

Contents

IP performance optimization commands.....	1
display icmp statistics.....	1
display ip statistics.....	1
display rawip	3
display rawip verbose.....	4
display tcp	6
display tcp statistics	7
display tcp verbose.....	9
display udp	11
display udp statistics	12
display udp verbose.....	13
ip forward-broadcast	15
ip icmp error-interval	16
ip icmp fragment discarding.....	17
ip icmp source	17
ip mtu	18
ip redirects enable	19
ip ttl-expires enable.....	19
ip unreachable enable.....	20
reset ip statistics	21
reset tcp statistics.....	21
reset udp statistics.....	22
tcp mss	22
tcp path-mtu-discovery	23
tcp syn-cookie enable.....	23
tcp timer fin-timeout	24
tcp timer syn-timeout.....	25
tcp window.....	25

IP performance optimization commands

The term "interface" in this chapter refers to VLAN interfaces.

display icmp statistics

Use **display icmp statistics** to display ICMP statistics.

Syntax

```
display icmp statistics [ slot slot-number ]
```

Views

Any view

Predefined user roles

network-admin
network-operator

Parameters

slot slot-number. Specifies an IRF member device by its member ID.

Usage guidelines

ICMP statistics include information about received and sent ICMP packets.

Examples

Display ICMP statistics.

```
<Sysname> display icmp statistics
```

```
Input: bad formats 0 bad checksum 0
       echo 175 destination unreachable 0
       source quench 0 redirects 0
       echo replies 201 parameter problem 0
       timestamp 0 information requests 0
       mask requests 0 mask replies 0
       time exceeded 0 invalid type 0
       router advert 0 router solicit 0
       broadcast/multicast echo requests ignored 0
       broadcast/multicast timestamp requests ignored 0
Output: echo 0 destination unreachable 0
       source quench 0 redirects 0
       echo replies 175 parameter problem 0
       timestamp 0 information replies 0
       mask requests 0 mask replies 0
       time exceeded 0 bad address 0
       packet error 1442 router advert 3
```

display ip statistics

Use **display ip statistics** to display IP packet statistics.

Syntax

display ip statistics [*slot slot-number*]

Views

Any view

Predefined user roles

network-admin

network-operator

Parameters

slot *slot-number*. Specifies an IRF member device by its member ID.

Usage guidelines

IP statistics include information about received and sent packets and reassembly.

Examples

Display IP packet statistics.

```
<Sysname> display ip statistics
```

```
Input:  sum          7120          local          112
        bad protocol  0           bad format     0
        bad checksum  0           bad options    0
Output: forwarding    0           local          27
        dropped       0           no route       2
        compress fails 0
Fragment:input       0           output         0
        dropped       0
        fragmented    0           couldn't fragment 0
Reassembling:sum    0           timeouts       0
```

Table 1 Command output

Field		Description
Input	sum	Total number of packets received.
	local	Total number of packets destined for the device.
	bad protocol	Total number of unknown protocol packets.
	bad format	Total number of packets with incorrect format.
	bad checksum	Total number of packets with incorrect checksum.
	bad options	Total number of packets with incorrect option.
Output	forwarding	Total number of packets forwarded.
	local	Total number of packets locally sent.
	dropped	Total number of packets discarded.
	no route	Total number of packets for which no route is available.
	compress fails	Total number of packets failed to be compressed.
Fragment	input	Total number of fragments received.

Field		Description
	output	Total number of fragments sent.
	dropped	Total number of fragments dropped.
	fragmented	Total number of packets successfully fragmented.
	couldn't fragment	Total number of packets failed to be fragmented.
Reassembling	sum	Total number of packets reassembled.
	timeouts	Total number of reassembly timeouts.

Related commands

- **display ip interface**
- **reset ip statistics**

display rawip

Use **display rawip** to display brief information about RawIP connections.

Syntax

display rawip [slot *slot-number*]

Views

Any view

Predefined user roles

network-admin
network-operator

Parameters

slot *slot-number*. Specifies an IRF member device by its member ID.

Usage guidelines

Brief RawIP connection information includes local and peer addresses, protocol, and PCB.

Examples

Display brief information about RawIP connections.

```
<Sysname> display rawip
Local Addr      Foreign Addr    Protocol  Slot  PCB
0.0.0.0         0.0.0.0        1         1     0x0000000000000009
0.0.0.0         0.0.0.0        1         1     0x0000000000000008
0.0.0.0         0.0.0.0        1         1     0x0000000000000002
```

Table 2 Command output

Field	Description
Local Addr	Local IP address.
Foreign Addr	Peer IP address.
Protocol	Protocol number.
Slot	ID of the IRF member device.
PCB	Protocol control block.

display rawip verbose

Use **display rawip verbose** to display detailed information about RawIP connections.

Syntax

```
display rawip verbose [ slot slot-number [ pcb pcb-index ] ]
```

Views

Any view

Predefined user roles

network-admin

network-operator

Parameters

pcb *pcb-index*: Specifies a PCB by its index in the range of 1 to 16.

slot *slot-number*: Specifies an IRF member device by its member ID.

Usage guidelines

Use the **display rawip verbose** command to display detailed information about socket creator, state, option, type, protocol number, and the source and destination IP addresses of RawIP connections.

Examples

Display detailed information about RawIP connections.

```
<Sysname> display rawip verbose
```

```
Total RawIP socket number: 1
```

```
Slot: 6
```

```
Creator: ping[320]
```

```
State: N/A
```

```
Options: N/A
```

```
Error: 0
```

```
Receiving buffer(cc/hiwat/lowat/drop/state): 0 / 9216 / 1 / 0 / N/A
```

```
Sending buffer(cc/hiwat/lowat/drop/state): 0 / 9216 / 512 / 0 / N/A
```

```
Type: 3
```

```
Protocol: 1
```

```
Connection info: src = 0.0.0.0, dst = 0.0.0.0
```

```
Inpcb flags: N/A
```

```
Inpcb vflag: INP_IPV4
```

```
TTL: 255(minimum TTL: 0)
```

```
Send VRF: 0xffff
```

```
Receive VRF: 0xffff
```

Table 3 Command output

Field	Description
Total RawIP socket number	Total number of RawIP sockets.
Slot	ID of the IRF member device.
Creator	Name of the operation that created the socket. The number in brackets is the process number of the creator.

Field	Description
State	State of the socket.
Options	Socket options.
Error	Error code.
Receiving buffer (cc/hiwat/lowat/drop/state)	<p>Displays receive buffer information in the following order:</p> <ul style="list-style-type: none"> • cc—Used space. • hiwat—Maximum space. • lowat—Minimum space. • drop—Number of dropped packets. • state—Buffer state: <ul style="list-style-type: none"> ○ CANTSENDMORE—Unable to send data to the peer. ○ CANTRCVMORE—Unable to receive data from the peer. ○ RCVATMARK—Receiving tag. ○ N/A—None of the above states.
Receiving buffer (cc/hiwat/lowat/drop/state)	<p>Displays send buffer information in the following order:</p> <ul style="list-style-type: none"> • cc—Used space. • hiwat—Maximum space. • lowat—Minimum space. • drop—Number of dropped packets. • state—Buffer state: <ul style="list-style-type: none"> ○ CANTSENDMORE—Unable to send data to the peer. ○ CANTRCVMORE—Unable to receive data from the peer. ○ RCVATMARK—Receiving tag. ○ N/A—None of the above states.
Type	<p>Socket type:</p> <ul style="list-style-type: none"> • 1—SOCK_STREAM. This socket uses TCP to provide reliable transmission of byte streams. • 2—SOCK_DGRAM. This socket uses UDP to provide datagram transmission. • 3—SOCK_RAW. This socket allows an application to change the next upper-layer protocol header. • N/A—None of the above types.
Protocol	Number of the protocol using the socket.
Connection info	Source IP address and destination IP address.

Field	Description
Inpcb flags	<p>Flags in the Internet PCB:</p> <ul style="list-style-type: none"> • INP_RECVOPTS—Receives IP options. • INP_RECVRETOPTS—Receives replied IP options. • INP_RECVSTADDR—Receives destination IP address. • INP_HDRINCL—Provides the entire IP header. • INP_REUSEADDR—Reuses the IP address. • INP_REUSEPORT—Reuses the port number. • INP_ANONPORT—Port number not specified. • INP_RECVIF—Records the input interface of the packet. • INP_RECVTTL—Receives TTL of the packet. Only UDP and RawIP support this flag. • INP_DONTFRAG—Sets the Don't Fragment flag. • INP_ROUTER_ALERT—Receives packets with the router alert option. Only RawIP supports this flag. • INP_PROTOCOL_PACKET—Identifies a protocol packet. • INP_RCVLANID—Receives the VLAN ID of the packet. Only UDP and RawIP support this flag. • INP_RCVMACADDR—Receives the MAC address of the frame. • INP_RECVTOS—Receives TOS of the packet. Only UDP and RawIP support this flag. • INP_USEICMP_SRC—Uses the specified IP address as the source IP address for outgoing ICMP packets. • INP_SYNCPCB—Waits until Internet PCB is synchronized. • N/A—None of the above flags.
Inpcb vflag	<p>IP version flags in the Internet PCB:</p> <ul style="list-style-type: none"> • INP_IPV4—IPv4 protocol. • INP_TIMEWAIT—In TIMEWAIT state. • INP_ONESBCAST—Sends broadcast packets. • INP_DROPPED—Protocol dropped flag. • INP_SOCKREF—Strong socket reference. • INP_DONTBLOCK—Do not block synchronization of the Internet PCB. • N/A—None of the above flags.
TTL	TTL value in the Internet PCB.

display tcp

Use **display tcp** to display brief information about TCP connections.

Syntax

```
display tcp [ slot slot-number ]
```

Views

Any view

Predefined user roles

network-admin

network-operator

Parameters

slot *slot-number*. Specifies an IRF member device by its member ID.

Usage guidelines

Brief TCP connection information includes local IP address, local port number, peer IP address, peer port number, and TCP connection state.

Examples

Display brief information about TCP connections.

```
<Sysname> display tcp
*: TCP MD5 Connection
Local Addr:port      Foreign Addr:port    State      Slot PCB
*0.0.0.0:21          0.0.0.0:0            LISTEN     1     0x00000000000000c387
192.168.20.200:23    192.168.20.14:1284   ESTABLISHED 1     0x0000000000000009
192.168.20.200:23    192.168.20.14:1283   ESTABLISHED 1     0x0000000000000002
```

Table 4 Command output

Field	Description
*	Indicates the TCP connection uses MD5 authentication.
Local Addr:port	Local IP address and port number.
Foreign Addr:port	Peer IP address and port number.
State	TCP connection state.
Slot	ID of the IRF member device.
PCB	PCB index.

display tcp statistics

Use **display tcp statistics** to display TCP traffic statistics.

Syntax

```
display tcp statistics [ slot slot-number ]
```

Views

Any view

Predefined user roles

network-admin
network-operator

Parameters

slot *slot-number*. Specifies an IRF member device by its member ID.

Usage guidelines

TCP traffic statistics include information about received and sent TCP packets and Syncookie/syncookie.

Examples

Display TCP traffic statistics.

```
<Sysname> display tcp statistics
Received packets:
```


Total: 4150
packets in sequence: 1366 (134675 bytes)
window probe packets: 0, window update packets: 0
checksum error: 0, offset error: 0, short error: 0
packets dropped for lack of memory: 0
packets dropped due to PAWS: 0
duplicate packets: 12 (36 bytes), partially duplicate packets: 0 (0 bytes)
out-of-order packets: 0 (0 bytes)
packets with data after window: 0 (0 bytes)
packets after close: 0
ACK packets: 3531 (795048 bytes)
duplicate ACK packets: 33, ACK packets for unsent data: 0

Sent packets:

Total: 4058
urgent packets: 0
control packets: 50
window probe packets: 3, window update packets: 11
data packets: 3862 (795012 bytes), data packets retransmitted: 0 (0 bytes)
ACK-only packets: 150 (52 delayed)
unnecessary packet retransmissions: 0

Synccache/syncookie related statistics:

entries added to synccache: 12
synccache entries retransmitted: 0
duplicate SYN packets: 0
reply failures: 0
successfully build new socket: 12
bucket overflows: 0
zone failures: 0
synccache entries removed due to RST: 0
synccache entries removed due to timed out: 0
ACK checked by synccache or syncookie failures: 0
synccache entries aborted: 0
synccache entries removed due to bad ACK: 0
synccache entries removed due to ICMP unreachable: 0
SYN cookies sent: 0
SYN cookies received: 0

SACK related statistics:

SACK recoveries: 1
SACK retransmitted segments: 0 (0 bytes)
SACK blocks (options) received: 0
SACK blocks (options) sent: 0
SACK scoreboard overflows: 0

Other statistics:

retransmitted timeout: 0, connections dropped in retransmitted timeout: 0

```

persist timeout: 0
keepalive timeout: 21, keepalive probe: 0
keepalive timeout, so connections disconnected: 0
fin_wait_2 timeout, so connections disconnected: 0
initiated connections: 29, accepted connections: 12, established connections:
23
closed connections: 50051 (dropped: 0, initiated dropped: 0)
bad connection attempt: 0
ignored RSTs in the window: 0
listen queue overflows: 0
RTT updates: 3518(attempt segment: 3537)
correct ACK header predictions: 0
correct data packet header predictions: 568
resends due to MTU discovery: 0
packets dropped with MD5 authentication: 0
packets permitted with MD5 authentication: 0

```

Related commands

reset tcp statistics

display tcp verbose

Use **display tcp verbose** to display detailed information about TCP connections.

Syntax

```
display tcp verbose [ slot slot-number [ pcb pcb-index ] ]
```

Views

Any view

Predefined user roles

network-admin

network-operator

Parameters

pcb *pcb-index*: Specifies a PCB by its index in the range of 1 to 16.

slot *slot-number*: Specifies an IRF member device by its member ID.

Usage guidelines

Detailed TCP connection information includes socket creator, state, option, type, protocol number, source IP address and port number, destination IP address and port number, and connection state.

Examples

Display detailed information about TCP connections.

```

<Sysname> display tcp verbose
TCP inpcb number: 1(tcpcb number: 1)

Slot: 6
Creator: telnetd_mips[199]
State: ISCONNECTED
Options: N/A

```

```

Error: 0
Receiving buffer(cc/hiwat/lowat/state): 0 / 65700 / 1 / N/A
Sending buffer(cc/hiwat/lowat/state): 0 / 65700 / 512 / N/A
Type: 1
Protocol: 6
Connection info: src = 192.168.20.200:23 , dst = 192.168.20.14:4181
Inpcb flags: N/A
Inpcb vflag: INP_IPV4
TTL: 255(minimum TTL: 0)
Connection state: ESTABLISHED
Send VRF: 0x0
Receive VRF: 0x0

```

Table 5 Command output

Field	Description
TCP inpcb number	Number of TCP IP PCBs.
tcpcb number	Number of TCP PCBs.
Slot	ID of the IRF member device.
Creator	Name of the operation that created the socket. The number in brackets is the process number of the creator.
State	State of the socket.
Options	Socket options.
Error	Error code.
Receiving buffer (cc/hiwat/lowat/state)	<p>Displays receive buffer information in the following order:</p> <ul style="list-style-type: none"> • cc—Used space. • hiwat—Maximum space. • lowat—Minimum space. • state—Buffer state: <ul style="list-style-type: none"> ○ CANTSENDMORE—Unable to send data to the peer. ○ CANTRCVMORE—Unable to receive data from the peer. ○ RCVATMARK—Receiving tag. ○ N/A—None of the above states.
Sending buffer (cc/hiwat/lowat/state)	<p>Displays send buffer information in the following order:</p> <ul style="list-style-type: none"> • cc—Used space. • hiwat—Maximum space. • lowat—Minimum space. • state—Buffer state: <ul style="list-style-type: none"> ○ CANTSENDMORE—Unable to send data to the peer. ○ CANTRCVMORE—Unable to receive data from the peer. ○ RCVATMARK—Receiving tag. ○ N/A—None of the above states.

Field	Description
Type	<p>Socket type:</p> <ul style="list-style-type: none"> • 1—SOCK_STREAM. This socket uses TCP to provide reliable transmission of byte streams. • 2—SOCK_DGRAM. This socket uses UDP to provide datagram transmission. • 3—SOCK_RAW. This socket allows an application to change the next upper-layer protocol header. • N/A—None of the above types.
Protocol	Number of the protocol using the socket.
Connection info	Source IP address and destination IP address.
Inpcb flags	<p>Flags in the Internet PCB:</p> <ul style="list-style-type: none"> • INP_RECVOPTS—Receives IP options. • INP_RECVRETOPTS—Receives replied IP options. • INP_RECVDSTADDR—Receives destination IP address. • INP_HDRINCL—Provides the entire IP header. • INP_REUSEADDR—Reuses the IP address. • INP_REUSEPORT—Reuses the port number. • INP_ANONPORT—Port number not specified. • INP_RECVIF—Records the input interface of the packet. • INP_RECVTTL—Receives TTL of the packet. Only UDP and RawIP support this flag. • INP_DONTFRAG—Sets the Don't Fragment flag. • INP_ROUTER_ALERT—Receives packets with the router alert option. Only RawIP supports this flag. • INP_PROTOCOL_PACKET—Identifies a protocol packet. • INP_RCVLANID—Receives the VLAN ID of the packet. Only UDP and RawIP support this flag. • INP_RCVMACADDR—Receives the MAC address of the frame. • INP_RECVTOS—Receives TOS of the packet. Only UDP and RawIP support this flag. • INP_SYNCPCB—Waits until Internet PCB is synchronized. • N/A—None of the above flags.
Inpcb vflag	<p>IP version flags in the Internet PCB:</p> <ul style="list-style-type: none"> • INP_IPV4—IPv4 protocol. • INP_TIMEWAIT—In TIMEWAIT state. • INP_ONESBCAST—Sends broadcast packets. • INP_DROPPED—Protocol dropped flag. • INP_SOCKREF—Strong socket reference. • INP_DONTBLOCK—Do not block synchronization of the Internet PCB. • N/A—None of the above flags.
TTL	TTL value in the Internet PCB.

display udp

Use **display udp** to display brief information about UDP connections.

Syntax

display udp [slot *slot-number*]

Views

Any view

Predefined user roles

network-admin
network-operator

Parameters

slot *slot-number*. Specifies an IRF member device by its member ID.

Usage guidelines

Brief UDP connection information includes local IP address and port number, and peer IP address and port number.

Examples

Display brief information about UDP connections.

```
<Sysname> display udp
  Local Addr:port      Foreign Addr:port      Slot PCB
  0.0.0.0:69           0.0.0.0:0              1    0x0000000000000003
  192.168.20.200:1024  192.168.20.14:69      1    0x0000000000000002
```

Table 6 Command output

Field	Description
Local Addr:port	Local IP address and port number.
Foreign Addr:port	Peer IP address and port number.
Slot	ID of the IRF member device.
PCB	PCB index.

display udp statistics

Use **display udp statistics** to display UDP traffic statistics.

Syntax

display udp statistics [slot *slot-number*]

Views

Any view

Predefined user roles

network-admin
network-operator

Parameters

slot *slot-number*. Specifies an IRF member device by its member ID.

Usage guidelines

UDP traffic statistics include information about received and sent UDP packets.

Examples

```
# Display UDP traffic statistics.
<Sysname> display udp statistics
Received packets:
    Total: 240
    checksum error: 0, no checksum: 0
    shorter than header: 0, data length larger than packet: 0
    no socket on port(unicast): 0
    no socket on port(broadcast/multicast): 240
    not delivered, input socket full: 0
Sent packets:
    Total: 0
```

Related commands

reset udp statistics

display udp verbose

Use **display udp verbose** to display detailed information about UDP connections.

Syntax

```
display udp verbose [ slot slot-number [ pcb pcb-index ] ]
```

Views

Any view

Predefined user roles

network-admin
network-operator

Parameters

pcb *pcb-index*: Specifies a PCB by its index in the range of 1 to 16.
slot *slot-number*: Specifies an IRF member device by its member ID.

Usage guidelines

Detailed UDP connection information includes the socket creator, status, option, type, the protocol number, the source IP address and port number, and the destination IP address and port number for UDP connections.

Examples

```
# Display detailed UDP connection information.
<Sysname> display udp verbose
Total UDP socket number: 1

Slot: 6
Creator: sock_test_mips[250]
State: N/A
Options: N/A
Error: 0
Receiving buffer(cc/hiwat/lowat/drop/state): 0 / 41600 / 1 / 0 / N/A
Sending buffer(cc/hiwat/lowat/drop/state): 0 / 9216 / 512 / 0 / N/A
```

```

Type: 2
Protocol: 17
Connection info: src = 0.0.0.0:69, dst = 0.0.0.0:0
Inpcb flags: N/A
Inpcb vflag: INP_IPV4
TTL: 255(minimum TTL: 0)
Send VRF: 0xffff
Receive VRF: 0xffff

```

Table 7 Command output

Field	Description
Total UDP socket number	Total number of UDP sockets.
Slot	ID of the IRF member device.
Creator	Name of the operation that created the socket. The number in brackets is the process number of the creator.
State	Socket state.
Options	Socket option.
Error	Error code.
Receiving buffer(cc/hiwat/lowat/drop/state)	<p>Displays receive buffer information in the following order:</p> <ul style="list-style-type: none"> • cc—Used space. • hiwat—Maximum space. • lowat—Minimum space. • drop—Number of dropped packets. • state—Buffer state: <ul style="list-style-type: none"> ○ CANTSENDMORE—Unable to send data to the peer. ○ CANTRCVMORE—Unable to receive data from the peer. ○ RCVATMARK—Receiving tag. ○ N/A—None of the above states.
Sending buffer(cc/hiwat/lowat/drop/state)	<p>Displays send buffer information in the following order:</p> <ul style="list-style-type: none"> • cc—Used space. • hiwat—Maximum space. • lowat—Minimum space. • drop—Number of dropped packets. • state—Buffer state: <ul style="list-style-type: none"> ○ CANTSENDMORE—Unable to send data to the peer. ○ CANTRCVMORE—Unable to receive data from the peer. ○ RCVATMARK—Receiving tag. ○ N/A—None of the above states.
Type	<p>Socket type:</p> <ul style="list-style-type: none"> • 1—SOCK_STREAM. This socket uses TCP to provide reliable transmission of byte streams. • 2—SOCK_DGRAM. This socket uses UDP to provide datagram transmission. • 3—SOCK_RAW. This socket allows an application to change the next upper-layer protocol header. • N/A—None of the above types.
Protocol	Number of the protocol using the socket.

Field	Description
Inpcb flags	<p>Flags in the Internet PCB:</p> <ul style="list-style-type: none"> • INP_RECVOPTS—Receives IP options. • INP_RECVRETOPTS—Receives replied IP options. • INP_RECVDSTADDR—Receives destination IP address. • INP_HDRINCL—Provides the entire IP header. • INP_REUSEADDR—Reuses the IP address. • INP_REUSEPORT—Reuses the port number. • INP_ANONPORT—Port number not specified. • INP_RECVIF—Records the input interface of the packet. • INP_RECVTTL—Receives TTL of the packet. Only UDP and RawIP support this flag. • INP_DONTFRAG—Sets the Don't Fragment flag. • INP_ROUTER_ALERT—Receives packets with the router alert option. Only RawIP supports this flag. • INP_PROTOCOL_PACKET—Identifies a protocol packet. • INP_RCVVLANID—Receives the VLAN ID of the packet. Only UDP and RawIP support this flag. • INP_RCVMACADDR—Receives the MAC address of the frame. • INP_RECVTOS—Receives TOS of the packet. Only UDP and RawIP support this flag. • INP_SYNCPCB—Waits until Internet PCB is synchronized. • N/A—None of the above flags.
Inpcb vflag	<p>IP version flags in the Internet PCB:</p> <ul style="list-style-type: none"> • INP_IPV4—IPv4 protocol. • INP_TIMEWAIT—In TIMEWAIT state. • INP_ONESBCAST—Sends broadcast packets. • INP_DROPPED—Protocol dropped flag. • INP_SOCKREF—Strong socket reference. • INP_DONTBLOCK—Do not block synchronization of the Internet PCB. • N/A—None of the above flags.
TTL	TTL value in the Internet PCB.

ip forward-broadcast

Use **ip forward-broadcast** to enable an interface to receive and forward directed broadcast packets destined for the directly connected network.

Use **undo ip forward-broadcast** to disable an interface from receiving and forwarding directed broadcast packets destined for the directly connected network.

Syntax

ip forward-broadcast

undo ip forward-broadcast

Default

An interface cannot receive or forward directed broadcasts destined for the directly connected network.

Views

Interface view

Predefined user roles

network-admin

Usage guidelines

A directed broadcast packet is destined for all hosts on a specific network. In the destination IP address of the directed broadcast, the network ID identifies the target network, and the host ID is made up of all ones.

If an interface is allowed to forward directed broadcasts destined for the directly connected network, hackers can exploit this vulnerability to attack the target network. In some scenarios, however, an interface must receive and send such directed broadcast packets to support UDP helper.

This command enables an interface to accept directed broadcast packets that are destined for and received from the directly connected network to support UDP helper. UDP helper converts the directed broadcasts to unicasts and forwards them to a specific server.

The command also enables the interface to forward directed broadcast packets that are destined for the directly connected network and are received from another subnet to support Wake on LAN. Wake on LAN sends the directed broadcasts to wake up the hosts on the target network.

Examples

```
# Enable VLAN-interface 2 to receive and forward directed broadcast packets destined for the
directly connected network.
<Sysname> system-view
[Sysname] interface vlan-interface 2
[Sysname-Vlan-interface2] ip forward-broadcast
```

ip icmp error-interval

Use **ip icmp error-interval** to set the bucket size and the interval for tokens to arrive in the bucket for ICMP error messages.

Use **undo ip icmp error-interval** to restore the default.

Syntax

ip icmp error-interval *milliseconds* [*bucketsize*]

undo ip icmp error-interval

Default

The bucket allows a maximum of 10 tokens, and tokens are placed in the bucket at the interval of 100 milliseconds.

Views

System view

Predefined user roles

network-admin

Parameters

milliseconds: Specifies the interval for tokens to arrive in the bucket. The value range is 0 to 2147483647 milliseconds, and the default is 100 milliseconds. To disable the ICMP rate limit, set the value to 0.

bucketsize: Specifies the maximum number of tokens allowed in the bucket. The value range is 1 to 200, and the default is 10.

Usage guidelines

This command limits the rate at which ICMP error messages are sent. Use this command to avoid sending excessive ICMP error messages within a short period that might cause network congestion. A token bucket algorithm is used with one token representing one ICMP error message. Tokens are placed in the bucket at intervals until the maximum number of tokens that the bucket can hold is reached. Tokens are removed from the bucket when ICMP error messages are sent. When the bucket is empty, ICMP error messages are not sent until a new token is placed in the bucket.

Examples

```
# Configure an interval of 200 milliseconds and bucket size of 40 tokens for ICMP error messages.
<Sysname> system-view
[Sysname] ip icmp error-interval 200 40
```

ip icmp fragment discarding

Use **ip icmp fragment discarding** to disable forwarding of ICMP fragments.

Use **undo ip icmp fragment discarding** to enable forwarding of ICMP fragments.

Syntax

```
ip icmp fragment discarding
undo ip icmp fragment discarding
```

Default

Forwarding of ICMP fragments is enabled.

Views

System view

Predefined user roles

network-admin

Usage guidelines

Disable forwarding of ICMP fragments can prevent ICMP fragment attacks.

Examples

```
# Disable forwarding of ICMP fragments.
<Sysname> system-view
[Sysname] ip icmp fragment discarding
```

ip icmp source

Use **ip icmp source** to enable specifying the source address for outgoing ICMP packets.

Use **undo ip icmp source** to restore the default.

Syntax

```
ip icmp source [ vpn-instance vpn-instance-name ] ip-address
undo ip icmp source [ vpn-instance vpn-instance-name ]
```

Default

The device uses the IP address of the sending interface as the source IP address for outgoing ICMP packets.

Views

System view

Predefined user roles

network-admin

Parameters

vpn-instance *vpn-instance-name*: Specifies the VPN instance to which the specified address belongs. The *vpn-instance-name* argument is a case-sensitive string of 1 to 31 characters. The specified VPN instance must exist. If the specified IP address is on the public network, do not use this option.

ip-address: Specifies an IP address.

Usage guidelines

It is a good practice to specify the IP address of the loopback interface as the source IP address for outgoing ping echo request and ICMP error messages. This feature helps users to locate the sending device easily.

Examples

```
# Specify 1.1.1.1 as the source address for outgoing ICMP packets.
```

```
<Sysname> system-view
```

```
[Sysname] ip icmp source 1.1.1.1
```

ip mtu

Use **ip mtu** to configure an MTU for an interface.

Use **undo ip mtu** to restore the default.

Syntax

ip mtu *mtu-size*

undo ip mtu

Default

No MTU is configured for an interface.

Views

Interface view

Predefined user roles

network-admin

Parameters

mtu-size: Specifies an MTU in the range of 128 to 2000 bytes.

Usage guidelines

When a packet exceeds the MTU of the output interface, the device processes it in one of the following ways:

- If the packet disallows fragmentation, the device discards it.
- If the packet allows fragmentation, the device fragments it and forwards the fragments.

Fragmentation and reassembling consume system resources, so set an appropriate MTU for an interface to avoid fragmentation.

If an interface supports both the **mtu** and **ip mtu** commands, the device fragments a packet based on the MTU set by the **ip mtu** command.

The MTU configured for an interface takes effect on only packets that are sent to the CPU for software forwarding, including packets sent from or destined for this interface.

Examples

```
# Set the MTU of VLAN interface 100 to 1280 bytes.
<Sysname> system-view
[Sysname] interface vlan-interface 100
[Sysname-Vlan-interface100] ip mtu 1280
```

ip redirects enable

Use **ip redirects enable** to enable sending ICMP redirect messages.

Use **undo ip redirects enable** to disable sending ICMP redirect messages.

Syntax

```
ip redirects enable
undo ip redirects enable
```

Default

Sending ICMP redirect messages is disabled.

Views

System view

Predefined user roles

network-admin

Usage guidelines

ICMP redirect messages simplify host management and enable hosts to gradually optimize its routing table.

A host that has only one route destined to the default gateway sends all packets to the default gateway. The default gateway sends an ICMP redirect message to inform the host of a correct next hop by following these rules:

- The receiving and sending interfaces are the same.
- The selected route is not created or modified by any ICMP redirect message.
- The selected route is not destined for 0.0.0.0.
- There is no source route option in the received packet.

Examples

```
# Enable sending ICMP redirect messages.
<Sysname> system-view
[Sysname] ip redirects enable
```

ip ttl-expires enable

Use **ip ttl-expires enable** to enable sending ICMP time-exceeded messages.

Use **undo ip ttl-expires enable** to disable sending ICMP time-exceeded messages.

Syntax

```
ip ttl-expires enable
undo ip ttl-expires enable
```

Default

Sending ICMP time-exceeded messages is disabled.

Views

System view

Predefined user roles

network-admin

Usage guidelines

A device sends ICMP time-exceeded messages by following these rules:

- If a received packet is not destined for the device and the TTL field of the packet is 1, the device sends an ICMP TTL Expired in Transit message to the source.
- When the device receives the first fragment of an IP datagram destined for the device itself, it starts a timer. If the timer expires before all the fragments of the datagram are received, the device sends an ICMP Fragment Reassembly Timeout message to the source.

A device disabled from sending ICMP time-exceeded messages does not send ICMP TTL Expired in Transit messages but can still send ICMP Fragment Reassembly Timeout messages.

Examples

```
# Enable sending ICMP time-exceeded messages.
```

```
<Sysname> system-view
```

```
[Sysname] ip ttl-expires enable
```

ip unreachable enable

Use **ip unreachable enable** to enable sending ICMP destination unreachable messages.

Use **undo ip unreachable enable** to disable sending ICMP destination unreachable messages.

Syntax

```
ip unreachable enable
```

```
undo ip unreachable enable
```

Default

Sending ICMP destination unreachable messages is disabled.

Views

System view

Predefined user roles

network-admin

Usage guidelines

A device sends ICMP destination unreachable messages by following these rules:

- If a packet does not match any route and there is no default route in the routing table, the device sends a Network Unreachable ICMP error message to the source.
- If a packet is destined for the device but the transport layer protocol of the packet is not supported by the device, the device sends a Protocol Unreachable ICMP error message to the source.
- If a UDP packet is destined for the device but the packet's port number does not match the running process, the device sends the source a Port Unreachable ICMP error message.

- If the source uses Strict Source Routing to send packets, but the intermediate device finds that the next hop specified by the source is not directly connected, the device sends the source a Source Routing Failure ICMP error message.
- If the MTU of the sending interface is smaller than the packet and the packet has a Don't Fragment set, the device sends the source a Fragmentation Needed and Don't Fragment-Set ICMP error message.

Examples

```
# Enable sending ICMP destination unreachable messages.
<Sysname> system-view
[Sysname] ip unreachable enable
```

reset ip statistics

Use **reset ip statistics** to clear IP traffic statistics.

Syntax

```
reset ip statistics [ slot slot-number ]
```

Views

User view

Predefined user roles

network-admin

Parameters

slot *slot-number*: Specifies an IRF member device by its member ID.

Usage guidelines

To collect new IP traffic statistics within a period of time, use this command to clear history IP traffic statistics first.

Examples

```
# Clear IP traffic statistics.
<Sysname> reset ip statistics
```

Related commands

- **display ip interface**
- **display ip statistics**

reset tcp statistics

Use **reset tcp statistics** to clear TCP traffic statistics.

Syntax

```
reset tcp statistics
```

Views

User view

Predefined user roles

network-admin

Examples

```
# Clear TCP traffic statistics.
```

```
<Sysname> reset tcp statistics
```

Related commands

display tcp statistics

reset udp statistics

Use **reset udp statistics** to clear UDP traffic statistics.

Syntax

reset udp statistics

Views

User view

Predefined user roles

network-admin

Examples

```
# Clear UDP traffic statistics.
```

```
<Sysname> reset udp statistics
```

Related commands

display udp statistics

tcp mss

Use **tcp mss** to configure the TCP maximum segment size (MSS).

Use **undo tcp mss** to restore the default.

Syntax

tcp mss *value*

undo tcp mss

Default

No TCP MSS is configured.

Views

Interface view

Predefined user roles

network-admin

Parameters

value: Specifies the TCP MSS in bytes. The minimum value for the *value* argument is 128 bytes. The maximum value equals the maximum MTU that the interface supports minus 40.

Usage guidelines

This configuration takes effect only on TCP connections that are established after the configuration and not on the TCP connections that already exist.

This configuration is effective only on IP packets.

The MSS option informs the receiver of the largest segment that the sender can accept. Each end announces its MSS during TCP connection establishment.

If the size of a TCP segment is smaller than the MSS of the receiver, TCP sends the TCP segment without fragmentation. If not, it fragments the segment according to the receiver's MSS.

If you configure a TCP MSS on an interface, the size of each TCP segment received or sent on the interface cannot exceed the MSS value.

Examples

```
# Set the TCP MSS to 300 bytes on VLAN-interface 100.
<Sysname> system-view
[Sysname] interface vlan-interface 100
[Sysname-Vlan-interface100] tcp mss 300
```

tcp path-mtu-discovery

Use **tcp path-mtu-discovery** to enable TCP path MTU discovery.

Use **undo tcp path-mtu-discovery** to disable TCP path MTU discovery.

Syntax

```
tcp path-mtu-discovery [ aging age-time | no-aging ]
undo tcp path-mtu-discovery
```

Default

TCP path MTU discovery is disabled.

Views

System view

Predefined user roles

network-admin

Parameters

aging *age-time*: Specifies the aging time for the path MTU, in the range of 10 to 30 minutes. The default aging time is 10 minutes.

no-aging: Does not age out the path MTU.

Usage guidelines

After you enable TCP path MTU discovery, all new TCP connections detect the path MTU. The device uses the path MTU to calculate the MSS to avoid IP fragmentation.

After you disable TCP path MTU discovery, the system stops all path MTU timers. The TCP connections established later do not detect the path MTU, but the TCP connections previously established still can detect the path MTU.

Examples

```
# Enable TCP path MTU discovery and set the path MTU aging time to 20 minutes.
<Sysname> system-view
[Sysname] tcp path-mtu-discovery aging 20
```

tcp syn-cookie enable

Use **tcp syn-cookie enable** to enable SYN Cookie to protect the device from SYN flood attacks.

Use **undo tcp syn-cookie enable** to disable SYN Cookie.

Syntax

```
tcp syn-cookie enable
undo tcp syn-cookie enable
```

Default

SYN Cookie is disabled.

Views

System view

Predefined user roles

network-admin

Usage guidelines

A TCP connection is established through a three-way handshake:

1. The sender sends a SYN packet to the server.
2. The server receives the SYN packet, establishes a TCP semi-connection in SYN_RECEIVED state, and replies with a SYN ACK packet to the sender.
3. The sender receives the SYN ACK packet and replies with an ACK packet. Then, a TCP connection is established.

An attacker can exploit this mechanism to mount SYN flood attacks. The attacker sends a large number of SYN packets, but they do not respond to the SYN ACK packets from the server. As a result, the server establishes a large number of TCP semi-connections and cannot handle normal services.

SYN Cookie can protect the server from SYN flood attacks. When the server receives a SYN packet, it responds to the request with a SYN ACK packet without establishing a TCP semi-connection.

The server establishes a TCP connection and enters ESTABLISHED state only when it receives an ACK packet from the sender.

Examples

```
# Enable SYN Cookie.
<Sysname> system-view
[Sysname] tcp syn-cookie enable
```

tcp timer fin-timeout

Use **tcp timer fin-timeout** to configure the TCP FIN wait timer.

Use **undo tcp timer fin-timeout** to restore the default.

Syntax

```
tcp timer fin-timeout time-value
undo tcp timer fin-timeout
```

Default

The TCP FIN wait timer is 675 seconds.

Views

System view

Predefined user roles

network-admin

Parameters

time-value: Specifies the TCP FIN wait timer in the range of 76 to 3600 seconds.

Usage guidelines

TCP starts the FIN wait timer when the state changes to FIN_WAIT_2. If no FIN packet is received within the timer interval, the TCP connection is terminated.

If a FIN packet is received, TCP changes connection state to TIME_WAIT. If a non-FIN packet is received, TCP restarts the timer and tears down the connection when the timer expires.

Examples

```
# Set the TCP FIN wait timer to 800 seconds.
<Sysname> system-view
[Sysname] tcp timer fin-timeout 800
```

tcp timer syn-timeout

Use **tcp timer syn-timeout** to configure the TCP SYN wait timer.

Use **undo tcp timer syn-timeout** to restore the default.

Syntax

```
tcp timer syn-timeout time-value
undo tcp timer syn-timeout
```

Default

The TCP SYN wait timer is 75 seconds.

Views

System view

Predefined user roles

network-admin

Parameters

time-value: Specifies the TCP SYN wait timer in the range of 2 to 600 seconds.

Usage guidelines

TCP starts the SYN wait timer after sending a SYN packet. If no response packet is received within the SYN wait timer interval, TCP fails to establish the connection.

Examples

```
# Set the TCP SYN wait timer to 80 seconds.
<Sysname> system-view
[Sysname] tcp timer syn-timeout 80
```

tcp window

Use **tcp window** to configure the size of the TCP receive/send buffer.

Use **undo tcp window** to restore the default.

Syntax

```
tcp window window-size
undo tcp window
```

Default

The size of the TCP receive/send buffer is 63 KB.

Views

System view

Predefined user roles

network-admin

Parameters

window-size: Specifies the size of the TCP receive/send buffer in KB, in the range of 1 to 64.

Examples

Configure the size of the TCP receive/send buffer as 3 KB.

```
<Sysname> system-view
```

```
[Sysname] tcp window 3
```