FlexVPN and Internet Key Exchange Version 2 Configuration Guide, Cisco IOS Release 15S

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

Introduction to FlexVPN

Internet Key Exchange Version (IKEv2), a next-generation key management protocol based on RFC 4306, is an enhancement of the IKE Protocol. IKEv2 is used for performing mutual authentication and establishing and maintaining security associations (SAs).

FlexVPN is Cisco's implementation of the IKEv2 standard featuring a unified paradigm and CLI that combines site to site, remote access, hub and spoke topologies and partial meshes (spoke to spoke direct). FlexVPN offers a simple but modular framework that extensively uses the tunnel interface paradigm while remaining compatible with legacy VPN implementations using crypto maps.

This configuration guide contains the following modules:

Configuring Internet Key Exchange Version 2 (IKEv2) and FlexVPN Site-to-Site

This module describes IKEv2 CLI required for FlexVPN site-to-site. The module is divided into basic and advanced sections.

The basic section introduces basic IKEv2 commands and describes IKEv2 smart defaults and the mandatory IKEv2 commands required for FlexVPN site-to-site. This module is a prerequisite for understanding subsequent chapters.

The advanced section describes global IKEv2 commands and how to override the default IKEv2 commands.

Appendix: IKEv2 and Legacy VPNs

This module contains configuration examples on how to configure legacy VPNs such as crypto maps and DMVPN with Internet Key Exchange Version 2 (IKEv2).

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Configuring Internet Key Exchange Version 2 and FlexVPN Site-to-Site

This module contains information about and instructions for configuring basic and advanced Internet Key Exchange Version 2 (IKEv2) and FlexVPN site-to-site. The tasks and configuration examples for IKEv2 in this module are divided as follows:

- Basic IKEv2—Provides information about basic IKEv2 commands, IKEv2 smart defaults, basic IKEv2 profile, and IKEv2 key ring.
- Advanced IKEv2—Provides information about global IKEv2 commands and how to override IKEv2 smart defaults.

Note Security threats, as well as the cryptographic technologies to help protect against them, are constantly changing. For more information about the latest Cisco cryptographic recommendations, see the *Next Generation Encryption* (NGE) white paper.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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.

Prerequisites for Configuring Internet Key Exchange Version 2

You should be familiar with the concepts and tasks described in the "Configuring Security for VPNs with IPsec" module.

Restrictions for Configuring Internet Key Exchange Version 2

You cannot configure an option that is not supported on a specific platform. For example, in a security protocol, the capability of the hardware-crypto engine is important, and you cannot specify the Triple Data Encryption Standard (3DES) or the Advanced Encryption Standard (AES) type of encryption transform in a nonexportable image, or specify an encryption algorithm that a crypto engine does not support.

Information About Internet Key Exchange Version 2

IKEv2 Supported Standards

Cisco implements the IP Security (IPsec) Protocol standard for use in Internet Key Exchange Version 2 (IKEv2).

Note Cisco no longer recommends using DES or MD5 (including HMAC variant); instead, you should use AES and SHA-256. For more information about the latest Cisco cryptographic recommendations, see the *Next Generation Encryption* (NGE) white paper.

The component technologies implemented in IKEv2 are as follows:

- AES-CBC—Advanced Encryption Standard-Cipher Block Chaining
- SHA (HMAC variant)—Secure Hash Algorithm
- Diffie-Hellman—A public-key cryptography protocol
- DES—Data Encryption Standard (No longer recommended)
- MD5 (HMAC [Hash-based Message Authentication Code] variant)—Message digest algorithm 5 (No longer recommended)

For more information about supported standards and component technologies, see the "Supported Standards for Use with IKE" section in the "Configuring Internet Key Exchange for IPsec VPNs" module in the *Internet Key Exchange for IPsec VPNs Configuration Guide*.

Benefits of IKEv2

Dead Peer Detection and Network Address Translation-Traversal

Internet Key Exchange Version 2 (IKEv2) provides built-in support for Dead Peer Detection (DPD) and Network Address Translation-Traversal (NAT-T).

Certificate URLs

Certificates can be referenced through a URL and hash, instead of being sent within IKEv2 packets, to avoid fragmentation.

Denial of Service Attack Resilience

IKEv2 does not process a request until it determines the requester, which addresses to some extent the Denial of Service (DoS) problems in IKEv1, which can be spoofed into performing substantial cryptographic (expensive) processing from false locations.

EAP Support

IKEv2 allows the use of Extensible Authentication Protocol (EAP) for authentication.

Multiple Crypto Engines

If your network has both IPv4 and IPv6 traffic and you have multiple crypto engines, choose one of the following configuration options:

- One engine handles IPv4 traffic and the other engine handles IPv6 traffic.
- One engine handles both IPv4 and IPv6 traffic.

Reliability and State Management (Windowing)

IKEv2 uses sequence numbers and acknowledgments to provide reliability, and mandates some error-processing logistics and shared state management.

Internet Key Exchange Version 2 CLI Constructs

IKEv2 Proposal

An Internet Key Exchange Version 2 (IKEv2) proposal is a collection of transforms used in the negotiation of Internet Key Exchange (IKE) security associations (SAs) as part of the IKE_SA_INIT exchange. The transform types used in the negotiation are as follows:

- Encryption algorithm
- Integrity algorithm
- Pseudo-Random Function (PRF) algorithm

- Diffie-Hellman (DH) group

See the “IKEv2 Smart Defaults” section for information about the default IKEv2 proposal. See the “Configuring Advanced IKEv2 CLI Constructs” section for information about how to override the default IKEv2 proposal and to define new proposals.

IKEv2 Policy

An IKEv2 policy contains proposals that are used to negotiate the encryption, integrity, PRF algorithms, and DH group in the IKE_SA_INIT exchange. It can have match statements, which are used as selection criteria to select a policy during negotiation.

See the “IKEv2 Smart Defaults” section for information about the default IKEv2 policy. See the “Configuring Advanced IKEv2 CLI Constructs” section for information about how to override the default IKEv2 policy and to define new policies.

IKEv2 Profile

An IKEv2 profile is a repository of nonnegotiable parameters of the IKE SA, such as local or remote identities and authentication methods and services that are available to authenticated peers that match the profile. An IKEv2 profile must be attached to either a crypto map or an IPSec profile on the initiator. An IKEv2 profile is not mandatory on the responder.

IKEv2 Key Ring

An IKEv2 key ring is a repository of symmetric and asymmetric preshared keys and is independent of the IKEv1 key ring. The IKEv2 key ring is associated with an IKEv2 profile and hence supports a set of peers that match the IKEv2 profile. The IKEv2 key ring gets its VPN routing and forwarding (VRF) context from the associated IKEv2 profile.

IKEv2 Smart Defaults

The IKEv2 Smart Defaults feature minimizes the FlexVPN configuration by covering most of the use cases. IKEv2 smart defaults can be customized for specific use cases, though this is not recommended.

See the “Configuring Advanced IKEv2 CLI Constructs” section for information about how to modify the default IKEv2 constructs.

The following rules apply to the IKEv2 Smart Defaults feature:

1. A default configuration is displayed in the corresponding **show** command with **default** as a keyword and with no argument. For example, the **show crypto ikev2 proposal default** command displays the default IKEv2 proposal and the **show crypto ikev2 proposal** command displays the default IKEv2 proposal, along with any user-configured proposals.
2. A default configuration is displayed in the **show running-config all** command; it is not displayed in the **show running-config** command.
3. You can modify the default configuration, which is displayed in the **show running-config all** command.
4. A default configuration can be disabled using the **no** form of the command; for example, **no crypto ikev2 proposal default**. A disabled default configuration is not used in negotiation but the configuration is displayed in the **show running-config** command. A disabled default configuration loses any user modification and restores system-configured values.

5 A default configuration can be reenabled using the default form of the command, which restores system-configured values; for example, **default crypto ikev2 proposal**.
6 The default mode for the default transform set is transport; the default mode for all other transform sets is tunnel.

Note Cisco no longer recommends using MD5 (including HMAC variant) and Diffie-Hellman (DH) groups 1, 2 and 5; instead, you should use SHA-256 and DH Groups 14 or higher. For more information about the latest Cisco cryptographic recommendations, see the *Next Generation Encryption* (NGE) white paper.

The following table lists the commands that are enabled with the IKEv2 Smart Defaults feature, along with the default values.

Table 1 IKEv2 Command Defaults

Command Name	Default Values

crypto ikev2 authorization policy	`Device# show crypto ikev2 authorization policy default`
	
	`IKEv2 Authorization policy: default`
	`route set interface`
	`route accept any tag: 1 distance: 2`

crypto ikev2 proposal	`Device# show crypto ikev2 proposal default`
	
	`IKEv2 proposal: default`
	`Encryption: AES-CBC-256 AES-CBC-192 AES-CBC-128`
	`Integrity: SHA512 SHA384 SHA256 SHA96 MD596`
	`PRF: SHA512 SHA384 SHA256 SHA1 MD5`
	`DH Group: DH_GROUP_1536_MODP/Group 5 DH_GROUP_1024_MODP/Group 2.`

crypto ikev2 policy	`Device# show crypto ikev2 policy default`
	
	`IKEv2 policy: default`
	`Match fvrf: any`
	`Match address local: any`
	`Proposal: default`

crypto ipsec profile	`Device# show crypto ipsec profile default`
	
	`IPSEC profile default`
	`Security association lifetime: 4608000 kilobytes/3600 seconds`
	`Responder-Only (Y/N): N`
	`PFS (Y/N): N`
	`Transform sets={`
	`default: { esp-aes esp-sha-hmac },`
	`}`

crypto ipsec transform-set	`Device# show crypto ipsec transform-set default`
	
	`Transform set default: { esp-aes esp-sha-hmac }`
	`will negotiate = { Tunnel, },`

Note Before you can use the default IPsec profile, explicitly specify the **crypto ipsec profile** command on a tunnel interface using the **tunnel protection ipsec profile default** command.

IKEv2 Suite-B Support

Suite-B is a set of cryptographic algorithms promulgated by the National Security Agency as part of its Cryptographic Modernization Program. Suite-B for Internet Key Exchange (IKE) and IPsec is defined in RFC 4869. The Suite-B components are as follows:

- Advanced Encryption Standard (AES) 128- and 256-bit keys configured in the IKEv2 proposal. For data traffic, AES should be used in Galois Counter Mode (GCM) that is configured in the IPsec transform set.
- Elliptic Curve Digital Signature Algorithm (ECDSA) configured in the IKEv2 profile.
- Secure Hashing Algorithm 2 (SHA-256 and SHA-384) configured in the IKEv2 proposal and IPsec transform set.

Suite-B requirements comprise four user-interface suites of cryptographic algorithms for use with IKE and IPsec. Each suite consists of an encryption algorithm, a digital-signature algorithm, a key-agreement algorithm, and a hash- or message-digest algorithm. See the “Configuring Security for VPNs with IPsec” feature module for detailed information about Cisco Suite-B support.

How to Configure Internet Key Exchange Version 2

Configuring Basic Internet Key Exchange Version 2 CLI Constructs

To enable IKEv2 on a crypto interface, attach an Internet Key Exchange Version 2 (IKEv2) profile to the crypto map or IPsec profile applied to the interface. This step is optional on the IKEv2 responder.

Note The difference between IKEv1 and IKEv2 is that you need not enable IKEv1 on individual interfaces because IKEv1 is enabled globally on all interfaces on a device.

Perform the following tasks to manually configure basic IKEv2 constructs:

Configuring the IKEv2 Key Ring

Perform this task to configure the IKEv2 key ring if the local or remote authentication method is a preshared key.

IKEv2 key ring keys must be configured in the peer configuration submode that defines a peer subblock. An IKEv2 key ring can have multiple peer subblocks. A peer subblock contains a single symmetric or

asymmetric key pair for a peer or peer group identified by any combination of the hostname, identity, and IP address.

IKEv2 key rings are independent of IKEv1 key rings. The key differences are as follows:

- IKEv2 key rings support symmetric and asymmetric preshared keys.
- IKEv2 key rings do not support Rivest, Shamir, and Adleman (RSA) public keys.
- IKEv2 key rings are specified in the IKEv2 profile and are not looked up, unlike IKEv1, where keys are looked up on receipt of MM1 to negotiate the preshared key authentication method. The authentication method is not negotiated in IKEv2.
- IKEv2 key rings are not associated with VPN routing and forwarding (VRF) during configuration. The VRF of an IKEv2 key ring is the VRF of the IKEv2 profile that refers to the key ring.
- A single key ring can be specified in an IKEv2 profile, unlike an IKEv1 profile, which can specify multiple key rings.
- A single key ring can be specified in more than one IKEv2 profile, if the same keys are shared across peers matching different profiles.
- An IKEv2 key ring is structured as one or more peer subblocks.

On an IKEv2 initiator, the IKEv2 key ring key lookup is performed using the peer's hostname or the address, in that order. On an IKEv2 responder, the key lookup is performed using the peer's IKEv2 identity or the address, in that order.

Note You cannot configure the same identity in more than one peer.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 keyring** *keyring-name*
4. **peer** *name*
5. **description** *line-of-description*
6. **hostname** *name*
7. **address** {*ipv4-address* [*mask*] | *ipv6-address prefix*}
8. **identity** {**address** {*ipv4-address* | *ipv6-address*} | **fqdn** *name* | **email** *email-id* | **key-id** *key-id*}
9. **pre-shared-key** {**local** | **remote**} [**0** | **6**] *line* **hex** *hexadecimal-string*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
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 kyr1`	Defines an IKEv2 key ring and enters IKEv2 key ring configuration mode.
Step 4	**peer** *name* **Example:** `Device(config-ikev2-keyring)# peer peer1`	Defines the peer or peer group and enters IKEv2 key ring peer configuration mode.
Step 5	**description** *line-of-description* **Example:** `Device(config-ikev2-keyring-peer)# description this is the first peer`	(Optional) Describes the peer or peer group.
Step 6	**hostname** *name* **Example:** `Device(config-ikev2-keyring-peer)# hostname host1`	Specifies the peer using a hostname.
Step 7	**address** {*ipv4-address* [*mask*] | *ipv6-address prefix*} **Example:** `Device(config-ikev2-keyring-peer)# address 10.0.0.1 255.255.255.0`	Specifies an IPv4 or IPv6 address or range for the peer. **Note** This IP address is the IKE endpoint address and is independent of the identity address.
Step 8	**identity** {**address** {*ipv4-address* | *ipv6-address*} | **fqdn** *name* | **email** *email-id* | **key-id** *key-id*} **Example:** `Device(config-ikev2-keyring-peer)# identity address 10.0.0.5`	Identifies the IKEv2 peer through the following identities: • E-mail • Fully qualified domain name (FQDN) • IPv4 or IPv6 address • Key ID **Note** The identity is available for key lookup on the IKEv2 responder only.

	Command or Action	Purpose
Step 9	**pre-shared-key** {**local** | **remote**} [**0** | **6**] *line* **hex** *hexadecimal-string* **Example:** `Device(config-ikev2-keyring-peer)# pre-shared-key local key1`	Specifies the preshared key for the peer.
Step 10	**end** **Example:** `Device(config-ikev2-keyring-peer)# end`	Exits IKEv2 key ring peer configuration mode and returns to privileged EXEC mode.

- What to Do Next, page 11

What to Do Next

After configuring the IKEv2 key ring, configure the IKEv2 profile. For more information, see the "Configuring IKEv2 Profile (Basic)" section.

Configuring an IKEv2 Profile (Basic)

Perform this task to configure the mandatory commands for an IKEv2 profile.

An IKEv2 profile is a repository of nonnegotiable parameters of the IKE security association (SA) (such as local or remote identities and authentication methods) and services available to authenticated peers that match the profile. An IKEv2 profile must be configured and associated with either a crypto map or an IPsec profile on the IKEv2 initiator. Use the **set ikev2-profile** *profile-name* command to associate a profile with a crypto map or an IPsec profile. To disassociate the profile, use the **no** form of the command.

The following rules apply to match statements:

- An IKEv2 profile must contain a match identity or a match certificate statement; otherwise, the profile is considered incomplete and is not used. An IKEv2 profile can have more than one match identity or match certificate statements.
- An IKEv2 profile must have a single match Front Door VPN routing and forwarding (FVRF) statement.
- When a profile is selected, multiple match statements of the same type are logically ORed, and multiple match statements of different types are logically ANDed.
- The match identity and match certificate statements are considered to be the same type of statements and are ORed.
- Configuration of overlapping profiles is considered a misconfiguration. In the case of multiple profile matches, no profile is selected.

Use the **show crypto ikev2 profile** *profile-name* command to display the IKEv2 profile.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 profile** *profile-name*
4. **description** *line-of-description*
5. **aaa accounting** {**psk** | **cert** | **eap**} *list-name*
6. **authentication** {**local** {**rsa-sig** | **pre-share** | **ecdsa-sig** | **eap**} | **remote** {**eap** [**query-identity** | **timeout** *seconds*] | **rsa-sig** | **pre-share** | **ecdsa-sig**}}
7. **dpd** *interval retry-interval* {**on-demand** | **periodic**}
8. **identity local** {**address** {*ipv4-address* | *ipv6-address*} | **dn** | **email** *email-string* | **fqdn** *fqdn-string* | **key-id** *opaque-string*}
9. **initial-contact** [**force**]
10. **ivrf** *name*
11. **keyring** {**local** *keyring-name* | **aaa** *list-name* **name-mangler** *mangler-name*}
12. **lifetime** *seconds*
13. **match** {**address local** {*ipv4-address* | *ipv6-address* | **interface** *name*} | **certificate** *certificate-map* | **fvrf** {*fvrf-name* | **any**} | **identity remote** {**address** {*ipv4-address* [*mask*] | *ipv6-address prefix*} | {**email** | **fqdn**} [*domain*] *string* | **key-id** *opaque-string*}}
14. **nat keepalive** *seconds*
15. **pki trustpoint** *trustpoint-label* [**sign** | **verify**]
16. **virtual-template** *number*
17. **end**

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 profile** *profile-name* **Example:** `Device(config)# crypto ikev2 profile profile1`	Defines an IKEv2 profile and enters IKEv2 profile configuration mode.

	Command or Action	Purpose
Step 4	**description** *line-of-description* **Example:** `Device(config-ikev2-profile)# description This is an IKEv2 profile`	(Optional) Describes the profile.
Step 5	**aaa accounting** {**psk** | **cert** | **eap**} *list-name* **Example:** `Device(config-ikev2-profile)# aaa accounting eap list1`	(Optional) Enables authentication, authorization, and accounting (AAA) accounting method lists for IPsec sessions. **Note** If the **psk**, **cert**, or **eap** keyword is not specified, the AAA accounting method list is used irrespective of the peer authentication method.
Step 6	**authentication** {**local** {**rsa-sig** | **pre-share** | **ecdsa-sig** | **eap**} | **remote** {**eap** [**query-identity** | **timeout** *seconds*] | **rsa-sig** | **pre-share** | **ecdsa-sig**}} **Example:** `Device(config-ikev2-profile)# authentication local ecdsa-sig`	Specifies the local or remote authentication method. **Note** You can specify only one local authentication method but multiple remote authentication methods.
Step 7	**dpd** *interval retry-interval* {**on-demand** | **periodic**} **Example:** `Device(config-ikev2-profile)# dpd 1000 250 periodic`	(Optional) Configures Dead Peer Detection (DPD) globally for peers matching the profile. **Note** DPD is disabled by default.
Step 8	**identity local** {**address** {*ipv4-address* | *ipv6-address*} | **dn** | **email** *email-string* | **fqdn** *fqdn-string* | **key-id** *opaque-string*} **Example:** `Device(config-ikev2-profile)# identity local email abc@example.com`	(Optional) Specifies the local IKEv2 identity type. **Note** If the local authentication method is a preshared key, the default local identity is the IP address. If the local authentication method is a Rivest, Shamir, and Adleman (RSA) signature, the default local identity is a Distinguished Name.
Step 9	**initial-contact** [**force**] **Example:** `Device(config-ikev2-profile)# initial-contact force`	Enforces initial contact processing if the initial contact notification is not received in the IKE_AUTH exchange.

Command or Action	Purpose
Step 10 **ivrf** *name* **Example:** `Device(config-ikev2-profile)# ivrf vrf1`	(Optional) Specifies a user-defined VPN routing and forwarding (VRF) or global VRF if the IKEv2 profile is attached to a crypto map. • If the IKEv2 profile is used for tunnel protection, the Inside VRF (IVRF) for the tunnel interface should be configured on the tunnel interface. **Note** IVRF specifies the VRF for cleartext packets. The default value for IVRF is FVRF.
Step 11 **keyring** {**local** *keyring-name* | **aaa** *list-name* **name-mangler** *mangler-name*} **Example:** `Device(config-ikev2-profile)# keyring aaa keyring1 name-mangler mangler1`	Specifies the local or AAA-based key ring that must be used with the local and remote preshared key authentication method. **Note** You can specify only one key ring. Local AAA is not supported for AAA-based preshared keys. **Note** Depending on your release, the **local** keyword and the **name-mangler** *mangler-name* keyword-argument pair should be used.
Step 12 **lifetime** *seconds* **Example:** `Device(config-ikev2-profile)# lifetime 1000`	Specifies the lifetime, in seconds, for the IKEv2 SA.
Step 13 **match** {**address local** {*ipv4-address* | *ipv6-address* | **interface** *name*} | **certificate** *certificate-map* | **fvrf** {*fvrf-name* | **any**} | **identity remote** {**address** {*ipv4-address* [*mask*] | *ipv6-address prefix*} | {**email** | **fqdn**} [*domain*] *string* | **key-id** *opaque-string*}} **Example:** `Device(config-ikev2-profile)# match address local interface Ethernet 2/0`	Uses match statements to select an IKEv2 profile for a peer.
Step 14 **nat keepalive** *seconds* **Example:** `Device(config-ikev2-profile)# nat keepalive 500`	(Optional) Enables NAT keepalive and specifies the duration in seconds. • NAT is disabled by default.

	Command or Action	Purpose
Step 15	**pki trustpoint** *trustpoint-label* [**sign** | **verify**] **Example:** `Device(config-ikev2-profile)# pki trustpoint tsp1 sign`	Specifies Public Key Infrastructure (PKI) trustpoints for use with the RSA signature authentication method. **Note** If the **sign** or **verify** keyword is not specified, the trustpoint is used for signing and verification. **Note** In contrast to IKEv1, a trustpoint must be configured in an IKEv2 profile for certificate-based authentication to succeed. There is no fallback for globally configured trustpoints if this command is not present in the configuration. The trustpoint configuration applies to the IKEv2 initiator and responder.
Step 16	**virtual-template** *number* **Example:** `Device(config-ikev2-profile)# virtual-template 125`	(Optional) Specifies the virtual template for cloning a virtual access interface (VAI). **Note** For the IPsec Dynamic Virtual Tunnel Interface (DVTI), a virtual template must be specified in an IKEv2 profile, without which an IKEv2 session is not initiated.
Step 17	**end** **Example:** `Device(config-ikev2-profile)# end`	Exits IKEv2 profile configuration mode and returns to privileged EXEC mode.

Configuring Advanced Internet Key Exchange Version 2 CLI Constructs

This section describes the global IKEv2 CLI constructs and how to override the IKEv2 default CLI constructs. IKEv2 smart defaults support most use cases and hence, we recommend that you override the defaults only if they are required for specific use cases not covered by the defaults.

Perform the following tasks to configure advanced IKEv2 CLI constructs:

Configuring Global IKEv2 Options

Perform this task to configure global IKEv2 options that are independent of peers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 certificate-cache** *number-of-certificates*
4. **crypto ikev2 cookie-challenge** *number*
5. **crypto ikev2 diagnose error** *number*
6. **crypto ikev2 dpd** *interval retry-interval* {**on-demand** | **periodic**}
7. **crypto ikev2 http-url cert**
8. **crypto ikev2 limit** {**max-in-negotiation-sa** *limit* | **max-sa** *limit*}
9. **crypto ikev2 nat keepalive** *interval*
10. **crypto ikev2 window** *size*
11. **crypto logging ikev2**
12. **end**

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 certificate-cache** *number-of-certificates* **Example:** `Device(config)# crypto ikev2 certificate-cache 750`	Defines the cache size for storing certificates fetched from HTTP URLs.
Step 4	**crypto ikev2 cookie-challenge** *number* **Example:** `Device(config)# crypto ikev2 cookie-challenge 450`	Enables an IKEv2 cookie challenge only when the number of half-open security associations (SAs) exceeds the configured number. • Cookie challenge is disabled by default.
Step 5	**crypto ikev2 diagnose error** *number* **Example:** `Device(config)# crypto ikev2 diagnose error 500`	Enables IKEv2 error diagnostics and defines the number of entries in the exit path database. • IKEv2 error diagnostics is disabled by default.

	Command or Action	Purpose
Step 6	**crypto ikev2 dpd** *interval retry-interval* {**on-demand** | **periodic**} **Example:** `Device(config)# crypto ikev2 dpd 500 50 on-demand`	Allows live checks for peers as follows: • Dead Peer Detection (DPD) is disabled by default.
Step 7	**crypto ikev2 http-url cert** **Example:** `Device(config)# crypto ikev2 http-url cert`	Enables the HTTP CERT support. • HTTP CERT is disabled by default.
Step 8	**crypto ikev2 limit** {**max-in-negotiation-sa** *limit* | **max-sa** *limit*} **Example:** `Device(config)# crypto ikev2 limit max-in-negotiation-sa 5000`	Enables connection admission control (CAC). • Connection admission control is disabled by default.
Step 9	**crypto ikev2 nat keepalive** *interval* **Example:** `Device(config)# crypto ikev2 nat keepalive 500`	Enables the Network Address Translation (NAT) keepalive that prevents the deletion of NAT entries in the absence of any traffic when there is NAT between Internet Key Exchange (IKE) peers. • NAT keepalive is disabled by default.
Step 10	**crypto ikev2 window** *size* **Example:** `Device(config)# crypto ikev2 window 15`	Allows multiple IKEv2 request-response pairs in transit. • The default window size is 5.
Step 11	**crypto logging ikev2** **Example:** `Device(config)# crypto logging ikev2`	Enables IKEv2 syslog messages. • IKEv2 syslog messages are disabled by default.
Step 12	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Configuring IKEv2 Proposal

Refer to the “IKEv2 Smart Defaults” section for information on the default IKEv2 proposal.

Perform this task to override the default IKEv2 proposal or to manually configure the proposals if you do not want to use the default proposal.

An IKEv2 proposal is a set of transforms used in the negotiation of IKEv2 SA as part of the IKE_SA_INIT exchange. An IKEv2 proposal is regarded as complete only when it has at least an encryption algorithm, an integrity algorithm, and a Diffie-Hellman (DH) group configured. If no proposal is configured and attached to an IKEv2 policy, the default proposal in the default IKEv2 policy is used in negotiation.

Note Security threats, as well as the cryptographic technologies to help protect against them, are constantly changing. For more information about the latest Cisco cryptographic recommendations, see the *Next Generation Encryption* (NGE) white paper.

Although the IKEv2 proposal is similar to the **crypto isakmp policy** command, the IKEv2 proposal differs as follows:

- An IKEv2 proposal allows configuring one or more transforms for each transform type.
- An IKEv2 proposal does not have any associated priority.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 proposal** *name*
4. **encryption** {**3des**} {**aes-cbc-128**} {**aes-cbc-192**} {**aes-cbc-256**}
5. **integrity** {**md5**} {**sha1**} {**sha256**} {**sha384**} {**sha512**}
6. **group** {**1**} {**14**} {**15**} {**16**} {**19**} {**2**} {**20**} {**24**} {**5**}
7. **end**
8. **show crypto ikev2 proposal** [*name* | **default**]

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.

	Command or Action	Purpose
Step 3	**crypto ikev2 proposal** *name* **Example:** `Device(config)# crypto ikev2 proposal proposal1`	Overrides the default IKEv2 proposal, defines an IKEv2 proposal name, and enters IKEv2 proposal configuration mode.
Step 4	**encryption** {**3des**} {**aes-cbc-128**} {**aes-cbc-192**} {**aes-cbc-256**} **Example:** `Device(config-ikev2-proposal)# encryption aes-cbc-128 aes-cbc-192`	Specifies one or more transforms of the encryption type, which are as follows: • **3des** (No longer recommended) • **aes-cbc-128** • **aes-cbc-192** • **aes-cbc-256**
Step 5	**integrity** {**md5**} {**sha1**} {**sha256**} {**sha384**} {**sha512**} **Example:** `Device(config-ikev2-proposal)# integrity sha1`	Specifies one or more transforms of the integrity algorithm type, which are as follows: • The **md5** keyword specifies MD5 (HMAC variant) as the hash algorithm. (No longer recommended) • The **sha1** keyword specifies SHA-1 (HMAC variant) as the hash algorithm. • The **sha256** keyword specifies SHA-2 family 256-bit (HMAC variant) as the hash algorithm. • The **sha384** keyword specifies SHA-2 family 384-bit (HMAC variant) as the hash algorithm. • The **sha512** keyword specifies SHA-2 family 512-bit (HMAC variant) as the hash algorithm.
Step 6	**group** {**1**} {**14**} {**15**} {**16**} {**19**} {**2**} {**20**} {**24**} {**5**} **Example:** `Device(config-ikev2-proposal)# group 14`	Specifies the Diffie-Hellman (DH) group identifier. • The default DH group identifiers are group 2 and 5 in the IKEv2 proposal. ◦ **1**—768-bit DH (No longer recommended). ◦ **2**—1024-bit DH (No longer recommended). ◦ **5**—1536-bit DH (No longer recommended). ◦ **14**—Specifies the 2048-bit DH group. ◦ **15**—Specifies the 3072-bit DH group. ◦ **16**—Specifies the 4096-bit DH group. ◦ **19**—Specifies the 256-bit elliptic curve DH (ECDH) group. ◦ **20**—Specifies the 384-bit ECDH group. ◦ **24**—Specifies the 2048-bit DH group. The group chosen must be strong enough (have enough bits) to protect the IPsec keys during negotiation. A generally accepted guideline recommends the use of a 2048-bit group after 2013 (until 2030). Either group 14 or group 24 can be selected to meet this guideline. Even if a longer-lived security method is needed, the use of Elliptic Curve Cryptography is recommended, but group 15 and group 16 can also be considered.

	Command or Action	Purpose
Step 7	**end** **Example:** `Device(config-ikev2-proposal)# end`	Exits IKEv2 proposal configuration mode and returns to privileged EXEC mode.
Step 8	**show crypto ikev2 proposal** [*name* | **default**] **Example:** `Device# show crypto ikev2 proposal default`	(Optional) Displays the IKEv2 proposal.

- What to Do Next, page 20

What to Do Next

After you create the IKEv2 proposal, attach it to a policy so that the proposal is picked for negotiation. For information about completing this task, see the “Configuring IKEv2 Policy” section.

Configuring IKEv2 Policies

See the “IKEv2 Smart Defaults” section for information about the default IKEv2 policy.

Perform this task to override the default IKEv2 policy or to manually configure the policies if you do not want to use the default policy.

An IKEv2 policy must contain at least one proposal to be considered as complete and can have match statements, which are used as selection criteria to select a policy for negotiation. During the initial exchange, the local address (IPv4 or IPv6) and the Front Door VRF (FVRF) of the negotiating SA are matched with the policy and the proposal is selected.

The following rules apply to the match statements:

- An IKEv2 policy without any match statements will match all peers in the global FVRF.
- An IKEv2 policy can have only one match FVRF statement.
- An IKEv2 policy can have one or more match address local statements.
- When a policy is selected, multiple match statements of the same type are logically ORed and match statements of different types are logically ANDed.
- There is no precedence between match statements of different types.
- Configuration of overlapping policies is considered a misconfiguration. In the case of multiple, possible policy matches, the first policy is selected.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 policy** *name*
4. **proposal** *name*
5. **match fvrf** {*fvrf-name* | **any**}
6. **match address local** {*ipv4-address* | *ipv6-address*}
7. **end**
8. **show crypto ikev2 policy** [*policy-name* | **default**]

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 policy** *name* **Example:** `Device(config)# crypto ikev2 policy policy1`	Overrides the default IKEv2 policy, defines an IKEv2 policy name, and enters IKEv2 policy configuration mode.
Step 4	**proposal** *name* **Example:** `Device(config-ikev2-policy)# proposal proposal1`	Specifies the proposals that must be used with the policy. • The proposals are prioritized in the order of listing. **Note** You must specify at least one proposal. You can specify additional proposals with each proposal in a separate statement.
Step 5	**match fvrf** {*fvrf-name*	**any**} **Example:** `Device(config-ikev2-policy)# match fvrf any`	(Optional) Matches the policy based on a user-configured FVRF or any FVRF. • The default is global FVRF. **Note** The **match fvrf any** command must be explicitly configured in order to match any VRF. The FVRF specifies the VRF in which the IKEv2 packets are negotiated.

	Command or Action	Purpose
Step 6	**match address local** {*ipv4-address* | *ipv6-address*} **Example:** `Device(config-ikev2-policy)# match address local 10.0.0.1`	(Optional) Matches the policy based on the local IPv4 or IPv6 address. • The default matches all the addresses in the configured FVRF.
Step 7	**end** **Example:** `Device(config-ikev2-policy)# end`	Exits IKEv2 policy configuration mode and returns to privileged EXEC mode.
Step 8	**show crypto ikev2 policy** [*policy-name* | **default**] **Example:** `Device# show crypto ikev2 policy policy1`	(Optional) Displays the IKEv2 policy.

Configuration Examples for Internet Key Exchange Version 2

Configuration Examples for Basic Internet Key Exchange Version 2 CLI Constructs

Example: Configuring the IKEv2 Key Ring

Example: IKEv2 Key Ring with Multiple Peer Subblocks

The following example shows how to configure an Internet Key Exchange Version 2 (IKEv2) key ring with multiple peer subblocks:

```
crypto ikev2 keyring keyring-1
 peer peer1
  description peer1
  address 209.165.200.225 255.255.255.224
  pre-shared-key key-1
 peer peer2
  description peer2
  hostname peer1.example.com
  pre-shared-key key-2
 peer peer3
  description peer3
  hostname peer3.example.com
  identity key-id abc
  address 209.165.200.228 255.255.255.224
  pre-shared-key key-3
```

Example: IKEv2 Keyring with Symmetric Preshared Keys Based on an IP Address

The following example shows how to configure an IKEv2 key ring with symmetric preshared keys based on an IP address. The following is the initiator's key ring:

```
crypto ikev2 keyring keyring-1
 peer peer1
  description peer1
  address 209.165.200.225 255.255.255.224
  pre-shared-key key1
```

The following is the responder's key ring:

```
crypto ikev2 keyring keyring-1
 peer peer2
  description peer2
  address 209.165.200.228 255.255.255.224
  pre-shared-key key1
```

Example: IKEv2 Key Ring with Asymmetric Preshared Keys Based on an IP Address

The following example shows how to configure an IKEv2 key ring with asymmetric preshared keys based on an IP address. The following is the initiator's key ring:

```
crypto ikev2 keyring keyring-1
 peer peer1
  description peer1 with asymmetric keys
  address 209.165.200.225 255.255.255.224
  pre-shared-key local key1
  pre-shared-key remote key2
```

The following is the responder's key ring:

```
crypto ikev2 keyring keyring-1
 peer peer2
  description peer2 with asymmetric keys
  address 209.165.200.228 255.255.255.224
  pre-shared-key local key2
  pre-shared-key remote key1
```

Example: IKEv2 Key Ring with Asymmetric Preshared Keys Based on a Hostname

The following example shows how to configure an IKEv2 key ring with asymmetric preshared keys based on the hostname. The following is the initiator's key ring:

```
crypto ikev2 keyring keyring-1
 peer host1
  description host1 in example domain
  hostname host1.example.com
  pre-shared-key local key1
  pre-shared-key remote key2
```

The following is the responder's keyring:

```
crypto ikev2 keyring keyring-1
 peer host2
  description host2 in abc domain
  hostname host2.example.com
  pre-shared-key local key2
  pre-shared-key remote key1
```

Example: IKEv2 Key Ring with Symmetric Preshared Keys Based on an Identity

The following example shows how to configure an IKEv2 key ring with symmetric preshared keys based on an identity:

```
crypto ikev2 keyring keyring-4
 peer abc
  description example domain
  identity fqdn example.com
  pre-shared-key abc-key-1
 peer user1
  description user1 in example domain
  identity email user1@example.com
  pre-shared-key abc-key-2
 peer user1-remote
  description user1 example remote users
  identity key-id example
  pre-shared-key example-key-3
```

Example: IKEv2 Key Ring with a Wildcard Key

The following example shows how to configure an IKEv2 key ring with a wildcard key:

```
crypto ikev2 keyring keyring-1
 peer cisco
  description example domain
  address 0.0.0.0 0.0.0.0
  pre-shared-key example-key
```

Example: Matching a Key Ring

The following example shows how a key ring is matched:

```
crypto ikev2 keyring keyring-1
 peer cisco
  description example.com
  address 0.0.0.0 0.0.0.0
  pre-shared-key xyz-key
 peer peer1
  description abc.example.com
  address 10.0.0.0 255.255.0.0
  pre-shared-key abc-key
 peer host1
  description host1@abc.example.com
  address 10.0.0.1
  pre-shared-key host1-example-key
```

In the example shown, the key lookup for peer 10.0.0.1 first matches the wildcard key example-key, then the prefix key example-key, and finally the host key host1-example-key. The best match host1-example-key is used.

```
crypto ikev2 keyring keyring-2
 peer host1
 description host1 in abc.example.com sub-domain
 address 10.0.0.1
 pre-shared-key host1-example-key
 peer host2
  description example domain
  address 0.0.0.0 0.0.0.0
  pre-shared-key example-key
```

In the example shown, the key lookup for peer 10.0.0.1 would first match the host key host1-abc-key. Because this is a specific match, no further lookup is performed.

Example: Configuring the Profile

Example: IKEv2 Profile Matched on Remote Identity

The following profile supports peers that identify themselves using fully qualified domain name (FQDN) example.com and authenticate with the RSA signature using trustpoint-remote. The local node authenticates itself with a preshared key using keyring-1.

```
crypto ikev2 profile profile2
 match identity remote fqdn example.com
 identity local email router2@example.com
 authentication local pre-share
 authentication remote rsa-sig
 keyring keyring-1
 pki trustpoint trustpoint-remote verify
 lifetime 300
 dpd 5 10 on-demand
 virtual-template 1
```

Example: IKEv2 Profile Supporting Two Peers

The following example shows how to configure an IKEv2 profile supporting two peers that use different authentication methods:

```
crypto ikev2 profile profile2
 match identity remote email user1@example.com
 match identity remote email user2@example.com
 identity local email router2@cisco.com
 authentication local rsa-sig
 authentication remote pre-share
 authentication remote rsa-sig
 keyring keyring-1
 pki trustpoint trustpoint-local sign
 pki trustpoint trustpoint-remote verify
 lifetime 300
 dpd 5 10 on-demand
 virtual-template 1
```

Example: Configuring FlexVPN Site-to-Site with Dynamic Routing Using Certificates and IKEv2 Smart Defaults

The following examples show a site-to-site connection between a branch device (initiator, using a static virtual tunnel interface [sVTI]) and a central device (responder, using a dynamic virtual tunnel interface [dVTI]) with dynamic routing over the tunnel. The example uses IKEv2 smart defaults, and the authentication is performed using certificates (RSA signatures).

Note A RSA modulus size of 2048 is recommended.

The peers use the FQDN as their IKEv2 identity, and the IKEv2 profile on the responder matches the domain in the identity FQDN.

The configuration on the initiator (branch device) is as follows:

```
hostname branch
ip domain name cisco.com
!
crypto ikev2 profile branch-to-central
 match identity remote fqdn central.cisco.com
 identity local fqdn branch.cisco.com
 authentication local rsa-sig
 authentication remote rsa-sig
 pki trustpoint CA
!
crypto ipsec profile svti
 set ikev2-profile branch-to-central
!
interface Tunnel0
 ip address 172.16.0.101 255.255.255.0
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel destination 10.0.0.100
 tunnel protection ipsec profile svti
!
interface Ethernet0/0
 ip address 10.0.0.101 255.255.255.0
!
interface Ethernet1/0
 ip address 192.168.101.1 255.255.255.0
!
router rip
 version 2
 passive-interface Ethernet1/0
 network 172.16.0.0
 network 192.168.101.0
 no auto-summary
```

The configuration on the responder (central router) is as follows:

```
hostname central
ip domain name cisco.com
!
crypto ikev2 profile central-to-branch
 match identity remote fqdn domain cisco.com
 identity local fqdn central.cisco.com
 authentication local rsa-sig
 authentication remote rsa-sig
 pki trustpoint CA
 virtual-template 1
!
interface Loopback0
 ip address 172.16.0.100 255.255.255.0
!
interface Ethernet0/0
 ip address 10.0.0.100 255.255.255.0
```

```
!
interface Ethernet1/0
 ip address 192.168.100.1 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered Loopback0
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile default
!
router rip
 version 2
 passive-interface Ethernet1/0
 network 172.16.0.0
 network 192.168.100.0
 no auto-summary
```

Configuration Examples for Advanced Internet Key Exchange Version 2 CLI Constructs

Example: Configuring the Proposal

Example: IKEv2 Proposal with One Transform for Each Transform Type

This example shows how to configure an IKEv2 proposal with one transform for each transform type:

```
crypto ikev2 proposal proposal-1
 encryption aes-cbc-128
 integrity sha1
 group 14
```

Example: IKEv2 Proposal with Multiple Transforms for Each Transform Type

This example shows how to configure an IKEv2 proposal with multiple transforms for each transform type:

```
crypto ikev2 proposal proposal-2
 encryption aes-cbc-128 aes-cbc-192
 integrity sha1
 group 14
```

Note Cisco no longer recommends using 3DES, MD5 (including HMAC variant), and Diffie-Hellman(DH) groups 1, 2 and 5; instead, you should use AES, SHA-256 and DH Groups 14 or higher. For more information about the latest Cisco cryptographic recommendations, see the *Next Generation Encryption* (NGE) white paper.

The IKEv2 proposal proposal-2 shown translates to the following prioritized list of transform combinations:

- aes-cbc-128, sha1, 14
- aes-cbc-192, sha1, 14

Example: IKEv2 Proposals on the Initiator and Responder

The following example shows how to configure IKEv2 proposals on the initiator and the responder. The proposal on the initiator is as follows:

```
crypto ikev2 proposal proposal-1
 encryption aes-cbc-192 aes-cbc-128
 integrity sha-256 sha1
 group 14 24
```

The proposal on the responder is as follows:

```
crypto ikev2 proposal proposal-2
 encryption aes-cbc-128 aes-cbc-192
 peer
 integrity sha1 sha-256
 group 24 14
```

The selected proposal will be as follows:

```
encryption aes-cbc-128
integrity sha1
group 14
```

In the proposals shown for the initiator and responder, the initiator and responder have conflicting preferences. In this case, the initiator is preferred over the responder.

Example: Configuring the Policy

Example: IKEv2 Policy Matched on a VRF and Local Address

The following example shows how an IKEv2 policy is matched based on a VRF and local address:

```
crypto ikev2 policy policy2
 match vrf vrf1
 match local address 10.0.0.1
 proposal proposal-1
```

Example: IKEv2 Policy with Multiple Proposals That Match All Peers in a Global VRF

The following example shows how an IKEv2 policy with multiple proposals matches the peers in a global VRF:

```
crypto ikev2 policy policy2
 proposal proposal-A
 proposal proposal-B
 proposal proposal-B
```

Example: IKEv2 Policy That Matches All Peers in Any VRF

The following example shows how an IKEv2 policy matches the peers in any VRF:

```
crypto ikev2 policy policy2
 match vrf any
 proposal proposal-1
```

Example: Matching a Policy

Do not configure overlapping policies. If there are multiple possible policy matches, the best match is used, as shown in the following example:

```
crypto ikev2 policy policy1
 match fvrf fvrf1
crypto ikev2 policy policy2
 match fvrf fvff1
 match local address 10.0.0.1
```

The proposal with FVRF as fvrf1 and the local peer as 10.0.0.1 matches policy1 and policy2, but policy2 is selected because it is the best match.

Where to Go Next

After configuring IKEv2, proceed to configure IPsec VPNs. For more information, see the “Configuring Security for VPNs with IPsec” module.

Additional References

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*
Suite-B ESP transforms	*Configuring Security for VPNs with IPsec*
Suite-B SHA-2 family (HMAC variant) and elliptic curve (EC) key pair configuration	*Configuring Internet Key Exchange for IPsec VPNs*

Related Topic	Document Title
Suite-B elliptic curve Diffie-Hellman (ECDH) support for IPsec SA negotiation	*Configuring Internet Key Exchange for IPsec VPNs*
Suite-B support for certificate enrollment for a PKI	*Configuring Certificate Enrollment for a PKI*
Supported standards for use with IKE	*Internet Key Exchange for IPsec VPNs Configuration Guide*
Recommended cryptographic algorithms	*Next Generation Encryption*

RFCs

RFC	Title
RFC 4306	*Internet Key Exchange (IKEv2) Protocol*
RFC 4869	*Suite B Cryptographic Suites for IPsec*
RFC 5685	*Redirect Mechanism for the Internet Key Exchange Protocol Version 2 (IKEv2)*

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 Configuring Internet Key Exchange Version 2 (IKEv2) and FlexVPN Site-to-Site

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 2 Feature Information for Configuring Internet Key Exchange Version 2 (IKEv2) and FlexVPN Site-to-Site

Feature Name	Releases	Feature Information
IKEv2 Site to Site	15.1(1)T 15.2(4)S 15.1(1)SY	IKEv2 is a component of IP Security (IPsec) and is used for performing mutual authentication and establishing and maintaining security associations (SAs). The following commands were introduced or modified: **aaa accounting (IKEv2 profile), address (IKEv2 keyring), authentication (IKEv2 profile), crypto ikev2 keyring, crypto ikev2 policy, crypto ikev2 profile, crypto ikev2 proposal, description (IKEv2 keyring), dpd, hostname (IKEv2 keyring), identity (IKEv2 keyring), identity local, ivrf, keyring, lifetime (IKEv2 profile), match (IKEv2 profile), nat, peer, pki trustpoint, pre-shared-key (IKEv2 keyring), proposal, virtual-template (IKEv2 profile), clear crypto ikev2 sa, clear crypto ikev2 stat, clear crypto session, clear crypto ikev2 sa, debug crypto ikev2, show crypto ikev2 diagnose error, show crypto ikev2 policy, show crypto ikev2 profile, show crypto ikev2 proposal, show crypto ikev2 sa, show crypto ikev2 session, show crypto ikev2 stats, show crypto session, show crypto socket.**

Feature Name	Releases	Feature Information
IPv6 support for IPSec and IKEv2	15.1(4)M 15.1(1)SY	This feature allows IPv6 addresses to be added to IPSec and IKEv2 protocols. The following commands were introduced or modified: **address (IKEv2 keyring), identity (IKEv2 keyring), identity local, match (IKEv2 policy), and match (IKEv2 profile), show crypto ikev2 session, show crypto ikev2 sa, show crypto ikev2 profile, show crypto ikev2 policy, debug crypto condition, clear crypto ikev2 sa.**

Feature Name	Releases	Feature Information
Suite-B support in IOS SW crypto	15.1(2)T	Suite-B adds support for the SHA-2 family (HMAC variant) hash algorithm used to authenticate packet data and verify the integrity verification mechanisms for the IKEv2 proposal configuration. HMAC is a variant that provides an additional level of hashing. Suite-B also allows the Elliptic Curve Digital Signature Algorithm (ECDSA) signature (ECDSA-sig), as defined in RFC 4754, to be the authentication method for IKEv2. Suite-B requirements comprise of four user interface suites of cryptographic algorithms for use with IKE and IPSec that are described in RFC 4869. Each suite is consists of an encryption algorithm, a digital signature algorithm, a key agreement algorithm, and a hash or message digest algorithm. See the Configuring Security for VPNs with IPsec module for more information about Cisco IOS Suite-B support. The following commands were introduced or modified: **authentication, group, identity (IKEv2 profile), integrity, match (IKEv2 profile).**

Appendix: IKEv2 and Legacy VPNs

This module provides examples on how to configure IKEv2 on crypto map based configurations.

Note Crypto maps are considered a legacy configuration construct. It is recommended that you migrate existing crypto map based setups to use tunnel protection and virtual interfaces.

Example: Configuring Crypto-Map-Based IKEv2 Peers Using Preshared Key Authentication Method

The following example shows how to configure crypto-map-based IKEv2 peers using the preshared key authentication method between a static crypto-map IKEv2 initiator and a dynamic crypto-map IKEv2 responder. The initiator configuration is as follows:

```
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 keyring v2-kr1
 peer abc
  address 209.165.200.231 255.255.255.224
  pre-shared-key abc
 !
!
!
crypto ikev2 profile prof
 match fvrf any
 match identity remote fqdn dmap-responder
 identity local fqdn smap-initiator
 authentication local pre-share
 authentication remote pre-share
```

```
 keyring v2-kr1
!
!
crypto ipsec transform-set trans esp-aes-cbc-128 esp-sha-hmac
!
crypto map cmap 1 ipsec-isakmp
 set peer 209.165.200.225
 set transform-set trans
 set ikev2-profile prof
 match address ikev2list
!
interface Loopback0
 ip address 209.165.200.226 255.255.255.224
!
interface Ethernet0/0
 ip address 209.165.200.227 255.255.255.224
 crypto map cmap
!
ip route 209.165.200.229 255.255.255.224 209.165.200.225
!
ip access-list extended ikev2list
 permit ip any any
!
```

The responder configuration is as follows:

```
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 keyring v2-kr1
 peer abc
  address 209.165.200.228
  pre-shared-key abc
 !
!
!
crypto ikev2 profile prof
 match fvrf any
 match identity remote fqdn smap-initiator
 identity local fqdn dmap-responder
 authentication local pre-share
 authentication remote pre-share
 keyring v2-kr1
 ivrf global
!
!
crypto ipsec transform-set trans esp-aes-cbc-128 esp-sha-hmac
!
crypto dynamic-map dmap 1
 set transform-set trans
 set reverse-route tag 222
 set ikev2-profile prof
 match address ikev2list
!
crypto map cmap 1 ipsec-isakmp dynamic dmap
!
interface Loopback0
 ip address 209.165.200.230 255.255.255.224
!
interface Ethernet0/0
 ip address 209.165.200.231 255.255.255.224
 crypto map cmap
!
ip route 209.165.200.233 255.255.255.224 209.165.200.228
!
ip access-list extended ikev2list
```

```
 permit ip any any
!
```

To initiate the connection between the initiator and the responder, enter the following command at the initiator's CLI:

```
ping 209.165.200.230 source 209.165.200.226
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.230, timeout is 2 seconds:
Packet sent with a source address of 209.165.200.226
%IKEV2-5-OSAL_INITIATE_TUNNEL: Received request to establish an IPsec tunnel; local
traffic selector = Address Range: 209.165.200.226-209.165.200.226 Protocol: 1 Port Range:
0-65535; remote traffic selector = Address Range: 209.165.200.230-209.165.200.230
Protocol: 1 Port Range: 0-65535
%IKEV2-5-SA_UP: SA UP
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 8/11/12 ms
```

To display the session details, enter the following **show** commands:

```
show crypto session
Crypto session current status
Interface: Ethernet0/0
Session status: UP-ACTIVE
Peer: 209.165.200.225 port 500
  IKEv2 SA: local 209.165.200.228/500 remote 209.165.200.231/500 Active
  IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
        Active SAs: 2, origin: crypto map
show crypto ikev2 sa detail
Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         209.165.200.228/500
209.165.200.231/500                                          (none)/(none)        READY
      Encr: AES-CBC, Keysize: 128, Hash: SHA96, DH Grp:14, Auth sign: PSK, Auth verify:
PSK
      Life/Active Time: 86400/21 sec
      CE id: 1002, Session-id: 2
      Status Description: Negotiation done
      Local spi: 687752902752A6FD       Remote spi: C9DCCFC65493D14F
      Local id: smap-initiator
      Remote id: dmap-responder
      Local req msg id:  2              Remote req msg id:  0
      Local next msg id: 2              Remote next msg id: 0
      Local req queued:  2              Remote req queued:  0
      Local window:      5              Remote window:      5
      DPD configured for 0 seconds, retry 0
      NAT-T is not detected
```

Example: Configuring Crypto Map-Based IKEv2 Peers Using Certification Authentication Method

The following example shows how to configure crypto-map-based IKEv2 peers using the certificate authentication method between a static crypto-map IKEv2 initiator, a dynamic crypto-map IKEv2 responder, and a CA server. The initiator configuration is as follows:

```
crypto pki trustpoint ca-server
 enrollment url http://10.1.1.3:80
 revocation-check none
!
crypto pki certificate map cmap-1 1
 subject-name eq hostname = responder
!
!
crypto pki certificate chain ca-server
 certificate 02
  308201AF 30820118 A0030201 02020102 300D0609 2A864886 F70D0101 04050030
```

```
  14311230 10060355 04031309 63612D73 65727665 72301E17 0D313030 33313031
  32353132 355A170D 31313033 31303132 35313235 5A301A31 18301606 092A8648
  86F70D01 09021609 494E4954 4941544F 52305C30 0D06092A 864886F7 0D010101
  0500034B 00304802 4100A47E 8C58BA89 8CCDC5A4 5A63BD29 C331A2A5 393F4616
  6B43FD2E 5ED4C81A 913E3B13 33A9B2DC CFC30391 24BB0DC8 B28FD6F1 C008D101
  34C10062 30F88CF7 9D630203 010001A3 4F304D30 0B060355 1D0F0404 030205A0
  301F0603 551D2304 18301680 144871D9 002C66DF D85FACB8 45D1D25F EA357455
  91301D06 03551D0E 04160414 E77C74E7 183AB530 83DC531B 1DE3DA1D 914A925D
  300D0609 2A864886 F70D0101 04050003 81810042 21934B77 7E485E6F EE717D75
  6407B361 45190CEF E1A29CF2 6FA29E9A 5ECC1CEE B273533D 1453F6CE 1FDDA747
  7E701B4B 2A2AE53F D67C2345 952325BA 30950435 0706C5EE A7A8B414 CFEEB7A2
  9CD46F8F 3F663268 A20C4CCF E75D61EF 03FBA85D EDD6B26E 63653F09 F97DAFA6
  6C76E44E C9CA3FDC 6CD85D30 169A1D9E 4E870B
        quit
 certificate ca 01
  30820201 3082016A A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  14311230 10060355 04031309 63612D73 65727665 72301E17 0D313030 33313031
  32343933 385A170D 31333033 30393132 34393338 5A301431 12301006 03550403
  13096361 2D736572 76657230 819F300D 06092A86 4886F70D 01010105 0003818D
  00308189 02818100 DA4ECE09 B998F670 598F32C1 7E9FA920 1D217AC4 293B842E
  7563CE11 B2F0F822 23077930 636C8293 00F6CFDD F6C9B0F5 8348BE58 6478F631
  7D44152F 494AEBCC A507FA6B 408D6BBB FAAB0A7A 2E7546A8 CA70F9A6 0F7F6824
  554BD833 060D657D ABDF406C 69EEF449 7A4F9AFE 6F0852E7 05DEDAC1 D433191E
  712868C2 A94E642B 02030100 01A36330 61300F06 03551D13 0101FF04 05300301
  01FF300E 0603551D 0F0101FF 04040302 0186301F 0603551D 23041830 16801448
  71D9002C 66DFD85F ACB845D1 D25FEA35 74559130 1D060355 1D0E0416 04144871
  D9002C66 DFD85FAC B845D1D2 5FEA3574 5591300D 06092A86 4886F70D 01010405
  00038181 00AFC36B 8A917284 06BD51CB 83BDC4E8 9457A361 6CAAF416 3BBEF691
  04215AC5 EDBC5730 C071C2FB 8A6C90CF D6AB39C2 3BC2147F D35553D9 028B2155
  802E50DB 48CDE067 B3857447 89A1C733 D81EFEF7 1115480F 70ED2F22 F27E35A1
  F3BB597C 7C8F717B FAAD79D3 0F469702 DE9190E4 B1B0808E 46A118EB 887CEAEB
  DFE2900E D2
        quit
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 profile prof
 match fvrf any
 match certificate cmap-1
 identity local dn
 authentication local rsa-sig
 authentication remote pre-share
 authentication remote rsa-sig
 pki trustpoint ca-server
!
!
crypto ipsec transform-set trans esp-aes-cbc-128 esp-sha-hmac
!
crypto map cmap 1 ipsec-isakmp
 set peer 209.165.200.225
 set transform-set trans
 set ikev2-profile prof
 match address ikev2list
!
interface Loopback0
 ip address 209.165.200.226 255.255.255.224
!
interface Ethernet0/0
 ip address 209.165.200.227 255.255.255.224
 crypto map cmap
!
interface Ethernet1/0
 ip address 209.165.200.228 255.255.255.224
!
ip route 209.165.200.229 255.255.255.224 209.265.200.231
!
ip access-list extended ikev2list
```

```
 permit ip any any
!
```

The responder configuration is as follows:

```
crypto pki trustpoint ca-server
 enrollment url http://10.1.1.3:80
 revocation-check none
!
!
!
crypto pki certificate map cmap-2 1
 subject-name eq hostname = initiator
!
crypto pki certificate chain ca-server
 certificate 03
  308201AF 30820118 A0030201 02020103 300D0609 2A864886 F70D0101 04050030
  14311230 10060355 04031309 63612D73 65727665 72301E17 0D313030 33313031
  32353231 325A170D 31313033 31303132 35323132 5A301A31 18301606 092A8648
  86F70D01 09021609 52455350 4F4E4445 52305C30 0D06092A 864886F7 0D010101
  0500034B 00304802 4100B517 EB8E64E1 B58CB014 07B3A6AF E6B69577 87486367
  9471B1DA BC66B847 DFA5073A 82121332 E787EA2D 3C433514 39033074 4095E7C7
  67A387A1 EBD24692 A76F0203 010001A3 4F304D30 0B060355 1D0F0404 030205A0
  301F0603 551D2304 18301680 144871D9 002C66DF D85FACB8 45D1D25F EA357455
  91301D06 03551D0E 04160414 DFF2401C 53276D96 89DE8C0A 786CCA71 C9EA792B
  300D0609 2A864886 F70D0101 04050003 8181002C 6E334273 CB832A95 3DDC6293
  669E416C A134D543 20952BC3 14A5C0B0 03AE011C 963AF523 C7C5C935 4FE9B2A5
  F24B3161 4D0D723A FA428BD1 85ADF172 B4007067 43C27D8A 1F74ED3D DEBE9F73
  1F515355 E77E766C AEACC303 39457991 29AB090C 99E21B5B 60DCB2C8 780B4479
  3EB3D46B B66C8C26 15311A7A B7A4ED97 32727C
        quit
 certificate ca 01
  30820201 3082016A A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  14311230 10060355 04031309 63612D73 65727665 72301E17 0D313030 33313031
  32343933 385A170D 31333033 30393132 34393338 5A301431 12301006 03550403
  13096361 2D736572 76657230 819F300D 06092A86 4886F70D 01010105 0003818D
  00308189 02818100 DA4ECE09 B998F670 598F32C1 7E9FA920 1D217AC4 293B842E
  7563CE11 B2F0F822 23077930 636C8293 00F6CFDD F6C9B0F5 8348BE58 6478F631
  7D44152F 494AEBCC A507FA6B 408D6BBB FAAB0A7A 2E7546A8 CA70F9A6 0F7F6824
  554BD833 060D657D ABDF406C 69EEF449 7A4F9AFE 6F0852E7 05DEDAC1 D433191E
  712868C2 A94E642B 02030100 01A36330 61300F06 03551D13 0101FF04 05300301
  01FF300E 0603551D 0F0101FF 04040302 0186301F 0603551D 23041830 16801448
  71D9002C 66DFD85F ACB845D1 D25FEA35 74559130 1D060355 1D0E0416 04144871
  D9002C66 DFD85FAC B845D1D2 5FEA3574 5591300D 06092A86 4886F70D 01010405
  00038181 00AFC36B 8A917284 06BD51CB 83BDC4E8 9457A361 6CAAF416 3BBEF691
  04215AC5 EDBC5730 C071C2FB 8A6C90CF D6AB39C2 3BC2147F D35553D9 028B2155
  802E50DB 48CDE067 B3857447 89A1C733 D81EFEF7 1115480F 70ED2F22 F27E35A1
  F3BB597C 7C8F717B FAAD79D3 0F469702 DE9190E4 B1B0808E 46A118EB 887CEAEB
  DFE2900E D2
        quit
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
!
crypto ikev2 profile prof
 match fvrf any
 match certificate cmap-2
 identity local dn
 authentication local rsa-sig
 authentication remote pre-share
 authentication remote rsa-sig
 pki trustpoint ca-server
!
!
crypto ipsec transform-set trans esp-aes-cbc-128 esp-sha-hmac
!
crypto dynamic-map dmap 1
```

```
 set transform-set trans
 set ikev2-profile prof
!
!
crypto map cmap 1 ipsec-isakmp dynamic dmap
interface Loopback0
 ip address 209.165.200.230 255.255.255.224
!
interface Ethernet0/0
 ip address 209.165.200.231 255.255.255.224
 crypto map cmap
!
interface Ethernet1/0
 ip address 209.165.200.232 255.255.255.224
!
ip route 209.165.200.233 255.255.255.224 209.165.200.228
!
ip access-list extended ikev2list
 permit ip host 209.165.200.231 host 209.165.200.228
```

The CA server configuration is as follows:

```
crypto pki server ca-server
 grant auto
!
crypto pki trustpoint ca-server
 revocation-check crl
 rsakeypair ca-server
!
!
crypto pki certificate chain ca-server
 certificate ca 01
  30820201 3082016A A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  14311230 10060355 04031309 63612D73 65727665 72301E17 0D303930 33303831
  36333335 395A170D 31323033 30373136 33333539 5A301431 12301006 03550403
  13096361 2D736572 76657230 819F300D 06092A86 4886F70D 01010105 0003818D
  00308189 02818100 99750598 EF4AF8B4 823DEF66 2F3BBA31 81C2DC5F D9B4040B
  99FB6020 22243CD6 B9F24C84 A543D7DB DD0B3018 2E36208C D0FD4015 EAF0DA69
  C1B0302B 87CEC34B 8646593F 0185AF02 0B86A3F3 5E5C3880 A992CD4A 79F13403
  411CC61F 07CEB4D9 0E967CB2 FAE0A899 5A3B6C87 73111F06 128465DA A45291F8
  F828C5DC 657487E7 02030100 01A36330 61300F06 03551D13 0101FF04 05300301
  01FF300E 0603551D 0F0101FF 04040302 0186301F 0603551D 23041830 1680147B
  D032BFB7 B3F70F1A 597B7C1E 1B42E472 5CCD6030 1D060355 1D0E0416 04147BD0
  32BFB7B3 F70F1A59 7B7C1E1B 42E4725C CD60300D 06092A86 4886F70D 01010405
  00038181 003838FA 628804EF E9FF69D9 3D5E299C 29074B2C AE33A563 8AF75976
  78FB68D4 5EF1E27B 04936FDF 78A09432 5348849D F79E17F5 70B233C9 2C1535D0
  506F0C35 99335012 84BBA3DC 050FD3C9 6E7B1D63 41ACC2B5 2B02432D BA2CC2CF
  E379DEA0 A9C208AC 0BEBB2D8 E6488815 EB12F1E0 19072D55 D5D11A49 739144D8
  271A842E ED
        quit
!
interface Ethernet1/0
 ip address 209.165.200.232 255.255.255.224
!
ip http server
```

To obtain the CA and device certificates, enter the **crypto pki authenticate ca-server** and **crypto pki enroll ca-server** commands. To initiate a connection between the initiator and the responder, enter the following command at the initiator's CLI:

```
ping 209.165.200.230 source 209.165.200.226
```

The output of the command is as follows:

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.230, timeout is 2 seconds:
Packet sent with a source address of 209.165.200.226
%IKEV2-5-OSAL_INITIATE_TUNNEL: Received request to establish an IPsec tunnel; local
traffic selector = Address Range: 209.165.200.226-209.165.200.226 Protocol: 1 Port Range:
0-65535; remote traffic selector = Address Range: 209.165.200.230-209.165.200.230
Protocol: 1 Port Range: 0-65535
```

```
%IKEV2-5-SA_UP: SA UP
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 8/11/12 ms
```

Enter the following **show** commands in the responder's CLI to display the session details:

```
show crypto session
Crypto session current status
Interface: Ethernet0/0
Session status: UP-ACTIVE
Peer: 1.1.1.1 port 500
  IKEv2 SA: local 209.165.200.231/500 remote 209.165.200.227/500 Active
  IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 host 209.165.200.226
        Active SAs: 2, origin: dynamic crypto map
show crypto ikev2 sa detailed
Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         209.165.200.231/500    209.165.200.227/500     (none)/
(none)                                                                    READY
      Encr: AES-CBC, Keysize: 128, Hash: SHA96, DH Grp:14, Auth sign: RSA, Auth verify:
RSA
      Life/Active Time: 86400/846 sec
      CE id: 1001, Session-id: 1
      Status Description: Negotiation done
      Local spi: F79756E978ED41C7       Remote spi: 188FB9A119516D34
      Local id: hostname=RESPONDER
      Remote id: hostname=INITIATOR
      Local req msg id:  0              Remote req msg id:  2
      Local next msg id: 0              Remote next msg id: 2
      Local req queued:  0              Remote req queued:  2
      Local window:      5              Remote window:      5
      DPD configured for 0 seconds, retry 0
      NAT-T is not detected
```

Example: Configuring Crypto Map- and dVTI-Based IKEv2 Peers

The following example shows how to configure crypto map-and dVTI-based IKEv2 peers using the preshared key authentication method between a static crypto map IKEv2 initiator and a dVTI-based IKEv2 responder. The initiator configuration is as follows:

```
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 keyring v2-kr1
 peer abc
  address 0.0.0.0 0.0.0.0
  pre-shared-key abc
 !
!
!
crypto ikev2 profile prof
 match fvrf any
 match identity remote address 0.0.0.0
 authentication local pre-share
 authentication remote pre-share
 keyring v2-kr1
!
crypto ipsec transform-set trans esp-aes-cbc-128 esp-sha-hmac
!
crypto map cmap 1 ipsec-isakmp
```

```
 set peer 206.165.200.235
 set transform-set trans
 set ikev2-profile prof
 match address ikev2list
!
interface Loopback0
 ip address 206.165.200.226 255.255.255.224
!
interface Ethernet0/0
 ip address 206.165.200.227 255.255.255.224
 crypto map cmap
!
ip route 206.165.200.229 255.255.255.224 206.165.200.235
!
ip access-list extended ikev2list
 permit ip host 206.165.200.227 host 206.165.200.235
 permit ip 206.165.200.233 255.255.255.224 206.165.200.229 255.255.255.224
```

The responder configuration is as follows:

```
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 keyring v2-kr1
 peer cisco
  address 0.0.0.0 0.0.0.0
  pre-shared-key cisco
 !
!
!
crypto ikev2 profile prof
 match fvrf any
 match identity remote address 0.0.0.0
 authentication local pre-share
 authentication remote pre-share
 keyring v2-kr1
 virtual-template 1
!
crypto ipsec transform-set set esp-aes-cbc-128 esp-sha-hmac
!
crypto ipsec profile vi
 set transform-set set
 set ikev2-profile prof
!
interface Loopback0
 ip address 206.165.200.230 255.255.255.224
!
interface Ethernet0/0
 ip address 206.165.200.235 255.255.255.224
!
interface Virtual-Template1 type tunnel
 ip unnumbered Ethernet0/0
 ip mtu 1000
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile vi
!
```

To initiate a connection between the initiator and the responder, enter the following command at the initiator's CLI:

```
ping 206.165.200.230 source 206.165.200.226
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 206.165.200.230, timeout is 2 seconds:
Packet sent with a source address of 206.165.200.226
%IKEV2-5-OSAL_INITIATE_TUNNEL: Received request to establish an IPsec tunnel; local
```

```
traffic selector = Address Range: 206.165.200.226-206.165.200.226 Protocol: 1 Port Range:
0-65535; remote traffic selector = Address Range: 206.165.200.230-206.165.200.230
Protocol: 1 Port Range: 0-65535
%IKEV2-5-SA_UP: SA UP
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 8/11/12 ms
```

Enter the following **show** command in an Easy VPN server to display the session details:

```
show crypto session
Crypto session current status
Interface: Virtual-Access2
Session status: UP-ACTIVE
Peer: 206.165.200.227 port 500
  IKEv2 SA: local 206.165.200.235/500 remote 206.165.200.227/500 Active
  IPSEC FLOW: permit ip 206.165.200.229/255.255.255.224 206.165.200.233/255.255.255.224
        Active SAs: 2, origin: crypto map
show crypto ikev2 sa detail
Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         206.165.200.235/500
206.165.200.227/500                          (none)/(none)        READY
      Encr: AES-CBC, Keysize: 128, Hash: SHA96, DH Grp: 14, Auth sign: PSK, Auth verify:
PSK
      Life/Active Time: 86400/8 sec
      CE id: 1001, Session-id: 1
      Status Description: Negotiation done
      Local spi: 305F610F57428834       Remote spi: D9D183B5689AEDCD
      Local id: 206.165.200.235
      Remote id: 206.165.200.227
      Local req msg id:  0              Remote req msg id:  2
      Local next msg id: 0              Remote next msg id: 2
      Local req queued:  0              Remote req queued:  2
      Local window:      5              Remote window:      5
      DPD configured for 0 seconds, retry 0
      NAT-T is not detected
show crypto route
VPN Routing Table: Shows RRI and VTI created routes
Codes: RRI - Reverse-Route, VTI- Virtual Tunnel Interface
        S - Static Map ACLs
Routes created in table GLOBAL DEFAULT
206.165.200.233/255.255.255.224 [1/0] via 206.165.200.227 tag 0
                                on Virtual-Access2 RRI
```

Example: Configuring IPsec Using sVTI-Based IKEv2 Peers

The following example shows how to configure IPsec using the preshared key authentication method between an sVTI IKEv2 initiator and an sVTI IKEv2 responder. The initiator configuration is as follows:

```
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 keyring v2-kr1
 peer abc
  address 209.165.200.225
  pre-shared-key abc
 !
!
!
crypto ikev2 profile prof
 match fvrf any
 match identity remote address 209.165.200.231 255.255.255.224
 authentication local pre-share
 authentication remote pre-share
```

```
 keyring v2-kr1
!
!
crypto ipsec transform-set trans esp-aes-cbc-128 esp-sha-hmac
!
crypto ipsec profile ipsecprof
 set transform-set trans
 set ikev2-profile prof
!
interface Loopback0
 ip address 209.165.200.226 255.255.255.224
!
interface Tunnel0
 ip address 10.0.0.1 255.255.255.0
 tunnel source 209.165.200.231
 tunnel mode ipsec ipv4
 tunnel destination 209.165.200.225
 tunnel protection ipsec profile ipsecprof
!
interface Ethernet0/0
 ip address 209.165.200.231 255.255.255.224
!
ip route 209.165.200.229 255.255.255.224 Tunnel0
!
```

The responder configuration is as follows:

```
crypto ikev2 proposal prop-1
 encryption aes-cbc-128
 integrity sha1
 group 14
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 keyring v2-kr1
 peer abc
  address 209.165.200.231
  pre-shared-key abc
 !
!
!
crypto ikev2 profile prof
 match fvrf any
 match identity remote address 209.165.200.231 255.255.255.224
 authentication local pre-share
 authentication remote pre-share
 keyring v2-kr1
!
!
crypto ipsec transform-set trans esp-aes-cbc-128 esp-sha-hmac
!
crypto ipsec profile ipsecprof
 set transform-set trans
 set ikev2-profile prof
!
crypto map cmap 1 ipsec-isakmp dynamic dmap
!
interface Loopback0
 ip address 209.165.200.230 255.255.255.224
!
interface Tunnel0
 ip address 10.0.0.2 255.255.255.0
 tunnel source 209.165.200.225
 tunnel mode ipsec ipv4
 tunnel destination 209.165.200.231
 tunnel protection ipsec profile ipsecprof
!
interface Ethernet0/0
 ip address 209.165.200.231 255.255.255.224
!
ip route 209.165.200.233 255.255.255.224 Tunnel0
```

With sVTI on IKEv2 peers, the session is initiated only when the sVTI interfaces are enabled. In other words, network traffic is not required to initiate the session. To verify the traffic between the initiator and the responder, enter the following command at the initiator's CLI:

```
ping 209.165.200.230 source 209.165.200.226
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.230, timeout is 2 seconds:
Packet sent with a source address of 209.165.200.226
%IKEV2-5-OSAL_INITIATE_TUNNEL: Received request to establish an IPsec tunnel; local
traffic selector = Address Range: 209.165.200.226-209.165.200.226 Protocol: 1 Port Range:
0-65535; remote traffic selector = Address Range: 209.165.200.230-209.165.200.23
Protocol: 1 Port Range: 0-65535
%IKEV2-5-SA_UP: SA UP
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 8/11/12 ms
```

Enter the following **show** command in the initiator's CLI to display the session details:

```
show crypto session
Crypto session current status
Interface: Ethernet0/0
Session status: UP-ACTIVE
Peer: 209.165.200.225 port 500
  IKEv2 SA: local 209.165.200.231/500 remote 209.165.200.225/500 Active
  IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
        Active SAs: 2, origin: crypto map
show crypto ikev2 sa detailed
Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         209.165.200.231/500
209.165.200.225/500                                          (none)/(none)        READY
      Encr: AES-CBC, Keysize: 128, Hash: SHA96, DH Grp: 14, Auth sign: PSK, Auth verify:
PSK
      Life/Active Time: 86400/21 sec
      CE id: 1002, Session-id: 2
      Status Description: Negotiation done
      Local spi: 687752902752A6FD       Remote spi: C9DCCFC65493D14F
      Local id: smap-initiator
      Remote id: dmap-responder
      Local req msg id:  2              Remote req msg id:  0
      Local next msg id: 2              Remote next msg id: 0
      Local req queued:  2              Remote req queued:  0
      Local window:      5              Remote window:      5
      DPD configured for 0 seconds, retry 0
      NAT-T is not detected
```

Example: Configuring IKEv2 on DMVPN Networks

DMVPN uses a tunnel protection CLI that is identical between IKEv1 and IKEv2. The IPsec profile applied on a DMVPN tunnel only refers to an IKEv2 profile. The DMVPN Hub configuration is as follows:

```
crypto ikev2 keyring cisco-ikev2-keyring
 peer dmvpn-node
 description symmetric pre-shared key for the hub/spoke
 address 0.0.0.0 0.0.0.0
 pre-shared-key cisco123
crypto ikev2 profile cisco-ikev2-profile
 keyring cisco-ikev2-keyring
 authentication pre-shared
 match local address 0.0.0.0
crypto ipsec profile cisco-ipsec-ikev2
 set transform-set cisco-ts
 set ikev2-profile cisco-ikev2-profile
! interface Tunnel 0
 description This is the Legacy IKEv1 facing tunnel on the hub
 ip address 1.1.1.99 255.255.255.0
 no ip redirects
 ip nhrp map multicast dynamic
```

```
 ip nhrp network-id 99
 ip nhrp redirect
 no ip split-horizon eigrp 1
 tunnel source Ethernet0/0
 tunnel mode gre multipoint
 tunnel protection ipsec profile cisco-ipsec
 !
interface Tunnel1
 description This would be the new IKEv2 facing tunnel on the hub
 ip address 2.2.2.99 255.255.255.0
 no ip redirects
 ip nhrp map multicast dynamic
 ip nhrp network-id 100
 no ip split-horizon eigrp 1
 tunnel source Ethernet0/1
 tunnel mode gre multipoint
 tunnel protection ipsec profile cisco-ipsec-ikev2
```

The IKEv2 configuration is as follows:

```
crypto ikev2 profile cisco-ikev2-profile
 keyring cisco-ikev2-keyring
 authentication pre-shared
 match local address 0.0.0.0
crypto ipsec profile cisco-ipsec-ikev2
 set transform-set cisco-ts
 set ikev2-profile cisco-ikev2-profile
interface Tunnel1
 ip address 2.2.2.11 255.255.255.0
 no ip redirects
 ip nhrp map 2.2.2.99 22.22.22.99
 ip nhrp map multicast 22.22.22.99
 ip nhrp network-id 100 ? Keep this same for all IKEv2 spokes for clarity
 ip nhrp nhs 2.2.2.99 ? This points to the hub's IKEv2 facing interface
 tunnel source Ethernet0/1
 tunnel mode gre multipoint
 tunnel protection ipsec profile cisco-ipsec-ikev2
```