ITUPublications
Recommendations

International Telecommunication Union
Standardization Sector

Recommendation

ITU-T Y.3803 (2020) Amd. 1 (11/2023)

SERIES Y: Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities

Quantum key distribution networks

Quantum key distribution networks – Key management
Amendment 1

ITU-T Y-SERIES RECOMMENDATIONS

Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities

GLOBAL INFORMATION INFRASTRUCTURE	Y.100-Y.999
General	Y.100-Y.199
Services, applications and middleware	Y.200-Y.299
Network aspects	Y.300-Y.399
Interfaces and protocols	Y.400-Y.499
Numbering, addressing and naming	Y.500-Y.599
Operation, administration and maintenance	Y.600-Y.699
Security	Y.700-Y.799
Performances	Y.800-Y.899
INTERNET PROTOCOL ASPECTS	Y.1000-Y.1999
General	Y.1000-Y.1099
Services and applications	Y.1100-Y.1199
Architecture, access, network capabilities and resource management	Y.1200-Y.1299
Transport	Y.1300-Y.1399
Interworking	Y.1400-Y.1499
Quality of service and network performance	Y.1500-Y.1599
Signalling	Y.1600-Y.1699
Operation, administration and maintenance	Y.1700-Y.1799
Charging	Y.1800-Y.1899
IPTV over NGN	Y.1900-Y.1999
NEXT GENERATION NETWORKS	Y.2000-Y.2999
Frameworks and functional architecture models	Y.2000-Y.2099
Quality of Service and performance	Y.2100-Y.2199
Service aspects: Service capabilities and service architecture	Y.2200-Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250-Y.2299
Enhancements to NGN	Y.2300-Y.2399
Network management	Y.2400-Y.2499
Computing power networks	Y.2500-Y.2599
Packet-based Networks	Y.2600-Y.2699
Security	Y.2700-Y.2799
Generalized mobility	Y.2800-Y.2899
Carrier grade open environment	Y.2900-Y.2999
FUTURE NETWORKS	Y.3000-Y.3499
CLOUD COMPUTING	Y.3500-Y.3599
BIG DATA	Y.3600-Y.3799
QUANTUM KEY DISTRIBUTION NETWORKS	**Y.3800-Y.3999**
INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES	Y.4000-Y.4999
General	Y.4000-Y.4049
Definitions and terminologies	Y.4050-Y.4099
Requirements and use cases	Y.4100-Y.4249
Infrastructure, connectivity and networks	Y.4250-Y.4399
Frameworks, architectures and protocols	Y.4400-Y.4549
Services, applications, computation and data processing	Y.4550-Y.4699
Management, control and performance	Y.4700-Y.4799
Identification and security	Y.4800-Y.4899
Evaluation and assessment	Y.4900-Y.4999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.3803

Quantum key distribution networks – Key management

Amendment 1

Summary

Recommendation ITU-T Y.3803 provides help for the design, deployment, and operation of key management of a quantum key distribution network (QKDN). The overall structure and basic functions of a QKDN are first reviewed along with Recommendation ITU-T Y.3800, then the requirements of a QKDN are reviewed along with Recommendation ITU-T Y.3801. Functional elements and procedures of key management are then described.

Quantum key distribution (QKD) protocols provide the means to distribute symmetric random bit strings as a secure key that can be proven to be secure even against an eavesdropper who has unbounded computational ability. A basic element of QKD is a pair of QKD modules linked by a QKD link that allows two remote parties to share secure keys. A QKDN consists of two or more QKD links and trusted nodes (QKD nodes), where any pair of two QKD nodes can share secure keys via QKD links and key relay. In the end, these keys are supplied to cryptographic applications in user networks. To implement a QKDN and appropriately integrate with the user network, an overview of QKD technologies, including network capabilities, conceptual structure, layered model, basic functions and components, and its relation to the user network, is given in Recommendation ITU-T Y.3800.

To operate a QKDN efficiently and securely, key management is the highest priority issue because without this, most meaningful QKD operations and services cannot be realized. Key management includes, at least, storing keys generated by QKD modules, relaying keys between the nodes of the QKDN, and supplying keys to cryptographic applications upon requests from users, all in a secure manner. The standardization of these issues is essential to realize the interoperability for QKDN, ensuring security and widening applications of QKD.

Amendment 1 revises Figure 2 to include the Mx reference point and an editorial correction on key relay.

History [*]

Edition	Recommendation	Approval	Study Group	Unique ID
1.0	ITU-T Y.3803	2020-12-07	13	11.1002/1000/14408
1.1	ITU-T Y.3803 (2020) Amd. 1	2023-11-29	13	11.1002/1000/15715

Keywords

Key management, key relay, key supply, quantum key distribution, QKD, QKD network

[*] To access the Recommendation, type the URL https://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID.

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.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T Y.3803

Quantum key distribution networks – Key management

Amendment 1

Editorial note: This is a complete-text publication. Modifications introduced by this amendment are shown in revision marks relative to Recommendation ITU-T Y.3803 (2020).

1 Scope

This Recommendation describes key management for quantum key distribution (QKD) networks that address technical specifications to help the implementation and operation.

In particular, the scope of this Recommendation includes:

– an overview of key management in a quantum key distribution network (QKDN);

– functional elements of key management;

– operations of key management; and

– key formats (key data and metadata).

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.1714] Recommendation ITU-T X.1714 (2020), *Key combination and confidential key supply for quantum key distribution networks*.

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

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 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.

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

3.1.3 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.

3.1.4 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.

3.1.5 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.

3.1.6 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.

3.1.7 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 [ITU-T Y.3800]: A communication link connecting key managers (KMs) to perform key management.

3.1.9 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.

3.1.10 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.

3.1.11 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.

3.1.12 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).

3.1.13 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.

3.1.14 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).

3.1.15 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.

3.1.16 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.

3.1.17 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.

3.1.18 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).

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

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 key data: Random bit strings that are used as a cryptographic key.

3.2.2 key management agent-key (KMA-key): Key data stored and processed in a key management agent (KMA), and securely shared between a KMA and a matching KMA.

3.2.3 key supply agent-key (KSA-key): Key data stored and processed in a key supply agent (KSA), and securely shared between a KSA and a matching KSA.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ID	Identifier
IT-secure	Information Theoretically secure
KM	Key Manager
KMA	Key Management Agent
KSA	Key Supply Agent
OTP	One-Time Pad
QBER	Quantum Bit Error Rate
QKD	Quantum Key Distribution
QKDN	Quantum Key Distribution Network
RNG	Random Number Generator
XOR	exclusive OR

5 Conventions

In this Recommendation:

The phrase "is required to" indicates a requirement that must be strictly followed and from which no deviation is permitted if conformance to this Recommendation is to be claimed.

The phrase "is recommended" indicates a requirement that is recommended but not absolutely required. Thus, this requirement need not be present to claim conformance.

6 Overview of key management

This Recommendation provides a description of key management for a QKDN, which is essential to design and realize networks supporting QKD technologies.

Once keys are generated by QKD modules, they are stored securely as classical bit strings in a server called a KM, until they are supplied to a cryptographic application. The KM is located in the QKD node, which is trusted, and protected against intrusion and attacks by unauthorized parties. Keys are often resized and formatted into a key of an appropriate length for the cryptographic application.

Point-to-point QKD links can be concatenated into a QKDN via the trusted QKD nodes to extend the reachability and availability of key supply. In the QKDN, keys can be relayed via the QKD nodes and shared between two parties, even if they are not directly connected by a QKD link. Keys are synchronized and authenticated in an appropriate manner, when they are transferred from a functional to another entity. These keys can be supplied to various cryptographic applications in the user network. All activities performed on keys during their life cycle, including their reception from the quantum layer, storage, formatting, relaying, synchronization, authentication, supply and deletion or preservation, are referred to as key management, which lies at the heart of QKD applications and services.

Basic functions and layered structures of a QKDN are specified in [ITU-T Y.3800]. Figure 1 depicts basic key management operations in a QKDN. Each pair of QKD modules connected by a QKD link generates keys in its own way. Generated keys are transferred to KMs. The KMs manage the keys and supply them to cryptographic applications in the service layer of the user network. The keys can be relayed via KMs and shared between any designated QKD nodes. The QKDN controller performs routing control of key relay. The QKDN manager monitors the status of the whole of the QKDN and supports key life cycle management for the KMs, as well as routing and rerouting control of key relay for the QKDN controller.

When QKD modules detect an alarm, such as an increase of their quantum bit error rates (QBERs), they alert the QKDN controller directly or indirectly via the KMs. The QKDN controller then reroutes the path for key relay to circumvent the faulty QKD link for successful key supply.

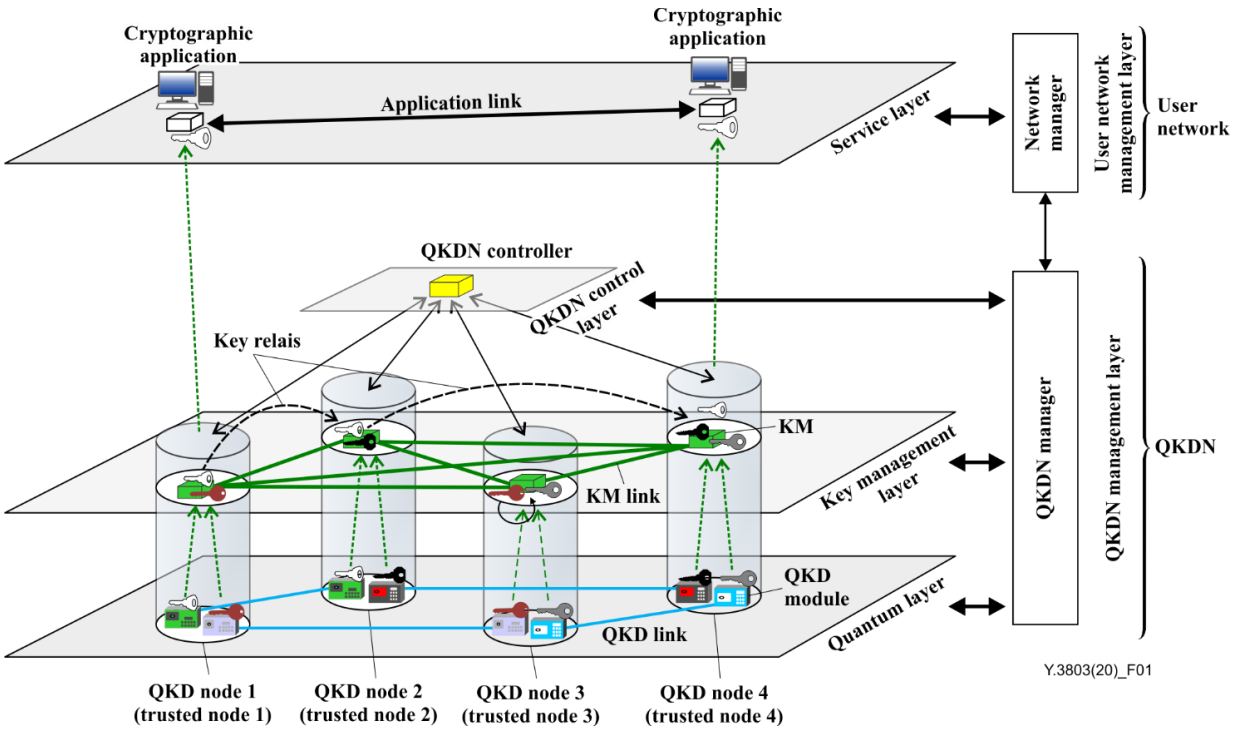
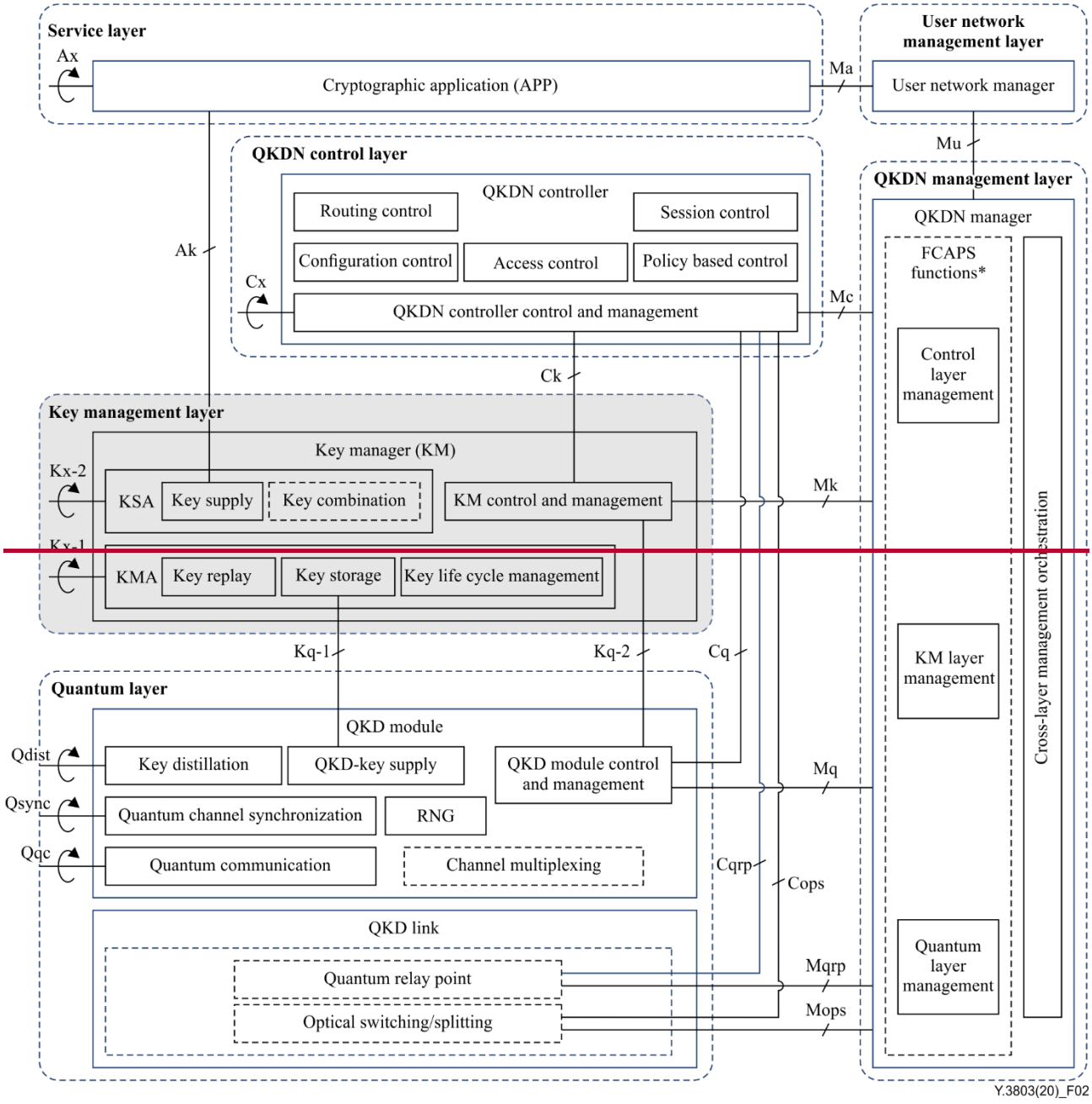


Figure 1 – Basic key management operations in a QKDN

7 Functional elements of key management

Figure 2 shows a functional architecture model of QKDN specified in [ITU-T Y.3802], highlighting the key management layer and reference points associated with it.


Service layer
Ax
Cryptographic application (APP)
Ma
User network management layer
User network manager
Mu
QKDN control layer
QKDN controller
Routing control
Session control
Configuration control
Access control
Policy based control
Ak
Cx
QKDN controller control and management
Mc
QKDN management layer
QKDN manager
FCAPS functions*
Control layer management
Ck
Key management layer
Key manager (KM)
Kx-2
KSA
Key supply
Key combination
KM control and management
Mk
Kx-1
KMA
Key replay
Key storage
Key life cycle management
Kq-1
Kq-2
Cq
KM layer management
Quantum layer
QKD module
Qdist
Key distillation
QKD-key supply
QKD module control and management
Mq
Qsync
Quantum channel synchronization
RNG
Qqc
Quantum communication
Channel multiplexing
Cqrp
Cops
QKD link
Quantum relay point
Optical switching/splitting
Mqrp
Mops
Quantum layer management
Cross-layer management orchestration
Y.3803(20)_F02

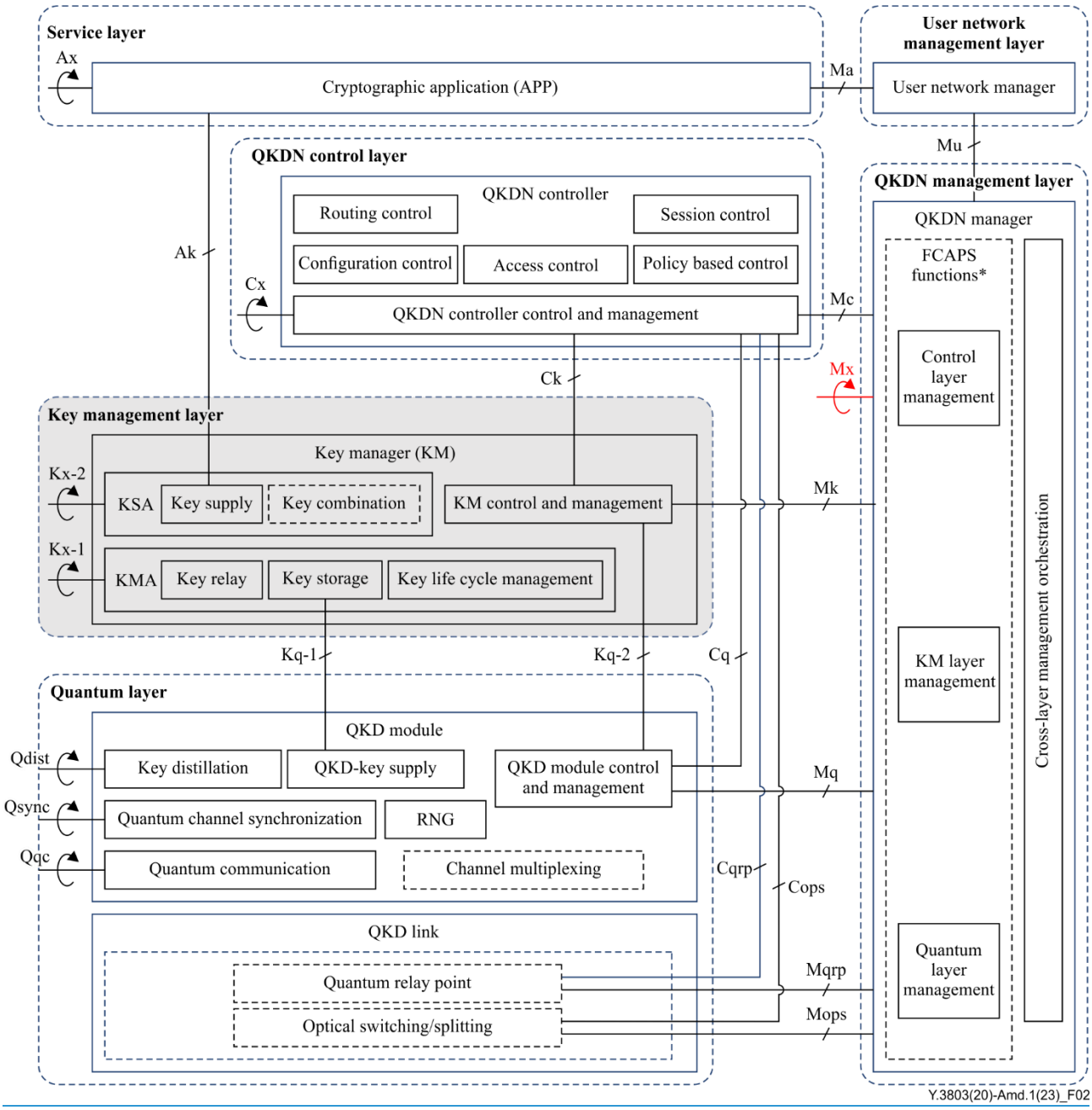


Figure 2 – A functional architecture model of QKDN

In order to design, implement, and execute the key management functions to fulfil the requirements specified in [ITU-T Y.3801], it is convenient and practical to identify and define two functional elements in the KM, i.e., a key management agent (KMA) and a key supply agent (KSA) as illustrated in Figure 3.

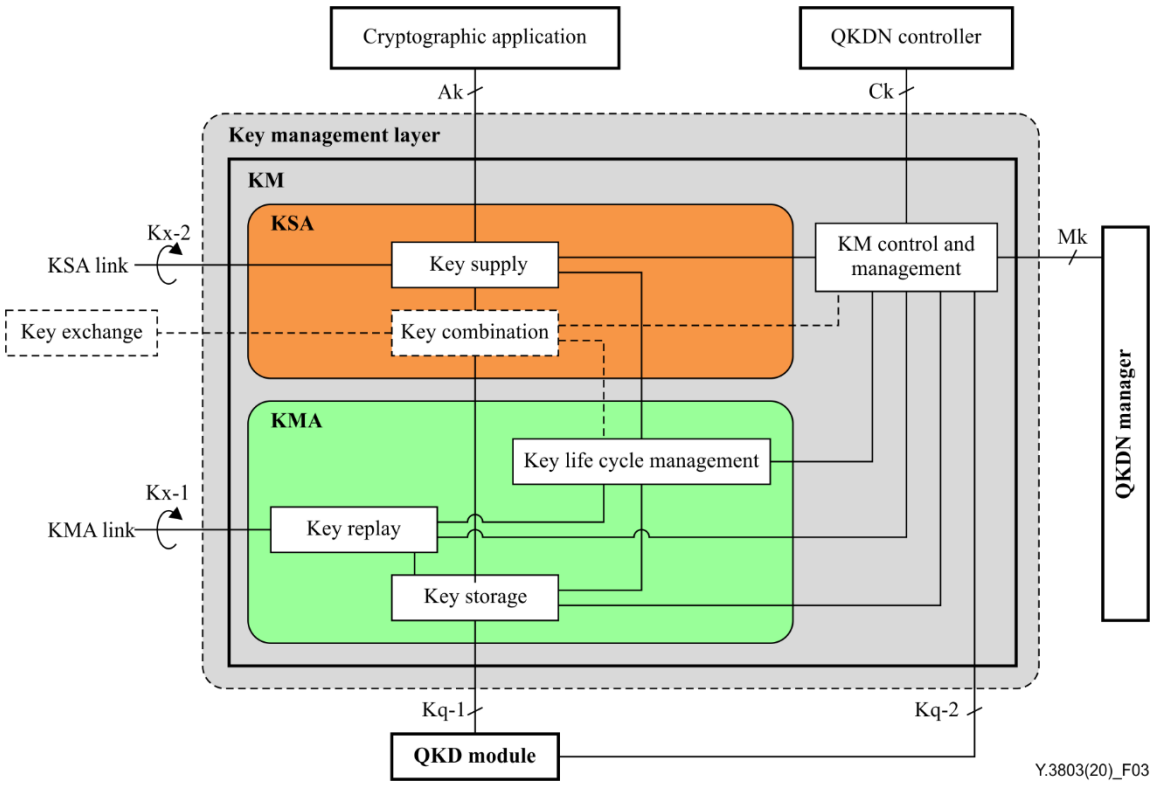


Figure 3 – Functional architecture model of the key management layer

The roles of these functional elements are described in clauses 7.1 to 7.4.

7.1 Agents for KM

a) KMA: It receives keys from QKD modules, and interconnects the QKD nodes by key relay, realizing the functions of:

– key storage;

– key relay;

– key life cycle management,

specified by requirements Req_KM 1~9 and 11 of [ITU-T Y.3801].

b) KSA: It is located between the KMA and a cryptographic application, and interfaces with the cryptographic application, realizing the function of

– key supply,

specified by requirement Req_KM 10 of [ITU-T Y.3801]. It houses libraries of application program interfaces to support various cryptographic applications. It optionally includes the function of

– key combination,

specified in [ITU-T X.1714]. The function of key combination associates a key from the KMA with another provided by a key exchange method, in such a way that the security of the combined key is unchanged from that of the input key from the KMA, and outputs the combined key to the function of key supply. Details of the security aspects of the key combination function lie outside the scope of this Recommendation.

c) In addition to the KMA and the KSA, the KM includes the function of

– KM control and management,

for communicating with the QKD module(s), the QKDN controller, and the QKDN manager to fulfil requirements Req_KM 4, 5 and 9 of [ITU-T Y.3801].

7.2 KM link

According to the identification of the two functional elements, the KM link consists of the KMA link and the KSA link. The KMA and KSA have processors, storage, and communication interfaces.

NOTE 1 – In a practical setting, the KMA and KSA, and the KMA link and KSA link can be implemented degenerately in a single server and a single link.

NOTE 2 – There would be the case where a QKD node has only the KMA but not the KSA, for example, acting as a key relay node only.

7.3 Reference points

The reference points relevant to the KM (Ak, Kq-1, Kq-2, Kx-1, Kx-2, Ck, Mk; see Figure 2) are specified in [ITU-T Y.3802].

7.4 Security demarcation boundary

A security domain is a common concept in information and network security. It is defined in [ITU-T Y.3800] as a boundary to demarcate one layer's responsibility on the keys to be supplied from another layer's responsibility on the use of the keys. It partitions a set of entities and parties that are subject to a single security policy. A security domain boundary often corresponds to a security responsibility demarcation point.

A boundary can be identified between the KSA and the cryptographic application to demarcate responsibility for management and consumption of the keys. In this case, once the keys are supplied to the cryptographic applications, they use the keys under their own responsibility, while the KMAs and KSAs should delete or preserve the keys according to the key management policy. It is highly recommended that the cryptographic application does not use the key more than once.

8 Operations of key management

Figure 4 depicts operations of key management based on the reference model, and illustrates how keys are managed and supplied to cryptographic applications.

In the case shown in Figure 4, QKD nodes 1 and 3 are not directly connected by a QKD link. In order for the two nodes to share the key, key relay is performed between KMA2 and KMA3 by using the key generated in the QKD modules at QKD node 2 and 3. The key is finally supplied from KSA1 and KSA3 to the cryptographic application that requested the key. The QKDN controller performs routing control for key relay. The QKDN manager monitors the status of the whole QKDN and manages the QKDN, supporting the KMAs, the KSAs and the QKDN controller when necessary. To maintain interconnectivity and expandability in the QKDN, an appropriate key format for key data with added metadata containing various types of information needs to be introduced. A logical set of the key data and the metadata is referred to as a key file in this Recommendation.

Based on the key file, the KMAs, the KSAs, and cryptographic applications communicate to each other the key data, the metadata and key management information. In addition, the QKDN controller and the QKDN manager communicate control and management information based on the metadata to the KMAs and the KSAs. In the following, basic key management procedures are described.

Security issues and methods lie outside the scope of this Recommendation.

NOTE 1 – In Figure 4, horizontal solid lines and vertical solid arrows represent KMA links and interfacing paths, respectively, on which keys are conveyed, and hence for which confidentiality, integrity and authenticity of the keys need to be considered. Horizontal dotted lines represent KSA links in which information on key synchronization and authentication are conveyed. Vertical dotted arrows from the cryptographic application to the KSA represent key requests. Other dotted arrows represent links for communicating QKDN control and

management information between connected entities. For these dotted lines and arrows, integrity of the information is a primary concern.

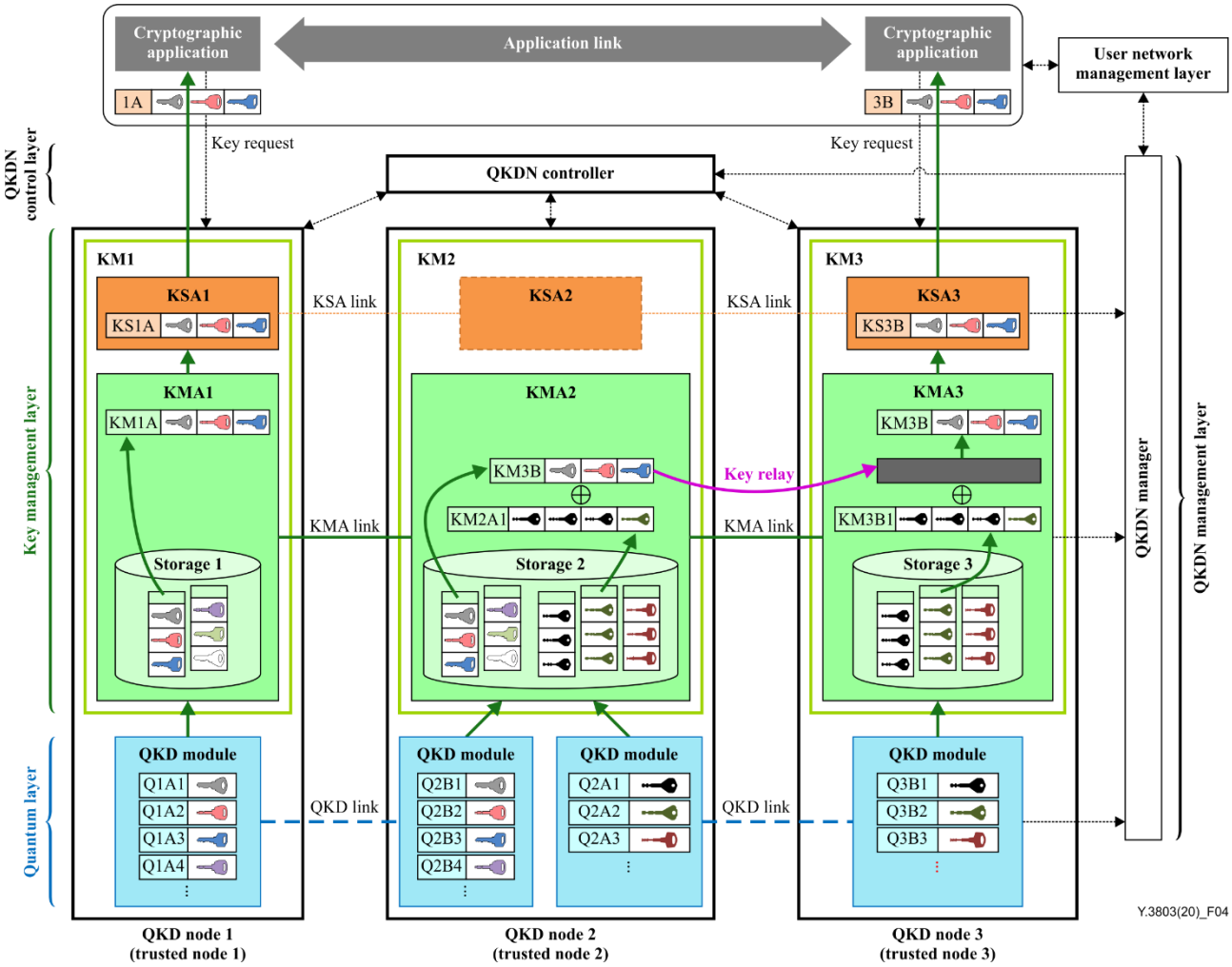


Figure 4 – Functional elements and operations of key management

NOTE 2 – Figure 4 describes a QKDN in the simplest way with three nodes. In such a simplification, while key relay can be featured, a routing control aspect cannot be well represented, but it is not excluded.

8.1 Generation of a QKD-key in the quantum layer

In the quantum layer, a pair of QKD modules generates a pair of symmetric (identical) random bit strings in its own way based on an information theoretically secure (IT-secure) protocol of QKD. Each QKD module is installed in a secure and reliable node against security threats (trusted node).

NOTE 1 – Conventionally, a sender (transmitter) and a receiver of keys or cipher texts are called "Alice" and "Bob", respectively, in the field of cryptography. In this Recommendation, Alice and Bob are often used to indicate QKD modules connected by a QKD link that share a key. In the case of QKD protocols in a so-called prepare-and-measure scheme, Alice and Bob are a sender and a receiver, respectively. In the case of QKD protocols based on measurement-assisted schemes, such as MDI-QKD and TF-QKD, mentioned in [ITU-T Y.3800], both Alice and Bob are transmitters, while receivers are located at an intermediate point on a quantum channel. In the case of entanglement-based QKD protocols, both Alice and Bob are receivers, while a transmitter of entangled quantum signals is located at an intermediate point on a quantum channel.

NOTE 2 – A QKD protocol means a list of steps for establishing symmetric cryptographic keys with information-theoretical security based on quantum information theory in this Recommendation.

The QKD protocol may differ for each pair of QKD modules, and each pair of QKD modules may be produced independently by a different vendor. In the following, the symmetric random bit string generated in the QKD module is referred to as a QKD-key, distinguished from a key which is re-sized

and formatted in the KMA and the key in the KSA. Unit lengths of QKD-keys produced by different QKD module pairs may be different from each other.

In each QKD module, metadata are generated and attached to the QKD-key, forming a key file. The pair of QKD modules transfer the QKD-key file to the corresponding KMAs.

The metadata for a QKD-key are specified in item (1) of Table 1.

8.2 Management of KMA-key and KSA-key in the key management layer

This clause specifies the operation executed in the key management layer, introducing two types of key data, i.e., KMA-key and KSA-key. The procedures include six main operations, which are explained in clauses 8.2.1 to 8.2.6. The order of these sub-clauses does not necessarily correspond to the actual temporal order of execution, which might vary depending on real situations and use cases.

8.2.1 Key acquisition, authentication, storage in KMA

Requirement Req_KM 2 of [ITU-T Y.3801] is that a KMA receives the QKD-key files from QKD module(s), which is/are located in the same QKD node, via an interface at the reference point Kq-1, and to store them securely when storage is necessary. The lengths of the acquired QKD-key files may differ from each other. Therefore, as recommended by Req_KM 3 of [ITU-T Y.3801], 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.

Before storing the buffered keys as key data, the pair of KMAs (say KMA1 and KMA2) which receive the pair of QKD-keys confirm the identity of the buffered keys. Therefore, as recommended by Req_KM 8 of [ITU-T Y.3801], the KMA has capabilities of key synchronization, entity authentication and message authentication. The pair of KMAs authenticate each other via the KMA link. Then one of the KMAs (say KMA1) sends the matching KMA (say KMA2) a key-authentication request including a hash value or a message authentication code of the buffered keys as well as the QKD-key identifier (ID). Then KMA2 synchronizes (in bit position) and authenticates the buffered keys by comparing the hash values or the message authentication code in its hand with that from KMA1. Security details of the hash value communication lie outside the scope of this Recommendation. If the hash values or the message authentication codes coincide with each other, KMA1 and KMA2 finally store the buffered keys as key data, which is referred to as KMA-key, with metadata in the key storage directory. Otherwise, KMA1 and KMA2 abort the buffered keys.

The metadata for KMA-key are specified in item (2) of Table 1.

8.2.2 Reception of key request from cryptographic application

A cryptographic application in the service layer sends a KSA a key request. The key request from cryptographic application may include information on a required security level, depending on key supply service policy, etc. Requirement Req_KM 10 of [ITU-T Y.3801] is that the KSA receives key requests from authorized cryptographic applications via a key supply interface at the reference point Ak. The control of reception of key request(s) by the KSA can be supported by the QKDN controller, especially when key requests are sent from multiple cryptographic applications.

The KSA then authenticates the cryptographic application by an appropriate means. Their certificate can be issued by an access control function of the QKDN controller, which manages an access control repository of registered functional components including cryptographic applications and KSAs. When the KSA and the cryptographic application establish authentication, they can exchange and share a common secret key for authentication for key request next time.

NOTE – The KSA and the cryptographic application can keep a part of the key as a secret key for the next authentication.

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 (e.g., key length, number of keys, node pair names or IDs, and KSA-key ID) for an individual key request. The KSA-key ID is a unique ID across the entire QKDN.

8.2.3 Key relay between KMAs

As recommended by Req_KM 6 of [ITU-T Y.3801], the KMAs support key relay through a key relay route between the two endpoint KMAs, employing highly secure encryption (e.g., one-time pad (OTP) [b-Shannon]). The key relay route is controlled by the QKDN controller as illustrated in Figure 1. The communication sessions between KMAs to establish the end-to-end key can be controlled by the QKDN controller.

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 encrypted by exclusive OR (XOR) with the other key shared by the neighbouring pair of QKD modules in an OTP manner, and are then sent from the source KMA to the destination KMA, thus realizing IT-secure key relay (see Figure 4). After decryption in the destination KMA, 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.

There is another option of key relay, namely, to use an extra random number generator (RNG) equipped in a KMA. The RNG in the source KMA generates a string of random numbers. The source KMA stores the random number string in its storage, and relays its copy to the destination KMA by the OTP-based key relay. Thus, a pair of symmetric random bit strings can be shared between the endpoint KMAs.

NOTE 1 – In Figure 4, KMA1 and KMA2 read the required number of keys from their storage as key files with the corresponding metadata, "KM1A" and "KM3B", respectively. In KMA2, the latter key file is encrypted by XOR with the other key (shared by the neighbouring pair of QKD modules) with metadata "KM2A1" in an OTP manner, then sent to KMA3 via the KMA link, and finally decrypted by the key with metadata "KM3B1". The key is thus shared between KMA1 and KMA3.

NOTE 2 – In the scheme shown in Figure 4, key relay is performed in a unit of a key file (metadata and key data) because implementation of key relay can be simplified, and the size of metadata is not long, typically a few 10s of bytes. For example, the relayed key file as a whole (the metadata "KM3B" and the key data) is encrypted by OTP using the other key data (with the metadata "KM2A1"). However, an option is not excluded in which metadata is not encrypted by the KMA-keys in an OTP manner.

NOTE 3 – The RNG should be non-deterministic: This can be realized with conventional physical noise-based schemes as specified in [b-ISO/IEC 18031] or with quantum principle-based schemes (quantum noise RNG).

As recommended by Req_KM 7 of [ITU-T Y.3801], in addition to the OTP encryption method, the KMA supports another encryption method (e.g., AES [b-ISO/IEC 18033-3], [b-FIPS PUB 197]) for key relay according to key management policy. If the necessary number of keys for OTP-encrypted key relay is not available, a backup method with symmetric key cipher such as AES can be used for key relay.

KMAs are recommended to create metadata to record an encryption method used for individual key relay, and use this metadata for key life cycle management of relayed KMA-keys.

In each key relay, it should be confirmed whether the content (identity) of the key shared by the source and destination KMAs has been altered in transit. Therefore, as recommended by Req_KM 8 of [ITU-T Y.3801], appropriate methods of entity and message authentication are employed to ensure

authenticity and integrity for the KMAs and KMA-key, respectively. In particular, for integrity protection of the KMA-key, a hash value or a message authentication code of the KMA-key data can be used. Integrity of key management information exchanged between KMAs is protected by performing message authentication. The detailed options for ensuring authenticity and integrity lie outside the scope of this Recommendation.

The metadata for a relayed KMA-key are specified in item (3) of Table 1.

8.2.4 Key supply from KSAs to cryptographic applications

After the KSA and the KMA authenticate each other, the KSA informs the KMA of the requested information (e.g., key length, number of keys, node pair names or IDs, KSA-key ID, and the security level of key). The KMA picks up the required number of keys 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.

When the cryptographic application requests keys that have the same security level as QKD-keys, the KMA is recommended to select KMA-keys that were relayed by OTP encryption, according to the metadata in the key relay encryption method.

The KMA then transfers this key to the KSA. The KSA receives this key, as exemplified in QKD nodes 1 and 3 in Figure 4.

NOTE – In order to adjust the key data length in the KMA in accordance with the requested length by the KSA, KMA-key data may sometimes be split and one part supplied to the KSA. In such a case, the remaining key data may be used in the subsequent key requests from the KSA. Then information to identify the remaining key data is needed in addition to the original KMA-key ID. For this purpose, some bit strings can be taken from the remaining key data and used as a new KMA-key ID.

The key received by the KSA is referred to as the KSA-key. In the KSA, the acquired KMA-key data can optionally be combined with another key provided by a key exchange method, and an output KSA-key is derived as specified in [ITU-T X.1714].

After the key is transferred from the KMA to the KSA, the KMA records its metadata in its storage, and may send metadata to the QKDN manager for key life cycle management as required by Req_KM 11 and Req_M 9 of [ITU-T Y.3801]. In the KSA, cryptographic application name, application source and destination IDs, KSA-key ID, and KSA-key length and supply timestamp are examples of KSA metadata that can be retained after key supply.

Finally, as recommended by Req_KM 8 of [ITU-T Y.3801], 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. Security details of the hash value communication lie outside the scope of this Recommendation.

After the preceding verification is completed, the key data with the KSA-key ID is supplied to the cryptographic application via the key supply interface at the reference point Ak, as required by Req_KM 10 of [ITU-T Y.3801]. KSA metadata is recorded in the storage of the KSA or sent to the QKDN manager for key life cycle management as required by Req_KM 11 and Req_M 9 of [ITU-T Y.3801]. The cryptographic application identifies the key data based on the KSA-key ID and uses it.

Once the keys are supplied to the cryptographic application, the KMAs and KSAs should apply the key management policy, such as deleting the key data from or preserving the key data in their storage, as required by Req_KM 10 of [ITU-T Y.3801].

The metadata for a KSA-key are specified in item (4) of Table 1.

The control of sessions for key supply can be supported by the QKDN controller, especially when keys are supplied to multiple cryptographic applications.

8.2.5 Rerouting of key relay between KMAs

Rerouting of key relay should be carried out, depending on the status of the key management layer or the quantum layer, to try to ensure continued availability of key supply. Typical cases include when the number of KMA-keys in the relay node(s) falls below a threshold, and when a high QBER prevents or reduces the rate of QKD key generation between the relay node(s) connected by a certain QKD link. For more details, see [ITU-T Y.3804].

As required by Req_KM 5 of [ITU-T Y.3801], the KMA provides information on key management, via the KM control and management function, to the QKDN controller and the QKDN manager.

As recommended by Req_KM 4 of [ITU-T Y.3801], the KM control and management function receives status information of the QKD module and optionally of the QKD link, such as QBER, key rate, QKD link status and alarm on fault, from QKD modules in the quantum layer.

The QKDN controller acquires the information on key management, such as key consumption rates, shared number of keys in the KMAs and KMA link status from the KMAs, as well as status information of the QKD module and optionally of the QKD link from the QKD modules.

Especially when a KMA is notified of an alarm on a QKD link connected to the relay node(s), the KMA forwards the alarm to the QKDN controller and the KMA continues key relay according to provision of rerouting by the QKDN controller.

Rerouting methods directed from the QKDN controller may include the following four cases.

a) Manual: A key relay route is set manually on the KMAs.

b) Fixed rate: Based on the key generation rates of the QKD links, the required frequencies of key relay are evaluated for node pairs, and a key relay instruction list is prepared. The KMAs carry out key relay according to this list.

c) Data-traffic adaptive: The KMAs perform key relay automatically based on recent statistical records of key relay traffic.

d) Scheduled: When it is possible to predict changes of a certain condition (e.g., key consumption or key generation rate), key relay is scheduled in advance.

8.2.6 Key life cycle management

As required by Req_KM 11 of [ITU-T Y.3801], each KMA stores information on key management activities on which it works, including reception, storage, formatting, relaying, synchronization, authentication, supply and deletion or preservation of keys. For example, each KMA collects and stores metadata in KMA-key files, and also sends them to the QKDN manager. Each KSA stores metadata in KSA-key files, and sends them to the QKDN manager.

The KMAs handle all transitions between phases of key life cycle, cooperating with the QKDN manager, which monitors, audits and keeps track of these workflows.

When an unexpected fault is detected, for example, if some KMA-keys are suspected to have been leaked to unauthorized parties or compromised somehow, the fault management function of the QKDN manager analyses the cause by tracing the key life cycle management information, and supervises the QKDN controller and the KMAs to take countermeasures. In the worst case, the KMAs delete relevant keys that might have correlations with the keys that are suspected to have been compromised.

9 Alternative scheme of key relay

A key relay scheme to share a key between the source node and destination node is illustrated in Figure 5. The Key_{AB} is generated between KMA-A and KMA-B. The Key_{AB} is relayed from KMA-B to KMA-C by OTP encryption with the Key_{BC}. It is relayed from KMA-C to KMA-D by OTP encryption with Key_{CD} and finally supplied to KSA-D.

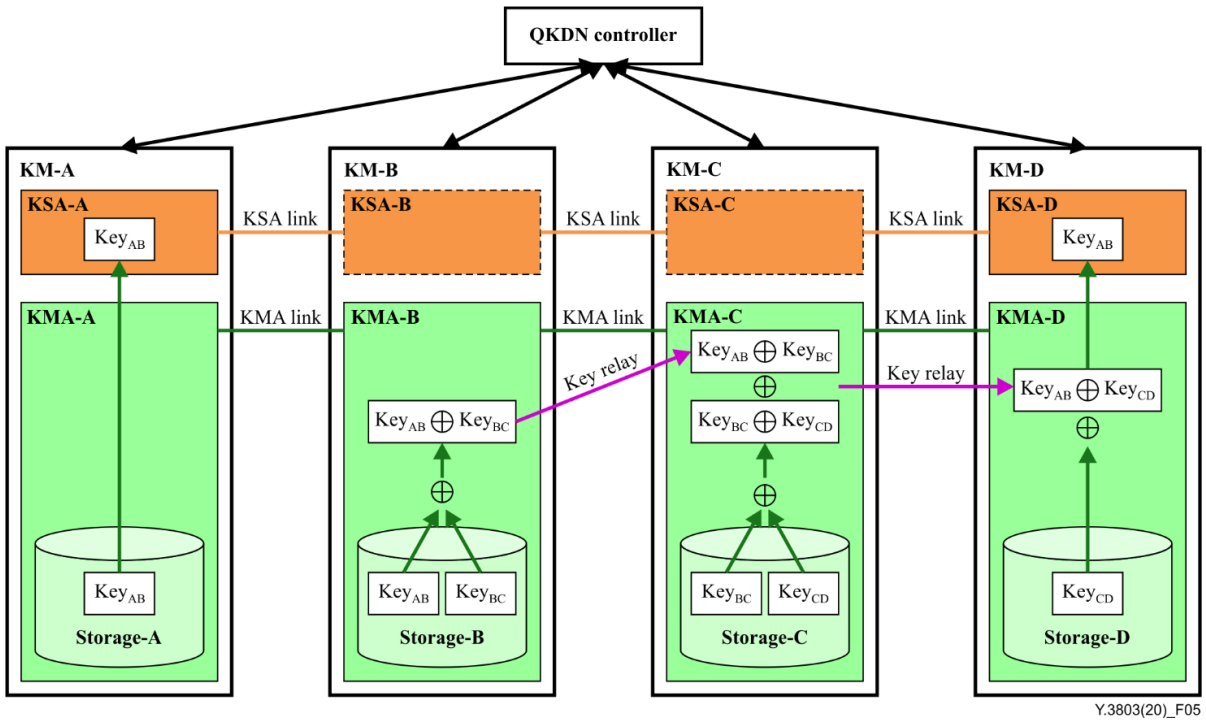


Figure 5 – Scheme of key relay from point to point in [ITU-T Y.3800] (case 1)

In case 2, illustrated in Figure 6, a random bit string Key_{RN} that is generated locally at KMA-A is used for key relay from KMA-A to KMA-D.

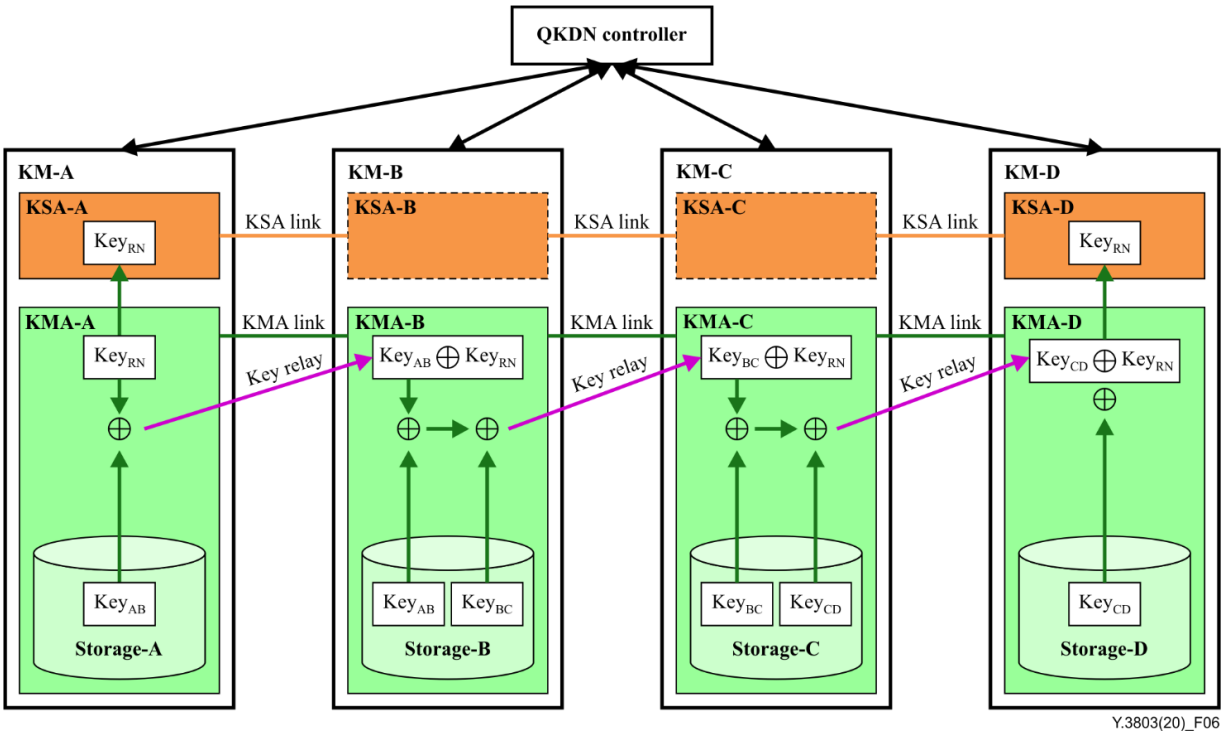


Figure 6 – Scheme of key relay from point to point in [ITU-T Y.3800] (case 2)

There are two problems for the schemes case 1 and case 2 of key relay from point to point in real application. One is that the XOR cipher text key and the corresponding decryption key may coexist on one key relay node, which has a security risk. The other is that the network has complex KMA

links between nodes, which results in implementation difficulty. To help alleviate these problems, two modified schemes of key relay are described in the following.

In contrast to the scheme in Figure 5, the scheme in Figure 7 is modified as follows:

1) the KMA link of one node connects to the destination node rather than a neighbouring node;

2) the XOR ciphertext key is directly sent to the destination node;

3) the relaying key is decrypted at the destination node by XORing all the ciphertext received.

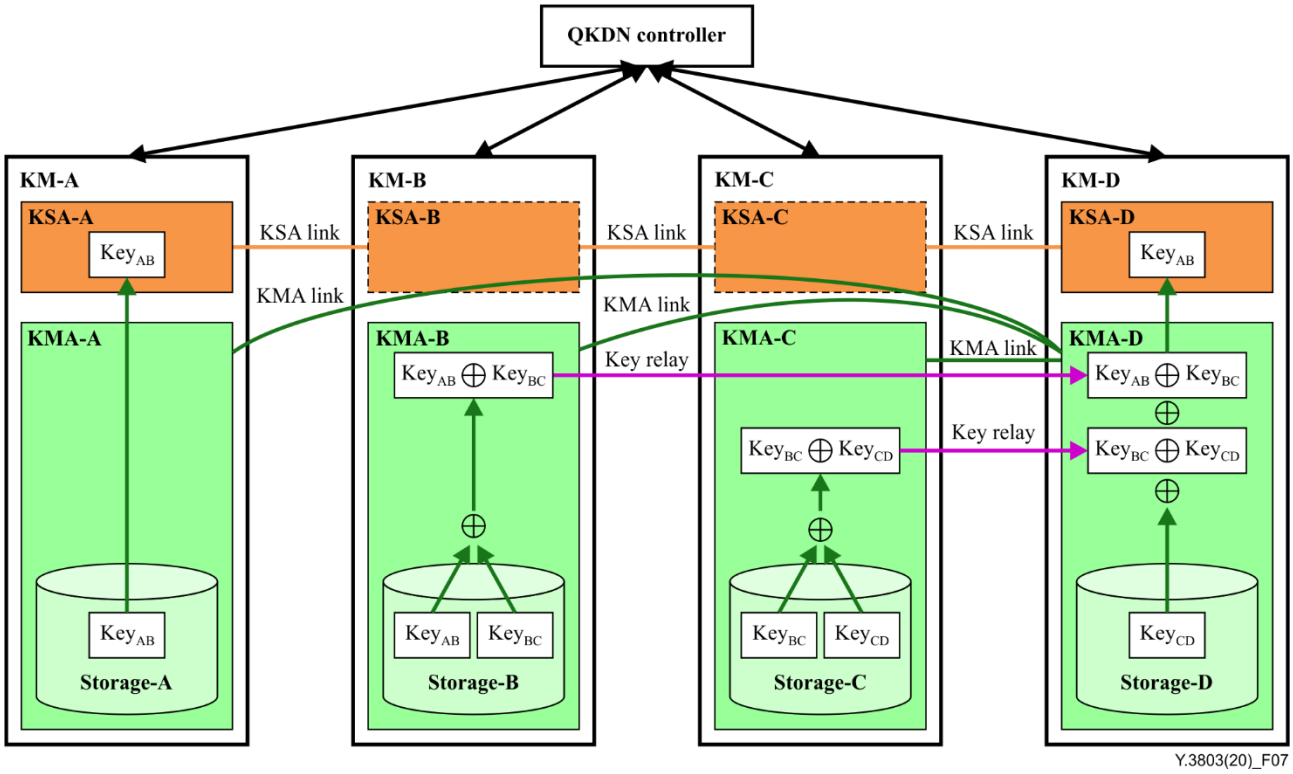


Figure 7 – Scheme of key relay with XORs uniformly processed at destination node

The function of the key relay node is simplified under this scheme. Only the XOR ciphertext key is sent out by the key relay node, and no neighbouring relays are required. In this way, the relaying key does not go through the nodes along the route and yet its XOR ciphertext does not coexist with the corresponding decryption key on the same node.

However, this scheme does not help to simplify the complex KMA links in the network. Therefore, it is only applied to some specific scenarios such as the trunk-line network for link length extension.

In contrast to the scheme in Figure 7, the scheme in Figure 8 is modified as follows:

1) XOR processes are all done in a centralized node, which will send the XOR ciphertext key to the destination node;

2) KMA links are set between each KMA and the centralized KMA;

3) a QKDN controller link is set between the QKDN controller and the centralized KMA.

KMA links are simplified under this scheme, and only KMA links between the nodes and the centralized node exist, thus facilitating network implementation.

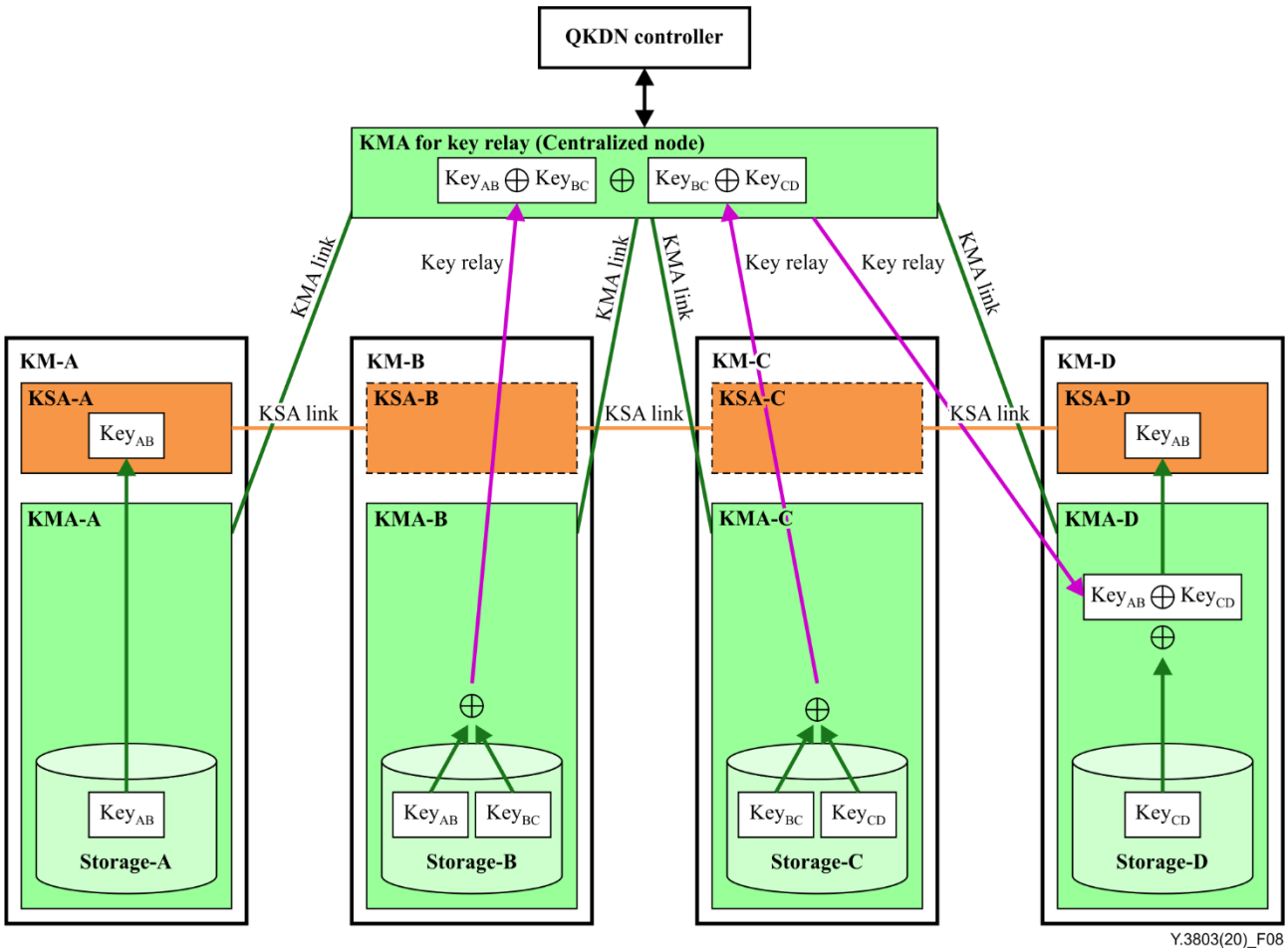


Figure 8 – Scheme of key relay with XORs collected at one centralized node

10 Key file format

A key file consists of key data in a prescribed size and metadata that includes items required for key management. Contents of metadata depend on QKDN architectures, for example, centralized architecture or distributed architecture, and also on use cases. Metadata of a key file may be stored in a distributed manner, e.g., on each KMA and KSA. Table 1 summarizes basic metadata items.

Table 1 – Metadata information (basic information)

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 that constitutes the pair of Alice and Bob	O
Generation time stamp	Time stamp of QKD-key generation at the pair of QKD modules	O

Table 1 – Metadata information (basic information)

Metadata	Description	M/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 QKDN. 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 that generates the QKD-key corresponding to the KMA-key data	O
Matching QKD module ID	ID to identify the matching QKD module that 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 NOTE – Some use cases will be studied, and metadata structures will be described in detail. New items of metadata, such as priority queuing control, quality of service control, and so on, will be described as extensions.		

Bibliography

[b-ETSI GR QKD 007] ETSI Group Report GR QKD 007 V1.1.1 (2018), *Quantum key distribution (QKD); Vocabulary*.

[b-ETSI GS QKD 008] ETSI Group Specification GS QKD 008 V1.1.1 (2010), *Quantum key distribution (QKD); QKD module security specification*.

[b-ISO/IEC 18031] ISO/IEC 18031:2011, *Information technology – Security techniques – Random bit generation.*

[b-ISO/IEC 18033-3] ISO/IEC 18033-3:2010, *Information technology Security techniques Encryption algorithms Part 3: Block ciphers.*

[b-FIPS PUB 197] Federal Information Processing Standards Publication 197 (2001), *Announcing the advanced encryption standard (AES)*.

[b-Shannon] Shannon, C.E. (1949). Communication theory of secrecy systems. *Bell Syst. Tech. J.* **28**, pp. 666–682.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities**
Series Z	Languages and general software aspects for telecommunication systems

Published in Switzerland
Geneva, 2024