Configuring NetFlow Accounting for Unicast and Multicast on GRE IP Tunnel Interfaces

Last Updated: November 28, 2011

This document contains information about and instructions for configuring NetFlow Accounting for Unicast and Multicast on generic routing encapsulation (GRE) IP Tunnel Interfaces. NetFlow multicast accounting allows you to capture multicast-specific data (both packets and bytes) for multicast flows.

NetFlow is a Cisco IOS application that provides statistics on packets flowing through a router. It is emerging as a primary network accounting and security technology.

Finding Feature Information

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

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.

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for Configuring NetFlow Accounting for Unicast and Multicast on GRE IP Tunnel Interfaces

- You must use the Per-interface NetFlow feature in conjunction with the NetFlow Accounting for Unicast and Multicast on GRE Tunnel Interfaces feature.
- The instructions for configuring IPv4 unicast routing are not included in this document. If you want to configure NetFlow accounting for IPv4 unicast traffic on a GRE IP interface, your switch must already be configured for IPv4 unicast routing.
- The instructions for configuring IPv4 multicast routing are not included in this document. If you want to configure NetFlow accounting for IPv4 multicast traffic on a GRE IP interface, your switch must already be configured for IPv4 multicast routing.

Restrictions for Configuring NetFlow Accounting for Unicast and Multicast on GRE IP Tunnel Interfaces

- Only Catalyst 6500 series switches with a supervisor 720 is supported.
- Multicast flow packet and byte counters will be updated only in PFC3B mode and above.
- Only hardware switched flows are supported.
- Only Version 9 NetFlow data export format is supported.

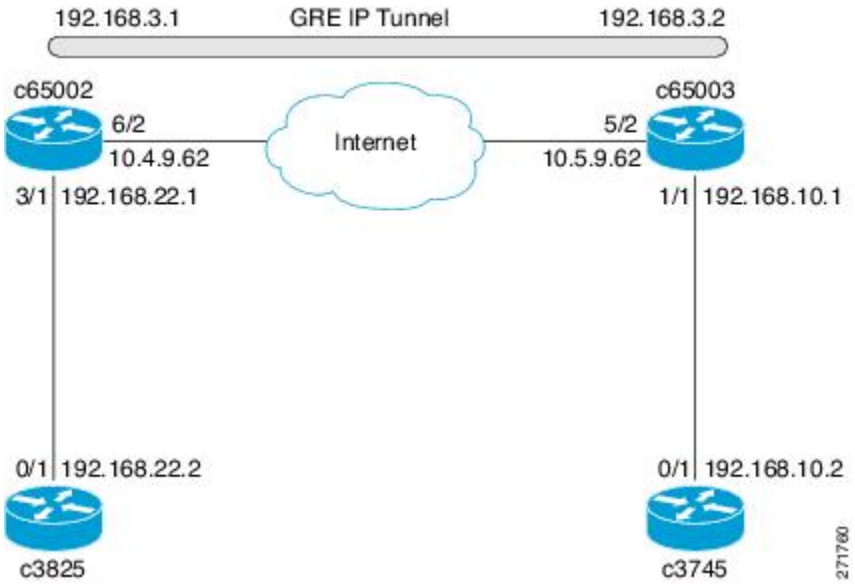
Information About NetFlow Accounting for Unicast and Multicast on GRE IP Tunnel Interfaces

GRE Tunneling

Generic routing encapsulation (GRE) tunneling is defined in RFC 2784. GRE is a carrier protocol that can be used with a variety of underlying transport protocols and that can carry a variety of passenger protocols. RFC 2784 also covers the use of GRE with IPv4 as the transport protocol and the passenger protocol. For

more information on GRE tunnels, see the *Cisco IOS Interface and Hardware Component Configuration Guide* . The figure below is an example of a typical implementation of a GRE IP tunnel.

Figure 1 Sample Network with a GRE IPv4 Tunnel



GRE Tunnel Keepalive

Keepalive packets can be configured to be sent over IP-encapsulated GRE tunnels. You can specify the rate at which keepalives will be sent and the number of times that a device will continue to send keepalive packets without a response before the interface becomes inactive. GRE keepalive packets may be sent from both sides of a tunnel or from just one side.

Tunnel Interfaces

A tunnel interface is used to pass protocol traffic across a network that does not normally support the protocol. To build a tunnel requires defining a tunnel interface on each of two routers. The tunnel interfaces must reference each other. At each router, the tunnel interface must be configured with a Layer 3 address. The tunnel endpoints, tunnel source, and tunnel destination must be defined, and the type of tunnel must be selected. Optional steps can be performed to customize the tunnel.

Remember to configure the router at each end of the tunnel. If only one side of a tunnel is configured, the tunnel interface may still come up and stay up (unless keepalive is configured), but packets going into the tunnel will be dropped.

In Cisco IOS Release 12.2(8)T and later releases, Cisco express forwarding (CEF) switching over multipoint GRE tunnels was introduced. Previously, only process switching was available for multipoint GRE tunnels.

NetFlow Accounting on GRE IP Tunnel Interfaces

To analyze traffic that is sent from c3825 to c3745 in the figure below, NetFlow accounting is configured as shown in the table below. The flows in the "Flows" column are shown in the Unicast IPv4 Traffic over

an IPv4 Unicast GRE Tunnel (Encapsulation) figure through the Multicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (De-encapsulation) figure.

Table 1 Where to Configure NetFlow Accounting and Which NetFlow Commands to Configure

Encapsulation/ De-encapsulation	Router	Ingress Physical Interface	Ingress Tunnel Interface	Egress Physical Interface	Egress Tunnel Interface	Flows
Traffic Direction						
Unicast over GRE (encap)	C650002 C3825 to C3745	**ip flow ingress** on interface gigabit 3/1	No configuration	No configuration	**ip flow egress** on interface tunnel 0	Flow (1) Flow (2)
Unicast over GRE (decap)	C65003 C3825 to C3745	**ip flow ingress** on interface gigabit 5/2	**ip flow ingress** on interface tunnel 0	No configuration	No configuration	Flow (1) Flow (2)
Multicast over GRE (encap)	C650002 C3825 to 3C745	**ip flow ingress** on interface gigabit 3/1	No configuration	**ip flow egress** on interface 6/2	**ip flow egress** on interface tunnel 0	Flow (1) Flow (2) Flow (3)
Multicast over GRE (decap)	C65003 C3825 to C3745	**ip flow ingress** on interface gigabit 5/2	**ip flow ingress** on interface tunnel 0	**ip flow egress** on interface 1/1	No configuration	Flow (1) Flow (2) Flow (3)

When you configure NetFlow accounting for IPv4 unicast traffic on a GRE tunnel interface, the traffic that is encapsulated or de-encapsulated on the router results in the creation of two flows. See the Unicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (Encapsulation) figure and the Unicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (De-encapsulation) figure. When you configure NetFlow accounting for IPv4 multicast traffic on a GRE tunnel interface, the traffic that is encapsulated or de-encapsulated on the router results in the creation of three flows. See the Multicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (Encapsulation) figure and the Multicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (De-encapsulation) figure. The increase in the number of flows created results in an increase in the usage of the hardware NetFlow table. You must monitor the hardware NetFlow table on your router to ensure that it is not oversubscribed.

If you are using NetFlow data export, the number of exported flows is also increased. Flows from the hardware table are converted to the Version 9 export format and then exported. Because the number of flows is doubled when you configure NetFlow Data Export, twice as much memory is required to convert the flows to Version 9 export format and then export them.

The table below provides the definitions of the terms used in the figures below.

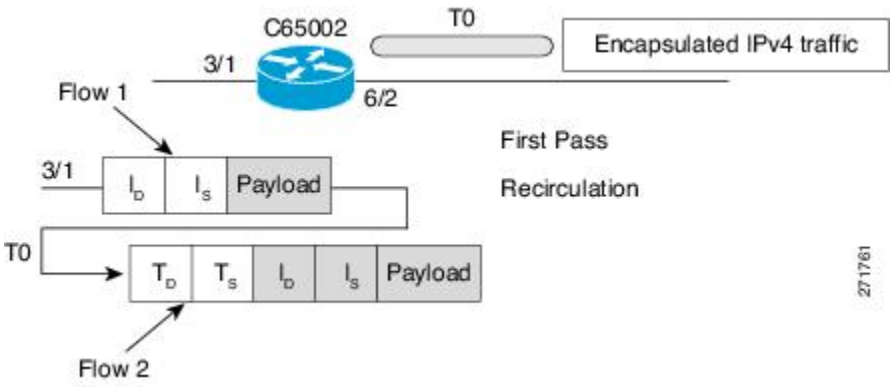
Table 2 Definition of Terms Used in Figures 2 through 5

Term	Definition
encapsulation	Adding the GRE tunnel header and trailer to the beginning and end respectively, of the packet being transmitted over the GRE tunnel.

Term	Definition
de-encapsulation	Removing the GRE tunnel header and trailer from the beginning and end respectively, of the packet being received from the GRE tunnel.
ingress	The inbound path of traffic. For example, the ingress interface is the interface over which traffic is received.
egress	The outbound path of traffic. For example, the egress interface is the interface over which traffic is transmitted.
ID	Destination IP address.
IS	Source IP address.
TD	Destination IP address for the tunnel interface.
TS	Source IP address for the tunnel interface.
MD	Multicast destination IP address.
MS	Multicast source IP address.
payload	The packet data.

The figure below shows the packet encapsulation process for unicast IPv4 traffic that is received on interface Gigabit Ethernet 3/1 on c65002 in the figure above. The first flow is the result of NetFlow accounting for the traffic after it is received on physical interface 3/1 (ingress NetFlow). The second flow is the result of NetFlow accounting for the traffic as it is being transmitted on the GRE tunnel interface T0 (egress NetFlow).

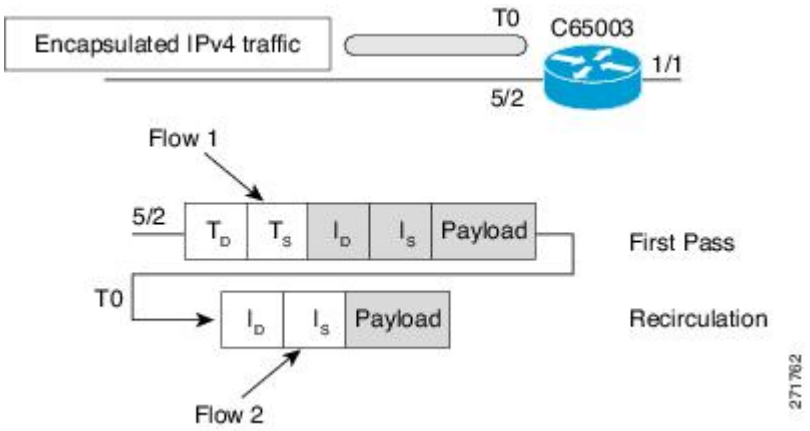
Figure 2 Unicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (Encapsulation)



The figure below shows the packet de-encapsulation process for unicast IPv4 traffic that is received on interface Gigabit Ethernet 3/1 on c65002 in the Sample Network with a GRE IPv4 Tunnel figure. The first flow is the result of NetFlow accounting for the traffic after it is received on the physical interface 5/2

(ingress NetFlow). The second flow is the result of NetFlow accounting for the traffic as it is being received and de-encapsulated on the tunnel interface T0 (ingress NetFlow).

Figure 3 Unicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (De-encapsulation)

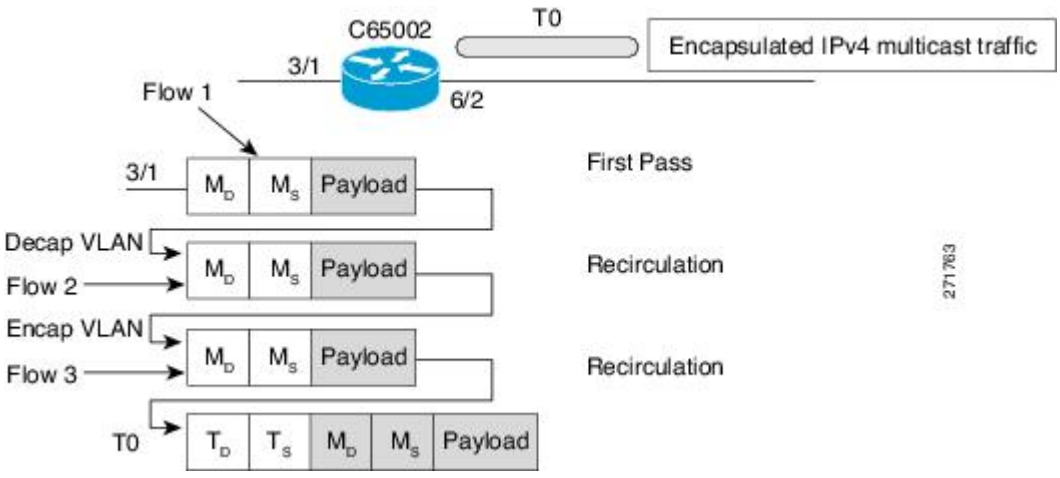


During de-encapsulation, only ingress features of the tunnel are applied on the packets, and during encapsulation, only egress features of the tunnel are applied.

Multicast replication can happen in either ingress or egress mode. GRE encapsulation of multicast flows is done on the line card on which the ingress physical interface resides, irrespective of the ingress or egress replication mode. So in the case of both ingress and egress multicast replication modes, egress flows are created on the ingress line card.

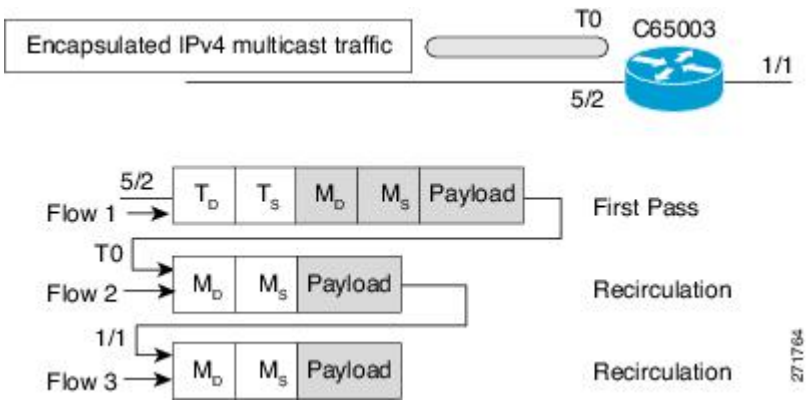
The examples in the figures below show how and why multiple flows are created during GRE handling of packets. In the figure below, Flow 1 is created when packets are received by physical interface 3/1. Flows 2 and 3 are created as part the multicast replication process using the internal virtual local area networks (VLANs) that are required for NetFlow accounting to keep track of the multicast traffic.

Figure 4 Multicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (Encapsulation)



In the figure below, Flow 1 is created when packets are received over physical interface 5/2. Flow 2 is created as part of the de-encapsulation process. Flow 3 is created as the multicast traffic is replicated and forwarded on interface 1/1.

Figure 5 Multicast IPv4 Traffic over an IPv4 Unicast GRE Tunnel (De-encapsulation)

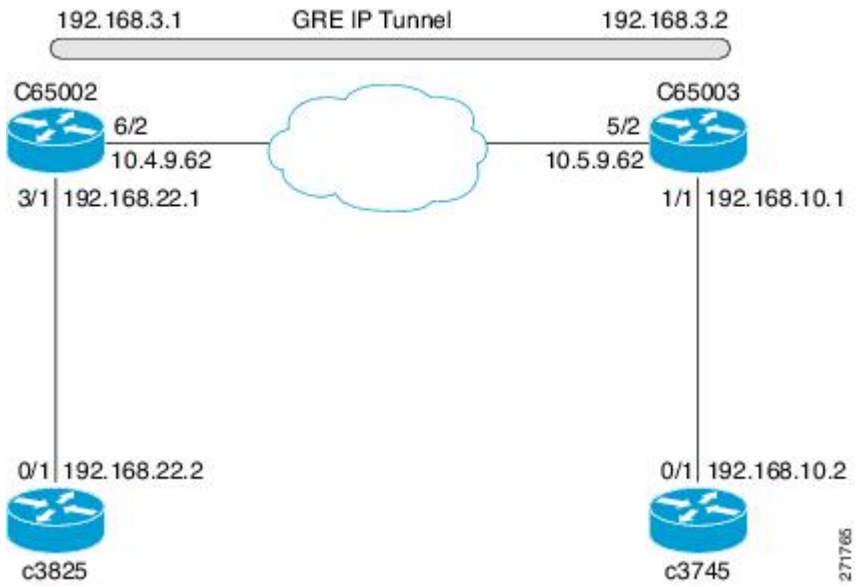


How to Configure NetFlow Accounting for Unicast and Multicast on GRE Tunnel Interfaces

Sample Network

The tasks in this section use the sample network shown in the figure below.

Figure 6 Sample Network with a GRE IPv4 Tunnel



Configuring a GRE IP Tunnel

To configure a GRE IP tunnel as shown in Configuring a GRE IP Tunnel, page 8, perform the task in this section.

Ensure that the physical interface to be used as the tunnel source in this task is up and configured with the appropriate IP address. For hardware technical descriptions and information about installing interfaces, see the hardware installation and configuration documentation for your product.

Note GRE tunnel keepalive is not supported in cases where virtual route forwarding (VRF) is applied to a GRE tunnel.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **bandwidth** *kbps*
5. **ip address** *address mask*
6. **keepalive** [*period* [*retries*]]
7. **tunnel source** {*ip-address* | *interface-type interface-number*}
8. **tunnel destination** {*hostname* | *ip-address*}
9. **tunnel key** *key-number*
10. **tunnel mode gre ip**
11. **ip mtu** *bytes*
12. **ip tcp mss** *mss-value*
13. **tunnel path-mtu-discovery** [**age-timer** {*aging-mins*| **infinite**}]
14. **end**
15. Repeat steps 1-14 on the router that hosts the other end of the GRE tunnel

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface tunnel 0`	Specifies the interface type and number and enters interface configuration mode. • To configure a tunnel, use *tunnel*for the *type* argument.

	Command or Action	Purpose
Step 4	**bandwidth** *kbps* **Example:** `Router(config-if)# bandwidth 1000`	Sets the current bandwidth value for an interface and communicates it to higher-level protocols. Specifies the tunnel bandwidth to be used to transmit packets. • Use the *kbps*argument to set the bandwidth, in kilobits per second (kbps). **Note** This is a routing parameter only; it does not affect the physical interface. The default bandwidth setting on a tunnel interface is 9.6 kbps. You should set the bandwidth on a tunnel to an appropriate value.
Step 5	**ip address** *address mask* **Example:** `Router(config-if)# ip address 192.168.3.1 255.255.255.0`	Specifies an IP address for the interface.
Step 6	**keepalive** [*period* [*retries*]] **Example:** `Router(config-if)# keepalive 3 7`	(Optional) Specifies the number of times that the device will continue to send keepalive packets without response before bringing the tunnel interface protocol down. • GRE keepalive packets may be configured either on only one side of the tunnel or on both. • If GRE keepalive is configured on both sides of the tunnel, the *period* and *retries* arguments can be different at each side of the link. **Note** This command is supported only on GRE point-to-point tunnels. **Note** The GRE tunnel keepalive feature should not be configured on a VRF tunnel. This combination of features is not supported.
Step 7	**tunnel source** {*ip-address* | *interface-type interface-number*} **Example:** `Router(config-if)# tunnel source GigabitEthernet6/2`	Configures the tunnel source. • Use the *ip-address*argument to specify the source IP address. • Use the *interface-type*and *interface-number*arguments to specify the interface to use. **Note** The tunnel source and destination IP addresses must be defined on two separate devices.
Step 8	**tunnel destination** {*hostname* | *ip-address*} **Example:** `Router(config-if)# tunnel destination 10.5.9.62`	Configures the tunnel destination. • Use the *hostname*argument to specify the name of the host destination. • Use the *ip-address*argument to specify the IP address of the host destination. **Note** The tunnel source and destination IP addresses must be defined on two separate devices.

	Command or Action	Purpose
Step 9	**tunnel key** *key-number* **Example:** `Router(config-if)# tunnel key 1000`	(Optional) Enables an ID key for a tunnel interface. • Use the *key-number*argument to identify a tunnel key that is carried in each packet. • Tunnel ID keys can be used as a form of weak security to prevent improper configuration or injection of packets from a foreign source. **Note** This command is supported only on GRE tunnel interfaces. We do not recommend relying on this key for security purposes.
Step 10	**tunnel mode gre ip** **Example:** `Router(config-if)# tunnel mode gre ip`	Specifies **GRE IP** as the encapsulation protocol to be used in the tunnel.
Step 11	**ip mtu** *bytes* **Example:** `Router(config-if)# ip mtu 1400`	(Optional) Set the maximum transmission unit (MTU) size of IP packets sent on an interface. • If an IP packet exceeds the MTU set for the interface, the Cisco IOS software will fragment it unless the don't fragment (DF) bit is set. • All devices on a physical medium must have the same protocol MTU in order to operate. **Note** If the **tunnel path-mtu-discovery** command is going to be enabled in Configuring a GRE IP Tunnel, page 8, do not configure this command.
Step 12	**ip tcp mss** *mss-value* **Example:** `Router(config-if)# ip tcp mss 250`	(Optional) Specifies the maximum segment size (MSS) for TCP connections that originate or terminate on a router. • Use the *mss-value* argument to specify the maximum segment size for TCP connections, in bytes.
Step 13	**tunnel path-mtu-discovery** [**age-timer** {*aging-mins*| **infinite**}] **Example:** `Router(config-if)# tunnel path-mtu-discovery`	(Optional) Enables Path MTU Discovery (PMTUD) on a GRE or IP-in-IP tunnel interface. • When PMTUD is enabled on a tunnel interface, PMTUD will operate for GRE IP tunnel packets to minimize fragmentation in the path between the tunnel endpoints.
Step 14	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 15	Repeat steps 1-14 on the router that hosts the other end of the GRE tunnel	--

Verifying the Status of the GRE IP Tunnel

To verify the tunnel configuration and operation, perform the following optional task:

SUMMARY STEPS

1. **enable**
2. **ping** *ip-address*
3. **ping** *ip-address*
4. **show interfaces tunnel** *number* [**accounting**

DETAILED STEPS

Step 1 **enable**
Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 **ping** *ip-address*
To verify that each router has IP connectivity to the tunnel endpoint on the other router, ping the IP address of the remote tunnel endpoint from the local router.

Example:

```
c65002# ping
 192.168.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/20 ms
```

Step 3 **ping** *ip-address*
To verify that each router has IP connectivity to the tunnel endpoint on the other router, ping the IP address of the remote tunnel endpoint from the local router.

Example:

```
c65003# ping
 192.168.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:
!!!!!
```

Step 4 **show interfaces tunnel** *number* [**accounting**
Displays the status and statistics of the tunnel interface

Example:

```
c65002# show interface tunnel 0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 192.168.3.1/24
  MTU 1514 bytes, BW 1000 Kbit, DLY 50000 usec,
     reliability 255/255, txload 115/255, rxload 57/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 10.4.9.62 (GigabitEthernet6/2), destination 10.5.9.62
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255, Fast tunneling enabled
  Path MTU Discovery, ager 10 mins, min MTU 92
  Last input 00:07:35, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 4139000 bits/sec, 659 packets/sec
  5 minute output rate 4117000 bits/sec, 669 packets/sec
  L2 Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes
  L3 in Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes mcast
  L3 out Switched: ucast: 0 pkt, 0 bytes mcast: 0 pkt, 0 bytes
     245049 packets input, 192533770 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     251500 packets output, 196216398 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Configuring NetFlow Accounting on a GRE IP Tunnel Interface

To configure NetFlow on a GRE IP tunnel interface, perform the following task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast netflow output-counters**
4. **interface tunnel** *number*
5. **ip flow** {**ingress** | **egress**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip multicast netflow output-counters** **Example:** `Router(config)# ip multicast netflow output-counters`	(Optional) Enables NetFlow accounting for the number of bytes and packets of multicast traffic forwarded from an ingress flow.
Step 4	**interface tunnel** *number* **Example:** `Router(conf)# interface tunnel 0`	Specifies the tunnel interface and enters interface configuration mode.
Step 5	**ip flow** {**ingress** | **egress**} **Example:** `Router(conf-if)# ip flow egress`	Configures NetFlow accounting on the interface. • **ingress** --Configures NetFlow accounting for traffic that is received by the interface. • **egress** --Configures NetFlow accounting for traffic that is transmitted by the interface.
Step 6	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring NetFlow Accounting on the Physical Interfaces

To configure NetFlow accounting on one or more physical interfaces, perform the following task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast netflow output-counters**
4. **interface** *type number*
5. Do one of the following:
   - **ip flow** {**ingress** | **egress**}
6. **exit**
7. Repeat Steps 4 through 6 to enable NetFlow on other interfaces.
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip multicast netflow output-counters** **Example:** `Router(config)# ip multicast netflow output-counters`	(Optional) Enables NetFlow accounting for the number of bytes and packets of multicast traffic forwarded from an ingress flow.
Step 4	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 3/1`	Specifies the interface on which you want to enable NetFlow and enters interface configuration mode.

Command or Action	Purpose

Step 5 Do one of the following:
• **ip flow** {**ingress** | **egress**}
Example:
`Router(config-if)# ip flow ingress`
Example:
Example:
`Router(config-if)# ip flow egress`
	Enables NetFlow on the interface.
	• **ingress** --Captures traffic that is being received by the interface.
	• **egress** --Captures traffic that is being transmitted by the interface.

Step 6 **exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode and returns to global configuration mode. **Note** You need to use this command only if you want to enable NetFlow on another interface.
Step 7 Repeat Steps 4 through 6 to enable NetFlow on other interfaces.	(Optional) --
Step 8 **end** **Example:** `Router(config-if)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Verifying NetFlow Accounting

To verify that NetFlow accounting for the tunnel interface is working, perform the following task.

Note This task uses the sample network shown in the figure below.

SUMMARY STEPS

1. **enable**
2. **show ip cache flow**
3. **show mls net ip module** *number*

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 **show ip cache flow**

The **show ip cache flow** command displays the NetFlow statistics in the cache. The tunnel interface (Tu0) appears in several rows of the statistics, indicating that NetFlow accounting is operational for the tunnel interface.

Example:

```
c65003# show ip cache flow
------------------------------------------------------------------------------
Displaying software-switched flow entries on the MSFC in Module 5:
IP packet size distribution (3721891 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 1.00 .000 .000 .000 .000 .000 .000
IP Flow Switching Cache, 278544 bytes
  2 active, 4094 inactive, 6 added
  5394 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 33992 bytes
  0 active, 1024 inactive, 0 added, 0 added to flow
  0 alloc failures, 0 force free
  1 chunk, 0 chunks added
  last clearing of statistics 05:58:56
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
ICMP                 4      0.0    406293  1499     75.4     626.5      12.3
Total:               4      0.0    406293  1499     75.4     626.5      12.3
SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Fa3/1         192.168.22.2    Tu0*          192.168.10.2    01 0000 0000  1052K
Fa3/1         192.168.22.2    Tu0           192.168.10.2    01 0000 0000  1052K
------------------------------------------------------------------------------
Displaying hardware-switched flow entries in the PFC (Active) Module 5:
 SrcIf            SrcIPaddress     DstIf            DstIPaddress    Pr SrcP DstP  Pkts
 Tu0              10.4.9.62        Gi6/2            10.5.9.62       2F 0000 0000   155K
 --               0.0.0.0          ---              0.0.0.0         00 0000 0000  1764K
 Fa3/1            192.168.22.2     Tu0              192.168.10.2    01 0000 0000    65K
 Tu0              192.168.10.2     Fa3/1            192.168.22.2    01 0000 0000   695K
 Tu0              192.168.10.2     Fa3/1            192.168.22.2    01 0008 0000    66K
 Tu0              192.168.10.2     Fa3/1            192.168.22.2    11 F378 F566    90K
 Fa3/1            192.168.22.2     Tu0              192.168.10.2    11 F566 F378    90K
```

Step 3 **show mls net ip module** *number*

The **show mls net ip mod** *number* command displays information about the hardware-switched NetFlow flows. The tunnel interface (Tu0) appears in several rows of the statistics, indicating that NetFlow accounting is operational for the tunnel interface.

Example:

```
c65003# show mls net ip module 5
Displaying NetFlow entries in Active Supervisor EARL in module 5
```

```
DstIP           SrcIP           Prot:SrcPort:DstPort  Src i/f          :AdjPtr
--------------------------------------------------------------------------------
Pkts         Bytes         Age   LastSeen  Attributes
------------------------------------------------
224.0.0.2       10.4.9.254      udp :646    :646      Gi6/2            :0x0
46           2852          200   00:30:28   Multicast
0.0.0.0         0.0.0.0         0   :0      :0        --               :0x0
238          17450         203   00:30:28   L3 - Dynamic
224.0.0.13      172.31.0.2      103 :0      :0        Gi6/2            :0x0
7            378           189   00:30:21   Multicast
224.0.0.5       192.168.255.254 89  :0      :0        Fa3/1            :0x0
204          16320         204   00:30:31   Multicast
224.0.0.1       172.31.0.2      2   :0      :0        Gi6/2            :0x0
3            138           174   00:29:38   Multicast
10.4.9.255      10.4.9.2        udp :138    :138      Fa3/1            :0x0
0            0             143   00:28:09   L3 - Dynamic
224.0.0.13      192.168.3.2     103 :0      :0        Tu0              :0x0
6            372           153   00:30:28   Multicast
224.192.16.1    172.31.0.1      icmp:0      :0        Fa3/1            :0x0
20435        940010        205   00:30:32   Multicast
224.0.0.1       192.168.3.2     2   :0      :0        Tu0              :0x0
2            64            103   00:29:49   Multicast
10.4.9.255      10.4.9.2        udp :137    :137      Fa3/1            :0x0
0            0             79    00:30:10   L3 - Dynamic
```

Configuring NetFlow Data Export Using the Version 9 Export Format

To configure NetFlow Data Export using the Version 9 data export format, perform the following task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mls flow ip** {**destination** | **destination-source** | **full** | **interface-destination-source** | **interface-full** | **source**}
4. **mls nde sender**
5. **ip flow-export destination** {*ip-address* | *hostname*} *udp-port*
6. Repeat Step 5 once to configure a second NetFlow export destination.
7. **ip flow-export source** *interface-type interface-number*
8. **ip flow-export version 9 [origin-as | peer-as] [bgp-nexthop]**
9. **ip flow-export template refresh-rate** *packets*
10. **ip flow-export template timeout-rate** *minutes*
11. **ip flow-export template options export-stats**
12. **ip flow-export template options refresh-rate** *packets*
13. **ip flow-export template options timeout-rate** *minutes*
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enters privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**mls flow ip** {**destination** | **destination-source** | **full** | **interface-destination-source** | **interface-full** | **source**} **Example:** `Router(conf)# mls flow ip interface-full`	Specifies the flow mask for NetFlow data export.
Step 4	**mls nde sender** **Example:** `Router(config)# mls nde sender`	Enables multi-layer switching (MLS) NetFlow data export (NDE).
Step 5	**ip flow-export destination** {*ip-address* | *hostname*} *udp-port* **Example:** `Router(config)# ip flow-export destination 172.16.10.2 99`	Specifies the IP address or hostname of the NetFlow collector and the UDP port on which the NetFlow collector is listening.
Step 6	Repeat Step 5 once to configure a second NetFlow export destination.	(Optional) You can configure a maximum of two export destinations for NetFlow.
Step 7	**ip flow-export source** *interface-type interface-number* **Example:** `Router(config)# ip flow-export source gigabitethernet 6/2`	(Optional) Specifies the interface from which the source IP address is derived for the UDP datagrams that are sent by NetFlow data export to the destination host.

	Command or Action	Purpose
Step 8	**ip flow-export version 9 [origin-as | peer-as] [bgp-nexthop]** **Example:** `Router(config)# ip flow-export version 9`	(Optional) Enables the export of information in NetFlow cache entries. • The **version 9** keyword specifies that the export packet uses the Version 9 format. • The **origin-as** keyword specifies that export statistics include the originating autonomous system for the source and destination. • The **peer-as** keyword specifies that export statistics include the peer autonomous system for the source and destination. • The **bgp-nexthop** keyword specifies that export statistics include border gateway protocol (BGP) next hop-related information. **Caution** Entering this command on a Cisco 12000 Series Internet Router causes packet forwarding to stop for a few seconds while NetFlow reloads the route processor and line card CEF tables. To avoid interruption of service to a live network, apply this command during a change window, or include it in the startup-config file to be executed during a router reboot.
Step 9	**ip flow-export template refresh-rate** *packets* **Example:** `Router(config)# ip flow-export template refresh-rate 15` **Example:**	(Optional) Enables the export of information in NetFlow cache entries. • The **template** keyword specifies template-specific configurations. • The **refresh-rate** *packets*keyword-argument pair specifies the number of packets exported before the templates are resent. Range is 1 to 600 packets. The default is 20 packets.
Step 10	**ip flow-export template timeout-rate** *minutes* **Example:** `Router(config)# ip flow-export template timeout-rate 90`	(Optional) Enables the export of information in NetFlow cache entries. • The **template** keyword specifies that the **timeout-rate** keyword applies to the template. • The **timeout-rate** *minutes*keyword-argument pair specifies the time elapsed before the templates are resent. You can specify from 1 to 3600 minutes. The default is 30 minutes.
Step 11	**ip flow-export template options export-stats** **Example:** `Router(config)# ip flow-export template options export-stats`	(Optional) Enables the export of information in NetFlow cache entries. • The **template** keyword specifies template-specific configurations. • The **options** keyword specifies template options. • The **export-stats**keyword specifies that the export statistics include the total number of flows exported and the total number of packets exported.

	Command or Action	Purpose
Step 12	**ip flow-export template options refresh-rate** *packets* **Example:** `Router(config)# ip flow-export template options refresh-rate 25`	(Optional) Enables the export of information in NetFlow cache entries. • The **template** keyword specifies template-specific configurations. • The **options** keyword specifies template options. • The **refresh-rate** *packets*keyword-argument pair specifies the number of packets exported before the templates are resent. Range is 1 to 600 packets. The default is 20 packets.
Step 13	**ip flow-export template options timeout-rate** *minutes* **Example:** `Router(config)# ip flow-export template options timeout-rate 120`	(Optional) Enables the export of information in NetFlow cache entries. • The **template** keyword specifies template-specific configurations. • The **options** keyword specifies template options. • The **timeout-rate** *minutes*keyword-argument pair specifies the time elapsed before the templates are resent. Range is 1 to 3600 minutes. The default is 30 minutes.
Step 14	**end** **Example:** `Router(config)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Verifying That NetFlow Data Export Is Operational

To verify that NetFlow data export is operational, perform the following optional task.

SUMMARY STEPS

1. **show ip flow export**
2. **show ip flow export template**

DETAILED STEPS

Step 1 **show ip flow export**

Use this command to display the statistics for the NetFlow data export, including statistics for the main cache and for all other enabled caches. The following is sample output from this command:

Example:

```
Router# show ip flow export
Flow export v9 is enabled for main cache
  Export source and destination details :
  VRF ID : Default
    Source(1)       10.4.9.62 (GigabitEthernet6/2)
    Source(2)       10.4.9.62 (GigabitEthernet6/2)
    Destination(1)  172.16.10.2 (99)
    Destination(2)  172.16.10.3 (99)
  Version 9 flow records
```

```
  11 flows exported in 11 udp datagrams
  0 flows failed due to lack of export packet
  0 export packets were sent up to process level
  0 export packets were dropped due to no fib
  0 export packets were dropped due to adjacency issues
  0 export packets were dropped due to fragmentation failures
  0 export packets were dropped due to encapsulation fixup failures
  0 export packets were dropped enqueuing for the RP
  0 export packets were dropped due to IPC rate limiting
  0 export packets were dropped due to Card not being able to export
```

Step 2 **show ip flow export template**

Use this command to display the statistics for the NetFlow data export (such as the template timeout rate and the refresh rate) for the template-specific configurations. The following is sample output from this command:

Example:

```
Router# show ip flow export template
Template Options Flag = 1
   Total number of Templates added = 1
   Total active Templates = 1
   Flow Templates active = 0
   Flow Templates added = 0
   Option Templates active = 1
   Option  Templates added = 1
   Template ager polls = 0
   Option Template ager polls = 388
Main cache version 9 export is enabled
 Template export information
   Template timeout = 90
   Template refresh rate = 15
 Option export information
   Option timeout = 120
   Option refresh rate = 25
```

Configuration Examples for NetFlow Accounting for Unicast and Multicast on GRE Tunnel Interfaces

Configuring a GRE IP Tunnel Example

The following example shows how to configure a GRE IP tunnel:

```
interface Tunnel0
 tunnel mode gre ip
 bandwidth 1000
 ip address 192.168.3.1 255.255.255.0
 tunnel source GigabitEthernet6/2
 tunnel destination 10.5.9.62
 tunnel path-mtu-discovery
```

The following display output shows that the GRE IP tunnel is operational because the tunnel is transmitting and receiving traffic:

```
c65002# show interface tunnel 0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 192.168.3.1/24
  MTU 1514 bytes, BW 1000 Kbit, DLY 50000 usec,
     reliability 255/255, txload 90/255, rxload 98/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 10.4.9.62 (GigabitEthernet6/2), destination 10.5.9.62
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255, Fast tunneling enabled
  Path MTU Discovery, ager 10 mins, min MTU 92
  Last input 00:11:44, output 00:11:44, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 380000 bits/sec, 125 packets/sec
  5 minute output rate 347000 bits/sec, 125 packets/sec
  L2 Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes
  L3 in Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes mcast
  L3 out Switched: ucast: 0 pkt, 0 bytes mcast: 0 pkt, 0 bytes
     3344121 packets input, 2452613051 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     3399211 packets output, 2431569783 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Configuring NetFlow Accounting on a GRE IP Tunnel Example

The following example shows how to configure NetFlow Accounting on a GRE IP Tunnel and a FastEthernet interface:

```
mls flow ip interface-full
interface tunnel 0
 ip flow egress
 ip flow ingress
interface FastEthernet3/1
 no shut
 ip address 192.168.22.1 255.255.255.0
 ip flow ingress
 ip flow egress
```

The following display output shows that NetFlow accounting is operational because the flow cache has NetFlow statistics data in it:

```
c65002# show ip cache flow
------------------------------------------------------------------------------
Displaying software-switched flow entries on the MSFC in Module 5:
IP packet size distribution (3721891 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 1.00 .000 .000 .000 .000 .000 .000
IP Flow Switching Cache, 278544 bytes
  2 active, 4094 inactive, 6 added
  5394 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 33992 bytes
  0 active, 1024 inactive, 0 added, 0 added to flow
```

```
  0 alloc failures, 0 force free
  1 chunk, 0 chunks added
  last clearing of statistics 05:58:56
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
ICMP                 4      0.0    406293  1499     75.4     626.5      12.3
Total:               4      0.0    406293  1499     75.4     626.5      12.3
SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Fa3/1         192.168.22.2    Tu0*          192.168.10.2    01 0000 0000  1052K
Fa3/1         192.168.22.2    Tu0           192.168.10.2    01 0000 0000  1052K
---------------------------------------------------------------------------
Displaying hardware-switched flow entries in the PFC (Active) Module 5:
 SrcIf            SrcIPaddress     DstIf            DstIPaddress    Pr SrcP DstP  Pkts
 Tu0              10.4.9.62        Gi6/2            10.5.9.62       2F 0000 0000   155K
 --               0.0.0.0          ---              0.0.0.0         00 0000 0000  1764K
 Fa3/1            192.168.22.2     Tu0              192.168.10.2    01 0000 0000    65K
 Tu0              192.168.10.2     Fa3/1            192.168.22.2    01 0000 0000   695K
 Tu0              192.168.10.2     Fa3/1            192.168.22.2    01 0008 0000    66K
 Tu0              192.168.10.2     Fa3/1            192.168.22.2    11 F378 F566    90K
 Fa3/1            192.168.22.2     Tu0              192.168.10.2    11 F566 F378    90K
```

The following display output shows that NetFlow accounting is operational because there are statistics for the hardware-switched NetFlow flows.

```
c65003# show mls net ip mod 5
Displaying NetFlow entries in Active Supervisor EARL in module 5
DstIP           SrcIP           Prot:SrcPort:DstPort  Src i/f          :AdjPtr
-------------------------------------------------------------------------------
Pkts         Bytes         Age   LastSeen  Attributes
-----------------------------------------------
224.0.0.2       10.4.9.254      udp :646    :646      Gi6/2            :0x0
46           2852          200   00:30:28   Multicast
0.0.0.0         0.0.0.0         0   :0      :0        --               :0x0
238          17450         203   00:30:28   L3 - Dynamic
224.0.0.13      172.31.0.2      103 :0      :0        Gi6/2            :0x0
7            378           189   00:30:21   Multicast
224.0.0.5       192.168.255.254 89  :0      :0        Fa3/1            :0x0
204          16320         204   00:30:31   Multicast
224.0.0.1       172.31.0.2      2   :0      :0        Gi6/2            :0x0
3            138           174   00:29:38   Multicast
10.4.9.255      10.4.9.2        udp :138    :138      Fa3/1            :0x0
0            0             143   00:28:09   L3 - Dynamic
224.0.0.13      192.168.3.2     103 :0      :0        Tu0              :0x0
6            372           153   00:30:28   Multicast
224.192.16.1    172.31.0.1      icmp:0      :0        Fa3/1            :0x0
20435        940010        205   00:30:32   Multicast
224.0.0.1       192.168.3.2     2   :0      :0        Tu0              :0x0
2            64            103   00:29:49   Multicast
10.4.9.255      10.4.9.2        udp :137    :137      Fa3/1            :0x0
0            0             79    00:30:10   L3 - Dynamic
```

Additional References

Related Documents

Related Topic	Document Title
Configuring Cisco IOS 12.2SX on Cisco Catalyst 6500 series switches	Catalyst 6500 Release 12.2SXH and Later Software Configuration Guide

Standards

Standard	Title
There are no standards associated with this feature.	--

MIBs

MIB	MIBs Link
There are no MIBs associated with this feature.	--

RFCs

RFC	Title
RFC 2784	Generic Routing Encapsulation (GRE)

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for Configuring NetFlow Accounting for Unicast and Multicast on GRE IP Tunnel Interfaces

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 3 Feature Information for Flexible NetFlow

Feature Name	Releases	Feature Configuration Information
Configuring NetFlow Accounting for Unicast and Multicast on GRE Tunnel Interfaces	12.2(33)SXI	The Configuring NetFlow Accounting for Unicast and Multicast on GRE Tunnel Interfaces feature allows NetFlow statistics to be gathered on traffic that is transmitted over a GRE IP tunnel interface. The following section provides information for configuring this feature: No commands were introduced or modified for this feature.