RTP Header Compression over Satellite Links

The RTP Header Compression over Satellite Links feature allows customers to use Real-Time Transport Protocol (RTP) header compression over an asymmetric link (such as a satellite link), where the uplink and downlink connections are on separate interfaces. This feature provides improved system performance by reducing network overhead and speeding up transmission of RTP packets.

Feature History for RTP Header Compression over Satellite Links

Feature History	
Release	**Modification**
12.3(2)T	This feature was introduced.
12.2(25)S	This feature was integrated into Cisco IOS Release 12.2(25)S.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Contents

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

Information About RTP Header Compression over Satellite Links

To configure the RTP Header Compression over Satellite Links feature, you should understand the following concepts:

RTP Header Compression over Satellite Links Feature Benefits

Improved System Performance

This feature provides improved system performance by reducing network overhead and speeding transmission of RTP packets. Enabling compression on both ends of a low-bandwidth serial link can greatly reduce the network overhead if there is a lot of RTP traffic on that slow link. This feature also increases the system robustness when data is transmitted over satellite (or other asymmetric) links.

Enhanced Functionality

This feature provides enhanced functionality because it provides a method for configuring periodic refreshes of the compressed data stream using FULL_HEADER packets.

This feature also provides a means for disabling the sending of CONTEXT_STATUS feedback messages. Disabling these messages is useful when the time it takes for the packet to traverse the uplink and the downlink portions of the data path is greater than the refresh period.

Using this feature, you can specify header-compression settings such as the time period for an automatic resend (that is, a “refresh”) of FULL_HEADER packets and the number of packets transmitted before a new FULL_HEADER packet is sent.

Periodic Refreshes of a Compressed Data Stream

RTP header compression is a mechanism that compresses the IP header in a data packet before the packet is transmitted. RTP header compression requires a CONTEXT_STATUS feedback mechanism to recover when the compressed data stream experiences packet channel loss. If the round-trip time of the packet between the uplink and the downlink is lengthy, or if a feedback path does not exist, the chance of loss propagation is greatly increased when a packet is dropped from the link. For instance, if a feedback path does not exist, a compressed data stream may never recover. This situation presented a need for a configurable option that allows periodic refreshes of the compressed data stream using FULL_HEADER packets.

New periodic-refresh Keyword

When you configure header compression, you can now configure periodic refreshes of the compressed data stream using the new **periodic-refresh** keyword. That is, you can configure the **periodic-refresh** keyword with the following commands:

- **frame-relay ip rtp header-compression**
- **frame-relay map ip rtp header-compression**
- **ip rtp header-compression**

For more information about these commands, see the "Command Reference" section of this document.

Configurable Packet Refresh Settings

The RTP Header Compression over Satellite Links feature allows you to configure the time period for an automatic resend (that is, a "refresh") of FULL_HEADER packets and the number of packets transmitted before a new FULL_HEADER is sent.

The time period and number of packets transmitted are configured using two new commands—the **ip header-compression max-time** command and the **ip header-compression max-period** command. For more information about these commands, see the "Command Reference" section of this document.

With this feature configured, the FULL_HEADER packet will be sent at the specified time period. Both the time period and the number of packets specified will be reset on the transmission of any FULL_HEADER packet.

Optional Disabling of CONTEXT_STATUS Feedback Messages

This feature also allows customers to disable the sending of CONTEXT_STATUS feedback messages in instances where either the time it takes for the packet to traverse the uplink and the downlink portions of the data path is greater than the refresh period (in which case the sending of the CONTEXT_STATUS feedback message would not be useful) or when a feedback path does not exist.

Disabling the CONTEXT_STATUS feedback messages can be accomplished by using the new **ip header-compression disable-feedback** command. For more information about this command, see the "Command Reference" section of this document.

Inherited Header Compression on Frame Relay Interfaces

This feature can be configured on the following types of Frame Relay interfaces:

- Main interfaces
- Point-to-point subinterfaces
- Multipoint subinterfaces

When you configure this feature on a Frame Relay main interface or a multipoint subinterface, there are two methods for configuring the header compression properties. Either you can use inherited compression, or you can specify the header compression properties you want to use on the interface.

Using Inherited Header Compression

With inherited header-compression, if you configure header compression on a Frame Relay main interface, all the data-link connection identifiers (DLCIs) configured on that interface, and *only* on that interface, will "inherit" header compression.

Note If you configure inherited header compression on a main interface, any subinterfaces associated with that interface will *not* have header compression enabled. The header-compression properties will not be passed down to any subinterfaces.

To use inherited header compression when configuring this feature on a Frame Relay main interface, follow the instructions in the section "Configuring RTP Header Compression over Satellite Links on a Frame Relay Main Interface (Header Compression Inherited)" in this document.

To use inherited header compression when configuring this feature on a Frame Relay multipoint subinterface, follow the instructions in the section "Configuring RTP Header Compression over Satellite Links on Frame Relay Multipoint Subinterfaces (Header Compression Inherited)" in this document.

Using Specified Header-Compression Properties

As an alternative to using inherited header-compression, you can specify the header-compression properties you want to use on the interface. That is, you can specify the type of header compression and the number of connections.

To specify the header-compression properties you want to use when configuring this feature on a Frame Relay main interface, follow the instructions in the section "Configuring RTP Header Compression over Satellite Links on a Frame Relay Main Interface (Header-Compression Properties Specified)" in this document.

To specify the header-compression properties you want to use when configuring this feature on a Frame Relay multipoint subinterface, follow the instructions in the section "Configuring RTP Header Compression over Satellite Links on a Frame Relay Multipoint Subinterface (Header-Compression Properties Specified)" in this document.

Overriding an Inherited Header-Compression Configuration

You can override an inherited header-compression configuration by specifying the header-compression settings in an individual Frame Relay map, as shown in the following example:

```
interface serial 2/0
  encap frame-relay
  ip rtp header-compression
  frame-relay map ip 192.168.1.1 100
  frame-relay map ip 192.168.2.1 200
  frame-relay map ip 192.168.3.1 300 nocompress
```

In this example, DLCIs 100 and 200 will inherit the header-compression properties, but DLCI 300 will not.

You can override the inherited header-compression settings in a similar way on any of the interfaces on which this feature can be configured. After overriding inherited header compression, you can specify the settings you want to use. For more information, see the section "Specifying the Header-Compression Settings" in this document.

How to Configure RTP Header Compression over Satellite Links

This section contains the following procedures. Choose the procedure for the type of interface or subinterface you have in your network.

Configuring RTP Header Compression over Satellite Links on a Point-to-Point or HDLC Interface

To configure this feature on a Point-to-Point or High-Level Data Link Control (HDLC) interface, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask* [**secondary**]
5. **ip rtp header-compression** [**passive** | **iphc-format**] [**periodic-refresh**]
6. **ip rtp compression-connections** [*number*]
7. **exit**

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* [*name-tag*] **Example:** `Router(config)# interface Serial2/0`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 192.168.1.27 255.255.255.0`	Sets a primary or secondary IP address for an interface. • Enter the IP address.
Step 5	**ip rtp header-compression** [**passive** | **iphc-format**] [**periodic-refresh**] **Example:** `Router(config-if)# ip rtp header-compression iphc-format`	Enables RTP header compression.
Step 6	**ip rtp compression-connections** [*number*] **Example:** `Router(config-if)# ip rtp compression-connections 64`	Specifies the total number of RTP header compression connections that can exist on an interface. • (Optional) Enter the number of compression connections. If no number is entered, the default number of connections (16) is used.
Step 7	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Configuring RTP Header Compression over Satellite Links on a Frame Relay Main Interface (Header Compression Inherited)

To configure this feature on a Frame Relay main interface and use inherited header-compression, perform the following steps.

Restrictions

The encapsulation type is specified using either the **cisco** or **ietf** keyword of the **encapsulation frame-relay** command and the **frame-relay interface-dlci** command. The **cisco** keyword specifies Cisco encapsulations, and the **ietf** keyword specifies IETF encapsulations. For Frame Relay interfaces, header compression is currently supported for Cisco encapsulations only.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **encapsulation frame-relay** [**cisco** | **ietf**]
5. **ip address** *ip-address mask* [**secondary**]
6. **frame-relay interface-dlci** *dlci* [**ietf** | **cisco**]
7. **frame-relay ip rtp header-compression** [**active** | **passive**] [**periodic-refresh**]
8. **frame-relay ip rtp compression-connections** [*number*]
9. **exit**

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* [*name-tag*] **Example:** `Router(config)# interface Serial2/0`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.
Step 4	**encapsulation frame-relay** [**cisco** | **ietf**] **Example:** `Router(config-if)# encapsulation frame-relay`	Enables Frame Relay encapsulation.

	Command or Action	Purpose
Step 5	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 192.168.1.27 255.255.255.0`	Sets a primary or secondary IP address for an interface. • Enter the IP address.
Step 6	**frame-relay interface-dlci** *dlci* [**ietf** | **cisco**] **Example:** `Router(config-if)# frame-relay interface-dlci 100`	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server. • Enter the DLCI number.
Step 7	**frame-relay ip rtp header-compression** [**active** | **passive**] [**periodic-refresh**] **Example:** `Router(config-if)# frame-relay ip rtp header-compression`	Enables RTP header compression for all Frame Relay maps on a physical interface.
Step 8	**frame-relay ip rtp compression-connections** [*number*] **Example:** `Router(config-if)# frame-relay ip rtp compression-connections 10`	Specifies the maximum number of RTP header compression connections that can exist on a Frame Relay interface. • (Optional) Enter the number of compression connections. If no number is entered, the default number of connections (256) is used.
Step 9	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Configuring RTP Header Compression over Satellite Links on a Frame Relay Main Interface (Header-Compression Properties Specified)

To configure this feature on a Frame Relay main interface and specify the header-compression properties, perform the following steps.

Restrictions

The encapsulation type is specified using either the **cisco** or **ietf** keyword of the **encapsulation frame-relay** command and the **frame-relay interface-dlci** command. The **cisco** keyword specifies Cisco encapsulations, and the **ietf** keyword specifies IETF encapsulations. For Frame Relay interfaces, header compression is currently supported for Cisco encapsulations only.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **encapsulation frame-relay** [**cisco** | **ietf**]

5. **ip address** *ip-address mask* [**secondary**]
6. **frame-relay interface-dlci** *dlci* [**ietf** | **cisco**]
7. **frame-relay map ip** *ip-address dlci* [**broadcast**] **rtp header-compression** [**active** | **passive**] [**periodic-refresh**] [**connections** *number*]
8. **exit**

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* [*name-tag*] **Example:** `Router(config-if)# interface Serial2/0`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.
Step 4	**encapsulation frame-relay** [**cisco** | **ietf**] **Example:** `Router(config)# encapsulation frame-relay`	Enables Frame Relay encapsulation.
Step 5	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 192.168.1.27 255.255.255.0`	Sets a primary or secondary IP address for an interface. • Enter the IP address.
Step 6	**frame-relay interface-dlci** *dlci* [**ietf** | **cisco**] **Example:** `Router(config-if)# frame-relay interface-dlci 100`	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server. • Enter the DLCI number.

	Command or Action	Purpose
Step 7	**frame-relay map ip** *ip-address dlci* [**broadcast**] **rtp header-compression** [**active** | **passive**] [**periodic-refresh**] [**connections** *number*] **Example:** `Router(config-if)# frame-relay map ip 10.108.175.220 180 rtp header-compression connections 20`	Enables RTP header compression per DLCI. • Enter the IP address and DLCI number.
Step 8	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Configuring RTP Header Compression over Satellite Links on a Frame Relay Point-to-Point Subinterface

To configure this feature on a Frame Relay point-to-point subinterface, perform the following steps.

Restrictions

The encapsulation type is specified using either the **cisco** or **ietf** keyword of the **encapsulation frame-relay** command and the **frame-relay interface-dlci** command. The **cisco** keyword specifies Cisco encapsulations, and the **ietf** keyword specifies IETF encapsulations. For Frame Relay interfaces, header compression is currently supported for Cisco encapsulations only.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **encapsulation frame-relay** [**cisco** | **ietf**]
5. **ip address** *ip-address mask* [**secondary**]
6. **frame-relay interface-dlci** *dlci* [**ietf** | **cisco**]
7. **frame-relay ip rtp header-compression** [**active** | **passive**] [**periodic-refresh**]
8. **frame-relay ip rtp compression-connections** [*number*]
9. **exit**

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* [*name-tag*] **Example:** `Router(config)# interface Serial2/0.1 point-to-point`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.
Step 4	**encapsulation frame-relay** [**cisco** | **ietf**] **Example:** `Router(config-if)# encapsulation frame-relay`	Enables Frame Relay encapsulation.
Step 5	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 192.168.1.27 255.255.255.0`	Sets a primary or secondary IP address for an interface. • Enter the IP address.
Step 6	**frame-relay interface-dlci** *dlci* [**ietf** | **cisco**] **Example:** `Router(config-if)# frame-relay interface-dlci 100`	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server. • Enter the DLCI number.
Step 7	**frame-relay ip rtp header-compression** [**active** | **passive**] [**periodic-refresh**] **Example:** `Router(config-if)# frame-relay ip rtp header-compression`	Enables RTP header compression for all Frame Relay maps on a physical interface.
Step 8	**frame-relay ip rtp compression-connections** [*number*] **Example:** `Router(config-if)# frame-relay ip rtp compression-connections 10`	Specifies the maximum number of RTP header compression connections that can exist on a Frame Relay interface. • (Optional) Enter the number of compression connections. If no number is entered, the default number of connections (256) is used.
Step 9	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Configuring RTP Header Compression over Satellite Links on Frame Relay Multipoint Subinterfaces (Header Compression Inherited)

To configure this feature on a Frame Relay multipoint subinterface and use the inherited header-compression properties, perform the following steps.

Restrictions

The encapsulation type is specified using either the **cisco** or **ietf** keyword of the **encapsulation frame-relay** command and the **frame-relay interface-dlci** command. The **cisco** keyword specifies Cisco encapsulations, and the **ietf** keyword specifies IETF encapsulations. For Frame Relay interfaces, header compression is currently supported for Cisco encapsulations only.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **encapsulation frame-relay** [**cisco** | **ietf**]
5. **ip address** *ip-address mask* [**secondary**]
6. **frame-relay interface-dlci** *dlci* [**ietf** | **cisco**]
7. **frame-relay ip rtp header-compression** [**active** | **passive**] [**periodic-refresh**]
8. **frame-relay ip rtp compression-connections** [*number*]
9. **exit**

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* [*name-tag*] **Example:** `Router(config)# interface Serial2/0.1 multipoint`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.

	Command or Action	Purpose
Step 4	**encapsulation frame-relay** [**cisco** | **ietf**] **Example:** `Router(config-if)# encapsulation frame-relay`	Enables Frame Relay encapsulation.
Step 5	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 192.168.1.27 255.255.255.0`	Sets a primary or secondary IP address for an interface. • Enter the IP address.
Step 6	**frame-relay interface-dlci** *dlci* [**ietf** | **cisco**] **Example:** `Router(config-if)# frame-relay interface-dlci 100`	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server. • Enter the DLCI number.
Step 7	**frame-relay ip rtp header-compression** [**active** | **passive**] [**periodic-refresh**] **Example:** `Router(config-if)# frame-relay ip rtp header-compression`	Enables RTP header compression for all Frame Relay maps on a physical interface.
Step 8	**frame-relay ip rtp compression-connections** [*number*] **Example:** `Router(config-if)# frame-relay ip rtp compression-connections 10`	Specifies the maximum number of RTP header compression connections that can exist on a Frame Relay interface. • (Optional) Enter the number of compression connections. If no number is entered, the default number of connections (256) is used.
Step 9	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Configuring RTP Header Compression over Satellite Links on a Frame Relay Multipoint Subinterface (Header-Compression Properties Specified)

To configure this feature on a Frame Relay multipoint subinterface and specify the header compression properties, perform the following steps.

Restrictions

The encapsulation type is specified using either the **cisco** or **ietf** keyword of the **encapsulation frame-relay** command and the **frame-relay interface-dlci** command. The **cisco** keyword specifies Cisco encapsulations, and the **ietf** keyword specifies IETF encapsulations. For Frame Relay interfaces, header compression is currently supported for Cisco encapsulations only.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **encapsulation frame-relay** [**cisco | ietf**]
5. **ip address** *ip-address mask* [**secondary**]
6. **frame-relay interface-dlci** *dlci* [**ietf | cisco**]
7. **frame-relay map ip** *ip-address dlci* [**broadcast**] **rtp header-compression** [**active | passive**] [**periodic-refresh**] [**connections** *number*]
8. Repeat Step 7 for each destination protocol address and the DLCI (or Frame Relay PVC bundle) for which you want to define a mapping scheme.
9. **frame-relay ip rtp header-compression** [**active | passive**] [**periodic-refresh**]
10. **frame-relay ip rtp compression-connections** [*number*]
11. **exit**

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* [*name-tag*] **Example:** `Router(config)# interface Serial2/0.1 multipoint`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.
Step 4	**encapsulation frame-relay** [**cisco** | **ietf**] **Example:** `Router(config-if)# encapsulation frame-relay`	Enables Frame Relay encapsulation.
Step 5	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 192.168.1.27 255.255.255.0`	Sets a primary or secondary IP address for an interface. • Enter the IP address.

	Command or Action	Purpose
Step 6	**frame-relay interface-dlci** *dlci* [**ietf** | **cisco**] **Example:** `Router(config-if)# frame-relay interface-dlci 100`	Assigns a DLCI to a specified Frame Relay subinterface on the router or access server. • Enter the DLCI number.
Step 7	**frame-relay map ip** *ip-address dlci* [**broadcast**] **rtp header-compression** [**active** | **passive**] [**periodic-refresh**] [**connections** *number*] **Example:** `Router(config-if)# frame-relay map ip 10.108.175.220 180 rtp-header compression`	Enables RTP header compression on a link. • Enter the IP address and the DLCI number.
Step 8	Repeat Step 7 for each destination protocol address and the DLCI (or Frame Relay PVC bundle) for which you want to define a mapping scheme.	
Step 9	**frame-relay ip rtp header-compression** [**active** | **passive**] [**periodic-refresh**] **Example:** `Router(config-if)# frame-relay ip rtp header-compression`	Enables RTP header compression for all Frame Relay maps on a physical interface.
Step 10	**frame-relay ip rtp compression-connections** [*number*] **Example:** `Router(config-if)# frame-relay ip rtp compression-connections 10`	Specifies the maximum number of RTP header compression connections that can exist on a Frame Relay interface. • (Optional) Enter the number of compression connections. If no number is entered, the default number of connections (256) is used.
Step 11	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Specifying the Header-Compression Settings

To specify the header compression settings such as the maximum size of the compressed IP header, perform one or more the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]

4. **ip header-compression max-header** *max-header-size*

   and/or

   ip header-compression max-time *length-of-time*

   and/or

   ip header-compression max-period *number-of-packets*
5. **exit**

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* [*name-tag*] **Example:** `Router(config)# interface Serial2/0.1`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.
Step 4	**ip header-compression max-header** *max-header-size* **Example:** `Router(config-if)# ip header-compression max-header 30` and/or	(Optional) Specifies the maximum size of the compressed IP header.
	ip header-compression max-time *length-of-time* **Example:** `Router(config-if)# ip header-compression max-time 5000` and/or	(Optional) Specifies the maximum amount of time to wait before refreshing the compressed IP header.

	Command or Action	Purpose
	ip header-compression max-period *number-of-packets* **Example:** `Router(config-if)# ip header-compression max-period 100`	(Optional) Specifies the maximum number of compressed packets between full headers.
Step 5	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Turning Off the CONTEXT_STATUS Feedback Messages

To turn off the CONTEXT_STATUS feedback messages, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip header-compression disable-feedback**
5. **exit**

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* [*name-tag*] **Example:** `Router(config)# interface Serial2/0.1`	Configures the interface type specified and enters interface configuration mode. • Enter interface type.

	Command or Action	Purpose
Step 4	**ip header-compression disable-feedback** **Example:** `Router(config-if)# ip header-compression disable-feedback`	Disables CONTEXT_STATUS feedback messages from the interface or link.
Step 5	**exit** **Example:** `Router(config-if)# exit`	(Optional) Exits interface configuration mode.

Verifying the Configuration

To display the header compression statistics and verify that the feature is configured as you intended, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **show ip rtp header-compression** [*interface-type interface-number*] [**detail**]

   and/or

   show frame-relay ip rtp header-compression
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip rtp header-compression** [*interface-type interface-number*] [**detail**] **Example:** `Router# show ip rtp header-compression Serial0/0.1` and/or	(Optional) Shows RTP header compression statistics.

	Command or Action	Purpose
	show frame-relay ip rtp header-compression [*interface interface-number*] **Example:** `Router# show frame-relay ip rtp header-compression Serial0/0.1`	(Optional) Shows Frame Relay RTP header compression statistics.
Step 3	**exit** **Example:** `Router# exit`	(Optional) Exits EXEC mode.

Configuration Examples for RTP Header Compression over Satellite Links

This section provides the following configuration examples:

PPP and HDLC Interfaces Configuration: Example

The following is an example of the feature configured on a PPP and HDLC interface:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# ip rtp header-compression iphc-format
Router(config-if)# ip rtp compression-connections 64
Router(config-if)# exit
```

Frame Relay Main Interface (Header-Compression Properties Inherited) Configuration: Example

The following is an example of the feature configured on a Frame Relay main interface. In this example, the inherited header compression properties were used when the feature was configured.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# encapsulation frame-relay
Router(config-if)# ip address 192.168.1.27 255.255.255.0
Router(config-if)# frame-relay interface-dlci 100
Router(config-if)# frame-relay ip rtp header-compression
Router(config-if)# frame-relay ip rtp compression-connections 10
Router(config-if)# exit
```

Frame Relay Main Interface (Header-Compression Properties Specified) Configuration: Example

The following is an example of the feature configured on a Frame Relay main interface. In this example, the header compression properties were specified when the feature was configured.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# encapsulation frame-relay
Router(config-if)# ip address 192.168.1.27 255.255.255.0
Router(config-if)# frame-relay interface-dlci 100
Router(config-if)# frame-relay map ip 10.108.175.220 180 rtp header-compression
connections 20
Router(config-if)# exit
```

Frame Relay Point-to-Point Subinterface Configuration: Example

The following is an example of the feature configured on a Frame Relay point-to-point subinterface:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0.2 point-to-point
Router(config-if)# encapsulation frame-relay
Router(config-if)# ip address 192.168.1.27 255.255.255.0
Router(config-if)# frame-relay interface-dlci 100
Router(config-if)# frame-relay ip rtp header-compression
Router(config-if)# frame-relay ip rtp compression-connections 10
Router(config-if)# exit
```

Frame Relay Multipoint Subinterface (Header-Compression Properties Inherited) Configuration: Example

The following is an example of the feature configured on a Frame Relay multipoint subinterface. In this example, the inherited header compression properties were used when the feature was configured.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial3/0.0 multipoint
Router(config-if)# encapsulation frame-relay
Router(config-if)# ip address 192.168.1.27 255.255.255.0
Router(config-if)# frame-relay interface-dlci 100
Router(config-if)# frame-relay ip rtp header-compression
Router(config-if)# frame-relay ip rtp compression-connections 10
Router(config-if)# exit
```

Frame Relay Multipoint Subinterface (Header-Compression Properties Specified) Configuration: Example

The following is an example of the feature configured on a Frame Relay multipoint subinterface. In this example, the header compression properties were specified when the feature was configured.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial3/0.0 multipoint
Router(config-if)# encapsulation frame-relay
Router(config-if)# ip address 192.168.1.27 255.255.255.0
Router(config-if)# frame-relay interface-dlci 100
Router(config-if)# frame-relay map ip 10.108.175.220 180 rtp header-compression
connections 16
Router(config-if)# frame-relay map ip 10.108.175.220 190
Router(config-if)# frame-relay map ip 10.108.175.220 200
Router(config-if)# frame-relay ip rtp header-compression
Router(config-if)# frame-relay ip rtp compression-connections 10
Router(config-if)# exit
```

Verifying the Configuration: Example

The following is sample output of the **show ip rtp header-compression** command. It displays the header compression statistics for the Serial2/0.1 subinterface.

```
RTP/UDP/IP header compression statistics:
       Interface Serial2/0.1:
         Rcvd:    502 total, 484 compressed, 0 errors
                  8 dropped, 0 buffer copies, 0 buffer failures
         Sent:    412 total, 398 compressed,
                  10324 bytes saved, 56714 bytes sent
                  2.7 efficiency improvement factor
         Connect: 16 rx slots, 16 tx slots,
                  96 misses, 6 collisions
                  3 negative cache hits
                  99% hit ratio, five minute miss rate 0 misses/sec, 0 max
```

Table 1 below describes the fields shown in the display.

Table 1 show ip rtp header-compression Command Field Descriptions

Field	Description
Rcvd:	
total	Total of compressed packets and full-headers received.
compressed	Number of compressed packets received.
errors	Number of packets with errors received.
dropped	Number of ignored packets. Packets are ignored when the links to and from the header-compression database are not synchronized.
buffer copies	Number of times the header-compression mechanism had to decompress the data into a new packet, rather than using the existing packet. This field in used primarily for debugging purposes.
buffer failures	Number of failed attempts at getting a new packet. This field in used primarily for debugging purposes.
Sent:	
total	Total number of packets sent and compressed, and number of bytes with full-headers.
compressed	Number of compressed packets sent.
bytes saved	Total number of bytes saved when the packets were sent.
bytes sent	Total number of bytes sent with compressed packets and full-headers.
efficiency improvement factor	Factor of bandwidth improvement achieved by IP header compression. For example, with RTP data streams, a factor of 2.9 (or 290%) in bandwidth improvement is achieved.
Connect:	
rx slots	Number of receive (rx) contexts in use.
tx slots	Number of transmit (tx) contexts in use.
misses	Number of times a packet not matching a known stream in the database was received. This represents a new stream.
collisions	Number of times a new stream put into the hash table matched an existing entry in the hash table.
negative cache hits	If an RTP stream does not behave as expected (for instance, it changes fields in a manner not defined in the RFC for RTP), this stream is not considered to be a true RTP stream. In this instance, the packets are sent as compressed User Datagram Protocol (UDP) until the RTP stream exhibits the expected behavior. As soon as the expected behavior is exhibited, the RTP stream will be fully compressed.
hit ratio	Percentage of packets found in the context database. In most cases, this percentage should be high. This field is use primarily for debugging purposes.

Table 1 show ip rtp header-compression Command Field Descriptions (continued)

Field	Description
five minute miss rate	The number of new data flows found per second, averaged over a five minute period.
max	The highest number of new data flows reported to date.

Additional References

The following sections provide references related to RTP Header Compression over Satellite Links.

Related Documents

Related Topic	Document Title
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*, Release 12.3 T
Express RTP and TCP Header Compression configuration information	*Cisco IOS IP Configuration Guide*
IP commands associated with header compression: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Command Reference, Volume 1 of 4: Addressing and Services*, Release 12.3 T
Frame Relay configuration information and information about DLCIs	*Cisco IOS Wide-Area Networking Configuration Guide*
Frame Relay commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Wide-Area Networking Command Reference*, Release 12.3 T

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 1144	*Compressing TCP/IP Headers for Low-Speed Serial Links*
RFC 2507	*IP Header Compression*
RFC 2508	*Compressing IP/UDP/RTP Headers for Low-Speed Serial Links*
RFC 2509	*IP Header Compression over PPP*

Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/public/support/tac/home.shtml

Command Reference

This section documents new and modified commands only.

New Commands

- **ip header-compression disable-feedback**
- **ip header-compression max-header**
- **ip header-compression max-period**
- **ip header-compression max-time**

Modified Commands

- **frame-relay ip rtp header-compression**
- **frame-relay map ip rtp header-compression**
- **ip rtp header-compression**

frame-relay ip rtp header-compression

To enable Real-Time Transport Protocol (RTP) header compression for all Frame Relay maps on a physical interface, use the **frame-relay ip rtp header-compression** command in interface configuration mode. To disable the compression, use the **no** form of this command.

frame-relay ip rtp header-compression [**active** | **passive**] [**periodic-refresh**]

no frame-relay ip rtp header-compression [**active** | **passive**] [**periodic-refresh**]

Syntax Description

active	(Optional) Compresses all outgoing RTP packets.
passive	(Optional) Compresses the outgoing RTP/User Datagram Protocol (UDP)/IP header only if an incoming packet had a compressed header.
periodic-refresh	(Optional) Indicates that the compressed IP header will be refreshed periodically.

Defaults

Disabled

By default, whatever type of header compression is configured on the interface will be inherited. If header compression is not configured on the interface, the **active** keyword will be used, but no **header-compression** keyword will appear on the **show running-config** command output.

Command Modes

Interface configuration

Command History

Release	Modification
11.3	This command was introduced.
12.3(2)T	This command was integrated into Cisco IOS Release 12.3(2)T. This command was modified to include the **periodic-refresh** keyword.
12.2(25)S	This command integrated into Cisco IOS Release 12.2(25)S.

Usage Guidelines

When the **frame-relay ip rtp header-compression** command is used on the physical interface, all the interface maps inherit the command; that is, all maps will perform UDP and RTP IP header compression.

Examples

The following example enables RTP header compression for all Frame Relay maps on a physical interface:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# frame-relay ip rtp header-compression
Router(config-if)# exit
```

In the following example, RTP header compression is enabled and the optional **periodic-refresh** keyword is specified:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# frame-relay ip rtp header-compression periodic-refresh
Router(config-if)# exit
```

Related Commands

Command	Description
frame-relay ip rtp compression-connections	Specifies maximum number of RTP header compression connections on a Frame Relay interface.
frame-relay map ip nocompress	Disables both RTP and TCP header compression on a link.
show frame-relay ip rtp header-compression	Displays RTP header compression statistics for Frame Relay.

frame-relay map ip rtp header-compression

To enable Real-Time Transport Protocol (RTP) header compression per data-link connection identifier (DLCI), use the **frame-relay map ip rtp header-compression** command in interface configuration mode. To disable RTP header compression per DLCI and delete the DLCI, use the **no** form of this command.

frame-relay map ip *ip-address dlci* [**broadcast**] **rtp header-compression** [**active | passive**] [**periodic-refresh**] [**connections** *number*]

no frame-relay map ip *ip-address dlci* [**broadcast**] **rtp header-compression** [**active | passive**] [**periodic-refresh**] [**connections** *number*]

Syntax Description

ip-address	IP address of the destination or next hop.
dlci	DLCI number.
broadcast	(Optional) Forwards broadcasts to the specified IP address.
active	(Optional) Compresses outgoing RTP packets.
passive	(Optional) Compresses the outgoing RTP/User Datagram Protocol (UDP/IP header only if an incoming packet had a compressed header.
periodic-refresh	(Optional) Refreshes the compressed IP header periodically.
connections *number*	(Optional) Specifies the maximum number of RTP header compression connections. The range is from 3 to 256.

Defaults

Disabled

By default, whatever type of header compression is configured on the interface will be inherited. If header compression is not configured on the interface, the **active** keyword will be used, but no **header-compression** keyword will appear on the **show running-config** command output.

The default maximum number of header-compression connections is 256.

Command Modes

Interface configuration

Command History

Release	Modification
11.3	This command was introduced.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T. This command was modified to enable the configuration of the maximum number of header compression connections.
12.3(2)T	This command was modified to include the **periodic-refresh** keyword.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.

Usage Guidelines

When this command is configured, the specified maps inherit RTP header compression. You can have multiple Frame Relay maps, with and without RTP header compression. If you do not specify the number of RTP header compression connections, the map will inherit the current value from the interface.

Examples

The following example enables RTP header compression on the Serial1/2 interface and sets the maximum number of RTP header compression connections at 64:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/2
Router(config-if)# encapsulation frame-relay
Router(config-if)# ip address 10.108.175.110 255.255.255.0
Router(config-if)# frame-relay map ip 10.108.175.220 180 rtp header-compression
connections 64
Router(config-if)# exit
```

In the following example, RTP header compression is enabled on the Serial1/1 interface, and the optional **periodic-refresh** keyword is included in the configuration:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/1
Router(config-if)# encapsulation frame-relay
Router(config-if)# ip address 10.108.175.110 255.255.255.0
Router(config-if)# frame-relay map ip 10.108.175.220 180 rtp header-compression
periodic-refresh
Router(config-if)# exit
```

Related Commands

Command	Description
frame-relay ip rtp compression-connections	Specifies the maximum number of RTP header compression connections on a Frame Relay interface.
frame-relay ip rtp header-compression	Enables RTP header compression for all Frame Relay maps on a physical interface.
frame-relay map ip compress	Enables both RTP and TCP header compression on a link.
show frame-relay ip rtp header-compression	Displays RTP header compression statistics for Frame Relay.

ip header-compression disable-feedback

To disable CONTEXT_STATUS feedback messages from the interface or link, use the **ip header-compression disable-feedback** command in interface configuration mode. To enable CONTEXT_STATUS feedback messages from the interface or link, use the **no** form of this command.

ip header-compression disable-feedback

no ip header-compression disable-feedback

Syntax Description

This command has no arguments or keywords.

Defaults

CONTEXT_STATUS feedback messages are enabled by default.

Command Modes

Interface configuration

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.

Usage Guidelines

The **ip header-compression disable-feedback** command is designed for use with satellite links where the path for the upward link is different from the path for the downward link. When the paths are different, CONTEXT_STATUS messages are not useful.

The **ip header-compression disable-feedback** command can be used with either Real-Time Transport Protocol (RTP) or TCP header compression.

Examples

The following example disables the CONTEXT_STATUS messages on the Serial2/0 interface:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# ip header-compression disable-feedback
Router(config-if)# exit
```

Related Commands

Command	Description
ip header-compression max-header	Specifies the maximum size of the compressed IP header.
ip header-compression max-period	Specifies the maximum number of compressed packets between full headers.
ip header-compression max-time	Specifies the maximum amount of time to wait before refreshing the compressed IP header.

ip header-compression max-header

To specify the maximum size of the compressed IP header, use the **ip header-compression max-header** command in interface configuration mode. To return the size of the compressed IP header to the default value, use the **no** form of this command.

ip header-compression max-header *max-header-size*

no ip header-compression max-header *max-header-size*

Syntax Description

max-header-size	Size of the IP header, in bytes. The size of the IP header can be in the range of 20 to 168 bytes.

Defaults

168 bytes

Command Modes

Interface configuration

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.

Usage Guidelines

The *max-header-size* argument of the **ip header-compression max-header** command can be used to restrict the size of the header to be compressed.

Examples

In the following example, the **ip header-compression max-header** command is configured to specify the maximum IP header size of the packet. In this configuration, the maximum IP header size is 100 bytes.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# ip header-compression max-header 100
Router(config-if)# exit
```

Related Commands

Command	Description
ip header-compression disable-feedback	Disables CONTEXT_STATUS feedback messages from the interface or link.
ip header-compression max-period	Specifies the maximum number of compressed packets between full headers.
ip header-compression max-time	Specifies the maximum amount of time to wait before refreshing the compressed IP header.

ip header-compression max-period

To specify the maximum number of compressed packets between full headers, use the **ip header-compression max-period** command in interface configuration mode. To return the number of compressed packets to the default value, use the **no** form of this command.

ip header-compression max-period *number-of-packets*

no ip header-compression max-period *number-of-packets*

Syntax Description

number-of-packets	Specifies a number of packets between full headers. The number can be in the range of 0 to 65535 packets.

Defaults

256 packets

Command Modes

Interface configuration

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.

Usage Guidelines

With the **ip header-compression max-period** command, full IP packet headers are sent in an exponentially increasing period after there has been a change in the context status. This exponential increase in the time period avoids the necessity of exchanging messages between the mechanism compressing the header and the mechanism decompressing the header.

By default, the **ip header-compression max-period** command operates on User Datagram Protocol (UDP) traffic only. However, if the **periodic refresh** keyword of either the **frame-relay ip rtp header-compression** command or the **frame-relay map ip rtp header-compression** command is configured, the **ip header-compression max-period** command operates on both UDP and Real-Time Transport Protocol (RTP) traffic.

Examples

In the following example, the **ip header-compression max-period** command is configured to specify the number of packets between full header packets. In this configuration, the packet number specified is 160.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# ip header-compression max-period 160
Router(config-if)# exit
```

Related Commands

Command	Description
frame-relay ip rtp header-compression	Enables RTP header compression for all Frame Relay maps on a physical interface.
frame-relay map ip rtp header-compression	Enables RTP header compression per DLCI.
ip header-compression disable-feedback	Disables CONTEXT_STATUS feedback messages from the interface or link.
ip header-compression max-header	Specifies the maximum size of the compressed IP header.
ip header-compression max-time	Specifies the maximum amount of time to wait before refreshing the compressed IP header.

ip header-compression max-time

To specify the maximum amount of time to wait before refreshing the compressed IP header, use the **ip header-compression max-time** command in interface configuration mode. To return to the default value, use the **no** form of this command.

ip header-compression max-time *length-of-time*

no ip header-compression max-time *length-of-time*

Syntax Description

length-of-time	Specifies a different amount of time (other than the default) to wait before refreshing the IP header. The amount of time can be in the range of 0 to 65535 seconds.

Defaults

5 seconds

Command Modes

Interface configuration

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.

Usage Guidelines

The **ip header-compression max-time** command is designed to avoid losing too many packets if the context status of the receiver has been lost.

If a packet is to be sent and the maximum amount of time has elapsed since the last time the IP header was refreshed, a full header is sent.

By default, the **ip header-compression max-time** command operates on User Datagram Protocol (UDP) traffic only. However, if the **periodic refresh** keyword of either the **frame-relay ip rtp header-compression** command or the **frame-relay map ip rtp header-compression** command is configured, the **ip header-compression max-time** command operates on UDP and Real-Time Transport Protocol (RTP) traffic.

Examples

In the following example, the **ip header-compression max-time** command is configured to specify the maximum amount of time to wait before refreshing the compressed IP header. In this configuration the amount of time to wait is 30 seconds.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0
Router(config-if)# ip header-compression max-time 30
Router(config-if)# exit
```

Related Commands

Command	Description
frame-relay ip rtp header-compression	Enables RTP header compression for all Frame Relay maps on a physical interface.
frame-relay map ip rtp header-compression	Enables RTP header compression per DLCI.
ip header-compression disable-feedback	Disables CONTEXT_STATUS feedback messages from the interface or link.
ip header-compression max-header	Specifies the maximum size of the compressed IP header.
ip header-compression max-period	Specifies the maximum number of compressed packets between full headers.

ip rtp header-compression

To enable Real-Time Transport Protocol (RTP) header compression, use the **ip rtp header-compression** command in interface configuration mode. To disable RTP header compression, use the **no** form of this command.

ip rtp header-compression [**passive** | **iphc-format** | **ietf-format**] [**periodic-refresh**]

no ip rtp header-compression [**passive** | **iphc-format** | **ietf-format**] [**periodic-refresh**]

Syntax Description

passive	(Optional) Compresses outgoing RTP packets only if incoming RTP packets on the same interface are compressed. If you do not specify the **passive** keyword, all RTP packets are compressed.
iphc-format	(Optional) Indicates that the IP Header Compression (IPHC) format of header compression will be used.
ietf-format	(Optional) Indicates that the Internet Engineering Task Force (IETF) format of header compression will be used.
periodic-refresh	(Optional) Indicates that the compressed IP header will be refreshed periodically.

Defaults

Disabled

For PPP interfaces, the default format for header compression is the IPHC format.

For High-Level Data Link Control (HDLC) and Frame Relay interfaces, the default format for header compression is the original proprietary Cisco format. The maximum number of compression connections for the proprietary Cisco format is 256.

Command Modes

Interface configuration

Command History

Release	Modification
11.3	This command was introduced.
12.0	This command was integrated into Cisco IOS Release 12.0. This command was modified to include the **iphc-format** keyword.
12.3(2)T	This command was integrated into Cisco IOS Release 12.3(2)T. This command was modified to include the **periodic-refresh** keyword.
12.3(4)T	This command was modified to include the **ietf-format** keyword.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.

Usage Guidelines

You can compress IP/User Datagram Protocol (UDP)/RTP headers to reduce the size of your packets. Compressing headers is especially useful for RTP because RTP payload size can be as small as 20 bytes, and the uncompressed header is 40 bytes.

Header Compression passive Keyword

By default, the **ip rtp header-compression** command compresses outgoing RTP traffic. This command includes an optional **passive** keyword. If you specify the **passive** keyword, outgoing RTP traffic is compressed only if *incoming* RTP traffic on the *same* interface is compressed. If you do not specify the **passive** keyword, *all* RTP traffic is compressed.

For PPP interfaces, the **passive** keyword is ignored. PPP interfaces negotiate the use of header-compression, regardless of whether the **passive** keyword is specified. Therefore, on PPP interfaces, the **passive** keyword is replaced by the IPHC format, the default format for PPP interfaces.

Header Compression iphc-format Keyword

This command includes the **iphc-format** keyword. The **iphc-format** keyword indicates the type of header compression that will be used. For PPP and HDLC interfaces, when the **iphc-format** keyword is specified, TCP header-compression is also enabled. For this reason, the **ip tcp header-compression** command appears in the output of the **show running-config** command. Since both RTP and TCP header compression are enabled, both UDP and TCP packets are compressed.

The **iphc-format** keyword includes checking whether the destination port number is even and in the ranges of 16385 to 32767 (for Cisco audio) or 49152 to 65535 (for Cisco video). Valid RTP packets that meet the criteria (that is, the port number is even and within the specified range) are compressed using the compressed RTP packet format. Otherwise, packets are compressed using the less-efficient compressed non-TCP packet format.

Note For Frame Relay interfaces, the **iphc-format** keyword is not available.

Header Compression ietf-format Keyword

This command includes the **ietf-format** keyword. The **ietf-format** keyword indicates the type of header compression that will be used. For HDLC interfaces, the ietf-format compresses only UDP packets. For PPP interfaces, when the **ietf-format** keyword is specified, TCP header-compression is also enabled. For this reason, the **ip tcp header-compression** command appears in the output of the **show running-config** command. Since both RTP and TCP header compression are enabled, both UDP and TCP packets are compressed.

However, with the **ietf-format** keyword, the requirement of checking whether a destination port number is in a specific range has been removed. Any even destination port number higher than 1024 can be used. Valid RTP packets that meet the criteria (that is, the port number is even and higher than 1024), are compressed using the compressed RTP packet format. Otherwise, packets are compressed using the less-efficient compressed non-TCP packet format.

Note For Frame Relay interfaces, the **ietf-format** keyword is not available.

Support for Serial Lines

RTP header compression is supported on serial lines using Frame Relay, HDLC, or PPP encapsulation. You must enable compression on both ends of a serial connection.

Unicast or Multicast RTP Packets

This command can compress unicast or multicast RTP packets, and, hence, multicast backbone (MBONE) traffic can also be compressed over slow links. The compression scheme is beneficial only when you have small payload sizes, as in audio traffic.

Examples

The following example enables RTP header compression on the Serial1/0.0 subinterface and limits the number of RTP header compression connections to 10. In this example, the optional **iphc-format** keyword of the **ip rtp header-compression** command is specified.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/0.0
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression iphc-format
Router(config-if)# ip rtp compression-connections 10
Router(config-if)# exit
```

The following example enables RTP header compression on the Serial2/0.0 subinterface and limits the number of RTP header compression connections to 20. In this example, the optional **ietf-format** keyword of the **ip rtp header-compression** command is specified.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0.0
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression ietf-format
Router(config-if)# ip rtp compression-connections 20
Router(config-if)# exit
```

In the following example, RTP header compression is enabled on the Serial1/0.1 subinterface and the optional **periodic-refresh** keyword of the **ip rtp header-compression** command is specified:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/0.1
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression iphc-format periodic-refresh
Router(config-if)# ip rtp compression-connections 10
Router(config-if)# exit
```

Related Commands

Command	Description
clear ip rtp header-compression	Clears RTP header compression structures and statistics.
ip rtp compression-connections	Specifies the total number of RTP header compression connections that can exist on an interface.
show ip rtp header-compression	Displays RTP header compression statistics.
show running-config	Displays the contents of the currently running configuration file or the configuration for a specific interface, or map class information.

Glossary

compression—Act of reducing the size of a header by removing header fields or reducing the size of header fields. Compression is done in a way such that a decompressor can reconstruct the header if its context state is identical to the context state used when compressing the header.

CID—context identifier. Small, unique number identifying the context that should be used to decompress a compressed header. Carried in full headers and compressed headers.

context—State which the compressor uses to compress a header and the decompressor uses to decompress a header. The context is the uncompressed version of the last header sent and includes other information used to compress and decompress the packet.

CONTEXT_STATE—Special packet sent from the decompressor to the compressor to communicate a list of (TCP or NON_TCP/RTP) CIDs for which synchronization has been lost. This packet is sent only over a single link so it requires no IP header.

decompression—Act of reconstructing a compressed header.

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

header—Chain of subheaders.

incorrect decompression—When a compressed and then decompressed header is different from the uncompressed header. This variance is usually due to mismatching context between the compressor and decompressor or bit errors during transmission of the compressed header.

packet stream—Sequence of packets whose headers are similar and share context. For example, headers in a TCP packet stream have the same source and final destination address, and the same port numbers in the TCP header. Similarly, headers in a UDP packet stream have the same source and destination address, and the same port numbers in the UDP header.

regular header—Normal, uncompressed, header. Does not carry CID or generation association.

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

Note Refer to the *Internetworking Terms and Acronyms* for terms not included in this glossary.