International Telecommunication Union

ITU-T Technical Report

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(24 November 2021)

ITU-T Focus Group on Quantum Information Technology for Networks (FG QIT4N)

FG QIT4N D2.3

Quantum key distribution network protocols: Key management layer, QKDN control layer and QKDN management layer

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

Quantum information technology (QIT) is a class of emerging technology that improves information processing capability by harnessing principles of quantum mechanics which is expected to have a profound impact to ICT networks.

The ITU Telecommunication Standardization Advisory Group established the ITU-T Focus Group on Quantum Information Technology for Networks (FG QIT4N) in September 2019 to provide a collaborative platform to study the pre-standardization aspects of QITs for ICT networks.

The procedures for establishment of focus groups are defined in Recommendation ITU-T A.7.

Deliverables of focus groups can take the form of technical reports, specifications, etc., and aim to provide material for consideration by the parent group in its standardization activities. Deliverables of focus groups are not ITU-T Recommendations.

FG QIT4N concluded and adopted all its Deliverables as technical reports on 24 November 2021.

Number	**Title**
FG QIT4N D1.1	QIT4N terminology: Network aspects of QITs
FG QIT4N D1.2	QIT4N use cases: Network aspects of QITs
FG QIT4N D1.4	Standardization outlook and technology maturity: Network aspects of QITs
FG QIT4N D2.1	QIT4N terminology: QKDN
FG QIT4N D2.2	QIT4N use cases: QKDN
FG QIT4N D2.3	QKDN protocols: Quantum layer
FG QIT4N D2.3	QKDN protocols: Key management layer, QKDN control layer and QKDN management layer
FG QIT4N D2.4	QKDN transport technologies
FG QIT4N D2.5	QKDN standardization outlook and technology maturity

The FG QIT4N Deliverables are available on the ITU webpage, at https://www.itu.int/en/ITU-T/focusgroups/qit4n/Pages/default.aspx.

For more information about FG QIT4N and its deliverables, please contact tsbfgqit4n@itu.int.

Technical Report ITU-T FG QIT4N D2.3

Quantum key distribution network protocols: Key management layer, QKDN control layer and QKDN management layer

Summary

This technical report is a deliverable of the ITU-T Focus Group on Quantum Information Technology for Networks (FG QIT4N) which studies classical communication protocols in the quantum key distribution network (QKDN) which include protocols with respect to the key management layer, QKDN control layer, and QKDN management layer.

The QKDN protocols are classified into different layers according to main functions of each layer. Representative operational procedures and corresponding message parameters are given for some protocols.

Note

This is an informative ITU-T publication. Mandatory provisions, such as those found in ITU-T Recommendations, are outside the scope of this publication. This publication should only be referenced bibliographically in ITU-T Recommendations.

Keywords

QKD; quantum key distribution; quantum key distribution network; protocols.

Chief editor:	Hongyu Wu QuantumCTek Co., Ltd. China	Email: hongyu.wu@quantum-info.com
Co-editors:	Hao Qin National Quantum-Safe Network | National University of Singapore Singapore	Email: hao.qin@nus.edu.sg
	Peng Huang XT Quantech | Shanghai Jiao Tong University China	Email: huang.peng@xtquantech.com
	Kaoru Kenyoshi National Institute of Information and Communications Technology (NICT) Japan	Email: kaoru.kenyoshi@nict.go.jp
	Eldar Gayfutdinov PJSC "Rostelecom" Russian Federation	Email: Eldar.Gayfutdinov@RT.RU

Acknowledgments

The editors express their appreciation to all the contributors of this report and all participants of Working Group 2 of the Focus Group on Quantum Information Technology for Networks (FG QIT4N) for their invaluable inputs, thorough review and all comments provided during the development of this report.

Table of Contents

Technical Report ITU-T FG QIT4N D2.3

Quantum key distribution network protocols: Key management layer, QKDN control layer and QKDN management layer

1 Scope

This Technical Report studies communication protocols related to key management layer, QKDN control layer, and QKDN management layer in the QKDN.

In particular, the scope of this draft technical report covers:

- Protocols with respect to key management layer;
- Protocols with respect to QKDN control layer;
- Protocols with respect to QKDN management layer; and
- Suggestions for future work.

2 References

[ITU-T X.1710] Recommendation ITU-T X.1710 (2020), *Security framework for quantum key distribution networks.*

[ITU-T X.1712] Recommendation ITU-T X.1712 (2021), *Security requirements and measures for quantum key distribution networks – key management.*

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

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

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

[ITU-T Y.3803] Recommendation ITU-T Y.3803 (2020), *Quantum key distribution networks – Key management.*

[ITU-T Y.3804] Recommendation ITU-T Y.3804 (2020), *Quantum key distribution networks – Control and Management.*

3 Terms and definitions

3.1 Terms defined elsewhere

This Technical Report uses the following terms defined elsewhere:

3.1.1 access control [b-ITU-T X.800]: The prevention of unauthorized use of a resource, including the prevention of use of a resource in an unauthorized manner.

3.1.2 classical channel [b-ETSI GR QKD 007]: Communication channel that is used by two communicating parties for exchanging data encoded in a form that may be non-destructively read and fully reproduced.

3.1.3 cryptographic hash function [b-ETSI GR QKD 007]: Computationally efficient function that maps binary strings of arbitrary length to binary strings of fixed length, such that it is computationally infeasible to invert it, or to find two distinct values that hash into a common value.

3.1.4 hash value [b-ETSI GS QKD 008]: Output of a cryptographic hash function.

3.1.5 key data [ITU-T Y.3803]: Random bit strings, which are used as a cryptographic key.

3.1.6 key management [ITU-T Y.3800]: All activities performed on keys during their life cycle starting from their reception from the quantum layer, storage, formatting, relay, synchronization, authentication, to supply to a cryptographic application and deletion or preservation depending on the key management policy.

3.1.7 key management agent (KMA) [ITU-T Y.3802]: A functional element to manage keys generated by one or multiple quantum key distribution (QKD) modules in a QKD node (trusted node).

NOTE – KMA acquires keys from one or multiple QKD modules, synchronizes, resizes, formats, and stores them. It also relays keys through key management agent (KMA) links.

3.1.8 key management agent-key (KMA-key) [ITU-T Y.3803]: Key data stored and processed in a key management agent (KMA), and securely shared between a KMA and a matching KMA.

3.1.9 key management agent link (KMA link) [ITU-T Y.3802]: A communication link connecting key management agents (KMAs) to perform key relay and communications for key management.

3.1.10 key manager (KM) [ITU-T Y.3800]: A functional module located in a quantum key distribution (QKD) node to perform key management in the key management layer.

3.1.11 key relay [ITU-T Y.3800]: A method to share keys between arbitrary quantum key distribution (QKD) nodes via intermediate QKD node(s).

3.1.12 key supply [ITU-T Y.3800]: A function providing keys to cryptographic applications.

3.1.13 key supply agent (KSA) [ITU-T Y.3802]: A functional element to supply keys to a cryptographic application, being located between a key management agent (KMA) and the cryptographic application.

NOTE – Application interfaces for cryptographic applications are installed into the key supply agent (KSA). The KSA synchronizes keys and verifies their integrity via a KSA link before supplying them to the cryptographic application.

3.1.14 key supply agent-key (KSA-key) [ITU-T Y.3803]: Key data stored and processed in a key supply agent (KSA), and securely shared between a KSA and a matching KSA.

3.1.15 key supply agent link (KSA link) [ITU-T Y.3802]: A communication link connecting key supply agents (KSAs) to perform key synchronization and integrity verification.

3.1.16 quantum channel [b-ETSI GR QKD 007]: Communication channel for transmitting quantum signals.

3.1.17 quantum key distribution (QKD) [b-ETSI GR QKD 007]: Procedure or method for generating and distributing symmetrical cryptographic keys with information theoretical security based on quantum information theory.

3.1.18 quantum key distribution link (QKD link) [ITU-T Y.3800]: A communication link between two quantum key distribution (QKD) modules to operate the QKD.

NOTE – A QKD link consists of a quantum channel for the transmission of quantum signals, and a classical channel used to exchange information for synchronization and key distillation.

3.1.19 quantum key distribution module (QKD module) [ITU-T Y.3800]: A set of hardware and software components that implements cryptographic functions and quantum optical processes, including quantum key distribution (QKD) protocols, synchronization, distillation for key generation, and is contained within a defined cryptographic boundary.

NOTE – A QKD module is connected to a QKD link, acting as an endpoint module in which a key is generated. These are two types of QKD modules, namely, the transmitters and the receivers.

3.1.20 quantum key distribution network (QKDN) [ITU-T Y.3800]: A network comprised of two or more quantum key distribution (QKD) nodes connected through QKD links.

NOTE – A QKDN allows sharing keys between the QKD nodes by key relay when they are not directly connected by a QKD link.

3.1.21 quantum key distribution network controller (QKDN controller) [ITU-T Y.3800]: A functional module, which is located in a quantum key distribution (QKD) network control layer to control a QKD network.

3.1.22 quantum key distribution network manager (QKDN manager) [ITU-T Y.3800]: A functional module, which is located in a quantum key distribution (QKD) network management layer to monitor and manage a QKD network.

3.1.23 quantum key distribution node (QKD node) [ITU-T Y.3800]: A node that contains one or more quantum key distribution (QKD) modules protected against intrusion and attacks by unauthorized parties.

NOTE – A QKD node can contain a key manager (KM).

3.1.24 quality of service (QoS) [b-ITU-T Q.1743]: The collective effect of service performances, which determine the degree of satisfaction of a user of a service. It is characterized by the combined aspects of performance factors applicable to all services, such as:

– service operability performance;
– service accessibility performance;
– service retainability performance;
– service integrity performance; and
– other factors specific to each service.

3.1.25 user network [ITU-T Y.3800]: A network in which cryptographic applications consume keys supplied by a quantum key distribution (QKD) network.

4 Abbreviations and acronyms

This Technical Report uses the following abbreviations and acronyms:

API	Application Programming Interface
FCAPS	Fault, Configuration, Accounting, Performance and Security
ID	Identifier
KM	Key Manager
KMA	Key Management Agent
KSA	Key Supply Agent
QK	Quantum Key
QKD	Quantum Key Distribution
QKDN	QKD Network
QoS	Quality of Service
TN	Trusted Node
TNE	Trusted Node Equipment

5 Introduction

Based on the conceptual structure of QKDN and its functional requirements identified in [ITU-T Y.3800] and [ITU-T Y.3801], respectively, a functional architecture model of QKDN is specified in [ITU-T Y.3802]. Figure 5-1 illustrates the functional elements in each layer and reference points between the elements where each reference point could be regarded as a logical interface with underlying communication protocols within or between layers.

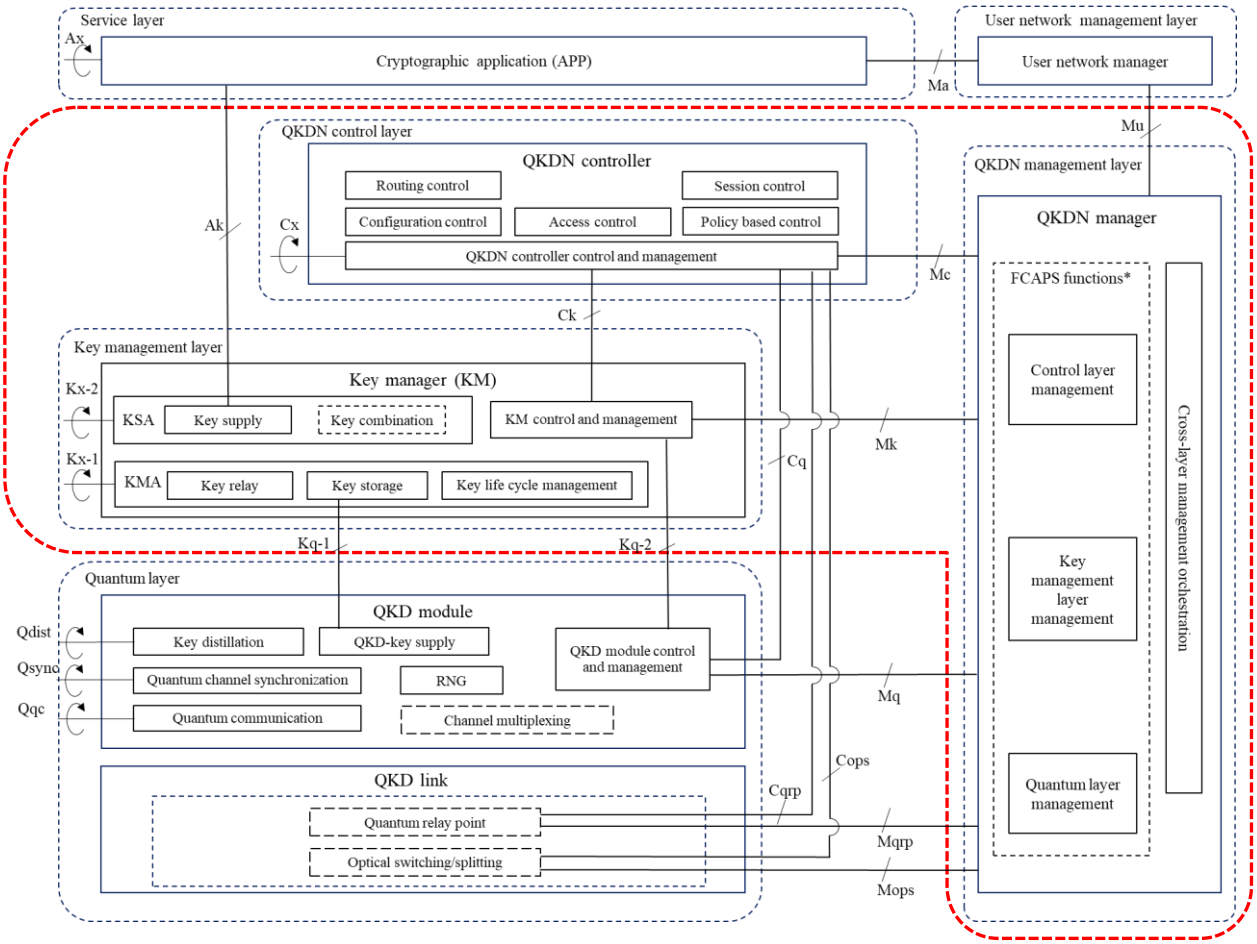


Figure 5-1 – A functional architecture model of QKDN

Since communication protocols are essential issues for today's networks, in which different vendors can provide compatible equipment or systems based on unified interfaces and protocols, the study of protocols in this report will try to help the implementation of QKDN with the same expectation.

Contrary to the quantum layer in the QKDN, the key management layer, the QKDN control layer, and the QKDN management layer do not deal with quantum signals. Thus, these three layers are grouped together in this report to study the specific communication protocols related to them in the QKDN.

Basic operational procedures such as service provisioning and system initialization, key generation, key request and supply, key relay, and key relay rerouting control procedures are discussed in [ITU-T Y.3802]; while basic operations for key management as well as typical examples of operational procedures for control, management and orchestration of QKDN are also respectively given in [ITU-T Y.3803] and [ITU-T Y.3804]. Specific operations and corresponding message parameters may, however, vary with different scenarios or implementations, thus, a protocol level discussion is necessary to help achieve compatibility as much as possible.

The QKDN protocols in this report are classified into three layers according to their main functions i.e., the key management layer, the QKDN control layer and the QKDN management layer. The representative operational procedures and corresponding message parameters are given for some protocols as examples and a summary of key findings and suggestions for further standardization and industrialization are provided at the end.

The aim of this report is to assist interested parties in recognizing QKDN protocol issues, to exchange information and best practices through peer learning and knowledge dissemination processes, and to identify possible standardization requirements.

6 QKDN protocol framework

A work item on the protocol framework of QKDN [b-ITU-T Q.QKDN_profr] has been initiated by ITU-T Study Group 11 with the objective of providing an overview of signalling and protocols for QKDN and to discuss signalling requirements and protocol suites for QKDN, wherein QKD protocols are outside its scope. Figures 6-1 and 6-2 [b-ITU-T Q.QKDN_profr] illustrate an overview of the protocol stacks at each reference point. Appropriate protocols of different network levels can be selected for each reference point.

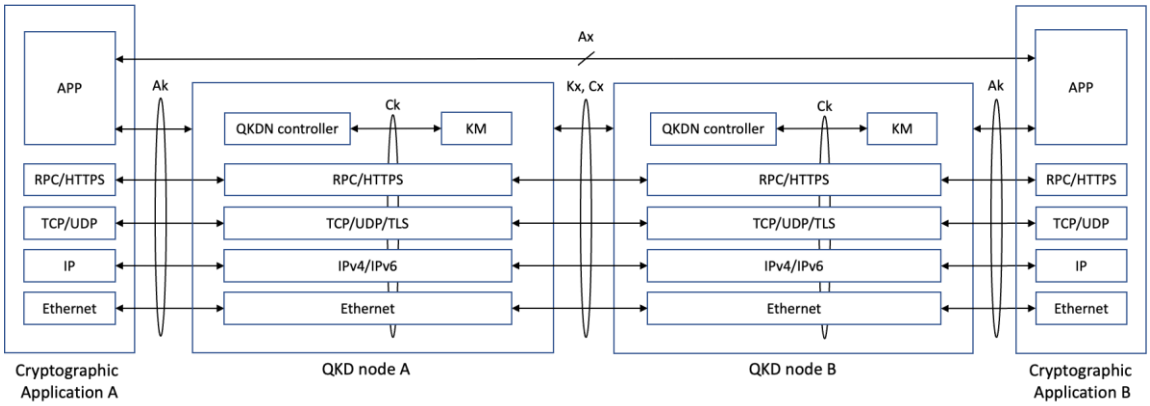


Figure 6-1 – Protocol stacks at reference points Ak, Ck, Kx and Cx

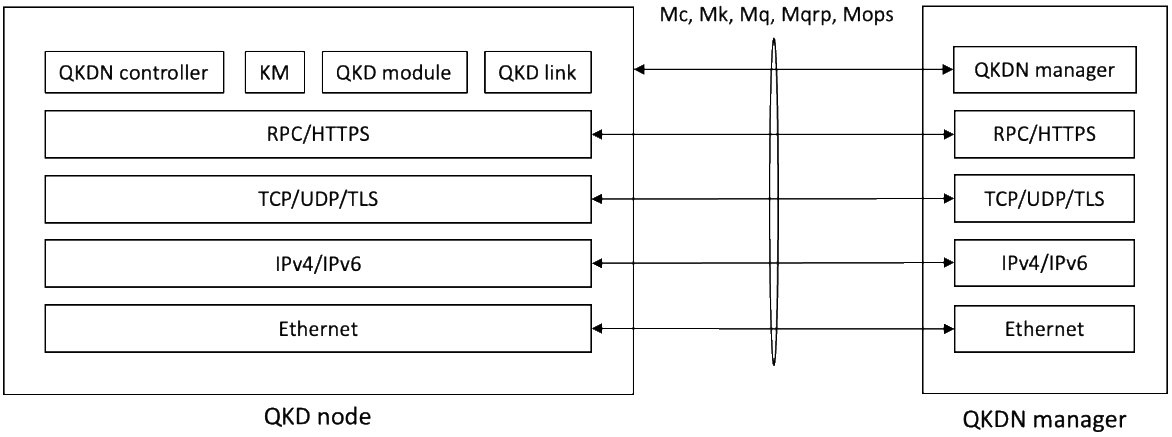


Figure 6-2 – Protocol stacks at reference points Mc, Mk, Mq, Mqrp and Mops

Interested readers can follow up with the [b-ITU-T Q.QKDN_profr] work item to have a better guidance before going into specific protocol issues.

7 Protocols with respect to key management layer

7.1 Basic operations of key management

In the following sub-clauses, basic operations of key management as discussed in [ITU-T Y.3803] are addressed, which serves as a basis for further discussions on protocol issues.

7.1.1 Generation of QKD-key in the quantum layer and supply it to KMA

In the quantum layer, a pair of QKD modules generate a pair of symmetric (identical) random bit strings in their own way based on an IT-secure protocol of QKD. The symmetric random bit string generated in the QKD module is referred to as a QKD-key. In each QKD module, metadata is generated and attached to the QKD-key, forming a key file. The pair of QKD modules then transfer the QKD-key file to the corresponding KMAs. This information flow is illustrated in Figure 7-1.

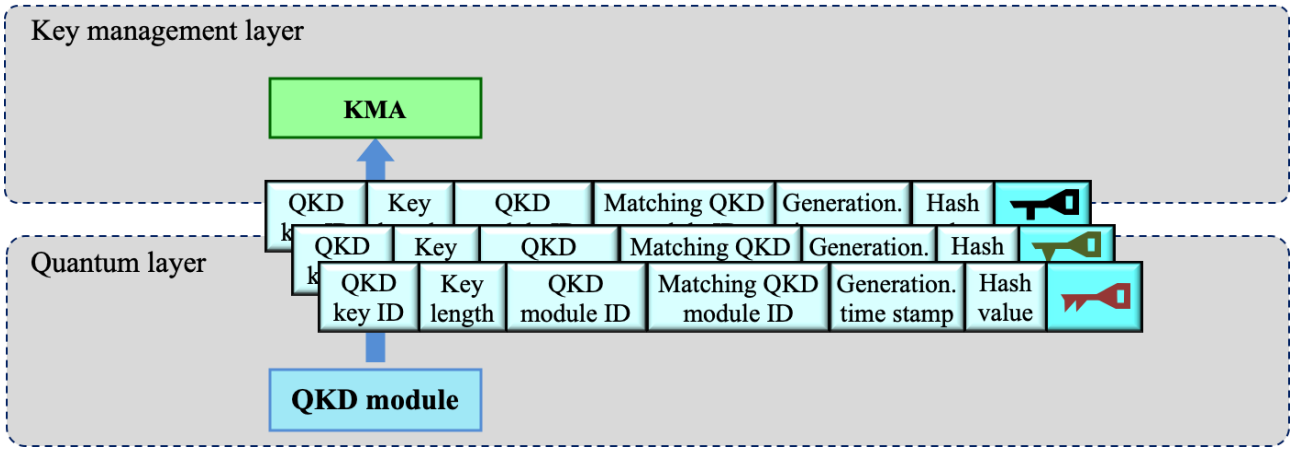


Figure 7-1 – Information flow of QKD-key from quantum layer to key management layer

7.1.2 KMA stores KMA-key in the key management layer

A KMA receives the QKD-key files from a QKD module located in the same QKD node and stores them securely when storage is necessary. The lengths of the acquired QKD-key files may differ from each other, therefore, the KMA re-formats (combines or splits) the QKD-keys into keys of a prescribed unit length, and then temporarily stores them in a buffer, see Figure 7-2.

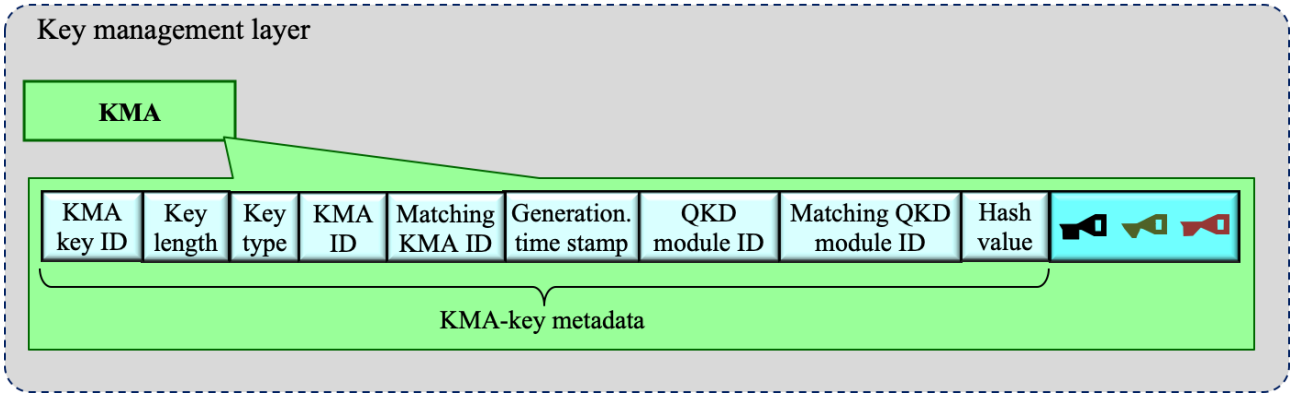


Figure 7-2 – KMA-key stored in KMA

7.1.3 Reception of key request from cryptographic application

A cryptographic application in the service layer sends a KSA a key request, see Figure 7-3. The key request from cryptographic application may include information on a required security level, depending on key supply service policy, etc. The KSA receives key requests from authorized cryptographic applications via a key supply interface.

The KSA then authenticates the cryptographic application by an appropriate means. After the KSA authenticates the cryptographic application, the KSA and the KMA in the same node are recommended to authenticate each other. After authentication, the KSA informs the KMA of the requested information.

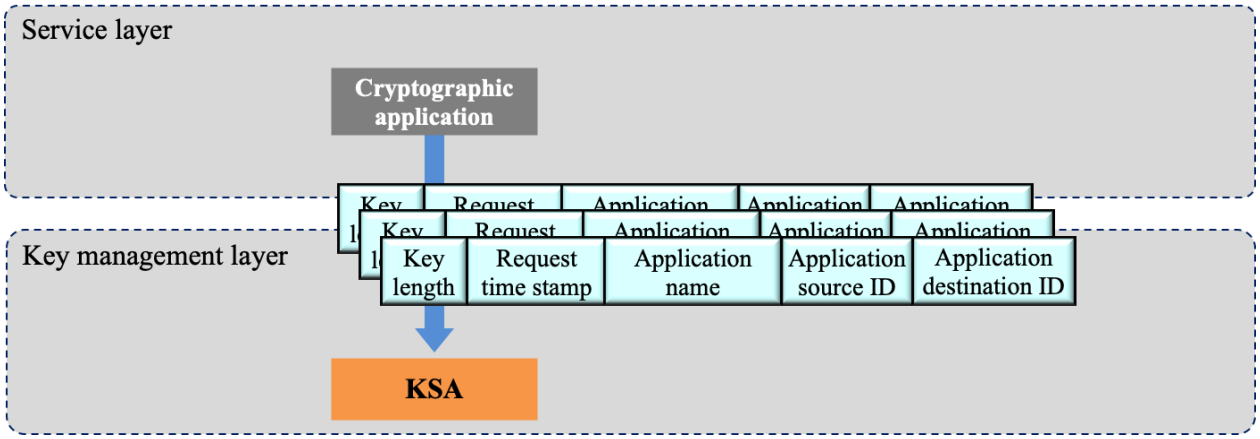


Figure 7-3 – Key request from cryptographic application to KSA

The KMAs support key relay through a key relay route between the two endpoint KMAs, employing highly secure encryption (e.g., OTP [b-Shannon]). A typical case of point-to-point key relay using OTP, which is an IT-secure protocol for ensuring confidentiality of the keys is explained in the following paragraphs. The key data and metadata of the KMA-key are exclusively ORed with the other key shared by the neighbouring pair of QKD modules in an OTP manner and are then sent from the source KMA (see Figure 7-4) to the destination KMA, thus realizing IT-secure key relay.

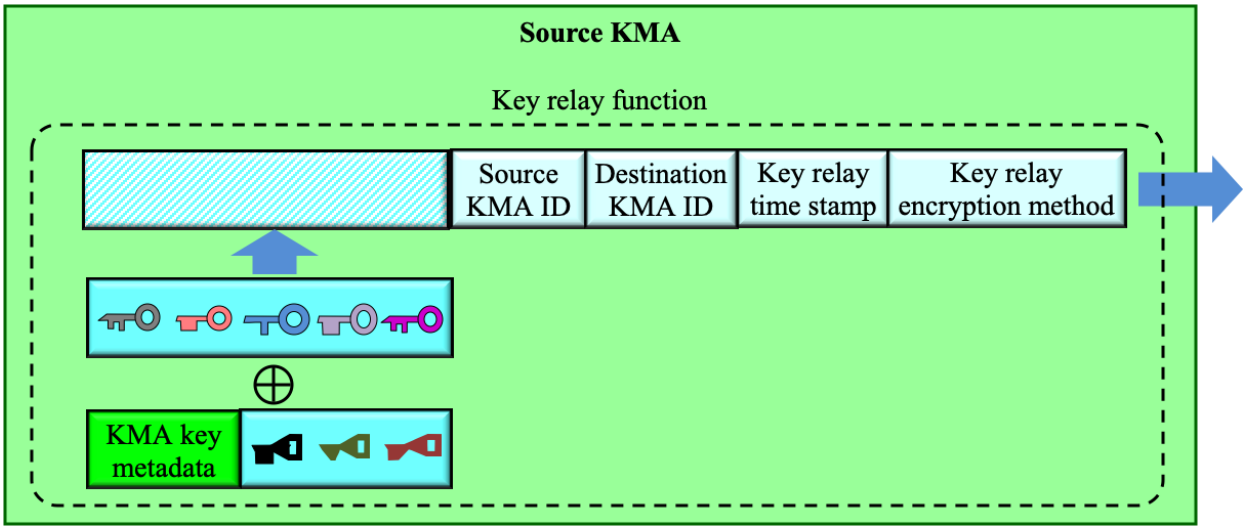


Figure 7-4 – Key encryption at the source KMA

After decryption in the destination KMA (see Figure 7-5), key relay information consisting of the source KMA, the destination KMA, and key relay time stamp is added to the source KMA-key metadata and stored in the key storage as the metadata for relayed KMA-key at the destination KMA.

When further relaying the key to the next destination node, it is encrypted just as before, including the key relay information. In this case, the key relay information is updated with the current node as the source KMA, the next destination node as the destination KMA, and new relay time stamp at the current node. This updated key relay information is added to the source KMA-key metadata and stored in the second destination KMA.

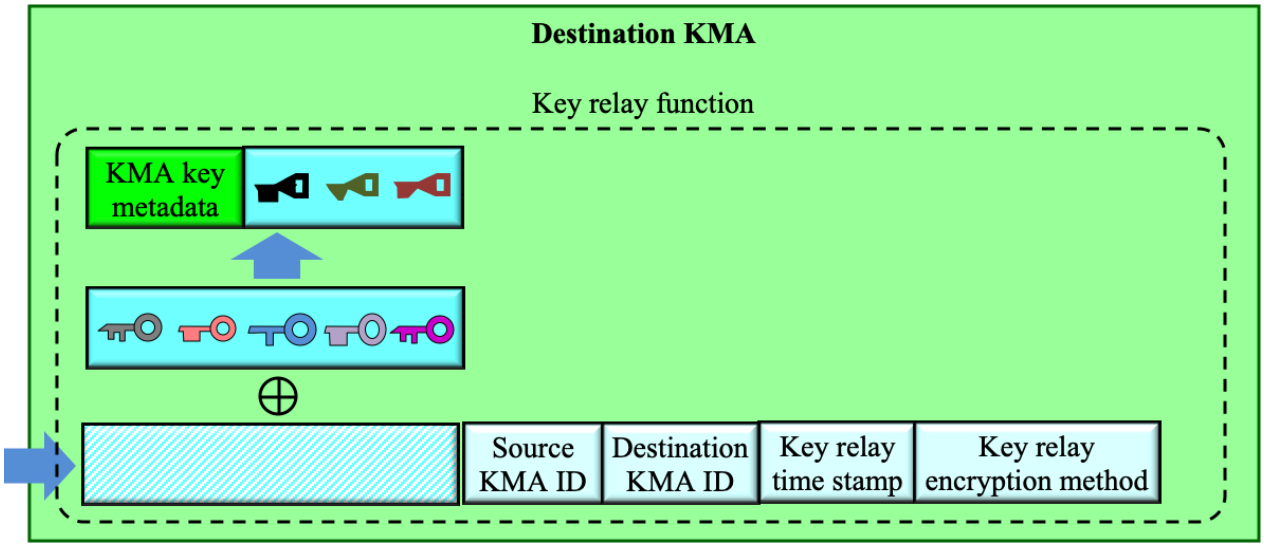


Figure 7-5 – Key decryption at the destination KMA

7.1.4 Relayed KMA-key supply from KMA to KSA

After the KSA and the KMA authenticate each other, the KSA informs the KMA of the requested information. The KMA picks up the required amount of key from the storage of KMA-key data, optionally taking into account the metadata on key relay encryption method based on the requested security level and key supply policy. The KMA then transfers this key to the KSA (see Figure 7-6) and the key received by the KSA is referred to as the KSA-key.

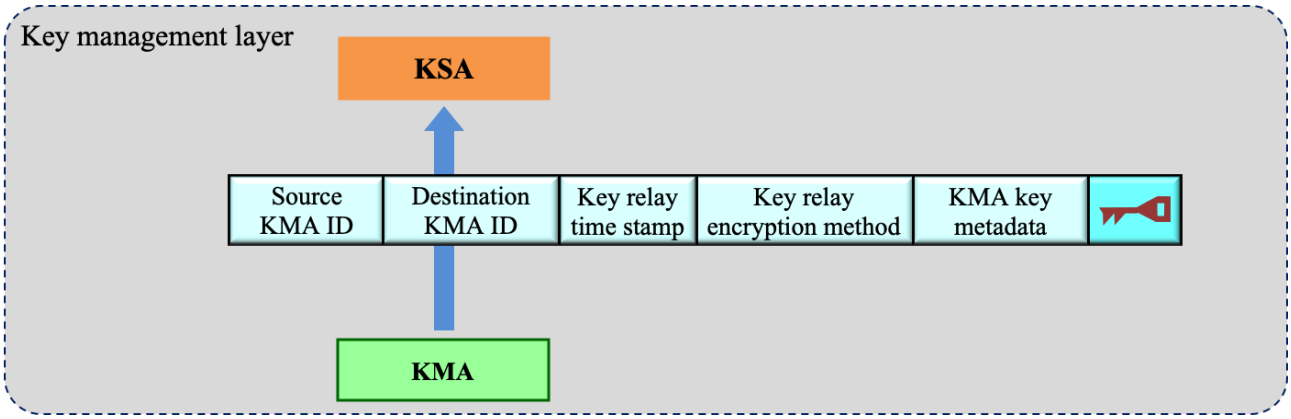


Figure 7-6 – Relayed KMA-key supply from KMA to KSA

7.1.5 KSA-key supply from KSA to cryptographic application

After the key is transferred from the KMA to the KSA, a hash value or a message authentication code is calculated from the KSA-key data in each KSA of the node pair for an individual key request. The pair of the KSAs compares their hash values or message authentication codes as well as the KSA-key ID via a KSA link, then synchronizes and authenticates the KSA key.

After the above verification is completed, the key data with the KSA-key ID is supplied to the cryptographic application via the key supply interface, see Figure 7-7. The KSA metadata is recorded in the storage of the KSA, and/or sent to the QKDN manager, for key life cycle management. The cryptographic application identifies the key data based on the KSA-key ID and uses it.

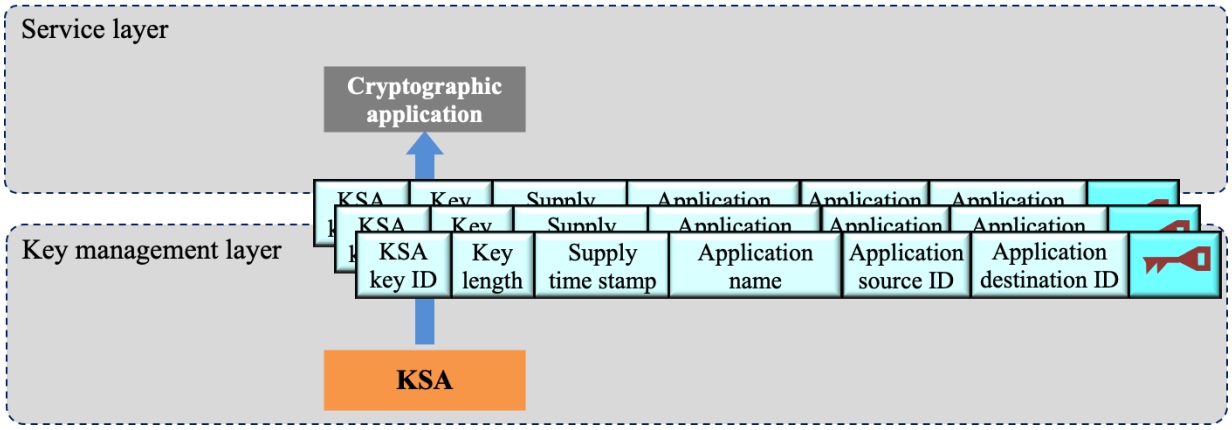


Figure 7-7 – KSA-key supply to cryptographic applications

7.1.6 Key life cycle management

Each KMA stores information on key management activities on which it works, including reception, storage, formatting, relaying, synchronization, authentication, supply, and deletion/preservation of keys. For example, each KMA collects, stores metadata in KMA-key files, and sends them to the QKDN manager while each KSA stores metadata in KSA-key files and sends them to the QKDN manager. See Figure 7-8 for an illustration on the metadata for key lifecycle management and Table 7-1 for a description of the metadata.

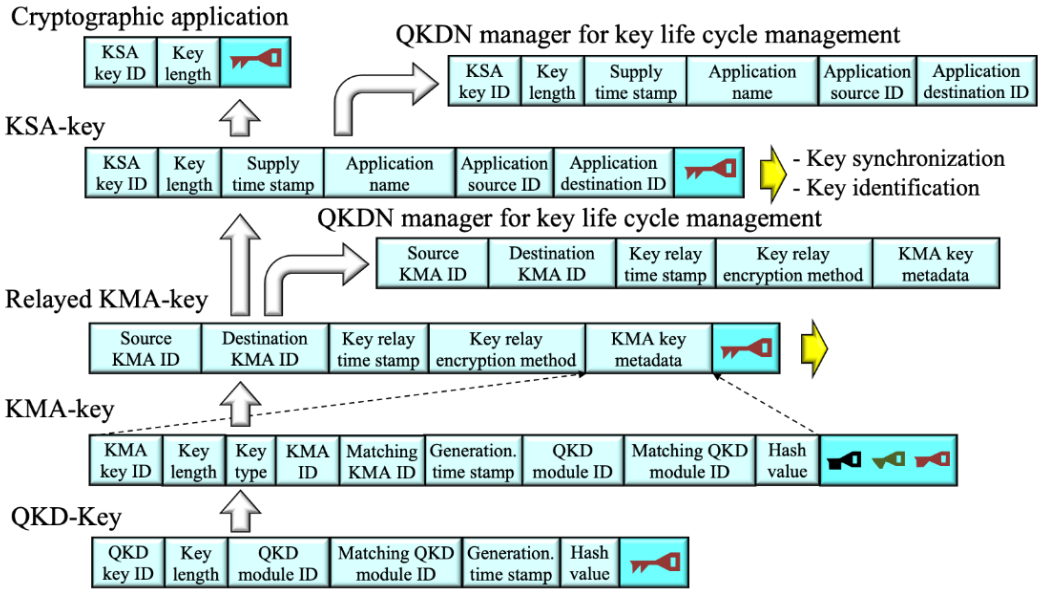


Figure 7-8 – Metadata for key lifecycle management

7.1.7 Key file format

Table 7-1 – Metadata information (basic information), see Table 1 of [ITU-T Y.3803]

Metadata	Description	M/O
(1) QKD-key		
QKD-key ID	ID of the QKD-key	M
Key length	Key length of the QKD-key	O
QKD module ID	ID of the QKD module (Alice or Bob) that generates the QKD-key	O
Matching QKD module ID	ID to identify the matching QKD module which constitutes the pair of Alice and Bob	O
Generation time stamp	Time stamp of QKD-key generation at the pair of QKD modules	O
Hash value	Hash value of the QKD-key data. (There are several options for hash function, which are discussed in other Recommendations.)	O
(2) KMA-key		
KMA-key ID	ID of the KMA-key, which is the same for the pair of keys for Alice and Bob, and unique in a QKD network. A part of the bits of the hash value generated from the names of the pair of QKD modules is often used for this ID.	M
Key length	Key length of the KMA-key	O
Key type	Index to specify either encrypting key or decrypting key	O
KMA ID	ID of the KMA that stores the KMA-key	O
Matching KMA ID	ID of the matching KMA	M
Generation time stamp	Time stamp of the KMA-key generation at the KMA	O
QKD module ID	ID to identify the QKD module which generates the QKD-key corresponding to the KMA-key data	O
Matching QKD module ID	ID to identify the matching QKD module which constitutes the pair of Alice and Bob	O
Hash value	Hash value of the KMA-key data. (There are several options for hash function, which are discussed in other Recommendations.)	O
(3) Relayed KMA-key		
Source KMA ID	ID of source KMA of the key relay	O
Destination KMA ID	ID of destination KMA of the key relay	O
Key relay time stamp	Time stamp of the key relay	O
Key relay encryption method	Encryption method used for the key relay	O
KMA-key metadata	Metadata of KMA-key of the source KMA	M
(4) KSA-key		
KSA-key ID	ID of the KSA-key	M
Key length	Key length of the KSA-key	O
Supply time stamp	Time stamp of the KSA-key supply from the KSA to a cryptographic application	O
Application name	Name of cryptographic application	O
Application source ID	Source ID of cryptographic application	O
Application destination ID	Destination ID of cryptographic application	O

O: Optional, M: Mandatory

7.2 Protocols with respect to the key management layer

The key management layer performs a series of actions on the key in and between QKD nodes in a QKDN, such as processing random bit strings from QKD modules into keys, performing key relay via trusted nodes, supplying keys upon requests from applications. The protocols with respect to the key management layer mainly include protocols in the key management layer and those via which the key management layer interacts with the quantum layer and applications. Such kind of protocols may include but not limited to the following:

- Key acquisition protocol;
- Key synchronization protocol;
- Key relay protocol;
- Key supply protocol.

In the following clauses, representative operational procedures and corresponding message parameters are given for the key supply protocol and the key relay protocol, which serve as examples to describe a protocol.

7.2.1 Key supply protocol

As identified in [ITU-T Y.3801], where a QKD node supplies keys by design or configuration to the user network, a KM in the QKD node is required to receive key requests from and supply keys to authorized cryptographic applications via a key supply interface. The key supply interface needs to have broad usability and flexible extensibility for current and future applications. With these concerns, the following key supply protocol corresponding to the key supply interface is addressed in terms of protocol operations and message parameters.

Three representative operational procedures and corresponding message parameters of authentication, key supply and key deletion are given. To guarantee the security, authentication is an important issue before key supply operations, and thus is specifically addressed in the beginning of the following sub-clauses. The key supply operations and corresponding message parameters have been studied in [b-ETSI GS QKD 004] and [b-ETSI GS QKD 014], where two kinds of APIs have been formulated to serve different scenarios. Since the messages could be in various formats, this report only focuses on basic parameters of a message to generalize the discussion and tries to accommodate Quality of Service (QoS) or status concerns in the messages for key supply. For further security concerns, the supplied keys in a KM may be requested for deletion by corresponding cryptographic applications. Hence, the key deletion is also specifically addressed below.

7.2.1.1 Protocol operations

a) Authentication

For security concerns, authentication between a KM and a cryptographic application needs to be done before key supply operations. Figure 7-9 illustrates operational procedures for authentication.

The cryptographic application sends an authentication message to the KM. Based on the received message, the KM authenticates and sends an authentication result to the cryptographic application. To further improve the security, another authentication operation can be initiated by the KM in a similar way to make mutual authentication.

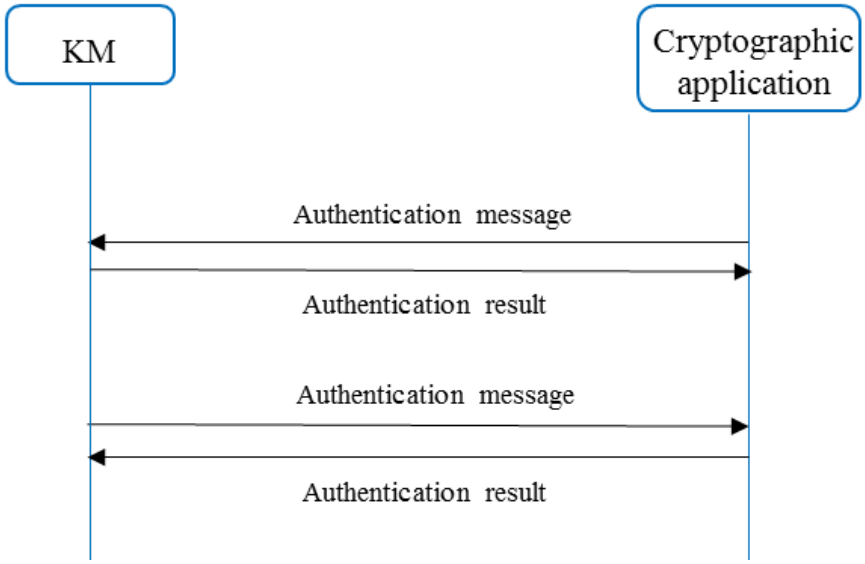


Figure 7-9 – Operational procedures for authentication

b) Key supply

After the authentication operations, the KM supplies keys upon request for the cryptographic application through key supply operations. Figure 7-10 illustrates operational procedures for key supply.

The cryptographic application sends a key request message to the KM and based on the received message, the KM sends a response message to the cryptographic application.

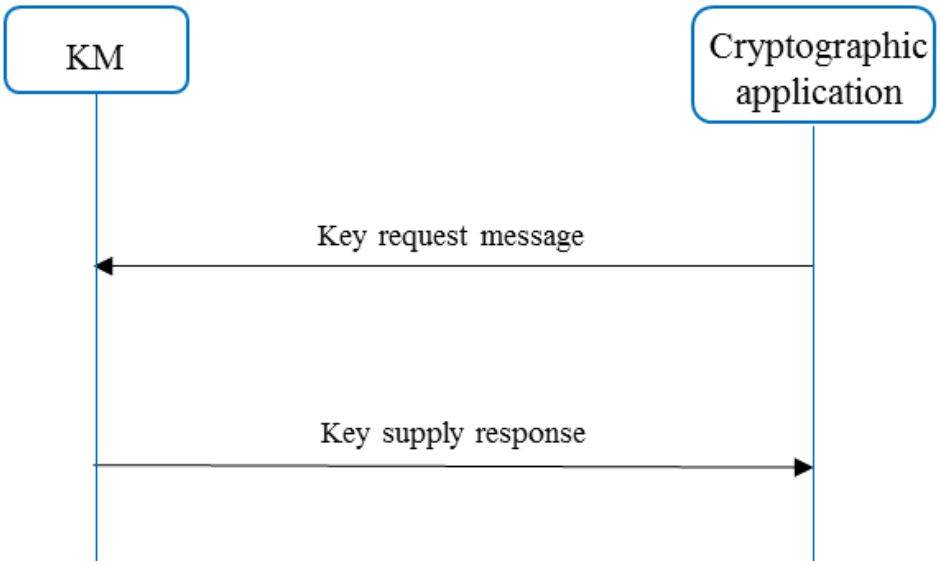


Figure 7-10 – Operational procedures for key supply

c) Key deletion

According to key management policy, the keys supplied by the KM may be requested to be deleted for security concerns, which is implemented through key deletion operations. Figure 7-11 illustrates operational procedures for key deletion.

The cryptographic application sends a key deletion message to the KM and based on the received message, the KM deletes corresponding keys and sends a key deletion result to the cryptographic application.

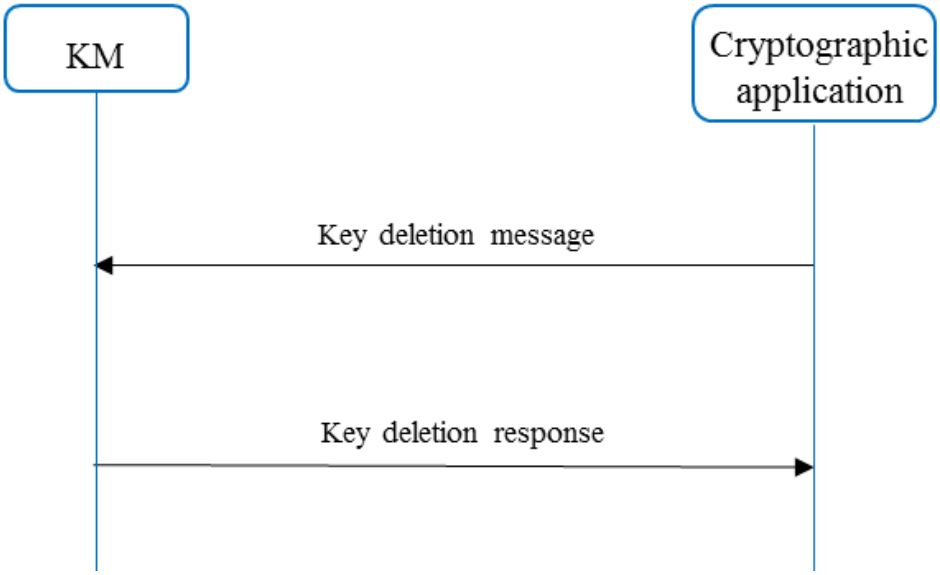


Figure 7-11 – Operational procedures for key deletion

7.2.1.2 Message parameters

The above operational procedures are realized through specific messages transmitted between the KM and the cryptographic application. In the following, basic parameters of a message are listed with corresponding descriptions. Additional information and/or examples of some parameters are listed in the "remark" column for better understanding. It should be noted that since messages could take on various formats, they are not discussed in this report.

a) Authentication

The messages for authentication transmitted between the KM and the cryptographic application are as follows, where parameters in command and response messages are listed respectively in Tables 7-2 and 7-3.

Table 7-2 – Command message for authentication

Item	Description	Remark
Message ID	A message identifier generated by a sender of the command message for authentication.	
Command code	A code that indicates the command message is used for authentication.	
Sender ID	Unique ID of the sender of the command message for authentication.	
Authentication info	Information used for authentication.	The authentication information is configurable and may be generated by algorithms.
……		

Table 7-3 – Response message for authentication

Item	Description	Remark
Message ID	A message identifier generated by a sender of the response message for authentication.	
OrigMessage ID	A message identifier received from a command message for authentication.	
Command code	A code that indicates the response message is used for authentication.	
Sender ID	Unique ID of the sender of the response message for authentication.	0x01: success 0x02: failure
Response code	A code that indicates an authentication result.	
……		

b) Key supply

The messages for key supply transmitted between the KM and the cryptographic application are as follows, where parameters in command and response messages are listed respectively in Tables 7-4 and 7-5. Some parameters may be set by configuration or negotiated by a pair of cryptographic applications in advance.

Table 7-4 – Command message for key supply

Item	Description	Remark
Message ID	A message identifier generated by a cryptographic application for the command message for key supply.	
Command code	A code that indicates the command message is used for key supply.	
Sender ID	Unique ID of the cryptographic application sending the command message for key supply.	
Source ID	An identifier of a KM to receive the command message for key supply.	ID of the cryptographic application sending the command message for key supply can be used as Source ID.
Target ID	An identifier of a KM to supply keys for a matching cryptographic application of the cryptographic application sending the command message for key supply.	ID of the matching cryptographic application can be used as Target ID.
Session ID	Unique ID of a key supply session.	
Application code	A code that indicates how keys are to be consumed.	0x01: for encryption 0x02: for decryption
Key amount	An amount of keys requested by the cryptographic application sending the command message for key supply.	
Key ID	An identifier of a key requested by the cryptographic application sending the command message for key supply.	
……		

Table 7-5 – Response message for key supply

Item	Description	Remark
Message ID	A message identifier generated by a KM for the response message for key supply.	
OrigMessage ID	A message identifier received from a command message for key supply.	
Command code	A code that indicates the response message is used for key supply.	
Sender ID	Unique ID of the KM sending the response message for key supply.	
Source ID	Same as that of the received command message for key supply.	
Target ID	Same as that of the received command message for key supply.	
Session ID	Same as that of the received command message for key supply.	
Application code	Same as that of the received command message for key supply.	
Key amount	Same as that of the received command message for key supply.	
Key size	Same as that of the received command message for key supply.	
Key ID	Same as that of the received command message for key supply.	
Count ID	An identifier that indicates the number of a serial of sub-sessions of a key supply session.	The requested amount of keys can be supplied through multiple sub-sessions within one key supply session.
Key data	Keys supplied by the KM sending the response message for key supply.	
Response code	A code that indicates the status of key supply.	0x01: success 0x02: sufficient 0x03: insufficient 0x04: status table* ……
……		

NOTE – *Detailed status of key supply may be requested by cryptographic applications, and those in Table 9 of [b-ETSI GS QKD 014] can be taken as examples for further information.

c) Key deletion

The messages for key deletion transmitted between the KM and the cryptographic application are as follows, where parameters in command and response messages are listed respectively in Tables 7-6 and 7-7.

Table 7-6 – Command message for key deletion

Item	Description	Remark
Message ID	A message identifier generated by a cryptographic application for the command message for key deletion.	
Command code	A code that indicates the command message is used for key deletion.	
Sender ID	Unique ID of the cryptographic application sending the command message for key deletion.	
Source ID	An identifier of a KM to receive the command message for key deletion.	ID of the cryptographic application sending the command message for key deletion can be used as Source ID.
Response code	A code that indicates an authentication result.	
Target ID	An identifier of a KM corresponding to a matching cryptographic application of the cryptographic application sending the command message for key deletion.	ID of the matching cryptographic application can be used as Target ID.
Session ID	Unique ID of a key supply session.	
Application code	A code that indicates how keys are consumed.	0x01: for encryption 0x02: for decryption
……		

Table 7-7 – Response message for key deletion

Item	Description	Remark
Message ID	A message identifier generated by a KM for the response message for key deletion.	
OrigMessage ID	A message identifier received from a command message for key deletion.	
Command code	A code that indicates the response message is used for key deletion.	
Sender ID	Unique ID of the KM sending the response message for key deletion.	
Source ID	Same as that of the received command message for key deletion.	
Target ID	Same as that of the received command message for key deletion.	
Session ID	Same as that of the received command message for key deletion.	
Application code	Same as that of the received command message for key deletion.	
Response code	A code that indicates a key deletion result.	0x01: success 0x02: failure
……		

7.2.2 Key relay protocol

The logical functions identified in Figure 5-1 can be implemented using various scenarios. For example, one KM can be implemented either as one physical entity with both KMA and KSA, or as two physical entities with KMA and KSA separated for each. Sometimes, even one equipment or device may be configured to have different functions.

Currently, two topologies are accepted for QKD networks – point-to-point and point-to-multipoint, using whichever is possible to build larger multi-user and multi-service networks using trusted nodes (TNs). When such networks are actively developed, increasing both in size and in the number of connected cryptographic applications, the question will arise: how can they be maintained and configured? For example, how can a shared key for a pair of cryptographic applications located in distant locations, or a shared key between the host's server and the client be generated? It is believed that such cases will require automation tools, in particular:

- Automatic generation of KMA-key ID based on cryptographic hash functions of the following parameters: {IP-address, MAC-address, port number (L4 OSI), unique ID of QKD module Tx (Alice), unique ID of KM and etc.};
- Protocol of routing key across the network.

Figure 7-12 shows an example of a point-to-point network with TNs. Generated keys from the KM 1 to KM 2 follow a single path, sequentially through TN 1, TN 2, and TN 3. Note that the generated key must have a unique identifier for a pair of cryptographic applications (red, green, and blue). KMA-key IDs are generated and approved by the KMs that cryptographic applications are connected to.

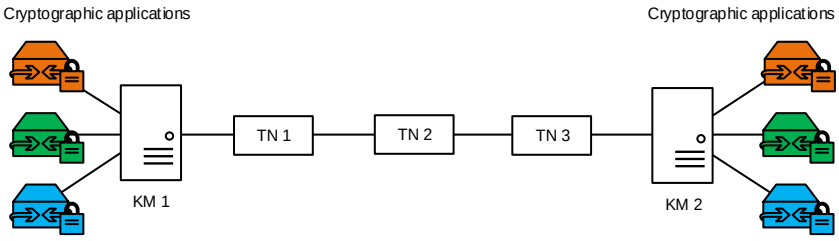


Figure 7-12 – Point-to-point network with a trusted node

Figure 7-13 shows TN 2 and TN 4 are not limited to a point-to-point connection and have several directions.

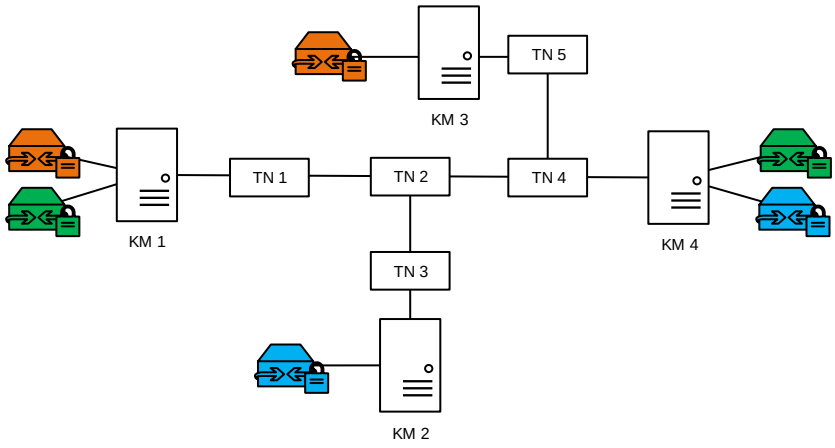


Figure 7-13 – Branched QKD network

For instance, there is a connection with the network of a third-party operator, see Figure 7-14.

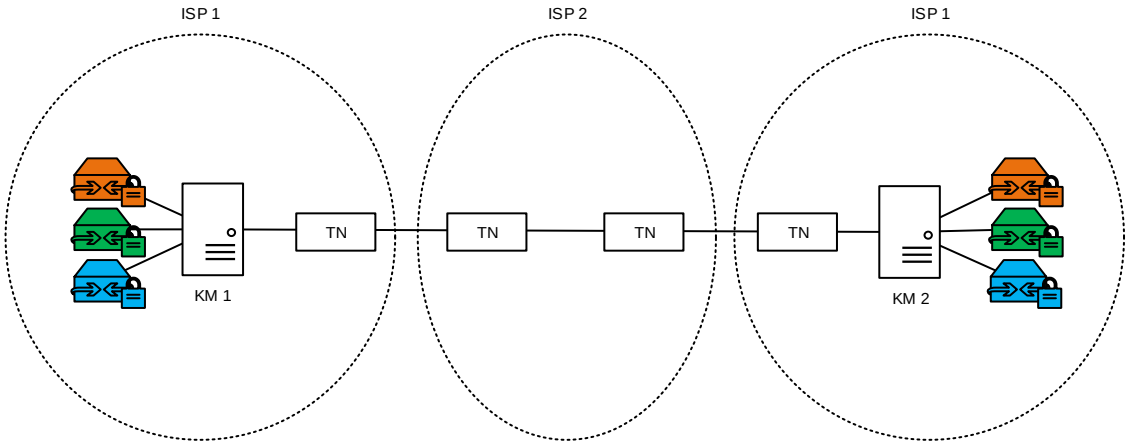


Figure 7-14 – QKD network with multiple operators

Figure 7-14 shows an example of connecting just two distinct QKD service providers. However, in reality, there may be N communication operators, and all of them pass a large number of keys between them in transit. In each case, this requires building a routing table with a division of responsibility (an analogy can be an Autonomous system in the BGP Protocol).

7.2.2.1 Protocol operations

It is necessary to develop a protocol for dynamic key relay in multi-user QKD networks in order to automatically create services over secure (encrypted) channels (QKD on demand). An example of logical separation of key relay routes is shown in Figure 7-15.

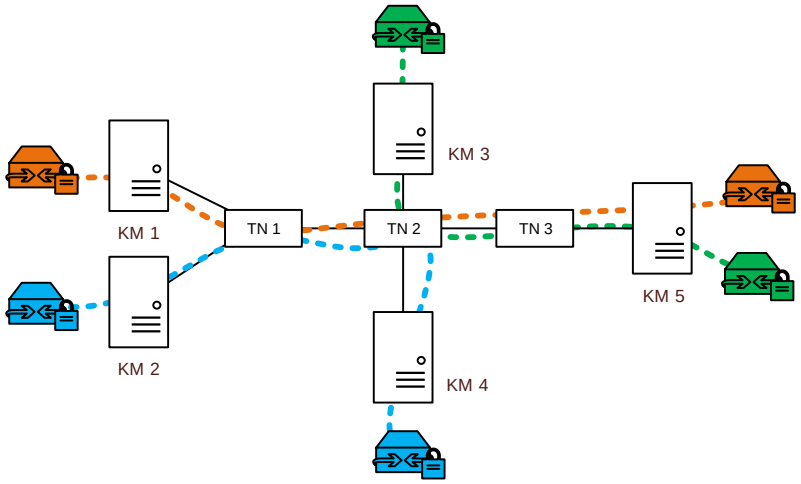


Figure 7-15 – Example of logical separation of key relay routes

Considering the example of Figure 7-16 which shows the physical diagram of a QKD network:

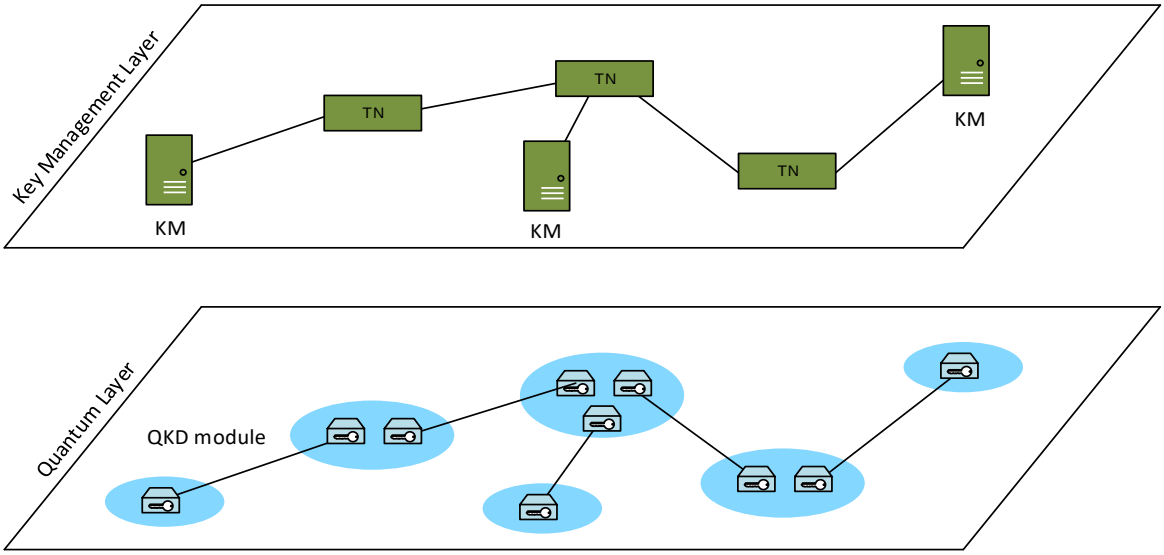


Figure 7-16 – The physical layout of a QKD network

In this case the algorithm of actions can be as follows:

1. KMs can use for example protocol like BGP to exchange information about the user networks they serve (Figure 7-17). Thus, when a cryptographic application from the 192.168.10.0/24 network receives a request to generate and exchange a key with the cryptographic application, for example, from the 10.0.0.0/8 network, KM 1 knows where the corresponding subnet is located, and it is necessary to contact KM 3;
2. KMs create ID for keys (KMA-key ID). KMA-key ID is created for a pair of cryptographic applications;
3. To transfer keys across the QKD network, KMs and Trusted Node Equipment (TNE) are required to know the network topology. Here as an option, one can consider the OSPF, IS-IS etc. This protocol must have methods of authentication and authorization of neighbor nodes;
4. In a case where the ID needs to go through another system (the network of another operator), an additional label is assigned to the ID, for example, an MPLS service label or Q-in-Q.

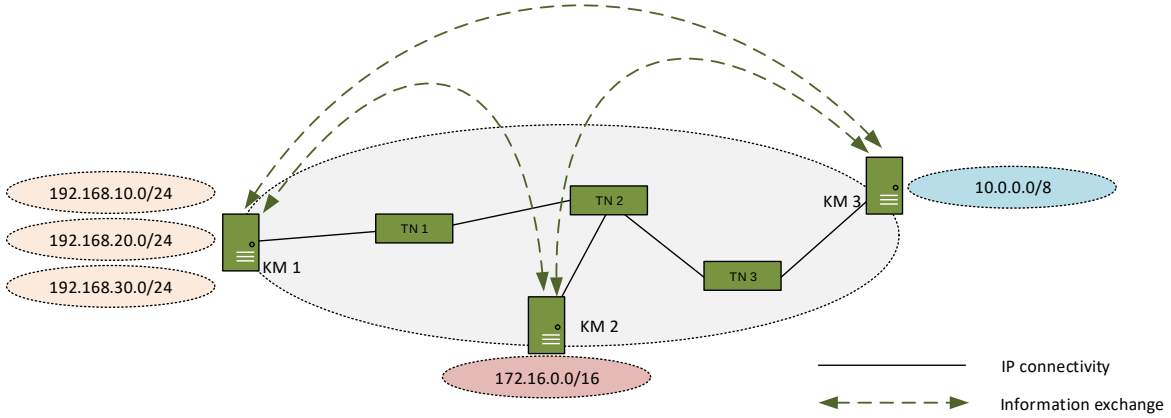
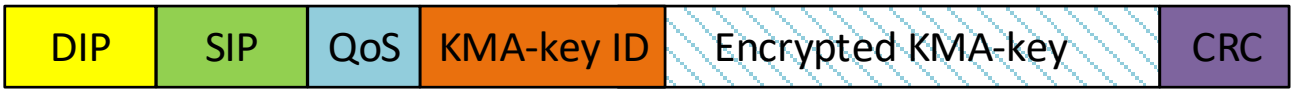


Figure 7-17 – The logical scheme of a QKD network

The theoretical representation of a frame/packet in a QKD network could appear as in Figure 7-18, without being limited to the fields represented.



NOTE – Abbreviations and terms used in the figure:

DIP Destination IP: IP address of destination KM

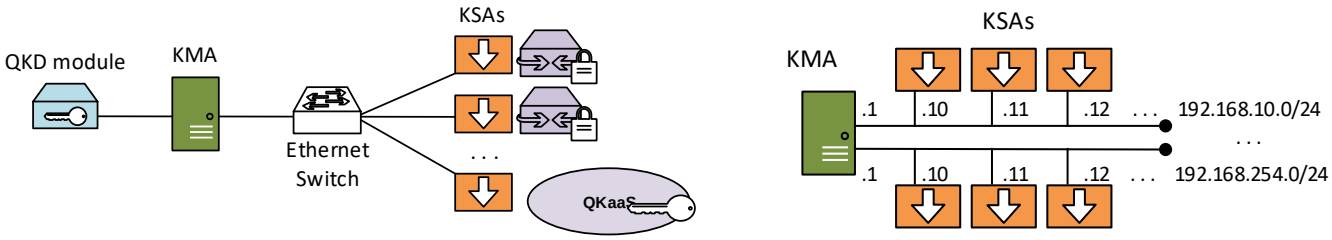
SIP Source IP: IP address of source KM

KMA-key ID ID of KMA-key

Figure 7-18 – Representation of the frame/packet structure

At the same time, special modules – KSA – should appear in the physical and logical circuits (this can be built into the encryptors or presented as a separate device).

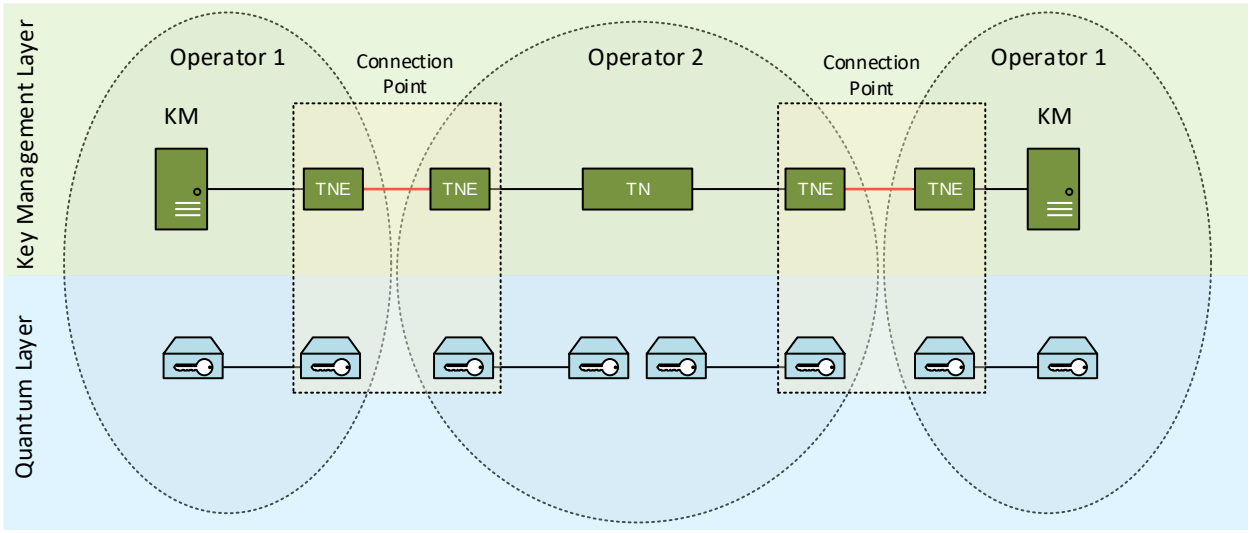
Figure 7-19 illustrates the physical and logical schemes that display KSA. These components are interfaces between the QKD network and cryptographic applications and are administered by KMAs in this case, and their IP addresses can be assigned either statically or dynamically.



a) Physical connection diagram for KSA **b) Logical connection diagram for KSA**

Figure 7-19 – Physical and logical connection diagram for KSA

The approach to packet generation described above is also intended to be used for passing through transit networks. Figure 7-20 shows a refined physical connection diagram for the transit QKD network.



NOTE – Abbreviations used in the figure:

TNE Trusted Node Equipment TN Trusted Node

Figure 7-20 – Physical connection diagram of the transit QKD network

Additional key identifiers must be included to separate the control plane and isolate the transmitted information between the communication operators. When passing through transit QKDNs, this frame/packet may appear as in Figure 7-21, although, it is not limited to the fields represented therein.

NOTE – Abbreviations and terms used in the figure:

DIP Destination IP: IP address of destination border trusted node

SIP Source IP: IP address of source border trusted node

Route tag Label of the route (DIP-SIP)

KMA-key ID ID of transit KMA-key

Figure 7-21 – Representation of the frame/packet structure for service providers interconnected

It is necessary to consider approaches to the formation and distribution of KMA-key ID in QKD networks of ISPs (TIER I, TIER II). The approaches developed later will allow formation of common standards for the construction of multi-user QKD networks and inter-operator interaction. This approach will allow large-scale implementation projects in the near future, including international connections of QKD service operators. The proposed approach will allow to create distributed extended networks using trusted nodes, but in the future, the same approach can also be used with the advent of quantum memory, repeaters, etc., which will allow for the creation of a truly new Quantum Internet.

7.2.2.2 Message parameters

Table 7-8 – HELLO messages related to neighbourhood setting between KMs for authentication

Item	Description	Remark
Sender IP	IP address of sender KM	
Sender ID	Hostname of sender KM	
Community ID	Community name	
Authentication info	Authentication string	
……		

Table 7-9 – SESSION messages aimed at creating and negotiating IDs

Item	Description	Remark
Session ID	Session ID of KMs	
Timer	Lifetime of the KMA-key ID	
Hash value	Hash value of the KMA-key ID	
……		

Table 7-10 – UPDATE messages related to route selection and ID distribution across the network, such as by a trusted node

Item	Description	Remark
Sender IP	IP address of sender KM	
Receiver IP	IP address of receiver KM	
KMA-key ID	Identifier of KMA-key	
QoS	Quality of Service	
……		

8 Protocols with respect to the QKDN control layer

The QKDN control layer provides control functions, which include key generation control, routing control for key relay, session control for QKD services, access control, and QoS control. All these control functions relate to communication protocols that need to be studied.

The protocols with respect to the QKDN control layer may include but are not limited to:

- Key generation control protocol;
- Routing control protocol;
- Session control protocol;
- Access control protocol;
- QoS control protocol.

In the following sub-clauses, representative operational procedures and corresponding message parameters are given for the routing control protocol, which serves as an example to describe a protocol.

8.1 Routing control protocol

As identified in [ITU-T Y.3802], both the key relay and key relay rerouting control procedures involve the process of supplying route information from the QKDN controller to the KM, which can be referred to two typical routing control scenarios i.e., control upon request and proactive control. To accommodate both routing control scenarios, the following routing control protocol is addressed in terms of protocol operations and message parameters.

To guarantee effective security, authentication operations between the QKDN controller and the KM, between the QKDN controller and the QKD module, are needed, similar to those of the key supply protocol. Besides authentication, two representative operational procedures and corresponding message parameters of status information supply and route information supply are given. Since the format of a message can be different, this report only focuses on basic parameters of a message and to initially accommodate both control upon request and proactive control scenarios in the messages.

8.1.1 Protocol operations

a) Authentication

For security concerns, authentications between a QKDN controller and a KM, between a QKDN controller and a QKD module need to be done before subsequent operations.

Figure 8-1 illustrates operational procedures for authentication. The operational procedures for authentication have been addressed for the key supply protocol in this report, which are also applicable here, see Clause 7.2.1 for more details. After the authentication operations, the KM (or the QKD module) and the QKDN controller can proceed to operations for status information supply or route information supply.

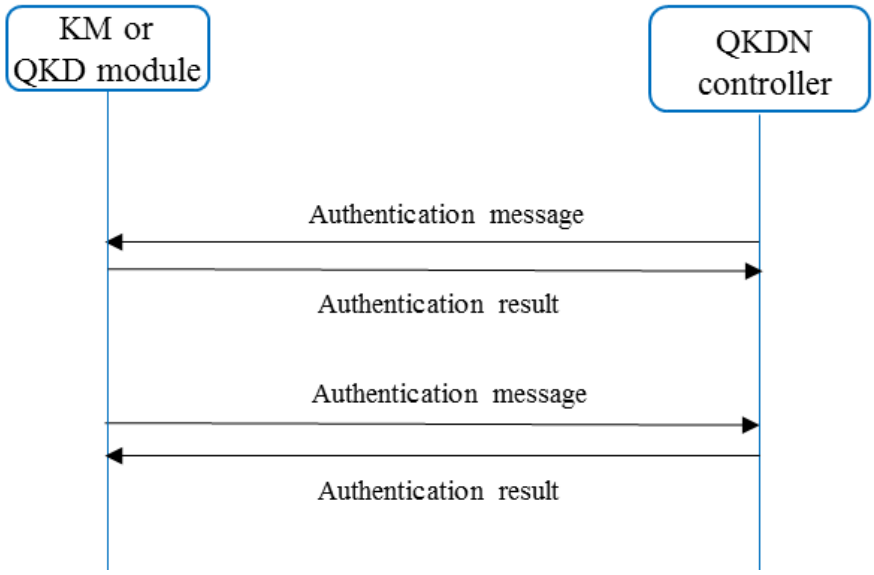


Figure 8-1 – Operational procedures for authentication

b) Status information supply

The KM or the QKD module sends status information to the QKDN controller through status information supply operations.

Figure 8-2 illustrates operational procedures for status information supply. The QKDN controller sends a status request message to the KM or the QKD module. Based on the received message, the KM or the QKD module sends a response message to the QKDN controller. In some cases, the KM or the QKD module may proactively supply status information for the QKDN controller.

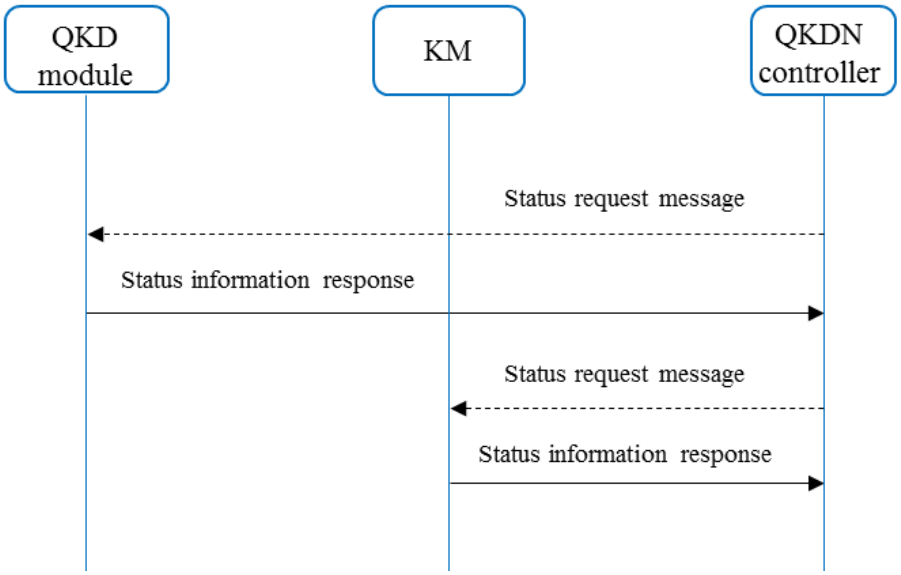


Figure 8-2 – Operational procedures for status information supply

c) Route information supply

The QKDN controller supplies route information for the KM through route information supply operations.

Figure 8-3 illustrates operational procedures for route information supply. The KM sends a route request message to the QKDN controller. Based on the received message, the QKDN controller decides the routing path and sends a response message to the KM. When there is a rerouting control needed, the QKDN controller may proactively supply route information for the KM.

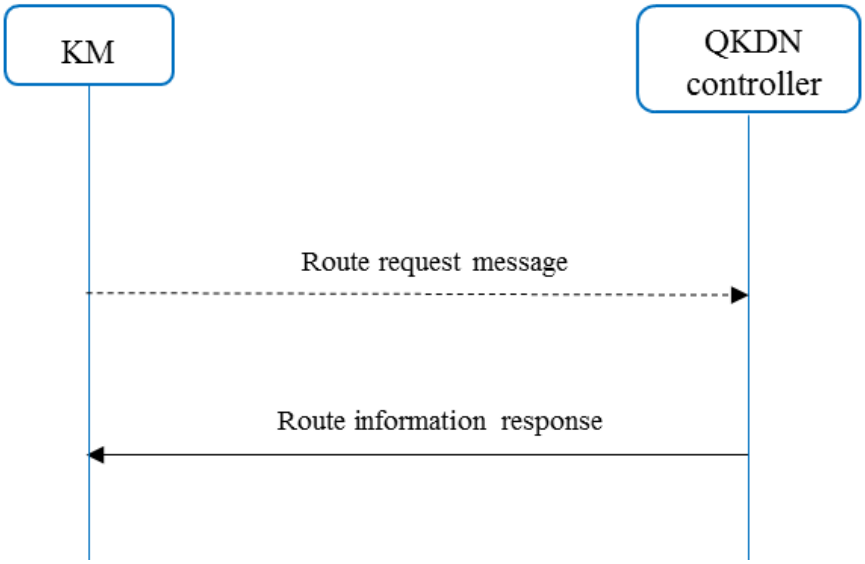


Figure 8-3 – Operational procedures for route information supply

8.1.2 Message parameters

The above operational procedures [Figure 8-1, 8-2 and 8-3] are realized through specific messages transmitted between the KM (or the QKD module) and the QKDN controller. In the following, basic parameters of a message are listed with corresponding descriptions. Additional information or examples of some parameters are added in the "remark" column for better understanding. However, the format of a message can differ, thus formats are not discussed in this report.

a) Authentication

The messages for authentication transmitted between the KM (or the QKD module) and the QKDN controller are the same as those in Table 7-2 and 7-3, see Clause 7.2.1 for details.

b) Status information supply

The messages for status information supply transmitted between the KM (or the QKD module) and the QKDN controller are as follows, where parameters in command and response messages are listed respectively in Table 8-1 and 8-2. Some parameters may be set by configuration or negotiated by a pair of KMs (or QKD modules) in advance.

Table 8-1 – Command message for status information supply

Item	Description	Remark
Message ID	A message identifier generated by a QKDN controller for the command message for status information supply.	
Command code	A code that indicates the command message is used for status information supply.	
Sender ID	Unique ID of the QKDN controller sending the command message for status information supply.	
Period	A time interval of status information supply requested by the QKDN controller.	
Status table	A table of parameters requested by the QKDN controller.	The parameters may include key amount of a KM link, key consumption rate of a KM link, status of a KM link, key generation rate of a QKD link, status of a QKD link, etc.
……		

Table 8-2 – Response message for status information supply

Item	Description	Remark
Message ID	A message identifier generated by a KM or a QKD module for the response message for status information supply.	
OrigMessage ID	A message identifier received from a command message for status information supply.	Not applicable for a proactive status information supply.
Command code	A code that indicates the response message is used for status information supply.	
Sender ID	Unique ID of the KM or the QKD module sending the response message for status information supply.	
Status table	A table of status information of parameters supplied by the KM or the QKD module sending the response message for status information supply.	The parameters may be the corresponding ones requested by the QKDN controller.
Response code	A code that indicates a result of status information supply.	0x01: success 0x02: failure
……		

c) Route information supply

The messages for route information supply transmitted between the KM and the QKDN controller are as follows, where parameters in command and response messages are listed respectively in Table 8-3 and 8-4.

Table 8-3 – Command message for route information supply

Item	Description	Remark
Message ID	A message identifier generated by a KM for the command message for route information supply.	
Command code	A code that indicates the command message is used for route information supply.	
Sender ID	Unique ID of the KM sending the command message for route information supply.	
Source ID	An identifier of a source KM of a key relay.	
Destination ID	An identifier of a destination KM of the key relay.	
Key amount	An amount of keys to be relayed.	
……		

Table 8-4 – Response message for route information supply

Item	Description	Remark
Message ID	A message identifier generated by a QKDN controller for the response message for route information supply.	
OrigMessage ID	A message identifier received from a command message for route information supply.	Not applicable for a proactive route information supply.
Command code	A code that indicates the response message is used for route information supply.	
Sender ID	Unique ID of the QKDN controller sending the response message for route information supply.	
Route table	A table of route information for a key relay from a source KM to a destination KM.	The source KM and the destination KM may be the corresponding ones indicated in the command message for route information supply.
Response code	A code that indicates a result of route information supply.	0x01: success 0x02: failure
……		

9 Protocols with respect to the QKDN management layer

The main tasks of the QKDN management layer may include fault, configuration, accounting, performance and security (FCAPS) management as well as monitoring information from the quantum layer, the key management layer and the QKDN control layer. The QKDN management layer also performs cross-layer orchestration that may support some QKDN functions such as key generation control and routing control.

The protocols with respect to the QKDN management layer may include but are not limited to the following:

- fault management protocol;
- configuration management protocol;
- accounting management protocol;
- performance management protocol;
- security management protocol.

Typical examples of operational procedures of FCAPS management and orchestration of QKDN are given in Clause 10 of [ITU-T Y.3804] and information components for reference points in Appendix I of [ITU-T Y.3804] can be taken as examples of message parameters for further information.

10 Security considerations

The QKDN is intrinsically related to security issues. Protocols are always addressed with security concerns in the QKDN community. The key data, metadata, control and management information are conveyed via protocols at each reference point in the QKDN. The security concerns on confidentiality, integrity, authentication, authorization, availability and accountability of conveyed data or information should be involved when designing QKDN protocols. Good references are [ITU-T X.1710] and [ITU-T X.1712], wherein, security requirements and measures are addressed for overall QKDN and key management. The details of security analysis on the QKDN protocols are out of the scope of this report.

11 Conclusions and standardization suggestions

As discussed in the previous clauses and following the functional architecture model of QKDN specified in [ITU-T Y.3802], protocols with respect to the key management layer, QKDN control layer and QKDN management layer have been identified in this report. To focus the discussion to a protocol level, some protocols have been addressed as examples with their representative operational procedures and corresponding message parameters.

Though basic operational procedures have been discussed in some ITU-T Recommendations, a next level specification on protocol issues is still needed by the community. This will help implementation of QKDN by addressing compatibility and interoperability concerns.

For the way forward with respect to standardization issues, some suggestions are as follows:

a) Authentication and authorization are common issues for many protocols which are correlated in terms of access control. Thus, a specific concern on authentication and authorization should be addressed for each applicable protocol in QKDN when being developed. The work item "Authentication and authorization in QKDN using quantum safe cryptography" [b-X.sec_QKDN_AA] under development in ITU-T SG17 is a good initiator at this point.

b) The reference points can be categorized into three following groups:

- Key management layer: Ak, Kx-1, Kx-2, Kq-1, Kq-2;
- QKDN control layer: Ck, Cq, Cqrp, Cops, Cx;
- QKDN management layer: Mu, Mc, Mk, Mq, Mqrp, Mops.

Each reference point represents an interface with underlying protocol to be developed.

c) For the protocols with respect to key management layer, each reference point (Ak, Kq-1, Kq-2, Kx-1 and Kx-2) corresponds to a specific protocol that differs from each other. The key supply protocol at Ak is quite demanded by different cryptographic application providers and is applicable to different QKDN implementations even if a QKDN is not designed for key relay. For compatibility and interoperability concerns, the key acquisition protocol at Kq-1 is expected to accommodate QKD-keys from QKD modules implementing different QKD protocols. When the key relay is applicable in a QKDN, the key relay protocol at Kx-1 needs to be carefully designed, wherein the key synchronization mechanism can also be applied to the reference point Kx-2. Hence, the following work items on key management related protocols are recommended to be initiated respectively in SG11:

- Protocols at the Ak interface between a KM and a cryptographic application;
- Protocols at the Kq-1 interface between a KM and a QKD module;
- Protocols at the Kx-1 and Kx-2 interfaces between KMs;
- Protocols at the Kq-2 interface between a KM and a QKD module.

d) For the protocols with respect to QKDN control layer, the key generation control protocol mainly relates to reference points Cq, Cqrp and Cops. However, a QKD link does not necessarily include a quantum relay point or an optical switching/splitting entity. So, it is better to develop protocols according to different functional entities in the quantum layer. The routing control protocol relates to the reference point Ck, wherein the session control issue is involved. Hence, the following work items on QKDN control related protocols are recommended to be initiated respectively in SG11:

- Protocols at the Ck interface between a QKDN controller and a KM;
- Protocols at the Cq interface between a QKDN controller and a QKD module;
- Protocols at the Cqrp and Cops interfaces between a QKDN controller and a QKD link;
- Protocols at the Cx interface between QKDN controllers.

e) For the protocols with respect to QKDN management layer, the FCAPS management protocol mainly relates to reference points Mc, Mk, Mq, Mqrp and Mops. To avoid interference with each other, it is better to develop protocols according to different functional entities. Hence, the following work items on QKDN management related protocols are recommended to be initiated respectively in SG11:

- Protocols at the Mc interface between a QKDN manager and a QKDN controller;
- Protocols at the Mk interface between a QKDN manager and a KM;
- Protocols at the Mq interface between a QKDN manager and a QKD module;
- Protocols at the Mqrp and Mops interfaces between a QKDN manager and a QKD link;
- Protocols at the Mu interface between a QKDN manager and a user network manager.

f) The QKDN is intrinsically related to security issues. Protocols are always addressed with security concerns in the QKDN community. In this way, collaboration and coordination should be emphasized between different SGs (e.g., SG11 and SG17) when developing work items on QKDN protocols.

Bibliography

[b-ETSI GR QKD 007] Group Report ETSI GR QKD 007 (2018), *Quantum Key Distribution (QKD); Vocabulary*.

[b-ETSI GS QKD 004] Group Specification ETSI GS QKD 004 (2020), *Quantum Key Distribution (QKD); Application Interface*.

[b-ETSI GS QKD 008] Group Specification ETSI GS QKD 008 (2010), *QKD module security specification*.

[b-ETSI GS QKD 014] Group Specification ETSI GS QKD 014 (2019), *Quantum Key Distribution (QKD); Protocol and data format of REST-based key delivery API*.

[b-ITU-T Q.1743] Recommendation ITU-T Q.1743 (2016), *IMT-Advanced references to Release 11 of LTE-Advanced evolved packet core network*.

[b-ITU-T Q.QKDN_profr] Draft Recommendation ITU-T Q.QKDN_profr, *Quantum key distribution networks – Protocol framework.*

[b-ITU-T X.800] Recommendation ITU-T X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications*.

[b-Shannon] Shannon, C. (1949), *Communication Theory of Secrecy Systems*. Bell System Technical Journal, Vol. 28, pp. 666-682.

[b-X.sec_QKDN_AA] Draft Recommendation ITU-T X.sec_QKDN_AA, *Authentication and authorization in QKDN using quantum safe cryptography*.