IG.18 Opportunities and Challenges for Hybrid (QKD and PQC) Scenarios

Version 1.0

20 October 2024

This is a Whitepaper of the GSMA

Security Classification: Non-Confidential

Copyright Notice

Disclaimer

Compliance Notice

The information contain herein is in full compliance with the GSM Association's antitrust compliance policy.

Table of Contents

1. Introduction

Quantum Key Distribution (QKD) is a security technology based on quantum physics to securely establish symmetric encryption keys. This technology in principle allows the agreement of cryptographic keys between two remote parties with information-theoretic security, guaranteed by the fundamental laws of physics. These keys can then be used securely with conventional cryptographic algorithms.

Post-quantum cryptography (PQC) refers to cryptographic algorithms which are resilient to attacks by quantum computers. In other words, unlike QKD, PQC relies on algorithms that are too complex for quantum computers to crack. PQC is still in active development, and it's currently undergoing standardization by NIST. Also, the model of Post Quantum Telco Network is under definition and development [GSMA4]

These two technologies, i.e., QKD and PQC are likely to be considered pillars complementing each other in hybrid security scenarios. The scope of this report is to analyse opportunities and challenges in Hybrid security scenarios based on the combined used of QKD and PQC.

It is important to underline that simultaneous use of PQC with QKD is permissable and in alignment with the the current state of the European Commission draft document: recommendation on a Coordinated Implementation Roadmap for the transition to Post-Quantum Cryptography [EC-1].

It should be mentioned that, in general, the term hybrid security is used, in cryptography, with several different meanings. A clear taxonomy of these meanings is required (and even standardised) for better analysing the options of the so-called QKD-PQC hybrid security scenarios.

1.1. Overview and scope

Scope of this report is to analyse opportunities and challenges in Hybrid security scenarios based on the combined use of QKD and PQC.

Before that, a taxonomy of the meanings of the term "hybrid" has to be provided in the context of security, in order to fix the analysis of this white paper. Having established this, we provide an analysis of the state of art of the international activities carried out by existing projects, industry bodies and standard fora.

Examples of analysis of hybrid security scenarios based on the combined used of QKD and PQC include the following questions:

- to understand challenges and opportunities of the combined used of QKD and PQC.
- to get a picture of the state of the art and the experiments.
- To provide a picture of current activities in the standards (e.g., ITU, ISO, IETF, ETSI, CEN-CENELEC…) with the aim at identifying gaps and proposing synergies to avoid overlapping efforts.

1.2. Intended audience

The intended audience for this document includes mainly stakeholders in the telecom industry, stakeholders in the supply chain, industry analysts, industry regulators, security policy makers, and security researchers.

1.3. Abbreviations

Term	Description
ASU2	Almost Strongly Universal2
CRQC	Cryptographic relevant Quantum Computer
CV-QKD	Continuous Variable Quantum Key Distribution
DSA	Digital Signature Algorithm
DV-QKD	Discrete Variable Quantum Key Distribution
FPGA	Field Programmable Gate Array
HMAC	Hash-based Message Authentication Code
IKE	Internet Key Exchange
ITS	Information Theoretically Secure
KDC	Key Distribution Center
KDF	Key Derivation Function
KEM	Key Encapsulation Mechanism
KMS	Key Management System
MAC	Message Authentication Codes
MITM	Man-in-theMmiddle
OTP	One Time Pad
PKIs	Public Key Infrastructures
PoCs	Proof-Of-Concepts
PoPs	Point of Presence
PQC	Post Quantum Cryptography
PPK	Post Quantum Pre-Shared Key
PRF	Pseudo Random Function
PSK	Pre-Shared Keys
PUF	Physical Unclonable Functions
QRNG	Quantum Random Number Generator
QKD	Quantum Key Distribution
SDN	Software Defined Network
TLS	Transport Layer Security
VPN	Virtual Private Network

1.4. References

Doc Number	Title
GSMAWP1	https://www.gsma.com/newsroom/resources/ig-11-quantum-computing-networking-and-security/attachment/ig-11-quantum-computing-networking-and-security-2/
GSMAWP2	https://www.gsma.com/newsroom/resources/quantum-networking-and-service/
GSMAWP3	https://www.gsma.com/aboutus/workinggroups/resources/ig-14-quantum-hardware-abstraction-layer-for-quantum-computing-and-networking
GSMAWP4	https://www.gsma.com/newsroom/resources/post-quantum-telco-network-impact-assessment-whitepaper/
HAC	A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone (1996) Handbook of Applied Cryptography ISBN 0-8493-8523-7.
EC-1	Recommendation on a Coordinated Implementation Roadmap for the transition to Post-Quantum Cryptography https://ec.europa.eu/newsroom/dae/redirection/document/104249 (11/04/2024)
ETSI-HYB	https://www.etsi.org/deliver/etsi_ts/103700_103799/103744/01.01.01_60/ts_103744v010101p.pdf
ITU-T-HYB-1	Recommendation ITU-T X.1714 - Key combination and confidential key supply for quantum key distribution networks
ITU-T-HYB-2	ITU-T Technical Report FG QIT4N D2.2 - Quantum information technology for networks use cases: Quantum key distribution network
IETF-HYB-2	RFC 8784 “Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security”
IETF-HYB-3	Draft IETF “Hybrid key exchange in TLS 1.3”
IETF-HYB-4	RFC 9370 "Multiple Key Exchanges in the Internet Key Exchange Protocol Version 2 (IKEv2)"
IETF-AAA	https://www.ietf.org/proceedings/50/I-D/1id-abstracts.txt
NIST-HYB-1	Recommendation NIST SP 800-133
NIST-HYB-2	Recommendation NIST SP 800-56C
QKD-BB84	C. H. Bennett and G. Brassard. “Quantum cryptography: Public key distribution and coin tossing”. In Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, volume 175, page 8. New York, 1984.
QKD-PQC-1	Max Brauer, Rafael J. Vicente, Jaime S. Buruaga, Ruben B. Mendez, Ralf-Peter Braun, Marc Geitz, Piotr Rydlichkowski, Hans H. Brunner, Fred Fung, Momtchil Peev, Antonio Pastor, Diego Lopez, Vicente Martin, Juan P. Brito - “Linking QKD testbeds across Europe” available at the link **https://www.mdpi.com/1099-4300/26/2/123 (31/01/2024)**
QKD-PQC-2	Lydia Garms, Taofiq K. Paraïso, Neil Hanley, Ayesha Khalid, Ciara Rafferty, James Grant, James Newman, Andrew J. Shields, Carlos Cid, Maire O'Neill - “Experimental Integration of Quantum Key Distribution and Post-Quantum Cryptography in a Hybrid Quantum-Safe Cryptosystem” available at the link https://onlinelibrary.wiley.com/doi/full/10.1002/qute.202300304 (19/02/2024)

Doc Number	Title
QKD-PQC-3	Quantum-resistant Transport Layer Security Rubio Garcia, C., Rommel, S., Takarabt, S., Vegas Olmos, J. J., Guilley, S., Nguyen, P., & Tafur Monroy, I. Quantum-resistant Transport Layer Security. https://www.sciencedirect.com/science/article/pii/S0140366423004012?via%3Dihub (22/11/2023)
WIRE-TAP	Wyner, A. D. (October 1975). "The Wire-Tap Channel". Bell System Technical Journal. 54 (8): 1355–1387. doi:10.1002/j.1538-7305.1975.tb02040.x -

2. Terms and Definitions

2.1. What's QKD?

Quantum Key Distribution (QKD) is a security technology based on quantum physics to securely establish shared-secret symmetric encryption keys. This technology allows the exchange of cryptographic keys between two remote parties based on a protocol, which is information-theoretically secure, guaranteed by the validity of Quantum Mechanics. These keys can then be used securely with conventional cryptographic algorithms [GSMAWP1], [GSMAWP2] [GSMAWP2].

2.2. What's PQC?

The goal of PQC is to develop cryptographic systems that are secure against both quantum and classical computers and can interoperate with existing communications protocols and networks [GSMWP4].

2.3. How is the term "Hybrid" used?

The terms Hybrid Key Exchange, Hybrid Key Combination, Key Combination Scheme or similar refer to the execution of multiple cryptographic key exchange methods (e.g., QKD, PQC key encapsulation mechanism (KEM), Diffie-Hellman key exchange, offline key exchange) and the subsequent combination of the keys obtained. In fact, once all the exchanges have been concluded, the keys obtained are combined into a single key through a key derivation function (KDF), which is typically a type of hash function.

The goal is to obtain a cryptographic key that is at least as secure as any of the components.. Essentially, for an attacker to obtain the final key, it would be necessary to break all the mechanisms of key exchange involved.

In the QKD context, the term hybrid most commonly refers to the parallel combination of PQC and QKD. Any risk to the security of a particular method of key exchange (such as an implementation flaw or an algorithmic breakthrough) can be mitigated by combining two or more methods of key exchange, which are not subject to the same type of vulnerability, in parallel.

On the other hand, in the PQC context, the term hybrid is generally attributed to a combination of PQC and traditional techniques such as Diffie-Hellman or RSA. To our knowledge there is not really a standard and unanimously applied vocabulary. An attempt was made by the IETF, but currently it does not involve QKD.

From the perspective of QKD-PQC integration, the possibility of using PQC to authenticate the classical discussion channel of QKD, and QKD to obtain a shared symmetric key can be described as a hybrid form. Furthermore, Hybrid encryption1 has nothing to do with the hybrid key exchange. However, since they are both cryptographic terms, there is a risk that they can be confused. Hybrid encryption , in fact, applies to most cryptographic protocols that we use in real contexts. Among the steps of a cryptographic protocol there are two fundamental ones, the cryptographic key exchange of the initial handshake, and the encryption with which the application data is protected once a symmetric key has been

exchanged. The term hybrid, in this case, refers to the fact that the key exchange is generally carried out with public key cryptography techniques (e.g., Diffie-Hellman, RSA, Post-Quantum), while the encryption uses symmetric key cryptography techniques (e.g., AES).

3. Taxonomy on Hybrid Security

3.1. Security primitives

Mathematical security primitives are well-established, low-level cryptographic algorithms that are frequently used to build cryptographic protocols for computer security systems. A graphic summary is presented in the “Handbook of Applied Cryptography”

Security primitives are one of the building blocks of every security system, e.g. the well-known ones, TLS, SSL, etc. Security system designers, not being in a position to definitively prove the security of primitives they use, must assume these to be secure. Choosing the best primitive available for use in a protocol usually provides the best available security.

Also procedures based on physical properties can be considered physical security primitives. These primitives are based on elementary physically based procedures, which in turn can also be described mathematically, similarly to traditional cryptographic primitives. The main difference to the latter is provided by the fact that a specific mathematical description of some physical model plays a significant role in their definition in addition to some (possibly very involved) purely computational ingredient. While in the case of mathematical security primitives it must be guaranteed that the abstract mathematical computation and communication are faithfully represented by a real computer, it is the implementation validity of the physical models that also needs to be justified in the case of physical security subsystems. Typical elementary physical processes (“physical primitives”) are pure Quantum Key Distribution (over an authentic classical post-processing channel), or physical layer encryption (over a wire-tqp channel)

3.2. Physical Security subsystem

Physical security subsystems are ultimately composed of mathematical and physical primitives. For the security properties of a subsystem it is essential to include these primitives into a sensible overall architecture. A misconceived architecture or workflow in a security subsystem can result in vulnerabilities that none of the separate primitives is subject to. Let us consider the example of a QKD physical security subsystem providing secret key sharing. A QKD system is a physical subsystem combining the physical version of the primitive of key agreement with the mathematical primitive ‘authentication’ for the classical channel communication. Implementing a combination of these physical and mathematical primitives forms a QKD subsystem

3.3. Organizational subsystems

Organizational security is typically provided by formalized organizational rules and procedures involving human interaction. E.g. a human being or an administrative entity, identified by its credentials and affiliation, receives pre-shared-key. It is up to the respective security establisher to secure the procedures against identity theft and forgery. Organizational security subsystems differ from (typically computer-based) cryptographic primitives only by the means to execute the procedure, involving human procedures and trust instead of computers.

3.4. Overall security systems

In cryptography, a cryptosystem is a suite of cryptographic primitives, physical and organizational subsystems needed to implement a particular security service, such as e.g., confidentiality (encryption). Typically, these are intertwined into composite security protocols. The latter perform security-related functions and apply the discussed methods, in a sequential form. A protocol describes how the algorithms/procedures should be used and includes details about data structures and representations, at which point it can be used to implement multiple, interoperable versions of e.g., a program.

A cryptographic protocol typically incorporates one or more of these subsystems:

- Key agreement or establishment
- Entity authentication
- Symmetric encryption and message authentication material construction
- Secured application-level data transport
- Non-repudiation methods
- Secret sharing methods
- Secure multi-party computation

For example, Transport Layer Security (TLS) is a cryptographic protocol that is used to secure web (HTTPS) connections. It has an entity authentication mechanism, based on the X.509 system; a key setup phase, where a symmetric encryption key is formed by employing public-key cryptography; and an application-level data transport function providing confidentiality. These three aspects have important interconnections. Standard TLS does not have non-repudiation support.

3.5. Hybrid security systems

A hybrid security system is a system where different instances of security systems are combined, so that each instance needs to be broken individually to break the security system as a whole. E.g. the key used in Public-key cryptography computing a ciphertext decodable with a different key used to encode can be mixed with a symmetric key known to each party to strengthen security. If two methods of the same kind are run one-by-one to strengthen each other, the resulting security system is considered hybrid.

Typically, such hybridization is used in composite protocols so that the methods of the same kind are combined to increase the robustness against attacks. E.g. RFC9370 describes multiple key-exchanges leveraging different cryptographic primitives to protect the IKEv2 protocol described in RFC 7296. RFC9370 therefore represents a hybrid cryptographic protocol.

4. Hybrid security solutions integrating PQC and QKD

In this section, hybrid security solutions of PQC and QKD are investigated.

In literature, techniques that integrate the two solutions are usually referred to as hybrid solutions.

This section describes in which sense an integration is addressed, clarifying the possible natures of the hybridization. In the first part, the realization of key exchange protocols combining different key exchange (equivalently key establishment or key agreement) methods

like QKD and post-quantum KEMs is described. The second part discusses the possibility of applying post-quantum solutions to authenticate the classical communication inherent in the post-processing of a QKD protocol.

4.1. Key combination approaches

Two (or more) different key exchange protocols are carried on in parallel and preferably without any dependencies. Once concluded, the obtained secrets become the inputs of a key combination function that outputs a unique key. The objective of this procedure is to derive an output key with security guarantees greater or equal to all the input ones. At a high level, the following diagram describes this process. Key combination techniques turn out to be particularly interesting in this specific historical period where we are witnessing an increasing worry concerning quantum computing threats described in the previous section. Combining traditional cryptographic solutions with quantum-resistant (but possibly immature) solutions provides security guarantees at least in the short-medium term.

While the versatility of post-quantum KEMs (Key Encapsulation Mechanism) allows them to be potentially included in all previously described hybrid schemes in an (almost) straightforward way, the same cannot be stated for QKD. There are two main reasons behind this difficulty.

The first one has been motivated in the previous section describing ITU-T X.1714. A priori, it is not guaranteed that a generic key combination scheme can preserve the unconditional security targeted by QKD since most available recommendations consider computational security. This may result in producing a key that is weaker than a QKD key even if the former is derived from the latter.

The second one is that QKD is not a public key scheme and requires a dedicated infrastructure. This is not contemplated by certain key-combination schemes, e.g., IETF Informational Draft in Hybrid TLS 1.3, which assumes the possibility of carrying out multiple key agreements through a unique communication session. Therefore, for native compatibility with QKD of a key combination scheme, it is required that the use of out-of-band exchanged keys is supported. As an example, QKD keys can be treated as preshared keys in protocols that accept them, e.g., Wireguard. A high-level representation of this scenario is reported in the following diagram distinguishing the classical layer and the quantum one.

4.2 Authentication in a QKD protocol

The “bare bones” basic QKD primitive is a key agreement one that is based (among others) on the availability of an authentic classical communication channel for ITS, so called post-processing phase. We note here in passing that the latter stage is a purely algorithmic/ classical communication one and involves a number of classical cryptographic (but Information Theoretically Secure - ITS) subprotocols necessary to guarantee overall correctness and security.

Moreover, the QKD primitive is not directly usable in a standalone fashion as authentic classical communication channels are not readily available. Additionally, it is well known that in the absence of an authenticated classical communication channel between the parties, , a direct usage of a QKD primitive can easily be broken by means of straightforward man-in-the-middle (MITM) attacks where the adversary cuts the channels and impersonates one party to the other and vice versa during the whole protocol.

More generally, communication authenticity is, in fact, a necessary condition for the security of any pure key generation primitive, as its absence immediately opens the way for its break by means of an MITM. (It is also part of the sufficient conditions for any sort of provably secure key generation.) In this sense, authentication (the security technique for providing an authenticated channel) is the basis of any form of secure communication in which it is necessary to defend against the presence of an active attacker and is therefore considered to be of equal or even greater importance than confidentiality. For this reason, key generation primitives of any sort are bundled with an authentication technique to give rise to a composite key-transport, -agreement or -growing protocol.

Authentication is achievable through the use of various types of classical security techniques. We specifically underline that any such technique is a combination of appropriate cryptographic but also organizational measures. The authentication constructions correspond to various trade-offs in terms of performance and security, but in all cases, an offline interaction between the involved parties is required.

In symmetric cryptography, the main tool used to achieve authentication are Message Authentication Codes (MAC), which depend on the message to be sent and a secret key shared by both communication parties. With a MAC, a tag is derived, whereby a concatenation of both message and tag is sent, so that the attacker cannot successfully modify the message and produce the corresponding tag as the probability of consistently modify the latter is low. Specifically, often a hash function in combination of the secret key is used to this end, which is equivalent to having a family of hash functions, parametrised by the value of the secret key. These are known as HMACs. Naturally, an attacker cannot produce a correct authentication tag for a message of his choosing, without knowing the secret key.

Another suitable solution, belonging to a public-key scenario, is represented by a Digital Signature which provides non-repudiation, an even stronger cryptographic property than message authentication. Combining his/her private key with the message, the author produces a signature that can be verified by anyone knowing the corresponding public key while the signature cannot be forged without the knowledge of the private key. In this scenario, the information that must be preshared in a secure way is the public key and a guarantee of authenticity of the identiy of the latter, in other words the registration of the public key to the identity must be enforced through organisational measures. Typically, this is done by repositories securely providing (by a public key registration process) the public key of legitimate participants, known as Public Key Infrastructures, or PKIs. The key of each new participant needs to be securely, traditionally physically, brought to the PKI. In a network of N participants this means N interactions with the PKI.

With respect to authentication, QKD provides no exception, compared to other key distribution methods. The QKD primitive, discussed above, can be extended to a composite key growing protocol by utilising an appropriate message authentication primitive. It is known that this can be done on the same level of security as QKD by using a special HMAC family, named Almost Strongly Universal$_2$ (ASU_2) hash functions. This family assures ITS authentication: no attacker will have a better advantage than randomly choosing a tag or a signature, regardless of her/his computing power. (More precisely authentication by ASU_2 hash functions is ε-composable and the combination with QKD is also ε-composable: the latter term means that any such protocol can be securely combined with any other one of

this type, whereby the ε is a linear ingredient in a quantitative measure of the potentially negligible probability for any attacker to break the combination. In what follows we use the more familiar term ITS, abstaining from the use of precise but somewhat cumbersome terminology.) The price to pay for using ASU_2 hashing, however, is that each member of the family is parametrised by a separate symmetric key. In practice, this effectively means that each new communication message in QKD post-processing requires a different (one time) key for each authentication. In a sense the ASU_2 hash functions can be dubbed to be one-time MACs. Note that this method provides implicit key authentication – for each legitimate party all the communication messages must have come from a party that possess the same key – i.e. the other legitimate part in the QKD protocol. (together with the key confirmation in post-processing this guarantees explicit key authentication).

The ITS requirement immediately rules out all available public-key solutions since they are based on computational security. The mentioned, one-time authentication itself does not represent a big deal in QKD, since authentication keys for subsequent QKD rounds, after the first one, can be obtained from the previous round. (This is based on the property of QKD-based key growing known as forward security: the key of a subsequent key session is independent of the authentication key in a previous key session.) What is critical, instead, is the initialization of the authentication key in the first QKD round. To this end, again, PSK distributed by organizational means is required. This is, naturally, a burden or scalability, i.e. addressing situations of significant proliferation, when no pre-shared key is already available.

Distributing pairwise PSK manually is really not an efficient and scalable solution in general but may be suitable for some specific scenarios. Other methods for distributing PSK exist, for example based on a trusted third party, such as first envisaged by Needham and Schroeder in the Kerberos protocol for symmetric key based authentication[2], from which further developments and improvements have been made. In this context, QKD can provide a strong method for refreshing PSKs, either between endpoints, or between each endpoint and the trusted third party/parties.

4.2.1 PQC to initialize the QKD authentication key

A possible solution to the highlighted issue can be identified through a security-flexibility trade-off.

From a security perspective, the best solution to realize an authenticated key agreement scheme is to combine QKD with one-time MACs, exchanging the initial authentication key with procedural security guarantees, e.g., by hand. Assuming that there is no gap between theory and practical implementation, the overall scheme is ITS and so it is resilient even in the face of an attacker with unlimited resources. However, with QKD depending on the use case scenario, the authentication key initialization is not always doable.

From a flexibility perspective, an authenticated key agreement scheme based on computational security, e.g., PQC, represents the most valuable direction in both scalability

[2] Roger Needham and Michael Schroeder, Using Encryption for Authentication in Large Networks of Computers, http://dl.acm.org/citation.cfm?doid=359657.359659

and usability since it can be directly integrated into already existing communication infrastructures.

A possible trade-off appears to consist of mixing the two strategies. Authentication and key agreement are carried out as in the unconditional secure solution, i.e., with QKD key exchange and one-time MAC authentication, while the initialization of the first authentication key is realized through a post-quantum KEM. With respect to the former solution, much more flexibility is gained at the price of reducing the security. Delving deeper into this point, the trade-off is describing a solution with quantum-resistant, but computational, security just in the first QKD round. Indeed, one may observe that the overall security of a protocol is determined by the security of its weaker component. However, if no attacker is able to intervene and break this first part in real-time, further QKD rounds guarantee unconditionally secure authenticated key agreement procedures. This resulting security model is usually defined as “Everlasting Security”. In the scope of this section, the trade-off presented above is in all respects an integration of PQC and QKD.

A couple of hints are useful for practical scenarios. The first one is actually an open problem: everlasting security holds as long as an attacker can’t force a reinitialization. Therefore, in concrete solutions it is crucial to mitigate attacks aiming at consuming all authentication keys to bring back the parties to a condition where no unconditional secure secret is already available.

Every time a one-time MAC is computed, the corresponding secret key must be discarded, and this requirement represents a surface where a hypothetic attacker, who might thwart PQC, may attempt the above strategy.

The second one is that in the face of the “store now, decrypt later” attack described in the previous section, which is the most relevant current threat rising from quantum computers, unconditional confidentiality - guaranteed by the QKD protocol - with initial computational authentication might be enough. Therefore, in the transition to quantum-resistant solutions, multiple trade-offs can be considered.

However, we note that in this discussion there is an important catch. As any sufficiently strong key distribution mechanism PQC KEM does not come for free. In order to avoid a trivial MITM it requires a message authentication mechanism that boils down to the asymmetric case, described above. As stated, to this end, a PKI is needed, and one must:

- trust its integrity,
- accept the burden of direct (best of all physical) interaction between each participant and the PKI, which is proportional to N – the number of participants, and
- be security-wise content with the key distribution method (we note that the PKI distributes the public key of Alice to Bob, and vice versa, by using a KEM and the public keys of Bob/Alice that it possesses).

At first glance the trade-off is acceptable as it replaces the quadratic (more precisely proportional to N(N-1)/2) burden of individual interactions that would be needed for a pairwise PSK distribution “by hand”. We show, however below that one can need only ii) N interactions, if i) trusting the integrity of an analogue to the PKI is acceptable but not iii) relying on public key-private key cryptographic methods.

4.2.2 Symmetric ITS Methods to initialize the QKD authentication key

The idea of a trusted repository or as it traditionally called – a Key Distribution Center (KDC) is crucial for this solution – distributing a secret (session) key between any two parties in a network of participants. This method is very old indeed. It dates back to the Kerberos protocol and even the preceding Needham-Schroeder Symmetric Key Protocol (from 1978). In essence, these protocols, very roughly, reduce to the following mechanism. The KDC initially shares a secret key with each of the participants. Essentially, a secure method has to be plannef for this sharing by the organisation before the QKD systems are deployed to site, or it can be done by trusted couriers, or by the participants physically getting such individual PSK from the KDC by simultaneously proving their identity to it. This is an organisational effort but it is equivalent to the one in ii) above. Moreover, also as above, it is scalable. Each new participant needs to contact the KDC, prove its identity and get its individual key. When each currently active party has its PSK, it asks the KDC for a session key with another party and the KDC enables such a key by utilizing the two individual PSKs. Here, we intentionally neglect significant technical details, allowing to thwart replay attacks and the exact sequence and direction of communication.

The important thing is that this traditional scheme can easily be modified and adapted to allow distributing individual keys between any two network participants by means of ITS (ε-composable) transport, specifically using One Time Pad (OTP) plus ASU_2 message authentication. In what follows we understand under “using a key for ITS transport of a string” three steps on behalf of the sender. The first involves encrypting by OTP the string to be sent by a sub-portion of the employed key-string. The second is computing an ASU_2 tag of the already computed cypher text, obtained by means of the remaining part of the key string. The third is step is concatenating the cypher text and the tag. This is followed by sending the concatenation to another participant, who possess the key string, employed by the sender. The latter first checks the validity of the sent cyphertext by computing the tag of the cyphertext, employing the second portion of the key. If these do not coincide the whole message is discarded (Note: to this end the receiver must know, where the tag starts.) Then it decrypts the cyphertext by the first portion of the key string. Finally, the sender and receiver discard the used key-string and move their “pointer” in their respective overall (long!) PSK to a position starting after the end of the used key-string. In a sense their PSK is shortened by the used key-string.

To do so each new network party should contact the KDC and get from it a very big amount of individual (to it) key: e.g. a party A gets the overall very long key-string $K_{KDC,A}$. If subsequently this party needs to share some amount of initial key with another party B, to serve as an initial key for a QKD session between A and B, then an ITS procedure as follows can be applied. To facilitate the mentioned above problem of identifying the end of the end of the cyphertext four standard string lengths l_1, l_2, l_3 and l_4 can be chosen, so that $l_2 > l_3 > l_4$, whereby $l_2 - l_3$ and $l_3 - l_4$ are two identical numbers, equal to the length of the key parameterising a member of the used ASU_2 hash family:

1. A communicates her request to get a key with B (R_B)using (in the above sense) a first portion of her key with the KDC K_{KDC,A_1}, of length l_1, which a substring of $K_{KDC,A}$. It must be chosen, whether overall Information Theoretical Security, or only information theoretical authenticity of this first request is deemed necessary. Naturally such a principle choice (to be made before any communication) will affect the length of l_1.

2. The KDC answers with a sub-string of the key-string $K_{KDC,B}$, namely: K_{KDC,B_1} of standard length l_3. To do so the KDC uses another portion of its key with A, namely K_{KDC,A_2} of length l_2.
3. A then generates a random key string $K_{A,B}$ of standard length l_4 and sends it to B, using the sub-string K_{KDC,B_1}.
4. A and B can now use $K_{A,B}$ for initial authentication of the communication channel between them. (Note that l_4 has to be dimensioned in such a way that the length of $K_{A,B}$ is typically sufficient for initial authentication. If it is longer than that, the superfluous part of the string can be discarded. If the length turns not to be sufficient, the procedure must be repeated with a securely communicated longer string length.)
5. The whole process needs to be repeated for obtaining initial authentication key of A with any other party or, generally, between any two parties.

The key-string used for a secure transport of another string must be obviously longer, as follows from the discussion above. Naturally, the whole process critically depends on distributing very long key-strings between the KDC and any individual “network party”. Nowadays, this is not a real technical problem and can be organized trivially. If for some reason some party runs out of initial authentication key, it needs to be “rekeyed” by “visiting” again the KDC.

Note that this method is equivalent in effort to that of i) and ii) presented in the previous subsection and allows PSK availability to A and B plus entity authentication of B for A. While the inverse is not true with the presented protocol, this is also possible by an appropriate extension.

The above analysis shows that while using PQC for authentication of the first round in post-processing communication in QKD only creates the appearance of a higher degree of practicability and scalability in a realistic scenario. This is not really the case. Indeed, such combination of PQC and QKD allows a significant reduction of effort in the distribution pair-wise PSK by hand at the expense of accepting trust in a PKI and the reduction of the overall level of security by using a communication-protocol of lower level of security, at least in the first communication round in a chain ensuring key growing. (As outlined above the forward security of the full process wards off attacks unless these can be successfully performed and combined with a MITM during the first communication round.) However, as shown above the same effect can be achieved by replacing a PKI by a KDC and the PQC communication methods by ITS ones. While the later is easy to organize in current and soon-to-be realized Quantum Communication Networks, the PQC based method might still have advantages on a very large scale, as existing PKIs can be simply reused and there is no need of establishing new analogous infrastructure.

5. Overview of state of standardization on key combination

In the context of key combination techniques, there is no specific standard leveraging on a consolidated approval of the whole cryptographic community.

However, there are multiple valuable documents, reports, recommendations discussing this topic and addressing different levels of generality. Some works describe a general-purpose solution trying to include, in the combination scheme, keys coming from any secure source.

Other approaches are related to an already existing cryptographic protocol proposing a little extension of it for accepting input keys from other key exchange methods. Some of these works are reported in the following.

5.1. ITU-T

Recommendation ITU-T X.1714 [ITU-T-HYB-1] “Key combination and confidential key supply for quantum key distribution networks” discusses methods to combine different key exchanges, including QKD, specifying some security requirements. In particular, since the security of the key supplied by QKD is unconditional, it is expected that the resulting combined key maintains the same guarantee. It must be pointed out that this is not trivially satisfied by a generic key combination method. An example of a technique, ensuring this requirement, is an exclusive or (XOR) operation of all input keys with the same length.

There are also two new Work Items in Study Group 17 (Security) dealing with the use of PQC. There aimed to deliver new Technical Reports (TR):

TR.ac-pqc: Guidance on use of advanced cryptography based on PQC

TR.QKDN-SP: Overview of security profile for Quantum Key Distribution Networks in hybrid mode.

ITU-T Technical Report FG QIT4N D2.2 [ITU-T-HYB-2] “Quantum information technology for networks use cases: Quantum key distribution network” lists some hints to integrate QKD in many modern cryptographic solutions. With respect to the scope of this deliverable, the most relevant one is the suggestion of a three-layered key exchange combining a QKD protocol, a post-quantum KEM and a traditional public key scheme. In the document, this solution is generically referred to as hybrid quantum-safe scheme but no more details are reported about its implementation.

5.2. ETSI

The document ETSI TS 103 744 [ETSI-HYB], entitled “Quantum-safe Hybrid Key Exchanges”, specifies several methods for deriving cryptographic keys from multiple shared secrets. The shared secrets are established using existing classical key agreement schemes, like elliptic curve Diffie-Hellman (ECDH) and post-quantum KEMs. The goal of this technique is to support the migration to PQC passing through a transition period where a coexistence with traditional public key schemes is expected. One may wonder why not to directly use a post-quantum solution. At this stage, an intermediate solution is needed to prevent possible unknown attacks arising from the technological immaturity of PQC. The document leaves the door open to include also a preshared key in a key combination scheme, which may be established out-of-band from a previous session or an alternative method like QKD.

5.3. IETF

The document RFC 8784 [IETF-HYB-2] “Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security” describes how to achieve post-quantum security (or quantum-resistant security) in the IKEv2 through a modification of the key derivation including a preshared key. This is a symmetric key, called PPK (Post-quantum Preshared Key) in the RFC, that is assumed to be configured within the IKE devices in an out-of-band fashion. Further specifications of this out-of-band procedure are

outside the scope of the document. Actually, this is a non-trivial issue and QKD can represent a possible solution in this direction, thus realizing a key combination scheme including it.

The IETF informational draft [IETF-HYB-3] “Hybrid key exchange in TLS 1.3” primarily aims at preparing for post-quantum algorithms. A key combination scheme integrating traditional public key cryptography and PQC to address the ephemeral key exchange in TLS 1.3 is described. As reported in the draft “the messages from the two or more algorithms being hybridized will be concatenated together and transmitted as a single value” which does not directly include QKD as an eligible algorithm since it needs a dedicated communication.

RFC 9370 [IETF-HYB-4] "Multiple Key Exchanges in the Internet Key Exchange Protocol Version 2 (IKEv2)" specifies a mechanism for combining up to seven additional key exchanges with the classical Diffie Hellman exchange. Whilst this provides a method for building hybrid classical/PQC key exchanges, or hybrid PQC/PQC, it does not provide a method for combining QKD.

5.4. NIST

Recommendation NIST SP 800-133 [NIST-HYB-1] for cryptographic key generations lists some approved methods for combining two or more symmetric keys, referred to in the document as “component keys”, into a unique one. These methods are:

- Concatenation of component keys;
- Exclusive-Or (XOR) of component keys;
- Key extraction process through an HMAC applied to the concatenation of component keys with HMAC-key a secret or public *salt*.

Recommendation NIST SP 800-56C [NIST-HYB-2] for key-derivation methods in key-establishment schemes allows to consider keys obtained in a *hybrid* way. This hybridization essentially consists in concatenating a symmetric key coming from a NIST approved key-establishment scheme with a symmetric key possibly generated using some other method.

For the current 4th round of PQC competition, the 5th PQC Standardization Conference is scheduled for April 10-12, 2024. Through many topics, information will be given on the NIST SP 1800-38, the Migration to Post-Quantum Cryptography NIST publication, the document which will also have an impact on future implementations of PQC solutions for current and future QKD equipments.

6. Proof-of-Concepts on Hybrid QKD- PQC

QKD is foreseen to greatly benefit from PQC (in fact, it will be not an option for QKD). So, to combine QKD and PQC in Proof-Of-Concepts (PoCs) works (both theoretical and practical field tests) are of utmost importance.

It is a general trend of work in progress, in many countries through the world (France, Italy, UK, Netherlands, South-Korea, China...).

For example, the combination of QKD and PQC technologies is illustrated in the most advanced country for the use of QKD: South-Korea

SK Telecom operator delivered an important 6G White Paper in 2023 [QKD-PQC-4 https://www.sktelecom.com/en/press/press_detail.do?idx=1575]. In this document, they anticipate the combined use of PQC and QKD as part of the evolution of networks for 6G:

In the figure above, they consider that the received quantum key from the QKD device and th

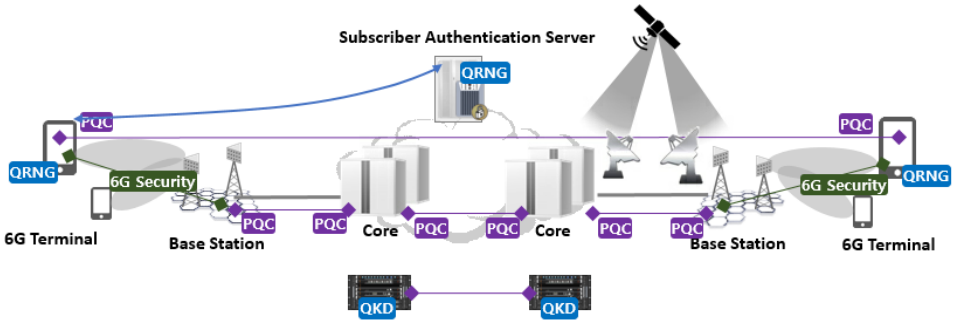

Figure *1:* 6G prediction based on quantum technology

e symmetric encryption key received from the PQC module should be hybridized for encryption. They judge that QKD and PQC are complementary technology to each other.

6.1. Example 1 of hybridization PoC

In the work [QKD-PQC-1] the hybridization is i) following the Muckle scheme Dowling et al. (2020) that puts forward a hybrid method for authenticated key exchange (specifically it is based on a secure hybridization, i.e.combination, of several keys of QKD, PQC and PKC origin, and the additionally, authentication of the key distribution using PSK, instead of PQC SIG), and, ii) additionally, is relying on extended hybridization involving multiple media/paths.

As reported in the paper [QKD-PQC-1], the key exchange module between PoPs has been modified to manage not only QKD keys, but also PQC ones. The systems also allow the use of several key exchange modules in parallel, even with the same PoPs. The KMS systems receive the keys (QKD and PQC or any other key exchange mechanism) in a transparent manner from these key exchange modules, so the KMS can establish different quantum safe key exchange session with other PoPs, using QKD links, PQC links and a combination of both in a simple way.

To do that, the key exchange modules are running in parallel on each node delivering keys to the KMS. The payload functionality of the key exchange modules is vendor- and technology-independent, and the only exposed interfaces are the ETSI GS QKD 004 or 014 ones. The multiple KMSs only get notice of a new link between two PoPs. This design enables generic integration of additional links, and the keys generated by PQC are internally managed by a KMS exactly as any key material generated by QKD. All these interfaces can be seen as quantum safe key delivery interfaces, whereby (Q)RNG serves as key source Herrero-Collantes and Garcia-Escartin (2017).

The PQC link is implemented as a TCP connection, with package delivery granted and in order. That allows to simulate a potentially infinite, protected stream of keys with PQC between any two peers of the network and identity authentication of the end points is required only when a new TCP stream is started. The use of PQC links allows to have long distance quantum safe links where QKD cannot reach right now, but the current design, exposing standard QKD interfaces, makes possible to replace PQC by QKD links when the

technology will be mature enough and with a minimal impact on the rest of the architecture. The hybridization of QKD and PQC keys will, as described earlier, deliver a key exchange system with significantly reduced side channels due to the use of principally different technologies and implementations thereof.

The key hybridization process is done applying a hybridization KDF. The key hybridization process in performed as an internal process on the KMS that manages different internal key stores. The key hybridization process needs to process the appropriate key bits so that a new, hybrid key is computed, stored and handed over to applications and encryptors.

Alternatively, hybridization may be left to the application, since the application oversees enforcing the required security level itself – by picking a key or a combination of keys exchanged under the right security paradigm.

6.2. Example 2 of hybridization PoC

The paper [QKD-PQC-2] provides an example of a proof-of-concept implementation. In particular, the cryptosystem on a QKD hardware prototype is integrated with the QKD processing, PQC key exchange and secret state masking via physical unclonable functions (PUFs) all running on a single field programmable gate array (FPGA).

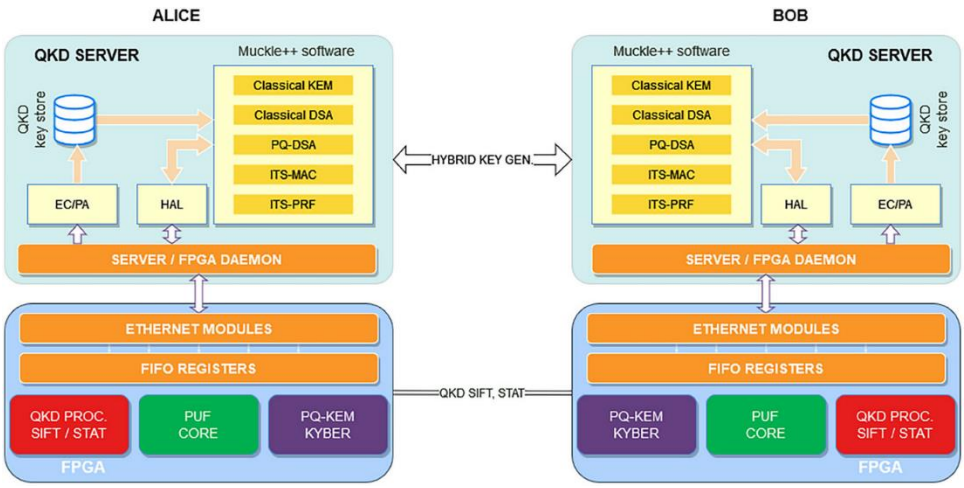

Figure 2: System architecture

The cryptosystem is deployed on a commercial QKD system prototype. The main software runs on the QKD servers where the classical primitives, the PQ-DSA, the MACs, and PRFs are executed along with the QKD error correction and privacy amplification. A hardware implementation of the PQ-KEM and the PUF was designed to execute them on the same FPGA that performs the core QKD functions.

6.3. Example 3 of hybridization PoC

In this example, the use of quantum-resistant technologies to enhance the security of Transport Layer Security (TLS) protocols is presented. This collaboration work (Netherlands, France and Israel) highlights that none of the existing works combine quantum key distribution (QKD) and post-quantum cryptography (PQC) at different stages of the TLS protocol for enhancing security. The authors present a hybrid quantum-resistant TLS protocol that integrates QKD and PQC to achieve enhanced security. They also aim to analyze the performance implications of this solution in an experimental networking

scenario. The document claims to be the first experimental demonstration where fully commercially available QKD devices are integrated with PQC algorithms.

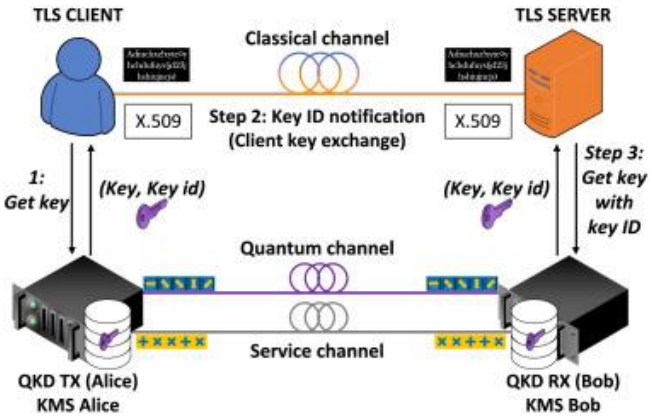

Figure 3: Integration of QKD into classical network security protocols (e.g., TLS): Architecture overview

Below are figured the use of PQC solutions in combination with QKD implementation:

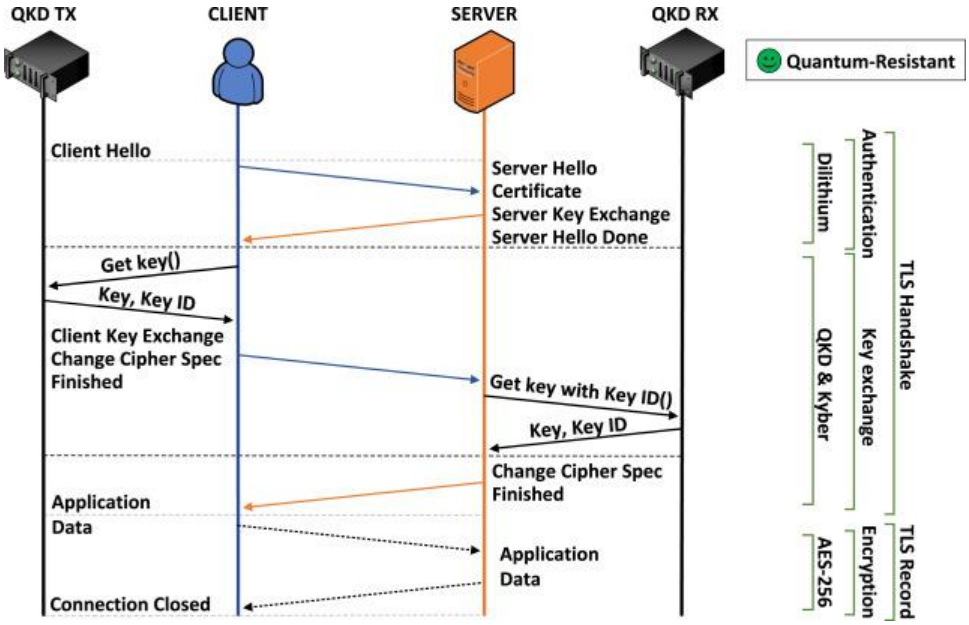

Figure 4: Hybrid quantum-resistant TLS 1.2 protocol: Quantum handshake steps

Here, the use of Dilithium (PQC signature algorithm) and Kyber (PQC ciphering algorithm) are illustrated in the Authentication and Key Exchange steps. In this publication, they present 2 possibilities to combine current and PQC security primitives: Concatenation and XORing.

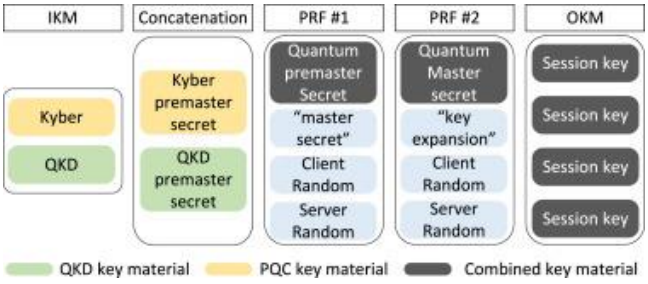

Figure 5*:* Generating a quantum-hybrid premaster secret via concatenation and hashing of cryptographically secure byte arrays

To compute the bitwise XOR, the QKD-based key is introduced inside the protocol at the master secret level, just after the first pseudo random function.

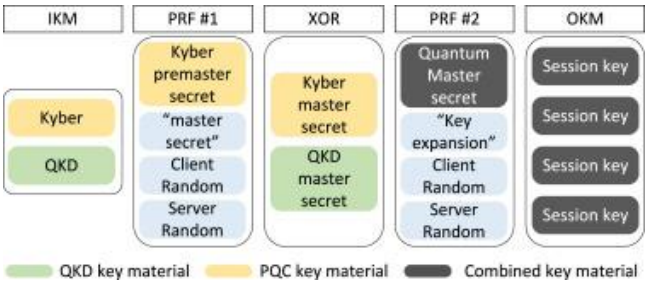

Figure 6: Generating a quantum-hybrid master secret by by a KDF which involves a XORing stage using byte arrays of the same size, as well as hashing stages

The last part of the publication deals performance analysis of classical and PQC cryptographic alternatives for TLS versus performance analysis of hybrid QKD-PQC quantum-resistant TLS (no real blocking point). They conclude by explaining their choice for TLS 1.2 (very wide availability) and that the architecture of TLS 1.2 is currently more suitable than TLS 1.3 for adopting QKD as a method for the key exchange.

6.4. Example 4 of hybridization PoC

A Telecom Operator was proving that QKD-keys are being synchronized via the direct quantum channel over fibre between the two QKD Devices. Furthermore, Firewall connectivity to Quantum Key Distribution devices using ETSI-QKD 014 REST API and formation of IPSec tunnel has been tested

For establishing quantum-safe IPsec connections, a version of the IKEv2 protocol defined in RFC8784 has been used. This standard provides a detailed description of how keys sourced from QKD (Pre-shared Post Quantum Keys PPK) are mixed with keys derived from a traditional PKI method and the necessary protocol steps in IKEv2. An attacker of the IKEv2 protocol is unable to determine the encryption key. A Cryptographic relevant Quantum Computer CRQC may be able to attack the PKI exchange, but can't attack the PPK since it is never exchanged in IKEv2. Since the PPK is a quantum-safe symmetric key itself, the IPsec connection is quantum-safe.

It is well known that generating new PKI key-pairs is a computationally demanding process involving many CPU cycles. Hence private/public key pairs are infrequently renewed. In contrast, a QKD system is quite proliferate in providing new keys and cryptographically combining keys is extremely resource efficient. As this setup showed, PPKs can be rolled

over frequently and IPsec tunnel services using IKEv2 can be implemented in such way that key-roll-over is not traffic affecting.

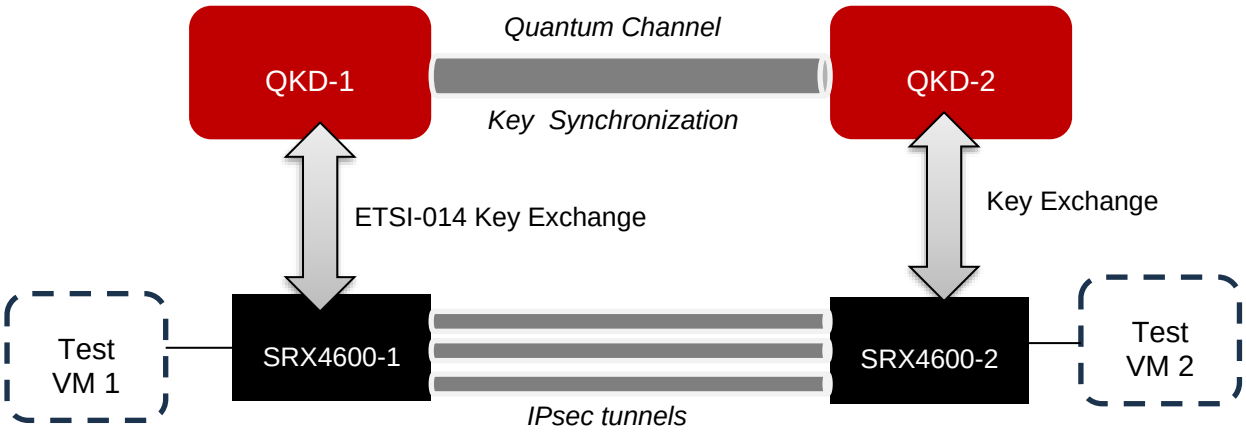

Figure 7: Proof of Concept: QKD - IPvpn service

7. Conclusions and Recommendations

QKD is a physical layer method that allows an unconditional secure distribution of random keys between remote users. For instance, QKD is a method that exploits the physical properties of photons to create and distribute secret keys that can then be used by existing ciphers. Considerable work on the standardisation of QKD, and the assurance of QKD implementations, is being led by the ETSI QKD ISG, including the publication of the first protection profile for a pair of prepare and measure QKD systems.

The goal of PQC is to develop cryptographic systems that are secure against both quantum and classical computers and can interoperate with existing communications protocols and networks.

In other words, unlike QKD, PQC relies on algorithms that are believed to be too complex for quantum computers to crack. PQC is still in active development, and it's currently undergoing standardization by NIST. Also, the model of Post Quantum Telco Network is under definition and development [GSMA4]

This white paper aimed at describing Hybrid scenarios based on QKD and PQC. In QKD, the term hybrid most commonly refers to the parallel combination of PQC and QKD. On the other hand, in the PQC context, the term hybrid is generally attributed to a combination of PQC and traditional techniques such as Diffie-Hellman or RSA.

Hybridisation of PQC with QKD can increase the flexibility of deployment, for example by allowing recovery of an authenticated channel after symmetric authentication key material has been exhausted, or by preventing exhaustion attacks on the stored symmetric key material, by first requiring PQC based authentication before any symmetric MAC is generated.

In the event of a discovery of a vulnerability in the algorithm being used for PQC, hybridising encryption using symmetric key material from QKD provides security against retrospective and zero day attacks.

Annex A Document Management

Annex A.1 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
V1.0	20 October 2024	First version of the document	IG/TG	Antonio Manzalini (TIM)

Other Information

Type	Description
Document Owner	Internet Group
Editor/Company	Antonio Manzalini / Telecom Italia (TIM)