

Per-Tunnel QoS for DMVPN

The Per-Tunnel QoS for DMVPN feature introduces per-tunnel QoS support for DMVPN and increases per-tunnel QoS performance for IPsec tunnel interfaces.

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

- Prerequisites for Per-Tunnel QoS for DMVPN, on page 1
- Restrictions for Per-Tunnel QoS for DMVPN, on page 1
- Information About Per-Tunnel QoS for DMVPN, on page 3
- How to Configure Per-Tunnel QoS for DMVPN, on page 5
- Configuration Examples for Per-Tunnel QoS for DMVPN, on page 9
- Additional References for Per-Tunnel QoS for DMVPN, on page 17
- Feature Information for Per-Tunnel QoS for DMVPN, on page 18

Prerequisites for Per-Tunnel QoS for DMVPN

- Before you configure the Per-Tunnel QoS for DMVPN feature, you must configure Cisco Express Forwarding switching.
- Before you can configure an Next Hop Resolution Protocol (NHRP) group on a spoke and map the NHRP group to a QoS policy on a hub, the spoke and the hub must already be configured for DMVPN without the per-tunnel QoS.

Restrictions for Per-Tunnel QoS for DMVPN

- The Per-Tunnel QoS for DMVPN feature only supports the following encapsulation and transport protocol combinations:
  - Per-Tunnel QoS for IPv4 over DMVPN with IPv4 transport (Effective from Cisco IOS XE Release 3.6S).

  - Per-Tunnel QoS for IPv6 over DMVPN with IPv4 transport (Effective from Cisco IOS XE Release 3.8S).
  - Per-Tunnel QoS for IPv4 over DMVPN with IPv6 transport (Effective from Cisco IOS XE Release 3.11S).
  - Per-Tunnel QoS for IPv6 over DMVPN with IPv6 transport (Effective from Cisco IOS XE Release 3.11S).
  - Per-Tunnel QoS for MPLS VPN over DMVPN with IPv4 transport (2547oDMVPN) (Effective from Cisco IOS XE Release 3.15S).
  - Per-Tunnel QoS for MPLS VPN over DMVPN with IPv6 transport (2547oDMVPN) (Effective from Cisco IOS XE Release 3.15S).

- For a given DMPVN tunnel interface, one transport protocol, either IPv4 or IPv6, can only be used. However, different DMVPN tunnel interfaces on the same device may use IPv4 or IPv6 transport protocol at the same time. Per-tunnel QoS can be configured for IPv4 and IPv6 DMVPN passenger traffic packets and be associated with an outbound physical interface that is either IPv4, IPv6 or both. This DMPVN tunnel traffic may be mixed with non-DMVPN IPv4 and IPv6 traffic, or both, on the outbound physical interface with its own QoS policy with restrictions.
- The Per-Tunnel QoS for DMVPN feature does not support the following:
  - Per-Tunnel QoS for IPv4 or IPv6 or Multiprotocol Label Switching (MPLS) VPN over DMVPN with Layer 2 Tunnel Protocol (L2TP) transport.
  - Per-Tunnel QoS for IPv4 or IPv6 or MPLS VPN over DMVPN.
- Per-Tunnel QoS service policies are only supported in the egress direction.
- This feature does not support adding the capability of user configurable queuing and schedules before the crypto engine.
- Fair queueing should not be used in a per-tunnel QoS for DMVPN policy map because the outer header with nonchanging IP addresses is used for individual flow queue selection. This results in the same queue being selected for all traffic flowing through the class with fair queueing.
- A QoS service policy is supported on the main interface or subinterface that the tunnel is sourced from in conjunction with a per-tunnel QoS service policy on the DMVPN tunnel interface. However, there are certain restrictions for the main or subinterface service policy, which are as follows:
  - A service policy is supported on either the main interface or the subinterface, but not both, in conjunction with the per-tunnel QoS service policy.
  - The main interface or subinterface QoS service policy is limited to only a class-default shaper (it can only contain the **class class-default** and **shape** commands). Additional QoS configurations are not supported on the main interface or subinterface when two different QoS service policies are applied to the main or subinterface and the tunnel interface simultaneously.
  - The main interface or subinterface QoS service policy must be applied before the tunnel interface service policy.
  - The main interface or subinterface QoS service policy is checked for validity only when a QoS service policy is applied on the tunnel interface. The main interface or subinterface service policy is not checked during a tunnel movement or modification.

- Adding new classes or features to the main interface or subinterface policy map is not supported. The classes or features may not be blocked on CLI and could result in unpredictable behavior.
- The policy-map counters for the main interface or subinterface service policy (from the **show policy-map interface** command) may not account for all packets and therefore should not be used or referenced. However, this does not affect the QoS functionality. The shaper will still limit the traffic on the main interface or subinterface, including all DMVPN tunnel traffic over that interface.

Information About Per-Tunnel QoS for DMVPN

Per-Tunnel QoS for DMVPN Overview

The Per-Tunnel QoS for DMVPN feature lets you apply a quality of service (QoS) policy on a Dynamic Multipoint VPN (DMVPN) hub on a per-tunnel instance (per-spoke basis) in the egress direction for DMVPN hub-to-spoke tunnels. The QoS policy on a DMVPN hub on a per-tunnel instance lets you shape tunnel traffic to individual spokes (a parent policy) and differentiate individual data flows going through the tunnel for policing (a child policy). The QoS policy that the hub uses for a specific spoke is selected according to the specific Next Hop Resolution Protocol (NHRP) group into which that spoke is configured. Although you can configure many spokes into the same NHRP group, the tunnel traffic for each spoke is measured individually for shaping and policing.

You can use this feature with DMVPN with or without Internet Protocol Security (IPsec).

When the Per-Tunnel QoS for DMVPN feature is enabled, queuing and shaping are performed at the outbound physical interface for generic routing encapsulation (GRE)/IPsec tunnel packets. The Per-Tunnel QoS for DMVPN feature ensures that the GRE header, the IPsec header, and the Layer 2 (for the physical interface) header are included in the packet-size calculations for shaping and bandwidth queuing of packets under QoS.

Benefits of Per-Tunnel QoS for DMVPN

Before the introduction of Per-Tunnel QoS for DMVPN feature, quality of service (QoS) on a Dynamic Multipoint VPN (DMVPN) hub could be configured to measure only either the outbound traffic in the aggregate (overall spokes) or outbound traffic on a per-spoke basis (with extensive manual configuration).

The Per-Tunnel QoS for DMVPN feature provides the following benefits:

- The QoS policy is attached to the DMVPN hub, and the criteria for matching the tunnel traffic are set up automatically as each spoke registers with the hub (which means that extensive manual configuration is not needed).
- Traffic can be regulated from the hub to spokes on a per-spoke basis.
- The hub cannot send excessive traffic to (and overrun) a small spoke.
- The amount of outbound hub bandwidth that a “greedy” spoke can consume can be limited; therefore, the traffic cannot monopolize a hub’s resources and starve other spokes.

NHRP QoS Provisioning for DMVPN

Next Hop Resolution Protocol (NHRP) performs the provisioning for the Per-Tunnel QoS for DMVPN feature by using NHRP groups.

An NHRP group, a new functionality introduced by this feature, is the group identity information signaled by a Dynamic Multipoint VPN (DMVPN) node (a spoke) to the DMVPN hub. The hub uses this information to select a locally defined quality of service (QoS) policy instance for the remote node.

You can configure an NHRP group on the spoke router on the DMVPN generic routing encapsulation (GRE) tunnel interface. The NHRP group name is communicated to the hub in each of the periodic NHRP registration requests sent from the spoke to the hub.

NHRP group-to-QoS policy mappings are configured on the hub DMVPN GRE tunnel interface. The NHRP group string received from a spoke is mapped to a QoS policy, which is applied to that hub-to-spoke tunnel in the egress direction.

After an NHRP group is configured on a spoke, the group is not immediately sent to the hub, but is sent in the next periodic registration request. The spoke can belong to only one NHRP group per GRE tunnel interface. If a spoke is configured as part of two or more DMVPN networks (multiple GRE tunnel interfaces), then the spoke can have a different NHRP group name on each of the GRE tunnel interfaces.

If an NHRP group is not received from the spoke, then a QoS policy is not applied to the spoke, and any existing QoS policy applied to that spoke is removed. If an NHRP group is received from the spoke when previous NHRP registrations did not have an NHRP group, then the corresponding QoS policy is applied. If the same NHRP group is received from a spoke similar to the earlier NHRP registration request, then no action is taken because a QoS policy would have already been applied for that spoke. If a different NHRP group is received from the spoke than what was received in the previous NHRP registration request, any applied QoS policy is removed, and the QoS policy corresponding to the new NHRP group is applied.

Per-Tunnel QoS for Spoke to Spoke Connections

The QoS: Spoke to Spoke per tunnel QoS for DMVPN feature enables a DMVPN client to establish a direct crypto tunnel with another DMVPN client leveraging the per-tunnel QoS policy, using Next Hop Resolution Protocol (NHRP) to build spoke-to-spoke connections.

This feature enhances the Adaptive QoS over DMVPN feature, which ensures effective bandwidth management using dynamic shapers based on available bandwidth.

A spoke-to-spoke connection is established when a group identity information, configured on the spokes using the **nhrp attribute group** command, is exchanged between the spokes through the NHRP Vendor Private Extension (VPE). The NHRP Vendor Private Extensions, encapsulated in NHRP control packets—NHRP resolution request and reply packets.

Assume a network with two spokes—Spoke A and Spoke B, connected to hub. If Spoke A is configured with the **nhrp attribute group** command and traffic exists between the Spoke A and Spoke B, a resolution request from the Spoke A carries the group identity information as part of Vendor Private Extension (VPE). On receiving the resolution request, Spoke B extracts the VPE header and checks the extension types received as part of the resolution request packet. If the VPE extension has group type, the NHRP VPE parser extracts the group information and checka if a matching map is present. If a matching map is present, QoS applies the policy on the target interface.

How to Configure Per-Tunnel QoS for DMVPN

To configure the Per-Tunnel QoS for DMVPN feature, you define a Next Hop Resolution Protocol (NHRP) group on the spokes and then map the NHRP group to a quality of service (QoS) policy on the hub.

Configuring an NHRP Group on a Spoke

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **nhrp group** *group-name*
5. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Device(config)# interface tunnel 1`	Configures a tunnel interface and enters interface configuration mode.
Step 4	**nhrp group** *group-name* **Example:** `Device(config-if)# nhrp group spoke_group1`	Configures a Next Hop Resolution Protocol (NHRP) group on the spoke.
Step 5	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring an NHRP Group Attribute on a Spoke

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **nhrp attribute group** *group-name*
5. **nhrp map group** *group-name* **service-policy output** *qos-policy-map-name*
6. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Device(config)# interface tunnel 1`	Configures a tunnel interface and enters interface configuration mode.
Step 4	**nhrp attribute group** *group-name* **Example:** `Device(config-if)# nhrp attribute group spoke1`	Configures the QoS group identity information on the spoke.
Step 5	**nhrp map group** *group-name* **service-policy output** *qos-policy-map-name* **Example:** `Device(config-if)# nhrp map group spoke_group1 service-policy output group1_parent`	Adds the Next Hop Resolution Protocol (NHRP) group to the quality of service (QoS) policy mapping.
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Mapping an NHRP Group to a QoS Policy on the Hub

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface tunnel** *number*
4. **nhrp map group** *group-name* **service-policy output** *qos-policy-map-name*
5. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface tunnel** *number* **Example:** `Device(config)# interface tunnel 1`	Configures a tunnel interface and enters interface configuration mode.
Step 4	**nhrp map group** *group-name* **service-policy output** *qos-policy-map-name* **Example:** `Device(config-if)# nhrp map group spoke_group1 service-policy output group1_parent`	Adds the Next Hop Resolution Protocol (NHRP) group to the quality of service (QoS) policy mapping on the hub.
Step 5	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Enabling DMVPN Per-tunnel QoS Sourced from Port Channel

To enable the feature, you must configure the command *platform qos port-channel-aggregate <port-channel number>* before configuring port channel.

The *platform qos port-channel-aggregate <port-channel number>* is required for this feature. The order of the configuration steps are important to enable DMVPN Per-tunnel QoS Sourced from Port-Channel feature. The *platform qos port-channel-aggregate <port-channel number>* command must be configured first. Then, the port-channel interface must be created. Lastly, *channel-group x* command must be applied to member ports.

Both port-channel main-interface and sub-interface are supported in aggregate mode.

Note Before configuring the command, you must remove the 'port channel interface' and ‘channel-group’ configuration from physical interface.

1. Enable the command *platform qos port-channel-aggregate <port-channel number>* before configuring port channel.
2. Configure per-tunnel QoS.
3. Reset the NHRP registration process to ensure the spokes register now that the new configuration is present on the hub BR. Use the command *show dmvpn detail* to display the NHRP group for each spoke.

Verifying Per-Tunnel QoS for DMVPN

SUMMARY STEPS

1. **enable**
2. **show dmvpn detail**
3. **show nhrp**
4. **show nhrp group** [*group-name*]
5. **show nhrp group-map** [*group-name*]
6. **show policy-map multipoint** [**tunnel** *tunnel-interface-number*]
7. **show tunnel endpoints**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show dmvpn detail** **Example:** `Device# show dmvpn detail`	Displays detailed Dynamic Multipoint VPN (DMVPN) information for each session, including the Next Hop Server (NHS) and NHS status, crypto session information, and socket details.

	Command or Action	Purpose
		• The output includes the Next Hop Resolution Protocol (NHRP) group received from the spoke and the quality of service (QoS) policy applied to the spoke tunnel.
Step 3	**show nhrp** **Example:** `Device# show nhrp`	Displays the NHRP cache and the NHRP group received from the spoke.
Step 4	**show nhrp group** [*group-name*] **Example:** `Device# show nhrp group`	Displays NHRP group mapping. • The output includes the associated QoS policy name and the list of tunnel endpoints using the QoS policy.
Step 5	**show nhrp group-map** [*group-name*] **Example:** `Device# show nhrp group-map group1-parent`	Displays the group-to-policy maps configured on the hub and also displays the tunnels on which the QoS policy is applied.
Step 6	**show policy-map multipoint** [**tunnel** *tunnel-interface-number*] **Example:** `Device# show policy-map multipoint tunnel 1`	Displays QoS policy details applied to multipoint tunnels.
Step 7	**show tunnel endpoints** **Example:** `Device# show tunnel endpoints`	Displays information about the source and destination endpoints for multipoint tunnels and the QoS policy applied on the spoke tunnel.

Configuration Examples for Per-Tunnel QoS for DMVPN

Example: Configuring an NHRP Group on a Spoke

The following example shows how to configure two Next Hop Resolution Protocol (NHRP) groups on three spokes:

Configuring the First Spoke

```
interface tunnel 1
 ip address 209.165.200.225 255.255.255.224
 no ip redirects
 ip mtu 1400
 ip nhrp authentication testing
 nhrp group spoke_group1
 ip nhrp map 209.165.200.226 203.0.113.1
 ip nhrp map multicast 203.0.113.1
```

```
 ip nhrp network-id 172176366
 ip nhrp holdtime 300
 ip tcp adjust-mss 1360
 ip nhrp nhs 209.165.200.226
 tunnel source fastethernet 2/1/1
 tunnel mode gre multipoint
 tunnel protection ipsec profile DMVPN
interface fastethernet 2/1/1
 ip address 203.0.113.2 255.255.255.0
```

Configuring the Second Spoke

```
interface tunnel 1
 ip address 209.165.200.227 255.255.255.224
 no ip redirects
 ip mtu 1400
 ip nhrp authentication testing
 nhrp group spoke_group1
 ip nhrp map 209.165.200.226 203.0.113.1
 ip nhrp map multicast 203.0.113.1
 ip nhrp network-id 172176366
 ip nhrp holdtime 300
 ip tcp adjust-mss 1360
 ip nhrp nhs 209.165.200.226
 tunnel source fastethernet 2/1/1
 tunnel mode gre multipoint
 tunnel protection ipsec profile DMVPN
interface fastethernet 2/1/1
 ip address 203.0.113.3 255.255.255.0
```

Configuring the Third Spoke

```
interface tunnel 1
 ip address 209.165.200.228 255.255.255.224
 no ip redirects
 ip mtu 1400
 ip nhrp authentication testing
 nhrp group spoke_group2
 ip nhrp map 209.165.200.226 203.0.113.1
 ip nhrp map multicast 203.0.113.1
 ip nhrp network-id 172176366
 ip nhrp holdtime 300
 ip tcp adjust-mss 1360
 ip nhrp nhs 209.165.200.226
 tunnel source fastethernet 2/1/1
 tunnel mode gre multipoint
 tunnel protection ipsec profile DMVPN
interface fastethernet 2/1/1
 ip address 203.0.113.4 255.255.255.0
```

Example: Configuring an NHRP Group Attribute on a Spoke

The following example shows how to configure two Next Hop Resolution Protocol (NHRP) groups attributes on two spokes:

Configuring the First Spoke

```
class-map match-any class2
match ip precedence 5
end
!
policy-map p2
class class2
priority percent 60
end
!
interface Tunnel0
 ip address 10.0.0.2 255.255.255.0
 no ip redirects
 ip mtu 1436
 ip nhrp authentication h1there
 ip nhrp attribute group1
 ip nhrp map group group1 service-policy output p2
 ip nhrp map multicast 172.17.0.1
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 253
 ip nhrp nhs 10.0.0.1
 ip nhrp registration timeout 600
 ip nhrp cache non-authoritative
 no ip mroute-cache
 tunnel source 172.17.0.2
 tunnel mode gre multipoint
 tunnel key 253
 tunnel protection ipsec profile dmvpn-profile
end
```

Configuring the Second Spoke

```
class-map match-any class1
 match ip precedence 5

policy-map policy p1
 class class1
  priority 70

interface Tunnel0
 ip address 10.0.0.1 255.255.255.0
 no ip redirects
 ip mtu 1436
 ip nhrp authentication h1there
 ip nhrp attribute group1
 ip nhrp map group group1 service-policy output p1
 ip nhrp map multicast 172.17.0.2
 ip nhrp map 10.0.0.2 172.17.0.2
 ip nhrp network-id 253
 ip nhrp nhs 10.0.0.2
 ip nhrp registration timeout 600
 ip nhrp cache non-authoritative
 no ip mroute-cache
 tunnel source 172.17.0.1
 tunnel mode gre multipoint
 tunnel key 253
 tunnel protection ipsec profile dmvpn-profile
end
```

Example: Mapping an NHRP Group to a QoS Policy on the Hub

The following example shows how to map Next Hop Resolution Protocol (NHRP) groups to a quality of service (QoS) policy on the hub. The example shows a hierarchical QoS policy (parent: group1_parent/group2_parent; child: group1/group2) that will be used for configuring Per-tunnel QoS for Dynamic Multipoint VPN (DMVPN) feature. The example also shows how to map the NHRP group spoke_group1 to the QoS policy group1_parent and map the NHRP group spoke_group2 to the QoS policy group2_parent on the hub:

```
class-map match-all group1_Routing
 match ip precedence 6
class-map match-all group2_Routing
 match ip precedence 6
class-map match-all group2_voice
 match access-group 100
class-map match-all group1_voice
 match access-group 100
policy-map group1
 class group1_voice
    priority 1000
 class group1_Routing
    bandwidth percent 20
policy-map group1_parent
 class class-default
    shape average 3000000
  service-policy group1
policy-map group2
 class group2_voice
    priority percent 20
 class group2_Routing
    bandwidth percent 10
policy-map group2_parent
 class class-default
    shape average 2000000
  service-policy group2
interface tunnel 1
 ip address 209.165.200.225 255.255.255.224
 no ip redirects
 ip mtu 1400
 ip nhrp authentication testing
 ip nhrp map multicast dynamic
 ip nhrp map group spoke_group1 service-policy output group1_parent
 ip nhrp map group spoke_group2 service-policy output group2_parent
 ip nhrp network-id 172176366
 ip nhrp holdtime 300
 ip nhrp registration unique
 tunnel source fastethernet 2/1/1
 tunnel mode gre multipoint
 tunnel protection ipsec profile DMVPN
interface fastethernet 2/1/1
 ip address 209.165.200.226 255.255.255.224
```

Example: Enabling DMVPN Per-tunnel QoS Sourced from Port Channel

The following example shows how to enable DMVPN Per-tunnel QoS Sourced from Port Channel.

Example: Configuring on hub

```
platform qos port-channel-aggregate 1
!
class-map match-any class2
match ip precedence 5
!
policy-map p1
class class2
  priority percent 60
!
interface Port-channel1
ip address 203.0.113.1 255.255.255.0
!
interface GigabitEthernet0/0/0
channel-group 1
!
interface GigabitEthernet0/0/1
channel-group 1
!
interface Tunnel1
ip address 10.9.9.1 255.255.255.0
no ip redirects
ip nhrp authentication cisco
nhrp map group group1 service-policy output p1
ip nhrp map multicast dynamic
ip nhrp network-id 1
tunnel source Port-channel 1
tunnel mode gre multipoint
```

Example: Configuring on spoke

```
platform qos port-channel-aggregate 1
!
interface Port-channel1
ip address 203.0.113.100 255.255.255.0
!
interface GigabitEthernet0/0/0
channel-group 1
!
interface GigabitEthernet0/0/1
channel-group 1
!
interface Tunnel1
ip address 10.9.9.11 255.255.255.0
no ip redirects
ip nhrp authentication cisco
ip nhrp map 10.9.9.1 203.0.113.1
ip nhrp map multicast 203.0.113.1
ip nhrp network-id 1
ip nhrp nhs 10.9.9.1
tunnel source Port-channel 1
nhrp group group1
tunnel mode gre multipoint
```

Example: Verifying Per-Tunnel QoS for DMVPN

The following example shows how to display the information about Next Hop Resolution Protocol (NHRP) groups received from the spokes and display the quality of service (QoS) policy that is applied to each spoke tunnel. You can enter this command on the hub.

```
Device# show dmvpn detail
```

```
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================
Interface Tunnel1 is up/up, Addr. is 209.165.200.225, VRF ""
   Tunnel Src./Dest. addr: 209.165.200.226/MGRE, Tunnel VRF ""
   Protocol/Transport: "multi-GRE/IP", Protect "DMVPN"
Type:Hub, Total NBMA Peers (v4/v6): 3
# Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb    Target Network
----- --------------- --------------- ----- -------- ----- -----------------
    1   209.165.200.227     192.0.2.2     UP 00:19:20    D        192.0.2.2/32
NHRP group: spoke_group1
 Output QoS service-policy applied: group1_parent
    1   209.165.200.228     192.0.2.3     UP 00:19:20    D        192.0.2.3/32
NHRP group: spoke_group1
 Output QoS service-policy applied: group1_parent
    1    209.165.200.229     192.0.2.4     UP 00:19:23    D        192.0.2.4/32
NHRP group: spoke_group2
 Output QoS service-policy applied: group2_parent
Crypto Session Details:
--------------------------------------------------------------------------------
Interface: tunnel1
Session: [0x04AC1D00]
  IKE SA: local 209.165.200.226/500 remote 209.165.200.227/500 Active
  Crypto Session Status: UP-ACTIVE
  fvrf: (none), Phase1_id: 209.165.200.227
  IPSEC FLOW: permit 47 host 209.165.200.226 host 209.165.200.227
        Active SAs: 2, origin: crypto map
   Outbound SPI : 0x9B264329, transform : ah-sha-hmac
    Socket State: Open
Interface: tunnel1
Session: [0x04AC1C08]
  IKE SA: local 209.165.200.226/500 remote 209.165.200.228/500 Active
  Crypto Session Status: UP-ACTIVE
  fvrf: (none), Phase1_id: 209.165.200.228
  IPSEC FLOW: permit 47 host 209.165.200.226 host 209.165.200.228
        Active SAs: 2, origin: crypto map
   Outbound SPI : 0x36FD56E2, transform : ah-sha-hmac
    Socket State: Open
Interface: tunnel1
Session: [0x04AC1B10]
  IKE SA: local 209.165.200.226/500 remote 209.165.200.229/500 Active
  Crypto Session Status: UP-ACTIVE
  fvrf: (none), Phase1_id: 209.165.200.229
  IPSEC FLOW: permit 47 host 209.165.200.226 host 209.165.200.229
        Active SAs: 2, origin: crypto map
   Outbound SPI : 0xAC96818F, transform : ah-sha-hmac
    Socket State: Open
Pending DMVPN Sessions:
```

The following example shows how to display information about the NHRP groups that are received from the spokes. You can enter this command on the hub.

```
Device# show ip nhrp

192.0.2.240/32 via 192.0.2.240
   Tunnel1 created 00:22:49, expire 00:01:40
   Type: dynamic, Flags: registered
   NBMA address: 209.165.200.227
     Group: spoke_group1
192.0.2.241/32 via 192.0.2.241
```

```
   Tunnel1 created 00:22:48, expire 00:01:41
   Type: dynamic, Flags: registered
   NBMA address: 209.165.200.228
      Group: spoke_group1
192.0.2.242/32 via 192.0.2.242
   Tunnel1 created 00:22:52, expire 00:03:27
   Type: dynamic, Flags: registered
   NBMA address: 209.165.200.229
      Group: spoke_group2
```

The following example shows how to display the details of NHRP group mappings on a hub and the list of tunnels using each of the NHRP groups defined in the mappings. You can enter this command on the hub.

```
Device# show nhrp group-map

Interface: tunnel1
          NHRP group: spoke_group1
  QoS policy: group1_parent
  Tunnels using the QoS policy:
  Tunnel destination overlay/transport address
  198.51.100.220/203.0.113.240
  198.51.100.221/203.0.113.241
          NHRP group: spoke_group2
  QoS policy: group2_parent
  Tunnels using the QoS policy:
  Tunnel destination overlay/transport address
  198.51.100.222/203.0.113.242
```

The following example shows how to display statistics about a specific QoS policy as it is applied to a tunnel endpoint. You can enter this command on the hub.

```
Device# show policy-map multipoint

Interface tunnel1 <--> 203.0.113.252
          Service-policy output: group1_parent
    Class-map: class-default (match-any)
      29 packets, 4988 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
      Queueing
      queue limit 750 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      shape (average) cir 3000000, bc 12000, be 12000
      target shape rate 3000000
      Service-policy : group1
        queue stats for all priority classes:
          queue limit 250 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
        Class-map: group1_voice (match-all)
          0 packets, 0 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match: access-group 100
          Priority: 1000 kbps, burst bytes 25000, b/w exceed drops: 0
        Class-map: group1_Routing (match-all)
          0 packets, 0 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match: ip precedence 6
          Queueing
          queue limit 150 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
```

```
          bandwidth 20% (600 kbps)
        Class-map: class-default (match-any)
          29 packets, 4988 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match: any
          queue limit 350 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
Interface tunnel1 <--> 203.0.113.253
          Service-policy output: group1_parent
    Class-map: class-default (match-any)
      29 packets, 4988 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
      Queueing
      queue limit 750 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      shape (average) cir 3000000, bc 12000, be 12000
      target shape rate 3000000
      Service-policy : group1
        queue stats for all priority classes:
          queue limit 250 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
        Class-map: group1_voice (match-all)
          0 packets, 0 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match: access-group 100
          Priority: 1000 kbps, burst bytes 25000, b/w exceed drops: 0
        Class-map: group1_Routing (match-all)
          0 packets, 0 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match: ip precedence 6
          Queueing
          queue limit 150 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
          bandwidth 20% (600 kbps)
        Class-map: class-default (match-any)
          29 packets, 4988 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match: any
          queue limit 350 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
Interface tunnel1 <--> 203.0.113.254
           Service-policy output: group2_parent
    Class-map: class-default (match-any)
      14 packets, 2408 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
      Queueing
      queue limit 500 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      shape (average) cir 2000000, bc 8000, be 8000
      target shape rate 2000000
      Service-policy : group2
        queue stats for all priority classes:
          queue limit 100 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
        Class-map: group2_voice (match-all)
```

```
  0 packets, 0 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: access-group 100
  Priority: 20% (400 kbps), burst bytes 10000, b/w exceed drops: 0
Class-map: group2_Routing (match-all)
  0 packets, 0 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: ip precedence 6
  Queueing
  queue limit 50 packets
  (queue depth/total drops/no-buffer drops) 0/0/0
  (pkts output/bytes output) 0/0
  bandwidth 10% (200 kbps)
Class-map: class-default (match-any)
  14 packets, 2408 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: any
  queue limit 350 packets
  (queue depth/total drops/no-buffer drops) 0/0/0
  (pkts output/bytes output) 0/0
```

Additional References for Per-Tunnel QoS for DMVPN

Related Documents

Related Topic	Document Title
Security commands	• *Cisco IOS Security Command Reference Commands A to C* • *Cisco IOS Security Command Reference Commands D to L* • *Cisco IOS Security Command Reference Commands M to R* • *Cisco IOS Security Command Reference Commands S to Z*
IP NHRP commands	Cisco IOS IP Addressing Services Command Reference
Configuring Basic Cisco Express Forwarding	*IP Switching Cisco Express Forwarding Configuration Guide*
Configuring NHRP	*IP Addressing: NHRP Configuration Guide*
Recommended cryptographic algorithms	Next Generation Encryption

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Per-Tunnel QoS for DMVPN

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 1: Feature Information for Per-Tunnel QoS for DMVPN

Feature Name	Releases	Feature Information
Per-Tunnel QoS	15.4(1)T / 3.11S	In , this feature was enhanced to provide support for IPv6 addresses. The following commands were introduced or modified: **ip nhrp map**, **nhrp group**, **nhrp map group**, **show dmvpn**, **show ip nhrp**, **show ip nhrp group-map**, **show nhrp group-map**, **show policy-map multipoint tunnel**. The commands **ip nhrp group** and **ip nhrp map group** were depreciated and hidden in the CLI. They are replaced with protocol agnostic **nhrp group** and **nhrp map group**. The configuration needs to be manually migrated to the new syntax.
	16.6.5, 16.8.1	The commands **ip nhrp group** and **ip nhrp map group** are removed from CLI. Manual migration before or after upgrade is required.
QoS: Spoke to Spoke Per-tunnel QoS for DMVPN		The following commands were introduced or modified: **nhrp attribute group**, **show dmvpn**, **show ip nhrp**. **Note** The command **show ip nhrp group** is deprecated and is not in use.
QoS: DMVPN Per-tunnel QoS over Aggregate GEC	Cisco IOS XE Everest 16.4.1	The QoS: DMVPN Per-tunnel QoS over Aggregate GEC feature is supported.