Telecommunications Standards Advisory Committee (TSAC)

Reference Specification

Quantum Key Distribution Networks

IMDA RS QKDN
Issue 1, June 2023

Info-communications Media Development Authority
10 Pasir Panjang Road
#03-01 Mapletree Business City
Singapore 117438

Acknowledgement

The Info-communications Media Development Authority (IMDA) and the Telecommunications Standards Advisory Committee (TSAC) would like to acknowledge the following members of the TSAC Focus Area (FA) 7 Quantum Communication Networks Task Force (TF) for their invaluable contributions to the preparation of this Reference Specification:

IMDA RS QKDN, Issue 1, June 2023	Reference Specification for Quantum Key Distribution Networks
Focus Area 7 Chairperson	Mr Lin Yih, Director, Digital Applied Research & Technology
Quantum Communication Networks Task Force Chairperson	Dr Michael Kasper, CEO, Fraunhofer Singapore Research Ltd Dr Hao Qin, Senior Manager (Quantum Communications Technologist), Centre for Quantum Technologies, National University of Singapore (NUS) | National Quantum-Safe Network (NQSN)
Editors	Dr Haw Jing Yan, Senior Manager (Quantum Communications Technologist), Centre for Quantum Technologies, National University of Singapore (NUS) | National Quantum-Safe Network (NQSN)
	Dr Xiao Duan, Research Fellow, Fraunhofer Singapore Research Ltd | National Quantum-Safe Network (NQSN)
Secretary	Ms Pei Wee Kong, Senior Manager, Infocomm Media Development Authority (IMDA)

List of TSAC FA 7 Quantum Communication Networks Task Force Members (2021-2024)

S/N	Organisation	Name
1	ADVA Optical Networking Singapore Pte Ltd	Dr Jim Zou, Global Business Development
2	Amazon Web Services Singapore Pte Ltd	Mr Albert Wang, Industry Solutions Architect
3	Defence Science and Technology Agency (DSTA)	Mr Too Huseh Tien, Deputy Director, Infocomm Digital Platforms
4	Government Technology Agency of Singapore (GovTech)	Mr Tan Peng Chong, Principal Cybersecurity Specialist
5	Infocomm Media Development Authority (IMDA)	Mr Koh Wee Sain, Director, Digital Infrastructure Technologies
6	Infocomm Media Development Authority (IMDA)	Mr Nga Chee Wei, Director, Emerging Technology Office
7	National Supercomputing Centre (NSCC)	Mr Michael Wong, Network Manager, Lead, Information Technology
8	NetLink Trust	Mr Melvin Chan, Director, Engineering Planning
9	S-Fifteen Instruments Pte. Ltd.	Ms Brenda Chng, Managing Director

10	Singapore Technologies Engineering Ltd (ST Engineering)	Mr Kan Pak Leng, Head Cybersecurity, Group Engineering Centre
11	Singapore Technologies Engineering Ltd (ST Engineering)	Dr Paul Waie Shew, Senior Cyber Product & Solution Architect
12	Singapore Telecommunications Limited (SingTel)	Mr Chinnadurai Thangiah, Lead Cybersecurity Specialist
13	ST Telemedia Global Data Centres (STT GDC)	Mr Yeo Teong Chuan, Senior Director, Technology, Project, Operations and Environment Department
14	SpeQtral Pte Ltd	Dr Robert Bedington, CTO
15	SPTel Pte Ltd	Ms Wang Kangni, Assistant Director, Business Development
16	Terrabit Networks Pte Ltd	Mr Sebastian Mock, Business Development
17	Toshiba Asia Pacific Pte Ltd	Mr Anandaraman Sankaran, Senior Manager, QKD Technical Marketing
18	UL Verification Services Pte Ltd	Mr Gavin Duan, Lead Technical Analyst
19	Utimaco IS Pte Ltd	Ms Michaela Klopstra, Regulation and Standardization Manager
20	Utimaco IS Pte Ltd	Dr Volker Krummel, Technical Manager, Cryptography

Telecommunications Standards Advisory Committee (TSAC)

The TSAC advises IMDA on the setting of ICT standards as well as on the development and recommendation of specifications, standards, information, guidelines and other forms of documentation for adoption and advancement of the standardisation effort of the Singapore ICT industry (hereafter termed "IMDA Standards").

Telecommunications standards-setting in Singapore is achieved with the assistance of TSAC, where professional, trade and consumer interest in telecommunications standards is represented on the TSAC with representatives from network and service operators, equipment suppliers and manufacturers, academia and researchers, professional bodies and other government agencies.

List of TSAC Members (2021-2024)

TSAC Chairman:

Dr Chin Woon Hau	Director (Standard Development & Regulatory Technology) Infocomm Media Development Authority

TSAC Members:

Mr Yip Yew Seng	Honorary Secretary Association of Telecommunications Industry of Singapore (ATIS)
Mr Adrian Chang	Director, Futures and Information Technology/ Chief Information Officer Civil Aviation Authority of Singapore (CAAS)
Mr Lim Wee Seng	Director, Energy Management System/Power System Operation Division Energy Market Authority (EMA)
Mr Kok Yixiong	Deputy Director, IT, Electrical & Infrastructure Enterprise Singapore (ESG)
Mr Mark Tan	Deputy Director, Security Engineering & Operations Section Housing Development Board (HDB)
Mr Andy Phang	Assistant Director, Infocomm Resource & Technology Infocomm Media Development Authority (IMDA)
Mr Marcus Tan Cheng Lin	Head of Cybersecurity Department Institute for Infocomm Research (I2R)
Mr Peter Quek	Group Director - IT, Cybersecurity & Digital Services Land Transport Authority (LTA)
Mr Denis Seek	CTO M1 Limited
Mr Dennis Khoo	Director, Port Systems Division Maritime and Port Authority of Singapore (MPA)
Mr Anil Nihalani	Head, Digital Products & Technology Mediacorp Pte Ltd
Dr Teh Kah Chan	Associate Professor, School of EEE Nanyang Technological University (NTU)

Dr Biplab Sikdar	Associate Professor, Department of ECE National University of Singapore (NUS)
Mr Kenneth Loh	RF Manager Simba Telecom Pte. Ltd.
Dr Forest Tan	Associate Professor, InfoComm Technology Cluster Singapore Institute of Technology (SIT)
Ms Lousia Lim	Head, Mobile Network Strategy and Access Engineering Singapore Telecommunications Ltd (Singtel)
Mr Lee Yeu Ching	Vice President, Fixed & TV Networks StarHub Ltd

This page is intentionally left blank.

Contents

This Reference Specification is a living document which is subject to review and revision.

Reference Specifications and Guides are informative documents and are not used for approval of customer equipment. They are either one of the following types of documents:

Informative and interim documents on customer equipment standards which are yet to be adopted by network operators; or

Informative documents describing network standards adopted by the public telecommunication networks in Singapore.

NOTICE

The Info-communications Media Development Authority (“IMDA”) makes no representation or warranty of any kind with regard to the material provided herein and excludes any express or implied warranties or conditions of non-infringement, merchantability, satisfactory quality and fitness for a particular purpose. Subject to the maximum extent permitted under law, IMDA shall not be responsible or liable to you or any third party for any errors and/or omissions contained herein or for any losses or damages (including any loss of profits, business, goodwill or reputation, and/or any special, incidental or consequential damages) in connection with the use of this material.

IMDA reserves the right to change, modify or add to any part of this document. Nothing herein is intended to create or impose any binding legal obligations or liability whatsoever on IMDA, whether expressed or implied, and whether contractual or otherwise. Without prejudice to the foregoing, nothing in this document shall bind IMDA to adopt any particular course of action. Consequently, nothing herein shall be construed as granting any expectation, whether procedural or substantive in nature, that IMDA will take or not take any particular course of action in the future, arising from or due to anything in this document or in the exercise of its discretion as a public authority.

IMDA draws attention to the possibility that ANY practice or implementation of this Standard/Specification may involve the use of Intellectual Property Rights and takes no position concerning the existence, validity and/or applicability of any such Intellectual Property Rights, whether asserted by TSAC members or any third party.

As of the date of issuance of this Standard/Specification, IMDA has not received written notice of any patent rights which may be relevant in relation to the implementation of this Standard/Specification. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to check with the relevant database in ITU, ISO, IEC or the relevant Standards Development Organisation for information of intellectual property rights. Implementers are advised to obtain their own professional, technical and/or legal advice and conduct all necessary due diligence, including but not limited to making such investigations or seeking clarifications as may be appropriate, in regard to any decision or action that they intend to take, or prior to the implementation of any Standard/Specification as may be required.

Reference Specification for Quantum Key Distribution Networks

1. Scope

The next decade will bring new possibilities and novel application services as quantum technologies get ready for the mainstream and become part of the future networking landscape. The ability to communicate securely is more important than ever to Singapore's society and industry. The integration of quantum communication technology into classical network infrastructures leads to new challenges in governance, deployment, and security. The realisation of quantum-safe networks needs to be built on a solid foundation, including standards and guidelines. The TSAC aims to support this effort with a first local reference specification for quantum key distribution networks (QKDNs).

This reference specification provides a general fundamental guideline for QKDNs deployment. It defines the general framework and functional architecture of QKDN. It also specifies the main functions and related specifications in the different layers of QKDNs, such as quantum layer, key management layer, QKDN control and management layer and service layer. While the defining characteristic of QKDN lies in its ability to provide highly secured communications, the security aspect of QKDNs (e.g., theoretical security, implementation security, security certification, protection profile, etc.) will be discussed separately from this document.

2. Abbreviations

This Reference Specification uses the following abbreviations:

AES	Advanced Encryption Standard
API	Application Programming Interface
ASE	Amplified Spontaneous Emission
BSS	Business Support System
CV	Continuous-variable
CWDM	Coarse Wavelength Division Multiplexing
DI	Device Independent
DV	Discrete-variable
DWDM	Dense Wavelength Division Multiplexing
EB	Entanglement Based
ECC	Elliptic Curve Cryptography
ECU	Electronic Control Units
ETSI	European Telecommunications Standards Institute
FCAPS	Fault, Configuration, Accounting, Performance, Security
gRPC	Google Remote Procedure Call
HSM	Hardware Security Module
HTTPS	Hypertext Transfer Protocol Secure
ICT	Information and Communications Technology
IP	Internet Protocol
ITS	Information-Theoretic Security
ISO	International Organization for Standardisation
JSON	JavaScript Object Notation
KM	Key Manager
KMA	Key Management Agent

KMIP	Key Management Interoperability Protocol
KMLM	Key Management Layer Management
KSA	Key Supply Agent
LEO	Low Earth Orbit
MAC	Message Authentication Code
MDI	Measurement Device Independent
NETCONF	Network Configuration Protocol
OGS	Optical Ground Station
OSS	Operations Support System
OTP	One-Time Pad
P&M	Prepare-and-Measure
PCIe	Peripheral Component Interconnect Express
PEP	Policy Enforcement Point
POS	Point-of-Sale
PQC	Post-Quantum Cryptography
PTP	Point-to-Point
QBER	Quantum Bit Error Rate
QCLM	QKDN Control Layer Management
QKD	Quantum Key Distribution
QKDN	QKD Network
QKD-Tx	QKD Transmitter
QKD-Rx	QKD Receiver
QLM	Quantum Layer Management
QoS	Quality of Service
QRNG	Quantum Random Number Generator
REST	Representational State Transfer
RSA	Rivest–Shamir–Adleman
SCION	Scalability Control and Isolation On Next-Generation Networks
SDN	Software Defined Networking
SFP	Small Form-factor Pluggable
SKIP	Secure Key Integration Protocol
SMF	Single Mode Fibre
SNMP	Simple Network Management Protocol
SPD	Single Photon Detector
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TSN	Time Sensitive Networking
VPN	Virtual Private Network
WDM	Wavelength Division Multiplexing
XLMO	Cross Layer Management Orchestration
XOR	Exclusive Or

3. Introduction

Quantum Key Distribution (QKD) is one of the most mature technologies stemming from quantum information science. QKD has already been made commercially available and has been deployed in test and production environments worldwide. A QKD protocol allows two remote parties to establish symmetric random bit strings as a secret key. Such QKD protocol process can be proven to be secure, even against an eavesdropper with unbounded computational resources under some assumptions supporting the security proof model. This type of security is known as the information-theoretic security (ITS), derived from the theory of quantum mechanics and quantum information. In principle, any eavesdropping attempts will unavoidably introduce quantum disturbances and will be detected by QKD users.

A QKD protocol is implemented on QKD modules, which consist of basic elements such as QKD transmitter(s) (QKD-Tx) and QKD receiver(s) (QKD-Rx). QKD-Tx and QKD-Rx are connected via a QKD link, which usually consists of a quantum channel and a classical channel. The quantum channel is a point-to-point (PTP) channel that transmits quantum signals such as single photons and weak coherent states of light. The classical channel is a traditional communication channel that is used to exchange digital data. The shared secure QKD-key is established through the quantum communication stage (via the quantum channel) and followed by the post-processing stage (via the classical channel). As in ITU-T Y.3800 [1], Figure 1 shows an example of securing a PTP communication link with QKD-keys. The QKD modules establish a pair of symmetric keys via the QKD link and supply them to the application user. It allows the data in transmission over the application link to be encrypted and decrypted with the QKD-keys.

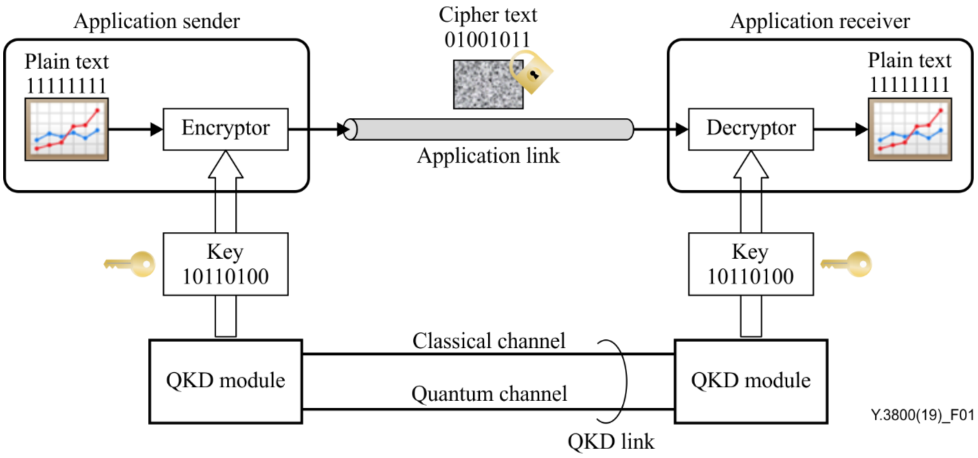

Figure 1: A typical PTP QKD application, as in ITU-T Y.3800 [1].

Due to the unavoidable loss in the quantum channel and security features of a QKD protocol, the range of a PTP QKD link is limited to typically 100 - 200 kilometres in a deployed fibre in practice. Free space or satellite-based QKD can achieve much longer distances (e.g., Chinese Micius satellite QKD demonstrated up to 1200 kilometres as in ITU-T TR FG QIT4N D2.5 [10]) owing to the low transmission loss in atmospheric or almost vacuum channels. There are several approaches to extend a PTP QKD link such as the trusted relay node, measurement-assisted relaying, and quantum repeaters. In relative comparison, the trusted relay node is mature enough to be implemented in practice, which is also the focus of this reference specification. Meanwhile, measurement-assisted relaying has been demonstrated and the quantum repeater is still in the early research stage as in ITU-T TR FG QIT4N D2.5 [10]. By using optical splitters/optical switches, PTP QKD links can be extended to multipoint QKD network (QKDN). The QKDN circumvents the distance limitations of PTP QKD links and enables more than two parties to share secure key materials, leading to the application of various cryptographic primitives.

This reference specification covers the general framework of QKDN and its main functions in different layers, including the quantum layer, the key management layer, the QKDN control layer, the QKDN management layer and the service layer.

4. QKDN Framework and Functions

4.1 QKDN Framework

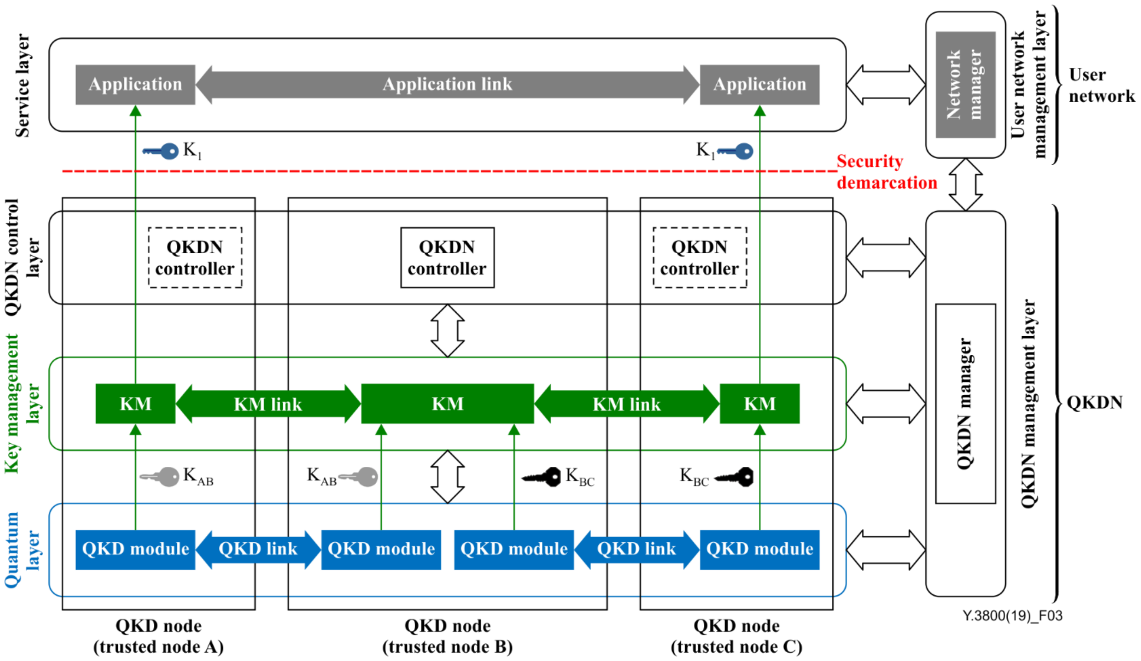

Figure 2: Conceptual structure of a QKDN as in ITU-T Y.3800 [1].

The conceptual structure and main functions of a QKDN are described in ITU-T Y.3800 [1]. As shown in Figure 2, a QKDN typically consists of five logical layers: a quantum layer, a key management layer, a QKDN control layer, a QKDN management layer and a service layer for applications. Vertically, the relevant entities of each layer are physically installed inside a QKD node. The QKD node is required to be secure against intrusion and attacks by any attackers, which is considered as a trusted node.

In a typical QKDN scenario, a pair of QKD modules located in two QKD nodes are connected by a QKD link in the quantum layer. QKD pairs and links can be further concatenated, or mesh-connected via QKD nodes. Upon the completion of QKD process through the QKD links, QKD modules push the generated QKD keys to the key manager (KM) that is located in the same QKD node and connected via KM links to other KMs. The KMs provide key materials to applications in the service layer. Various use cases and applications can be realised using keys from the QKDN in the user network.

In the key management layer, the KM has several key management functions, such as the QKD keys request, key supply, key relay capability, key process, storage etc. The key management layer plays an essential role to interconnect different layers in the QKDN, which enables the interoperability and scalability of the QKDN. QKD modules and KMs are often controlled by the QKDN controller(s) in the QKDN control layer. As shown in Figure 2, there can be a single QKDN controller communicating with all QKD nodes. Alternatively, there can also be a dedicated QKDN controller in each QKD node. The QKDN manager usually monitors the status of different layers and manages the QKDN as a whole in the QKDN management layer. In terms of QKDN design and deployment, there are several aspects needed to be considered: security, scalability, stability, efficiency, robustness, interoperability, ability of integration/migration, manageability, and application.

NOTE 1 - Each layer is a logical function and not necessarily a separate physical entity/layer. For example, a KM can be integrated with QKD modules; QKDN control and management layers can be implemented by software on a server.

NOTE 2 – PTP QKD configuration is also aligned with the QKDN in Figure 2, but certain functions are simplified: the key relay function in the KM is not needed; QKDN control and management layers mainly control and manage the QKD modules.

4.2 QKDN Layer Elements and Functions

This clause briefly summarises the layer elements and functions as shown in Figure 2 (Figure 3 in ITU-T Y.3800 [1]), including the quantum layer, the key management layer, the QKDN control and management layer, and the service layer.

4.2.1 Quantum Layer

This layer consists of QKD modules and QKD links. Connected via the QKD links, the QKD modules produce identical and uniformly random bit strings known as QKD-keys based on the implemented QKD protocol. The QKD modules then output the generated QKD-keys into a KM within the same QKD node. Other QKD link parameters, such as the quantum bit error rate (QBER) and QKD key generation rates, are provided to the QKDN manager. The pairwise linked QKD modules are joined together via QKD nodes.

4.2.2 Key Management Layer

This layer consists of KMs and KM links. The main role of a KM in each QKD node is to manage key materials between and inside QKD nodes. Upon receiving random bit strings from the QKD module within the same QKD node, the KMs synchronise and re-format these bit strings via KM links in between. The resulting outputs are called keys and are stored for later usage. A cryptographic application can perform a key request from the KM via a cryptographic application interface installed in the KM. The KM processes this request by serving the acquired number of keys from the storage. This key is further synchronised and authenticated with the corresponding KM via a KM link, which is then provided to the cryptographic application with the desired format.

4.2.3 QKDN Control Layer

This layer consists of QKDN controllers with links connected to other layers. In this layer, the QKDN controllers can be distributed in each QKD node as shown in Figure 2, or one centralised QKDN that locates in one of the QKDN nodes. The QKDN control functions are handled by the QKDN controller(s). Several functions handled by these controllers include the key relay routing, QKD services sessions, authentication and authorization, QKD links and KM links control, Quality of Service (QoS) and charging policy control.

4.2.4 QKDN Management Layer

This layer consists of QKDN managers with links connected to other layers. The role of a QKD manager is to oversee and regulate the QKDN operations. Generally, the aspects of monitoring and management are fault, performance, security, configuration and accounting. The status of the key management layer and the quantum layer are observed by collecting their operational information.

4.2.5 Service Layer

This layer contains user-defined applications. By supplying keys provided by the QKDN to cryptographic applications, users can communicate securely through the application link.

5. Quantum Layer: QKD Modules and Links

In the quantum layer, QKD protocols are implemented on the QKD modules, which are connected via QKD links. The main objective of this layer is for the pairs of QKD modules to establish symmetric QKD keys.

5.1 QKD Protocol

QKD protocols give instructions to QKD modules on how to interact and output QKD keys. The so-called security proofs and ITS are only applicable to the QKD protocols instead of the actual QKD module implementations. Factors such as implementation security and side channels need to be considered on QKD modules as well. The first QKD protocol was invented by Charles Bennet and Gilles Brassard in 1984, which is known as BB84 as in ITU-T TR FG QIT4N D2.3.1 [7]. Since then, many QKD protocols have been proposed, studied and demonstrated, with some being commercialised and deployed in QKDNs. Although specific QKD protocols vary in detailed steps, they all follow similar patterns in the workflow. In this clause, the general QKD protocol framework is presented, and it follows the cryptographic jargon of the actors i.e., Alice who refers to the QKD-Tx, Bob who refers to the QKD-Rx and Eve refers to the attacker/eavesdropper.

As in ITU-T TR FG QIT4N D2.3.1 [7], two main stages are involved in a QKD protocol: the raw key exchange stage (also known as the quantum communication stage) and the classical post-processing stage. The raw key exchange is carried out via the quantum channel, while the classical post-processing is performed via the classical (authenticated) channel.

5.1.1 Quantum Communication Stage

The steps involved in the raw key exchange stage are dependent on the QKD protocol used. Broadly speaking, QKD protocols can be divided into three categories based on the nature of the raw key exchange stage: prepare-and-measure (P&M), measurement device assisting and entanglement-based (EB) schemes. The raw key exchange under a prepare-and-measure (P&M) scheme is as follows (shown in Figure 3):

Step 1: Alice encodes a classical random variable ***a*** on a set of non-orthogonal quantum states.

Step 2: Alice then sends these quantum states via a communication channel, i.e., the quantum channel to Bob.

Step 3: Bob measures the quantum states at the output of the quantum channel, to obtain a classical random variable ***b*** which is partially correlated with Alice's random variable ***a***.

Step 4: By repeating steps 1-3, Alice and Bob exchange a significant number of quantum states and generate two sets of partially correlated data on each side. These two sets of data are called the *raw key* (or *raw data*).

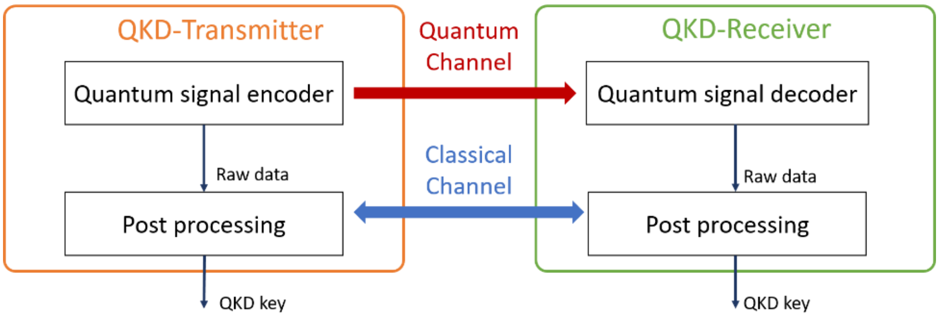

Figure 3: Structure of prepare-&-measure QKD protocol.

For entanglement-based (EB) schemes, the transmitter is distinct from Alice, and consists of a source which produces entangled particle pairs (such as photons). The quantum states of the entangled pairs are realisations of a random variable. One particle in the pair is sent to Alice, while the other is sent to Bob. Both parties then measure the quantum states of the received particles to obtain random variables which are correlated.

In the measurement device assisting schemes, two QKD-Tx send quantum states simultaneously to one intermediate QKD-Rx to perform Bell state measurements.

5.1.2 Classical Post-Processing

Alice and Bob proceed to the classical post-processing stage after the raw key exchange stage. The raw key (partially correlated and partially secure) is processed by Alice and Bob via exchanging information over a classical channel. This stage includes the following steps:

Step 1*: *Sifting: Alice and Bob exchange classical messages to indicate which orthogonal subsets of ***a*** have been used for the encoding (typically basis or quadrature) and the measurement in the raw key exchange stage. The two parties then discard part of the raw key for which encoding and measurement basis are inconsistent. The remaining data they keep is called the *sifted keys* (or *sifted data*).

Step 2: Parameter estimation: In this step, different parameters from the quantum channel, such as QBER, excess noise, and channel transmission, are evaluated. QBER is the ratio of mismatch between Alice's and Bob's sifted key bit strings. These parameters are estimated based on the statistics from a subset of the sifted keys from Alice and Bob. This procedure further allows Alice and Bob to compute the upper limit of the information accessible to Eve from the mutual information between their sifted keys. Concretely, for a given assumption on the attacking strategy, whenever Eve's information exceeds the mutual information between Alice and Bob, which means no secret QKD keys can be established and thus they will abort the QKD protocol. If they have more mutual information compared to Eve, they carry on with the next steps to distil the QKD keys.

Step 3: Error correction (information reconciliation): In this step, Alice and Bob use classical error correction techniques to transform the sifted keys (partially correlated bit strings) into identical bit strings. Depending on the directionality of the information reconciliation, there is either direct reconciliation, where Bob corrects his key to match Alice's data, or reverse reconciliation, where Alice corrects her key to match Bob's data. However, information may be leaked to Eve in all preceding steps (including this step) and during the quantum communication stage in the quantum channel, which needs to be addressed in the next step. The output data of this step is known as the *corrected keys* (or *corrected data*).

Step 4*: *Privacy amplification: In this step, Alice and Bob erase the information that Eve may have from the fully correlated corrected keys from the previous step. Based on the maximum amount of information that Eve can have (computed from the step 2), a portion of the shared keys need to be removed to ensure the secrecy. To this end, public, randomly selected two-universal hash functions are utilised. Finally, Alice and Bob share a pair of identical and secret bit strings, which is called *QKD-key* (or *final key*) and secure up to a negligible ε failure probability (when the keys are not identical, random or secure, e.g. $\varepsilon=10^{-8}$).

Some protocols feature additional steps such as pre-processing, advantage distillation, post-selection etc. However, these additional steps can be typically subsumed in the four general steps outlined above.

5.2 QKD Transmitter and Receiver

QKD protocols are implemented on QKD modules which usually consist of QKD-Tx(s) and Rx(s). Depending on the different protocols (mainly in the raw key exchange stage), the required technologies for the QKD transmitter and receiver also vary. The QKD protocols and their implementations can thus be classified into categories from different perspectives.

Depending on the sending and measurement settings in the raw key exchange stage (as described in *clause 5.1.1*), there are prepare-and-measure (P&M), measurement device assisting and entanglement-based (EB) schemes. In P&M QKD, one QKD-Tx sends encoded quantum states to one QKD-Rx, which measures the quantum states to decode them into the raw key. In the measurement device assisting scheme, two QKD-Txs send quantum states simultaneously to one QKD-Rx to perform a Bell state measurement jointly on the received states. In the EB scheme, an entangled particle pair source simultaneously sends one particle to one QKD-Rx locally and the other one to another QKD-Rx via the quantum channel, so that two QKD-Rxs measure the quantum states of the received quantum signals to establish correlated variables.

NOTE 3 - There is no entanglement source involved in the P&M scheme.

Depending on whether the QKD-Tx and Rx are trusted or not, there are device-dependent QKD or device-independent (DI) QKD and semi-DI QKD including Measurement-device-independent (MDI) QKD and source DI QKD.

NOTE 4 - The so-called “device independent” concept only relates to the security aspects, such that the compromise of the QKD-Tx and/or QKD-Rx will not affect the security of QKD-key generation. However, the performance of QKD (secret key rate) still highly depends on the modules of QKD-Tx and/or Rx even if they are called “device independent”.

Depending on the direction of quantum communication performed by QKD-Txs and QKD-Rxs, there are two-way QKD for bi-directional quantum states exchange and one-way QKD when quantum states are sent from one to another.

Depending on the different encoding in QKD-Txs and decoding in QKD-Rxs, there are discrete-variable (DV) QKD and continuous-variable (CV) QKD. These are the most often mentioned two categories of QKD transmitters and receivers:

- In DV QKD schemes, the sender typically encodes information with discrete variables of finite dimension such as phase, polarization or time bin of single photons and the receiver uses single photon detectors (SPDs) to decode information. Some examples of DV QKD schemes include (decoy state) BB84 protocol, E91 protocol, B92 protocol, six-state protocol, BBM92 protocol, coherent-one way protocol, DPS protocol, Twin-Field protocol QKD, DV-MDI protocol and DI-QKD protocol as in ITU-T TR FG QIT4N D2.3.1 [7].

- In CV QKD schemes, the sender typically encodes information using the position and momentum quadrature of a quantized electromagnetic field in an infinite dimensional Hilbert space. The receiver then uses the coherent detection such as homodyne or heterodyne detection to decode information. Some examples of CV QKD schemes include Gaussian-modulation-based CV protocol, discrete-modulation-based CV protocol and CV-MDI protocol as in ITU-T TR FG QIT4N D2.3.1 [7].

The specifications and related parameters for DV and CV QKD Tx/Rx are referred to in ETSI GS QKD 003 [20] and ITU-T TR FG QIT4N D2.4 [9].

5.3 QKD Links

Typically, two QKD modules are connected via a QKD link, which consists of two logical channels: a quantum channel for the quantum communication stage and a classical channel for the post-processing stage. Besides these two channels, an additional synchronisation channel is used to synchronise and reference the quantum signals in the quantum channel between QKD-Tx and QKD-Rx.

5.3.1 Quantum Channel

Quantum channel is the medium for the quantum signals (one or few photons) to be transferred from QKD-Tx to QKD-Rx in the quantum communication stage. It is in fact an optical communication channel, either a fibre channel or a free space channel.

The quantum channel is assumed to be open and unprotected. In principle, an eavesdropper can perform any actions on the quantum channel that are allowed by the quantum physics, while the security of QKD can be still guaranteed. The realisation of a quantum channel is typically in the form of a given wavelength of a fibre link or an open free space.

NOTE 5 - No active device (e.g., optical amplifier) is allowed in the quantum channel as it will destroy the quantum state, while passive devices such as optical switches are allowed but will introduce additional losses.

Optical loss of the quantum channel is one of the main factors impacting the QKD performance, while such loss is usually associated with the distance, via the attenuation coefficient (a typical value in QKD literature for optical fibre is: 0.2dB/km). This is the so-called distance limitation for QKD, which mainly refers to the loss of the quantum channel. Since the optical loss is associated with the wavelength of the quantum signal light, it is critical to select a proper wavelength of QKD-Tx to be compatible with the quantum channel. Optical loss or attenuation of the quantum channel is the main parameter that needs to be determined in its deployment.

Channel loss and quantum signal wavelength are the main factors to be considered in the deployment of the quantum layer of a QKDN.

5.3.1.1 Fibre Channel

Single mode fibre (SMF) is usually considered for the deployment of a PTP quantum channel, while multimode fibre is not common to be used. The optical loss or attenuation of the fibre is specified in fibre standards, while the optical fibre cables used in Singapore are typically ITU-T G.652D [11], ITU-T G.655 [12] and ITU-T G.657A [13].

The loss of a PTP fibre link can be estimated as in the equation (I-1) in ITU-T G.652 [11]: $A=\alpha L+\alpha_s x+\alpha_c y$, in which α is the attenuation coefficient of the fibre cables in a link, α_s is the splice loss, x is the number of splices in a link, α_c is the loss of line connectors, y is the number of line connectors in a link, L is the fibre link length. Thus, these losses in a fibre link can be summarised as follows:

- Optical fibre cable losses αL: As per ITU-T G.652D [11] and ITU-T G.657A [13] the maximum value of the attenuation coefficient α for 1310 nm - 1625 nm is 0.4dB/km, and for 1530 nm -1565 nm is 0.3dB/km. The fibre length L is usually longer than the physical distance due to fibre routing.
- Splicing losses α_sx: fusion splicing is needed to interconnect the fibre cables for a certain distance with an additional loss for each fusion splice, the splice loss α_s is typically up to 0.1 dB;
- Patching loss α_cy: Typical value of connector loss α_c is up to 0.3 dB per patch.

As an illustration, for a 45 km SMF fibre link with 10 splices and 2 connector pairs at 1310 nm, the total loss (taking maximum values) can be summarised as follows: Fibre loss 45 km x 0.4 dB/km = 18 dB; Splicing losses of 10 splices x 0.1 dB/splice = 1 dB; Patching losses at the two endpoints 2 patches x 0.3 dB/patch = 0.6 dB; thus reaching a total loss of 19.6 dB.

NOTE 6 - In an actual deployment, the actual fibre loss needs to be verified with the respective fibre provider/operator. During the fibre losses resource planning, it is also good to have a sufficient loss budget as a safety margin.

5.3.1.2 Free Space Channel

Compared to the fibre channel, the free-space channel has some advantages as in ITU-T TR FG QIT4N D2.4 [9]:

- Atmospheric transmission windows 780–850 and 1520–1600 nm have a loss with an attenuation coefficient of less than 0.1 dB/km in clear weather. The attenuation is even negligible in the outer space above the earth's atmosphere. This condition allows QKD to be performed over much larger distances, such as between a satellite in Low Earth Orbit (LEO) and the ground. The Micius satellite has demonstrated satellite-to-ground QKD over a distance of up to 1200 km, with a key rate up to 20 orders of magnitude more efficient than what is expected using an optical fibre of the same length as in ITU-T TR FG QIT4N D2.2 [6] and ITU-T TR FG QIT4N D2.5 [10].
- In practice, the de-coherence of polarization or of any other degree of freedom is negligible.

However, there are also some disadvantages as in ITU-T TR FG QIT4N D2.4 [9] and ITU-T TR FG QIT4N D2.5 [10]:

- The free space loss is heavily affected by the weather conditions;
- Alignment, movements and atmospheric turbulence will reduce the effective apertures of the sending and receiving telescopes, which further increases the coupling losses;
- The free space channel requires a direct line of sight and usually are only effective at night time since daylight can cause significant noise photons.

For a free-space channel, the sources of loss include:

- Transmission losses due to atmospheric attenuation;

- Geometric losses due to beam diffraction and pointing error in the link between the QKD-Tx and QKD-Rx optical terminals;
- Clipping losses, which increase the beam divergence.

5.3.2 Classical Post-processing Channel

Classical post-processing channel is used for the post-processing stage to output the final QKD keys. To do that, this channel will need to exchange data between the two parties. “Classical” means the information processed here are digital signals or optical signals carrying classical digital information, as compared to “quantum”. This channel is also known as the “classical channel”, “post- processing channel”, “service channel” or “reconciliation channel”.

Data integrity protection is needed for the classical channel, which means the information over this channel cannot be modified, altered, or deleted, but they are publicly available (no confidentiality requirement). Usually, such protection is provided by the QKD module.

The implementations of the classical channel can be in any form of a communication channel, such as Ethernet cable, fibre link, radio wave link or even Internet. The limitations on the loss or distance on the classical channel are dependent on the relevant communication technology. Depending on the implemented QKD protocol, the classical channel needs to exchange a certain amount of data to process the raw key, in some cases it may need a high-speed link connection (e.g. tens of Gbits/s for CV QKD). Small form-factor pluggable (SFP) optical transceivers are often used in the deployment of the classical channel in QKDN. In practice, data speed requirement may also limit the distance of the classical channel.

5.3.3 Synchronisation Channel

As in ITU-T TR FG QIT4N D2.4 [9] and ETSI GS QKD 012 [22], in the implementations of QKD modules, synchronous signals are needed to exchange reference information between QKD-Tx and QKD-Rx in the synchronisation channel. The synchronous signal is an optical signal which is considered as a classical signal and typically contains many photons.

Synchronous signals are mainly used to perform clock synchronisation between quantum signals emitted from the QKD-Tx and the detections in the QKD-Rx. In some implementations, synchronous signals also serve as phase reference and compensation as well as polarization drift monitoring and compensation.

The propagation characteristics of the synchronisation channel should be close to or correlated with the quantum channel, this is to ensure proper synchronisation and compensation for the quantum signals. For this reason, the synchronisation channel may share a same channel with the quantum channel via multiplexing. However, the synchronisation channel is determined in practical implementations of QKD modules and is not defined by the QKD protocol.

5.4 Channel Multiplexing

Under a fibre-based QKDN, using Wavelength Division Multiplexing (WDM) technique is a cost-effective way to save fibre resources. In a QKDN, channel multiplexing is possible in the QKD link (quantum and classical post-processing channel), with other links in different layers, such as key management links, application links, as well as with other data traffic that is not part of QKDN. For the channel multiplexing that does not involve the quantum channel, all the channels can be multiplexed into one fibre channel, under the configuration of coarse wavelength division multiplexing (CWDM) and/or dense wavelength division multiplexing (DWDM) settings. DWDM channels are mainly located in the C band from 1528.77 nm to 1563.86 nm with 0.4 nm wavelength spacing for 80 channels, or 0.8 nm spacing for 40 channels as in ITU-T G.694.1 [14]. The optical parameters of physical interfaces for the DWDM setting are specified in ITU-T G.698.1 [17] and ITU-T G.698.2 [18]. CWDM setting has 18 wavelength channels from 1270 to 1610 nm with a channel space of 20 nm as in ITU-T G.694.2 [15], and its physical interfaces and parameters are specified in ITU-T G.695 [16].

As in ITU-T TR FG QIT4N D2.4 [9], it is also possible to multiplex the quantum channel with other channels via DWDM or CWDM into one fibre under different configurations. Since the propagating optical signals from other channels are much stronger than the quantum signals in the quantum channel, noise photons can be induced due to several nonlinear effects and leakage of the optical signals, which may further prevent the QKD modules from producing any QKD keys. Possible noise sources include light leakage, in-band amplified spontaneous emission (ASE) noise, four-wave-mixing and Raman scattering noise. Among these, Raman scattering is the dominant noise that impacts the QKD module performance. To reduce the noise impacts, wavelength isolation and narrowband filtering are added in the QKD-Rx, with a price of additional loss. Meanwhile, CV QKD can efficiently filter the noise photons even without an isolator or filter in the QKD-Rx, thanks to its coherent detection.

6. QKDN Key Management Layer

In a QKDN, the key management layer is responsible for managing and supplying keys from the quantum layer to the cryptographic applications in the service layer. In the trusted node based QKDN, QKD nodes are trusted and protected against unauthorised access and attacks. Each QKD node contains a key manager (KM) system. The KMs are connected through KM links and receive random bit strings (QKD keys) from QKD modules within the same node. The KM synchronises and formats these bit strings, storing them as keys in the storage. Interfaces for various cryptographic applications are also installed in the KM.

When a cryptographic application requests keys, the KM acquires the necessary keys from storage, synchronises them, authenticates them through the KM link, and supplies them in the appropriate format to the cryptographic application. Beyond the simple PTP architecture, if KMs do not have direct connections, they can share keys through the key relay function with the help of the QKDN controller(s) to identify a suitable relay route. The KMs then transfer the keys through KM links with highly secure encryption. Finally, the shared symmetric keys are supplied to the cryptographic applications. Once the keys have been used, the KMs implement their key management policy, such as deleting or preserving the keys.

The functional elements and operations of the key management layer in a QKDN are specified according to ITU-T Y.3800 [1], ITU-T Y.3801 [2], ITU-T Y.3802 [3] and ITU-T Y.3803 [4].

6.1 Key Management Layer Functions

In the key management layer, a key manager (KM) receives and manages the QKD keys produced from the QKD modules and QKD links. The KM also relays the keys between the QKD nodes and supplies the keys to user-defined cryptographic applications. It also performs key re-size, and key re-format with metadata and key storage. QKD link parameters, including QBER, key rate, link status, etc., are also acquired by the KM. From a functional perspective, as shown in Figure 4, three logical functional components within a KM can be defined: a key management agent (KMA), a key supply agent (KSA) and a KM control and management function. KMA and KSA in different QKD nodes are further connected via the KMA link and KSA link, respectively. In practice, these logical functional components can be implemented in a single entity or separate ones. KMA links and KSA links are both classical ones, which can share one single physical channel.

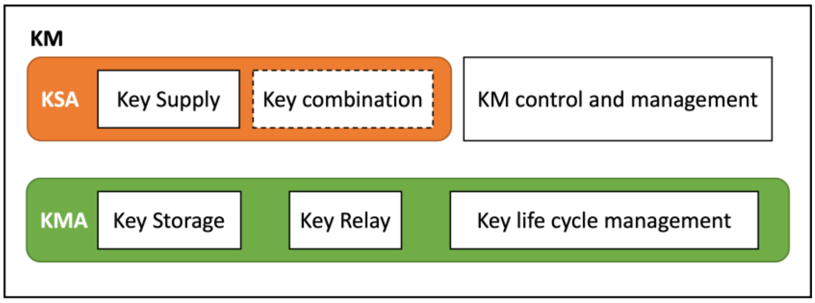

Figure 4: Logical functional components within a KM**.** Dashed lines denote optional element(s).

6.1.1 Key Management Agent

The Key Management Agent (KMA) role is to receive the keys from the QKD modules and to establish links between the QKD nodes by key relays. This is done by the following sub-functions:

(a) Key Storage Function

The following operations are performed:

- Reception of keys via an appropriate interface from various kinds of QKD modules (different protocols, vendors);
- Synchronisation of the keys (in bit position) between the KMAs;
- Entity authentication and message authentication;
- Resizing (combines or splits) of the key;
- Reformatting of the key with metadata such as key ID, key size, key type and generation of the time stamp;
- Storage of the processed key and the metadata.

(b) Key Relay Function

It performs the key relay from end to end in a QKDN through a key relay route with either an encryption with ITS such as one-time pad (OTP) or any other supported encryption method according to key management policy. More details will be described in *clause 6.2 (c)*.

(c) Key Life Cycle Management Function

- It manages the life cycle of the key from reception by KM to usage by cryptographic applications.
- Based on the key management policies, it determines key deletion or preservation in the key storage function.
- Information managed here includes key ID, key generation timestamp, QKD module ID, name of the target cryptographic application, key supply timestamp, etc.

6.1.2 Key Supply Agent

The Key Supply Agent (KSA) provides the key processed by KMA to a cryptographic application. It includes the following sub-functions.

(a) Key Supply Function

- It synchronises and authenticates the keys between the KSAs through KSA links.

- It also supplies the requested number of keys to the authorised cryptographic applications in the user network via a key supply interface, subject to the existing key management policy and the availability of the keys.
- The key supply interface is equipped with security capabilities and is usually compatible with various cryptographic applications.

More info about the key supply interface can be found in *clause 6.3*.

(b) Key Combination Function
This optional element combines the keys from the KMA and other key exchange methods, for example, keys produced by post-quantum cryptography. This is done in such a way that the security of the combined key is preserved from the input key from the KMA.

6.1.3 KM Control and Management Function

This function deals with the overall control and management function within the KM layer, and communicates with other QKDN layer elements.

- It receives status information of QKD module(s) and QKD link(s) in the quantum layer.
- It also provides (i) information on key management for QKDN control functions to the QKDN controller (ii) information on key management for QKDN management functions to the QKDN manager (iii) fault and performance information of the KM and KM links to the QKDN manager.

NOTE 7 - The information on the key management may cover information such as the origin of the key (from which QKD module), the destination of the key relay, timestamp, target cryptographic application for the key supply, key consumption rate, amount of key shared in a KM link, KM link status, accounting, and alarm on fault.

6.2 Key Management Layer Operation

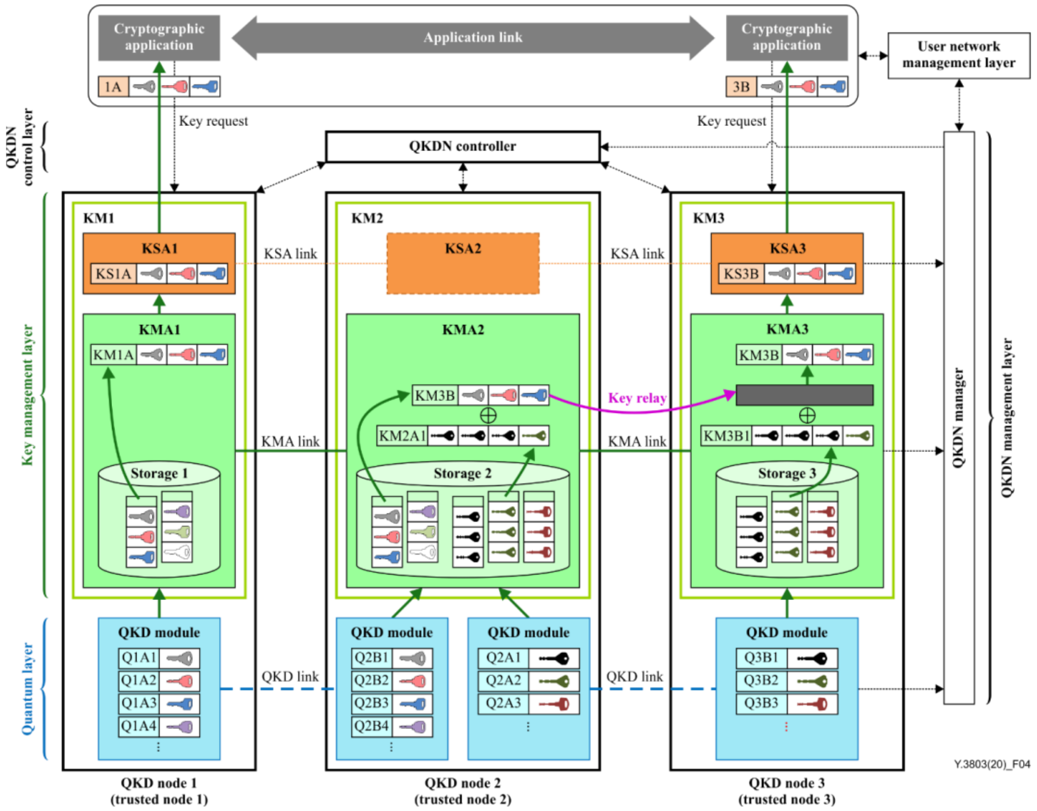

Figure 5: Functional elements and operation of key management in single QKDN controller configuration (Figure 4 in ITU Y.3803 [4]).

As shown in Figure 5, the symmetric keys (QKD-keys) are supplied by the QKD modules into the KM layers. Each QKD-key, together with its metadata, forms a QKD-key file, which will be managed by the KM layer in the following operations:

(a) Key Acquisition, Authentication and Storage (KMA)
In this step, the KMA receives the QKD-key files from the QKD modules, which are reformatted into a prescribed unit length before storing them in a buffer. The communicating nodes will then authenticate each other via the KMA link, followed by authentication and synchronisation of the keys via hash values/message authentication codes comparison. Finally, the buffered keys are stored as the key-data called *KMA-key*, with its metadata stored in a directory.

(b) Reception of Key Request (KSA and KMA)
First, the KSA receives a key request from an authorised cryptographic application. The KSA then authenticate the application via an appropriate means. Upon the establishment of authentication, common secret keys can be shared among the application and the KSA for the next key request authentication. The KSA then informs the KMA of the requested information from the applications, such as the required key length and the number of keys.

(c) Relaying of the Key (KMA)
As shown in Figure 5, the KMAs relay the keys between the endpoint KMAs, using highly secured encryption protocol such as OTP. OTP is a symmetric key encryption method in which each bit of the plaintext is encrypted by combining it with the corresponding bit from the secret key using exclusive OR (XOR). The key relay route is coordinated by the QKDN controller. The scheme depicted in Figure 5 is a typical case of a PTP key relay using OTP, which allows for a key relay with ITS between

QKD node 1 and node 3. It involves an exclusive OR of the key to be relayed upon the other key shared by the neighbouring QKD nodes. Specifically, in the case of the key relay in Figure 5, its operations are as the following:

Step 1: KM1A and KM3B, a pair of identical QKD keys, are generated from the pair of QKD modules in QKD node 1 and node 2.

Step 2: KM2A1 and KM3B1 are another pair of the identical QKD keys generated from the pair of QKD modules in QKD node 2 and node 3.

Step 3: KM3B is OTP encrypted by KM2A1 and then relayed to KMA3 via the KMA link.

Step 4: After the key relay, KM3B is recovered by the corresponding OTP decryption via KM3B1.

Step 5: Node 1 and Node 2 now share the identical keys, KM1A and KM3B, respectively.

NOTE 8 - The keys used for OTP operations are provided by QKD modules in the quantum layer, no pre-shared keys or other key materials are involved.

In another key relay scheme, the key for relaying can be also provided by the end user or the KMA equipped with a non-deterministic random number generator (RNG), including quantum random number generator (QRNG) to provide random number strings as the key. This key can be relayed to the destination KMA by using the OTP encryption and decryption with keys generated by QKD modules between two neighbouring QKD nodes along the relay route. An example of this key relay scheme is shown in Figure 6, in which the end-to-end key from the RNG for cryptographic encryption is relayed from QKD node 1 to node 3 using local QKD keys (key 1 and key 2), achieving the goal of relaying a key that is not provided by QKD modules.

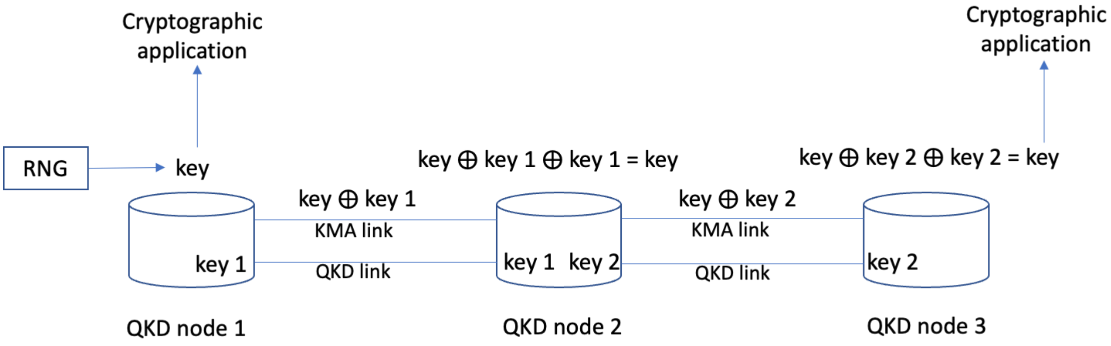

Figure 6: Key relay scheme using external RNG.

NOTE 9 - During the relay in both schemes, appropriate methods for entity and message authentication should be utilised to ensure the authenticity and integrity of the KMAs and the KMA-key, respectively. For example, the integrity protection of the KMA-key can be done via a hash value or a message authentication code.

The KMA key relay function also supports other encryption methods, such as Advanced Encryption Standard (AES). It can adopt a suitable encryption method with respect to available keys in the key storage that are supplied by QKD modules. KMA also announces the encryption methods for the relaying in the metadata, such that the security level is known. One example of key relay protocol is introduced in *clause 7.2.2* of ITU-T TR FG QIT4N D2.3.2 [8].

(d) Supplying the Key (KSA)
Upon receiving the requested key information from the KSA, KMA retrieves the required keys from the storage of the KMA-key data, optionally considering the metadata on the key relay encryption method based on the requested security level and key supply policy. The KMA then passes the key to the KSA, in which the key is designated as the *KSA-key*. As before, to ensure the keys shared by the KSA node pair are reliable and secure, the KSA pair synchronises and authenticates the KSA key via the KSA link.

During the key supply process, both the KSA and the KMA record their respective metadata in their storage, which may be sent to QKDN management for life cycle management. Once the keys are supplied to the required application, the KMAs and KSAs apply the desired key management policy, e.g., deletion/preservation of the key data in their storage.

(e) Rerouting the Key (KMA)
The key rerouting ensures the continued availability of the key supply under instances such as low KMA keys or low QKD key generation rate between the relay nodes. This is orchestrated by the QKDN manager and controller, which collects the information on key management, such as key consumption rates, from KMA via the KM control and management function. Additionally, the KM control and management function also provides the status information of the QKD modules and optionally the QKD link.

In the event of an alarm triggered by the KMA, the alarm is forwarded to the QKDN controller, which will direct the KMA to continue the key relay according to the designated rerouting methods. These methods may include manual, fixed rate, data traffic adaptive or scheduled key relay.

(f) Managing the Key Life Cycle (KM Control and Management)
The KMAs handle the key life cycle, including reception, storage, formatting, relaying, synchronization, authentication, supply and deletion or preservation of keys together with the QKDN manager, which monitors, audits, and keeps track of these workflows. The fault management function of the QKDN manager deals with unexpected fault situations by supervising the QKDN controller and KMAs to take the necessary countermeasures. For instance, in the event where some keys are compromised, the remaining keys that could be affected will be deleted.

6.3 Key Supply Interface

The key supply (or key delivery) interface for applications is between the KM in the key management layer and the cryptographic application in the service layer, as shown in Figure 2. This interface is for the key management layer to provide keys to the applications as requested. Currently available specifications include ETSI GS QKD 004 [21] and ETSI GS QKD 014 [23]. There are also some other interfaces developed by the industry without standardisation such as Secure Key Integration Protocol (SKIP). One key supply interface protocol is also introduced in *clause 7.2.1* of ITU-T TR FG QIT4N D2.3.2 [8].

As recommended by ETSI GS QKD 004 [21], the key supply interface includes three Application Programming Interface (API) functions:

- *OPEN_CONNECT: create a session for a Key_stream_ID for a set of keys to be delivered;*
- *CLOSE: terminate the session for a Key_stream_ID;*
- *GET_KEY: obtain the required amount of key material requested for this Key_stream_ID;*
- *Key_stream_ID is a unique identifier for the keys provided by the KM to the application.*

NOTE 10 - Such key supply API can maximise key throughput with a minimal overhead by creating a continuous stream of keys for applications.

It is also implementation agnostic as the developer can choose the programming language that suits the specific requirements. Furthermore, it can serve as the API for the KM at multiple levels, i.e., the interface between the KMA and the QKD module, and between the KSA and the KMA, also between the KSA and the cryptographic application, as illustrated in Figure 5.

As recommended by ETSI GS QKD 014 [23], the key supply interface can be implemented in the form of a Representational State Transfer (REST) API. In general, a REST API describes a client-server interface between different components in the networks. REST APIs have been widely adopted by the software developer's community due to their simplicity and scalability.

In the case of the QKDN, the cryptographic application is the client and the KM is the server. Each KM has a unique ID (KM ID) in a QKDN. Each cryptographic application also has a unique ID (application ID). The cryptographic application requests the keys from the KM. Then the KM delivers the keys. All communications between the application and the KM are implemented in the Hypertext Transfer Protocol Secure (HTTPS) protocols. HTTPS is widely used on the Internet for secure communication over computer networks. Data in the message body of HTTPS requests from the application to the KM and HTTPS responses from the KM to the application is encoded in JavaScript Object Notation (JSON) format. JSON is an open standard file format and data interchange format that uses human-readable text to store and transmit data objects including attribute-value pairs and arrays.

There are three methods specified in the REST API key supply interface:

- Get status: returns information on keys available from the KM to the application;
- Get key: returns key container data from the KM to the application. Key container data includes one or more keys;
- Get key with key IDs: returns key container from the KM to the application. Key container includes keys matching those previously delivered to the application based on the key IDs from that application in response to its call on Get key.

An example format of the key container with two keys is as follows:

```
{
        "keys": [
                {
                        "key_ID": "abc",
                        "key": "123"
                },
                {       "key_ID": "def",
                        "key": "456"
                }
        ]
}
```

Figure 7 shows a use case to utilise the key supply API. Cryptographic application 1/2 is directly connected to the KM 1/2. Application 1 can launch secure communication with application 2 based on the following steps:

Step 1: application 1 calls the "Get key" method of the key supply API with the application 2 ID to get keys from KM 1.

Step 2: KM 1 delivers the key materials with the associated key IDs to application 1. Such key materials and key IDs are also (to be) shared with KM 2.

Step 3: application 1 sends the key IDs to application 2.

Step 4: application 2 calls the "Get key with key IDs" method of the key supply API with the application 1 ID and the notified key IDs information from application 1 to get the identical keys from KM 2.

Step 5: KM 2 delivers the identical key materials with the associated key IDs shared with KM 1 to application 2.

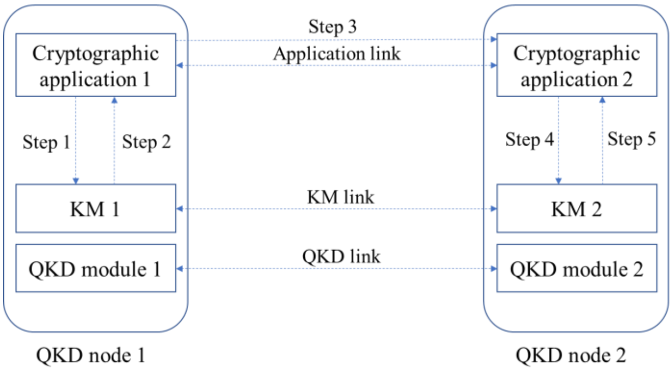

Figure 7: Use case of the key supply API.

NOTE 11 - The application and KM links are logical which means that QKD node 1 and node 2 are either connected directly or through multiple intermediate links.

NOTE 12 - The standardisation of the key supply interface between the KM and the QKD module is still in progress. In principle, the above-mentioned ETSI GS QKD 014 [23] may also be used for this interface. However, ETSI GS QKD 014 is not optimised or designed for such an interface or purpose.

7. QKDN Control Layer

The QKDN control layer consists of QKDN controller(s) to achieve secure, robust and efficient operations and services in a QKDN. As shown in Figure 2 and Figure 5, The QKDN controller controls the key management layer and the quantum layer by sending control information to the KMs, the QKD modules and the QKD links. The QKDN controller also provides functions for the QKDN management layer and the service layer by communicating management information with the QKDN manager.

The functions of the QKDN control layer include Routing control, Configuration control, Policy-based control, Access control and Session control.

7.1 Routing Control Function

Routing is the process of selecting a path(route) in a QKDN to deliver keys. As recommended by ITU-T Y.3804 [5], the routing control includes the following functions.

7.1.1 Routing of the Key Relay between two end-point KMs.

As an example of routing of the key relay, between KMs, in Figure 2 and Figure 5, QKD nodes 1 and 3 are two end-points that deliver symmetric QKD keys to their encryption/decryption applications. Then the keys generated in QKD node 1 need to be relayed to QKD node 3. In a practical QKDN as illustrated in Figure 8, either QKD nodes 2, 4 or 5 can be the relay node. There are three possible routes:

- Route 1: QKD node 1 → QKD node 2 → QKD node 3
- Route 2: QKD node 1 → QKD node 4 → QKD node 3
- Route 3: QKD node 1 → QKD node 5 → QKD node 3

Each route contains two hops. Then the routing is performed in the following steps:

- Receive requests of the required number of keys from the KMs in QKD nodes 1 and 3 based on their cryptographic applications.
- Obtain information on the remaining available number of keys and key consumption rate of the KMs in the nodes along each route from the key management layer, QKD module and link parameters from the quantum layer.
- Update and manage the routing table after analysing the above information. The routing table is a database that contains information about the source and destination (e.g., IP address) for all hops along each route.
- Determine and provide the key relay route according to the routing table. In this case, all three routes can be used if they meet the requirements in 2). In addition, the key relay route optimisation can be done with additional information (e.g., topology) from the QKDN manager. This is because the QKDN manager monitors the entire status of the quantum layer and the key management layer and store and update such information in a database.

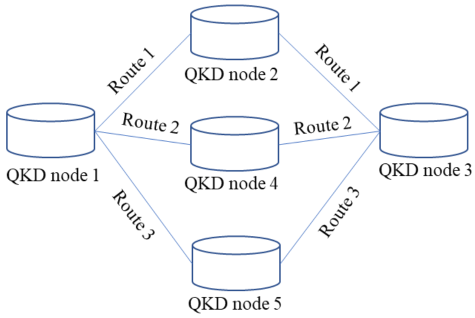

Figure 8: Conceptual diagram of the routing control.

7.1.2 Rerouting of the Key Relay based on the status of the Quantum Layer and/or the Key Management Layer.

This function is performed when any of the following cases occurs:

- The remaining available number of keys in the relay nodes is insufficient (the key amount is below the required threshold).
- A fault is detected in the KMs of the relay nodes and the KM links between the relay nodes.
- The QBER or excess noise surpasses the required threshold in QKD links between relay nodes.
- A fault is detected in the QKD modules of the relay nodes.

In Figure 8, if route 1 fails, the QKDN controller will automatically switch to another available route such as route 2 or 3 to ensure continuous operation and service in the QKDN. Any faulty KMs, KM links, QKD modules and QKD links are de-activated, allowing the QKDN manager to handle the root cause analysis and countermeasures. A specific routing control protocol in the QKDN control layer is introduced in *clause 8.1* of ITU-T TR FG QIT4N D2.3.2 [8].

7.2 Configuration Control Function

This function mainly includes:

- To obtain control-related configuration information of KMs and KM links in the key management layer and QKD modules and QKD links in the quantum layer.
- To control the state (in service, out of service, standby, reserved) of the above components.
- To reconfigure KM links and QKD links if an alarm or failure occurs.

For KM links, the reconfiguration is done by replacing faulty classical channels with new ones and putting them into service. For example, in Figure 9(a), classical channel 1 is the default channel while classical channel 2 is a backup channel. When classical channel 1 is broken, classical channel 2 will be put into service to replace classical channel 1.

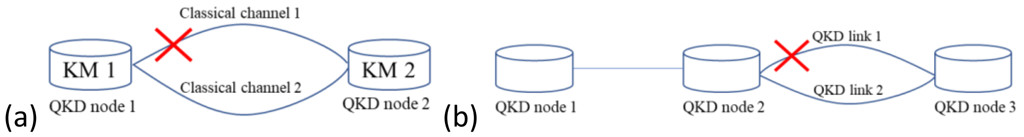

Figure 9: (a) Reconfiguration in KM links; (b) Reconfiguration in QKD links.

In QKD links, for example, an alarm or failure occurs when the QBER or excess noise increases, which can be due to eavesdropping or device imperfections, as well as the loss in the quantum channel surpassing a certain threshold. Once such failure occurs, the reconfiguration can be done by controlling optical switch/splitter modules in the relay nodes or putting a backup quantum channel (if applicable) into service. For example, in Figure 9(b), when QKD link 1 becomes problematic, the backup link (QKD link 2) will be put into service to replace QKD link 1.

7.3 Policy-based Control Function

In a QKDN, a policy describes the statements/specifications that a network administrator uses to give priority to key supply for some service/applications at a certain period on a particular part of the network. For example, a policy statement could be: “deliver keys in the fastest way to cryptographic applications in QKD node A and QKD node B between 9 am to 9 pm”. As recommended by ITU-T Y.3804 [5], such policies usually include:

- Quality of Service (QoS) policy. This is a policy to prioritise application-critical key traffic over lower priority traffic, i.e., provide guaranteed bandwidth to key supply for the most important cryptographic applications in the service layer.
- Key management policy. This is a policy imposed by requirements from the key management layer in *clause 6*.
- Charging policy. This is a policy determined by cryptographic applications that the users employ in the service layer.

The policy-based control is executed with support from the QKDN manager at which policies are entered and updated. The policy-based control in each QKD node enforces the policies by controlling the policy enforcement points (PEPs) such as KMs and KM links in the key management layer and QKD modules and QKD links in the quantum layer.

7.4 Access Control Function

The access control allows QKDN administrators to keep unauthorised users and components out of a QKDN, hence ensuring secure operations and services. The access control is performed by verifying the functional components (e.g., QKD modules, KMs, cryptographic applications) on their authorised rights. As recommended by ITU-T Y.3804 [5], the access control has a database storing authorised access rights and priorities of the above functional components according to enforced policies. The access control includes the following functions:

- To register and delete IDs of functional components in the access control repository.
- To issue certificates to the registered functional components.
- To authenticate between the QKDN controller and functional components based on their certificates.
- To authenticate between functional components based on their certificates and records in the access control repository.

7.5 Session Control Function

In a QKDN, a session is a time-delimited communication link between KMs to set up the end-to-end key or to deliver keys to cryptographic applications in the service layer of the user network. A session has a starting and ending time. The session control starts, maintains and ends the session.
As recommended by ITU-T Y.3804 [5], the session control includes the following functions:

- To support the session control for KMAs to establish the end-to-end key via KMA links, as shown in Figure 5. The KMA controls the session based on the key management policy.
- To support the session control for KSAs to deliver keys to cryptographic applications in the service layer, as shown in Figure 5. The KSA controls the session based on charging policies imposed by the policy-based control described in *clause 7.3*.

8. QKDN Management Layer

As recommended by ITU-T Y.3804 [5], the QKDN management layer consists of the QKDN manager to provide fault, configuration, accounting, performance, and security (FCAPS) functions to manage the entire QKDN and support user network management. Fault, configuration, accounting, performance, and security are the five areas of functions into which the International Organization for Standardisation (ISO) model defines network management tasks. FCAPS is the ISO Telecommunications Management Network (TMN) model and framework for network management. Some of those specifications also apply to the QKDN management layer functions, while some other specifications have been covered previously in *clause 6* and *clause 7.1*.

8.1 Common Management Function

The common management function is mostly related to the FCAPS management for the quantum layer, the key management layer and the QKDN control layer.

8.1.1 Fault Management

The objective of fault management is to identify, isolate, correct and record faults that occur in the QKDN. Moreover, faults' data and their trend can be analysed to predict future faults so that the QKDN operation and service disruption can be minimised.

When a fault occurs, the faulty QKDN component will send a notification to the QKDN manager to collect the relevant information. Then, the QKDN manager will notify a QKDN administrator about the problem, allowing suitable action to be taken. Such action can be either manual or automatic. For instance, to put backup components into service or to collect more data to diagnose the problem. There are various fault management software available commercially.

The QKDN manager supports the following fault management functions:

- To monitor QKD link failures for recovery of QKD modules. The recovery action includes rerouting of key relay routes and reconfiguration of QKD links as shown in Figure 8 and Figure 9.
- To offer fault detection and root-cause diagnosis for QKDN control, key management and quantum layers.
- To generate failure-resolving policies and communicate with each layer for restoration actions with support from the QKDN controller as described in *clause 7.1*.
- To discover functions, managed resources and bootstrap in each layer to get them ready for the operation based on the bootstrapping policies.

8.1.2 Configuration Management

The configuration management involves gathering and monitoring the QKDN component configuration information, tracking any changes made to the configuration, and simplifying the configuration of the QKDN components. Some commercially available configuration management software can be used to achieve these goals.

The QKDN manager supports the following configuration management functions:

- To provision and configure the managed resources (software and hardware) of the QKDN components in each layer. For example, to upgrade the operating system of a KM, or to add a new optical device in a QKD module.
- To manage the configuration status of the QKDN components in each layer.
- To perform inventory management for all the managed resources of the QKDN components in each layer.
- To manage the life cycle of the managed resource repositories (create, remove, modify, retrieve, store, etc.) of the QKDN components in each layer.
- To manage the network topology of each layer. For example, to configure a key relay route as described in *clause 7.1*, or to plan for future scaling of the QKDN.

8.1.3 Accounting Management

The objective of the accounting management is to track QKDN utilisation information so that external users or business units can be appropriately billed or charged for accounting purposes.

The QKDN manager supports the following accounting management function:

- To measure the resource usage data of each layer (e.g., usage of keys in the quantum layer) and create accounting policies for charging/billing.

8.1.4 Performance Management

The objective of the performance management is to ensure that the QKDN performance is maintained at acceptable levels. It provides the QKDN administrator with a broad view of the performance of the current QKDN. The QKDN performance includes network response times, link utilisation, key rate, QBER etc. The performance data can be gathered through the implementation of a Simple Network Management Protocol (SNMP) management system. By actively monitoring current performance data, potential problems can be identified before they occur. Alternatively, the SNMP management system can be configured to send an alarm to QKDN administrators when performance drops below predefined thresholds. Such alarm notification is handled by the fault management described in *clause 8.1.1*. Some commercial performance management software is also available.

The QKDN manager supports the following performance management functions:

- To collect status and performance data of each layer, store and update them in a database.
- To analyse the collected performance data and create performance reports. Trends that indicate issues can be identified before they affect services.
- To manage the key supply service policies as described in *clause 7.1*.

8.1.5 Security Management

The objective of the security management is to control access to resources in the QKDN. The security management does not only keep the QKDN environment secure, but also analyse collected security-related information regularly.

The QKDN manager supports the following security management functions:

- To collect event logs, audit trails, metadata, etc of each layer to detect security anomalies.
- To support key life cycle management by KMs and use the log database to ensure the traceability of keys.
- To issue root certificates to the QKDN controller, and support the QKDN controller for the access control as described in *clause 7.4*.
- To manage the key management policies and send them to the QKDN controller as described in *clause 7.3*.

8.2 Layer Specific Management Functions

As the QKDN manager provisions FCAPS management for the quantum layer, the key management layer, the QKDN control layer, the layer specific management includes the following functions: quantum layer management (QLM), key management layer management (KMLM), QKDN control layer management (QCLM).

8.2.1 Quantum Layer Management

The quantum layer specific FCAPS management includes the following functions:

- To detect eavesdropping attempts on the quantum channel.
- To collect and analyse QKD module performance information (e.g., key rates).
- To manage the availability and reliability of QKD operations based on the redundancy of QKD links in the quantum layer.
- To support metadata abstraction for mapping component-dependent data into component-independent data for component interoperability in the quantum layer.

8.2.2 Key Management Layer Management

The key management layer specific FCAPS management includes the following function:

- To collect and analyse the available number of keys in KMs for key relays, key supply services and the key life cycle management.

8.2.3 QKDN Control Layer Management

The QKDN control layer specific FCAPS management includes the following functions:

- To support the QKDN controller in the routing and rerouting of key relays, and in carrying out rules and policies due to performance degradation or faults.
- To support the QKDN controller for provision of routing and rerouting of key relay routes if the QKDN supports the key relay as the configuration management as described in *clause 8.1.2*.

8.3 Cross-layer Management Orchestration

The cross-layer management orchestration (XLMO) orchestrates the quantum, key management and QKDN control layers. It also orchestrates control and management functions of the quantum layer, the key management layer and the QKDN control layer. It interacts and coordinates with external management entities, e.g., the user network management layer, network operator's operations support system (OSS), business support system (BSS), etc.

As recommended by ITU-T Y.3804 [5], the XLMO supports the following management functions:

8.3.1 Orchestration for cross-layers management

This orchestration mainly includes:

- To provide management coordination of the quantum layer, the key management layer and the QKDN control layer.
- To provide management orchestration of the QKDN control layer and QKDN management layer to support the QKDN controller in addressing anomalies (e.g., performance degradation, fault event, security attack).
- To divide the provisioning information into three types of initialisations and configuration information for three layers (the quantum layer, the key management layer and the QKDN control layer) and to carry out provisioning tasks per layer in sequence.

8.3.2 Orchestration for External Management

This orchestration mainly includes:

- To provide management orchestration with external management systems, especially with the user network management system.

8.4 SDN-based QKDN

As recommended by ETSI GS QKD 015 [24], in addition to the conventional QKDN architecture in Figure 2, a software-defined networking (SDN)-based QKDN architecture can also be adopted. The core idea of SDN is to decouple the control plane from the data plane to enhance the programmability, flexibility and intelligence of the network. Figure 10 shows a conceptual SDN-based QKDN architecture. Compared to the conventional QKDN architecture, the main difference is that the QKDN controller is extracted from the QKD node. A centralised QKDN SDN manager and controller that combines the management and control functions and logically sits on top of all the QKD nodes is introduced. The QKDN SDN manager and controller are connected with the SDN agent in each QKD node via standard SDN protocols such as OpenFlow to communicate control information described in *clause 7*. The QKDN SDN manager and controller has an abstracted global view of the whole QKDN, thus reducing the complexity of managing and controlling all the QKD nodes separately and making the optimisation of the QKDN more efficient. All the control and management functions described in *clause 7* and *clause 8* remain the same in the SDN paradigm. The SDN agent acts as a simplified QKDN controller as in the conventional QKDN architecture, but with much of the functions moved to the centralised QKDN SDN manager and controller.

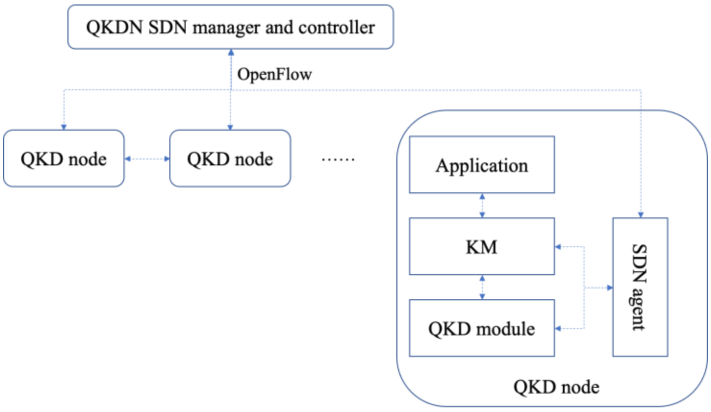

Figure 10*:* Conceptual architecture of an SDN-based QKDN.

8.5 Implementation Considerations

The QKDN control and management layer functions can be implemented in software under the specifications listed in *clause 7 and clause 8*, and with references from several commercially available classical network control and management software from different vendors.

There are various widespread network control and management protocols that can be used in software implementation such as:

- SNMP, as mentioned in *clause 8.1.4*, is a common approach to be deployed on networking components for obtaining network information.
- Network Configuration Protocol (NETCONF) is a protocol that provides mechanisms for network management tools and for administrators to configure network components.
- RESTCONF is a protocol built on top of NETCONF using a RESTful API to change and upgrade network configuration.
- gRPC (Google Remote Procedure Call) Network Management Interface (gNMI) is a protocol that provides ways to obtain the status of the network components and modify the configuration of the components.

A feasible approach is to implement the quantum layer and the key management layer as new functionalities added to the existing control and management software.

9. QKDN Use Cases and Considerations

This clause introduces some QKDN use cases as examples, to demonstrate how a QKDN can be used in practice. It includes use cases for each layer and the whole of the QKDN: quantum layer, key management layer, QKDN control and management layer, as well as the service layer, in which different applications can be realised with the keys provided by a QKDN.

9.1 Overview of QKDN Use Cases and Applications

With keys provided by the QKDN and coupled with cryptographic primitives, various use cases and applications can be realised. QKD use cases can be classified from different angles as in ITU-T TR FG QIT4N D2.2 [6] and ETSI GS QKD 002 [19]: QKD combined with other cryptographic primitives; integrated with various Transmission Control Protocol/Internet Protocol (TCP/IP) protocols; deployed in various network topologies; with different user device categories; integrated into various network forms; applied in different vertical sectors.

9.1.1 QKD-keys Combined with Cryptographic Primitives:

QKD-keys can be used for symmetric encryption such as OTP or AES. QKD-keys combine with message authentication functions such as, e.g., universal-II hash functions, and message authentication code (MAC). QKD combined with Shamir's secret sharing algorithm to perform secure storage. QKD raw key can be used to implement oblivious key transfer to perform secure multi-party computation. QKD-key can be further combined with Post-quantum cryptography (PQC) and Quantum random number generator (QRNG).

Some concepts:

- Backup and Disaster Recovery – hybrid encryption solution, incorporating layer 2/3 encryption. QKD and Ethernet encryptors to secure the backbone.
- High-Value Assets and Data Protection – secure transport and storage of assets/data using QKD and secret sharing protocols (e.g. Shamir secret sharing).
- Trusted Randomness Beacons – implementation and deployment of cryptographic beacon as verifiable cloud service to provide a source of quantum randomness along with QKD-keys for applications such as contract signing, e-voting, privacy enhancements and data masking.
- PQC can be used as an authentication method to initialise the QKD operation.

9.1.2 QKD-keys Integrated with Various TCP/IP Protocols

QKD can be integrated with TCP/IP protocols in different layers, such as point-to-point protocol (PPP), virtual private network (VPN) tunnel, MACSec protocol at the data link layer, IPSec protocol at the network layer, Transport Layer Security (TLS) protocol at the transport layer.

Some concepts:

- Secure Video Conference – Secure VPN solution for quantum safe access and communication services. QKD and Ethernet encryptors to secure backbone.

9.1.3 QKDN Implemented in Various Network Topologies

There are various QKDN topologies such as line, ring, star, mesh and mixed type which are connected via fibre and/or free-space channels. The QKDN can be deployed in fibre-based metropolitan access networks, fibre-based inter-city backbone networks, free-space satellite-ground or inter-satellite networks.

Some concepts:

- Secure Firmware Upgrade – terrestrial and free space QKD for securing the communication channel for upgrading firmware of platforms such as maritime/drone communication.

9.1.4 QKD Modules with Different Types of User Devices

QKD modules can be integrated with terminal devices at different integration levels. Fixed User device connected to a standalone QKD module or integrates QKD module as an internal component. Remote user device consumes offline keys provided by QKDN. Remote user device integrated QKD module consumes QKD keys in real time.

Some concepts:

- QKD-key as a Service – supporting request from the end-users at the application level to obtain QKD-key from QKDN instead of deploying their dedicated QKD node.
- Quantum safe end-to-end encrypted smartphones – QRNG enabled smartphones integration with QKDN, complete with secure applications/services.

9.1.5 QKDN Integrated in Different Network Forms

QKDN can be integrated into various Information and Communications Technology (ICT) network forms which require high-security guarantee, e.g., 4G/5G, SDN/NFV-based, cloud computing, blockchain, Time Sensitive Networking (TSN), service chain and other future network evolutions, e.g., Scalability, Control, and Isolation On Next-Generation Networks (SCION), quantum internet.

Some concepts:

- 5G network slicing – distribute encryption keys for encrypting traffic in network slices.

9.1.6 QKD Technologies Adopted in Different Vertical Sectors

QKD technologies can be adopted in sectors that need high-level and long-term security, e.g., finance, government, health care, energy, telecom and critical infrastructure.
Some concepts:

- Data Centre interconnect network protection – secure data in transit and at rest for data centres.
- Quantum Security for the Industrial Internet – quantum-safe communication for Industrial Internet and mission-critical infrastructures.

NOTE 13 - The following use cases introduced are only for the purpose of technical information sharing and does not imply that they are endorsed or recommended by IMDA, TSAC or the Quantum Communication Networks Task Force (TF).

9.2 QKDN Use Case 1: Secure Data Centre Interconnect

Problem Statement:
Data centres are facilities that centralise information technology infrastructures to support enterprise applications and services. These include e.g., productivity and business applications, voice/video conferencing, remote collaboration, compute engines, data storage and analytics.

To achieve service scaling and resiliency, data centre interconnects are established to link up multiple data centres. The interconnects transfer aggregation of the enterprise’s valuable information such as applications and workload processing, synchronisation data, and backups. It is thus important that the interconnects are secured for confidentiality, integrity, and availability.

Use Case Description:
The data centres' interconnects are often secured using Virtual Private Network (VPN). The VPN can be implemented with link/network encryptors, which use symmetric cryptography and are typically capable of performing encryption and decryption at Giga-bits per second speed. To further safeguard the confidentiality of the VPN, the encryptors can be augmented with QKD technology as in ITU-T TR FG QIT4N D2.2 [6] and ITU-T TR FG QIT4N D2.5 [10]. The QKD enables information-theoretic secure distribution of the symmetric cryptographic keys and facilitates frequent re-keying to limit the amount of communication data protected with any given key.

Benefits:
The augmentation of QKD to encryptors (that implement mathematical cryptographic algorithms such as PQC and AES that are deemed quantum-safe) provides a hybrid and layered defence-in-depth security assurance for the VPN that interconnects the data centres.

Actors/Domains:
Data centres

9.3 QKDN Use Case 2: Key Management with Hardware Security Module

Problem Statement:
The key management (KM) layer of a QKDN relies on the existence of the keys generated, and propagated, via that network.

While the (human, or machine) user of the KM system is aware that the bits it is receiving are "keys" or otherwise to be used as key material, to the key management system, these are simply blocks of data. By having a secure key management system, the source of the data is not critical, only that the data are securely generated, securely propagated, and securely stored (i.e., data in use, data in motion, and data at rest). A good key management system provides its functions via both a machine-readable back-end format, as well as a human-readable and accessible front-end.

Use Case Description:
The root of trust for a secure key management system is a hardware security module (HSM). The HSM is a separate computer system, with its own on-board hardening and security awareness, dedicated to the prevention of exposure of the key material it holds, uses or makes available.

On use, an application will assemble the necessary inputs and send the command and its inputs to the HSM as in ITU-T TR FG QIT4N D2.2 [6]. The HSM will prepare the key, process the inputs based on the command issued, and return the output of the command. The key is never exposed to the host application outside the HSM in an unencrypted form.

Two possible scenarios with the HSM application are considered here:

- For installations with only a few standalone QKD nodes, it is expected that each QKD node that is managing key material between QKD nodes will have its own Peripheral Component Interconnect Express (PCIe) HSM.
- For major relay stations, or for nodes with multiple or complex QKD systems and system topologies, these may benefit from a single cluster of high-performance, appliance form-factor HSMs, usable independently by each of the KM systems in the node.

In addition to the features required by QKD and QKD KM, it is also possible for the HSM to be compatible with asymmetric algorithms beyond the classical Rivest–Shamir–Adleman (RSA) and Elliptic Curve Cryptography (ECC), such as those being investigated by NIST for use as "Quantum-Safe" algorithms. This ensures that as more quantum-safe functionality matures, the QKDN can benefit therefrom.

As a usage example, an application requiring cryptographic key support can then issue a key request to any KM in the QKDN, using for example KMIP (Key Management Interoperability Protocol). The KM will retrieve one or more requested keys from its storage, use the HSM to decrypt the keys, and then return them over a secure channel to the requestor, in the format desired by the requestor. Certain HSMs can be extended to do the format conversion and encryption translation internally, via custom firmware modules, for instance.
Some other functions to augment the KM functions described in Sections 7.1 and 7.2 in the context of clusters topology described above include:

(a) Key Replication (within a cluster)
Replication of key material between KM systems within the (local) classical network, i.e. the local cluster.

(b) Key Relay (between clusters)
Relay is replication between clusters, and may be "on-demand" for specific keys. This is for chunks of the topology that do not have a point of ingestion, i.e. they do not have a QKD-capable node.

(c) Key Injection

For high-security demands, the KM can inject the keys directly into attached target security devices like point-of-sale (POS) terminals or electronic control units (ECUs) at the (secured) manufacturer site. To meet all necessary security requirements, the KM, and the target run a cryptographic protocol for secure handshaking and secure messaging. This approach enables a secure end-to-end channel from the initiator of the key distribution process to the target platform, e.g. an ECU.

Benefits:
The HSM provides general purpose cryptographic functionality, as well as secure key and key material storage. The internal key storage is protected by state-of-the-art hardware protection mechanisms to also thwart combinations of physical and logical attacks. In addition, since the capacity of the internal key storage is limited, there is an option to extend the key storage to an external memory. This external key storage is protected by the cryptographic mechanisms of the HSM and hence provides a high level of security (comparable to the internal key storage). Each key may be independently gauged, for whether it is suitable for internal or external storage.

Actors/Domains:
Key Management layer

9.4 QKDN Use Case 3: Satellite-based QKD

Problem Statement:
QKD has a distance limitation in establishing QKD keys between two remote parties. It is difficult for governments and other commercial organizations to achieve end-to-end QKD globally only through fibre networks. Satellite networks also need effective encryption method to ensure the communication security.

Use Case Description:
In fibre QKD installations, the QKD links may be fixed fibre line connections between two trusted nodes, or optical switches may be used so that signals can be routed to different nodes (within the allowable distance/loss budget range). In trusted node satellite-to-ground systems, the satellite nodes are in constant motion, connecting with the various optical ground station (OGS) that they fly over for only a few minutes every day, weather permitting. In order to establish continuous coverage, a geostationary satellite or a complex constellation of low earth orbit satellites would need to be used, but these are significantly more challenging technically and are unlikely to be widely available in the near future.

The required technologies for the QKD-Tx and Rx also vary depending on whether fibre-based QKD, free-space QKD on the ground, or satellite-to-ground QKD is implemented. At a high level, a transmitter for a fibre-based QKD link would consist of the source that generates the particles (typically photons) being used for QKD. The receiver in a fibre-based link would include the detectors for measuring quantum states. For a free-space link on the ground, there would additionally be optical terminals (telescopes) at the transmitter and receiver. Finally, for a satellite-to-ground link, the transmitter would consist of the source and an optical terminal on the satellite, while the receiver would consist of an OGS and detectors on the ground. The satellite transmitter and ground receiver must track each other with great accuracy as the satellite passes overhead to maintain a sufficiently low-loss optical link. For keys generated onboard satellites, the satellite may implement KM by acting as a trusted node which receives, stores and supplies QKD keys (see Figures 11-14 on page 28 in ETSI GS QKD 002 [19]). If two users establish QKD keys with the satellite node, the satellite can then supply XORs of the keys via classical (e.g. radio) communications on demand. This allows the users to determine each other’s keys and communicate securely. Since the satellite can access the stored keys, it must be trusted to be secure.

To realise a larger QKD network, a constellation of trusted-node QKD satellites would need to be used. In this case, the entire network would be queried for key XORs. These would be broadcast to the users via relay satellites with quantum inter-satellite links. The ground station nodes, which may also be trusted nodes that connect to further fibre QKD networks, must ration the usage of satellite-based keys which are likely far less abundant than the fibre-based keys.

Benefits:

Counter space satellite network attacks and improve the defence performance of space networks.

Achieve end-to-end QKD sessions between two places worldwide, achieve the global availability of QKD key.

Wider and denser coverage of QKD encryption services and easier for mobile and remote user terminals to access.

Actors/Domains:

End users who cannot connect to optical fibres; with strong mobility; looking for a high security network solution; network operators.

10. References

[1] ITU-T Recommendation Y.3800 (2019), plus Corrigendum 1 (2020), Overview on networks supporting quantum key distribution, 2020.

[2] ITU-T Recommendation Y.3801, Functional requirements for quantum key distribution networks, 2020.

[3] ITU-T Recommendation Y.3802, Quantum key distribution networks – Functional architecture, 2020.

[4] ITU-T Recommendation Y.3803, Quantum key distribution networks – Key management.

[5] ITU-T Recommendation Y.3804, Quantum key distribution networks – Control and management, 2020.

[6] ITU-T Technical Report (TR) FG QIT4N D2.2, Quantum information technology for networks use cases: Quantum key distribution network, 2021.

[7] ITU-T Technical Report (TR) FG QIT4N D2.3-part 1, Quantum key distribution network protocols: Quantum layer, 2021.

[8] ITU-T Technical Report (TR) FG QIT4N D2.3-part 2, Quantum key distribution network protocols: Key management layer, QKDN control layer and QKDN management layer, 2021.

[9] ITU-T Technical Report (TR) FG QIT4N D2.4, Quantum key distribution network transport technologies, 2021.

[10] ITU-T Technical Report (TR) FG QIT4N D2.5, Standardization outlook and technology maturity: Quantum key distribution network, 2021.

[11] ITU-T Recommendation G.652, Characteristics of a single-mode optical fibre and cable, 2016.

[12] ITU-T Recommendation G.655, Characteristics of a non-zero dispersion-shifted single-mode optical fibre and cable, 2009.

[13] ITU-T Recommendation G.657, Characteristics of a bending-loss insensitive single-mode optical fibre and cable, 2016.

[14] ITU-T Recommendation G.694.1, Spectral grids for WDM applications: DWDM wavelength grid, 2021.

[15] ITU-T Recommendation G.694.2, Spectral grids for WDM applications: CWDM wavelength grid, 2021.

[16] ITU-T Recommendation G.695, Optical interfaces for coarse wavelength division multiplexing applications, 2018.

[17] ITU-T Recommendation G.698.1, Multichannel DWDM applications with single channel optical interfaces, 2018.

[18] ITU-T Recommendation G.698.2, Multichannel DWDM applications with single channel optical interfaces, 2018.

[19] ETSI GS QKD 002, Quantum key distribution; use case, 2010.

[20] ETSI GR QKD 003, Quantum key distribution; Components and Internal Interfaces, 2018.

[21] ETSI GS QKD 004, Quantum key distribution; Application Interface, 2020. © ETSI 2023. All rights reserved.

[22] ETSI GS QKD 012, Quantum key distribution; Device and Communication Channel Parameters for QKD Deployment, 2019. © ETSI 2023. All rights reserved.

[23] ETSI GS QKD 014, Quantum key distribution; Protocol and data format of key delivery API to Applications, 2019. © ETSI 2023. All rights reserved.

[24] ETSI GS QKD 015, Quantum key distribution; Control Interface for Software Defined Networks, 2021. © ETSI 2023. All rights reserved.

NOTE 14 - Please be aware that the ETSI standards might contain patent-protected technologies. Standards Essential Patents (SEPs) disclosed to ETSI can be checked in the ETSI IPR database at https://ipr.etsi.org/.