



# Serial Interface Commands on the Cisco IOS XR Software

---

This module provides CLI commands for configuring serial interfaces on the Cisco XR 12000 Series Router.

- [clear iphc ipv4](#), page 3
- [crc \(serial\)](#), page 5
- [description \(IPHC profile\)](#), page 7
- [encapsulation \(serial\)](#), page 8
- [feedback disable](#), page 10
- [fragment end-to-end](#), page 11
- [fragment-counter](#), page 13
- [interface serial](#), page 15
- [invert](#), page 18
- [iphc non-tcp connections](#), page 19
- [iphc profile](#), page 20
- [iphc tcp connections](#), page 22
- [ipv4 iphc profile](#), page 23
- [keepalive \(serial\)](#), page 25
- [max-header](#), page 27
- [non-tcp compression](#), page 28
- [non-tcp context absolute](#), page 29
- [pvc \(serial\)](#), page 31
- [refresh max-period](#), page 33
- [refresh rtp](#), page 35
- [rtp](#), page 36

- [scramble](#), page 37
- [serial](#), page 39
- [show iphc idb](#), page 40
- [show iphc ipv4 rtp](#), page 42
- [show iphc ipv4 tcp](#), page 44
- [show iphc platform trace](#), page 46
- [show iphc profile](#), page 48
- [show iphc trace all](#), page 51
- [show tech-support iphc](#), page 53
- [tcp compression](#), page 54
- [tcp context absolute](#), page 55
- [transmit-delay \(serial\)](#), page 56

# clear iphc ipv4

To clear all Real Time Protocol (RTP) and Transport Control Protocol (TCP) statistics for IP header compression (IPHC) packets sent and received on an interface, use the **clear iphc ipv4** command in EXEC mode.

```
clear iphc ipv4 {interface {serial| multilink} interface-path-id} location node-id}
```

## Syntax Description

<b>interface</b>	Specifies the interface to be configured, by type and the <i>interface-path-id</i> argument.
<b>serial</b>	Specifies a serial network interface.
<b>multilink</b>	Specifies a multilink network interface.
<i>interface-path-id</i>	Physical interface or virtual interface. <b>Note</b> Use the <b>show interfaces</b> command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
<b>location</b>	Specifies the interface to be configured by its <i>node-id</i> .
<i>node-id</i>	Fully qualified path of the node in the <i>rack/slot/module notation</i> .

## Command Default

No default behavior or values

## Command Modes

EXEC

## Command History

Release	Modification
Release 3.9.0	This command was introduced.

## Usage Guidelines

### Note

The **clear counters** command also clears the IPHC statistics for all the interfaces.

## Task ID

Task ID	Operations
ip-services	read, write

**Examples**

The following example shows how to clear RTP and TCP statistics on an interface:

```
RP/0/0/CPU0:router# clear iphc ipv4 interface Serial 0/1/0/1/26:0
Thu Jan  8 20:30:38.155 UTC
```

The following example shows how to clear RTP and TCP statistics on a node:

```
RP/0/0/CPU0:router# clear iphc ipv4 location 0/3/CPU0
Mon Oct 12 22:47:51.430 DST
```

## crc (serial)

To set the length of the cyclic redundancy check (CRC) on a serial interface, use the **crc** command in serial configuration mode. To return the CRC setting on a serial interface to the default setting, use the **no** form of this command.

```

crc {16| 32}
no crc {16| 32}

```

Syntax Description	16	Sets 16-bit CRC mode.
	32	Sets 32-bit CRC mode.

**Command Default** The default is 16 bits for serial interfaces.

**Command Modes** Serial configuration

**Command History**

Release 3.3.0	This command was introduced.
---------------	------------------------------

**Usage Guidelines** *CRC* is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. The designators 16 and 32 indicate the length (in bits) of the frame check sequence (FCS). A CRC of 32 bits provides more powerful error detection, but adds overhead. Both the sender and receiver must use the same setting.

CRC-16, the most widely used error checking method throughout the United States and Europe, is used extensively with WANs. CRC-32 is specified by IEEE standard 802 and as an option by some point-to-point transmission standards. It is often used on Switched Multimegabit Data Service (SMDS) networks and LANs.

Task ID	Task ID	Operations
	hdlc	read, write

**Examples** In the following example, the 32-bit CRC on serial interface 0/3/0/0:10 is enabled:

```

RP/0/0/CPU0:router(config)# interface serial 0/3/0/0:10
RP/0/0/CPU0:router(config-if)# serial
RP/0/0/CPU0:router(config-if-serial)# crc 32

```

**Related Commands**

Command	Description
<a href="#">show interfaces</a>	Displays statistics for all interfaces configured on the router or for a specific node.

## description (IPHC profile)

To add a description to an IPHC profile, use the **description** command in IPHC profile configuration mode. To remove a description for an IPHC profile, use the **no** form of this command.

**description** *description*

**no description**

### Syntax Description

<i>description</i>	Description to be added to the IPHC profile.
--------------------	--

### Command Default

By default, no description is attached to an IPHC profile.

### Command Modes

IPHC profile configuration

### Command History

Release	Modification
Release 3.9.0	This command was introduced.

### Usage Guidelines

#### Task ID

Task ID	Operations
ip-services	read, write

### Examples

In the following example, a description is attached to the IPHC profile test:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile test type iphc
RP/0/0/CPU0:router(config-iphc-profile)# description testprofile
RP/0/0/CPU0:router(config-iphc-profile)# commit
```

# encapsulation (serial)

To set the Layer 2 encapsulation of an interface, use the **encapsulation** command in interface configuration mode. To restore the system to the default encapsulation, use the **no** form of this command.

**encapsulation** {hdlc| ppp| frame-relay| mfr}

**no encapsulation**

## Syntax Description

<b>hdlc</b>	Enables Cisco High-Level Data Link Control (HDLC) encapsulation on the interface. This is the default encapsulation type.
<b>ppp</b>	Enables PPP encapsulation on the interface.
<b>frame -relay</b>	Enables Frame Relay encapsulation on the interface.
<b>mfr</b>	Enables multilink Frame Relay encapsulation on the interface.

## Command Default

For serial interfaces, the default encapsulation is HDLC.

## Command Modes

Interface configuration

## Command History

Release 3.2	This command was first supported.
Release 3.4.0	Frame Relay encapsulation was supported on serial interfaces.
Release 3.6.0	Multilink Frame Relay encapsulation was supported on serial interfaces.

## Usage Guidelines

### Task ID

Task ID	Operations
hdlc	read, write
interface	read, write

## Examples

The following example shows how to enable PPP encapsulation on serial interface 0/3/0/1:

```
RP/0/0/CPU0:router(config)# interface serial 0/3/0/1
```



```
RP/0/0/CPU0:router(config-if)# encapsulation ppp
```

**Related Commands**

Command	Description
<a href="#">show interfaces</a>	Displays statistics for all interfaces configured on the router or for a specific node.
<a href="#">show ppp interfaces</a>	Displays PPP state information for an interface.

## feedback disable

To disable the IP header compression (IPHC) context status feedback messages on an interface, use the **feedback disable** command in IPHC profile configuration mode. To re-enable feedback messages after they are disabled, use the **no** form of this command.

**feedback disable**

**no feedback disable**

**Syntax Description** This command has no keywords or arguments.

**Command Default** Feedback messages are enabled by default.

**Command Modes** IPHC profile configuration

Command History	Release	Modification
	Release 3.9.0	This command was introduced.

### Usage Guidelines

**Note** Feedback disable can be configured only within an IPHC profile.

Task ID	Task ID	Operations
	ip-services	read, write

**Examples** The following example shows how to disable the IP header compression (IPHC) context status feedback messages within an IPHC profile:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
RP/0/0/CPU0:router(config-iphc-profile)# feedback disable
```

## fragment end-to-end

To enable fragmentation of Frame Relay frames on an interface and enter Frame Relay virtual circuit fragment configuration mode, use the **fragment end-to-end** command in serial Frame Relay PVC configuration mode. To disable Frame Relay fragmentation, use the **no** form of this command.

**fragment end-to-end** *fragment-size* [**fragment-counter**]

**no fragment end-to-end**

### Syntax Description

<i>fragment-size</i>	Number of payload bytes from the original Frame Relay frame that go into each fragment. This number excludes the Frame Relay header of the original frame.  All the fragments of a Frame Relay frame, except the last, have a payload size equal to <i>fragment-size</i> ; the last fragment has a payload less than or equal to <i>fragment-size</i> . Valid values are from 64 to 512 bytes, depending on your hardware.
<b>fragment-counter</b>	(Optional) Enables fragmentation counters.

### Command Default

### Command Modes

### Command History

Release 3.5.0	This command was introduced.
Release 4.0.0	<ul style="list-style-type: none"> <li>The supported fragment size range was changed to 64 to 512 bytes.</li> <li>The <b>fragment-counter</b> keyword was added.</li> </ul>

### Usage Guidelines

The Cisco 8-Port Channelized T1/E1 SPA supports fragment sizes of 128, 256, and 512 bytes.

### Task ID

Task ID	Operations
fr	read, write

### Examples

The following example shows how to enter serial Frame Relay virtual circuit configuration mode, set the fragmentation size of Frame Relay frames on subinterface 0/6/2/4.1 to 512 bytes and enable fragmentation counters:

```
RP/0/0/CPU0:router (config)# interface serial 0/6/2/4.1 point-to-point
```

```
RP/0/0/CPU0:router(config-subif)# pvc 100
RP/0/0/CPU0:router(config-fr-vc)# fragment end-to-end 512 fragment-counter
```

**Related Commands**

Command	Description
<a href="#">interface serial, on page 15</a>	Configures a serial interface and enters interface or subinterface configuration mode.
<a href="#">fragment-counter, on page 13</a>	Enables fragmentation counters for a Frame Relay subinterface and PVC.

# fragment-counter

To enable fragmentation counters for a Frame Relay subinterface and PVC, use the **fragment-counter** command in Frame Relay virtual circuit fragment configuration mode. To disable collection of fragmentation counters, use the **no** form of this command.

**fragment-counter**

**no fragment-counter**

**Syntax Description** This command has no keywords or arguments.

**Command Default** Fragmentation counters are disabled.

**Command Modes** Frame Relay virtual circuit fragment configuration

Command History	Release	Modification
	Release 4.0.0	This command was introduced.

## Usage Guidelines

Task ID	Task ID	Operations
	fr	read, write

**Examples** The following example shows how to enter serial Frame Relay PVC configuration mode, set the fragmentation size of Frame Relay frames on subinterface 0/6/2/4.1 to 512 bytes and enable the fragmentation counter:

```
RP/0/0/CPU0:router(config)# interface serial 0/6/2/4.1 12transport
RP/0/0/CPU0:router(config-subif)# pvc 100
RP/0/0/CPU0:router(config-fr-vc)# fragment end-to-end 512
RP/0/0/CPU0:router(config-fr-vc-frag)# fragment-counter
```

Related Commands	Command	Description
	<a href="#">interface serial</a> , on page 15	Configures a serial interface and enters interface or subinterface configuration mode.
	<a href="#">pvc (serial)</a> , on page 31	Creates a Frame Relay PVC under a serial subinterface and enters Frame Relay virtual circuit configuration mode.

Command	Description
<a href="#">fragment end-to-end, on page 11</a>	
<a href="#">show frame-relay pvc</a>	Displays statistics about Frame Relay PVCs.

# interface serial

To configure a serial interface and enter interface or subinterface configuration mode, use the **interface serial** command in global configuration mode. To delete a serial configuration, use the **no** form of this command.

**interface serial** *interface-path-id* [. *subinterface*] {**point-to-point**| **l2transport**}

**no interface serial** *interface-path-id* [. *subinterface*] {**point-to-point**| **l2transport**}

## Syntax Description

*interface-path-id*[.*subinterface*] Physical interface or virtual interface followed by the optional subinterface path ID. Naming notation is *interface-path-id.subinterface*. The period in front of the subinterface value is required as part of the notation.

For more information about the syntax for the router, use the question mark (?) online help function.

**point-to-point** Interface functions as one endpoint of a point-to-point link.

**l2transport** Interface functions as one endpoint on an Layer 2 link.

## Command Default

No default behavior or values

## Command Modes

Global configuration

## Command History

Release 3.3.0 This command was introduced.

Release 3.5.0 The **l2transport** keyword was added to support Layer 2 configuration on serial interfaces.

## Usage Guidelines

For the *interface-path-id* argument, use the following guidelines:



### Note

A slash between values is required as part of the notation.

- If specifying a physical interface, the naming notation is *rack/slot/module/port*. The slash between values is required as part of the notation. An explanation of each component of the naming notation is as follows:
  - *rack*—Chassis number of the rack.
  - *slot*—Physical slot number of the line card.
  - *module*—Module number. A physical layer interface module (PLIM) is always 0.

◦ *port*—Physical port number of the interface.

- If specifying a virtual interface, the number range varies, depending on interface type.
- The naming notation for T1 interfaces on a channelized SPA is *rack/slot/module/port/channel-num:channel-group-number*, as shown in the following example:

```
interface serial 0/0/1/2/4:3
```

- If a subinterface is configured under the serial interface, then the router includes the subinterface number at the end of the serial interface address. In this case, the naming notation is *rack/slot/module/port[/channel-num:channel-group-number].subinterface*, as shown in the following example:

```
interface serial 0/0/1/2.1
```

- The naming notation syntax for serial interfaces is as follows:
  - *rack*—Chassis number of the rack.
  - *slot*—Physical slot number of the modular services card or line card.
  - *module*—Module number. Shared port adapters (SPAs) are referenced by their subslot number.
  - *port*—Physical port number of the controller.
  - *channel-num*:—T1 channel number. T1 channels range from 0 to 23.
  - *channel-group-number*:—Time slot number. T1 time slots range from 1 to 24. The *channel-group-number* is preceded by a colon and not a slash.
  - *subinterface*—Subinterface number.
- Use the question mark (?) online help function following the **serial** keyword to view a list of all valid interface choices.

Serial interfaces on channelized T3 can be deleted using the **no channel-group** command in T1 configuration mode. If there are nondefault serial parameters defined, you need to use the **no interface serial** command first to revert to the default configuration, and then delete the serial interface using the **no channel-group** command.

## Task ID

Task ID	Operations
interface	read, write

## Examples

The following example shows how to enter interface configuration mode for a serial interface in slot 6, subslot 2, port 4, T1 channel number 10 and channel group 8:

```
RP/0/0/CPU0:router(config)# interface serial 0/6/2/4/10:8
RP/0/0/CPU0:router(config-if)#
```



The following example shows how to create a subinterface on a serial interface in slot 6, subslot 2, port 3 and enter subinterface configuration mode:

```
RP/0/0/CPU0:router(config)# interface serial 0/6/2/3.1
RP/0/0/CPU0:router(config-if)#
```

The following example shows how to reference the serial interface on channel group 3 of T1 channel group 4 on port 2 of a SPA in subslot 1 and enter subinterface configuration mode:

```
RP/0/0/CPU0:router(config)# interface serial 0/0/1/2/4:3
RP/0/0/CPU0:router(config-if)#
```

### Related Commands

Command	Description
<a href="#">channel-group</a>	Configures a DS0 channel group and enters channel group configuration mode.
<a href="#">show interfaces</a>	Displays statistics for all interfaces configured on the router or for a specific node.

# invert

To invert the data stream on a serial interface, use the **invert** command in serial configuration mode. To disable data inversion, use the **no** form of this command.

**invert**

**no invert**

**Syntax Description** This command has no keywords or arguments.

**Command Default** Data is not inverted.

**Command Modes** Serial configuration

Command History	Release	Modification
	Release 3.3.0	This command was introduced.

**Usage Guidelines** To verify that data inversion is configured on the interface, use the **show interfaces serial** command.

Task ID	Task ID	Operations
	hdlc	read, write

**Examples** In the following example, data inversion is enabled on serial interface 0/3/0/0/0:10:

```
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0/0:10
RP/0/0/CPU0:router(config-if)# serial
RP/0/0/CPU0:router(config-if-serial)# invert
```

Related Commands	Command	Description
	<a href="#">show interfaces</a>	Displays statistics for all interfaces configured on the router or for a specific node.

## iphc non-tcp connections

To set the maximum number of non-TCP connections that may be configured for IP header compression (IPHC) on a line card, use the **iphc tcp connections** command in configuration mode. To remove this setting, use the **no** form of this command.

**iphc non-tcp connections** *max-number* **location** *node-id*

**no iphc non-tcp connections** *max-number* **location** *node-id*

### Syntax Description

<i>max-number</i>	Maximum number of non-TCP connections that may be configured for IPHC. The range is 1 to 20000.
<b>location</b>	Location, specified by <i>node-id</i> , on which to set the maximum number of connections for IPHC.
<i>node-id</i>	Fully qualified path of the node in the format <i>rack/slot/port</i> .

### Command Default

No default behavior or values

### Command Modes

Configuration

### Command History

Release	Modification
Release 3.9.0	This command was introduced.

### Usage Guidelines

#### Task ID

Task ID	Operations
ip-services	read, write

### Examples

The following example shows how to set the maximum number of non-TCP connections that may be configured for IP header compression (IPHC) on a line card:

```
RP/0/0/CPU0:router# config
RP/0/0/CPU0:router(config)# iphc non-tcp connections 20000 location 0/1/cpu0
```

## iphc profile

To create an IP header compression (IPHC) profile and enter the IPHC profile configuration mode, use the **iphc profile** command in configuration mode. To remove the profile, use the **no** form of this command.

```
iphc profile profile-name type {ietf|iphc}
no iphc profile profile-name [type {ietf|iphc}]
```

### Syntax Description

<i>profile-name</i>	Text name for the IPHC profile. The maximum number of characters is 50.
<b>type</b>	Specifies the type of compression format.
<b>ietf</b>	Specifies Internet Engineering Task Force (IETF) standard format. Uses RFC2507 and RFC2508 compression schemes.
<b>iphc</b>	Specifies Internet Protocol Header Compression (IPHC) format. Provides options similar to IETF.

### Command Default

No default behavior or values

### Command Modes

Configuration

### Command History

Release	Modification
Release 3.9.0	This command was introduced.

### Usage Guidelines

After you create a profile and enter the IPHC profile configuration mode, you can configure IPHC features in the profile and attach the profile to multiple interfaces. The maximum number of profiles allowed on a router is 250.

A profile cannot be deleted if it is attached to any interfaces. You must remove the profile from all interfaces first. Then, delete the profile using the **no** form of this command.

On-the-fly modifications to IPHC profiles are not supported.

A profile name cannot exceed 50 characters. If you attempt to create a profile name that exceeds 50 characters, you receive the following error message:

```
iphc profile test1test1test1test1test1test1test1test1test1test11 type iphc
!!% 'iphc_profilemgr' detected the 'warning' condition
'Name is longer than allowed character length of 50'
!
```

**Task ID**

<b>Task ID</b>	<b>Operations</b>
ip-services	read, write

**Examples**

The following example shows how to create the IPHC profile Profile\_1

```
RP/0/0/CPU0:router(config)# config  
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc  
RP/0/0/CPU0:router(config-iphc-profile)#
```

## iphc tcp connections

To set the maximum number of TCP connections that may be configured for IP header compression (IPHC) on a line card, use the **iphc tcp connections** command in configuration mode. To remove this setting, use the **no** form of this command.

**iphc tcp connections** *max-number* **location** *node-id*

**no iphc tcp connections** *max-number* **location** *node-id*

### Syntax Description

<i>max-number</i>	Maximum number of TCP connections that may be configured for IPHC. The range is 1 to 2000.
<b>location</b>	Location of the card, specified by <i>node-id</i> .
<i>node-id</i>	Fully qualified path of the node in the format <i>rack/slot/port</i> .

### Command Default

No default behavior or values

### Command Modes

Configuration

### Command History

Release	Modification
Release 3.9.0	This command was introduced.

### Usage Guidelines

#### Task ID

Task ID	Operations
ip-services	read, write

### Examples

The following example shows how to set the maximum number of TCP connections that may be configured for IP header compression (IPHC) on a line card:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc tcp connections 2000 location 0/1/cpu0
```

## ipv4 iphc profile

To attach an IP header compression (IPHC) profile to an interface, use the **ipv4 iphc profile** command in interface configuration mode. To remove the profile from the interface, use the **no** form of this command.

**ipv4 iphc profile** *profile-name* [**mode service-policy**]

**no ipv4 iphc profile** [*profile-name* [**mode service-policy**]]

### Syntax Description

<i>profile-name</i>	Text name of the configured IPHC profile to attach to this interface.
<b>mode service-policy</b>	(Optional) Specifies that the IPHC profile applies to a QoS service policy.

### Command Default

No default behavior or values

### Command Modes

Interface configuration

### Command History

Release	Modification
Release 3.9.0	This command was introduced.

### Usage Guidelines

If the profile name is not recognized the system returns the following error message:

```
!!% 'iphc_ma' detected the 'warning' condition 'Profile doesn't exist'
```

If the encapsulation on the interface is not supported, the system returns the following error message:

```
!!% 'iphc_ma' detected the 'warning' condition 'IPHC capability: Encap type not supported'
```

### Task ID

Task ID	Operations
ip-services	read, write
ipv4	read, write

### Examples

The following example shows how to attach an IP header compression (IPHC) profile to an interface.

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# interface serial 0/1/0/1
RP/0/0/CPU0:router(config-if)# ipv4 iphc profile Profile_1
```

The following example shows how to attach an IPHC profile that applies to a QoS service policy to an interface:

```
RP/0/0/CPU0:router(config)# config  
RP/0/0/CPU0:router(config)# interface serial 0/1/0/1  
RP/0/0/CPU0:router(config-if)# ipv4 iphc profile Profile_1 mode service-policy
```



# keepalive (serial)

To set the keepalive timer for a specific serial interface, use the **keepalive** command in interface configuration mode. To reset the keepalive timer to the default of 10 seconds, use the **no** form of this command.

**keepalive** {*interval* [*retry* ]} **disable**}

**no keepalive**

## Syntax Description

<i>interval</i>	Number of seconds (from 1 to 30) between keepalive messages. The default is 10.
<b>disable</b>	Turns off the keepalive timer.
<i>retry</i>	(Optional) Number of keepalive messages (from 1 to 255) that can be sent to a peer without a response before transitioning the link to down state. The default is 5.

## Command Default

The default interval is 10 seconds between keepalive messages. The default retry is 5 keepalive messages that can be sent without a response. However, when more than 5 keepalive messages are sent to a peer without a response, the link transitions to the down state.

## Command Modes

Interface configuration

## Command History

Release 3.2	This command introduced.
-------------	--------------------------

## Usage Guidelines

HDLC keepalives require that the **keepalive** command is configured the same way on both ends of a single connection. The two connected routers have no way of negotiating the keepalive value because there is no way for one router to tell the other about its configured values. The keepalive value configured on each router (local or partner) sets the rate at which the Cisco IOS XR software sends packets. It also sets the rate at which the local end expects to receive incoming packets.

To set the keepalive value to the default value, use the **keepalive** command without specifying a value for the *interval* argument.

By default, if more than five keepalive messages are sent to a peer and no response is received from the peer, then the link transitions to the down state.

## Task ID

Task ID	Operations
hdlc	read, write

**Examples**

The following example shows how to configure keepalives for 3 seconds on serial interface 0/7/0/1:

```
RP/0/0/CPU0:router(config)# interface serial 0/7/0/1  
RP/0/0/CPU0:router(config-if)# keepalive 3
```

**Related Commands**

Command	Description
<a href="#">show interfaces</a>	Displays statistics for all interfaces configured on the router or for a specific node.

# max-header

To define the maximum size header that can be compressed, use the **max-header** command in IPHC profile configuration mode. To return to the default maximum size, use the **no** form of this command.

**max-header** *number-of-bytes*

**no max-header** [ *number-of-bytes* ]

## Syntax Description

<i>number-of-bytes</i>	Maximum size, in bytes, of a header that can be compressed. The range is from 20 to 40. The default is 40.
------------------------	--

## Command Default

*Number-of-bytes*; 40.

## Command Modes

IPHC profile configuration

## Command History

Release	Modification
Release 3.9.0	This command was introduced.

## Usage Guidelines

### Note

The maximum header size can be configured only within an IPHC profile.

## Task ID

Task ID	Operations
ip-services	read, write

## Examples

The following example shows how to define the maximum size header that can be compressed.

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
RP/0/0/CPU0:router(config-iphc-profile)# max-header 20
```

## non-tcp compression

To enable non-TCP compression in an IP header compression (IPHC) profile, use the **non-tcp compression** command in IPHC profile configuration mode. To disable non-TCP compression in the profile, use the **no** form of this command.

**non-tcp compression**

**no non-tcp**

**Syntax Description** This command has no keywords or arguments.

**Command Default** No default behavior or values

**Command Modes** IPHC profile configuration

Command History	Release	Modification
	Release 3.9.0	This command was introduced.

### Usage Guidelines

**Note** NON-TCP compression can be enabled only within an IPHC profile. Non-TCP compression does not work unless it is enabled under a profile.

Task ID	Task ID	Operations
	ip-services	read, write

**Examples** The following example shows how to enable NON-TCP compression within an IP header compression (IPHC) profile:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
RP/0/0/CPU0:router(config-iphc-profile)# non-tcp compression
```

## non-tcp context absolute

To configure the maximum number of non-TCP contexts that are allowed for IPHC under a profile, use the **non-tcp context absolute** command in IPHC profile configuration mode. To remove the non-TCP context from the profile, use the **no** form of this command.

**non-tcp context absolute** *number-of-contexts*

**no non-tcp context** [**absolute** [ *number-of-contexts* ]]

### Syntax Description

<i>number-of-contexts</i>	Numeric value that specifies the maximum number of non-TCP contexts allowed for IPHC under this profile. The range is from 0 to 6000.
---------------------------	---

### Command Default

If the number of contexts is not specified, and only non-TCP compression is enabled, the default number of contexts is 16.

### Command Modes

IPHC profile configuration

### Command History

Release	Modification
Release 3.9.0	This command was introduced.

### Usage Guidelines

The maximum allowed number of non-tcp contexts on a Line Card, across all IPHC profiles and interfaces, is 6000.



#### Note

Non-TCP context can be set only within an IPHC profile.

### Task ID

Task ID	Operations
ip-services	read, write

### Examples

The following example shows how to enable non-TCP compression within an IP header compression (IPHC) profile:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
```

**non-tcp context absolute**

```
RP/0/0/CPU0:router(config-iphc-profile)# non-tcp context absolute 255
```

## pvc (serial)

To create a Frame Relay permanent virtual circuit (PVC) under a serial subinterface and enter Frame Relay virtual circuit configuration mode, use the **pvc** command in subinterface configuration mode. To remove a PVC from a subinterface, use the **no** form of this command.

```
pvc dlsi
no pvc dlsi
```

### Syntax Description

<i>dlsi</i>	DLCI number used to identify the PVC. Range is from 16 to 1007.
-------------	---

### Command Default

No PVC is defined.

### Command Modes

Subinterface configuration

### Command History

Release	Modification
Release 3.4.0	This command was introduced.

### Usage Guidelines

The **pvc** command creates a PVC and attaches it to the specified DLCI.

The Cisco IOS XR software dynamically creates rate queues as necessary to satisfy the requests of the PVC commands.

When you issue the **pvc** command in global configuration mode, the CLI prompt changes to “config-fr-vc,” indicating that you have entered the Frame Relay virtual circuit configuration submenu.

In the following sample output, the question mark (?) online help function displays all the commands available under the Frame Relay virtual circuit configuration submenu for the serial subinterface:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/3/2/0.1
RP/0/0/CPU0:router(config-subif)# pvc 20
RP/0/0/CPU0:router(config-fr-vc)# ?

  commit      Commit the configuration changes to running
  describe    Describe a command without taking real actions
  do          Run an exec command
  exit        Exit from this submenu
  no          Negate a command or set its defaults
  show        Show contents of configuration
```

Task ID	Task ID	Operations
	fr	read, write

**Examples**

The following example shows how to create a Frame Relay PVC on a serial subinterface, and enter Frame Relay virtual circuit configuration mode:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/6/0/1.1 point-to-point
RP/0/0/CPU0:router(config-if)# pvc 16
RP/0/0/CPU0:router(config-fr-vc)#
```



# refresh max-period

To configure the maximum number of compressed IP header packets exchanged on a link before IPHC context is refreshed, use the **refresh max-period** command in IPHC profile configuration mode. To return to the default context refresh settings, use the **no** form of this command.

**refresh max-period** {*max-number*| **infinite**}

**no refresh max-period** [*max-number*| **infinite**]

## Syntax Description

<i>max-number</i>	Maximum number of compressed IP header packets allowed between full headers or before the context is refreshed. Range is from 0 to 65535.
<b>infinite</b>	Allows an unlimited number of packets to be exchanged before context refresh.

## Command Default

*max-number*: 256

## Command Modes

IPHC profile configuration

## Command History

Release	Modification
Release 3.9.0	This command was introduced.

## Usage Guidelines

To enable the configured context refresh settings for RTP packets, the **refresh rtp** command must be used.



### Note

The maximum period between context refreshes can be set only within an IPHC profile.

## Task ID

Task ID	Operations
ip-services	read, write

## Examples

The following example shows how to configure the maximum number of compressed IP header packets that are exchanged on a link before the context is refreshed, in an IPHC profile.

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
```

**refresh max-period**

```
RP/0/0/CPU0:router(config-iphc-profile)# refresh max-period 50
```

# refresh rtp

To enable the configured context refresh settings for RTP packets, use the **refresh rtp** command in IPHC profile configuration mode. To disable context refresh settings for RTP packets, use the **no** form of this command.

**refresh rtp**

**no refresh rtp**

**Syntax Description** This command has no keywords or arguments.

**Command Default** By default, refresh RTP is disabled and only the first packet in the flow is sent as a 'full-header' packet.

**Command Modes** IPHC profile configuration

Command History	Release	Modification
	Release 3.9.0	This command was introduced.

## Usage Guidelines

Task ID	Task ID	Operations
	ip-services	read, write

**Examples** The following example shows how to enable the configured refresh settings for RTP packets:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
RP/0/0/CPU0:router(config-iphc-profile)# refresh rtp
```

# rtp

To enable Real Time Protocol (RTP) compression and decompression on the interface, use the **rtp** command in IPHC profile configuration mode. To remove RTP from the interface, use the **no** form of this command.

**rtp**

**no rtp**

**Syntax Description** This command has no keywords or arguments.

**Command Default** No default behavior or values

**Command Modes** IPHC profile configuration

Command History	Release	Modification
	Release 3.9.0	This command was introduced.

## Usage Guidelines

### Note

RTP can be enabled only within an IPHC profile.

You must enable RTP before attaching a profile to an interface. If you do not enable RTP first, the router will display the following message: '!!% 'iphc\_capability' detected the 'warning' condition 'IPHC Capability: RTP Compression NOT enabled in the profile!'

Task ID	Task ID	Operations
	ip-services	read, write

**Examples** The following example shows how to

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
RP/0/0/CPU0:router(config-iphc-profile)# rtp
```

# scramble

To enable payload scrambling (encryption) on a serial interface, use the **scramble** command in interface configuration mode. To disable scrambling, use the **no** form of this command.

**scramble**

**no scramble**

**Syntax Description** This command has no keywords or arguments.

**Command Default** Scrambling is disabled.

**Command Modes** Interface configuration

Command History	Release	Modification
	Release 3.3.0	This command was introduced.

**Usage Guidelines** Scrambling is used to assist clock recovery on the receiving end. Scrambling is designed to randomize the pattern of 1s and 0s carried in the physical layer frame. Randomizing the digital bits can prevent continuous, nonvariable bit patterns—in other words, long strings of all 1s or all 0s. Several physical layer protocols rely on transitions between 1s and 0s to maintain clocking.

Scrambling can prevent some bit patterns from being mistakenly interpreted as alarms by switches placed between the Data Service Units (DSUs).

The local interface configuration must match the remote interface configuration. For example, if you enable scrambling on the local port, you must also do the same on the remote port.

To verify that scrambling is configured on the interface, use the **show interfaces serial** command.

Task ID	Task ID	Operations
	hdlc	read, write

## Task ID

**Examples** In the following example, scrambling is enabled on serial interface 0/3/0/0/0:10:

```
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0/0:10
RP/0/0/CPU0:router(config-if)# serial
```

```
RP/0/0/CPU0:router(config-if-serial)# scramble
```

**Related Commands**

<b>Command</b>	<b>Description</b>
<a href="#">show controllers t3</a>	Displays information about the T3 links and hardware and software drivers for the T3 controller.

# serial

To configure the serial parameters and enter serial configuration mode, use the **serial** command in interface configuration mode. To return to the default state of the serial interface, use the **no** form of this command.

**serial**

**no serial**

**Syntax Description** This command has no keywords or arguments.

**Command Default** No default behavior or values

**Command Modes** Interface configuration

Command History	Release	Modification
	Release 3.3.0	This command was introduced.

**Usage Guidelines** Serial interfaces are automatically created for unchannelized ports; for channelized ports, serial interfaces are created when you add T1/E1 channel groups.

Task ID	Task ID	Operations
	hdlc	read, write

**Examples** The following example shows how to enter serial configuration mode:

```
RP/0/0/CPU0:router(config-if)# serial
RP/0/0/CPU0:router(config-if-serial)#
```

# show iphc idb

To display status information for an IP header compression (IPHC) interface description block (IDB), use the **show iphc idb** command in EXEC mode.

**show iphc idb** {**detail**| **interface** *type interface-path-id* [**detail**]| **location** *node-id* [**detail**]}

## Syntax Description

<b>detail</b>	Includes statistics information and internal data.
<b>interface</b>	Specifies the interface for which IPHC information is to be displayed.
<i>type</i>	Interface type. For more information, use the question mark (?) online help function.
<i>interface-path-id</i>	Physical interface or virtual interface. <b>Note</b> Use the <b>show interfaces</b> command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
<b>location</b>	Specifies the node location for which IPHC information is to be displayed.
<i>node-id</i>	Specifies the fully qualified path of a node. For more information about the syntax for the router, use the question mark (?) online help function.

## Command Default

The default (no parameters) displays information for all interfaces configured for IPHC.

## Command Modes

EXEC

## Command History

Release	Modification
Release 3.9.0	This command was introduced.

## Usage Guidelines

### Task ID

Task ID	Operations
ip-services	read



**Examples**

The following examples show how to display status information for an IP header compression (IPHC) interface description block (IDB).

```
RP/0/0/CPU0:router# show iphc idb interface Serial 0/1/0/1/26:0
```

```
Thu Jan  8 20:25:41.079 UTC
EA Status Codes:
  CFG_AS: Cfg Apply Succeed
  CFG_AF: Cfg Apply Failed
  NEG_AS: Neg Apply Succeed
  NEG_AF: Neg Apply Failed
Neg Status Code:
  NEG_I: Negotiation Init
  NEG_P: Negotiation Progress
  NEG_D: Negotiation Done
  NEG_F: Negotiation Failed

Interface_Name:  Serial0/1/0/1/26:0  Ifhandle  : 0x02008e00
EA Status       :  NEG_AS             Neg Status: NEG_D
MQC Mode        :  F                 Prof_Name  : iphcfmt
```

```
RP/0/0/CPU0:router# show iphc idb interface Serial 0/1/0/1/26:0 detail
```

```
Thu Jan  8 20:25:44.731 UTC
EA Status Codes:
  CFG_AS: Cfg Apply Succeed
  CFG_AF: Cfg Apply Failed
  NEG_AS: Neg Apply Succeed
  NEG_AF: Neg Apply Failed
Neg Status Code:
  NEG_I: Negotiation Init
  NEG_P: Negotiation Progress
  NEG_D: Negotiation Done
  NEG_F: Negotiation Failed

Interface_Name:  Serial0/1/0/1/26:0  Ifhandle  : 0x02008e00
EA Status       :  NEG_AS             Neg Status: NEG_D
MQC Mode        :  F                 Prof_Name  : iphcfmt
```

	Tcp Space	Non-Tcp Space	Max Header	Max Period	Max Time	RTP
Cfg_Option	1	60	40	256	5	T
Neg_Option	1	50	40	256	5	T

# show iphc ipv4 rtp

To display IPv4 statistics for Real Time Protocol (RTP) and User Datagram Protocol (UDP) packets sent and received on an interface, use the **show iphc ipv4 rtp** command in EXEC mode.

**show iphc ipv4 rtp interface** *type interface-path-id* [**location** *node-id*]

## Syntax Description

<i>type</i>	Interface type. For more information, use the question mark (?) online help function.
<i>interface-path-id</i>	Physical interface or virtual interface.  <b>Note</b> Use the <b>show interfaces</b> command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
<b>location</b>	(Optional) Specifies the location of the interface
<i>node-id</i>	(Optional) Node-id entered in the <i>rack/slot/module</i> notation.

## Command Default

No default behavior or values

## Command Modes

EXEC

## Command History

Release 3.9.0	This command was introduced.
---------------	------------------------------

## Usage Guidelines

### Caution

These Cisco support commands are normally reserved for use by Cisco Technical Support personnel only. If used incorrectly, there is some risk that they may cause performance or other issues that impact products, and we highly recommend that you contact Cisco Technical Support before using any of these commands.

## Task ID

Task ID	Operations
ip-services	read
cisco-support	read

**Examples**

The following example shows how to display IPv4, Real Time Protocol (RTP), User Datagram Protocol (UDP), and Non-Transmission Control Protocol (non-TCP) statistics about IP header compression (IPHC) packets sent and received on an interface:

```
RP/0/0/CPU0:router# show iphc ipv4 rtp interface Serial 0/1/0/1/26:0

Thu Jan  8 20:28:47.569 UTC
RTP/UDP/IP header compression statistics:
Interface Serial0/1/0/1/26:0
  Rcvd:   100 total, 93 compressed, 7 full header
         0 dropped, 0 status msgs
  Sent:   0 total, 0 compressed, 0 fullheader, 0 status msgs
         0 bytes saved, 0 bytes sent
         1.00 efficiency improvement factor
```

## show iphc ipv4 tcp

To display IPv4 Transport Control Protocol (TCP) statistics about IP header compression (IPHC) packets sent and received on an interface, use the **show iphc ipv4 tcp** command in EXEC mode.

**show iphc ipv4 tcp interface** *type interface-path-id* [**location** *node-id*]

### Syntax Description

<i>type</i>	Interface type. For more information, use the question mark (?) online help function.
<i>interface-path-id</i>	Physical interface or virtual interface.  <b>Note</b> Use the <b>show interfaces</b> command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
<b>location</b>	(Optional) Specifies the location of the interface
<i>node-id</i>	(Optional) Node-id entered in the <i>rack/slot/module</i> notation.

### Command Default

No default behavior or values

### Command Modes

EXEC

### Command History

Release 3.9.0	This command was introduced.
---------------	------------------------------

### Usage Guidelines

#### Caution

These Cisco support commands are normally reserved for use by Cisco Technical Support personnel only. If used incorrectly, there is some risk that they may cause performance or other issues that impact products, and we highly recommend that you contact Cisco Technical Support before using any of these commands.

### Task ID

Task ID	Operations
ip-services	read
cisco-support	read

**Examples**

The following example shows how to display IPv4, Transport Control Protocol (TCP) statistics about IP header compression (IPHC) packets sent and received on an interface:

```
RP/0/0/CPU0:router# show iphc ipv4 tcp interface Serial 0/1/0/1/26:0

Thu Jan  8 20:28:54.407 UTC
TCP/IP header compression statistics:
Interface Serial0/1/0/1/26:0
  Rcvd:   100 total, 93 compressed, 7 full header
         0 dropped, 0 status msgs
  Sent:   0 status msgs
```

# show iphc platform trace

**show iphc platform trace** [**error**| **internal** [**error**]] [**flow**] [**unique**| **wrapping**] [**hexdump**] [**last** *number-of-entries*] [**reverse**] [**stats**] [**tailf**] [**verbose**] [**file** *file-name* **original** *location* *node-id*] **location** {*node-id*| **all**| **mgmt-nodes**;}]

## Syntax Description

<b>events</b>	(Optional) Displays event platform trace information.
<b>error</b>	(Optional) Displays errors found in the trace.
<b>internal</b>	(Optional) Displays internal trace information.
<b>flow</b>	(Optional) Displays trace information for the flow.
<b>unique</b>	(Optional) Displays trace information for unique entries with counts.
<b>wrapping</b>	(Optional) Displays wrapping entries.
<b>hexdump</b>	(Optional) Displays trace information in hexadecimal format.
<b>last</b> <i>number_of_entries</i>	(Optional) Displays trace information for the last specified number of entries. The range is 1 to 4294967295.
<b>reverse</b>	(Optional) Displays trace information in reverse order (latest traces first).
<b>stats</b>	(Optional) Displays statistics information for the trace.
<b>tailf</b>	(Optional) Displays new traces as they are added.
<b>verbose</b>	(Optional) Displays internal debugging information.
<b>file</b> <i>file_name</i>	(Optional) Displays trace information for the specified file.
<b>original</b>	(Optional) Specifies the original location of file.
<b>location</b> <i>node_id</i>	(Optional) Displays trace information for the specified card location.
<b>all</b>	(Optional) Displays trace information for all nodes.
<b>mgmt-nodes</b>	(Optional) Displays trace information for all management nodes.

## Command Default

No default behavior or values

## Command Modes

EXEC

**Command History**

Release	Modification
Release 3.9.0	This command was introduced.

**Usage Guidelines**

The keywords **hexdump**, **last**, **reverse**, **stats**, **tailf**, and **verbose** may be entered in any order.

Use of the keywords **file** and **location** allows any number of desired files or locations to be entered. For more information, use the question mark (?) online help function.

**Caution**

These Cisco support commands are normally reserved for use by Cisco Technical Support personnel only. If used incorrectly, there is some risk that they may cause performance or other issues that impact products, and we highly recommend that you contact Cisco Technical Support before using any of these commands.

**Task ID**

Task ID	Operations
ip-services	read
cisco-support	read

**Examples**

The following example shows how to display platform trace information for a specified location:

```
RP/0/0/CPU0:router# show iphc platform trace location 0/3/CPU0
```

```
Thu Aug 13 14:41:13.983 UTC
5 wrapping entries (8192 possible, 0 filtered, 5 total)
Aug 6 12:27:16.840 iphc_ea/internal 0/3/CPU0 t1 Platform IPHC - Calling LC Platform init
Aug 6 12:27:17.183 iphc_ea/internal 0/3/CPU0 t1 Registering with hfa
Aug 6 12:27:19.481 iphc_ea/internal 0/3/CPU0 t1 Registering with uIDB Manager
Aug 6 12:27:19.841 iphc_ea/internal 0/3/CPU0 t1 Registering with fm
Aug 6 12:27:21.733 iphc_ea/internal 0/3/CPU0 t1 fsram_virtual_addr = 0x46000000
```

# show iphc profile

To display the configuration information of an IP header compression (IPHC) profile, use the **show iphc profile** command in EXEC mode.

**show iphc profile** *{profile-name| all}* [**detail**]

## Syntax Description

<i>profile-name</i>	Text name of the IPHC profile for which to display information.
<b>all</b>	Displays information for all profiles on the router.
<b>detail</b>	(Optional) Displays the interfaces to which the profile is attached.

## Command Default

No default behavior or values

## Command Modes

EXEC

## Command History

Release	Modification
Release 3.9.0	This command was introduced.

## Usage Guidelines

### Task ID

Task ID	Operations
ip-services	read

## Examples

The following examples show how to display information about an IPHC profile:

```
RP/0/0/CPU0:router# show iphc profile ietf-test1
Tue Aug 10 06:53:19.711 PDT
IPHC Profile: ietf-test1
Type: IETF
Compressing : TCP NON-TCP (RTP)
Context     : TCP fixed at 1 NON-TCP fixed at 10
Refresh     : NON-TCP every 60 seconds or 100 packets
Feedback    : ON
Max_Header  : 40
Refresh RTP : OFF
```

```
RP/0/0/CPU0:router# show iphc profile Profile_1 detail
Thu Jan  8 20:22:24.276 UTC
```



```

IPHC Profile: Profile_1
Type: IPHC
  Compressing : TCP NON-TCP (RTP)
  Context     : TCP fixed at 1 NON-TCP fixed at 60
  Refresh     : NON-TCP every 5 seconds or 256 packets
  Feedback    : ON
  Max_Header  : 40

*** No of Intf 1 ****
Serial0_4_3_1_1:0

RP/0/0/CPU0:router# show iphc profile all
Thu Mar 12 11:05:35.987 UTC
IPHC Profiles : 3
IPHC Profile: p1
Type: IETF
  Compressing : TCP NON-TCP (RTP)
  Context     : TCP fixed at 1 NON-TCP fixed at 16
  Refresh     : NON-TCP every 5 seconds or 256 packets
  Feedback    : ON
  Max_Header  : 40

IPHC Profile: p2
Type: IETF
  Compressing : TCP NON-TCP (RTP)
  Context     : TCP fixed at 1 NON-TCP fixed at 16
  Refresh     : NON-TCP every 5 seconds or 256 packets
  Feedback    : ON
  Max_Header  : 40

IPHC Profile: test
Type: IETF
  Compressing : TCP NON-TCP (RTP)
  Context     : TCP fixed at 1 NON-TCP fixed at 16
  Refresh     : NON-TCP every 5 seconds or 256 packets
  Feedback    : ON
  Max_Header  : 40

RP/0/0/CPU0:router# show iphc profile all detail
Thu Mar 12 11:06:26.902 UTC
IPHC Profiles : 3
IPHC Profile: p1
Type: IETF
  Compressing : TCP NON-TCP (RTP)
  Context     : TCP fixed at 1 NON-TCP fixed at 16
  Refresh     : NON-TCP every 5 seconds or 256 packets
  Feedback    : ON
  Max_Header  : 40

*** No of Intf 1 ****
Serial0_4_3_1_1:0

IPHC Profile: p2
Type: IETF
  Compressing : TCP NON-TCP (RTP)
  Context     : TCP fixed at 1 NON-TCP fixed at 16
  Refresh     : NON-TCP every 5 seconds or 256 packets
  Feedback    : ON
  Max_Header  : 40

*** No of Intf 2 ****
Serial0_4_3_1_2:0
Serial0_4_3_1_8:0

IPHC Profile: test
Type: IETF
  Compressing : TCP NON-TCP (RTP)
  Context     : TCP fixed at 1 NON-TCP fixed at 16
  Refresh     : NON-TCP every 5 seconds or 256 packets
  Feedback    : ON
  Max_Header  : 40

```

```
show iphc profile
```

```
*** No of Intf 0 ***
```

# show iphc trace all

To display trace results for all IP header compression (IPHC) configurations on the router, use the **show iphc trace all** command in EXEC mode.

```
show iphc trace all [unique|wrapping] [hexdump] [last number-of-entries] [reverse] [stats] [tailf] [verbose]
[file file-name original location node-id] location {node-id} all|mgmt-nodes}]
```

## Syntax Description

<b>unique</b>	(Optional) Displays trace information for unique entries with counts.
<b>wrapping</b>	(Optional) Displays wrapping entries.
<b>hexdump</b>	(Optional) Displays trace information in hexadecimal format.
<b>last number_of_entries</b>	(Optional) Displays trace information for the last specified number of entries. The range is from 1 to 4294967295.
<b>reverse</b>	(Optional) Displays trace information in reverse order (latest traces first).
<b>stats</b>	(Optional) Displays statistics information for the trace.
<b>tailf</b>	(Optional) Displays new traces as they are added.
<b>verbose</b>	(Optional) Displays internal debugging information.
<b>file file_name</b>	(Optional) Displays trace information for the specified file.
<b>original</b>	(Optional) Specifies the original location of file.
<b>location node_id</b>	(Optional) Displays trace information for the specified card location.
<b>all</b>	(Optional) Displays trace information for all nodes.
<b>mgmt-nodes</b>	(Optional) Displays trace information for all management nodes.

## Command Default

No default behavior or values

## Command Modes

EXEC

## Command History

Release 3.9.0	This command was introduced.
---------------	------------------------------

**Usage Guidelines**

The keywords **hexdump**, **last**, **reverse**, **stats**, **tailf**, and **verbose** may be entered in any order. The keywords **unique** or **wrapping** may only be entered as the first keyword in the command. The keywords **file** or **location** may only be entered as the last keyword in the command.

Use of the keywords **file** and **location** allows any number of desired files or locations to be entered. For more information, use the question mark (?) online help function.

**Caution**

These Cisco support commands are normally reserved for use by Cisco Technical Support personnel only. If used incorrectly, there is some risk that they may cause performance or other issues that impact products, and we highly recommend that you contact Cisco Technical Support before using any of these commands.

**Task ID**

Task ID	Operations
ip-services	read
cisco-support	read

**Examples**

The following example shows how to display IPHC trace information:

```
RP/0/0/CPU0:router# show iphc trace all

Wed Jul 22 21:48:07.339 DST
20 wrapping entries (3072 possible, 0 filtered, 20 total)
Jul 22 03:31:39.770 iphc/profilemgr/int 0/5/CPU0 t1 : Event Mgr Create Successful
Jul 22 03:31:39.799 iphc/profilemgr/int 0/5/CPU0 t1 : Sysmgr Init Successful
Jul 22 03:31:39.894 iphc/profilemgr/int 0/5/CPU0 t1 : Ens Init Successful
Jul 22 03:31:39.910 iphc/profilemgr/int 0/5/CPU0 t1 : Sysdb Init Successful
Jul 22 03:31:39.911 iphc/profilemgr/int 0/5/CPU0 t1 : Stats thread Init Successful
Jul 22 03:31:39.942 iphc/profilemgr/int 0/5/CPU0 t1 : Cfg thread Init Successful
Jul 22 03:31:39.951 iphc/profilemgr/int 0/5/CPU0 t1 : Registered verifier call7
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Profile: *** Pr*
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Option (format)
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Option (tcp_com)
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Option (tcp_con)
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Option (non_tcp)
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Option (rtp) in)
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Option (max-per)
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Scanning Option (non_tcp)
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : +++++ Profile Verificati+
Jul 22 03:31:39.952 iphc/profilemgr/eve 0/5/CPU0 t1 : Verify Profile (Profile_n
Jul 22 03:31:39.959 iphc/profilemgr/eve 0/5/CPU0 t1 : +++++ Profile Verificati+
Jul 22 03:31:39.981 iphc/profilemgr/int 0/5/CPU0 t1 : Registered applier calle7
Jul 22 03:31:39.999 iphc/profilemgr/eve 0/5/CPU0 t1 : Registered Profile (Profy
```

# show tech-support iphc

**show tech-support iphc** [*file*| *interface type interface-path-id* [*location node-id*| *rack rack\_name*]] *location node-id*| *rack rack\_name*] **file** *location file\_name* [**background**] [**compressed**| **uncompressed**]

## Syntax Description

<b>background</b>	(Optional) Runs this command in the background.
<b>compressed</b>	(Optional) Compresses the output.
<b>uncompressed</b>	(Optional) Does not compress the output.

## Command Default

No default behavior or values

## Command Modes

EXEC

## Command History

Release	Modification
Release 3.9.0	This command was introduced.

## Usage Guidelines

### Task ID

Task ID	Operations
basic-services	read
cisco-support	read
ipv4	read

## Examples

The following example shows how to collect and display IP header compression (IPHC) data:

```
RP/0/0/CPU0:router# show tech-support iphc

Mon Oct 12 20:30:58.660 DST
++ Show tech start time: 2009-Oct-12.203059.DST ++
Mon Oct 12 20:31:05 DST 2009 Waiting for gathering to complete
.....
Mon Oct 12 20:31:46 DST 2009 Compressing show tech output
Show tech output available at 0/5/CPU0 : disk0:/showtech/showtech-iphc-2009-Octz
++ Show tech end time: 2009-Oct-12.203148.DST ++
```

## tcp compression

To enable TCP compression in an IP header compression (IPHC) profile, use the **tcp compression** command in IPHC profile configuration mode. To disable TCP compression in the profile, use the **no** form of this command.

**tcp compression**

**no tcp**

**Syntax Description** This command has no keywords or arguments.

**Command Default** No default behavior or values

**Command Modes** IPHC profile configuration

**Command History**

Release 3.9.0	This command was introduced.
---------------	------------------------------

**Usage Guidelines** Where the IPHC profile used by a router is configured using this command, the router will negotiate TCP compression with its peer router and decompress any compressed TCP packets sent by its peer. TCP packets transmitted to the peer are transmitted uncompressed.



**Note** TCP compression can be enabled only within an IPHC profile. TCP compression does not work unless it is enabled under a profile.

Task ID	Task ID	Operations
	ip-services	read, write

**Examples** The following example shows how to enable TCP compression within an IP header compression (IPHC) profile:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
RP/0/0/CPU0:router(config-iphc-profile)# tcp compression
RP/0/0/CPU0:router(config-iphc-profile)#
```

## tcp context absolute

To configure the maximum number of TCP contexts that are allowed for IPHC under a profile, use the **tcp context absolute** command in IPHC profile configuration mode. To remove the TCP context from the profile, use the **no** form of this command.

**tcp context absolute** *number-of-contexts*

**no tcp context** [**absolute**] [ *number-of-contexts* ]

### Syntax Description

*number-of-contexts*

Numeric value that specifies the maximum number of TCP contexts allowed for IPHC under this profile. The range is from 0 to 255.

### Command Default

If *number-of-contexts* is not specified, and only TCP compression is enabled, the default *number-of-contexts* is 1.

### Command Modes

IPHC profile configuration

### Command History

Release 3.9.0

This command was introduced.

### Usage Guidelines

The maximum allowed number of tcp contexts on a Line Card, across all IPHC profiles and interfaces, is 255.



#### Note

TCP context can be set only within an IPHC profile.

### Task ID

#### Task ID

#### Operations

ip-services

read, write

### Examples

The following example shows how to enable TCP compression within an IP header compression (IPHC) profile:

```
RP/0/0/CPU0:router(config)# config
RP/0/0/CPU0:router(config)# iphc profile Profile_1 type iphc
RP/0/0/CPU0:router(config-iphc-profile)# tcp context absolute 255
```

# transmit-delay (serial)

**transmit-delay** *microseconds*

**no transmit-delay** *microseconds*

## Command Default

## Command Modes

## Command History

Release 3.2	This command was introduced.
-------------	------------------------------

## Usage Guidelines

## Task ID

Task ID	Operations
hdlc	read, write

## Examples

In the following example, a delay of 2 microseconds is specified on serial interface 0/3/0/0/0:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0/0:0
RP/0/0/CPU0:router(config-if)# serial
RP/0/0/CPU0:router(config-if-serial)# transmit-delay 2
```

In the following example, the transmit delay on serial interface 0/3/0/0/0 is disabled:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface serial 0/3/0/0/0:0
RP/0/0/CPU0:router(config-if)# serial
RP/0/0/CPU0:router(config-if-serial)# no transmit-delay
```

## Related Commands

Command	Description
<a href="#">show interfaces</a>	Displays statistics for all interfaces configured on the router or for a specific node.