Configuring the FlexVPN Server

This module describes FlexVPN server features, IKEv2 commands required to configure the FlexVPN server, remote access clients, and the supported RADIUS attributes.

Note Security threats, as well as cryptographic technologies to help protect against such threats, 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.

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

Restrictions for the FlexVPN Server

Dual-Stack Tunnel Interface and VRF-Aware IPsec

When configuring a dual-stack tunnel interface in a VPN routing and forwarding (VRF)-aware IPsec scenario, you cannot use the **ip vrf forwarding** command to configure an Inside VPN routing and forwarding (IVRF)

instance because this is not a valid configuration. Use the **vrf forwarding** *vrf-name* command to define the IVRF of the tunnel interface, where the *vrf-name* argument is defined using the **vrf definition** command with IPv4 and IPv6 address families inside the definition.

SSO Restrictions

- The Cisco ASR 1000 Series Routers support stateful IPSec sessions on Embedded Services Processor (ESP) switchover. During ESP switchover, all IPSec sessions will stay up and no user intervention is needed to maintain IPSec sessions.
- For an ESP reload (no standby ESP), the SA sequence number restarts from 0. The peer router drops packets that do not have the expected sequence number. You may need to explicitly reestablish IPSec sessions to work around this issue for systems that have a single ESP after an ESP reload. Traffic disruption might happen over the IPSec sessions in such cases for the duration of the reload.
- The Cisco ASR 1000 Series Router currently does not support Stateful Switchover (SSO) IPSec sessions on Route Processors (RPs). The IPSec sessions will go down on initiation of the switchover, but will come back up when the new RP becomes active. No user intervention is needed. Traffic disruption might happen over the IPSec sessions for the duration of the switchover, until the sessions are back up.
- The Cisco ASR 1000 Series Router does not support stateful ISSU for IPSec sessions. Before performing an ISSU, you must explicitly terminate all existing IPSec sessions or tunnels prior to the operation and reestablish them post ISSU. Specifically, ensure that there are no half-open or half-established IPSec tunnels present before performing ISSU. To do this, we recommend a interface shutdown in the case of interfaces that may initiate a tunnel setup, such as a routing protocol initiating a tunnel setup, or interfaces that have keepalive enabled, or where there is an auto trigger for an IPSec session. Traffic disruption over the IPSec sessions during ISSU is obvious in this case.

Information About the FlexVPN Server

Peer Authentication Using EAP

The FlexVPN server supports peer authentication using the Extensible Authentication protocol (EAP) and acts as a pass-through authenticator relaying EAP messages between the client and the backend EAP server. The backend EAP server is typically a RADIUS server that supports EAP authentication.

Note While a FlexVPN client authenticates the FlexVPN client using EAP, the FlexVPN server must authenticated by using certificates.

The FlexVPN server is configured to authenticate FlexVPN clients that use EAP by configuring the **authentication remote eap** command in IKEv2 profile configuration mode. FlexVPN clients authenticate using EAP by skipping the AUTH payload in the IKE_AUTH request.

If the **query-identity** keyword is configured, the FlexVPN server queries the EAP identity from the client; otherwise, the FlexVPN client's IKEv2 identity is used as the EAP identity. However, if the **query-identity** keyword is not configured and the FlexVPN client's IKEv2 identity is an IPv4 or IPv6 address, the session is terminated because IP addresses cannot be used as the EAP identity.

The FlexVPN server starts the EAP authentication by passing the FlexVPN client's EAP identity to the EAP server; the FlexVPN server then relays EAP messages between the remote access (RA) client and the EAP server until the authentication is complete. If the authentication succeeds, the EAP server is expected to return the authenticated EAP identity to the FlexVPN server in the EAP success message.

After EAP authentication, the EAP identity used for the IKEv2 configuration is obtained from the following sources in the given order:

- The EAP identity provided by the EAP server with the EAP success message.
- The EAP identity queried from the client when the **query-identity** keyword is configured.
- The FlexVPN client IKEv2 identity used as the EAP identity.

The figure below shows IKEv2 exchange for EAP authentication without the **query-identity** keyword.

Figure 1: IKEv2 Exchange Without the query-identity Keyword

IKEv2 RA client	IKEv2 RA server	RADIUS-EAP server
HDR, SAi1, KEi, Ni →		
	← HDR, SAr1, KEr, Nr, [CERTREQ]	
HDR, SK {IDi, [CERTREQ,] [IDr,] SAi2, TSi, TSr} →		
	RADIUS Access-Request/ EAP-Message/EAP-Response/(EAP-ID(IKEv2-ID)) →	
		← RADIUS Access-Challenge/EAP-Message/ EAP-Request/(EAP-method)
	← HDR, SK {IDr, [CERT,] AUTH, EAP(EAP-Request(EAP-method))}	
HDR, SK {EAP(EAP-Response(EAP-method)} →		
	RADIUS Access-Request/EAP-Message/EAP-Response/EAP-method) →	
		← RADIUS Access-Accept/EAP-Message/EAP-Success (other attributes))
	← HDR, SK {EAP (success)}	
HDR, SK {AUTH} →		
	← HDR, SK {AUTH, SAr2, TSi, TSr }	

209140

The figure below shows the IKEv2 exchange for EAP authentication with the **query-identity** keyword.

Figure 2: IKEv2 Exchange with the query-identity Keyword

IKEv2 RA client	IKEv2 RA server	RADIUS-EAP server
HDR, SAi1, KEi, Ni →		
	← HDR, SAr1, KEr, Nr, [CERTREQ]	
HDR, SK {IDi, [CERTREQ,] [IDr,] SAi2, TSi, TSr} →		
	← HDR, SK {IDr, [CERT,] AUTH, EAP (EAP-request (Identity)) }	
HDR, SK {EAP(EAP-Response(Identity)} →		
	RADIUS Access-Request/ EAP-Message/EAP-Response/(EAP-ID) →	
		← RADIUS Access-Challenge/EAP-Message/ EAP-Request/(EAP-method)
	← HDR, SK {IDr, [CERT,] AUTH, EAP(EAP-Request(EAP-method))}	
HDR, SK {EAP(EAP-Response(EAP-method)} →		
	RADIUS Access-Request/EAP-Message/EAP-Response/EAP-method) →	
		← RADIUS Access-Accept/EAP-Message/EAP-Success (EAP-identity) (other attributes))
	← HDR, SK {EAP (success)}	
HDR, SK {AUTH} →		
	← HDR, SK {AUTH, SAr2, TSi, TSr }	

209141

IKEv2 Configuration Mode

IKEv2 configuration mode allows IKE peers to exchange configuration information such as IP addresses and routes. The configuration information is obtained from IKEv2 authorization. Both pull and push models are supported. The pull model involves the exchange of configuration requests and replies; the push model involves the exchange of configuration sets and acknowledgements.

The following table describes the conditions when the initiator and the responder send different configuration payload types:

Table 1: Configuration Payload Types

Configuration Payload Type	Sent By...	When...
CFG_REQUEST	Initiator	The initiator is the FlexVPN client or if the **config-exchange request** command is enabled in the IKEv2 profile.
CFG_REPLY	Responder	The responder receives the CFG_REQUEST.
CFG_SET	Initiator and responder	Initiator—The **config-exchange set send** command is enabled in the IKEv2 profile. Responder—The CFG_REQUEST is not received, the configuration data is available, and the **config-exchange set send** command is enabled in the IKEv2 profile.
CFG_ACK	Initiator and responder	Initiator—The **config-exchange set accept** command is enabled in the IKEv2 profile. Responder—The **config-exchange set accept** command is enabled in the IKEv2 profile.

Note The commands to send configuration requests and configuration set payloads are enabled by default.

Depending on your release, the IKEv2 initiator can trigger a configuration mode when the initiator is a FlexVPN client, or any static tunnel interface initiating IKEv2 can trigger configuration mode by enabling the **config-mode** command in the IKEv2 profile.

The IKEv2 FlexVPN server supports the following standard IPv4 configuration attributes:

- INTERNAL_IP4_ADDRESS
- INTERNAL_IP4_NETMASK
- INTERNAL_IP4_DNS
- INTERNAL_IP4_NBNS
- INTERNAL_IP4_SUBNET

The IKEv2 FlexVPN server supports the following standard IPv6 configuration attributes:

- INTERNAL_IP6_ADDRESS
- INTERNAL_IP6_DNS
- INTERNAL_IP6_SUBNET

Note IPv6 configuration attributes are only supported by the Microsoft Windows IKEv2 client.

The INTERNAL_IP4_SUBNET and INTERNAL_IP6_SUBNET configuration attributes, controlled by the **route set** and **aaa attribute list** commands in the IKEv2 authorization policy, are not supported when you configure a static virtual tunnel interface (SVTI)-to-SVTI tunnel. In such cases, static routing or dynamic routing must be used instead of the IKEv2-based route exchange.

The IKEv2 FlexVPN server supports the following standard common configuration attribute:

- APPLICATION_VERSION

Note This attribute is only sent for Cisco Anyconnect and FlexVPN clients.

The IKEv2 FlexVPN server supports the following Cisco Unity configuration attributes:

- MODECFG_BANNER
- MODECFG_DEFDOMAIN
- MODECFG_SPLITDNS_NAME
- MODECFG_BACKUPSERVERS
- MODECFG_PFS
- MODECFG_SMARTCARD_REMOVAL_DISCONNECT

Note The Cisco Unity attributes are sent only for Cisco Anyconnect and FlexVPN clients.

The IKEv2 FlexVPN server supports the following Cisco FlexVPN configuration attributes:

- MODECFG_CONFIG_URL
- MODECFG_CONFIG_VERSION

Note The Cisco FlexVPN attributes are sent only for Cisco FlexVPN clients.

The INTERNAL_IP4_ADDRESS attribute value is derived from the following sources in the given order:

- The Framed-IP-Address attribute received in AAA user authorization.
- The local IP address pool.
- The DHCP server.

The DHCP server, if configured, allocates addresses only if the local IP address pool is not configured. However, if an error occurs when allocating IP addresses from the local pool, the next address source DHCP server is not used for allocating the addresses.

The value for INTERNAL_IP4_NETMASK attribute is derived as follows:

- If the IP address is obtained from the DHCP server, the netmask is also obtained from the DHCP server.

- If the IP address is obtained from either the Framed-IP-Address attribute in AAA user authorization or the local IP address pool, the netmask is derived from the IPv4 netmask attribute received in the user or group authorization. If the netmask is not available, the INTERNAL_IP4_NETMASK attribute is not included in the configuration reply. If the netmask is available, the INTERNAL_IP4_NETMASK attribute is included only if the INTERNAL_IP4_ADDRESS attribute is included in the configuration reply.

An IPv4 address is allocated and included in the reply only if the client requests an address. If the client requests multiple IPv4 addresses, only one IPv4 address is sent in the reply. If available, the remaining attributes are included in the reply even though the client does not request them. If the client requests an IPv4 address and the FlexVPN server is unable to assign an address, an INTERNAL_ADDRESS_FAILURE message is returned to the client.

It is always recommended that the prefix length should be used as 128 on ipv6 local pool configuration.

For example, if clients are 4 , **ipv6 local pool pool1 afe0::/126 128** needed to be configured for the prefix length. If clients are 16, **ipv6 local pool pool1 afe0::/124 128** needed to be configured for the prefix length.

IKEv2 Authorization

IKEv2 authorization provides a policy for an authenticated session by using the AAA. The policy can be defined locally or on the RADIUS server, and contains local and/or remote attributes. The username for authorization can either be derived from the peer identity using the **name-mangler** keyword or be directly specified in the command. IKEv2 authorization is mandatory only if the peer requests an IP address via configuration mode.

IKEv2 authorization types are as follows:

- User authorization—Use the **aaa authorization user** command in the IKEv2 profile to enable user authorization. User authorization is based on the user-specific portion of the peer IKE identity such as fqdn-hostname. The attributes from user authorization are called user attributes.
- Group authorization—Use the **aaa authorization group** command in the IKEv2 profile to enable group authorization. Group authorization is based on the generic portion of the peer IKE identity such as fqdn-domain. The attributes from group authorization are called group attributes.
- Implicit user authorization—Use the **aaa authorization user cached** command in the IKEv2 profile to enable implicit user authorization. Implicit authorization is performed as part of EAP authentication or when obtaining the AAA preshared key. The attributes from implicit user authorization are called cached attributes.

Note Depending on your release, the **aaa authorization user cached** command may or may not be available. Explicit user authorization is performed only when implicit user authorization does not return any attributes or does not have the Framed-IP-Address attribute.

Merging and Overriding Attributes

Attributes from different sources are merged before they are used. The precedence of merging attributes is as follows:

- When merging duplicate attributes, the source of the attribute has a higher precedence.
- When merging user and cached attributes, user attributes have higher precedence.

- When merging merged-user-attributes and group attributes, merged-user attributes have a higher precedence, by default. However, this precedence can be reversed using the **aaa author group override** command.

IKEv2 Authorization Policy

An IKEv2 authorization policy defines the local authorization policy and contains local and/or remote attributes. Local attributes, such as VPN routing and forwarding (VRF) and the QOS policy, are applied locally. Remote attributes, such as routes, are pushed to the peer via the configuration mode. Use the **crypto ikev2 authorization policy** command to define the local policy. The IKEv2 authorization policy is referred from the IKEv2 profile via the **aaa authorization** command.

IKEv2 Name Mangler

The IKEv2 name mangler is used to derive the username for IKEv2 authorization and obtain the AAA preshared key from the peer IKE identity.

IKEv2 Multi-SA

The IKEv2 Multi-SA feature allows an IKEv2 Dynamic Virtual Tunnel Interface (DVTI) session on the IKEv2 responder to support multiple IPsec Security Associations (SA). The maximum number of IPsec SAs per DVTI session is either obtained from AAA authorization or configured on the IPsec profile. The value from AAA has a higher priority. Any change to the *max-flow-limit* argument in the IPsec profile is not applied to the current session but is applied to subsequent sessions. The IKEv2 Multi-SA feature makes the configuration of the IKEv2 profile in the IPsec profile optional. This optional configuration allows IPsec DVTI sessions using the same virtual template to have different IKEv2 profiles, thus saving the number of virtual template configurations.

Note The IKEv2 Multi-SA feature allows multiple IPsec SAs that have non-any-any proxies. However, when the IPsec SA proxies are any-any, a single IPsec SA is allowed.

For more information, see the "Multi-SA Support for Dynamic Virtual Tunnel Interfaces for IKEv2" module in the *Security for VPNs with IPsec Configuration Guide*.

Supported RADIUS Attributes

The following tables list the RADIUS attributes supported by the IKEv2 FlexVPN server:

- The Scope field defines the direction of the attribute and the usage on the FlexVPN server or client.
  - Inbound—FlexVPN server to RADIUS
  - Outbound—RADIUS to the FlexVPN server
  - Local—Used locally by the FlexVPN server
  - Remote—Pushed to the client by the FlexVPN server

- The “Local configuration” field specifies the IKEv2 authorization policy command that is used to configure the attribute locally on the FlexVPN server.
- Cisco AV Pair is a Cisco Vendor Specific Attribute (VSA) with vendor-id 9 and vendor-type 1. The VSAs are encapsulated in the Radius IETF attribute 26 Vendor-Specific. The Cisco AV pair is specified as a string of format “protocol:attribute=value”.

Example:

```
cisco-avpair = “ipsec:ipv6-addr-pool=v6-pool”
```

The following example shows the Cisco AV pair for a standard access-list.

```
cisco-avpair = "ipsec:route-set=access-list 99"
```

Table 2: Inbound and Bidirectional IETF RADIUS Attributes

Attribute	Scope
User-Name	Inbound and outbound (bidirectional)
User-Password	Inbound
Calling-Station-Id	Inbound
Service-Type	Inbound
EAP-Message	Bidirectional
Message-Authenticator	Bidirectional

Table 3: Outbound IETF and Cisco AV Pair RADIUS Attributes

Attribute	Type	Scope	Local configuration
Tunnel-Type	IETF	Local	N/A
Tunnel-Medium-Type	IETF	Local	N/A
Tunnel-Password	IETF	Local	N/A
ipsec:ikev2-password-local	Cisco AV Pair	Local	N/A
ipsec:ikev2-password-remote	Cisco AV Pair	Local	N/A
ipsec:addr-pool	Cisco AV Pair	Local	pool
ipsec:group-dhcp-server	Cisco AV Pair	Local	dhcp server
ipsec:dhcp-giaddr	Cisco AV Pair	Local	dhcp giaddr
ipsec:dhcp-timeout	Cisco AV Pair	Local	dhcp timeout
ipsec:ipv6-addr-pool	Cisco AV Pair	Local	ipv6 pool
ipsec:route-set=interface	Cisco AV Pair	Local	route set interface

Attribute	Type	Scope	Local configuration
ipsec:route-set=prefix	Cisco AV Pair	Local	N/A
ipsec:route-accept	Cisco AV Pair	Local	route accept any
ip:interface-config	Cisco AV Pair	Local	aaa attribute list
ipsec:ipsec-flow-limit	Cisco AV Pair	Local	ipsec flow-limit
Framed-IP-Address	IETF	Remote	N/A
Framed-IP-Netmask	IETF	Remote	netmask
ipsec:dns-servers	Cisco AV Pair	Remote	DNS
ipsec:wins-servers	Cisco AV Pair	Remote	wins
ipsec:route-set=access-list (See Note 1.)	Cisco AV Pair	Remote	route set access-list (See Note 1.)
ipsec:addrv6	Cisco AV Pair	Remote	n/a
ipsec:prefix-len	Cisco AV Pair	Remote	n/a
ipsec:ipv6-dns-servers-addr	Cisco AV Pair	Remote	ipv6 dns
ipsec:route-set=access-list ipv6	Cisco AV Pair	Remote	route set access-list ipv6
ipsec:banner	Cisco AV Pair	Remote	banner
ipsec:default-domain	Cisco AV Pair	Remote	def-domain
ipsec:split-dns	Cisco AV Pair	Remote	split-dns
ipsec:ipsec-backup-gateway	Cisco AV Pair	Remote	backup-gateway
ipsec:pfs	Cisco AV Pair	Remote	pfs
ipsec:include-local-lan	Cisco AV Pair	Remote	include-local-lan
ipsec:smartcard-removal-disconnect	Cisco AV Pair	Remote	smartcard-removal- disconnect
ipsec:configuration-url	Cisco AV Pair	Remote	configuration url
ipsec:configuration-version	Cisco AV Pair	Remote	configuration version

Note

- 1. The RADIUS attribute to set an access list on IKEv2 FlexVPN server only supports a standard access list. An extended access list is not supported.

Supported Remote Access Clients

The FlexVPN server interoperates with the Microsoft Windows7 IKEv2 client, Cisco IKEv2 AnyConnect client, and Cisco FlexVPN client.

Microsoft Windows7 IKEv2 Client

The Microsoft Windows 7 IKEv2 client sends an IP address as the Internet Key Exchange (IKE) identity that prevents the Cisco IKEv2 FlexVPN server from segregating remote users based on the IKE identity. To allow the Windows 7 IKEv2 client to send the email address (user@domain) as the IKE identity, apply the hotfix documented in KB975488 (http://support.microsoft.com/kb/975488) on Microsoft Windows 7 and specify the email address string in either the Username field when prompted or the CommonName field in the certificate depending on the authentication method.

For certificate-based authentication, the FlexVPN server and Microsoft Windows 7 client certificates must have an Extended Key Usage (EKU) field as follows:

- For the client certificate, EKU field = client authentication certificate.
- For the server certificate, EKU field = server authentication certificate
- The certificates can be obtained from the Microsoft Certificate Server or the IOS CA server.

For EAP authentication, the Microsoft Windows 7 IKEv2 client expects an EAP identity request before any other EAP requests. Ensure that you configure the **query-identity** keyword in the IKEv2 profile on the IKEv2 FlexVPN server to send an EAP identity request to the client.

Cisco IKEv2 AnyConnect Client

For certificate-based authentication, the FlexVPN server and the AnyConnect client certificates must have an Extended Key Usage (EKU) field as follows:

- For the client certificate, EKU field = client authentication certificate
- For the server certificate, EKU field = server authentication certificate

If the FlexVPN server authenticates to AnyConnect client using certificates, a SubjectAltName extension is required in the FlexVPN server certificate that contains the server's IP address or fully qualified domain name (FQDN). Additionally, HTTP certified URLs must be disabled on the FlexVPN server using the **no crypto ikev2 http-url cert** command.

The following example displays the XML tags specific to EAP-MD5 authentication of IKEv2 sessions in the AnyConnect client profile:

```
<PrimaryProtocol>IPsec
       <StandardAuthenticationOnly>true
           <AuthMethodDuringIKENegotiation>
               EAP-MD5
           </AuthMethodDuringIKENegotiation>
           <IKEIdentity>DEPT24</IKEIdentity>
       </StandardAuthenticationOnly>
</PrimaryProtocol>
```

Note For every flap or FlexVPN tunnel that is enabled, the following message is displayed:

```
*Jan 22 22:52:09.833: %SYS-5-CONFIG_P: Configured programmatically by process Crypto INT
from console as console
*Jan 22 22:52:09.840: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access2,
changed state to up
```

For more information, refer to AnyConnect client 3.0 documentation at this link: http://www.cisco.com/en/US/docs/security/vpn_client/anyconnect/anyconnect30/release/notes/anyconnect30rn.html#wp1268255.

How to Configure the FlexVPN Server

Configuring the IKEv2 Profile for the FlexVPN Server

This task describes the IKEv2 profile commands required for configuring the FlexVPN server in addition to the basic IKEv2 profile commands. Refer to the "Configuring IKEv2 Profile (Basic)" task in the *Configuring Internet Key Exchange Version 2 (IKEv2)* feature module for information about configuring the basic IKEv2 profile.

Perform this task to configure the IKEv2 profile for the FlexVPN Server:

Step 1 **enable**

Example:

Enables privileged EXEC mode.

```
Device> enable
```

Enter your password, if prompted.

Step 2 **configure terminal**

Example:

Enters the global configuration mode.

```
Device# configure terminal
```

Step 3 **crypto ikev2 profile** *profile-name*

Defines an IKEv2 profile name and enters IKEv2 profile configuration mode.

Example:

```
Device(config)# crypto ikev2 profile profile1
```

Step 4 **aaa authentication eap** *list-name*

Example:

```
Device(config-ikev2-profile)# aaa authentication eap list1
```

(Optional) Specifies the AAA authentication list for the EAP authentication when implementing the IKEv2 remote access server.

- **eap**—Specifies the external EAP server.
- *list-name*—The AAA authentication list name.

Step 5 **authentication** {**local** {**rsa-sig** | **pre-share** [**key** {**0** | **6**} *password*}] | **ecdsa-sig** | **eap** [**gtc** | **md5** | **ms-chapv2**] [**username** *username*] [**password** {**0** | **6**} *password*}]} | **remote** {**eap** [**query-identity** | **timeout** *seconds*] | **rsa-sig** | **pre-share** [**key** {**0** | **6**} *password*}] | **ecdsa-sig**}}

Example:

```
Device(config-ikev2-profile)# authentication local ecdsa-sig
```

Specifies the local or remote authentication method.

- **rsa-sig**—Specifies RSA-sig as the authentication method.
- **pre-share**—Specifies the preshared key as the authentication method.
- **ecdsa-sig**—Specifies ECDSA-sig as the authentication method.
- **eap**—Specifies EAP as the remote authentication method.
- **query-identity**—Queries the EAP identity from the peer.
- **timeout** *seconds*—Specifies the duration, in seconds, to wait for the next IKE_AUTH request after sending the first IKE_AUTH response.

Note You can specify only one local authentication method but multiple remote authentication methods.

Step 6 Execute both or one of the following:

- **aaa authorization user** {**eap** | **psk**} {**cached** | **list** *aaa-listname* [*aaa-username* | **name-mangler** *mangler-name*]}
- **aaa authorization user cert list** *aaa-listname* {*aaa-username* | **name-mangler** *mangler-name*}

Example:

```
Device(config-ikev2-profile)# aaa authorization user eap cached
```

Example:

```
Device(config-ikev2-profile)# aaa authorization user cert list list1 name-mangler mangler1
```

Specifies the AAA method list and username for user authorization.

- **user**—Specifies user authorization.
- **cert**—Specifies that the peers must be authenticated using certificates.
- **eap**—Specifies that the peers must be authenticated using EAP.
- **psk**—Specifies that the peers must be authenticated using preshared keys.
- **cached**—Specifies that the attributes received during EAP authentication or obtained from the AAA preshared key must be cached.
- *aaa-listname*—AAA method list name.
- *aaa-username*—Specifies the username that must be used in the AAA authorization request.
- **name-mangler**—Specifies the name mangler that derives the AAA authorization username from the peer identity.
- *mangler-name*—Name mangler to be used.

Note
- For **psk** and **eap** authentication methods, specifying the *aaa-username* argument or the **name-mangler** keyword is optional and if not specified, the peer identity is used as the username.
- For **psk** and **eap** authentication methods, you can simultaneously configure two variants for user authorization with the **cached** and **list** keyword respectively.
- Specifying the *aaa-username* argument or the **name-mangler** keyword is mandatory for **cert** authentication, as the peer identity of type distinguished name (DN) cannot be used.

Step 7 Execute both or one of the following:

- **aaa authorization group** [**override**] {**eap** | **psk**} **list** *aaa-listname* [*aaa-username* | **name-mangler** *mangler-name*]
- **aaa authorization group** [**override**] **cert list** *aaa-listname* {*aaa-username* | **name-mangler** *mangler-name*}

Example:

```
Device(config-ikev2-profile)# aaa authorization group override psk list list1
```

Example:

```
Device(config-ikev2-profile)# aaa authorization group cert list list1 name-mangler mangler1
```

Specifies the AAA method list and username for group authorization.

- **group**—Specifies group authorization.
- **override**—(Optional) Specifies that attributes from group authorization should take precedence while merging attributes. By default, user attributes take precedence.
- **cert**—Specifies that peers must be authenticated using certificates.
- **eap**—Specifies that peers must be authenticated using EAP.
- **psk**—Specifies that peers must be authenticated using preshared keys.
- *aaa-listname*—AAA method list name.
- *aaa-username*—Username that must be used in the AAA authorization request.
- **name-mangler**—Specifies the name mangler that derives the AAA authorization username from the peer identity.
- *mangler-name*—Name mangler to be used.

Note
- For **psk** and **eap** authentication methods, specifying the *aaa-username* argument or the **name-mangler** keyword is optional and if not specified, the peer identity is used as the username.
- For **psk** and **eap** authentication methods, you can simultaneously configure two variants for user authorization with the **cached** and **list** keyword respectively.
- Specifying the *aaa-username* argument or the **name-mangler** keyword is mandatory for **cert** authentication, as the peer identity of type distinguished name (DN) cannot be used.

Step 8 **config-exchange** {**request** | **set** {**accept** | **send**}}

Example:

```
Device(config-ikev2-profile)# config-exchange set accept
```

(Optional) Enables configuration exchange options.

- **request**—Enables the configuration exchange request.

- **set**—Enables the configuration exchange request set options.
- **accept**—Accepts the configuration exchange request set.
- **send**—Enables sending of the configuration exchange set.

Note The request and set options are enabled by default.

Step 9 **end**

Example:

```
Device(config-ikev2-profile)# end
```

Exits IKEv2 profile configuration mode and returns to privileged EXEC mode.

Configuring the IKEv2 Name Mangler

Perform this task to specify the IKEv2 name mangler, which is used to derive a name for authorization requests and obtain AAA preshared keys. The name is derived from specified portions of different forms of remote IKE identities or the EAP identity. The name mangler specified here is referred to in the IKEv2 profile.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 name-mangler** *mangler-name*
4. **dn** {**common-name** | **country** | **domain** | **locality** | **organization** | **organization-unit** | **state**}
5. **eap** {**all** | **dn** {**common-name** | **country** | **domain** | **locality** | **organization** | **organization-unit** | **state**} | **prefix** | **suffix** {**delimiter** {**.** | **@** | \}}}
6. **email** {**all** | **domain** | **username**}
7. **fqdn** {**all** | **domain** | **hostname**}
8. **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 name-mangler** *mangler-name* **Example:** `Device(config)# crypto ikev2 name-mangler mangler1`	Defines a name mangler and enters IKEv2 name mangler configuration mode.

	Command or Action	Purpose
Step 4	**dn** {**common-name** | **country** | **domain** | **locality** | **organization** | **organization-unit** | **state**} **Example:** `Device(config-ikev2-name-mangler)# dn state`	Derives the name from any of the following fields in the remote identity of type DN (distinguished name). • **common-name** • **country** • **domain** • **locality** • **organization** • **organization-unit** • **state**

Step 5	**eap** {**all** | **dn** {**common-name** | **country** | **domain** | **locality** | **organization** | **organization-unit** | **state**} | **prefix** | **suffix** {**delimiter** {**.** | **@** | \}}}
	Example:
	`Device(config-ikev2-name-mangler)# eap prefix delimiter @`
		Derives the name from the remote identity of type EAP (Extensible Authentication Protocol).
		• **all**—Derives the name from the entire EAP identity.
		• **dn**—Derives the name from any of the following fields in the remote EAP identity of type DN:
		  • **common-name**
		  • **country**
		  • **domain**
		  • **locality**
		  • **organization**
		  • **organization-unit**
		  • **state**
		• **prefix**—Derives the name from the prefix in the EAP identity.
		• **suffix**—Derives the name from the suffix in the EAP identity.
		• **delimiter** {**.** | **@** | \}—Specifies the delimiter in the EAP identity that separates the prefix and the suffix.

Step 6	**email** {**all** | **domain** | **username**} **Example:** `Device(config-ikev2-name-mangler)# email username`	Derives the name from the remote identity of type e-mail. • **all**—Derives the name from the entire remote IKE identity of type e-mail. • **domain**—Derives the name from the domain part of the remote IKE identity. • **username**—Derives the name from the username part of the remote IKE identity.

	Command or Action	Purpose
Step 7	**fqdn** {**all** | **domain** | **hostname**} **Example:** `Device(config-ikev2-name-mangler)# fqdn domain`	Derives the name from the remote identity of type FQDN (Fully Qualified Domain Name). • **all**—Derives the name from the entire remote IKE identity of type FQDN. • **domain**—Derives the name from the domain part of the remote IKE identity. • **hostname**—Derives the name from the hostname part of the remote IKE identity.
Step 8	**end** **Example:** `Device(config-ikev2-name-mangler)# end`	Exits IKEv2 name mangler configuration mode and returns to privileged EXEC mode.

Configuring the IKEv2 Authorization Policy

Perform this task to configure the IKEv2 authorization policy.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 authorization policy** *policy-name*
4. **aaa attribute list** *list-name*
5. **backup-gateway** *string*
6. **banner** *banner-text*
7. **configuration url** *url*
8. **configuration version** *version*
9. **def-domain** *domain-name*
10. **dhcp** {**giaddr** *ip-address* \| **server** {*ip-address* \| *hostname*} \| **timeout** *seconds*}
11. [**ipv6**] **dns** *primary-server* [*secondary-server*]
12. **include-local-lan**
13. **ipsec flow-limit** *number*
14. **netmask** *mask*
15. **pfs**
16. [**ipv6**] **pool** *name*
17. **route set** {**interface** *interface* \| **access-list** {*access-list-name* \| *access-list-number* \| **ipv6** *access-list-name*}}
18. **route accept any** [**tag** *value*] [**distance** *value*]
19. **route redistribute** *protocol* [**route-map** *map-name*]
20. **route set remote** {**ipv4** *ip-address mask* \| **ipv6** *ip-address*/*mask*}
21. **smartcard-removal-disconnect**
22. **split-dns** *string*
23. **session-lifetime** *seconds*

24. **route set access-list** {*acl-number* | [**ipv6**] *acl-name*}
25. **wins** *primary-server* [*secondary-server*]
26. **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 authorization policy** *policy-name* **Example:** `Device(config)# crypto ikev2 authorization policy policy1`	Specifies the IKEv2 authorization policy and enters IKEv2 authorization policy configuration mode.
Step 4	**aaa attribute list** *list-name* **Example:** `Device(config-ikev2-author-policy)# aaa attribute list list1`	Specifies an AAA attribute list. **Note** The AAA attribute list referred to in this command should be defined in global configuration mode.
Step 5	**backup-gateway** *string* **Example:** `Device(config-ikev2-author-policy)# backup-gateway gateway1`	Allows you to specify up to ten backup server names. This parameter is pushed to the client via the nonstandard Cisco Unity configuration attribute. This parameter specifies the backup servers that the client can use.
Step 6	**banner** *banner-text* **Example:** `Device(config-ikev2-author-policy)# banner This is IKEv2`	Specifies the banner. This parameter is sent to the client via the nonstandard Cisco Unity configuration attribute.
Step 7	**configuration url** *url* **Example:** `Device(config-ikev2-author-policy)# configuration url http://www.cisco.com`	Specifies the configuration URL. This parameter is sent to the client via the nonstandard Cisco FlexVPN configuration attribute. The client can use this URL to download the configuration.
Step 8	**configuration version** *version* **Example:** `Device(config-ikev2-author-policy)# configuration version 2.4`	Specifies the configuration version. This parameter is sent to the client via the nonstandard Cisco FlexVPN configuration attribute. This parameter is sent with the configuration URL to specify the version that the client can download.
Step 9	**def-domain** *domain-name* **Example:**	Specifies the default domain. This parameter is sent to the client via the nonstandard Cisco Unity configuration

	Command or Action	Purpose
	`Device(config-ikev2-author-policy)# def-domain cisco`	attribute. This parameter specifies the default domain that the client can use.
Step 10	**dhcp** {**giaddr** *ip-address* | **server** {*ip-address* | *hostname*} | **timeout** *seconds*} **Example:** `Device(config-ikev2-author-policy)# dhcp giaddr 192.0.2.1`	Specifies the DHCP server to lease an IP address that is assigned to the remote access client. • **giaddr** *ip-address*—Specifies the gateway IP address (giaddr). • **server** {*ip-address* | *hostname*}—Specifies the IP address or hostname of the DHCP server. The hostname is resolved during configuration. • **timeout** *seconds*—Specifies the wait time in seconds for the response from the DHCP server. **Note** You can specify only one DHCP server. It is assumed that the DHCP server can be reached via the global routing table, and therefore, the DHCP packets are forwarded to the global routing table.
Step 11	[**ipv6**] **dns** *primary-server* [*secondary-server*] **Example:** `Device(config-ikev2-author-policy)# dns 198.51.100.1 198.51.100.100`	Specifies the IP addresses of primary and secondary Domain Name Service (DNS) servers that are sent to the client in the configuration reply. • **ipv6**—(Optional) Specifies an IPv6 address for the DNS server. To specify an IPv4 address, execute the command without this keyword. • *primary-server*—IP address of the primary DNS server. • *secondary-server*—(Optional) IP address of the secondary DNS server.
Step 12	**include-local-lan** **Example:** `Device(config-ikev2-author-policy)# include-local-lan`	Includes local LAN. This parameter is sent to the client via the nonstandard Cisco Unity configuration attribute.
Step 13	**ipsec flow-limit** *number* **Example:** `Device(config-ikev2-author-policy)# ipsec flow-limit 12500`	Specifies the maximum number of IPsec SAs that an IKev2 dVTI session on the IKev2 responder can have. The range is from 0 to 50000. By default, the command is disabled, and there is no limit on the number of IPsec flows per dVTI session. A value of 0 will not allow any IPsec SAs.
Step 14	**netmask** *mask* **Example:**	Specifies the netmask of the subnet from which the IP address is assigned to the client. • *mask*—Subnet mask address.

	Command or Action	Purpose
	`Device(config-ikev2-author-policy)# netmask 255.255.255.0`	
Step 15	**pfs** **Example:** `Device(config-ikev2-author-policy)# pfs`	Enables Password Forward Secrecy (PFS). This parameter is sent to the client via the nonstandard Cisco Unity configuration attribute. This parameter specifies whether the client should use PFS.
Step 16	[**ipv6**] **pool** *name* **Example:** `Device(config-ikev2-author-policy)# pool abc`	Defines a local IP address pool for assigning IP addresses to the remote access client. • **ipv6**—(Optional) Specifies an IPv6 address pool. To specify an IPv4 address, execute the command without this keyword.. • *name*—Name of the local IP address pool. **Note** The local IP address pool must already be defined using the **ip local pool** command.
Step 17	**route set** {**interface** *interface* | **access-list** {*access-list-name* | *access-list-number* | **ipv6** *access-list-name*}} **Example:** `Device(config-ikev2-author-policy)# route set interface`	Specifies the route set parameters to the peer via configuration mode and allows running routing protocols such as Border Gateway Protocol (BGP) over VPN. • **interface**—Specifies the route interface. • **access-list**—Specifies the route access list. • *access-list-name*—Access list name. • *access-list-number*—Standard access list number. • **ipv6**—Specifies an IPv6 access list.
Step 18	**route accept any** [**tag** *value*] [**distance** *value*] **Example:** `Device(config-ikev2-author-policy)# route accept any tag 10`	Filters the routes received from the peer and specify the tag and metric values to install these routes. • **any**—Accepts all routes received from the peer. • **tag** *value*—(Optional) Specifies the tag ID for the static routes added by IKEv2. The range is from 1 to 497777. • **distance** *value*—(Optional) Specifies the distance for the static routes added by IKEv2. The range is from 1 to 255.
Step 19	**route redistribute** *protocol* [**route-map** *map-name*] **Example:** `Device(config-ikev2-author-policy)# route redistribute connected`	Filters the routes received from the peer and specify the tag and metric values to install these routes. • *protocol*—Source protocol from which routes are redistributed. It can be one of the following keywords: connected or static.

	Command or Action	Purpose
		• **route-map** *map-name*—(Optional) Route map that should be filtered to import routes from one source routing protocol to another routing protocol. If a map name is not specified, all routes are redistributed.
Step 20	**route set remote** {**ipv4** *ip-address mask* | **ipv6** *ip-address/mask*} **Example:** `Device(config-ikev2-author-policy)# route set remote ipv6 2001:DB8::1/32`	Configures IP addresses of inside networks.
Step 21	**smartcard-removal-disconnect** **Example:** `Device(config-ikev2-author-policy)# smartcard-removal-disconnect`	Enables smartcard removal disconnect. This parameter is sent to the client via the nonstandard Cisco Unity configuration attribute. This parameter specifies that the client should terminate the session when the smart card is removed.
Step 22	**split-dns** *string* **Example:** `Device(config-ikev2-author-policy)# split-dns abc1`	Allows you to specify up to ten split domain names. This parameter is sent to the client via the nonstandard Cisco Unity configuration attribute. This parameter specifies the domain names that the client should use for private networks.
Step 23	**session-lifetime** *seconds* **Example:** `Device(config-ikev2-author-policy)# session-lifetime 1000`	Specifies the IKEv2 session lifetime. • **seconds** *seconds*—The range is from 120 to 25920000, which converts to two minutes to 300 days.
Step 24	**route set access-list** {*acl-number* | [**ipv6**] *acl-name*} **Example:** `Device(config-ikev2-client-config-group)# route set access-list 110`	Specifies the subnets that are pushed to the remote peer via configuration mode. • *acl-number*—Access list number (ACL). The ACL number can only be specified for an IPv4 ACL. • **ipv6**—(Optional) Specifies an IPv6 access control list (ACL). To specify an IPv4 attribute, execute the command without this keyword. • *acl-name*—Access list name. **Note** You can only specify standard, simple access lists for IPv4 addresses.
Step 25	**wins** *primary-server* [*secondary-server*] **Example:** `Device(config-ikev2-author-policy)# wins 203.0.113.1 203.0.113.115`	Specifies the internal Windows Internet Naming Service (WINS) server addresses that are sent to the client in the configuration reply. • *primary-server*—IP address of the primary WINS server. • *secondary-server*—(Optional) IP address of the secondary WINS server.

	Command or Action	Purpose
Step 26	**end** **Example:** `Device(config-ikev2-author-policy)# end`	Exits IKEv2 authorization policy configuration mode and returns to privileged EXEC mode.

Configuration Examples for the FlexVPN Server

Example: Configuring the FlexVPN Server

Example: Configuring the FlexVPN Server to Authenticate Peers Using EAP

This example shows how to configure the FlexVPN server to authenticate peers using EAP.

```
aaa new-model
!
aaa group server radius eap-server
 server 192.168.2.1
!
aaa authentication login eap-list group eap-server
!
crypto pki trustpoint trustpoint1
 enrollment url http://192.168.3.1:80
 revocation-check crl
!
crypto ikev2 profile ikev2-profile1
 match identity remote address 0.0.0.0
 authentication local rsa-sig
 authentication remote eap query-identity
 pki trustpoint trustpoint1
 aaa authentication eap eap-list
 virtual-template 1
!
crypto ipsec transform-set transform1 esp-aes esp-sha-hmac
!
crypto ipsec profile ipsec-profile1
 set transform-set trans transform1
 set ikev2-profile ikev2-profile1
!
interface Ethernet0/0
 ip address 192.168.1.1 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile ipsec-profile1
!
radius-server host 192.168.2.1 key key1
!
```

Example: Configuring the FlexVPN Server for Group Authorization (External AAA)

The following example shows how to configure the FlexVPN server for group authentication through an external AAA, which would be the RADIUS or TACACS server.

```
aaa new-model
!
aaa group server radius cisco-acs
 server 192.168.2.2
!
aaa authorization network group-author-list group cisco-acs
!
crypto pki trustpoint trustpoint1
 enrollment url http://192.168.3.1:80
 revocation-check crl
!
crypto pki certificate map certmap1 1
 subject-name co cisco
!
crypto ikev2 name-mangler group-author-mangler
 dn domain
!
crypto ikev2 profile ikev2-profile1
 match certificate certmap1
 authentication local rsa-sig
 authentication remote rsa-sig
 pki trustpoint trustpoint1
 aaa authorization group cert list group-author-list name-mangler group-author-mangler
 virtual-template 1
!
crypto ipsec transform-set transform1 esp-aes esp-sha-hmac
!
crypto ipsec profile ipsec-profile1
 set transform-set trans transform1
 set ikev2-profile ikev2-profile1
!
interface Ethernet0/0
 ip address 192.168.1.1 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile ipsec-profile1
!
radius-server host 192.168.2.2 key key2
!
```

Example: Configuring the FlexVPN Server for Group Authorization (Local AAA)

The following example shows how to configure the FlexVPN server for group authorization through the local AAA using the IKEv2 authorization policy. The authorization policy specifies standard IPv4 and IPv6 attributes, and Cisco Unity, and FlexVPN attributes to be sent to the client through configuration mode. The authorization policy also specifies per user attributes through **aaa attribute list** command for local use.

```
aaa new-model
!
aaa authorization network local-group-author-list local
!
!
aaa attribute list attr-list1
 attribute type interface-config "ip mtu 1100"
 attribute type interface-config "tunnel key 10"
!

crypto pki trustpoint trustpoint1
 enrollment url http://192.168.3.1:80
```

```
 revocation-check crl
!
crypto pki certificate map certmap1 1
 subject-name co cisco
!
crypto ikev2 authorization policy author-policy1
 pool pool1
 dhcp server 192.168.4.1
 dhcp timeout 10
 dhcp giaddr 192.168.1.1
 dns 10.1.1.1 10.1.1.2
 route set access-list acl1
 wins 192.168.1.2 192.168.1.3
 netmask 255.0.0.0
 banner ^C flexvpn server ^C
 configuration url http://www.abc.com
 configuration version 10
 def-domain abc.com
 split-dns dns1
 split-dns dns2
 split-dns dns3
 backup-gateway gw1
 backup-gateway gw2
 backup-gateway gw3
 smartcard-removal-disconnect
 include-local-lan
 pfs
 aaa attribute list attr-list1
!
crypto ikev2 profile ikev2-profile1
 match certificate certmap1
 authentication local rsa-sig
 authentication remote rsa-sig
 pki trustpoint trustpoint1
 aaa authorization group cert list local-group-author-list author-policy1
 virtual-template 1
!
crypto ipsec transform-set transform1 esp-aes esp-sha-hmac
!
crypto ipsec profile ipsec-profile1
 set transform-set trans transform1
 set ikev2-profile ikev2-profile1
!
interface Ethernet0/0
 ip address 192.168.1.1 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile ipsec-profile1
!
ip local pool pool11 192.168.2.10 192.168.2.100
!
ip access-list extended acl-1
 permit ip 192.168.3.10 192.168.4.100 any
 permit ip 192.168.10.1 192.168.10.100 any
!
```

Example: Configuring the FlexVPN Server for User Authorization

The following example shows how to configure the FlexVPN server for user authentication.

```
aaa new-model
```

```
!
aaa group server radius cisco-acs
 server 192.168.2.2
!
aaa authorization network user-author-list group cisco-acs
!
crypto pki trustpoint trustpoint1
 enrollment url http:// 192.168.3.1:80
 revocation-check crl
!
crypto pki certificate map certmap1 1
 subject-name co cisco
!
crypto ikev2 name-mangler user-author-mangler
 dn common-name
!
crypto ikev2 profile ikev2-profile1
 match certificate certmap1
 authentication local rsa-sig
 authentication remote rsa-sig
 pki trustpoint trustpoint1
 aaa authorization user cert list user-author-list name-mangler user-author-mangler
 virtual-template 1
!
crypto ipsec transform-set transform1 esp-aes esp-sha-hmac
!
crypto ipsec profile ipsec-profile1
 set transform-set trans transform1
 set ikev2-profile ikev2-profile1
!
interface Ethernet0/0
 ip address 192.168.1.1 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile ipsec-profile1
!
radius-server host 192.168.2.2 key key2
!
```

Example: Configuring the FlexVPN Server for IPv6 Session with IPv6 Configuration Attributes

The following example shows how to configure the FlexVPN server for an IPv6 dynamic Virtual Tunnel Interfaces (dVTI) session. The example uses local AAA group authorization using the IKEv2 authorization policy. The IPv6 configuration attributes are configured under the IKEv2 authorization policy.

```
aaa new-model
!
aaa authorization network local-group-author-list local
!
crypto pki trustpoint trustpoint1
 enrollment url http://192.168.3.1:80
 revocation-check crl
!
crypto pki certificate map certmap1 1
 subject-name co cisco
!
crypto ikev2 authorization policy author-policy1
 ipv6 pool v6-pool
 ipv6 dns 2001:DB8:1::11 2001:DB8:1::12
 ipv6 subnet-acl v6-acl
```

```
!
crypto ikev2 profile ikev2-profile1
 match certificate certmap1
 authentication local rsa-sig
 authentication remote rsa-sig
 pki trustpoint trustpoint1
 aaa authorization group cert list local-group-author-list author-policy1
 virtual-template 1
!
crypto ipsec transform-set transform1 esp-aes esp-sha-hmac
!
crypto ipsec profile ipsec-profile1
 set transform-set trans transform1
 set ikev2-profile ikev2-profile1
!
interface Ethernet0/0
 ipv6 address 2001:DB8:1::1/32
!
interface Virtual-Template1 type tunnel
 ipv6 unnumbered Ethernet0/0
 tunnel mode ipsec ipv6
 tunnel protection ipsec profile ipsec-profile1
!
ipv6 local pool v6-pool 2001:DB8:1::10/32 48
!
ipv6 access-list v6-acl
 permit ipv6 host 2001:DB8:1::20 any
 permit ipv6 host 2001:DB8:1::30 any
!
```

Example: Configuring AnyConnect Profile Download

The following example shows how to configure the FlexVPN AnyConnect Profile Download feature:

Note You do not modify the Local Policy files on the Anyconnect Client machine. After the configuration of Anyconnect Profile Download feature on IKEv2, the required XML profiles get automatically downloaded on the client device.

Note You should disable either the HTTPS server (ip http secure-server) or SSL policy (crypto ssl policy) for the profile download feature, otherwise, if both these features are enabled at the same time and the device receives an incoming SSL VPN connection, the device may crash.

```
no ip http secure-server
crypto ssl policy ssl-policy
  pki trustpoint CA1 sign
  ip address local 10.0.0.1 port 443
  no shutdown
crypto ssl profile ssl_prof
    match policy ssl-policy
crypto vpn anyconnect profile ANY-PROF bootflash:profile.xml
crypto ikev2 profile ikev2_profile
    anyconnect profile ANY-PROF
```

Additional References for Configuring the FlexVPN Server

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
Cisco AnyConnect Secure Mobility Client	https://www.cisco.com/c/en/us/support/security/anyconnect-secure-mobility-client/tsd-products-support-series-home.html
IPsec configuration	*Configuring Security for VPNs with IPsec*
Recommended cryptographic algorithms	Next Generation Encryption

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 the FlexVPN Server

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 4: Feature Information for Configuring the FlexVPN Server

Feature Name	Releases	Feature Information
IKEv2 headend support for remote access clients	Cisco IOS XE Release 3.5S	This features provides IKEv2 support for Anyconnect 3.0, FlexVPN hardware client, and multi SA support for VTI. The following commands were introduced or modified: **aaa attribute list, backup-gateway, banner, config-mode set, configuration url, configuration version, def-domain, dhcp, dns, include-local-lan, max flow limit, pfs, pool, route accept, route set interface, smartcard-removal-disconnect, split-dns, subnet-acl.**