CHAPTER 22

Configuring VPNs in VRF Mode

This chapter provides information about configuring IPsec VPNs in Virtual Routing and Forwarding (VRF) mode, one of the two VPN configuration modes supported by the IPsec VPN SPA. For information on the other VPN mode, crypto-connect mode, see Chapter 21, "Configuring VPNs in Crypto-Connect Mode."

This chapter includes the following topics:

For general information on configuring IPsec VPNs with the IPsec VPN SPA, see the "Overview of Basic IPsec and IKE Configuration Concepts" section on page 20-3.

Note The procedures in this chapter assume you have familiarity with security configuration concepts, such as VLANs, ISAKMP policies, preshared keys, transform sets, access control lists, and crypto maps. For detailed information on configuring these features, refer to the following Cisco IOS documentation:

Cisco IOS Security Configuration Guide, *Release 12.2*, at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/security/configuration/guide/fsecur_c.html

Cisco IOS Security Command Reference, *Release 12.2*, at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/security/command/reference/fsecur_r.html

For additional information about the commands used in this chapter, see the *Catalyst 6500 Series Cisco IOS Command Reference, Release 12.2SX* and the related Cisco IOS Release 12.2 software configuration guide and master index publications. For more information about accessing these publications, see the "Related Documentation" section on page xlv.

Tip To ensure a successful configuration of your VPN using the IPsec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Configuring VPNs in VRF Mode

VRF mode, also known as VRF-Aware IPsec, allows you to map IPsec tunnels to VPN routing and forwarding instances (VRFs) using a single public-facing address.

A VRF instance is a per-VPN routing information repository that defines the VPN membership of a customer site attached to the Provider Edge (PE) router. A VRF comprises an IP routing table, a derived Cisco Express Forwarding (CEF) table, a set of interfaces that use the forwarding table, and a set of rules and routing protocol parameters that control the information that is included in the routing table. A separate set of routing and CEF tables is maintained for each VPN customer.

Each IPsec tunnel is associated with two VRF domains. The outer encapsulated packet belongs to one VRF domain, called the front door VRF (FVRF), while the inner, protected IP packet belongs to another domain called the inside VRF (IVRF). Stated another way, the local endpoint of the IPsec tunnel belongs to the FVRF while the source and destination addresses of the inside packet belong to the IVRF, the unprotected (LAN) side.

Note Front door VRF (FVRF) is only supported as of Cisco IOS Release 12.2(33)SXH and later.

One or more IPsec tunnels can terminate on a single interface. The FVRF of all these tunnels is the same and is set to the VRF that is configured on that interface. The IVRF of these tunnels can be different and depends on the VRF that is defined in the ISAKMP profile that is attached to a crypto map entry.

With VRF mode, packets belonging to a specific VRF are routed through the IPsec VPN SPA for IPsec processing. Through the CLI, you associate a VRF with an interface VLAN that has been configured to point to the IPsec VPN SPA. An interface VLAN must be created for each VRF. Packets traveling from an MPLS cloud to the Internet that are received from an inside VRF are routed to an interface VLAN, and then to the IPsec VPN SPA for IPsec processing. The IPsec VPN SPA modifies the packets so that they are placed on a special Layer 3 VLAN for routing to the WAN-side port after they leave the IPsec VPN SPA.

Packets traveling in the inbound direction from a protected port on which the **crypto engine slot** command has been entered are redirected by a special ACL to the IPsec VPN SPA, where they are processed according to the Security Parameter Index (SPI) contained in the packet's IPsec header. Processing on the IPsec VPN SPA ensures that the decapsulated packet is mapped to the appropriate interface VLAN corresponding to the inside VRF. This interface VLAN has been associated with a specific VRF, so packets are routed within the VRF to the correct inside interface.

Note Tunnel protection is supported in VRF mode. For information on configuring tunnel protection, see the "Configuring VPNs in VRF Mode with Tunnel Protection (GRE)" section on page 22-12 and the "VRF Mode Tunnel Protection Configuration Example" section on page 22-33.

When configuring a VPN using VRF mode, you have these additional tunneling options: tunnel protection (TP) using GRE, and Virtual Tunnel Interface (VTI). With either of these options, you can terminate tunnels in VRFs (normal VRF mode) or in the global context.

The following subsections describe how to configure a VPN in VRF mode on the IPsec VPN SPA:

Note For additional information on configuring VPNs in VRF mode, refer to the Cisco IOS documentation at this URL:
http://www.cisco.com/en/US/docs/ios/sec_secure_connectivity/configuration/guide/sec_vrf_aware_ipsec_ps6017_TSD_Products_Configuration_Guide_Chapter.html

Understanding VPN Configuration in VRF Mode

In the traditional crypto-connect mode, a VPN is configured by attaching crypto maps to interface VLANs and then crypto-connecting a physical port to the interface VLAN. When configuring a VPN in VRF mode using the IPsec VPN SPA, the model of interface VLANs is preserved, but the **crypto connect vlan** CLI command is not used. When a packet comes into an interface on a specific VRF, the packet must get to the proper interface VLAN. A route must be installed so that packets destined for that particular subnet in that particular VRF are directed to that interface VLAN. This function can be achieved through the following configuration options:

- Configuring an IP address on the interface VLAN that is in the same subnet as the packets' destination IP address. For example, packets are trying to reach subnet 10.1.1.x and their destination IP address is 10.1.1.1 as follows:

```
int vlan 100
 ip vrf forwarding coke
 ip address 10.1.1.254  255.255.255.0 <-- same subnet as 10.1.1.x that we are trying
to reach.
 crypto map mymap
 crypto engine slot 4/1
```

- Configuring a static route as follows:

```
ip route vrf coke 10.1.1.0 255.255.255.0 vlan 100
```

- Configuring routing protocols. You configure BGP, OSPF, or other routing protocols so that remote switches broadcast their routes.

  Note Do not configure routing protocols unless you are using tunnel protection.

- Configuring Reverse Route Injection (RRI). You configure RRI so that a route gets installed when the remote end initiates an IPsec session (as in remote access situations).

With VRF mode, the switch sees the interface VLAN as a point-to-point connection; the packets are placed directly onto the interface VLAN. Each VRF has its own interface VLAN.

When a crypto map is attached to an interface VLAN and the **ip vrf forwarding** command has associated that VLAN with a particular VRF, the software creates a point-to-point connection so that all routes pointing to the interface VLAN do not attempt to run the Address Resolution Protocol (ARP). Through normal routing within the VRF, packets to be processed by the IPsec VPN SPA are sent to the interface VLAN. You may configure features on the interface VLAN. The IP address of the interface VLAN must be on the same subnet as the desired destination subnet for packets to be properly routed.

When you enter the **ip vrf forwarding** command on an inside interface, all packets coming in on that interface are routed correctly within that VRF.

When you enable the **crypto engine mode vrf** command and enter the **crypto engine slot outside** command on an interface, a special ACL is installed that forces all incoming Encapsulating Security Payload (ESP)/Authentication Header (AH) IPsec packets addressed to a system IP address to be sent to the IPsec VPN SPA WAN-side port. NAT Traversal (NAT-T) packets are also directed to the IPsec VPN SPA by the special ACL.

Note You must enter the **vrf** *vrf_name* command from within the context of an ISAKMP profile. This command does not apply to the VRF-aware crypto infrastructure; it applies only to generic crypto processing. When the ISAKMP profile is added to a crypto map set, the VRF becomes the default VRF for all of the crypto maps in the list. Individual crypto maps may override this default VRF by specifying another policy profile that contains a different VRF. If no profile is applied to a crypto map tag, it inherits the VRF from the interface if you have configured the interface with the **ip vrf forwarding** command.

All packets destined for a protected outside interface received in this VRF context are placed on the associated interface VLAN. Similarly, all decapsulated ingress packets associated with this VRF are placed on the appropriate interface VLAN so that they may be routed in the proper VRF context.

VRF Mode Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring a VPN for the IPsec VPN SPA using VRF mode:

Note After enabling or disabling VRF mode using the [**no**] **crypto engine mode vrf** command, you must reload the supervisor engine. In addition, MPLS tunnel recirculation must be enabled for VRF mode. That is, you must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command.

- The procedure for configuring a VPN in VRF mode varies based on whether you are using tunnel protection or not.
- Unlike IPsec VPN SPA crypto-connect mode configurations, when configuring VPNs in VRF mode, you do not use the **crypto connect vlan** command.
- In Cisco IOS Release 12.2(33)SXH and later releases, the **crypto engine subslot** command used in previous releases has been replaced with the **crypto engine slot** command (of the form **crypto engine slot** *slot/subslot* {**inside** | **outside**}). The **crypto engine subslot** command is no longer supported. In Cisco IOS Release 12.2(33)SXI and later releases, it is not necessary to specify the **slot** *slot/subslot* information with the **outside** keyword. When upgrading, ensure that the **crypto engine** command has been modified in your start-up configuration to avoid extended maintenance time.
- As of Cisco IOS Release 12.2(33)SXH, the **ip vrf forwarding** command is no longer required when configuring GRE with tunnel protection.
- Crypto ACLs support only the EQ operator. Other operators, such as GT, LT, and NEQ, are not supported.

Note When configuring a permit policy for multiple ports with the EQ operator, you must use multiple lines as in this example:

```
permit ip any any port eq 300
permit ip any any port eq 400
permit ip any any port eq 600
```

In Cisco IOS Release 12.2(33)SXH1 and later releases, when configuring a deny policy for multiple ports with the EQ operator, you can use commas to declare the ports as in this example:

```
deny ip any any port eq 300,400,600
```

- Noncontiguous subnets in a crypto ACL, as in the following example, are not supported:

```
deny ip   10.0.5.0   0.255.0.255   10.0.175.0   0.255.0.255
deny ip   10.0.5.0   0.255.0.255   10.0.176.0   0.255.0.255
```

- ACL counters are not supported for crypto ACLs.
- An egress ACL is not applied to packets generated by the route processor. An ingress ACL is not applied to packets destined for the route processor.
- When you create an ISAKMP profile, note the following guidelines regarding the use of the **vrf** command:
  - You must use the **vrf** command if you are using the ISAKMP profile with a crypto map.
  - You are not required to use the **vrf** command if you are using the ISAKMP profile with tunnel protection.
  - You should not use the **vrf** command if you are using the ISAKMP profile with DMVPN.
- When the **ip vrf forwarding** command is applied to a VLAN, any previously existing IP address assigned to that VLAN is removed. To assign an IP address to the VLAN, enter the **ip address** command after the **ip vrf forwarding** command, not preceding it.
- Although more than one IPsec VPN SPA in a chassis is supported beginning with Cisco IOS Release 12.2(18) SXE, in VRF mode, there is no configuration difference between multiple IPsec VPN SPA operation and single IPsec VPN SPA operation. For multiple IPsec VPN SPA operation, the only change is to the output of the **show crypto vlan** command. The following is an example:

```
Interface Tu1 on IPSec Service Module port Gi7/1/1 connected to VRF vrf1
Interface VLAN 2 on IPSec Service Module port Gi7/1/1 connected to VRF vrf2
```

- Applying an ACL to the ingress interface will interfere with the packet flow in releases earlier than Cisco IOS Release 12.2(33) SXI. .

Note Do not apply an ACL during the configuration of VRF mode in releases earlier than Cisco IOS Release 12.2(33) SXI.

- The number of outside interfaces supported by the IPsec VPN SPA is determined by your system resources.
- Inbound and outbound traffic for the same tunnel must use the same outside interface. Asymmetric routing, in which encrypted traffic uses a different outside interface than decrypted traffic for the same tunnel, is not supported.
- A loopback interface can be used as tunnel source address.

- A crypto map local address (for example, the interface VLAN address if the crypto map is applied to the interface VLAN) can share the same address as the TP tunnel source address, but it cannot share the same address as a DMVPN tunnel source address.
- In VRF mode, crypto map interfaces that share the same local address must be bound to the same crypto engine.
- When two tunnels share the same tunnel source address, they will be taken over by the IPsec VPN SPA only if one of the following two conditions are met:
  - Both tunnels share the same FVRF.
  - The **crypto engine gre vpnblade** command is entered.
- You can configure the FVRF to be the same as the IVRF.
- In VRF mode, ingress ACLs are installed on crypto engine outside interfaces. In combination with other configured ACLs, these ACLs may cause the ACL-TCAM usage to become excessive. To reduce the TCAM usage, share the TCAM resources by entering the **mls acl tcam share-global** command in the configuration. You can view the ACL usage using the **show tcam counts** command.
- In Cisco IOS Release 12.2(33)SXF and earlier releases, IPsec can be configured with manual keying instead of IKE. If you configure manual keying, you must configure SPI to be greater than 4096.

Supported and Unsupported Features in VRF Mode

A list of the supported and unsupported features in VRF mode can be found in the "IPsec Feature Support" section on page 20-6. Additional details are as follows:

- Remote access into a VRF (provider edge [PE]) is supported with the following:
  - Reverse Route Injection (RRI) only with crypto maps
  - Proxy AAA (one VRF is proxied to a dedicated AAA)
- Customer edge-provider edge (CE-PE) encryption using tunnel protection is supported with the following:
  - Routing update propagation between CEs
  - IGP/eBGP routing update propagation between the PE and CEs

Configuring VPNs in VRF Mode without Tunnel Protection

To configure a VPN in VRF mode with crypto maps and without tunnel protection, perform this task beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# mls mpls tunnel-recir`	Enables tunnel-MPLS recirculation.
Step 2	`Router(config)# crypto engine mode vrf`	Enables VRF mode for the IPsec VPN SPA. **Note** After enabling or disabling VRF mode using the **crypto engine mode vrf** command, you must reload the supervisor engine.
Step 3	`Router(config)# ip vrf` *vrf-name*	Configures a VRF routing table and enters VRF configuration mode. • *vrf-name*—Name assigned to the VRF.

	Command	Purpose
Step 4	`Router(config-vrf)# rd route-distinguisher`	Creates routing and forwarding tables for a VRF. • *route-distinguisher*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1).
Step 5	`Router(config-vrf)# route-target export route-target-ext-community`	Creates lists of export route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 6	`Router(config-vrf)# route-target import route-target-ext-community`	Creates lists of import route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 7	`Router(config-vrf)# exit`	Exits VRF configuration mode.
Step 8	`Router(config)# crypto keyring keyring-name [vrf fvrf-name]`	Defines a crypto keyring to be used during IKE authentication and enters keyring configuration mode. • *keyring-name*—Name of the crypto keyring. • *fvrf-name*—(Optional) Front door virtual routing and forwarding (FVRF) name to which the keyring will be referenced. fvrf-name must match the FVRF name that was defined during virtual routing and forwarding (VRF) configuration
Step 9	`Router(config-keyring)# pre-shared-key {address address [mask] | hostname hostname} key key`	Defines a preshared key to be used for IKE authentication. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • *hostname*—Fully qualified domain name of the peer. • *key*—Specifies the secret key.
Step 10	`Router(config-keyring)# exit`	Exits keyring configuration mode.

	Command	Purpose
Step 11	`Router(config)# crypto ipsec transform-set transform-set-name transform1[transform2[transform3]]`	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPsec security protocols and algorithms. Accepted values are described in the *Cisco IOS Security Command Reference*.
Step 12	`Router(config-crypto-trans)# exit`	Exits crypto transform configuration mode
Step 13	`Router(config)# crypto isakmp policy priority`	Defines an IKE policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest.
Step 14	`Router(config-isakmp)# authentication pre-share`	Specifies the authentication method with an IKE policy. • **pre-share**—Specifies preshared keys as the authentication method.
Step 15	`Router(config-isakmp)# lifetime seconds`	Specifies the lifetime of an IKE SA. • *seconds*—Number of seconds each SA should exist before expiring. Use an integer from 60 to 86,400 seconds. Default is 86,400 (one day).
Step 16	`Router(config-isakmp)# exit`	Exits ISAKMP policy configuration mode.
Step 17	`Router(config)# crypto isakmp profile profile-name`	Defines an ISAKMP profile and enters ISAKMP profile configuration mode. • *profile-name*—Name of the user profile.
Step 18	`Router(config-isa-prof)# vrf ivrf`	Defines the VRF to which the IPsec tunnel will be mapped. • *ivrf*—Name of the VRF to which the IPsec tunnel will be mapped. Enter the same value specified in Step 3.
Step 19	`Router(config-isa-prof)# keyring keyring-name`	Configures a keyring within an ISAKMP profile. • *keyring-name*—Keyring name. This name must match the keyring name that was defined in global configuration. Enter the value specified in Step 8.

	Command	Purpose
Step 20	`Router(config-isa-prof)# match identity address address [mask] [vrf]`	Matches an identity from a peer in an ISAKMP profile. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • [*vrf*]—(Optional) This argument is only required when configuring a front door VRF (FVRF). This argument specifies that the address is an FVRF instance.
Step 21	`Router(config-isa-prof)# exit`	Exits ISAKMP profile configuration mode.
Step 22	`Router(config)# access list access-list-number {deny | permit} ip host source host destination`	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Number of the host from which the packet is being sent. • *destination*—Number of the host to which the packet is being sent.
Step 23	`Router(config)# crypto map map-name seq-number ipsec-isakmp`	Creates or modifies a crypto map entry and enters the crypto map configuration mode. • *map-name*—Name that identifies the crypto map set. • *seq-number*—Sequence number you assign to the crypto map entry. Lower values have higher priority. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPsec security associations.
Step 24	`Router(config-crypto-map)# set peer {hostname | ip-address}`	Specifies an IPsec peer in a crypto map entry. • {*hostname* | i*p-address*}—IPsec peer host name or IP address. Enter the value specified in Step 20.
Step 25	`Router(config-crypto-map)# set transform-set transform-set-name`	Specifies which transform sets can be used with the crypto map entry. • *transform-set-name*—Name of the transform set. Enter the value specified in Step 11.
Step 26	`Router(config-crypto-map)# set isakmp-profile profile-name`	Sets the ISAKMP profile name. • *profile-name*—Name of the ISAKMP profile. Enter the value entered in Step 17.

	Command	Purpose
Step 27	`Router(config-crypto-map)# match address [access-list-id | name]`	Specifies an extended access list for the crypto map entry. • *access-list-id*—Identifies the extended access list by its name or number. Enter the value specified in Step 22. • *name*—(Optional) Identifies the named encryption access list. This name should match the name argument of the named encryption access list being matched.
Step 28	`Router(config-crypto-map)# exit`	Exits crypto map configuration mode.
Step 29	`Router(config)# crypto map map-name local-address interface-id`	Specifies and names an identifying interface to be used by the crypto map for IPsec traffic. • *map-name*—Name that identifies the crypto map set. Enter the value specified in Step 23. • **local-address** *interface-id*—Name of interface that has the local address of the switch. **Note** The local address must belong to the FVRF. **Note** In VRF mode, the VPN feature supports up to 1024 local addresses. This limit is across the chassis (not per VPN module).
Step 30	`Router(config)# interface fastethernet slot/port`	Configures a Fast Ethernet interface and enters interface configuration mode.
Step 31	`Router(config-if)# ip vrf forwarding vrf-name`	Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF. Enter the value specified in Step 3.
Step 32	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 33	`Router(config-if)# no shutdown`	Enables the interface.
Step 34	`Router(config-if)# interface gigabitethernet slot/subslot/port`	Configures a Gigabit Ethernet interface. Match the value specified as the *interface-id* in Step 29.
Step 35	`Router(config-if)# ip vrf forwarding vrf-name`	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 36	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask.

	Command	Purpose
Step 37	`Router(config-if)# crypto engine slot slot/subslot outside`	Assigns the specified crypto engine to the interface. • *slot/subslot*—The slot where the IPsec VPN SPA is located. **Note** In Cisco IOS Release 12.2(33)SXI and later releases, do not specify **slot** *slot/subslot* with the **outside** keyword.
Step 38	`Router(config-if)# no shutdown`	Enables the interface.
Step 39	`Router(config-if)# exit`	Exits interface configuration mode.
Step 40	`Router(config)# interface vlan-id`	Configures a VLAN interface and enters interface configuration mode. • *vlan-id*—VLAN identifier.
Step 41	`Router(config-if)# ip vrf forwarding vrf-name`	Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF. Enter the value specified in Step 3.
Step 42	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 43	`Router(config-if)# crypto map map-name`	Applies a previously defined crypto map set to an interface. • *map-name*—Name that identifies the crypto map set. Enter the value specified in Step 232.
Step 44	`Router(config-if)# crypto engine slot slot/subslot inside`	Assigns the specified crypto engine to the interface. • *slot/subslot*—The slot where the IPsec VPN SPA is located.
Step 45	`Router(config-if)# exit`	Exits interface configuration mode.
Step 46	`Router(config)# ip route vrf vrf-name prefix mask interface-number`	Establishes static routes for a VRF. • *vrf-name*—Name of the VRF for the static route. Enter the value specified in Step 3. • *prefix*—IP route prefix for the destination, in dotted-decimal format. • *mask*—Prefix mask for the destination, in dotted decimal format. • *interface-number*—Number identifying the network interface to use. Enter the *vlan-id* value specified in Step 40.
Step 47	`Router(config)# end`	Returns to privileged EXEC mode.

For complete configuration information for VRF-Aware IPsec, refer to this URL:

http://www.cisco.com/en/US/docs/ios/sec_secure_connectivity/configuration/guide/sec_vrf_aware_ipsec_ps6017_TSD_Products_Configuration_Guide_Chapter.html

For a configuration example, see the "VRF Mode Basic Configuration Example" section on page 22-23.

Configuring VPNs in VRF Mode with Tunnel Protection (GRE)

This section describes how to configure a VPN in VRF mode with tunnel protection (TP). Tunnel protection is GRE tunneling in VRF mode.

When you configure IPsec, a crypto map is attached to an interface to enable IPsec. With tunnel protection, there is no need for a crypto map or ACL to be attached to the interface. A crypto policy is attached directly to the tunnel interface. Any traffic routed by the interface is encapsulated in GRE and then encrypted using IPsec. The tunnel protection feature can be applied to point-to-point GRE.

VRF Mode Using Tunnel Protection Configuration Guidelines and Restrictions

When configuring tunnel protection on the IPsec VPN SPA, follow these guidelines and restrictions:

- Do not configure any options (such as sequence numbers or tunnel keys) that prevent the IPsec VPN SPA from seizing the GRE tunnel.
- Do not configure the GRE tunnel keepalive feature.
- When applied to the GRE tunnel interface, the **ip tcp adjust-mss** command is ignored. Apply the command to the ingress LAN interface instead. (CSCsl27876)
- Do not use crypto maps to protect GRE traffic in VRF mode.
- When a crypto map interface and a tunnel protection interface (either VTI or GRE/TP) share the same outside interface, they cannot share the same local source address.
- To avoid fragmentation after encryption, set the tunnel IP MTU to be equal to or less than the egress interface MTU minus the GRE and IPsec overheads. The egress interface MTU must be the smallest MTU of all the active crypto outside interfaces.

To configure a VPN in VRF mode using tunnel protection, perform this task beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# mls mpls tunnel-recir`	Enables tunnel-MPLS recirculation.
Step 2	`Router(config)# crypto engine mode vrf`	Enables VRF mode for the IPsec VPN SPA. **Note** After enabling or disabling VRF mode using the **crypto engine mode vrf** command, you must reload the supervisor engine.
Step 3	`Router(config)# ip vrf` *vrf-name*	Configures a VRF routing table and enters VRF configuration mode. • *vrf-name*—Name assigned to the VRF.
Step 4	`Router(config-vrf)# rd` *route-distinguisher*	Creates routing and forwarding tables for a VRF. • *route-distinguisher*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1).

	Command	Purpose
Step 5	`Router(config-vrf)# route-target export route-target-ext-community`	Creates lists of export route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 6	`Router(config-vrf)# route-target import route-target-ext-community`	Creates lists of import route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 7	`Router(config-vrf)# exit`	Exits VRF configuration mode.
Step 8	`Router(config)# crypto keyring keyring-name [vrf fvrf-name]`	Defines a crypto keyring to be used during IKE authentication and enters keyring configuration mode. • *keyring-name*—Name of the crypto keyring. • *fvrf-name*—(Optional) Front door virtual routing and forwarding (FVRF) name to which the keyring will be referenced. *fvrf-name* must match the FVRF name that was defined during virtual routing and forwarding (VRF) configuration.
Step 9	`Router(config-keyring)# pre-shared-key {address address [mask] | hostname hostname} key key`	Defines a preshared key to be used for IKE authentication. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • *hostname*—Fully qualified domain name of the peer. • *key*—Specifies the secret key.
Step 10	`Router(config-keyring)# exit`	Exits keyring configuration mode.
Step 11	`Router(config)# crypto ipsec transform-set transform-set-name transform1[transform2[transform3]]`	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPsec security protocols and algorithms. Accepted values are described in the *Cisco IOS Security Command Reference*.
Step 12	`Router(config-crypto-trans)# exit`	Exits crypto transform configuration mode

	Command	Purpose
Step 13	`Router(config)# crypto isakmp policy priority`	Defines an IKE policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest.
Step 14	`Router(config-isakmp)# authentication pre-share`	Specifies the authentication method with an IKE policy. • **pre-share**—Specifies preshared keys as the authentication method.
Step 15	`Router(config-isakmp)# lifetime seconds`	Specifies the lifetime of an IKE SA. • *seconds*—Number of seconds each SA should exist before expiring. Use an integer from 60 to 86,400 seconds. Default is 86,400 (one day.)
Step 16	`Router(config-isakmp)# exit`	Exits ISAKMP policy configuration mode.
Step 17	`Router(config)# crypto isakmp profile profile-name`	Defines an ISAKMP profile and enters ISAKMP profile configuration mode • *profile-name*—Name of the user profile.
Step 18	`Router(config-isa-prof)# keyring keyring-name`	Configures a keyring within an ISAKMP profile. • *keyring-name*—Keyring name. This name must match the keyring name that was defined in global configuration. Enter the value specified in Step 8.
Step 19	`Router(config-isa-prof)# match identity address address [mask]`	Matches an identity from a peer in an ISAKMP profile. • *address* [*mask*]—IP address of the remote peer or a subnet and mask.
Step 20	`Router(config-isa-prof)# exit`	Exits ISAKMP profile configuration mode.
Step 21	`Router(config)# access list access-list-number {deny | permit} ip host source host destination`	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Number of the host from which the packet is being sent. • *destination*—Number of the host to which the packet is being sent.
Step 22	`Router(config)# crypto ipsec profile profile-name`	Defines an IPsec profile and enters IPsec profile configuration mode. • *profile-name*—Name of the user profile.

	Command	Purpose
Step 23	`Router(config-ipsec-profile)# set transform-set transform-set-name`	Specifies which transform sets can be used with the crypto map entry. • *transform-set-name*—Name of the transform set. Enter the value specified in Step 11.
Step 24	`Router(config-ipsec-profile)# set isakmp-profile profile-name`	Sets the ISAKMP profile name. • *profile-name*—Name of the ISAKMP profile. Enter the value entered in Step 17.
Step 25	`Router(config-ipsec-profile)# exit`	Exits IPsec profile configuration mode.
Step 26	`Router(config)# interface tunnel-number`	Configures a tunnel interface and enters interface configuration mode. • *tunnel-number*—Name assigned to the tunnel interface.
Step 27	`Router(config-if)# ip vrf forwarding vrf-name`	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF. Enter the value specified in Step 3.
Step 28	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 29	`Router(config-if)# tunnel source ip-address`	Sets the source address of a tunnel interface. • *ip-address*—IP address to use as the source address for packets in the tunnel.
Step 30	`Router(config-if)# tunnel vrf vrf-name`	(Optional) Associates a VPN routing and forwarding instance (VRF) with a specific tunnel destination, interface or subinterface. This step is only required when configuring a front door VRF (FVRF). • *vrf-name*—Name assigned to the VRF.
Step 31	`Router(config-if)# tunnel destination ip-address`	Sets the destination address of a tunnel interface. • *ip-address*—IP address to use as the destination address for packets in the tunnel.
Step 32	`Router(config-if)# tunnel protection ipsec crypto-policy-name`	Associates a tunnel interface with an IPsec profile. • *crypto-policy-name*—The value as specified in Step 22.
Step 33	`Router(config-if)# crypto engine slot slot/subslot inside`	Assigns the specified crypto engine to the interface. • *slot/subslot*—The slot where the IPsec VPN SPA is located.
Step 34	`Router(config-if)# interface fastethernet slot/subslot`	Configures a Fast Ethernet interface.

	Command	Purpose
Step 35	`Router(config-if)# ip vrf forwarding vrf-name`	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 36	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask.
Step 37	`Router(config-if)# no shutdown`	Enables the interface.
Step 38	`Router(config-if)# interface type slot/subslot/port`	Configures the physical egress interface.
Step 39	`Router(config-if)# ip vrf forwarding vrf-name`	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 40	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for an interface. • *address*—IP address. Enter the value specified in Step 29. • *mask*—Subnet mask.
Step 41	`Router(config-if)# crypto engine slot slot/subslot outside`	Assigns the crypto engine to the interface. • *slot/subslot*—The slot where the IPsec VPN SPA is located. **Note** In Cisco IOS Release 12.2(33)SXI and later releases, do not specify **slot** *slot/subslot* with the **outside** keyword.
Step 42	`Router(config-if)# no shutdown`	Enables the interface.
Step 43	`Router(config-if)# exit`	Exits interface configuration mode.

For a configuration example, see the "VRF Mode Tunnel Protection Configuration Example" section on page 22-33.

Configuring an IPsec Virtual Tunnel Interface

The IPsec Virtual Tunnel Interface (VTI) provides a routable interface type for terminating IPsec tunnels that greatly simplifies the configuration process when you need to provide protection for remote access, and provides a simpler alternative to using GRE tunnels and crypto maps with IPsec. In addition, the IPsec VTI simplifies network management and load balancing.

Note IPsec VTI is supported in Cisco IOS Release 12.2(33)SXH and later releases, and is not supported in crypto-connect mode.

Note the following details about IPsec VTI routing and traffic encryption:

- You can enable routing protocols on the tunnel interface so that routing information can be propagated over the virtual tunnel. The router can establish neighbor relationships over the virtual tunnel interface. Interoperability with standard-based IPsec installations is possible through the use of the IP ANY ANY proxy. The static IPsec interface will negotiate and accept IP ANY ANY proxies.
- The IPsec VTI supports native IPsec tunneling and exhibits most of the properties of a physical interface.
- In the IPsec VTI, encryption occurs in the tunnel. Traffic is encrypted when it is forwarded to the tunnel interface. Traffic forwarding is handled by the IP routing table, and dynamic or static IP routing can be used to route the traffic to the virtual tunnel interface. Using IP routing to forward the traffic to encryption simplifies the IPsec VPN configuration because the use of ACLs with a crypto map in native IPsec configurations is not required. When IPsec VTIs are used, you can separate applications of NAT, ACLs, and QoS, and apply them to clear text or encrypted text, or both. When crypto maps are used, there is no easy way to specify forced encryption features.

IPsec Virtual Tunnel Interface Configuration Guidelines and Restrictions

When configuring IPsec VTI, follow these guidelines and restrictions:

- A VTI tunnel can terminate either in a VRF (normal VRF mode) or in the global context (with no **ip vrf forwarding** command on the tunnel interface).
- Only static VTI is supported.
- Only strict IP ANY ANY proxy is supported.
- The IPsec transform set must be configured only in tunnel mode.
- The IKE security association (SA) is bound to the virtual tunnel interface. Because it is bound to the virtual tunnel interface, the same IKE SA cannot be used for a crypto map.
- When the **mls mpls tunnel-recir** command is applied in a VTI configuration, one reserved VLAN is allocated to each tunnel. As a result, there will be a maximum limit of 1000 VTI tunnels.
- In releases earlier than Cisco IOS Release 12.2(33)SXI, the following guidelines apply:
  - The IPsec virtual tunnel interface is limited to IP unicast, as opposed to GRE tunnels, which have a wider application for IPsec implementation.
  - Multicast over VTI is not supported except for control plane traffic such as routing protocol updates.
- In Cisco IOS Release 12.2(33)SXI and later releases, the following guidelines apply:
  - A static VTI tunnel interface supports multicast traffic.
  - ACLs can be applied to GRE and static VTI tunnel interfaces participating in multicast traffic.
  - Platform QoS features can be applied to GRE and static VTI tunnel interfaces participating in multicast traffic.

Configuring an IPsec Static Tunnel

To configure a static IPsec virtual tunnel interface, perform this task beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# mls mpls tunnel-recir`	Enables tunnel-MPLS recirculation.
Step 2	`Router(config)# crypto engine mode vrf`	Enables VRF mode for the IPsec VPN SPA. **Note** After enabling or disabling VRF mode using the **crypto engine mode vrf** command, you must reload the supervisor engine.
Step 3	`Router(config)# ip vrf` *vrf-name*	Configures a VRF routing table and enters VRF configuration mode. • *vrf-name*—Name assigned to the VRF.
Step 4	`Router(config-vrf)# rd` *route-distinguisher*	Creates routing and forwarding tables for a VRF. • *route-distinguisher*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1).
Step 5	`Router(config-vrf)# route-target export` *route-target-ext-community*	Creates lists of export route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 6	`Router(config-vrf)# route-target import` *route-target-ext-community*	Creates lists of import route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 7	`Router(config-vrf)# exit`	Exits VRF configuration mode.

	Command	Purpose
Step 8	`Router(config)# crypto keyring keyring-name [vrf fvrf-name]`	Defines a crypto keyring to be used during IKE authentication and enters keyring configuration mode. • *keyring-name*—Name of the crypto keyring. • *fvrf-name*—(Optional) Front door virtual routing and forwarding (FVRF) name to which the keyring will be referenced. *fvrf-name* must match the FVRF name that was defined during virtual routing and forwarding (VRF) configuration.
Step 9	`Router(config-keyring)# pre-shared-key {address address [mask] | hostname hostname} key key`	Defines a preshared key to be used for IKE authentication. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • *hostname*—Fully qualified domain name of the peer. • *key*—Specifies the secret key.
Step 10	`Router(config-keyring)# exit`	Exits keyring configuration mode.
Step 11	`Router(config)# crypto ipsec transform-set transform-set-name transform1[transform2[transform3]]`	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPsec security protocols and algorithms. Accepted values are described in the *Cisco IOS Security Command Reference*.
Step 12	`Router(config-crypto-trans)# exit`	Exits crypto transform configuration mode
Step 13	`Router(config)# crypto isakmp policy priority`	Defines an IKE policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest.
Step 14	`Router(config-isakmp)# authentication pre-share`	Specifies the authentication method with an IKE policy. • **pre-share**—Specifies preshared keys as the authentication method.
Step 15	`Router(config-isakmp)# lifetime seconds`	Specifies the lifetime of an IKE SA. • *seconds*—Number of seconds each SA should exist before expiring. Use an integer from 60 to 86,400 seconds. Default is 86,400 (one day.)
Step 16	`Router(config-isakmp)# exit`	Exits ISAKMP policy configuration mode.

	Command	Purpose
Step 17	`Router(config)# crypto ipsec profile profile-name`	Defines an IPsec profile and enters IPsec profile configuration mode. The IPsec profile defines the IP Security (IPsec) parameters that are to be used for IPsec encryption between two IPsec routers. • *profile-name*—Name of the user profile.
Step 18	`Router(config-ipsec-profile)# set transform-set transform-set-name [transform-set-name2 ...transform-set-name6]`	Specifies which transform sets can be used with the crypto map entry. • *transform-set-name*—Name of the transform set.
Step 19	`Router(config)# interface type slot/[subslot]/port`	Configures an interface type. • *type*—Type of interface being configured. • *slot*/[*subslot*]/*port*—Number of the slot, subslot (optional), and port to be configured.
Step 20	`Router(config-if)# ip vrf forwarding vrf-name`	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 21	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask.
Step 22	`Router(config-if)# tunnel mode ipsec ipv4`	Defines the mode for the tunnel as IPsec and the transport as IPv4.
Step 23	`Router(config-if)# tunnel source ip-address`	Sets the source address of a tunnel interface. • *ip-address*—IP address to use as the source address for packets in the tunnel.
Step 24	`Router(config-if)# tunnel destination ip-address`	Sets the destination address of a tunnel interface. • *ip-address*—IP address to use as the destination address for packets in the tunnel.
Step 25	`Router(config-if)# tunnel vrf vrf-name`	(Optional) Associates a VPN routing and forwarding instance (VRF) with a specific tunnel destination. This step is only required when configuring a front door VRF (FVRF). • *vrf-name*—Name assigned to the VRF.
Step 26	`Router(config-if)# tunnel protection ipsec profile name`	Associates a tunnel interface with an IPsec profile. • *name*—Name of the IPsec profile; this value must match the name specified in the **crypto ipsec profile** command in Step 1.
Step 27	`Router(config-if)# crypto engine slot slot/subslot inside`	Assigns the specified crypto engine to the interface. • *slot/subslot*—The slot where the IPsec VPN SPA is located.
Step 28	`Router(config-if)# interface type slot/subslot/port`	Configures the physical egress interface.

	Command	Purpose
Step 29	`Router(config-if)# ip vrf forwarding vrf-name`	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 30	`Router(config-if)# ip address address mask`	Sets a primary or secondary IP address for an interface. • *address*—IP address. Enter the value specified in Step 23. • *mask*—Subnet mask.
Step 31	`Router(config-if)# crypto engine outside`	Assigns the crypto engine to the interface.
Step 32	`Router(config-if)# no shutdown`	Enables the interface.
Step 33	`Router(config-if)# exit`	Exits interface configuration mode.

Verifying the IPsec Virtual Tunnel Interface Configuration

To confirm that your IPsec virtual tunnel interface configuration is working properly, enter the **show interfaces tunnel**, **show crypto session**, and **show ip route** commands.

The **show interfaces tunnel** command displays tunnel interface information, the **show crypto session** command displays status information for active crypto sessions, and the **show ip route** command displays the current state of the routing table.

In this display the Tunnel 0 is up and the line protocol is up. If the line protocol is down, the session is not active.

```
Router1# show interfaces tunnel 0

Tunnel0 is up, line protocol is up
Hardware is Tunnel
Internet address is 10.0.51.203/24
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
reliability 255/255, txload 103/255, rxload 110/255
Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 10.0.149.203, destination 10.0.149.217
Tunnel protocol/transport IPSEC/IP, key disabled, sequencing disabled
Tunnel TTL 255
Checksumming of packets disabled, fast tunneling enabled
Tunnel transmit bandwidth 8000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
Tunnel protection via IPSec (profile "P1")
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 1/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/0 (size/max)
30 second input rate 13000 bits/sec, 34 packets/sec
30 second output rate 36000 bits/sec, 34 packets/sec
191320 packets input, 30129126 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
59968 packets output, 15369696 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
```

```
Router1# show crypto session
Crypto session current status
Interface: Tunnel0
Session status: UP-ACTIVE
Peer: 10.0.149.217 port 500
IKE SA: local 10.0.149.203/500 remote 10.0.149.217/500 Active
IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
Active SAs: 4, origin: crypto map

Router1# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C 10.0.35.0/24 is directly connected, Ethernet3/3
S 10.0.36.0/24 is directly connected, Tunnel0
C 10.0.51.0/24 is directly connected, Tunnel0
C 10.0.149.0/24 is directly connected, Ethernet3/0
```

For more complete information about IPsec Virtual Tunnel Interface, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/12_3t/12_3t14/feature/guide/gtIPSctm.html

For IPsec Virtual Tunnel Interface configuration examples, see the "IPsec Virtual Tunnel Interfaces Configuration Examples" section on page 22-36.

Configuring VTI in the Global Context

With Cisco IOS Release 12.2(33)SXH and later releases, you can configure IPsec VTI without having to configure VRFs. Although VRF mode must be configured globally using the **crypto engine mode vrf** command, tunnels can be terminated in the global context rather than in VRFs.

The configuration steps for VTI in the global context are similar to the steps for IPsec VTI shown in the "Configuring an IPsec Static Tunnel" section on page 22-18 with the exception that the **ip vrf forwarding** *vrf-name* command and the **tunnel vrf** *vrf-name* command are not required.

For a configuration example of IPsec VTI in the global context, see the "IPsec Virtual Tunnel Interfaces Configuration Examples" section on page 22-36.

Configuration Examples

The following sections provide examples of VRF mode configurations:

- IPsec Virtual Tunnel Interfaces Configuration Examples, page 22-36

Note When the **ip vrf forwarding** command is applied to a VLAN, any previously existing IP address assigned to that VLAN is removed. To assign an IP address to the VLAN, enter the **ip address** command after the **ip vrf forwarding** command, not preceding it.

Note The following examples use commands at the level of Cisco IOS Release 12.2(33)SXH.

In Cisco IOS Release 12.2(33)SXH and later releases, the **crypto engine subslot** command used in previous releases has been replaced with the **crypto engine slot** command (of the form **crypto engine slot** *slot/subslot* {**inside** | **outside**}). The **crypto engine subslot** command is no longer supported. In Cisco IOS Release 12.2(33)SXI and later releases, do not specify the **slot** *slot/subslot* information with the **outside** keyword. When upgrading, ensure that this command has been modified in your start-up configuration to avoid extended maintenance time.

VRF Mode Basic Configuration Example

The following example shows a basic IPsec VPN SPA configuration using VRF mode:

Switch 1 Configuration

```
hostname router-1
!
ip vrf ivrf
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
crypto engine mode vrf
!
vlan 2,3
!
crypto keyring key0
  pre-shared-key address 11.0.0.2 key 12345
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
!
crypto isakmp profile prof1
   vrf ivrf
   keyring key0
   match identity address 11.0.0.2 255.255.255.255
!
!
crypto ipsec transform-set proposal1  esp-3des esp-sha-hmac
!
crypto map testtag local-address Vlan3
crypto map testtag 10 ipsec-isakmp
 set peer 11.0.0.2
 set transform-set proposal1
 set isakmp-profile prof1
 match address 101
```

```
!
interface GigabitEthernet1/1
 !switch inside port
 ip vrf forwarding ivrf
 ip address 12.0.0.1 255.255.255.0
!
!
interface GigabitEthernet1/2
 !switch outside port
 switchport
 switchport access vlan 3
 switchport mode access
!
interface GigabitEthernet4/0/1
 !IPsec VPN SPA inside port
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 !IPsec VPN SPA outside port
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface Vlan2
 ip vrf forwarding ivrf
 ip address 13.0.0.252 255.255.255.0
 crypto map testtag
 crypto engine slot 4/0 inside
!
interface Vlan3
 ip address 11.0.0.1 255.255.255.0
 crypto engine slot 4/0 outside
!
access-list 101 permit ip host 12.0.0.2 host 13.0.0.2
```

Switch 2 Configuration

```
hostname router-2
!
ip vrf ivrf
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
crypto engine mode vrf
!
vlan 2,3
!
crypto keyring key0
  pre-shared-key address 11.0.0.1 key 12345
```

```
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
!
crypto isakmp profile prof1
   vrf ivrf
   keyring key0
   match identity address 11.0.0.1 255.255.255.255
!
!
crypto ipsec transform-set proposal1  esp-3des esp-sha-hmac
!
crypto map testtag local-address Vlan3
crypto map testtag 10 ipsec-isakmp
 set peer 11.0.0.1
 set transform-set proposal1
 set isakmp-profile prof1
 match address 101
!
interface GigabitEthernet1/1
 !switch inside port
 ip vrf forwarding ivrf
 ip address 13.0.0.1 255.255.255.0
!
interface GigabitEthernet1/2
 !switch outside port
 switchport
 switchport access vlan 3
 switchport mode access
!
interface GigabitEthernet4/0/1
 !IPsec VPN SPA inside port
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 !IPsec VPN SPA outside port
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface Vlan2
 ip vrf forwarding ivrf
 ip address 12.0.0.252 255.255.255.0
 crypto map testtag
 crypto engine slot 4/0 inside
!
interface Vlan3
 ip address 11.0.0.2 255.255.255.0
 crypto engine slot 4/0 outside
!
```

```
access-list 101 permit ip host 13.0.0.2 host 12.0.0.2
```

VRF Mode Remote Access Using Easy VPN Configuration Example

The following examples show VRF mode configurations for remote access using Easy VPN, first using RADIUS authentication, then using local authentication:

Using RADIUS Authentication

```
aaa group server radius acs-vrf1
 server-private 192.1.1.251 auth-port 1812 acct-port 1813 key allegro
 ip vrf forwarding vrf1
!
aaa authentication login test_list group acs-vrf1
aaa authorization network test_list group acs-vrf1
aaa accounting network test_list start-stop group acs-vrf1
!
ip vrf ivrf
 rd 1:1
 route-target export 1:1
 route-target import 1:1
!
!
crypto isakmp policy 5
 encr 3des
 authentication pre-share
 group 2

crypto isakmp client configuration group test
 key world
 pool pool1
!
crypto isakmp profile test_pro
   vrf ivrf
   match identity group test
   client authentication list test_list
   isakmp authorization list test_list
   client configuration address respond
   accounting test_list
crypto ipsec transform-set t3 esp-3des esp-sha-hmac
!
crypto dynamic-map remote 1
 set transform-set t3
 set isakmp-profile test_pro
 reverse-route
!
!
crypto map map-ra local-address GigabitEthernet2/1
crypto map map-ra 10 ipsec-isakmp dynamic remote
!
interface GigabitEthernet2/1
  mtu 9216
 ip address 120.0.0.254 255.255.255.0
 ip flow ingress
 logging event link-status
 mls qos trust ip-precedence
 crypto engine slot 1/0 outside
!
interface GigabitEthernet1/0/1
```

```
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,100,1002-1005
 switchport mode trunk
 mtu 9216
 mls qos trust ip-precedence
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet1/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 mls qos trust ip-precedence
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!

interface Vlan100
 ip vrf forwarding vrf1
 ip address 120.0.0.100 255.255.255.0
 no mop enabled
 crypto map map-ra
 crypto engine slot 1/0 inside

ip local pool pool1 100.0.1.1 100.0.5.250
```

Using Local Authentication

```
username t1 password 0 cisco
aaa new-model
!
aaa authentication login test_list local
aaa authorization network test_list local
!
aaa session-id common
!
ip vrf ivrf
 rd 1:2
 route-target export 1:2
 route-target import 1:2


!
crypto isakmp policy 5
 encr 3des
 authentication pre-share
 group 2
!
crypto isakmp client configuration group test
 key world
 pool pool1
crypto isakmp profile test_pro
   vrf ivrf
   match identity group test
   client authentication list test_list
   isakmp authorization list test_list
   client configuration address respond
```

```
   accounting test_list
crypto ipsec transform-set t3 esp-3des esp-sha-hmac
!
crypto dynamic-map remote 10
 set transform-set t3
 set isakmp-profile test_pro
 reverse-route

!
!
crypto map map-ra local-address GigabitEthernet2/1
crypto map map-ra 11 ipsec-isakmp dynamic remote
!
!

!
interface GigabitEthernet2/1
  mtu 9216
 ip address 120.0.0.254 255.255.255.0
 ip flow ingress
 logging event link-status
 mls qos trust ip-precedence
 crypto engine slot 1/0 outside
!
!
interface GigabitEthernet1/0/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,100,1002-1005
 switchport mode trunk
 mtu 9216
 mls qos trust ip-precedence
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet1/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 mls qos trust ip-precedence
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface Vlan100
 ip vrf forwarding ivrf
 ip address 120.0.0.100 255.255.255.0
 ip flow ingress
 crypto map map-ra
 crypto engine slot 1/0 inside
!
!
ip local pool pool1 100.0.1.1 100.0.5.250
```

VRF Mode PE Configuration Example

The following example shows a VRF mode configuration for a provider edge (PE):

```
!
version 12.2
!
hostname EXAMPLE-PE
!
no aaa new-model
ip subnet-zero
!
ip vrf vrf1
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
crypto engine mode vrf
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
  auto-sync standard
spanning-tree mode pvst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
!
power redundancy-mode combined
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
crypto keyring key0
  pre-shared-key address 11.0.0.1 key mykey
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 lifetime 500
crypto isakmp profile prof1
   vrf vrf1
   keyring key0
   self-identity user-fqdn a@example.com
   match identity address 11.0.0.1 255.255.255.255
!
crypto ipsec transform-set proposal1 ah-sha-hmac esp-3des esp-sha-hmac
!
crypto map testtag local-address Vlan3
crypto map testtag 10 ipsec-isakmp
 set peer 11.0.0.1
 set security-association lifetime seconds 1000
 set transform-set proposal1
 set pfs group2
 set isakmp-profile prof1
 match address 101
!
interface GigabitEthernet1/1
 no ip address
 shutdown
!
interface GigabitEthernet1/2
```

```
 switchport
 switchport access vlan 3
 switchport mode access
 no ip address
!
interface GigabitEthernet1/14
 ip vrf forwarding vrf1
 ip address 13.0.0.1 255.255.255.0
!
interface GigabitEthernet6/0/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2
 switchport mode trunk
 mtu 9216
 no ip address
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet6/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan none
 switchport mode trunk
 mtu 9216
 no ip address
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet7/1
 no ip address
 shutdown
!
interface GigabitEthernet7/2
 ip address 17.1.5.4 255.255.0.0
 media-type rj45
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan2
 ip vrf forwarding vrf1
 ip address 12.0.0.252 255.255.255.0
 crypto map testtag
 crypto engine subslot 6/0
!
interface Vlan3
 ip address 11.0.0.2 255.255.255.0
 crypto engine subslot 6/0
!
ip classless
ip route 223.255.254.0 255.255.255.0 17.1.0.1
!
no ip http server
!
access-list 101 permit ip host 13.0.0.2 host 12.0.0.2
!
control-plane
!
dial-peer cor custom
!
```

```
line con 0
 exec-timeout 0 0
line vty 0 4
 login
!
end
```

VRF Mode CE Configuration Example

The following example shows a VRF mode configuration for a customer edge (CE):

```
!
version 12.2
!
hostname EXAMPLE-CE
!
no aaa new-model
ip subnet-zero
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
  auto-sync standard
spanning-tree mode pvst
!
power redundancy-mode combined
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 lifetime 500
crypto isakmp key mykey address 11.0.0.2
!
crypto ipsec transform-set proposal1 ah-sha-hmac esp-3des esp-sha-hmac
!
crypto map testtag 10 ipsec-isakmp
 set peer 11.0.0.2
 set security-association lifetime seconds 1000
 set transform-set proposal1
 set pfs group2
 match address 101
!
interface GigabitEthernet1/1
 ip address 12.0.0.1 255.255.255.0
 load-interval 30
 no keepalive
!
interface GigabitEthernet1/2
 switchport
 switchport access vlan 3
 switchport mode access
 no ip address
!
interface GigabitEthernet5/2
 ip address 17.1.5.3 255.255.0.0
 media-type rj45
```

```
!
interface GigabitEthernet6/0/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2
 switchport mode trunk
 mtu 9216
 no ip address
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet6/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 3
 switchport mode trunk
 mtu 9216
 no ip address
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet6/1/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan none
 switchport mode trunk
 mtu 9216
 no ip address
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet6/1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan none
 switchport mode trunk
 mtu 9216
 no ip address
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan2
 ip address 11.0.0.1 255.255.255.0
 no mop enabled
 crypto map testtag
 crypto engine subslot 6/0
!
interface Vlan3
 no ip address
 crypto connect vlan 2
!
ip classless
ip route 13.0.0.0 255.0.0.0 11.0.0.2
ip route 223.255.254.0 255.255.255.0 17.1.0.1
!
no ip http server
```

```
!
access-list 101 permit ip host 12.0.0.2 host 13.0.0.2
!
control-plane
!
dial-peer cor custom
!
line con 0
 exec-timeout 0 0
line vty 0 4
 login
!
end
```

VRF Mode Tunnel Protection Configuration Example

The following example shows a VRF mode configuration with tunnel protection:

```
ip vrf coke
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
crypto keyring key1
 pre-shared-key address 100.1.1.1 key happy-eddie
!
crypto isakmp policy 1
 authentication pre-share

crypto isakmp profile prof1
 keyring key1
 match identity address 100.1.1.1 255.255.255.255
!
crypto ipsec transform-set TR esp-des esp-md5-hmac
 mode transport
!
crypto ipsec profile tp
 set transform-set TR
 set isakmp-profile prof1
!
!
crypto engine mode vrf
!
interface Tunnel1
 ip vrf forwarding coke
 ip address 10.1.1.254 255.255.255.0
 tunnel source 172.1.1.1
 tunnel destination 100.1.1.1
 tunnel protection ipsec profile tp
 crypto engine slot 4/0 inside
!
interface GigabitEthernet4/0/1
 !IPsec VPN SPA inside port
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 cdp enable
 spanning-tree portfast trunk
```

```
!
interface GigabitEthernet4/0/2
 !IPsec VPN SPA outside port
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 cdp enable
 spanning-tree portfast trunk
!
interface GigabitEthernet6/1
 ip address 172.1.1.1 255.255.255.0
 crypto engine slot 4/0 outside
!
interface FastEthernet7/13
 ip vrf forwarding coke
 ip address 13.1.1.2 255.255.255.0
!
ip route 100.1.1.1 255.255.255.255 Tunnel1
```

IP Multicast in VRF Mode Configuration Example

Note If two IPsec VPN SPAs are present in the Cisco 7600 SSC-400, one will be shut down if the **hw-module slot** *X* **subslot** *Y* **only** command is in the configuration. In this case, the IPsec VPN SPA in subslot Y will be active, and the IPsec VPN SPA in the other subslot will be disabled.

The following example shows how to configure IP multicast over GRE:

```
hostname router-1
!
ip vrf ivrf
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
!
!
ip multicast-routing vrf ivrf
!
crypto engine mode vrf
!
!
hw-module slot 4 subslot 0 only
!
crypto keyring key1
  pre-shared-key address 11.0.0.0 255.0.0.0 key 12345
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
crypto isakmp profile isa_prof
   keyring key1
   match identity address 11.0.0.0 255.0.0.0
```

```
!
crypto ipsec transform-set proposal esp-3des
 mode transport
!
crypto ipsec profile vpnprof
 set transform-set proposal
 set isakmp-profile isa_prof
!
!
!
interface Tunnel1
 ip vrf forwarding ivrf
 ip address 20.1.1.1 255.255.255.0
 ip mtu 9216
 ip hold-time eigrp 1 3600
 ip pim sparse-mode
 tunnel source 1.0.1.1
 tunnel destination 11.1.1.1
 tunnel protection ipsec profile vpnprof
 crypto engine slot 4/0 inside
!
interface Loopback1
 ip address 1.0.1.1 255.255.255.0
!
interface GigabitEthernet1/1
 mtu 9216
 ip vrf forwarding ivrf
 ip address 50.1.1.1 255.0.0.0
 ip pim sparse-mode
!
interface GigabitEthernet1/2
 mtu 9216
 ip address 9.1.1.1 255.255.255.0
 crypto engine slot 4/0 outside
!
!
interface GigabitEthernet4/0/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
router eigrp 1
 !
 address-family ipv4 vrf ivrf
  autonomous-system 1
  network 20.1.1.0 0.0.0.255
  network 50.1.1.0 0.0.0.255
  no auto-summary
  no eigrp log-neighbor-changes
```

```
 exit-address-family
!
router ospf 1
 log-adjacency-changes
 network 1.0.0.0 0.255.255.255 area 0
 network 9.0.0.0 0.255.255.255 area 0
!
ip pim vrf ivrf rp-address 50.1.1.1
!
```

IPsec Virtual Tunnel Interfaces Configuration Examples

The following examples show VRF mode configurations that use VTI:

- IPsec Virtual Tunnel Interface FVRF Configuration Example, page 22-36
- IPsec Virtual Tunnel Interface in the Global Context Configuration Example, page 22-38
- IPsec Virtual Tunnel Interface Multicast Configuration Example, page 22-39

IPsec Virtual Tunnel Interface FVRF Configuration Example

The following example configuration shows an FVRF VTI configuration:

```
hostname router-1
!
!
ip vrf fvrf
 rd 2000:1
 route-target export 2000:1
 route-target import 2000:1
!
ip vrf ivrf
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
crypto engine mode vrf
!
crypto keyring key1 vrf fvrf
  pre-shared-key address 11.1.1.1 key cisco47
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
!
crypto isakmp profile isa_prof
   keyring key1
   match identity address 11.1.1.1 255.255.255.255 fvrf

crypto ipsec transform-set proposal esp-3des esp-sha-hmac
!
!
crypto ipsec profile vpnprof
 set transform-set proposal
 set isakmp-profile isa_prof
!
!
!
```

```
!
!
interface Tunnel1
 ip vrf forwarding ivrf
 ip address 20.1.1.1 255.255.255.0
 ip pim sparse-mode
 ip ospf network broadcast
 ip ospf priority 2
 tunnel source 1.0.0.1
 tunnel destination 11.1.1.1
 tunnel mode ipsec ipv4
 tunnel vrf fvrf
 tunnel protection ipsec profile vpnprof
 crypto engine slot 4/0 inside
!
interface Loopback1
 ip vrf forwarding fvrf
 ip address 1.0.0.1 255.255.255.0
!
interface GigabitEthernet1/1
 !switch inside port
 ip vrf forwarding ivrf
 ip address 50.0.0.1 255.255.255.0
!
interface GigabitEthernet1/2
 !switch outside port
 ip vrf forwarding fvrf
 ip address 9.1.1.1 255.255.255.0
 crypto engine slot 4/0 outside
!
interface GigabitEthernet4/0/1
 !IPsec VPN SPA inside port
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 !IPsec VPN SPA outside port
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
router ospf 1 vrf ivrf
 log-adjacency-changes
 network 20.1.1.0 0.0.0.255 area 0
 network 21.1.1.0 0.0.0.255 area 0
 network 50.0.0.0 0.0.0.255 area 0
!
ip classless
ip route vrf fvrf 11.1.1.0 255.255.255.0 9.1.1.254
```

IPsec Virtual Tunnel Interface in the Global Context Configuration Example

The following example configuration shows IPsec VTI configuration in the global context:

```
!
crypto engine mode vrf
!
crypto keyring key1
  pre-shared-key address 14.0.0.2 key 12345
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share

!
crypto isakmp profile prof1
   keyring key1
   match identity address 14.0.0.2 255.255.255.255
!
crypto ipsec transform-set t-set1 esp-3des esp-sha-hmac
!
crypto ipsec profile prof1
 set transform-set t-set1
 set isakmp-profile prof1
!
!
interface Tunnel1
 ip address 122.0.0.2 255.255.255.0
 tunnel source 15.0.0.2
 tunnel destination 14.0.0.2
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile prof1
 crypto engine slot 2/0 inside
!
interface Loopback2
 ip address 15.0.0.2 255.255.255.0
!

interface GigabitEthernet1/3
 ip address 172.2.1.1 255.255.255.0
 crypto engine slot 2/0 outside
!
interface GigabitEthernet2/0/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet2/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
```

```
!
ip route 14.0.0.0 255.0.0.0 172.2.1.2
ip route 172.0.0.0 255.0.0.0 172.2.1.2
```

IPsec Virtual Tunnel Interface Multicast Configuration Example

The following example shows how to configure multicast over VTI:

```
hostname router-1
!
ip vrf ivrf
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
!
!
ip multicast-routing vrf ivrf
!
crypto engine mode vrf
!
!
!
crypto keyring key1
  pre-shared-key address 11.0.0.0 255.0.0.0 key 12345
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
crypto isakmp profile isa_prof
   keyring key1
   match identity address 11.0.0.0 255.0.0.0
!
crypto ipsec transform-set proposal esp-3des
!
crypto ipsec profile vpnprof
 set transform-set proposal
 set isakmp-profile isa_prof
!
!
!
interface Tunnel1
 ip vrf forwarding ivrf
 ip address 20.1.1.1 255.255.255.0
 ip mtu 9216
 ip hold-time eigrp 1 3600
 ip pim sparse-mode
 tunnel source 1.0.1.1
 tunnel destination 11.1.1.1
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile vpnprof
 crypto engine slot 4/0 inside
!
interface Loopback1
 ip address 1.0.1.1 255.255.255.0
!
interface GigabitEthernet1/1
 mtu 9216
 ip vrf forwarding ivrf
```

```
 ip address 50.1.1.1 255.0.0.0
 ip pim sparse-mode
!
interface GigabitEthernet1/2
 mtu 9216
 ip address 9.1.1.1 255.255.255.0
 crypto engine slot 4/0 outside
!
!
interface GigabitEthernet4/0/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
router eigrp 1
 !
 address-family ipv4 vrf ivrf
  autonomous-system 1
  network 20.1.1.0 0.0.0.255
  network 50.1.1.0 0.0.0.255
  no auto-summary
  no eigrp log-neighbor-changes
 exit-address-family
!
router ospf 1
 log-adjacency-changes
 network 1.0.0.0 0.255.255.255 area 0
 network 9.0.0.0 0.255.255.255 area 0
!
ip pim vrf ivrf rp-address 50.1.1.1
!
```