DNS-Based X.25 Routing

This document contains the following sections:

Feature Overview

Managing a large TCP/IP network requires accurate and up-to-date maintenance of IP addresses and X.121 address mapping information on each router database in the network. Currently, this data is managed manually. Because these addresses are constantly being added and removed in the network, administration of the routing table of every router needs to be updated, which is a time consuming and error-prone task. This has also been a problem for mnemonics.

X.25 has long operated over an IP network, specifically using Transmission Control Protocol (TCP) as a reliable transport mechanism. This method is known as X.25 over TCP (XOT). However, large networks and financial legacy environments experienced problems with the amount of route configuration that needed to be done manually because each router switching calls over TCP needed every destination configured. Every destination from the host router needed a static IP route statement, and for larger environments, these destinations could be as much as several thousand per router. Until now, the only way to map X.121 addresses and IP addresses was on a one-to-one basis using the **x25 route** *x121address* **xot** *ipaddress* command.

The solution to this problem was to centralize route configuration that routers could then access for their connectivity needs. This centralization is the function of the Domain Name System (DNS)-based X.25 routing feature, because the DNS server is a database of all domains and addresses on a network.

DNS has three components:

- Domain name space or resource records—Defines the specifications for a tree-structured domain name space.
- Name servers—Programs to hold information about the domain tree structure.
- Resolvers—Receive a client request and return the desired information in a form compatible with a local hosts data formats.

Only one route statement needs to be maintained in the host router to connect it to the DNS, and all the ongoing route statement maintenance can be done in the DNS. Until now, X.25 never utilized the DNS because it was not thought possible to use IP functionality in legacy environments such as X.25. When using DNS, you must adhere to these rules:

- Names must consist of printable characters.
- No embedded white space is permitted.
- Periods must separate subdomains.
- Names are case sensitive.
- Append any domain configured for the router to the user-specified name format.
- Total length of the name must not exceed 255 characters.

For more information on configuring the DNS, see the "Configuring the DNS Service" chapter in the *Cisco DNS/DHCP Manager Administrator's Guide*.

With the DNS-Based X.25 Routing feature, it is easy to manage the X.121-to-IP addressing correlation and the mnemonic-to-X.121 addressing correlation. Instead of the router needing a route statement going to all destinations, all that is needed is a wildcard route statement that covers all addresses in the DNS.

The **x25 route** *disposition* **xot** command option has been modified to include the **dns** *pattern* argument after the **xot** keyword, where *pattern* is a rewrite element that works in the same way that address substitution utilities works (see Table 6 in the "Command Reference" section).

The wildcard **^.*** characters and **\0** pattern of the modified **x25 route ^.* xot dns \0** command give it more universality and effectiveness, and make DNS-based X.25 routing simple and easy to use. These characters and pattern already exist, and are explained in detail under the **x25 route** command (see Table 4, Table 5, and Table 6 in the "Command Reference" section).

This command only functions if the DNS route table mapping has been configured in a method recognized and understood by X.25 and the DNS server.

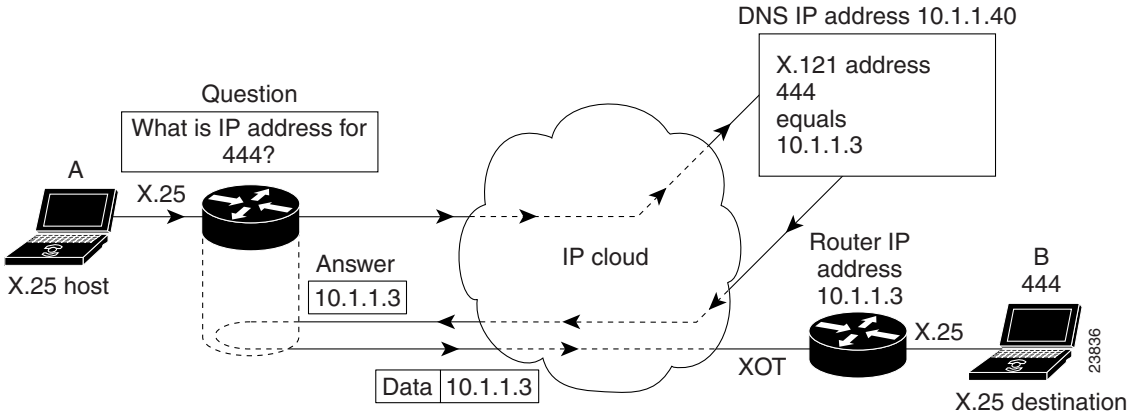
The following example is a setup from a DNS route table showing which X.121 address relates to which IP address:

```
222 IN      A       172.18.79.60
444 IN      A       10.1.1.3
555 IN      A       10.1.1.2
```

The command line **x25 route 444 xot dns \0** shown in the DNS-based X.25 routing configuration example is what extracts the IP address from the DNS. The **\0** pattern replaces itself with 444. The 444 is then used as the index into the DNS route table to generate the IP address 10.1.1.3. Other characters can be combined with the pattern, for example, **A-\0**. In the DNS database, the index would show as A-444.

Using Figure 1 as an example, when the router sends a call, it goes to the DNS. The DNS checks its route table, identifies the X.121 address 444 and its related IP address 10.1.1.3. The DNS returns the IP address to the host's router, which then creates a route statement and forwards the data to the IP address of the destination's router (10.1.1.3).

Figure 1 How DNS-Based X.25 Routing Functions Using XOT over an IP Cloud



Mnemonic Resolution

The DNS-Based X.25 Routing feature can also be used for mnemonic resolution with or without use of XOT routing. For more information on mnemonic addressing, refer to the "Configuring the Cisco PAD Facility for X.25 Connections" chapter in the Cisco IOS Release 12.0 *Dial Configuration Guide*.

When using mnemonics (an easy-to-remember alias name for an X.121 address) with XOT, the same communication with the DNS occurs, except that the router needs to contact the DNS twice—first to get the X.121 address using the mnemonic, then the IP address using the X.121 address. However, there is no significant performance issue because the process happens very fast.

The following example is a setup from the DNS route table showing a mnemonic and its related X.121 address ("destination_host" represents 222). The **X25** keyword ensures that this line will be recognized by the DNS-Based X.25 Routing feature in the DNS server.

```
destination_host IN      X25     222
```

Using X.28 to retrieve this address, you would enter the following commands:

```
Router# x28
*destination_host
Translating "destination_host"...domain server (10.1.1.40)
```

Notice the output line requesting mnemonic resolution from the DNS server with IP address 10.1.1.40. If you were using PAD, you would need to enter only the mnemonic name, as in the following example:

```
Router# pad destination_host
```

Note This feature should not be used in the public Internet. It should only be used for private network implementations, because in the Internet world the DNS has conventions for names and addresses, which the DNS-Based X.25 Routing feature does not comply with.

Caution You must remove any permanent entry for X.25 located in the router's host table that has been duplicated in the DNS route table (as part of the enabling process for the DNS-Based X.25 Routing feature). Otherwise, DNS-based X.25 routing will be overridden by the router's host table entries.

Benefits

Scalability

Scales well with networks that have multiple XOT routers.

Improved Maintenance

This feature simplifies maintenance of routing tables and creation of new routes.

Increased Productivity

This feature reduces labor intensive tasks and the possibility of human error when routing tables are maintained.

Restrictions

- DNS-based X.25 routing should not be implemented in a public Internet environment, only in private networks.
- The X.25 address or mnemonic sent to the DNS for address resolution should not exceed 255 bytes per the DNS restriction on length.
- Existing Cisco IOS software configuration commands for a name server must be used for the DNS name lookup to succeed.
- You cannot specify any **x25 route** command options on the DNS. These options must be configured within the **x25 route** command itself.
- Mnemonic name restrictions that apply when using X.28 still apply with the DNS-Based X.25 Routing feature (for example, not using **-**, **.**, **P**, or **D** in the mnemonic).

Related Documents

- ITU-T X.25
- ITU-T X.28/X.29/X.3
- *X.25 over TCP/IP* (RFC #1613)
- *Domain Names—Concepts and Facilities* (RFC #1034)
- *Domain Names—Implmentation and Specification* (RFC #1035)

Supported Platforms

- Cisco 1000 series
- Cisco 1600 series
- Cisco 2500 series
- Cisco 2600 series
- Cisco Catalyst 3000 series

- Cisco 3600 series
- Cisco MC3810 Mutiservice Concentrator
- Cisco 4000 series (Cisco 4000, 4000-M, 4500, 4500-M, 4700, 4700-M)
- Cisco 7000 series
- Cisco 7200 series
- Cisco 7500 series

Supported Standards, MIBs, and RFCs

MIBs

No new or modified MIBs are supported by this feature.

For descriptions of supported MIBs and how to use MIBs, see the Cisco MIB web site on CCO at http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.

RFCs

None

Prerequisites

You must have DNS activated and X.25 configured for XOT to enable the DNS-Based X.25 Routing feature. Details can be found in the Cisco IOS Release 12.0 *Wide-Area Networking Configuration Guide*.

Configuration Tasks

To configure DNS-Based X.25 Routing, use the following command in configuration mode. This task assumes that you already have XOT and DNS configured and enabled, and the route table in the DNS server has been correctly organized. For more information on configuring XOT, DNS, and setting up the route table in the DNS server correctly, refer to the Cisco IOS Release 12.0 *Wide-Area Networking Configuration Guide*.

Configuring DNS-Based X.25 Routing

Step	Command	Purpose
1	`Router(config)# x25 route x121address xot dns pattern`	Configures XOT routing to search for IP addresses in the DNS.

Verifying DNS-Based X.25 Routing

To verify that the DNS-Based X.25 Routing feature is configured, use the **show x25 route** command in EXEC mode.

```
Router# show x25 route
  #  Match                          Substitute          Route to
  1  dest 444                                           xot dns \0
  2  dest 555                                           xot dns \0
```

Verifying DNS-Based X.25 Mnemonic Resolution

To verify DNS-based X.25 mnemonic resolution, use the **show hosts** command in EXEC mode. All permanent (perm) entries of type X.121 should be removed from the route table for DNS-based X.25 routing to work. In the following example the mnemonic "destination_host" is showing itself to be a permanent entry:

```
Router# show hosts
Default domain is home.com
Name/address lookup uses domain service
Name servers are 10.1.1.40

Host                    Flags      Age Type   Address(es)
destination_host        (perm, OK)  1  X.121  222
```

The following example uses the **no x25 host** command to remove the permanent entry mnemonic "destination_host" from the route table, and illustrates the subsequent absence of any mnemonics in the route table by using the **show hosts** command again:

```
Router(config# no x25 host destination_host

Router# show hosts
Default domain is cisco.com
Name/address lookup uses domain service
Name servers are 10.1.1.40

Host                    Flags      Age Type   Address(es)
```

Verification and Troubleshooting Tips

If you are having trouble with the DNS-Based X.25 Routing feature not functioning correctly, it is worth checking to see that your DNS is configured properly and operating correctly as follows:

- Use the **show hosts** command to display temporary entries cached by DNS at the router
- Use **debug x25 events** and **debug domain** commands to display current data flow. See the "Debug Commands" section for more information.

Configuration Examples

This section provides the following configuration example for XOT switch configuration to use XOT switching via the DNS:

```
Router(config)# ip tcp synwait-time 5
Router(config)# ip name-server 10.1.1.40
Router(config)# x25 routing
Router(config)# !
Router(config)# service pad to-xot
Router(config)# service pad from-xot
Router(config)# !
Router(config)# ip domain-name home.com
Router(config)# ip domain-list home.com
Router(config)# ip domain-lookup
Router(config)# !
Router(config)# interface Ethernet1
Router(config-if)# ip address 10.1.1.2 255.255.255.0
Router(config-if)# no shut
Router(config-if)# exit
Router(config)# interface Serial0
Router(config-if)# encapsulation x25 dce
Router(config-if)# no shut
Router(config-if)# exit
Router(config)# x25 route 444 xot dns \0
Router(config)# x25 route 555 xot dns \0
```

Command Reference

This section documents modified commands. All other commands used with this feature are documented in the Cisco IOS Release 12.0 command reference publications.

- **show x25 route**
- **x25 route**

In Cisco IOS Release 12.0(1)T or later, you can search and filter the output for **show** and **more** commands. This functionality is useful when you need to sort through large amounts of output, or if you want to exclude output that you do not need to see.

To use this functionality, enter a **show** or **more** command followed by the "pipe" character (|), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search or filter on:

command **|** {**begin | include | exclude**} *regular-expression*

Following is an example of the **show atm vc** command in which you want the command output to begin with the first line where the expression "PeakRate" appears:

show atm vc | begin PeakRate

For more information on the search and filter functionality, refer to the Cisco IOS Release 12.0(1)T feature module titled *CLI String Search*.

show x25 route

To display the X.25 routing table, use the **show x25 route** EXEC command.

show x25 route

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Command History

Release	Modification
10.0	This command was introduced.
12.0(5)T	The **dns** keyword was added to the output field.

Examples

The following example shows output from the **show x25 route** command:

```
Router# show x25 route

# Match                     Substitute              Route To
1 dest ^1311001$                                    Serial0, 0 uses
2 dest ^1311002$                                    xot 172.20.170.10
3 dest 444                                          xot dns \0
4 dest 555                                          xot dns \0
```

Table 1 describes significant fields shown in the display.

Table 1 show x25 Route Display Field Descriptions

Field	Description
#	Number identifying the entry in the X.25 routing table.
Match	The match criteria and patterns associated with this entry.
Route To	Destination to which the router will forward a call; X.25 destinations identify an interface; CMNS destinations identify an interface and host MAC address; XOT destinations either identify up to six IP addresses (#2), or the **x25 route** pattern for retrieving up to six IP addresses from the DNS (#3 and #4).

Related Commands

Command	Description
x25 route	Creates an entry in the X.25 routing table.

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-options*] [*modification-options*] *disposition-options* [*xot-keepalive-options*]

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

Syntax Description

#position	(Optional) A pound sign (#) followed by a number designates the position in the routing table at which to insert the new entry. If no value for the *position* argument is given, the entry is appended to the end of the routing table.
selection-options	(Optional) The selection options identify when the subsequent modification and disposition options apply to an X.25 call; any or all variables may be specified for a route. For selection keyword and argument options, see Table 2 in the "Usage Guidelines" section. For selection and modification pattern and character matching and replacement see Table 4, Table 5, and Table 6 in the "Usage Guidelines" section. Although each individual selection criterion is optional, at least one selection or modification option must be specified in the **x25 route** command.
modification-options	(Optional) The modification options modify 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. For modification keyword and argument options, see Table 3 in the "Usage Guidelines" section. For selection and modification pattern and character matching and replacement see Table 4, Table 5, and Table 6 in the "Usage Guidelines" section. Although each individual modification is optional, at least one selection or modification option must be specified in the **x25 route** command.
disposition-options	Specifies the disposition of a call matching the specified selection pattern. For disposition keyword and argument options, see Table 7 in the "Usage Guidelines" section.
xot-keepalive-options	(Optional) The XOT-keepalive options specify an X.25 over TCP (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 sent data goes unacknowledged for a given number of attempts over a period of time. For XOT-keepalive keyword and argument options, see Table 8 in the "Usage Guidelines" section.

Default

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

Command Mode

Global configuration

Command History

Release	Modification
11.3	The selection option keywords **source** and **dest-ext** and the interface *disposition* to a Connection-Mode Network Service (CMNS) destination first appeared. In prior releases, CMNS routing information was implied by maps defining a network service access point (NSAP) prefix for a CMNS host's Media Access Control (MAC) address. The **clear** interface *disposition* option was introduced. In prior releases, the disposition was implicit in a route to the Null 0 interface.
12.0(5)T	For the DNS-Based X.25 Routing feature, the keyword **dns** followed by *pattern* (see Table 6) was introduced, to be used after the **xot** in the **x25 route** *destination-pattern* **xot** command.

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 modification options are long-standing but newly applicable to all dispositions in Cisco IOS Release 11.3 and later.

Note The entire command must be entered on one line.

Selection Options

Selection arguments 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 or source host address)
- Called address extension (destination NSAP address)
- X.25 packet's call user data (CUD) field
- Input interface from which the call was received (input-interface)

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

Table 2 x25 route Command Selection Options

Selection Option	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) 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.
input interface *interface number*	(Optional) Specifies interface number on which the call will be received.

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 use prefix matching on the X.121 Data Network Identification Code (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 option must be specified.

Table 3 x25 route Command Modification Options

Modification Option	Description
substitute-source *rewrite-source*	(Optional) Calling X.121 address rewrite pattern. The source address, *source-pattern*, and this *rewrite-source* pattern are used to form a new source address. If no *source-pattern* is specified, any *destination-pattern* 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 *rewrite-dest*	(Optional) Called X.121 address rewrite pattern. The destination address, *destination-pattern*, and this *rewrite-dest* pattern are used to form a new destination address. If no *destination-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.

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.

Pattern and Character Matching and Replacement for Selection and Modification Options

See Table 4, Table 5, and Table 6, respectively, for summaries of pattern matching, character matching, and pattern replacement 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* publication.

Table 4 Pattern Matching for x25 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 x25 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 *char* specified.
.	Matches any single character.

Table 6 Pattern Replacements for x25 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.

Using the **continue** keyword provides flexibility by reducing the number of X.25 route configurations necessary in the route table by breaking them into separate, simpler, and more manageable tasks. It allows the **x25 route** command to cumulatively hold all specified route entries and carry whatever selection or modification options you may have just specified on the command line. The route table lookup terminates when a matching route is found among the remaining entries in the route table. The **continue** disposition must be the last option on the **x25 route** command line.

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

Table 7 x25 route Command Disposition Options

Disposition Option	Description
continue	(Optional) Combines sequential route table lookups, holding onto any "selections" and "modifications" specified on the **x25 route** statement.
interface *interface number*	Routes the selected call to the specified X.25 serial interface.

Table 7 x25 route Command Disposition Options (continued)

Disposition Option	Description
interface *interface number* **dlci** *number*	(Optional) Routes the X.25 call to the specified Annex G link. You must include the interface number and enter the data link connection identifier (DLCI) number. You only need to do this if you want the router to accept switched calls, as well as originate them.
interface *cmns-interface* **mac** *mac-address*	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 Fiber Distributed Data Interface (FDDI). The interface numbering scheme depends on the router interface hardware.
xot *ip-address* [*ip2-address* [...[*ip6-address*]]] [**xot-source** *interface*]	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.
xot dns *pattern*	Used with the DNS for X.25 feature, this option consults the DNS to get up to six destination IP addresses using whatever lookup pattern you choose (see Table 6).
hunt-group *name*	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 *seconds*	Number of seconds between keepalives for XOT connections. The default is 60 seconds.
xot-keepalive-tries *count*	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 packet 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 on the remote host, 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 Annex G 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 33 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 ^333$ 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 DNS-based X.25 routing example shows an X.25 request to the DNS. The **\0** pattern indicates that the entire incoming X.121 address is being used as the index into the DNS, which will return the required IP address.

```
x25 route ^.* xot dns \0
```

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 8s 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 of the Calling Address Insertion based on the Input Interface feature strips the destination address off a call coming from the X.25 (serial 0) network with a destination address beginning with 02 and ending in anything, replaces it with a network-assigned address of \1, and forwards it to the substitute destination of interface serial 2:

```
x25 route ^02(.*) input-interface serial 0 substitute-destination \1 interface serial 2
```

The following example of the Calling Address Insertion based on the Input Interface feature inserts a 2 to the source address of any call coming from serial 2 and forwards the call to serial 0:

```
x25 route ^02 input-interface serial 2 substitute-source 2\0 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

Command	Description
show x25 route	Displays the X.25 routing table.

Debug Commands

The **debug x25** command has been modified to accommodate DNS-based X.25 routing. Issue the **debug x25 events** command to show events occurring while the X.25 address is resolving itself to an IP address using a DNS server.

- **debug x25**

debug x25

To display information about X.25 traffic, use one of the following **debug x25** commands. The commands allow you to display all information or an increasingly restrictive part of the information.

Caution This command is processor intensive and can render the router useless. Use this command only when the aggregate of all reportable X.25 traffic is fewer than five packets per second. The generic forms of this command should be restricted to low-speed, low-usage links running at less than 19.2 kbps. Because the **debug x25 vc** command and the **debug x25 vc events** command display traffic for only a small subset of virtual circuits, they are safer to use under heavy traffic conditions, as long as events for that virtual circuit are fewer than 25 packets per second.

To display information about all X.25 traffic, including traffic for X.25, Connection Mode Network Service (CMNS), and X.25 over TCP (XOT) services, use the **debug x25** EXEC command (default **all**).

[**no**] **debug x25**

To display information about all X.25 traffic except data and resource record packets, use the **debug x25 events** command (see the "debug x25 events for DNS-Based X.25 Routing" section in the "Examples" section for more details.). The **no** form of this command disables debugging output.

[**no**] **debug x25 events**

To display information about a specific X.25 service class, use the following form of the **debug x25** EXEC command:

[**no**] **debug x25** [**only** | **cmns** | **xot**} [**events** | **all**]

To display information about a specific X.25 or CMNS context, use the following form of the **debug x25** EXEC command:

[**no**] **debug x25 interface** {*serial-interface* | *cmns-interface* **mac** *mac-address*} [**events** | **all**]

To display information about a specific X.25 or CMNS virtual circuit, use the following form of the **debug x25** EXEC command:

[**no**] **debug x25 interface** {*serial-interface* | *cmns-interface* **mac** *mac-address*} **vc** *number* [**events** | **all**]

To display information about traffic for all virtual circuits using a given number, use the following form of the **debug x25** EXEC command. The **no** form of this command removes the filter for a particular virtual circuit from the **debug x25 all** or **debug x25 events** output:

[**no**] **debug x25 vc** *number* [**events** | **all**]

To display information about traffic to or from a specific XOT host, use the following form of the **debug x25 xot** EXEC command:

[**no**] **debug x25 xot** [**remote** *ip-address* [**port** *number*]] [**local** *ip-address* [**port** *number*]] [**events** | **all**]

Use the **debug x25** EXEC command with the **aodi** keyword to display information about an interface running Point-to-Point Protocol (PPP) over an X.25 session. The **no** form of this command disables debugging output.

[no] debug x25 aodi

Syntax Description

events	(Optional) Displays all traffic except data and Receiver Ready (RR) packets.
all	(Optional) Displays all traffic. This is the default.
only | **cmns** | **xot**	Displays information about the specified services: X.25 only, CMNS, or XOT.
aodi	Causes the **debug x25** command to display Always On/Dynamic ISDN (AO/DI) events and processing information.
serial-interface	X.25 serial interface.
cmns-interface **mac** *mac-address*	CMNS interface and remote host's MAC address. The interface type can be Ethernet, Token Ring, or Fiber Distributed Data Interface (FDDI).
vc *number*	Virtual circuit number, in the range 1 to 4095.
remote *ip-address* [**port** *number*]	(Optional) Remote IP address and, optionally, a port number in the range 1 to 65535.
local *ip-address* [**port** *number*]	(Optional) Local host IP address and, optionally, a port number in the range 1 to 65535.

Default

all

Command History

Release	Modification
10.0	This command was introduced.
12.0(5)T	For DNS-based X.25 routing, additional functionality was added to the **debug x25 events** command to describe the events occurring while resolving the X.25 address to an IP address using a DNS server. The **debug domain** command can be used along with **debug x25 events** to observe the whole DNS-based X.25 routing data flow.

Usage Guidelines

This command is particularly useful for diagnosing problems encountered when placing calls. The **debug x25 all** output includes data, control messages, and flow control packets for all of the router's virtual circuits.

All **debug x25** command forms can take either the **events** or **all** keyword. The keyword **all** is the default and causes all packets meeting the other debug criteria to be reported. The keyword **events** omits reports of any Data or Receiver Ready (RR) flow control packets; the normal flow of Data and RR packets is commonly large and less interesting to the user, so event reporting can significantly decrease the processor load induced by debug reporting.

The **debug x25 interface** command is useful for diagnosing problems encountered with a single X.25 or CMNS host or virtual circuit.

Because no interface is specified by the **debug x25 vc** command, traffic on any virtual circuit that has the specified number is reported.

Virtual circuit zero (**vc 0**) cannot be specified. It is used for X.25 service messages, such as RESTART packets, not virtual circuit traffic. Service messages can be monitored only when no virtual circuit filter is used.

The **debug x25 xot** output allows you to restrict the debug output reporting to XOT traffic for one or both hosts or host/port combinations. Because each XOT virtual circuit uses a unique TCP connection, an XOT debug request that specifies both host addresses and ports will report traffic only for that virtual circuit. Also, you can restrict reporting to sessions initiated by the local or remote router by, respectively, specifying 1998 for the remote or local port. (XOT connections are received on port 1998.)

Use the **debug x25 aodi** command to display interface PPP events running over an X.25 session and to debug X.25 connections between a client and server configured for AO/DI.

Examples

The following is sample output from the **debug x25** command, displaying output concerning the functions. X.25 restart, call setup, data exchange, and clear:

```
Router# debug x25

Serial0: X.25 I R/Inactive Restart (5) 8 lci 0
  Cause 7, Diag 0 (Network operational/No additional information)
Serial0: X.25 O R3 Restart Confirm (3) 8 lci 0
Serial0: X.25 I P1 Call (15) 8 lci 1
From(6): 170091 To(6): 170090
   Facilities: (0)
   Call User Data (4): 0xCC000000 (ip)
Serial0: X.25 O P3 Call Confirm (3) 8 lci 1
Serial0: X.25 I D1 Data (103) 8 lci 1 PS 0 PR 0
Serial0: X.25 O D1 Data (103) 8 lci 1 PS 0 PR 1
Serial0: X.25 I P4 Clear (5) 8 lci 1
  Cause 9, Diag 122 (Out of order/Maintenance action)
Serial0: X.25 O P7 Clear Confirm (3) 8 lci 1
```

Table 9 describes the fields in the display.

Table 9 debug x25 Field Descriptions

Field	Description
Serial0	Interface on which the X.25 event occurred.
X.25	Type of event this message describes.
I	Letter indicating whether the X.25 packet was input (I) or output (O) through the interface.

Table 9 debug x25 Field Descriptions (continued)

Field	Description
R3	State of the service or virtual circuit (VC). Possible values follow: • R/Inactive—Packet layer awaiting link layer service • R1—Packet layer ready • R2—data terminal equipment (DTE) restart request • R3—data communications equipment (DCE) restart indication • P/Inactive—VC awaiting packet layer service • P1—Idle • P2—DTE waiting for DCE to connect CALL • P3—DCE waiting for DTE to accept CALL • P4—Data transfer • P5—CALL collision • P6—DTE clear request • P7—DCE clear indication • D/Inactive—VC awaiting setup • D1—Flow control ready • D2—DTE reset request • D3—DCE reset indication See Annex B of the 1984 ITU-T X.25 Recommendation for more information on these states.
Restart	The type of X.25 packet. Possible values follow: **R Events** • Restart • Restart Confirm • Diagnostic **P Events** • Call • Call Confirm • Clear • Clear Confirm **D Events** • Reset • Reset Confirm **D1 Events** • Data • RNR (Receiver Not Ready) • RR (Receiver Ready) • Interrupt • Interrupt Confirm **XOT Overhead** • PVC Setup

Table 9 debug x25 Field Descriptions (continued)

Field	Description
(5)	Number of bytes in the packet.
8	Modulo of the virtual circuit. Possible values are 8 or 128.
lci 0	Virtual circuit number. See Annex A of the X.25 Recommendation for information on virtual circuit assignment.
Cause 7	Code indicating the event that triggered the packet. The Cause field can only appear in entries for Clear, Reset, and Restart packets. Possible values for the cause field can vary, depending on the type of packet. Refer to the "X.25 Cause and Diagnostic Codes" appendix in the Cisco IOS Release 12.0 *Debug Command Reference* publication for an explanation of these codes.
Diag 0	Code providing an additional hint as to what, if anything, went wrong. The Diag field can only appear in entries for Clear, Diagnostic (as "error 0"), Reset and Restart packets. Refer to the "X.25 Cause and Diagnostic Codes" appendix for an explanation of these codes.
(Network operational/No additional information)	The standard explanations of the Cause and Diagnostic codes (*cause*/*diag*).

The following example shows a sequence of increasingly restrictive **debug x25** commands:

```
Router# debug x25
X.25 packet debugging is on

Router# debug x25 events
X.25 special event debugging is on

Router# debug x25 interface serial 0
X.25 packet debugging is on
X.25 debug output restricted to interface Serial0

Router# debug x25 vc 1024
X.25 packet debugging is on
X.25 debug output restricted to VC number 1024

Router# debug x25 interface serial 0 vc 1024
X.25 packet debugging is on
X.25 debug output restricted to interface Serial0
X.25 debug output restricted to VC number 1024

Router# debug x25 interface serial 0 vc 1024 events
X.25 special event debugging is on
X.25 debug output restricted to interface serial 0
X.25 debug output restricted to VC number 1024
```

The following examples show the normal sequence of events for both the AO/DI client and server sides:

Client Side

```
Router# debug x25 aodi
PPP-X25: Virtual-Access1: Initiating AODI call request
PPP-X25: Bringing UP X.25 AODI VC
PPP-X25: AODI Client Call Confirm Event Received
PPP-X25: Cloning interface for AODI is Di1
PPP-X25: Queuing AODI Client Map Event
PPP-X25: Event:AODI Client Map
```

```
PPP-X25: Created interface Vi2 for AODI service
PPP-X25: Attaching primary link Vi2 to Di1
PPP-X25: Cloning Vi2 for AODI service using Di1
PPP-X25: Vi2: Setting the PPP call direction as OUT
PPP-X25: Vi2: Setting vectors for RFC1598 operation on BRI3/0:0 VC 0
PPP-X25: Vi2: Setting the interface default bandwidth to 10 Kbps
PPP-X25: Virtual-Access2: Initiating AODI call request
PPP-X25: Bringing UP X.25 AODI VC
PPP-X25: AODI Client Call Confirm Event Received
```

Server Side

```
Router# debug x25 aodi
PPP-X25: AODI Call Request Event Received
PPP-X25: Event:AODI Incoming Call Request
PPP-X25: Created interface Vi1 for AODI service
PPP-X25: Attaching primary link Vi1 to Di1
PPP-X25: Cloning Vi1 for AODI service using Di1
PPP-X25: Vi1: Setting vectors for RFC1598 operation on BRI3/0:0 VC 1
PPP-X25: Vi1: Setting the interface default bandwidth to 10 Kbps
PPP-X25: Binding X.25 VC 1 on BRI3/0:0 to Vi1
```

debug x25 events for DNS-Based X.25 Routing

The following example of the **debug x25 events** command shows output related to the DNS-Based X.25 Routing feature. It shows messages concerning accessing the DNS server. In the following example, there are nine alternate addresses for one XOT path entered in the DNS server database. All nine addresses are returned to the router's host cache by the DNS server. However, only six addresses will be used during the XOT switch attempt, because this is the limit that XOT allows:

```
Router# debug x25 events
00:18:25:Serial1:X.25 I R1 Call (11) 8 lci 1024
00:18:25:  From (0): To (4):444
00:18:25:  Facilities:(0)
00:18:25:  Call User Data (4):0x01000000 (pad)
00:18:25:X.25 host name sent for DNS lookup is "444"
00:18:26:%3-TRUNCATE_ALT_XOT_DNS_DEST:Truncating excess XOT addresses (3)
returned by DNS
00:18:26:DNS got X.25 host mapping for "444" via network
00:18:32:[10.1.1.8 (pending)]:XOT open failed (Connection timed out; remote host not
responding)
00:18:38:[10.1.1.7 (pending)]:XOT open failed (Connection timed out; remote host not
responding)
00:18:44:[10.1.1.6 (pending)]:XOT open failed (Connection timed out; remote host not
responding)
00:18:50:[10.1.1.5 (pending)]:XOT open failed (Connection timed out; remote host not
responding)
00:18:56:[10.1.1.4 (pending)]:XOT open failed (Connection timed out; remote host not
responding)
00:20:04:[10.1.1.3,1998/10.1.1.3,11007]:XOT O P2 Call (17) 8 lci 1
00:20:04:  From (0): To (4):444
00:20:04:  Facilities:(6)
00:20:04:    Packet sizes:128 128
00:20:04:    Window sizes:2 2
00:20:04:  Call User Data (4):0x01000000 (pad)
00:20:04:[10.1.1.3,1998/10.1.1.3,11007]:XOT I P2 Call Confirm (11) 8 lci 1
00:20:04:  From (0): To (0):
00:20:04:  Facilities:(6)
00:20:04:    Packet sizes:128 128
00:20:04:    Window sizes:2 2
00:20:04:Serial1:X.25 O R1 Call Confirm (5) 8 lci 1024
00:20:04:  From (0): To (0):
00:20:04:  Facilities:(0)
```

Related Commands

Command	Description
debug ppp bap	Displays general Bandwidth Allocation Control Protocol (BACP) transactions.
debug ppp bap negotiation	Displays general BACP transactions, as well as successive steps in negotiations between peers.
debug ppp multilink	Displays information about important multilink events.

Glossary

CMNS—Connection-Mode Network Service. Extends local X.25 switching to a variety of media (Ethernet, FDDI, Token Ring).

DCE—Data communications equipment.

DLCI—Data link connection identifier.

DNS—Domain Name System. System used for translating network node names into IP addresses.

DTE—Data terminating equipment.

MAC—Media Access Control. A sublayer in the data link layer which handles access to shared media.

FDDI—Fiber Distributed Data Interface.

TCP—Transmission Control Protocol.

VC—Virtual circuit.

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

X.121—ITU-T standard describing a 14-digit addressing scheme used in X.25 networks (sometimes called IDNs)

XOT—X.25 over Transmission Control Protocol (TCP). Protocol frequently used to transport X.25 data over a WAN.