ETSI GS QKD 015 V1.1.1 (2021-03)

GROUP SPECIFICATION

Quantum Key Distribution (QKD); Control Interface for Software Defined Networks

Disclaimer

The present document has been produced and approved by the Quantum Key Distribution (QKD) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference

DGS/QKD-015_ContIntSDN

Keywords

control interface, quantum cryptography, Quantum Key Distribution, Software-Defined Networking

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Quantum Key Distribution (QKD).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

The present document deals with the interface between a SDN-QKD node and a SDN controller. It describes the flow of information between both entities, The SDN-QKD node part being embodied by an SDN-Agent that collects the local node information. The information model is given in YANG [1] and [2], a language well suited and widely used for these purposes. The information model is agnostic from the vendor and the implementation, permitting to control any type of QKD systems, whilst also enabling to the centralized SDN controller to build an end-to-end view of the network for managing and optimizing the transmission of quantum signals and also to deliver the QKD-derived keys.

Introduction

Quantum Key Distribution relies on the creation, transmission and detection of signals at the quantum level. This is difficult to achieve if the network used for the transmission is also in use with classical signal, which are much more powerful. On the other hand, the quantum transmission can be neither amplified nor regenerated - at least without quantum repeaters, which are not feasible with current technology - implying a limited reach for quantum communications and the need to resort to trusted repeaters to increase the distance. To optimize the transmission of quantum signals together with classical communications - whether they share the same physical media or not - over a network and manage the key relay required for longer distances, it is necessary to integrate the QKD systems such that they can communicate with the network control and also receive commands from it. These network-aware QKD systems have to be integrated at the physical level (e.g. to allocate spectrum for the quantum channel, dynamically change the peer, or use a new optical path, etc.), but also logically connected to the management architectures. To achieve this integration, the required capabilities of the QKD devices have to be described to the network controller. YANG [1] and [2] is the major modelling language used to describe network elements. Any new elements, services or capabilities being defined usually come together with a YANG model for enabling a faster integration into management systems.

The purpose of the information model presented in the present document, regardless of the protocol chosen to implement the control channel, is to simplify the management of the QKD resources by implementing an abstraction layer described in YANG. This will allow to optimize the creation and usage of the QKD-derived keys by introducing a central element through the SDN controller. This is a standard component of SDN networks that has a global view of the whole network. This abstraction layer will enable the SDN controller to simultaneously manage both, the classical and quantum parts of the network. The integration has the added benefit of using well-known mechanisms and tools in the classical community, which will facilitate its adoption and deployment by the telecommunications world.

1 Scope

The present document provides a definition of management interfaces for the integration of QKD in disaggregated network control plane architectures, in particular with Software-Defined Networking (SDN). It defines abstraction models and workflows between a SDN-enabled QKD node and the SDN controller, including resource discovery, capabilities dissemination and system configuration operations. Application layer interfaces and quantum-channel interfaces are out of scope.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] IETF RFC 6020 (October 2010): "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)".

[2] IETF RFC 7950 (August 2016): "The YANG 1.1 Data Modeling Language".

[3] IETF RFC 6241 (June 2011): "Network Configuration Protocol (NETCONF)".

[4] IETF RFC 8040 (January 2017): "RESTCONF Protocol".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GR QKD 007: "Quantum Key Distribution (QKD); Vocabulary".

3 Definitions of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

NOTE: Where possible, the definitions from ETSI GR QKD 007 [i.1] are used.

entity: set of hardware, software or firmware components providing specific functionalities

QKD application: entity consuming QKD-derived keys from the key management system

NOTE: They can be either external applications (similar to SAE, see below) or internal applications running in the QKD system.

QKD-derived key: secret key derived from QKD system(s) operating QKD protocol(s) over a QKD link

QKD interface: interface that is a high-level abstraction of a QKD system

NOTE: A QKD interface defines only attributes that are relevant from the point of view of the network. These attributes are revealed to a SDN controller to establish and manage QKD.

QKD link: set of active and/or passive components that connect a pair of QKD modules to enable them to perform QKD and where the security of symmetric keys established does not depend on the link components under any of the one or more QKD protocols executed

QKD module: set of hardware, software or firmware components contained within a defined cryptographic boundary that implements part of one or more QKD protocol(s) to be capable of securely establishing symmetric keys with at least one other QKD module

QKD network: network comprised of two or more QKD nodes

QKD node: set of QKD modules installed in the same location within the same security perimeter

QKD protocol: operations on quantum and classical signals that allow two parties to agree on commonly shared bit strings between two ends of a QKD link that remain secret

QKD system: pair of QKD modules connected by a QKD link designed to provide Quantum Key Distribution functionality using QKD protocols

quantum channel: communication channel for transmitting quantum signals

Quantum Key Distribution (QKD): procedure involving the transport of quantum states to establish symmetric keys between remote parties using a protocol with security based on quantum entanglement or the impossibility of perfectly cloning the transported quantum states

SDN agent: entity that is responsible of managing one or more QKD Systems (through their respective QKD interfaces) within a secure location, abstracting the information of QKD resources under its control and communicating with a SDN controller for the QKD network

SD-QKD node: logical and abstracted representation of the QKD resources under the responsibility of a single SDN Agent

Secure Application Entity (SAE): entity that requests one or more keys from a Key Management System for one or more applications running in cooperation with one or more other Secure Application Entities

secure location: location assumed to be secured against access by adversaries

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ACK	Acknowledgement
API	Application Programming Interface
CRUD	Create, Read, Update and Delete
DoS	Denial of Service
HSM	Hardware Security Module
HTTP	Hypertext Transfer Protocol
ID	IDentifier

IETF	International Engineering Task Force
JSON	JavaScript Object Notation
NBI	North Bound Interface
NMS	Network Management System
PHYS	Physical link
QKD	Quantum Key Distribution
QoS	Quality of Service
RFC	Request For Comments
SAE	Secure Application Entity
SDN	Software-Defined Networking
SD-ONC	Software-Defined Optical Network Controller
SD-QKD	Software-Defined Quantum Key Distribution
SD-QNC	Software-Defined Quantum Network Controller
SKR	Secure Key generation Rate
SSH	Secure Shell
TLS	Transport Layer Security
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UUID	Universally Unique Identifier
XML	Extensible Markup Language
YANG	Yet Another Next Generation

4 Software-Defined Quantum Key Distribution

4.1 Introduction

The parametrization and modelling defined in the present document relates to the management interface of QKD modules (one or multiple) that connects them to a SDN controller. The requirements for such an interface and the further integration is described as a YANG model and as associated workflows for the main functional use cases (see Annex A). This architectural design permits a controller to centrally orchestrate the QKD resources to optimize the key allocation per link based on demands and automate the creation of either direct (physically connected through an uninterrupted quantum channel) or virtual (multi-hop-based) QKD links, where the keys are relayed from one hop (direct QKD link) to the next in the chain connecting the initial with the final points. The workflows described in Annex A are thought to be implemented by using any of the well-accepted network management protocols used in SDN architectures, which are based on YANG information models for their internal data structures. However, it is out of the scope of the present document to define which specific protocol, data structures or specific implementation is chosen to carry the YANG-structured information defined in the present document. These specifics are left aside to permit some flexibility during the system design and implementation phases.

In addition, this YANG model is designed to be a base or core model for the integration of QKD technologies in operator's management architectures, but it is not closed for experimentation and for further extensions, as YANG provides such flexibility to easily integrate new capabilities inside a given model. Future revisions of the present document may include additional parameters.

4.2 SD-QKD node

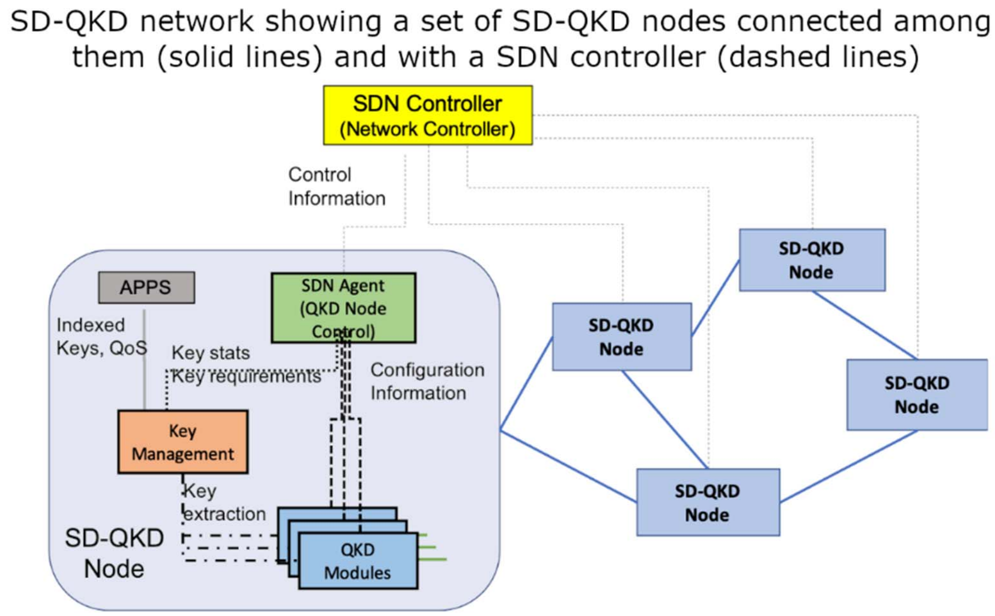
A Software-Defined Quantum Key Distribution (SD-QKD) node is an aggregation of one or multiple QKD modules that interface with a SDN controller using standard protocols (i.e. it is SDN-enabled). The connection between node and controller allows information to be retrieved from the QKD domain and dynamically and remotely configure the behaviour of the QKD systems to create, remove or update key associations (either through a quantum channel, or via multi-hop) between remote secure locations. A SD-QKD node shall be compliant with some specific requirements:

- A SD-QKD node shall comprise at least one QKD module, exposed to the controller as a QKD interface.
- It may comprise multiple QKD modules, creating an abstraction of a single node with multiple interfaces.
- It shall be located within a secure location.
- A single location may comprise one or multiple SD-QKD nodes.

- A SD-QKD node shall contain a key management system aggregating the key material from the different key associations. The key management system may be implemented using multiple logical key stores to distinguish groups of applications.
- It shall provide a single access for applications to retrieve keys from the key store via an API.
- It shall connect to (at least) one SDN controller, via standard protocols and mechanisms to enable discovery, control and telemetry and statistics streaming.
- It should expose applications information to the controller, for discovery purposes and to better optimize the utilization of QKD keys.
- It should expose QKD interface with QKD system information to the SDN controller, and allow to configure some parameters of the systems to create the quantum channel.
- It should expose information of the key associations (links) with other SD-QKD systems.
- It should expose the classical channel requirements for each of the systems within the node (e.g. attenuation, supported wavelengths, etc.).

The modelling defined in the next clauses provides an abstracted view of the QKD domain. It can abstract the QKD systems within a secure location as interfaces of a network element. This network element, the SD-QKD node, is able to communicate with its neighbours and with the central controller to create end-to-end services, or key associations. When possible (enough QKD systems, reachability over the physical media), these associations are created over a direct quantum channel. In other cases, a multi-hop link or key association is created granting a fully connected QKD network. Also, the information exchanged across the control plane is not critical (e.g. keys are not forwarded to the controller). Therefore, the introduction of the SDN paradigm for QKD networks, should not imply any further security risks different from the already known in trusted node QKD networks (e.g. DoS attacks). In particular, the aim of this abstraction model is to:

- Enable centralized management of the QKD resources based on demands, capabilities and network (quantum and classical) status.
- Aggregate different QKD systems within a secure perimeter under a single key management, to better detect demands and provision the necessary links or key associations.
- Reduce the complexity of operating separately all the QKD modules within a secure location and handling statistics from the QKD systems.
- Abstract the complexity of managing low level parameters and behaviour of each QKD system and technology, as each node can take the responsibility of low-level configurations.
- Optimize the configuration and the distribution of QKD links in the QKD network to cope with all demands, based on application's QoS information and generation rate statistics of each link.
- Coordinate quantum and classical channels (the configuration of the optical network), whether they share the same physical media or not, to enhance the performance of the QKD systems.



NOTE: The SD-QKD Nodes are connected among them (solid lines) and with the SDN controller (dashed lines). One of the nodes is detailed to show a typical set of components and the type of information that flows among them. In particular the SDN Agent that connects the node with the SDN controller is shown. The present document deals with this connection. See the text for additional information.

Figure 1: Depiction of a SD-QKD network showing a set of SD-QKD nodes

Figure 1 shows, in a high-level design, a SD-QKD network as a set of connected nodes under the control of the SDN controller. One of the nodes is shown in more detail with the fundamental components which are required to build a SD-QKD node in order to illustrate the typical flow of information between components. Note that the figure is for illustrative purposes and does not imply a mandatory node structure. The Applications are included as part of the node to illustrate that they are contained in the same security perimeter. At the hardware level, the SD-QKD system shall comprise at least one QKD module (in the example figure, there are three modules). These modules are used to physically connect the SD-QKD node to other remote peers through a quantum link, composing a QKD system for key generation purposes (note that the scheme can be easily extended to include other services allowed by the quantum device, like entanglement distribution). The generated keys are pushed (or extracted) to a key management system, which is responsible for maintaining and distributing them. The key management system registers incoming applications and their QoS, and monitors the real demands of each of them. It also exposes the parameters needed to monitor the utilization of the QKD-derived keys for each link. This information allows to optimize the planning of the QKD network.

The following clauses describe the different data structures (YANG grouping) to be handled by the SD-QKD node and the SDN controller. The YANG data model for the SD-QKD node is divided in four main structures (groupings): SD-QKD node capabilities, QKD applications, QKD links (or key associations) and QKD interfaces (or systems). In addition, YANG notifications are also included for server (node) to client (controller) communications.

4.3 SD-QKD node capabilities

The SD-QKD node capabilities structure contains essential parameters to expose to the SDN controller its support for some basic functionalities. An example is the capability (or policy) of exporting statistics about the key usage, or if the node is allowed (capable) of forwarding keys (key relay) in a multi-hop environment. Other submodules could also include their own capabilities, while this clause just refers to the capabilities of the node as a whole.

Table 1: SD-QKD node capabilities

Name	Type	Details	Description
link_stats_support	boolean	Default: true	If true, this node exposes link-related statistics (secure key generation rate-SKR, link consumption, status, QBER).
application_stats_support	boolean	Default: true	If true, this node exposes application related statistics (application consumption, alerts).
key_relay_mode_enable	boolean	Default: true	QKD node support for key relay mode services.

4.4 QKD Interfaces

As described in the introductory clause, QKD interfaces are an abstraction of the QKD systems which are contained within a secure location as part of a SD-QKD nodes. This abstraction allows a SDN controller to identify all the QKD systems within a location and aggregate them as a single network element with multiple interfaces (e.g. as a switch or a router, with very different capabilities).

Due to interoperability issues, the current version of the model shall specify the QKD technology implemented by the device and the vendor and model, as mix-matching different QKD modules in the current state of development will lead to inoperative links with no key generation.

The QKD interfaces within a SD-QKD node shall include the following parameters:

Table 2: Parameters of QKD interfaces

Name	Type	Details	Description
qkdi_id (interface ID)	ietf_yang_types:uuid	None	Interface id. It is described as a locally unique number, which is globally unique when combined with the SD-QKD node ID.
capabilities	container	None	Capabilities of the QKD system (interface).
capabilities/ role_support	etsi-qkdn-types: QKD-ROLE-TYPES	None	QKD node support for key relay mode services.
capabilities/ wavelength_range	etsi-qkdn-types: QKD-ROLE-TYPES	None	Range of supported wavelengths (nm) (multiple if it contains a tunable laser).
capabilities/ max_absorption	uint32	None	Maximum absorption supported (in dB).
model	string	None	Device model (vendor/device).
type	etsi-qkdn-types: QKD-TECHNOLOGY-TYPES	None	Interface type (QKD technology).
att_point	container	None	Interface attachment point to an optical switch.
att_point/ device	string	None	Unique ID of the optical switch (or passive component) to which the interface is connected.
att_point/ port	uint32	None	Port ID from the device to which the interface is connected.

4.5 QKD Key Association Links

A QKD Key Association Link is a logical key association between two remote SD-QKD nodes. These links associations can be of two different types: direct (also called physical), if there is a direct quantum channel through which keys are generated i.e. a physical QKD link connecting the pair of QKD modules, or virtual if keys are forwarded (key relay) through several SD-QKD -trusted- nodes to form an end-to-end key association. i.e. there is no direct quantum channel connecting the end points, and a set of them have to be concatenated such that for each a secret key is produced and then used to relay a key from the initial to the end point in a multi-hop way.

Any new key association link created in a SD-QKD node has to be tracked, labelled and isolated from other links. Virtual links are also registered as internal applications, as they make use of QKD-derived keys from other QKD key association links for the key transport.

The information exported to the SDN controller should be kept minimal but sufficient to allow analysis and optimization of the deployed links and applications, while other crucial information (e.g. the QKD-derived keys) shall be kept within the SD-QKD node security perimeter. The parameters that define a QKD key association link within the SD-QKD node abstraction are:

Table 3: Parameters of a QKD key association link

Name	Type	Details	Description
link_id	ietf_yang_types:uuid	None	Unique ID of the QKD link (key association).
enable	boolean	Default true	This value allows to enable of disable the key generation process for a given link.
local/qkd_node	ietf_yang_types:uuid	None	Unique ID of the local SD-QKD node.
local/interface	uint32	None	Interface used to create the key association link.
remote/qkd_node	ietf_yang_types:uuid	None	Unique ID of the remote QKD node. This value is provided by the SDN controller when the key association link request arrives.
remote/interface	uint32	None	Interface used to create the link.
type	etsi-qkdn-types: QKD-LINK-TYPES	None	Key Association Link type: Virtual (multi-hop) or Direct.
state	etsi-qkdn-types: LINK-STATUS-TYPES	None	Status of the QKD key association link.
applications	List: ietf_yang_types:uuid	None	Applications which are consuming keys from this key association link.
prev_hop	ietf_yang_types:uuid	(if type=VIRTUAL)	Previous hop in a multihop/virtual key association link config.
next_hop	Leaf-list: ietf_yang_types:uuid	(if type=VIRTUAL)	Next hop(s) in a multihop/virtual key association link config. Defined as a list for multicast over shared sub-paths.
bandwidth	uint32	(if type=VIRTUAL)	Required bandwidth (in bits per second) for that key association link. Used to reserve bandwidth from the physical QKD links to support the virtual key association link as an internal application.
channel_att	uint8	(if type=PHYS)	Expected attenuation on the quantum channel (in dB) between the Source/qkd_node and Destination/qkd_node.
wavelength	etsi-qkdn-types: wavelength	(if type=PHYS)	Wavelength (in nm) to be used for the quantum channel. If the interface is not tunable, this configuration could be bypassed.
qkd_role	etsi-qkdn-types: QKD-ROLE-TYPES	(if type=PHYS)	Transmitter/receiver mode for the QKD module. If there is no multi-role support, this could be ignored.
Performance/ expected_consumption	uint32	Config false	Sum of all the application's bandwidth (in bits per second) that are on this particular key association link.
Performance/skr	uint32	Config false	Secret key rate generation (in bits per second) of the key association link.
Performance/eskr	uint32	Config false	Effective secret key rate (in bits per second) generation of the key association link available after internal consumption.
Performance/ phys_perf	list	(if type=PHYS) Config false Key "type";	List of physical performance parameters.
Performance/ phys_perf/ type	etsi-qkdn-types: PHYS-PERF-TYPES	(if type=PHYS) config false;	type of the physical performance value to be exposed to the controller.