Integration of PQC and QKD: Applications, Challenges and Implementation Frameworks

Elina Kalnina[▽][0009−0004−2680−3220], Rihards Balodis[0000−0002−9641−080X], Edgars Celms[0000−0001−9608−3792], Sergejs Kozlovics[0000−0002−7085−383X], Inara Opmane[0000−0003−0760−9268], Krisjanis Petrucena[0009−0008−5713−5914], Edgars Rencis[0000−0002−1606−4944], and Juris Viksna[0000−0003−2283−2978]

Institute of Mathematics and Computer Science, University of Latvia
Raiņa bulv. 29, Riga, Latvia, LV-1459
rihards.balodis@lumii.lv, edgars.celms@lumii.lv, sergejs.kozlovics@lumii.lv, inara.opmane@lumii.lv, krisjanis.petrucena@lumii.lv, edgars.rencis@lumii.lv, juris.viksna@lumii.lv
[▽] Corresponding author: elina.kalnina@lumii.lv

Abstract. The advent of quantum computing poses a significant threat to classical cryptographic systems, necessitating the development of quantum-resistant solutions. Post-Quantum Cryptography (PQC) and Quantum Key Distribution (QKD) have emerged as complementary approaches to achieving quantum-safe communications. This paper explores the integration of PQC and QKD technologies, emphasising their potential applications, practical implementation frameworks, and the challenges associated with their deployment. We have also analysed key use cases that are compatible with our existing network equipment, such as the Centauris encryptors, Juniper devices, and Cisco routers. Through an examination of the use cases and experimental findings, this work provides valuable insights into building scalable, robust, and quantum-resistant infrastructures for long-term data security.

Keywords: PQC, QKD, quantum resistant cryptography, post-quantum cryptography, quantum cryptography.

1 Introduction

The rapid advancement of quantum computing poses a significant threat to current cryptographic systems, which are heavily reliant on classical algorithms that are vulnerable to quantum attacks. To address this, two primary technologies have emerged: Post-Quantum Cryptography (PQC) and Quantum Key Distribution (QKD). While PQC aims to replace traditional cryptographic algorithms with quantum-resistant alternatives, QKD enables secure key exchange through quantum mechanics, offering theoretically unbreakable encryption when combined with one-time-pad encryption. However, both methods have distinct limitations. The resilience of PQC against future quantum computing developments

remains uncertain, as there are currently no definitive security proofs confirming its complete resistance to all quantum attacks [1]. Conversely, QKD, although promising, faces practical limitations due to hardware requirements and infrastructure dependencies, which increase the costs and complexities of deployment.

"It is important to clarify that the choice between QKD and PQC is not binary. The integration of QKD with PQC can offer a hybrid solution that is based on the strengths of both technologies" [1]. It has sparked interest in a hybrid approach that combines PQC and QKD. Notably, PQC can provide essential authentication for QKD processes, safeguarding against potential man-in-the-middle attacks, which is crucial in maintaining the integrity of QKD's key exchange mechanism. By combining QKD with PQC-based authentication, it becomes possible to achieve a balance between short-term and long-term security: PQC can secure immediate authentication, while QKD enables the generation of long-term secure keys [2]. This integration ensures that even if PQC were compromised in the future, previous authentication and QKD-generated keys would remain unaffected, adding a layer of resilience to cryptographic infrastructure.

Ultimately, the integration of PQC and QKD can adapt to diverse security needs, creating a robust and flexible framework for quantum-safe communications. Although this approach requires investment in both advanced algorithms and QKD-compatible hardware, the combined security benefits make it a promising solution for environments where information integrity and confidentiality are of paramount importance.

We start with an overview of post-quantum cryptography in Section 2. Section 3 discusses the math problems that PQC algorithms rely on. Section 4 discusses QKD algorithms. The literature review of PQC and QKD integration examples and approaches is given in Section 5. We analyse PQC and QKD support by the network equipment in Section 6. We complete with the analyses of potential use case scenarios in Section 7 and the testbed description in Section 8.

2 Overview of PQC Algorithms

PQC develops cryptographic algorithms that are considered secure against potential quantum computer threats while remaining compatible with existing digital communication protocols and infrastructure. Unlike QKD, which also resists quantum attacks but requires specialised and costly equipment, PQC relies on mathematically hard problems for classical and quantum computers. PQC enables a gradual transition to quantum-resistant algorithms without disrupting current systems.

2.1 Why We Need PQC

RSA, one of the earliest and widely used public-key cryptosystems, relies on the prime factorisation problem. Its difficulty lies in finding the product of two large prime numbers. Elliptic-curve cryptography (ECC), another approach, uses the algebraic structure of elliptic curves over finite fields and achieves equivalent

security to RSA with smaller key sizes, making it suitable for key exchange, digital signatures, and pseudorandom generation.

In 1994, Peter Shor introduced a quantum algorithm for factoring integers and solving discrete logarithm problems [3]. The algorithm can break RSA and ECC in polynomial time on a sufficiently large quantum computer. While such computers are not yet available, advancements in quantum technology suggest they might be implemented in 5–10 years. This has enabled "harvest now, decrypt later" attacks, where encrypted network traffic is stored and decrypted once quantum computers are available. Long-term sensitive data, such as government documents or financial records, is already at risk.

2.2 NIST Standardisation

To address quantum threats, NIST initiated a standardisation process in 2016 to identify quantum-resistant cryptographic algorithms. Out of 69 initial submissions, the process, involving four evaluation rounds, categorised algorithms into "Public-Key Encryption and Key-Establishment Algorithms" and "Digital Signature Algorithms." Some algorithms were broken, and others merged, leading to the selection of four algorithms in 2022: CRYSTALS-KYBER (encryption) [4] and three digital signature schemes: CRYSTALS-DILITHIUM [5], FALCON [6], and SPHINCS+ [7].

In August 2024, NIST published three PQC standards:

- **FIPS 203:** Based on CRYSTALS-KYBER for key encapsulation, relying on the Module Learning with Errors problem [8].
- **FIPS 204:** Based on CRYSTALS-DILITHIUM, a digital signature scheme also leveraging Module Learning with Errors [9].
- **FIPS 205:** Based on SPHINCS+, a hash-based signature scheme that combines few-time (FORS) and multi-time (XMSS) signature schemes [10].

FALCON, another digital signature scheme, has been selected but not standardised. Its reliance on floating-point arithmetic raises concerns regarding processor compatibility and resistance to side-channel attacks, such as timing and energy consumption. Additionally, the larger signature sizes of PQC schemes compared to the classical ones may hinder crypto agility, particularly in use cases requiring long signature chains that could exceed protocol limits. Code-based KEM - HQC [11] was selected for standardisation by NIST in March 2025.

3 Math Problems Used in PQC Algorithms

This section discusses key mathematical problems that serve as the foundation for post-quantum cryptographic algorithms, highlighting their principles and applications.

3.1 Hash-Based Cryptography

Hash-based cryptography relies on the collision resistance of cryptographic hash functions to provide secure digital signatures. A prominent example is the Merkle signature scheme [12], which employs binary hash trees to sign messages securely. In this scheme, a root hash serves as the public key, and each message signature reveals a specific path within the tree, proving authenticity without relying on vulnerable mathematical assumptions. The core building block of all hash-based schemes is the concept of one-time signatures introduced by [13]. The NIST standard SLH-DSA [10] integrates other hash-based signature schemes: Forest of random subsets (FORS) [14], the eXtended Merkle Signature Scheme (XMSS) [15] and Winternitz One-Time Signature Plus (WOTS+) [16]. SLH-DSA attacks include side channel and fault attacks [17–21].

Despite its quantum resistance, hash-based cryptography faces limitations such as large signature sizes and the need for careful tracking of used keys in stateful schemes. However, it remains a highly secure and practical alternative, exemplified by its adoption in the NIST SLH-DSA digital signature standard [10].

3.2 Code-Based Cryptography

Code-based cryptography derives its security from the computational hardness of decoding random linear codes, as demonstrated by the McEliece cryptosystem [22]. The public key is a scrambled error-correcting code, while the private key enables efficient decoding. Encryption involves introducing random errors that make decryption without the private key infeasible. Classic McEliece advanced to the final rounds but was not standardized due to key size [23]. However, a severe attack has recently occurred on the McElice cryptosystem [24]. Despite challenges, the simplicity and robustness of code-based cryptography make it a valuable post-quantum candidate. Code-based KEM - HQC [11] was selected for standardization by NIST in March 2025. It is based on Hamming Quasi-Cyclic codes [25].

3.3 Lattice-Based Cryptography

Lattice-based cryptography leverages the structure of n-dimensional lattices, utilizing problems such as the Shortest Vector Problem (SVP) and the Closest Vector Problem (CVP). These problems, especially in their approximate forms, are computationally hard to solve even with quantum computers. The Learning With Errors (LWE) problem [26, 27], a variant involving noisy linear algebra, forms the foundation for many lattice-based schemes.

The inherent hardness of these problems makes lattice-based cryptography highly secure and versatile. For example, the NIST standards ML-KEM [8] and ML-DSA [9] are based on the Modulus Learning with Errors problem. Lattice-based cryptography is also favoured for its efficiency and scalability, with applications in encryption, digital signatures, and key exchanges.

The best-known attacks on LWE and its variants include lattice sieving [28, 29], enumeration [30, 31], basis reduction algorithms like LLL [32] and BKZ [33] and machine learning-based techniques like SALSA [34].

3.4 Multivariate Quadratic Cryptography

Multivariate Quadratic (MQ) cryptography is based on the difficulty of solving systems of multivariate quadratic equations over finite fields, a problem known to be NP-hard. MQ cryptography is particularly robust against quantum and classical attacks due to the lack of efficient algorithms to solve these equations.

MQ cryptography is widely used for digital signatures. In these schemes, the private key allows efficient signature generation, while the public key enables verification by checking specific quadratic equations. However, the Rainbow [35] signature scheme, once a prominent MQ system, was compromised in 2022 [36] due to vulnerabilities that exposed its private keys. Despite this, MQ cryptography remains an active research area, with ongoing efforts to enhance its robustness and utility.

3.5 Isogeny-Based Cryptography

Isogeny-based cryptography leverages the complexity of finding isogenies between elliptic curves. The Supersingular Isogeny Key Encapsulation (SIKE) scheme was a notable candidate in this field but was broken in 2022 [37] when researchers discovered an efficient attack on SIDH.

Despite this setback, isogeny-based cryptography continues to attract interest due to its compact key sizes and bandwidth efficiency. Alternative approaches and modified schemes are being explored to address vulnerabilities, ensuring the field remains a promising avenue for quantum-resistant cryptographic protocols.

4 Overview of QKD Algorithms

Quantum Key Distribution is a secure key agreement protocol that enables two parties, commonly referred to as Alice and Bob, to generate a shared secret key using the principles of quantum mechanics and classical communication. Unlike traditional cryptographic methods, QKD leverages the fundamental laws of physics to ensure that any attempt by a third party (commonly called Eve) to intercept or tamper with the key can be detected. The security of QKD originates from the no-cloning theorem and the probabilistic nature of quantum measurements, which make eavesdropping inherently detectable. This unique property provides perfect forward secrecy, making QKD resistant to future quantum-based threats, including so-called "harvest now, decrypt later" attacks. As a result, QKD offers a robust foundation for long-term secure communication in the era of quantum computing.

The foundational QKD protocol, BB84 [38], was proposed as early as 1984, although a formal security proof was not published until 2000 [39]. Since then, several other notable protocols have been introduced, including B92 [40], SARG04

[41], and the Ekert protocol [42]. These protocols fall into different categories - some are based on the prepare-and-measure paradigm, others leverage quantum entanglement, and some exist in both forms. A crucial requirement for all QKD implementations is an authenticated classical communication channel, without which the system remains vulnerable to man-in-the-middle attacks. Typically, authentication is achieved using a pre-shared key (PSK) between the two communicating parties (Alice and Bob). In this context, QKD functions as a mechanism to amplify or extend the initial PSK, generating a long (potentially unbounded) sequence of secure, shared key bits that are guaranteed to be free from tampering or interception.

Several commercial QKD devices are available on the market. The most mature manufacturers in the market are Toshiba[1] and IDQ[2]. The Toshiba devices use efficient BB84 protocol with decoy states and phase encoding. The newest ID Quantique devices (e.g. Clavis XG QKD System) also use BB84 protocol with decoy states. Some models (e.g. Cerberis XGR QKD System) use the COW (Coherent One-Way) protocol [43].

5 Overview of PQC and QKD Integration

In this chapter, we review and summarise information from the scientific literature and documented case studies on the integration of PQC and QKD. This overview includes theoretical insights and practical experiences, highlighting current challenges, solutions, and advancements in merging these two technologies.

In [44], the potential for integrating PQC and QKD in mobile networks is discussed. To protect communication between user equipment and the base station, PQC is employed. QKD is integrated within the backbone of the communication network to enhance overall security. QKD is utilised to secure communication between base stations and the 6G core. PQC is also used for authenticating devices with a software-defined network controller. On the other hand, authors admit that QKD equipment is expensive and requires dedicated optical lines; therefore, for large networks, creating QKD links between all base stations and 6G core nodes would be very expensive. The work is theoretical and only provides suggestions where QKD and PQC could be used but doesn't discuss implementation and adoption details. It focuses more on different PQC algorithm types and where each of them could be useful in the mobile network. However, some of the claims, at the time of writing, are already outdated, as some of the suggested algorithms are broken and others have been standardised.

In [45], a hybrid protocol Muckle is presented. It integrates various potential key sources: QKD, PQC KEM and Classical KEM. The Muckle relies on the existence of a pre-shared key. A HAKE framework for the security evaluation of such hybrid protocols is also proposed. The work doesn't cover real QKD im-

[1] https://www.toshiba.eu/quantum/

[2] https://www.idquantique.com/quantum-safe-security/products/#quantum_key_distribution

plementations. It abstracts QKD as an array of independent, uniformly random bits available to both parties.

Muckle++ protocol is proposed in [46]. The protocol integrates PQC and QKD if a QKD link is available. If QKD is not available, it combines PQC with classical cryptography. The authors claim that protocol is provably ITS (information-theoretically secure) if QKD is used and provable quantum secure otherwise. The paper does not contain the proofs. Muckle++ augments an earlier presented Muckle protocol [45], by excluding the need to pre-share keys and instead rely on quantum-resistant digital signature schemes for authentication [46]. The authors were able to run the protocol for 20 hours using a physical QKD link (with Toshiba QKD equipment) and FPGA (Field-Programmable Gate Array). However, there were technical problems running it longer due to error correction overload. The authors also used PUF (Physically Unclonable Function) to identify devices. For PQC, the CRYSTALS-Kyber [4] was used as KEM and Falcon [6] as digital signature schema.

Muckle+ [47] for joining PQC and QKD uses KDF (key derivation functions) with security proofs and provides signature authentication. Muckle# is inspired by KEM TLS. It is a modification of TLS where KEMS are used for authentication instead of digital signatures. The reason is that, currently, PQC KEMs are shorter than PQC digital signatures.

In [2], the significance of QKD device authentication is explained. It is proposed to replace pre-shared keys for authentication with PQC algorithms. The Aigis.Sig algorithm is used here. However, the idea is still valid by replacing the PQC algorithm. The stability of PQC authentication was tested with a pair of QKD devices. The fibre length is 40 km, and it was running continuously for 30 h. The PQC program keeps running normally, and the QKD systems continuously generate keys.

In [48], PQC algorithm Aigis-Sig [49] is used for authentication in the Jinan metropolitan QKD network (14 QKD nodes, 5 optical switching nodes). The PQC algorithm replaced pre-shared symmetric keys used previously. The PQC algorithm was integrated into the ARM chip of the QKD device to realise the authentication process. The average key rate and QBER of each connection during 36 days of network operation were analysed. There were no significant performance issues.

According to [50], their encryptors are stated to support PQC and can be integrated with their QKD equipment. Meanwhile, [51] reports that the backbone of the Paris QCI network was implemented using QKD, while relays were secured with PQC.

6 PQC and QKD Support in the Equipment

This section analyses the capabilities of the equipment to support Quantum Key Distribution and Post-Quantum Cryptography in secure communication systems, referencing specific examples.

6.1 PQC and QKD Support in Centauris

Centauris encryptors secure data links between remote sites using symmetric key cryptography with AES-256, which is currently considered quantum-safe when appropriate key lengths are used. For enhanced long-term security, AES-512 may be preferred, but Centauris encryptors do not yet support it.

The encryptors integrate with QKD systems, such as Clavis XG, to regularly update symmetric keys (e.g., every minute). Clavis XG can generate four new keys per second, ensuring frequent updates and quantum-safe keys. If the QKD system becomes unavailable, the last obtained key is reused, and a warning is issued via a light on the device. To extend the secure usage of symmetric keys, AES key modification techniques are employed [52].

Device authentication and management of Centauris encryptors are handled via the CM7 software tool. CM7 authenticates devices using digital signatures, allowing the selection of predefined certificates from a list that includes NIST PQC standards, such as CRYSTALS-DILITHIUM and SPHINCS+. However, compatibility between certificate types, CM7 versions, and encryptor releases varies.

Alternative encryptors, such as the Thales High-Speed Encryptor, offer similar functionality, integrating with QKD devices that conform to the ETSI eQKD v14.01 standard. Thales also includes a PQC Starter Kit that supports built-in PQC algorithms [52].

6.2 PQC and QKD Support in Juniper

Juniper devices do not directly support QKD and/or PQC but can integrate pre-shared keys, namely post-quantum pre-shared keys, into IPSec IKEv2 encryption. These keys may then be periodically replaced via calls to the key management system (KMS). The KMS, on the other hand, may use a QKD network or external PQC systems to acquire and distribute symmetric keys.

An experiment conducted by Juniper Networks, ID Quantique, and Deutsche Telekom tested ETSI's REST API in a multivendor environment. Two Juniper SRX380 firewalls connected via a classical 10GbE link were secured using MACsec (AES-256). The QKD devices used were ID Quantique's Cerberis XG systems, connected via standard single-mode fibre optic links and an IDQ eavesdropping simulator. The experiment confirmed that the API is ready for productized environments, though further enhancements to standards are required [53].

Juniper devices running JUNOS 22.4R1 support quantum-safe IPsec using IETF RFC 8784 and the ETSI QKD014 [54] REST API. This implementation merges classical and quantum-safe key material (e.g., from QKD or PQC) to create quantum-resistant IPsec tunnels. Despite these developments, Juniper is cautious about PQC's long-term security and instead focuses on Distributed Symmetric Key Establishment (DSKE) to achieve quantum-safe communications [53, 55].

6.3 PQC and QKD Support in Cisco

Cisco devices support secure connections through IPsec protocols such as SKIP (1995), IKEv1, and IKEv2. SKIP (1995) is an early protocol designed for stateless key distribution but is now largely obsolete. However, the SKIP (2024) draft integrates QKD systems to generate symmetric encryption keys dynamically. In this workflow, QKD devices generate Key IDs and keys, share these with routers, and enable secure key distribution for IPsec communications. SKIP (2024) aligns with ETSI GS QKD 014 but includes additional features such as an optional entropy source endpoint [56, 57].

IKEv2 is widely adopted for its efficiency and improved security. However, IKEv2 systems relying on Elliptic Curve Diffie-Hellman (ECDH) are vulnerable to quantum attacks if sufficiently powerful quantum computers become available. These vulnerabilities can be mitigated by integrating post-quantum preshared keys (PPKs) obtained through QKD or PQC mechanisms. For example, IKEv2 Key Derivation with PPK, described in [58], requires at least 256 bits of entropy for PPKs to ensure 128 bits of post-quantum security and protection against dictionary attacks. Though the current approach supports only static PPKs, extensions to dynamic PPKs are possible [59].

Cisco devices also support MACsec (IEEE 802.1AE), a Layer 2 protocol that secures traffic on a frame-by-frame basis defending against attacks such as replay and MAC address spoofing. However, MACsec relies on symmetric preshared keys (PSKs), which are not quantum-resistant. Using QKD-generated keys or PQC mechanisms for PSKs could significantly enhance MACsec's quantum safety. Additionally, EAP-TLS can establish mutual authentication using certificates, though it does not provide quantum resistance [60, 61].

In summary, Cisco's IPsec and MACsec implementations are being enhanced to integrate QKD and PQC capabilities, aligning with emerging quantum-safe standards [56, 57, 62].

7 Use Cases Integrating PQC and QKD

We discuss use cases that integrate QKD and PQC technologies to enhance security.

7.1 Connecting Two LANs Using Bump-in-the-Wire Encryptors

This use case connects two local area networks (A and B) using a QKD link and an encrypted classical communication channel. While traffic within the local area networks remains unencrypted, encryptors secure the communication between the networks using symmetric cryptography keys derived from the QKD system. Centauris encryptors (discussed in Section 6.1) are employed for this purpose, with authentication currently performed using ECC, but it is planned to transition to SPHINCS+. The communication scheme is shown in Fig. 1.

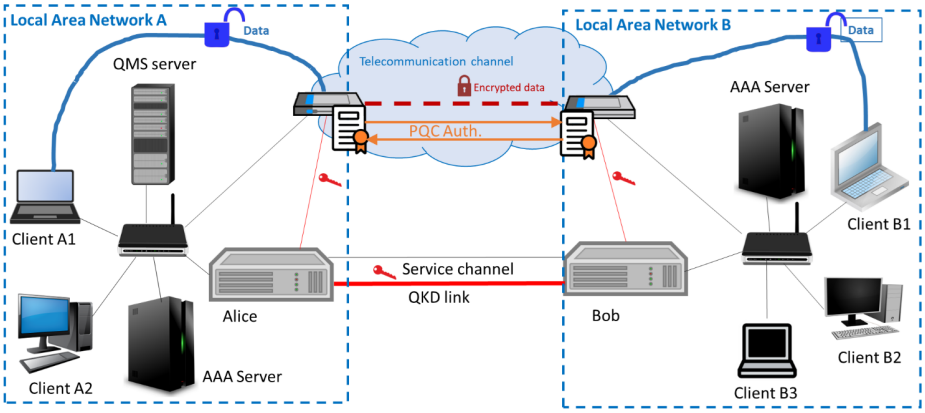

Fig. 1. Schema of the use case QKD as a Service without local area network encryption.

Cisco routers could also replace Centauris encryptors in this use case by employing MACsec for traffic encryption and SKIP to inject QKD keys. Authentication between QKD devices is performed using pre-shared keys and digital certificates. Combining PQC (e.g., SPHINCS+) with a classical algorithm (e.g., ECDH) in certificates provides multi-layer security, ensuring protection even if one algorithm is compromised.

7.2 Integration of QKD into TLS-Based Communication

This modification extends the previous use case by encrypting traffic within the local area networks. In addition to QKD-generated keys used by encryptors and post-quantum cryptography mechanisms for authentication, network users must also ensure mutual identity verification and secure their communications through encryption.

To facilitate this, a Key Management Server (KMS) with shared entropy is introduced. The server acts as a Relying Party (RP), authenticating users via client-server authentication in a TLS v1.3 flow. Certificates signed with PQC algorithms (e.g., SPHINCS+) are negotiated alongside post-quantum KEMs, as explained in [63]. Implementation can utilize repositories like Bouncy Castle [64–66].

The KMS also serves as a key provider, managing QKD-derived and QRNG-generated keys. Symmetric keys are distributed to users who wish to communicate. The scheme is shown in Fig. 2, with PQC-based authentication indicated in orange and KMS-provided keys in green.

7.3 QKD as a Service for External Partners

This use case involves QKD links connecting two local area networks, with external users accessing the networks via VPN servers at the perimeter. The schema

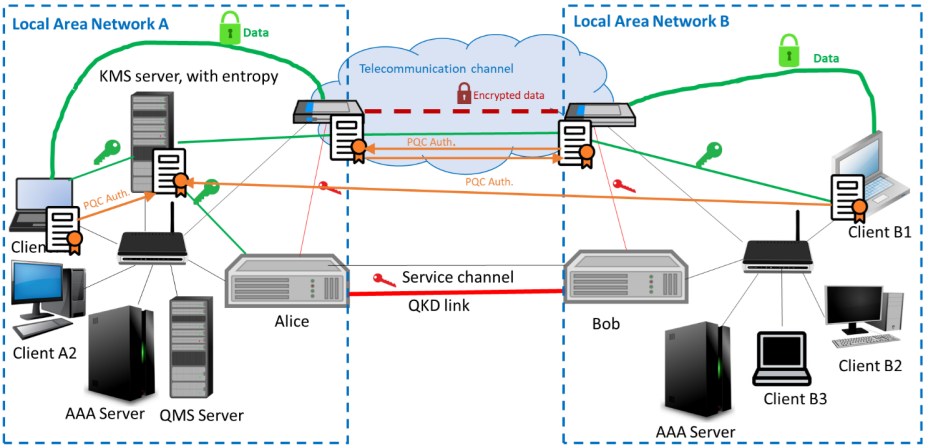

Fig. 2. Schema of the use case QKD as a Service with local area network encryption.

within the networks mirrors the one described in Section 7.2, as shown in Fig. 3.

A critical vulnerability in this scenario is the VPN link between a client and the VPN server. Post-quantum VPNs can mitigate this risk. Microsoft's post-quantum OpenVPN fork [67, 68] integrates PQC and is part of the Open Quantum Safe project's OpenVPN subproject [69]. Other VPNs, such as Mullvad (using Kyber and Classic McEliece) and ExpressVPN (evaluated by Cure53), also offer PQC support [70, 71].

PAN-OS 11.2 Quasar enables quantum-safe hybrid keys for IKEv2 VPNs, implementing RFC 8784 and RFC 9370 [72, 73]. Despite being experimental, the Open Quantum Safe OpenVPN subproject appears to be the most promising candidate for the implementation.

7.4 QKD-Protected Messenger

In this use case, two clients with direct access to a QKD link securely exchange text messages. Keys from the QKD link encrypt messages using OTP as the encryption algorithm. Authentication is essential in the use case. A hybrid signature scheme that combines classical signatures (e.g., ECC) with PQC signatures (ML-DSA or SLH-DSA) is preferred.

This use case leverages the QKD link's direct access for secure communication while ensuring robust authentication through hybrid cryptographic methods.

8 Site-to-Site Quantum-Safe Interconnect Testbed

Our experimental two LAN quantum-safe interconnect testbed integrated QKD devices with post-quantum cryptography. It utilized ID Quantique's QKD hardware – specifically the Clavis and Cerberis models – which implement the BB84

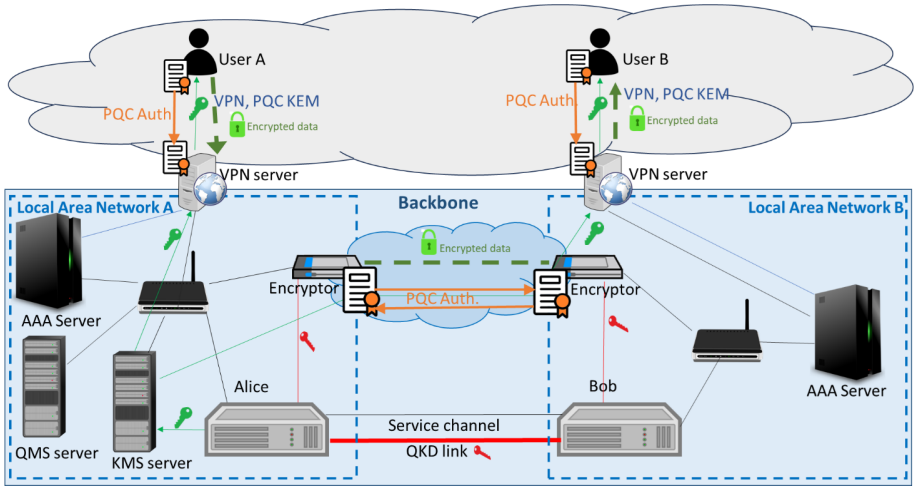

Fig. 3. Schema of the use case QKD as a Service for external partners.

protocol and the COW protocol respectively. The testbed was tested with Cisco NCS 540 routers, Juniper SRX1500 firewalls, and IDQ Centauris bump-in-the-wire encryptors. For brevity, we later briefly detail the deployment with Juniper devices as encryptors.

Both QKD devices rely on a small pre-shared secret for initial mutual authentication; this PSK is used to authenticate their classical communications (e.g. error-correction data) and thus securely bootstrap the quantum key exchange.

The two QKD nodes were interconnected by a dedicated quantum channel (a dark fibre carrying single-photon signals) and a parallel classical service channel. The service channel operated as a 2.5 Gbps bidirectional optical link on standard telecom wavelengths in the C-band (compliant with ITU-T G.694.1 DWDM grid), and was used for synchronization, basis reconciliation, error correction and other auxiliary communications.

To utilize the keys generated by these QKD links, a Key Management System (KMS) layer is necessary, abstracting away vendor-specific details and enabling integration with standard network encryptors. Essentially, the KMS ingests raw keys from the QKD devices (via the devices' proprietary interfaces) and presents them to the network layer through well-defined, interoperable APIs. In our testbed, KMS service is integrated into QKD devices. It provides support for the following two APIs that allow encryptors (referred to as Secure Application Entities, SAE-s) to request QKD keys from Alice and Bob endpoints:

- the ETSI GS QKD 014 API [54], used by the majority of QKD-capable devices; it is a REST-based API, which has a standard since 2019;
- the Cisco SKIP v00 [74], used only by Cisco routers with recent firmware update. The SKIP protocol is currently in draft status. Our IDQ and Cisco

devices support the draft version 00 (as of August 2024); however, the draft version 01 was proposed in March 2025.

For the ETSI interface, mutual TLS was used, requiring each encryptor (client) and the KMS (server) to authenticate with X.509 certificates. In the case of Cisco encryptors that use the SKIP API, the PSK mode was used (where both sides share a pre-established key for authentication). Although the SKIP protocol draft supports certificate-based authentication, it is not yet implemented by the latest Cisco firmware available at the moment.

ETSI and SKIP authentication relies on TLSv1.2 or TLSv1.3, which does not support PQC out-of-the-box. PQC can be added to TLSv1.3 if appropriate codepoints are used, which have not been standardized yet. However, since ETSI and SKIP are used only between a QKD device and an encryptor, which are usually located in the same room, PQC algorithms are generally unnecessary. Still, if an update to TLSv1.3 (or a subsequent version of TLS) will introduce PQC capabilities, that should also be reflected in both ETSI and SKIP protocols.

Juniper SRX1500 firewall devices were chosen for their support of quantum-safe key integration support (the SRX can utilize external keys via the RFC 8784 mechanism). The IPsec tunnels were set with short-lived keys so that new keys would be frequently requested from the KMS. Instead of a continuously running QKD link during testing, a simulated QKD key server nicknamed "SNEK" (Single-node ETSI KMS) was also deployed for observability. SNEK emulated the behaviour of real QKDs to test the encryptors' compliance with the ETSI GS QKD 014 protocol and to observe request/response patterns over time. The experiments showed that the encryptors successfully retrieved keys in sync, applied them to the IPsec SAs (Security Associations), and maintained encrypted traffic flow without packet loss or downtime.

Cisco devices, in their turn, rely on the proprietary MACSec protocol, while Centauris devices use a proprietary TLS-like protocol. Interestingly, Centauris devices can use PQC algorithms for mutual authentication. However, they have to rely on algorithm identifiers and key encodings that are subject to change since not all PQC algorithms have these standards yet.

9 Conclusions

We analysed the state-of-the-art advancements in Post-Quantum Cryptography. Currently, three algorithms have been standardised by NIST: one key encapsulation mechanism (KEM) and two digital signature schemes. The KEM and one of the signature schemes are based on lattice-based cryptography using the Learning With Errors problem, while the second signature scheme relies on hash-based cryptography. Despite the standardisation of these two signature schemes, an ongoing competition seeks to identify additional signature schemes, primarily due to concerns regarding the large signature sizes of the currently standardised schemes. Additionally, some vendors remain hesitant to integrate post-quantum standards into their network solutions, citing insufficient real-world testing and the lack of long-term validation.

In this context, we also investigated the extent to which PQC and QKD are supported by the network equipment we use. Our findings indicate that not all devices natively support PQC or QKD keys. Nevertheless, in most cases, workarounds can be implemented to enable the integration of QKD keys and PQC algorithms with the equipment.

We reviewed scientific literature addressing the integration of PQC and QKD. While several papers discuss experimental implementations of PQC and QKD integration, no widely accepted or standardised approach currently exists for achieving such integration. This highlights the need for further research and development in this domain.

Based on our experience and the literature review, we discussed four use cases. The use cases explicitly define the utilisation of QKD links. Consequently, we focused on identifying where PQC could provide additional value within these scenarios. Three primary areas for the application of PQC were identified:

1. **Device Authentication**: QKD does not inherently provide authentication. Pre-shared keys are typically used for QKD authentication, but this approach can become problematic in larger networks due to scalability issues. PQC offers a robust solution for device authentication in such scenarios.
2. **Quantum-Safe Communication for Remote Users**: PQC can enable secure communication for users who do not have direct access to the local area network. This is typically achieved through PQC-based VPNs, which replace classical VPNs to ensure quantum-safe communication. While several PQC-based VPN solutions are currently available, many remain experimental.
3. **Quantum-Safe IPSec Communication**: Integrating PQC libraries into IPSec implementations enables quantum-safe communication between users. We identified libraries that provide both PQC support and compatibility with IPSec, making them well-suited for adaptation in the targeted use cases.

We have developed a testbed currently supporting some of the proposed use cases. We plan to extend the testbed to support all discussed use cases. Given that many of these solutions incorporate experimental components, modifications or adaptations are likely necessary to implement the planned approaches effectively.

Acknowledgment

The research is supported by the European Union, project No. 101091559 "Development of experimental quantum communication infrastructure in Latvia (LATQN)".

References

1. Regulatory Horizons Council, Regulating Quantum Technology Applications, 28 February (2024).

2. Wang, L.J., Zhang, K.Y., Wang, J.Y., et al.: Experimental authentication of quantum key distribution with post-quantum cryptography. npj Quantum Inf 7, 67, (2021). https://doi.org/10.1038/s41534-021-00400-7
3. Shor, P.W.: Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In: Goldwasser, S. (ed.) Proceedings of the 35th Annual Symposium on Foundations of Computer Science (FOCS), pp. 124–134. IEEE Computer Society Press, Los Alamitos (1994). https://doi.org/10.1109/SFCS.1994.365700
4. Avanzi, R.,, Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehlé, D.: CRYSTALS-Kyber algorithm specifications and supporting documentation, Third-round submission to the NIST's post-quantum cryptography standardization process (2020). https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions
5. Bai, S., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., Stehlé, D.: CRYSTALS-Dilithium: Algorithm specifications and supporting documentation, Version 3.1. (2021). https://pq-crystals.org/dilithium/data/dilithium-specification-round3-20210208.pdf
6. Prest, T., Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU. Submission to the NIST's post-quantum cryptography standardization process, Version 1.2. (2020). https://falcon-sign.info/falcon.pdf
7. Aumasson, J.P., Bernstein, D.J., Beullens, W., Dobraunig, C., Eichlseder, M., Fluhrer, S., Gazdag, S.L., Hülsing, A., Kampanakis, P., Kölbl, S., Lange, T., Lauridsen, M.M., Mendel, F., Niederhagen, R., Rechberger, C., Rijneveld, J., Schwabe, P., Westerbaan, B.: SPHINCS+ – Submission to the NIST post-quantum project, Version 3.1. (2022). https://sphincs.org/data/sphincs+-r3.1-specification.pdf
8. NIST: FIPS 203 Module-Lattice-Based Key-Encapsulation Mechanism Standard. Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8900, August 13 (2024), https://doi.org/10.6028/NIST.FIPS.203
9. NIST: FIPS 204 Module-Lattice-Based Digital Signature Standard. Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8900, August 13 (2024). https://doi.org/10.6028/NIST.FIPS.204
10. NIST: FIPS 205 Stateless Hash-Based Digital Signature Standard. Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8900, August 13 (2024). https://doi.org/10.6028/NIST.FIPS.205
11. Aguilar-Melchor, C., Aragon, N., Bettaieb, S., Bidoux, L., Blazy, O., Bos, J., Deneuville, J.C., Dion, A., Gaborit, G., Lacan, P., Persichetti E., Robert, J.M, Veron, P., Zémor, J.,: Hamming Quasi-Cyclic (HQC) – Submission to the NIST post-quantum project, Fourth round version, Updated version 19/02/2025 (2025) https://pqc-hqc.org/doc/hqc-specification_2025-02-19.pdf
12. Merkle R.C.: Secrecy, Authentication, and Public Key Systems. Ph.D. thesis, Stanford university (1979). http://www.ralphmerkle.com/papers/Thesis1979.pdf
13. Lamport, L.: Constructing digital signatures from a one way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory (1979).

14. Bernstein, D.J., Hülsing, A., Kölbl, S., Niederhagen, R., Rijneveld, J., Schwabe, P.: The SPHINCS+ Signature Framework. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19). pp. 2129–2146, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3319535.3363229
15. Buchmann, J., Dahmen, E., Hülsing, A.: XMSS - A Practical Forward Secure Signature Scheme Based on Minimal Security Assumptions. Yang, B.Y. (eds) Post-Quantum Cryptography. PQCrypto 2011. Lecture Notes in Computer Science, vol 7071, pp 117–129, Springer, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25405-5_8
16. Hülsing A.: W-OTS+ – Shorter Signatures for Hash-Based Signature Schemes. In: Youssef, A., Nitaj, A., Hassanien, A.E. (eds) Progress in Cryptology – AFRICACRYPT 2013. AFRICACRYPT 2013. Lecture Notes in Computer Science, vol 7918, pp. 173–188, Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38553-7_10
17. Kannwischer, M.J., Genêt, A., Butin, D., Krämer, J., Buchmann, J.: Differential Power Analysis of XMSS and SPHINCS. Fan J., Gierlichs B. (eds) Constructive Side-Channel Analysis and Secure Design, pp. 168–188, Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-89641-0_10
18. Castelnovi L., Martinelli A., Prest T.: Grafting Trees: A Fault Attack Against the SPHINCS Framework. Lange, T., Steinwandt, R. (eds) Post-Quantum Cryptography, pp. 165–184, Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-79063-3_8
19. Genêt, A., Kannwischer, M.J., Pelletier, H., McLauchlan, A.: Practical Fault Injection Attacks on SPHINCS, Cryptology ePrint Archive, Paper 2018/674 (2018). https://ia.cr/2018/674
20. Amiet, D., Leuenberger, L., Curiger, A., Zbinden, P.: FPGA-based SPHINCS+ Implementations: Mind the Glitch. 2020 23rd Euromicro Conference on Digital System Design (DSD), Kranj, Slovenia, pp 229–237, IEEE (2020). https://doi.org/10.1109/DSD51259.2020.00046
21. Genêt, A.: On Protecting SPHINCS+ Against Fault Attacks. IACR Transactions on Cryptographic Hardware and Embedded Systems vol. 2023(2) pp. 80–114 (2023). https://doi.org/10.46586/tches.v2023.i2.80-114
22. McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. DSN Progress Report. pp. 42-44 (1978)
23. Niebuhr, R., Meziani, M., Bulygin, S., Buchmann, J.: Selecting parameters for secure McEliece-based cryptosystems. Int. J. Inf. Secur. Vol. 11, pp. 137–147 (2012). https://doi.org/10.1007/s10207-011-0153-2
24. Randriambololona, H.: The syzygy Distinguisher. In: Fehr, S., Fouque, P.A. (eds) Advances in Cryptology – EUROCRYPT 2025. EUROCRYPT 2025. Lecture Notes in Computer Science, vol 15606, pp. 324–354, Springer, Cham (2025). https://doi.org/10.1007/978-3-031-91095-1_12
25. Aguilar-Melchor, C., Blazy, O., Deneuville, J.C., Gaborit, P., Zémor, G.: Efficient Encryption From Random Quasi-Cyclic Codes. In: IEEE Transactions on Information Theory, vol. 64, no. 5, pp. 3927-3943, May (2018). https://doi.org/10.1109/TIT.2018.2804444
26. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Journal of the ACM Vol. 56, No. 6, Article 34 pp.1-40, September (2009). https://doi.org/10.1145/1568318.1568324

27. Lyubashevsky, V., Peikert, C., Regev, O.: On Ideal Lattices and Learning with Errors over Rings. In: Journal of the ACM Vol. 60, No.6, Article 43, pp. 1-35, November (2013). https://doi.org/10.1145/2535925
28. Ajtai, M., Kumar, R., Sivakumar, D.: A Sieve Algorithm for the Shortest Lattice Vector Problem. In: Proceedings of the 33th Annual ACM Symposium on Theory of Computing, STOC 2001, pp. 601–610, ACM (2001). https://doi.org/10.1145/380752.380857
29. Nguyen, P.Q., Vidick, T.: Sieve Algorithms for the Shortest Vector Problem Are Practical. Journal of Mathematical Cryptology 2, 181–207 (2008). https://doi.org/10.1515/JMC.2008.009
30. Schnorr, C.P.: Lattice Basis Reduction: Improved Practical Algorithms and Solving Subset Sum Problems. In: Journal of Mathematical Programming, Vol. 66, pp. 181–191 (1994). https://doi.org/10.1007/BF01581144
31. Gama, N., Nguyen, P., Regev, O.: Lattice Enumeration Using Extreme Pruning. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 257–278, Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_13
32. Lenstra, A., Lenstra, H., Lovasz, L.: Factoring Polynomials with Rational Coefficients. Journal of Mathematische Annalen 261(4), pp. 515–534 (1982). https://doi.org/10.1007/BF01457454
33. Schnorr, C.P.: A Hierarchy of Polynomial Time Lattice Basis Reduction Algorithms. Journal of Theoretical Computer Science Vol. 53, Issue 2-3, ISSN 0304-3975, pp. 201–224 (1987) https://doi.org/10.1016/0304-3975(87)90064-8
34. Wenger, E., Chen, M., Charton, F., Lauter, K.: SALSA: attacking lattice cryptography with transformers. In: Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS '22), Article 2535, pp. 34981–34994, Curran Associates Inc., Red Hook, NY, USA (2022).
35. Ding J., Schmidt, D.: Rainbow, a new multivariable polynomial signature scheme. In: Ioannidis, J., Keromytis, A., Yung, M. (eds) Applied Cryptography and Network Security. ACNS 05, LNCS, Vol. 3531, pp. 164–175, Springer, Heidelberg (2005). https://doi.org/10.1007/11496137_12
36. Beullens, W.: Breaking Rainbow Takes a Weekend on a Laptop. In: Dodis, Y., Shrimpton, T. (eds) Advances in Cryptology – CRYPTO 2022. CRYPTO 2022, Lecture Notes in Computer Science, vol 13508, pp. 464–479, Springer, Cham (2022). https://doi.org/10.1007/978-3-031-15979-4_16
37. Maino, L., Martindale, C., Panny, L., Pope, G., Wesolowski, B.: A Direct Key Recovery Attack on SIDH. In: Hazay, C., Stam, M. (eds) Advances in Cryptology – EUROCRYPT 2023. EUROCRYPT 2023. Lecture Notes in Computer Science, vol 14008. pp. 448-471, Springer, Cham, (2023). https://doi.org/978-3-031-30589-4_16
38. Bennett, C.H., Brassard, G.: Quantum cryptography: Public key distribution and coin tossing. In: Proceedings of IEEE International Conference on Computers, Systems and Signal Processing. pp. 175–179, Bangalore, India, December (1984)
39. Shor, P.W., Preskill, J.: Simple Proof of Security of the BB84 Quantum Key Distribution Protocol. Physical Review Letters, 85(2), pp. 441–444, July (2000). https://doi.org/10.1103/PhysRevLett.85.441
40. Bennett, C.H.: Quantum cryptography using any two nonorthogonal states. Physical Review Letters, 68(21), pp. 3121–3124, May (1992). https://doi.org/10.1103/PhysRevLett.68.3121
41. Scarani, V., Acín, A., Ribordy, G., Gisin, N.: Quantum Cryptography Protocols Robust against Photon Number Splitting Attacks for Weak Laser Pulse

Implementations. Physical Review Letters, 92(5), 057901, Feb (2004). https://doi.org/10.1103/PhysRevLett.92.057901
42. Ekert, A.K.: Quantum cryptography based on Bell's theorem. Physical Review Letters 67(6), pp. 661–663, Aug (1991). https://doi.org/10.1103/PhysRevLett.67.661
43. Stucki, D., Brunner, N., Gisin, N., Scarani, V., Zbinden, H.: Fast and simple one-way quantum key distribution. In: Applied Physics Letters 87 (19): 194108, 7 November (2005). https://doi.org/10.1063/1.2126792
44. Hoque, S., Aydeger, A., Zeydan, E.: Exploring Post Quantum Cryptography with Quantum Key Distribution for Sustainable Mobile Network Architecture Design. Proc. of the 4th Workshop on Performance and Energy Efficiency in Concurrent and Distributed Systems (PECS '24), pp. 9-16, Association for Computing Machinery, NY, USA (2024). https://doi.org/10.1145/3659997.3660033
45. Dowling, B., Hansen, T.B., Paterson, K.G.: Many a mickle makes a muckle: A framework for provably quantum-secure hybrid key exchange. International Conference on Post-Quantum Cryptography, Lecture Notes in Computer Science, vol 12100, pp. 483-502, Springer, Cham (2020). https://doi.org/10.1007/978-3-030-44223-1_26
46. Garms, L., Paraïso, T.K., Hanley, N., Khalid, A., Rafferty, C., Grant, J., Newman, J., Shields, A.J., Cid, C., O'Neill, M.: Experimental Integration of Quantum Key Distribution and Post-Quantum Cryptography in a Hybrid Quantum-Safe Cryptosystem. Adv. Quantum Technol. **7**(4), 2300304 (2024). https://doi.org/10.1002/qute.202300304
47. Bruckner, S., Ramacher, S., Striecks, C.: Muckle+: End-to-End Hybrid Authenticated Key Exchanges. In: Johansson, T., Smith-Tone, D. (eds.) Post-Quantum Cryptography (PQCrypto 2023). LNCS, vol. 14154, pp. 601–633. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-40003-2_22
48. Yang, Y.H., Li, P.Y., Ma, S.Z., Qian, X.C., Zhang, K.Y., Wang, L.J., Zhang, W.L., Zhou, F., Tang, S.B., Wang, J.Y., Yu, Y., Zhang, Q., Pan, J.W.: All Optical Metropolitan Quantum Key Distribution Network with Post-Quantum Cryptography Authentication. Optics Express **29**(16), pp. 25859–25867 (2021). https://doi.org/10.1364/OE.432944
49. Zhang, J., Yu, Y., Fan, S., Zhang, Z., Yang, K.: Tweaking the Asymmetry of Asymmetric-Key Cryptography on Lattices: KEMs and Signatures of Smaller Sizes. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds) Public-Key Cryptography – PKC 2020. PKC 2020. LNCS, vol. 12110, pp. 37–65. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45388-6_2
50. Telsy SpA: Quantum Key Distribution (QKD) and Post-Quantum Cryptography (PQC). https://www.telsy.com/en/quantum-key-distribution-qkd-and-post-quantum-cryptography-pqc/
51. ID Quantique SA: PQC-QKD Hybridization in Orange's Fiber Network. June (2024). https://www.idquantique.com/pqc-qkd-hybridization-in-orange-fiber-network/
52. Ginga, S.: Shielding Your Network: Preparing for a Quantum-Safe Future Now. In: THALES BLOG, July 23 (2024). https://cpl.thalesgroup.com/blog/encryption/preparing-for-a-quantum-safe-future-now
53. Juniper Networks, Inc.: Validation of Quantum Safe MACsec Implementation. July (2022). https://www.juniper.net/content/dam/www/assets/white-papers/us/en/2022/validation-of-quantum-safe-macsec-white-paper.pdf

54. European Telecommunications Standards Institute (ETSI): Quantum Key Distribution (QKD); Protocol and data format of REST-based key delivery API. ETSI GS QKD 014 V1.1.1, February (2019).
55. Aelmans, M.: NIST Finalizes Post-Quantum Encryption Standards. Blog post in Juniper Networks, Inc., August 29 (2024). https://blogs.juniper.net/en-us/industry-solutions-and-trends/nist-finalizes-post-quantum-encryption-standards
56. Singh, R., Hill, C., Kawaguchi, S., Lupo, J.: Secure Key Integration Protocol (SKIP). In: draft-cisco-skip-00, August (2024). https://datatracker.ietf.org/doc/draft-cisco-skip/00/
57. ETSI: Quantum Key Distribution (QKD); Protocol and Data Format of REST-based Key Delivery API. In: ETSI GS QKD 014 V1.1.1 (2019-02), February (2019). https://www.etsi.org/deliver/etsi_gs/QKD/001_099/014/01.01.01_60/gs\textbf_QKD014v010101p.pdf
58. Fluhrer, S., Kampanakis, P., McGrew, D., Smyslov, V.: Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-Quantum Security. In: RFC 8784, Internet Engineering Task Force (IETF), June (2020). https://datatracker.ietf.org/doc/html/rfc8784
59. Cisco Systems, Inc.: Chapter: Configuring Quantum-Safe Encryption Using Postquantum Preshared Keys. In: Security and VPN Configuration Guide, Cisco IOS XE 17.x, January (2021). https://www.cisco.com/c/en/us/td/docs/routers/ios/config/17-x/sec-vpn/b-security-vpn/m-sec-cfg-quantum-encryption-ppk.html
60. Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., Kivinen, T.: Internet Key Exchange Protocol Version 2 (IKEv2). In: STD 79, RFC 7296, October (2014). https://doi.org/10.17487/RFC7296
61. Hoffman, P.: The Transition from Classical to Post-Quantum Cryptography. In: Network Working Group, May 26, (2020). https://datatracker.ietf.org/doc/html/draft-hoffman-c2pq-07
62. Aziz, A., Markson, T., Prafullchandra, H.: Simple Key-Management For Internet Protocols (SKIP). August (1996). https://datatracker.ietf.org/doc/draft-ietf-ipsec-skip/06/
63. Kozlovičs, S., Petručeņa, K., Lāriņš, D., Vīksna, J.: Quantum Key Distribution as a Service and Its Injection into TLS. In: Meng, W., Yan, Z., Piuri, V. (eds) Information Security Practice and Experience (ISPEC 2023), LNCS, vol. 14341, pp. 527–545. Springer, Singapore (2023). https://doi.org/10.1007/978-981-99-7032-2_31
64. IMCS SysLab: Bouncy Castle Fork at IMCS SysLab for Implementing the TLS Injection Mechanism (for Adding PQC/QKD-Related Code to TLS). https://github.com/LUMII-Syslab/tls-injection-mechanism
65. IMCS SysLab: Integration of PQC Algorithms into the BouncyCastle TLS Injection Mechanism. https://github.com/LUMII-Syslab/tls-injection-pqc
66. IMCS SysLab: TLS Client Example with Injected PQC Algorithm. https://github.com/LUMII-Syslab/tls-injection-pqc
67. Microsoft: Post-Quantum Crypto and VPNs. https://www.microsoft.com/en-us/research/project/post-quantum-crypto-vpn
68. Microsoft: Post-Quantum Cryptography VPN. https://github.com/microsoft/PQCrypto-VPN
69. Open Quantum Safe: OQS-Demos / OpenVPN. https://github.com/open-quantum-safe/oqs-demos/tree/main/openvpn

70. Powell, O.: Why Every VPN Should Use Post-Quantum Encryption. Tom's Guide, June 18 (2024). https://www.tomsguide.com/computing/vpns/why-every-vpn-should-use-post-quantum-encryption
71. Tamašiūnas, L.: NordVPN Launches First App with Post-Quantum Encryption Support. NordVPN Blog, September 30 (2024). https://nordvpn.com/blog/nordvpn-linux-post-quantum-encryption-support/
72. Meshi, Y.: PAN-OS 11.2 Quasar Helps Customers Secure Networks Everywhere, Faster. Palo Alto Networks Blog, May 2 (2024). https://www.paloaltonetworks.com/blog/network-security/quasarlaunch/
73. Palo Alto Networks, Inc.: Network Security, Quantum Security Concepts. December 8 (2023). https://docs.paloaltonetworks.com/network-security/quantum-security/administration/quantum-security-concepts
74. Internet Engineering Task Force (IETF): Secure Key Integration Protocol (SKIP). Version: draft-cisco-skip-00. https://datatracker.ietf.org/doc/draft-cisco-skip/00/