



Configuring Asynchronous Serial Traffic over UDP

This chapter describes how to communicate with a modem using the Asynchronous Serial Traffic over UDP feature in the following main sections:

- [UDPTN Overview](#)
- [How to Configure Asynchronous Serial Traffic over UDP](#)

See the “[Configuration Examples for UDPTN](#)” section for configuration examples.

To identify the hardware platform or software image information associated with a feature, use the Feature Navigator on Cisco.com to search for information about the feature or refer to the software release notes for a specific release. For more information, see the “Identifying Supported Platforms” section in the “Using Cisco IOS Software” chapter.

For a complete description of the UDP commands mentioned in this chapter, refer to the *Cisco IOS Dial Technologies Command Reference*, Release 12.2. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

UDPTN Overview

The Asynchronous Serial Traffic over UDP feature provides the ability to encapsulate asynchronous data into User Datagram Protocol (UDP) packets and then unreliably send this data without needing to establish a connection with a receiving device. This process is referred to as UDP Telnet (UDPTN), although it does not—and cannot—use the Telnet protocol. UDPTN is similar to Telnet in that both are used to send data, but UDPTN is unique in that it does not require that a connection be established with a receiving device. You load the data that you want to send through an asynchronous port, and then send it, optionally, as a multicast or a broadcast. The receiving device(s) can then receive the data whenever it wants. If the receiver ends reception, the transmission is unaffected.

The Asynchronous Serial Traffic over UDP feature provides a low-bandwidth, low-maintenance method to unreliably deliver data. This delivery is similar to a radio broadcast: It does not require that you establish a connection to a destination; rather, it sends the data to whatever device wants to receive it. The receivers are free to begin or end their reception without interrupting the transmission.

It is a low-bandwidth solution for delivering streaming information for which lost packets are not critical. Such applications include stock quotes, news wires, console monitoring, and multiuser chat features.

This feature is particularly useful for broadcast, multicast, and unstable point-to-point connections. This feature may not work as expected when there are multiple users on the same port number in a nonmulticast environment. The same port must be used for both receiving and sending.

How to Configure Asynchronous Serial Traffic over UDP

To configure the Asynchronous Serial Traffic over UDP feature, perform the tasks described in the following sections:

- [Preparing to Configure Asynchronous Serial Traffic over UDP](#) (Required)
- [Configuring a Line for UDPTN](#) (Required)
- [Enabling UDPTN](#) (Required)
- [Verifying UDPTN Traffic](#) (Optional but Recommended)

See the “[Configuration Examples for UDPTN](#)” section at the end of this chapter for multicast, broadcast, and point-to-point UDPTN configuration examples.

Preparing to Configure Asynchronous Serial Traffic over UDP

When configuring the Asynchronous Serial Traffic over UDP feature for multicast transmission, you must configure IP multicast routing for the entire network that will receive or propagate the multicasts. When configuring the feature for broadcast transmission, you must configure broadcast flooding on the routers between network segments. Refer to the “[Configuring IP Multicast Routing](#)” chapter of this guide for information on how to configure IP multicast routing. See the section “[Configuring Broadcast Packet Handling](#)” in the *Cisco IOS IP Configuration Guide* for information on how to configure broadcast flooding.

Configuring a Line for UDPTN

To configure the line that will be used to send or receive UDP packets, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# line <i>line-number</i>	Enters line configuration mode for the line number specified.
Step 2	Router(config-line)# transport output udptn	Enables the line to transport UDP packets.
Step 3	Router(config-line)# dispatch-timeout 1000	Sends packets every 1000 milliseconds.
Step 4	Router(config-line)# dispatch-character 13	Sends packets after every new line.
Step 5	Router(config-line)# no session-timeout	Disables timeout connection closing.

Enabling UDPTN

There are two methods of enabling UDPTN. You can manually enable UDPTN when you want to begin transmission or reception, or you can configure the router to automatically enable UDPTN when a connection is made to the line.

To manually enable UDPTN and begin UDPTN transmission or reception, use the following command in EXEC mode:

Command	Purpose
Router# udptn <i>ip-address</i> [<i>port</i>] [/ transmit] [/ receive]	Enables UDPTN to the specified IP address (optionally, using the specified port). Use the /transmit or /receive keyword if the router will only be sending or receiving UDPTN.

To automatically enable UDPTN when a connection is made to the line, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# line <i>line-number</i>	Enters line configuration mode for the line number specified.
Step 2	Router(config-line)# autocommand udptn <i>ip-address</i> [<i>port</i>] [/ transmit] [/ receive]	Enables UDPTN automatically when a connection is made to the line (optionally, using the specified port). Use the /transmit or /receive keyword if the router will only be sending or receiving UDPTN.

Verifying UDPTN Traffic

To verify that UDPTN is enabled correctly, perform the following steps:

- Step 1 Enable UDPTN debugging by using the **debug udptn** EXEC command.
- Step 2 Enable UDPTN by using the **udptn ip-address** EXEC command, and then observe the debug output.

The following debug output shows a UDPTN session being successfully established and then disconnected.

```
Router# debug udptn
Router# udptn 172.16.1.1
Trying 172.16.1.1 ... Open

*Mar 1 00:10:15.191:udptn0:adding multicast group.
*Mar 1 00:10:15.195:udptn0:open to 172.16.1.1:57 Loopback0jjaassdd
*Mar 1 00:10:18.083:udptn0:output packet w 1 bytes
*Mar 1 00:10:18.087:udptn0:Input packet w 1 bytes
Router# disconnect
Closing connection to 172.16.1.1 [confirm] y
Router#
```

- Step 3** While the **udptn** command is enabled, enter the **show ip socket** command to verify that the socket being used for UDPTN opened correctly.

```
Router# show ip socket
Proto  Remote      Port      Local      Port  In  Out  Stat  TTY  OutputIF
17     --listen--          172.21.14.90  67  0   0   89   0
17     0.0.0.0      520      172.21.14.90  520  0   0   1    0
17     1.1.1.2      57       1.1.1.1      57   0   0   48   0
17     224.1.1.1    57       1.2.2.2      57   0   0   48   0 Loopback0
```

Configuration Examples for UDPTN

This section provides the following UDPTN configuration examples:

- [Multicast UDPTN Example](#)
- [Broadcast UDPTN Example](#)
- [Point-to-Point UDPTN Example](#)

Multicast UDPTN Example

These configurations are for multicast UDPTN. The router that is multicasting does not require a multicast configuration—it simply sends to the multicast IP address.

Router That Is Multicasting

```
ip multicast-routing
interface ethernet 0
 ip address 10.1.1.1 255.255.255.0
 ip pim dense-mode
!
line 5
 no session-timeout
 transport output udptn
 dispatch-timeout 10000
 dispatch-character 13
 modem in
 autocommand udptn 172.1.1.1 /transmit
```

Receiving Routers

```
ip multicast-routing
interface ethernet 0
 ip address 10.99.98.97 255.255.255.192
 ip pim dense-mode
!
line 0 16
 transport output udptn telnet lat rlogin
 autocommand udptn 172.1.1.1 /receive
```

Broadcast UDPTN Example

These configurations are for broadcast UDPTN. This is the simplest method to send to multiple receivers. The broadcasting router sends to the broadcast IP address, and any router that wants to receive the transmission simply connects to the broadcast IP address by using the **udptn** command.

Router That Is Broadcasting

```
interface ethernet 0
 ip address 10.1.1.1 255.255.255.0
 !
 line 5
  no session-timeout
  transport output udptn
  dispatch-timeout 10000
  dispatch-character 13
  modem in
  autocommand udptn 255.255.255.255 /transmit
```

Receiving Routers

```
interface ethernet 0
 ip address 10.99.98.97 255.255.255.192
 !
 line 0 16
  transport output udptn telnet lat rlogin
  autocommand udptn 255.255.255.255 /receive
```

Point-to-Point UDPTN Example

These configurations are for two routers in mobile, unstable environments that wish to establish a bidirectional asynchronous tunnel. Because there is no way to ensure that both routers will be up and running when one of the routers wants to establish a tunnel, they cannot use connection-dependent protocols like Telnet or local area transport (LAT). They instead use the following UDPTN configurations. Each router is configured to send to and receive from the IP address of the other. Because both routers will be sending and receiving, they do not use the **/transmit** or **/receive** keywords with the **udptn** command.

Router A

```
interface ethernet 0
 ip address 10.54.46.1 255.255.255.192
 !
 line 5
  no session-timeout
  transport output udptn
  dispatch-timeout 10000
  dispatch-character 13
  modem in
  autocommand udptn 10.54.46.2
```

Router B

```
interface ethernet 0
 ip address 10.54.46.2 255.255.255.192
!
line 10
 no session-timeout
 transport output udptn
 dispatch-timeout 10000
 dispatch-character 13
 modem in
 autocommand udptn 10.54.46.1
```