Implementation and Prototype Verification: PQC Hybrid Key Exchange for Base Station/Security Gateway Version 1.0

27 May 2026

Security Classification: Non-confidential

Copyright Notice

Disclaimer

Compliance Notice

Table of Contents

1 Introduction

1.1 Overview

The GSMA Post-Quantum Telco Network Task-Force (PQTN) provides guidance for post-quantum migration in telecom networks. PQTN publishes technical reports [1, 2]. PQTN also encourages the community to share real-world experience with Post Quantum Cryptography (PQC) in telecoms through events at Mobile World Congress (MWC) and publications.

This document is one of the series sharing practical experience. This document shares details of a vendor's hybrid key exchange mechanism and its implementation. Hybrid systems are one option for migrating security connections in telecom network (e.g. IPsec, etc.). The objective is to provide telecommunications-specific insights to vendors, operators and researchers. Results from demonstrations provide information for future networks: to address quantum attacks such as hoarding attacks and to progress quantum readiness in telecom.

1.2 Scope

This document describes a telecom post-quantum use case including the hybrid key exchange solution design, prototype implementation and verification. The work is based on Guidelines for Telecom Use Cases [1], with the following focus:

1. Provide the migration solution analysis for operator's network, including:

    a) hybrid key exchanges in IPSec/IKEv2 of the base station

2. Test the solution and migration process, including:

    b) hybrid key exchanges in IPSec/IKEv2 of the base station

This document makes recommendations on updates to key negotiation in IKE based on evidence gathered during implementation of the demo.

1.3 Definitions

The key words "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", and "MAY" in this document SHALL be interpreted as described in RFC 2119 [21]

1.4 Abbreviations.

Term	Description
3GPP	Third generation partnership program
AES	Advanced Encryption Standard
API	Application Programming Interface
CRQC	Cryptographically Relevant Quantum Computer
ECDH	Elliptic Curve Diffie Hellman
eNB	e-Node B
gNB	g Node B
HNDL	Harvest Now, Decrypt Later
IETF	Internet Engineering Task Force

Term	Description
IKE	Internet Key Exchange
IPsec	Internet Protocol Security
KEM	Key encapsulation mechanism
ML-DSA	Module-Lattice Digital Signature Algorithm
ML-KEM	Module Lattice based Key Encapsulation Mechanism
MTU	Maximum transmission unit
NAT	Network address translation
ng-eNB	Next generation e-Node B
NIST	National Institute of Standards and Technology
OAM	Operations and Maintenance
OSS	Operational Support Systems
PKI	Public Key Infrastructure
PQC	Post Quantum Cryptography
PQ/T	Post-Quantum/Traditional
RAN	Radio Access Network
RSA	Rivest Shamir Adelman
SCP	Secure Copy Protocol
SFTP	Secure File Transfer Protocol
SecGW	Security Gateway
SNMP	Simple Network Management Protocol
SNDL	Store Now, Decrypt Later
SSH	Secure shell
TLS	Transport Layer Security (a major Internet secure communication protocol)
UDP	User datagram protocol
UE	User equipment

Table 1 Abbreviations

1.5 References

Ref	Doc Number	Title
[1]	PQ.03	GSMA PQ.03 – Post Quantum Cryptography – Guidelines for Telecom Use Cases, Version 2, Sept 2024
[2]	PQ.01	GSMAPQ.01 – Post Quantum Telco Network Impact Assessment Whitepaper, Version 1.0, 17 Feb 2023

Ref	Doc Number	Title
[3]	FIPS 203	National Institute of Standards and Technology (2024) Module-Lattice-Based Key-Encapsulation Mechanism Standard. (Department of Commerce, Washington,D.C.), Federal Information Processing Standards Publication (FIPS) NIST FIPS 203. https://doi.org/10.6028/NIST.FIPS.203
[4]	FIPS 204	National Institute of Standards and Technology (2024) Module-Lattice-Based Digital Signature Standard. (Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) NIST FIPS 204. https://doi.org/10.6028/NIST.FIPS.204
[5]	FIPS 205	National Institute of Standards and Technology (2024) Stateless Hash-Based Digital Signature Standard. (Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) NIST FIPS 205. https://doi.org/10.6028/NIST.FIPS.205
[6]	RFC 3411 SNMP v3	An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks, RFC 3411, Dec 2002 https://datatracker.ietf.org/doc/rfc3411/ (Accessed 27 Jan 2026)
[7]	RFC 4301	Security Architecture for the Internet Protocol, RFC 4301, Dec 2005 https://datatracker.ietf.org/doc/rfc4301/ (Accessed 27 Jan 2026)
[8]	RFC 7296	Internet Key Exchange Protocol Version 2 (IKEv2), October 20214 RFC 7296, https://datatracker.ietf.org/doc/rfc7296/ (Accessed 20 Jan 2026)
[9]	RFC 7383	Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation, RFC 7383, Nov 2014 https://datatracker.ietf.org/doc/rfc7383/ (Acessed 27 Jan 2026)
[10]	RFC 8784	Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security, June 2020 https://datatracker.ietf.org/doc/rfc8784 (Accessed 27 Jan 2026)
[11]	RFC 9242	Intermediate Exchange in the Internet Key Exchange Protocol Version 2 (IKEv2), RFC 9242, May 2022 https://datatracker.ietf.org/doc/rfc9242/ (Accessed 27 Jan 2026)
[12]	RFC 9370	Multiple Key Exchanges in the Internet Key Exchange Protocol Version 2 (IKEv2), RFC 9370, May 2023 https://datatracker.ietf.org/doc/rfc9370/ (accessed 27 Jan 2026)
[13]	RFC 9794	Terminology for Post-Quantum Traditional Hybrid Schemes, RFC 9794, June 2025, IETF https://datatracker.ietf.org/doc/rfc9794/ (Accessed 20 Jan 2026)

Ref	Doc Number	Title
[14]	-	PQ/T Hybrid Key Exchange with ML-KEM in SSH draft-ietf-sshm-mlkem-hybrid-kex-08, Jan 2026 https://datatracker.ietf.org/doc/draft-ietf-sshm-mlkem-hybrid-kex/ (accessed 27 Jan 2026)
[15]	-	Post-quantum Hybrid Key Exchange with ML-KEM in the Internet Key Exchange Protocol Version 2 (IKEv2), draft-ietf-ipsecme-ikev2-mlkem-03 https://datatracker.ietf.org/doc/draft-ietf-ipsecme-ikev2-mlkem/
[16]	Frodo KEM	FrodoKEM: Learning With Errors Key Encapsulation, Preliminary Standardization Proposal, 29 Sep 2025 https://frodokem.org/ (Acessed 2026-01-27)
[17]	Shor	Shor, Peter W. "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer." SIAM review 41.2 (1999): 303-332.
[18]	33.210	3GPP TS 33.210 IP network layer security
[19]	33.501	3GPP TS 33.501 Security architecture and procedures for 5G system
[20]		"Reducing the Number of Qubits in Quantum Discrete Logarithms on Elliptic Curves", Clémence Chevignard, Pierre-Alain Fouque, André Schrottenloher, Eurocrypt 2026
[21]	RFC 2119	Key words for use in RFCs to Indicate Requirement Levels, S. Bradner http://www.ietf.org/rfc/rfc2119.tx
[22]		Reducing the Number of Qubits in Quantum Discrete Logarithms on Elliptic Curves", Clémence Chevignard, Pierre-Alain Fouque, André Schrottenloher, Eurocrypt 2026

Table 2 References

2 Background

Telecom networks rely upon cryptographic algorithms for security: ensuring the confidentiality and communications, ensuring the integrity of network equipment and authenticating the communicating parties. Telecoms uses *asymmetric* or *public key* cryptographic algorithms, whose security is based on mathematical hard problems.[1] The current versions of these asymmetric algorithms, such as Elliptic Curve Diffie-Hellman and RSA, will bevulnerable according to [17][20] to attack by adversaries possessing a cryptographically relevant quantum computer (CRQC) [22]: a sufficiently powerful and accurate quantum computer capable of executing variant of Shor's quantum algorithm at scale [17] [20]. CRQC attacks

[1] Telecom also uses symmetric cryptographic algorithms, e.g. AES, but they are not the subject of this report.

offer an exponential acceleration relative to known classical attacks, making the underlying asymmetric cryptographic methods insecure. This threat is relevant in the present day and near future, owing to the *harvest now, decrypt later* (HNDL) attack, in which bad actors can harvest and store communications traffic now, to decrypt in the future, once a CRQC is available. This HNDL attack endangers long-lived secrets.

This document is concerned with the threat that quantum attacks pose to the connection between network base stations (e.g., gNBs and ng-eNBs, etc.) and the security gateways that provide access to network functions in the core network. Use of SecGWs between the RAN and core network is not mandated by 3GPP standards but is commonly adopted by operators to enhance security. The connections typically involve multiple components using public key cryptography.

- The connection between a gNodeB and a SecGW (or between two gNodeBs), may use an encrypted IPsec tunnel whose establishment depends on the key exchange protocol such as IKEv2 specified in [8]. However, current versions of key exchange protocols typically use traditional asymmetric cryptographic algorithms that are vulnerable to attack by a CRQC.
- The communications between OSS/OAM systems and network elements (in addition to the IPsec tunnel between a base station and the SecGW). These communications use SSH or TLS which current specifications rely upon quantum-vulnerable algorithms. As of writing this document, IETF drafts defining post-quantum variants of the latter protocols are still under discussion in IETF Working Groups. However, implementations of TLS 1.3 and SSHv2 based on quantum-safe key exchange mechanisms are already deployed to a greater or lesser extent (e.g. OpenSSH since release 9.0).

Securing the connection between base stations and SecGWs against quantum attacks requires modifications to all these quantum-vulnerable components and is an important and necessary step when migrating to quantum safe networks.

There are multiple ways to modify the cryptography used between gNodeBs and the SecGW to achieve quantum safety:

1. Switch to the use of pre-shared keys.

2. Replace currently used quantum-vulnerable public key methods with post-quantum secure cryptographic algorithms using NIST standards.

3. Adopt a hybrid solution that combines one of the above methods with a traditional (i.e., quantum vulnerable) key exchange algorithm. For example, by combining a PQC key exchange with a traditional method (ECDH).

Post-Quantum Traditional (PQ/T) hybrid scheme as defined in [13], is a multi-algorithm scheme where at least one component algorithm is a post-quantum algorithm and at least one is a traditional algorithm. During the transition from traditional to post-quantum algorithms, there may be a desire or a requirement for protocols that use both algorithm types. The hybrid solution listed above (3) protects against quantum attacks but also retains existing methods to ensure that present-day levels of security (e.g., against traditional attacks) are maintained, if unanticipated issues are discovered in the newer quantum safe algorithms.

Organisations are increasingly enabling hybrid PQC approaches to protect communications. Examples include the enablement of hybrid key agreement methods for TLS across multiple services, such as served websites and APIs, by major companies (e.g. Akamai, Cloudflare). This also includes the introduction of hybrid methods for browsers, in some cases initially using Kyber, then later ML-KEM, as the PQC algorithm. This industry-wide shift demonstrates a pragmatic approach to quantum readiness, leveraging hybrid encryption as a transitional safeguard while PQC standards and software and hardware implementations mature. The telecommunications industry itself involves many contexts where quantum vulnerable VPN methods such as IPsec or TLS are deployed This includes both internal functionalities, such as communications between network components, and external services or products, such as SD-WANs. In some such instances it may be desired or necessary to consider hybrid PQC approaches, as standards and implementations mature.

This document describes the use of hybrid key exchange methods to secure the base station to SecGW connection, and direct connections between base stations via, e.g., the X2 interface as used to forward control signaling, which similarly relies on IPsec. The document describes prototype implementations using a hybrid key exchange by combining traditional public key methods with newer post-quantum algorithms standardized by NIST.

3 Use Case Flashback

3.1 Use case introduction

The use case described in this document is the connectivity between gNodeB/eNodeB and the mobile packet core. This connection encompasses both a control-plane interface (S1-MME, or N2) and a user-plane interface (S1-U or N3). The combination is typically protected by one or more IPSec tunnels, terminating on a Security Gateway (SecGW), which provide integrity protection and confidentiality of the signaling and subscriber data.

The quantum threat affects several components in this use-case:

- Authentication between base station and SecGW
- Key exchange between base station and SecGW
- Certificate management of base station and SecGW

The second item on the list may be the most urgent, because of the HNDL problem: if an attacker stores eavesdropped data today, then that data may be decrypted once a CRQC becomes available. This makes key exchange an issue that must be solved as soon as possible.

Authentication on the other hand is only at risk when a CRQC is available. Other factors mean that it should also be considered a priority: PKI systems are often central components that provide certificates to many different components, covering multiple vendors. The migration of the PKI system to post-quantum algorithms involves many dependencies; the complexity of the migration of PKI should not be underestimated.

See [1] section 5.2 for more details on this use-case.

3.2 Cryptographic Inventory

The cryptographic protocols to be considered include:

3.2.1 Protection of data between base stations and SecGW:

Data protection is provided by IPSec protocol as specified in IETF [7] for SecGW interfaces as specified by 3GPP [18]. It consists of a mutual authentication step based on x.509 certificates, followed by a key exchange to establish session keys for integrity protection and encryption of signalling and subscriber data. 3GPP specified interface instructions and PKI management when IPsec is used between the base station and SecGW [19].

3.2.2 Protection of management data between network elements and OSS/OAM systems

Management systems use protocols like SSH and HTTPS for configuration and monitoring, as well as SNMP for monitoring, and SCP, SFTP or FTPS for file transfer. These protocols, relying upon SSH or TLS to protect the data in transit, are quantum-vulnerable when they use (Elliptic Curve) Diffie-Hellman ((EC)DH) or RSA for the key exchange, or other quantum-unsafe algorithms during the authentication phase (e.g. ECDSA, RSA-PSS.). SNMP has an encrypted version (version 3) but since keys are manually configured, it is not quantum-vulnerable.

3.2.3 PKI

PKI systems are responsible for creation, distribution, verification and renewal of X.509 certificates used by network entities to authenticate themselves using a chain of trust which is anchored on one or more trusted certificate authorities. The CMPv2 protocol is generally used to communicate between network elements and PKI functions. This protocol uses quantum-vulnerable methods to transfer certificate material, and the certificates themselves are based on quantum-vulnerable cryptographic algorithms such as RSA.

See [1] section 5.2.4 for more detail

3.3 Implementation inventory

The section is referred to the section 6.1.2

4 Risk Assessment

This section evaluates the security risks posed by quantum computing to both user plane and the control plane link between the Base Station (gNB) and the Security Gateway (SecGW). In current 5G architectures, the N2 (Control Plane) and N3 (User Plane) interfaces are protected by IPsec tunnels when traversing non-secure domains. These tunnels rely on the IKEv2 for mutual authentication and session key establishment.

4.1 Key Risks

The most immediate risk is the HNDL. Adversaries may intercept and store encrypted N2 and N3 traffic. While this information or data is secure against classical computers, it can be retroactively decrypted once a CRQC becomes available. For telecommunication operators,

if the IPsec protection is compromised via the gNB and SecGW link, all user data may lose its confidentiality in the user plane level.

4.2 Requirements from the Assessment

The current reliance on IPsec tunnel establishment represents a significant long-term vulnerability. To mitigate these risks in base station and SecGW, the migration toward Post-Quantum Cryptography (PQC) of the key exchange mechanisms is necessary to ensure the continued confidentiality and integrity of both the user/control plane links.

5 Migration Strategy for Demo Implementation

Based on the guidelines for telecom use cases [1], the following migration is selected for demo implementation

IPSec has two post-quantum vulnerabilities that need to be considered: key exchange mechanism (Diffie-Hellman), and the authentication mechanism used to identify the endpoints to each other (e.g. RSA, ECDSA).

- the authentication mechanism used to identify the endpoints to each other (e.g. RSA, ECDSA), and the key exchange mechanism (Diffie-Hellman).For key exchange, using pre-shared keys may also be an option [10] although since this standard is itself relatively new and may not be supported by all vendors, it may be preferable to migrate directly to quantum-safe key exchange algorithms (e.g., Hybrid).
- For authentication, operators need to evaluate the benefits of
    - introducing hybrid certificates directly via corresponding upgrades or replacement of PKI systems, versus
    - using pre-shared keys (considering them quantum safe) for a transition period before upgrading the PKI infrastructure.

In terms of the best practice, GSMA PQTN TF recommends working with vendors and evolving standards and planning for the migration to Post Quantum Cryptography.

Three areas to consider in post-quantum key exchange migration:

- Communication overheads: Hybrid key exchange increases computing and communication overheads and may cause service performance bottlenecks.
- Compatibility: There are many devices on the live network. Third-party devices need to be interconnected; Compatibility needs to be considered.
- Urgently needed: There are urgency needs regarding to the fully standardization of the hybrid key exchange in basic protocols such as IPsec. Attacks such as HNDL may result to potential risks

6 Detailed Implementations

6.1 Overview and Dependencies

6.1.1 Summary of Demos

In this section a demonstration implementation and evaluation results are provided. A summary table is as follows:

Table 6-X Summary of demo of key exchanges

Demo No.	labels	Scenario Description	partners
1.	IPsec/IKEv2 Hybrid KE over X2/Xn	Analysis of the impact of interoperability and performance on services over the interface X2/Xn with the Hybrid KE of IPsec/IKEv2	Huawei
2.	IPsec/IKEv2 Hybrid KE over gNB/eNodeB and SecGW	Analysis of the impact of interoperability and performance on services over the gNB/eNodeB to SecGW with the Hybrid KE of IPsec/IKEv2	TBD

6.1.2 Dependencies

Based on the description of the use case in section 3, There are potential data size increase, procedure update, and additional indication caused by post-quantum security context. To implementation the demo, some dependencies are provided as the potential principle and reference for the detailed implementation including the re-design of IKEv2 Fragmentation [6.1.2.1], supportive of additional key exchanges[6.1.2.2] and the re-design of hybrid KE of ML-KEM [6.1.2.3].

6.1.2.1 Consideration of improved IKEv2 Fragmentation

The transition toward hybrid key exchange mechanisms requires an update in how the Internet Key Exchange Protocol Version 2 (IKEv2) handles large payloads. Therefore, RFC 7383 [9] is introduced in this proposed implementation, which specifies a method for managing large IKEv2 messages into smaller sets of messages called IKE fragment messages to avoid IP fragmentation. This is because the public key of the post-quantum KEM is usually larger than the maximum transmission unit (MTU) of current networks.

Some implementations policies may cause IP fragments to be discarded. Therefore, RFC 7383 [9] is introduced to avoid IP fragments and the IKEV2_FRAGMENTATION_SUPPORTED specified in RFC is also enabled to determine whether the two ends support IKEv2 fragmentation.

6.1.2.2 .Consideration of Supportive IKEv2 additional key exchanges

As described in the previous section, the first packet in the IKEv2 protocol does not support fragmentation, the post-quantum public key may be longer than the maximum size of the packet. Therefore, a new message format is required in the IKE protocol to support the transmission of additional keys due to the increased key size. A new key exchange protocol is introduced in [11], named "intermediate exchange" improved from existing IKE

fragmentation mechanism in [9]. It defines how to use a new message named "IKE_INTERMediaTE" to exchange additional large messages.
Additional key exchanges can be performed by using IKE_INTERMEDIATE or IKE_FOLLOWUP_KE messages and deriving new SKEYSEED and KEYMAT keying material, which supports up to 7 additional key exchanges [12]. This new feature allows the use of new post-quantum key exchanges in derived IKE and its child SA keys.

6.1.2.3 Consideration of ML-KEM key lengths in hybrid KE demos

Earlier IETF standards define the mechanism for hybrid key agreement in IKEv2. The latest IETF group draft [15] defines Hybrid Key Exchange with ML-KEM, which is the latest post-quantum algorithm that can be applied in IKEv2/IPsec

As shown in the Table 6.1.2-X, the draft currently plans to define the following three key lengths as specifications:

Table 6.1.2-X Payload field of ML-KEM for different key size (from draft [15])

KEM size	Payload Length (initiator/responder)	Key Exchange Method Num.	Data Size in Octets (initiator/responder)
ML-KEM-512	808/776	TBD35	800/768
ML-KEM-768	1192/1096	TBD36	1184/1088
ML-KEM-1024	1576/1576	TBD37	1568/1568

ML-KEM-512, ML-KEM-768, and ML-KEM-1024 are planned to be included in the demo specification. NIST recommends using ML-KEM-768 as the default parameter set, as it provides a large security margin at a reasonable performance cost [3]. ML-KEM-1024 is also included for a more secure option. ML-KEM-512 is also included as an option because the size of the ML-KEM-512 public key is within one MTU, so fragmentation can be reduced compared to ML-KEM-768 and ML-KEM-1024. Further changes may occur based on the progress of NIST, IETF, etc.

6.2 Demo 1: IPsec/IKEv2 Hybrid KE over X2/Xn

6.2.1 System Architecture and Entities

X2/Xn is a transmission interface between base stations defined in a 3GPP specification, and is used to forward control signaling, for example, a UE context. Session 6.2 mainly considers scenario and implementation where base stations (e.g., eNodeB or gNB) are directly connected. The architecture connected through the SecGW will be discussed in other demos. The following Figure 6.2.1-x shows the system architecture for different connection cases of the direct IPsec tunnel.

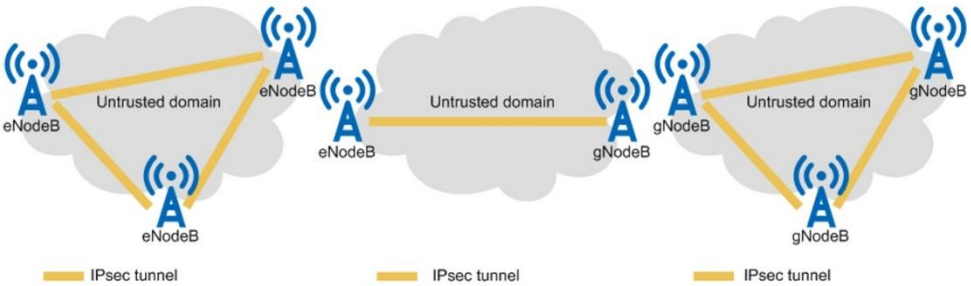


Figure 6.2.1-x example cases of the direct IPsec tunnel over the X2/Xn interface

There are 3 cases to be considered in the system architecture:

1. direct IPsec tunnel over the X2 interface between eNodeB(s),
2. direct IPsec tunnel over the X2 interface between the eNodeB and gNodeB,
3. direct IPsec tunnel over the Xn/eXn interface between gNodeB(s)

6.2.2 Impact Analysis and Algorithm Selection

The following Figure **6.2.2-y1** gives the general IPsec configuration model of the X2 interface on the eNodeB:

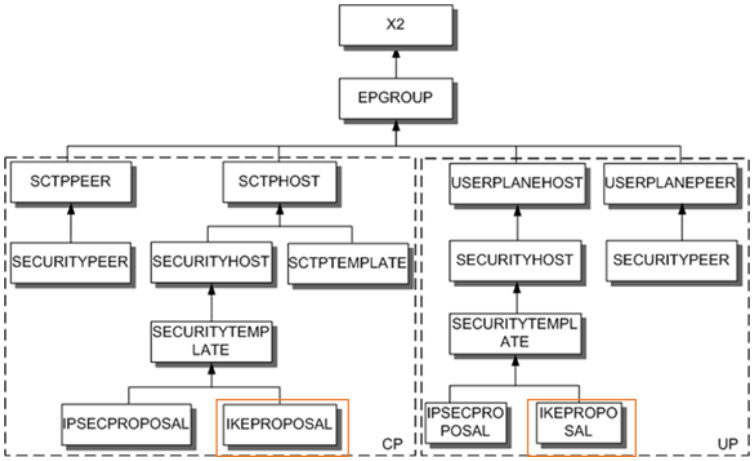


Figure 6.2.2-x1 IPsec configuration in X2 interface of eNodeB

Based on the analysis in dependencies, the update of Hybrid key exchange affects the IKEPROPOSAL shown in Figure 6.2.2-x1. Currently, the IKE Proposal supports the configuration of cryptographic algorithms during IKEv2 establishment, as shown in the following table. According to the [12] protocol, several ADDKEs are added in the IKE configuration due to the increasing key size shown in the following table 6.2.2-y1.

Table 6.2.2-y1 configuration and Id sets of IKEv2

ID	Name	Description	NEW
PROPID	Proposal ID	Indicates the ID of the IKE proposal.	
ENCALG	Encryption Algorithm	Indicates the encryption algorithm used in the IKE proposal.	
AUTHALG	Authentication Algorithm	Indicates the authentication algorithm used in the IKE proposal.	
AUTHMETH	Authentication Method	Indicates the authentication mode used in the IKE proposal.	
DHGRP	Diffie-Hellman Group	Indicates the Diffie-Hellman (DH) group of the IKE proposal. Actual Value Range: DH_GROUP1, DH_GROUP2, DH_GROUP14, DH_GROUP15, DH_GROUP19, DH_GROUP20, DH_GROUP31	

ADDKENUM	Additional Key Exchange Number	Meaning: Indicates the Number of Additional Key Exchanges	
ADDKE1	Additional Key Exchange 1	Meaning: Indicates Additional Key Exchanges 1 used In the IKE proposal	**NEW**
ADDKE2	Additional Key Exchange 2	Meaning: Indicates Additional Key Exchanges 2 used In the IKE proposal	**NEW**
……	……	……	**…**
ADDKEn	Additional Key Exchange n	Meaning: Indicates Additional Key Exchanges n used In the IKE proposal	**NEW**
PRFALG	PRF Algorithm	Meaning: Indicates the Pseudo-random Function (PRF) algorithm used in IKEv2.	
.....	.....	.....	

For the selection of PQC KEM algorithm in the validation, ML-KEM, the NIST-standardized post-quantum key-encapsulation mechanism, was chosen for this demo. .

For this demo NIST standardised ML-KEM was selected as the key encapsulation mechanism. Different algorithms will have performance implications that should be evaluated in the context of the specific implementation, taking into account factors such as key sizes, bandwidth requirements, computational overhead on constrained devices, latency impact on control and user planes, and storage requirements for keys and certificates.

Considering the complexity of verification and implementation, Demo 1 implements with one extended ADDKE supporting of Post-quantum Hybrid Key Exchange with ML-KEM in the Internet Key Exchange Protocol Version 2 (IKEv2) [15]. The ML-KEM algorithms are implemented. In practice, if there are multiple algorithms available, it is better to implement a selection of algorithms. Algorithm negotiation can then be conducted through protocol. E.g. the base station (i.e., responder) can receive a multiple-algorithm negotiation request with candidate algoIDs during IKE_SA_INIT. The algorithm is selected and returns selected algoIDs in ADDKEs for generating keys (e.g., at least two) of both sides, which is for the initiator/responder to complete the Hybrid KE and establish a quantum secure IPsec.

6.2.3 Design and Development Approach

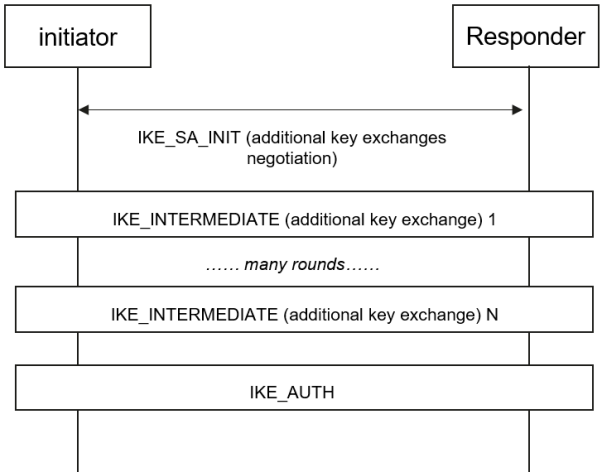


Figure 6.2.3-x1 Multiple IKE Exchange Procedure in [12]

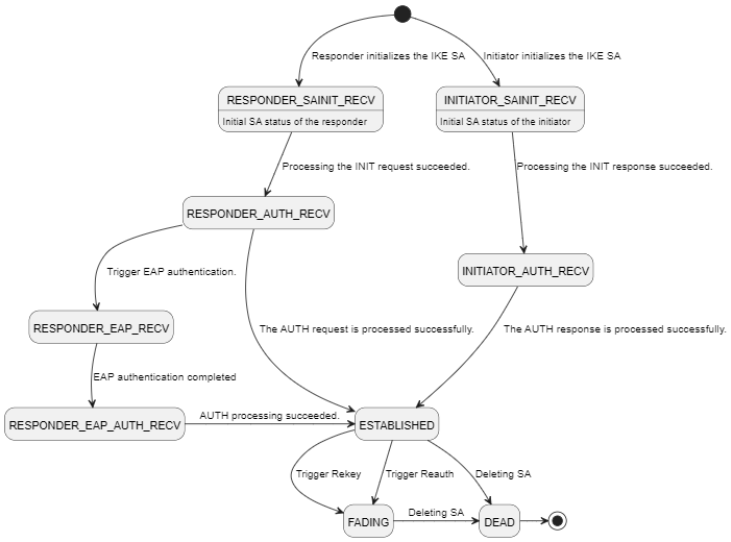


Figure 6.2.3-x2 Current IKE SA Finite State Machine (FSM)

Based on the **Figure 6.2.3-x1,** the following new features and capabilities are supported in the approach, including:

- Support for the new exchange type ' IKE_INTERMEDIATE ' for IKE
  - Packet encapsulation/resolving of the new type ' IKE_INTERMEDIATE '
  - IKE_AUTH supports the new type ‘IKE_INTERMEDIATE '
- Support the configuration of mixed key algorithms in Proposal.
- Support the new negotiation item of 'ADDKE Transform '.

The state machine path of the current IKE SA establishment process is shown in **Figure 6.2.3-x2.** Based on the state machine analysis and [12], we concluded that a new exchange type message should be developed, i.e., `IKE_INTERMEDTE`, and implemented between `IKE_INIT` and `IKE_AUTH`.

The current state machine does not support `IKE_INTERMEDTE` switching. Therefore, the current state machine needs to be reconstructed. New packet processing in the negotiate authentication phase is added in `IKE_INTERMEDTE` to Demo1, as shown in the following:

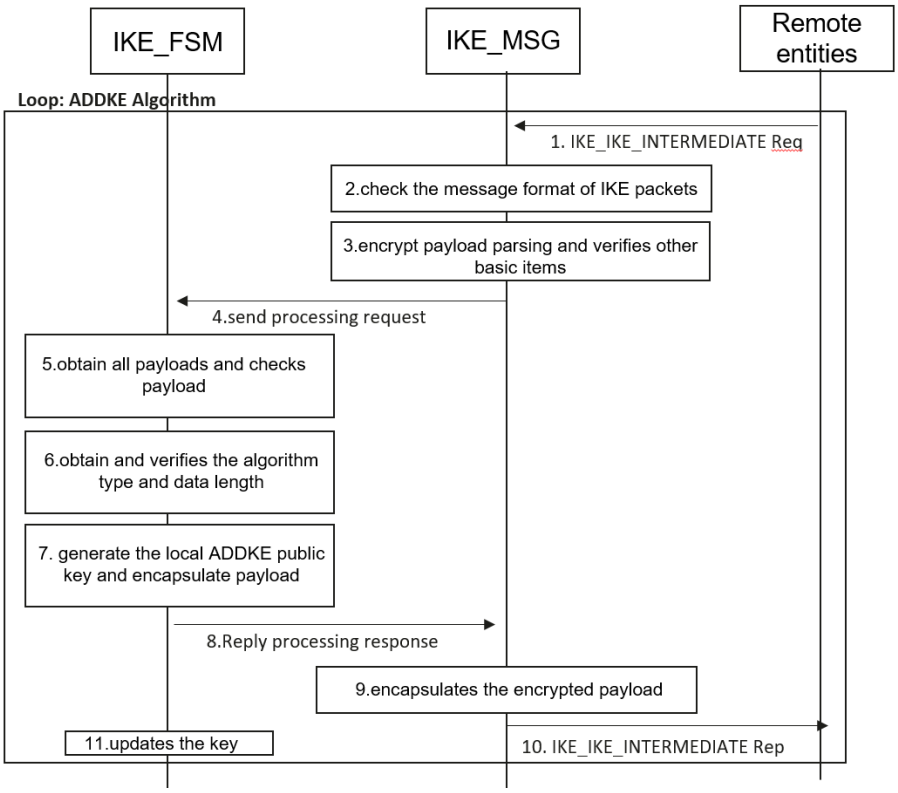


Figure 6.2.3-x3 Detailed implementation of ADDKE processing

Shown in Figure 6.2.3-x3, for each round of ADDKE in the Demo1 for implementation.

1. A remote entity sends IKE_IKE_INTERMEDIATE Req to IKE_MSG.
2. IKE_MSG checks the message format of IKE packets.
3. IKE_MSG encrypts payload parsing and verifies other basic items in the payload.
4. IKE_MSG sends processing request to IKE_FSM.
5. IKE_FSM obtains all payloads and checks the completeness of the KE payload.
6. IKE_FSM obtains and verifies the algorithm type and data length of the KE payload.
7. IKE_FSM generates the local ADDKE public key and encapsulates the KE payload.
8. IKE_FSM Replies to the response packet to the IKE_MSG.
9. IKE_MSG encapsulates the encrypted payload based of the response.
10. IKE_MSG sends IKE_IKE_INTERMEDIATE Rsp
11. IKE_FSM updates the key using the key derivation function.

NOTE: The steps may be varied from the standard due to the implementation of the demo.

6.2.4 Deployment Configuration and Analysis

In this section, we first discuss the deployment configuration, then analyze the packet of the demo solution, and discuss the impact of benchmarking before deployment. Then, we deploy the test equipment and conduct the test.

Finally, a conclusion of demo1 is given in Section 6.2.5.

6.2.4.1 Configuration

The detailed IKEv2 configuration and environment of the validation prototype is described in the following Table 6.2.4-y1:

Table 6.2.4-y1 IKEv2 configuration and environment of Demo1

Configuration	Item	Description
Certificate	Signature algorithms	RSA 4096
IKE Proposal	Key negotiation algorithms (non-PQC)	DH_Group 31 (i.e. X25519)
	Key negotiation algorithms (PQC)	ML_KEM 1024
	Encryption algorithm	AES_GCM_256
	pseudo-random function (PRF)	HMAC_SHA384
Board	model number	UMPT boards (set of two)
	product	BTS3900 series for testing only
Crypto-Library	PQC library	oqs-provider 0.8.0
Maximum IKE SA links between two nodes	-	520

6.2.4.2 Demo Packet Analysis and Pre-deployed Analysis

Before enabling hybrid key negotiation

355	9.000000	91.0.107.191	91.0.107.114	ISAKMP	282 IKE_SA_INIT MID=00 Initiator Request
365	9.000000	91.0.107.114	91.0.107.191	ISAKMP	303 IKE_SA_INIT MID=00 Responder Response
375	9.000000	91.0.107.191	91.0.107.114	ISAKMP	2270 IKE_AUTH MID=01 Initiator Request
381	9.000000	91.0.107.114	91.0.107.191	ISAKMP	2129 IKE_AUTH MID=01 Responder Response
383	9.000000	91.0.107.114	91.0.107.191	ISAKMP	99 INFORMATIONAL MID=00 Responder Request
399	10.000000	91.0.107.191	91.0.107.114	ISAKMP	99 INFORMATIONAL MID=02 Initiator Request
400	10.000000	91.0.107.191	91.0.107.114	ISAKMP	99 INFORMATIONAL MID=00 Initiator Response
417	10.000000	91.0.107.114	91.0.107.191	ISAKMP	99 INFORMATIONAL MID=02 Responder Response

After enabling hybrid key negotiation

4	0.000000	91.0.107.191	91.0.107.134	ISAKMP	282 IKE_SA_INIT MID=00 Initiator Request
11	0.000000	91.0.107.134	91.0.107.191	ISAKMP	367 IKE_SA_INIT MID=00 Responder Response
15	0.000000	91.0.107.191	91.0.107.134	ISAKMP	1675 IKE_INTERMEDIATE MID=01 Initiator Request
18	0.000000	91.0.107.134	91.0.107.191	ISAKMP	1675 IKE_INTERMEDIATE MID=01 Responder Response
21	0.000000	91.0.107.191	91.0.107.134	ISAKMP	1675 IKE_INTERMEDIATE MID=02 Initiator Request
26	0.000000	91.0.107.134	91.0.107.191	ISAKMP	1675 IKE_INTERMEDIATE MID=02 Responder Response
29	0.000000	91.0.107.191	91.0.107.134	ISAKMP	1675 IKE_INTERMEDIATE MID=03 Initiator Request
34	0.000000	91.0.107.134	91.0.107.191	ISAKMP	1675 IKE_INTERMEDIATE MID=03 Responder Response
37	1.000000	91.0.107.191	91.0.107.134	ISAKMP	1675 IKE_INTERMEDIATE MID=04 Initiator Request
42	1.000000	91.0.107.134	91.0.107.191	ISAKMP	1675 IKE_INTERMEDIATE MID=04 Responder Response
45	1.000000	91.0.107.191	91.0.107.134	ISAKMP	1675 IKE_INTERMEDIATE MID=05 Initiator Request
50	1.000000	91.0.107.134	91.0.107.191	ISAKMP	1675 IKE_INTERMEDIATE MID=05 Responder Response
55	1.000000	91.0.107.191	91.0.107.134	ISAKMP	1675 IKE_INTERMEDIATE MID=06 Initiator Request
58	1.000000	91.0.107.134	91.0.107.191	ISAKMP	1675 IKE_INTERMEDIATE MID=06 Responder Response
61	1.000000	91.0.107.191	91.0.107.134	ISAKMP	1675 IKE_INTERMEDIATE MID=07 Initiator Request
66	1.000000	91.0.107.134	91.0.107.191	ISAKMP	1675 IKE_INTERMEDIATE MID=07 Responder Response
75	1.000000	91.0.107.191	91.0.107.134	ISAKMP	2270 IKE_AUTH MID=08 Initiator Request
79	1.000000	91.0.107.134	91.0.107.191	ISAKMP	2129 IKE_AUTH MID=08 Responder Response
81	1.000000	91.0.107.134	91.0.107.191	ISAKMP	99 INFORMATIONAL MID=00 Responder Request
83	1.000000	91.0.107.191	91.0.107.134	ISAKMP	99 INFORMATIONAL MID=09 Initiator Request
86	1.000000	91.0.107.134	91.0.107.191	ISAKMP	99 INFORMATIONAL MID=09 Responder Response
87	1.000000	91.0.107.191	91.0.107.134	ISAKMP	99 INFORMATIONAL MID=00 Initiator Response

Figure 6.2.4-x1 System log before/after hybrid key negotiation (see Note)

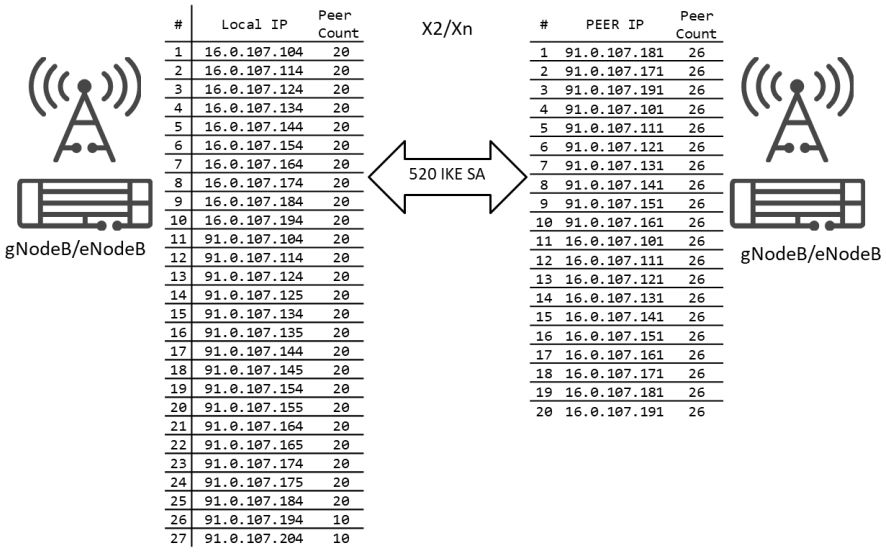
According to the previous implementation inventory analysis, generally only one ADDKE is required, in the case of post-quantum algorithms such as ML-KEM. Support for multiple ADDKEs is not anticipated for implementation soon. The reason to consider in total of a 7 ADDKEs complex case, is that maximum of 7 ADDKEs is required to support as specified in [12] (e.g., session 2.2.1 in [12]). It is also can be considered as a 'extreme' benchmark performance since usually the performance can be better than the result since one ADDKE case is easier for implementation. It is worth noting, In the general scenario, one ADDKE can fit the regular situation.

Note: The flow in Fig 6.2.4-x1 does not describe any fragmentation in the IKE protocol.Therefore, for the simulation of the above scenario, the protocol supports a maximum of totally 7 ADDKE expansion slots. A test is implemented with the most complex ML-KEM-1024 algorithm listed in the **Table 6.1.2-Y**. In this scenario, a total of seven IKE_INTERMEDIATE extra exchanges (double sided) is introduced and tested after the procedure of IKE_INIT. Based on the statistic of packets in the simulation environments, the total size is 23,450 (1,675 x 7 x 2) bytes including headers and other security related payload specified in RFC. The **Figure 6.2.4-x1** shows the log of the demo1, which the newly added IKE_INTERMEDIATE is implemented between IKE_SA_INIT and IKE_AUTH.

The performance impact in a single SA procedure is evaluated with seven IKE_INTERMEDIATE extra exchanges. After the PQC hybrid key negotiation is enabled, the IKE SA establishment time increases from 140 ms to 335 ms (on average). Seven simulated additional ML-KEM-1024 algorithms increase the delay by about 195 ms..

6.2.4.3 Deployed Test Description

The maximum number of IKE SAs in the test is set to 520. Again, we consider the worst case of testing using the maximum number of links.



#	Local IP	Peer Count
1	16.0.107.104	20
2	16.0.107.114	20
3	16.0.107.124	20
4	16.0.107.134	20
5	16.0.107.144	20
6	16.0.107.154	20
7	16.0.107.164	20
8	16.0.107.174	20
9	16.0.107.184	20
10	16.0.107.194	20
11	91.0.107.104	20
12	91.0.107.114	20
13	91.0.107.124	20
14	91.0.107.125	20
15	91.0.107.134	20
16	91.0.107.135	20
17	91.0.107.144	20
18	91.0.107.145	20
19	91.0.107.154	20
20	91.0.107.155	20
21	91.0.107.164	20
22	91.0.107.165	20
23	91.0.107.174	20
24	91.0.107.175	20
25	91.0.107.184	20
26	91.0.107.194	10
27	91.0.107.204	10

#	PEER IP	Peer Count
1	91.0.107.181	26
2	91.0.107.171	26
3	91.0.107.191	26
4	91.0.107.101	26
5	91.0.107.111	26
6	91.0.107.121	26
7	91.0.107.131	26
8	91.0.107.141	26
9	91.0.107.151	26
10	91.0.107.161	26
11	16.0.107.101	26
12	16.0.107.111	26
13	16.0.107.121	26
14	16.0.107.131	26
15	16.0.107.141	26
16	16.0.107.151	26
17	16.0.107.161	26
18	16.0.107.171	26
19	16.0.107.181	26
20	16.0.107.191	26

Figure 6.2.4-x2 IKE SA likes and gNodeB/eNodeB settings

As shown in **Figure 6.2.4-x2**, for this configuration, 27 IP addresses are configured for gNodeB/eNodeB-1 and 20 IP addresses are configured for gNodeB/eNodeB-2. This ensures that a maximum of 520 IKE SAs can be established between the two gNodeB/eNodeB

nodes. Plese note that 520 links are the maximum connections that the current test environment supports. When setting 520 connections, it takes 269 seconds to enable hybrid key negotiation, compared with 255 seconds in the case that this is disabled (on average), as shown in Table 6.2.4-y1. this is disabled (on average), as shown in Table 6.2.4-y1.

There is an interval for setting up an IKE SA link is about 1000 ms. Therefore, the delay does not increase significantly as long as the delay of key hybrid negotiation is not close to 1000 ms. Based on the test results, we conclude that the IKE SA establishment time for X2/Xn is not significantly affected after hybrid key negotiation is enabled.

Table 6.2.4-y1 Test result HY-KE of 520 IKE SA links

Status	Number of IKE SA	Time(s)
Before enabling hybrid key negotiation	520	255
After enabling hybrid key negotiation	520	269

6.2.5 Verification conclusion

In this verification, ML-KEM-1024 and 7 ADDKEs are tested for maximum support and capabilities. In general cases, other algorithms such as ML-KEM-768 with 1 ADDKE may be recommended for general scenarios. When PQC-based hybrid key negotiation is enabled, the time for establishing a single IKE SA is increased from 140 ms to 335 ms (on average), and the number of packet exchanges and bytes increases due to the newly introduced IKE_INTERMEDIATE procedure. In product testing, IKE SA establishment is often speed-limited with time intervals. Therefore, when there are 520 IKE SA links in the test environment, the time increase does not change significantly, that is only from 255 seconds to 269 seconds

In the current test, the performance of the SUT (system under test) is measured and may not reflect the performance aspects of the algorithm itself. The measured outcome has limitations because it can be impacted by SUT specific configuration or settings.

Void

7 Future consideration

Multi-group testing: More detailed multi-group testing can take into consideration different connection numbers (e.g., 10, 100, 200, 300, 400 and 520). For all these test cases it would be good to get the time that is needed to set up the connections.

Standards: PQC standards remain in ongoing development and improvement. Although NIST has announced the first batch of standardized algorithms, within the scope of use cases, the adaptation of relevant technical details and application scenarios is still being continuously optimized. This creates uncertainty for PQC based protocols and applications, necessitating close monitoring of standard updates.

National Guidelines: The prevalence of security gateways and base stations varies between countries, and in addition, some countries are actively developing sovereign PQC algorithms.

National guidelines should be consulted to ensure that any recommendations or regulations are followed appropriately.

Vendors: Vendors play a crucial role in pre-standard protocol implementation and preliminary proof-of-concept testing for algorithms, providing foundational data for standards-based solutions. As standards mature, vendors need to invest in software and hardware upgrades. These upgrades ensures their equipment can smoothly support PQC algorithms and protocol extensions.

3rd-Parties: Typically, use cases involve communication between different entities. There should be active collaboration with 3rd-Parties. This ensures that 3rd-Parties components and services in the network can support quantum secure protocols. It also encourages their support and integration of PQC.

Performance: Performance tests in this document reflect extreme configurations (7 ADDKEs, ML-KEM-1024), Operators, vendors, and 3rd-Parties should work together to validate with realistic parameters (e.g., 1 ADDKE, ML-KEM-768) for production planning. This quantifies the impact of PQC algorithms on network performance. It's essential to balance security and performance needs. Select suitable PQC algorithms and implementation plans based on specific network requirements.

Hardware performance: The computational complexity of ML-KEM increases CPU load. But the specific value hasn’t been quantified. When deploying, a comprehensive evaluation of the hardware capacity of base stations is needed to ensure it meets the computational requirements of PQC hybrid key exchange. This involves upgrading or optimizing existing base station hardware to enhance computational and processing capabilities.

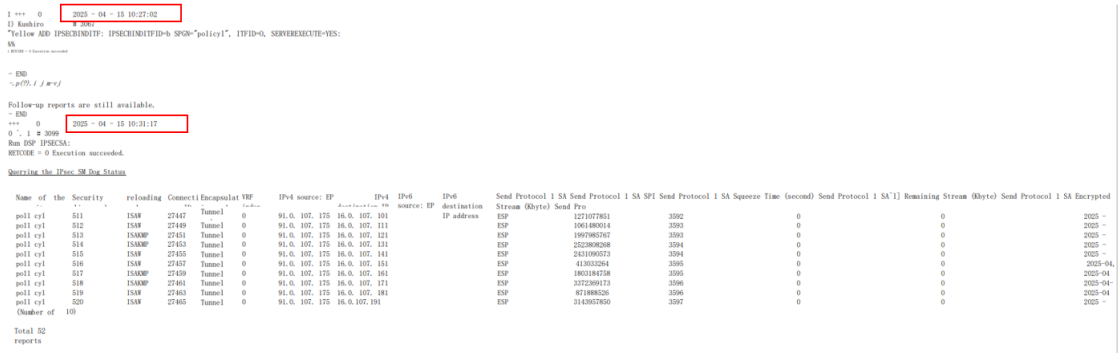
Legacy Impact: Under this scheme, legacy equipment may ignore ADDKE and IKE_INTERMEDIATE messages, responding only to traditional DH group parameters, thereby falling back to pure classical mode negotiation. New equipment can fully execute the hybrid key exchange process. For intermediate network devices, [9] fragmentation avoids IP-layer modifications, and [11] new message types avoid interception, ensuring a smooth transition and compatibility. However, migration strategies (e.g., via software/hardware feature upgrades or procurement of new equipment) still require careful consideration.

Annex A Related standardization documents

A.1 System Log

System benchmark before/after HY-KE of 520 IKE SA links:

Before enabling hybrid key negotiation of 520 links of IKE SAs



After enabling hybrid key negotiation of 520 links of IKE SAs

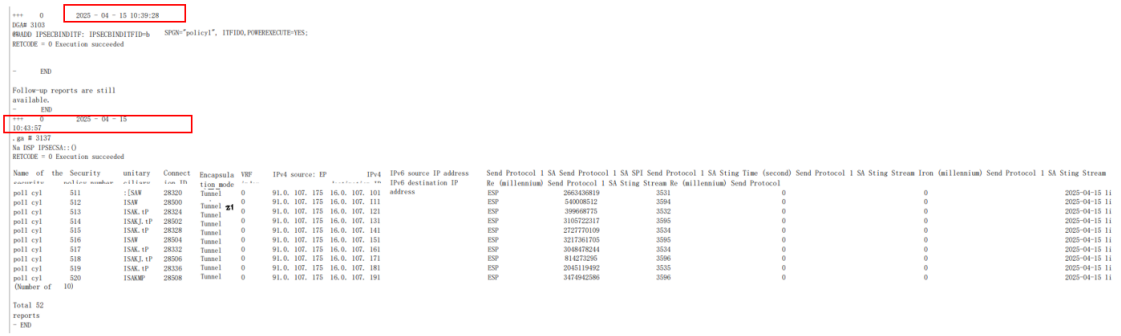


Figure A.4-1 System benchmark before/after HY-KE of 520 IKE SA links

Annex B Document Management

B.1 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	27 May 2026	Skeleton and First draft	PQTN TF/ TG	Z.A.Lozinski, IBM

B.2 Other Information

Type	Description
Document Owner	PQTN TF
Editor / Company	Z.A.Lozinski, IBM

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at prd@gsma.com

Your comments or suggestions & questions are always welcome.