



**QTECH**  
МИР ДОСТУПНЕЕ

NETWORK PROTOCOL CONFIGURATION COMMANDS

# Network Protocol Configuration Commands

## Table of contents

<b>CHAPTER 1 IP ADDRESS CONFIGURATION COMMANDS</b>	<b>5</b>
1.1. IP Address Configuration Commands	5
1.1.1. arp	5
1.1.2. arp pending-time	6
1.1.3. arp max-incomplete	7
1.1.4. arp max-gw-retries	7
1.1.5. arp retry-allarp	8
1.1.6. arp timeout	8
1.1.7. arp send-gratuitous	9
1.1.8. clear arp-cache	10
1.1.9. ip address	10
1.1.10. ip directed-broadcast	12
1.1.11. ip forward-protocol	12
1.1.12. ip helper-address	13
1.1.13. ip host	14
1.1.14. ip default-gateway	15
1.1.15. ip proxy-arp	15
1.1.16. show arp	16
1.1.17. show hosts	17
1.1.18. show ip interface	18
<b>CHAPTER 2 DHCP CLIENT CONFIGURATION COMMANDS</b>	<b>20</b>
2.1. DHCP Client Configuration Commands	20
2.1.1. ip address dhcp	20
2.1.2. ip dhcp client	21
2.1.3. ip dhcp-server	22
2.1.4. show dhcp lease	23
2.1.5. show dhcp server	24
2.1.6. debug dhcp	25
<b>CHAPTER 3 DHCP SEVER CONFIGURATION COMMANDS</b>	<b>26</b>
3.1. DHCPD Configuration Commands	26
3.1.1. ip dhcpd ping packet	26
3.1.2. ip dhcpd ping timeout	26
3.1.3. ip dhcpd write-time	27
3.1.4. ip dhcpd pool	28
3.1.5. ip dhcpd enable	28

3.2. DHCPD Address Pool Configuration Commands	29
3.2.1. network	29
3.2.2. range	30
3.2.3. default-router	31
3.2.4. dns-server	31
3.2.5. domain-name	32
3.2.6. lease	32
3.2.7. netbios-name-server	33
3.2.8. ip-bind	34
3.2.9. debug ip dhcpd packet	34
3.2.10. debug ip dhcpd event	35
3.2.11. debug ip dhcpd all	35
3.3. DHCPD Management Commands	36
3.3.1. show ip dhcpd statistic	36
3.3.2. show ip dhcpd binding	36
3.3.3. clear ip dhcpd statistic	37
3.3.4. clear ip dhcpd binding	37
CHAPTER 4 IP SERVICE CONFIGURATION COMMANDS	39
4.1. IP Service Configuration Commands	39
4.1.1. clear tcp	39
4.1.2. clear tcp statistics	41
4.1.3. debug arp	41
4.1.4. debug ip icmp	43
4.1.5. debug ip packet	47
4.1.6. debug ip raw	52
4.1.7. debug ip tcp packet	53
4.1.8. debug ip tcp transactions	55
4.1.9. debug ip udp	58
4.1.10. ip mask-reply	59
4.1.11. ip mtu	60
4.1.12. ip redirects	61
4.1.13. ip route-cache	61
4.1.14. ip route-cache hit-numbers	63
4.1.15. ip route-cache age-exf	64
4.1.16. ip route-cache cache-pbr	64
4.1.17. ip route-cache age-delay	65
4.1.18. ip route-cache softcache-alive-time	66
4.1.19. ip route-cache software-index	67
4.1.20. ip route-cache hardware-index	67

4.1.21.	ip route-cache-aging-time	68
4.1.22.	ip source-route	69
4.1.23.	ip tcp synwait-time	69
4.1.24.	ip tcp window-size	70
4.1.25.	ip unreachable	71
4.1.26.	show ip cache	72
4.1.27.	show ip irdp	73
4.1.28.	show ip sockets	74
4.1.29.	show ip traffic	75
4.1.30.	show tcp	77
4.1.31.	show tcp brief	82
4.1.32.	show tcp statistics	83
4.1.33.	show tcp tcb	86
<b>4.2.</b>	<b>ACL Configuration Commands</b>	<b>87</b>
4.2.1.	deny	88
4.2.2.	ip access-group	91
4.2.3.	ip access-list	92
4.2.4.	permit	93
4.2.5.	show ip access-list	97
<b>4.3.</b>	<b>IP ACL Configuration Commands Based on Physical Ports</b>	<b>98</b>
4.3.1.	deny	98
4.3.2.	ip access-group	101
4.3.3.	ip access-list	102
4.3.4.	permit	103
4.3.5.	show ip access-list	106

# 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

#### Syntax

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

#### Default value

No entries are permanently installed in the ARP cache.

### Command mode

global configuration mode

### Usage guidelines

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 pending-time

#### Syntax

To set the waiting time of ARP cache resolution, run the following command:

**arp pending-time** *seconds*

#### Parameter

parameter	description
<i>seconds</i>	Sets the waiting time of ARP cache resolution, whose unit is second.

#### Default value

15 seconds

### Command mode

Global configuration mode

### Usage guidelines

The first ARP cache resolution will generate an incomplete entry and this command will then be used to set the life-time of this incomplete entry.

### Example

The following example shows how to set the waiting time of ARP cache resolution to 10 seconds.

```
arp pending-time 10
```

### Related command

**show arp**

### 1.1.3. arp max-incomplete

#### Syntax

To set the maximum number of incomplete ARP entries., run the preceding command.

**arp max-incomplete** *number*

#### Parameter

parameter	description
<i>number</i>	Sets the maximum number of incomplete ARP entries.

#### Default value

The default value is 0, meaning no threshold exists.

#### Command mode

Global configuration mode

#### Usage guidelines

This command can be used to set the maximum number of the incomplete entries during ARP cache resolution.

#### Example

The following example shows how to set the maximum number of the incomplete ARP cache entries to 10:

```
arp max-incomplete 10
```

#### Related command

**show arp**

### 1.1.4. arp max-gw-retries

#### Syntax

To set the maximum retransmissions of the Re-Detect packets, run the following command:

**arp max-gw-retries** *number*

#### Parameter

parameter	description
<i>number</i>	Sets the maximum retransmissions of the Re-Detect packets.

#### Default value

3

## Command mode

Global configuration mode

## Usage guidelines

The ARP entries, which the routing entry gateway depends on, require being redetected at their aging so that the promptness and correctness of the hardware subnet routing can be guaranteed. This command is here used for setting the maximum ARP retransmissions in the redetection process. The bigger its value is, the greater chance the detection has.

## Example

The following example shows how to set the maximum retransmissions of the Re-Detect packets to 5:

```
arp max-gw-retries 5
```

## Related command

**show arp**

### 1.1.5. arp retry-allarp

## Syntax

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

## Usage guidelines

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**

### 1.1.6. arp timeout

## Syntax



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
<i>seconds</i>	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 value

14400 seconds (4 hours)

#### Command mode

interface configuration mode

#### Usage guidelines

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**

### 1.1.7. arp send-gratuitous

#### Syntax

To configure ARP send-gratuitous function, use the arp send-gratuitous command **arp send-gratuitous [ interval value ]**

#### Parameter

parameter	description
-----------	-------------

<b>interval</b>	Set the intervals of arp send-gratuitous
<i>value</i>	Set time interval, the default is 120 seconds. The range is 15 to 600 seconds

### Command mode

routing interface configuration mode

### Usage guidelines

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

### 1.1.8. clear arp-cache

#### Syntax

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

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

#### Parameter

parameter	description
<i>ip-address</i>	IP or subnets
<i>mask</i>	Subnets mask

### Command mode

EXEC

### Example

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

```
clear arp-cache
```

### Related commands

arp

### 1.1.9. ip address

#### Syntax

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 of 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 generated by the system, if the upper application does not specify the source 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
<i>ip-address</i>	IP address
<i>mask</i>	IP mask
<b>secondary</b>	(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 value

No IP address is defined for the interface.

#### Command mode

interface configuration mode

#### Usage guidelines

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
```

### 1.1.10. ip directed-broadcast

#### Syntax

To enable the translation of a directed broadcast to physical broadcasts, use the **ip directed-broadcast** interface configuration command

```
ip directed-broadcast [access-list-name]
```

```
no ip directed-broadcast
```

#### Parameter

parameter	description
<i>access-list-name</i>	(Optional) Access list name. If specified, a broadcast must pass the access list to be forwarded.

#### Default value

By default, all IP directed broadcasts are dropped.

#### Command mode

interface configuration mode

#### Example

The following example enables forwarding of IP directed broadcasts on Ethernet interface 1/0:

```
interface vlan 10
ip directed-broadcast
```

### 1.1.11. ip forward-protocol

#### Syntax

After the **ip helper-address** command is configured on interface, the **ip forward-protocol** command is used to specify which UDP to be forwarded.

```
ip forward-protocol udp [port]
no ip forward-protocol udp [port]
default ip forward-protocol udp
```

#### Parameter

parameter	description
<i>port</i>	(optional) Destination port that controls which UDP requests are forwarded

#### Default value

Request that forwards NETBIOS Name Service.

#### Command mode

global configuration mode

#### Usage guidelines

The netbios name service request is forwarded by default. Not to forward netbios name service request, use one of the following commands:

```
no ip forward-protocol udp netbios-ns
```

```
no ip forward-protocol udp 137
```

To stop forwarding all UDP broadcasting requests, use the following command.

```
no ip forward-protocol udp
```

#### Example

```
switch_config#ip forward-protocol udp 137
```

#### Related commands

```
ip helper-address
```

#### 1.1.12. ip helper-address

#### Syntax

To enable the forwarding of User Datagram Protocol (UDP) broadcasts to specific IP helper-addresses which can be used with this command.

```
ip helper-address address
```

```
no ip helper-address [address]
```

#### Parameter

parameter	description
-----------	-------------

<i>address</i>	IP helper address
----------------	-------------------

### Default value

IP helper-address is not configured

### Command mode

interface configuration mode

### Usage guidelines

The ip helper-address command does not work on an X.25 interface on a destination router because the router cannot determine if the packet was intended as a physical broadcast.

### Example

```
The following example configures IP helper-address 1.0.0.1 on the interface vlan 10
interface vlan 10
ip helper-address 1.0.0.1
```

### Related commands

**ip forward-protocol udp**

### 1.1.13. ip host

#### Syntax

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
<i>name</i>	Host name
<i>address</i>	IP address

### Default value

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
```

#### 1.1.14. ip default-gateway

##### Syntax

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
<i>address</i>	IP address

##### Default value

no configuration

##### Command 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
```

#### 1.1.15. ip proxy-arp

##### Syntax

To enable proxy Address Resolution Protocol (ARP) on an interface, use the `ip proxy-arp` command. To disable proxy ARP on the interface, use the `no` form of this command.

```
ip proxy-arp
```

```
no ip proxy-arp
```

##### Parameter

This command has no parameters

##### Default value

proxy ARP

##### Command mode

interface configuration mode

## Usage guidelines

When routing switch receives ARP requirements, if routing switch has the routing path to required IP address, and the routing interface and the interface receive the requirements are different, routing switch will send ARP reaction from its own MAC address, and then it forward actual data packets when receive them. In this case, even if one host does not know the network topology, or without configured accurate routing, it can also communicate with remote terminal. The remote host is connected with the routing switch in the same physical subnet.

If the host need routing switch to provide such function, then the host and the routing switch must be in the same IP network, or its IP address is regarded in the IP subnet of routing switch, or rather, they can use different masks, or routing switch can not provide such service.

## Example

The following example open the proxy ARP function on the interface VLAN 10

```
interface vlan 10
```

```
ip proxy-arp
```

### 1.1.16. show arp

## Syntax

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

## Command mode

EXEC

## Usage guidelines

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

### 1.1.17. show hosts

#### Syntax

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
```

### 1.1.18. show ip interface

#### Syntax

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

#### Usage guidelines

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 unreachable : 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 DHCP CLIENT CONFIGURATION COMMANDS

## 2.1. DHCP Client Configuration Commands

The following are DHCP Client configuration command:

- ip address dhcp
- ip dhcp client
- ip dhcp-server
- show dhcp lease
- show dhcp server
- debug dhcp

The chapter describes the DHCP configuration commands. These commands are used to configure and monitor the DHCP running on the switch.

### 2.1.1. ip address dhcp

#### Syntax

Run the **ip address dhcp** command to obtain an IP address for the interface through the dynamic host configuration protocol (DHCP). Run the **no ip address dhcp** command to delete the obtained IP address.

**ip address dhcp**

**no ip address dhcp**

#### Parameter

None

#### Default value

None

#### Command mode

Interface configuration mode

#### Usage guidelines

The **ip address dhcp** command is used to allow the interface to obtain the IP address through DHCP, which is very helpful for dynamically connecting the Internet service provider (ISP) through the Ethernet interface. When the dynamic IP address is obtained and the **ip address dhcp** command is configured, the switch sends the DHCPDISCOVER message to the DHCP server in the network. When the dynamic IP address is obtained and the **no ip address dhcp** command is configured, the switch sends the DHCPRELEASE message.

#### Example

The following example shows that the VLAN11 interface obtains the IP address through the DHCP protocol.

!

```
interface vlan11
```

```
ip address dhcp
```

### Related command

**ip dhcp client**

**ip dhcp-server**

**show dhcp lease**

**show dhcp server**

### 2.1.2. ip dhcp client

#### Syntax

It is used to configure parameters at the DHCP client server of the local switch.

```
ip dhcp client { minlease seconds | retransmit count | select seconds }
```

```
no ip dhcp client { minlease | retransmit | select }
```

#### Parameter

Parameter	Description
<b>minlease</b> <i>seconds</i>	Acceptable minimum lease time (optional), ranging from 60 to 86400 seconds
<b>retransmit</b> <i>count</i>	Times of retransmitting the protocol message (optional), ranging from 1 to 10
<b>select</b> <i>seconds</i>	Interval for selecting default parameter values (optional), ranging from 0 to 30

#### Default value

`minleaseParameterDefault` is 60 seconds.

`retransmitParameterDefault` is four times.

`selectParameterDefault` is zero second.

#### Command mode

Global configuration mode

#### Usage guidelines

These commands can adjust these parameters according to network structure and requirements of DHCP server.

If the negative forms of these commands are configured, these parameters are resumed to the default settings.

### Example

The following example shows that the **minlease** parameter is set to 100 seconds at the client side of the switch:

```
ip dhcp client minlease 100
```

The following example shows that the **retransmit** parameter is set to 3 times at the client side of the switch:

```
ip dhcp client retransmit 3
```

The following example shows that the **SELECT** parameter is set to 10 seconds at the client side of the switch:

```
ip dhcp client select 10
```

### Related command

**ip address dhcp**

**ip dhcp-server**

**show dhcp lease**

**show dhcp server**

### 2.1.3. ip dhcp-server

#### Syntax

Run the **ip dhcp-server** command to specify the IP address of the DHCP server:

```
ip dhcp-server ip-address
```

```
no ip dhcp-server ip-address
```

#### Parameter

Parameter	Description
<i>ip-address</i>	IP address of the DHCP server

#### Default value

The DHCP server has no default IP address.

#### Command mode

Global configuration mode

#### Usage guidelines

It is used to specify an IP address for the DHCP server. The command cannot replace the previously specified IP address of the DHCP server.

You can run the negative form of the command to clear the previously configured IP address of the DHCP server.

### Example

The following example shows that the server whose IP address 192.168.20.1 is designated as the DHCP server:

```
ip dhcp-server 192.168.20.1
```

### Related command

**ip address dhcp**

**ip dhcp client**

**show dhcp lease**

**show dhcp server**

### 2.1.4. show dhcp lease

#### Syntax

Run the **show dhcp lease** command to check the information about the DHCP server distribution used by the current switch:

**Show dhcp lease**

#### Parameter

None

#### Default value

None

#### Command mode

Management mode

#### Usage guidelines

It is used to check the information about the DHCP server distribution used by the current switch.

### Example

The following example shows that the information about the DHCP server distribution used by the current switch is displayed:

```
switch#show dhcp lease
```

```
Temp IP addr: 192.168.20.3 for peer on Interface: vlan11
```

```
Temp sub net mask: 255.255.255.0
```

```
DHCP Lease server: 192.168.1.3, state: 4 Rebinding
```

```
DHCP transaction id: 2049
Lease: 86400 secs, Renewal: 43200 secs, Rebind: 75600 secs
Temp default-gateway addr: 192.168.1.2
Next timer fires after: 02:34:26
Retry count: 1 Client-ID: router-0030.80bb.e4c0-v11
```

### Related command

```
ip address dhcp
ip dhcp client
ip dhcp-server
show dhcp server
debug dhcp
```

### 2.1.5. show dhcp server

#### Syntax

Run the **show dhcp server** command to display the known DHCP server information:

```
show dhcp server
```

#### Parameter

None

#### Default value

None

#### Command mode

Management mode

#### Usage guidelines

It is used to display the known DHCP server information.

#### Example

The following example shows that the known DHCP server information is displayed:

```
switch#show dhcp sever
DHCP server: 255.255.255.255
Leases: 0
Discovers: 62 Requests: 0 Declines: 0 Releases: 0
Offers: 0 Acks: 0 Naks: 0 Bad: 0
Subnet: 0.0.0.0, Domain name:
```

### Related command



**ip address dhcp**  
**ip dhcp client**  
**ip dhcp-server**  
**show dhcp lease**

### 2.1.6. debug dhcp

#### Syntax

Run the **debug dhcp** command to check the running state of the DHCP:

**debug dhcp [detail]**  
**no debug dhcp [detail]**

#### Parameter

Parameter	Description
<b>detail</b>	Displays the content of the DHCP message.

#### Default value

The relevant information is not displayed.

#### Command mode

Management mode

#### Usage guidelines

The following example shows that important information about the running of DHCP is displayed.

```
switch#debug dhcp
switch#2000-4-22 10:50:40 DHCP: Move to INIT state, xid: 0x7
2000-4-22 10:50:40 DHCP: SDISCOVER attempt # 1, sending 277 byte DHCP
packet
2000-4-22 10:50:40 DHCP:      B'cast on vlan11 interface from 0.0.0.0
2000-4-22 10:50:40 DHCP: Move to SELECTING state, xid: 0x7
2000-4-22 10:50:46 DHCP: SDISCOVER attempt # 2, sending 277 byte DHCPpacket
2000-4-22 10:50:46 DHCP:      B'cast on vlan11 interface from 0.0.0.0
2000-4-22 10:50:54 DHCP: SDISCOVER attempt # 3, sending 277 byte DHCPpacket
```

#### Related command

**show dhcp lease**

## CHAPTER 3 DHCP SEVER CONFIGURATION COMMANDS

### 3.1. DHCPD Configuration Commands

The following are DHCPD configuration commands:

- ip dhcpd ping packet
- ip dhcpd ping timeout
- ip dhcpd write-time
- ip dhcpd pool
- ip dhcpd enable
- ip dhcpd disable

#### 3.1.1. ip dhcpd ping packet

##### Syntax

```
ip dhcpd ping packet pkgs
```

##### Parameter

Parameter	Description
<i>pkgs</i>	Number of ICMP packets sent when the DHCP server detects whether the address is distributed

##### Default value

2

##### Command mode

Global configuration mode

##### Usage guidelines

You can run the following command to configure the DHCP server to send *n* ICMP packets when the DHCP server checks whether the IP address is distributed.

```
ip dhcpd ping packets n
```

##### Example

The following example shows that the DHCP server has sent one ICMP packet when it checks whether the IP address is distributed.

```
ip dhcpd ping packets 1
```

#### 3.1.2. ip dhcpd ping timeout

##### Syntax

### **ip dhcpd ping timeout *timeout***

#### **Parameter**

Parameter	Description
<i>timeout</i>	Timeout time waiting for the ICMP response message when the DHCP server checks whether the IP address is distributed (Unit: 100 ms)

#### **Default value**

5

#### **Command mode**

Global configuration mode

#### **Usage guidelines**

The following command can be used to configure the timeout time of the ICMP response message to **n\*100 ms** when the DHCP server checks whether the IP address is distributed.

```
ip dhcpd ping timeout n
```

#### **Example**

The following example shows that the timeout time of the ICMP response message is set to **300 ms** when the DHCP server checks whether the IP address is distributed

```
ip dhcpd ping timeout 3
```

### **3.1.3. ip dhcpd write-time**

#### **Syntax**

```
ip dhcpd write-time time
```

#### **Parameter**

Parameter	Description
<i>time</i>	Interval when the DHCP server stores the address distribution information into the database (unit: minute)

#### **Default value**

60

#### **Command mode**

Global configuration mode

#### **Usage guidelines**

The following command can be used to configure that the DHCP server stores the address distribution information into the database every **n** minutes.

```
ip dhcpd write-time n
```

It is recommended not to set the write-time to a value smaller than the default value.

### Example

The following example shows that the DHCP server stores the address distribution information into the database every 1440 minutes.

```
ip dhcpd write-time 1440
```

### 3.1.4. ip dhcpd pool

#### Syntax

```
ip dhcpd pool name
```

#### Parameter

Parameter	Description
<i>name</i>	Name of the DHCP address pool

#### Default value

None

#### Command mode

Global configuration mode

#### Usage guidelines

The following command is used to add a **name** DHCP address pool and enter the dhcp address pool configuration mode.

```
ip dhcpd pool name
```

### Example

The following example shows that a **test** DHCP address pool is added and the dhcp address pool configuration mode is entered.

```
ip dhcpd pool test
```

### 3.1.5. ip dhcpd enable

#### Syntax

```
ip dhcpd enable
```

```
no ip dhcpd enable
```

#### Parameter

None

### Default value

The DHCP server is disabled.

### Command mode

Global configuration mode

### Usage guidelines

The following command is used to enable the DHCP service. In this case, the DHCP server supports the **relay** operation. For the address request that cannot be distributed, the port where **ip helper-address** is configured can forward the DHCP request.

ip dhcpd pool name

### Example

The following example shows that the DHCP server is enabled

ip dhcpd enable

## 3.2. DHCPD Address Pool Configuration Commands

The following are DHCPD address pool configuration commands:

- network
- range
- default-router
- dns-server
- domain-name
- lease
- netbios-name-server
- host
- hardware-address
- client-identifier
- client-name

### 3.2.1. network

#### Syntax

**network** ip-addr netmask

#### Parameter

Parameter	Description
<i>ip-addr</i>	Network address of the address pool for automatic allocation
<i>netmask</i>	Subnet mask

### Default value

None

### Command mode

DHCP address pool configuration mode

### Usage guidelines

The command is used to configure the network address of the address pool for automatic allocation. The command is only used in automatic allocation mode.

When configuring the command, make sure that the port receiving the DHCP message has a network number of the port IP address similar to **network**.

### Example

The following example shows that the network address of the DHCP address pool is set to 192.168.20.0 and the subnet mask is set to 255.255.255.0.

```
network 192.168.20.0 255.255.255.0
```

## 3.2.2. range

### Syntax

```
range low-addr high-addr
```

### Parameter

Parameter	Description
<i>low-addr</i>	Original address for automatically allocating the address range
<i>hogh-addr</i>	End address for automatically allocating the address range

### Default value

None

### Command mode

DHCP address pool configuration mode

### Usage guidelines

The command is used to configure the address range for automatic allocation. Up to eight address ranges can be configured for each address pool. Each range must be in the network. The command is used only in automatic allocation mode.

### Example

The following example shows that the address allocation range of the DHCP address pool is between 192.168.20.210 and 192.168.20.219.

```
range 192.168.20.210 192.168.20.219
```

### 3.2.3. default-router

#### Syntax

```
default-router ip-addr
```

#### Parameter

Parameter	Description
<i>ip-addr</i>	Default route distributed to the client

#### Default value

None

#### Command mode

DHCP address pool configuration mode

#### Usage guidelines

The command is used to configure the default route. Up to four default routes can be configured, which are separated by space.

#### Example

The following example shows that the default route allocated to the DHCP client is 192.168.20.1.

```
default-router 192.168.20.1
```

### 3.2.4. dns-server

#### Syntax

```
dns-server ip-addr ...
```

#### Parameter

Parameter	Description
<i>ip-addr</i>	DNS server address allocated to the client

#### Default value

None

#### Command mode

DHCP address pool configuration mode

#### Usage guidelines

The command is used to configure the DNS server address that is allocated to the client. Up to four DNS servers can be configured, which are separated by space.

### Example

The following example shows that the DNS server address allocated to the client is 192.168.1.3.

```
dns-server 192.168.1.3
```

### 3.2.5. domain-name

#### Syntax

```
domain-name name
```

#### Parameter

Parameter	Description
<i>name</i>	Domain name distributed to the client

#### Default value

None

#### Command mode

DHCP address pool configuration mode

#### Usage guidelines

The command is used to configure the domain name which is distributed to the client.

### Example

The following example shows that the domain name of the client is configured to **test.domain**.

```
domain-name test.domain
```

### 3.2.6. lease

#### Syntax

```
lease {days [hours][minutes] | infinite}
```

#### Parameter

Parameter	Description
<b>days</b>	Days after the address is allocated
<i>hours</i>	Hours after the address is allocated



<i>minutes</i>	Minutes after the address is allocated
<i>infinite</i>	Permanent address allocation

#### Default value

One day

#### Command mode

DHCP address pool configuration mode

#### Usage guidelines

The command is used to configure the lease time of the distributed address.

#### Example

The following example shows that the lease time of the address distributed to the client is set to 2 days and 12 hours.

```
Lease 2 12
```

### 3.2.7. netbios-name-server

#### Syntax

```
netbios-name-server ip-addr
```

#### Parameter

Parameter	Description
<i>ip-addr</i>	IP address of the netbios name server allocated to the client

#### Default value

None

#### Command mode

DHCP address pool configuration mode

#### Usage guidelines

The command is used to configure the address of the netbios name server. Up to four netbios name servers can be configured, which is separated by space.

#### Example

The following example shows that the address of the of the netbios name server is set to 192.168.1.10.

```
netbios-name-server 192.168.1.10
```

### 3.2.8. ip-bind

#### Syntax

```
ip-bind ip-addr hardware-address WORD [type] | host-name WORD | identifier WORD
```

```
no ip-bind ip-addr
```

#### Parameter

Parameter	Description
<b>hardware-address</b> <i>WORD</i> [ <i>type</i> ]	WORD is used to match the hardware address of the client machine. type used for matching the network type of the user
<b>host-name</b> <i>WORD</i>	Used for matching the host name of the user

#### Default value

None

#### Command mode

DHCP address pool configuration mode

#### Usage guideline

The command can be used to configure the host address of the manual distributed address pool.

#### Example

The following example shows how to configure the hardware address to 10:a0:0c:13:64:7d of DHCP manual distributed address 1.1.1.1.

```
ip-bind 1.1.1.1 hardware-address 10:a0:0c:13:64:7d
```

The following example shows how to configure the client ID to 01:10:a0:0c:13:64:7d of DHCP manual distributed address 1.1.1.2.

```
ip-bind 1.1.1.2 identifier 01.10.a0.0c.13.64.7d
```

The following example shows how to configure the host name as Router-test of DHCP manual distributed address 1.1.1.3.

```
ip-bind 1.1.1.3 host-name Router-test
```

### 3.2.9. debug ip dhcpd packet

#### Syntax

```
debug ip dhcpd packet
```

#### Parameter

None

**Default value**

None

**Command mode**

Exec

**Usage guideline**

The command can be used to enable debug on-off of DHCPD data packet information.

**Example**

The following example shows how to enable debug on-off of DHCPD data packet information.

```
debug ip dhcpd packet
```

**3.2.10. debug ip dhcpd event**

```
debug ip dhcpd event
```

**Parameter**

None

**Default value**

None

**Command mode**

Exec

**Usage guideline**

The command can be used to enable debug on-off of DHCPD event information.

**Example**

The following example shows how to enable debug on-off of DHCPD data event information.

```
debug ip dhcpd event
```

**3.2.11. debug ip dhcpd all**

```
debug ip dhcpdall
```

**Parameter**

None

**Default value**

None

**Command mode**

Exec

## Usage guideline

The command can be used to enable debug on-off of DHCPD all debug information.

## Example

The following example shows how to enable debug on-off of DHCPD debug information.

```
debug ip dhcpd all
```

## 3.3. DHCPD Management Commands

DHCPD management command include:

- show ip dhcpd statistic
- show ip dhcpd binding
- clear ip dhcpd statistic
- clear ip dhcpd binding

### 3.3.1. show ip dhcpd statistic

#### Syntax

```
show ip dhcpd statistic
```

#### Parameter

None

#### Default value

None

#### Command mode

All modes except the user mode

#### Usage guideline

The command can be used to display DHCPD statistics, including number of packets and address numbers of auto-distribution and manual-distribution.

## Example

The following example shows how to display statistics of DHCPD.

```
show ip dhcpd statistic
```

### 3.3.2. show ip dhcpd binding

```
show ip dhcpd binding {ip-addr}
```

#### Parameter

Parameter	Description
-----------	-------------

<i>ip-addr</i>	Shows address of the binding information.
----------------	---

**Default value**

Display the binding information of all addresses.

**Command mode**

All modes except the user mode

**Usage guideline**

The command can be used to display DHCPD address binding information, including IP address, hardware address, binding type and time-out.

**Example**

The following example shows how to display the binding information of DHCPD.

```
show ip dhcpd binding
```

**3.3.3. clear ip dhcpd statistic****Syntax**

```
clear ip dhcpd statistic
```

**Parameter**

None

**Default value**

None

**Command mode**

Exec

**Usage guideline**

The command can be used to delete statistics about numbers of DHCPD packets.

**Example**

The following example shows how to delete statistics about numbers of DHCPD packets.

```
clear ip dhcpd statistic
```

**3.3.4. clear ip dhcpd binding****Syntax**

```
clear ip dhcpd binding {ip-addr}*}
```

**Parameter**

Parameter	Description
-----------	-------------

<i>ip-addr</i>	Address needs to delete binding information
*	Delete all binding information

**Default value**

Delete binding information of designated address.

**Command mode**

Exec

**Usage guideline**

The command can be used to delete binding information of designated address.

**Example**

The following example shows how to delete binding information of 192.168.20.210.

```
clear ip dhcpd binding 192.168.20.210
```

The following example shows how to delete binding information of 192.168.20.210 and 192.168.20.211.

```
clear ip dhcpd binding 192.168.20.210 192.168.20.211
```

The following example shows how to delete all binding information.

```
clear ip dhcpd binding *
```

## CHAPTER 4 IP SERVICE CONFIGURATION COMMANDS

### 4.1. 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 unreachable
- show ip cache
- show ip irdp
- show ip sockets
- show ip traffic
- show tcp
- show tcp brief
- show tcp statistics
- show tcp tcb

#### 4.1.1. clear tcp

##### Syntax

It is used to delete a TCP connection.

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

##### Parameter

Parameter	Description
-----------	-------------

<b>local</b> host-name port	IP address and TCP port of the local host
<b>remote</b> host-name port	IP address and TCP port of the remote host
<b>tcb</b> 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 <b>show tcp brief</b> .

### Command mode

Management mode

### Usage guidelines

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**

### 4.1.2. clear tcp statistics

#### Syntax

To clear the TCP statistics data, run the following command.

**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**

### 4.1.3. debug arp

#### Syntax

To display the ARP interaction information, such as sending ARP requests, receiving ARP requests, sending ARP response and receiving ARP response, run the following command. 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, 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, 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

#### 4.1.4. debug ip icmp

##### Syntax

To display the ICMP interaction information, run the following command. 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

##### Usage guidelines

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	<p>ICMP message which means that the original parameters of the IP message are incorrect and incorrect domain is pointed out</p> <p>The following are other types of ICMP message:</p> <ul style="list-style-type: none"><li>echo reply</li><li>dst unreachable:<ul style="list-style-type: none"><li>---net unreachable</li><li>---host unreachable</li><li>---protocol unreachable</li><li>---port unreachable</li><li>---fragmentation needed and DF set</li><li>---source route failed</li><li>---net unknown</li><li>---destination host unknown</li><li>---source host isolated</li><li>---net prohibited</li><li>---host prohibited</li><li>---net tos unreachable</li><li>---host tos unreachable</li></ul></li><li>source quench</li><li>redirect messages:<ul style="list-style-type: none"><li>---net redirect</li><li>---host redirect</li><li>---net tos redirect</li><li>---host tos redirect</li></ul></li><li>echo</li><li>router advertisement</li></ul>

	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 host 192.168.20.26, 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.

### 4.1.5. debug ip packet

#### Syntax

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
<b>detail</b>	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
<i>ip-access-list-name</i>	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.
<i>access-group</i>	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.
<i>interface</i>	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

#### Usage guidelines

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

	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 transmission 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 destination 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 reception 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**

#### 4.1.6. debug ip raw

##### Syntax

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
<b>detail</b>	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
<i>access-group</i>	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.
<i>interface</i>	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

### Usage guidelines

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

debug ip tcp packet

#### 4.1.7. debug ip tcp packet

### Syntax

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 <b>debug ip tcp transactions</b> .
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**

#### 4.1.8. debug ip tcp transactions

#### Syntax

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.



<pre>state was LISTEN -&gt; SYN_RCVD</pre>	<p>Means that the state of the TCP state machine changes from the LISTEN state to the SYN_RCVD state.</p> <p>The TCP state may be one of the following:</p> <p>LISTEN---waiting for the TCP connection request from any remote host</p> <p>SYN_SENT---the connection request for creating TCP connection negotiation has been sent and the reply is being waited.</p> <p>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.</p> <p>ESTABLISHED---the connection is successful; the data is being transmitted; the data of the upper application can be received and sent.</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>TIME_WAIT---the period when the system waits for the peer to receive the acknowledgement of the connection</p>
--	---

	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

**debug ip tcp packet**

### 4.1.9. debug ip udp

#### Syntax

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.

#### 4.1.10. ip mask-reply

##### Syntax

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 value

The mask request of the IP address is not replied.

### Command mode

Interface configuration mode

### Example

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

#### 4.1.11. ip mtu

### Syntax

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
<i>bytes</i>	Maximum transmission unit of the IP message, which is calculated by byte

### Default value

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

### Usage guidelines

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 affect the 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**

### 4.1.12. ip redirects

#### Syntax

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 value

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

#### Usage guidelines

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
```

### 4.1.13. ip route-cache

#### Syntax

It is used to enable the route cache to forward the IP message. You can run the command **no ip route-cache** to deny the route cache.

**ip route-cache**  
**no ip route-cache**  
**ip route-cache same-interface**  
**no ip route-cache same-interface**

#### Parameter

Parameter	Description
<i>same-interface</i>	Enables the IP message to be rapidly forwarded from the reception interface.

#### Default value

The fast switching is allowed on the interface. The fast switching is forbidden on the same interface.

#### Command mode

Interface configuration mode

#### Usage guidelines

The route cache performs load balance to the messages transmitted or received through the source address and the destination address.

The route cache helps improving the message forwarding performance of the routing switch. However, the route cache need be denied on the low-speed path (64 k or below).

You can run the command **ip route-cache same-interface** to allow the rapid IP switching on the same interface, that is, the reception interface is the same as the transmitter interface. Normally, you are not recommended to enable the function because it collides with the **redirect** function of the routing switch. If you have an incompletely connected network, such as frame relay, you can enable the function on the interface of the frame relay. For example, the routing switches A, B and C constitute the frame relay network. Only the link from A to B and the link from B to C exist. The communication from switch A to switch B must be relayed through switch B. Switch B receives the message from switch A through a DLCI, and then forwards the message to switch C through another DLCI.

#### Example

The following example shows that the rapid switching is permitted on the same interface:

```
ip route-cache same-interface
```

The following command is used to forbid the rapid switching, including on the same interface:

```
no ip route-cache
```

The following command is used to forbid the rapid switching only on the same interface:

```
no ip route-cache same-interface
```

The following command is used to resume the system to the default settings (the rapid switching is permitted, but the rapid switching is forbidden on the same interface):

```
ip route-cache
```

#### Related command

```
show ip cache
```

#### 4.1.14. ip route-cache hit-numbers

##### Syntax

Run the following command in global configuration mode to configure the hit number needed when the items in the software route cache are added to the hardware route cache.

```
[no] ip route-cache hit-numbers hit-number
```

##### Parameter

Parameter	Description
<i>hit-number</i>	Hit number needed when the items in the software route cache are added to the hardware route cache

##### Default value

The default value is 5.

##### Command mode

Global configuration mode

##### Usage guidelines

The command is used to configure the hit number needed when the items in the software route cache are added to the hardware route cache.

##### Example

The following example shows that the items in the software route cache are added to the hardware route cache when the items are hit twice.

```
ip route-cache hit-numbers 2
```

##### Related command

```
show ip cache
```

#### 4.1.15. **ip route-cache age-exf**

##### Syntax

To delete the hardware route of a host in case the next hop of the route of the indirectly connected host is same as that of a subnet route, run the following command in global mode:

**[no] ip route-cache age-exf**

##### Parameter

None

##### Default value

It is enabled by default.

##### Command mode

Global configuration mode

##### Usage guidelines

In case the next hop of the route of the indirectly connected host is same as that of a subnet route, the command will be used to decide whether to delete the hardware route of a host.

##### Example

After the **no ip exf** command is run in global mode, the **dst:192.200.1.1 nh:192.3.3.2** route of the indirectly connected host will be generated due to the forwarding of L3 packets. If you want to open IP EXF globally in this case, two results, as shown below, will be obtained according to the shutdown or opening of **ip route-cache age-exf**.

If the **ip route-cache age-exf** command is enabled, the hardware subnet route, **dst:192.200.1.0/24 nh:192.3.3.2**, will be generated and at the same time the hardware host's route, **dst:192.200.1.1 nh:192.3.3.2**, will be deleted.

If the **ip route-cache age-exf** command is disabled, the hardware subnet route, **dst:192.200.1.0/24 nh:192.3.3.2**, will be generated and at the same time the hardware host's route, **dst:192.200.1.1 nh:192.3.3.2**, will be kept.

##### Related command

**ip exf**

**show ip cache**

#### 4.1.16. **ip route-cache cache-pbr**

##### Syntax

If you run this command in global configuration mode, the hosts that detect the routes through policy routing will also be added to the hardware host list; by default, these hosts will not be added to the list.

**[no] ip route-cache cache-pbr**



### Parameter

None

### Default value

Disabled by default

### Command mode

Global configuration mode

### Usage guidelines

After the policy routing is set, those route caches that look for routes through the policy routing cannot be added to the hardware table, and in this case the software forwarding reduces in its performance; after this command is set, these route caches can be added to the hardware table to improve the performance of the system.

### Example

The following example shows how to enable the function in global mode:

```
ip route-cache cache-pbr
```

### Related command

**show ip cache**

## 4.1.17. ip route-cache age-delay

### Syntax

To set the delay of the route cache, which is caused by ARP change, run the following command in global mode:

```
[no] ip route-cache age-delay age-delay
```

### Parameter

Parameter	Description
<i>age-delay</i>	Sets the delay of route hardware cache.

### Default value

The default value is 0.

### Command mode

Global configuration mode

### Usage guidelines

If the delay is set, the related hardware route cache will be kept from being deleted promptly at the change of ARP until the delay is done.

Note: In a network with a lot of directly connected hosts, ARP change will lead to the cancelation of related hardware route cache, leaving here a lot of packets to impact

CPU, while the delay settings can provide a temporary protection to CPU. This command is always used together with another command, `arp retry-allarp`, and when both commands are executed, the system will learn ARP again and then reset the correct egress for IP cache as soon as possible.

### Example

The following example shows how to set the delay of hardware route cache, which is caused by ARP change, to 60:

```
ip route-cache age-delay 60
```

### Related command

**show ip cache**

### 4.1.18. ip route-cache softcache-alive-time

#### Syntax

To set the lifetime of the route entries in the software cache, run the following command in global mode:

```
[no] ip route-cache softcache-alive-time alive-time
```

#### Parameter

Parameter	Description
<i>alive-time</i>	Stands for the lifetime of the route entries in the software cache, whose unit is 10ms.

#### Default value

The default value of the lifetime is 3000, that is, 30s.

#### Command mode

Global configuration mode

#### Usage guidelines

This command is used to set the lifetime of the entries in the software route cache.

### Example

The following example shows how to set the lifetime of the entries in the software route cache to 40s:

```
ip route-cache softcache-alive-time 4000
```

### Related command

**show ip cache**

#### 4.1.19. ip route-cache software-index

##### Syntax

To set the maximum time for the timer to operate the entries in the software route cache each time, run the following command in global mode:

**[no] ip route-cache software-index** *ticks*

##### Parameter

Parameter	Description
<i>ticks</i>	Sets the maximum time for the timer to operate the entries in the software route cache each time.

##### Default value

The default value of the lifetime is 1, that is, 10ms.

##### Command mode

Global configuration mode

##### Usage guidelines

This command is used to set the maximum time for the timer to operate the entries in the software route cache. The bigger the maximum time is, the sooner the invalid software route cache ages, especially when the system is busy. Hence, this command can be used to restrain the quantity of the invalid entries in the software route cache.

##### Example

The following example shows how to set the maximum time, which is for the timer to operate the entries in the software route cache, to 500ms:

```
ip route-cache software-index 50
```

##### Related command

**show ip cache**

#### 4.1.20. ip route-cache hardware-index

##### Syntax

To set the maximum time for the timer to operate the hardware route cache each time, run the following command in global mode:

**[no] ip route-cache hardware-index** *ticks*

##### Parameter

Parameter	Description
-----------	-------------

<i>ticks</i>	Sets the maximum time for the timer to operate the entries in the software route cache each time.
--------------	---

### Default value

The default value of the lifetime is 50, that is, 0.5s.

### Command mode

Global configuration mode

### Usage guidelines

This command is used to set the maximum time for the timer to operate the entries in the hardware route cache. The bigger the maximum time is, the sooner the invalid hardware route cache ages, especially when the system is busy.

### Example

The following example shows how to set the maximum time, which is for the timer to operate the entries in the software route cache, to 600ms:

```
ip route-cache hardware-index 60
```

### Related command

**show ip cache**

## 4.1.21. ip route-cache-aging-time

### Syntax

To set the lifetime of the entries in the hardware cache, run the following command in global mode:

```
[no] ip route-cache-aging-time seconds
```

### Parameter

Parameter	Description
<i>seconds</i>	Sets the lifetime of the hardware route cache.

### Default value

The default value is 300s.

### Command mode

Global configuration mode

### Usage guidelines

This command is used to set the lifetime of the entries in the hardware route cache.

### Example

The following example shows how to set the lifetime of the entries in the hardware route cache to 600s:

```
ip route-cache-aging-time 600
```

#### Related command

**show ip cache**

#### 4.1.22. ip source-route

##### Syntax

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 value

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**

#### 4.1.23. ip tcp synwait-time

##### Syntax

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
-----------	-------------

<i>seconds</i>	Time for waiting for the TCP connection, which ranges from 5 to 300 seconds  Its default value is 75 seconds.
----------------	---

### Default value

75 seconds

### Command mode

Global configuration mode

### Usage guidelines

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
```

#### 4.1.24. ip tcp window-size

### Syntax

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
<i>bytes</i>	Size of the window whose unit is second  The maximum size is 65535 bytes. The default size is 2000

	bytes.
--	--------

### Default value

2000 bytes

### Command mode

Global configuration mode

### Usage guidelines

Do not hastily 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
```

## 4.1.25. ip unreachable

### Syntax

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

**ip unreachable**

**no ip unreachable**

### Parameter

The command has no parameter or keyword.

### Default value

The ICMP unreachable message is sent.

### Command mode

Interface configuration mode

### Usage guidelines

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 unreachable
```

#### 4.1.26. show ip cache

##### Syntax

It is used to display the route cache, which is used for the rapid switching.

```
show ip cache [prefix mask][software][hardware]
```

##### Parameter

Parameter	Description
<i>prefix mask</i>	An optional parameter, which means that the destination address of the items and the items matching the designated prefix/mask are displayed
<b>software</b>	Means that only the items stored in the software route cache are displayed.
<b>hardware</b>	Means that only the items stored in the hardware route cache are displayed.

##### Command mode

Management mode

##### Example

The following example shows that the route cache is displayed:

```
switch#show ip cache
```

Source	Destination	Interface	Next Hop
192.168.20.125	2.0.0.124	vlan 210	2.0.0.124
192.168.20.124	192.168.30.124	vlan 210	2.0.0.124
2.0.0.124	192.168.20.125	vlan 11	192.168.20.125

Field	Description
Source	Source address
Destination	Destination address



Interface	Type and number of the transmitter interface
Next Hop	Address of the gateway

The following example shows that the route cache whose destination address matches the designated prefix or mask is displayed:

```
switch#show ip cache 192.168.20.0 255.255.255.0  
Source      Destination      Interface      Next Hop  
2.0.0.124   192.168.20.125  vlan 101      192.168.20.125
```

The following example shows that the route cache whose transmitter interface matches the designated interface type or mask is displayed:

```
switch#show ip cache vlan210  
Source      Destination      Interface      Next Hop  
192.168.20.125  2.0.0.124      vlan 210      2.0.0.124  
192.168.20.124  192.168.30.124  vlan 210      2.0.0.124
```

#### 4.1.27. show ip irdp

##### Syntax

It is used to display the **irdp** information.

##### Parameter

The command has no parameter or keyword.

##### Command mode

Management mode

##### Example

```
xuhao_config_vlan10# show ip irdp  
Async0/0 ICMP router discovery protocol(IRDP) : OFF  
vlan 10 ICMP router discovery protocol(IRDP) : ON  
  Advertisements occur between every 450 and 600 seconds  
  Advertisements are sent as broadcasts  
  Advertisements valid in 1800 seconds  
  Default preference: 0  
vlan 11 ICMP router discovery protocol(IRDP) : OFF  
Null0 ICMP router discovery protocol(IRDP) : OFF  
Loopback7 ICMP router discovery protocol(IRDP) : OFF
```

Loopback10 ICMP router discovery protocol(IRDP) : OFF

#### 4.1.28. show ip sockets

##### Syntax

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

#### 4.1.29. show ip traffic

##### Syntax

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

Frag: 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.

### 4.1.30. show tcp

#### Syntax

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	Current 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 reception 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 reception 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 assure that the transmission window resume to the normal size after the TCP acknowledgement information is dropped
KeepAlive	Keep-alive timer, which is used to assure 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 guarantee 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 reception sequence number, that is, original transmission sequence number of the remote host
rcvnxt:	Reception 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**

#### 4.1.31. show tcp brief

#### Syntax

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

**show tcp brief [all]**

#### Parameter

Parameter	Description
<b>all</b>	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 <b>show tcp</b> command.

#### Related command

**show tcp**

**show tcp tcb**

#### 4.1.32. show tcp statistics

##### Syntax

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 reception 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 dublicately acknowledged
ack packets with unsend 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**

**clear tcp statistics**

**4.1.33. show tcp tcb**

**Syntax**

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

**show tcp tcb address**

**Parameter**

Parameter	Description
<i>address</i>	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**

## 4.2. 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

### 4.2.1. deny

#### Syntax

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 applied to the ICMP protocol:

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

The following syntax can be applied to the IGMP protocol:

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

The following syntax can be applied 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 applied to the UDP protocol:

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

#### Parameter

Parameter	Description
<i>protocol</i>	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 <b>ip</b> 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:



	<ul style="list-style-type: none"> <li>● Binary system in 32 bits</li> <li>● Decimal system separated by four dots</li> </ul> <p>The keyword <b>any</b> is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.</p>
<i>source-mask</i>	<p>Network mask of the source address</p> <p>The keyword <b>any</b> is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.</p>
<i>destination</i>	<p>Destination network or host number</p> <p>The following two ways can be used to specify the destination:</p> <ul style="list-style-type: none"> <li>● Binary system in 32 bits</li> <li>● Decimal system separated by four dots</li> </ul> <p>The keyword <b>any</b> is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.</p>
<i>destination-mask</i>	<p>Network mask of the destination address</p> <p>The keyword <b>any</b> is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.</p>
<b>precedence</b> <i>precedence</i>	<p>An optional parameter, which means that the packets can be filtered by precedence</p> <p>It is specified with an integer from 0 to 7.</p>
<b>tos tos</b>	<p>An optional parameter, which means that the packets can be filtered by the service layer.</p> <p>It is specified with an integer from 0 to 15.</p>
<i>icmp-type</i>	<p>An optional parameter, which means that the ICMP packets can be filtered by the ICMP message type.</p> <p>The ICMP message type is specified with an integer from 0 to 225.</p>
<i>igmp-type</i>	<p>An optional parameter, which means that the IGMP packets can be filtered by the IGMP message type or the IGMP message name.</p> <p>The IGMP message type is specified with an integer from 0 to 15.</p>
<i>operator</i>	<p>An optional parameter, which means to compare the source</p>

	<p>or the destination port</p> <p>The operations include the <b>lt</b> (less than) operation, the <b>gt</b> (larger than) operation, the <b>eq</b> (equal to) operation and the <b>neq</b> (unequal to) operation. If the operator is behind the parameter <b>source</b> and <b>source-mask</b>, it must match the source port. If the operator is behind the parameter <b>destination</b> and <b>destination-mask</b>, it must match the destination port.</p>
port	<p>An optional parameter, which means a decimal number or name of the TCP/UDP port</p> <p>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 “Usage guidelines”. When the UDP is filtered, only the name of the UDP port can be used.</p>
established	<p>An optional parameter to the TCP protocol, which means a connection has been established</p> <p>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.</p>
log	<p>An optional parameter, which is used to record the log</p>

### Command mode

IP access control list configuration mode

### Usage guidelines

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
```

### 4.2.2. ip access-group

#### Syntax

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
<i>access-list-name</i>	Name of the access control list, which is a character string with up to 20 characters
<b>in</b>	Access control list used on the incoming interface
<b>out</b>	Access control list used on the outgoing interface

#### Command mode

Interface configuration mode

### Usage guidelines

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 checks the source address of the packet according to the ACL. For the extensible ACL, the switch also checks the ACL at the reception 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
```

### 4.2.3. ip access-list

#### Syntax

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
<b>standard</b>	Specifies a standard ACL
<b>extended</b>	Specifies an extensible ACL

<i>name</i>	Name of the ACL, which is a character string with up to 20 characters
-------------	---

### Default value

No IP access control list is defined.

### Command mode

Global configuration mode

### Usage guidelines

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**

## 4.2.4. permit

### Syntax

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
<b>protocol</b>	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 <b>ip</b> is used for matching any Internet protocol, including ICMP, TCP and UDP. Some protocols can be further limited. See the following description.
<b>source</b>	Source network or host number The following two ways can be used to specify the source: <ul style="list-style-type: none"><li>● Binary system in 32 bits</li><li>● Decimal system separated by four dots</li></ul> The keyword <b>any</b> is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.
<i>source-mask</i>	Network mask of the source address The keyword <b>any</b> is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.
<b>destination</b>	Destination network or host number The following two ways can be used to specify the destination: <ul style="list-style-type: none"><li>● Binary system in 32 bits</li><li>● Decimal system separated by four dots</li></ul> The keyword <b>any</b> is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.
<i>destination-mask</i>	Network mask of the destination address

	The keyword <b>any</b> is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.
<b>precedence</b> <i>precedence</i>	An optional parameter, which means that the packets can be filtered by precedence It is specified with an integer from 0 to 7.
<b>tos tos</b>	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.
<b>icmp-type</b>	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.
<b>igmp-type</b>	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.
<b>operator</b>	An optional parameter, which means to compare the source or the destination port The operations include the <b>lt</b> (less than) operation, the <b>gt</b> (larger than) operation, the <b>eq</b> (equal to) operation and the <b>neq</b> (unequal to) operation. If the operator is behind the parameter <b>source</b> and <b>source-mask</b> , it must match the source port. If the operator is behind the parameter <b>destination</b> and <b>destination-mask</b> , it must match the destination port.
<b>port</b>	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 “Usage guidelines”. When the UDP is filtered, only the name of the UDP port can be used.
<b>established</b>	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.
<b>log</b>	An optional parameter, which is used to record the log

### Command mode

IP access control list configuration mode

### Usage guidelines

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**

#### **4.2.5. show ip access-list**

##### **Syntax**

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

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

##### **Parameter**

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

##### **Default value**

All standard and extensible IP ACLs are displayed.

##### **Command mode**

Management mode

##### **Usage guidelines**

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
```

### 4.3. 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

#### 4.3.1. deny

##### Syntax

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
<b>protocol</b>	Protocol name or IP number It can be a keyword, such as icmp, igmp, igmp, ip, ospf, tcp or udp; it can be an integer from 0 to 255. The keyword <b>ip</b> is used for matching any Internet protocol, including ICMP, TCP and UDP. Some protocols can be further limited. See

	the following description.
<b>source</b>	Source network or host number The following two ways can be used to specify the source: <ul style="list-style-type: none"><li>● Binary system in 32 bits</li><li>● Decimal system separated by four dots</li></ul> The keyword <b>any</b> is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.
<i>source-mask</i>	Network mask of the source address The keyword <b>any</b> is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.
<b>destination</b>	Destination network or host number The following two ways can be used to specify the destination: <ul style="list-style-type: none"><li>● Binary system in 32 bits</li><li>● Decimal system separated by four dots</li></ul> The keyword <b>any</b> is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.
<i>destination-mask</i>	Network mask of the destination address The keyword <b>any</b> is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.
<b>precedence</b> <i>precedence</i>	An optional parameter, which means that the packets can be filtered by precedence It is specified with an integer from 0 to 7.
<b>tos tos</b>	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.
<i>icmp-type</i>	An optional parameter, which means that the ICMP packets

	<p>can be filtered by the ICMP message type.</p> <p>The ICMP message type is specified with an integer from 0 to 225.</p>
<i>igmp-type</i>	<p>An optional parameter, which means that the IGMP packets can be filtered by the IGMP message type or the IGMP message name.</p> <p>The IGMP message type is specified with an integer from 0 to 15.</p>
<b>operator</b>	<p>An optional parameter, which means to compare the source or the destination port</p> <p>The operations include the <b>lt</b> (less than) operation, the <b>gt</b> (larger than) operation, the <b>eq</b> (equal to) operation and the <b>neq</b> (unequal to) operation. If the operator is behind the parameter <b>source</b> and <b>source-mask</b>, it must match the source port. If the operator is behind the parameter <b>destination</b> and <b>destination-mask</b>, it must match the destination port.</p>
<b>port</b>	<p>An optional parameter, which means a decimal number or name of the TCP/UDP port</p> <p>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 “Usage guidelines”. When the UDP is filtered, only the name of the UDP port can be used.</p>
<b>established</b>	<p>An optional parameter to the TCP protocol, which means a connection has been established</p> <p>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.</p>
<b>log</b>	<p>An optional parameter, which is used to record the log</p>

### Command mode

IP access control list configuration mode

## Usage guidelines

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**

## 4.3.2. ip access-group

### Syntax

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
<i>access-list-name</i>	Name of the access control list, which is a character string with up to 20 characters

### Command mode

Interface configuration mode

## Usage guidelines

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
```

### 4.3.3. ip access-list

## Syntax

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
<b>standard</b>	Specifies a standard ACL.
<b>extended</b>	Specifies an extensible ACL.
<i>name</i>	Name of an ACL, which is a character string with up to 20 characters

## Default value

No IP ACL is defined.

## Command mode

Global configuration mode

### Usage guidelines

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**

#### 4.3.4. permit

### Syntax

It is used in IP ACL configuration mode to configure the **permit** regulation. You can run 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
<b>protocol</b>	<p>Protocol name or IP number</p> <p>It can be a keyword, such as <code>icmp</code>, <code>igmp</code>, <code>igrp</code>, <code>ip</code>, <code>ospf</code>, <code>tcp</code> or <code>udp</code>; it can be an integer from 0 to 255. The keyword <b>ip</b> is used for matching any Internet protocol, including ICMP, TCP and UDP. Some protocols can be further limited. See the following description.</p>
<b>source</b>	<p>Source network or host number</p> <p>The following two ways can be used to specify the source:</p> <ul style="list-style-type: none"> <li>● Binary system in 32 bits</li> <li>● Decimal system separated by four dots</li> </ul> <p>The keyword <b>any</b> is used to stand for the abbreviation of the source 0.0.0.0 and the source mask.</p>
<i>source-mask</i>	<p>Network mask of the source address</p> <p>The keyword <b>any</b> is used to stand for the abbreviation of the source host 0.0.0.0 and the source mask.</p>
<b>destination</b>	<p>Destination network or host number</p> <p>The following two ways can be used to specify the destination:</p> <ul style="list-style-type: none"> <li>● Binary system in 32 bits</li> <li>● Decimal system separated by four dots</li> </ul> <p>The keyword <b>any</b> is used to stand for the abbreviation of the destination host 0.0.0.0 and the destination mask.</p>
<i>destination-mask</i>	<p>Network mask of the destination address</p> <p>The keyword <b>any</b> is used to stand for the abbreviation of the destination 0.0.0.0 and the mask of the destination address.</p>
<b>precedence</b> <i>precedence</i>	<p>An optional parameter, which means that the packets can be filtered by precedence</p> <p>It is specified with an integer from 0 to 7.</p>
<b>tos tos</b>	<p>An optional parameter, which means that the packets can be filtered by the service layer.</p> <p>It is specified with an integer from 0 to 15.</p>
<i>icmp-type</i>	<p>An optional parameter, which means that the ICMP packets can be filtered by the ICMP message type.</p>



	The ICMP message type is specified with an integer from 0 to 225.
<i>igmp-type</i>	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.
<b>operator</b>	An optional parameter, which means to compare the source or the destination port The operations include the <b>lt</b> (less than) operation, the <b>gt</b> (larger than) operation, the <b>eq</b> (equal to) operation and the <b>neq</b> (unequal to) operation. If the operator is behind the parameter <b>source</b> and <b>source-mask</b> , it must match the source port. If the operator is behind the parameter <b>destination</b> and <b>destination-mask</b> , it must match the destination port.
<b>port</b>	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 “Usage guidelines”. When the UDP is filtered, only the name of the UDP port can be used.
<b>established</b>	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.
<b>log</b>	An optional parameter, which is used to record the log

### Command mode

IP access control list configuration mode

### Usage guidelines

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**

**4.3.5. show ip access-list**

**Syntax**

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

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

**Parameter**

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

**Default value**

All standard and extensible IP ACLs are displayed.

**Command mode**

Management mode

**Usage guidelines**

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
```