INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 2017-2020

SG13-TD552/WP3

STUDY GROUP 13

Original: English

Question(s): 16/13 | Virtual, 1 -12 March 2021

TD

Source:	Editor
Title:	Draft Recommendation ITU T Y.QKDN_frint: "Framework for integration of QKDN and secure network infrastructures"
Purpose:	Proposal

Contact:	Kaoru Kenyoshi NICT Japan	E-mail: kaoru.kenyoshi@nict.go.jp

Keywords: integration of QKDN (quantum key distribution), phase in scenarios, PKI (public key infrastructure), QKD, QKDN (QKD network), secure storage network (SSN)

Abstract: This document includes the revised draft of Recommendation ITU-T Y.QKDN_frint "Framework for integration of QKDN and secure network infrastructures"

Summary

This TD is the outcome of Q16/13 (e-meeting, 1- 12 March 2021). This output document is the revised draft of Recommendation ITU T Y.QKDN_frint "Framework for integration of QKDN and secure network infrastructures" based on the contribution C1181 with modifications in the Q16/13 meetings during SG13 meeting.

This TD includes proposed texts for further review and editor's comments for guidance to invite contributions at the next meeting. The current status of this document including structure is premature and it needs discussions based on the contribution.

It is also recognized that close collaboration and harmonization with associated WI X.sec_QKDN_intrq in Q4/17.

Attachments:

Annex I Draft Recommendation ITU T Y.QKDN_frint "Framework for integration of QKDN and secure network infrastructures"

Annex I: Draft of Y.QKDN_frint

Draft Recommendation ITU-T Y.QKDN_frint

Framework for integration of QKDN and secure network infrastructures

Summary

For quantum key distribution networks (QKDN), Recommendation ITU-T Y.QKDN_frint specifies overview of secure storage networks (SSNs). It also specifies functional requirements, functional architecture model, reference points and phase-in scenarios for SSNs.

Keywords

integration of QKDN (quantum key distribution), phase in scenarios, PKI (public key infrastructure), QKD, QKDN (QKD network), secure storage network (SSN)

Table of Contents

Draft Recommendation ITU-T Y.QKDN_frint

Framework for integration of QKDN and secure network infrastructures

Editor's note – Another option for the title of this draft Recommendation is "*Framework for integration of QKDN and secure storage network".*

1 Scope

This Recommendation describes framework for integrating QKDN with conventional and emerging secure network infrastructures.

In particular, the scope of this Recommendation includes:

- overview of secure storage network (SSN);
- functional requirements for SSN;
- functional architecture model of SSN;
- reference points;
- operational procedures;
- phase-in scenarios.

2 References

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

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

[ITU-T Y.3802] Recommendation ITU-T Y.3802 (2020), *Functional architecture of the quantum key distribution network*.

[ITU-T Y.3803] Recommendation ITU-T Y.3803 (2020), *Key management for quantum key distribution Networks*

[ITU-T Y.3804] Recommendation ITU-T Y.3804 (2020), *Control and* Management for Quantum Key Distribution Networks.

[ITU-T X.509] Recommendation ITU-T X.509 (2016), *Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 **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.

3.1.2 **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.

3.1.3 **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.

3.1.4 **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).

3.1.5 **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.

3.1.6 **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.

3.1.7 **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.

3.1.8 **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).

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AES	Advanced Encryption Standard
CA	Certification Authority
FCAPS	Fault, Configuration, Accounting, Performance and Security
IT-secure	Information-theoretically secure
KM	Key manager
OTP	One-time pad encryption
PKI	Public Key infrastructure
QKD	Quantum Key Distribution
QKDN	QKD Network
SSA	Secure Storage Agent
SSN	Secure Storage Network

5 Conventions

In this Recommendation:

The keywords "is required to" indicate a requirement which must be strictly followed and from which no deviation is permitted if conformance to this document is to be claimed.

The keywords "is recommended" indicate a requirement which is recommended but which is not absolutely required. Thus this requirement need not be present to claim conformance.

6 Introduction

Editor' Note: some sentences are added to clarify the scope of this Recommendation.

The purpose of introducing the QKDN into current communication networks and cryptographic infrastructures is to enhance their security level by supplying highly secure symmetric keys to cryptographic applications. Introducing the QKDN into these existing infrastructures can impose significant overhead cost/impacts on systems and elements, and in the worst case may also introduce new vulnerabilities if the QKDN is not appropriately designed, operated, and interfaced to the cryptographic applications.

In order to support the QKDN, various kinds of cryptographic methods need to be used in appropriate combinations. To realize confidentiality protection of keys, e.g., in key relay via trusted nodes, one-time pad encryption (OTP), which is an information theoretically secure scheme, is recommended for ensuring a long-term confidentiality of keys. To realize integrity protection of the keys to ensure that the keys remain unaltered, cryptographic methods such as public key cryptography and hash functions, which are computationally secure, can be employed. These methods also play important roles to realize authentication and access control of functional elements in the QKDN. Control and management information in the QKDN needs to be protected by the combination of public key cryptography (especially for authentication and key exchange) and symmetric cipher such as AES (especially for data encryption). Cipher suites of these cryptographic technologies are implemented in IPsec and TLS based on public key infrastructure (PKI). Thus building the QKDN means integration of QKD technologies and existing secure network infrastructures.

Keys supplied by the QKDN can be used to encrypt sensitive and high-value data in transmission. Although the QKDN itself cannot protect confidentiality of data storage, it can be used to enhance the security of storage networks. In fact, today digital data are stored in data centers forever, and can easily be targeted by malicious attacks or even be threatened by non-malicious incidents like natural disasters. Protection of critical data in storage networks for a long term warrants the use of the QKDN, and should be worth the overhead cost. A secure storage network (SSN) consists of multiple data servers and is supported by a secret sharing scheme. Some secret sharing schemes, such as Shamir threshold scheme, ensure information theoretic confidentiality of storage, provided that the number of corrupted servers is less than a certain threshold, and data shares are exchanged through highly private channels. These highly private channels can be realized by using OTP encryption with keys supplied by the QKDN. To realize authentication, access control, and integrity protection in the SSN, PKI plays again an essential role.

A concept of integration of the QKDN with PKI and the SSN can be depicted in Figure 1. This is a typical example of integration of QKDN and secure network infrastructures.

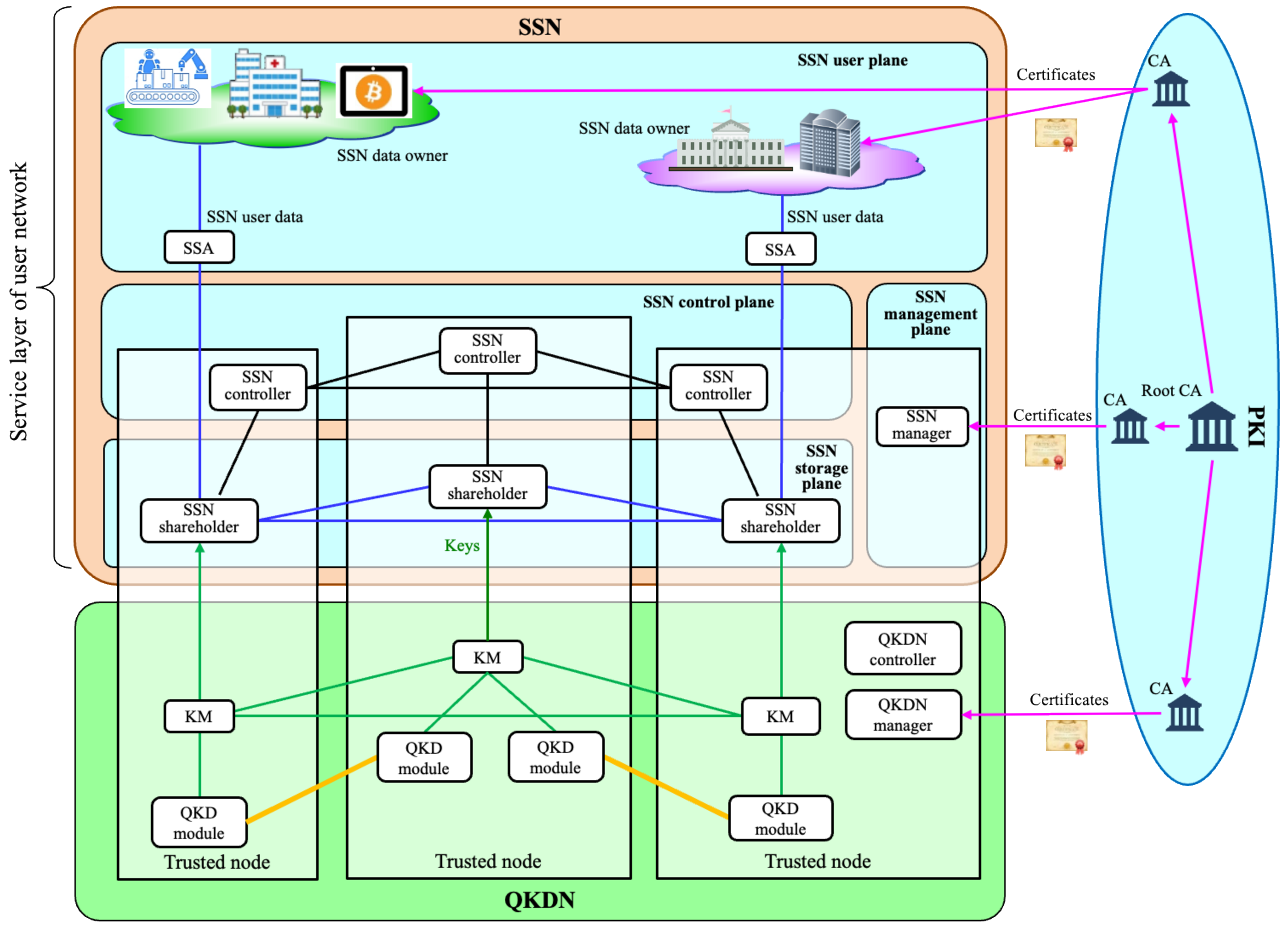

Figure 1. A conceptual view of integration of the QKDN with PKI and the SSN

The following functional elements are contained in the SSN in figure 1:

- Secure storage agent (SSA): a functional element which create shares from the original data and reconstruct the original data from shares.
- SSN controller: a functional element which controls the secret sharing process, i.e. receive the original data, encrypt them appropriately (e.g. transform them to shares by a secret sharing protocol), and control communication for the SSN shareholder.
- SSN manager: a functional element which manages FCAPS functions of the SSN.
- SSN shareholder: a functional element which distributes and stores shares.
- SSN shareholder link: a communication link between SSAs and SSN shareholders and among SSN shareholders. SSN shareholder links are shown in blue in figure 1. These links transmit shares with highly secure encryption such as OTP.
- SSN control link: a communication link among SSN controllers and between an SSN controller and an SSN share holder. SSN control links are shown in black in figure 1. These links transmit control and management information between the SSN controller and the SSN shareholder.

7 PKI

Editor's Note: This clause refers to the existing and on-going works on QKDN and PKI, especially from the viewpoint of how to support the QKDN with PKI. Missing specifications for integration of the QKDN and PKI will then be identified.

Editor's Note: We study the relations between the QKDN and PKI. An architecture and functional elements of QKDN are specified in [ITU-T Y.3800], [ITU-T Y.3801] and [ITU-T Y.3802]. The details of public-key certificates and attribute certificates are specified in [ITU-T X.509]. We consider how PKI support the QKDN and introduce necessary concepts and functions to integrate them. Some relevant work items are going on in ETSI ISG-QKD, harmonization with them need to be considered.

8 SSN

Editor's Note: Secure storage network is one of representative examples in the service layer. We review basic concepts and underlying technologies of the secure storage network, including secret sharing, control plane, storage planes, shareholders, private channels etc. Particular attention is put to the long-term security. Some relevant work items on storage security are going on in ISO/IEC, harmonization with them need to be considered. We introduce several subclauses as follows.

Editor's note: The following sentences are duplicated in X.sec_QKDN_intrq. Harmonization between them need to be considered.

8.1 Secret sharing

Editor's note – This clause will specify the secret sharing technologies.

Secret sharing satisfies confidentiality of storage, availability, and functionality. In secret sharing, new multiple data shares are created from the original data by using a polynomial, and stored in multiple data servers (shareholders). Shamir's (k, n) threshold scheme uses n shareholders, and restores the original data by collecting at least k $(\leq n)$ of shares. With shares of k–1 or less, the original data can never be reconstructed even with unlimited computing power. Provided that the number of corrupted shareholders is less than k, and shares are exchanged through private channels, Shamir's (k, n) threshold scheme ensures information theoretic confidentiality of storage, that is, confidentiality is satisfied. Shares can be added and multiplied, meaning that full homomorphism—functionality—can be met. Even if shares up to n–k are lost, the original data can be reconstructed by using the k remaining shares, which provides availability. However, this scheme cannot protect integrity. Private channels should also be implemented somehow to protect confidentiality of data transmission, which is another important confidentiality requirement.

8.2 Private channels supported by QKDN

Editor's note – This clause will specify how the QKDN can be combined for the secure storage network.

The secret sharing method itself has a mathematical algorism and does not provide a solution to transmit a share securely to the remote storage (i.e., ensure the confidentiality of transmission). Combined with the QKDN, which realizes confidentiality of data transmission, and secret sharing method can be used for information theoretically SSN in a protocol level.

8.3 Secure operation supported by PKI

Editor's note – This clause will specify how PKI supports secure storage network.

QKD and secret sharing can realize confidentiality and availability of data, but they cannot prevent corruption of long-term preservation of data. Therefore, it is necessary to introduce security technologies such as digital signatures in the system. These functions are performed at the certification authority (CA) in the PKI in figure 1. It should be noted that for integrity protection, it is sufficient to ensure short-term security for a certain period. Timestamp chains are used to prolong the validity of digital signatures for the original data for any length of time. For example, a

Pedersen commitment scheme is adopted, and commitments to the original data are timestamped and shared. While this scheme can protect the secrecy of the original data information theoretically, the correctness of the data must inevitably be computational. So the commitments as well as the timestamps are renewed regularly. Thus the long-term integrity protection can be realized.

Editor's note – Proposed texts in clause 9 and 10 are the first version for reviews.

9 Functional requirements for SSN

Editor's note – Additional requirements might be needed for QKDN to supply keys to SSN.

9.1 SSN user plane

An SSA should meet the following requirements.

Req_SSN_A 1. The SSA is required to receive the original data from data owners.

Req_SSN_A 2. The SSA is required to create shares of the original data.

Req_SSN_A 3. The SSA is required to send shares to SSN shareholders.

Req_SSN_A 4. When the data owners request restoring the original data, the SSA is required to reconstruct the original data from shares.

9.2 SSN control plane

An SSN controller should meet the following requirements.

Req_SSN_C 1. The SSN controller is required to control distribution of shares to SSN shareholders.

Req_SSN_C 2. The SSN controller is required to control collection of shares from SSN shareholders to reconstruct the original data.

Req_SSN_C 3. When failure occurs in an SSN shareholder, the SSN controller is required to control re-sharing of shares.

Req_SSN_C 4. The SSN controller is required to receive certificates from CAs and use them for security functions.

Req_SSN_C 5. The SSN controller is required to encrypt control and management information between SSN controllers.

Req_SSN_C 6. The SSN controller is required to manage configuration of SSN shareholders.

9.3 SSN storage plane

An SSN shareholder should meet the following requirements.

Req_SSN_S 1. The SSN shareholder is required to receive shares from SSAs.

Req_SSN_S 2. The SSN shareholder is required to transmit the shares to other SSN shareholder with IT-secure encryption such as OTP and store them under the control of the SSN controller.

Req_SSN_S 3. The SSN shareholder is required to send shares to SSAs with IT-secure encryption such as OTP when the original data are requested.

Req_SSN_S 4. When failure occurs in an SSN shareholder, the SSN controller is required to perform re-sharing of shares.

9.4 SSN management plane

An SSN manager should meet the following requirements.

Req_SSN_M 1. The SSN manager is required to provide FCAPS management of the SSN control plane and the SSN storage plane.

10 Functional architecture model of SSN

Figure 2 illustrates the functional architecture model of SSN.

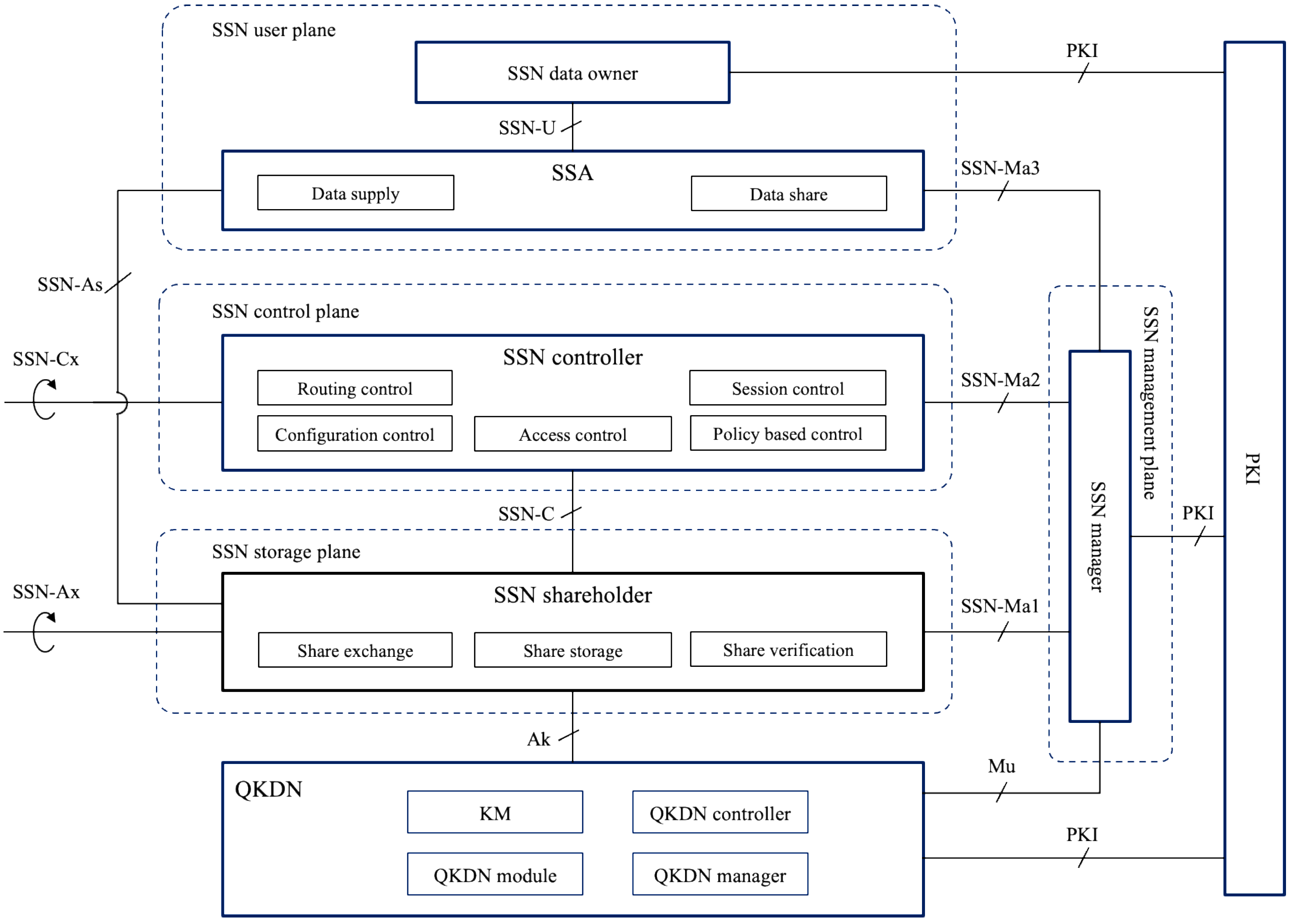

Figure 2 Functional architecture model of SSN

Editor's note – Further interfaces between SSA and SSN controller, and SSN controller and PKI might be necessary.

10.1 Functions of SSA

In the SSN user plane, an SSA is to create shares and reconstruct the original data. It is further comprised of the following functional elements.

- Data supply function: It receives original data from data owners and sends the original data to data owners with highly secure encryption (e.g., OTP is recommended).
- Data share function: It supports creating shares of the original data and reconstruct the original data from shares.

10.2 Functions of SSN controller

In the SSN control plane, an SSN controller is to control functions in SSN storage plane. It is further comprised of the following functional elements.

- Session control function: It supports the controls the session procedures among SSN shareholders, between the SSN controller and the shareholders and between the SSA and the SSN shareholders.

- Routing control function: It provisions an appropriate distribution route among SSN shareholders and also performs routing of re-sharing shares depending on fault, performance, and/or availability status of the SSN shareholder links for ensuring the continuation of secret sharing.
- Configuration control function: It performs the acquisition of configuration information on SSN shareholders, SSN controllers, SSN shareholder links and SSN control links, and the state of these components (e.g., in service, out of service, standby, or reserved). It conducts the reconfiguration of SSN shareholders and SSN shareholder links if an alarm including the result of failure diagnosis is notified.
- Policy based control function: It controls the SSN resources based on the quality of service (QoS) and charging policies for SSN data owners.
- Access control function: It provides capabilities to verify the claimed identity of functions and functional elements under control and support by the SSN controller (i.e., authentication), and to restrict them to pre-authorized activities or roles by access rights based on enforced policies (i.e., authorization).

10.3 Functions of SSN shareholder

In the SSN storage plane, an SSN shareholder is to exchange shares with other shareholders and store them. It is further comprised of the following functional elements.

- Share exchange function: It receives shares from SSAs and exchanges shares with other SSN shareholders with highly secure encryption (e.g., OTP is recommended).
- Share storage function: It stores shares securely.
- Share verification function: It verifies integrity of shares with certificates.

10.4 Functions of SSN manager

In the SSN management plane, an SSN manager supports FCAPS functions of SSAs, SSN controllers and SSN shareholders.

11 Reference points

11.1 Reference points of SSA

The following reference points are relevant to connections with an SSA.

- SSN-U: a reference point connecting an SSN data owner and an SSA. It is responsible for sending the SSN user data.

11.2 Reference points of SSN controller

The following reference points are relevant to connections with an SSN controller.

- SSN-C: a reference point connecting an SSN controller and an SSN shareholder. It is responsible for the SSN controller to communicate control information with the SSN shareholder.
- SSN-Cx: a reference point connecting two SSN controllers. It is responsible for the two SSN controllers to communicate control information each other.

11.3 Reference points of SSN shareholder

The following reference points are relevant to connections with an SSN shareholder.

- SSN-As: a reference point connecting an SSA and an SSN shareholder. It is responsible for sending shares from the SSA to the SSN shareholder, and supplying the shares from the SSN shareholder to the SSA.
- SSN-Ax: a reference point connecting an SSN shareholder and other SSN shareholder. It is responsible for sending shares to other SSN shareholders.

11.4 Reference points of SSN manager

The following reference points are relevant to connections with an SSN manager.

- SSN- Ma1: a reference point connecting an SSN manger with an SSN shareholder. It is responsible for the SSN manager to communicate management information with the SSN shareholder.
- SSN- Ma2: a reference point connecting an SSN manager with an SSN controller. It is responsible for the SSN manager to communicate management information with the SSN controller.
- SSN- Ma3: a reference point connecting an SSN manager with an SSA. It is responsible for the SSN manager to communicate management information with the SSA.

11.5 Reference points of QKDN

Ak and Mu reference points are defined in [ITU-T Y.3802].

11.6 Reference points of PKI

Editor's note – Details of interfaces of PKI might be outside the scope of this Recommendation. Further gap analysis is needed.

12 Operational procedures

Editor's note - Descriptions will be added.

13 Phase-in scenarios

Editor's note – Texts will be added. Migration issues will also be considered.