Post-Quantum Migration

Table of contents

1. Introduction

Currently widely used cryptographic hash functions such as for example SHA-2 and SHA-3 (Box 1.1), Message Authentication Codes (MACs) such as for example HMAC and Poly1305 (Box 1.2), and symmetric encryption algorithms such as for example the AES block cipher and the ChaCha20 stream cipher (Box 1.3), are deemed to be resistant to future attacks by means of powerful quantum computers, provided that sufficiently large (underlying) hash values, authentication codes and cryptographic keys are being used.

A cryptographic hash function is a mathematical algorithm that maps data of an arbitrary size to a bitstring of a fixed size (the “hash” or "hash value"), by means of a one-way function. Ideally it should have the following properties:

- it is fast to compute the hash value for any given piece of data;
- the computed hash value is always the same for given piece of data, i.e. the hash function is deterministic;
- it is (practically) infeasible to generate a piece of data that yields a given hash value, i.e. it is impossible to reverse the process that generated the given hash value (pre-image resistance);
- for any given piece of data, it is (practically) infeasible to find another piece of data that has the same hash value (second pre-image resistance);
- it is (practically) infeasible to find (at least) two different pieces of data that have the same hash value (collision resistance);
- a small change to a piece of data should change its hash value so extensively that the new hash value appears uncorrelated with the old hash value (avalanche effect).

Box 1.1: Cryptographic hash function

A Message Authentication Code (MAC) is used to verify the authenticity (and at the same time, to protect the integrity) of a piece of data (a file, a document, a message, etc.). A MAC provides message authentication provided that there exists mutual trust, but will not resist repudiation (because the mutual trust relationship breaks with repudiation).

Box 1.2: Message Authentication Code (MAC)

Symmetric cryptography uses only one cryptographic key (known as the ”secret key”) for both the encryption of plaintext and the decryption of the corresponding ciphertext. For some algorithms, the key value used for decryption is derived from the key value used for encryption by a simple transformation.

Symmetric encryption algorithms are categorised into block and stream ciphers:

- Block ciphers convert data in plaintext into ciphertext in fixed-size blocks. The block size generally depends on the encryption scheme. If the plaintext length is not a multiple of the block size the encryption scheme uses padding to ensure complete blocks are encrypted.

- Stream ciphers encrypt a continuous string of binary digits by applying time-varying transformations on plaintext data. Therefore, this type of encryption works bit-by-bit, using keystreams to generate ciphertext for arbitrary lengths of plaintext messages. Stream encryption ciphers achieve this using feedback shift registers to generate a unique nonce (number used only once) to create the keystream.

Box 1.3: Symmetric cryptography

It is generally believed that Cryptographically Relevant Quantum Computers (CRQCs, see Box 1.4) will become available in the (not so far) future, which are capable of breaking many widely used classical state-of-the-art public-key cryptographic schemes (Box 1.5), including secret key exchange mechanisms (Box 1.6) and digital signature mechanisms (Box 1.7).

The term Cryptographically Relevant Quantum Computer (CRQC) is used to specifically describe powerful future quantum computers that are capable of actually attacking real world cryptographic schemes that would be infeasible to attack with a classical computer.

Box 1.4: Cryptographically Relevant Quantum Computer (CRQC)

Public-key cryptography uses pairs of cryptographic keys. Each pair consists of a public key (which may be known to others) and a private key (which must not be known by anyone except the owner). The generation of such key pairs depends on asymmetric cryptographic algorithms, which are based on hard mathematical problems (one-way functions).

Box 1.5: Public-key cryptography

A key exchange (aka key establishment) mechanism is a method by which symmetric cryptographic keys are exchanged between two or parties. Key transport (aka key distribution) is the process whereby one entity generates a secret key and then transfers that secret key by secure means to the other entity. Key agreement is the process of establishing a shared secret key between two entities in such a way that neither of them can predetermine the value of the shared secret key. Key transport usually involves non-interactive techniques while key agreement usually involves interactive techniques. Key transport protocols and key agreement protocols can be based on either symmetric or asymmetric cryptographic techniques.

In many cases, the shared secret key that is established by a key transport or key agreement mechanism is not directly used, but is subject to further processing in order to derive the cryptographic key(s) that is (are) used for subsequent encryption and/or decryption.

Box 1.6: Key exchange mechanism

A digital signature is the electronic analogue of a hand-written signature and must satisfy the following requirements:

- the receiver should be able to validate the sender's signature;
- the signature must not be forgeable;
- the sender must not be able to successfully repudiate the signing of a message.

A major difference between digital (electronic) signatures and hand-written signatures is that a digital signature cannot be constant. Given its digital nature (a string of bits), a constant digital signature

could easily be attached (copied) to any piece of data. A usable digital signature therefore needs to be a function of the entire piece of data (message, document, file, etc.) that is signed by it. Furthermore, digital signatures often include sequence numbers, timestamps, etc., to ensure different digital signatures for otherwise equal pieces of data.

Most digital signatures are based on public-key cryptography schemes, many of which are based on specialised algorithms that are not suitable for encipherment purposes. It is usually not desirable to apply a digital signature directly to a possibly long piece of data, given the inefficiency of public-key encryption. Nonetheless, the entire piece of data should be protected by the signature. A way of satisfying both requirements is to use a cryptographic hash function as an intermediary. The hash function takes the entire piece of data and produces a fixed-length message digest (hash value), which is then digitally signed.

Box 1.7: Digital signature mechanism

Significant damage could be caused by future CRQCs by breaking secret key exchange mechanisms based on current public-key encryption. Current publicly known quantum computers are certainly not capable of doing so. Nonetheless, by intercepting and recording data encrypted with secret keys established by means of key exchange mechanisms based on current public-key encryption, CRQCs could be used in the future to decrypt encrypted data that has been recorded earlier on ("store-now-decrypt-later attack" aka "harvest attack"). Significant damage could thus be caused retrospectively if no action is taken to mitigate this risk, e.g. by migrating to quantum-resistant key exchange schemes before the CRQC threat becomes reality (taking into account the amount of time during which the confidentiality of the previously encrypted data must be ensured).

Significant damage could also be caused by future CRQCs by forging digital signatures based on classical public-key cryptography with CRQCs. Digital signatures are used for different purposes, including signing of public-key certificates, which in turn is used for a variety of purposes: identity authentication, privilege authorisation, etc. Migrating to quantum-resistant digital signature schemes before the CRQC threat becomes reality is needed for mitigating this risk.

Classical public-key cryptography mechanisms are also used in several contemporary cryptographic security protocols (Box 1.8) for the purpose of "on-the-fly" entity authentication or privilege authorisation. Unlike "store-now-decrypt-later" attacks on key exchange mechanisms and attacks on digital signature mechanisms, attacking these entity authentication and privilege authorisation mechanisms will require a much more powerful quantum computer since the time available for performing the attacks is severely constrained, because the entity authentication and the privilege authorisation is done in real-time.

A cryptographic security protocol is an abstract or concrete protocol that performs security-related functions by applying cryptographic methods, often by means of a sequences of cryptographic primitives. It describes how the cryptographic algorithms should be used and includes details about data structures and representations.

Box 1.8: Cryptographic security protocol

2. Migration scenarios

First of all, any cryptographic hash function (such as for example MD5), MAC scheme (such as for example CBC-MAC) or symmetric encryption algorithm (such as for example TDES) that is not quantum-resistant should either be upgraded to use a larger (underlying) hash value, authentication code or cryptographic key size (if at all possible) or else replaced by a quantum-resistant mechanism.

One approach for mitigating the risks posed by the future availability of CRQCs to vulnerable public-key cryptographic schemes (including key exchange and digital signature mechanisms) is to physically isolate and strongly protect critical data assets, to prevent eavesdropping and unauthorised manipulation. A major issue with this approach is that physically isolating data usually makes it far less valuable.

There are several other approaches for mitigating these risks that mainly large organisations can choose from (in contrast, small organisations and consumers often have little choice other than relying on their technology providers to migrate to quantum-resistant cryptography):

- Using secure physical distribution of secret keys, e.g. by means of physical transport using cryptographic hardware tokens. This approach is very cumbersome, extremely slow and precludes many use cases.

- Using Quantum Key Distribution (QKD) or QKD Network (QKDN) solutions for the establishment of shared secret keys. This approach is (very) expensive and, in the case of QKD, is currently mostly limited to point-to-point secret key establishment using fibre-optic cables (over relatively short geographical distances) or free-space (satellite) communication channels.

  Furthermore, in general, QKD(N) is not considered a direct solution to the quantum computing cryptanalysis threat (though it could be part of such a solution) because QKD(N) security is inherently tied to the physical layer and QKDN cannot be used to protect information sent through its network nodes hence these nodes have to be trusted ("trusted relays"). Consequently, QKD(N) is not aligned with modern information security principles, such as end-to-end encryption and zero-trust.

  It is also important to note that QKD(N) updates will typically require changing hardware and/or firmware of QKD(N) equipment, whereas Quantum-Resistant Cryptography (QRC) upgrades will typically be delivered as software updates.

  QKD(N) solutions will mostly only be used for specific use cases such as for example encryption of communication links between data centres.

- Using a Key Derivation Function (KDF, see Box 2.1), to mix keying material obtained from different sources, such as Pre-Shared Keys (PSKs, see Box 2.2), QKD(N) keys, classic key exchange schemes and QRC key exchange schemes, into shared secret keys.

> A Key Derivation Function (KDF) is used in cryptography to derive multiple secrets (KDF outputs) from one or more other secrets (KDF inputs). A KDF is often used in security protocols that require participants to rederive the same key several times and is therefore expected to be deterministic. A KDF is usually not designed to produce a lot of derived secrets.

Box 2.1: Key Derivation Function (KDF)

> A Pre-Shared Key (PSK) is a secret key which was previously shared between two parties using a secure (typically out-of-band) communication channel, before it is put into use by some cryptographic mechanism.

Box 2.2: Pre-Shared Key (PSK)

When using PSKs, this approach necessitates keeping pairwise shared cryptographic key material, which is very cumbersome to implement and is therefore only an option for use cases where a limited set of entities is involved. This is also the case when using QKD(N) keys.

- Replacing key exchange schemes based on vulnerable public-key cryptography with PSK-based schemes (note that the PSK could be a QKD key). Many cryptographic security protocols, including IPsec, TLS and SSH, support the use of PSK-based key exchange schemes. This approach has the same disadvantage as the KDF approach.

- Using key exchange or digital signature QRC schemes that have already been standardised by NIST.

- Using other QRC schemes for which vendor-supported or open-source products are available. A major issue with this approach is that, in many cases, the security of such QRC schemes and/or products has not been independently verified.

- Using a combination of a classical public-key scheme and one or more QRC schemes (this may involve using hybrid public-key certificates that contain multiple sets of public keys and signatures). Security is guaranteed as long as the classical scheme remains secure against classical attacks and at least one of the QRC schemes remains secure against CRQC attacks.

  This can be done in such a way that backwards compatibility is achieved for entities that do not yet support the QRC scheme(s). However, care should be taken to prevent downgrade attacks (i.e. attacks where an attacker makes a party supporting QRC believe that the other party does not support it). Such combined classical/QRC schemes have already been

described for both key exchange schemes and signature schemes. For example, there is an IETF standard for quantum-safe VPNs and an IETF draft standard for hybrid key exchange in TLS1.3.

- Using a combination of two or more QRC schemes. Security is guaranteed as long as at least one of the QRC schemes remains secure against classical attacks and at least one of the QRC schemes (the same one or another one) remains secure against CRQC attacks. Such combined QRC schemes have already been described for both key exchange schemes and signature schemes.

- Using quantum-resistant hash-based signature schemes, such as eXtended Merkle Signature Scheme (XMSS), Multi-Tree XMSS (XMSSMT), Leighton-Micali Signature (LMS) or Hierarchical Signature System (HSS), which have already been standardised. However, the security of these particular signature schemes is dependent on careful state management (to ensure that signatures are only used once or a few times) and therefore, they are limited to specific use cases (an example is the use of XMSS for code signing).

- Waiting until standardised QRC signature schemes become available to replace classical signature schemes that are not quantum-resistant. The public keys used by these classical signature schemes must be revoked before the advent of CRQCs, so as to render all their signatures invalid before the CRQC threat becomes manifest. This may also require replacing existing classical signatures with quantum-resistant signatures.

It is of course possible, and will often be unavoidable, to follow an hybrid mitigation scenario by concurrently implementing multiple mitigation approaches described above as the appropriate risk mitigation technique will depend on several factors. For example: if highly sensitive data is transferred that is protected by means of vulnerable classical cryptography, and it is not feasible or not practical to change the sending and/or receiving application, an option may be to set up a quantum-safe VPN through which the application traffic is routed. An other example: the combination of an existing classical cryptographic scheme and a new QRC scheme may be required for maintaining the security certification obtained for a specific application, if certification mandates the use of the classical cryptographic scheme.

3. Migration considerations

A particular quantum-resistant public-key cryptographic scheme will most likely support only a limited set of use cases and it is therefore expected that NIST will standardise different Post-Quantum Cryptography (PQC) schemes for different types of applications and usage contexts.

Furthermore, existing cryptographic security protocols need to be modified to accommodate the particular characteristics of quantum-resistant public-key cryptographic schemes, e.g. long cryptographic keys, long ciphertexts or long digital signatures. So called “drop-in replacements” are not likely to be feasible in many cases for adapting these security protocols and (partial) protocol redesign will be required.

In some cases, replacing cryptography schemes might even be impossible. Examples: legacy systems that are no longer supported by their vendors, hardwired systems that cannot be changed/updated and systems with restricted accessibility (e.g. satellite systems).

It should be noted that QRC schemes may require higher-quality entropy (Box 3.1) than classical public-key cryptographic schemes. Furthermore, the security proofs for these QRC schemes require that their entropy be obtained by sampling gaussian distributions rather than uniformly distributed entropy, which is what is often produced by current random number generators. A possible solution is to use Quantum Random Number Generator (QRNG) solutions.

Entropy is a scientific concept as well as a measurable physical property that is most commonly associated with a state of disorder, randomness or uncertainty.

Box 3.1: Entropy

QKD solutions also depend on high-quality random numbers for their security, but generally the required entropy sources will be embedded in the QKD solution.

Migrating to new cryptographic schemes typically requires changing or replacing the following components: cryptographic libraries, implementation validation and certification tools, hardware that implements cryptographic algorithms or accelerates cryptographic algorithm performance, cryptography supporting operating system and application code, communications equipment, etc. Furthermore, security procedures need to be adapted and also, installation, configuration and system administration documentation needs to be changed or replaced.

It is important for organisations to ensure that all use cases of cryptographic schemes currently deployed to protect (critical) data assets are documented, together with the cryptographic parameters being used (algorithm domain parameters, cryptographic key lengths, etc.). Any dependencies between these cryptographic schemes must also be documented.

Cryptographic schemes that are deemed vulnerable to CRQCs need to be identified, and availability of potential solutions need to be investigated and documented. For each potential

solution, it must be determined how it will affect the ICT infrastructure and the applications, to identify potential future infrastructure shortcomings and, if needed, to develop plans for addressing them.

Based on the information gathered, migration scenarios can be worked out and their priorities determined, preferably using a risk-based approach. Also, it is of vital importance that post-quantum cryptography considerations are discussed with (potential) vendors, service providers, contractors, business partners and other relevant third parties; these vendors and service providers should have a post-quantum roadmap in place.

It goes without saying that a lot of effort is required to accomplish all of the above. However, most of it is in fact always required when cryptography solutions are deployed, to ensure that there are adequate plans for smooth migration to new cryptographic schemes, whenever currently used schemes are compromised or run the risk of being compromised in the near future.

Today, very few organisations have such plans readily available; most of them have only recently become aware of this issue due to the enormous amount of publicity given to the emerging quantum computer threats to existing cryptography in (social) media and professional journals.

Furthermore, few organisations have a centralised policy in place for the use of cryptography because it has become very easy to implement and use cryptographic solutions. Consequently, these organisations are neither aware of the types of encryption used by their IT infrastructure and applications, nor where such cryptography is being is used.

Therefore, organisations should immediately implement so-called “low-regret moves”:

- create awareness about the extent of information security that is provided by cryptography (aka crypto visibility);
- create awareness for emerging quantum computing threats to cryptography;
- monitor progress of quantum computing, quantum security and quantum-resistant cryptography technologies;
- develop a strategy for adopting and integrating new cryptographic schemes (aka crypto agility).

In general, migration to quantum-resistant public-key cryptographic schemes will take a significant amount of time. For example, NIST cautions that, after publication of the first set of four PQC standards (expected in 2024), five to fifteen more years will be needed for completing migration to PQC cryptography.

In some cases, migration to new cryptographic schemes will take a very long time to implement because there are many parties involved. Examples: Public Key Infrastructure (PKI) infrastructures operated by Trust Service Providers (TSPs), Distributed Ledger Technology (DLT) infrastructures and electronic payment infrastructures.

Migration to quantum-resistant cryptography as described above should not be considered to constitute a long-term cryptographic solution for an organisation, for several reasons:

- There is no guarantee that the proposed QRC schemes will remain secure for a reasonable amount of time. Firstly, these schemes have been subjected to far less classical cryptanalysis than the widely used pre-quantum public-key cryptographic schemes they are meant to replace. Secondly, in many cases, we don't know whether there will soon be a (yet unknown) quantum algorithm for breaking a particular QRC scheme.

- The focus of QRC security is currently on providing resistance against Shor's and Grover's quantum algorithms. However, many a time a new quantum algorithm is discovered that is capable of breaking a (specific type of) QRC scheme (e.g. Abelian hidden shift algorithm, BDD algorithm, BHT algorithm, claw finding attack algorithm, dHSP algorithm, EDCP algorithm, HHL algorithm, Kuperberg's algorithm, Tami's algorithm, etc.). Furthermore, lattice-based cryptography QRC schemes can be (and have already been) attacked by means of Quantum Annealers (QAs, see Box 3.2), and NISQ quantum computers (Box 3.3), because the Closest Vector Problem (CVP) hard problem is merely an optimisation problem.

Quantum annealing is a metaheuristic for finding the global minimum of a given objective function over a given set of candidate solutions, by a process using quantum fluctuations. A quantum fluctuation is the temporary random change in the amount of energy in a point in space as prescribed by Heisenberg's uncertainty principle. Quantum fluctuations are minute random fluctuations in the values of the fields which represent elementary particles, such as electric and magnetic fields. Although the particles are not directly detectable, the cumulative effects of these particles are measurable.

Heisenberg's uncertainty principle is asserting a fundamental limit to the accuracy with which the values for certain pairs of physical quantities of a particle, such as position and momentum, can be predicted from initial conditions.

Box 3.2: Quantum annealing

Noisy Intermediate-Scale Quantum (NISQ) applies to current state-of-the-art quantum computers. The term 'noisy' refers to the fact that these quantum computers are very sensitive to the environment and may lose their quantum state due to quantum decoherence because they are not sophisticated enough to implement quantum error correction. Quantum decoherence is the loss of quantum coherence and represents a challenge for the practical realisation of quantum computers, since such machines are expected to rely heavily on the undisturbed evolution of quantum coherences. The term 'intermediate-scale' refers to the not-so-large number of qubits.

Box 3.3: Noisy Intermediate-Scale Quantum (NISQ)

- Even if the quantum algorithm(s) for breaking a particular QRC scheme would be known, it is very difficult to determine the computing cost and the memory cost of (a) quantum computer(s) capable of breaking the scheme, given the current state-of-the-art of quantum computing technology. It is therefore in many cases still unclear how to choose the QRC scheme parameter settings that are needed for withstanding quantum computer attacks.

These parameter settings often have a profound effect on the key size, the ciphertext size or signature size, and the encryption/decryption time or signature generation/verification time.

- Last but not least: cheaper, improved or entirely new quantum security technologies might (and probably will) emerge and could possibly be used as viable replacements for QRC-based solutions or parts thereof.

Appendix A - References

NCCoE Migration to Post-Quantum Cryptography

[ANSSI 2022] ANSSI views on the Post-Quantum Cryptography transition

[CSA 2021] Practical Preparations for the Post-Quantum World

[ENISA 2021] Post-Quantum Cryptography – Current state and quantum mitigation

[ENISA 2021] Post-Quantum Cryptography – Integration study

[ETSI 2020] TR 103 619 - Migration strategies and recommendations to Quantum Safe schemes

[IETF 2005] RFC 4279 - Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)

[IETF 2009] RFC 5487 - Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode

[IETF 2020] RFC 8784 - Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security

[IETF 2024] draft-ietf-tls-hybrid-design-09 - Hybrid key exchange in TLS 1.3

[IETF 2023] draft-kampanakis-curdle-ssh-pq-ke-01 - Post-quantum Hybrid Key Exchange in SSH

[Gartner 2022] Preparing for the Quantum World With Crypto-Agility

[NCSC 2022] Guidelines for quantum-safe transport-layer encryption

[NIST 2021] Getting Ready for Post-Quantum Cryptography

[NOREA 2024] Quantum Computing and Cryptography

[NOREA 2024] Quantum Key Distribution

[NOREA 2024] Quantum Networks

[NOREA 2024] Quantum Random Number Generation

[TNO 2020] Migration to quantum-safe cryptography

[TNO/CWI 2019] Towards Quantum-Safe VPNs and Internet

[TNO/CWI/AIVD 2023] The PQC Migration Handbook

[WEF 2022] Transitioning to a Quantum Secure Economy

Appendix B - Acronyms and abbreviations

AES	Advanced Encryption Standard
AIVD	Algemene Inlichtingen- en Veiligheidsdienst
aka	also known as
ANSSI	Agence nationale de la sécurité des systèmes d'information
BDD	Bounded-Distance-Decoding
BHT	Brassard, Høyer and Tapp
CBC	Cipher Block Chaining
CRQC	Cryptographically Relevant Quantum Computer
CSA	Cloud Security Alliance
CVP	Closest Vector Problem
CWI	Centrum Wiskunde & Informatica
DES	Data Encryption Standard
dHSP	dehedral Hidden Subgroup Problem
DLT	Distributed Ledger Technology
e.g.	exempli gratia
EDCP	Extrapolated Dihedral Coset Problem
EDP	Electronic Data Processing
ENISA	*European Union Agency for Cybersecurity* (former European Network and Information Security Agency)
etc.	et cetera
ETSI	European Telecommunications Standards Institute
HHL	Harrow, Hassidim and Lloyd
HMAC	Hash-based Message Authentication Code
HRSS	Hülsing - Rijneveld - Schanck - Schwabe
HSS	Hierarchical Signature System
i.e.	id est
ICT	Information and Communication Technology

IETF	Internet Engineering Task Force
IKEv2	Internet Key Exchange version 2
IP	Internet Protocol
IPsec	IP security
KDF	Key Derivation Function
KEM	Key Encapsulation Method
LMS	Leighton-Micali Scheme
MAC	Message Authentication Code
MD5	Message Digest 5
NCCoE	National Cybersecurity Center of Excellence
NCSC	Nationaal Cyber Security Centrum
NISQ	Noisy Intermediate-Scale Quantum
NIST	National Institute of Standards and Technology
nonce	number used only once
NOREA	Nederlandse Orde van Register EDP-Auditors
NTRU	N-th *D*egree Truncated *P*olynomial Ring Units
PKI	Public Key Infrastructure
PQC	Post-Quantum Cryptography
PSK	Pre-Shared Key
QA	Quantum Annealer
	Quantum Annealing
QED-C	Quantum Economic Development Consortium
QKD	Quantum Key Distribution
QKDN	Quantum Key Distribution Network
QRC	Quantum-Resistant Cryptography
QRNG	Quantum Random Number Generator
RFC	Request *f*or Comments

SHA	Secure Hash Algorithm
SHA-2	Secure Hash Algorithm 2
SHA-3	Secure Hash Algorithm 3
SIKE	Supersingular Isogeny Key Encapsulation
SSH	Secure SHell
TDES	Triple DES
TLS	Transport Layer Security
TLS1.3	TLS version 1.3
TNO	*N*ederlandse *O*rganisatie voor Toegepast Natuurwetenschappelijk Onderzoek
TR	Technical Report
TSP	Trust Service Provider
US	United States
VPN	Virtual Private Network
WEF	World Economic Forum
XMSS	eXtended Merkle Signature Scheme
XMSSMT	Multi-*t*ree eXtended Merkle Signature Scheme