

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 also is 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 of the *Router Products Configuration Guide*.

clear vines cache

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

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

Syntax Description

no argument	Deletes the entire fast-switching cache.
interface <i>interface</i>	(Optional.) Deletes from the fast-switching cache table any entry that has one or more paths that go through the specified interface.
neighbor <i>address</i>	(Optional.) Deletes from the fast-switching cache table any entry that has one or more paths via the specified neighbor router.
server <i>network</i>	(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 <i>network</i> 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

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

Examples

The following example deletes from the fast-switching cache table 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:1:

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

Related Commands

show vines cache
vines decimal
vines route-cache

clear vines neighbor

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

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

Syntax Description

<i>network</i>	Network number of the neighbor whose entry should be deleted from the neighbor table. The argument <i>network</i> 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).

* Delete 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 neighbor
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 zeros 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 hostname | [vines] address]
```

Syntax Description

vines	(Optional.) Specifies the VINES protocol. If you omit this keyword, the router prompts for it.
<i>address</i>	(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 Display

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

no argument	Displays all VINES access lists.
<i>access-list-number</i>	(Optional.) Number of the access list to display.

Command Mode

EXEC

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 1-1 describes the fields shown in the display.

Table 1-1 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 (standard)
vines access-list (extended)
vines access-list (simple)

show vines cache

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

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

Syntax Description

no arguments	Displays all entries in the VINES fast-switching cache table.
<i>address</i>	(Optional.) Displays the entry in the fast-switching cache table for the specified station.
interface <i>interface</i>	(Optional.) Displays all neighbors in the fast-switching cache table that are accessible via the specified interface. <i>interface</i> is the interface type and unit number. For example, specify Ethernet interface 0 as “ethernet0” or “e0.”
neighbor <i>address</i>	(Optional.) Displays all routes in the VINES fast-switching cache table that have the specified neighbor as their first hop. <i>address</i> is a 6-byte hexadecimal number in the format <i>network:host</i> , where <i>network</i> is 4 bytes and <i>host</i> 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 <i>network</i>	(Optional.) Displays all entries in the VINES fast-switching cache table that are in the specified logical network. <i>network</i> 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

Sample Display

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

```
router# show vines cache
Vines fast switching cache version is 36

Hash Destination      Int   Age  Length  MAC Header
13/00 Router1          *T0   46   16/18  10005A746A3600003080FB06BCBC03BA
27/00 Router2          E1    11   14/14  0000C01D87C00000C0158010BAD
                               *T0   11   16/18  00003000435500003080FB06BCBC03BA
3E/00 Router3          *T0   42   16/18  10005A6FBC1500003080FB06BCBC03BA
72/00 30002E6D:0001    E1    32   14/14  0000C01D87C00000C0158010BAD
                               *T0   32   16/18  00003000435500003080FB07BCBC03BA
                               T0    32   16/18  10005A6FBC1500003080FB06BCBC03BA
                               T0    32   16/18  10005A6FBC1500003080FB06BCBC03BA
FE/00 Router4          *E2   264   14/14  0000C0124EA00000C0151AF0BAD
```

Table 1-2 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 1-2 Show Vines Cache Field Descriptions

Field	Description
Vines fast switching cache version ...	Version number of the VINES fast-switching cache table. The number is incremented each time an entry is added to or deleted from this table.
Hash	Position of this entry in the neighbor table.
Destination	Name or address of the destination station.
Int	Interface out which the packet will be 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.
MAC Header	MAC header that will be used to reach the destination.

Related Commands

clear vines cache
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

no argument	Displays the entire contents of the VINES host name table.
<i>name</i>	(Optional.) Displays the entry in the VINES name table that has the specified name.

Command Mode

EXEC

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 1-3 describes the fields shown in the display.

Table 1-3 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 [interface unit]
```

Syntax Description

no arguments	Displays values for all interfaces, and displays all VINES global parameters.
<i>interface unit</i>	(Optional.) Displays values for a specific interface. The arguments are an interface type and a unit number. An example is "ethernet 0."

Command Mode

EXEC

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.
Hssi0 is down, line protocol is down
  VINES protocol processing disabled
Ethernet0 is up, line protocol is up
  VINES broadcast encapsulation is ARPA
  Interface metric is 2 (0.4 seconds)
  Split horizon is enabled
  ARP processing enabled
  Special serverless net processing enabled
  Outgoing access list is not set
  Fast switching is enabled
  Routing updates every 90 seconds. Next in 50 seconds.
Ethernet1 is up, line protocol is up
  VINES protocol processing disabled
```

Table 1-4 describes the fields that may be shown in the display.

Table 1-4 Show Vines Interface Field Descriptions

Field	Description
VINES address is ...	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.
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."

Field	Description
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 is ...	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.
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.”
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 contain ...	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.

Related Commands

- vines arp-enable**
- vines encapsulation**
- vines metric**
- vines route-cache**
- vines serverless**
- vines split-horizon**
- vines update deltas**
- vines update interval**

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 1-5 describes the fields shown in the display.

Table 1-5 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.
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.

Field	Description
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 in ...	Time, in seconds, until the next acknowledgment message is sent by the local host.
Retransmit in ...	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 interface | server number]
```

Syntax Description

no arguments	Displays all entries in the VINES neighbor table.
<i>address</i>	(Optional.) Displays the entry for the specified neighbor.
interface <i>interface</i>	(Optional.) Displays all neighbor paths in the neighbor table that use the specified interface.
server <i>number</i>	(Optional.) Displays all entries in the neighbor table that have the specified network number.

Command Mode

EXEC

Sample Display

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

Hash  Address          Hardware Address  Type  Int  Flg  Age  Metric  Uses
-----
27/00 Router1         -                HDLC  Se0  R*   n/a   35    7
68/00 Router2         -                -     -    C    -     -     -
FE/00 Router3         0000.0c01.24ea   ARPA  Et2  R*   42    2     9
1C/00 Router4         0000.00c00.2e6e  PPP   Et1  R    2     2     0
      -              0000.0c01.0506   ARPA  Ft0  R.   2     2     0
      -              0000.0c01.9ac9   VINES To0  R*   12    2     0
```

Table 1-6 describes the fields shown in the display.

Table 1-6 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.
Hash	Position of this entry in the neighbor table.
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.

Field	Description
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
Flg	<p>This field is a three-column fields.</p> <p>The first column indicates how the path was learned. It can be one of the following values:</p> <ul style="list-style-type: none"> • 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). <p>The second column indicates how this path will be used. It can be one of the following values:</p> <ul style="list-style-type: none"> • *—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. <p>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.</p> <p>The third 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.</p>
Age	Age of this VINES neighbor table entry, in seconds. In this example, interfaces Hssi0 and Ethernet 2 are both set for delta-only updates. The entry for Ethernet2 displays an age because periodic hello messages are sent from the stations on a LAN media. The entry for Hssi0 shows an age of n/a because there are no hello messages on WAN media.
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.

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

no arguments	Displays all entries in the VINES routing table.
<i>number</i>	(Optional.) Displays the routing table entry for the specified network.
neighbor <i>address</i>	(Optional.) Displays all routes in the VINES routing table that have the specified neighbor as their first hop.

Command Mode

EXEC

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

Hash  Network      Neighbor      Flags  Age  Metric  Uses
-----
0C/00 Router1       Router2       R*     n/a   37     0
12/00 0027AF9A   3000577A:0001 R*n    73    4     0
      -         30019AC7:0001 R      13    4     0
26/00 Router2       Router3       R*     n/a   35     0
3F/00 002ABFAA   3000577A:0001 R*     73    4     0
      -         30019AC7:0001 R      13    4     0
69/00 Router4       -             C      -     -     -
6C/00 30019AC7     30019AC7:0001 R*     n/a   37     0
FE/00 Router5       Router6       R*     14    4     0
FF/00 Router7       Router6       R*     n/a   2     0
```

Table 1-7 describes the fields shown in the display.

Table 1-7 Show VINES Route Field Descriptions

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

Field	Description
Flags	<p>This field is a series of single-column fields.</p> <p>The first column indicates how the route was learned. It can be one of the following values:</p> <ul style="list-style-type: none"> • 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). <p>An asterisk in the second column indicates that this route will be used next when forwarding a frame to that server.</p> <p>The third column indicates whether that route will be used to forward a broadcast from a serverless network. It can be one of the following values:</p> <ul style="list-style-type: none"> • 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. <p>The fourth column contains the letter “H” if the route is in a holddown state.</p> <p>The fifth 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.</p>
Age	<p>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.</p>
Metric	<p>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.</p>
Uses	<p>Number of times this route has been used to forward a packet.</p>

Related Commands

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

show vines services

To display information about the router's current time, use the **show vines services EXEC** command.

show vines services

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

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

```
router# show vines services
Server Service:
  Time is 22:20:20 UTC Apr 22 1993
  Time last adjusted (+1 sec) by Server1, 1:47:18 ago.
  Time epoch is SS@Aloe@Servers-132, started 3w3d ago.
  Next wakeup in 0:29:54
```

Table 1-8 describes the fields shown in the display.

Table 1-8 Show Vines Services Field Descriptions

Field	Description
Time is ...	Current time (in the format <i>hours:minutes:seconds</i>) and date.
Time last adjusted ...	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 <i>hours:minutes:seconds</i> . For times longer ago than 24 hours, the time format is <i>weekswdaysd</i> .
Time epoch is ...	Name of the current time epoch (in the format <i>name-number</i>), and when it was established.
Next wakeup in ...	Time, in seconds, until the router will send a "time synchronization" message.

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 [*interface unit*]

Syntax Description

no arguments	Displays values for all interfaces and display all VINES global parameters.
<i>interface unit</i>	(Optional.) Displays values for a specific interface. The arguments are an interface type and a unit number. An example is "ethernet 0."

Command Mode

EXEC

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
  ARP: Rcvd 0/0/0/0/0, Sent 0/0/0/0
  ICP: Rcvd 0/0/0, Sent 0/0
  IPC: Rcvd 17, Sent 8
  RTP: Rcvd 2/10/0/0/170/0/0/5, Sent 0/6/00/0/91/10/0
  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 helpared, 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
ARP: Rcvd 0/0/0/0/0, Sent 0/0/0/0
ICP: Rcvd 0/0/0, Send 0/0
IPC: Rcvd 0, Sent 8
RTP: Rcvd 0/10/0/0/0/0/0/0, Sent 0/0/00/0/0/0/0
SPP: Rcvd 0, Sent 0
Echo: Rcvd 0, Sent 0
Proxy: Rcvd 0, Sent 0

```

Table 1-9 describes the fields shown in the display.

Table 1-9 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).
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.

Field	Description
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 helpere," and "dup bcast" fields will not equal the total number of broadcast packets received.)
bcast helpere	Number of broadcast packets accepted by the router (as shown in the "bcast" field) that were "helpere" to a Banyan server. (Note that the sum of the "bcast forwarded," "bcast helpere," 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 helpere," 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.
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.

Field	Description
ARP:	This section displays statistics about VINES ARP packets sent and received by the router.
Rcvd <i>x/x/x/x/x</i>	Number of ARP packets received of type 0, 1, 2, 3, and other. The “other” field will be nonzero if there are VINES 5.50 clients on that LAN segment. Otherwise, this number should always be zero.
Sent <i>x/x/x/x</i>	Number of ARP packets send of type 0, 1, 2, and 3.
ICP:	This section displays statistics about VINES ICP packets sent and received by the router.
Rcvd <i>x/x/x</i>	Number of ICP packets received of type 0, 1, and other.
Sent <i>x/x</i>	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.
RTP:	This section displays statistics about VINES routing protocol (RTP) packets sent and received by the router.
Rcvd <i>x/x/x/x/x/x/x/x</i>	Number of RTP packets received of type 0, 1, 2, 3, 4, 5, 6, and other. The count of type 0 RTP packets will be nonzero if there are VINES 5.50 clients on that LAN segment. Otherwise, this number should always be zero. The counts of type 2, type 3, and other RTP packets should always be zero.
Sent <i>x/x/x/x/x/x/x/x</i>	Number of RTP packets sent of type 0, 1, 2, 3, 4, 5, and 6.
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 (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.

Field	Description
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] [address]
```

Syntax Description

vines (Optional.) Specifies the VINES protocol. If you omit this keyword, the router prompts for it.

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

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

The **trace** test characters are explained in Table 1-10.

Table 1-10 Trace Test Characters

Char	Meaning
<i>nm</i> msec	For each node, the round-trip time for each probe in <i>nm</i> milliseconds.
*	The probe timed out.
?	Unknown packet type.

Sample Display

The following is sample output from the VINES **trace** command:

```
router# trace 27AF92:1
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

router# trace
Protocol [ip]: vines
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
```

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

<i>access-list-number</i>	Number of the access list. All outgoing packets defined with either standard or extended access lists and forwarded through the interface are filtered by the entries in this access list. For standard access lists, <i>access-list-number</i> is a decimal number from 1 to 100. For extended access lists, <i>access-list-number</i> is a decimal number from 101 to 200.
---------------------------	--

Default

None

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 Commands

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

vines access-list (standard)

To specify a standard VINES access list, use the following 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-address-mask [source-port] destination-address destination-address-mask
[destination-port]
no vines access-list access-list-number
```

Syntax Description

<i>access-list-number</i>	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.
<i>protocol</i>	VINES protocol ID number or name. It can be a value from 1 to 255 or one of the following protocol keywords: <ul style="list-style-type: none"> • ARP—Address Resolution Protocol • ICP—Internet Control Protocol • IP—VINES Internet Protocol • IPC—Interprocess Communications • RTP—Routing Update Protocol • SPP—Sequence Packets Protocol
<i>source-address</i>	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format <i>network:host</i> , where <i>network</i> is 4 bytes and <i>host</i> is 2 bytes.
<i>source-mask</i>	Mask to be applied to <i>source-address</i> . 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.
<i>source-port</i>	(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 0xFFFFE. Table 1-13 in the “Usage Guidelines” section lists some IPC port numbers.
<i>destination-address</i>	Address of the network to which the packet is being sent. This is a 6-byte hexadecimal number in the format <i>network:host</i> , where <i>network</i> is 4 bytes and <i>host</i> is 2 bytes.

<i>destination-address-mask</i>	Mask to be applied to <i>destination-address</i> . 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.
<i>destination-port</i>	(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 1-11 in the “Usage Guidelines” section following lists some IPC port numbers.

Default

None

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 1-11.

Table 1-11 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
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
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 (extended)

vines access-list (simple)

vines access-list (extended)

To create an extended VINES access list, use the following 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-address-mask [source-port source-port-mask] destination-address
destination-address-mask [destination-port destination-port-mask]
no vines access-list access-list-number
```

Syntax Description

<i>access-list-number</i>	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.
<i>protocol</i>	VINES protocol ID number or name. The number can be a value from 1 to 255 or one of the following protocol keywords: <ul style="list-style-type: none"> • ARP—Address Resolution Protocol • ICP—Internet Control Protocol • IP—VINES Internet Protocol • IPC—Interprocess Communications • RTP—Routing Update Protocol • SPP—Sequence Packets Protocol
<i>source-address</i>	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format <i>network:host</i> , where <i>network</i> is 4 bytes and <i>host</i> is 2 bytes.
<i>source-mask</i>	Mask to be applied to <i>source-address</i> . 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.
<i>source-port</i>	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 1-12 in the “Usage Guidelines” section lists some IPC port numbers.
<i>source-port-mask</i>	(Optional.) Mask to be applied to <i>source-port</i> . 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.
<i>destination-address</i>	VINES address of the network to which the packet is being sent. This is a 6-byte hexadecimal number in the format <i>network:host</i> , where <i>network</i> is 4 bytes and <i>host</i> is 2 bytes.

<i>destination-address-mask</i>	Mask to be applied to <i>destination-address</i> . 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.
<i>destination-port</i>	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 1-12 in the “Usage Guidelines” section lists some IPC port numbers.
<i>destination-port-mask</i>	(Optional.) Mask to be applied to <i>destination-port</i> . 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.

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 1-12.

Table 1-12 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
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 (standard)

vines access-list (simple)

vines access-list (simple)

To create a simple VINES access list, use the following version of the **vines access-list** global configuration command. To remove a standard 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

<i>access-list-number</i>	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.
<i>source-address</i>	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format <i>network:host</i> , where <i>network</i> is 4 bytes and <i>host</i> is 2 bytes.
<i>source-mask</i>	Mask to be applied to <i>source-address</i> . 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.

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 accept time updates only from the stations on networks 30015800 and 30004355; it defines 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 access-group 201
```

Related Commands

- show vines access**
- vines access-list** (standard)
- vines access-list** (simple)
- vines access-group**
- vines time access-group**
- vines time participate**
- vines time set-system**
- vines time use-system**

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  
no vines arp-enable
```

Syntax Description

This command has no arguments or keywords.

Default

no vines arp-enable

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. Our routers normally do not respond to these broadcast requests. However, on a network that has only clients and no servers (called a serverless network), the router needs to respond to the broadcast requests so that all the clients on that serverless network can acquire network addresses. You do this by configuring the router to provide address resolution (ARP) functionality. When ARP is enabled, the router responds to address requests and assigns addresses to network clients. The router then acts as a network communication service provider for the client. The router generates 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 shows how to configure 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  
vines arp-enable  
vines serverless
```

Related Command

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 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	ARPA encapsulation. This is the default encapsulation for Ethernet interfaces.
snap	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	Our VINES Token Ring encapsulation. This is the default encapsulation for Token Ring interfaces.

Default

For Ethernet: **arpa**
For Token Ring: **vines-tr**
For all other media: **snap**

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 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

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

Default

Display hosts 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 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 [number]
no vines metric
```

Syntax Description

number (Optional.) Cost value associated with the interface. It is optional for all interface types. If you omit *metric*, the router automatically chooses a reasonable value. These values are listed in Table 1-13 in the “Usage Guidelines” section. 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 like 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, first issue the **show interface** command to determine the speed of the interface. Then 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 1-13 lists some example delay metric values.

Table 1-13 Example Delay Metric Values

Interface Type	Delay Metric Value
FDDI	1
Ethernet	2
16-Mb Token Ring	2
4-Mb Token Ring	4
T1 HDLC	35

Interface Type	Delay Metric Value
56-kb HDLC	45
9600 HDLC	90
4800 HDLC	150
2400 HDLC	250
1200 HDLC	450
T1 X.25	45
56-kb X.25	55
9600 X.25	100
4800 X.25	160
2400 X.25	260
1200 X.25	460

Example

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

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

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 [metric]
no vines neighbor address mac-address
```

Syntax Description

<i>address</i>	VINES IP address of the station to which to add or remove a static path.
<i>mac-address</i>	MAC-level address used to reach the neighbor station.
<i>encapsulation</i>	Encapsulation type to use on the media. It can be one of the following values: <ul style="list-style-type: none"> • 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.
<i>metric</i>	(Optional.) Delay metric to use on the neighbor. For more information, refer to the discussion and table in the “Usage Guidelines” section. If you omit this argument, the metric used is that specified with the vines metric command for the selected interface.

Default

No static paths

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 1-14 lists some example delay metric values.

Table 1-14 Example Delay Metric Values

Interface Type	Delay Metric Value
FDDI	1
Ethernet	2
16-Mb Token Ring	2
4-Mb Token Ring	4
T1 HDLC	35
56-kb HDLC	45
9600 HDLC	90
4800 HDLC	150
2400 HDLC	250
1200 HDLC	450
T1 X.25	45
56-kb X.25	55
9600 X.25	100
4800 X.25	160
2400 X.25	260
1200 X.25	460

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 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 **no** form of this command.

vines propagate
no vines propagate

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

The **vines propagate** command affects how the router decides whether to forward a broadcast packet on an interface. The normal decision is based on the settings of both the “hop count” and “class” fields of the VINES IP header. Enabling this command tells the router to ignore the “class” field and make the routing decision solely based on the “hop count” field.

A common use of this command would be for a serverless network that is connected via a serial line. Because of the way the VINES client software operates, file service requests are sent with a “class” field, which indicates that they should only be sent on LANs. Enabling this command allows these broadcast packets to cross the serial line and reach a network that contains a server.

Example

The following example ignores the “class” field of the VINES IP header when deciding whether to forward a broadcast packet:

```
interface serial 0
vines propagate
```

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 [*time*]
no vines redirect

Syntax Description

time (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 metric  
no vines route number address metric
```

Syntax Description

<i>number</i>	Number of the server to which to add or remove the static route.
<i>address</i>	VINES IP address of the neighbor station to use to reach the server.
<i>metric</i>	Metric value assigned to this route.

Default

No static routes

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.

Example

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

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

Related Commands

```
clear vines neighbor  
clear vines route  
show vines neighbor  
show vines route  
vines neighbor
```

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

Fast switching 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

<i>address</i>	(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

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 our 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 interface Ethernet 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 turn off this functionality, use the **no** form of this command.

vines serverless
no vines serverless

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

The **vines serverless** interface subcommand 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, you must configure it with the **vines serverless** command to 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 allows a router to find the services that are provided by a server on another network. This feature is especially important when two networks, one with a server and one without a server, are connected via one or more routers; in this situation, you must use this command to build a path between the two networks.

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.

Example

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
vines arp-enable
vines serverless
```

Related Command
vines arp-enable

vines split-horizon

To use split horizon when sending out 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 determines how much information is included in routing updates sent out an 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. This is because it is assumed that the physical network is fully connected and that all other stations on that network have already heard the information.

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.

Example

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

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

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

vines time participate

To control whether the router participates 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

Disabled

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 sets the router to participate in the sending of VINES time messages:

```
vines time participate
```

Related Commands

show vines services
vines access-list (simple)
vines time 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 services
vines access-list (simple)
vines time access-group
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 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 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 services
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 routes  
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 time  
no vines update interval time
```

Syntax Description

time 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 *time* 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 routes  
vines metric
```

