Configuring Quantum-Safe Encryption Using Postquantum Preshared Keys

This module describes quantum-safe encryption using Postquantum Preshared Keys (PPK). This feature implements RFC 8784 and Cisco Secure Key Integration Protocol (SKIP) for quantum-safe encryption of IKEv2 and IPsec packets using PPKs.

Restrictions for Quantum-Safe Encryption Using Postquantum Preshared Keys

- The Quantum-Safe Encryption Using Postquantum Preshared Keys feature is applicable to all IKEv2 and IPsec VPNs such as, FlexVPN (SVTI-DVTI) and DMVPN, except for GETVPN.

Supported Platforms

The Quantum-Safe Encryption Using Postquantum Preshared Keys feature is available on the following platforms:

From Cisco IOS XE Release 17.12.1a	From Cisco IOS XE Release 17.11.1a
Cisco 1000 Series Integrated Services Routers	Cisco Catalyst 8000V Edge Software
Cisco Catalyst 8500 Series Edge Platforms	Cisco Catalyst 8300 Series Edge Platforms
	Cisco ASR 1000 Series Aggregation Services Routers

Information About Quantum-Safe Encryption Using Postquantum Preshared Keys

The following sections provide detailed information relating to the Quantum-Safe Encryption Using Postquantum Preshared Keys feature.

Impact of Quantum Computers on Cryptography

Quantum computers pose a serious challenge to the cryptographic algorithms and protocols deployed widely today. A quantum computer can solve Diffie-Hellman (DH) and Elliptic Curve Diffie-Hellman (ECDH) problems in polynomial time, and this can compromise the security of existing IKEv2 systems. A man-in-the-middle storing the VPN communications today can decrypt them later when a quantum computer is available.

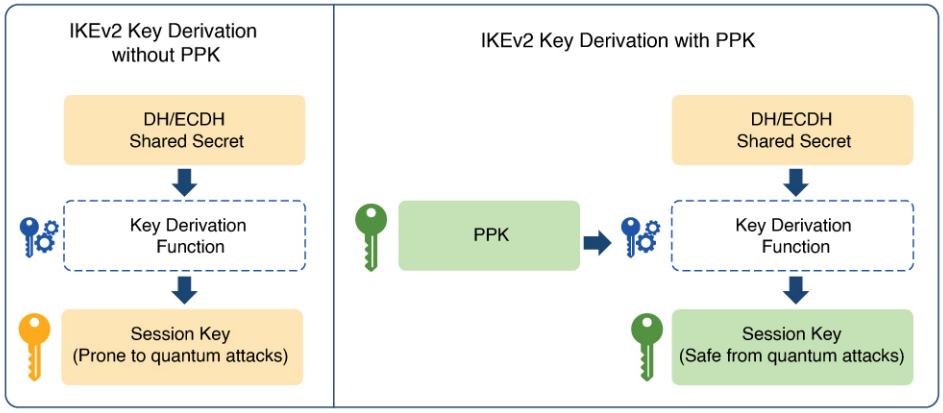
Postquantum Preshared Keys

Session keys that are based on preshared keys are not vulnerable to quantum attacks if the preshared keys have sufficient entropy and the pseudorandom function (PRF), encryption, and authentication transformations are quantum secure. The resulting system is then believed to be secure against classical attackers of today or future attackers with a quantum computer.

RFC 8784 (Mixing Preshared Keys in IKEv2 for Postquantum Security) describes an extension to the IKEv2 protocol to allow it to be resistant to a quantum computer by using preshared keys known as PPKs. The RFC defines negotiation of PPK capability, communication of PPK ID, mixing of PPK as an additional input in the session key derivation, and optional fallback to non-PPK-based session.

Figure 1 shows the IKEv2 key derivation with and without PPK.

Figure 1: IKEv2 Key Derivation - With and Without PPK



DH: Diffie-Hellman
ECDH: Elliptic-curve Diffie-Hellman
PPK: Postquantum Preshared Key

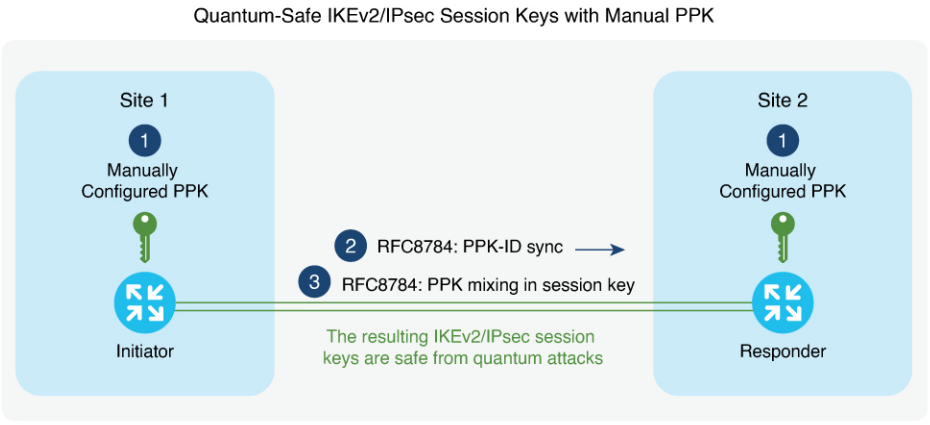
Manual Postquantum Preshared Keys

The simplest provisioning mechanism to provide the same PPKs on the IKEv2 and IPsec initiator and responder pair is to manually configure the PPKs on both sides. The PPKs configured manually are known as manual PPKs.

With a manual PPK, the administrator must ensure that the PPK is of sufficient size and entropy and it is rotated frequently.

Figure 2 shows quantum-safe IKEv2 and IPsec session keys with a manual PPK.

Figure 2: Quantum-Safe IKEv2 and IPsec Session Keys with Manual PPK



Cisco Secure Key Integration Protocol and Dynamic Postquantum Preshared Keys

Cisco SKIP is an HTTPS-based protocol that allows encryption devices such as routers, to import PPKs from an external key source. The externally imported PPKs known as dynamic PPKs offer the benefits of automated provisioning and refresh and better entropy of PPKs.

Cisco SKIP uses TLS1.2 with PSK-DHE cipher suite to make the SKIP protocol quantum-safe. The encryption devices must implement the SKIP client and the external key source must implement the SKIP server.

For an external key source to be SKIP compliant, it must implement the Cisco SKIP protocol and must use an out-of-band synchronization mechanism to provide the same PPK to the two encryption devices—initiator and responder. The external key source can be a Quantum Key Distribution (QKD) device, software, or cloud-based key source or service.

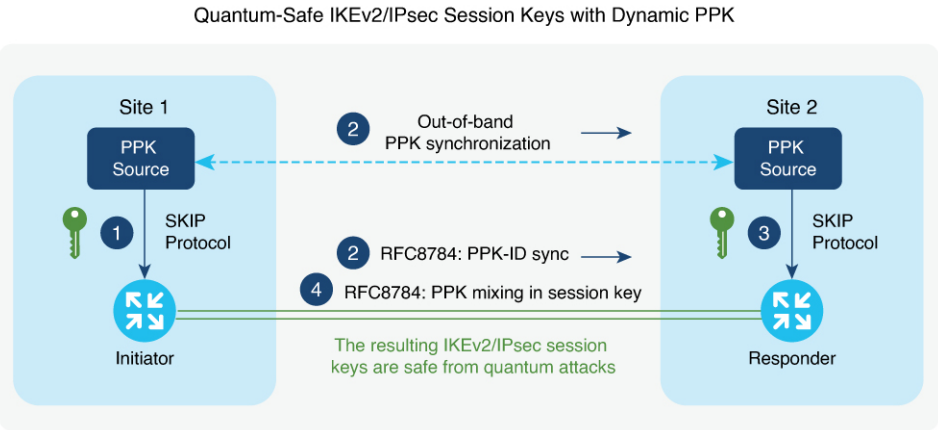
The external key source must meet the following expectations to be SKIP-compliant:

- Must implement SKIP protocol or API, as defined in the Cisco SKIP specification.
- Must provide the same PPK to the encryption device pair—initiator and responder—using an out-of-band synchronization mechanism.

Note Key source vendors, such as QKD vendors, should contact their Cisco representative to implement the Cisco SKIP protocol.

Figure 3 shows quantum-safe IKEv2 and IPsec session keys using dynamic PPK.

Figure 3: Quantum-Safe IKEv2 and IPsec Session Keys with Dynamic PPK



The IKEv2 initiator and responder are connected to their local key source and configured with the SKIP client that specifies the IP address and port of the key source and the preshared key for the TLS1.2 session. The PPK sources are configured with the SKIP parameters, including the local key source identity and the list of identities of the peer key sources.

The following is a high-level operation of the Cisco SKIP protocol:

1. The IKEv2 initiator places a request for a PPK from its key source. The key source replies with a PPK and the corresponding PPK ID.
2. The initiator-side key source synchronizes the PPK to the responder-side key source using an out-of-band mechanism that is specific to the type of key source. The IKEv2 initiator communicates the PPK ID to the IKEv2 responder over IKEv2 using the RFC 8784 extensions.
3. The IKEv2 responder requests from its key source, the PPK corresponding to the PPK ID received from the IKEv2 initiator. The key source replies with the PPK corresponding to the PPK ID.
4. The IKEv2 initiator and responder mix the PPK in the key derivation, as specified in RFC 8784. The resulting IKEv2 and IPsec session keys are quantum-safe.

How to Configure Quantum-Safe Encryption Using Postquantum Preshared Keys

The following sections describe the processes involved in configuring quantum-safe encryption using postquantum preshared keys.

Configuring Manual Postquantum Preshared Keys

Perform the following tasks to configure the manual PPK.

Configuring Manual Postquantum Preshared Keys in an IKEv2 Keyring

Follow these steps to configure the manual PPK for one or more peers or groups of peers, in the IKEv2 keyring.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 keyring** *keyring-name*
4. **peer** *name*
5. Run one of the following commands:
   - **address** {*ipv4-address mask* | *ipv6-address prefix*}
   - **identity** {**address** {*ipv4-address* | *ipv6-address*} | **fqdn domain** *domain-name* | **email domain** *domain-name* | **key-id** *key-id*}
6. **ppk manual id** *ppk-id* **key** [**0** | **6** | **hex**] *password* [**required**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto ikev2 keyring** *keyring-name* **Example:** `Device(config)# crypto ikev2 keyring keyring1`	Defines an IKEv2 keyring and enters IKEv2 keyring configuration mode.
Step 4	**peer** *name* **Example:** `Device(config-ikev2-keyring)# peer peer1`	Defines the peer or peer group and enters IKEv2 keyring peer configuration mode.
Step 5	Run one of the following commands: • **address** {*ipv4-address mask* | *ipv6-address prefix*} • **identity** {**address** {*ipv4-address* | *ipv6-address*} | **fqdn domain** *domain-name* | **email domain** *domain-name* | **key-id** *key-id*} **Example:**	Specifies the remote IKEv2 peers based on WAN IP address or IKEv2 identity. • The **address** command specifies an IPv4 or IPv6 address or range for the peer or group of peers. **Note** This IP address is the IKE endpoint address and is independent of the identity address.

	Command or Action	Purpose

	`Device(config-ikev2-keyring-peer)# address 10.0.0.1 255.0.0.0`
	Example:
	`Device(config-ikev2-keyring-peer)# identity address 10.0.0.1`
		• The **identity** command identifies the IKEv2 peer through the following identities:
		• E-mail
		• Fully qualified domain name (FQDN)
		• IPv4 or IPv6 address
		• Key ID
		Note The **identity** command is available for key lookup only on the IKEv2 responder.

Step 6	**ppk manual id** *ppk-id* **key** [**0** | **6** | **hex**] *password* [**required**] **Example:** `Device(config-ikev2-keyring-peer)# ppk manual id ppk_id key cisco123`	Configures PPK ID and PPK for the identified peers. • **ppk manual**: Indicates that the PPK ID and the PPK are configured manually. • **id** *ppk-id*: Specifies the PPK ID. • **key** *password*: Specifies the PPK. • **required**: Indicates that the quantum-safe encryption using PPK is mandatory and there must be no fallback to a normal IKEv2 or IPsec session. **Note** The *ppk-id* and the PPK must match on both the peers.

Configuring an IKEv2 Keyring in an IKEv2 Profile

SUMMARY STEPS

1. **crypto ikev2 profile** *profile-name*
2. **keyring ppk** *keyring-name*
3. **exit**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**crypto ikev2 profile** *profile-name* **Example:** `Device(config-ikev2-keyring-peer)# crypto ikev2 profile profile1`	Defines an IKEv2 profile and enters IKEv2 profile configuration mode.
Step 2	**keyring ppk** *keyring-name* **Example:**	Specifies the keyring that has either manual or dynamic PPK configured.

	Command or Action	Purpose
	`Device(config-ikev2-profile)# keyring ppk keyring1`	**Note** To remove the keyring from the IKEv2 profile, use the **no keyring** {**aaa** | **local** | **ppk**} *keyring-name* command.
Step 3	**exit** **Example:** `Device(config-ikev2-profile)# exit`	Exits IKEv2 profile configuration mode and returns to global configuration mode.
Step 4	**exit** **Example:** `Device(config)# exit`	Exits global configuration mode and enters privileged EXEC mode.

Configuring Dynamic Postquantum Preshared Keys

Perform the following tasks to configure the dynamic PPK.

Configuring a Secure Key Integration Protocol Client

SKIP client configuration specifies the parameters required to securely communicate with and request PPKs from an external SKIP-compliant key source.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto skip-client** *skip-client-name*
4. **server** {**ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **fqdn** *domain-name*} **port** *port-number*
5. **psk id** *id-name* **key** [**0** | **6** | **hex**] *password*
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto skip-client** *skip-client-name* **Example:** `Device(config-crypto-skip-client)# crypto skip-client skip-client-cfg`	Specifies the name of SKIP client configuration block and enters SKIP client configuration mode.

	Command or Action	Purpose
Step 4	**server** {**ipv4** *ipv4-address* | **ipv6** *ipv6-address* | **fqdn** *domain-name*} **port** *port-number* **Example:** `Device(config-crypto-skip-client)# server ipv4 10.10.0.3 port 9993`	Specifies the IP address or FQDN and port to connect to the external key source.
Step 5	**psk id** *id-name* **key** [**0** | **6** | **hex**] *password* **Example:** `Device(config-crypto-skip-client)# psk id psk-id key 0 cisco123`	Specifies the preshared key identity and the preshared key for the SKIP TLS session.
Step 6	**exit** **Example:** `Device(config-crypto-skip-client)# exit`	Exits SKIP client configuration mode and returns to global configuration mode.

Configuring a Secure Key Integration Protocol Client in an IKEv2 Keyring

Follow these steps to configure the manual PPK for one or more peers or groups of peers in the IKEv2 keyring.

SUMMARY STEPS

1. **crypto ikev2 keyring** *keyring-name*
2. **peer** *name*
3. Execute one of the following commands:
   - **address** {*ipv4-address mask* | *ipv6-address prefix*}
   - **identity** {**address** {*ipv4-address* | *ipv6-address*} | **fqdn domain** *domain-name* | **email domain** *domain-name* | **key-id** *key-id*}
4. **ppk dynamic** *skip-client-name* [**required**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**crypto ikev2 keyring** *keyring-name* **Example:** `Device(config)# crypto ikev2 keyring keyring1`	Defines an IKEv2 keyring and enters IKEv2 keyring configuration mode.
Step 2	**peer** *name* **Example:** `Device(config-ikev2-keyring)# peer peer1`	Defines the peer or peer group and enters IKEv2 keyring peer configuration mode.
Step 3	Execute one of the following commands: • **address** {*ipv4-address mask* | *ipv6-address prefix*} • **identity** {**address** {*ipv4-address* | *ipv6-address*} | **fqdn domain** *domain-name* | **email domain** *domain-name* | **key-id** *key-id*}	Specifies the remote IKEv2 peers based on WAN IP address or IKEv2 identity. The **address** command specifies an IPv4 or IPv6 address or range for the peer or group of peers.

	Command or Action	Purpose

	Example:
	`Device(config-ikev2-keyring-peer)# address 10.0.0.1 255.0.0.0`
	Example:
	`Device(config-ikev2-keyring-peer)# identity address 10.0.0.1`
		Note This IP address is the IKE endpoint address and is independent of the identity address.
		The **identity** command identifies the IKEv2 peer through the following identities:
		• E-mail
		• Fully qualified domain name (FQDN)
		• IPv4 or IPv6 address
		• Key ID
		Note The **identity** command is available for key lookup only on the IKEv2 responder.

Step 4	**ppk dynamic** *skip-client-name* [**required**] **Example:** `Device(config-ikev2-keyring-peer)# ppk dynamic skip-client1`	Specifies the external key source to use for dynamic PPKs. • **ppk dynamic**: Indicates that the PPK is imported dynamically from an external key source. • **required**: Indicates that the quantum-safe encryption using PPK is mandatory and there must be no fallback to a normal IKEv2 or IPsec session.

Configuring an IKEv2 Keyring in an IKEv2 Profile

SUMMARY STEPS

1. **crypto ikev2 profile** *profile-name*
2. **keyring ppk** *keyring-name*
3. **exit**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**crypto ikev2 profile** *profile-name* **Example:** `Device(config-ikev2-keyring-peer)# crypto ikev2 profile profile1`	Defines an IKEv2 profile and enters IKEv2 profile configuration mode.
Step 2	**keyring ppk** *keyring-name* **Example:** `Device(config-ikev2-profile)# keyring ppk keyring1`	Specify the keyring that has either manual or dynamic PPK configured. **Note** To remove the keyring from the IKEv2 profile, use the **no keyring** {**aaa** | **local** | **ppk**} *keyring-name* command.

	Command or Action	Purpose
Step 3	**exit** **Example:** `Device(config-ikev2-profile)# exit`	Exits IKEv2profile configuration mode and returns to global configuration mode.
Step 4	**exit** **Example:** `Device(config)# exit`	Exits global configuration mode and enters privileged EXEC mode.

Configuration Examples for Quantum-Safe Encryption Using Postquantum Preshared Keys

The following sections provide detailed configuration examples relating to the configuration of quantum-safe encryption using PPKs.

Example: Configuring the Manual Postquantum Preshared Keys

Example: Initiator Configuration

The following example shows how to manually configure a PPK for an initiator:

```
conf t
hostname Router1
!
crypto ikev2 keyring ppk-keyring
peer 1
address 10.10.0.1 255.255.255.0
ppk manual id ppk_id key cisco123
!
crypto ikev2 profile prof
match identity remote address 10.10.0.1
authentication local pre-share key cisco
authentication remote pre-share key cisco
keyring ppk ppk-keyring
!
crypto ipsec profile prof
set ikev2-profile prof
!
interface Tunnel0
ip address 10.10.0.1 255.255.255.0
tunnel source GigabitEthernet1
tunnel destination 10.10.10.1
tunnel protection ipsec profile prof
!
interface GigabitEthernet1
ip address 10.10.10.2 255.255.255.0
no shut
!
```

Example: Responder Configuration

The following example shows how to manually configure a PPK for a responder:

```
conf t
hostname Router2
!
crypto ikev2 keyring ppk-keyring
peer 1
address 10.10.0.1 255.255.255.0
ppk manual id ppk_id key cisco
!
crypto ikev2 profile prof
match identity remote address 10.10.0.1
authentication local pre-share key cisco
authentication remote pre-share key cisco
keyring ppk ppk-keyring
!
crypto ipsec profile prof
set ikev2-profile prof
!
interface Tunnel0
ip address 10.10.0.2 255.255.255.0
tunnel source GigabitEthernet1
tunnel destination 10.10.10.2
tunnel protection ipsec profile prof
!
interface GigabitEthernet1
ip address 10.10.0.1 255.255.255.0
no shut
!
```

Example: Configuring the Dynamic Postquantum Preshared Keys

Example: Initiator Configuration

The following example shows how to configure a dynamic PPK for an initiator:

```
conf t
hostname Router1
!
crypto skip-client skip-client-cfg
server ipv4 10.10.0.4  port 9991
psk id psk-id1 key 0 cisco123
!
crypto ikev2 keyring ppk-keyring
peer 1
address 10.10.0.1 255.255.255.0
ppk dynamic skip-client-cfg
!
crypto ikev2 profile prof
match identity remote address 10.10.0.1
authentication local pre-share key cisco
authentication remote pre-share key cisco
keyring ppk ppk-keyring
!
crypto ipsec profile prof
set ikev2-profile prof
!
interface Tunnel0
ip address 10.10.0.2 255.255.255.0
```

```
tunnel source GigabitEthernet1
tunnel destination 10.10.10.1
tunnel protection ipsec profile prof
!
interface GigabitEthernet1
ip address 10.10.10.2 255.255.255.0
no shut
!
interface GigabitEthernet1
ip address 10.10.10.3 255.255.255.0
no shut
!
```

Example: Responder Configuration

The following example shows how to configure a dynamic PPK for a responder:

```
conf t
hostname Router2
!
crypto skip-client skip-client-cfg
server ipv4 10.10.0.4 port 9992
psk id vedge-sim-1 key 0 cisco123
!
crypto ikev2 keyring ppk-keyring
peer 1
address 10.10.0.1 255.255.255.0
ppk dynamic skip-client-cfg
!
crypto ikev2 profile prof
match identity remote address 10.10.0.1
authentication local pre-share key cisco
authentication remote pre-share key cisco
keyring ppk ppk-keyring
!
crypto ipsec profile prof
set ikev2-profile prof
!
interface Tunnel0
ip address 10.10.0.2 255.255.255.0
tunnel source GigabitEthernet1
tunnel destination 10.10.10.2
tunnel protection ipsec profile prof
!
interface GigabitEthernet1
ip address 10.10.10.1 255.255.255.0
no shut
!
interface GigabitEthernet1
ip address 10.10.10.4 255.255.255.0
!
```

Verifying the Postquantum Preshared Keys Configuration

Use the **show crypto ikev2 sa detailed** command to display information about the current IKEv2 security associations. The `Quantum Resistance Enabled` message displayed in the output indicates that PPK-based quantum-safe encryption is enabled.

The following is a sample output from the **show crypto ikev2 sa detailed** command:

```
IPv4 Crypto IKEv2  SA
Tunnel-id       Local               Remote                  fvrf/ivrf          Status

   3        <src IP>/SrcPort     <Dst IP>/DstPort             none/none        READY

Encr: AES-CBC, keysize: 256, PRF: SHA512, Hash: SHA512, DH Grp:19,
Auth sign:
.
.
.
Initiator of SA : No
Quantum Resistance Enabled
```

Additional References for Quantum-Safe Encryption Using Postquantum Preshared Keys

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	• Cisco IOS Security Command Reference Commands A to C • Cisco IOS Security Command Reference Commands D to L • Cisco IOS Security Command Reference Commands M to R • Cisco IOS Security Command Reference Commands S to Z
IPsec configuration	Configuring Security for VPNs with IPsec

RFCs

RFC	Title
RFC 8784	*Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Postquantum Security*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Quantum-Safe Encryption Using Postquantum Preshared Keys

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Quantum-Safe Encryption Using Postquantum Preshared Keys

Feature Name	Releases	Feature Information
Quantum-Safe Encryption Using Postquantum Preshared Keys	Cisco IOS XE Release 17.11.1a	The feature implements RFC 8784 and Cisco Secure Key Integration Protocol (SKIP) for quantum-safe encryption of IKEv2 and IPsec packets using Postquantum Preshared Keys (PPKs). The PPKs that are configured manually are known as manual PPKs, and the PPKs that are imported from an external key source using the SKIP protocol are known as dynamic PPKs.
Quantum-Safe Encryption Using Postquantum Preshared Keys	Cisco IOS XE Release 17.12.1a	This enhancement introduces support for Quantum-Safe Encryption Using Postquantum Preshared Keys for the following platforms: • Cisco 1000 Series Integrated Services Routers • Cisco Catalyst 8500 Series Edge Platforms