# Configuring the Serial Interface

This chapter describes configuring serial interface management.

## Configuring the Serial Interface

This chapter describes configuring serial interface management.

The IR807 supports asynchronous serial interface protocols used for Raw Socket, SCADA Protocol Translation or reverse Telnet. It has two serial interfaces, designated async 0 and async 1.

### Specifying an Asynchronous Serial Interface

To specify an asynchronous serial interface and enter interface configuration mode, use one of the following commands in global configuration mode.

| Command or Action | Purpose |
|---|---|
| Router(config)# interface async 0 | Enters interface configuration mode. |

### Specifying Asynchronous Serial Encapsulation

By default, asynchronous serial lines use the SCADA serial encapsulation method, which provides the communication between the Control Center and the RTU. The asynchronous serial interfaces support the following serial encapsulation methods:

- PPP
- Raw-TCP
- Raw-UDP
- SCADA
- SLIP

| Command or Action | Purpose |
|---|---|
| `Router(config-if)# encapsulation {scada | ppp | raw-tcp | raw-udp | slip}` | Configures asynchronous serial encapsulation. |

Encapsulation methods are set according to the type of protocol or application you configure in the Cisco IOS software.

The remaining encapsulation methods are defined in their respective books and chapters describing the protocols or applications.

# Configuring the Serial Port

To configure the serial port perform the steps in the following example:

```
IR807#sh run int async 0
Building configuration...
Current configuration : 62 bytes
!
interface Async0
no ip address
encapsulation raw-tcp
end
IR807#sh run | sec line
line con 0
stopbits 1
line 1
exec-timeout 0 0
raw-socket tcp server 502
no exec
transport preferred none
transport input all
transport output none
stopbits 1
```

# Configuring Raw Socket

On the IR807, async 0 is associated with Line 5, and async 1 is associated with Line 4. In the following example, raw-socket (tcp) is configured on async 0:

```
IR807# interface Async0
    no ip address
    encapsulation raw-tcp
```

If the IR807 is configured as a raw-socket TCP client:

```
line 5
    raw-socket tcp client 10.0.0.254 5000
    stopbits 1
```

If IR807 is configured as a raw-socket TCP server:

```
line 5
   raw-socket tcp server 10000
   stopbits 1
```

**Note:** In the example above, 10000 is for raw-socket tcp server configuration. Avoid using the ports from 1 to 1023 for the raw-socket TCP server's local port.

# Configuring Common Raw Socket Line Options

You can configure options common to all connections on a line. The common options apply to both TCP and UDP.

**Note** The corresponding port used for configuration mapping should be changed on the Head end application server as well.

.

**Procedure**

|  | Command or Action | Purpose |
|---|---|---|
| Step 1 | **configure terminal** | Enters global configuration mode. |
| Step 2 | **interface async** *number* | Enters interface command mode for the serial slot/port. |
| Step 3 | **no ip address** | Disables IP processing on the interface. |
| Step 4 | Do one of the following:<br>• **encapsulation raw-tcp**<br>• **encapsulation raw-udp** | Enables Raw Socket TCP encapsulation or UDP encapsulation for the serial port. |
| Step 5 | **raw-socket packet-timer** *timeout* | Specifies the maximum time in milliseconds the router waits to receive the next character in a stream. If a character is not received by the time the packet-timer expires, the accumulated data is packetized and forwarded to the Raw Socket peer.<br><br>Options are 25ms or 50ms. |
| Step 6 | **raw-socket packet-length** *length* | Specifies the packet size that triggers the IR807 to transmit the data to the peer. When the IR807 accumulates this much data in its buffer, it packetizes the data and forwards it to the Raw Socket peer.<br><br>*length*— 2 to 1400 bytes.<br><br>By default, the packet-length trigger is disabled. |
| Step 7 | **raw-socket tcp server** *port* | Starts the Raw Socket Transport TCP server for an asynchronous line interface. In Raw Socket server mode, the IR807 listens for incoming connection requests from Raw Socket clients.<br><br>*port*–Port number the server listens on. |

| | Command or Action | Purpose |
|---|---|---|
| | | *ip_address*–(Optional) Local IP address on which the server listens for connection requests. |
| Step 8 | **raw-socket tcp keepalive** *interval* | Sets the Raw Socket Transport TCP session keepalive interval for the asynchronous line interface. The router sends keepalive messages based on the configured interval. You may need to configure this interval, for example, when sending raw TCP traffic over a cellular interface. Currently configured keepalive interval in seconds. Range is 1-864000 seconds. |
| Step 9 | **raw-socket tcp idle-timeout** *session_timeout* | Sets the Raw Socket Transport TCP session timeout for the asynchronous line interface. If no data is transferred between the client and server over this interval, then the TCP session is closed. The client then automatically attempts to reestablish the TCP session with the server |
| Step 10 | **raw-socket mode best-effort** | Enable best-effort mode for the serial line. When this mode is enabled, older packets are dropped from the head of the queue when the queue is full. By default, best-effort mode is off. |
| Step 11 | **exit** | Exits global configuration mode. |

## Example Configuration

```
!
interface Async1
no ip address
encapsulation raw-tcp
!
line 4
raw-socket tcp keepalive 300
raw-socket tcp server 8004
raw-socket packet-timer 50
raw-socket tcp idle-timeout 60
transport input all
transport output all
!
```

# Configuring Raw Socket TCP

After enabling Raw Socket TCP encapsulation, you configure the TCP server and/or clients.

# Configuring the Raw Socket TCP Server

### Procedure

|  | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal** | Enters configuration mode. |
| **Step 2** | **line slot/port** | Enters line command mode for the serial slot/port. |
| **Step 3** | **raw-socket tcp server** *port* [*ip_address*] | Starts the Raw Socket Transport TCP server for an asynchronous line interface. In Raw Socket server mode, the IR807 listens for incoming connection requests from Raw Socket clients. |
|  |  | *port*–Port number the server listens on. |
|  |  | *ip_address*–(Optional) Local IP address on which the server listens for connection requests. |
| **Step 4** | **raw-socket tcp idle-timeout** *session_timeout* | Sets the Raw Socket Transport TCP session timeout for the asynchronous line interface. If no data is transferred between the client and server over this interval, then the TCP session closes. The client then automatically attempts to reestablish the TCP session with the server. |
|  |  | This timeout setting applies to all Raw Socket Transport TCP sessions under this particular line. |
|  |  | *session_timeout*–Currently configured session idle timeout in minutes. The default is 5 minutes. |

### What to do next

To remove a Raw Socket TCP server, use the **no raw-socket tcp server** command.

### EXAMPLE

This example shows how to configure a Raw Socket TCP server for an asynchronous serial line. The TCP server listens for TCP client connection requests on local port 4000 and local IP address 10.0.0.1. If no data is exchanged between the Raw Socket TCP server and one of the TCP clients for 10 minutes, then the TCP session closes, and the Raw Socket client attempts to reestablish the session with the Raw Socket server.

```
router# configure terminal
router(config)# line 1/1
router(config-line)# raw-socket tcp server 4000 10.0.0.1
router(config-line)# raw-socket tcp idle-timeout 10
router(config-line)# exit
router(config)#
```

# Configuring the Raw Socket TCP Client

### Procedure

|  | Command or Action | Purpose |
|---|---|---|
| Step 1 | **configure terminal** | Enters configuration mode. |
| Step 2 | **line slot/port** | Enters line command mode for the serial slot/port. |
| Step 3 | **raw-socket tcp client** *dest_ip_address dest_port* *[local_ip_address] [local_port]* | Specifies settings for Raw Socket Transport TCP client sessions. |
|  |  | *dest_ip_address*–Destination IP address of the remote Raw Socket server. |
|  |  | *dest_port*–Destination port number to use for the TCP connection to the remote server. |
|  |  | *local_ip_address*–(Optional) Local IP address that the client can also bind to. |
|  |  | *local_port*–(Optional) Local port number that the client can also bind to. |

#### What to do next

To remove a Raw Socket TCP client, use the **no raw-socket tcp client** command.

## EXAMPLE

This example shows how to configure a Raw Socket TCP client for an asynchronous serial line. The IR807 (router), serving as a Raw Socket client, initiates TCP sessions with a Raw Socket server and forwards packetized serial data to it. The router collects streams of serial data in its buffer; when it accumulates 827 bytes in its buffer, the router packetizes the data and forwards it to the Raw Socket server. If the router and the Raw Socket server do not exchange any data for 10 minutes, then the TCP session with the Raw Socket server closes, and the router attempts to reestablish the session with the Raw Socket server.

```
router# configure terminal
router(config)# line 1/1
router(config-line)# raw-socket tcp client 10.0.0.1 4000
router(config-line)# raw-socket packet-length 827
router(config-line)# raw-socket tcp idle-timeout 10
router(config-line)# exit
router(config)#
```

# Configuring a Raw Socket UDP Peer-to-Peer Connection

After enabling Raw Socket UDP encapsulation and the common line options, you configure the Raw Socket UDP peer-to-peer connection. The local port on one end of the connection should be the destination port on the other end.

**Procedure**

|       | Command or Action | Purpose |
|-------|-------------------|---------|
| Step 1 | **configure terminal** | Enters configuration mode. |
| Step 2 | **line slot/port** | Enters line command mode for the serial slot/port. |
| Step 3 | **raw-socket udp connection** *dest_ip_address dest_port local_port [local_ip_address]* | Specifies settings for Raw Socket Transport UDP connections.<br><br>*dest_ip_address*–Destination IP address to use for the UDP connection.<br><br>*dest_port*–Destination port number to use for the UDP connection.<br><br>*local_port*–Local port number for the UDP connection.<br><br>*local_ip_address*–(Optional) Local IP address for the UDP connection. |

**What to do next**

To remove a Raw Socket UDP connection, use the **no raw-socket udp connection** command.

## EXAMPLE

This example shows how to configure a Raw Socket UDP connection between router A (local IP address 192.168.0.8) and router B (local IP address 192.168.0.2).

### Router A

```
router# configure terminal
router(config)# line 1/1
router(config-line)# raw-socket udp connection 192.168.0.2 5000 7000
router(config-line)# raw-socket mode best-effort
router(config-line)# exit
router(config)#
```

### Router B

```
router# configure terminal
router(config)# line 1/1
router(config-line)# raw-socket udp connection 192.168.0.8 7000 5000
router(config-line)# raw-socket mode best-effort
router(config-line)# exit
router(config)#
```

# Reverse Telnet

Reverse telnet allows you to telnet to a device, and then connect from that device to the console of another device. This is useful for devices that do not have remote access built into them. This section provides an overview of what is required for setup.

> **Note** This setup requires a straight through cable that goes from the console port of the console-only device to the AUX port on your router.

Follow these steps to configure the IR807:

### Configure the AUX Port

```
router#configure terminal
router(config)#line aux 0
router(config-line)#modem InOut
router(config-line)#transport input all
router(config-line)#speed 19200
router(config-line)#exit
```

### Create a Loopback Address

```
router#configure terminal
router(config)#int loopback 0
router(config-if)#ip address 10.0.0.1 255.0.0.0
router(config-if)#no shut
router(config-if)#exit
```

### Determine Which Line is the AUX Port

Outside of configuration mode (hit CTRL-Z to get out), enter the command **show line**

```
router#sh line
   Tty Typ     Tx/Rx     A Modem  Roty AccO AccI  Uses  Noise  Overruns  Int
     0 CTY             - -       -    -    -     0     0    0/0      -
   225 AUX  19200/19200 - inout   -    -    -     0     0    0/0      -
*  226 VTY             - -       -    -    -    10     0    0/0      -
   227 VTY             - -       -    -    -     0     0    0/0      -
   228 VTY             - -       -    -    -     0     0    0/0      -
   229 VTY             - -       -    -    -     0     0    0/0      -
   230 VTY             - -       -    -    -     0     0    0/0      -
```