Quality of Service Options on GRE Tunnel Interfaces

Document ID: 10106

Contents

Introduction
Prerequisites
Requirements
Components Used
Conventions
Overview of GRE
Cisco QoS for GRE Tunnels
Shaping
Policing
Congestion Avoidance
The qos pre−classify command
Characterizing Traffic for QoS Policies

Where Do I Apply the Service Policy?
Multipoint Tunnel Interfaces
Known Issues
Related Information

Introduction

This document reviews which Quality of Service (QoS) features can be configured on tunnel interfaces using generic routing encapsulation (GRE). Tunnels configured with IP Security (IPsec) are outside the scope of this document.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Overview of GRE

Before learning about QoS over GRE tunnels, you first need to understand the format of a tunneled packet.

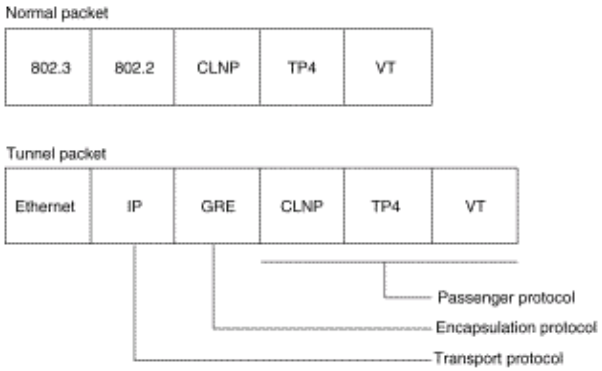
A tunnel interface is a virtual or logical interface on a router running Cisco IOS® Software. It creates a virtual point−to−point link between two Cisco routers at remote points over an IP internetwork.

GRE is an encapsulation protocol supported by IOS and defined in RFC 1702 . Tunneling protocols encapsulate packets inside of a transport protocol.

A tunnel interface supports a header for each of these:

- A passenger protocol or encapsulated protocol, such as IP, AppleTalk, DECnet, or IPX.
- A carrier protocol (GRE in this case).
- A transport protocol (IP only in this case).

The format of a tunnel packet is illustrated here:



Refer to Configuring Logical Interfaces for more information on configuring GRE tunnels.

Cisco QoS for GRE Tunnels

A tunnel interface supports many of the same QoS features as a physical interface. These sections describe the supported QoS features.

Shaping

Cisco IOS Software Release12.0(7)T introduced support for applying generic traffic shaping (GTS) directly on the tunnel interface. The following sample configuration shapes the tunnel interface to an overall output rate of 500 kbps. Refer to Configuring Generic Traffic Shaping for more information.

```
interface Tunnel0
  ip address 130.1.2.1 255.255.255.0
  traffic−shape rate 500000 125000 125000 1000
  tunnel source 10.1.1.1
  tunnel destination 10.2.2.2
```

Cisco IOS Software Release 12.1(2)T added support for class−based shaping using the modular QoS command−line interface (MQC). The following sample configuration shows how to apply the same shaping policy to the tunnel interface with the MQC commands. Refer to Configuring Class−Based Shaping for more information.

```
policy-map tunnel
  class class-default
    shape average 500000 125000 125000
interface Tunnel0
  ip address 130.1.2.1 255.255.255.0
  service-policy output tunnel
  tunnel source 130.1.35.1
  tunnel destination 130.1.35.2
```

Policing

When an interface becomes congested and packets start to queue, you can apply a queuing method to packets waiting to be transmitted. Cisco IOS logical interfaces do not inherently support a state of congestion and do not support the direct application of a service policy that applies a queuing method. Instead, you need to apply a hierarchical policy as follows:

1. Create a "child" or lower−level policy that configures a queueing mechanism, such as low latency queueing with the **priority** command and class−based weighted fair queueing (CBWFQ) with the **bandwidth** command. Refer to Congestion Management for more information.

   ```
   policy-map child
    class voice
     priority 512
   ```
2. Create a "parent" or top−level policy that applies class−based shaping. Apply the child policy as a command under the parent policy since admission control for the child class is done based on the shaping rate for the parent class.

   ```
   policy-map tunnel
    class class-default
     shape average 2000000
     service-policy child
   ```
3. Apply the parent policy to the tunnel interface.

   ```
   interface tunnel0
      service-policy tunnel
   ```

The router prints this log message when a tunnel interface is configured with a service policy that applies queuing without shaping.

```
router(config)# interface tunnel1
router(config-if)# service-policy output child
 Class Based Weighted Fair Queueing not supported on this interface
```

Tunnel interfaces also support class−based policing, but they do not support committed access rate (CAR).

Note: Service Policies are not supported on tunnel interfaces on 7500.

Congestion Avoidance

Cisco IOS Software Release 11.3T introduced GRE Tunnel Marking and DSCP or IP Precedence Values, which configures the router to copy the IP precedence bit values of the ToS byte to the tunnel or GRE IP header that encapsulates the inner packet. Previously, those bits were set to zero. Intermediate routers between the tunnel endpoints can use the IP precedence values to classify the packets for QoS features such as policy routing, WFQ, and weighted random early detection (WRED).

The qos pre−classify command

When packets are encapsulated by tunnel or encryption headers, QoS features are unable to examine the original packet headers and correctly classify the packets. Packets traveling across the same tunnel have the same tunnel headers, so the packets are treated identically if the physical interface is congested. With the introduction of the Quality of Service for Virtual Private Networks (VPNs) feature, packets can now be classified before tunneling and encryption occur.

In this example, tunnel0 is the tunnel name. The **qos pre−classify** command enables the QoS for VPNs feature on tunnel0:

```
Router(config)# interface tunnel0
Router(config−if)# qos pre−classify
```

Note: The **qos pre−classify** command can be used in order to classify traffic based on values other than IP precedence or DSCP. For example, you might want to classify packets based on IP flow or Layer 3 information, such as source and destination IP address for which this command can be used. The **qos pre−classify** command is required only if you classify traffic on IP, protocol, or port. If classification is based on DSCP code, then **qos pre−classify** is not required.

Characterizing Traffic for QoS Policies

When configuring a service policy, you first might need to characterize the traffic that is traversing the tunnel. Cisco IOS supports Netflow and IP Cisco Express Forwarding (CEF) accounting on logical interfaces like tunnels. Refer to the NetFlow Services Solutions Guide for more information.

Where Do I Apply the Service Policy?

You can apply a service policy to either the tunnel interface or to the underlying physical interface. The decision of where to apply the policy depends on the QoS objectives. It also depends on which header you need to use for classification.

- Apply the policy to the tunnel interface without **qos−preclassify** when you want to classify packets based on the pre−tunnel header.
- Apply the policy to the *physical* interface without **qos−preclassify** when you want to classify packets based on the post−tunnel header. In addition, apply the policy to the physical interface when you want to shape or police all traffic belonging to a tunnel, and the physical interface supports several tunnels.
- Apply the policy to a *physical* interface and enable **qos−preclassify** on a tunnel interface when you want to classify packets based on the pre−tunnel header.

Multipoint Tunnel Interfaces

CBWFQ inside class−based shaping is not supported on a multipoint interface. Cisco bug ID CSCds87191 configures the router to print an error message when rejecting the policy.

Known Issues

In rare conditions, applying a service−policy configured with the **shape** command leads to high CPU utilization and alignment errors. The CPU load is caused by logging the alignment errors, which in turn are caused by CEF incorrectly setting the output interface and adjacency rewrite information. This problem affects only non−RSP platforms (low−end) and platforms using particle−based CEF switching, and is resolved via Cisco bug IDs CSCdu45504 and CSCuk30302. You also can consider these workarounds:

- Replace GRE encapsulation with **tunnel mode ipip**.
- Replace the **shape** command with the **police** command.
- Configure shaping on the physical interface supporting the tunnel.

Related Information

- **Quality of Service for Virtual Private Networks**
- **Configuring GRE Tunnel Over Cable**
- **QoS Technology Support**
- **Configuring a GRE Tunnel over IPSec with OSPF**
- **Technical Support & Documentation − Cisco Systems**

Contacts & Feedback | Help | Site Map
© 2014 − 2015 Cisco Systems, Inc. All rights reserved. Terms & Conditions | Privacy Statement | Cookie Policy | Trademarks of Cisco Systems, Inc.

Updated: Mar 20, 2015 Document ID: 10106
