Network Protocol Configuration Commands

Table of Contents

Chapter 1 IP Address Configuration Commands

1.1 IP Address Configuration Commands

IP address configuration commands include：

- arp
- arp timeout
- clear arp-cache
- ip address
- ip directed-broadcast
- ip forward-protocol
- ip helper-address
- ip host
- ip default-gateway
- ip proxy-arp
- show arp
- show hosts
- show ip interface

1.1.1 arp

To add a static and permanent entry in the Address Resolution Protocol (ARP) cache, use the **arp** command in global configuration mode. To remove an entry from the ARP cache, use the no form of this command.

arp *ip-address hardware-address* [**alias**]

no arp *ip-address*

parameter

parameter	description
ip-address	IP address corresponding to the local data-link address.
hardware-address	Physical address of local data-link address
alias	(optional) router responds to ARP requests as if it were the interface of the specified address.

default

No entries are permanently installed in the ARP cache.

command mode

global configuration mode

instruction

The common host all supports dynamic ARP analysis, so user doesn't need to configure static ARP entries for host.

Example

The following is an example of a static ARP entry for a typical Ethernet host:

arp 1.1.1.1 00:12:34:56:78:90

related commands

clear arp-cache

1.1.2 arp retry-allarp

To set whether to carry on redetection at the aging of ARP entries (not just meaning the gateway-related ARP entries), run the following command:

arp retry-allarp

Parameter

None

Command mode

Global configuration mode

Instruction

By default, redetection is conducted only to the aging ARPs, which the routing entry gateway depends on. However, if this command is enabled, redetection will be conducted towards all types of aging ARP entries.

Example

The following example shows how to enable redetection to be carried out to all aging ARP entries.

arp retry-allarp

Related command

show arp

1.1.3 arp timeout

To configure the exist time that a dynamic ARP entry remains in the Address Resolution Protocol (ARP) cache, use the **arp timeout**. To restore the default value, use the no form of this command or default arp timeout command.

arp timeout *seconds*

no arp timeout

default arp timeout

parameter

parameter	description
seconds	Time in seconds that an entry remains in the ARP cache. A value of zero means that entries are never cleared from the cache.

default

14400 seconds (4 hours)

mode

interface configuration mode

instruction

This command is ignored when it is not configured on interfaces using ARP. The show interface command displays the ARP timeout value, as seen in the following example from the show interfaces command:

ARP type: ARPA, ARP timeout 04:00:00

example

The following example sets the ARP timeout to 900 seconds on Ethernet 1/0 to allow entries to time out more quickly than the default

interface vlan 10
arp timeout 900

related commands

show interface

1.1.4 arp send-gratuitous

To configure ARP send-gratuitous function, use the arp send-gratuitous command

arp send-gratuitous [**interval** *value*]

parameter

parameter	descruption
interval	Set the intervals of arp send-gratuitous
value	Set time interval, the default is 120 seconds. The range is 15 to 600 seconds

mode

routing interface configuration mode

instruction

The following command start arp send-gratuitous on Interface Vlan 1, and set the send interval as 3 minutes

switch_config_v1#arp send-gratuitous interval 180

related commands

arp

1.1.5 clear arp-cache

To clear all dynamic entries from the ARP cache, use the clear arp-cache **command.**

clear arp-cache [*ip-address* [*mask*]]

parameter

parameter	description
ip-address	IP or subnets
mask	Subnets mask

mode

EXEC

example

The following example removes all dynamic entries from the ARP cache:

clear arp-cache

related commands

arp

1.1.6 ip address

To set an IP address and mask for an interface, use the **ip address** command. Currently, there is no strict regulation to distinguish A.B.C IP address. But multicast address and broadcast address can not be used(all host section is ‘1’). Other than the Ethernet,multiple interfaces of other types can be connected to the same network. Other than the unnumbered interface, the configured network range ot the Ethernet interface can not be the same as the arbitrary interfaces of other types. You should configure the primary address before configuring the secondary address. Also you should delete all secondary addresses before deleting the primary address. IP packets generanted by the system, if the upper application does not specify the soruce address, the router will use the IP address configured on the sending interface that on the same network range with the gateway as the source address of the packet. If the IP address is uncertain (like interface route), the router will use the primary address of the sending interface. If the ip address is not configured on an interface, also it is not the unnumbered interface, and then this interface will not deal with any IP packet.To remove an IP address or disable IP processing, use the no form of this command.

ip address *ip-address mask* [secondary]

no ip address *ip-address mask*

no ip address

parameter

parameter	description
ip-address	IP address
mask	IP mask
secondary	(optional) Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.

default

No IP address is defined for the interface.

command mode

interface configuration mode

instruction

If any router on a network segment uses a secondary address, all other devices on that same segment must also use a secondary address from the same network or subnet. Inconsistent use of secondary addresses on a network segment can very quickly cause routing loops. When you are routing using the Open Shortest Path First (OSPF) algorithm, ensure that all secondary addresses of an interface fall into the same OSPF area as the primary addresses

example

In the following example, 202.0.0.1 is the primary address, 255.255.255.0 is the mask and 203.0.0.1 and 204.0.0.1 are secondary addresses for Ethernet interface 1/0:

```
interface vlan 10
ip address 202.0.0.1 255.255.255.0
ip address 203.0.0.1 255.255.255.0 secondary
ip address 204.0.0.1 255.255.255.0 secondary
```

1.1.7 ip host

To define a static host name-to-address mapping in the host cache, use the **ip host** command in global configuration mode. To remove the host name-to-address mapping, use the no form of this command.

ip host *name address*

no ip host *name*

parameter

parameter	description
name	Host name
Address	IP address

default

disabled

command mode

global configuration mode

example

The following example shows how to configure host name dns-server to IP host address 202.96.1.3:

ip host dns-server 202.96.1.3

1.1.8 ip default-gateway

TO configure the default gateway of switch, use the ip default-gateway command. To delete the default gateway of switch, use the no form of this command.

ip default-gateway *address*

no ip default-gateway

parameter

parameter	description
address	IP address

default

no configuration

mode

global configuration mode

example

The following example configure the IP address 202.96.1.3 as default-gateway

ip default-gateway 202.96.1.3

1.1.9 show arp

To display the entries in the Address Resolution Protocol (ARP) table, including the ARP mapping of interface IP address, the static ARP mapping that user configures and the dynamic ARP mapping, use the **show arp** command.

show arp

parameter

this command has no parameters or keywords

mode

EXEC

instruction

The display includes:

parameter	description
Protocol	Displays the type of the network address that maps with the physical address. IP, for example.
Address	Displays the network address that maps with the physical address. IP address, for example.
Age	Displays the age in seconds. The router will refresh the time to 0 when using this ARP entry.
Hardware Address	Displays the physical address that corresponds to the network address. It is empty for the unanalyzed entries.
Type	Specifies request encapsulation types that the interface use, including ARPA, SNAP and so on.

example

The following command displays ARP cache.

```
switch#show arp
Protocol   IP Address        Age(min)   Hardware Address     Type    Interface
  IP       192.168.20.77        11     00:30:80:d5:37:e0    ARPA    vlan 10
  IP       192.168.20.33         0        Incomplete
  IP       192.168.20.22         -     08:00:3e:33:33:8a    ARPA    vlan 10
  IP       192.168.20.124        0     00:a0:24:9e:53:36    ARPA    vlan 10
  IP       192.168.0.22          -     08:00:3e:33:33:8b    ARPA    vlan 11
```

1.1.10 show hosts

To display all entries of the host name—address cathe, use the **show hosts** command.

show hosts

parameter

This command has no parameters or keywords.

command mode

EXEC

example

The following command shows how to display all host names/address mappings.

show hosts

related commands

clear host

1.1.11 show ip interface

To display the IP configuration on interface, use the **show ip interface** command

show ip interface [**type** *number*]

parameter

parameter	description
type	(Optional) Interface type.
number	(Optional) Interface number.

command mode

EXEC

instruction

If the interface link layer is usable, the line protocol is marked "Protocol up." If you configure IP address on this interface, the router will add a direct route to the routing table. If the link layer protocol is marked “Protocol down”, the direct route will be deleted. This command displays the specified interface information if specified interface type and number, or IP configuration information of all interfaces will be displayed.

Example

The following example shows how to display IP configuration on interface e0/1.

```
switch#show ip interface vlan 11
    vlan 10 is up, line protocol is up
IP address : 192.168.20.167/24
       Broadcast address : 192.168.20.255
       Helper address : not set
       MTU : 1500(byte)
       Forward Directed broadcast : OFF
       Multicast reserved groups joined:
         224.0.0.9 224.0.0.6 224.0.0.5 224.0.0.2
         224.0.0.1
```

Outgoing ACL : not set
Incoming ACL : not set
IP fast switching : ON
IP fast switching on the same interface : OFF
ICMP unreachables : ON
ICMP mask replies : OFF
ICMP redirects : ON

display description :

domain	description
Ethernet1/0 is up	If the interface hardware is usable, the interface is marked "up." For an interface to be usable, both the interface hardware and line protocol must be up.
line protocol is up	If the interface can provide two-way communication, the line protocol is marked "up." For an interface to be usable, both the interface hardware and line protocol must be up.
IP address	IP address and mask for interface
Broadcast address	Displays broadcast address
MTU	Displays the MTU value set on the interface.

Chapter 2 IP Service Configuration Commands

2.1 IP Service Configuration Commands

The following are IP service configuration commands:

- clear tcp
- clear tcp statistics
- debug arp
- debug ip icmp
- debug ip packet
- debug ip raw
- debug ip tcp packet
- debug ip tcp transactions
- debug ip udp
- ip mask-reply
- ip mtu
- ip redirects
- ip route-cache
- ip source-route
- ip tcp synwait-time
- ip tcp window-size
- ip unreachables
- show ip cache
- show ip irdp
- show ip sockets
- show ip traffic
- show tcp
- show tcp brief
- show tcp statistics
- show tcp tcb

2.1.1 clear tcp

It is used to delete a TCP connection.

clear tcp {**local** *host-name* **port remote** *host-name* **port** | *tcb address*}

Parameter

Parameter	Description

local host-name port	IP address and TCP port of the local host
remote host-name port	IP address and TCP port of the remote host
tcb address	TCB address of the to-be-deleted TCP connection TCB is an identifier of TCP connection in the inner system, which can be obtained by the command **show tcp brief**.

Command mode

Management mode

Instruction

The **clear tcp** command is mainly used to delete the terminated TCP connection. In some cases, such as faulty in communication lines, restarting TCP connection or the peer host, the TCP connections are terminated in fact. However, the system cannot obtain information about the terminated TCP connection because there is no communication on the TCP connections. In this case, you can run the **clear tcp** command to terminate these invalid TCP connections. The command **clear tcp local host-name port remote host-name port** is used to terminate the connections between the specified host's IP address/port and the remote host's IP address/port. The command **clear tcp tcb address** is used to terminate the TCP connections identified by the TCB address.

Example

The following example shows that the TCP connection between 192.168.20.22:23 and 192.168.20.120:4420 is deleted. The **show tcp brief** command is used to show the information about the local host and the remote host in TCP connection.

```
switch#show tcp brief
TCB          Local Address          Foreign Address        State
0xE85AC8     192.168.20.22:23       192.168.20.120:4420    ESTABLISHED
0xEA38C8     192.168.20.22:23       192.168.20.125:1583    ESTABLISHED
switch#clear tcp local 192.168.20.22 23 remote 192.168.20.120 4420
switch#show tcp brief
TCB          Local Address          Foreign Address        State
0xEA38C8     192.168.20.22:23       192.168.20.125:1583    ESTABLISHED
```

In the following example, the TCP connection whose TCB address is **0xea38c8** is deleted. The command **show tcp brief** displays the TCB address of the TCP connection.

```
switch#show tcp brief
TCB          Local Address          Foreign Address        State
0xEA38C8     192.168.20.22:23       192.168.20.125:1583    ESTABLISHED
switch#clear tcp tcb 0xea38c8
switch#show tcp brief
TCB          Local Address          Foreign Address        State
```

Related command

show tcp

show tcp brief

show tcp tcb

2.1.2 clear tcp statistics

It is used to clear the TCP statistics data.

clear tcp statistics

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

The following command is used to delete the TCP statistics data:

switch#clear tcp statistics

Related command

show tcp statistics

2.1.3 debug arp

It is used to display the ARP interaction information, such as sending ARP requests, receiving ARP requests, sending ARP response and receiving ARP response. When the switch cannot communicate with the host, the command is used to analyze the ARP interaction. You can run the **no debug arp** command to stop displaying the relative information.

debug arp

no debug arp

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

switch#debug arp

switch#IP ARP: rcvd req src 192.168.20.116 00:90:27:a7:a9:c2, dst 192.168.20.111, vlan 10

IP ARP: req filtered src 192.168.20.139 00:90:27:d5:a9:1f, dst 192.168.20.82 00:

00:00:00:00:00, wrong cable, vlan 11

IP ARP: created an incomplete entry for IP address 192.168.20.77, vlan 10

IP ARP: sent req src 192.168.20.22 08:00:3e:33:33:8a, dst 192.168.20.77, vlan 10

IP ARP: rcvd reply src 192.168.20.77 00:30:80:d5:37:e0, dst 192.168.20.22, vlan 10

The first information indicates: the switch receives an ARP request on interface vlan 10; the IP address of the host that sends the ARP request is 192.168.20.116 and the MAC address of the host is 00:90:27:a7:a9:c2; the MAC address of the host 192.168.20.111 is **IP ARP: rcvd req src 192.168.20.116 00:90:27:a7:a9:c2, dst 192.168.20.111, vlan 10**.

The second information indicates that the switch receives an ARP request from 192.168.20.139 host on interface vlan 10. However, the interface is not in the network the host declares according to the interface configuration on the switch. The host may not be correctly configured. If the switch creates the ARP cache according to the information, it may not communicate with the host that is configured the same address and connected to the normal interface

IP ARP: req filtered src 192.168.20.139 00:90:27:d5:a9:1f, dst 192.168.20.82 00:
00:00:00:00:00, wrong cable, vlan 11

In the third information, to resolve the MAC address of host 192.168.20.77, the switch first creates an incomplete ARP item in the ARP cache. After receiving an ARP response, the MAC address is then added to the ARP cache. According to the location of the switch, the host connects the interface vlan 10.

IP ARP: created an incomplete entry for IP address 192.168.20.77, vlan 10

In the fourth information, the switch sends out the ARP request from the interface vlan 10. The IP address of the switch is 192.168.20.22. The MAC address of the interface is 08:00:3e:33:33:8a. The IP address of the requested host is 192.168.20.77. The fourth information is relative with the third information.

IP ARP: sent req src 192.168.20.22 08:00:3e:33:33:8a, dst 192.168.20.77, vlan 10

In the fifth information, the switch receives the ARP response on interface vlan 10 from host 192.168.20.77 to host 192.168.20.22. The switch is then informed that the MAC address of the host that returns the ARP response is 00:30:80:d5:37:e0. The information is relative to the third and fourth information.

IP ARP: rcvd reply src 192.168.20.77 00:30:80:d5:37:e0, dst 192.168.20.22, vlan 10

2.1.4 debug ip icmp

It is used to display the ICMP interaction information. You can run the command **no debug ip icmp** to close the debugging output.

debug ip icmp

no debug ip icmp

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Instruction

The command is used to display the received or transmitted ICMP message, which helps to solve end-to-end connection problems. To know the detailed meaning of the command debug ip icmp, refer to RFC 792, “Internet Control Message Protocol”.

Example

switch#debug ip icmp
switch#ICMP: sent pointer indicating to 192.168.20.124 (dst was 192.168.20.22), len 48
ICMP: rcvd echo from 192.168.20.125, len 40
ICMP: sent echo reply, src 192.168.20.22, dst 192.168.20.125, len 40
ICMP: sent dst (202.96.209.133) host unreachable to 192.168.20.124, len 36

ICMP: sent dst (192.168.20.22) protocol unreachable to 192.168.20.124, len 36
ICMP: rcvd host redirect from 192.168.20.77, for dst 22.0.0.3 use gw 192.168.20.26, len 36
ICMP: rcvd dst (22.0.0.3) host unreachable from 192.168.20.26, len 36
ICMP: sent host redirect to 192.168.20.124, for dst 22.0.0.5 use gw 192.168.20.77, len 36
ICMP: rcvd dst (2.2.2.2) host unreachable from 192.168.20.26, len 36

Details about the first information are shown in the following table:

ICMP: sent pointer indicating to 192.168.20.124 (dst was 192.168.20.22), len 48

Field	Description
ICMP	Information about the ICMP message
Sent	Sending the ICMP message
pointer indicating	ICMP message which means that the original parameters of the IP message are incorrect and incorrect domain is pointed out The following are other types of ICMP message: echo reply dst unreachable: ---net unreachable ---host unreachable ---protocol unreachable ---port unreachable ---fragmentation needed and DF set ---source route failed ---net unknown ---destination host unknown ---source host isolated ---net prohibited ---host prohibited ---net tos unreachable ---host tos unreachable source quench redirect messages: ---net redirect ---host redirect ---net tos redirect ---host tos redirect echo router advertisement router solicitation time exceeded : ---ttl exceeded ---reassembly timeout

	parameter problem : ---pointer indicating ---option missed ---bad length timestamp timestamp reply information request information reply mask request mask reply If the ICMP type is unknown, the system is to display the values of the ICMP type and code.
to 192.168.20.124	Destination address of the ICMP message, which is also the source address of the original message that generates the ICMP message
(dst was 192.168.20.22)	Destination address of the original message that generates the ICMP message
len 48	Length of the ICMP message, excluding the length of the IP header

Details about the second information are shown in the following table:

ICMP: rcvd echo from 192.168.20.125, len 40

Field	Description
rcvd	Receiving the ICMP message
echo	Echo request message, which is a type of the ICMP message
from 192.168.20.125	Source address of the ICMP message

Details about the third information are shown in the following table:

ICMP: sent echo reply, src 192.168.20.22, dst 192.168.20.125, len 40

Field	Description
src 192.168.20.22	Means that the source address of the ICMP message is 192.168.20.22.
dst 192.168.20.125	Means that the destination address of the ICMP message is 192.168.20.125.

According to the type of the ICMP message, the information that generates the ICMP message adopts different formats to display the message content.

For example, the **redirect** message of ICMP is printed in the following format:

ICMP: rcvd host redirect from 192.168.20.77, for dst 22.0.0.3 use gw 192.168.20.26, len 36
ICMP: sent host redirect to 192.168.20.124, for dst 22.0.0.5 use gw 192.168.20.77, len 36

In the first information, an ICMP redirect message from host 192.168.20.77 is received. Gateway 192.168.20.26 is recommended to reach the destination host 22.0.0.3. The length of the ICMP message is 36 bytes.

In the second information, the ICMP redirect message is sent to from host 192.168.20.124 to host 22.0.0.5 through gateway 192.168.20.77. The length of the ICMP message is 36 bytes.

The **dst unreachable** message of ICMP adopts the following format for printing:

ICMP: sent dst (202.96.209.133) host unreachable to 192.168.20.124, len 36
ICMP: rcvd dst (2.2.2.2) host unreachable from 192.168.20.26, len 36

In the first information, the switch cannot route a certain IP message, so it sends the **destination** (202.96.209.133) **unreachable** message to the source host (192.168.20.124). The length of the ICMP message is 36 bytes.

In the second information, after receiving an ICMP message from host192.168.20.26, the switch notifies host 192.168.20.26 that the destination address (2.2.2.2) cannot be reached. The length of the ICMP message is 36 bytes.

2.1.5 debug ip packet

It is used to display the IP interaction information. The command **no debug ip packet** is used to stop displaying information.

debug ip packet [**detail**] *[ip-access-list-name*]

no debug ip packet

Parameter

Parameter	Description
detail	An optional parameter, which is used to export the protocol information about IP message encapsulation, such as protocol number, UDP, TCP port number and ICMP message type
ip-access-list-name	An optional parameter, which is used to filter the names of the IP access control list in the exported information Only the information about the IP message in the specified IP access control list can be exported.
access-group	An optional parameter, which is used to filter the names of the IP access control list in the exported information Only the information about the IP message in the specified IP access control list can be exported.
interface	An optional parameter, which is used to filter the port name of the exported information Only the information about the IP message satisfied the designated port can be exported.

Command mode

Management mode

Instruction

The command is used to find the destination of each received or locally generated IP message, which helps to detect the reason of communication problems.

The command is used in the following cases:

- forwarded
- forwarded as the multicast message or the broadcast message
- addressing failure during message forwarding
- Sending the **redirect** message
- Rejected because of having the original routing option
- Rejected because of illegal IP options
- Original route
- Message sent from the local machine should be segmented, but the DF is reset.
- Receiving message
- Receiving IP segment
- Sending message
- Sending broadcast/multicast
- Addressing failure when message is generated locally
- Locally generated message is segmented
- Received message is filtered
- Transmitted message is filtered
- Link layer fails to be encapsulated (only for Ethernet)
- Unknown protocol

This command may export lots of information. You'd better use it when the switch is in the free state. Otherwise, the performance of the system will be badly affected. You are recommended to filter the output information through the IP access control list, enabling the system to export the useful message.

Command mode

Management mode

Example

switch#debug ip packet
switch#IP: s=192.168.20.120 (vlan 10), d=19.0.0.9 (vlan 10), g=192.168.20.1, len=60, redirected
IP: s=192.168.20.22 (local), d=192.168.20.120 (vlan 10), g=192.168.20.120, len=56, sending
IP: s=192.168.20.120 (vlan 10), d=19.0.0.9 (vlan 10), g=192.168.20.1, len=60, forward
IP: s=192.168.20.81 (vlan 10), d=192.168.20.22 (vlan 10), len=56, rcvd

Field	Description
IP	Means that the information is about the IP message.
s=192.168.20.120 (vlan 10)	Source address of the IP message and the interface name that receives message (for message that is not locally generated)
d=19.0.0.9 (vlan 10)	Destination address of the IP message and the interface name that sends message (if routing is successful)
g=192.168.20.1	Next-hop destination address of the IP message, which may be the gateway's address or the destination address

len	Length of the IP message
redirected	Means that the routing switch is to send the ICMP redirect message to the source host. Other cases are shown in the following: **forward** --- the message is forwarded. **forward directed broadcas**t---the message is forwarded as the **redirect** message and the message will become the physical broadcast on the transmitting interface. unroutable---the message addressing fails and the message will be dropped. source route---source route rejected source route---the current system does not support the source route, therefore, the message with the IP source route is declined. bad options---the IP option is incorrect and the message will be dropped. need frag but DF set---the local message need be fragmented,while the DF is set. rcvd---the message is locally received. rcvd fragment---the message fragment is received. sending---the locally generated message is sent. sending broad/multicast---the locally generated broadcast/muticast message is sent. sending fragment--- the IP message locally fragmented is sent. denied by in acl---It is declined by the access control list on the reception interface. denied by out acl---It is declined by the transmitter access control on the transmitter interface. unknown protocol--- unknown protocol encapsulation failed---The protocol fails to be encapsulated.It is only for the Ethernet. When the message on the Ethernet is dropped because of the ARP resolution failure, the information is displayed.

In the first information, the switch receives an IP message; the source address of the received message is 192.168.20.120; the message is from the network segment the vlan 10 interface connects; its destination address is 19.0.0.9. According to the routing table, the transmitter interface is vlan 10, the address of the gateway is 192.168.20.1 and the message length is 60 bytes. The gateway and the source host are directly connected in the same network, that is, the network that vlan 10 connects. In this case, the switch sends out the ICMP redirect message.

IP: s=192.168.20.120 (vlan 10), d=19.0.0.9 (vlan 10), g=192.168.20.1, len=60, redirected

In the second information, the transimmission of the ICMP redirect message is described. The source address is the local address 192.168.20.22. The destination address is 192.168.20.120. The message is directly sent from the vlan 10 interface to the destination address. Therefore, the gateway's address is the detination address 192.168.20.120. The length of the ICMP redirect message is 56 bytes.

IP: s=192.168.20.22 (local), d=192.168.20.120 (vlan 10), g=192.168.20.120, len=56, sending

The third information shows that the IP layer receives an IP message. The source address and destination address of the IP message are 192.168.20.120 and 19.0.0.9 respectively. The reception interface is vlan 10. By checking the routing table, the system finds that the IP message need be forwarded to the vlan10 interface. The length of the IP message is 60 bytes. The third information shows that the message shown in the first information will be forwarded after the system sends the ICMP redirect message.

IP: s=192.168.20.120 (vlan 10), d=19.0.0.9 (vlan 10), g=192.168.20.77, len=60, forward

The fourth information shows that the IP layer receives an IP message. The source address and destination address of the IP message are 192.168.20.81 and 192.168.20.22 respectively. The reception interface is vlan 10. The length of the IP message is 56 bytes. The IP message is locally received.

IP: s=192.168.20.81 (vlan 10), d=192.168.20.22 (vlan 10), len=56, rcvd

The following is an example about the output information after running the **debug ip packet detail** command. Only the newly added parts are described.

switch#debug ip packet detail

switch#IP: s=192.168.12.8 (vlan 10), d=255.255.255.255 (vlan 10), len=328, rcvd, UDP: src=68, dst=67

IP: s=192.168.20.26 (vlan 10), d=224.0.0.5 (vlan 10), len=68, rcvd, proto=89

IP: s=192.168.20.125 (vlan 10), d=192.168.20.22 (vlan 10), len=84, rcvd, ICMP: type=0, code = 0

IP: s=192.168.20.22 (local), d=192.168.20.124 (vlan 10), g=192.168.20.124, len=40, sending, TCP: src=1024, dst=23, seq=75098622, ack=161000466, win=17520, ACK

Field	Description
UDP	Name of the protocol, such as UDP, ICMP and TCP Other protocols are represented by their protocol number.
type, code	Type and code of the ICMP message
src, dst	Source address and destination address of the UDP message and the TCP message
seq	Sequence number of the TCP message
ack	Acknowledge number of the TCP message
win	Window value of the TCP message
ACK	If ACK is set in the control bit of the TCP message, the acknowledge number is valid. Other control bits include SYN, URG, FIN, PSH and RST.

The first information indicates that the UDP message is received. The source port is port 68 and the destination port is port 67.

IP: s=192.168.12.8 (vlan 10), d=255.255.255.255 (vlan 10), len=328, rcvd, UDP: src=68, dst=67

The second information indicates that the protocol number of the received message is 89.

IP: s=192.168.20.26 (vlan 10), d=224.0.0.5 (vlan 10), len=68, rcvd, proto=89

The third information indicates that the ICMP message is received. Both the type and the code of the message are represented by the number **0**.

IP: s=192.168.20.125 (vlan 10), d=192.168.20.22 (vlan 10), len=84, rcvd, ICMP: type=0, code = 0

The fourth information indicates that the TCP message is sent. The source port and destination port are port 1024 and port 23 respectively. The sequence number and the acknowledge number are 75098622 and 161000466 respectively. The size of the receiption window is 17520. The ACK logo is set. For details, refer to RFC 793—Transmission Control Protocol.

IP: s=192.168.20.22 (local), d=192.168.20.124 (vlan 10), g=192.168.20.124, len=40, sending,
TCP: src=1024, dst=23, seq=75098622, ack=161000466, win=17520, ACK

The access control list is described in the following. For example, if the messages with the source address 192.168.20.125 require to be displayed, you need to define the standard access control list to permit only the IP message whose source address is 192.168.20.125. You then run the command **debug ip packet** to use the access control list.

```
switch#config
switch_config#ip access-list standard abc
switch_config_std_nacl#permit 192.168.20.125
switch_config_std_nacl#exit
switch_config#exit
switch#debug ip packet abc
switch#IP: s=192.168.20.125 (vlan 101), d=192.168.20.22 (vlan 101), len=48, rcvd
```

In the previous commands, the standard access control list is used. You can also use the extensible access control list.

Related command

debug ip tcp packet

2.1.6 debug ip raw

It is used to display the IP interaction information. Run the command **no debug ip raw** to stop displaying the information.

debug ip raw [**detail**] [*access-list-group*] [**interface**]

no debug ip raw

Parameter

Parameter	Description
detail	An optional parameter, which is used to export the protocol information about IP message encapsulation, such as protocol number, UDP, TCP port number and ICMP message type
access-group	An optional parameter, which is used to filter the names of the IP access control list in the exported information Only the information about the IP message in the specified IP access control list can be exported.
interface	An optional parameter, which is used to filter the port name of the exported information Only the information about the IP message satisfied the designated port can be exported.

Command mode

Management mode

Instruction

The command is used to find the destination of each received or locally generated IP message, which helps to detect the reason of communication problems.

The command is used in the following cases:

- Forwarded
- Forwarded as the multicast message or the broadcast message
- Addressing failure during message forwarding
- Sending the **redirect** message
- Rejected because of having the original routing option
- Rejected because of illegal IP options
- Original route
- Message sent from the local machine should be segmented, but the DF is reset.
- Receiving message
- Receiving IP segment
- Sending message
- Sending broadcast/multicast
- Addressing failure when message is generated locally
- Locally generated message is segmented
- Received message is filtered
- Transmitted message is filtered
- Link layer fails to be encapsulated (only for Ethernet)
- Unknown protocol

This command may export lots of information. You'd better use it when the switch is in the free state. Otherwise, the performance of the system will be badly affected. You are recommended to filter the output information through the IP access control list, enabling the system to export the useful message.

Example

Similar to the **debug ip packet** command

Related command

2.1.7 debug ip tcp packet

It is used to display the TCP message. To stop displaying the TCP message, run the command **no debug ip tcp packet**.

debug ip tcp packet

no debug ip tcp packet

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

```
switch#debug ip tcp packet
switch#tcp: O ESTABLISHED 192.168.20.22:23 192.168.20.125:3828 seq 50659460
        DATA 1 ACK 3130379810 PSH WIN 4380
tcp: I ESTABLISHED 192.168.20.22:23 192.168.20.125:3828 seq 3130379810
        DATA 2 ACK 50659460 PSH WIN 16372
tcp: O ESTABLISHED 192.168.20.22:23 192.168.20.125:3828 seq 50659461
        DATA 50 ACK 3130379812 PSH WIN 4380
tcp: O FIN_WAIT_1 192.168.20.22:23 192.168.20.125:3828 seq 50659511
        ACK 3130379812 FIN WIN 4380
tcp: I FIN_WAIT_1 192.168.20.22:23 192.168.20.125:3828 seq 3130379812
        ACK 50659511 WIN 16321
tcp: I FIN_WAIT_1 192.168.20.22:23 192.168.20.125:3828 seq 3130379812
        ACK 50659512 WIN 16321
tcp: I FIN_WAIT_2 192.168.20.22:23 192.168.20.125:3828 seq 3130379812
        ACK 50659512 FIN WIN 16321
tcp: O TIME_WAIT 192.168.20.22:23 192.168.20.125:3828 seq 50659512
        ACK 3130379813 WIN 4380
tcp: I LISTEN 0.0.0.0:23 0.0.0.0:0 seq 3813109318
        DATA 2 ACK 8057944 PSH WIN 17440
tcp: O LISTEN 0.0.0.0:23 0.0.0.0:0 seq 8057944
        RST
```

Field	Description
tcp:	Information about the TCP message
O	Sending the TCP message
ESTABLISHED	Current state of the TCP connection For the description of the TCP connection state, refer to the description of the command **debug ip tcp transactions**.
192.168.20.22:23	Means that the source address of the message is 192.168.20.22 and the source port is port 23.
192.168.20.125:3828	Means that the destination address of the message is 192.168.20.125 and the destination port is port 3828.
seq 50659460	Means that the sequence number of the message is 50659460.
DATA 1	Means that the number of valid data bytes contained in the message is **1**.
ACK 3130379810	Means that the acknowledge number of the message is 3130379810.
PSH	Means that PSH in the control bits of the message is set.

	Other control bits include ACK, FIN, SYN, URG and RST.
WIN 4380	It is used to notify the peer reception end of the cache size. The current cache size is 4380 sizes.
I	Receiving the TCP message

If the previous fields are not displayed, the field in the TCP message does not have the valid value.

Related command

debug ip tcp transactions

2.1.8 debug ip tcp transactions

It is used to display the TCP interaction information, such as the change of the TCP connection state. Run the command **no debug ip tcp transactions** to stop displaying the information.

debug ip tcp transactions

no debug ip tcp transactions

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

switch#debug ip tcp transactions
switch#TCP: rcvd connection attempt to port 23
TCP: TCB 0xE88AC8 created
TCP: state was LISTEN -> SYN_RCVD [23 -> 192.168.20.125:3828]
TCP: sending SYN, seq 50658312, ack 3130379657 [23 -> 192.168.20.125:3828]
TCP: state was SYN_RCVD -> ESTABLISHED [23 -> 192.168.20.125:3828]
TCP: connection closed by user, state was LISTEN [23 -> 0.0.0.0:0]
TCP: state was TIME_WAIT -> CLOSED [23 -> 192.168.20.125:3827]
TCP: TCB 0xE923C8 deleted
TCP: TCB 0xE7DBC8 created
TCP: connection to 192.168.20.124:513 from 192.168.20.22:1022, state was CLOSED to SYN_SENT
TCP: sending SYN, seq 52188680, ack 0 [1022 -> 192.168.20.124:513]
TCP: state was SYN_SENT -> ESTABLISHED [1022 -> 192.168.20.124:513]
TCP: rcvd FIN, state was ESTABLISHED -> CLOSE_WAIT [1022 -> 192.168.20.124:513]
TCP: connection closed by user, state was CLOSE_WAIT [1022 -> 192.168.20.124:513]
TCP: sending FIN [1022 -> 192.168.20.124:513]
TCP: connection closed by user, state was LAST_ACK [1022 -> 192.168.20.124:513]
TCP: state was LAST_ACK -> CLOSED [1022 -> 192.168.20.124:513]
TCP: TCB 0xE7DBC8 deleted

Field	Description

TCP:	Means that the TCP interaction information is displayed.
rcvd connection attempt to port 23	Means that the connection request from peer port 23 (telnet port) is received.
TCB 0xE88AC8 created	Means a new TCP connection control block is generated and its logo is 0xE88AC8.
state was LISTEN -> SYN_RCVD	Means that the state of the TCP state machine changes from the LISTEN state to the SYN_RCVD state. The TCP state may be one of the following: LISTEN---waiting for the TCP connection request from any remote host SYN_SENT---the connection request for creating TCP connection negotiation has been sent and the reply is being waited. SYN_RCVD---the connection request from the peer has been received and the acknowledgement information and its own connection request have also been sent out; the acknowledge information about the peer's connection is being waited. ESTABLISHED---the connection is successful; the data is being transmitted; the data of the upper application can be received and sent. FIN_WAIT_1---the connection termination request has been sent to the peer; the acknowledgement information and the connection termination request from the peer are being waited. FIN_WAIT_2---the connection termination request has been sent to the peer and the acknowledgement information from the peer has been received; the connection termination request from the peer is being waited. CLOSE_WAIT--- the connection termination request from the peer has been received and the acknowledgement information has been sent out; the local user is being waited to close the connection. Once the user demands to close the connection, the system sends out the connection termination request. CLOSING--- the connection termination request has been sent to the peer and the connection termination request from the peer has been received and the acknowledgement information has been sent out; the system is waiting for the local connection termination request acknowledge from the peer. LAST_ACK---The system has received the connection termination request from the peer and acknowledged it; the system has already sent out connection termination request; the acknowledge is being waited for. TIME_WAIT---the period when the system waits for the peer to receive the acknowledgement of the connection termination request CLOSED---the connection is closed. For details, refer to RFC 793, Transmission Control Protocol.

[23 -> 192.168.20.125:3828]	The first field (23) in the bracket means the local TCP port. The second field (192.168.20.125) in the bracket means the remote IP address. The third field (3828) in the bracket means the remote TCP port.
sending SYN	Means a connection request message is sent out (SYN in the control bits of the TCP header is set). Other TCP control bits include SYN, ACK, FIN, PSH, RST and URG.
seq 50658312	Means that the sequence number for sending the message is 50658312.
ack 3130379657	Means that the acknowledgement number for sending the message is 3130379657.
rcvd FIN	Means that the connection termination request is received (FIN in the control bits of the TCP header is set).
connection closed by user	Means that the upper application requires closing the TCP connection.
connection timed out	Means that connection timeout is closed.

Related command

2.1.9 debug ip udp

It is used to display the UDP interaction information. Run the command **no debug ip udp** to stop displaying the information.

debug ip udp

no debug ip udp

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

```
switch#debug ip udp
switch#UDP: rcvd src 192.168.20.99(520), dst 192.168.20.255(520), len = 32
UDP: sent src 192.168.20.22(20001), dst 192.168.20.43(1001), len = 1008
```

Field	Description
UDP:	Means that the information is about the UDP message.
rcvd	Means that the message is received.
sent	Means that the message is sent.
src	Means the source IP address of the UDP message and the UDP port.

dst	Means the destination IP address of the UDP message and the UDP port.
len	Means the length of the UDP message.

The first line in the previous information shows that a UDP message is received. The UDP message is sent from host 192.168.20.99. Both the source port and the destination port are port 520. The destination address is 192.168.20.255. The length of the message is 32 bytes.

The second line in the previous information shows that a UDP message is sent. The local address and the destination address are 192.168.20.22 and 192.168.20.43 respectively. The source port and the destination port are port 20001 and port 1001 respectively. The length of the message is 1008 bytes.

2.1.10 ip mask-reply

It is used to enable the switch to reply the mask request of the IP address on the designated interface. Run the command **no ip mask-reply** to disable the function.

ip mask-reply

no ip mask-reply

default ip mask-reply

Parameter

The command has no parameter or keyword.

Default

The mask request of the IP address is not replied.

Command mode

Interface configuration mode

Example

```
interface vlan 11
ip mask-reply
```

2.1.11 ip mtu

It is used to set the MTU of the IP message. To reuse **MTUDefault**, run the command **no ip mtu**.

ip mtu *bytes*

no ip mtu

Parameter

Parameter	Description
bytes	Maximum transmission unit of the IP message, which is calculated by byte

Default

It varies with different physical media of the interface. It is the same as MTU. The minimum value is 68 bytes.

Command mode

Interface configuration mode

Instruction

If the length of the IP message exceeds IP MTU configured on the interface, the switch fragments the message. All devices connecting on the same physical media need be configured the same MTU. The MTU affects the IP MTU. If the value of IP MTU is the same as that of the MTU, the value of IP MTU automatically changes to the new value of the MTU when the MTU value changes. The change of the IP MTU does not affectthe MTU.

The minimum value of IP MTU is 68 bytes and the maximum value of IP MTU cannot exceed the MTU value configured on the interface.

Example

The following example shows that IP MTU on interface vlan 10 is set to 200:

```
interface vlan 10
ip mtu 200
```

Related command

mtu

2.1.12 ip redirects

It is used to send the IP ICMP **redirect** message. You can run the command **no ip redirects** not to send the IP ICMP **redirect** message.

ip redirects

no ip redirects

Parameter

The command has no parameter or keyword.

Default

The IP redirect message is sent by default. However, if you configure the hot standby switch protocol, the function is disabled automatically. If the hot standby switch protocol is cancelled, the function cannot be automatically enabled.

Command mode

Interface configuration mode

Instruction

When the switch finds that the forwarding interface of the gateway is the same as the the reception interface and the source host directly connects the logical network of the interface, the switch sends an ICMP **redirect** message, notifying the source host to take the switch as the gateway to the destination address.

If the hot standby switch protocol is configured on the interface, the message may be dropped when the IP **redirect** message is sent.

Example

The following example shows that the ICMP **redirect** message can be sent on interface vlan 10:

```
interface vlan 10
ip redirects
```

2.1.13 ip source-route

It is used to enable the routing switch to process the IP message with the source IP route. To enable the routing switch to drop the IP message with the source IP route, run the command **no ip source-route**.

ip source-route

no ip source-route

Parameter

None

Default

The IP message with the source IP route is processed.

Command mode

Global configuration mode

Example

The following command enables the routing switch to process the IP message with the source IP route.

```
ip source-route
```

Related command

ping

2.1.14 ip tcp synwait-time

It is used to set the timeout time, which is used in the case when the switch waits for the successful TCP connection. To resume to the default time, run the command **no ip tcp synwait-time**.

ip tcp synwait-time *seconds*

no ip tcp synwait-time

Parameter

Parameter	Description
seconds	Time for waiting for the TCP connection, which ranges from 5 to 300 seconds Its default value is 75 seconds.

Default

75 seconds

Command mode

Global configuration mode

Instruction

When the switch originates the TCP connection, if the TCP connection is unsuccessful after the waiting time, the switch considers that the connection fails and sends the result to the upper application. You can set the waiting time for the successful TCP connection. The default value is 75 seconds. The option has nothing with the TCP connection message forwarded by the switch. However, it is relevant with the local TCP connection of the switch.

To know the current value of the waiting time, run the command **ip tcp synwait-time ?**. The value in the square bracket is the current value.

Example

The following example shows that the waiting time of the TCP connection is set to 30 seconds:

```
switch_config#ip tcp synwait-time 30
switch_config#ip tcp synwait-time ?
  <5-300>[30] seconds        -- wait time
```

2.1.15 ip tcp window-size

It is used to set the size of the TCP window. To resume to the default value, run the command **no ip tcp window-size**.

ip tcp window-size *bytes*

no ip tcp window-size

Parameter

Parameter	Description
bytes	Size of the window whose unit is second The maximum size is 65535 bytes. The default size is 2000 bytes.

Default

2000 bytes

Command mode

Global configuration mode

Instruction

Do not hastly modify the default value of the window size unless you have a definite purpose. You can run the command **ip tcp window-size ?** to know the current value. The value in the square bracket is the current value.

Example

The following example shows that the size of the TCP window is set to 6000 bytes:

```
switch_config#ip tcp window-size 6000
switch_config#ip tcp window-size ?
  <1-65535>[6000] bytes        -- Window size
```

2.1.16 ip unreachables

It is used to enable the switch to send the ICMP unreachable message. To stop sending the message, run the command **no ip unreachables**.

ip unreachables

no ip unreachables

Parameter

The command has no parameter or keyword.

Default

The ICMP unreachable message is sent.

Command mode

Interface configuration mode

Instruction

When the switch forwards the IP message, the message is dropped if the relevant route is not in the routing table. In this case, the switch sends the ICMP unreachable message to the source host. According to the information in the ICMP unreachable message, the source host promptly detects the fault and removes it.

Example

The following example shows that the interface vlan 10 is set to send the ICMP unreachable message:

```
interface vlan 10
ip unreachables
```

2.1.17 show ip sockets

It is used to display the socket information.

show ip sockets

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

switch#show ip sockets

Proto	Local	Port	Remote	Port	In	Out
17	0.0.0.0	0	0.0.0.0	0	161	0
6	0.0.0.0	0	0.0.0.0	0	513	0
17	0.0.0.0	0	0.0.0.0	0	1698	0
17	0.0.0.0	0	0.0.0.0	0	69	0
6	0.0.0.0	0	0.0.0.0	0	23	0
17	0.0.0.0	0	0.0.0.0	0	137	122590

Field	Description
Proto	IP number The protocol number of UDP is 17 and the number of TCP is 6.
Remote	Remote address
Port	Remote port
Local	Local address
Port	Local port
In	Total number of the received bytes
Out	Total number of the transmitted bytes

2.1.18 show ip traffic

It is used to display the statistics information about the IP traffic.

show ip traffic

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

```
switch#show ip traffic
IP statistics:
Rcvd:  0 total, 0 local destination, 0 delivered
          0 format errors, 0 checksum errors, 0 bad ttl count
          0 bad destination address, 0 unknown protocol, 0 discarded
          0 filtered , 0 bad options, 0 with options
  Opts:  0 loose source route, 0 record route, 0 strict source route
          0 timestamp, 0 router alert, 0 others
  Frags: 0 fragments, 0 reassembled, 0 dropped
          0 fragmented, 0 fragments, 0 couldn't fragment
  Bcast: 0 received, 0 sent
  Mcast: 0 received, 0 sent
  Sent:  230 generated, 0 forwarded
          0 filtered, 0 no route, 0 discarded
ICMP statistics:
Rcvd: 0 total, 0 format errors, 0 checksum errors
0 redirect, 0 unreachable, 0 source quench
0 echos, 0 echo replies, 0 mask requests, 0 mask replies
0 parameter problem, 0 timestamps, 0 timestamp replies
0 time exceeded, 0 router solicitations, 0 router advertisements
Sent: 0 total, 0 errors
0 redirects, 0 unreachable, 0 source quench
0 echos, 0 echo replies, 0 mask requests, 0 mask replies
0 parameter problem, 0 timestamps, 0 timestamp replies
0 time exceeded, 0 router solicitations, 0 router advertisements

UDP statistics:
Rcvd: 28 total, 0 checksum errors, 22 no port, 0 full sock
Sent: 0 total

TCP statistics:
Rcvd: 0 total, 0 checksum errors, 0 no port
Sent: 3 total

IGMP statistics:
Rcvd: 0 total, 0 format errors, 0 checksum errors
0 host queries, 0 host reports
Sent: 0 host reports

ARP statistics:
Rcvd: 8 total, 7 requests, 1 replies, 0 reverse, 0 other
Sent: 5 total, 5 requests, 0 replies (0 proxy), 0 reverse
```

Field	Description
format errors	Means that the format of the message is incorrect, such as the incorrect length of the IP header.
bad hop count	Means that the TTL value decreases to 0 when the routing

	switch forwards the message. In this case, the message will be dropped.
no route	Means that the routing switch does not have relevant route message.

2.1.19 show tcp

It is used to display the state of all TCP connections.

show tcp

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

```
switch#show tcp
TCB 0xE9ADC8
Connection state is ESTABLISHED, unread input bytes: 934
Local host: 192.168.20.22, Local port: 1023
Foreign host: 192.168.20.124, Foreign port: 513

Enqueued bytes for transmit: 0, input: 934    mis-ordered: 0 (0 packets)

Timer          Starts     Wakeups            Next(ms)
Retrans            33           1                   0
TimeWait            0           0                   0
SendWnd             0           0                   0
KeepAlive         102           0             7199500

iss: 29139463  snduna: 29139525  sndnxt: 29139525     sndwnd:     17520
irs: 709124039  rcvnxt: 709205436  rcvwnd:      4380

SRTT: 15 ms, RXT: 2500 ms, RTV: 687 ms
minRXT: 1000 ms, maxRXT: 64000 ms, ACK hold: 200 ms

Datagrams (max data segment is 1460 bytes):
Rcvd: 102 (out of order: 0), with data: 92, total data bytes: 81396
Sent: 104 (retransmit: 0), with data: 31, total data bytes: 61
```

Field	Description
TCB 0xE77FC8	Internal identifier of the TCP connection control block
Connection state is ESTABLISHED	Currrent state of the TCP connection The TCP connection may be in one of the following state: LISTEN---waiting for the TCP connection request from any remote host SYN_SENT---the connection request has been sent and the

	reply is being waited. SYN_RCVD---the connection request from the peer has been received and the acknowledgement information and its own connection request have also been sent out; the acknowledge information about the peer's connection is being waited. ESTABLISHED---the connection is successful; the data is being transmitted; the data of the upper application can be received and sent. FIN_WAIT_1---the connection termination request has been sent to the peer; the acknowledgement information and the connection termination request from the peer are being waited. FIN_WAIT_2---the connection termination request has been sent to the peer and the acknowledgement information from the peer has been received; the connection termination request from the peer is being waited. CLOSE_WAIT--- the connection termination request from the peer has been received and the acknowledgement information has been sent out; the local user is being waited to close the connection. Once the user demands to close the connection, the system sends out the connection termination request. CLOSING--- the connection termination request has been sent to the peer and the connection termination request from the peer has been received and the acknowledgement information has been sent out; the system is waiting for the local connection termination request acknowledge from the peer. LAST_ACK---The system has received the connection termination request from the peer and acknowledged it; the system has already sent out connection termination request; the acknowledgement is being waited for. TIME_WAIT---the period when the system waits for the peer to receive the acknowledgement of the connection termination request CLOSED---the connection is closed. For details, refer to RFC 793, Transmission Control Protocol.
unread input bytes:	Data that is processed by the lower-layer TCP and the upper application has not received
Local host:	Local IP address
Local port:	Local TCP port
Foreign host:	Remote IP address
Foreign port:	Remote TCP port
Enqueued bytes for transmit:	Bytes in the transmitter queue, including the data that is sent but not yet acknowledged and the data that is not sent
input:	Bytes in the reception queue After sorting, these data waits for the upper application to accept.

mis-ordered:	Number of bytes and messages in the misordered queue After other data is received, these data can enter the receiption queue in turn and then can be received by the upper application. For example, after messages 1, 2, 4, 5 and 6 are received, messages 1 and 2 can enter the receiption queue, but messages 4, 5 and 6 have to enter the misordered queue and wait for message 3.

After that, the information about the timer of the current connection is displayed, including its startup times, timeout times and the next-time timeout time. The value 0 means that the timer does not run currently. Each connection has its own unique timer. The timeout times is less than the startup times because the timer may be reset in its process. For example, when the retransmission timer works, the system will receive the acknowledgements for all data from the peer. In this case, the retransmission timer stops running.

Timer	Starts	Wakeups	Next(ms)
Retrans	33	1	0
TimeWait	0	0	0
SendWnd	0	0	0
KeepAlive	102	0	7199500

Field	Description
Timer	Name of the timer
Starts	Startup times of the timer
Wakeups	Timeout times of the timer
Next(ms)	Next-time timeout time (unit: ms) The value 0 means the timer does not run.
Retrans	Retransmission timer, which is used to trigger resending data The timer is started up after the data is sent. If the data is not acknowledged by the peer within the timeout time, the data will be resent.
TimeWait	Time Waiting timer, which is used to know that the peer has already received the acknowledgement of the connection termination request.
SendWnd	Timer of the transmission window, which is used to asure that the transmission wind resume to the normal size after the TCP acknowledgement information is dropped
KeepAlive	Keep-alive timer, which is used to asure that the communication link is in normal state and the peer is still in the connection state It triggers the testing message to be sent for testing the state of the communication link and the peer.

After the timer is displayed, the sequence number of the TCP connection is displayed. TCP uses the sequence number to gurantee reliable and orderly data transmission. The local or remote host can control the traffic and send the acknowledgement information according to the sequence number.

iss: 29139463 snduna: 29139525 sndnxt: 29139525 sndwnd: 17520

irs: 709124039 rcvnxt: 709205436 rcvwnd: 4380

Field	Description
iss:	Sequence number of original transmission
snduna:	Sequence number of the first byte in the data that is already sent but whose acknowledgement information has not been received
sndnxt:	Transmission sequence number of the first data in the data that is sent later
sndwnd:	TCP window size of the remote host
irs:	Original receiption sequence number, that is, original transmission sequence number of the remote host
rcvnxt:	Receiption sequence number that is acknowledged recently
rcvwnd:	TCP window size of the local host

The transmission time recorded by the local host is displayed afterwards. The system can adapt itself to different networks according to the transmission time.

SRTT: 15 ms, RXT: 2500 ms, RTV: 687 ms
minRXT: 1000 ms, maxRXT: 64000 ms, ACK hold: 200 ms

Field	Description
SRTT:	Round-trip time after smooth processing
RXT:	Retransmission timeout time
RTV:	Change value of the round-trip time
MinRXT:	Permissible minimum retransmission timeout time
MaxRXT:	Permissible maximum retransmission timeout time
ACK hold:	Maximum delay time when the acknowledgement is delayed for being sent together with the data

Datagrams (max data segment is 1460 bytes):
Rcvd: 102 (out of order: 0), with data: 92, total data bytes: 81396
Sent: 104 (retransmit: 0), with data: 31, total data bytes: 61

Field	Description
max data segment is	Maximum length of the data segment which is permitted by the connection
Rcvd:	Number of messages that the local host receives during the connection procedure, including the number of the misordered messages
with data:	Number of messages that contain valid data
total data bytes:	Number of data bytes contained by the message
Sent:	Number of messages that are sent or resent by the local host during the connection procedure
with data:	Number of messages that contain valid data
total data bytes:	Number of data bytes contained by the message

Related command

show tcp brief

show tcp tcb

2.1.20 show tcp brief

It is used to display the brief information about the TCP connection.

show tcp brief [**all**]

Parameter

Parameter	Description
all	An optional parameter, which means that all ports are displayed If the parameter is not entered, the system does not display the ports in the LISTEN state.

Command mode

Management mode

Example

```
switch#show tcp brief
TCB          Local Address          Foreign Address          State
0xE9ADC8     192.168.20.22:1023     192.168.20.124:513       ESTABLISHED
0xEA34C8     192.168.20.22:23       192.168.20.125:1472      ESTABLISHED
```

Field	Description
TCB	Internal identifier of the TCP connection
Local Address	Local IP address and the TCP port
Foreign Address	Remote IP address and the TCP port
State	State of the connection For details, refer to the description of the **show tcp** command.

Related command

show tcp

show tcp tcb

2.1.21 show tcp statistics

It is used to display the TCP statistics data.

show tcp statistics

Parameter

The command has no parameter or keyword.

Command mode

Management mode

Example

```
switch#show tcp statistics
Rcvd: 148 Total, 0 no port
0 checksum error, 0 bad offset, 0 too short
131 packets (6974 bytes) in sequence
0 dup packets (0 bytes)
0 partially dup packets (0 bytes)
0 out-of-order packets (0 bytes)
0 packets (0 bytes) with data after window
0 packets after close
0 window probe packets, 0 window update packets
0 dup ack packets, 0 ack packets with unsend data
127 ack packets (247 bytes)
Sent: 239 Total, 0 urgent packets
6 control packets
123 data packets (245 bytes)
0 data packets (0 bytes) retransmitted
110 ack only packets (101 delayed)
0 window probe packets, 0 window update packets
4 Connections initiated, 0 connections accepted, 2 connections established
3 Connections closed (including 0 dropped, 1 embryonic dropped)
5 Total rxmt timeout, 0 connections dropped in rxmt timeout
1 Keepalive timeout, 0 keepalive probe, 1 Connections dropped in keepalive
```

Field	Description
Rcvd:	Statistics data about the messages received by the routing switch
Total	Total number of the received messages
no port	Number of messages showing the destination port does not exist
checksum error	Number of messages showing that sum check is incorrect
bad offset	Number of messages showing that the data offset is incorrect
too short	Number of messages showing that the message length is less than the minimum effective length
packets in sequence	Number of messages that are received in turn
dup packets	Number of received duplicate messages
partially dup packets	Number of received messages that are partly duplicated
out-of-order packets	Number of misordered messages
packets with data after window	Number of messages whose data exceeds the receiption window
packets after close	Number of messages that are received after the connection is closed

window probe packets	Number of received messages about window probe
window update packets	Number of received messages about window update
dup ack packets	Number of received messages that are duplicately acknowledged
ack packets with unsent data	Number of received messages that are acknowledged but has not been sent
ack packets	Number of received messages that are acknowledged
Sent	Statistics data about messages that are sent by the routing switch
Total	Total number of the transmitted messages
urgent packets	Number of the transmitted urgent messages
control packets	Number of the transmitted control messages (SYN, FIN or RST)
data packets	Number of the transmitted data messages
data packets retransmitted	Number of the retransmitted data messages
ack only packets	Number of the purely acknowledged messages
window probe packets	Number of the transmitted window probe messages
window update packets	Number of the transmitted window update messages
Connections initiated	Number of the locally initiated connections
connections accepted	Number of the locally received connections
connections established	Number of the locally established connections
Connections closed	Number of the locally closed connections
Total rxmt timeout	Total number of retransmission timeouts
Connections dropped in rxmit timeout	Number of the connections dropped because of retransmission timeout
Keepalive timeout	Number of Keepalive timeouts
keepalive probe	Number of the transmitted messages for keepalive probe
Connections dropped in keepalive	Number of the connections dropped because of Keepalive

Related command

2.1.22 show tcp tcbl

It is used to display the state of a certain TCP connection.

show tcp tcb address

Parameter

Parameter	Description

address	TCB address of the TCP connection TCB is an identifier of the TCP connection in the system, which can be obtained by the command **show tcp brief**.

Command mode

Management mode

Example

For detailed explanation, refer to the command **show tcp**.

```
switch_config#show tcp tcb 0xea38c8

TCB 0xEA38C8
Connection state is ESTABLISHED, unread input bytes: 0
Local host: 192.168.20.22, Local port: 23
Foreign host: 192.168.20.125, Foreign port: 1583

Enqueued bytes for transmit: 0, input: 0    mis-ordered: 0 (0 packets)

Timer          Starts     Wakeups            Next(ms)
Retrans             4           0                   0
TimeWait            0           0                   0
SendWnd             0           0                   0
KeepAlive          +5           0             6633000

iss: 10431492  snduna: 10431573  sndnxt: 10431573     sndwnd:     17440
irs: 915717885  rcvnxt: 915717889  rcvwnd:      4380

SRTT: 2812 ms, RXT: 18500 ms, RTV: 4000 ms
minRXT: 1000 ms, maxRXT: 64000 ms, ACK hold: 200 ms

Datagrams (max data segment is 1460 bytes):
Rcvd: 5 (out of order: 0), with data: 1, total data bytes: 3
Sent: 4 (retransmit: 0), with data: 3, total data bytes: 80
```

Related command

show tcp

show tcp brief

2.2 ACL Configuration Commands

The following are the access control list (ACL) configuration commands:

- deny
- ip access-group
- ip access-list
- permit
- show ip access-list

2.2.1 deny

You can run the **deny** command in IP ACL configuration mode to forbid some regulations. You can run the **no deny** command to remove the forbidden regulations from the IP ACL.

deny source [*source-mask*] [**log**]

no deny source [*source-mask*] [**log**]

deny protocol source source-mask destination destination-mask [**precedence** precedence] [**tos** tos] [log]

no deny protocol source source-mask destination destination-mask [**precedence** precedence] [**tos** tos] [log]

The following syntax can be appled to the ICMP protocol:

deny icmp source source-mask destination destination-mask [icmp-type] [**precedence** precedence] [**tos** tos] [log]

The following syntax can be appled to the IGMP protocol:

deny igmp source source-mask destination destination-mask [igmp-type] [**precedence** precedence] [**tos** tos] [log]

The following syntax can be appled to the TCP protocol:

deny tcp source source-mask [operator port] destination destination-mask [operator port] [**established**] [**precedence** precedence] [**tos** tos] [log]

The following syntax can be appled to theUDP protocol:

deny udp source source-mask [operator port] destination destination-mask [operator port] [**precedence** precedence] [**tos** tos] [log]

Parameter

Parameter	Description
protocol	Protocol name or IP number It can be a keyword, such as icmp, igmp, igrp, ip, ospf, tcp or udp; it can be an integer from 0 to 255. The keyword **ip** is used for matching any Internet protocol, including ICMP, TCP and UDP. Some protocols can be further limited. See the following description.
source	Source network or host number The following two ways can be used to specify the source: ● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.
source-mask	Network mask of the source address The keyword **any** is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.
destination	Destination network or host number The following two ways can be used to specify the destination:

	● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.
destination-mask	Network mask of the destination address The keyword **any** is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.
precedence *precedence*	An optional parameter, which means that the packets can be filtered by precedence It is specified with an integer from 0 to 7.
tos *tos*	An optional parameter, which means that the packets can be filtered by the service layer. It is specified with an integer from 0 to 15.
icmp-type	An optional parameter, which means that the ICMP packets can be filtered by the ICMP message type. The ICMP message type is specified with an integer from 0 to 225.
igmp-type	An optional parameter, which means that the IGMP packets can be filtered by the IGMP message type or the IGMP message name. The IGMP message type is specified with an integer from 0 to 15.
operator	An optional parameter, which means to compare the source or the destination port The operations include the **lt** (less than) operation, the **gt** (larger than) operation, the **eq** (equal to) operation and the **neq** (unequal to) operation. If the operator is behind the parameter **source** and **source-mask**, it must match the source port. If the operator is behind the parameter **destination** and **destination-mask**, it must match the destination port.
port	An optional parameter, which means a decimal number or name of the TCP/UDP port The port number is a number from 0 to 65535. The names of TCP ports are listed in the part “Usage Policy”. When the TCP is filtered, only the name of the TCP port can be used. The names of UDP ports are listed in the part “Instruction”. When the UDP is filtered, only the name of the UDP port can be used.
established	An optional parameter to the TCP protocol, which means a connection has been established If the ACK bit or the RST bit in the TCP packet is set, a TCP connection is matched. You also can initialize a TCP packet to establish a connection.
log	An optional parameter, which is used to record the log

Command mode

IP access control list configuration mode

Instruction

You can control the virtual terminal path access and limit the content in the route choice update by transmitting the ACL control packet. After the matching operation, the extensible ACL stops to be checked. The IP segment is promptly received by any extensible IP ACL. The extensible ACL is used to control the virtual terminal path access and limit the content in the route choice update. The source ICP port, type of the service value or precedence of the packet need not be matched.

Note:

After an ACL is originally created, any following content is written to the bottom of the list.

The following are the TCP port names which are used to replace the port numbers. You can find relative references about these protocols according to the current distribution number. You can find the corresponding port numbers of these protocols by entering a question mask.

Bgp、ftp、ftp-data、login、pop2、pop3、smtp、telnet、www

The following are the UDP port names which are used to replace the port numbers. You can find relative references about these protocols according to the current distribution number. You can find the corresponding port numbers of these protocols by entering a question mask.

Domain、snmp、syslog、tftp

Example

The following example shows that network segment is forbidden:

ip access-list standard filter
deny 192.168.5.0 255.255.255.0

Note:

The IP ACL ends with an implicit **deny** regulation.

Related command

ip access-group

ip access-list

permit

show ip access-list

2.2.2 ip access-group

It is used to control an interface access. To delete the designated access group, run the command **no ip access-group**.

ip access-group {*access-list-name*}{**in** | **out**}

no ip access-group {*access-list-name*}{**in** | **out**}

Parameter

Parameter	Description
access-list-name	Name of the access control list, which is a character string with up to 20 characters
in	Access control list used on the incoming interface
out	Access control list used on the outgoing interface

Command mode

Interface configuration mode

Instruction

The ACL can be used both on the incoming interface and the outgoing interface. For the standard incoming ACL, the source address of the packet can be checked according to the ACL after the packet is received. For the standard extensible ACL, the switch also checks the destination address. If the ACL permits the address, the system continues to process the packet. If the ACL denies the packet, the system drops the packet and returns an ICMP unreachable message.

For the standard outgoing ACL, after a packet is received and routed to the control interface, the switch ckecks the source address of the packey according to the ACL. For the extensible ACL, the switch also checks the ACL at the receiption end. If the ACL permits the address, the switch transmits the packet. If the ACL denies the address, the switch drops the packet and returns an ICMP unreachable message.

If the designated ACL does not exist, all packets can pass through.

Example

The following example shows that the filter list is applied on Ethernet interface 0:

```
interface ethernet 0
ip access-group filter out
```

Related command

ip access-list

show ip access-list

2.2.3 ip access-list

It is used to enter the IP ACL configuration mode where you can add or delete the access regulation. You can run the **exit** command to go back to the configuration mode.

You can run the command **no ip access-list** to delete an IP ACL.

ip access-list {**standard** | **extended**} *name*

no ip access-list {**standard** | **extended**} *name*

Parameter

Parameter	Description
standard	Specifies a standard ACL
extended	Specifies an extensible ACL
name	Name of the ACL, which is a character string with up to 20 characters

Default

No IP access control list is defined.

Command mode

Global configuration mode

Instruction

The command is used to enter the IP ACL configuration mode. In IP ACL configuration mode,you can run the **deny** command or the **permit** command to configure the access regulation.

Example

The following example shows that a standard ACL is configured.

```
ip access-list standard filter
deny 192.168.1.0 255.255.255.0
permit any
```

Related command

deny

ip access-group

permit

show ip access-list

2.2.4 permit

It is used in IP ACL configuration mode to configure the **permit** regulations. To remove the **permit** regulations, run the command **no permit**.

permit source [*source-mask*] [**log**]

no permit source *[source-mask*] [**log**]

permit protocol source *source-mask* **destination** *destination-mask* [**precedence** *precedence*] [**tos** *tos*] [**log**]

no permit protocol source *source-mask* **destination** *destination-mask* [**precedence** *precedence*] [**tos** *tos*] [**log**]

The following syntax can be applied to the ICMP protocol:

permit icmp source *source-mask* **destination** *destination-mask* [*icmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**log**]

The following syntax can be applied to the IGMP protocol:

permit igmp source *source-mask* **destination** *destination-mask* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**log**]

The following syntax can be applied to the TCP protocol:

permit tcp source *source-mask* [**operator** *port*] **destination** *destination-mask* [**operator** *port*] [**established**] [**precedence** *precedence*] [**tos** tos] [log]

The following syntax can be applied to the UDP protocol:

permit udp source *source-mask* [**operator port** [*port*]] **destination** *destination-mask* [**operator** *port]* [**precedence** *precedence*] [**tos** *tos*] [**log**]

Parameter

Parameter	Description
protocol	Protocol name or IP number It can be a keyword, such as icmp, igmp, igrp, ip, ospf, tcp or udp; it can be an integer from 0 to 255. The keyword **ip** is used for matching any Internet protocol, including ICMP, TCP and UDP. Some protocols can be further limited. See the following description.
source	Source network or host number The following two ways can be used to specify the source: ● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.
source-mask	Network mask of the source address The keyword **any** is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.
destination	Destination network or host number The following two ways can be used to specify the destination: ● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.
destination-mask	Network mask of the destination address The keyword **any** is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.
precedence *precedence*	An optional parameter, which means that the packets can be filtered by precedence It is specified with an integer from 0 to 7.

tos *tos*	An optional parameter, which means that the packets can be filtered by the service layer. It is specified with an integer from 0 to 15.
icmp-type	An optional parameter, which means that the ICMP packets can be filtered by the ICMP message type. The ICMP message type is specified with an integer from 0 to 225.
igmp-type	An optional parameter, which means that the IGMP packets can be filtered by the IGMP message type or the IGMP message name. The IGMP message type is specified with an integer from 0 to 15.
operator	An optional parameter, which means to compare the source or the destination port The operations include the **lt** (less than) operation, the **gt** (larger than) operation, the **eq** (equal to) operation and the **neq** (unequal to) operation. If the operator is behind the parameter **source** and **source-mask**, it must match the source port. If the operator is behind the parameter **destination** and **destination-mask**, it must match the destination port.
port	An optional parameter, which means a decimal number or name of the TCP/UDP port The port number is a number from 0 to 65535. The names of TCP ports are listed in the part “Usage Policy”. When the TCP is filtered, only the name of the TCP port can be used. The names of UDP ports are listed in the part “Instruction”. When the UDP is filtered, only the name of the UDP port can be used.
established	An optional parameter to the TCP protocol, which means a connection has been established If the ACK bit or the RST bit in the TCP packet is set, a TCP connection is matched. You also can initialize a TCP packet to establish a connection.
log	An optional parameter, which is used to record the log

Command mode

IP access control list configuration mode

Instruction

You can control the virtual terminal path access and limit the content in the route choice update by transmitting the ACL control packet. After the matching operation, the extensible ACL stops to be checked. The IP segment is promptly received by any extensible IP ACL. The extensible ACL is used to control the virtual terminal path access and limit the content in the route choice update. The source ICP port, type of the service value or precedence of the packet need not be matched.

Note:

After an ACL is originally created, any following content is written to the bottom of the list.

The following are the TCP port names which are used to replace the port numbers. You can find relative references about these protocols according to the current distribution number. You can find the corresponding port numbers of these protocols by entering a question mask.

Bgp、ftp、ftp-data、login、pop2、pop3、smtp、telnet、www

The following are the UDP port names which are used to replace the port numbers. You can find relative references about these protocols according to the current distribution number. You can find the corresponding port numbers of these protocols by entering a question mask.

Domain、snmp、syslog、tftp

Example

The following example shows that network segment 192.168.5.0 is permitted:

```
ip access-list standard filter
permit 192.168.5.0 255.255.255.0
```

Note:

The IP ACL ends with an implicit **deny** regulation.

Related command

deny

ip access-group

ip access-list

show ip access-list

2.2.5 show ip access-list

It is used to display the content of the current IP AC.

show ip access-list[*access-list-name*]

Parameter

Parameter	Description
access-list-name	Name of the access control list, which is a character string with up to 20 characters

Default

All standard and extensible IP ACLs are displayed.

Command mode

Management mode

Instruction

The command is used to specify a specific ACL.

Example

The following information appears after you run the command **show ip access-list** without a designated ACL:

```
Switch# show ip access-list
ip access-list standard aaa
permit 192.2.2.1
permit 192.3.3.0 255.255.255.0
ip access-list extended bbb
permit tcp any any eq www
permit ip any any
```

The following information appears after you run the command **show ip access-list** with a designated ACL:

```
ip access-list extended bbb
permit tcp any any eq www
permit ip any any
```

2.3 IP ACL Configuration Commands Based on Physical Ports

The following are ACL configuration commands based on physical ports:

- deny
- ip access-group
- ip access-list
- permit
- show ip access-list

2.3.1 deny

You can run the **deny** command in IP ACL configuration mode to configure **deny** regulations. You can run the **no deny** command to remove the forbidden regulations from the IP ACL.

deny source [*source-mask*]

no deny source [*source-mask*]

deny protocol source source-mask destination destination-mask [**tos** tos]

no deny protocol source source-mask destination destination-mask [**tos** tos]

The following syntax can be applied to the ICMP protocol:

deny icmp source source-mask destination destination-mask [icmp-type] [**tos** tos]

The following syntax can be applied to the IGMP protocol:

deny igmp source source-mask destination destination-mask [igmp-type] [**tos** tos]

The following syntax can be applied to the TCP protocol:

deny tcp source source-mask [operator port] destination destination-mask [operator port] [**tos** tos]

The following syntax can be applied to the UDP protocol:

deny udp source source-mask [operator port] destination destination-mask [operator port] [**tos** tos]

Parameter

Parameter	Description
protocol	Protocol name or IP number It can be a keyword, such as icmp, igmp, igrp, ip, ospf, tcp or udp; it can be an integer from 0 to 255. The keyword **ip** is used for matching any Internet protocol, including ICMP, TCP and UDP. Some protocols can be further limited. See the following description.
source	Source network or host number The following two ways can be used to specify the source: ● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.
source-mask	Network mask of the source address The keyword **any** is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.
destination	Destination network or host number The following two ways can be used to specify the destination: ● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.
destination-mask	Network mask of the destination address The keyword **any** is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.
precedence *precedence*	An optional parameter, which means that the packets can be filtered by precedence It is specified with an integer from 0 to 7.
tos *tos*	An optional parameter, which means that the packets can be filtered by the service layer. It is specified with an integer from 0 to 15.
icmp-type	An optional parameter, which means that the ICMP packets can be filtered by the ICMP message type. The ICMP message type is specified with an integer from 0 to 225.
igmp-type	An optional parameter, which means that the IGMP packets can

	be filtered by the IGMP message type or the IGMP message name. The IGMP message type is specified with an integer from 0 to 15.
operator	An optional parameter, which means to compare the source or the destination port The operations include the **lt** (less than) operation, the **gt** (larger than) operation, the **eq** (equal to) operation and the **neq** (unequal to) operation. If the operator is behind the parameter **source** and **source-mask**, it must match the source port. If the operator is behind the parameter **destination** and **destination-mask**, it must match the destination port.
port	An optional parameter, which means a decimal number or name of the TCP/UDP port The port number is a number from 0 to 65535. The names of TCP ports are listed in the part “Usage Policy”. When the TCP is filtered, only the name of the TCP port can be used. The names of UDP ports are listed in the part “Instruction”. When the UDP is filtered, only the name of the UDP port can be used.
established	An optional parameter to the TCP protocol, which means a connection has been established If the ACK bit or the RST bit in the TCP packet is set, a TCP connection is matched. You also can initialize a TCP packet to establish a connection.
log	An optional parameter, which is used to record the log

Command mode

IP access control list configuration mode

Instruction

You can control the virtual terminal path access and limit the content in the route choice update by transmitting the ACL control packet. After the matching operation, the extensible ACL stops to be checked. The IP segment is promptly received by any extensible IP ACL. The extensible ACL is used to control the virtual terminal path access and limit the content in the route choice update. The source ICP port, type of the service value or precedence of the packet need not be matched.

Note:

After an ACL is originally created, any following content is written to the bottom of the list.

Example

The following example shows that network segment 192.168.5.0 is forbidden:

ip access-list standard filter
deny 192.168.5.0 255.255.255.0

Note:

The IP ACL ends with an implicit **deny** regulation.

Related command

ip access-group

ip access-list

permit

show ip access-list

2.3.2 ip access-group

It is used to control an interface access. To delete the designated access group, run the command **no ip access-group**.

ip access-group {*access-list-name*}

no ip access-group {*access-list-name*}

Parameter

Parameter	Description
access-list-name	Name of the access control list, which is a character string with up to 20 characters

Command mode

Interface configuration mode

Instruction

The ACL can be used on the incoming interface. For the standard incoming ACL, the source address of the packet can be checked according to the ACL after the packet is received. For the extensible ACL, the switch also checks the destination address. If the ACL permits the address, the system continues to process the packet. If the ACL denies the packet, the system drops the packet and returns an ICMP unreachable message.

If the designated ACL does not exist, all packets can pass through.

Example

The following example shows that the **filter** list is applied on Ethernet interface 10:

```
interface f0/10
ip access-group filter
```

Related command

ip access-list

show ip access-list

2.3.3 ip access-list

After you run the command, the system enters the IP ACL configuration mode. In this mode, you can add or delete the access regulations. You can run the **exit** command to enable the system to enter the configuration mode

You can run the command **no ip access-list** to delete the IP ACL.

ip access-list {**standard** | **extended**} *name*

no ip access-list {**standard** | **extended**} *name*

Parameter

Parameter	Description
standard	Specifies a standard ACL.
extended	Specifies an extensible ACL.
name	Name of an ACL, which is a character string with up to 20 characters

Default

No IP ACL is defined.

Command mode

Global configuration mode

Instruction

It is used to enter the IP ACL configuration mode. In this mode, you can run the **deny** or **permit** command to configure the access regulation.

Example

The following example shows that a standard ACL is configured.

```
ip access-list standard filter
deny 192.168.1.0 255.255.255.0
permit any
```

Related command

deny

ip access-group

permit

show ip access-list

2.3.4 permit

It is used in IP ACL configuration mode to configure the **permit** regulation. You can rum the command **no permit** to remove the **permit** regulations from the IP ACL.

permit source [*source-mask*]

no permit source *[source-mask*]

permit protocol source *source-mask* **destination** *destination-mask* [**tos** *tos*]

no permit protocol source *source-mask* **destination** *destination-mask* [**tos** *tos*]

The following syntax can be applied to the ICMP protocol:

permit icmp source *source-mask* **destination** *destination-mask* [*icmp-type*] [**tos** *tos*]

The following syntax can be applied to the IGMP protocol:

permit igmp source *source-mask* **destination** *destination-mask* [*igmp-type*] [**tos** *tos*]

The following syntax can be applied to the TCP protocol:

permit tcp source *source-mask* [**operator** *port*] **destination** *destination-mask* [**operator** *port*] [**tos** tos]

The following syntax can be applied to the UDP protocol:

permit udp source *source-mask* [**operator port** [*port*]] **destination** *destination-mask* [**tos** *tos*]

Parameter

Parameter	Description
protocol	Protocol name or IP number It can be a keyword, such as icmp, igmp, igrp, ip, ospf, tcp or udp; it can be an integer from 0 to 255. The keyword **ip** is used for matching any Internet protocol, including ICMP, TCP and UDP. Some protocols can be further limited. See the following description.
source	Source network or host number The following two ways can be used to specify the source: ● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.
source-mask	Network mask of the source address The keyword **any** is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.
destination	Destination network or host number The following two ways can be used to specify the destination: ● Binary system in 32 bits ● Decimal system separated by four dots The keyword **any** is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.
destination-mask	Network mask of the destination address The keyword **any** is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.
precedence *precedence*	An optional parameter, which means that the packets can be

	filtered by precedence It is specified with an integer from 0 to 7.
tos *tos*	An optional parameter, which means that the packets can be filtered by the service layer. It is specified with an integer from 0 to 15.
icmp-type	An optional parameter, which means that the ICMP packets can be filtered by the ICMP message type. The ICMP message type is specified with an integer from 0 to 225.
igmp-type	An optional parameter, which means that the IGMP packets can be filtered by the IGMP message type or the IGMP message name. The IGMP message type is specified with an integer from 0 to 15.
operator	An optional parameter, which means to compare the source or the destination port The operations include the **lt** (less than) operation, the **gt** (larger than) operation, the **eq** (equal to) operation and the **neq** (unequal to) operation. If the operator is behind the parameter **source** and **source-mask**, it must match the source port. If the operator is behind the parameter **destination** and **destination-mask**, it must match the destination port.
port	An optional parameter, which means a decimal number or name of the TCP/UDP port The port number is a number from 0 to 65535. The names of TCP ports are listed in the part “Usage Policy”. When the TCP is filtered, only the name of the TCP port can be used. The names of UDP ports are listed in the part “Instruction”. When the UDP is filtered, only the name of the UDP port can be used.
established	An optional parameter to the TCP protocol, which means a connection has been established If the ACK bit or the RST bit in the TCP packet is set, a TCP connection is matched. You also can initialize a TCP packet to establish a connection.
log	An optional parameter, which is used to record the log

Command mode

IP access control list configuration mode

Instruction

You can control the virtual terminal path access and limit the content in the route choice update by transmitting the ACL control packet. After the matching operation, the extensible ACL stops to be checked. The IP segment is promptly received by any extensible IP ACL. The extensible ACL is used to control the virtual terminal path

access and limit the content in the route choice update. The source ICP port, type of the service value or precedence of the packet need not be matched.

Note:

After an ACL is originally created, any following content is written to the bottom of the list.

Example

The following example shows that network segment 192.168.5.0 is permitted:

ip access-list standard filter
permit 192.168.5.0 255.255.255.0

Note:

The IP ACL ends with an implicit **deny** regulation.

Related command

deny

ip access-group

ip access-list

show ip access-list

2.3.5 show ip access-list

It is used to display the content of the current IP ACL.

show ip access-list[*access-list-name*]

Parameter

Parameter	Description
access-list-name	Name of an ACL, which is a character string with up to 20 characters

Default

All standard and extensible IP ACLs are displayed.

Command mode

Management mode

Instruction

It is used to specify a specific ACL.

Example

The following information appears after you run the command **show ip access-list** without specifying an ACL.

Switch# show ip access-list

```
ip access-list standard aaa
permit 192.2.2.1
permit 192.3.3.0 255.255.255.0
ip access-list extended bbb
permit tcp any any eq 25
permit ip any any
```

The following information appears after you run the command **show ip access-list** with a specified ACL.

```
ip access-list extended bbb
permit tcp any any eq 25
permit ip any any
```