

X.25 Load Balancing

Feature Summary

As the number of users accessing the same host has grown, competition for these application resources has become a problem. Internet service providers (ISPs) have had to increase the number of users they could support by increasing the number of X.25 lines to the host.

In order to support a large number of virtual circuits (VCs) to a particular destination, configuration of more than one serial interface to that destination was needed. When a serial interface is configured to support X.25, there is a fixed number of VCs available for use.

However, the current X.25 allocation method for VCs across multiple serial lines fills one serial line to its VC capacity before utilizing the second line at all. As a result, the first serial line is frequently carrying its maximum data traffic before it runs out of VCs.

Now, using a facility called “hunt-group” (the method for X.25 load balancing), a switch is able to view a pool of X.25 lines going to the same host as one address and assign VCs on an “idle logical channel” basis. With this feature, X.25 calls can be load balanced among all configured outgoing interfaces to fully use and balance all managed lines.

Benefits

X.25 load balancing provides the following benefits:

- The choice of two load-balancing distribution methods (rotary or vc-count) utilizing multiple serial lines.
- Reconfiguration of hunt groups does not affect functionality.
- Improved performance of serial lines.

List of Terms

packet assembler/disassembler (PAD)—Device used to connect simple devices (like character mode terminals) that do not support the full functionality of a particular protocol to a network.

virtual circuit (VC)—A logical association between two physically separated DTEs.

X.25—Interface between a DTE and a DCE for packet-mode operation on a public data network (PDN).

Restrictions

The following restrictions apply to X.25 load balancing:

- Only one distribution method can be selected for each hunt group, although one interface can participate in one or more hunt groups.
- Distribution methods are limited to “rotary” and “vc-count” only.
- XOT cannot be used in hunt groups configured with the vc-count distribution method. XOT does not limit the number of calls that can be sent to a particular destination, so the method of selecting the hunt group member with the largest number of available VCs will not work. XOT can be used in hunt groups configured with the rotary distribution method.

Platforms

This feature is supported on these platforms:

- Cisco 1600 series
- Cisco 2500 series
- Cisco 2600 series
- Cisco 3600 series
- Cisco 4000 series (Cisco 4000, 4000-M, 4500, 4500-M, 4700, 4700-M)
- Cisco 7200 series
- Cisco 7500 series

Prerequisites

Before enabling X.25 load balancing you must:

- Activate the X.25 routing software
- Configure the interfaces participating in the hunt group for X.25 encapsulation.

See the sections “Configuration Tasks” and “Configuration Examples” for further details. For more information about configuring X.25, refer to the chapter “Configuring X.25 and LAPB” in the *Wide-Area Networking Configuration Guide* for Cisco IOS Release 12.0.

Supported MIBs and RFCs

None.

Configuration Tasks

To configure X.25 load balancing, use the following commands beginning in global configuration mode:

Step	Command	Purpose
1	x25 routing	Activates X.25 routing software.

Step	Command	Purpose
2	encapsulation x25	Specifies X.25 encapsulation on each interface to be included in the hunt group.
3	x25 hunt-group name {rotary vc-count}	Creates the hunt group. Note XOT cannot be used in hunt groups configured with the vc-count distribution method.
4	x25 route [#position][selection][modification] disposition [xot-keepalive]	Adds the hunt group to the routing table.

Configuration Examples

The following example enables X.25 encapsulation on relevant serial interfaces and configures serial interfaces 1 and 2 to participate in X.25 hunt group “HG1,” and serial interfaces 0 and 3 to participate in X.25 hunt group “HG2.” Serial interfaces 1 and 2, and XOT IP addresses 172.17.125.54 and 172.17.125.34 are then associated with hunt group “HG1” (with rotary distribution assigned); and serial interfaces 0 and 3 “HG2” (with vc-count distribution assigned). These hunt groups are then added to the routing table, where X.25 route 1111 will use “HG1,” and X.25 route 1112 will use “HG2.”

```

Router(config)# x25 routing
Router(config)# interface serial 0
Router(config-if)# encapsulation x25
Router(config-if)# interface serial 1
Router(config-if)# encapsulation x25
Router(config-if)# interface serial 2
Router(config-if)# encapsulation x25
Router(config-if)# interface serial 3
Router(config-if)# encapsulation x25
Router(config)# x25 hunt-group HG1 rotary
Router(config-x25-huntgro)# interface serial 1
Router(config-x25-huntgro)# interface serial 2
Router(config-x25-huntgro)# xot 172.17.125.54
Router(config-x25-huntgro)# xot 172.17.125.34
Router(config-x25-huntgro)# exit
Router(config)# x25 hunt-group HG2 vc-count
Router(config-x25-huntgro)# interface serial0
Router(config-x25-huntgro)# interface serial3
Router(config-x25-huntgro)# exit
Router(config)# x25 route 1111 hunt-group HG1
Router(config)# x25 route 1112 hunt-group HG2

```

Command Reference

This section documents modifications to the **x25 route** command and describes the new commands **x25 hunt-group** and **show x25 hunt-group**. All other commands used with this feature are documented in the Cisco IOS Release 12.0 command references.

- **show x25 hunt-group**
- **x25 hunt-group**
- **x25 route**

show x25 hunt-group

To display hunt groups and view detailed interface statistics and distribution methods, use the **show x25 hunt-group EXEC** command.

```
show x25 hunt-group [name]
```

Syntax Description

name (Optional) Displays the specific hunt group named.

Command Mode

EXEC

Usage Guidelines

This command first appeared in the Cisco IOS Release 12.0(3)T.

Sample Display

The following is sample output from the **show x25 hunt-group** command:

```
Router# show x25 hunt-group
  ID      Type      Target          uses      status      traffic(out/in)
=====
HG1      rotary   Serial1         2         last used  17080 /10100
          Serial2         2         last used  15640 /10100
          xot 172.17.125.54 1         next       3120 /4424
          xot 172.17.125.34 1         next       63 /88
=====
HG2      vc-count Serial0         5         2 VC       3120 /4424
          Serial3         7         1 VC       15640 /10100
=====
```

Table 1 describes significant fields shown in the display.

Table 1 show x25 hunt-group Field Descriptions

Field	Description
ID	Hunt group name.
Type	Method of load balancing (rotary or vc-count).
Target	Range of interfaces that a call within the hunt group can go to.
uses	Frequency of use indicator for each interface.
status	State of interface at that moment (next, last used, unavailable, full or # VC).
traffic (out/in)	Number of data bytes transmitted through the interface.

Status of an interface may be:

- next—Interface that will be used next for rotary distribution method.
- last used—Interface just used for rotary distribution method.
- unavailable—Interface is shutdown.

- **full**—All logical channels on the interface are in use.
- **# VC**—(vc-count only) Number of VCs currently in use on the interface.

Use the **clear x25** command in EXEC mode to reset the “traffic,” “# VC,” and “uses” fields to zero.

Related Commands

You can use the master index or search online to find documentation of related commands.

- **clear x25**
- **x25 hunt-group**

x25 hunt-group

To create and maintain a hunt group, use the **x25 hunt-group** global configuration command. To delete this hunt group, use the **no** form of this command.

```
x25 hunt-group name {rotary | vc-count}
```

```
no x25 hunt-group name
```

Syntax Description

<i>name</i>	Name you assign to the particular hunt group.
rotary	Each call steps to the next interface.
vc-count	Each call is placed on the interface with most available logical channels.

Command Mode

Global configuration.

Usage Guidelines

This command first appeared in the Cisco IOS Release 12.0(3)T. The following restrictions apply:

Only one load-balancing distribution method can be selected for a hunt group, although one interface can participate in one or more hunt groups.

XOT cannot be used in hunt groups configured with the vc-count distribution method. XOT does not limit the number of calls that can be sent to a particular destination, so the method of selecting the hunt group member with the largest number of available VCs will not work. XOT can be used in hunt groups configured with the rotary distribution method.

Example

The following example shows the creation of hunt group “HG1” using the rotary distribution method, with X.25 configuration on serial interfaces 1 and 2 for that hunt group; as well as showing x25 configuration for the Ethernet interface (**xot**) with two specific target IP addresses (172.17.125.54 and 172.17.125.34). The example also shows the creation of hunt group “HG2” using the vc-count distribution method, with X.25 configuration on serial interfaces 0 and 3 for that hunt group:

```
Router(config)# x25 hunt-group HG1 rotary
Router(config-x25-huntgro)# interface serial 1
Router(config-x25-huntgro)# interface serial 2
Router(config-x25-huntgro)# xot 172.17.125.54
Router(config-x25-huntgro)# xot 172.17.125.34
Router(config-x25-huntgro)# exit
Router(config)# x25 hunt-group HG2 vc-count
Router(config-x25-huntgro)# interface serial 0
Router(config-x25-huntgro)# interface serial 3
```

Related Commands

You can use the master index or search online to find documentation of related commands.

```
show x25 hunt-group
```

x25 route

To create an entry in the X.25 routing table (to be consulted for forwarding incoming calls and for placing outgoing packet assembler/disassembler (PAD) or protocol translation calls), use an appropriate form of the **x25 route** global configuration command. To remove an entry from the table, use the **no** form of the command.

```
x25 route [#position] [selection] [modification] disposition [xot-keepalive]  
no x25 route [#position] [selection] [modification] disposition
```

Syntax Description

<i>#position</i>	(Optional) A pound sign (#) followed by a number designates the position in the routing table at which to insert the new entry. If no <i>position</i> value is given, the entry is appended to the end of the routing table.
<i>selection</i>	(Optional) The <i>selection</i> options identify when the subsequent <i>modification</i> and <i>disposition</i> elements apply to an X.25 call; any or all variables may be specified for a route. See Table 2 in the “Usage Guidelines” section for the valid selection keyword and argument options. Although each individual selection criterion is optional, at least one <i>selection</i> or <i>modification</i> element must be specified in the x25 route command.
<i>modification</i>	(Optional) Modifies the source or destination addresses of the selected calls. The standard regular expression substitution rules are used, where a match pattern and rewrite string direct the construction of a new string. See Table 3 in the “Usage Guidelines” section for the valid modification keyword and argument options. Although each individual modification is optional, at least one <i>selection</i> or <i>modification</i> element must be specified in the x25 route command.
<i>disposition</i>	Specifies the disposition of a call matching the specified selection pattern. See Table 7 in the “Usage Guidelines” section for the valid disposition keyword and argument options.
<i>xot-keepalive</i>	(Optional) Specifies an XOT keepalive period and number of XOT keepalive retries. XOT relies on TCP to detect when the underlying connection is dead. TCP detects a dead connection when transmitted data goes unacknowledged for a given number of attempts over a period of time. See Table 8 in the “Usage Guidelines” section for keepalive options.

Default

No entry is created in the X.25 routing table.

Command Mode

Global configuration

Usage Guidelines

The enhanced **x25 route** command replaces the **x25 map cmns** command. The **x25 route alias** form of this command (supported in earlier releases) has been replaced by the **x25 alias** command.

The *selection* criteria **source** and **dest-ext** first appeared in Cisco IOS Release 11.3. The interface *disposition* to a CMNS destination first appeared in Cisco IOS Release 11.3; in prior releases, CMNS routing information was implied by maps defining an NSAP prefix for a CMNS hosts MAC address. The **clear** interface *disposition* option first appeared in Cisco IOS Release 11.3; in prior releases, the disposition was implicit in a route to the Null 0 interface. The *modification* elements are long-standing but newly applicable to all dispositions in Cisco IOS Release 11.3.

Note The entire command must be entered on one line.

Selection Options

Selection options specify match criteria. When a call matches all selection criteria in an X.25 route, then the specified modification and disposition are used for the call.

As many as four selection options can be used to determine the route:

- Called X.121 network interface address (destination host address)
- Calling X.121 network interface address (source host address)
- Called address extension (destination NSAP address)
- X.25 packet's call user data (CUD) field

Table 2 lists the *selection* options for the **x25 route** command. At least one *selection* or *modification* element must be specified.

Table 2 x25 route Command Selection Options

Selection Options	Description
destination-pattern	(Optional) Destination address pattern, which is a regular expression that can represent either one X.121 address (such as ^1111000\$) or any address in a group of X.121 addresses (such as ^1111.*).
source source-pattern	(Optional) Source address pattern, which is a regular expression that can represent either one X.121 source address (such as ^2222000\$) or any address in a group of X.121 addresses (such as ^2222.*).
dest-ext nsap-destination-pattern	(Optional) NSAP destination address pattern, which is a regular expression that can represent either an NSAP destination address (such as ^11.1111.0000\$) or an NSAP prefix (such as ^11.1111.*). Note: A period (.) in the pattern is interpreted as a character wildcard, which will not interfere with a match to the actual period in the NSAP; if desired, an explicit character match may be used (such as ^11\,1111\,.*).
cud user-data-pattern	(Optional) Call user data (CUD) pattern, which is specified as a regular expression of printable ASCII text. The CUD field may be present in a call packet. The first few bytes (commonly 4 bytes long) identify a protocol; the specified pattern is applied to any user data after the protocol identification.
hunt-group name	Routes the selected call to the X.25 hunt group. The chosen router may vary depending on the hunt group configuration.

Note The X.121 and NSAP addresses are specified as regular expressions. A common error is to specify the address digits without anchoring them to the beginning and end of the address. For example, the regular expression 1111 will match an X.121 address that has four successive 1s somewhere in the address; to specify the single X.121 address, the form ^1111\$ must be used.

Regular expressions are used to allow pattern-matching operations on the addresses and user data. A common operation is to do prefix matching on the X.121 DNIC field and route accordingly. The caret (^) is a special regular expression character that anchors the match at the beginning of the pattern. For example, the pattern ^3306 will match all X.121 addresses with a DNIC of 3306.

Modification Options

Addresses typically need to be modified when traffic from a private network that uses arbitrary X.121 addresses must transit a public data network, which must use its own X.121 addresses. The easiest way to meet the requirement is to specify in the **x25 route** command a way to modify the private address into a network X.121 address or to modify a network X.121 address into a private address. The addresses are modified so that no change to the private addressing scheme is required.

The *modification* options use the standard UNIX regular expression substitution operations to change an X.25 field. A pattern match is applied to an address field, which is rewritten as directed by a rewrite pattern.

Table 3 lists the *modification* options for the **x25 route** command. At least one *selection* or *modification* element must be specified.

Table 3 x25 route Command Modification Options

Modification Option	Description
substitute-source <i>rewrite-source</i>	(Optional) Calling X.121 address rewrite pattern. The source address, <i>source-pattern</i> , and this <i>rewrite-source</i> pattern are used to form a new source address. If no <i>source-pattern</i> is specified, any <i>destination-pattern</i> match pattern is used. If neither match pattern is specified, a default match pattern of .* is used. See Table 4 and Table 5 for summaries of pattern and character matching, respectively. See Table 6 for a summary of pattern rewrite elements.
substitute-dest <i>rewrite-dest</i>	(Optional) Called X.121 address rewrite pattern. The destination address, <i>destination-pattern</i> , and this <i>rewrite-dest</i> pattern are used to form a new destination address. If no <i>destination-pattern</i> is specified, a default match pattern of .* is used. See Table 4 and Table 5 for summaries of pattern and character matching, respectively. See Table 6 for a summary of pattern rewrite elements.

Note As of Cisco IOS Release 11.3, the **substitute-source** and **substitute-dest** options also apply to PAD calls.

Source address. A modification of the source address is directed by the rewrite string using one of three possible match patterns. If the **source** *source-pattern* selection option is defined, it is used with the *source-rewrite* string to construct the new source address; otherwise, a *destination-pattern*

regular expression is used (for backward compatibility) or a wildcard regular expression (.*) is used. In the *rewrite-source* argument, the backslash character (\) indicates that the digit immediately following the argument selects a portion of the matched address to be inserted into the new called address.

Destination address. A modification of the destination address is directed by the rewrite string using one of two possible match patterns. If the *destination-pattern* selection option is defined, it is used with the *destination-rewrite* string to construct the new destination address; otherwise, a wildcard regular expression (.*) is used. In the *rewrite-dest* argument, the backslash character (\) indicates that the digit immediately following the argument selects a portion of the original called address to be inserted into the new called address.

Refer to Table 4, Table 5, and Table 6 respectively, for summaries of pattern matching, character matching, and pattern rewrite elements. Note that up to nine pairs of parentheses can be used to identify patterns to be included in the modified string. A more complete description of the pattern-matching characters is found in the “Regular Expressions” appendix in the *Dial Solutions Command Reference*.

Table 4 Pattern Matching for X.25 Route Selection and Modification Options

Pattern	Description
*	Matches 0 or more occurrences of the preceding character.
+	Matches 1 or more occurrences of the preceding character.
?	Matches 0 or 1 occurrences of the preceding character. ¹

1. Precede the question mark with **Ctrl-V** to prevent the question mark from being interpreted as a **help** command.

Table 5 Character Matching for X.25 Route Selection and Modification Options

Character	Description
^	Matches the beginning of the input string.
\$	Matches the end of the input string.
\char	Matches the single character <i>char</i> specified.
.	Matches any single character.

Table 6 Pattern Replacements for X.25 Route Selection and Modification Options

Pattern	Description
\0	The pattern is replaced by the entire original address.
\1...9	The pattern is replaced by strings that match the first through ninth parenthetical part of the X.121 address.

Disposition Option

The **xot-source** disposition option can improve the resilience of the TCP connection if, for instance, a loopback interface is specified. By default, a TCP connection’s source IP address is that of the interface used to initiate the connection; a TCP connection will fail if either the source or destination IP address is no longer valid. Because a loopback interface never goes down, its IP address is always valid. Any TCP connections originated using a loopback interface can be maintained as long as a path exists to the destination IP address, which may also be the IP address of a loopback interface.

Table 7 lists the *disposition* choices for the **x25 route** command. You must select one of these choices.

Table 7 x25 route Command Dispositions

Disposition	Description
interface <i>serial-interface</i>	Routes the selected call to the specified X.25 serial interface.
interface <i>serial-interface</i> dcli <i>number</i>	(Optional) Routes the X.25 call to the specified Annex G link. You must include the interface number as well as entering the DLCI number. You only need to do this if you want the router to accept switched calls, as well as originate them.
interface <i>cmns-interface</i> mac <i>mac-address</i>	Routes the selected call out the specified broadcast interface via CMNS to the LAN destination station. The broadcast interface type can be Ethernet, Token Ring, or FDDI. The interface numbering scheme depends on the router interface hardware.
xot <i>ip-address</i> [<i>ip2-address</i> [... <i>ip6-address</i>]]] [xot-source <i>interface</i>]	Routes the selected call to the XOT host at the specified IP address. Subsequent IP addresses are tried, in sequence, only if XOT is unable to establish a TCP connection with a prior address.
hunt-group <i>name</i>	Routes the selected call to the X.25 hunt group. The chosen route may vary depending on the hunt group configuration.
clear	Terminates the call.

XOT Keepalive Options

TCP maintains each connection using a keepalive mechanism that starts with a default time period and number of retry attempts. If a received XOT connection is dispatched using a route with explicit keepalive parameters, those values will be used for the TCP connection. If an XOT connection is sent using a route with explicit keepalive parameters, those values will be used for the TCP connection.

Table 8 lists and describes the *xot-keepalive* options for the **x25 route** command.

Table 8 x25 route Command XOT Keepalive Options

XOT-Keepalive Option	Description
xot-keepalive-period <i>seconds</i>	Number of seconds between keepalives for XOT connections. The default is 60 seconds.
xot-keepalive-tries <i>count</i>	Number of times TCP keepalives should be sent before dropping the connection. The default value is 4 times.

X.25 Routing Action when a Match Is Found

If a matching route is found, the incoming call is forwarded to the next hop depending on the routing entry. If no match is found, the call is cleared. If the route specifies a serial interface running X.25 or a broadcast interface running CMNS, the router attempts to forward the call to that host. If the interface is not operational, the subsequent routes are checked for forwarding to an operational interface. If the interface is operational but out of available virtual circuits, the call is cleared. Otherwise, the expected Clear Request or Call Accepted message is forwarded back toward the originator. A call cannot be forwarded out the interface on which it arrived.

If the matching route specifies an XOT disposition, a TCP connection is established to port 1998 at the specified IP address, which must be an XOT host. The Call Request packet is forwarded to the remote host, which applies its own criteria to handle the call. If, upon receiving an XOT call, a routing table entry is not present, or the destination is unavailable, a Clear Request is sent back and the TCP connection is closed. Otherwise, the call is handled and the expected Clear Request or Call

Accepted packet is returned. Incoming calls received via XOT connections that match a routing entry specifying an XOT destination are cleared. This restriction prevents Cisco routers from establishing an XOT connection to another router that would establish yet another XOT connection.

Examples

The following example uses regular expression pattern matching characters to match just the initial portion of the complete X.25 address. Any call with a destination address beginning with 3107 that is received on an interface other than serial 0 is forwarded to serial 0.

```
x25 route ^3107 interface serial 0
```

The following example routes the X.25 call to the specified Annex G DLCI link. You must include both interface number and DLCI number. It is this combination of both these numbers that indicates the logical X.25 interface over Frame Relay.

```
x25 route ^2222 interface serial 1 dlci 20
```

The following example prevents X.25 routing for calls that do not specify a source address:

```
x25 route source ^$ clear
```

The following example configures alternate XOT hosts for the routing entry. If the first address listed is not available, subsequent addresses are tried until a connection is made. If no connection can be formed, the call is cleared.

```
x25 route ^3106$ xot 172.20.2.5 172.20.7.10 172.10.7.9
```

The following example clears calls that contain a 3 in the source address. The disposition keyword **clear** is new:

```
x25 route source 3 clear
```

The following example clears calls that contain two consecutive 3's in the source address:

```
x25 route source 33 clear
```

The following example clears a call to the destination address 9999:

```
x25 route ^9999$ clear
```

The following example specifies a route for specific source and destination addresses. (The ability to combine source and destination patterns is a new feature.)

```
x25 route ^9999$ source ^3333$ interface serial 0
```

The following example routes the call to the XOT host at the specified IP address. The disposition keyword **xot** is new. In prior releases the keyword **ip** was used.

```
x25 route ^3333$ xot 172.21.53.61
```

The following example routes calls containing the destination extension address preamble 11.1234:

```
x25 route dest-ext ^11.1234.* interface serial 0
```

The following example rewrites the destination address as 9999. There must be a minimum of four 8's in the address. (8888888 will change to 9999.)

```
x25 route 8888 substitute-dest 9999 interface serial 0
```

The following example substitutes only part of the destination address. “^88” specifies the original destination string must begin with 88. “(.*)” indicates the string can end with any number, 0-9, and can be more than one digit. “99\1” changes the destination address to 99 plus whatever matches “.*” in the original destination address. For example, 8881 will change to 9981.

```
x25 route ^88(.*) substitute-dest 99\1 interface serial 0
```

The following example substitutes only part of the destination address and also removes a specified number of digits from the address. “^88” specifies the original destination string must begin with 88. “(..)” matches any two digits. “(.*)” specifies the string can end with any number, 0-9, and can occur zero or more times. Thus any address that starts with 88 and has four or more digits will be rewritten to start with 99 and omit the third and fourth digits. For example, 881234 will change to 9934.

```
x25 route ^88(..)(.*) substitute-dest 99\2 interface serial 0
```

The following example looks for a specified destination address and changes the source address. “9999” is the destination address. The original source address changes to “2222” because the call is made to the destination 9999.

```
x25 route ^9999$ substitute-source 2222 interface serial 0
```

The following example rewrites the source address based on the source address. “9999” matches any destination address with four consecutive 9s. “^...(*)” matches any source address with at least three digits; the command removes the first three digits and rewrites any digits after the first three as the new source address. For example, a call to 9999 from the source address 77721 will be forwarded using the calling address 21 and the called address 9999.

```
x25 route 9999 source ^...(*) substitute-source \1 interface serial 0
```

The following example adds a digit to the source and destination addresses patterns. “09990” is the destination address pattern. The source can be any address. “9\0” specifies to add a leading 9 to the destination address pattern. “3\0” specifies to add a leading 3 to the source address pattern. For example, a call using source 03330 and destination 09990 will change to 303330 and 909990, respectively.

```
x25 route 09990 source .* substitute-dest 9\0 substitute-source 3\0 interface serial 0
```

Related Commands

You can use the master indexes or search online to find documentation of related commands.

show x25 route