CHAPTER 16

Banyan VINES Commands

The Banyan VINES protocol is a networking system for personal computers. "VINES" is an acronym for Virtual Network System. This proprietary protocol was developed by Banyan and is derived from Xerox's XNS protocol. Cisco's implementation of VINES has been designed in conjunction with Banyan.

Cisco's implementation of Banyan VINES provides routing of VINES packets on all media types. Although the software automatically determines a metric value that it uses to route updates based on the delay set for the interface, Cisco's software implementation allows you to customize the metric. Cisco's implementation also offers address resolution to respond to address requests. MAC-level echo support is also available for Ethernet, IEEE 802.2, Token Ring, and FDDI media. Name-to-address mapping for VINES host names is also supported, as are access lists to filter outgoing packets.

Use the commands in this chapter to configure and monitor VINES networks. For VINES configuration information and examples, refer to the "Configuring Banyan VINES" chapter in the *Router Products Configuration Guide*.

clear vines cache

To delete entries from the VINES fast-switching cache, use the **clear vines cache** EXEC command.

clear vines cache [**interface** *interface* | **neighbor** *address* | **server** *network*]

Syntax Description

interface *interface*	(Optional) Deletes from the fast-switching cache table any entry that has one or more paths that go through the specified interface.
neighbor *address*	(Optional) Deletes from the fast-switching cache table any entry that has one or more paths via the specified neighbor router.
server *network*	(Optional) Deletes from the fast-switching cache table any entry whose network number part of the destination address matches the specified network address. The argument *network* can be either a 4-byte hexadecimal number or a 4-byte decimal number (if you have issued a **vines decimal** command).

Command Mode

EXEC

Usage Guidelines

If you do not specify any keywords or arguments, all entries in the fast-switch cache are deleted.

The fast-switching cache is a table of routes used when fast switching is enabled.

Examples

The following example deletes all entries from the VINES fast-switching cache table:

```
clear vines cache
```

The following example deletes all entries whose destination server has the address 30002E6D:

```
clear vines cache server 30002E6D
```

Related Commands

show vines cache
vines decimal
vines route-cache

clear vines ipc

To delete VINES IPC connection blocks from the router, use the **clear vines ipc** EXEC command.

clear vines ipc *number*

Syntax Description

number	Hexadecimal number of the IPC connection to delete.

Command Mode

EXEC

Usage Guidelines

An IPC connection entry is built each time the router initiates or receives an IPC DATA message from a router that is not already in this table.

Examples

The following example deletes IPC connection 0x1D from the table of VINES IPC connections:

```
clear vines ipc 1D
```

Related Command

show vines ipc

clear vines neighbor

To delete entries from the neighbor table, use the **clear vines neighbor** EXEC command.

clear vines neighbor {*network* | *****}

Syntax Description

network	Network number of the neighbor whose entry should be deleted from the neighbor table. The argument *network* can be either a 4-byte hexadecimal number or a 4-byte decimal number (if you have issued a **vines decimal** command).
*	Deletes all entries from the neighbor path table except the entry for the local router.

Command Mode

EXEC

Usage Guidelines

The neighbor table contains an entry for each of the router's neighbor nodes.

Deleting an entry from the neighbor table also deletes any routes in the routing table that have that neighbor as the first hop and all fast-switching cache entries that have that neighbor as the first hop in any of their paths.

Example

The following example deletes all entries from the neighbor table:

```
clear vines neighbor *
```

Related Commands

clear vines route
show vines neighbor
show vines route
vines decimal
vines neighbor
vines route

clear vines route

To delete network addresses from the routing table, use the **clear vines route** EXEC command.

clear vines route {*network* | *****}

Syntax Description

network	Network number of the entry to delete from the routing table. The argument *network* can be either a 4-byte hexadecimal number, a 4-byte decimal number (if you have issued a **vines decimal** command), or a host name (if you have issued a **vines host** command).
*****	Deletes all entries from the routing table.

Command Mode

EXEC

Usage Guidelines

Deleting an entry from the routing table with the **clear vines route** command also deletes any entries in the fast-switching table that are a part of that logical network.

Example

The following example deletes all entries from the VINES routing table:

```
clear vines route *
```

Related Commands

clear vines route
show vines neighbor
show vines route
vines decimal
vines host
vines route

clear vines traffic

To clear all VINES-related statistics that are displayed by the **show vines traffic** command, use the **clear vines traffic** EXEC command.

clear vines traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **clear vines traffic** command clears only the statistics displayed by the **show vines traffic** command. It has no effect on the value of the VINES counters retrieved by SNMP.

Example

The following example zeroes
all VINES-related traffic statistics:

```
clear vines traffic
```

Related Command

show vines traffic

ping

To determine basic network connectivity, use the **ping** EXEC command.

ping [**vines**] [*address*]

Syntax Description

vines	(Optional) Specifies the VINES protocol. If you omit this keyword, the router prompts for it.
address	(Optional) Address of system to ping. If you omit the address, the router prompts for it.

Command Mode

EXEC

Usage Guidelines

The **ping** command determines network connectivity by sending datagrams to another host on the network.

Sample Displays

The following is sample output from the **ping** command:

```
Router# ping vines 27AF92:1

Type escape sequence to abort.
Sending 5, 100-byte VINES Echos to 27AF92:1,
timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/7/8 ms

Router# ping

Protocol [ip]: vines
Target VINES address: 27AF92:1
Repeat count [5]: 10
Datagram size [100]: 500
Timeout in seconds [2]:
Verbose [n]:
Type escape sequence to abort.
Sending 10, 500-byte VINES Echos to 27AF92:1,
timeout is 2 seconds:
!!!!!!!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/7/8 ms
```

show vines access

To display the VINES access lists currently defined, use the **show vines access** EXEC command.

show vines access [*access-list-number*]

Syntax Description

access-list-number	(Optional) Number of the access list to display.

Command Mode

EXEC

Usage Guidelines

If no access list number is specified, all access lists are displayed.

Sample Display

The following is sample output from the **show vines access** command:

```
Router# show vines access

Vines access list 1
  deny   SPP 30015800:0001 00000000:00000000 202 00123456:8005 00000000:0000 249
  permit IP 00000000:0000 FFFFFFFF:FFFF 00000000:0000 FFFFFFFF:FFFF
Vines access list 101
  deny   SPP 00112233:0001 00000000:0000 0006 0000
             00123456:8005 00000000:00000000 0000 FFFF
  permit IP 00000000:0000 FFFFFFFF:FFFF 00000000:0000 FFFFFFFF:FFFF
```

Table 16-1 describes the fields shown in the display.

Table 16-1 Show VINES Access Field Descriptions

Field	Description
Vines access list ...	Number of the VINES access list.
deny	Networks to which access is denied.
permit	Networks to which access is permitted.

Related Commands

vines access-list (extended)
vines access-list (simple)
vines access-list (standard)

show vines cache

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

show vines cache [*address* | **interface** *type number* | **neighbor** *address* | **server** *network*]

Syntax Description

address	(Optional) Displays the entry in the fast-switching cache for the specified station.
interface *type number*	(Optional) Displays all neighbors in the fast-switching cache that are accessible via the specified interface type and number.
neighbor *address*	(Optional) Displays all routes in the VINES fast-switching cache that have the specified neighbor as their first hop. The argument *address* is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes, a 4-byte decimal number in the same format (if you have issued a **vines decimal** command), or a host name (if you have issued a **vines host** command).
server *network*	(Optional) Displays all entries in the VINES fast-switching cache that are in the specified logical network. The argument *network* can be either a 4-byte hexadecimal number or a 4-byte decimal number (if you have issued a **vines decimal** command).

Command Mode

EXEC

Usage Guidelines

If no keywords or arguments are specified, all entries in the fast-switching cache are displayed.

Sample Display

The following is sample output from **show vines cache** command. This sample shows all entries in the VINES fast-switching cache.

```
Router# show vines cache

VINES fast switching cache information:
  Current: 0 entries, 0 paths
  History:
      Added:          0 server,          0 router,          0 client
    Updated:          0 server,          0 router,          0 client
    Expired:          0 server,          0 router,          0 client
    Removed:          0 server,          0 router,          0 client
    Flushes:          4 by neighbor,           1 by server
                      8 by interface,          1 entire table

Hash  Destination     Int    Age  Length  Type  MAC Header
13/00 Router1        *T0      46   16/18  1     10005A746A3600003080FB06BCBC03BA
27/00 Router2         E1      11   14/14  1     00000C01D87C00000C0158010BAD
```

```
                      *T0      11   16/18  1     0000300043550000 3080FB06BCBC03BA
3E/00 Router3         *T0      42   16/18  1     10005A6FBC15000003080FB06BCBC03BA
72/00 30002E6D:0001    E1      32   14/14  1     00000C01D87C00000C0158010BAD
                      *T0      32   16/18  1     0000300043550000 3080FB07BCBC03BA
                       T0      32   16/18  1     10005A6FBC1500003080FB06BCBC03BA
                       T0      32   16/18  1     10005A6FBC1500003080FB06BCBC03BA
FE/00 Router4         *E2     264   14/14  1     00000C0124EA00000C0151AF0BAD
```

Table 16-2 describes fields shown in the display.

Note that neighbor information is not explicitly displayed by the **show vines cache** command. However, you can determine it by looking at the neighbor and routing tables (using the **show vines neighbor** and **show vines route** commands, respectively).

Table 16-2 Show VINES Cache Field Descriptions

Field	Description
Current:	The number of entries and paths currently in the cache.
History:	Number of events since the last time the counters were cleared.
Added:	Number of server, router, and client entries added to the cache.
Updated:	Number of server, router, and client entry updates.
Expired:	Number of server, router, and client entries that timed out.
Removed:	Number of server, router, and client entries removed from the cache.
Flushes:	Number of neighbor, server, interface, and entire table flushes.
Hash	Position of this entry in the neighbor table.
Destination	Name or address of the destination station.
Int	Interface out which the packet is sent. An asterisk preceding the interface name indicates that this is the next entry that will be used for the destination.
Age	Age of the entry, in seconds.
Length	Stored length of the packet's MAC header, followed by a slash and the actual length of the MAC header. Both lengths do not include the length of the Type field. These two lengths may differ because the initial bytes of Token Ring and FDDI frames are not stored.
Type	Local encapsulation type.
MAC Header	MAC header used to reach the destination.

Related Commands

clear vines route
show vines neighbor
show vines route
vines decimal
vines route-cache

show vines host

To display the entries in the VINES host name table, use the **show vines host** EXEC command.

show vines host [*name*]

Syntax Description

name	(Optional) Displays the entry in the VINES name table that has the specified name.

Command Mode

EXEC

Usage Guidelines

If no name is specified, all entries in the host name table are displayed.

Sample Display

The following is sample output from the **show vines host** command:

```
Router# show vines host

Name          Address
Router1       0027AF9A:0001
Router2       0027D0E4:0001
Router3       002ABFAA:0001
Router4       30015800:0001
```

Table 16-3 describes the fields shown in the display.

Table 16-3 Show VINES Host Field Descriptions

Field	Description
Name	Name of the VINES host.
Address	Address of the VINES host.

Related Command

vines host

show vines interface

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

show vines interface [*type number*]

Syntax Description

type	(Optional) Interface type.
number	(Optional) Interface number.

Command Mode

EXEC

Usage Guidelines

If you omit all keywords, this command displays values for all interfaces, and displays all VINES global parameters.

Sample Display

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

```
Router# show vines interface

VINES address is 3000902D:0001
  Next client will be 3000902D:8001
  Addresses are displayed in hexadecimal format.
  Slowest update interval is 90 seconds
  Roll Call timer queue:
    Neighbor Router3-Et2-0000.0c01.24ea in 180 seconds
  Sequence: 01029DD7, Packet ID: 00000003
  Reassembly timer queue:  (empty)
  Retry timer queue:  (empty)
   Participating in vines time of day synchronization
Hssi0 is down, line protocol is down
  VINES protocol processing disabled
Fddi0 is up, line protocol is up
  VINES broadcast encapsulation is ARPA
  Interface metric is 0008 [0 5000] (0.1000 seconds)
  Split horizon is enabled
  ARP processing is dynamic, state is learning (for another 18 seconds)
  Special serverless net processing enabled
  Outgoing access list is not set
  Fast switching is enabled
  Routing updates every 90 seconds. Next in 50 seconds.
  Next synchronization update in 11:58:17.
  Nodes present: 0 5.5x servers, 0 5.5x routers, 0 5.5x clients
                 0 4.11 servers, 0 4.11 routers, 0 4.11 clients
  Neighbors:  none.
```

Table 16-4 describes the fields that may be shown in the display.

Table 16-4 Show VINES Interface Field Descriptions

Field	Description
VINES address	Address of the router.
Next client will be	Address the router will assign to the next client that requests an address. This line is interesting only if the router has been configured via the **vines arp-enable** command to respond to address assignment requests.
Addresses	Indicates whether addresses will be displayed as decimal or hexadecimal numbers.
Slowest update interval	Indicates the longest time interval (in seconds) between routing updates on any of the router's interfaces.
Roll Call timer Neighbor	Displays a list of all neighbor paths for which an RTP request will be sent on a regular basis, and the interval until that timer expires.
Sequence	Current SRTP sequence number for this router.
Packet ID	Identifier number that will be used on the last SRTP update message sent by this router
Reassembly timer	Displays a list of all neighbor paths for which an SRTP update is currently being reassembled, and the interval until that timer expires.
Retry timer	Displays a list of all neighbor paths for which an SRTP request is currently being retried, and the interval until that timer expires.
Participating in vines time of day synchronization	Indicates whether the router is participating in VINES time-of-day synchronization. This is controlled by the **vines time participate** global configuration command.
Hssi0/Ethernet 0/Ethernet 1 is up/down	Type and number of interface, and whether it is currently active and inserted into network (up) or inactive and not inserted (down).
Line protocol is	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful). This field can report the values "up," "down," and "administratively down."
VINES protocol processing disabled	Indicates that VINES processing is not enabled on the interface (that is, you have not issued a **vines metric** command on the interface).
VINES broadcast encapsulation	Type of encapsulation used for VINES broadcast packets, as defined with the **vines encapsulation** command. This field can report the values "arpa," "vines-tr," and "snap."
Interface metric	Metric that has been configured for the interface with the **vines metric** command. The metric is shown in internal form, configuration form, and in seconds.
Split horizon	Indicates whether split horizon has been enabled or disabled (via the **vines split-horizon** command).
ARP processing	Indicates whether this interface will process ARP packets, as specified by the **vines arp-enable** command.
Special serverless net processing	Indicates whether this interface is defined via the **vines serverless** command as being connected to a serverless network.
Outgoing access list	Indicates whether an access list is set.
Fast switching	Indicates whether fast switching has been enabled via the **vines route-cache** command). The value reported in this field can be "enabled," "disabled," or "not supported."

Field	Description
Routing updates every Next in	Frequency of routing updates, in seconds. This also indicates when the next routing update will be transmitted on the interface. You set the update interval with the **vines update interval** command.
Routing updates	Indicates whether routing updates contain all entries in the routing table or just changes to the table since the last update was sent. You set the method used with the **vines update deltas** command.
Next synchronization	Indicates when the next SRTP synchronization update will be sent.
Nodes Present	Indicates the number and type of all VINES-speaking devices present on the given physical network segment.
Neighbors 0 Router2	List of all VINES neighbor on that interface and what version of the RTP protocol they are running. 0 means RTP, and 1 means SRTP.

Related Commands

vines arp-enable
vines encapsulation
vines metric
vines route-cache

show vines ipc

To display information about any currently active IPC connections, use the **show vines ipc** EXEC command.

show vines ipc

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

Information about the IPC protocol formats, data sequences, and state machines can be found in Banyan documentation.

Sample Display

The following is sample output from the **show vines ipc** command:

```
Router# show vines ipc

Vines IPC Status:

Next Port: 513
Next Connection: 3
Next check in: 27 sec

Connection 2, state: connected
  Local address: Router1, id 0002, last port: 0200
  Remote address: Router2, id 0002, last port: 0001
  Last send seq: 0005, Last rcvd seq: 0005
  Next send ack: 0005, Last sent ack: 0005
  Server metric 4, last hop 0, bias 0, total 800 (ms)
  Send ACK in 0 ms, Retransmit in 0 ms
  Idle check in 0 sec
  Retransmit queue contains 0 packets
  No packet in reassembly
```

Table 16-5 describes the fields shown in the display.

Table 16-5 Show VINES IPC Field Descriptions

Field	Description
Next Port:	IPC port number that the router will use when a new, unique IPC port number is needed.
Next Connection:	IPC connection number that the router will use when a new, unique IPC connection number is needed.
Next check in:	When the router will make the next pass of the IPC connection table to examine each of the connection-specific timers.

Field	Description
Connection 2, state:	State of a particular connection. Possible states are connecting, connected, idle, and dead.
Local address:	VINES IP address of the local side of the connection.
last port:	Last port number used on this particular connection by the local host.
Remote address:	VINES IP address of the remote side of the connection.
last port:	Last port number used on this particular connection by the remote host.
Last send seq:	Last sequence number sent on this particular connection used by the local host.
Last rcvd seq:	Last sequence number received on this particular connection used by the local host.
Next send ack:	Next acknowledgment number that will be sent on this particular connection by the local host.
Last sent ack:	Last acknowledgment number that has been sent on this particular connection by the local host.
Server metric	Metric value from this host to the remote host's server or router.
last hop	Metric value from the remote host's server or router to the remote host itself. If the remote host is a server or router, this value should be zero.
bias	Bias added to the metric to account for variance in the round-trip delay of a message going to the remote host.
total	Total metric value used to reach the remote host. It is the sum of the three previous numbers.
Send ACK	Time, in seconds, until the next acknowledgment message is sent by the local host.
Retransmit	Time, in seconds, until a message is retransmitted by the local host.
Idle check in	Time, in seconds, until this connection will be checked to see if it has been idle for 30 seconds.
Retransmit queue contains ... packets	Number of messages that have been sent but not acknowledged.
No packet in reassembly	Number of packets that have been received and are being reassembled into a larger message.

show vines neighbor

To display the entries in the VINES neighbor table, use the **show vines neighbor** EXEC command.

show vines neighbor [*address* | **interface** *type number* | **server** *number*]

Syntax Description

address	(Optional) Displays the entry for the specified neighbor.
interface *type number*	(Optional) Displays all neighbor paths in the neighbor table that use the specified interface.
server *number*	(Optional) Displays all entries in the neighbor table that have the specified network number.

Command Mode

EXEC

Usage Guidelines

If no keywords or arguments are specified, all entries in the neighbor table are displayed.

Sample Displays

The following is sample output from the **show vines neighbor** command. This sample shows all entries in the VINES neighbor table.

```
Router# show vines neighbor

6 neighbors, 7 paths, version 14, next update 34 seconds

Address        Hardware Address   Type  Int    Flag Age  Metric  Uses

Router1        -                  HDLC  Se0    R0*  n/a   0230      7
Router2        -                  -     -      C1   -       -       -
Router3        0000.0c01.24ea     ARPA  Et2    R0*  42    0020      9
Router4        -                  PPP   Se1    R1   n/a   0230      0
 Router4       0000.0c01.0506     ARPA  Et0    R1.  n/a   0020      0
 Router4       0000.0c01.9ac9     VINES To0    R1*  n/a   0020      0
```

The following is sample output from the **show vines neighbor** command for a specific server. This sample shows all entries in the VINES neighbor table for router3.

```
Router# show vines neighbor router3

3 neighbors, 4 paths, version 7, next update 24 seconds

Address          Hardware Address     Type  Int       Flag Age  Metric    Uses

Router3          0000.0c01.24ea       ARPA  Et2       R0*   42   0020        9

  RTP Counters:

    Interface Ethernet2, address Router3-Et2-0000.0c01.24ea
      Timers:
        Roll Call:  00:03:00
       Received counters:
```

```
Requests:   00000000
Responses:  00000000
Updates:    00000000
Redirects:  00000000
Unknown:    00000000
```

Table 16-6 describes the fields shown in the display.

Table 16-6 Show VINES Neighbor Field Descriptions

Field	Description
neighbors	Number of neighbors in the neighbor table.
paths	Number of paths to the neighbor.
version	Version number of the VINES neighbor table. The number is incremented each time a route or path is added to or deleted from this table.
next update	Time, in seconds, until the next routing update is sent.
Address	Address of the neighbor station. The neighbor's name is displayed if you have issued a **vines host** command.
Hardware Address	MAC address of the router interface through which the VINES neighbor in this entry can be reached.
Type	Type of MAC-level encapsulation used to communicate with this neighbor.
Int	Type and number of interface through which the VINES neighbor can be reached

Field	Description
Flag	This field is a three-column field. The first column indicates how the path was learned. It can be one of the following values: • C—Connected (that is, this is the entry for this router). • D—Learned via an RTP redirect message. • P—Placeholder. This neighbor is currently used as the next hop for a static route. • R—Learned via an RTP update message. • S—Static path entry (entered with the **vines neighbor** command). The second column indicates what version of the RTP protocol this neighbor is running. It can be one of the following values: • 0—Version 0 of the RTP protocol. This is the version used by VINES servers prior to VINES version 5.50. • 1—Version 1 of the RTP protocol, commonly called SRTP. This is the version used by VINES servers in VINES version 5.50 and later. The third column indicates how this path will be used. It can be one of the following values: • *—An asterisk means that this is the next path that will be used next when forwarding a frame to that neighbor. • .—A dot means that this is the alternate path that will be used in round-robin fashion. • Blank—No value means this is backup path that will not be used. In the sample output, there are two paths to Router4 with the same metric. These two paths will be used in a round-robin fashion, and the Token Ring path will be the next one of the two used. There is a third path to Router4 via the serial line, but this will not be used unless both of the other paths are lost.
Age	Age of this VINES neighbor table entry, in seconds. This entry will show an age of "n/a" for RTP Version 0 neighbors on WAN interfaces, when the interface has been configured for delta-only updates. In all other cases, this entry will contain a number.
Metric	Distance to this neighbor. This normally is the same as the interface metric, but may be different because of network topology or router configuration.
Uses	For all entries except placeholders, indicates the number of times that path was used to forward a packet. For placeholder entries, indicates the number of static routes that use the neighbor as the first hop.
RTP Counter:	This section shows counters that are specific to a neighbor port that is running the RTP protocol only. If the neighbor has multiple interfaces, then multiple sections will show up in this part of the display.
Interface ...	Identifies the network interface and full identifier for a neighbor port.
Timers: Roll Call	Identifies whether or not the roll call timer is active for this neighbor, and if so, when it will expire.
Received Counters	Indicates the number and type of RTP packets received from this neighbor port.
SRTP Counter:	This section shows counters that are specific to a neighbor port that is running the SRTP protocol. If the neighbor has multiple interfaces, then multiple sections will show up in this part of the display.

Field	Description
Interface	Identifies the network interface and full identifier for a neighbor port.
Timers: Reassembly	Identifies whether or not the reassembly timer is active for this neighbor, and if so, when it will expire.
Timers: Retry	Identifies whether or not the retry timer is active for this neighbor, and if so, when it will expire.
Received Counters	Indicates the number, type, and sequence number of matching SRTP packets received from this neighbor port.
Transmitted Counters	Indicated the number and type of SRTP packets transmitted explicitly to this neighbor port.

Related Commands

clear vines neighbor
clear vines route
show vines cache
vines host
vines neighbor
vines update deltas
vines update interval

show vines route

To display the contents of the VINES routing table, use the **show vines route** EXEC command.

show vines route [*number* | **neighbor** *address*]

Syntax Description

number	(Optional) Displays the routing table entry for the specified network.
neighbor *address*	(Optional) Displays all routes in the VINES routing table that have the specified neighbor as their first hop.

Command Mode

EXEC

Usage Guidelines

If no keywords or arguments are specified, all entries in the routing table are displayed.

Sample Display

The following is sample output from the **show vines route** command. This sample shows all entries in the VINES routing table.

```
Router# show vines route

Worf              Worf              R0*       2       2       0
Succubus          Succubus          R1*       2       2       0
Aloe              -                 C1        -       -       -
Vera              Vera              R0*       2       2       0
Falcon            Falcon            R0*       2       2       0
Zangbutt          Worf              R0*       2       4       0
Zangbutt          Vera              R0        2       4       0
```

The following is sample output from the **show vines route** command for a specific neighbor. This sample shows all entries in the VINES routing table for router1.

```
Router# show vines route router1

8 servers, 10 routes, version 58, next update 32 seconds

Network       Neighbor     Flags     Age      Metric   Uses    Origin    Local    Flags

Router1       Router2      R0*       n/a       0250      0     001AFE7B  00010FCA  0009
```

Table 16-7 describes the fields shown in the display.

Table 16-7 Show VINES Route Field Descriptions

Field	Description
servers	Number of servers in the routing table.
routes	Number of routes in the routing table.

Field	Description
version	Version number of the VINES routing table. This number is incremented each time a server or route is added to or deleted from this table.
next update	Time, in seconds, until the next routing update is sent.
Hash	Position of this entry in the routing table.
Network	Name or number of the remote network. Networks take the name of the server that defines the network.
Neighbor	Next hop to the destination network.
Flags	This field is a series of single-column fields. The first column indicates how the route was learned. It can be one of the following values: • C—Connected (that is, this is the entry for this router). • D—Learned via an RTP redirect message. • R—Learned via an RTP update message. • S—Static entry (entered with the **vines route** command). The second column indicates what version of the RTP protocol this router is running. It can be one of the following values: • 0—Version 0 of the RTP protocol. This is the version used by VINES servers prior to VINES version 5.50. This version number will also be shown if the route was learned via a pre-5.50 server, and thus the version information was lost. • 1—Version 1 of the RTP protocol, commonly called SRTP. This is the version used by VINES servers in VINES version 5.50 and later. An asterisk in the third column indicates that this route will be used next when forwarding a frame to that server. The fourth column indicates whether that route will be used to forward a broadcast from a serverless network. It can be one of the following values: • N—This server is considered to be the nearest server and is on a directly connected network. • n—This server is considered to be the nearest server but is not on a directly connected network. The fifth column contains the letter “S” if the route is in a suppression state. The sixth column contains the letter “h” if this path has a metric that is higher than the best metric for this neighbor. This indicates that the path is not eligible for use in load sharing.
Age	Age of this VINES routing table entry, in seconds. An age of n/a indicates the destination is accessible via a neighbor that is sending delta-only updates. Note that even though the neighbor entry for Pica has an age, there is no age available for its routing table entry or other routing entries reachable via Pica. This is because the periodic hello messages from Pica contain no routing information, only neighbor reachability information.
Metric	Distance to this server. This normally is the distance to the neighbor router plus the distance advertised by that neighbor. This does not hold for static routes.
Uses	Number of times this route has been used to forward a packet.

Field	Description
Origin	Last known timestamp that originated from this server. If this field is not valid, as indicated by the following set of flags, then it will be zero.
Local	Local timestamp then this route entry was learned or last changed.
Flags	This field is a series of bit flags presented as a hexadecimal number. The following are the defined values: • 0001—The neighbor of this server reaches it through a LAN interface. • 0002—The neighbor of this server reaches it through a WAN interface. • 0004—The neighbor of this server reaches it through a non-VINES interface. • 0008—The origin timestamp for this entry is not valid. The entry is either for a pre-5.50 server, or the entry was learned via a pre-5.50 server.

Related Commands

clear vines neighbor
clear vines route
show vines cache
vines route
vines neighbor
vines update deltas
vines update interval

show vines service

To display information about the router's application layer support, use the **show vines service** EXEC command.

show vines service [**fs** | **nsm** | **ss** | **vs**]

Syntax Description

fs	(Optional) Displays file service information.
nsm	(Optional) Displays network and system management service information.
ss	(Optional) Displays server service information.
vs	(Optional) Displays security service information.

Command Mode

EXEC

Sample Display

The following is sample output from the **show vines service** command:

```
Router# show vines service

Vines Files Service:
    Name:     FS@Doc-ags+1@Servers  (FS)
    Ports:    Well Known 6, Transient 0
    Timer:    not running

Network & System Management Service:
    Name:     NSM@Doc-ags+1@Servers  (NSM)
    Ports:    Well Known 25, Transient 0
    Timer:    not running

Server Service:
    Name:     SS@Doc-ags+1@Servers  (SS)
    Ports:    Well Known 7, Transient 0
    Emulates: 5.50(0),  Supports: 3.22(49) - 6.99(49)
    Timer:    not running

VINES Security Service:
    Name:     VS@Doc-ags+1@Servers  (VS)
    Ports:    Well Known 19, Transient 0
    Timer:    not running
```

Table 16-8 describes the fields shown in the display.

Table 16-8 Show VINES Service Field Descriptions

Field	Description
Name:	Name of the service.
Ports:	Ports on which the service is running.
Timer:	Time at which this service will wake up and perform some periodic functions.

The following is sample output from the **show vines service** command using the **fs**, **nsm**, **ss**, and **vs** keywords:

```
Router# show vines service fs

Vines Files Service:
  Periodic timer not running.

Router# show vines service nsm

Network & System Management Service:
  Next wakeup in 00:00:29.

Router# show vines service ss

Server Service:
  Next wakeup in 00:01:51.
  Time is 17:12:55 PDT Jun 23 1994
  Time last set by Doc-ags, 0:28:09 ago.
  Time epoch is SS@Doc-ags@Servers-9, started 00:28:09 ago.
    Participating in vines time of day synchronization.
    Sending time messages to the broadcast address.
    Synchronizing vines time with system time.

Router# show vines service vs

VINES Security Service:
  Periodic timer not running.
```

Table 16-9 describes the fields shown in the displays.

Table 16-9 Show VINES Service Field Descriptions

Field	Description
Periodic timer not running	Indicates that this service has no periodic functions to perform.
Next wakeup in ...	Time, in seconds, until the service performs its periodic actions. For the Server service, this is to send a time synchronization message. For the NSM service, this is to send any requested trace packets. The periodic interval for the NSM service is 30 seconds when no trace messages are pending.
Time is ...	Current time (in the format *hours*:*minutes*:*seconds*) and date.

Field	Description
Time last set ...	Server that last adjusted the time, how much it adjusted the time, and how long ago it was adjusted. For times within the last 24 hours, the time format is *hours*:*minutes*:*seconds*. For times longer ago than 24 hours, the time format is *weeks*w*days*d.
Time epoch is ...	Name of the current time epoch (in the format *name-number*), and when it was established.

Related Commands

vines time access-group
vines time participate
vines time set-system
vines time use-system

show vines traffic

To display the statistics maintained about VINES protocol traffic, use the **show vines traffic** EXEC command.

show vines traffic [*type number*]

Syntax Description

type	(Optional) Interface type.
number	(Optional) Interface number.

Command Mode

EXEC

Usage Guidelines

If no interface is specified, values for all interfaces are displayed.

Sample Display

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

```
Router# show vines traffic

SYSTEM TRAFFIC:
  Rcvd: 204 total, 12708 bytes, 0 format errors, 0 not enabled,
        15 local dst, 189 bcast, 0 forwarded
        0 no route, 0 zero hops
        0 checksum errors, 3 IP unknown, 0 IPC unknown
        3 bcast forwarded, 1 bcast helpered, 0 dup bcast
  Sent: 21 packets, 1278 bytes
        0 unicast, 21 bcast, 0 forwarded
        0 encap failed, 0 access failed, 0 down
        0 bcast fwd, 3 not fwd (toward source)
        0 notlan, 0 not gt4800, 0 no pp charge
   ARPv0: Rcvd 0/0/0/0/0, Sent 0/0/0/0
   ARPv1: Rcvd 0/0/0/0/0, Sent 0/0/0/0
     ICP: Rcvd 0/0/0, Send 0/0
     IPC: Rcvd 17, Sent 8
   RTPv0: Rcvd 2/10/0/0/170/0/0/5, Sent 0/6/00/0/91/10/0
   RTPv1: Rcvd 0/0/0/0/0/0, Sent 0/3/60/0
     SPP: Rcvd 0, Sent 0
    Echo: Rcvd 5, Sent 5
   Proxy: Rcvd 0, Sent 0
IPC TRAFFIC BY PORT NUMBER:
Broadcast: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000
 Helpered: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
```

```
              19:00000000
  Unicast: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000
 Proxied: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000
P_Replies: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000

Interface Hssi0:
  Rcvd: 0 packets, 0 bytes, 0 format errors, 0 not enabled,
        0 local dst, 0 bcast, 0 forwarded,
        0 no route, 0 zero hops
        0 checksum errors, 0 IP unknown, 0 IPX unknown
        0 bcast forwarded, 0 bcast helpered, 0 dup bcast
  Sent: 0 packets, 0 bytes
        0 unicast, 0 bcast, 0 forwarded
        0 encap failed, 0 access failed, 0 down
        0 bcast fwd, 0 not fwd (toward source)
        0 notlan, 0 not gt4800, 0 no pp charge
   ARPv0: Rcvd 0/0/0/0/0, Sent 0/0/0/0
   ARPv1: Rcvd 0/0/0/0/0, Sent 0/0/0/0
     ICP: Rcvd 0/0/0, Send 0/0
     IPC: Rcvd 0, Sent 8
  RTPv0: Rcvd 0/10/0/0/0/0/0/0, Sent 0/0/00/0/0/0/0
  RTPv1: Rcvd 0/0/0/0/0/0, Sent 0/3/60/0
    SPP: Rcvd 0, Sent 0
   Echo: Rcvd 0, Sent 0
  Proxy: Rcvd 0, Sent 0
```

Table 16-10 describes the fields shown in the display.

Table 16-10 Show VINES Traffic Field Descriptions

Field	Description
SYSTEM TRAFFIC:	This section displays statistics about all VINES packets handled by the router.
Rcvd:	This section displays statistics about VINES packets received by the router.
total packets	Total number of VINES packets received.
bytes	Total bytes in all the VINES packets received.
format errors	Number of VINES packets that had errors in the format of the VINES IP header. Currently, the only thing checked is the length field in the header. The number of packets with format errors is included in the count of total packets received (in the Rcvd: field).

Field	Description
not enabled	Number of VINES packets received on an interface on which VINES was not enabled. These packets are not included when counting the total packets received (in the Rcvd: field).
local dst	Number of packets accepted for further processing because they were addressed to the router's unicast address.
bcast	Number of packets accepted for further processing because they were addressed to the router's broadcast address.
forwarded	Number of packets not accepted for further processing but that were simply forwarded out another interface.
no route	Number of packets discarded because the router did not know how to reach the destination.
zero hops	Number of packets discarded because the hop count field in the VINES IP header was zero.
checksum errors	Number of packets accepted by the router for further processing (the sum of the "local dest" and "bcast" fields) that were discarded because the checksum was bad.
IP unknown	Number of packets accepted by the router (the sum of the "local dest" and "bcast" fields) that were discarded because the IP protocol type was unknown.
IPC unknown	Number of packets accepted by the router for further processing (the sum of the "local dest" and "bcast" fields) that were discarded because the IPC port number was unknown.
bcast forwarded	Number of broadcast packets accepted by the router for further processing (as shown in the "bcast" field) that were forwarded because they had a nonzero hop count. (Note that the sum of the "bcast forwarded," "bcast helpered," and "dup bcast" fields will not equal the total number of broadcast packets received.)
bcast helpered	Number of broadcast packets accepted by the router (as shown in the "bcast" field) that were "helpered" to a Banyan server. (Note that the sum of the "bcast forwarded," "bcast helpered," and "dup bcast" fields will not equal the total number of broadcast packets received.)
dup bcast	Number of broadcast packets accepted by the router (as shown in the "bcast" field) that were classified as duplicates and discarded. (Note that the sum of the "bcast forwarded," "bcast helpered," and "dup bcast" fields will not equal the total number of broadcast packets received.)
Sent:	This section displays statistics about VINES packets sent by the router.
packets	Total number of VINES packets sent.
bytes	Total bytes in all the VINES packets sent.
unicast	Number of unicast packets originating at the router.
bcast	Number of broadcast packets originating at the router.
forwarded	Number of unicast packets that were forwarded from another interface.
encap failed	Number of packets not sent because of an encapsulation failure. This usually happens when entries in a map for a public data network, such as X.25 or Frame Relay, are missing.

Field	Description
access failed	Number of packets not sent because the destination was denied by an access list.
down	Number of packets not sent because the interface was down.
bcast fwd	Number of broadcast packets that were forwarded from another interface.
not fwd (toward source)	Number of broadcast packets that were not forwarded because this interface is the interface on which the broadcast was received.
not lan	Number of broadcast packets that were not forwarded because they were marked for LANs only and this interface is not a LAN (for example, it might be a serial interface.)
not gt	Number of broadcast packets that were not forwarded because they were marked for high-speed interfaces only and this interface is a low-speed interface (line speed of 4800 baud or less).
no pp charge	Number of broadcast packets that were not forwarded because they were marked to send only to networks that do not have per-packet charging and this interface is to a network that has per-packet charging.
ARPv0:	This section displays statistics about VINES ARP packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*	Number of ARP packets received of type 0, 1, 2, 3, and other.
Sent *x*/*x*/*x*/*x*	Number of ARP packets sent of type 0, 1, 2, and 3.
ARPv1:	This section displays statistics about VINES SARP packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*	Number of SARP packets received of type 0, 1, 2, 3, and other.
Sent *x*/*x*/*x*/*x*	Number of SARP packets sent of type 0, 1, 2, and 3.
ICP:	This section displays statistics about VINES ICP packets sent and received by the router.
Rcvd *x*/*x*/*x*	Number of ICP packets received of type 0, 1, and other.
Sent *x*/*x*	Number of ICP packets sent of type 0 and 1.
IPC:	This section displays statistics about VINES IPC packets sent and received by the router.
Rcvd	Number of IPC packets received.
Sent	Number of IPC packets sent.
RTPv0:	This section displays statistics about VINES routing protocol (RTP) packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*/*x*/*x*	Number of RTP packets received of type 0, 1, 2, 3, 4, 5, 6, and other. The counts of type 0, type 2, type 3, and other RTP packets should always be zero.
Sent *x*/*x*/*x*/*x*/*x*/*x*/*x*	Number of RTP packets sent of type 0, 1, 2, 3, 4, 5, and 6.
RTPv1:	This section displays statistics about VINES routing protocol (SRTP) packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*	Number of SRTP packets received of type 0, 1, 2, 3, and other. The count of other SRTP packets should always be zero.
Sent *x*/*x*/*x*/*x*/*x*	Number of SRTP packets sent of type 0, 1, 2, 3.

Field	Description
SPP:	This section displays statistics about VINES SPP packets sent and received by the router.
Rcvd	Number of SPP packets received.
Sent	Number of SPP packets sent.
Echo:	This section displays statistics about VINES echo packets sent and received by the router.
Rcvd	Number of MAC-level echo packets received.
Sent	Number of MAC-level echo packets sent.
Proxy:	This section displays statistics about VINES proxies sent and received by the router. A proxy is when a client sends a query directly to the router for which the router does not have the intelligence to respond. The router then sends these queries to a Banyan server, and when it receives the response from the server, the router relays it back to the client.
Rcvd	Number of proxy queries received by the router.
Sent	Number of proxy queries sent by the router.
IPC TRAFFIC BY PORT NUMBER:	This section displays statistics about VINES Interprocess Communications Protocol (IPC) packets. The information displayed in this section is particularly useful when a serverless network is connected to the router.
Broadcast:	Number of VINES IPC messages, by destination port number, received by the router because they were addressed to the VINES IP broadcast address.
Helpered:	Number of broadcast messages that were sent toward a Banyan server because they were received on an interface for a serverless network.
Unicast:	Number of VINES IPC messages, by destination port number, received by the router because they were specifically addressed to the VINES IP address of the router.
Proxied:	Number of unicast messages received that were sent to a Banyan server because they were received on a serverless interface and because the router did not know how to respond to the message.
P_Replies:	Number of responses to a proxy query that were received from a Banyan server.
Interface	This section displays statistics about the individual interfaces in the router. The fields in this section have the same meanings as the fields of the same name in the "SYSTEM TRAFFIC" section, except that the statistics are for the particular interface, not for the entire router.

Related Commands

clear vines traffic
vines serverless

trace

To determine the path that a packet takes when traversing a VINES network, use the **trace** EXEC command.

trace [**vines** | **oldvines**] [*address*]

Syntax Description

vines	(Optional) Specifies the VINES protocol. This trace is compatible with the Banyan VINES traceroute function.
oldvines	(Optional) Specifies the VINES protocol. This trace is compatible with our **trace** function prior to Cisco IOS Release 10.2.
address	(Optional) Address of a node. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.

Command Mode

EXEC

Usage Guidelines

The **trace** EXEC command supports the Banyan traceroute function. This enables trace requests on a VINES network to reach all servers on the network.

This command does not produce the names of any VINES servers that are traversed.

Sample Displays

The following is sample output from the VINES **trace** command when you specify the **vines** keyword:

```
Router# trace vines

Target Vines address: wayfinder
Source Vines address: coinspinner
From: 0002801578 Coinspinner        To: 0002609380 Wayfinder

Server                    Gate                        metric media address
0002801578 Coinspinner    0805371606 Router               4     40 000030C0FEB6
0805371606 Router         0002609380 Wayfinder            2   2560 10005A746A36
```

The following is sample output from the VINES **trace** command when you specify the **oldvines** keyword:

```
Router# trace oldvines

Target vines address: 27AF92:1
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [0]:
Maximum Time to Live [15]:
Type escape sequence to abort.
```

```
Tracing the route to COINSPINNER (27AF92:1)
 0 Farslayer (30002A2D:1) 0 msec 4 msec 4 msec
 1 Coinspinner (27AF92:1) 4 msec 4 msec 8 msec
```

The value *nn* `msec` indicates the round-trip time for each probe in milliseconds, for each node.

vines access-group

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

vines access-group *access-list-number*
no vines access-group *access-list-number*

Syntax Description

access-list-number	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, *access-list-number* is a decimal number from 1 to 100. For extended access lists, *access-list-number* is a decimal number from 101 to 200.

Default

No access list is applied.

Command Mode

Interface configuration

Usage Guidelines

The **vines access-group** command applies an access list created with the **vines access-list** command to an interface.

You can apply only one access list to an interface.

Example

In the following example, access list 1 is applied to Ethernet interface 0:

```
interface ethernet 0
vines access-group 1
```

Related Command

vines access-list (extended)

vines access-list (extended)

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

vines access-list *access-list-number* {**deny** | **permit**} *protocol source-address source-mask* [*source-port source-port-mask*] *destination-address destination-mask* [*destination-port destination-port-mask*]
no vines access-list *access-list-number*

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 101 to 200.
deny	Denies access if the conditions are matched.
permit	Allows access if the conditions are matched.
protocol	VINES protocol ID number or name. The number can be a value from 1 to 255 or one of the following protocol keywords: • **arp**—Address Resolution Protocol • **icp**—Internet Control Protocol • **ip**—VINES Internet Protocol • **ipc**—Interprocess Communications • **rtp**—Routing Update Protocol • **spp**—Sequence Packets Protocol **arp**—Address Resolution Protocol • **icp**—Internet Control Protocol • **ip**—VINES Internet Protocol • **ipc**—Interprocess Communications • **rtp**—Routing Update Protocol • **spp**—Sequence Packets Protocol
source-address	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
source-mask	Mask to be applied to *source-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.

source-port	Number of the local port from which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 16-11 in the "Usage Guidelines" section lists some IPC port numbers.
source-port-mask	(Optional) Mask to be applied to *source-port*. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. These bits correspond to the bits in the port that should be ignored.
destination-address	VINES address of the network to which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
destination-mask	Mask to be applied to *destination-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.
destination-port	Number of the local port to which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 16-11 lists some IPC port numbers.
destination-port-mask	(Optional) Mask to be applied to *destination-port*. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. These bits correspond to the bits in the port that should be ignored.

Default

No extended VINES access list is specified.

Command Mode

Global configuration

Usage Guidelines

An extended VINES access list filters packets based on their protocol, source and destination addresses, and source and destination address masks, and optionally on their source and destination ports, and source and destination port masks. This differs from the standard access list filters in that you can specify port masks.

Use the **vines access-group** command to assign an access list to an interface.

Keep the following in mind when configuring VINES network access control:

- You can apply only one access list to an interface.
- The conditions in the access list are applied to all outgoing packets that are forwarded by the router. Packets generated by the router are not subject to the access list.
- Access list entries are scanned in the order you enter them. The first matching entry is used.
- An implicit *deny everything* entry is defined at the end of an access list unless you include an explicit *permit everything* entry at the end of the list.
- All new entries to an existing list are placed at the end of the list. You cannot add an entry to the middle of a list. This means that if you have previously included an explicit *permit everything* entry, new entries will never be scanned. The solution is to delete the access list and retype it with the new entries.

If you specify a protocol type of IPC, the port (either *source-port* or *destination-port*) can be one of the values shown in Table 16-11.

Table 16-11 Some VINES IPC Port Numbers

IPC Port Number (Hexadecimal)	Service
0x0003	Back End (only on PCs; it is the 25th line notification)
0x0004	Mail Service
0x0006	“VINES Files” File Service
0x0007	Server Service
0x000F	StreetTalk Service
0x0012	Network Management
0x0013	VINES Security
0x0016	StreetTalk Directory Assistance
0x0017	StreetTalk Directory Assistance Service Listening Port
0x0019	Systems and Network Management

Example

In the following example, the first line prohibits communication from any client process to the service on IPC port 0x14; the second line permits all other communication:

```
vines access-list 101 deny   IPC 0:0 ffffffff:ffff 0x14 0 0:0 ffffffff:ffff 0 0xFFFF
vines access-list 101 permit IP 0:0 ffffffff:ffff     0:0 ffffffff:ffff
```

Related Commands

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

priority-list protocol †
show vines access
vines access-group
vines access-list (simple)

vines access-list (simple)

To create a simple VINES access list, use this version of the **vines access-list** global configuration command. To remove a simple access list, use the **no** form of this command.

vines access-list *access-list-number* {**deny** | **permit**} *source-address source-mask*
no vines access-list *access-list-number*

Syntax Description

access-list-number	Access list number. It is a number from 201 to 300.
deny	Denies access if the conditions are matched.
permit	Allows access if the conditions are matched.
source-address	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
source-mask	Mask to be applied to *source-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.

Default

No simple VINES access list is specified.

Command Mode

Global configuration

Usage Guidelines

A simple VINES access list filters packets based on their source address and source address mask. These access lists are used to decide which stations to accept time updates from.

Use the **vines access-group** command to assign an access list to an interface.

Keep the following in mind when configuring VINES network access control:

- You can assign only one access list to an interface.
- The conditions in the access list are applied to all outgoing packets that are forwarded by the router. Packets generated by the router are not subject to the access list.
- Access list entries are scanned in the order you enter them. The first matching entry is used.
- An implicit *deny everything* entry is defined at the end of an access list unless you include an explicit *permit everything* entry at the end of the list.
- All new entries to an existing list are placed at the end of the list. You cannot add an entry to the middle of a list. This means that if you have previously included an explicit *permit everything* entry, new entries will never be scanned. The solution is to delete the access list and retype it with the new entries.

Example

The following example defines an access list that accepts time updates only from the servers on networks 30015800 and 30004355; it denies time updates from all other sources:

```
vines access-list 201 permit 30015800:0001 00000000:0000
vines access-list 201 permit 30004355:0001 00000000:0000
vines access-list 201 deny 00000000:0000 FFFFFFFF:FFFF
interface ethernet 0
vines time access-group 201
```

Related Commands

show vines access
vines access-group
vines access-list (extended)
vines time access-group
vines time participate
vines time set-system
vines time use-system

vines access-list (standard)

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

vines access-list *access-list-number* {**deny | permit**} *protocol source-address source-mask* [*source-port*] *destination-address destination-mask* [*destination-port*]
no vines access-list *access-list-number*

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 1 to 100.
deny	Denies access if the conditions are matched.
permit	Allows access if the conditions are matched.
protocol	VINES protocol ID number or name. It can be a value from 1 to 255 or one of the following protocol keywords: • **arp**—Address Resolution Protocol • **icp**—Internet Control Protocol • **ip**—VINES Internet Protocol • **ipc**—Interprocess Communications • **rtp**—Routing Update Protocol • **spp**—Sequence Packets Protocol
source-address	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
source-mask	Mask to be applied to *source-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bit in the address that should be ignored.
source-port	(Optional) Number of the local port from which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 16-12 lists some IPC port numbers.
destination-address	Address of the network to which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.

destination-mask	Mask to be applied to *destination-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.
destination-port	(Optional) Number of the local port to which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 16-12 in the "Usage Guidelines" section following lists some IPC port numbers.

Default

No standard VINES access list is specified.

Command Mode

Global configuration

Usage Guidelines

A standard VINES access list filters packets based on their protocol, source and destination addresses, and source and destination address masks, and optionally on their source and destination ports.

Use the **vines access-group** command to apply an access list to an interface.

Keep the following in mind when configuring VINES network access control:

- You can apply only one access list to an interface.
- The conditions in the access list are applied to all outgoing packets that are forwarded by the router. Packets generated by the router are not subject to the access list.
- Access list entries are scanned in the order you enter them. The first matching entry is used.
- An implicit *deny everything* entry is defined at the end of an access list unless you include an explicit *permit everything* entry at the end of the list.
- All new entries to an existing list are placed at the end of the list. You cannot add an entry to the middle of a list. This means that if you have previously included an explicit *permit everything* entry, new entries will never be scanned. The solution is to delete the access list and retype it with the new entries.

If you specify a protocol type of IPC, the port (either *source-port* or *destination-port*) can be one of the values shown in Table 16-12.

Table 16-12 Some VINES IPC Port Numbers

IPC Port Number (Hexadecimal)	Service
0x0003	Back End (only on PCs; it is the 25th line notification)
0x0004	Mail Service
0x0006	"VINES Files" File Service
0x0007	Server Service
0x000F	StreetTalk Service
0x0012	Network Management
0x0013	VINES Security
0x0016	StreetTalk Directory Assistance
0x0017	StreetTalk Directory Assistance Service Listening Port
0x0019	Systems and Network Management

Examples

In the following example, the first line prohibits any communication on StreetTalk port (port number 0xF); the second line permits all other communication:

```
vines access-list 1 deny   IPC 0:0 ffffffff:ffff 0xf 0:0 ffffffff:ffff 0xf
vines access-list 1 permit IP 0:0 ffffffff:ffff     0:0 ffffffff:ffff
```

The following example filters all mail service on Ethernet interface 0 and permits all other traffic:

```
interface Ethernet 0
vines access-group 101
!
vines access-list 101 deny ipc 0:0 FFFFFFFF:FFFF 4 0 0:0 FFFFFFFF:FFFF 0 0xF FFF
```

Related Commands

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

priority-list protocol †
show vines access
vines access-group
vines access-list (extended)
vines access-list (simple)

vines arp-enable

To enable the processing of ARP packets, use the **vines arp-enable** interface configuration command. To disable the processing of ARP packets, use the **no** form of this command.

vines arp-enable [**dynamic**]
no vines arp-enable [**dynamic**]

Syntax Description

dynamic	(Optional) Responds to ARP and SARP requests on this interface only if there are no other VINES servers present.

Default

The interface always responds to ARP and SARP requests.

Command Mode

Interface configuration

Usage Guidelines

Client systems on VINES networks are assigned network addresses dynamically. When a VINES client boots, it has no knowledge of their addresses and preferred servers. Immediately after it initializes its hardware interface, the client sends broadcast requests asking a server to provide it with a network-layer address. In a network that has a server, our routers do not normally respond to these broadcast requests. However, on a network that has only clients and no servers (called a serverless network), the router does need to respond to the broadcast requests so that all the clients on that serverless network can acquire network addresses. By default, the router will respond to ARP requests and assign addresses to network clients only if there is no VINES server present on that network segment. When it does, the router then acts as a network communication service provider for the client. You may configure the router to respond to these requests even if a VINES servers is present, or never to respond to these requests. If the router assigns an address, it will generate a unique network number based on its own VINES address.

A VINES file server must still be present somewhere on the network in order for the client to continue the booting process.

Example

The following example configures a router when Ethernet interface 1 is a network that does not contain any VINES servers:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
```

The following example configures a router to always provide ARP service on Ethernet interface 1, even when VINES servers are present on that network:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
vines arp-enable
```

Related Commands

vines propagate
vines serverless

vines decimal

To display VINES addresses in decimal notation, use the **vines decimal** global configuration command. To return to displaying the addresses in hexadecimal, use the **no** form of this command.

vines decimal
no vines decimal

Syntax Description

This command has no arguments or keywords.

Default

Addresses are displayed in hexadecimal.

Command Mode

Global configuration

Usage Guidelines

When displaying addresses, the router always uses a name if one has been configured via the **vines host** command. The **vines decimal** command affects the radix in which the address is presented when a name is not available.

Example

The following example displays VINES addresses in decimal:

```
vines decimal
```

Related Commands

clear vines cache
clear vines neighbor
clear vines route
show vines cache
vines host

vines encapsulation

To set the MAC-level encapsulation used for VINES broadcast packets, use the **vines encapsulation** interface configuration command. To disable encapsulation, use the **no** form of this command.

vines encapsulation [**arpa** | **snap** | **vines-tr**]
no vines encapsulation

Syntax Description

arpa	(Optional) ARPA encapsulation. This is the default encapsulation for Ethernet interfaces.
snap	(Optional) SNAP encapsulation. This encapsulation uses an IEEE 802.2 SNAP header. It is the default encapsulation for all media except Ethernet and Token Ring.
vines-tr	(Optional) Our VINES Token Ring encapsulation. This is the default encapsulation for Token Ring interfaces.

Default

ARPA encapsulation for Ethernet
VINES-TR Token Ring encapsulation for Token Ring
SNAP encapsulation for all other media

Command Mode

Interface configuration

Usage Guidelines

You can choose a MAC-level encapsulation type for each Ethernet, Token Ring, or IEEE 802.2 interface.

Setting the MAC-level encapsulation type with the **vines encapsulation** command affects broadcast packets sent by the router. The router keeps track of which encapsulation is used by each of its neighbors and uses the same style of encapsulation when talking directly to a neighbor.

You should not use this command with the current versions of VINES software that are available. This command is present for future interoperability when Banyan begins using encapsulations other than the current default ones.

Example

The following example configures IEEE 802.2 SNAP encapsulation on Ethernet interface 0:

```
vines routing
!
interface ethernet 0
vines metric 2
vines encapsulation snap
```

vines enhancements

To enable split-horizon for routing updates and to generate flash updates, use the **vines enhancements v**global configuration command. To turn VINES enhancement off, use the **no** form of this command.

vines enhancements
no vines enhancements

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The **vines enhancements** command applies only to non-sequenced RTP protocol and deals only with routing updates.

The **vines enhancements** command helps VINES enable split horizon when generating regular, periodic routing updates (routes learned from an interface are not advertised back to the same interface) and helps generates flash updates that are sent out to indicate topology changes for only the changed routes instead of full updates. Full updates are still sent periodically. When periodic or flash updates are sent out on a given interface, the updates do not include any information that was originally sent from that interface. This behavior is slightly different from the VINES server running pre-5.5.*x* Banyan VINES OS, which sends out the entire routing table in a flash update, even if only one route changed.

Note For routing updates only, when **vines enhancements** is enabled in global configuration mode by default, **vines split-horizon** is also enabled on the interface by default. In this case, if required, you can disable **vines split-horizon** on an interface like Frame Relay and X.25.

When **vines enhancements** is disabled in global configuration mode, **vines split-horizon** for RTP routing updates is disabled on all interfaces; however, one may still see **vines split-horizon** as enabled on the VINES interface when **show vines interface** *interface* command is entered. Split horizon remains enabled because **vines split-horizon** on individual VINES interface, in addition to controlling RTP updates, also controls whether or not retransmission of broadcasts is permitted on the receiving interface.

Example

The following example specifies the VINES enhancements:

```
vines routing
vines enhancements
```

Related Commands

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

vines split-horizon
vines update deltas
vines update interval

vines host

To associate a host name with a VINES address, use the **vines host** global configuration command. To delete the association, use the **no** form of this command.

vines host *name address*
no vines host *name*

Syntax Description

name	VINES host name. It can be any length and sequence of characters separated by white space.
address	Number of a VINES network. You enter it in the current VINES radix, in the format *network*:*host*, where *network* is 4 bytes and *host* is 2 bytes.

Default

Hosts are displayed by address.

Command Mode

Global configuration

Usage Guidelines

The router maintains a table of the mappings between host names and addresses.

When displaying addresses, the router uses the name instead of the numerical address if you have configured one with the **vines host** command.

Our software provides only static name-to-address bindings for the VINES protocol. This is completely separate from Banyan's distributed naming system, StreetTalk. The router does not learn names from StreetTalk, nor does the router provide names to StreetTalk.

Example

The following example assigns names to four VINES servers:

```
! cisco names
vines host FARSLAYER 30002A2D:0001
vines host DOOMGIVER 30000A83:0001
! VINES PS/2 server
vines host COINSPINNER 0027AF92:0001
! PC clone client
vines host STUFF 0027AF92:8001
```

Related Commands

clear vines neighbor
clear vines route
show vines host
vines decimal

vines input-network-filter

To filter the information contained in routing messages received from other stations, use the **vines input-network-filter** interface configuration command. To disable this filtering, use the **no** form of this command.

vines input-network-filter *access-list-number*
no vines input-network-filter

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

No filtering.

Command Mode

Interface configuration

Usage Guidelines

VINES routing messages contain topological entries that allow service and client nodes to select the best paths to destinations. This command provides filtering ability to an administrator so that they may selectively determine which routing entries should be accepted from other routers and which routing entries should be dropped. This command may be useful in enforcing administrative policies of local server usage.

Example

The following example prevents a route to one specific server from ever being learned via Ethernet interface 0:

```
vines routing
!
vines access-list 201 deny 27AF9A:1 0:0
vines access-list 201 permit 0:0 FFFFFFFF:FFFF
!
interface ethernet 0
vines metric 2
vines input-network-filter 201
```

vines input-router-filter

To filter received routing messages based upon the address of the sending station, use the **vines input-router-filter** interface configuration command. To disable this filtering, use the **no** form of this command.

vines input-router-filter *access-list-number*
no vines input-router-filter

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

No filtering.

Command Mode

Interface configuration

Usage Guidelines

VINES routing messages contain topological entries that allow service and client nodes to select the best paths to destinations. This command provides filtering ability to an administrator so that they may selectively determine the routers from which routing entries will be accepted.

Example

The following example prevents the router from ever learning routing information from one specific server on Ethernet interface 0:

```
vines routing
!
vines access-list 201 deny 27AF9A:1 0:0
vines access-list 201 permit 0:0 FFFFFFFF:FFFF
!
interface ethernet 0
vines metric 2
vines input-router-filter 201
```

vines metric

To enable VINES routing on an interface, use the **vines metric** interface configuration command. To disable VINES routing, use the **no** form of this command.

vines metric [*whole* [*fractional*]]
no vines metric

Syntax Description

whole	(Optional) Integer cost value associated with the interface. It is optional for all interface types. If you omit *whole*, the router automatically chooses a reasonable value. These values are listed in Table 16-13. For additional information, refer to the discussion and table in the "Usage Guidelines" section. If whole is zero, then a fractional portion must be supplied.
fractional	(Optional) Fractional cost value associated with the interface expressed in 10,000ths. It is optional for all interface types, but may only be present if a whole number portion is specified. This number will be rounded to the nearest 1/16. If you omit both whole and fractional numbers, the router automatically chooses a reasonable value. These values are listed in Table 16-13. For additional information, refer to the discussion and table in the "Usage Guidelines" section.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

The metric is the cost value associated with the interface media type. It is generally inversely proportional to the speed of the interface. The lower the delay metric, the more likely it is that the router will use that interface.

Our router automatically chooses a reasonable metric. These numbers match as closely as possible the numbers a Banyan server would choose for an interface of the same type and speed.

When enabling VINES for a serial interface, you should keep in mind that the VINES metric is based upon the configured bandwidth for the interface. To insure that the router selects the correct VINES metric, you need to make sure that the correct bandwidth has been configured. To do this, first issue the **show interface** command to determine the speed of the interface. Then issue the **bandwidth** command to set the bandwidth rate that is appropriate for that interface type and speed.After that, issue the **vines metric** command and the router will choose a metric appropriate to that speed. If you do not issue the **bandwidth** command first, you will need to either reissue the **vines metric** command or issue it with a metric number to get an appropriate metric.

Banyan servers use these metrics to compute timeouts when communicating with other hosts. If you do specify a metric, be careful that you do not set this number too high or too low. Doing so could disrupt the normal function of the Banyan servers.

Table 16-13 lists some example delay metric values.

Table 16-13 Example Delay Metric Values

Interface Type	Old Format	New Internal Format	New Configuration File Format	Seconds
FDDI	1	0010	1 0000	0.2000
Ethernet	2	0020	2 0000	0.4000
16-Mb Token Ring	2	0020	2 0000	0.4000
4-Mb Token Ring	4	0040	4 0000	0.8000
T1 HDLC	35	0230	35 0000	7.0000
56-kb HDLC	45	02D0	45 0000	9.0000
9600 baud HDLC	90	05A0	90 0000	18.0000
4800 baud HDLC	150	0960	150 0000	30.0000
2400 baud HDLC	250	0F00	250 0000	50.0000
1200 baud HDLC	450	1C20	450 0000	90.0000
T1 X.25	45	02D0	45 0000	9.0000
56-kb X.25	55	0370	55 0000	11.0000
9600 baud X.25	100	0640	100 0000	20.0000
4800 baud X.25	160	0A00	160 0000	32.0000
2400 baud X.25	260	1040	260 0000	52.0000
1200 baud X.25	460	1CC0	460 0000	92.0000

Examples

The following example enables VINES routing on Ethernet interface 0 and sets the metric to 2:

```
vines routing
!
interface ethernet 0
vines metric 2
```

The following example enables VINES routing on FDDI interface 0 and sets the metric to 0.25:

```
vines routing
!
interface fddi 0
vines metric 0 2500
```

Related Commands

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

bandwidth †
vines routing
vines update deltas
vines update interval

vines neighbor

To specify a static path to a neighbor station, use the **vines neighbor** interface configuration command. To remove a static path from the neighbor table, use the **no** form of this command.

vines neighbor *address mac-address encapsulation* [*whole* [f*ractional*]]
no vines neighbor *address mac-address*

Syntax Description

address	VINES IP address of the station to which to add or remove a static path.
mac-address	MAC-level address used to reach the neighbor station.
encapsulation	Encapsulation type to use on the media. It can be one of the following values: • **arpa**—Use ARPA encapsulation. This is recommended for Ethernet interfaces. • **snap**—Use an IEEE 802.2 SNAP header. This is recommended for FDDI interfaces. • **vines-tr**—Use our VINES Token Ring encapsulation. This is recommended for Token Ring interfaces.
whole	(Optional) Delay metric to use on the neighbor. If you omit this argument, the metric used is that specified with the **vines metric** command for the selected interface.
fractional	(Optional) Fractional metric value associated with this neighbor. This number will be rounded to the nearest 1/16. If you omit both whole and fractional numbers, then the interface metric will be used.

Default

No static paths are specified.

Command Mode

Interface configuration

Usage Guidelines

You can configure static neighbor entries only on Ethernet, FDDI, and Token Ring interfaces.

The decision to use a static path or a dynamic path is always determined by the relative metric numbers.

Be careful when assigning static paths. If a static path is assigned with a better metric than the dynamic paths and the link associated with the static path is lost, traffic may stop being forwarded, even though an alternative path might be available.

The metric is the cost value associated with the interface media type. It is generally inversely proportional to the speed of the interface. The lower the delay metric, the more like it is that the router will use that interface.

This command is useful for testing VINES networks with test equipment that does not generate hello packets.

Table 16-14 lists some example delay metric values.

Table 16-14 Example Delay Metric Values

Interface Type	Old Format	New Internal Format	New Configuration File Format	Seconds
FDDI`	1	0010	1 0000	0.2000
Ethernet	2	0020	2 0000	0.4000
16-Mb Token Ring	2	0020	2 0000	0.4000
4-Mb Token Ring	4	0040	4 0000	0.8000
T1 HDLC	35	0230	35 0000	7.0000
56-kb HDLC	45	02D0	45 0000	9.0000
9600 baud HDLC	90	05A0	90 0000	18.0000
4800 baud HDLC	150	0960	150 0000	30.0000
2400 baud HDLC	250	0F00	250 0000	50.0000
1200 baud HDLC	450	1C20	450 0000	90.0000
T1 X.25	45	02D0	45 0000	9.0000
56-kb X.25	55	0370	55 0000	11.0000
9600 baud X.25	100	0640	100 0000	20.0000
4800 baud X.25	160	0A00	160 0000	32.0000
2400 baud X.25	260	1040	260 0000	52.0000
1200 baud X.25	460	1CC0	460 0000	92.0000

Example

The following example defines a static path to the neighbor station at address 12345678:0001 using ARPA encapsulation:

```
interface ethernet 0
vines neighbor 12345678:0001 0001.0002.0003 arpa 20
```

Related Commands

clear vines neighbor
show vines neighbor
show vines route
vines route

vines output-network-filter

To filter the information contained in routing updates transmitted to other stations, use the **vines output-network-filter** interface configuration command. To disable this filtering, use the **no** form of this command.

vines output-network-filter *access-list-number*
no vines output-network-filter

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

No filtering.

Command Mode

Interface configuration

Usage Guidelines

VINES routing messages contain topological entries that allow service and client nodes to select the best paths to destinations. This command provides filtering ability to an administrator so that they may selectively determine which routing entries should be passed on to other routers. This command may be useful in enforcing administrative policies of local server usage.

Example

The following example prevents all routes from being advertised to Ethernet interface 0 except the route to one single server:

```
vines routing
!
vines access-list 201 permit 27AF9A:1 0:0
vines access-list 201 deny 0:0 FFFFFFFF:FFFF
!
interface ethernet 0
vines metric 2
vines output-network-filter 201
```

vines propagate

To modify how routers forward a broadcast packet, use the **vines propagate** interface configuration command. To return to the default forwarding scheme, use the **dynamic** form of this command.

vines propagate [**dynamic**]
no vines propagate [**dynamic**]

Syntax Description

dynamic	(Optional) Propagate broadcasts on this interface only if there are no servers on any local network.

Default

Dynamic forwarding

Command Mode

Interface configuration

Usage Guidelines

If you specify the **vines propagate** command with no keywords, broadcast messages are always propagated on the interface.

The **vines propagate** command affects how the router decides whether to forward a broadcast packet out an interface. The normal decision is based on the settings of both the "hop count" and "class" fields of the VINES IP header, and also whether or not there are any servers present on any of the local network segments. In the default configuration, the router first looks to see if there are any local servers, and if so, follows the normal rules of VINES IP and forwards the broadcast out this interface based upon the "hop count" and the "class" field. If there are no local servers, then the router looks only at the "hop count" field before forwarding the broadcast out this interface. Enabling this command with no argument tells the router to always ignore the "class" field and make the forwarding decision based solely upon the "hop count" field. The **no** form of this command tells the router to always examine both the "hop count" and "class" fields.

Example

The following example always ignores the "class" field of the VINES IP header when deciding whether to forward a broadcast packet on serial interface 0:

```
interface serial 0
vines propagate
```

Related Commands

vines arp-enable
vines serverless

vines redirect

To determine how frequently a router sends an RTP redirect message on an interface, use the **vines redirect** interface configuration command. To restore the default, use the **no** form of this command.

vines redirect [*seconds*]
no vines redirect

Syntax Description

seconds	(Optional) Interval, in seconds, that the router waits after sending a redirect message on an interface before it sends another redirect message on that same interface. If you specify a value of 0, the router never sends redirect messages on that interface.

Default

1 second

Command Mode

Interface configuration

Usage Guidelines

VINES routing redirect packets contain topological entries that allow service and client nodes to select the best paths to destinations. When a service node determines that it should not be forwarding packets between two nodes, it sends a redirect packet to the sending node informing it of the better path.

Example

The following example prevents redirect messages from ever being sent on Ethernet interface 0:

```
vines routing
!
interface ethernet 0
vines metric 2
vines redirect 0
```

vines route

To specify a static route to a server, use the **vines route** global configuration command. To remove a static route from the routing table, use the **no** form of this command.

vines route *number address* [*whole* [*fractional*]]
no vines route *number address* [*whole* [*fractional*]]

Syntax Description

number	Number of the server to which to add or remove the static route.
address	VINES IP address of the neighbor station to use to reach the server.
whole	(Optional) Metric value assigned to this route.
fractional	(Optional) Fractional cost value associated with this route.

Default

No static routes are specified.

Command Mode

Global configuration

Usage Guidelines

The decision to use a static route or a dynamic route is always determined by the relative metric numbers.

Be careful when assigning static routes. If a static route is assigned with a better metric than the dynamic routes and the links associated with the static routes are lost, traffic may stop being forwarded, even though an alternative route might be available.

Floating static routes are static routes that can be overridden by dynamically learned routes. Floating static routes allow you to switch to another path whenever routing information for a destination is lost. One application of floating static routes is to provide back-up routes in topologies where dial-on-demand routing is used.

To configure a floating static route, assign a metric to the static route that is worse (higher) than all dynamic routes. If you configure a floating static route, the router checks to see if an entry for the route already exists in its routing table. If a dynamic route already exists, the floating static route is placed in reserve as part of a floating static route table. When the router detects that the dynamic route is no longer available, it replaces the dynamic route with the floating static route for that destination. If the route is later relearned dynamically, the dynamic route replaces the floating static route and the floating static route is again placed in reserve.

Note By default floating static routes are not redistributed into other dynamic protocols.

Examples

The following example establishes a static route to the server at ABCD1234:

```
vines route ABCD1234 12345678:1 35
```

The following example establishes a floating static route to the server at 3000000:

```
vines route 3000000 3001000:1
```

Related Commands

clear vines neighbor
clear vines route
show vines neighbor
show vines route
vines neighbor
vines output-network filter

vines route-cache

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

vines route-cache
no vines route-cache

Syntax Description

The command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

The **vines route-cache** command enables the fast switching of VINES packets being transmitted out of the interface. However, forwarding of broadcast packets and responding to packets destined for the local router still occurs at the process level. When fast switching is disabled, all packets are forwarded at the process level.

Fast switching allows higher throughput by switching a packet using a cache created by previous packets. Fast switching provides load sharing on a per-packet basis just as slow switching does. Fast switching is enabled by default on all interfaces where it is supported. It is not supported on very old Ethernet, serial, and Token Ring interfaces, nor is it supported on serial interfaces using an encapsulation other than HDLC.

Packet transfer performance is generally better when fast switching is enabled. However, you may want to disable fast switching in order to save memory space on interface cards and help avoid congestion when high-bandwidth interfaces are writing large amounts of information to low-bandwidth interfaces.

When fast switching is enabled, the router maintains a fast-switching cache table. When transmitting a packet that is eligible to be fast switched, the router first checks the fast-switching cache table. If it finds an entry for the destination, the router uses that path. Otherwise, it searches the standard routing table and places the route it finds into the fast-switching cache table. The next time the router receives a packet for that destination, it uses the route in the fast-switching cache table.

Example

The following example disables fast switching on serial interface 0:

```
interface serial 0
bandwidth 19200
vines metric
no vines route-cache
```

Related Commands

clear vines cache
show vines cache
show vines route

vines routing

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

vines routing [*address* | **recompute**]
no vines routing

Syntax Description

address	(Optional) Network address of the router. You should specify an address on a router that does not have any Ethernet or FDDI interfaces. You also can specify an address in the unlikely event that two routers map themselves to the same address.
recompute	(Optional) Dynamically redetermine the router's network address.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

Enabling VINES routing with the **vines routing** command starts both the VINES RTP and SRTP protocols. The router software dynamically determines which version of the VINES routing protocol stations on the network are using and then uses one or the other, or both protocols, as appropriate.

If a router contains Ethernet or FDDI interfaces, you do not need to specify an address because the router automatically maps itself into the VINES address space that is reserved for our routers. If you do specify an address, the router will use the specified address.

If a router contains only Token Ring interfaces (or Token Ring and serial interfaces), either the Token Ring interface must be fully initialized before you issue the **vines routing** command or you must specify an address in the **vines routing** command. This is because Token Ring interfaces have MAC addresses of 0000.0000.0000 until they are fully initialized.

Banyan has assigned us a portion of the overall VINES network number space. This portion is the set of all numbers that begin with the first 11 bits (of the 32) of 0011 0000 000. This number set appears in all our displays as a hexadecimal number beginning with 0x300 or 0x301. Routers attempt to automatically map themselves into our number space based upon the first nonzero Ethernet, Token Ring, or FDDI address found.

In theory, address conflicts are impossible, because VINES servers use their Banyan-assigned, unique key serial numbers as their network numbers and use a subnetwork number of one. Because the keys are unique, the server addresses are unique. VINES clients do not have addresses per se. The clients use a modified version of the address of the first file server found on the physical network: they assume the server's network number and are assigned a subnetwork number by that server. This address-assignment scheme means that it is likely that two clients on the same physical LAN will have different addresses. It requires that the router keep a cache of local neighbors as well as a cache of routing entries.

If you do not specify a network address and the router cannot compute one from a MAC address, the router selects a random address. There is no guarantee that this will be a unique address.

If you find that two routers have the same VINES network address, you should issue the **vines routing recompute** command on both routers. When recomputing its address, the router uses the same method used when originally determining its network address. If you issue this command on a router on which you have enabled the processing of ARP packets (with the **vines arp-enable** command) and if the router's address changes when it is recomputed, any clients that received their VINES network addresses from the router will lose all network connectivity, and you will have to reboot them.

Older implementations of out software mapped themselves to numbers beginning with 0xF80. This was done before Banyan made the address assignment.

Example

The following example enables VINES routing on Ethernet interface 0:

```
vines routing
!
interface ethernet 0
vines metric 2
```

Related Commands

vines arp-enable
vines metric

vines serverless

To configure a Banyan VINES network that does not have a server, use the **vines serverless** interface configuration command. To disable this feature, use the **no** form of this command.

vines serverless [**dynamic | broadcast**]
no vines serverless [**dynamic | broadcast**]

Syntax Description

dynamic	(Optional) Forward broadcasts toward one server only if there are no servers present on this interface.
broadcast	(Optional) Always flood broadcasts out all other router interfaces in order to reach all servers.

Default

Dynamic forwarding

Command Mode

Interface configuration

Usage Guidelines

If all keywords are omitted, broadcasts are always forwarded toward one server.

The **vines serverless** command provides special processing for certain broadcast packets and certain packets directed at the router.

When you have a Banyan VINES network that has no server, by default the router will provide special processing for certain broadcast packets and certain packets directed at the router. This is necessary for proper functioning of the clients on a network without a server. This special processing allows a client to find the services that are provided by a server on another network. The dynamic nature of this processing allows the router to switch over from not providing serverless support to providing serverless support if the last server on a network fails. If you want the router to always provide serverless support, even when there are local servers present, you may override the default processing by issuing the **vines serverless** command with no argument. If you do not want the router to ever provide serverless support, you may also override the default in this way by issuing the **no vines serverless** command.

When the router receives a zero-hop broadcast on a serverless network, it does not follow the normal processing rules for VINES packets and discard the frames. Instead, it looks in its routing table for the nearest Banyan server. If this server is on a directly connected network, the router resends the broadcast message on that network as a MAC-level broadcast so that server and any others present can respond to it. If the nearest Banyan server is not on a directly connected network, the router resends the broadcast message on that network as a MAC-level unicast message directed at the first hop to that server. The next router will perform these same steps, assuming it is also configured for serverless support. The router can also be configured to always flood these broadcasts on all interfaces by using the command **vines serverless broadcast**. The decision on whether or not to flood is a trade-off between network bandwidth and finding more servers.

If you have configured this interface to forward towards a single destination, you may see which server has been selected as the forwarding target by looking at the output of the **show vines route** command. All servers on the same physical network as the target server will receive the broadcast.

Examples

The following example configures Ethernet interface 1, which is a network with no VINES servers:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
```

Note The **vines serverless** command is not necessary because the default setting is what is desired.

The following example configures Ethernet interface 1, which is a network with no VINES servers to always flood broadcasts to all other interfaces in the router:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
vines serverless broadcast
```

The **vines serverless** command is necessary here because a nondefault setting is desired.

Related Commands

show vines route
vines arp-enable
vines propagate

vines single-route

To maintain only one route per server, use the **vines single-route** global configuration command. To allow multiple routes per server, use the **no** form of this command.

vines single-route
no vines single route

Syntax Description

This command has no arguments or keywords.

Default

Multiple routes

Command Mode

Global configuration

Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

VINES servers and clients do not handle out-of-sequence packets well. If a VINES connection experiences slow performances due to low window size, enable the **vines single-route** command. This command can be enabled at any time after VINES routing has been enabled.

Example

The following example specifies the VINES single route:

```
vines routing
vines single-route
```

Related Command

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

vines routing

vines split-horizon

To use split horizon when sending routing updates, use the **vines split-horizon** interface configuration command. To disable split horizon, use the **no** form of this command.

vines split-horizon
no vines split-horizon

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

The **vines split-horizon** command also affects whether broadcasts packets received on an interface are resent on the same interface.

The **vines split-horizon** command determines how much information is included in routing updates sent out an interface. It also determines whether received broadcasts will be retransmitted on the same interface. When you enable split horizon, routing updates sent out on a given interface will not include any information that was originally learned from that interface, and broadcasts will not be retransmitted on the receiving interface. This is because split horizon is designed for networks that are either broadcast networks, or are fully connected mesh networks. In these types of networks, resending this information is a waste of network bandwidth because all other stations on that network have already heard the information. Disabling split horizon will cause the router to include all information in routing updates, and to resend broadcast packets on the network from which they were received.

Note When **vines enhancements** is disabled in global configuration mode, **vines split-horizon** for RTP routing updates is disabled on all interfaces; however, one may still see **vines split-horizon** as enabled on the VINES interface when **show vines interface** *interface* command is entered. Split horizon remains enabled because **vines split-horizon** on individual VINES interface, in addition to controlling RTP updates, also controls whether or not retransmission of broadcasts is permitted on the receiving interface.

You can use this command on any interface, but generally it makes sense to use it only for X.25 and Frame Relay interfaces. You should disable split horizon on X.25 and Frame Relay networks that are not fully connected mesh topologies.

Example

The following example disables split horizon on an X.25 network:

```
interface serial 0
vines metric 2
no vines split-horizon
```

Related Command

vines enhancements

vines srtp-enabled

To enable Sequenced Routing Update Protocol (SRTP), use the **vines srtp-enabled** global configuration command. To disable SRTP, use the **no** form of this command.

vines srtp-enabled
no vines srtp-enabled

Syntax Description

This command has no arguments or keywords.

Default

The router runs Banyan's Routing Update Protocol (RTP) routing protocol only.

Command Mode

Global configuration

Usage Guidelines

When SRTP is enabled, the router dynamically determines whether it needs to send RTP messages, SRTP messages, or both.

Example

The following example enables SRTP on the router:

```
interface serial 0
vines routing
vines srtp-enabled
```

Related Command

vines routing

vines time access-group

To control the servers from which the router will accept VINES network time, use the **vines time access-group** global configuration command. To accept VINES network time messages from any server, use the **no** form of this command.

vines time access-group *access-list-number*
no vines time access-group

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

Disabled

Command Mode

Global configuration

Example

The following example applies an access list to incoming time messages:

```
vines access-list 201 permit 27AF9A:1 0:0
vines access-list 201 deny 0:0 FFFFFFFF:FFFF
!
vines time participate
vines time access-group 201
```

Related Commands

show vines service
vines access-list (simple)
vines time destination
vines time participate
vines time set-system
vines time use-system

vines time destination

To control the servers to which the router sends VINES network time, use the **vines time destination** global configuration command. To send VINES network time messages to all servers, use the **no** form of this command.

vines time destination *address*
no vines time destination

Syntax Description

address	Destination VINES address for the network time messages.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

By default, the router sends VINES network time messages to the broadcast address.

You can enter the **vines time destination** command up to 20 times for 20 destination addresses.

Example

The following example specifies the servers to receive VINES time messages:

```
vines time participate
vines time destination 0027AF9F:0001
vines time destination 300001239:001
```

Related Commands

show vines service
vines time access-group
vines time participate
vines time set-system
vines time use-system

vines time participate

To enable the router's participation in the synchronization of time across a VINES network, use the **vines time participate** global configuration command. To disable the router's participation in time synchronization, use the **no** form of this command.

vines time participate
no vines time participate

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

The router always listens to the time synchronization messages on the network, and it tracks the network time. This command controls only the sending of time synchronization messages by the router. This arrangement means that you can use the **show vines services** EXEC command to see the network time even if the router is not actively participating in time synchronization.

Example

The following example disables the router's participation in the sending of VINES time messages:

```
no vines time participate
```

Related Commands

show vines service
vines access-list (simple)
vines access-group
vines time destination
vines time set-system
vines time use-system

vines time set-system

To set the router's internal time based upon the received VINES network time, use the **vines time set-system** global configuration command. To uncouple the router's time from VINES network time, use the **no** form of this command.

vines time set-system
no vines time set-system

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

You should not use the **vines time set-system** command when running NTP on a router, because this command has no effect on these systems. NTP is considered to be a higher-priority clock than VINES, because it is a much more accurate timekeeping system.

Example

The following example sets the router's time from received VINES time messages:

```
vines time participate
vines time set-system
```

Related Commands

show vines service
vines access-list (simple)
vines time destination
vines time participate
vines time use-system

vines time use-system

To set VINES network time based upon the router's internal time, use the **vines time use-system** global configuration command. To uncouple VINES network time from the router's time, use the **no** form of this command.

vines time use-system
no vines time use-system

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **vines time use-system** command causes the router to import the locally available time source (such as NTP, the Cisco 7000 clock, or DNSIX time) into the VINES time world as an authoritative clock. This is most useful when running NTP on the router. The router appears to the VINES network as a server dialing the NIST clock.

When you specify the **vines time use-system** command, VINES will extract the system time and propagate it into the VINES world only if the system time is valid. If you are running NTP, the system time becomes valid when NTP synchronizes with a master. If you are not running NTP, but you do have an internal clock (such as exists on the Cisco 7000 router), you can force that time to be valid by specifying the **clock calendar-valid** command. This will allow VINES to propagate time based upon the Cisco 7000's clock chip.

Example

The following example sets VINES network time from the router's internal time:

```
ntp peer 131.108.13.111 version 2
!
vines time participate
vines time use-system
```

Related Commands

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

clock calendar-valid †
show vines service
vines access-list (simple)
vines time access-group
vines time destination
vines time participate
vines time set-system

vines update deltas

To modify the manner in which routing updates are sent, use the **vines update deltas** interface configuration command. To return to the default method, use the **no** form of this command.

vines update deltas
no vines update deltas

Syntax Description

This command has no arguments or keywords.

Default

No deltas

Command Mode

Interface configuration

Usage Guidelines

The **vines update deltas** command significantly modifies the way that routing information is propagated across the network.

On LAN media, using this command causes the router to stop transmitting and to stop expecting periodic routing updates. Instead, the router transmits and expects a periodic hello message. The difference between these two messages is whether routing information is included. The router will continue to send flash updates to inform its neighbors of any changes to current routing table information. This is the same frequency and type of routing updates used on LANs by VINES version 5.50, but our packet format differs from the VINES format.

On WAN media, using this command causes the router to transmit three normally spaced routing updates and then cease transmission. The router does *not* send periodic hello messages. The router will, however, continue to send flash updates to inform its neighbors of any changes to current routing table information. This is the same frequency and type of routing updates used on LANs by all versions of VINES, but our packet format differs from the VINES format.

Example

The following example modifies the propagation of routing update information on the WAN interface connected to serial interface 0:

```
interface serial 0
vines metric
vines update deltas
```

Related Commands

show vines interface
show vines neighbor
show vines route
vines metric

vines update interval

To modify the frequency at which routing updates are sent, use the **vines update interval** interface configuration command. To return to the default frequency, use the **no** form of this command.

vines update interval [*seconds*]
no vines update interval [*seconds*]

Syntax Description

seconds	Interval, in seconds, between the sending of periodic VINES routing updates. This can be a number in the range 0 to 2^{32} and will be rounded up to the nearest 5 seconds. The default value is 90 seconds. If you omit *seconds* or specify a value of 0, the default value of 90 seconds is used.

Default

90 seconds

Command Mode

Interface configuration

Usage Guidelines

The **vines update interval** command controls the interval at which the router sends routing updates. The routing update interval should be the same on all VINES-speaking entities on the same physical network.

For networks on which other vendors' entities are present, it is safe to use any setting in the range 30 to 100 seconds on networks. This is the range of update intervals supported by Banyan servers. You should use values outside of this range (with the exception of zero) only on networks that contain only our routers. You can use a value of zero on networks with only our routers or on WAN links connecting our routers and Banyan servers. In this configuration, you must also address application-level security requirements.

For Banyan VINES sites that support "change-only" updates on LAN networks, you can use the **vines update interval** command in LAN networks with both our routers and Banyan servers.

Example

The following example sets the update interval on serial interface 0 to a value of 270 seconds:

```
interface serial 0
vines metric
vines update interval 270
```

Related Commands

show vines interface
show vines neighbor
show vines route
vines metric