Configuring Class-Based RTP and TCP Header Compression

Last Updated: December 5, 2011

Header compression is a mechanism that compresses the IP header in a packet before the packet is transmitted. Header compression reduces network overhead and speeds up the transmission of either Real-Time Transport Protocol (RTP) packets or Transmission Control Protocol (TCP) packets.

Cisco provides two types of header compression: RTP header compression and TCP header compression. RTP and TCP header compression are typically configured on a per-interface (or subinterface) basis. Class-based RTP and TCP header compression allows you to configure either type of header compression on a per-class basis. This module describes the concepts and tasks related to configuring class-based RTP and TCP header compression.

Note

If you want to configure RTP or TCP header compression on a per-interface (or subinterface) basis, see the "Configuring RTP Header Compression" module or the "Configuring TCP Header Compression" module, respectively.

Finding Feature Information, page 1
Prerequisites for Class-Based RTP and TCP Header Compression, page 2
Restrictions for Class-Based RTP and TCP Header Compression, page 2
Information About Class-Based RTP and TCP Header Compression, page 2
How to Configure Class-Based RTP and TCP Header Compression, page 4
Configuration Examples for Class-Based RTP and TCP Header Compression, page 8
Additional References, page 11
Glossary, page 13
Feature Information for Class-Based RTP and TCP Header Compression, page 13

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.

Prerequisites for Class-Based RTP and TCP Header Compression

Before configuring class-based RTP and TCP header compression, read the information in the "Header Compression" module.

Restrictions for Class-Based RTP and TCP Header Compression

Class-based RTP and TCP header compression can be enabled on PPP interfaces, High-Level Data Link Control (HDLC) interfaces, and interfaces that use Frame Relay encapsulation. However, note the following points about the header-compression formats supported on these interfaces:

- For PPP and HDLC interfaces, the only supported format for header compression is the IPHC (IP Header Compression) format.
- For interfaces that use Frame Relay encapsulation, the IPHC format is not available. The only supported format for header compression is the Cisco proprietary format.

Information About Class-Based RTP and TCP Header Compression

- Class-Based Header Compression and the MQC, page 2
- Benefits of Class-Based Header Compression, page 3
- Header Compression on Local and Remote Routers, page 3
- About Header-Compression Connections, page 3

Class-Based Header Compression and the MQC

Class-based RTP and TCP header compression allows you to configure either RTP or TCP header compression for a specific class within a policy map (sometimes referred to as a traffic policy). You configure the class and the policy map by using the Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC). The MQC is a CLI that allows you to create classes within policy maps (traffic policies) and then attach the policy maps to interfaces (or subinterfaces). The policy maps are used to configure and apply specific QoS features (such as RTP or TCP header compression) to your network. For more information about the MQC, see the "Applying QoS Features Using the MQC" module.
Benefits of Class-Based Header Compression

Class-based header compression allows you to compress (and then decompress) a subset of the packets on your network. Class-based header compression acts as a filter; it allows you to specify at a much finer level the packets that you want to compress. For example, instead of compressing all RTP (or TCP) packets that traverse your network, you can configure RTP header compression to compress only those packets that meet certain criteria (for example, protocol type “ip” in a class called "voice)."

Header Compression on Local and Remote Routers

In a typical network topology, header compression is configured at both a local router and a remote router. If you configure class-based RTP header compression (or class-based TCP header compression) on the local router, you must also configure RTP header compression (or TCP header compression) on the remote router.

However, when you configure either RTP or TCP header compression on the remote router, you can choose one of the following:

- You can configure class-based RTP or TCP header compression on the remote router (by using the instructions in this module)

or

- You can configure RTP or TCP header compression directly on the interface of the remote router (by using the instructions in the "Configuring RTP Header Compression" module or the "Configuring TCP Header Compression" module, respectively).

Note

If you configure RTP or TCP header compression directly on the interface of the remote router, you must specify the iphc-format keyword for PPP and HDLC interfaces. For Frame Relay interfaces, the iphc-format keyword is not supported; only the Cisco proprietary format (that is, the cisco keyword) is supported. For more information about the iphc-format keyword, see either the "Configuring RTP Header Compression" module or the "Configuring TCP Header Compression" module.

About Header-Compression Connections

Number of Connections Calculated on the Basis of Bandwidth

In class-based RTP and TCP header compression, the number of header-compression connections is calculated on the basis of the amount of available bandwidth.

Note the following points about how bandwidth is used:

- The setting of the bandwidth command determines the amount of bandwidth available on the interface.
- The number of header-compression connections is calculated by dividing the available bandwidth by 4 (that is, 4 kilobits per connection).

Header-Compression Connections on HDLC and Frame Relay Interfaces

For HDLC interfaces and Frame Relay interfaces (that is, interfaces that use Frame Relay encapsulation), the number of header-compression connections on both sides of the network must match. That is, the
number calculated (from the bandwidth setting) for use on the local router must match the number configured (or calculated from the bandwidth setting) for use on the remote router.

**Header-Compression Connections on PPP Interfaces**

For PPP interfaces, if the header-compression connection numbers on both sides of the network do not match, the number used is "autonegotiated." That is, any mismatch in the number of header-compression connections between the local router and the remote router will be automatically negotiated to the lower of the two numbers. For example, if the local router is configured to use 128 header-compression connections, and the remote router is configured to use 64 header-compression connections, the negotiated number will be 64.

---

**Note**

This autonegotiation function applies to PPP interfaces only. For HDLC interfaces and interfaces that use Frame Relay encapsulation, no autonegotiation occurs.

---

**How to Configure Class-Based RTP and TCP Header Compression**

- Enabling RTP or TCP Header Compression for a Class in a Policy Map, page 4
- Attaching the Policy Map to an Interface, page 6
- Verifying the Class-Based RTP and TCP Header Compression Configuration, page 7

**Enabling RTP or TCP Header Compression for a Class in a Policy Map**

With class-based header compression, you can configure either RTP or TCP header compression for a specific class inside a policy map. To specify the class, to create a policy map, and to configure either RTP or TCP header compression for the class inside the policy map, perform the following steps.

---

**Note**

In the following task, the match protocol command is shown in step Enabling RTP or TCP Header Compression for a Class in a Policy Map, page 4 The match protocol command matches traffic on the basis on the protocol type and is only an example of a match command you can use. You may want to use a different match command to specify another criterion. The match commands vary by Cisco IOS release. See the command documentation for the Cisco IOS release that you are using for a complete list of match commands.
**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `class-map [match-all|match-any] class-map-name`
4. `match protocol protocol-name`
5. `exit`
6. `policy-map policy-map-name`
7. `class {class-name|class-default}`
8. `compression header ip {rtp | tcp}`
9. `end`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Router&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> class-map [match-all</td>
<td>match-any] class-map-name</td>
</tr>
<tr>
<td>Example: Router(config)# class-map class1</td>
<td>• Enter the class map name.</td>
</tr>
<tr>
<td><strong>Step 4</strong> match protocol protocol-name</td>
<td>(Optional) Matches traffic on the basis of the specified protocol.</td>
</tr>
<tr>
<td>Example: Router(config-cmap)# match protocol ip</td>
<td>• Enter the protocol name.</td>
</tr>
<tr>
<td>Note</td>
<td>The match protocol command matches traffic on the basis of the protocol type. The match protocol command is just an example of one of the match commands that can be used. The match commands vary by Cisco IOS release. See the command documentation for the Cisco IOS release that you are using for a complete list of match commands.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Step 5</strong> exit</td>
<td>(Optional) Exits class-map configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-cmap)# exit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> policy-map policy-map-name</td>
<td>Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy and enters policy-map configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config)# policy-map policy1</td>
<td>• Enter the policy map name.</td>
</tr>
<tr>
<td><strong>Step 7</strong> class {class-name</td>
<td>class-default}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-pmap)# class class1</td>
<td>• Enter the class name or the <strong>class-default</strong> keyword.</td>
</tr>
<tr>
<td><strong>Step 8</strong> compression header ip {rtp</td>
<td>tcp}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-pmap-c)# compression header ip rtp</td>
<td>• Enter either the <strong>rtp</strong> keyword (for RTP header compression) or the <strong>tcp</strong> keyword (for TCP header compression).</td>
</tr>
<tr>
<td><strong>Step 9</strong> end</td>
<td>(Optional) Exits policy-map class configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-pmap-c)# end</td>
<td></td>
</tr>
</tbody>
</table>

### Attaching the Policy Map to an Interface

After a policy map is created, the next step is to attach the policy map to an interface (or subinterface). To attach the policy map to an interface or subinterface, perform the following steps.

#### Note

You configure class-based RTP and TCP header compression in policy maps. Then you attach those policy maps to an interface by using the **service-policy** command. The **service-policy** command gives you the option of specifying either an input service policy (for input interfaces) or an output service policy (for output interfaces). For class-based RTP and TCP header compression, you can specify output service policies only.

---

>
**SUMMARY STEPS**

1. enable
2. configure terminal
3. interface type number [name-tag]
4. service-policy output policy-map-name
5. end

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> interface type number [name-tag]</td>
<td>Configures an interface type and enters interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter the interface type and the interface number.</td>
</tr>
<tr>
<td>Router(config)# interface serial0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> service-policy output policy-map-name</td>
<td>Specifies the name of the policy map to be attached to the interface in the output direction.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter the policy map name.</td>
</tr>
<tr>
<td>Router(config-if)# service-policy output policy1</td>
<td>Note Policy maps can be attached in the input or output direction of an interface. For class-based RTP and TCP header compression, always use the output keyword.</td>
</tr>
<tr>
<td><strong>Step 5</strong> end</td>
<td>(Optional) Exits interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config-if)# end</td>
<td></td>
</tr>
</tbody>
</table>

**Verifying the Class-Based RTP and TCP Header Compression Configuration**

This task allows you to verify that you created the intended configuration and that the feature is functioning correctly. To verify the configuration, perform the following steps.
### SUMMARY STEPS

1. enable
2. **show policy-map interface** *type number output*
3. 
4. **show policy-map** *policy-map class class-name*
5. **end**

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
• Enter your password if prompted. |
| **Step 2** **show policy-map interface** *type number output* | Displays the packet statistics of all classes that are configured for all service policies on the specified interface.  
• Enter the interface type and the interface number. |
| **Step 3** **show policy-map** *policy-map class class-name* | Displays the configuration for the specified class of the specified policy map.  
• Enter the policy map name and the class name. |
| **Step 4** **end** | (Optional) Exits privileged EXEC mode. |

### Configuration Examples for Class-Based RTP and TCP Header Compression

- Example Enabling RTP or TCP Header Compression for a Class in a Policy Map, page 9
- Example Attaching the Policy Map to an Interface, page 9
- Example Verifying the Class-Based RTP and TCP Header Compression Configuration, page 9
Example Enabling RTP or TCP Header Compression for a Class in a Policy Map

In the following example, a class map called class1 and a policy map called policy1 have been configured. Policy1 contains the class called class1, within which RTP header compression has been enabled by using the `compression header ip rtp` command.

```
Router> enable
Router# configure terminal
Router(config)# class-map class1
Router(config-cmap)# match protocol ip
Router(config-cmap)# exit
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# compression header ip rtp
Router(config-pmap-c)# end
```

Example Attaching the Policy Map to an Interface

In the following example, the policy map called policy1 has been attached to serial interface 0.

```
Router> enable
Router# configure terminal
Router(config)# interface serial0
Router(config-if)# service-policy output policy1
Router(config-if)# end
```

Example Verifying the Class-Based RTP and TCP Header Compression Configuration

This section provides sample output from a typical `showpolicy-mapinterface` command.
Depending upon the interface in use and the QoS feature enabled (such as Class-Based Weighted Fair Queuing [CBWFQ]), the output you see may vary from that shown below.

The following sample displays the statistics for serial interface 0. In this sample configuration, three classes, called gold, silver, and voice, have been configured. Traffic is classified and grouped into classes on the basis of the IP precedence value and RTP port protocol number.

```plaintext
class-map match-all gold
match ip precedence 2
class-map match-all silver
match ip precedence 1
class-map match-all voice
match ip precedence 5
match ip rtp 16384 1000
```

This sample configuration also contains a policy map called mypolicy, configured as shown below. QoS features such as RTP header compression and CBWFQ are enabled for specific classes within the policy map.

```plaintext
policy-map mypolicy
class voice
  priority 128
  compress header ip rtp
  class gold
  bandwidth 100
  class silver
  bandwidth 80
  random-detect
```

Given the classes and policy map configured as shown above, the following content is displayed for serial interface 0:

```plaintext
Router# show policy-map interface serial0 output
Serial0
Service-policy output: mypolicy

Class-map: voice (match-all)
  880 packets, 58080 bytes
  30 second offered rate 1000 bps, drop rate 0 bps
  Match: ip precedence 5
  Match: ip rtp 16384 1000
Queueing
  Strict Priority
  Output Queue: Conversation 136
  Bandwidth 128 (kbps) Burst 3200 (Bytes)
  (pkts matched/bytes matched) 880/26510
  (total drops/bytes drops) 0/0
compress:
  header ip rtp
  UDP/RTP (compression on, IPHC, RTP)
  Sent: 880 total, 877 compressed,
  31570 bytes saved, 24750 bytes sent
  2.27 efficiency improvement factor
  99% hit ratio, five minute miss rate 0 misses/sec, 0 max rate 0 bps

Class-map: gold (match-all)
  100 packets, 53000 bytes
  30 second offered rate 0 bps, drop rate 0 bps
  Match: ip precedence 2
Queueing
  Output Queue: Conversation 137
  Bandwidth 100 (kbps) Max Threshold 64 (packets)
  (pkts matched/bytes matched) 100/53000
  (depth/total drops/no-buffer drops) 0/0/0
```
Class-map: silver (match-all)
878 packets, 1255540 bytes
30 second offered rate 56000 bps, drop rate 0 bps
Match: ip precedence 1
Queueing
Output Queue: Conversation 138
Bandwidth 64 (kbps)
(pkts matched/bytes matched) 878/1255540
(depth/total drops/no-buffer drops) 0/0/0
exponential weight: 9
mean queue depth: 0

<table>
<thead>
<tr>
<th>class</th>
<th>Transmitted pkts/bytes</th>
<th>Random drop pkts/bytes</th>
<th>Tail drop pkts/bytes</th>
<th>Minimum thresh</th>
<th>Maximum thresh</th>
<th>Mark prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>20</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>1</td>
<td>878/1255540</td>
<td>0/0</td>
<td>0/0</td>
<td>22</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>2</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>24</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>3</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>26</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>4</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>28</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>5</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>30</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>6</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>32</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>7</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>34</td>
<td>40</td>
<td>1/10</td>
</tr>
<tr>
<td>rsvp</td>
<td>0/0</td>
<td>0/0</td>
<td>0/0</td>
<td>36</td>
<td>40</td>
<td>1/10</td>
</tr>
</tbody>
</table>

Additional References

The following sections provide references related to configuring class-based RTP and TCP header compression.

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>QoS commands: complete command syntax,</td>
<td>Cisco IOS Quality of Service Solutions Command</td>
</tr>
<tr>
<td>command modes, command history, defaults, usage</td>
<td>Reference</td>
</tr>
<tr>
<td>guidelines, and examples</td>
<td></td>
</tr>
<tr>
<td>MQC</td>
<td>&quot;Applying QoS Features Using the MQC&quot; module</td>
</tr>
<tr>
<td>Header compression overview</td>
<td>&quot;Header Compression&quot; module</td>
</tr>
<tr>
<td>RTP header compression</td>
<td>&quot;Configuring RTP Header Compression&quot; module</td>
</tr>
<tr>
<td>TCP header compression</td>
<td>&quot;Configuring TCP Header Compression&quot; module</td>
</tr>
<tr>
<td>IPHC profiles and header compression</td>
<td>&quot;Configuring Header Compression Using IPHC Profiles&quot; module</td>
</tr>
</tbody>
</table>
### Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards are supported, and support for existing standards has not been modified.</td>
<td>--</td>
</tr>
</tbody>
</table>

### MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified MIBs are supported, and support for existing MIBs has not been modified.</td>
<td>To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

### RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 1144</td>
<td>Compressing TCP/IP Headers for Low-Speed Serial Links</td>
</tr>
<tr>
<td>RFC 2507</td>
<td>IP Header Compression</td>
</tr>
<tr>
<td>RFC 2508</td>
<td>Compressing IP/UDP/RTP Headers for Low-Speed Serial Links</td>
</tr>
<tr>
<td>RFC 3544</td>
<td>IP Header Compression over PPP</td>
</tr>
<tr>
<td>RFC 3545</td>
<td>Enhanced Compressed RTP (CRTP) for Links with High Delay, Packet Loss and Reordering</td>
</tr>
<tr>
<td>RFC 3550</td>
<td>A Transport Protocol for Real-Time Applications</td>
</tr>
</tbody>
</table>

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
bandwidth --The rated throughput capacity of a given network medium.

compression --The running of a data set through an algorithm that reduces the space required to store the data set or the bandwidth required to transmit the data set.

full header (header refresh) --An uncompressed header that updates or refreshes the context for a packet stream. It carries a context identifier (CID) that will be used to identify the context. Full headers for non-TCP packet streams also carry the generation of the context that they update or refresh.

HDLC --High-Level Data Link Control. A bit-oriented synchronous data link layer protocol developed by the International Organization for Standardization (ISO). Derived from Synchronous Data Link Control (SDLC), HDLC specifies a data encapsulation method on synchronous serial links using frame characters and checksums.

header --A chain of subheaders.

MQC --Modular Quality of Service Command-Line Interface. The MQC is a CLI that allows you to create traffic classes and policy maps and then attach the policy maps to interfaces. The policy maps apply QoS features to your network.

PPP --Point-to-Point Protocol. A protocol that provides router-to-router and host-to-network connections over synchronous and asynchronous circuits.

regular header --A normal, uncompressed header. A regular header does not carry a context identifier (CID) or generation association.

RTP --Real-Time Transport Protocol. A protocol that is designed to provide end-to-end network transport functions for applications that transmit real-time data, such as audio, video, or simulation data, over unicast or multicast network services. RTP provides such services as payload type identification, sequence numbering, timestamping, and delivery monitoring to real-time applications.

subheader --An IPv6 base header, an IPv6 extension header, an IPv4 header, a UDP header, an RTP header, or a TCP header.

TCP --Transmission Control Protocol. A connection-oriented transport layer protocol that provides reliable full-duplex data transmission. TCP is part of the TCP/IP protocol stack.

Feature Information for Class-Based RTP and TCP Header Compression

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 Class-Based RTP and TCP Header Compression

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class-Based RTP and TCP Header Compression</td>
<td>12.2(13)T</td>
<td>This feature allows you to configure Real-Time Transport Protocol (RTP) or Transmission Control Protocol (TCP) IP header compression on a per-class basis, when a class is configured within a policy map. Policy maps are created using the Modular Quality of Service (QoS) Command-Line Interface (CLI) (MQC).</td>
</tr>
</tbody>
</table>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2011 Cisco Systems, Inc. All rights reserved.