massimiliano Pala. Docsis pki: A Proposal for a Next-Generation Quantum-Resistant Infrastructure. SCTE ISBE, 2020.
[30] Confidence in Post Quantum Algorithms. Cloud Security Alliance. 2021. https://cloudsecurityalliance.org/artifacts/confidence-in-post-quantum-algorithms/
[31] Jeremy Kirk. Google Tests Post-Quantum Crypto, 2016. https://www.bankinfosecurity.com/google-adds-quantum-computing-armor-to-chrome-a-9253
[32] Joppe Bos, Craig Costello, Leo Ducas, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Ananth Raghunathan, and Douglas Stebila. Frodo: Take Off the Ring! Practical, Quantum-Secure Key Exchange From LWE. ACM Conference on Computer and Communications Security, 2016.
[33] Introduction to Quantum-Safe Cryptography in TLS for IBM Key Protect, 2021. https://www.ibm.com/cloud/blog/introducing-quantum-safe-crypto-tls-for-ibm-key-protect
[34] Post-Quantum Cryptography. Microsoft. 2021. https://www.microsoft.com/en-us/research/project/post-quantum-cryptography/
[35] Elaine Barker and Allen Roginsky. Transitioning the Use of Cryptographic Algorithms and Key Lengths. Technical report, National Institute of Standards & Technology, 2019.

[36] Chujiao Ma, Luis Colon, Joe Dera, Bahman Rashidi, and Vaibhav Garg. CARAF: Crypto Agility Risk Assessment Framework. Journal of Cybersecurity, 7(1), 05 2021. tyab013.
[37] Hassane Aissaoui Mehrez and Othmane EL OMRI. The Crypto Agility Properties. The 12th International Multi-Conference on Society, Cybernetics and Informatics, 2018.
[38] Matt Campagna, Brian LaMacchia, and David Ott. Post Quantum Cryptography: Readiness Challenges and the Approaching Storm. Computing Community Consortium, 2020
[39] "External Users of OQS," Open Quantum Safe. 2021. https://openquantumsafe.org/applications/external.html.
[40] Dustin Moody, "NIST PQC Standardization," 2021. https://www.nccoe.nist.gov/sites/default/files/3-PQC%20NCCoE.pdf .
[41] Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner and Daniel Smith-Tone. Report on Post-Quantum Cryptography. Technical report, NIST, 2016.
[42] Elaine Baker, Lily Chen and Richard Davis "Recommendation for Key-Derivation Methods in Key-Establishment Schemes," NIST, 2020.
[43] Andreas Hülsing, Stefan-Lukas Gazdag, Denis Butin and Johannes Buchmann. " Hash-based Signatures: An Outline for a New Standard," NIST, 2015.
[44] "OQS Algorithm Performance Visualizations," Open Quantum Safe. 2021. https://openquantumsafe.org/benchmarking/.
[45] "eBACS: ECRYPT Benchmarking of Cryptographic Systems." https://bench.cr.yp.to/supercop.html.
[46] Alexander Truskovsky, Daniel Van Geest, Scott Fluhrer, Panos Kampanakis, Mike Ounsworth and Serge Mister, "Multiple Public-Key Algorithm X.509 Certificates," Internet Engineering Task Force, 2018. https://datatracker.ietf.org/doc/html/draft-truskovsky-lamps-pq-hybrid-x509-01.
[47] William Barker and Murugiah Souppaya, "Migration to Post-Quantum Cryptography," NIST, 2021. https://www.nccoe.nist.gov/sites/default/files/library/project-descriptions/pqc-migration-project-description-draft.pdf.
[48] Cong Peng, Jianhua Chen, Sherali Zeadally and Debiao He, "Isogeny-Based Cryptography: A Promising Post-Quantum Technique" in IT Professional, vol. 21, no. 06, pp. 27-32, 2019.

