ISBN: 978-89-294-1163-3

양자키관리 표준화 분석 연구 보고서

2020. 11 . 30

한국과학기술정보연구원

차 례

그 림 차 례

표 차 례

제1장 서 론

제1장 서 론

1.1 연구 배경

QKD를 통하여 양자적 현상으로 분배된 키를 관리하기 위해서는 보안성을 비롯한 다양한 분야를 고려하여 설계되어야 한다. 특히 ETSI, ITU-T 등에서 제정되고 있는 표준화기법을 따라 설계되어야할 것이다.

1) QKD 시스템은 단대단(Point to Point) P2P 키분배만이 가능하므로 네트워크 레벨의 키 분배를 위해서는 양자키 관리시스템이 필수적으로 구축되어야 하며, 2) QKD시스템의 공급자에 따라 양자키 관리시스템이 다를 수 있으므로 이를 통합, 관리할 수 있는 양자키 관리 및 운영체계가 필요하다. 3) 국가연구망 구축사업의 데이터 전송 보안에 대한 통합적인 관리를 위해서는 현재 개발되고 있는 QKD 시스템의 기술 특성과 개발 방향, 키관리 계층의 암호화 동향 등을 폭넓게 검토하여, 종합적으로 키관리 시스템 설계 방안을 도출하기 위한 설계 전략 수립이 필요하다. 4) 또한 국가과학기술연구망의 백본회선 구축 및 연구지원 회선 간의 서비스를 제공하기 위한, 양자암호기반의 키분배 시스템 환경 구축 및 양자암호통신망의 서비스 제공을 위한 종합적인 키관리 시스템 체계 연구가 반드시 필요하다.

1.2 목적

본 연구에서는 ① QKD 기술 전반 및 표준화 동향을 분석하여 ② 양자키 관리 연계 QKD 네트워크 구조를 연구하고 이를 바탕으로 ③ 양자키 관리시스템 기술연구 및 설계 전략을 수립 연구 수행한다.

제2장 양자암호통신 기술표준화 및 동향분석

제2장 양자암호통신 기술표준화 및 동향분석

2.1. QKD네트워크 KMS 표준화 및 기술 동향 분석

2.1.1. 주요 표준 조사

○ **요구사항 및 기능(ITU-T)**
▷ QKD KMS에 요구되는 기능
○ **구성요소 및 구조(ITU-T)**
▷ 서비스 제공을 위해 필요한 구성 요소 및 구조
○ **관리 시스템 및 기능별 동작 절차(ITU-T)**
▷ QKD와 KMS 관리 및 제어를 위한 시스템 구성 및 인터페이스, 주요 기능별 동작 절차
○ **연동 인터페이스(ETSI)**
▷ QKD와 KMS간 키스트림 전달을 위한 연동인터페이스
▷ KMS에서 키 사용자에게 키 공급을 위한 연동인터페이스
○ **보안 (ITU-T, ISO)**
▷ 시스템 보호, 구성 요소간 통신 보호, QKD를 통해 분배된 키 보호, 인증 및 접근제어, 키 보호, 보안경계
○ **키관리 (NIST)**
▷ 키 생성, 사용, 폐기 등의 생애 주기 관리
○ **데이터모델링(TTA, ITU-T, ETSI, NIST등 종합적으로 자료 참고)**
▷ 시스템 동작을 위한 운영정보, 키 관리 정책, 키 데이터 저장을 위한 개체 구분 및 필요 정보
○ **기타요소기술**
▷ 키 식별 및 장비식별을 위한 UUID, 키 데이터 유효성 검사 방안

2.2. 요소별 주요 표준 사항

2.2.1. 이종 장비 연동(CID – 1)

이종 장비의 연동을 위해서는 구조적으로 키관리 개체와 QKD개체가 분리되어야 한다[R-1].

[R-1] QKD 개체 - 키 관리 개체 분리구조
QKD 개체 - 키 관리 개체 분리구조는 QKD 개체와 키 관리 개체가 독립적으로 존재하며 벤더에 상관없이 개방형, 표준화된 인터페이스를 통해 멀티벤더 QKD 모듈/애플리케이션을 수용할 수 있는 구조이다. QKD 모듈의 추가, 삭제와 같은 양자암호 전달 네트워크의 구성 변경이 용이하며 N:N 네트워크를 수용할 수 있는 구조이다.

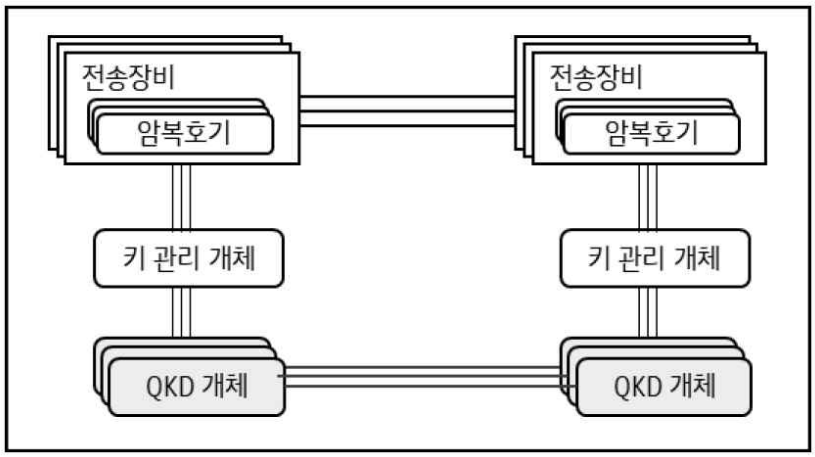

그림 2.1 QKD & 키 관리 개체 분리구조

2.2.2. QKD-KMS 연동(CID - 2)

현재 KMS와 QKD 연동에 대해서는 표준화 진행이 미흡한 상황이며 [R-7]에 QKD Key File과 KMS와의 연동 인터페이스에 대한 언급이 되어 있으나 구체적인 내용은 부족한 상황으로 본 기술에 대해서는 [R-7]의 내용에 기초하여 연동을 위한 data의 구성과 인터페이스를 정의 하였다.

[R-7] QKD Key File

The QKD protocol may differ for each pair of QKD modules, and each pair of QKD modules may be produced independently by different vendor. Below, 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 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.**

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

Metadata	Description	M/O
(1) QKD-key		
QKD-key ID	ID of the QKD-key.	M
Key length	Key length of the QKD-key	O
QKD module ID	ID of the QKD module (Alice or Bob) that generates the QKD-key	O
Matching QKD module ID	ID to identify the matching QKD module which constitutes the pair of Alice and Bob	O
Generation time stamp	Time stamp of QKD-key generation at the pair of QKD modules	O
Hash value	Hash value of the QKD-key data. (There are several options for hash function, which are discussed in other Recommendations.)	O

2.2.3. 키 공급 연동(CID – 3)

KMS와 키 사용자 간의 연동 인터페이스는 이종 장비간 호환성 및 구현 용의성을 위해 Rest Based API 프로토콜[R-10]을 ETSI의 표준을 참고한다. 그러나, [R-10] 표준에서는 GetKey 시에는 요청 Key size가 전달이 되나, GetKeyWithId 시에는 Key size 전달이 없어 KMS에서 미리 다양한 size의 Key를 생성해 놔야 해서 효율적인 키 관리가 어렵다는 단점이 있다. 본 설계에서는 KMS에서 key 요청 수신 시 요청 key size에 맞게 Master Key로부터 Session Key를 파생하여 전달하는 방안을 적용 예정이며, 따라서 GetKeyWithId에 Key size를 추가 한다.

[R-10] GetKey, GetKeyWithId

The SAE making an initial "Get key" request is referred to as the Master SAE for the key(s) returned. An SAE making a subsequent "Get key with key IDs" request is called the Slave SAE for the key(s) returned.

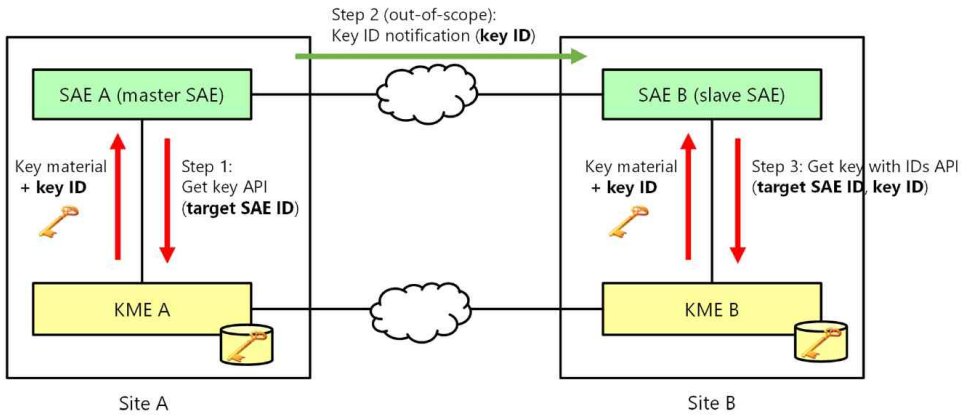

그림 2.2 ETSI 키 공급 인터페이스

2.2.4. 장비제어(CID – 4)

복수의 QKD장비와 KMS를 효율적으로 운영하기 위해 [R-13]과 같은 계층 구성이 필요하다. 본 설계에서는 QKDN control layer의 기능에 QKDN management layer의 일부 일반 기능을 포함한 Q-EMS를 구성한다.
Q-EMS는 QKD와 KMS 제어관리 역할을 수행하며 이를 위한 제어 인터페이스는 표준화가 미흡한 관계로 본 설계에서는 별도로 정의한다.

[R-13] QKDN control layer, QKDN management layer
○ QKDN control layer (QKD 제어 계층) ▷ Routing control function: Key relay를 위한 routing ▷ Configuration control function: QKD 및 KMS 설정 ▷ Policy-based control function: KM 정책 설정 ▷ Access control function: KMS 구성 요소간 상호 인증 및 접근 제어 ▷ Session control function: 키 공급을 위한 session제어 ○ QKDN management layer (QKD망 관리 계층 – 일반 기능) ▷ fault management functions: 장애관리 ▷ configuration management functions:망구성, 자원, 설정상태 관리 ▷ performance management: 성능/상태 정보 수집/분석/보고, 키 공급 정책 관리

2.2.5. QKD망관리(CID – 5)

현재 QKD 노드 간 키 교환을 위한 거리에 제약이 있어 장거리 키 교환을 위해서는 Trusted node를 통한 key relay가 필요하며 key relay를 위해서는 QKD망 구성을 통한 key 전달이 필요하다. QKD망 구성을 통해 key를 전달하기 위해서는 출발지 KMS부터 목적지 KMS까지의 라우팅 경로가 필요하며 이를 위해 Q-EMS에서는 KMS로부터 QKD의 연결정보를 수집하고, 이 정보를 통해 Key relay를 위한 라우팅 경로를 생성한다[C-13].

Key relay routing 생성
① Q-EMS에서 Domain간 KEY 생성이 필요한 경우 Domain간 QKD 연결관계를 그래프 자료구조에 입력, ②경로알고리즘(다익스트라등)을 통해 모든 경로를 생성하여 비용 최소순으로 정렬, ③ 가장 최소 비용인 경로 사용(1-hop이면 direct mode, 2-hop이상 이면 relay mode 동작), ④ 2단계에서 후순위 경로는 장애 발생 시 사용할 예비 경로로 저장

2.2.6. 키 릴레이(CID – 6)

현재 QKD 노드 간 키 교환을 위한 거리에 제약이 있어 장거리 키 교환을 위해서는 Trusted node를 통한 키 릴레이가 필요하며 이를 위해 QKD망 구성을 통한 key 전달이 필요하다.[R-4, 7]

키 릴레이시 KMS 관점에서 고려사항은 다음과 같다.

- ○ A와 D간 사용할 Key 생성
  - ▷ QKD 자원 부족으로 별도의 난수 생성기 이용 필요
- ○ A와 D간 KEY pair 정보 관리
- ○ A와 D간 Key 전송 인터페이스
  - ▷ A → B → C → D Key 전송
- ○ A에서 D까지 Key Relay 라우팅 설정
  - ▷ Q-EMS에서 라우팅 설정 후 KMS에 전달 필요[C-5]

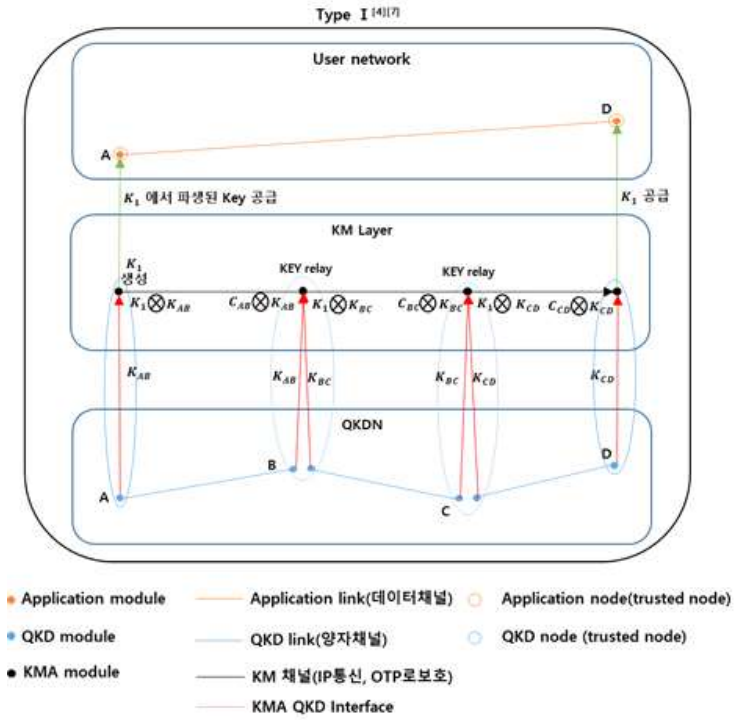

그림 2.3 키 릴레이 구조

[R-4, 7] Key Relay

[R-4] Trusted relaying:

[R-7] the KMAs support key relay through a key relay route between the two endpoint KMAs, employing highly secure encryption (e.g., OTP [b-Shannon 1949]). The key relay route is controlled by the QKDN controller as illustrated in Figure 1. The sessions of communications between KMAs to establish the end-to-end key can be controlled by the QKDN controller. A case of key relay using OTP, which is an IT-secure protocol for ensuring confidentiality of the keys is explained in the following paragraphs. The key data and metadata of the KMA-key are exclusively ORed with the other key shared by the neighboring 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 relay time stamp is added to the original KMA-key metadata, and stored in the key storage at the destination KMA.

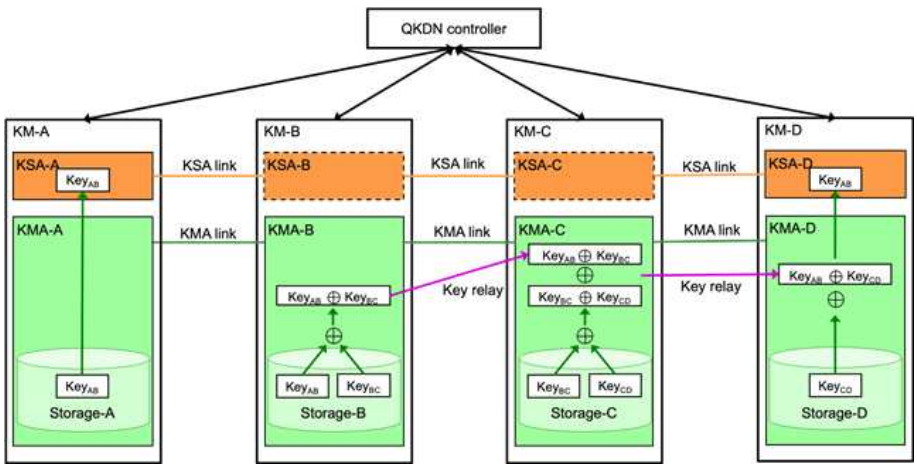

그림 2.4 QKDN Controller에 의해 제어되는 키 릴레이 구조

2.2.7. 연동 인터페이스 보호(CID – 7)

구성 요소간 연동 시 인터페이스는 보호되어야 하며 HTTPS를 이용하여 보호한다.[R-10]

[R-10, 35]
[R-10] The present document makes the following security assumptions about the use of this API with a QKD network: • each Trusted Node is securely operated and managed; • **this API is used between SAEs and KMEs within a secure site;** • each SAE is secure; • each KME is secure. All communications between SAE and KME shall use the **HTTPS protocols (with TLS version 1.2 or higher)** (IETF RFC 7230 [1], IETF RFC 7231 [2], IETF RFC 7235 [3], IETF RFC 5246 [4], IETF RFC 8446 [5]). HTTPS: Hypertext Transfer Protocol Secure **[R-35]** Security considerations in communications between QKD systems and applications (cryptographic applications) communication entity A. There are various protocols to provide keys to cryptographic applications, for example, [b ETSI GS QKD 004] and [b-ETSI GS QKD 014]. B. To secure such protocols or APIs, the reliance on conventional secure communication methods such as **TLS/SSL, HTTPS, SNMP** and IPSec, among others, are vital. C. Vendors generally use their own special APIs or various protocols for providing keys. Therefore, QKD key management layer sometimes need to deploy various interfaces and security functions at the same time.

2.2.8. 키 보호(CID – 8)

Q-KMS 키는 보안경계(Domain, Trusted Node) 외부로 유출이 되서는 안 되며 키 릴레이 시에는 OTP와 같은 강력한 암호화 기술로 보호되어야 한다. 보안경계 내에서 키 공급시도 HTTPS와 같은 기술을 이용하여 데이터를 보호한다.[C-7]

[R-29]
Protection of confidentiality A QKDN is required to provide capabilities to ensure **the confidentiality of stored and transferred key data**. **The key data confidentiality is critical**. The key data is usually required to be **protected with an IT-secure manner**. The key data are transferred between the trusted nodes (QKD nodes) via QKD links and KM links. In the QKD links, that contain quantum and classical channels, QKD protocol is executed to generate the secure keys. The QKD protocol should be implemented to guarantee the IT-secure confidentiality. In the KM links, the keys are relayed via classical channels. **The encryption of keys in the key relay** is required to meet the IT-secure confidentiality, which is, for example, attained by the **OTP encryption**.

2.2.9. 시스템 보호(CID – 9)

Q-KMS을 구성하는 모든 시스템은 보안 경계 안에 있어야 하며, 운영 및 관리를 위한 접근 시 인가된 사용자만 접근이 가능해야한다. 주요 상용 제품에서는 운영관리 목적으로 시스템 접근 시 사용자 ID/PW 기반 인증을 통해 인가된 사용자만 SSH 또는 HTTPS기반 WEB-GUI를 통해 접근을 허용한다.

[R-29, 35]
[R-29] **Controlled access and authorization** A QKDN is required to provide capabilities to ensure that the entities are prevented from gaining access to information or resources that they are not authorized to access. The security service supporting this requirement is access control. The access control service provides means to ensure that resources are accessed by subjects only in an authorized manner. Resources concerned may be the physical system, the system software, applications and data. The limitations of access are laid out in access control information, which specify: – The means to determine which entities are authorized to have access; –What kind of access is allowed (reading, writing, modifying, creating, deleting). In QKDN, all entities (QKD-Tx/Rx, KM, QKDN controller, QKDN manager) should be in a trusted node (QKD node). Thus, the access control of the trusted node should be carefully specified for its implementation. **[R-35]** security considerations in communications between QKD systems and management and monitoring systems A. Conventional management and monitoring methods are also available for QKDNs. B. All elements should be securely connected to management and monitoring systems, only if they are operated in dedicate physical units. C. Management and monitoring systems (or servers) always gather all the information about performance and status from every single unit and should also recognize any failure or any levels of events immediately.

2.2.10. 인증(CID – 10)

KMS와 키 사용자(NE) 연동 시 상호 인증이 필요하며 TLS기반 인증서를 이용한 인증을 수행하고자한다. 현재 KMIP와 주요 상용제품(QuintessenceLabs, Toshiba QKD system)에서 인증서 기반 인증을 사용하고 있다. 이 경우 상호간 키 정보를 사전에 공유하며 Q-EMS와 T-EMS를 통해 각 관리 대상 KMS와 NE에 인증정보를 전달하는 과정을 추가하고자 한다. 현재 [R-11]에서는 KMIP의 인증프로토콜의 보안성을 강화하는 방안을 모색 중이며 향후 업데이트된 결과를 반영하고자 한다.

[R-11]
To enhance security of KMIP **authentication protocols**. KMIP does not specify, as part of the protocol, the mechanisms by which **key management systems and cryptographic clients identify themselves to each other**. Rather, KMIP relies on existing standards for mutual authentication that specify how this identification is to be established. KMIP currently defines two authentication profiles, the first based on TLS, the second on HTTPS. In both profiles, **digital certificates are used by the client and the server to identify themselves** as participants in KMIP requests and responses. Registration mechanisms by which the enterprise key manager learns the identity of cryptographic clients are not defined in KMIP. The credentials used by the cryptographic client to identify itself can be included in the protocol as part of a request message, to simplify processing of the request by the key management system. However, the credential element is not guaranteed to be authenticated and is therefore not intended for use in authentication. Message integrity for KMIP exchanges, as well as entity authentication, is provided by TLS. Other mechanisms (for example with the inclusion of QKD inside TLS or HTTPS key exchange process) that could also be used for enhanced security of KMIP messages are not currently defined for KMIP.

2.2.11. 정보관리(CID – 11)

Q-KMS시스템 운영을 위해 필요한 정보는 크게 운영정보, 정책정보, 키 정보 3가지로 나누어 볼 수 있으며, 이러한 정보를 안정적으로 관리하기 위해 RDBMS를 사용한다. QKD-KMS운영관련 정보는 표준화가 미흡하여 기지국제어관리 프로토콜(TR-069)에서 사용하는 TR-196 데이터모델 표준을 참고한다.

[R-7, 13]

[R-13] 운영정보와 정책에 대한 언급만 있으며 구체적인 정보는 없음

○ 운영정보: 일반적인 장비관리(구성, 성능, 장애)를 위한 정보 구성

○ 정책정보: 키 사용 정책을 위한 정보 구성

[R-7] KMA Key file 참고

○ 키 정보: Key 데이터 저장/조회를 위한 정보 구성

(2) KMA-key		
KMA-key ID	ID of the KMA-key, which is the same for the pair of keys for Alice and Bob, and unique in a QKD network. A part of the bits of the hash value generated from the names of the pair of QKD modules is often used for this ID.	M
Key length	Key length of the KMA-key	O
Key type	Index to specify whether encrypting key or decrypting key	O
KMA ID	ID of the KMA that stores the KMA-key	O
Matching KMA ID	ID of the matching KMA	M
Generation time stamp	Time stamp of the KMA-key generation at the KMA	O
QKD module ID	ID to identify the QKD module which generates the KMA-key data	O
Matching QKD module ID	ID to identify the matching QKD module which constitutes the pair of Alice and Bob	O
Hash value	Hash value of the KMA-key data. (There are several options for hash function, which are discussed in other Recommendations.)	O

2.2.12. 시스템관리 환경(CID – 12)

시스템 운영관리를 위해서는 운영자가 시스템에 접근하여 장비 상태 모니터링, 장비 설정 관리 등의 작업이 필요하며 이를 위한 관리자 환경이 필요하다. 이러한 관리자 환경 제공시도 관리자 인증, 관리/제어정보 보호 등의 기능[C-9]이 필요하다. 본 설계에서는 관리환경에 접근하기 인터페이스로써 RestAPI를 통한 제어 인터페이스를 제공하고 Q-EMS, NMS, T-EMS, WEB-GUI, CLI등에서 상기 인터페이스를 이용함으로써 일원화된 통합 관리 환경을 제공하고자 한다.

2.2.13. KMA 키 동기화 (CID – 13)

KMS는 QKD에서 Key Stream을 수신한 후 KMS 키 정책에 맞게 reformat을 수행하고, 각각의 분할된 Key data에 Global하게 고유한 Key-ID를 할당한다. 이렇게 reformat된 키는 KMS간 Key-ID별로 key가 맞는지 키 동기화를 수행 한다[R-7]. 유효성 검사를 위해 각각의 KMS는 키 동기화를 위한 Key-ID와 키에 대한 해쉬값으로 구성된 인터페이스를 제공하고, Master KMS에서 Slave KMS간 키 동기화 요청/응답 수행 후 해쉬값이 일치하는 키에 대해서 키 저장소에 저장한다.

[R-7]
[R-7] 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 the Req_KM 8 in [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 KMA (say KMA1) send the matching KMA (say KMA2) a key-authentication request including a hash value or a message authentication code of the buffered keys. 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 is 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 the item (2) of table 1 in clause 10.

KMA Key file 참고

○ 키 정보: Key 데이터 저장/조회를 위한 정보 구성

(2) KMA-key		
KMA-key ID	ID of the KMA-key, which is the same for the pair of keys for Alice and Bob, and unique in a QKD network. A part of the bits of the hash value generated from the names of the pair of QKD modules is often used for this ID.	M
Key length	Key length of the KMA-key	O
Key type	Index to specify whether encrypting key or decrypting key	O
KMA ID	ID of the KMA that stores the KMA-key	O
Matching KMA ID	ID of the matching KMA	M
Generation time stamp	Time stamp of the KMA-key generation at the KMA	O
QKD module ID	ID to identify the QKD module which generates the KMA-key data	O
Matching QKD module ID	ID to identify the matching QKD module which constitutes the pair of Alice and Bob	O
Hash value	Hash value of the KMA-key data. (There are several options for hash function, which are discussed in other Recommendations.)	O

2.2.14. 고유 식별자(CID – 14)

Q-KMS에서는 정확한 시스템 운영 및 서비스 제공을 위해 장비ID와 Key-ID에 대해 고유한 식별자를 사용해야 하며 본 설계에서는 고유식별자 형식으로 UUID(16bytes)를 사용한다. UUID 생성은 [R-36] 표준을 반영하며 추후 최신 해쉬알고리즘 사용을 목적으로 변경이 될 수 있다.([R-36]에서는 MD5, 또는 SHA-1등의 오래된 알고리즘을 사용하게 되어 있음)

○ 장비ID

▷ 각 이종 장비(NE, QKD)의 고유한 장비ID 생성 시 장비별 ID 체계가 다르며 이를 고려한 UUID 생성 방식이 고려되어야 한다.

○ Key-ID

▷ 고유한 값을 생성하기 위해 QKD-KEY-ID, QKD-ID, Matching QKD-ID등의 값을 사용한다.

[R-7, 10]

[R-10]

Table 11

Items		Data type	Description
Keys		array of objects	Array of keys. The number of keys is specified by the "number" parameter in "Get key". If not specified, the default number of keys is 1.
	key_ID	string	ID of the key: UUID format (example: "550e8400-e29b-41d4-a716-446655440000").
	key_ID_extension	object	(Option) for future use
	key	string	Key data encoded by base64 [7]. The key size is specified by the "size" parameter in "Get key". If not specified, the "key_size" value in Status data model is used as the default size.
	key_extension	object	(Option) for future use.
key_container_extension		object	(Option) for future use.

[R-7] QKD-Key ID, QKD module ID, Matching QKD module ID

Metadata	Description	M/O
(1) QKD-key		
QKD-key ID	ID of the QKD-key	M
Key length	Key length of the QKD-key	O
QKD module ID	ID of the QKD module (Alice or Bob) that generates the QKD-key	O
Matching QKD module ID	ID to identify the matching QKD module which constitutes the pair of Alice and Bob	O
Generation time stamp	Time stamp of QKD-key generation at the pair of QKD modules	O
Hash value	Hash value of the QKD-key data. (There are several options for hash function, which are discussed in other Recommendations.)	O

2.2.15. 파생키 생성(CID – 15)

본 설계에서는 KMS에서 QKD자원을 효율적으로 사용하기 위해 Master Key size 정책에 따라 QKD Key stream을 Master Key size로 분할하여 저장하고, 키 사용자(NE)로부터 Key 공급 요청 시 전달된 Key size로 key를 derivation하여 Session Key를 생성 한 후 공급한다. 본 설계에서는 Key Derivation시 [R-37]의 표준을 사용한다.

제3장 표준화 문서에 근거한 양자키관리 설계

제3장 표준화 문서에 근거한 양자키관리 설계

3.1 QKD-KMS-전송교환장비 인터페이스 및 연계 구조 설계

QKDN KMS에서 키생성 및 전달을 위해서는 키 교환 및 관리를 위한 양자계층과 암호화 키를 이용한 데이터채널을 구성하는 전달계층으로 나누어 볼 수 있다.

- 양자계층: 키 교환/관리/공급을 수행하는 계층으로 내부적으로 키 관리/공급을 위한 LKMS(KSA, KMA)와 키 교환을 위한 QKD로 나누어진다.
- 전달계층: LKMS-KSA로부터 암.복호화 키를 공급받아 데이터채널을 구성한다.

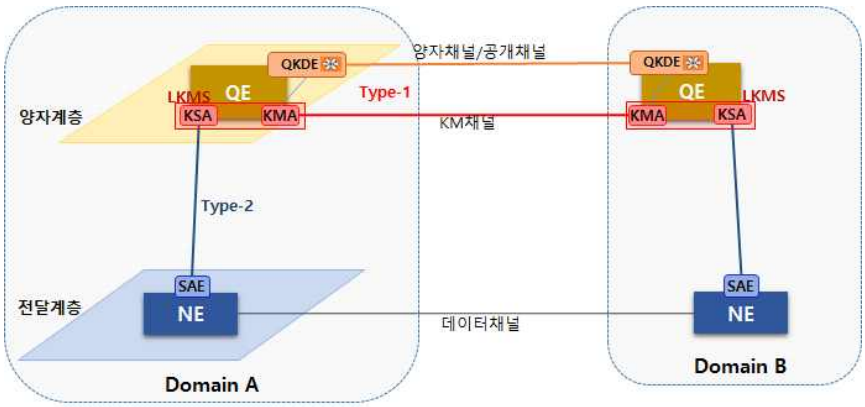

그림 3.1 양자계층 및 전달계층 구성요소 및 인터페이스

KMS에서 키 생성/관리/공급을 위해서 필요한 인터페이스는 표 3.1과 같다.

표 3.1 인터페이스 정의

인터페이스	정의
Type-1	KMA-KMA: Key 동기화, Key Relay를 위한 인터페이스
Type-2	KSA-SAE(NE, 전송장비): 암.복호화 키 공급을 위한 인터페이스
QKDE-KMA	QKDE-KMA: QKDE상태 모니터링 및 제어, KEY STREAM 전달을 위한 인터페이스

3.2 QKD Node 단위 및 시험망 통합 운영관리 구조 설계

3.2.1 KMS 계층 구성

본 설계에서는 QKD Node 및 KM영역을 효과적으로 관리하기 위해 관리에 필요한 기능을 계층적으로 표 3.2과 같이 정의 하였다.

표 3.2 QKDN 계층 정의

계층	정의
망관리계층	QKD 망 구성 관리
장비관리계층	관리대상장비(QE, NE)에 대한 구성, 성능, 장애 관리 및 KM관리
QKD계층	QE간 Key stream 생성 및 Key분배, Key 관리, Key 사용 정책 적용 NE에게 데이터채널 암.복호화에 사용할 Key 공급
전달계층	Domain간 암호화된 데이터를 전송하는 데이터 채널 생성/관리

3.2.2 KMS 계층별 구성 요소

KMS의 계층별 구성요소는 표 4.3와 같으며 본 설계의 범위는 Q-EMS, KMA, KSA로 한정 된다.

표 3.3 QKDN 계층별 구성 요소

계층	구성요소	역할
망관리 계층	NMS	◆ QKD/전송망 구성 관리
장비 관리 계층	T-EMS	◆ 데이터 채널 설정 및 제어
	Q-EMS	[QKDE관리] ◆ QKD 모듈, QKD 채널 설정 및 제어 [KEY정책 관리] ◆ Key-Consumer에 대한 QoS 및 Key관리정책설정 ◆ 복수의 Key-Consumer에 대해 QoS정책 기반 session관리 ◆ Key relay를 위한 routing control ◆ KM 채널 설정 및 제어
QKD 계층	QKDE	QE간 Key stream 생성 및 동기화
	KMA(LKMS)	◆ QKDE가 생성한 Key stream 수신 및 관리 (KMA:QKDE = 1:N) ◆ Key 라이프사이클을 관리, Key Relay, Key 저장
	KSA(LKMS)	◆ SAE에게 Key공급 (1:N)
전달 계층	SAE	◆ KSA에 암.복호화에 사용할 KEY 요청
	CM	◆ 데이터 암.복호화 수행

그림 3.2는 계층별 구성 요소간 관계와 통신채널을 도식화한 것이다. Q-EMS에서 QKD계층 및 KM관리를 수행하며 QE는 내부적으로 QKDE(QKD 영역)와 LKMS(KM영역)으로 분리되어 있다[C-1].

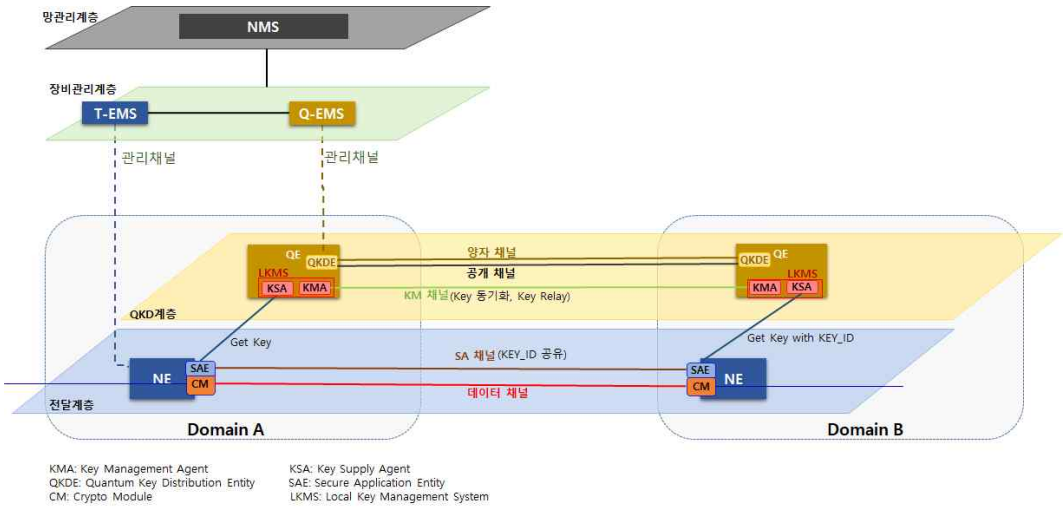

그림 3.2 KMS 계층 구조

3.3 이종 양자키 관리 구조 설계

이종 양자키 프로토콜 수용을 위서는 키 관리 개체와 QKD개체가 분리되어 야한다[C-1]. 본 설계에서는 QE시스템 안에 QKDE(QKD모듈)와 LKMS(KM 모듈)로 논리적으로 분리하는 구조를 선택 하여 KM모듈인 LKMS에서 다양한 QKD모듈과 연동할 수 있는 방안 마련하였다. LKMS의 블록 설계는 메인 프레임의 변경 없이 다양한 Version과 새로운 프로토콜 추가가 용이한 형태로 방향을 잡았으며 블록설계에 대해서는 5장에서 상세히 다루고자 한다.

3.4 Trusted Node 및 Key Relay 연동 구조 설계

본 설계에서는 키 릴레이를 위한 구조로 [R-4]표준 Type-I구조[C-6]를 참고하였다. Type-I구조는 Trusted Node가 외부 공격으로부터 보호를 받는다는 전제아래[C-8]에 다음과 같은 절차로 키의 릴레이가 이루어진다.

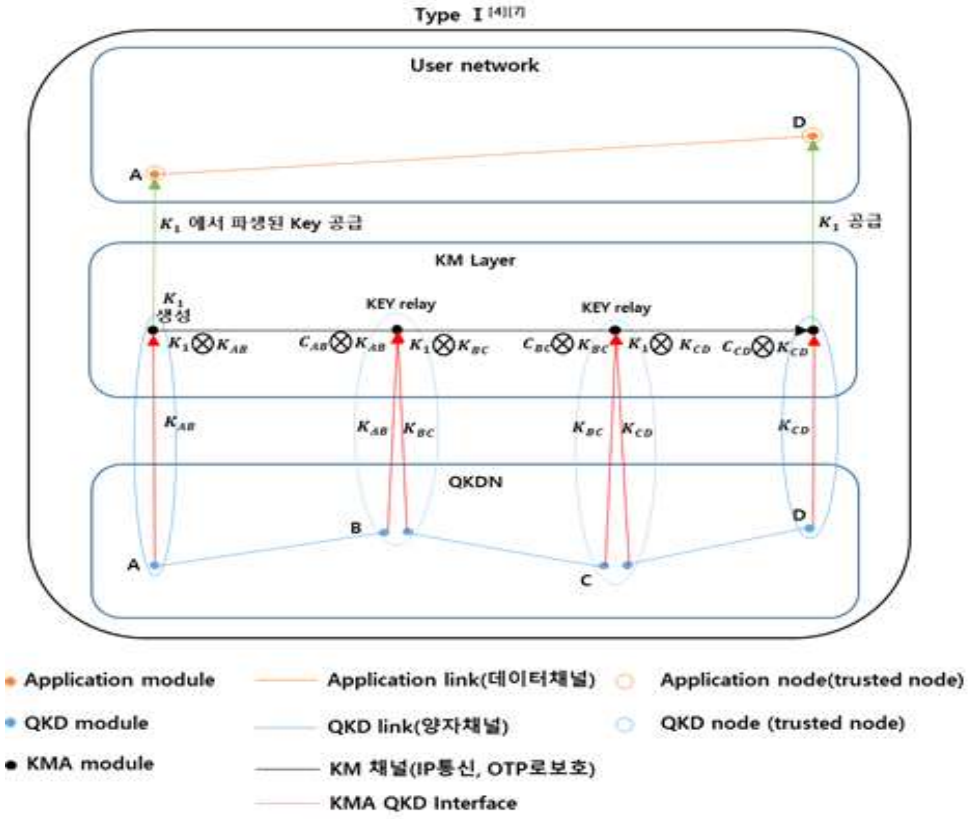

그림 3.3 키 릴레이 구조

① T-EMS에서 KEY가 필요한 Domain ID pair정보를 Q-EMS에 전달
② Q-EMS에서 해당되는 Master KMA(A), Relay KMA(B & C), Slave KMA(D)에 전달 경로 정보 전달
③ Q-EMS에서 A에 D와 Key stream 생성 요청
④ A에서 난수생성기(QKD와 별도)를 이용하여 Key stream을 생성

⑤ A에서 B로 Key Relay 요청(Key pair 정보 및 Key data를 A와 B간 Key stream으로 XOR 연산 암호화 전송)
⑥ B에서 C로 Key Relay 요청(Key pair 정보 및 Key data를 B와 C간 Key stream으로 XOR 연산 암호화 전송)
⑦ C에서 D로 전달(Key pair 정보 및 Key data를 C와 D간 Key stream으로XOR 연산 암호화 전송) - KEY Relay 완료

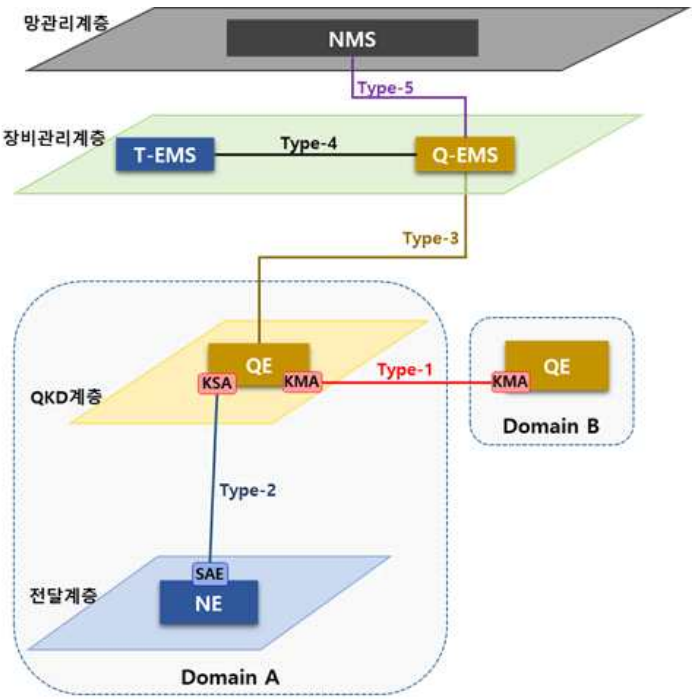

그림 3.4 KMS 구조 연동인터페이스

3.5 관리 인터페이스

본장 3.1절에서는 구성요소 중 일부인 NE(전송장비)와 LKMS-KSA, KMA와 QKDE, KMA와 이웃 KMA간 인터페이스에 대해서 정의 하였다. 본 3.5절에서는 이에 더하여, QKD와 KM관리를 목적으로 관리시스템과 장비의 연동을 위한 인터페이스를 추가적으로 정의하고자 한다. 그림 3.4에서는 구성요소간 관계 및 인터페이스를 도식화 했으며, 각 인터페이스에 대한 정의는 표 3.4을 따른다.

표 3.4 인터페이스 정의(관리인터페이스 추가)

인터페이스	정의
Type-1	KMA-KMA: Key 동기화, Key Relay를 위한 인터페이스
Type-2	KSA-SAE(NE, 전송장비): 암.복호화 키 공급을 위한 인터페이스
Type-3	Q-EMS - QE: Domain간 키 생성 요청, NE 인증 정보 전달, 키 릴레이 경로 전달, 키 생성상태 보고
Type-4	T-EMS - Q-EMS: Domain간 키 생성 요청, NE 인증 정보 전달, 공새채널 구성 요청
Type-5	NMS - Q-EMS: 망 구성정보 전달
QKDE-KMA	QKDE-KMA: QKDE상태 모니터링 및 제어, KEY STREAM 전달을 위한 인터페이스

본 장에서는 키 생성/관리/공급에 대한 연결 구조와 연동 인터페이스 그리고 이종 QKD를 수용하기 위한 구조와 키 릴레이 방안에 대해서 다루었다. 이종 QKD를 수용하고, 키 릴레이가 가능하게 하도록 하기 위해 기능별 계층을 나누었으며 QKD개체와 KMS를 논리적으로 분리하는 방향으로 구조를 설계 하였다. QKD 망 구성 관리 및 장비 관리를 위해서는 추가적으로 관리 인터페이스에 대한 정의가 필요하며 4.5.1절에서 추가되는 인터페이스에 대해서 다루고자 한다.

제4장 양자 KMS 설계 항목

제4장 양자 KMS 설계 항목

4.1 전송망과 연계한 KMS 상세 구조 설계

4.1.1 계층 및 구성요소

본 설계에서 KMS 구조는 관련 표준들을 조사 결과에 따라 설계 하였으며 구성요소는 표 4.1과같다. 그림 4.1은 KMS 구조와 영역별 관련표준 정보를 맵핑한 것이다.

표 4.1 QKDN 계층별 구성 요소

계층	구성요소	역할
망관리 계층	NMS	◆ QKD/전송망 구성 관리
장비 관리 계층	T-EMS	◆ 데이터 채널 설정 및 제어
	Q-EMS	[QKDE관리] ◆ QKD 모듈, QKD 채널 설정 및 제어 [KEY정책 관리] ◆ Key-Consumer에 대한 QoS 및 Key관리정책설정 ◆ 복수의 Key-Consumer에 대해 QoS정책 기반 session관리 ◆ Key relay를 위한 routing control ◆ KM 채널 설정 및 제어
QKD 계층	QKDE	QE간 Key stream 생성 및 동기화
	KMA(LKMS)	◆ QKDE가 생성한 Key stream 수신 및 관리 (KMA:QKDE = 1:N) ◆ Key 라이프사이클을 관리, Key Relay, Key 저장
	KSA(LKMS)	◆ SAE에게 Key공급 (1:N)
전달 계층	SAE	◆ KSA에 암.복호화에 사용할 KEY 요청
	CM	◆ 데이터 암.복호화 수행

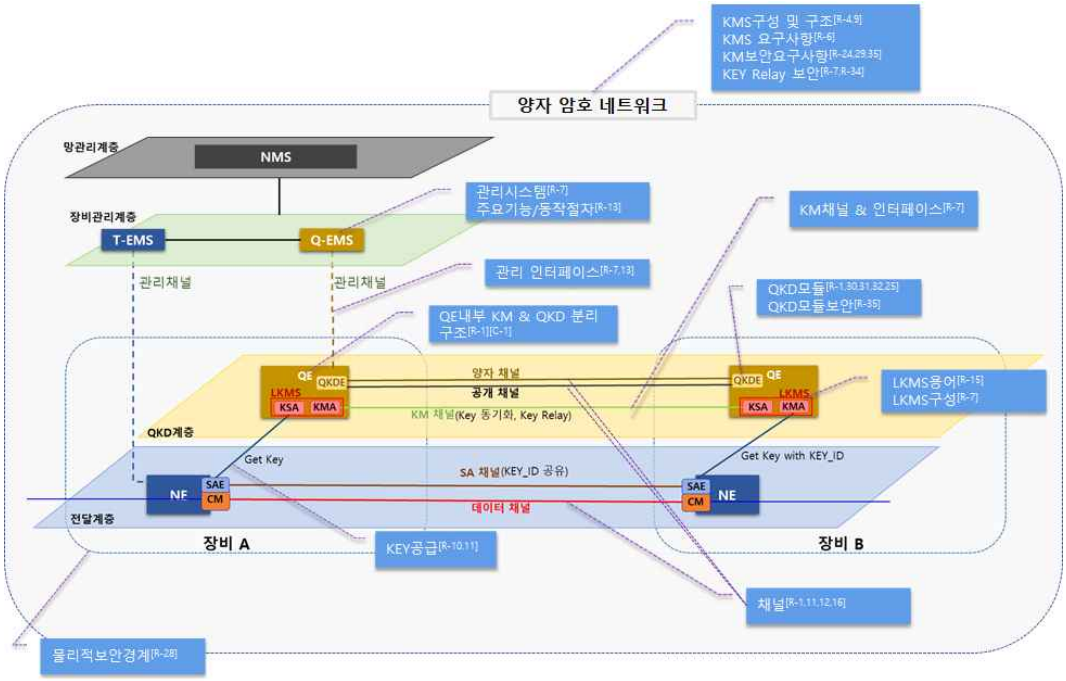

그림 4.1 KMS 구조 및 관련 표준 매핑

4.1.2 채널 구성

KMS을 구성하고 있는 구성 요소간 연동을 위해서는 상호간 통신이 필요하며 이러한 통신을 위한 채널의 구성은 표 4.2와 같다.

4.2 채널 구성

채널	정의
양자채널	양자암호 키 분배를 수행하는데 사용되는 QKD모듈 사이의 양자 광 채널 TTA: 양자채널[R-1]
공개채널	QKD 노드 간 공유된 양자 신호에서 양자키를 생성하기 위한 양자암호프로토콜, QKD 네트워킹 제어, 키 관리 및 전달 프로토콜을 위한 채널 TTA: 일반채널[R-1], ETSI: Classical Channel[R-11,12], ETSI: Public Channel[R-11], ITU-T: Clasical Channel[R-13]
데이터채널	기존 네트워크에서 데이터 전달을 위한 채널을 의미하며, 양자키로 암호화된 데이터 전달 TTA: 양자채널[R-1]
KM채널	KMA간 키 동기화, 키 릴레이를 위한 통신 채널 ITU-T: KM Link[R-7]
SAE채널	SAE간 Key-ID 정보 공유를 위한 통신 채널
관리채널	Q-EMS와 KMS간, T-EMS와 NE간 관리를 위한 통신 채널 [R-13] The QKDN controller communicates control information with the KMs, the QKD modules, and the QKD link via the reference points Ck, Cq, and Cops, respectively. In the distributed architecture, the QKD controllers communicates with each other via the reference point Cx.

4.1.3 연동 인터페이스 Method 구성

가. Type-1 인터페이스 Method 구성

키 릴레이[c-6] 와 KMA 키 동기화[c-13]등 KMA간 기능 수행을 위한 인터페이스로 Domain간 중요정보의 이동이 있어 OTP[C-8]를 이용한 높은 보안강도의 데이터 보호 알고리즘을 적용하여 데이터를 전송한다.

표 4.3 Type 1 method

MID	method	정의
M101	postRelayKey (Required)	KMA에서 이웃 KMA로 키 릴레이 요청
M102	postValidateKey (Required)	이웃 KMA간 QKD로부터 수신한 Key stream을 reformat한 키에 대한 동기화 수행

나. Type-2 인터페이스 Method 구성

SAE(전송장비)에서 KSA로 키를 공급받기 위한 인터페이스[C-3]로 키를 공급하는 절차는 다음과 같다. ① Master SAE에서는 KSA에 키 요청 시 Slave SAE-ID를 이용하여 필요한 키를 공급 받은 후 ② Slave SAE에게 공급받은 키의 Key-ID를 전달한다. ③ Slave SAE에서는 KSA에게 Master SAE와 동일한 키를 공급받기 위해 Master로부터 수신한 Key-ID를 이용하여 KSA에 키를 요청한다. 표 5.3는 이러한 기능 수행을 위한 method 목록이다.

4.4 Type 2 method

MID	method	정의
M201	getKey (Required)	NE(SAE)에서 KSA로 Key 요청 (KEY: Slave SAE ID)
M202	getKeyWithId (Required)	NE(SAE)에서 KSA로 Key 요청 (KEY: Key-ID)
M203	getStatus (Required)	NE에서 KMS로 사용가능한 Key가 있는지 조회 요청 (KEY: Slave SAE ID)

다. Type-3 인터페이스 Method 구성

표 4.5 Type 3 method

MID	method	정의
M301	putDeviceInfo (Required)	Q-EMS -> QE: NE장비정보 전송
M302	postDeviceInfo (Required)	Q-EMS -> QE: NE장비정보 갱신
M303	delete DeviceInfo (Required)	Q-EMS -> QE: NE장비정보 삭제
M304	getDeviceInfo (Required)	Q-EMS -> QE: NE장비정보 조회
M305	postGenerateKey (Required)	Q-EMS -> QE: 키 생성 요청
M306	postRelayKey (Required)	Q-EMS -> QE: 키 릴레이 요청
M307	postKeyPolicy (Required)	Q-EMS -> QE: 키 관리 정책 전달
M308	postKeyStatus (Required)	QE -> Q-EMS: 키 생성상태 보고
M309	postStatistics (Required)	QE -> Q-EMS: 통계정보 전달
M310	postEvent (Required)	QE -> Q-EMS: Fault/Alarm/운영정보등의 Event 전달
M311	postStatisticsMgmt (Required)	Q-EMS -> QE: M309의 통계항목/생성주기/보고주기 설정
M312	postInitQkd (Required)	Q-EMS -> QE: QKD모듈 초기화
M313	postInform (Required)	QE -> Q-EMS: LKMS 및 QKDE모듈 구성 정보 전달

라. Type-4 인터페이스 Method 구성

표 4.6 Type 4 method

MID	method	정의
M401	putDeviceInfo (Required)	T-EMS -> Q-EMS: NE장비정보 전송
M402	postDeviceInfo (Required)	T-EMS -> Q-EMS: NE장비정보 갱신
M403	delete DeviceInfo (Required)	T-EMS -> Q-EMS: NE장비정보 삭제
M404	getDeviceInfo (Required)	T-EMS -> Q-EMS: NE장비정보 조회
M405	postGenerateKey (Required)	T-EMS -> Q-EMS: 키 생성 요청
M406	postKeyStatus (Required)	Q-EMS -> T-EMS: 키 생성 상태 보고
M407	postNotifyFault (Required)	T-EMS -> Q-EMS: NE장애정보 전달
M408	postNotifyFault (Required)	Q-EMS -> T-EMS: QE장애정보 전달
M409	postPublicChannel (Optional)	Q-EMS -> T-EMS: QKDE간, KMA간 통신을 위한 공개채널 구성 요청
M410	deletPublicChannel (Optional)	Q-EMS -> T-EMS: 공개채널 삭제 요청
M411	postCapabilites (Required)	T-EMS -> Q-EMS: 지원하는 method 목록 교환

마. Type-5 인터페이스 Method 구성

표 4.7 Type 5 method

MID	method	정의
M501	putQkdtopology (Required)	NMS -> Q-EMS: QKD망구성정보 전달
M502	postKeepAlive (Required)	NMS -> Q-EMS: Q-EMS 상태 확인

바. QKDE-KMA 인터페이스

KMA에서 QKDE간 교환된 Key stream을 수신하기 위한 인터페이스가 필요하다. 본 인터페이스에 대해서는 현재는 표준화가 진행 시작 단계로 QKD Key stream전달을 위한 메타정보에 대한 정의만 되어 있는 상황이다. 본 설계에서는 [C-2]의 내용대로 그림 x.x와 같이 QKD-Key file을 정의 하였으다. Key stream이 QKDE에서 KMA로 전달되는 과정은 다음과 같다.

① QKDE에서 QKD-Key file을 생성한다. ② QKDE에서 LKMS의 특정 디렉토리로(LKMS-KMA 설정) sftp등과 같은 보안 프로토콜을 이용하여 QKD-Key-file을 전송한다. ③LKMS-KMA에서 주기적으로 파일을 읽어서 Key stream을 추출 한다.

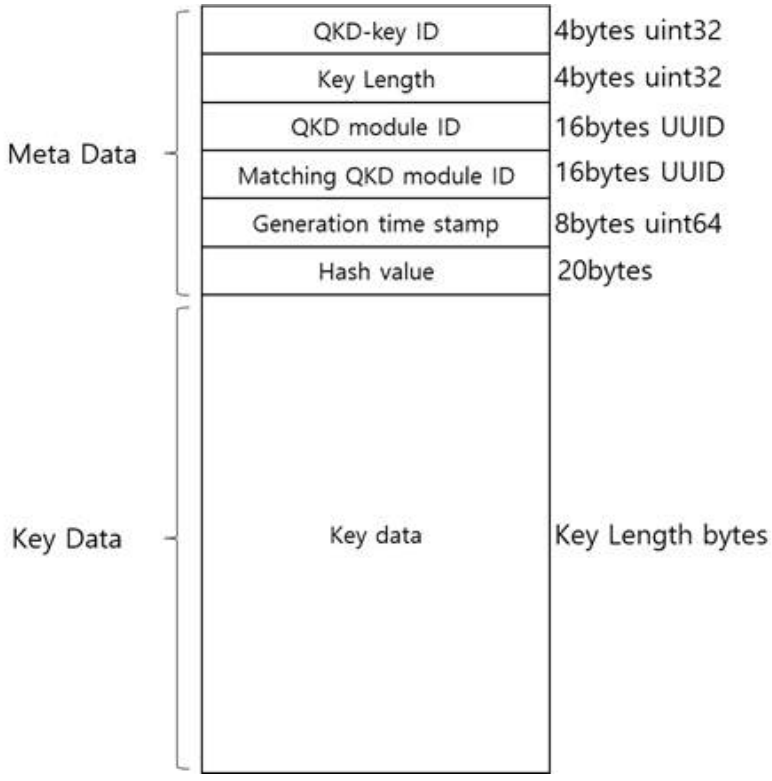

그림 4.2 QKD-Key-file 구조

4.1.4 KMS 기능 구성 및 동작 절차

가. KMS 주요 기능 구성

표 4.8 KMS 주요 기능 구성

FID	기능	설명
F101	장비 등록/변경/삭제/조회 (구성)	장비간 연동을 위한 연동대상 장비의 정보를 등록/변경/삭제/조회 하는 기능
F102	Bootstrap (구성)	KMS 서비스 준비를 위한 절차 QE에서 Q-EMS에 자신의 정보(LKMS 및 QKD 연결관계 등)를 등록하고 Q-EMS에서는 키관리 정책, 운영 정책 등을 설정)
F103	구성정보 설정/조회 (구성)	주요 설정 SET/GET 수행
F104	Capablity 교환(구성)	Q-EMS와 T-EMS간 지원되는 기능 교환
F201	Master Key 생성 (운영)	Domain간 Master Key 생성
F202	Session Key 공급 (운영)	Domain간 Data채널 구성에 필요한 키 공급
F203	Key Life Cycle 관리 (운영)	키 상태 관리
F204	Master Key 저장소 (운영)	Master Key 저장/폐기
F205	Key 사용 상태 보고 (운영)	NE에서 키 사용 상태 보고
F301	통계정보 보고 (성능)	QE의 통계정보 항목 및 수집주기 보고 주기 정책 설정 정책에 따른 보고 수행
F302	Event 생성 및 Action (장애)	QE의 Event(Alarm, Fault, Run)생성 및 그에 다른 Action 수행
F401	접근제어 (보안)	장비 연동 요청 시 인가된 장비인지 식별 및 권한 확인

나. KMS 주요 기능별 동작 절차

[F-101 - 장비 등록/변경/삭제/조회]

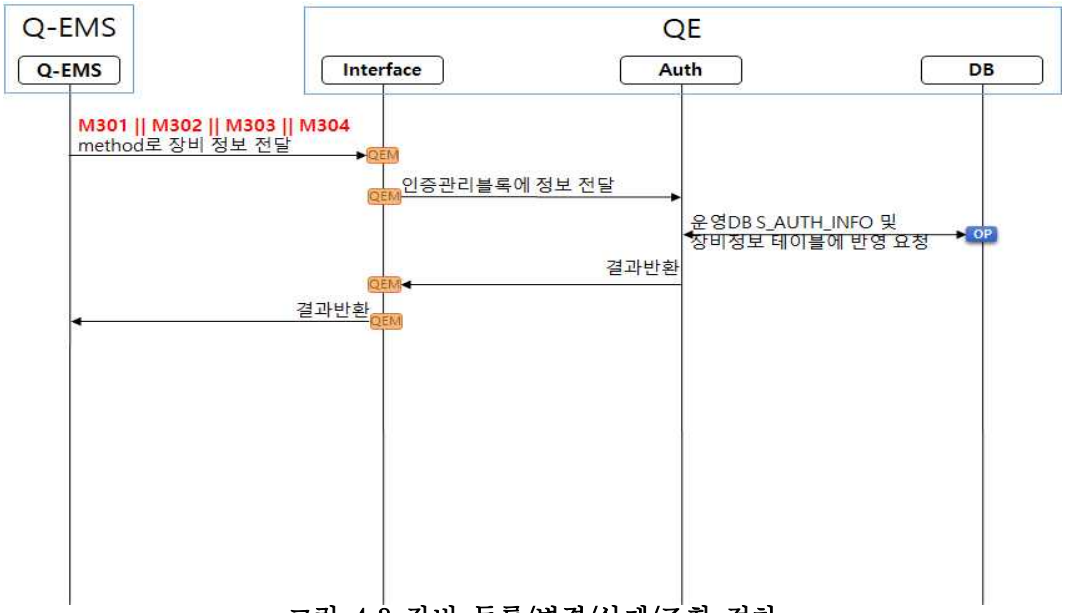

그림 4.3 장비 등록/변경/삭제/조회 절차

[F-102 - Bootstrap]

Bootstrap - 1/2

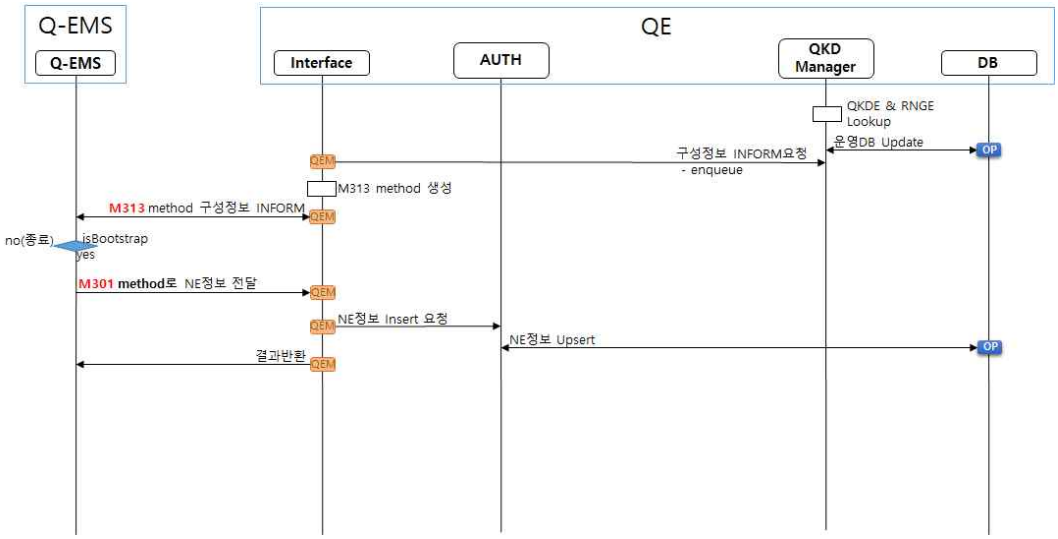

그림 4.4 Bootstrap 절차 1

Bootstrap – 2/2

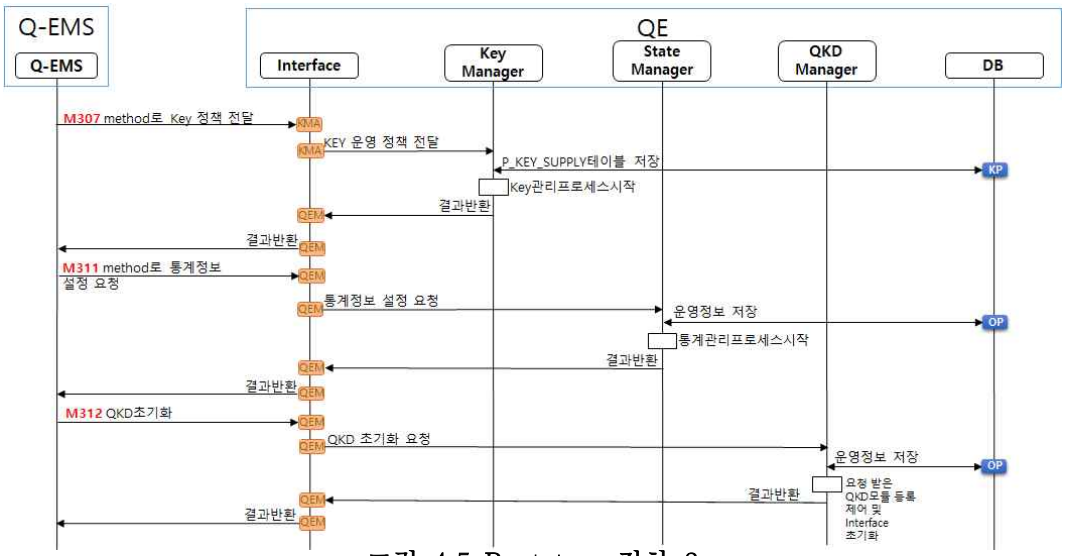

그림 4.5 Bootstrap 절차 2

[F-103 – 구성정보 설정/조회]

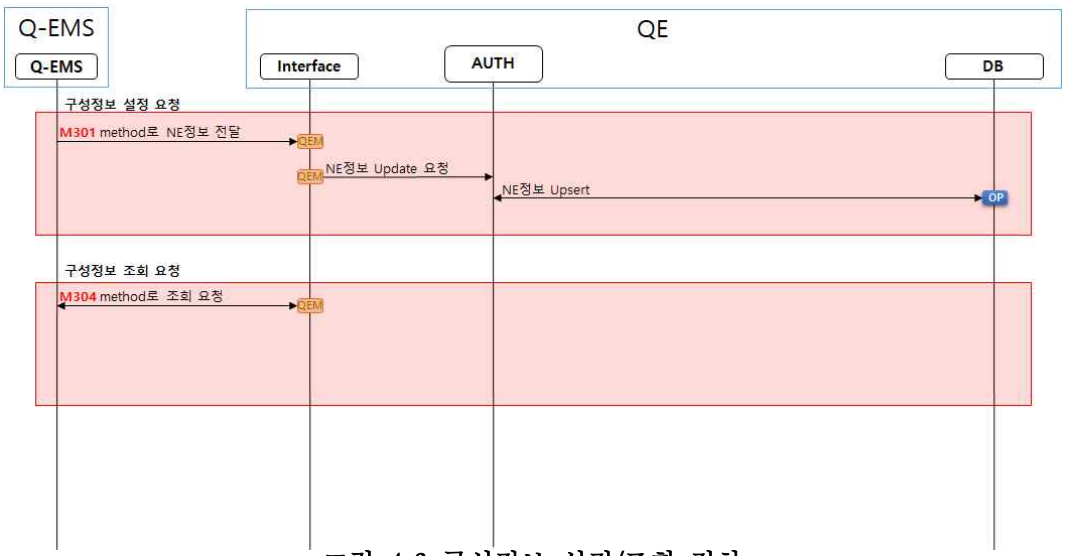

그림 4.6 구성정보 설정/조회 절차

[F-104 - Capability 교환]

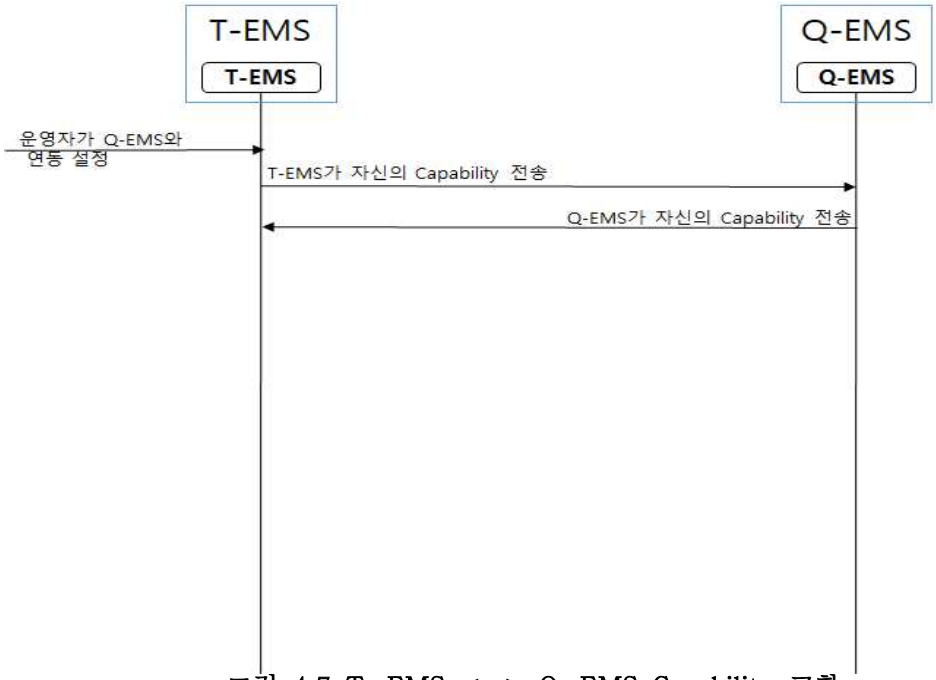

그림 4.7 T-EMS <-> Q-EMS Capability 교환

[F-201 - Master Key 생성]

Master Key 생성 - 1/2 - Direct mode

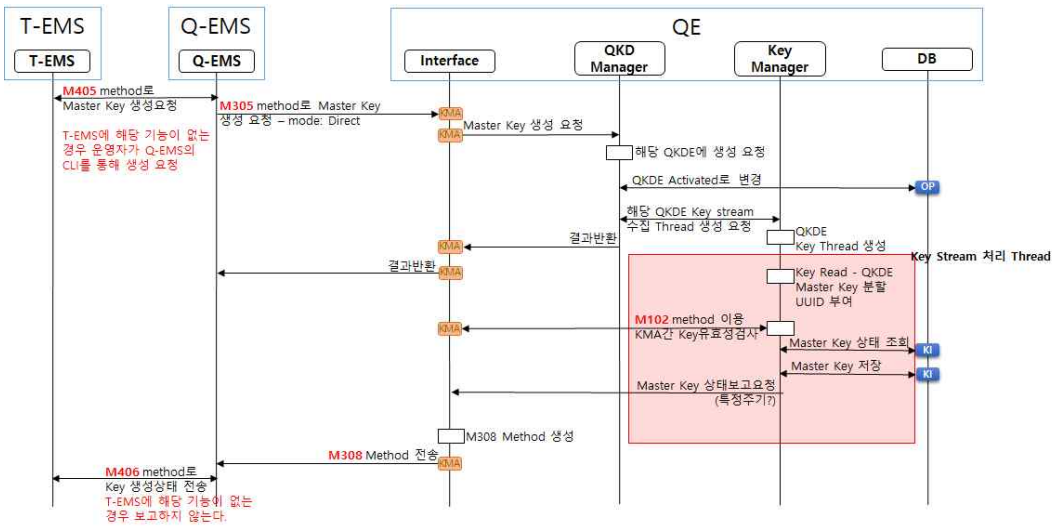

그림 4.8 Master Key 생성 절차 1(Direct)

Master Key 생성 – 2/2 – Relay mode

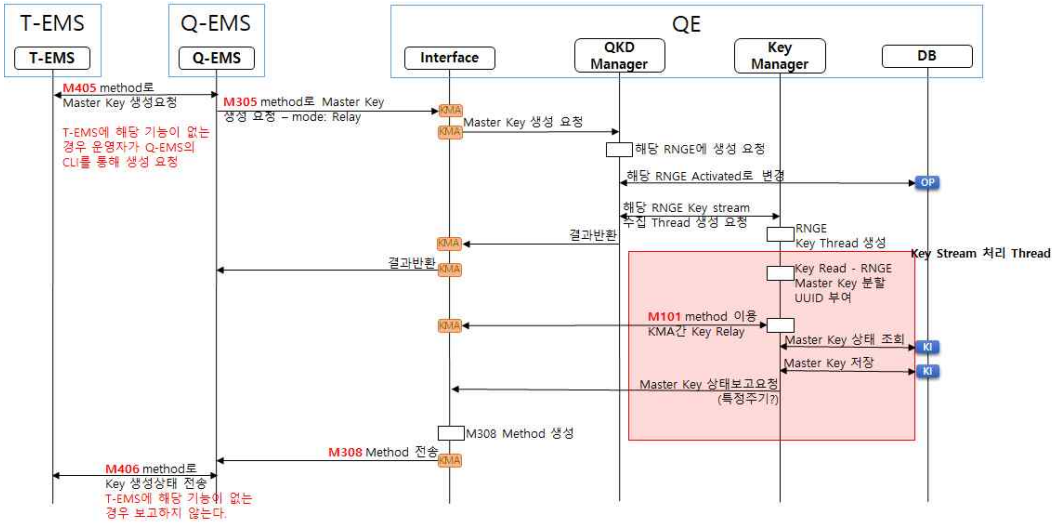

그림 4.9 Master Key 생성 절차 2(Relay)

[F-202 – Session Key 공급]

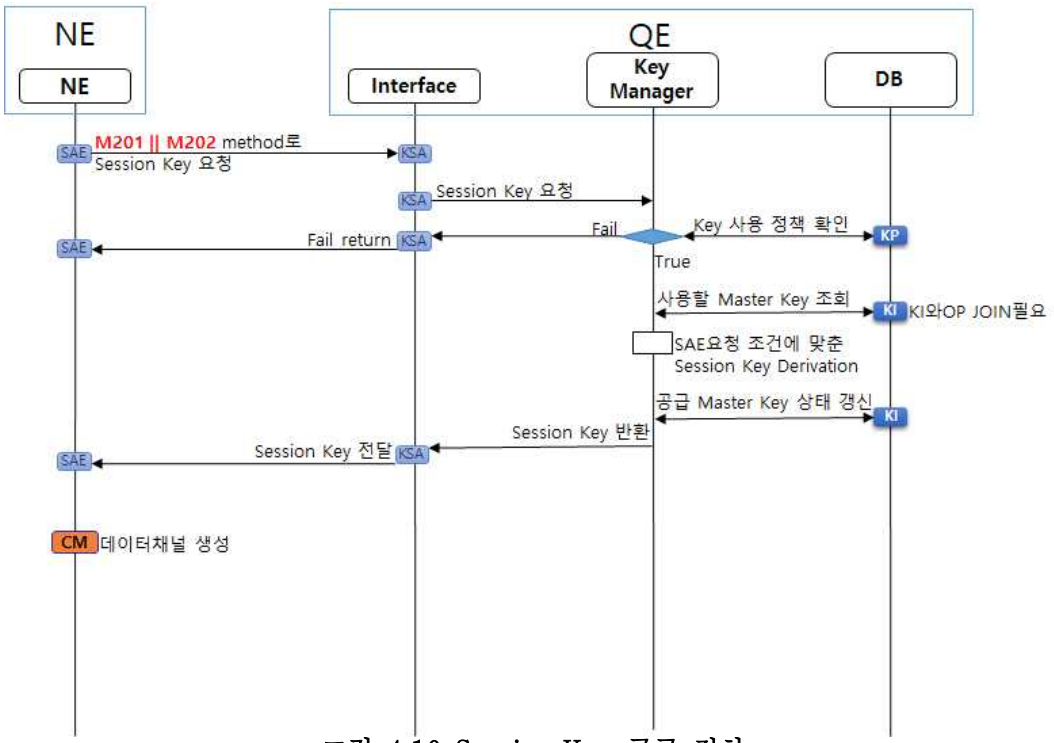

그림 4.10 Session Key 공급 절차

[F-203/204/205 - Key Life Cycle 관리]

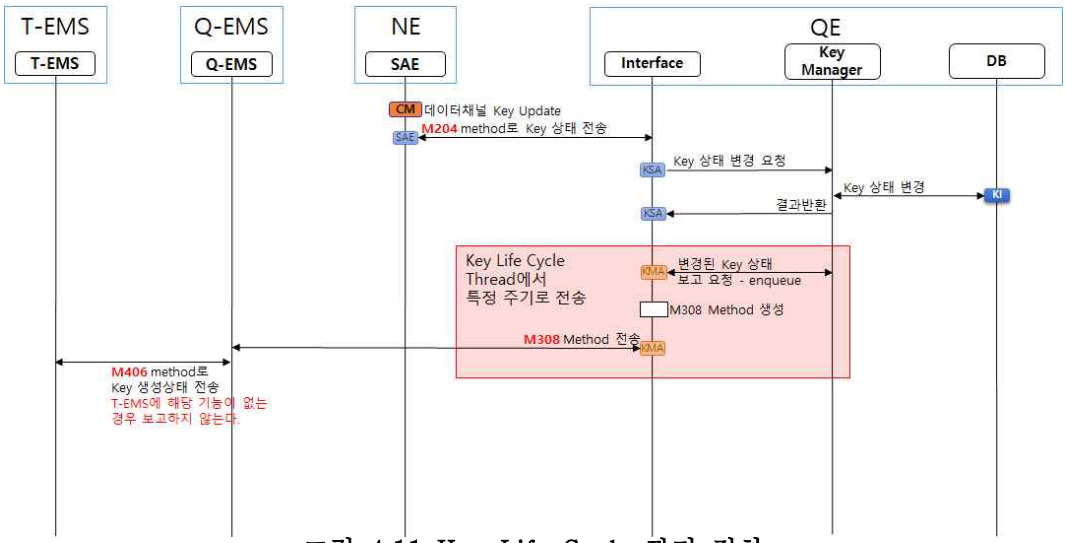

그림 4.11 Key Life Cycle 관리 절차

[F-301 - 통계정보 보고]

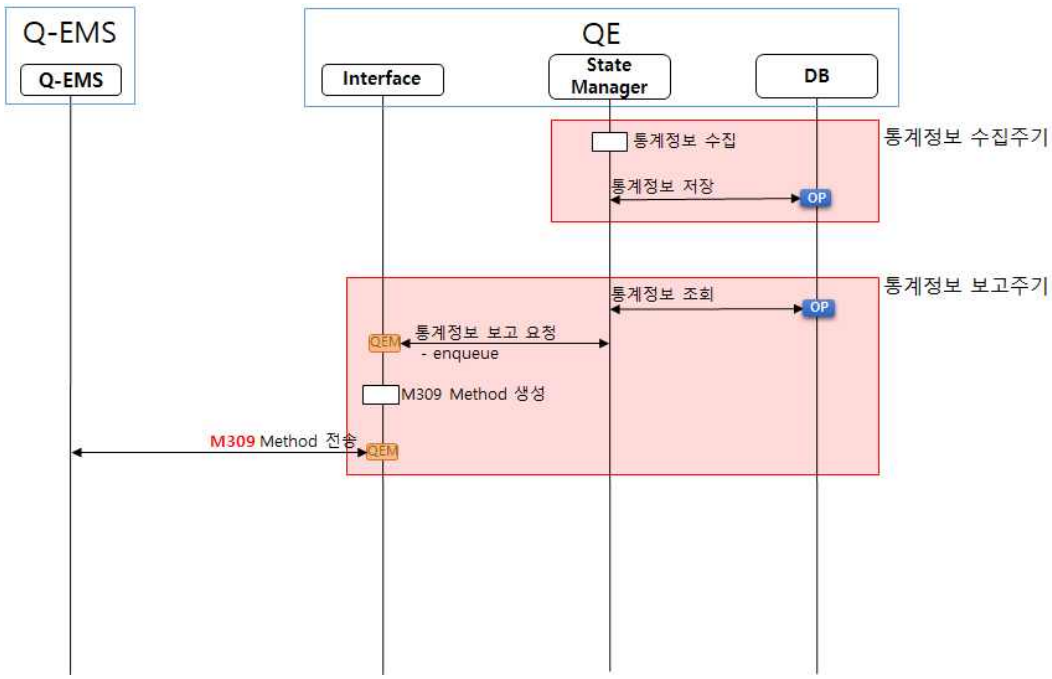

그림 4.12 통계정보 보고 절차

[F-302 - Event 생성 및 Action]

Event 생성 및 Action - 1/2

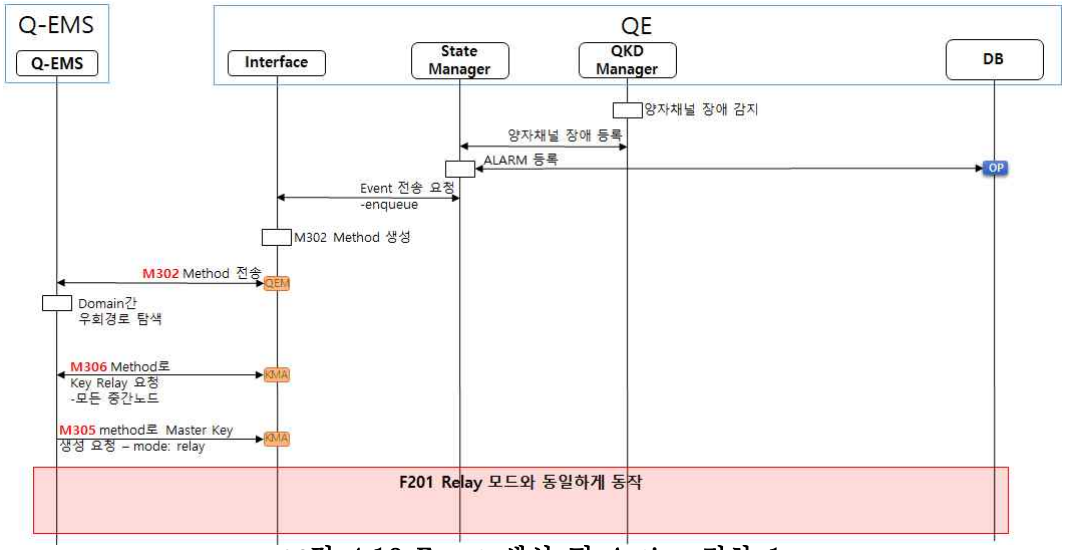

그림 4.13 Event 생성 및 Action 절차 1

Event 생성 및 Action - 2/2

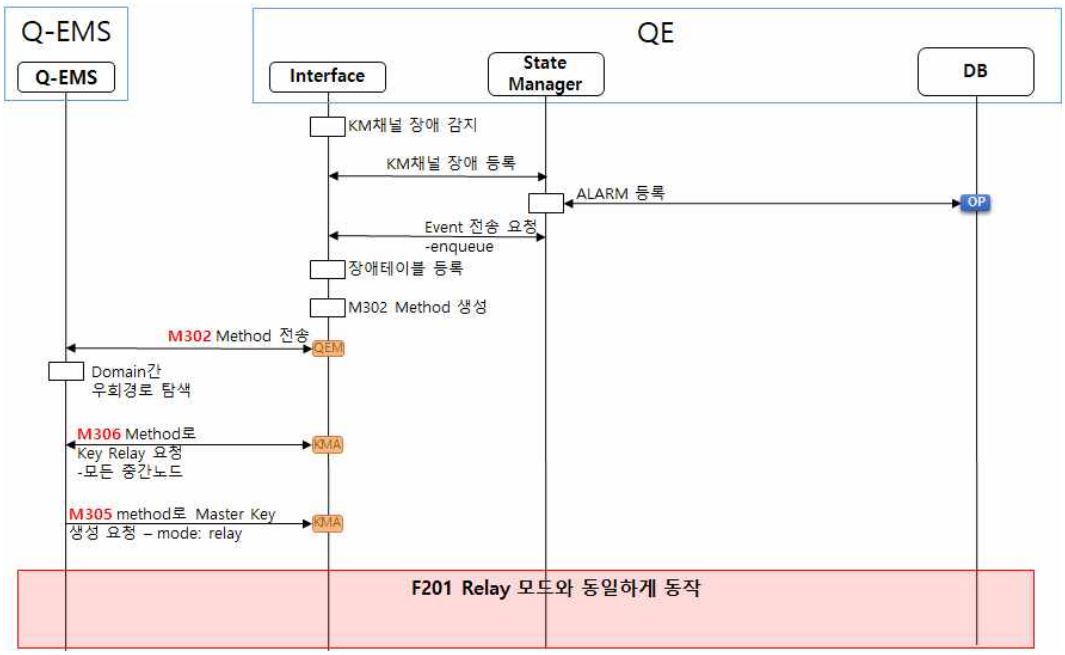

그림 4.14 Event 생성 및 Action 절차 2

[F-401 - 접근제어]

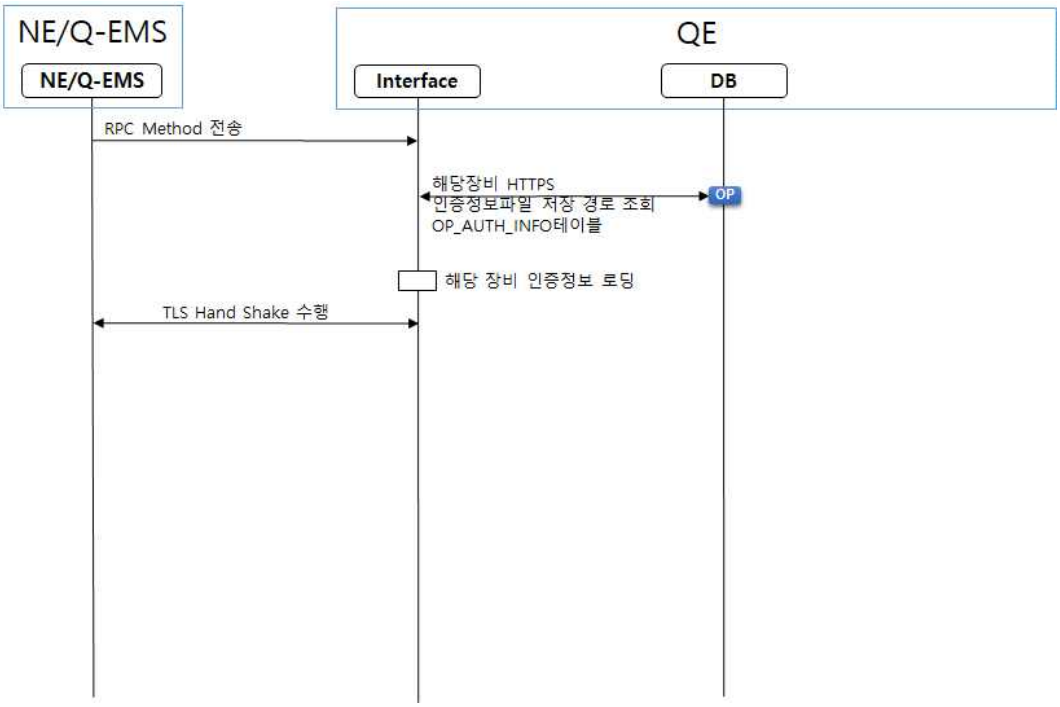

그림 4.15 접근제어 절차

4.2 시험망 적용을 위한 통합 KMS 상세 설계안 도출

4.2.1 설계 방향

표 4.9 설계 고려 사항

항목	설명
이종 장비 호환성	Key관리 계층과 QKD계층 분리 이종 NE(SAE), QKD장비 등과 연동이 용이한 방향 -표준화 인터페이스 사용 이종 벤더 장비ID를 수용 가능한 장비ID 체계 구성
확장성(Scale out) 및 고가용성	수행 영역과 업무에 따른 계층 구조로 개별 시스템간 확장 가능 및 이를 통한 고가용성
연동 편의성	Human readable한 프로토콜(JSON) 사용
보안성	구성요소간 연동 시 상호 인증 및 데이터 보호 물리적 보안경계 외부로 키 이동 불가 키 릴레이시 QKD Key stream 이용 OPT로 키보호 인가된 장비에게만 키 공급
버전 관리 용의성	메인 프레임의 변경 없이 동적 라이브러리 추가하는 형태로 블록별 버전별 기능 추가가 용이하며 하위 호환이 가능한 구조
관리자 환경	외부시스템, 운영자 WEB, 운영자 CLI등의 관리 요청을 JSON/REST로 일원화

4.2.2 LKMS 상세 설계안

가. 주요 기능 구성

표 4.10 LKMS 주요 기능

기능	설명
키 관리	QKD-KEY-FILE로 부터 KEY 추출 및 Reformat KMS간 KEY 유효성 검사 NE에서 KEY 공급 시 KEY 공급
운영정보관리	운영상 필요 설정 정보 관리 KEY 생성/공급/생애주기등의 정책 정보 관리 연동 대상(NE, Q-EMS 등) 인증정보 관리
구성관리	QKDE 구성정보 수집 및 Q-EMS로 전송
장애관리	FAULT, ALARM Event 생성 및 관리
성능관리	통계데이터 생성 및 전송
QKD관리	QKDE 상태 모니터링
외부연동 인터페이스	Q-EMS, CLI, WEB-GUI등을 통한 관리 인터페이스 제공

나. 블록 구성 및 역할

○ LKMS 블록 및 역할

표 4.11 LKMS 블록 및 역할

블록	역할
Interface(IF)	외부 시스템 연동 및 관리를 인터페이스 제공 –Restful API, GUI - HTTPS, CLI
QKDE Manager(QM)	QKDE 제어/관리, 양자채널 상태관리(State Manager와 연동), RNGE 관리
Access Control(AC)	장비 인증정보 등록 및 인증 수행
Key Manager(KM)	Key stream 처리, Q-KMS간 Key 동기화 관리, Key UUID 부여, Key Derivation 및 Key pair정보 관리, Key lifecycle 관리, Key Policy에 따른 QoS관리
State Manager(SM)	성능관리: 시스템 성능 통계정보 생성 장애관리: 장애 event 생성 프로세스관리
DB Access(DA)	Database 입.출력
Key Inventory(KI)	KEY Inventory - Key stream 데이터 관리 Keying material 저장소, Key 사용 이력 저장
Key Policy(KP)	KEY 사용정책정보 관리
Operation DB(OP)	LKMS 운영에 필요한 정보 관리
Data Replication(DR)	주요 데이터 (정책, 운영정보) 이중화 기능 수행
Block Manager(BM)	주요 블록 초기화 및 실행/관리 주요 작업 수행 관리

○ LKMS 블록 구성도

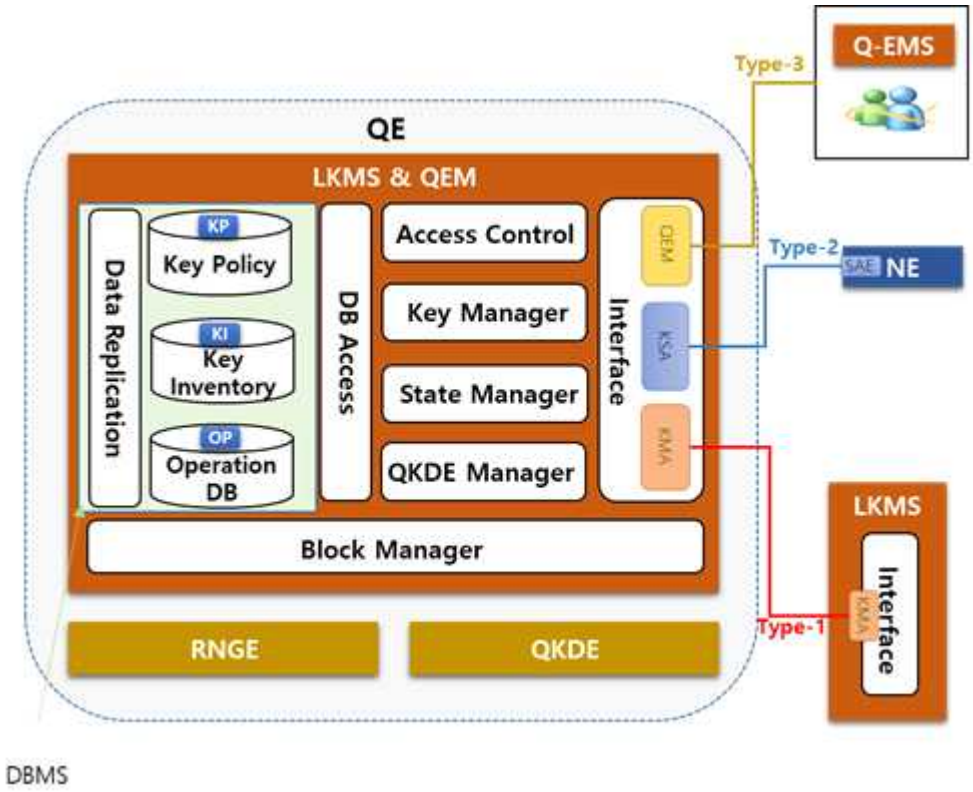

그림 4.20 LKMS 블록 구성도

4.2.3 Q-EMS 상세 설계안

가. 주요 기능 구성

표 4.12 Q-EMS 주요 기능

기능	설명
키 정책정보 관리	KEY 생성/공급/생애주기등의 정책 정보 관리 KMS 관리
운영정보관리	운영상 필요 설정 정보 관리 연동 대상(LKMS) 인증정보 관리 LKMS로 NE 인증정보 전달
구성관리	LKMS로 부터 LKMS 및 QKDE 구성정보 수집
장애관리	LKMS로 부터 FAULT, ALARM Event 수신 및 Action 장애 감지
성능관리	LKMS로 부터 통계데이터 수신 및 시각화
QKD 장비 및 망 관리	QKD 장비 및 망 상태 모니터링 QKDE 연결관계 그래프 생성 및 KEY전달 경로 생성 장애관리와 연동, KEY전달 우회경로 생성
외부연동 인터페이스	T-EMS, WEB-GUI등을 통한 관리 인터페이스 제공 LKMS 연동 인터페이스 제공

나. 블록 구성 및 역할

○ KMS 블록 및 역할

표 4.13 Q-EMS 블록 및 역할

블록	역할
Interface(IF)	외부 시스템 연동 및 관리를 인터페이스 제공 -Restful API, GUI - HTTPS, CLI
Access Control(AC)	장비 인증정보 등록 및 인증 수행
KMS Manager(KSM)	LKMS 구성 관리, T-EMS에 KMA간 KM채널 생성 요청, KMA에 키 생성 요청, 키 생성 상태 수신 및 T-EMS 전달
QKDN Manager (QNM)	QKD망구성 관리, 키 분배를 위한 경로 관리
State Handler(SH)	Event 처리 / 통계 수집 및 저장 가공
DB Access(DA)	Database 입.출력
Key Policy(KP)	키 사용정책정보 관리
Operation DB(OP)	QKD 망구성 정보 저장 Q-EMS 운영에 필요한 정보 관리
Data Replication(DR)	주요 데이터 (정책, 운영정보) 이중화 기능 수행
Block Manager(BM)	주요 블록 초기화 및 실행/관리 주요 작업 수행 관리

○ Q-EMS 블록 구성도

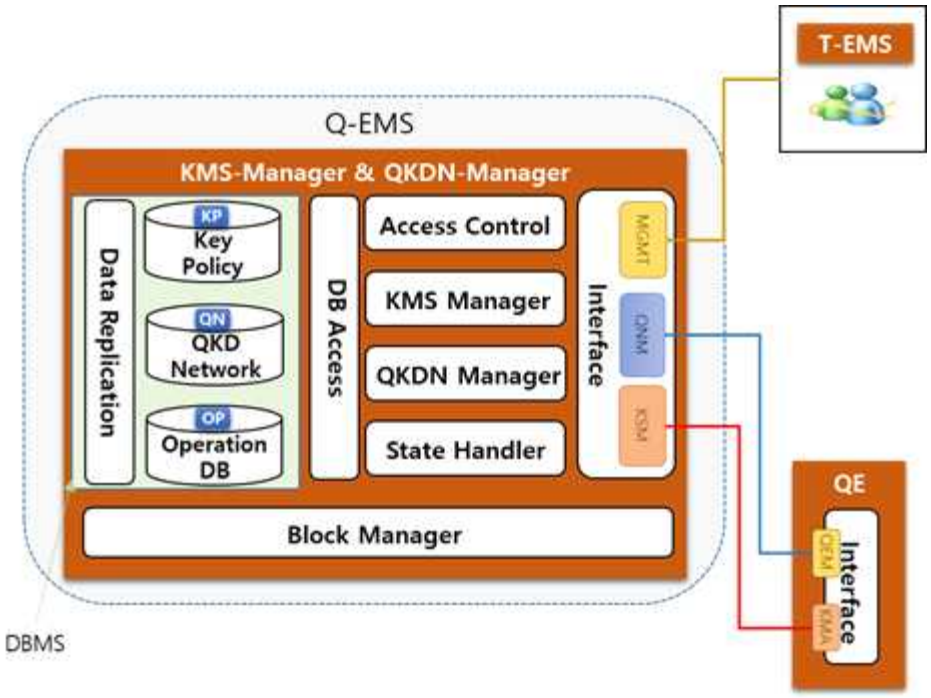

그림 4.22 Q-EMS 블록 구성도

제5장 결론

제5장 결론

본 연구는 양자키분배시스템(QKD)간 주고받은 양자키를 각 사이트 및 도메인별로 양자 키관리를 위한 키관리 계층 및 시스템의 설계연구를 수행하였다.

첫 번째는 양자암호통신 기술표준화 및 동향분석이다. 첫 번째 세부 과업 수행을 위해 3단계로 나누어서 업무를 진행하였다. 1단계는 QKD네트워크 및 KMS 표준화 및 기술 동향을 주요 표준화 단체인 ETSI, ITU-T, TTA등을 토대로 “요구 및 기능”, “구성요소 및 구조”, “관리 시스템 및 기능별 동작 절차”, “연동 인터페이스”, “보안”, “키 관리”, “데이터모델링”, “기타요소기술” 등의 관점에 맞추어서 관련 표준을 조사 및 분석하였다. 2단계 에서는 양자암호통신 시험망의 국내외 구축 사례 분석이며 주요 상용제품을 위주로 “시스템 구조” 및 “연동 인터페이스” 그리고 “주요 사용 기술 및 참고 표준”을 위주로 참고할 만한 사항을 조사 및 분석하였다.

두 번째로 세부 과업에서는 두 번째 과업의 결과에 기초하여 국가연구망에 적합한 QKD네트워크 연구 및 KMS 구조 설계를 4단계로 나누어서 진행 하였다. 1단계는 “QKD-KMS-전송교환장비 인터페이스 및 연계 구조”를 설계 하였으며 “QKD와 KMS간”인터페이스와 “KMS와 전송교환장비간” 인터페이스에 대해서 ETSI와 ITU-T 표준을 참고하여 설계하였다. 2단계에서는 “QKD Node 단위 및 시험망 통합 운영관리구조 설계”를 진행했으며 KMS를 구성하기 위한 계층을 “망관리계층”, “장비관리계층”, “QKD계층”, “전달계층”으로 구분하고 구성요소 및 역할에 대해서 정의하고 구성 요소간 관계를 KMS 계층 구조로써 도식화 하였다. 3단계 에서는 “이종 양자키 관리 구조 설계”를 위하여 TTA표준을 참고하여 QKD개체와 KM(KMS)개체를 분리하는 구조를 채택하였으며 다양한 프로토콜 및 기능 확장이 가능한 프레임워크에 대해서 연구하였다.

마지막으로 “설계안 및 설계전략 검증”으로 2단계로 진행이 되었다. 향후, 이러한 표준화 분석기반을 바탕으로, “통합 KMS 설계안 모델링 및 시뮬레이션을 통한 검증”을 수행하고, “단계별 시험망을 위한 KMS 설계 전략 도출” 연구를 수행하여, 연계시킬 예정이다.

ISBN 978-89-294-1163-3