

# XNS Commands

---

The Xerox Network Systems (XNS) protocols, which were developed by the Xerox Corporation, are designed to be used across a variety of communication media, processors, and office applications. Ungermann-Bass, Inc. (now a part of Tandem Computers) adopted XNS in developing its Net/One XNS routing protocol. Standard XNS routing uses the routing information protocol (RIP) update packets and the hop count metric. Ungermann-Bass Net/One uses hello packets and a path-delay metric.

This chapter describes the commands to configure both standard Xerox Network System (XNS) routing and Ungermann-Bass Net/One XNS routing. You can also configure our routers to interoperate with Ungermann-Bass routers. For XNS configuration information and examples, refer to the “Configuring XNS” chapter in the *Router Products Configuration Guide*.

## access-list (standard)

To define a standard XNS access list, use the standard version of the **access-list** global configuration command. To remove a standard access list, use the **no** form of this command.

```
access-list access-list-number {deny | permit} source-network [.source-address
  [source-address-mask]] [destination-network] [.destination-address
  [destination-address-mask]]]
no access-list access-list-number
```

### Syntax Description

<i>access-list-number</i>	Number of the access list. This is a decimal number from 400 to 499.
<b>deny</b>	Denies access if the conditions are matched.
<b>permit</b>	Permits access if the conditions are matched.
<i>source-network</i>	Number of the network from which the packet is being sent. This is a 32-bit decimal number. You can omit leading zeros. A network number of -1 matches all networks.  Note that you enter the network number in decimal, and this number is expressed in decimal format in our configuration files and routing tables. However, the router internally converts the network number into hexadecimal. This means, for instance, that a network analyzer will display the network number in hexadecimal.
<i>source-address</i>	(Optional) Host on <i>source-network</i> from which the packet is being sent. This is a 48-bit hexadecimal value represented as a dotted triplet of four-digit hexadecimal numbers (xxxx.xxxx.xxxx).
<i>source-address-mask</i>	(Optional) Mask to be applied to <i>source-address</i> . This is a 48-bit value represented as a dotted triplet of four-digit hexadecimal numbers (xxxx.xxxx.xxxx). Place ones in the bit positions you want to mask.
<i>destination-network</i>	(Optional) Number of the network to which the packet is being sent. This is a 32-bit decimal number. A network number of -1 matches all networks.  You can omit leading from the network number.  Note that you enter the network number in decimal, and this number is expressed in decimal format in our configuration files and routing tables. However, the router internally converts the network number into hexadecimal. This means, for instance, that a network analyzer will display the network number in hexadecimal.

<i>destination-address</i>	(Optional) Host on <i>destination-network</i> to which the packet is being sent. This is a 48-bit hexadecimal value represented as a dotted triplet of four-digit hexadecimal numbers ( <i>xxx.xxx.xxx</i> ).
<i>destination-address-mask</i>	(Optional) Mask to be applied to <i>destination-address</i> . This is a 48-bit value represented as a dotted triplet of four-digit hexadecimal numbers ( <i>xxx.xxx.xxx</i> ). Place ones in the bit positions you want to mask.

## Default

None

## Command Mode

Global configuration

## Usage Guidelines

Standard XNS access lists filter on the source network only. All other parameters are optional. This means that you cannot use them to prevent traffic from going to or coming from specific hosts.

Use the **xns access-group** command to assign an access list to an interface. You can apply only one extended or one standard access list to an interface. The access list filters all outgoing packets on the interface.

## Examples

The following example denies access to packets from source network 1 that are destined for network 2. It permit all other traffic.

```
access-list 400 deny 1 2
access-list 400 permit -1 -1
```

The following example adds masks for the source and destination networks:

```
access-list 400 deny 1.0011.1622.0015 0000.0000.0000 2.301D3.020C.0022
0000.00ff.ffff
access-list 400 permit -1 0000.0000.0000 -1 0000.0000.0000
```

## Related Commands

**access-list (extended)**

**xns access-group**

**xns input-network-filter**

## access-list (extended)

To define an extended XNS access list, use the extended version of the **access-list** global configuration command. To remove an extended access list, use the **no** form of this command.

```
access-list access-list-number { deny | permit } protocol [source-network [source-host
[source-network-mask.]source-host-mask] source-socket
[destination-network [destination-host
[destination-network-mask.destination-host-mask] [destination-socket[/pep]]]]]
no access-list access-list-number { deny | permit } protocol [source-network [source-host
[source-network-mask.source-host-mask]]] source-socket
[destination-network [destination-host
[destination-network-mask.destination-host-mask] destination-socket[/pep]]]]]
```

---

**Note** If network masks are used, then all fields are required, except the destination socket and the destination Packet Exchange Protocol (PEP) type.

---

### Syntax Description

<i>access-list-number</i>	Number of the access list. This is a decimal number from 500 to 599.
<b>deny</b>	Denies access if the conditions are matched.
<b>permit</b>	Permits access if the conditions are matched.
<i>protocol</i>	Number of an XNS protocol, in decimal. See the documentation accompanying your host's XNS implementation for a list of protocol numbers.
<i>source-network</i>	(Optional) Number of the network from which the packet is being sent. This is a 32-bit decimal number. A network number of -1 matches all networks.  You can omit leading from the network number.  Note that you enter the network number in decimal, and this number is expressed in decimal format in our configuration files and routing tables. However, the router internally converts the network number into hexadecimal. This means, for instance, that a network analyzer will display the network number in hexadecimal.
<i>source-host</i>	(Optional) Host on <i>source-network</i> from which the packet is being sent. This is a 48-bit hexadecimal value represented as a dotted triplet of 4-digit hexadecimal numbers ( <i>xxxx.xxxx.xxxx</i> ).
<i>source-network-mask</i>	(Optional) Mask to be applied to <i>source-network</i> . The mask is a 32-bit decimal number. The mask must immediately be followed by a period, which must in turn immediately be followed by <i>source-host-mask</i> .

<i>source-host-mask</i>	(Optional) Mask to be applied to <i>source-host</i> . This is a 48-bit value represented as a dotted triplet of four-digit hexadecimal numbers ( <i>xxxx.xxxx.xxxx</i> ). Place ones in the bit positions you want to mask.
<i>source-socket</i>	Number of the socket from which the packet is being sent. This is a 16-bit decimal value. See the documentation accompanying your host's XNS implementation for a list of socket numbers.
<i>destination-network</i>	(Optional) Number of the network to which the packet is being sent. This is a 32-bit decimal number. A network number of -1 matches all networks.  You can omit leading from the network number.  Note that you enter the network number in decimal, and this number is expressed in decimal format in our configuration files and routing tables. However, the router internally converts the network number into hexadecimal. This means, for instance, that a network analyzer will display the network number in hexadecimal.
<i>destination-host</i>	(Optional) Host on <i>destination-network</i> to which the packet is being sent. This is a 48-bit hexadecimal value represented as a dotted triplet of four-digit hexadecimal numbers ( <i>xxxx.xxxx.xxxx</i> ).
<i>destination-network-mask</i>	(Optional) Mask to be applied to <i>destination-network</i> . The mask is a 32-bit decimal number. The mask must immediately be followed by a period, which must in turn immediately be followed by <i>destination-host-mask</i> .
<i>destination-host-mask</i>	(Optional) Mask to be applied to <i>destination-host</i> . This is a 48-bit value represented as a dotted triplet of four-digit hexadecimal numbers ( <i>xxxx.xxxx.xxxx</i> ). Place ones in the bit positions you want to mask.
<i>destination-socket</i>	(Optional) Number of the socket to which the packet is being sent. This is a 16-bit decimal value. See the documentation accompanying your host's XNS implementation for a list of socket numbers.
<i>pep</i>	(Optional) Packet Exchange Protocol type. PEP is a connectionless-oriented protocol that uses XNS Type 4 IDP frames.

**Default**

None

**Command Mode**

Global configuration

### Usage Guidelines

Extended XNS access lists filter on protocol type. All other parameters are optional.

Use the **xns access-group** command to assign an access list to an interface. You can apply only one extended or one standard access list to an interface. The access list filters all outgoing packets on the interface.

### Examples

The following example shows two ways to allow protocol type 20 on any socket (from a certain make of machine [Cisco Ethernet]) on network 10 to access any hosts on networks 1000 through 1015 on any socket.

In the following example, the access-list commands were issued without using the network mask option.

```
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1000 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1001 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1002 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1003 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1004 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1005 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1006 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1007 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1008 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1009 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1010 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1011 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1012 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1013 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1014 0
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0 1015 0
```

In the following example, the access-list command performs the same operation, but in a much shorter time, because the netmask option is used.

```
access-list 505 permit 20 10.0000.0C00.0000 0000.0000.FFFF 0
1000.0000.0000.0000 15.FFFF.FFFF.FFFF 0
```

### Related Commands

**access-list (standard)**

**xns access-group**

**xns input-network-filter**

## ping (user)

To check host reachability and network connectivity, use the user **ping** EXEC command.

**ping** *xns address*

### Syntax Description

**xns** Specifies the XNS protocol.

*address* Address of system to ping.

### Command Mode

EXEC

### Usage Guidelines

The user **ping** (packet internet groper function) command provides a basic ping facility for users who do not have system privileges. This command is equivalent to the nonverbose form of the privileged **ping** command. It sends five 100-byte ping packets.

The **ping** command works only on our network servers running Software Release 8.2 or later.

If the system cannot map an address for a host name, it will return an “%Unrecognized host or address” error message.

To abort a **ping** session, type the escape sequence. By default, this is Ctrl-^ X. You enter this by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, and then pressing the X key.

Table 21-1 describes the test characters displayed in **ping** responses.

**Table 21-1 Ping Test Characters**

Character	Meaning
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted the test.
?	Unknown packet type.
&	Packet lifetime exceeded.

### Sample Display

The following display shows sample output from the XNS **ping** command:

```
router> ping xns 1.0000.0c01.f4cf
Type escape sequence to abort.
Sending 5, 100-byte XNS Echoes to 1.0000.0c01.f4cf, timeout is 2 seconds:
!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/5/12 ms
```

Related Command  
**ping (privileged)**



## ping (privileged)

To check host reachability and network connectivity, use the **ping** privileged EXEC command.

### ping

#### Syntax Description

This command has no arguments or keywords.

#### Command Mode

Privileged EXEC

#### Usage Guidelines

The privileged **ping** (packet internet groper function) command provides a complete **ping** facility for users who have system privileges.

If the system cannot map an address for a host name, it will return an “%Unrecognized host or address” error message.

To abort a **ping** session, type the escape sequence. By default, this is Ctrl-^ X. You enter this by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, and then pressing the X key.

Table 21-2 describes the test characters displayed in **ping** responses.

**Table 21-2 Ping Test Characters**

Character	Meaning
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted the test.
?	Unknown packet type.
&	Packet lifetime exceeded.

#### Sample Display

The following display shows sample input to and output from the **ping** command:

```
Router# ping
Protocol [ip]: xns
Target XNS address: 2001.aa00.0400.6508
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Verbose [n]:n
Type escape sequence to abort.
Sending 5, 100-byte XNS Echos to 2001.aa00.0400.6508, timeout is 2 seconds:
Success rate is 100 percent, round-trip min/avg/max = 4/5/12 ms
```

## ping (privileged)

---

The following display shows sample input to and output from the XNS **ping** command in verbose mode. In this mode, the command shows the round-trip time, in milliseconds, for each XNS echo packet sent. In this display, the packets are labeled 0, 1, 2, 3, and 4.

```
Router# ping
Protocol [ip]: xns
Target XNS address: 2001.aa00.0400.6508
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Verbose [n]: y
Type escape sequence to abort.
Sending 5, 100-byte XNS Echos to 2001.aa00.0400.6508, timeout is 2 seconds:
0 in 12 ms
1 in 4 ms
2 in 4 ms
3 in 4 ms
4 in 4 ms
Success rate is 100 percent, round-trip min/avg/max = 4/5/12 ms
```

## show xns cache

To display the contents of the XNS fast-switching cache, use the **show xns cache** EXEC command.

**show xns cache**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Sample Display

The following is sample output from the **show xns cache** command:

```
Router# show xns cache

XNS routing cache version is 23

Destination          Interface      MAC Header
* 2.0000.00c0.1234    Ethernet1     000000C012340000C00D8DB0600
```

Table 21-3 describes the fields shown in the display.

**Table 21-3 Show XNS Cache Field Descriptions**

Field	Description
XNS routing cache version is 23	Number identifying the fast-switching cache table. It increments each time the table changes.
Destination	Destination network for this packet. Valid entries are marked by an asterisk (*).
Interface	Router interface through which this packet is transmitted.
MAC Header	First bytes of this packet's MAC header.

### Related Command

**xns route-cache**

## show xns interface

To display the status of the XNS interfaces configured in the router and the parameters configured on each interface, use the **show xns interface EXEC** command.

**show xns interface** [*type number*]

### Syntax Description

*type* (Optional) Interface type. It can be one of the following types: asynchronous, dialer, Ethernet (IEEE 802.3), loopback, null, serial, or tunnel.

*number* (Optional) Interface number.

### Command Mode

EXEC

### Sample Display

The following is sample output from the **show xns interface** command:

```
Router# show xns interface
Ethernet 0 is up, line protocol is up
XNS address is 60.0000.0c00.1d23
xns encapsulation is ARPA
Helper address is 912.ffff.ffff.ffff
Outgoing address list is not set
Input filter list is not set
Output filter list is not set
Router filter list is not set
Update timer is not set
XNS fast-switching enabled

Ethernet 1 is administratively down, line protocol is down
XNS protocol processing disabled

Serial 1 is up, line protocol is up
XNS protocol processing disabled
```

Table 21-4 describes the fields shown in the display.

**Table 21-4 Show XNS Interface Field Descriptions**

Field	Description
Ethernet 0 is up	Type of interface and whether it is currently active and inserted into the network (up) or inactive and not inserted (down).
line protocol is up	Indicates whether the software processes that handle the line protocol believe that the interface is usable (that is, whether keepalives are successful).
XNS address	Network and host number of the local router interface.
xns encapsulation	Type of encapsulation configured on the interface.

---

Field	Description
Helper address	Address of a target XNS server or network to which broadcast XNS packets are forwarded, as configured with the <b>xns helper-address</b> command.
Outgoing address list	Indicates whether an access list has been enabled with the <b>access-list</b> command.
Input filter list	Number of the input filter applied to the interface with the <b>xns input-network-filter</b> command.
Output filter list	Number of the output filter applied to the interface with the <b>xns output-network-filter</b> command.
Router filter list	Number of the router filter applied to the interface with the <b>xns router-filter</b> command.
Update timer	How often the router sends RIP updates, as configured with the <b>xns update-time</b> command.
XNS fast-switching	Indicates whether XNS fast switching is enabled (default) or disabled for this interface.
administratively down	Hardware has been taken down by an administrator.

#### Related Commands

**access-list (standard)**

**access-list (standard)**

**xns helper-address**

**xns input-network-filter**

**xns output-network-filter**

**xns router-filter**

**xns update-time**

## show xns route

To display the contents of the XNS routing table, use the **show xns route** EXEC command.

```
show xns route [network]
```

### Syntax Description

*network* (Optional) Number of the network that the route is to. This is a 32-bit decimal number. You can omit leading zeros.

### Command Mode

EXEC

### Sample Display

The following is sample output from the **show xns route** command:

```
Router# show xns route

Codes: R - RIP derived, C - connected, S - static, 1 learned routes

Maximum allowed path(s) are/is 1
C Net 14 is directly connected, 0 uses, Ethernet0
C Net 15 is directly connected, 0 uses, Ethernet1
R Net 16 [1/0] via 14.0000.0c00.3e3b, 10 sec, 0 uses, Ethernet0
```

Table 21-5 describes the fields shown in the display.

**Table 21-5 Show XNS Route Field Descriptions**

Field	Description
Codes	Codes defining how the route was learned.
R	Route learned from a RIP update.
C	Directly connected network.
S	Statically defined route via the <b>xns route</b> command.
learned routes	Number of routes learned from RIP updates.
Maximum allowed paths	Maximum number of paths for which the router has been configured with the <b>xns helper-address</b> command.
Net 14	XNS network number.
is directly connected	Indicates that the network is directly connected to the router.
uses	Fair estimate of the number of times a route gets used. It actually indicates the number of times the route has been selected for use prior to operations such as access list filtering.
Ethernet0	Possible interface through which you can reach the remote network via the specified router.
[1/0]	Delay/Metric. The delay is the delay between sending routing updates. The metric is the XNS metric used in making routing decisions.
via	Address of a router that is the next hop to the remote network.

<b>Field</b>	<b>Description</b>
sec	Number of seconds since information about this network was last heard.

Related Commands

**xns maximum-paths**

**xns route**

## show xns traffic

To display information about the number and type of XNS packets transmitted and received by the router, use the **show xns traffic EXEC** command.

### show xns traffic

#### Syntax Description

This command has no arguments or keywords.

#### Command Mode

EXEC

#### Sample Display

The following is sample output from the **show xns traffic** command:

```
Router# show xns traffic

Rec: 3968 total, 0 format errors, 0 checksum errors, 0 bad hop count,
    3968 local destination, 0 multicast
Bcast: 2912 received, 925 sent
Sent: 5923 generated, 500 forwarded, 0 encapsulation failed, 0 not routable
Errors: 10 received, 20 sent
Echo: Recd: 100 requests, 89 replies   Sent: 20 requests, 20 replies
Unknown: 5 packets
```

Table 21-6 describes significant fields shown in the display.

**Table 21-6 Show XNS Traffic Statistics Field Descriptions**

Field	Description
Rec:	Description of the XNS packets the router has received.
3968 total	Total number of packets the router has received.
0 format errors	Number of bad packets discarded (for example, packets with a corrupted header).
0 checksum errors	Number of packets discarded because they contained checksum errors.
0 bad hop count	Number of packets discarded because their hop count exceeded 16 (that is, the packets timed out).
3968 local destination	Number of packets sent to the local broadcast address or specifically to the router.
0 multicast	Number of packets received that were addressed to multiple destinations.
Bcast:	Number of broadcast packets received and sent.
Sent:	Description of the XNS packets the router has sent.
5923 generated	Number of packets the router transmitted that it generated itself.
500 forwarded	Number of packets the router transmitted that it forwarded from other sources.
0 encapsulation failed	Number of packets the router was unable to encapsulate.



---

<b>Field</b>	<b>Description</b>
0 not routable	Number of times the router could not locate in the routing table a route to the destination.
Errors:	Number of packets the router sent and received that contained errors.
Echo:	Number of ping packets the router received and sent, and the number of replies it received.
Unknown:	Number of packets the router was unable to forward, for example, because of a misconfigured helper address or because no route was available.

---

## xns access-group

To apply a generic filter to an interface, use the **xns access-group** interface configuration command. To remove the access list, use the **no** form of this command.

```
xns access-group access-list-number  
no xns access-group access-list-number
```

### Syntax Description

<i>access-list-number</i>	Number of the access list. All outgoing packets defined with either standard or extended access lists and forwarded through the interface are filtered by the entries in this access list. For standard access lists, <i>access-list-number</i> is a decimal number from 400 to 499. For extended access lists, <i>access-list-number</i> is a decimal number from 500 to 599.
---------------------------	--

### Default

None

### Command Mode

Interface configuration

### Usage Guidelines

The **xns access-group** command applies a generic filter to an interface. These filters control which packets are sent out an interface based on the packet's source and destination addresses, XNS protocol type, and source and destination socket numbers. You use the **access-list (standard)** and **access-list (extended)** commands to specify the filtering conditions.

You can apply only one filter to an interface.

### Example

In the following example, the access list 500 is applied to Ethernet interface 0.

```
interface ethernet 0  
xns access-group 500
```

### Related Commands

**access-list (standard)**  
**access-list (extended)**

## xns encapsulation

To select the type of encapsulation used on a Token Ring interface, use the **xns encapsulation** interface configuration command. To disable the encapsulation, use the **no** form of this command.

```
xns encapsulation {snap | ub | 3com}
no xns encapsulation {snap | ub | 3com}
```

### Syntax Description

<b>snap</b>	802.2 LLC encapsulation. This is the default encapsulation type. Use this encapsulation type with IBM Token Ring networks.
<b>ub</b>	Ungermann-Bass encapsulation.
<b>3com</b>	3Com encapsulation. Use this encapsulation type when older 3Com Corporation products are present on the network.

### Default

**snap**

### Command Mode

Interface configuration

### Usage Guidelines

You must specify this command on an interface if you want a Token Ring interface to forward XNS packets.

Some 3Com 3+ hosts do not recognize Token Ring packets with the source-route bridging RIF field set. You can work around this discrepancy by using the **no multiring xns** interface configuration command on Token Ring interfaces that are used for 3Com XNS routing. See the “Source-Route Bridging Commands” chapter for more information.

### Example

In the following example, Ungermann-Bass-style encapsulation is used when forwarding Token Ring packets across an interface:

```
interface tokenring 0
xns network 23
xns encapsulation ub
```

### Related Command

A dagger (†) indicates that the command is documented in another chapter.

```
multiring xns †
xns hear-rip
xns ub-emulation
```

## xns flood broadcast allnets

To flood broadcast packets whose destination address is `-1.FFFF.FFFF.FFFF`, use the **xns flood broadcast allnets** interface configuration command. To disable this type of flooding, use the **no** form of this command.

**xns flood broadcast allnets**  
**no xns flood broadcast allnets**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

The **xns flood broadcast allnets** command configures all-nets flooding. In this type of routing, all broadcast packets are sent to all networks (as indicated by the network address `-1`) and to all host on those networks (as indicated by the host address `FFFF.FFFF.FFFF`).

You can specify a network address of `-1` only with the **xns flood broadcast** commands. In all other commands, it is an invalid address.

Flooding is applied to the packets received on an interface.

It is most closely in accordance with the XNS specification to flood packets with destinations of `-1.FFFF.FFFF.FFFF` and destinations of `-1.specific-host`, but not to flood packets with destinations of `0.FFFF.FFFF.FFFF`.

### Example

The following example configures the interface to flood broadcast packets:

```
interface ethernet 0
xns network 20
xns broadcast allnets
```

### Related Commands

**xns flood broadcast net-zero**  
**xns hear-rip**

## xns flood broadcast net-zero

To flood packets whose destinations address is 0.FFFF.FFFF.FFFF, use the **xns flood broadcast net-zero** interface configuration command. To disable this type of flooding, use the **no** form of this command.

**xns flood broadcast net-zero**  
**no xns flood broadcast net-zero**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

The **xns flood broadcast net-zero** command sends broadcast packets to all hosts (as indicated by the host address FFFF.FFFF.FFFF) on the local network (as indicated by the network address 0). This broadcast configuration is required in some 3Com environments.

Flooding is applied to the packets received on an interface.

### Example

In the following example, broadcast packets destined for the local network are sent to all hosts on that network:

```
interface Ethernet 0
xns network 20
xns flood broadcast net-zero
```

### Related Commands

**xns flood broadcast allnets**  
**xns flood specific allnets**  
**xns hear-rip**

## xns flood specific allnets

To flood packets whose destination address is *-1.specific-host*, use the **xns flood specific allnets** interface configuration command. To disable this type of flooding, use the **no** form of this command.

**xns flood specific allnets**  
**no xns flood specific allnets**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

The **xns flood specific allnets** command forwards broadcast packets as MAC-layer broadcasts so that they can be picked up and further flooded by other routers.

You can specify a network address of *-1* only with the **xns flood broadcast** commands. In all other commands, it is an invalid address.

Flooding is applied to the packets received on an interface.

It is most closely in accordance with the XNS specification to flood packets with destinations of *-1.FFFF.FFFF.FFFF* and destinations of *-1.specific-host*, but not to flood packets with destinations of *0.FFFF.FFFF.FFFF*.

### Example

In the following example, packets with destinations of *-1.specific-host* are flooded:

```
interface Ethernet 0
xns network 20
xns flood broadcast specific allnets
```

### Related Commands

**xns flood broadcast allnets**  
**xns flood broadcast net-zero**  
**xns hear-rip**

## xns forward-protocol

To forward packets of a specific XNS protocol to a helper address, use the **xns forward-protocol** global configuration command. To disable the forwarding of these packets, use the **no** form of this command.

**xns forward-protocol** *protocol*  
**no xns forward-protocol** *protocol*

### Syntax Description

*protocol*

**Number of an XNS protocol, in decimal.** See the documentation accompanying your host's XNS implementation for a list of protocol numbers.

### Default

Disabled

### Command Mode

Global configuration

### Example

In the following example, packets of protocol type 2 are forwarded to the specified helper address:

```
xns forward-protocol 2
interface ethernet 0
xns helper-address 26.FFFF.FFFF.FFFF
```

### Related Command

**xns helper-address**

## xns hear-rip

To receive RIP updates, use the **xns hear-rip** interface configuration command. To disable the receipt of RIP updates, use the **no** form of this command.

```
xns hear-rip [access-list-number]  
no xns hear-rip
```

### Syntax Description

*access-list-number*

(Optional) Number of the access list. This list defines the routes the router is to learn through standard RIP. The list is applied to individual routes within the RIP packet, not to the address of the packet's sender. For standard access lists, *access-list-number* is a decimal number from 400 to 499. For extended access lists, *access-list-number* is a decimal number from 500 to 599.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

Delay metrics are computed as if each hop mentioned by the RIP update were a 9.6-kbps serial link. Ordinarily, the result is that our router prefers an all-Ungermann-Bass path over an all-RIP path. If you want the router to learn only certain routes through standard RIP, specify an access list number as an argument to the **xns hear-rip** command. The router will then learn from RIP packets only routes to networks permitted by the access list. Note that the access list is applied to individual routes within the RIP packet, not to the address of the packet's sender.

In an Ungermann-Bass environment, you should configure all interfaces with the **xns flood broadcast allnets** and **xns flood specific allnets** commands. You should *not* configure them with the **xns flood broadcast net-zero** command. You should configure Token Ring interfaces that are directly connected to Ungermann-Bass nodes with the **xns encapsulation ub** command.

### Example

In the following example, serial interface 0 receives RIP updates:

```
interface serial 0  
xns network 24  
xns hear-rip
```

### Related Commands

**xns encapsulation ub**  
**xns flood broadcast allnets**  
**xns flood broadcast net-zero**  
**xns flood specific allnets**  
**xns ub-emulation**



## xns helper-address

To forward broadcast packets to a specified server, use the **xns helper-address** interface configuration command. To disable this function, use the **no** form of this command.

```
xns helper-address network.host  
no xns helper-address network.host
```

### Syntax Description

<i>network</i>	Network on which the target XNS server resides. This is a 32-bit decimal number.
<i>host</i>	Host number of the target XNS server. This is a 48-bit hexadecimal value represented as a dotted triplet of four-digit hexadecimal numbers ( <i>xxxx.xxxx.xxxx</i> ). The host must be directly connected to one of the router's directly attached networks. A number of FFFF.FFFF.FFFF indicates all hosts on the specified network.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

Routers normally block all broadcast requests and do not forward them to other network segments. This is done to prevent the degradation of performance over the entire network. The **xns helper-address** command allows broadcasts to be forwarded to other networks. This is useful when a network segment does not have a server capable of handling broadcasts. This command lets you forward the broadcasts to a server, network, or networks that can process them. Incoming unrecognized broadcast packets that match the access list created with the **xns helper-list** command, if it is present, are forwarded.

When a packet is helped, the router changes its destination address to be the configured helper address, and the packet is routed toward that address. The host at the helper address is expected to process the packet and (usually) to reply to the packet's sender. A helper address can be a directed broadcast address, in which case the helped packet will be forwarded to a remote network and rebroadcast there.

You can specify multiple **xns helper-address** commands on a given interface.

### Example

In the following example, the server at address 0000.0c00.23fe receives all broadcasts on network 51:

```
xns helper-address 51.0000.0c00.23fe
```

Related Command  
**xns forward-protocol**

## xns input-network-filter

To control which networks are added to the routing table, use the **xns input-network-filter** interface configuration command. To remove the filter from the interface, use the **no** form of this command.

```
xns input-network-filter access-list-number  
no xns input-network-filter access-list-number
```

### Syntax Description

*access-list-number*                      Number of the access list. All incoming packets defined with either standard or extended access lists are filtered by the entries in this access list. For standard access lists, *access-list-number* is a decimal number from 400 to 499. For extended access lists, it is a decimal number from 500 to 599.

### Default

None

### Command Mode

Interface configuration

### Usage Guidelines

The **xns input-network-filter** command controls which networks are added to the routing table based on the networks learned in incoming XNS routing updates (RIP updates) on the interface.

You can issue only one **xns input-network-filter** command on each interface.

### Example

In the following example, access list 476 controls which networks are added to the routing table when RIP packets are received on Ethernet interface 1. Network 16 is the only network whose information will be added to the routing table. Routing updates for all other networks are implicitly denied and are not added to the routing table.

```
access-list 476 permit 16  
interface ethernet 1  
xns input-network-filter 476
```

### Related Commands

**access-list (standard)**

**access-list (extended)**

**xns output-network-filter**

## xns maximum-paths

To set the maximum number of paths the router uses when sending packets, use the **xns maximum-paths** global configuration command. To restore the default value, use the **no** form of this command.

```
xns maximum-paths number  
no xns maximum-paths
```

### Syntax Description

*number* Maximum number of equal-cost paths from which the router chooses. It can be a number from 1 to 512. The default is 1.

### Default

1 path

### Command Mode

Global configuration

### Usage Guidelines

A router can use multiple paths to reach an XNS destination in order to increase throughput in the network. By default, the router will pick one best path and send all traffic on this path, but you can configure it to remember two or more paths that have equal costs (the cost metric is hop count for standard XNS RIP) and to balance the traffic load across all the available paths. (Note that when paths have differing costs, the router chooses lower-cost routes in preference to higher-cost routes.) Packets are distributed over the multiple paths in round-robin fashion on a packet-by-packet basis. That is, the first packet is sent along the first path, the second packet along the second path, and so on. If the final path is reached before all packets are sent, the next packet is sent to the first path, the next to the second path, and so on.

Limiting the number of equal-cost paths can save memory on routers with limited memory or very large configurations. Additionally, in networks with a large number of multiple paths and systems with limited ability to cache out-of-sequence packets, performance might suffer when traffic is split between many paths.

### Example

In the following example, the router uses up to two alternate paths:

```
xns maximum-paths 2
```

### Related Command

**show xns route**

## xns network

To enable XNS routing on a particular interface by assigning a network number to the interface, use the **xns network** interface configuration command. To disable XNS routing on an interface, use the **no** form of this command.

**xns network** *number*  
**no xns network**

### Syntax Description

*number* Network number. This is a 32-bit decimal number. You can omit leading zeros.

### Default

XNS routing disabled

### Command Mode

Interface configuration

### Usage Guidelines

Interfaces not enabled to run XNS ignore any XNS packets that they receive.

Every XNS interface must have a unique XNS network number.

### Example

The following example enables XNS routing, specifying that XNS networks 20 and 21 are connected to two of the router's Ethernet interfaces:

```
xns routing
interface ethernet 0
xns network 20
interface ethernet 1
xns network 21
```

### Related Commands

**xns routing**  
**show xns interface**

## xns output-network-filter

To control the list of networks included in routing updates sent out an interface, use the **xns output-network-filter** interface configuration command. To remove the filter from the interface, use the **no** form of this command.

```
xns output-network-filter access-list-number  
no xns output-network-filter access-list-number
```

### Syntax Description

<i>access-list-number</i>	Number of the access list. All outgoing packets defined with either standard or extended access lists are filtered by the entries in this access list. For standard access lists, <i>access-list-number</i> is a decimal number from 400 to 499. For extended access lists, it is a decimal number from 500 to 599.
---------------------------	---

### Default

None

### Command Mode

Interface configuration

### Usage Guidelines

The **xns output-network-filter** command controls which networks the router advertises in its routing updates (RIP updates).

You can issue only one **xns output-network-filter** command on each interface.

### Example

In the following example, access list 496 controls which networks are specified in routing updates sent out the serial 1 interface. This configuration causes network 27 to be the only network advertised in routing updates sent on the defined serial interface.

```
access-list 496 permit 27  
interface serial 1  
xns output-network-filter 496
```

### Related Commands

**xns input-network-filter**  
**xns router-filter**

## xns route

To add a static route to the XNS routing table, use the **xns route** global configuration command. To remove a route from the routing table, use the **no** form of this command.

```
xns route network network.host  
no xns route network network.host
```

### Syntax Description

<i>network</i>	Network to which you want to establish a static route. This is a 32-bit decimal number. You can omit leading zeros.
<i>network.host</i>	Router to which to forward packets destined for the specified network.  The argument <i>network</i> is a 32-bit decimal number. You can omit leading zeros.  The argument <i>host</i> is the host number of the target router. This is a 48-bit value represented by a dotted triplet of four-digit hexadecimal numbers ( <i>xxxx.xxxx.xxxx</i> ).

### Default

None

### Command Mode

Global configuration

### Usage Guidelines

The **xns route** command forwards packets destined for the specified network (*network*) to the specified router (*network.host*), regardless of whether that router is sending dynamic routing information.

Static routes usually are not used in XNS environments, because nearly all XNS routers support dynamic routing via RIP. Dynamic routing is enabled by default in our routers.

Be careful when assigning static routes. When links associated with static routes are lost, traffic may stop being forwarded, even though alternative paths might be available.

### Example

In the following example, the router at address 21.0456.acd3.1243 handles all traffic destined for network 25:

```
xns routing  
xns route 25 21.0456.acd3.1243
```

### Related Command

**show xns route**

## xns route-cache

To enable XNS fast switching, use the **xns route-cache** interface configuration command. To disable fast switching, use the **no** form of this command.

**xns route-cache**  
**no xns route-cache**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Interface configuration

### Usage Guidelines

XNS fast switching allows higher throughput by switching packets using a cache created by previous transit packets.

You might want to disable fast switching in two situations. One is if you want to save memory on the interface cards: fast-switching caches require more memory than those used for standard switching. The second situation is to avoid congestion on interface cards when a high-bandwidth interface is writing large amounts of information to a low-bandwidth interface.

### Example

In the following example, XNS fast switching is disabled for interface serial 1:

```
interface serial 1
no xns route-cache
```

### Related Command

**show xns cache**



## xns router-filter

To control the routers from which packets are accepted, use the **xns router-filter** interface configuration command. To remove the filters from the interface, use the **no** form of this command.

```
xns router-filter access-list-number  
no xns router-filter access-list-number
```

### Syntax Description

*access-list-number*      Number of the access list. All incoming packets defined with either standard or extended access lists are filtered by the entries in this access list. For standard access lists, *access-list-number* is a decimal number from 400 to 499. For extended access lists, it is a decimal number from 500 to 599.

### Default

None

### Command Mode

Interface configuration

### Usage Guidelines

You can issue only one **xns router-filter** command on each interface.

### Example

In the following example, access list 466 controls the routers from which packets are accepted. For serial interface 0, only packets from the router at 26.0000.00c0.047d are accepted. All other packets are implicitly denied.

```
access-list 466 permit 26.0000.00c0.047d  
interface serial 0  
xns router-filter 466
```

### Related Commands

```
xns input-network-filter  
xns output-network-filter
```

## xns routing

To enable XNS routing, use the **xns routing** global configuration command. To disable XNS routing, use the **no** form of this command.

```
xns routing [address]  
no xns routing
```

### Syntax Description

*address*

(Optional) Host number of the router. This is a 48-bit value represented by a dotted triplet of four-digit hexadecimal numbers (*xxxx.xxxx.xxxx*). It must not be a multicast address.

If you omit *address*, the router uses the address of the first IEEE-compliant (Token Ring, FDDI, or Ethernet) interface MAC address it finds in its interface list. The router uses the address 0123.4567.abcd for non-IEEE-compliant interfaces.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

The **xns routing** command enables the RIP service on the router.

### Example

The following example enables XNS routing on the router:

```
xns routing
```

### Related Command

**xns network**

## xns ub-emulation

To enable Ungermann-Bass Net/One routing, use the **xns ub-emulation** global configuration command. To disable Net/One routing and restore standard routing mode, use the **no** form of this command.

```
xns ub-emulation  
no xns ub-emulation
```

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

The **xns ub-emulation** command enables Ungermann-Bass Net/One routing. This means that Hello packets and routing updates on all XNS interfaces are sent out in Ungermann-Bass format.

Net/One is a distance-vector, or Bellman-Ford, protocol, similar to standard XNS RIP. The major difference between the two protocols is in the metrics used. Standard XNS RIP uses a hop count to determine the best route to distant networks and maintains information only about hop counts. The Ungermann-Bass protocol uses a path-delay metric and maintains information about both hop counts and its own metrics.

Ungermann-Bass routers generate standard RIP updates by extracting the hop-count values from the Ungermann-Bass routing protocol. When configured in Ungermann-Bass emulation mode, routers participate in this protocol and behave (insofar as routing protocols are concerned) like Ungermann-Bass routers.

You can use the **xns hear-rip** command to configure routers to listen to standard RIP updates when in Ungermann-Bass emulation mode. When our router in Ungermann-Bass emulation mode receives a RIP packet, each route in that packet is treated as though it had come from an Ungermann-Bass routing packet. The hop count used is the actual hop count from the RIP packet. The delay metric used is computed by assuming that each hop is the longest-delay link used by Ungermann-Bass, which is a 9.6-kbps serial link. Information from RIP packets is used in creating outgoing Ungermann-Bass updates, and vice versa.

To enable Ungermann-Bass routing on routers running Software Release 8.3 or earlier, use the **xns ub-routing** global configuration command. This command has been retained for backward compatibility with earlier software releases. To disable Ungermann-Bass routing, use the **no** form of this command.

```
xns ub-routing  
no xns ub-routing
```

This command is never written to nonvolatile configuration memory. Instead, the equivalent individual commands are written. These are an **xns ub-emulation** command for the router, and **xns hear-rip**, **xns flood broadcast allnets**, **no xns flood broadcast net-zero**, and **xns flood** specific allnets commands for all interfaces on which XNS is enabled. The **xns ub-routing** command does not modify the encapsulation used on Token Ring interfaces.

Older versions of our software implemented a restricted version of the Ungermann-Bass routing protocol, and in certain configurations could create routing instability and forwarding loops. Before using Software Releases 8.3 and earlier in Ungermann-Bass environments, consult the 8.3 documentation for information about these restrictions.

### Example

The following example enables Net/One routing:

```
xns routing
xns ub-emulation
interface tokenring 0
xns network 23
xns encapsulation ub
```

### Related Commands

**xns encapsulation ub**

**xns hear-rip**

## xns update-time

To set the XNS routing update timers, use the **xns update-time** interface configuration command. To restore the default value, use the **no** form of this command.

```
xns update-time interval  
no xns update-time
```

### Syntax Description

*interval* Interval, in seconds, at which XNS routing updates are sent. The minimum interval is 10 seconds, and the maximum is 2,493,644 seconds, which is about 29 days. The default is 30 seconds.

### Default

30 seconds

### Command Mode

Interface configuration

### Usage Guidelines

The **xns update-time** command sets the routing update timer on a per-interface basis. To display the current value, use the **show xns route** command.

Routers exchange information about routes by sending broadcast messages when they are brought up and shut down, and periodically while they are running. The **xns update-time** command lets you modify the periodic update interval.

You can set RIP timers only in a configuration in which all routers are our routers. The timers should be the same for all routers connected to the network.

The update value you choose affects the internal XNS timers as follows:

- XNS routes are marked invalid and placed in holddown if no routing updates for those routes are heard within three times the value of the update interval ( $3 \times interval$ ).
- XNS routes are removed from the routing table if no routing updates are heard within six times the value of the update interval ( $6 \times interval$ ).

This command has no effect on the Ungermann-Bass routing protocol.

### Example

The following example sets the routing update time to 20 seconds:

```
interface ethernet 0  
  xns network 20  
  xns update-time 20
```

### Related Command

**show xns route**

