INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 2022-2024

SG11-TD455 /GEN

STUDY GROUP 11

Original: English

Question(s): 2/11 Geneva, 10-19 May 2023

TD

Source: Editors

Title: Output – draft baseline text of draft Recommendation ITU-T Q.QKDN_Kq-1: Protocols for Kq-1 interface for QKDN (Geneva, 10-19 May 2023)

Contact: Hongyu Wu
QuantumCTek Co., Ltd.
China
E-mail: hongyu.wu@quantum-info.com

Contact: Zhangchao Ma
CAS Quantum Network Co., Ltd.
China
E-mail: mazhangchao@qtict.com

Contact: Junsen Lai
CAICT, Ministry of Industry and Information Technology (MIIT)
China
E-mail: laijunsen@caict.ac.cn

Contact: Kaoru KENYOSHI
NICT
Japan
Tel: +81 50 3566 5852
E-mail: kaoru.kenyoshi@nict.go.jp

Contact: Mariko Honda
NICT
Japan
E-mail: mariko.honda@ntt-at.co.jp

Abstract: This TD is the output of draft Recommendation Q.QKDN_Kq-1: Protocols for Kq-1 interface for QKDN (Geneva, 10-19 May 2023).

Summary

This is the output of draft Recommendation ITU-T Q.QKDN_Kq-1: Protocols for Kq-1 interface for QKDN, based on the discussion results on C128R1 and C201 with modifications at the Q2/11 meeting (Geneva, 10-19 May 2023).

Draft Recommendation ITU-T Q.QKDN_Kq-1

Protocols for Kq-1 interface for QKDN

Summary

Recommendation ITU-T Q.QKDN_Kq-1 specifies protocols for Kq-1 interface in quantum key distribution networks (QKDN).

Keywords

Protocol, QKD (quantum key distribution), QKDN (QKD network), signalling procedure, message parameters

Table of Contents

Draft new Recommendation ITU-T Q.QKDN_Kq-1

Protocols for Kq-1 interface for QKDN

1. Scope

This Recommendation specifies protocols at Kq-1 interface for quantum key distribution network (QKDN) especially the following areas.

- signalling procedures for Kq-1 interface for QKDN;
- signalling messages and parameters for Kq-1 interface for QKDN;
- security considerations.

2. References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T X.1712] Recommendation ITU-T X.1712 (2021)/Cor.1 (02/2022), *Security requirements and measures for quantum key distribution networks - key management.*

[ITU-T Y.3800] Recommendation ITU-T Y.3800 (2019)/Cor.1 (04/2020), *Overview on networks supporting quantum key distribution*.

[ITU-T Y.3802] Recommendation ITU-T Y.3802 (2020)/Cor.1 (04/2021), *Quantum key distribution networks - Functional architecture*.

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

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

3. Definitions

3.1. Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

~~3.1.1 **information theoretically secure (IT-secure)** [ITU-T Y.3800]: Secure against any deciphering attack with unbounded computational resources.~~

~~3.1.2 **key life cycle** [ITU-T Y.3800]: A sequence of steps that a key undergoes from its reception by a key manager (KM) through its use in a cryptographic application and until deletion or preservation depending on the key management policy.~~

~~3.1.3~~3.1.1 **key management** [ITU-T Y.3800]: All activities performed on keys during their life cycle starting from their reception from the quantum layer, the storage, formatting, relay, synchronization, authentication, to supply to cryptographic application and delete or preserve depending on the key management policy.

~~3.1.4 **key management agent (KMA)** [ITU-T Y.3802]**:** A functional element to manage keys generated by a quantum key distribution (QKD) module/QKD modules in a QKD node (trusted node).~~

~~NOTE – KMA acquires keys from a QKD module/QKD modules, synchronizes, resize, formats, and stores them. It also relays keys through key management agent (KMA) links.~~

~~3.1.5 **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.~~

~~3.1.6 **key management agent link (KMA link)** [ITU-T Y.3802]**:** A communication link connecting KMAs to perform key relay and communications for key management.~~

~~3.1.7~~3.1.2 **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.

~~3.1.8 **key manager link (KM link)** [ITU-T Y.3800]: A communication link connecting key managers (KMs) to perform key management.~~

~~3.1.9~~3.1.3 **key relay** [ITU-T Y.3800]: A method to share keys between arbitrary quantum key distribution (QKD) nodes via intermediate QKD node(s).

~~3.1.10 **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 client.~~

~~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 client.~~

~~3.1.11 **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.~~

~~3.1.12 key supply agent link (KSA link) [ITU-T Y.3802]: A communication link connecting KSAs to perform key synchronization and integrity verification.~~

3.1.4 **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.

~~3.1.13~~3.1.5 **quantum key distribution-key (QKD-key)** [ITU-T Y.3802]: A pair of symmetric random bit strings generated by a pair of quantum key distribution (QKD) modules, particularly referring to random bit strings before being resized and formatted in a key manager (KM).

~~3.1.14~~3.1.6 **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.

~~3.1.15~~3.1.7 **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 (QKD-Tx) and the receivers (QKD-Rx).

~~3.1.16~~3.1.8 **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.

~~3.1.17~~3.1.9 **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.

~~3.1.18 **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.~~

~~3.1.19~~3.1.10 **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).

~~This Recommendation uses the following terms defined elsewhere:~~

3.2. Terms defined in this Recommendation

None.

4. Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

HTTPS	HyperText Transfer Protocol Secure
ID	Identifier
KM	Key manager
~~KSA~~	~~Key Supply Agent~~
QKD	Quantum Key Distribution
QKDN	QKD Network
TCP	Transmission Control Protocol

5. Conventions

None.

6. Kq-1 interface

The Kq-1 interface is defined between the key manager (KM) and the QKD module. The Kq-1 interface is used for key acquisition between the key storage function in the KM and the QKD-key supply function in the QKD module.

7. Signalling procedure

The following two modes are specified for key supply at the Kq-1 interface.

1) Proactive key supply mode: QKD module initiates the supply of QKD-keys to KM.

2) Key supply upon request mode: KM initiates the procedure by requesting QKD module to supply QKD-keys and QKD module supplies QKD-keys to the KM in response to the request.

The protocol suites applied for the signalling are specified in clause 8 of [ITU-T Q.QKDN_profr].

7.1. Signalling procedure for proactive key supply mode

This procedure is initiated when the QKD-keys are generated in the QKD module. The amount of the QKD-keys supplied mainly depends on the key generation request from the QKDN controller.

Figure 1 shows signalling procedures for proactive key supply mode at the Kq-1 interface.

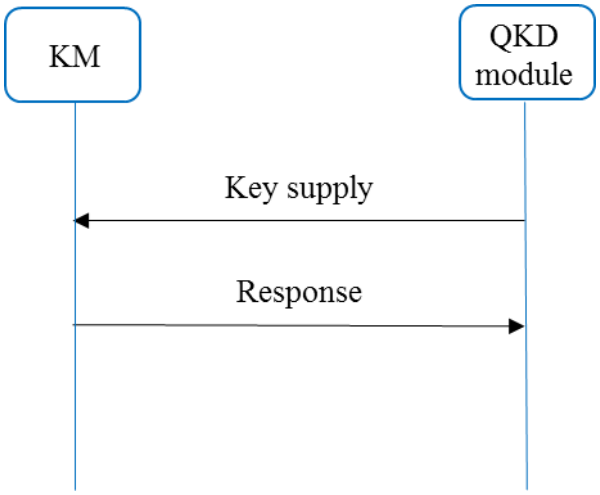

Figure 1 – Signalling procedures for proactive key supply mode at the Kq-1 interface

7.2. Signalling procedure for key supply upon request mode

In this procedure, the KM sends a key request to the QKD module when the KM needs QKD-keys. The QKD module supplies QKD-keys to the KM in response to the request.

Figure 2 shows signalling procedures for key supply upon request mode at the Kq-1 interface.

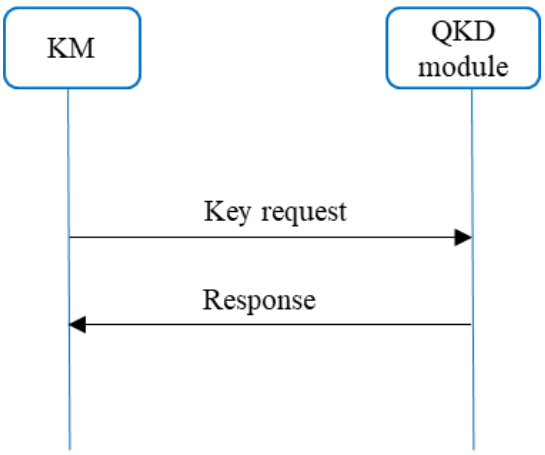

Figure 2 – Signalling procedures for key supply upon request mode at the Kq-1 interface

8. Signalling messages and parameters

This clause specifies messages and their parameters for the Kq-1 interface.

In the M/O column of the tables in this clause, M indicates that the parameter is mandatory for signalling, and O indicates that the parameter is optional for signalling.

NOTE – The messages and parameters defined in this clause are independent of a specific protocol. Different protocols can have different implementations of these messages and parameters. The recommended protocol implementations are described in Annex A and B. A message parameter described in the following tables is not necessarily mapped to a field in the message payload and might be a part of control parameters of one specific protocol. The Data type column of the tables may vary with specific protocols.

8.1. Messages and parameters for proactive key supply mode

8.1.1. Key supply message

Key supply message is sent from the QKD module to the KM at the same QKD node. The QKD module supplies the QKD-key with a unique QKD-key ID to the KM.

Table 1 shows parameters of Key supply message.

Table 1 – Parameters of Key supply message

Parameter	Description	Data type	M/O	Remarks
QKD-key	QKD-key data supplied	string	M	
QKD-key ID	ID of the QKD-key supplied	string	M	
QKD module ID	ID of the QKD module (Alice or Bob) that supplies the QKD-key	string	M	
Matching QKD module ID	ID to identify the matching QKD module that constitutes the pair of Alice and Bob	string	O	
Key length	Length of each QKD-key supplied	integer	O	A default value is applied if omitted.
Generation time stamp	Time stamp of QKD-key generation at the pair of QKD modules	string	O	
Hash value	Hash value of the QKD-key data.	string	O	
Extension	Array of extension parameters	Array of objects	O	For future use

8.1.2. Response to key supply message

Response to key supply message is sent from the KM to the QKD module in response to the key supply. The KM notifies the results of the receipt of the QKD-keys to the QKD module.

Table 2 shows parameters of Response to key supply message.

Table 2 – Parameters of Response to key supply message

Parameter	Description	Data type	M/O	Remarks
QKD-key ID	ID of the QKD-key received.	string	M	
QKD module ID	ID of the QKD module (Alice or Bob)	string	O	

	that supplies the QKD-key			
Matching QKD module ID	ID to identify the matching QKD module that constitutes the pair of Alice and Bob	string	O	
Response	Result of the receipt of the QKD-key	string	M	Success or failure reason
Extension	Array of extension parameters	Array of objects	O	For future use

8.2. Messages and parameters for key supply upon request mode

8.2.1. Key request message

Key request message is sent from the KM to the QKD module to request QKD-keys.

Table 3 shows parameters of Key request message.

Table 3 – Parameters of Key request message

Parameter	Description	Data type	M/O	Remarks
Number of ~~KSA~~QKD-keys	Number of ~~KSA~~QKD-keys requested	integer	O	A default value is applied if omitted.
Size of ~~KSA~~QKD-key	Length of each ~~KSA~~QKD-key requested	integer	O	A default value is applied if omitted.
Extension	Array of extension parameters	Array of objects	O	

8.2.2. Response to key request message

Response to key request message is sent from the QKD module to the KM in response to the key request from the cryptographic application. The QKD module supplies the requested QKD-keys to the cryptographic application.

Table 4 shows parameters of Response to key request message.

Table 4 – Parameters of Response to key request message

Parameter		Description	Data type	M/O	Remarks
Keys		Key file consists of key data and metadata.	Array of objects	M	
	~~QKD k~~Key	~~KSA~~QKD-key data provided for the request	string	M	
	~~QKD k~~Key ID	ID of the ~~KSA~~QKD-key provided	string	M	
	Key extension	Extensions to key file	object	O	
Response		Result of key supply	string	M	
Extension		Array of extension parameters	Array of objects	O	

Editor's note: the name of parameters of Keys and Key should be considered in future meeting.

9. Security considerations

Key data and associated metadata are transferred through Kq-1 reference point. Security requirements and measures to protect them are specified in [ITU-T X.1712].

Annex A

Protocol implementation ~~for~~using TCP

(This annex forms an integral part of this Recommendation.)

This annex describes a protocol implementation for messages and parameters ~~for~~using TCP which are described in clause 8.

NOTE 1 - Some of the parameters are mapped to a part of control information of the protocol instead of being mapped to a field in the data payload.

The QKD module can connect to the KM using TCP protocol [b-IETF RFC 9293~~793~~]. The corresponding message format over TCP is as follows.

Version	MessageID	CommandCode	Length	Payload

Figure A.1 – Message format over TCP

Version: the current version of the message format adopted, 2 bytes;

MessageID: the unique identifier of each message, 4 bytes;

CommandCode: a unique code that denotes different Command/Response messages transferred at the Kq-1 interface, 2 bytes;

Length: the length of the message payload, 2 bytes;

Payload: the message parameters of a specific Command/Response message, JSON data format [b-IETF RFC 8259].

NOTE 2 – TLS protocol [b-IETF RFC 5246] can be implemented with TCP protocol for enhanced security.

~~*Editor's note - Detail implementation of TLS protocol needs to be addressed.*~~

At the connection establishment, mutual authentication between the QKD module and the KM shall be performed. After the mutual authentication, a Command/Response message can be transferred at the Kq-1 interface for key supply from the QKD module to the KM.

NOTE 3 – When applying TLS protocol, the QKD module can verify the validity of a certificate the KM possesses and confirm the ID of the KM it is connecting to, based on the certificate. Similarly, the KM can verify the validity of a certificate the QKD module possesses and confirm the ID of the connecting QKD module based on the certificate.

Table A.1 shows a list of CommandCode vs. Command/Response message name.

Table A.1 – CommandCode vs. Command/Response message name

CommandCode	Command/Response message name
0x01	Key supply
0x02	Response to key supply
0x03	Key request
0x04	Response to key request

Annex B

Protocol implementation for key supply upon request mode using HTTPS

(This annex forms an integral part of this Recommendation.)

The signalling ~~procedure~~messages and parameters for key supply upon request mode specified in clause ~~7.2~~8.2 can be implemented ~~as the REST-based key delivery procedure~~using HTTPS according to the protocol and data format of REST-based key delivery API specified in [b-ETSI GS QKD 014]. ~~To illustrate how the signalling procedure is implemented as the REST-based key delivery procedure, t~~This annex describes the mapping of the messages and parameters ~~for that mode~~ specified in clause 8.2 to the corresponding data format specified in [b-ETSI GS QKD 014].

NOTE – In this implementation, the KM and the QKD module play the roles of SAE (Secure Application Entity) and the KME (Key Management Entity) defined in [b-ETSI GS QKD 014] respectively.

B.1 Key request message

In this implementation ~~T~~the Key request message specified in clause 8.2.1 ~~is implemented as~~corresponds to the HTTPS request ~~phase~~ of the HTTPS transaction performed as the 'Get Key' method specified in [b-ETSI GS QKD 014]. Table B.1 shows the mapping of the Key request message to the 'Get Key' method.

Table B.1 – Mapping of Key request message to Get Key method

Parameter	M/O	Data type	Implementation in 'Get Key' method
Number of QKD-keys	O	integer	The 'number' item in the Key request data format
Size of QKD-key	O	integer	The 'size' item in the Key request data format
Extension	O	array of objects	The 'extension_mandatory' or 'extension_optional' item in the Key request data format

B.2 Response to key request message

In this implementation ~~T~~the Response to key request message specified in clause 8.2.2 ~~is implemented as~~corresponds to the HTTPS response ~~phase~~ of the HTTPS transaction performed as the 'Get Key' method specified in [b-ETSI GS QKD 014]. Table ~~C~~B.2 shows the mapping of the Response to key request message to the 'Get Key' method.

Table B.2 – Mapping of Response to key request message to Get Key method

Parameter	M/O	Data type	Implementation in 'Get Key' method
~~Key~~Keys	M	array of objects	The 'Keys' item in the Key container data format
~~QKD k~~Key	M	string	The 'key' item in the Key container data format
~~QKD k~~Key ID	M	string	The 'key_ID' item in the Key container data format
Key extension	O	object	The 'key_ID_extension' item in the Key container data format
Response	M	string	The status code of HTTPS transaction performed as 'Get Key' method

Extension	O	array of objects	The ‘key_container_extension’ item in the Key container data format

This annex describes a protocol implementation for messages and parameters for HTTPS which are described in clause 8.

B.1 Key supply message

Table B.1 shows HTTPS profiles for Key supply message.

Table B.1 – HTTPS profiles for Key supply message

Parameter	Mapped to	Data type
QKD key	HTTPS request body	JSON key:value pair
QKD key ID	HTTPS request body	JSON key:value pair
QKD module ID	Implicit (source address of HTTPS connection)	
Matching QKD module ID	HTTPS request body	JSON key:value pair
Key length	HTTPS request body	JSON key:value pair
Generation time stamp	HTTPS request body	JSON key:value pair
Hash value	HTTPS request body	JSON key:value pair
Extension	HTTPS request body	JSON Array

B.2 Response to key supply message

Table B.2 shows HTTPS profiles for Response to key supply message.

Table B.2 – HTTPS profiles for Response to key supply message

Parameter	Mapped to	Data type
QKD key ID	HTTPS response body	JSON key:value pair
QKD module ID	Implicit (source address of HTTPS connection)	
Matching QKD module ID	HTTPS response body	JSON key:value pair
Response	Status code in HTTP response header	String(integer)
Extension	HTTP response body	JSON object

B.3 Key request message

Table B.3 shows HTTPS profiles for Key request message.

Table B.3 – HTTPS profiles for Key request message

Parameter	Mapped to	Data type
Number of QKD keys	HTTPS request body (‘number’ in key request container)	JSON (‘number’: value)
Size of QKD key	HTTPS request body (‘size’ in key request container)	JSON (‘size’: size)
Extension	HTTPS request body	JSON array

B.4 Response to key request message

Table B.4 shows HTTPS profiles for Response to key request message.

Table B.4 – HTTPS profiles for Response to key request message

Parameter		Mapped to	Data type
Key		HTTPS response body	Array of objects
	QKD key	HTTPS response body	JSON (‘key’: base64string)
	QKD key ID	HTTPS response body	JSON (‘key_ID’: uuid)
	Key extension	HTTPS response body	object
Response		HTTPS response header	String(integer)
Extension		HTTPS response body	object

Bibliography

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

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

[b-IETF RFC 9293~~793~~] IETF RFC 9293~~793~~, ~~*TRANSMISSION CONTROL PROTOCOL*~~ *Transmission Control Protocol (TCP).*

[b-IETF RFC 5246] IETF RFC 5246, *The Transport Layer Security (TLS) Protocol Version 1.2.*

[b-IETF RFC 8259] IETF RFC 8259, *The JavaScript Object Notation (JSON) Data Interchange Format.*