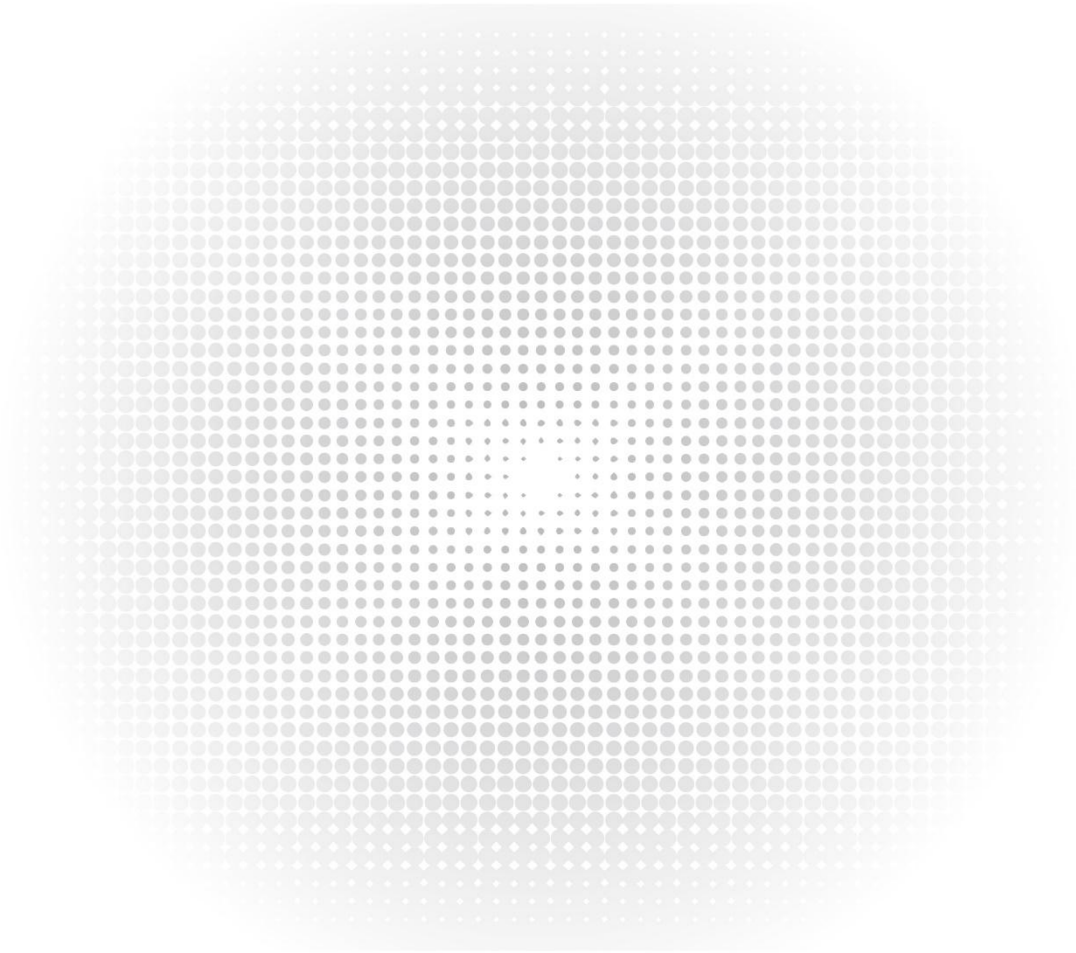
S3950-4T12S Switch Configuration Guide

Model: S3950-4T12S



CONFIGURATION GUIDE

Contents

Chapter 1 Preface

1.1 Declaration

This document updates at irregular intervals because of product upgrade or other reason.

This document is for your reference only.

1.2 Audience

This document is for the following audiences:

- System maintenance engineers
- Debugging and testing engineers
- Network monitoring engineers
- Field maintenance engineers

Chapter 2 Basic Configuration Guide

2.1 Configuring System Management

2.1.1 Overview

Function Introduction

Banner function is used for configuring messages on the devices. User can specify any messages to notify other users. Improper operations might cause critical situation such as service interrupt, in this case, a notification in advance is necessary. (E.g. to notify users "Don't reboot")

Three types of messages are supported by now:

- MOTD(message-of-the-day). Messages will display on the terminal when user connect to the device.
- login banner. Messages will display on the terminal when user login to the device. "Login mode" is required for displaying this message. Please reference the section of "Configuring User Management".
- exec banner. Messages will display on the terminal when user enter the EXEC mode.

Principle Description

This function displays notification on the terminal to reduce misoperation.

2.1.2 Configuration

Configuring a MOTD Login Banner

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create the notification

User can create a notification (one line or multiple lines) to display on all connected terminals. In the following example, the delimiting character is #. All characters between two delimiting characters will display on the terminals when user connect the device.

The message length is at most 99 lines with 1023 character in each line.

```
Switch(config)# banner motd # This is a switch #
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 Validation

Use the following command to display the configuration:

```
switch# show running
banner motd ^C
 This is a switch
^C
```

Configuring a Login Banner

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create the notification

User can create a notification (one line or multiple lines) to display on all connected terminals. "Login mode" is required for displaying this message. Please reference the section of "Configuring User Management".

In the following example, the delimiting character is #. All characters between two delimiting characters will display on the terminals when user connect the device.

The message length is at most 99 lines with 1023 character in each line.

```
banner login # admin login #
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 Validation

Use the following command to display the configuration

```
switch# show running
banner login ^C
 admin login
^C
```

Configuring an Exec Banner

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create the notification

User can create a notification (one line or multiple lines) to display on all connected terminals. In the following example, the delimiting character is #. All characters between two delimiting characters will display on the terminals when user enter the EXEC mode.

The message length is at most 99 lines with 1023 character in each line.

```
Switch(config)# banner exec # do not reboot! #
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 Validation

Use the following command to display the configuration:

```
switch# show running
banner exec ^C
 do not reboot!
^C
```

2.1.3 Application Cases

Case 1: mark the usage of the device

Set the MOTD message as "This is a switch of some area/department", user can see this message when connect to the device. If the user needs to operate a switch of another department, he can realize that he connected to a wrong device and stop misoperation.

Configuration steps

```
Switch# configure terminal
Switch(config)# banner motd # This is a switch of IT DEPARTMENT ！ ！ ！ #
Switch(config)# exit
```

Configuration files

```
switch# show running
banner motd ^C
 This is a switch of IT DEPARTMENT ！ ！ ！
^C
```

2.2 Configuring User Management

2.2.1 Overview

Function Introduction

User management increases the security of the system by keeping the unauthorized users from guessing the password. The user is limited to a specific number of attempts to successfully log in to the switch.

There are three load modes in the switch.

- In "no login" mode, anyone can load the switch without authentication.
- In "login" mode, there is only one default user.
- In "login local" mode, if you want to load the switch you need to have a user account. Local user authentication uses local user accounts and passwords that you create to validate the login attempts of local users. Each switch has a maximum of 32 local user accounts. Before you can enable local user authentication, you must define at least one local user account. You can set up local user accounts by creating a unique username and password combination for each local user. Each username must be fewer than 32 characters. You can configure each local user account with a privilege level; the valid privilege levels are 1 or 4. Once a local user is logged in, only the commands those are available for that privilege level can be displayed.

There is only one user can enter the configure mode at the same time.

Principle Description

N/A

2.2.2 Configuration

Configuring the user management in login local mode

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 et username and password

```
Switch(config)# username testname privilege 4 password 123abc<>
```

step 3 Enter the configure mode and set user management mode

```
Switch(config)# line vty 0 7
Switch(config-line)# login local
Switch(config-line)# exit
```

step 4 Exit the configure mode

```
Switch(config)# exit
```

step 5 Validation

After the above setting, login the switch will need a username and password, and user can login with the username and password created before. This is a sample output of the login prompt.

```
Username:
```

After the input the username, a password is required.

```
Username: testname
Password:
```

Authentication succeed:

```
Password:

Switch#
```

Configuring the user management in login mode

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the configure mode and set password

```
Switch(config)# line vty 0 7
Switch(config-line)# login
Switch(config-line)# line-password abc
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 Validation

After the above setting, login the switch will need the line password, and user can login with the password created before. This is a sample output of the login prompt.

```
Password:
```

Configuring Password recovery procedure

If the password is forgotten unfortunately, it can be recovered by following steps.

Step 1 Power on the system. Boot loader will start to run. The follow information will be printed on Console.

```
CPU:   MPC8247 (HiP7 Rev 14, Mask 1.0 1K50M) at 350 MHz
Board: 8247 (PCI Agent Mode)
I2C:   ready
DRAM:  256 MB
In:    serial
Out:   serial
Err:   serial
Net:   FCC1 ETHERNET, FCC2 ETHERNET [PRIME]
Press ctrl+b to stop autoboot: 3
```

Step 2 Press ctrl+b. stop autoboot.

```
Bootrom#
```

Step 3 Under boot loader interface, use the following instructions.

```
Bootrom# boot_flash_nopass
Bootrom# Do you want to  revert  to the default config file ? [Y|N|E]:
```

NOTE: Please remember your username and password.

Recovering the password may lead configuration lost or service interrupted; we strongly recommend that user should remember the username and password.

2. 2. 3 Application Cases

N/A

2.3 Configuring FTP

2.3.1 Overview

Function Introduction

You can download a switch configuration file from an FTP server or upload the file from the switch to an FTP server. You download a switch configuration file from a server to upgrade the switch configuration. You can overwrite the current startup configuration file with the new one. You upload a switch configuration file to a server for backup purposes. You can use this uploaded configuration for future downloads to the switch or another switch of the same type.

Principle Description

N/A

2.3.2 Configuration

You can copy configurations files to or from an FTP server. The FTP protocol requires a client to send a remote username and password on each FTP request to a server.

Before you begin downloading or uploading a configuration file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same network if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the ping command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download.
- When you upload a configuration file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

Downloading a configuration file by using FTP in IPv4 network

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set username and password

```
Switch(config)# ftp username test
Switch(config)# ftp password test
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 copy the configuration file

```
Switch# copy mgmt-if ftp://test:test@10.10.10.163/ startup-config.conf  flash:/startup-config.conf
```

step 5 Validation

Use the following command to display the configuration

```
Switch# show startup-config
```

Uploading a configuration file by using FTP in IPv4 network #

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set username and password

```
Switch(config)# ftp username test
Switch(config)# ftp password test
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 copy the configuration file

```
Switch# copy flash:/startup-config.conf mgmt-if ftp://test:test@10.10.10.163/startup-config.conf
```

Downloading a configuration file by using FTP in IPv6 network

Username and password settings are same as IPv4 network.

step 1 copy the configuration file

```
Switch# copy ftp://root: root@2001:1000::2/startup-config.conf  flash:/startup-config.conf
```

Uploading a configuration file by using FTP in IPv6 network

Username and password settings are same as IPv4 network.

step 1 copy the configuration file

```
Switch# copy flash:/startup-config.conf mgmt-if ftp://root:root@2001:1000::2 startup-config.conf
```

2.3.3 Application Cases

N/A

2.4 Configuring TFTP

2.4.1 Overview

Function Introduction

You can download a switch configuration file from a TFTP server or upload the file from the switch to a TFTP server. You download a switch configuration file from a server to upgrade the switch configuration. You can overwrite the current file with the new one. You upload a switch configuration file to a server for backup purposes; this uploaded file can be used for future downloads to the same or another switch of the same type.

Principle Description

N/A

2.4.2 Configuration

Before you begin downloading or uploading a configuration file by using TFTP, do these tasks:

Ensure that the workstation acting as the TFTP server is properly configured.

Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same network if you do not have a router to route traffic between subnets. Check connectivity to the TFTP server by using the ping command.

Ensure that the configuration to be downloaded is in the correct directory on the TFTP server.

For download operations, ensure that the permissions on the file are set correctly.

During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly.

Downloading a configuration file by using TFTP in IPv4 network

```
Switch# copy mgmt-if tftp://10.10.10.163/startup-config.conf flash:/startup-config.conf
```

Uploading a configuration file by using TFTP in IPv4 network

```
Switch# copy flash:/startup-config.conf mgmt-if tftp://10.10.10.163/startup-config.conf
```

Downloading a configuration file by using TFTP in IPv6 network

```
Switch# copy mgmt-if tftp://2001:1000::2/startup-config.conf flash:/startup-config.conf
```

Uploading a configuration file by using TFTP in IPv6 network

```
Switch# copy flash:/startup-config.conf mgmt-if tftp://2001:1000::2/startup-config.conf
```

2.4.3 Application Cases

N/A

2.5 Configuring SCP

2.5.1 Overview

Function Introduction

SCP, which is short for secure copy, is a part of SSH protocol. It is a remote copy technology which is based on SSH protocol. User can download a switch configuration file from a SCP server or upload the file from the switch to a SCP server. User can download a switch configuration file from a server to upgrade the switch configuration and overwrite the current file with the new one. User can upload a switch configuration file to a server for backup purposes; this uploaded file can be used for future downloads to the same or another switch of the same type.

Principle Description

N/A

2.5.2 Configuration

Before you begin downloading or uploading a configuration file by using SCP, do these tasks:

Ensure that the workstation acting as the SCP server is properly configured.

Ensure that the switch has a route to the SCP server. The switch and the SCP server must be in the same network if you do not have a router to route traffic between subnets. Check connectivity to the SCP server by using the ping command.

Ensure that the configuration to be downloaded is in the correct directory on the SCP server.

For download operations, ensure that the permissions on the file are set correctly.

During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly.

Downloading a configuration file by using SCP in IPv4 network

```
Switch# copy mgmt-if scp://10.10.10.163/startup-config.conf flash:/startup-config.conf
```

Uploading a configuration file by using SCP in IPv4 network

```
Switch# copy flash:/startup-config.conf mgmt-if scp://10.10.10.163/startup-config.conf
```

Downloading a configuration file by using SCP in IPv6 network

```
Switch# copy mgmt-if scp://2001:1000::2/startup-config.conf flash:/startup-config.conf
```

Uploading a configuration file by using SCP in IPv6 network

```
Switch# copy flash:/startup-config.conf mgmt-if scp://2001:1000::2/startup-config.conf
```

2.5.3 Application Cases

N/A

2.6 Configuring Telnet

2.6.1 Overview

Function Introduction

Telnet is a network protocol used on the Internet or local area networks to provide a bidirectional interactive text-oriented communications facility using a virtual terminal connection. User data is interspersed in-band with Telnet control information in an 8-bit byte oriented data connection over the Transmission Control Protocol (TCP). Telnet was developed in 1969 beginning with RFC 15, extended in RFC 854, and standardized as Internet Engineering Task Force (IETF) Internet Standard STD 8, one of the first Internet standards. Historically, Telnet provided access to a command-line interface (usually, of an operating system) on a remote host. Most network equipment and operating systems with a TCP/IP stack support a Telnet service for remote configuration (including systems based on Windows NT). Because of security issues with Telnet, its use for this purpose has waned in favor of SSH.

Principle Description

N/A

2.6.2 Configuration

Telnet switch with inner port

Example 1 IPv4 Network

```
Switch# telnet 10.10.29.247
Entering character mode
Escape character is '^]'.
Switch #
```

Example 2 IPv6 Network

```
Switch# telnet 2001:1000::71
Entering character mode
Escape character is '^]'.
Switch #
```

Telnet switch with management port

Example 1 IPv4 Network

```
Switch# telnet mgmt-if 10.10.29.247
Entering character mode
Escape character is '^]'.
Switch #
```

Example 2 IPv6 Network

```
Switch# telnet mgmt-if 2001:1000::2
Entering character mode
Escape character is '^]'.
Switch #
```

Configure telnet server

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable Telnet service

```
Switch(config)# service telnet enable
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

2.6.3 Application Cases

N/A

2.7 Configuring SSH

2.7.1 Overview

Function Introduction

The Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. SSH supports the Data Encryption Standard (DES) encryption algorithm, the Triple DES (3DES) encryption algorithm, and password-based user authentication. The SSH feature has an SSH server and an SSH integrated client, which are applications that run on the switch. You can use an SSH client to connect to a switch running the SSH server. The SSH server works with the SSH client supported in this release and with SSH clients. The SSH client also works with the SSH server supported in this release and with SSH servers.

Principle Description

N/A

2.7.2 Configuration

Figure 1-1 SSH system application

Create key for SSH

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create a key

```
Switch(config)# rsa key a generate
```

step 3 Create a private key named a.pri with key a and save it to flash

```
Switch(config)# rsa key a export url flash:/a.pri private ssh2
```

step 4 Create a private key named a.pub with key a and save it to flash

```
Switch(config)# rsa key a export url flash:/a.pub public ssh2
```

step 5 Exit the configure mode

```
Switch(config)# exit
```

Import the key

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Import the key a.pub we created as importKey

```
Switch(config)# rsa key importKey import url flash:/a.pub public ssh2
```

step 3 Create username and password

```
Switch(config)# username aaa privilege 4 password abc
```

step 4 Assign the key to user aaa

```
Switch(config)# username aaa assign rsa key importKey
```

step 5 Exit the configure mode

```
Switch(config)# exit
```

Use SSH to connect

step 1 Download the a.pri key on SSH client

step 2 Connect to the client

```
[root@test1 tftpboot]# ssh -i a.pri aaa@10.10.39.101
aaa@10.10.39.101's password:
Switch#
```

2.7.3 Application Cases

N/A

2.8 Configuring Time&timezone

2.8.1 Overview

Function Introduction

If no other source of time is available, you can manually configure the time and date after the system is restarted. The time remains accurate until the next system restart. We recommend that you use manual configuration only as a last resort. If you have an outside source to which the switch can synchronize, you do not need to manually set the system clock.

Principle Description

N/A

2.8.2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Configuring time and timezone

```
Switch(config)# clock set datetime 11:30:00 10 26 2013
Switch(config)# clock set summer-time dst date 6 1 2013 02:00:00 10 31 2013 02:00:00 120
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 Validation

Use the following command to display the information of time and date:

```
Switch# show clock detail
13:31:10 dst Sat Oct 26 2013
Time zone: (GMT + 08:00:00)  beijing
Summer time starts at beijing 02:00:00  06/01/2013
Summer time ends at dst 02:00:00  10/31/2013
Summer time offset:  120 minutes
```

2.8.3 Application Cases

N/A

2.9 RPC API Configuration Guide

2.9.1 Overview

Function Introduction

RPC API service allows user to configure and monitor the switch system through Remote Procedure Calls (RPC) from your program.

The service currently supports JSON-RPC over HTTP protocol together with HTTP Basic authentication.

Principle Description

RPC API service uses standard JSON-RPC over HTTP protocol to communicate the switch and your program. User may issue switch CLI commands through JSON-RPC method: 'executeCmds'. By default, the CLI mode is in privileged EXEC mode (#).

User could send JSON-RPC request via an HTTP POST request to URL: http://:/command-api. The detailed JSON-RPC request and response are show below:

JSON-RPC Request

```
{
  "params":[                          Parameters for command
    {
      "format":"text",                Expected response format, can be 'text' or 'json',
the default format is 'text'
      "version":1,                    The API version
      "cmds":[                        List of CLI commands
        "show run",                   CLI command 1
        "config t",                   CLI command 2
        "vlan database",              CLI command 3
        "vlan 1-8",                   CLI command 4
        "interface eth-0-1",          CLI command 5
        "switchport mode trunk",      CLI command 6
        "switchport trunk allowed vlan add 2",  CLI command 7
        "shutdown",                   CLI command 8
        "end",                        CLI command 9
```

```
        "show interface switchport"          CLI command 10
      ]
    }
  ],
  "jsonrpc":"2.0",                          JSON RPC protocol version. Always 2.0.
  "method":"executeCmds",                   Method to run the switch CLI commands
  "id":"70853aff-af77-420e-8f3c-fa9430733a19"      JSON RPC unique identifier
}
```

JSON-RPC Response

```
{
  "jsonrpc":"2.0",                          JSON RPC protocol version. Always 2.0.
  "id":"70853aff-af77-420e-8f3c-fa9430733a19",      JSON RPC unique identifier
  "result":[                                Result list of objects from each CLI command executed.
    {
      "sourceDetails":"version 5.1.6.fcs\n!\n …",  Output information of CLI Command 1.
                                            The Original ASCII output information returned from CLI command if this command is successfully executed.
      "errorCode":-1003,                    Error code if it is available.
      "errorDesc":"unsupported command…",       Error description if it is available.
      "warnings":"% Invalid…",              Warnings if it is available.
                                            Formatted JSON object will also be returned if it is available.
    },
    {},                                     Output information of CLI Command 2.
    {},                                     Output information of CLI Command 3.
    {},                                     Output information of CLI Command 4.
    {},                                     Output information of CLI Command 5.
    {},                                     Output information of CLI Command 6.
    {},                                     Output information of CLI Command 7.
    {},                                     Output information of CLI Command 8.
    {},                                     Output information of CLI Command 9.
    {
      "sourceDetails":" Interface name        : eth-0-1\n Switchport mode       : trunk\n …\n"
    }                                       Output information of CLI Command 10.
  ]
}
```

Python Client Example Code

Here is an example code using 'pyjsonrpc' library:

```
import pyjsonrpc
import json

http_client = pyjsonrpc.HttpClient(
  url = "http://10.10.39.64:80/command-api",
  username = "username",
  password = "password"
)

cmds = {}
cmd_list = ["show run", "config t", "vlan database", "vlan 1-8", "interface eth-0-1", "switchport mode trunk", "switchport trunk allowed vlan
add 2", "shutdown", "end", "show interface switchport"]

cmds['cmds'] = cmd_list
cmds['format'] = 'text'
cmds['version'] = 1

try:
  response = http_client.call("executeCmds", cmds)
  print("json response:");
  json_result = json.dumps(response, indent=4)
  print(json_result)
except Exception, e:
  if e.code == 401:
    print "Unauthorized user"
  else:
    print e.message
    print e.data
```

Error code

Here is a list of JSON-RPC 2.0 error code:

Error Code	Description
-32700	Parse error
-32600	Invalid Request
-32601	Method not found
-32602	Invalid param
-32603	Internal error

Here is a list of RPC-API error code:

Error Code	Description
-1000	General error
-2001	JSON RPC API Error: unsupported API version
-2002	JSON RPC API Error: must specify 'params' with 'cmds' in JSON RPC
-2003	JSON RPC API Error: unsupported command response format
-3001	Command execution failed: timed out
-3002	Command execution failed: unsupported command
-3003	Command execution failed: unauthorized command
-3004	Command execution failed: the string does not match any command in current mode
-3005	Command execution failed: can't convert to JSON format
-3006	Command execution failed: command list too short
-3007	Command execution failed: command list too long

2.9.2 Configuration

Configuring RPC API service

User could enable the RPC API service by the following steps.

The default port is 80.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable RPC API service

```
Switch(config)# service rpc-api enable
```

NOTE: Use the following command to disable rpc-api service:

```
Switch(config)# service rpc-api disable
```

step 3 Exit the configure mode

```
Switch(config)# end
```

Configuring RPC API service with HTTP Authentication

User could configure the HTTP authentication mode of RPC API service.

Currently, only HTTP Basic authentication is supported. User will receive status code: 401 (Unauthorized access) if user provides invalid user name or password.

step 1 Enter the configure mode

```
Switch# configure terminal
```

Step 2 Set the username and password, then enable the rpc-api authentication

```
Switch(config)# username myuser password mypass
Switch(config)# service rpc-api auth-mode basic
```

NOTE: Use the following command to disable authentication:

```
Switch(config)# no service rpc-api auth-mode
```

NOTE: HTTP authentication settings of RPC API service will take effect after you restart this service or reboot the system.

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show services rpc-api
RPC API service configuration:
 Server State      : enable
 Port              : 80
 Authentication Mode : basic
 VRF               : default
```

2.9.3 Application Cases

N/A

Chapter 3 Ethernet Configuration Guide

3.1 Configuring Interface

3.1.1 Overview

Function Introduction

Interface status, speed and duplex are configurable.

When the interface is configured as "no shutdown", it can work normally after cable is connected. When the interface is configured as "shutdown", no matter the cable is connected or not, the interface can not work.

If the device supports combo ports, user can choose to enable copper or fiber mode. The two modes of one port can not work together at same time. The configuration of speed or duplex at combo ports cannot be effective when combo port is working at fiber mode.

The rule of physical port name is as following: interface name format is eth-[slot]-[port]; [slot] is 0 for single pizza-box switch; when stacking is enabled, the [slot] number is according to the configuration. The [port] number is begin with 1, and increase from up to down, from left to right. The following figure shows the interface name of the device:

eth-0-1	eth-0-3	···	eth-0-23
eth-0-2	eth-0-4	...	eth-0-24

Figure 1-2 Interface Name

NOTE: To get more information about the interface type and number, please reference to the product spec.

Principle Description

N/A

3.1.2 Configuration

Configuring Interface State

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Turn on an interface

```
Switch#(config)# interface eth-0-1
Switch(config-if)# no shutdown
```

step 3 Shut down an interface

```
Switch(config-if)# interface eth-0-2
Switch(config-if)# shutdown
```

step 4 Exit the configure mode

```
Switch(config-if)# end
```

step 5 Validation

Use the following command to display the status of the interfaces:

```
Switch# show interface status
Port     Status    Duplex  Speed   Mode   Type
```

```
-------------------------------------------------------------------
eth-0-1   up       a-full  a-1000  access  1000BASE_T
eth-0-2   admin down auto    auto    access  1000BASE_T
```

Configuring Interface Speed

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the speed

Set speed of interface eth-0-1 to 100M

```
Switch(config)# interface eth-0-1
Switch(config-if)# speed 100
Switch(config-if)# no shutdown
```

Set speed of interface eth-0-2 to 1000M

```
Switch(config-if)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# speed 1000
```

Set speed of interface eth-0-3 to auto

```
Switch(config-if)# interface eth-0-3
Switch(config-if)# no shutdown
Switch(config-if)# speed auto
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

Use the following command to display the status of the interfaces:

```
Switch# show interface status
Port     Status    Duplex  Speed   Mode   Type
-------------------------------------------------------------------
eth-0-1   up       a-full  100     access  1000BASE_T
eth-0-2   up       a-full  1000    access  1000BASE_T
eth-0-3   up       a-full  a-1000  access  1000BASE_T
```

Configuring Interface Duplex

There are 3 duplex mode supported on the device:

- full mode: the interface can transmit and receive packets at same time.
- half mode: the interface can transmit or receive packets at same time.
- auto mode: the interface should negotiate with the other side to decide the duplex mode.

User can choose proper duplex mode according to the network state.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the duplex

Set duplex of interface eth-0-1 to full

```
Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config-if)# duplex full
```

Set duplex of interface eth-0-1 to half

```
Switch(config-if)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# duplex half
```

Set duplex of interface eth-0-1 to auto

```
Switch(config)# interface eth-0-3
Switch(config-if)# no shutdown
Switch(config-if)# duplex auto
```

step 4 Validation

Use the following command to display the status of the interfaces:

```
Switch# show interface status
Port     Status   Duplex  Speed   Mode   Type
-------------------------------------------------------------
eth-0-1  up       full    a-1000  access 1000BASE_T
eth-0-2  up       half    a-100   access 1000BASE_T
eth-0-3  up       a-full  a-1000  access 1000BASE_T
```

3.1.3 Application Cases

N/A

3.2 Configuring Layer3 Interfaces

3.2.1 Overview

Function Introduction

3 types of Layer3 interface are supported:

- VLAN interfaces: Logical interface with layer3 features. Connect different VLANs via IP address on the VLAN interface. VLAN interfaces can be created and deleted.
- Routed Ports: Ports are physical ports configured to be in Layer 3 mode by using the no switchport in interface configuration command.
- Layer 3 Link Aggregation Ports: Link Aggregation interfaces made up of routed ports.
- A Layer 3 switch can have an IP address assigned to each routed port and VLAN interface. All Layer 3 interfaces require an IP address to route traffic. This section shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface.

Principle Description

N/A

3.2.2 Configuration

Configuring Routed Port

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set IP address

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 1.1.1.1/24
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

Use the following command to display the brief status of the interfaces:

```
Switch# show ip interface brief
Interface          IP-Address     Status          Protocol
eth-0-1            1.1.1.1        up              up
Switch# show ip interface
Interface eth-0-1
 Interface current state: UP
 Internet address(es):
  1.1.1.1/24 broadcast 1.1.1.255
 Joined group address(es):
  224.0.0.1
 The maximum transmit unit is 1500 bytes
 ICMP error messages limited to one every 1000 milliseconds
 ICMP redirects are always sent
ICMP unreachables are always sent
 ICMP mask replies are always sent
 ARP timeout 01:00:00,  ARP retry interval 1s
 VRRP master of:  VRRP is not configured on this interface
```

Configuring VLAN Interfaces

This chapter describes configuring VLAN interfaces and using them. Several Virtual LAN (VLAN) interfaces can be configured on a single Ethernet interface. Once created, a VLAN interface functions the same as any physical interface, and it can be configured and displayed like any physical interface. Routing protocols, such as, RIP, OSPF and BGP can run across networks using VLAN interfaces.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create a vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode and set switch port attributes

```
Switch(config)# interface eth-0-2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 4 Enter the vlan interface configure mode and set IP address

```
Switch(config)# interface vlan10
Switch(config-if)# ip address 2.2.2.2/24
```

step 5 Exit the configure mode

```
Switch(config-if)# end
```

step 6 Validation

Use the following command to display the brief status of the interfaces:

```
Switch# show ip interface brief
Interface          IP-Address     Status          Protocol
vlan10             2.2.2.2        up              up

Switch# show ip interface
```

```
Interface vlan10
 Interface current state: UP
 Internet address(es):
  2.2.2.2/24 broadcast 2.2.2.255
 Joined group address(es):
  224.0.0.1
 The maximum transmit unit is 1500 bytes
 ICMP error messages limited to one every 1000 milliseconds
 ICMP redirects are always sent
 ICMP redirects are always sent
 ICMP unreachables are always sent
 ICMP mask replies are always sent
 ARP timeout 01:00:00,  ARP retry interval 1s
 VRRP master of :  VRRP is not configured on this interface
```

3.3 Configuring Interface Errdisable

3.3.1 Overview

Function Introduction

Errdisable is a mechanism to protect the system through shutdown the abnormal interface. If an interface enters errdisable state, there are two ways to recovery it from errdisabled state. The first one is to enable errdisable recovery of this reason before errdisable detection; the interface will be recovered automatically after the configured time. But if errdisable occurred first, then errdisable recovery is enabled, the errdisable will not be recovered automatically. The secondary one is configuring "no shutdown" command on the errdisabled interface.

The flap of interface link state is a potential error caused by hardware or line problem. The administrator can also configure the detection conditions of interface link flap to suppress the flap.

Principle Description

N/A

3.3.2 Configuration

Configuring Errdisable Detection

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable detect link flap errdisable

```
Switch(config)# errdisable detect reason link-flap
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the configuration of error disable:

```
Switch# show errdisable detect
ErrDisable Reason       Detection status
-----------------       ----------------
bpduguard               Enabled
bpduloop                Enabled
link-monitor-failure    Enabled
oam-remote-failure      Enabled
port-security           Enabled
link-flap               Enabled
monitor-link            Enabled
udld                    Disabled
fdb-loop                Disabled
loopback-detection      Enabled
reload-delay            Enabled
```

Configuring Errdisable Recovery

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable errdisable and set recovery interval

```
Switch(config)# errdisable recovery reason link-flap
Switch(config)# errdisable recovery interval 30
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the configuration of error disable recovery:

```
Switch# show errdisable recovery
ErrDisable Reason       Timer Status
-----------------       --------------
bpduguard               Disabled
bpduloop                Disabled
link-monitor-failure    Disabled
oam-remote-failure      Disabled
port-security           Disabled
link-flap               Enabled
udld                    Disabled
fdb-loop                Disabled
loopback-detection      Disabled
Timer interval: 30 seconds
```

Configuring suppress Errdisable link Flap

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set link flap condition

```
Switch(config)# errdisable flap reason link-flap 20 60
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the configuration of error disable flap:

```
Switch# show errdisable flap
ErrDisable Reason       Flaps    Time (sec)
-----------------       ------   ----------
link-flap               20       60
```

Checking Errdisable Status

Administrator can check the interface errdisable status though two commands.

Case 1 Enable errdisable recovery

If link flap errdisable is enabled recovery, the command will display the left time for recovery; Otherwise, will display "unrecovery".

```
Switch# show errdisable recovery
ErrDisable Reason       Timer Status
```

```
-----------------       --------------
bpduguard               Disabled
bpduloop                Disabled
link-monitor-failure    Disabled
oam-remote-failure      Disabled
port-security           Disabled
link-flap               Enabled
udld                    Disabled
fdb-loop                Disabled
loopback-detection      Disabled
Timer interval: 300 seconds
Interfaces that will be enabled at the next timeout:
Interface Errdisable Reason Time Left(sec)
--------- ----------------- --------------
eth-0-3   link-flap         25
```

Case 2 Disalbe errdisable recovery

```
Switch# show errdisable recovery
ErrDisable Reason       Timer Status
-----------------       --------------
bpduguard               Disabled
bpduloop                Disabled
link-monitor-failure    Disabled
oam-remote-failure      Disabled
port-security           Disabled
link-flap               Disabled
udld                    Disabled
fdb-loop                Disabled
loopback-detection      Disabled
Timer interval: 300 seconds
```

case 3 Display interface brief information to check errdisable state.

```
Switch# show interface status
Port     Status    Duplex  Speed   Mode   Type       Description
--------------------------------------------------------------------------------
eth-0-1   up       a-full  a-1000  TRUNK  1000BASE_SX
eth-0-2   down     auto    auto    TRUNK  Unknown
eth-0-3   errdisable a-full  a-1000  TRUNK  1000BASE_SX
eth-0-4   down     auto    auto    ACCESS  Unknown
```

3.3.3 Application Cases

N/A

3.4 Configuring MAC Address Table

3.4.1 Overview

Function Introduction

MAC address table contains address information for the switch to forward traffic between ports. The address table includes these types of address:

- Dynamic address: the source address learnt by the switch and will be aged after aging time if this address is not hit. We only support IVL learning mode.
- Static address: the source address manually added by administrators.
- Following is a brief description of terms and concepts used to describe the MAC address table:
- IVL: Independent VLAN Learning: for a given set of VLANs, if a given individual MAC Address is learned in one VLAN, it can't be used in forwarding decisions taken for that address relative to any other VLAN in the given set.
- SVL: Shared VLAN Learning: for a given set of VLANs, if an individual MAC Address is learned in one VLAN, it can be used in forwarding decisions taken for that address relative to all other VLANs in the given set.
- Reference to standard:IEEE 802.1D, IEEE 802.1Q

Principle Description

N/A

3.4.2 Configuration

Configuring Address Aging Time

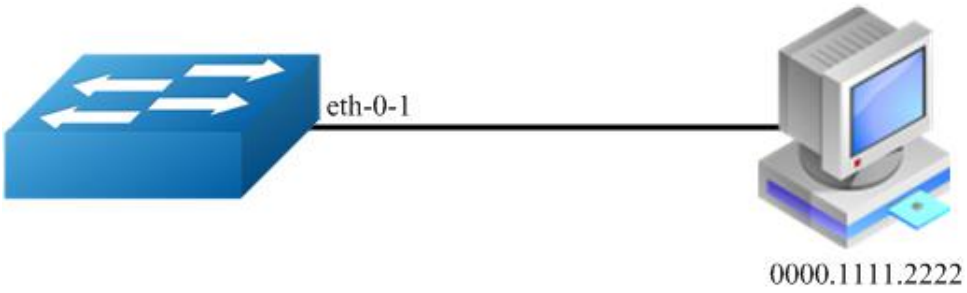


Figure 1-3 Mac address aging

The aging time is not exact time. If aging time set to N, then the dynamic address will be aged after N~2N interval. The default aging time is 300 seconds.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set dynamic address aging time

```
Switch(config)# mac-address-table ageing-time 10
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the aging time:

```
Switch# show mac address-table ageing-time
MAC address table ageing time is 10 seconds
```

Configuring Static Unicast Address

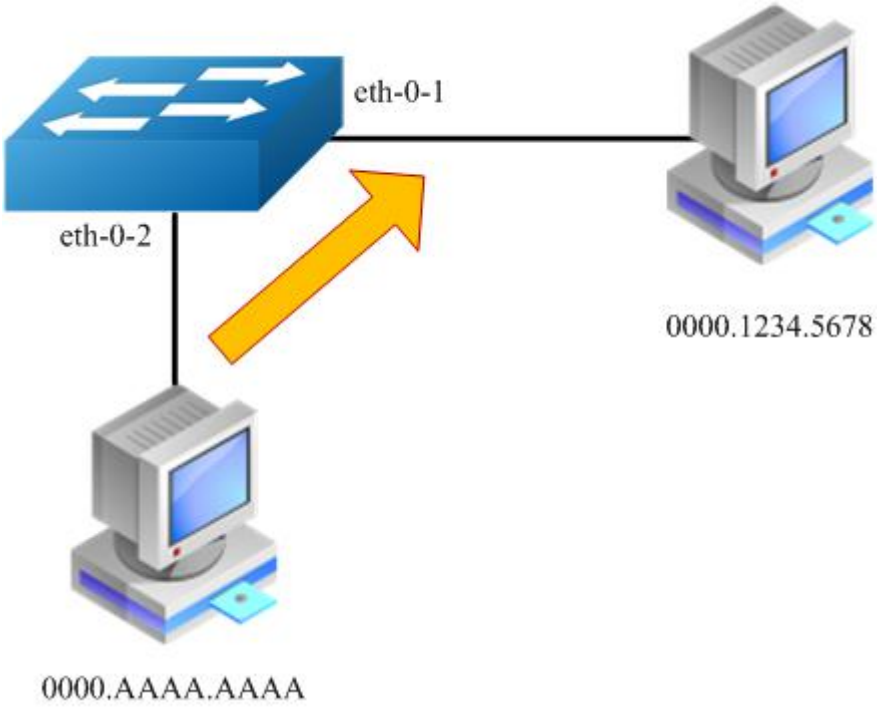


Figure 1-4 Static mac address table

Unicast address can be only bound to one port. According to the picture, Mac-Da 0000.1234.5678 should forward via eth-0-1.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set static mac address table

```
Switch(config)# mac-address-table 0000.1234.5678 forward eth-0-1 vlan 1
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the mac address table:

```
Switch# show mac address-table
Mac Address Table
------------------------------------------
(*) - Security Entry
Vlan   Mac Address     Type     Ports
----   --------------  -------  --------
1      0000.1234.5678  static   eth-0-1
```

Configuring Static Multicast Address

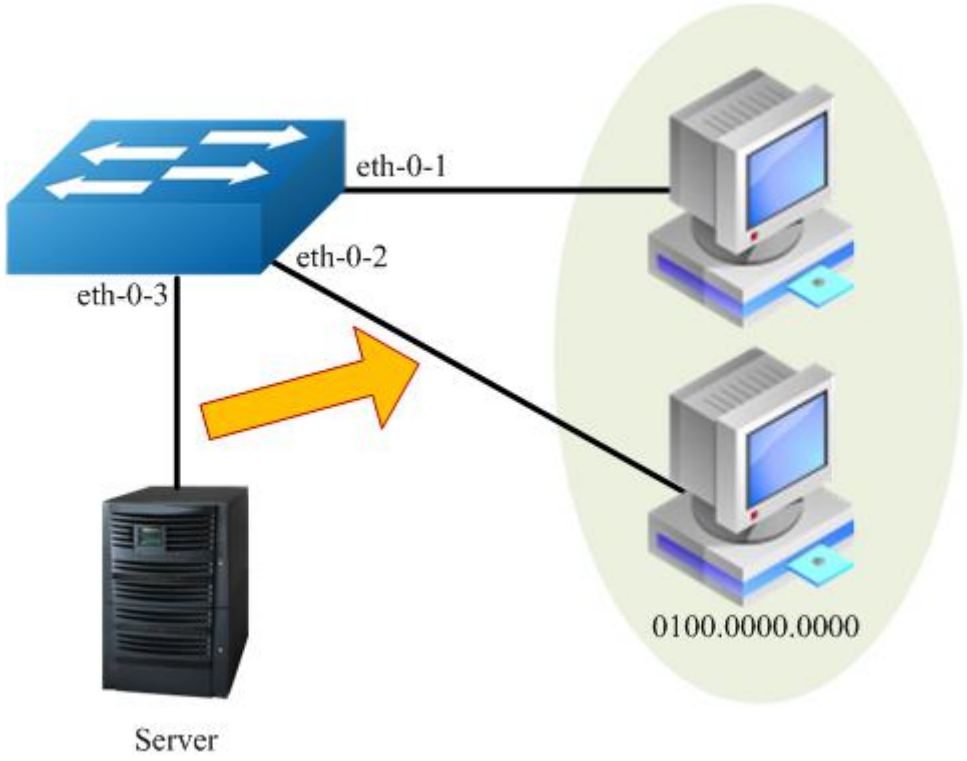


Figure 1-5 Static multicast mac address table

Multicast address can be bound to multi-port.According to the picture, Mac-Da 0100.0000.0000 can forward via eth-0-1 and eth-0-2.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set static multicast mac address table

```
Switch(config)# mac-address-table 0100.0000.0000 forward eth-0-1 vlan 1
Switch(config)# mac-address-table 0100.0000.0000 forward eth-0-2 vlan 1
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the mac address table:

```
Switch# show mac address-table
  Mac Address Table
------------------------------------------
(*) - Security Entry
Vlan   Mac Address      Type      Ports
----   --------------   -------   -------
1    0100.0000.0000   static    eth-0-1
                               eth-0-2
```

Configuring MAC Filter Address

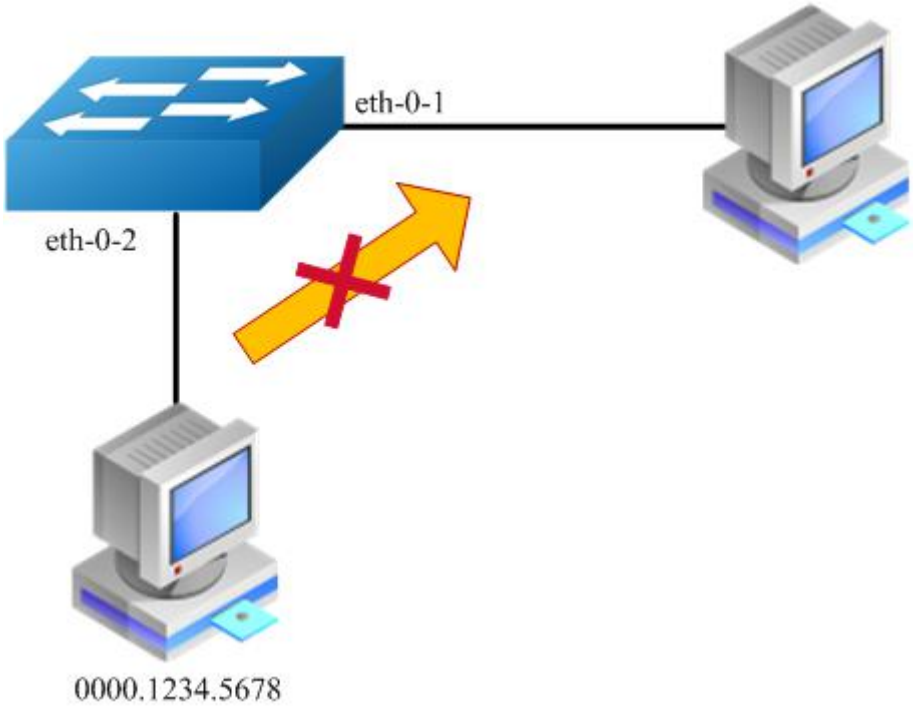


Figure 1-6 mac address filter

MAC filter will discard these frames whose source or destination address is set to discard. The MAC filter has higher priority than MAC address.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Add unicast address to be discarded

```
Switch(config)# mac-address-table 0000.1234.5678 discard
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the mac address filter:

```
Switch# show mac-filter address-table
   MAC Filter Address Table
-----------------------------------
Current count     : 1
Max count         : 128
Left count        : 127
Filter address list :
-----------------------------------
0000.1234.5678
```

3. 4. 3 Application Cases

N/A

3.5 Configuring VLAN

3.5.1 Overview

Function Introduction

VLAN (Virtual Local Area Network) is a switched network that is logically segmented the network into different broadcast domain so that packets are only switched between ports that are designated for the same VLAN. Each VLAN is considered as a logical network, and packets send to stations that do not belong to the same VLAN must be forwarded through a router.

Reference to standard: IEEE 802.1Q

Principle Description

Following is a brief description of terms and concepts used to describe the VLAN:

- VID: VLAN identifier
- LAN: Local Area Network
- VLAN: Virtual LAN
- PVID: Port VID, the untagged or priority-tagged frames will be assigned with this VID

Tagged Frame: Tagged Frame is inserted with 4 Bytes VLAN Tag, show in the picture below:

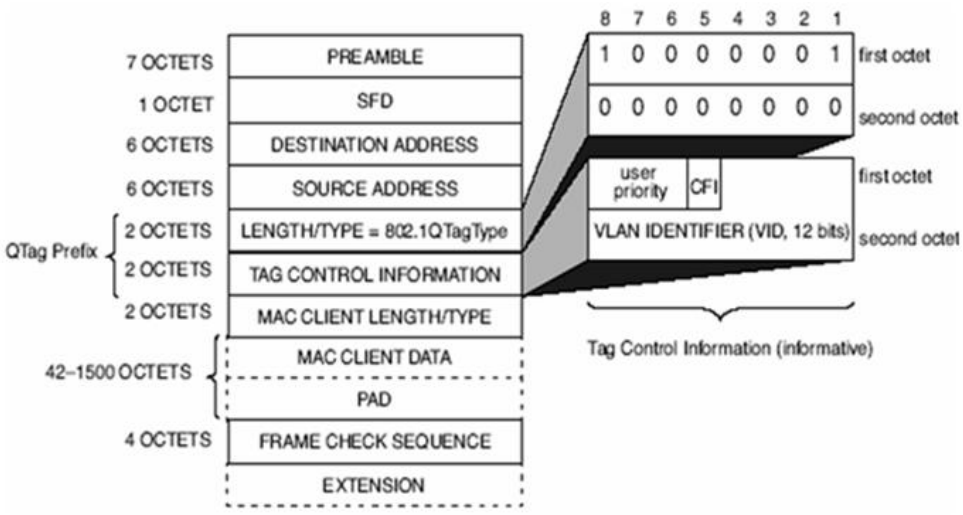


Figure 1-7 Tagged Frame

Trunk Link: Both tagged and untagged frames can be transmitted on this link. Trunk link allow for multiple VLANs to cross this link, show in the picture below:

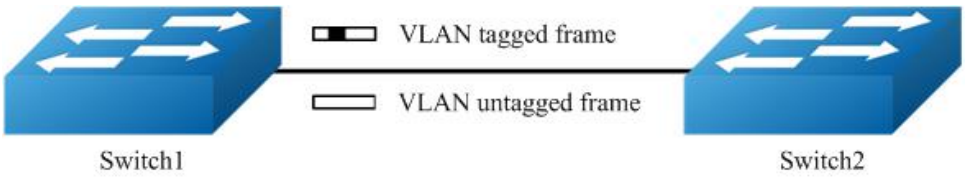


Figure 1-8 Trunk link

Access Link: Only untagged frames can be transmitted on this link. Access link is at the edge of the network, where end stations attach, show in the picture below:

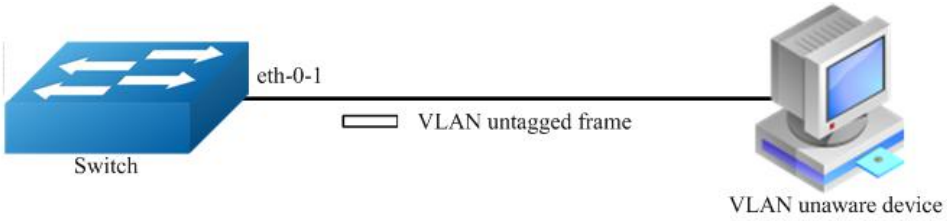


Figure 1-9 Access link

3.5.2 Configuration

Configuring Access Port

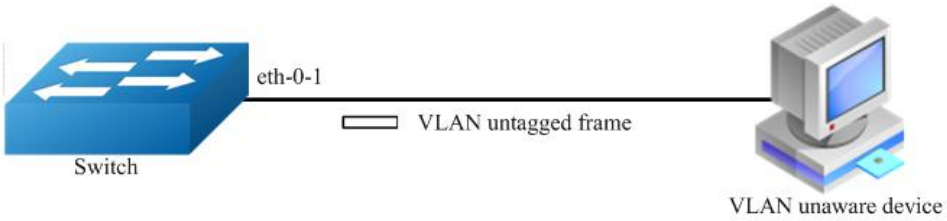


Figure 1-10 Access link

Access port only receives untagged or priority-tagged frames, and transmits untagged frames.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 2
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode, set the switch port mode and bind to the vlan

```
Switch(config)# interface eth-0-1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
```

step 4 Exit the configure mode

```
Switch(config-if)# end
```

step 5 Validation

Use the following command to display the information of the switch port interface:

```
Switch# show interface switchport interface eth-0-1
 Interface name        : eth-0-1
 Switchport mode       : access
 Ingress filter        : enable
 Acceptable frame types  : vlan-untagged only
 Default Vlan          : 2
 Configured Vlans      :   2
```

Use the following command to display the vlan brief information:

```
Switch# show vlan brief
VLAN ID  Name          State  STP ID     Member ports
                                (u)-Untagged, (t)-Tagged
```

```
======= ================ ======= ======= ========================
1    default       ACTIVE  0     eth-0-2(u) eth-0-3(u)
                                 eth-0-4(u) eth-0-5(u)
                                 eth-0-6(u) eth-0-7(u)
                                 eth-0-8(u) eth-0-9(u)
                                 eth-0-10(u) eth-0-11(u)
                                 eth-0-12(u) eth-0-13(u)
                                 eth-0-14(u) eth-0-15(u)
                                 eth-0-16(u) eth-0-17(u)
                                 eth-0-18(u) eth-0-19(u)
                                 eth-0-20(u) eth-0-21(u)
                                 eth-0-22(u) eth-0-23(u)
2    VLAN0002      ACTIVE  0     eth-0-1(u)
```

Configuring Trunk Port

Trunk port receives tagged, untagged, and priority-tagged frames, and transmits both untagged and tagged frames. If trunk port receives an untagged frame, this frame will be assigned to the VLAN of the trunk port's PVID; if a frame send out from the trunk port and the frame's VID is equal to the trunk port's PVID, this frame will be send out without VLAN tag.

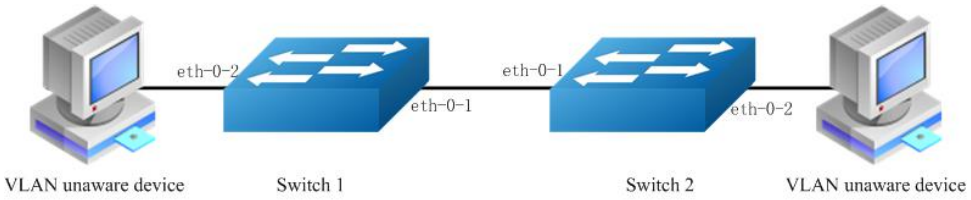


Figure 1-11 Trunk link

Network topology is shown in the picture above. The following configuration steps are same for Switch1 and Switch2.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10,20
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode, set the switch port mode and bind to the vlan

Set eth-0-1's switch port mode as trunk, set native vlan as 10, and allow all VLANs on this interface:

```
Switch(config)# interface eth-0-1
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# switchport trunk native vlan 10
Switch(config-if)# exit
```

Set eth-0-2's switch port mode as access, and bind to vlan 10:

```
Switch(config)# interface eth-0-2
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# exit
```

step 4 Exit the configure mode

```
Switch(config-if)# end
```

step 5 Validation

Use the following command to display the information of the switch port interface:

```
Switch# show interface switchport
 Interface name        : eth-0-1
 Switchport mode       : trunk
 Ingress filter        : enable
 Acceptable frame types  : all
 Default Vlan          : 10
 Configured Vlans      :   1   10   20
 Interface name        : eth-0-2
 Switchport mode       : access
 Ingress filter        : enable
 Acceptable frame types  : vlan-untagged only
 Default Vlan          : 10
 Configured Vlans      :   10
```

Use the following command to display the vlan brief information:

```
Switch# show vlan brief
VLAN ID  Name          State  STP ID     Member ports
                              (u)-Untagged, (t)-Tagged
======= ================ ======= ========  ========================
1     default        ACTIVE  0      eth-0-1(t) eth-0-3(u)
                              eth-0-4(u) eth-0-5(u)
                              eth-0-6(u) eth-0-7(u)
                              eth-0-8(u) eth-0-9(u)
                              eth-0-10(u) eth-0-11(u)
                              eth-0-12(u) eth-0-13(u)
                              eth-0-14(u) eth-0-15(u)
                              eth-0-16(u) eth-0-17(u)
                              eth-0-18(u) eth-0-19(u)
                              eth-0-20(u) eth-0-21(u)
                              eth-0-22(u) eth-0-23(u)
10    VLAN0010       ACTIVE  0      eth-0-1(t) eth-0-2(u)
20    VLAN0020       ACTIVE  0      eth-0-1(t)
```

3.5.3 Application Cases

N/A

3.6 Configuring Voice VLAN

3.6.1 Overview

Function Introduction

With the development of the voice technology, the use of IP Phone/IAD(Integrated Access Device) is becoming more and more widespread in broadband community. Voice and data traffics are usually present in the network at the same time, therfore, voice traffics need higher priority to improve the performance and reduce the packet loss rate.

The traditional method to improve the quality of voice traffic is using ACL to separate the voice packets, and using QoS to ensure the transmit quality.

The voice VLAN feature can identify the voice packets by source mac, which makes the conguration more convenient.

Principle Description

N/A

3.6.2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 2
Switch(config-vlan)# exit
```

step 3 Set the cos of voice vlan (Optional)

The default cos is 5.

```
Switch(config)# voice vlan set cos to 7
```

step 4 Set the voice vlan and create a mac entry for it

```
Switch(config)# voice vlan 2
Switch(config)# voice vlan mac-address 0055.0000.0000 ffff.ff00.0000 description test
```

step 5 Enter the interface configure mode and enable voice vlan

```
Switch(config)# interface eth-0-1
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# voice vlan enable

Switch(config-if)# interface eth-0-2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
```

step 6 Validation

Send packet to eth-0-1, the format of the packet is as below（priority in Vlan tag is 0）:

```
0x0000:  0000 0a02 0001 0055 0000 0011 8100 0002  .....k.....
0x0010:  0800 aadd aadd aadd aadd aadd aadd aadd  ..............
0x0020:  aadd aadd aadd aadd aadd aadd aadd aadd  ..............
0x0030:  aadd aadd aadd aadd aadd aadd            ............
```

Receive packet from eth-0-2, the format of the packet received is as below（priority in Vlan tag is 5）:.

```
0x0000:  0000 0a02 0001 0055 0000 0011 8100 a002  .....k.....
0x0010:  0800 aadd aadd aadd aadd aadd aadd aadd  ..............
0x0020:  aadd aadd aadd aadd aadd aadd aadd aadd  ..............
0x0030:  aadd aadd aadd aadd aadd aadd            ............
```

3.6.3 Application Cases

N/A

3.7 Configuring VLAN Classification

3.7.1 Overview

Function Introduction

VLAN classification is used to define specific rules for directing packets to selected VLANs based on protocol or subnet criteria. Sets of rules can be grouped (one group per interface).

VLAN classification rules have 3 types: mac based, ip based and protocol based. MAC based vlan classification rule will classify packets to specified VLAN according to the source MAC address of incoming packets; IP based vlan classification rule will classify packets according to the source IP address of incoming packets; And protocol based vlan classification rule will classify packets according to the layer3 type of incoming packets. The following layer3 types can be supported: ARP, IP(v4), MPLS, Mcast MPLS, PPPoE, RARP.

Different types of vlan classification rules can be added to same vlan classification group. VLAN classification group can only be applied on switchport. Only one type of vlan classification rules can take effect on one switchport.

Principle Description

N/A

3. 7. 2 Configuration

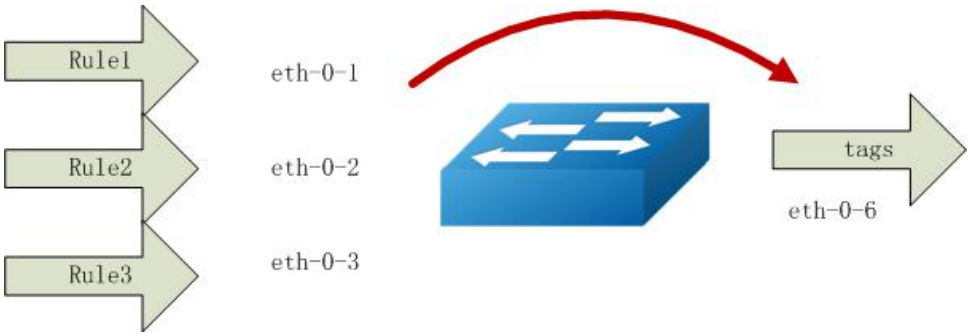


Figure 1-12 vlan classification

In this configuration example, three VLAN classifier rules are created:

Rule 1 is mac based rule, it will classify the packets with MACSA 2222.2222.2222 to vlan 5;

Rule 2 is ip based rule, it will classify the packets sourced from IP adress 1.1.1.1 to vlan 5;

Rule 3 is protocol based rule, it will classify all arp packets to vlan 5.

Add rule 1, rule2, rule3 to group 31. Then apply group 31 to 3 interfaces: eth-0-1, eth-0-2, eth-0-3. These 3 interfaces have different vlan classification type. eth-0-1 is configured to ip based vlan class, this means only ip based rules can take effect on this interface. eth-0-2 is configured to mac based vlan class, this means only mac based rules can take effect on this interface. eth-0-3 is configured to protocol based vlan class, this means only protocol based rules can take effect on this interface.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 5
Switch(config-vlan)# vlan 6
Switch(config-vlan)# exit
```

step 3 Create vlan classifier rule and add the rules to the group

```
Switch(config)# vlan classifier rule 1 mac 2222.2222.2222 vlan 5
Switch(config)# vlan classifier rule 2 ip 1.1.1.1 vlan 5
Switch(config)# vlan classifier rule 3 protocol arp vlan 5

Switch(config)# vlan classifier group 31 add rule 1
Switch(config)# vlan classifier group 31 add rule 2
Switch(config)# vlan classifier group 31 add rule 3
```

step 4 Apply the vlan classifier group on the interface

interface eth-0-1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# switchport access vlan 6
Switch(config-if)# switchport access allowed vlan add 5
Switch(config-if)# vlan classifier activate 31 based ip
Switch(config-if)# exit
```

interface eth-0-2:

```
Switch(config)# interface eth-0-2
Switch(config-if)# switchport access vlan 6
Switch(config-if)# switchport access allowed vlan add 5
Switch(config-if)# vlan classifier activate 31 based mac
Switch(config-if)# exit
```

interface eth-0-3:

```
Switch(config)# interface eth-0-3
Switch(config-if)# switchport access vlan 6
Switch(config-if)# switchport access allowed vlan add 5
Switch(config-if)# vlan classifier activate 31 based protocol
Switch(config-if)# exit
```

interface eth-0-6:

```
Switch(config)# interface eth-0-6
Switch(config)#switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 5
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Verify the VLAN classifier rules:

```
Switch# show vlan classifier rule
vlan classifier rule 1 mac 2222.2222.2222 vlan 5
vlan classifier rule 2 ip 1.1.1.1 vlan 5
vlan classifier rule 3 protocol arp vlan 5
```

Verify the VLAN classifier group:

```
Switch# show vlan classifier group
vlan classifier group 31 add rule 1
vlan classifier group 31 add rule 2
vlan classifier group 31 add rule 3
```

Verify the VLAN classifier interface:

```
Switch# show vlan classifier interface grou
vlan classifier group 31 on interface eth-0-2, based mac
vlan classifier group 31 on interface eth-0-1, based ip
vlan classifier group 31 on interface eth-0-3, based protocol
```

3.7.3 Application Cases

N/A

3.8 Configuring Link Aggregation

3.8.1 Overview

Function Introduction

This chapter contains a sample configuration of Link Aggregation Control Protocol (LACP) . LACP is based on the 802.3ad IEEE specification. It allows bundling of several physical interfaces to form a single logical channel providing enhanced performance and redundancy. The aggregated interface is viewed as a single link to each switch. The spanning tree views it as one interface. When there is a failure in one physical interface, the other interfaces stay up and there is no disruption. This implementation supports the aggregation of maximum 16 physical Ethernet links into a single logical channel. LACP enables our device to manage link aggregation group between other devices that conform to the 802.3ad protocol. By using the LACP, the switch learns the identity of partners supporting LACP and the capabilities of each port. It then dynamically groups ports with same properties into a single logical bundle link.

Reference to standard IEEE 802.3ad.

Principle Description

N/A

3. 8. 2 Configuration

Configure dynamic lacp

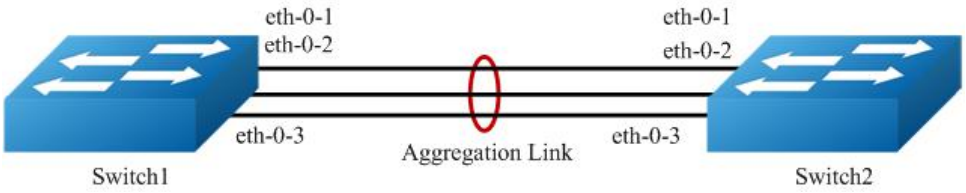


Figure 1-13 Dynamic LACP

The configurations of Switch1 and Switch2 are as below:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the global attributes of LACP

Set the dynamic lacp mode of aggregation groups.

Switch1 configuration:

```
Switch(config)# port-channel 1 lacp-mode dynamic
```

Switch2 configuration:

```
Switch(config)# port-channel 1 lacp-mode dynamic
```

step 3 Enter the interface configure mode and add the interface to the channel group

```
Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config-if)# channel-group 1 mode active
Switch(config-if)# exit
Switch(config)# interface eth-0-2
Switch(config-if)# channel-group 1 mode active
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface eth-0-3
Switch(config-if)# channel-group 1 mode active
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

Use the following command to display the information of the channel-group:

```
Switch# show channel-group summary
port-channel load-balance hash-arithmetic: xor
port-channel load-balance hash-field-select:
   macsa
Flags:  s - suspend         T - standby
     D - down/admin down   B - in Bundle
     R - Layer3          S - Layer2
     w - wait            U - in use
```

```
Mode:  SLB  - static load balance
       DLB  - dynamic load balance
       SHLB - self-healing load balance
       RR   - round robin load balance
Aggregator Name  Mode     Protocol    Ports
----------------+---------+-------------+-----------------------------------------------
agg1(SU)        SLB      LACP(Dynamic)  eth-0-1(B) eth-0-2(B) eth-0-3(B)
```

Use the following command to display the information of the interface agg:

```
Switch1# show interface agg1
Interface agg1
 Interface current state: UP
 Hardware is AGGREGATE, address is cce3.33fc.330b (bia cce3.33fc.330b)
 Bandwidth 3000000 kbits
 Index 1025 , Metric 1 , Encapsulation ARPA
 Speed - 1000Mb/s , Duplex - Full  , Media type is Aggregation
 Link speed type is autonegotiation, Link duplex type is autonegotiation
 Input flow-control is off, output flow-control is off
 The Maximum Frame Size is 1534 bytes
 VRF binding: not bound
 Label switching is disabled
 No virtual circuit configured
 ARP timeout 01:00:00,  ARP retry interval 1s
 5 minute input rate 0 bits/sec, 0 packets/sec
 5 minute output rate 2 bits/sec, 0 packets/sec
  13 packets input, 1184 bytes
  Received 0 unicast, 0 broadcast, 0 multicast
  0 runts, 0 giants, 0 input errors, 0 CRC
  0 frame, 0 overrun, 0 pause input
  0 input packets with dribble condition detected
  20 packets output, 2526 bytes
  Transmitted 0 unicast, 0 broadcast, 0 multicast
  0 underruns, 0 output errors, 0 pause output
```

Configure channel-group

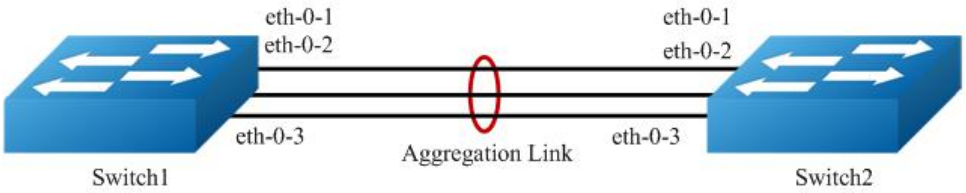


Figure 1-14 LACP

The configurations of Switch1 and Switch2 are as below:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the global attributes of LACP

Set the system priority of this switch. This priority is used for determining the system that is responsible for resolving conflicts in the choice of aggregation groups. A lower numerical value has a higher priority.Set the load balance mode. In this case we choose source MAC address for load balance.

Switch1 configuration:

```
Switch(config)# lacp system-priority 2000
Switch(config)# port-channel load-balance hash-field-select macsa
```

Switch2 configuration:

```
Switch(config)# lacp system-priority 1000
Switch(config)# port-channel load-balance hash-field-select macsa
```

step 3 Enter the interface configure mode and add the interface to the channel group

```
Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config-if)# channel-group 1 mode active
Switch(config-if)# exit
Switch(config)# interface eth-0-2
Switch(config-if)# channel-group 1 mode active
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface eth-0-3
Switch(config-if)# channel-group 1 mode active
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

Use the following command to display the information of the channel-group:

```
Switch# show channel-group summary
port-channel load-balance hash-arithmetic: xor
port-channel load-balance hash-field-select:
    macsa
Flags:  s - suspend         T - standby
       D - down/admin down   B - in Bundle
       R - Layer3          S - Layer2
       w - wait           U - in use
Mode:   SLB  - static load balance
       DLB  - dynamic load balance
       SHLB - self-healing load balance
       RR   - round robin load balance
Aggregator Name  Mode     Protocol    Ports
----------------+----------+-------------+-----------------------------------------------
agg1(SU)        SLB      LACP        eth-0-1(B) eth-0-2(B) eth-0-3(B)
```

Use the following command to display the information of the interface agg:

```
Switch1# show interface agg1
Interface agg1
 Interface current state: UP
 Hardware is AGGREGATE, address is cce3.33fc.330b (bia cce3.33fc.330b)
 Bandwidth 3000000 kbits
 Index 1025 , Metric 1 , Encapsulation ARPA
 Speed - 1000Mb/s , Duplex - Full  , Media type is Aggregation
 Link speed type is autonegotiation, Link duplex type is autonegotiation
 Input flow-control is off, output flow-control is off
 The Maximum Frame Size is 1534 bytes
 VRF binding: not bound
 Label switching is disabled
 No virtual circuit configured
 ARP timeout 01:00:00,  ARP retry interval 1s
 5 minute input rate 0 bits/sec, 0 packets/sec
 5 minute output rate 2 bits/sec, 0 packets/sec
   13 packets input, 1184 bytes
   Received 0 unicast, 0 broadcast, 0 multicast
   0 runts, 0 giants, 0 input errors, 0 CRC
   0 frame, 0 overrun, 0 pause input
   0 input packets with dribble condition detected
   20 packets output, 2526 bytes
   Transmitted 0 unicast, 0 broadcast, 0 multicast
   0 underruns, 0 output errors, 0 pause output
```

Configuring Static-channel-group

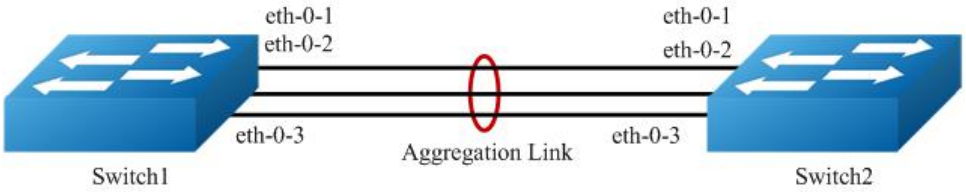


Figure 1-15 Static Agg

The configurations of Switch1 and Switch2 are as below:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and add the interface to the channel group

```
Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config-if)# static-channel-group 1
Switch(config-if)# exit
Switch(config)# interface eth-0-2
Switch(config-if)# static-channel-group 1
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface eth-0-3
Switch(config-if)# static-channel-group 1
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the information of the channel-group:

```
Switch1# show channel-group summary
port-channel load-balance hash-arithmetic: xor
port-channel load-balance hash-field-select:
   macsa
Flags:  s - suspend         T - standby
      D - down/admin down   B - in Bundle
      R - Layer3          S - Layer2
      w - wait            U - in use
Mode:   SLB  - static load balance
      DLB  - dynamic load balance
      SHLB - self-healing load balance
      RR   - round robin load balance
Aggregator Name  Mode     Protocol    Ports
---------------+---------+-----------+-----------------------------------------------
agg1(SU)       SLB      Static       eth-0-1(B) eth-0-2(B) eth-0-3(B)
```

Use the following command to display the information of the interface agg:

```
Switch1# show interface agg 1
Interface agg1
 Interface current state: UP
 Hardware is AGGREGATE, address is cce3.33fc.330b (bia a876.6b2c.9c01)
 Bandwidth 3000000 kbits
 Index 1025 , Metric 1 , Encapsulation ARPA
 Speed - 1000Mb/s , Duplex - Full  , Media type is Aggregation
 Link speed type is autonegotiation, Link duplex type is autonegotiation
 Input flow-control is off, output flow-control is off
```

```
The Maximum Frame Size is 1534 bytes
VRF binding: not bound
Label switching is disabled
No virtual circuit configured
ARP timeout 01:00:00,  ARP retry interval 1s
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 140 bits/sec, 0 packets/sec
  0 packets input, 0 bytes
  Received 0 unicast, 0 broadcast, 0 multicast
  0 runs, 0 giants, 0 input errors, 0 CRC
  0 frame, 0 overrun, 0 pause input
  0 input packets with dribble condition detected
  1080 packets output, 60614 bytes
  Transmitted 0 unicast, 0 broadcast, 0 multicast
0 underruns, 0 output errors, 0 pause output
```

3.8.3 Application Cases

N/A

3.9 Configuring Flow Control

3.9.1 Overview

Function Introduction

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port to stop sending until the condition clears. When the local device detects any congestion at its end, it can notify the link partner or the remote device of the congestion by sending a pause frame. You can use the flow control interface configuration command to set the interface's ability to receive and send pause frames to on, off. The default state for ports is receive off and send off. In auto-negotiation link, local device's flow control ability can be notified to link partner by link up/down.

NOTE: Flow control send/receive on ability only works on full duplex link

Principle Description

N/A

3.9.2 Configuration

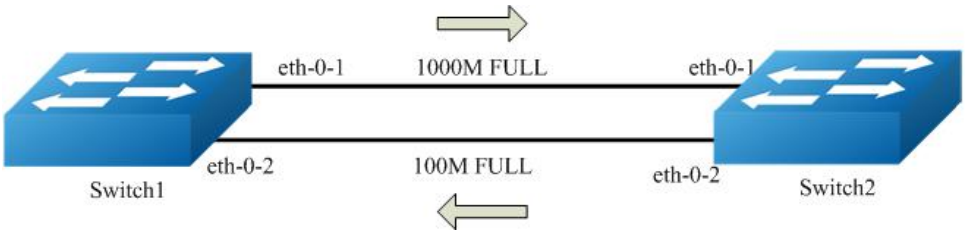


Figure 1-16 Flow control

Configuring Flow Control Send

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and enable flowcontrol send

```
Switch(config)# interface eth-0-1
Switch(config-if)# flowcontrol send on
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

Use the following command to display the information of flow control:

```
Switch# show flowcontrol
Port      Receive FlowControl  Send FlowControl  RxPause     TxPause
          admin    oper        admin    oper
--------- -------- ----------  -------- --------  ----------  ----------
eth-0-1   off      off         on       on        0           0
eth-0-2   off      off         off      off       0           0
eth-0-3   off      off         off      off       0           0
```

Use the following command to display the information of flow control on specified interface:

```
Switch# show flowcontrol eth-0-1
Port      Receive FlowControl  Send FlowControl  RxPause     TxPause
          admin    oper        admin    oper
--------- -------- ----------  -------- --------  ----------  ----------
eth-0-1   off      off         on       on        0           0
```

Configuring Flow Control Receive

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and enable flowcontrol send

```
Switch(config)# interface eth-0-1
Switch1(config-if)# flowcontrol receive on
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

Use the following command to display the information of flow control:

```
Switch1# show flowcontrol
Port      Receive FlowControl  Send FlowControl  RxPause     TxPause
          admin    oper        admin    oper
--------- -------- ----------  -------- --------  ----------  ----------
eth-0-1   on       on          off      off       0           0
eth-0-2   off      off         off      off       0           0
eth-0-3   off      off         off      off       0           0
```

Use the following command to display the information of flow control on specified interface:

```
Switch1# show flowcontrol eth-0-1
Port      Receive FlowControl  Send FlowControl  RxPause     TxPause
          admin    oper        admin    oper
--------- -------- ----------  -------- --------  ----------  ----------
eth-0-1   on       on          off      off       0           0
```

3.9.3 Application Cases

N/A

3.10 Configuring Storm Control

3.10.1 Overview

Function Introduction

Storm control prevents traffic on a LAN from being disrupted by a broadcast, a multicast, or a unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance.

Storm control uses one of these methods to measure traffic activity:

- Bandwidth as a percentage of the total available bandwidth of the port (Level mode).
- Traffic rate in packets per second of the port (PPS mode).

PPS = Packets per second

Principle Description

N/A

3.10.2 Configuration

Configuring Bandwidth Percentage Storm Control

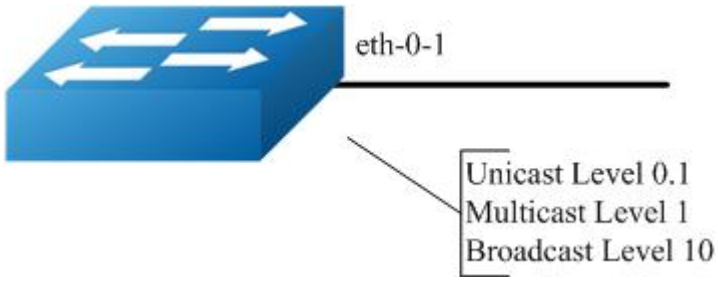


Figure 1-17 Percentage Storm Control

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, and set the storm control level

User can set different level for Unknown unicast/multicast/broad cast packets:

```
Switch(config)# interface eth-0-1
Switch(config-if)# storm-control unicast level 0.1
Switch(config-if)# storm-control multicast level 1
Switch(config-if)# storm-control broadcast level 10
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

```
Switch# show storm-control interface eth-0-1
Port     ucastMode ucastlevel   bcastMode bcastLevel   mcastMode mcastLevel
-------------------------------------------------------------------------------
eth-0-1  Level    0.10       Level    10.00      Level    1.00
```

Configuring Packets per-Second Storm Control

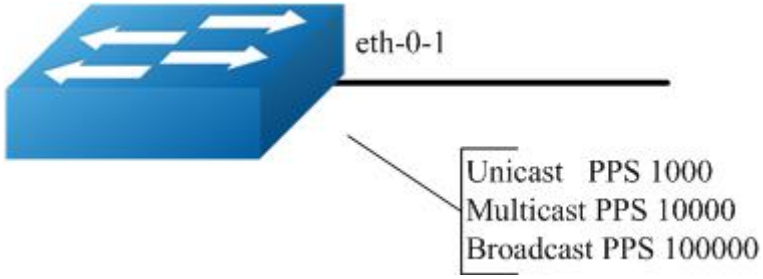


Figure 1-18 PPS Storm Control

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, and set the storm control pps

User can set different pps for Unknown unicast/multicast/broad cast packets:

```
Switch(config)# interface eth-0-1
Switch(config-if)# storm-control unicast pps 1000
Switch(config-if)# storm-control multicast pps 10000
Switch(config-if)# storm-control broadcast pps 100000
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

```
Switch# show storm-control interface eth-0-1
Port     ucastMode ucastlevel   bcastMode bcastLevel   mcastMode mcastLevel
-------------------------------------------------------------------------------
eth-0-1  PPS      1000       PPS      100000      PPS      10000
```

3.10.3 Application Cases

N/A

3.11 Configuring Loopback Detection

3.11.1 Overview

Function Introduction

The loopback in the networks would cause the device continued to send broadcast, multicast and unknow unicast packets. It will waste the resource of network even paralysis the whole network. To detect the loopback in the layer 2 network rapidly and avoid to effect the whole network, system need to provide a detection function to notice the user checking the network connection and configuration, and control the error interface when the network appears loopback.

Loopback Detection can detects whether the interface of device exists loopback. When enable loopback detection on a interface, device will send detection packets from this interface by periodically. If the device receives detection packets sent from the interface, this interface is considered that there is a loop existed and the device can send alarm information to network management system. Administraitors discover loopback problem througt alarm information and resolve the problem to avoid longtime network abnormal. In addition, the device can control the specific interface and configured Trap according the requirement, and disable the interface to quickly reduce the impact in the network of loopback to the minimum.

Principle Description

N/A

3.11.2 Configuration

Enable Loopback Detect

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, and enable Loopback Detect

```
Switch(config)# interface eth-0-1
Switch(config-if)# loopback-detect enable
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

By default, loopback detection is disable. When the interface enable loopback detection, system send the detection packets to detect the loopback. Default detection packets transmission interval is 5 second.

Use the following command to display the loopback detection states:

```
Switch# show loopback-detect
Loopback detection packet interval(second): 5
Loopback detection recovery time(second):  15
Interface      Action         Status
eth-0-2        shutdown       NORMAL
```

Configuring Loopback Detect packet interval

The network is changing all the time, therefor the loopback detection is an continued process. The interface sent loopback detection packets in a certain interval of time, the packets transimission time is loopback detection packets sending period.

The device send the lopback detection packets time interval range is 1 to 300 seconds.The loopback status recover period default is 3 times of the interface send interval.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 set the packet interval of Loopback Detect

```
Switch(config)# loopback-detect packet-interval 10
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the packet interval of Loopback Detect:

```
Switch# show loopback-detect packet-interval
Loopback detection packet interval(second): 10
```

Configuring Loopback Detect action

If a loopback is detected on the interface and loopback is enabled on this interfac, the system can configure an action to send alarm, shutdown the interface, block the interface or other action.

After loopback detection is enabled on an interface, the interface sends loopback detection packets at intervals. When a loopback is detected on the interface, the system performs an action to minimize the impact on the entire network.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, and set the action of Loopback Detect

```
Switch(config)# interface eth-0-1
Switch(config-if)# loopback-detect action shutdown
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the information of Loopback Detect on the interface:

```
Switch# show loopback-detect interface eth-0-1
Interface      Action         Status
eth-0-1        shutdown       NORMAL
```

Configuring specify VLAN Loopback Detection

specify the VLAN IDs of loopback detection packets on an interface After loopback detection is enabled on an interface, system send untagged loopback detection packets by default. It means the device dosen't detect any specify vlan loopback packets. When interface is configured Tagged mode in vlan, the loopback detection packets sent by this interface will be discard on the link, and interface won't receive the loop packets which is sent by itself. So we should specify the VLAN IDs of loopback detection packets on an interface.

After the loopback-detect packet vlan command is executed on an interface, the interface sends an untagged loopback detection packet and the loopback detection packets with the specified VLAN tags. The specified VLANs exist and the interface has been added to the VLANs in tagged mode. If you run the loopback-detect packet vlan command multiple times in the same interface view, multiple VLAN IDs are specified. You can specify a maximum of eight VLAN IDs

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, and set the specify vlan of Loopback Detect

```
Switch(config)# interface eth-0-1
Switch(config-if)# loopback-detect packet vlan 20
```

step 3 Exit the configure mode

```
Switch(config-if)# end
```

step 4 Validation

Use the following command to display the configuration of Loopback Detect:

```
Switch# show running-config interface eth-0-1
Building configuration...
!
interface eth-0-1
 loopback-detect enable
 loopback-detect packet vlan 20
!
```

3.11.3 Application Cases

N/A

3. 12 Configuring MSTP

3. 12. 1 Overview

Function Introduction

The MSTP (Multiple Spanning Tree Algorithm and Protocol (IEEE 802.1Q-2005)) enables multiple VLANs to be mapped to the same spanning-tree instance, thereby reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic and enables load balancing. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths). The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network; this deployment provides the highly-available network required in a service-provider environment. When the switch is in the multiple spanning-tree (MST) modes, the Rapid Spanning Tree Protocol (RSTP), which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Principle Description

N/A

3. 12. 2 Configuration

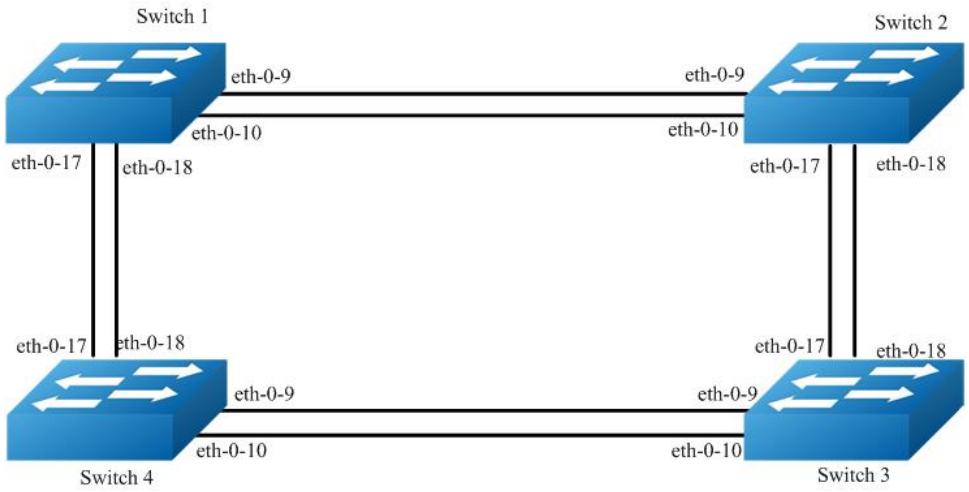


Figure 1-19 MSTP

The configurations of Switch1-Switch4 are as blow. The configurations of these 4 Switches are same if there is no special description.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the mode of STP

```
Switch(config)# spanning-tree mode mstp
```

step 3 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10
Switch(config-vlan)# vlan 20
Switch(config-vlan)# exit
```

step 4 Enter the MSTP configure mode, create region and instance. Bind the vlan to the instance.

```
Switch(config)# spanning-tree mst configuration
Switch(config-mst)# region RegionName
Switch(config-mst)# instance 1 vlan 10
```

```
Switch(config-mst)# instance 2 vlan 20
Switch(config-mst)# exit
```

step 5 Enter the interface configure mode, set the attributes of the interfaces

```
Switch(config)# interface eth-0-9
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-10
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-17
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-18
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 6 Enable STP and set priority for each swicth

Switch1:

```
Switch# configure terminal
Switch(config)# spanning-tree priority 0
Switch(config)# spanning-tree enable
```

Switch2:

```
Switch# configure terminal
Switch(config)# spanning-tree instance 1 priority 0
Switch(config)# spanning-tree enable
```

Switch3:

```
Switch# configure terminal
Switch(config)# spanning-tree instance 2 priority 0
Switch(config)# spanning-tree enable
```

Switch4:

```
Switch# configure terminal
Switch(config)# spanning-tree enable
```

step 7 Exit the configure mode

```
Switch(config)# end
```

step 8 Validation

Use the following command to display the information of MSTP on Switch1:

```
Switch# show spanning-tree mst brief
##### MST0:  Vlans: 1
Multiple spanning tree protocol Enabled
Root ID    Priority      0 (0x0000)
        Address     2225.fa28.c900
        Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID   Priority      0 (0x0000)
```

```
        Address    2225.fa28.c900
        Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
        Aging Time  300 sec
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Designated    Forwarding     20000         128.9      P2p
eth-0-10   Designated    Forwarding     20000         128.10     P2p
eth-0-17   Designated    Forwarding     20000         128.17     P2p
eth-0-18   Designated    Forwarding     20000         128.18     P2p
##### MST1:  Vlans: 10
Root ID    Priority      1 (0x0001)
        Address    9c9a.7d91.9f00
Bridge ID  Priority    32769 (0x8001)
        Address    2225.fa28.c900
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Rootport      Forwarding     20000         128.9       P2p
eth-0-10   Alternate     Discarding     20000         128.10      P2p
eth-0-17   Designated    Forwarding     20000         128.17      P2p
eth-0-18   Designated    Forwarding     20000         128.18      P2p
##### MST2:  Vlans: 20
Root ID    Priority      2 (0x0002)
        Address    304c.275b.b200
Bridge ID  Priority    32770 (0x8002)
        Address    2225.fa28.c900
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Alternate     Discarding     20000         128.9       P2p
eth-0-10   Alternate     Discarding     20000         128.10      P2p
eth-0-17   Rootport      Forwarding     20000         128.17      P2p
eth-0-18   Alternate     Discarding     20000         128.18      P2p
```

Use the following command to display the information of MSTP on Switch2:

```
Switch# show spanning-tree mst brief
##### MST0:  Vlans: 1
Multiple spanning tree protocol Enabled
Root ID    Priority      0 (0x0000)
        Address    2225.fa28.c900
        Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID  Priority    32768 (0x8000)
        Address    9c9a.7d91.9f00
        Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
        Aging Time  300 sec
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Rootport      Forwarding     20000         128.9       P2p
eth-0-10   Alternate     Discarding     20000         128.10      P2p
eth-0-17   Designated    Forwarding     20000         128.17      P2p
eth-0-18   Designated    Forwarding     20000         128.18      P2p
##### MST1:  Vlans: 10
Root ID    Priority      1 (0x0001)
        Address    9c9a.7d91.9f00
Bridge ID  Priority      1 (0x0001)
        Address    9c9a.7d91.9f00
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Designated    Forwarding     20000         128.9       P2p
eth-0-10   Designated    Forwarding     20000         128.10      P2p
eth-0-17   Designated    Forwarding     20000         128.17      P2p
eth-0-18   Designated    Forwarding     20000         128.18      P2p
##### MST2:  Vlans: 20
Root ID    Priority      2 (0x0002)
        Address    304c.275b.b200
Bridge ID  Priority    32770 (0x8002)
        Address    9c9a.7d91.9f00
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Designated    Forwarding     20000         128.9       P2p
eth-0-10   Designated    Forwarding     20000         128.10      P2p
```

```
eth-0-17   Rootport     Forwarding     20000        128.17      P2p
eth-0-18   Alternate    Discarding     20000        128.18      P2p
```

Use the following command to display the information of MSTP on Switch3:

```
Switch# show spanning-tree mst brief
##### MST0:  Vlans: 1
Multiple spanning tree protocol Enabled
Root ID    Priority      0 (0x0000)
           Address     2225.fa28.c900
           Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID  Priority    32768 (0x8000)
           Address     304c.275b.b200
           Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
           Aging Time  300 sec
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Rootport     Forwarding     20000        128.9       P2p
eth-0-10   Alternate    Discarding     20000        128.10      P2p
eth-0-17   Alternate    Discarding     20000        128.17      P2p
eth-0-18   Alternate    Discarding     20000        128.18      P2p
##### MST1:  Vlans: 10
Root ID    Priority      1 (0x0001)
           Address     9c9a.7d91.9f00
Bridge ID  Priority    32769 (0x8001)
           Address     304c.275b.b200
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Designated   Forwarding     20000        128.9       P2p
eth-0-10   Designated   Forwarding     20000        128.10      P2p
eth-0-17   Rootport     Forwarding     20000        128.17      P2p
eth-0-18   Alternate    Discarding     20000        128.18      P2p
##### MST2:  Vlans: 20
Root ID    Priority      2 (0x0002)
           Address     304c.275b.b200
Bridge ID  Priority      2 (0x0002)
           Address     304c.275b.b200
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Designated   Forwarding     20000        128.9       P2p
eth-0-10   Designated   Forwarding     20000        128.10      P2p
eth-0-17   Designated   Forwarding     20000        128.17      P2p
eth-0-18   Designated   Forwarding     20000        128.18      P2p
```

Use the following command to display the information of MSTP on Switch4:

Switch# show spanning-tree mst brief

```
Switch# show spanning-tree mst brief
s##### MST0:  Vlans: 1
无 Multiple spanning tree protocol Enabled
Root ID    Priority      0 (0x0000)
           Address     2225.fa28.c900
           Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID  Priority    32768 (0x8000)
           Address     80a4.be55.6400
           Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
           Aging Time  300 sec
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Designated   Forwarding     20000        128.9       P2p
eth-0-10   Designated   Forwarding     20000        128.10      P2p
eth-0-17   Rootport     Forwarding     20000        128.17      P2p
eth-0-18   Alternate    Discarding     20000        128.18      P2p
##### MST1:  Vlans: 10
Root ID    Priority      1 (0x0001)
           Address     9c9a.7d91.9f00
Bridge ID  Priority    32769 (0x8001)
           Address     80a4.be55.6400
Interface   Role        State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
```

```
eth-0-9    Alternate    Discarding    20000        128.9       P2p
eth-0-10   Alternate    Discarding    20000        128.10      P2p
eth-0-17   Rootport     Forwarding    20000        128.17      P2p
eth-0-18   Alternate    Discarding    20000        128.18      P2p
##### MST2:  Vlans: 20
Root ID    Priority     2 (0x0002)
           Address      304c.275b.b200
Bridge ID  Priority     32770 (0x8002)
           Address      80a4.be55.6400
Interface  Role         State         Cost     Priority.Number    Type
--------------------------------------------------------------------------------
eth-0-9    Rootport     Forwarding    20000        128.9       P2p
eth-0-10   Alternate    Discarding    20000        128.10      P2p
eth-0-17   Designated   Forwarding    20000        128.17      P2p
eth-0-18   Designated   Forwarding    20000        128.18      P2p
```

3.12.3 Application Cases

N/A

3.13 Configuring MLAG

3.13.1 Overview

Function Introduction

High availability data center topologies typically provide redundancy protection at the expense of oversubscription by connecting top-of-rack (TOR) switches and servers to dual aggregation switches. In these topologies, Spanning Tree Protocol prevents network loops by blocking half of the links to the aggregation switches. This reduces the available bandwidth by 50%.

Deploying MLAG removes oversubscription by configuring an MLAG link between two aggregation switches to create a single logical switching instance that utilizes all connections to the switches. Interfaces on both devices participate in a distributed port channel, enabling all active paths to carry data traffic while maintaining the integrity of the Spanning Tree topology.

MLAG provides these benefits:

- Provides higher bandwidth links as network traffic increases.
- Utilizes bandwidth more efficiently with fewer uplinks blocked by STP.
- Connects to other switches and servers by static LAG or LACP without other proprietary protocols.
- Supports normal STP operation to prevent loops.
- Supports active-active Layer-2 redundancy

Principle Description

N/A

NOTE: STP can not be used with MLAG.

3.13.2 Configuration

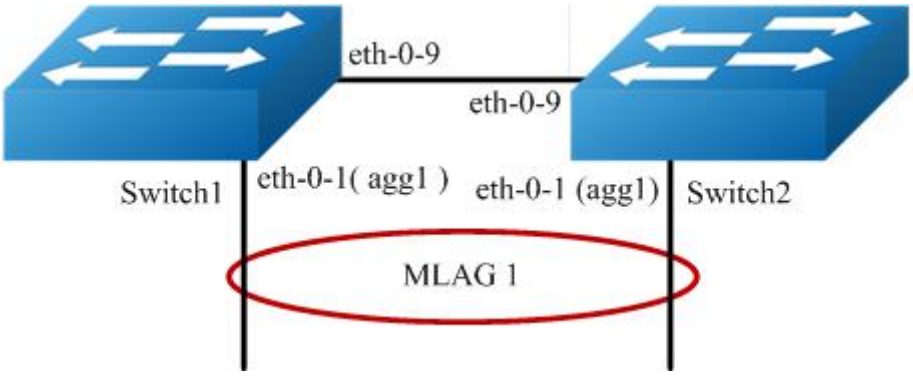


Figure 1-20 MLAG

The configurations of Switch1-Switch2 are as blow. The configurations of these 2 Switches are same if there is no special description.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10,4094
Switch(config-vlan)# exit
```

step 3 Create a static agg

```
Switch(config)# interface eth-0-1
Switch(config-if)# static-channel-group 1
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 4 Set the attributes of the peer link interface

interface eth-0-9 will be set as the peer link interface later

```
Switch(config)# interface eth-0-9
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# spanning-tree port disable
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 5 Bind the agg interface to the mlag

```
Switch(config)# interface agg1
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 10
Switch(config-if)# mlag 1
Switch(config-if)# exit
```

step 6 Set the attributes of the vlan interface

Switch1:

```
Switch(config)# interface vlan4094
Switch(config-if)# ip address 12.1.1.1/24
Switch(config-if)# exit
```

Switch2:

```
Switch(config)# interface vlan4094
Switch(config-if)# ip address 12.1.1.2/24
Switch(config-if)# exit
```

step 7 Enter the mlag configure mode and set the attributes of the mlag

Switch1:

```
Switch(config)# mlag configuration
Switch(config-mlag)# peer-link eth-0-9
Switch(config-mlag)# peer-address 12.1.1.2
Switch(config-mlag)# exit
```

Switch2:

```
Switch(config)# mlag configuration
Switch(config-mlag)# peer-link eth-0-9
Switch(config-mlag)# peer-address 12.1.1.1
Switch(config-mlag)# end
```

step 8 Validation

Use the following command to display the information of mlag on Switch1

```
Switch# show mlag
MLAG configuration:
-----------------
role       : Master
local_sysid : ea90.aecc.cc00
mlag_sysid  : ea90.aecc.cc00
peer-link   : eth-0-9
peer conf   : Yes

Switch# show mlag interface
mlagid  local-if  local-state    remote-state
1      agg1     up           up

Switch# show mlag peer
MLAG neighbor is 12.1.1.2, MLAG version 1
MLAG state = Established, up for 00:13:07
Last read 00:00:48, hold time is 240, keepalive interval is 60 seconds
Received 19 messages,Sent 23 messages
Open     : received 1, sent 2
KAlive   : received 15, sent 16
Fdb sync : received 0, sent 0
Failover : received 0, sent 0
Conf     : received 1, sent 1
STP Total: received 2, sent 4
 Global  : received 2, sent 3
 Packet  : received 0, sent 0
 Instance: received 0, sent 0
 State   : received 0, sent 1
Connections established 1; dropped 0
Local host: 12.1.1.1, Local port: 61000
Foreign host: 12.1.1.2, Foreign port: 46157
remote_sysid: baa7.8606.8b00



Switch# show mac address-table
        Mac Address Table
-------------------------------------------
(*) - Security Entry
Vlan    Mac Address       Type       Ports
----    -----------       --------   -----
```

Use the following command to display the information of mac address table on Switch1

```
Switch# show mlag
MLAG configuration:
-----------------
role       : Slave
local_sysid : baa7.8606.8b00
mlag_sysid  : ea90.aecc.cc00
peer-link   : eth-0-9
peer conf   : Yes

Switch# show mlag interface
mlagid  local-if  local-state    remote-state
1      agg1     up           up

Switch# show mlag peer
MLAG neighbor is 12.1.1.1, MLAG version 1
MLAG state = Established, up for 00:14:29
Last read 00:00:48, hold time is 240, keepalive interval is 60 seconds
Received 23 messages,Sent 21 messages
Open     : received 1, sent 1
KAlive   : received 17, sent 17
Fdb sync : received 0, sent 0
Failover : received 0, sent 0
Conf     : received 1, sent 1
```

```
STP Total: received 4, sent 2
 Global  : received 3, sent 2
 Packet  : received 0, sent 0
 Instance: received 0, sent 0
 State   : received 1, sent 0
Connections established 1; dropped 0
Local host: 12.1.1.2, Local port: 46157
Foreign host: 12.1.1.1, Foreign port: 61000
remote_sysid: ea90.aecc.cc00
```

Use the following command to display the information of mlag on Switch2:

```
Switch# show mac address-table
       Mac Address Table
------------------------------------------
(*) - Security Entry
Vlan    Mac Address      Type       Ports
----    -----------      --------   -----
```

3.13.3 Application Cases

N/A

3.14 Configuring PORT-XCONNECT

3.14.1 Overview

Function Introduction

This feature can forward the packet directly according to the destination-interface configured without looking up any table items and forwarding.

Only physical and aggregate port are currently supported.

Principle Description

N/A

3.14.2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface mode

```
Switch(config)# interface eth-0-1
```

step 3 Set eth-0-1 port-xconnect destination interface

```
Switch(config-if)# port-xconnect destination-interface eth-0-2
Switch(config-if)# end
```

step 4 Display configuration

```
Switch# show running-config
Building configuration...
version 5.3.9.18
!
no service password-encryption
!
!
!
!
!
!
!
```

```
temperature 0 0 0
!
vlan database
!

interface eth-0-1
port-xconnect destination-interface eth-0-2
shutdown
!
interface eth-0-2
shutdown
!
interface eth-0-3
Switch#
```

3. 14. 3 **Application Cases**

N/A

Chapter 4 IP Service Configuration Guide

4.1 Configuring Arp

4.1.1 Overview

Function Introduction

The Address Resolution Protocol (ARP) is a protocol used to dynamically map between Internet host addresses and Ethernet addresses. ARP caches Internet-Ethernet address mappings. When an interface requests a mapping for an address not in the cache, ARP queues the message, which requires the mapping, and broadcasts a message on the associated network requesting the address mapping. If a response is provided, the new mapping is cached and any pending message is transmitted. ARP will queue at most one packet while waiting for a response to a mapping request; only the most recently transmitted packet is kept. If the target host does not respond after 3 requests, the host is considered to be down, allowing an error to be returned to transmission attempts during this interval. If a target host does not send message for a period (normally one hour), the host is considered to be uncertainty, and several requests (normally 6, 3 unicast and 3 broadcast) will send to the host before delete the ARP entry. ARP entries may be added, deleted or changed manually. Manually added entries may be temporary or permanent.

Principle Description

N/A

4.1.2 Configuration

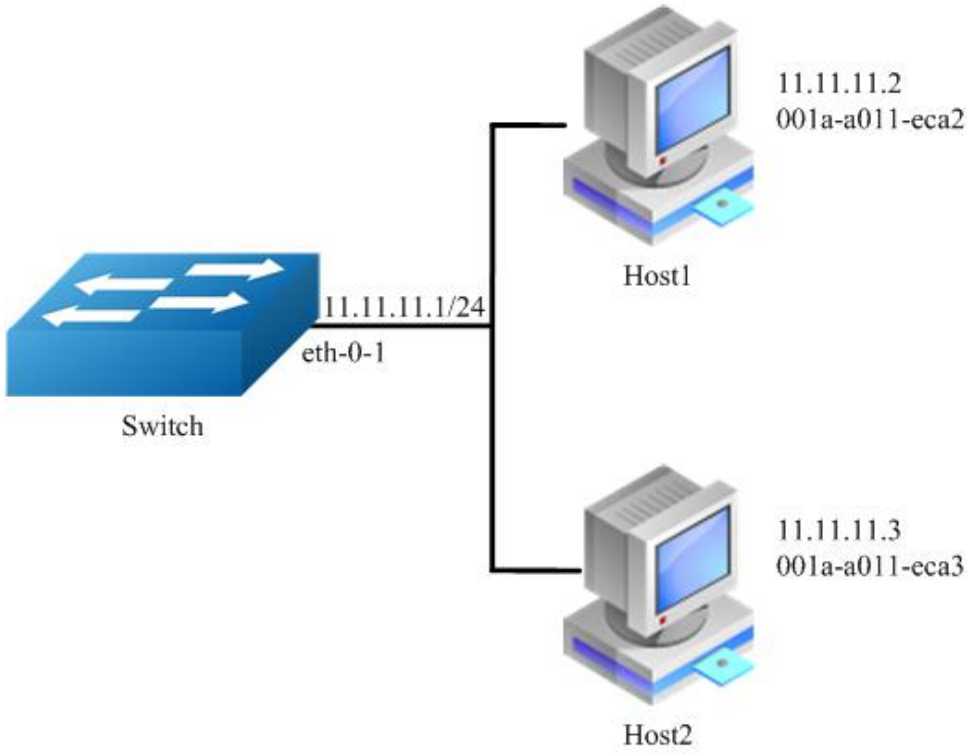


Figure 1-21 arp

In this configuration example, interface eth-0-1 assigned with address 11.11.11.1/24, on subnet 11.11.11.0/24, there are two hosts, and their IP addresses are 11.11.11.2, 11.11.11.3, MAC address are 001a-a011-eca2, 001a-a011-eca3. ARP entry of host 11.11.11.2 is added manually, the entry of host 11.11.11.3 is added dynamically. Time-out period of ARP entries for interface eth-0-1 configure to 20 minutes, ARP request retry delay on interface eth-0-1 configure to 2 seconds.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Configure the layer 3 interface and set the ip address

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 11.11.11.1/24
```

step 3 Configure arp aging timeout value and the arp retry interval value

```
Switch(config-if)# arp timeout 1200
Switch(config-if)# arp retry-interval 2
Switch(config-if)# exit
```

step 4 Add a static arp entry

```
Switch(config)# arp 11.11.11.2 1a.a011.eca2
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Use the following command to display the information of the arp entry:

```
Switch# show ip arp
 Protocol   Address         Age (min)  Hardware Addr   Interface
 Internet   11.11.11.2           -   001a.a011.eca2    eth-0-1

Switch# show ip arp summary
 1 IP ARP entries, with 0 of them incomplete
(Static:0,  Dyamic:0, Interface:1)
 ARP Pkt Received is: 0
 ARP Pkt Send number is: 0
 ARP Pkt Dicard number is: 0
```

Use the following command to display the information of the arp configurations on the interface:

```
Switch# show interface eth-0-1
Interface eth-0-1
  Interface current state: Administratively DOWN
  Hardware is Ethernet, address is 6c02.530c.2300 (bia 6c02.530c.2300)
  Bandwidth 1000000 kbits
  Index 1 , Metric 1 , Encapsulation ARPA
  Speed - Auto , Duplex - Auto  , Media type is 1000BASE_T
  Link speed type is autonegotiation, Link duplex type is autonegotiation
  Input flow-control is off, output flow-control is off
  The Maximum Frame Size is 1534 bytes
  VRF binding: not bound
  Label switching is disabled
  No virtual circuit configured
  VRRP master of :  VRRP is not configured on this interface
  ARP timeout 00:20:00,  ARP retry interval 2s
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes
    Received 0 unicast, 0 broadcast, 0 multicast
    0 runts, 0 giants, 0 input errors, 0 CRC
    0 frame, 0 overrun, 0 pause input
    0 input packets with dribble condition detected
    0 packets output, 0 bytes
    Transmitted 0 unicast, 0 broadcast, 0 multicast
0 underruns, 0 output errors, 0 pause output
```

4.1.3 Application Cases

N/A

4. 2 Configuring Arp proxy

4. 2. 1 Overview

Function Introduction

Proxy ARP, the most common method for learning about other routes, enables an Ethernet host with no routing information to communicate with hosts on other networks or subnets. The host assumes that all hosts are on the same local Ethernet and that they can use ARP to determine their MAC addresses. If a switch receives an ARP request for a host that is not on the same network as the sender, the switch evaluates whether it has the best route to that host. If it does, it sends an ARP reply packet with its own Ethernet MAC address, and the host that sent the request sends the packet to the switch, which forwards it to the intended host. Proxy ARP treats all networks as if they are local and performs ARP requests for every IP address. Proxy ARP can be separated to 2 parts: Proxy ARP and local Proxy ARP. Local Proxy ARP is always used in the topology where the Device is enabled port isolate but still need to do communicating via routing. Internet Control Message Protocol (ICMP) redirects are disabled on interfaces where the local proxy ARP feature is enabled.

Principle Description

N/A

4. 2. 2 Configuration

Configuring ARP Proxy

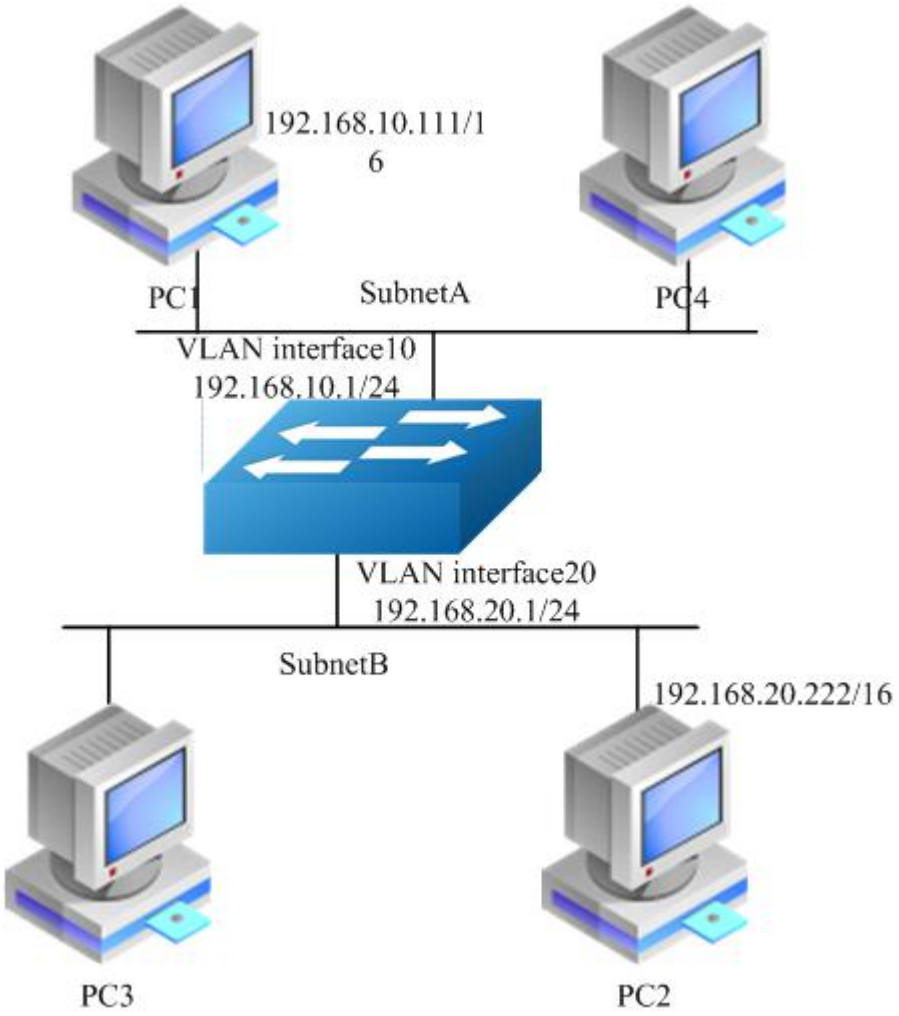


Figure 1-22 arp proxy

As seen in the above topology, PC1 is belonged to VLAN10 and PC2 is belonged to VLAN20. If ARP proxy feature is not enabled, then PC1 and PC2 can not communicate with each other. As following, these steps are shown to enable ARP proxy feature for both VLAN interface 10 and VLAN interface 20.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10,20
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode, set the switch port mode and bind to the vlan

```
Switch(config)# interface eth-0-22
Switch(config-if)# switchport access vlan 10
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-23
Switch(config-if)# switchport access vlan 20
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 4 Create the vlan interface, configure the ip address, and enable arp proxy

```
Switch(config)# interface vlan 10
Switch(config-if)# ip address 192.168.10.1/24
Switch(config-if)# proxy-arp enable
Switch(config-if)# exit

Switch(config)# interface vlan 20
Switch(config-if)# ip address 192.168.20.1/24
Switch(config-if)# proxy-arp enable
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Use the following command to display the information of the arp proxy configuration on the switch:

```
Switch# show ip interface vlan 10
Interface vlan10
 Interface current state: UP
 Internet address(es):
  192.168.10.1/24 broadcast 192.168.10.255
 Joined group address(es):
  224.0.0.1
 The maximum transmit unit is 1500 bytes
 ICMP error messages limited to one every 1000 milliseconds
 ICMP redirects are always sent
 ICMP unreachables are always sent
 ICMP mask replies are always sent
 ARP timeout 01:00:00,  ARP retry interval 1s
 ARP Proxy is enabled, Local ARP Proxy is disabled
 VRRP master of :  VRRP is not configured on this interface

Switch# show ip interface vlan 20
Interface vlan20
 Interface current state: UP
 Internet address(es):
  192.168.20.1/24 broadcast 192.168.20.255
 Joined group address(es):
  224.0.0.1
```

```
The maximum transmit unit is 1500 bytes
ICMP error messages limited to one every 1000 milliseconds
ICMP redirects are always sent
ICMP unreachables are always sent
ICMP mask replies are always sent
ARP timeout 01:00:00,  ARP retry interval 1s
ARP Proxy is enabled, Local ARP Proxy is disabled
VRRP master of :  VRRP is not configured on this interface
```

Use the following command to display the information of the arp entry on the switch:

```
Switch# show ip arp
 Protocol   Address        Age (min)  Hardware Addr   Interface
 Internet   192.168.10.1          -   7cc3.11f1.aa00  vlan10
 Internet   192.168.10.111        5   0cf9.11b6.6e2e  vlan10
 Internet   192.168.20.1          -   7cc3.11f1.aa00  vlan20
 Internet   192.168.20.222        6   5a94.031f.2357  vlan20
```

Use the following command to display the information on PC1:

```
[Host:~]$ ifconfig eth0
eth0     Link encap:Ethernet  HWaddr 0C:F9:11:B6:6E:2E
         inet addr:192.168.10.111  Bcast:192.168.255.255  Mask:255.255.0.0
         UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
         RX packets:11 errors:0 dropped:0 overruns:0 frame:0
         TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:588 (588.0 b)  TX bytes:700 (700.0 b)
         Interrupt:5

[Host:~]$ arp –a
? (192.168.20.222) at 7c:c3:11:f1:aa:00 [ether] on eth0
[Host: ~]$ route -v
Kernel IP routing table
Destination    Gateway       Genmask       Flags Metric Ref    Use Iface
192.168.0.0    *             255.255.0.0    U     0      0        0 eth0

[Host:~]$ ping 192.168.20.222
PING 192.168.20.222 (192.168.20.222) 56(84) bytes of data.
64 bytes from 192.168.20.222: icmp_seq=0 ttl=63 time=189 ms
64 bytes from 192.168.20.222: icmp_seq=1 ttl=63 time=65.2 ms
--- 192.168.20.222 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 65.209/127.226/189.244/62.018 ms, pipe 2
```

Use the following command to display the information on PC2:

```
[Host:~]$ ifconfig eth0
eth0     Link encap:Ethernet  HWaddr 5A:94:03:1F:23:57
         inet addr:192.168.20.222  Bcast:192.168.255.255  Mask:255.255.0.0
         UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
         RX packets:14 errors:0 dropped:0 overruns:0 frame:0
         TX packets:17 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:784 (784.0 b)  TX bytes:1174 (1.1 KiB)
         Interrupt:5

[Host:~]$ arp -a
? (192.168.10.111) at 7c:c3:11:f1:aa:00 [ether] on eth0

[Host: ~]$ route -v
Kernel IP routing table
Destination    Gateway       Genmask       Flags Metric Ref    Use Iface
192.168.0.0    *             255.255.0.0    U     0      0        0 eth0

[Host: ~]$ ping 192.168.10.111
PING 192.168.10.111 (192.168.10.111) 56(84) bytes of data.
64 bytes from 192.168.10.111: icmp_seq=0 ttl=63 time=53.8 ms
64 bytes from 192.168.10.111: icmp_seq=1 ttl=63 time=65.8 ms
--- 192.168.10.111 ping statistics ---
```

```
2 packets transmitted, 2 received, 0% packet loss, time 1007ms
rtt min/avg/max/mdev = 53.832/59.842/65.852/6.010 ms, pipe 2
```

Configuring Local ARP Proxy

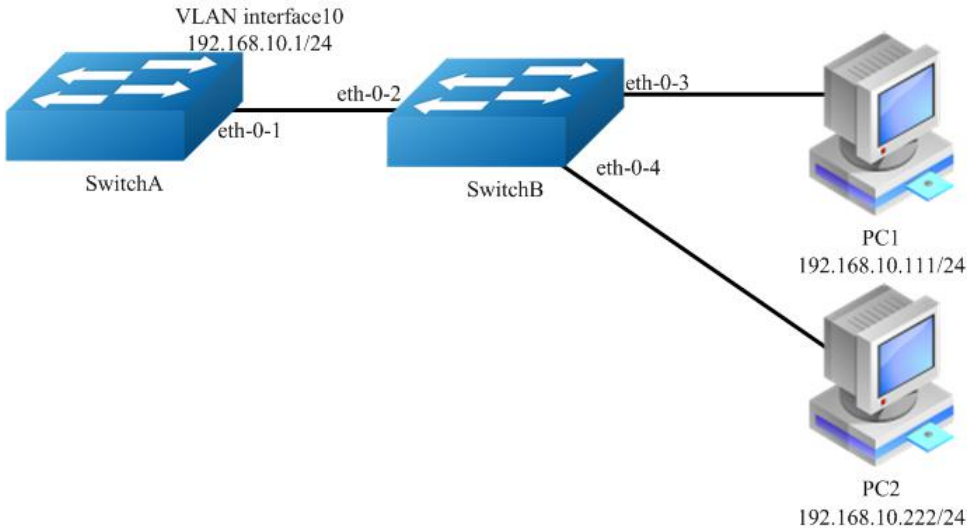


Figure 1-23 local arp proxy

As the above topology, eth-0-2, eth-0-3 and eth-0-4 are belonging to VLAN 10. eth-0-3 and eth-0-4 are both in port isolate group 1, and eth-0-2 is in port isolate group 3, so packets received in eth-0-3 can not flood to eth-0-4, but packets received in eth-0-2 can flood to both eth-0-3 and eth-0-4. PC1 is connecting with port eth-0-3 and PC2 is connecting with port eth-0-4.Configure as the following step for communicating with PC1 and PC2.

The configurations of switch A and switch B are same if there is no special description.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode, set the switch port mode and bind to the vlan

Switch A configuration:

```
Switch(config)# interface eth-0-1
Switch(config-if)# switchport access vlan 10
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Switch B configuration:

```
Switch(config)# interface range eth-0-2 - 4
Switch(config-if-range# switchport access vlan 10
Switch(config-if-range# no shutdown
Switch(config-if-range# exit
```

step 4 Create the vlan interface, configure the ip address, and enable local arp proxy

Switch A configuration:

```
Switch(config)# interface vlan 10
Switch(config-if)# ip address 192.168.10.1/24
```

```
Switch(config-if)# local-proxy-arp enable
Switch(config-if)# exit
```

step 5 Configuring port isolation(optional)

Switch B configuration:

After configuring port isolation as blow, eth-0-3 and eth-0-4 on swichB are isolated in layer 2 network.

```
Switch(config)# port-isolate mode l2
Switch(config)# interface eth-0-3 - 4
Switch(config-if-range# port-isolate group 1
Switch(config-if-range# exit

Switch(config)# interface eth-0-2
Switch(config-if)# port-isolate group 3
Switch(config-if)# exit
```

step 6 Validation

Use the following command to display the information of the arp entry on switchA:

```
Switch# show ip arp
 Protocol   Address         Age (min)  Hardware Addr   Interface
 Internet   192.168.10.1          -   eeb4.2a8d.6c00  vlan10
 Internet   192.168.10.111        0   34b0.b279.5f67  vlan10
 Internet   192.168.10.222        0   2a65.9618.57fa  vlan10
```

Use the following command to display the information of the arp configurations on the interface of switchA:

```
Switch# show ip interface vlan 10
Interface vlan10
 Interface current state: UP
 Internet address(es):
  192.168.10.1/24 broadcast 192.168.10.255
 Joined group address(es):
  224.0.0.1
 The maximum transmit unit is 1500 bytes
 ICMP error messages limited to one every 1000 milliseconds
 ICMP redirects are never sent
 ICMP unreachables are always sent
 ICMP mask replies are always sent
 ARP timeout 01:00:00,  ARP retry interval 1s
 ARP Proxy is disabled, Local ARP Proxy is enabled
 VRRP master of :  VRRP is not configured on this interface
```

Use the following command to display the information on PC1:

```
[Host: ~]$ ifconfig eth0
eth0     Link encap:Ethernet  HWaddr 34:B0:B2:79:5F:67
         inet addr:192.168.10.111  Bcast:192.168.10.255  Mask:255.255.255.0
         UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
         RX packets:22 errors:0 dropped:0 overruns:0 frame:0
         TX packets:28 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:1344 (1.3 KiB)  TX bytes:2240 (2.1 KiB)
         Interrupt:5

[Host: ~]$ arp -a
? (192.168.10.222) at ee:b4:2a:8d:6c:00 [ether] on eth0

[Host: ~]$ ping 192.168.10.222
PING 192.168.10.222 (192.168.10.222) 56(84) bytes of data.
64 bytes from 192.168.10.222: icmp_seq=0 ttl=63 time=131 ms
64 bytes from 192.168.10.222: icmp_seq=1 ttl=63 time=159 ms
--- 192.168.10.222 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 131.078/145.266/159.454/14.188 ms, pipe 2
```

Use the following command to display the information on PC2:

```
[Host:~]$ ifconfig eth0
eth0     Link encap:Ethernet  HWaddr 2A:65:96:18:57:FA
         inet addr:192.168.10.222  Bcast:192.168.10.255  Mask:255.255.255.0
         UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
         RX packets:19 errors:0 dropped:0 overruns:0 frame:0
         TX packets:20 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:1148 (1.1 KiB)  TX bytes:1524 (1.4 KiB)
         Interrupt:5

[Host:~]$ arp -a
? (192.168.10.111) at ee:b4:2a:8d:6c:00 [ether] on eth0

[Host: ~]$ ping 192.168.10.111
PING 192.168.10.111 (192.168.10.111) 56(84) bytes of data.
64 bytes from 192.168.10.111: icmp_seq=0 ttl=63 time=198 ms
64 bytes from 192.168.10.111: icmp_seq=1 ttl=63 time=140 ms
64 bytes from 192.168.10.111: icmp_seq=2 ttl=63 time=146 ms
--- 192.168.10.111 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2008ms
rtt min/avg/max/mdev = 140.196/161.959/198.912/26.267 ms, pipe 2
```

4.2.3 Application Cases

N/A

4.3 Configuring DHCP Client

4.3.1 Overview

Function Introduction

Dynamic Host Configuration Protocol(DHCP) client can acquire IP address and configuration dynamically from DHCP server by DHCP. If client and server is on the same physical subnet, client can communicate with server directly, otherwise they need DHCP relay agent which is used to forward DHCP messages. DHCP client can request IP address from DHCP server by broadcasting DHCP messages. After received IP address and lease correspond to it, client will configure itself and set the expired time. When half past the lease, client will sent DHCP messages for a new lease to use the IP address continually. If it success, DHCP client will renew the lease. DHCP client can send option request to server, which may be one or several of router, static-route, classless-static-route, classless-static-route-ms, tftp-server-address, dns-nameserver , domain-name, netbios-nameserver and vendor-specific. By default, options include router, static-route, classless-static-route, classless-static-route-ms, tftp-server-address will be requested from server. We can cancel one or several of these option requests by command.

Principle Description

N/A

4.3.2 Configuration

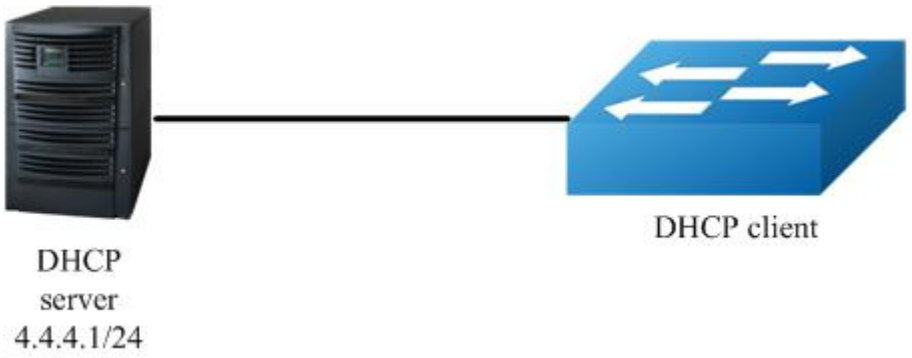


Figure 1-24 dhcp client

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
```

step 3 disable static-route and enable DHCP client

```
Switch(config-if)# no dhcp client request static-route
Switch(config-if)# ip address dhcp
```

step 4 Exit the configure mode

```
Switch(config-if)# end
```

step 5 Validation

Check interface configuration:

```
Switch# show running-config interface eth-0-1
Building configuration...
!
interface eth-0-1
 no switchport
 ip address dhcp
 no dhcp client request static-route
!
```

Check all DHCP client status:

```
Switch# show dhcp client verbose
DHCP client informations:
============================================================
eth-0-1 DHCP client information:
  Current state: BOUND
  Allocated IP: 4.4.4.199 255.255.255.0
  Lease/renewal/rebinding: 1187/517/1037 seconds
  Lease from 2011-11-18 05:59:59 to 2011-11-18 06:19:59
  Will Renewal in 0 days 0 hours 8 minutes 37 seconds
  DHCP server: 4.4.4.1
  Transaction ID: 0x68857f54
  Client ID: switch-7e39.3457.b700-eth-0-1
```

Show DHCP client statistics:

```
Switch# show dhcp client statistics
DHCP client packet statistics:
============================================================
DHCP OFFERS    received: 1
DHCP ACKs      received: 2
DHCP NAKs      received: 0
DHCP Others    received: 0
DHCP DISCOVER      sent: 1
DHCP DECLINE       sent: 0
DHCP RELEASE       sent: 0
DHCP REQUEST       sent: 2
DHCP packet send failed: 0
```

4.3.3 Application Cases

N/A

4. 4 Configuring DHCP Relay

4. 4. 1 Overview

Function Introduction

DHCP relay agent is any host that forwards DHCP packets between clients and servers. Relay agents are used to forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is distinct from the normal forwarding of an IP router, where IP datagram are switched between networks somewhat transparently. By contrast, relay agents receive DHCP messages and then generate a new DHCP message to send out on another interface. The relay agent sets the gateway address (girder field of the DHCP packet) and, if configured, adds the relay agent information option (option82) in the packet and forwards it to the DHCP server. The reply from the server is forwarded back to the client after removing option 82.

Principle Description

N/A

4. 4. 2 Configuration

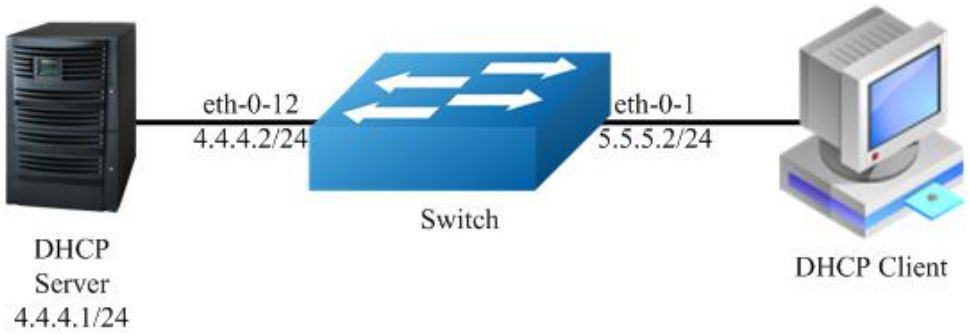


Figure 1-25 DHCP relay

This figure is the networking topology for testing DHCP relay functions. We need two Linux boxes and one Switch to construct the test bed.

Computer A is used as DHCP server.

Computer B is used as DHCP client.

Switch is used as DHCP relay agent.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, set the attributes and ip address

```
Switch(config)# interface eth-0-12
Switch(config-if)# no switchport
Switch(config-if)# ip address 4.4.4.2/24
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 5.5.5.2/24
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 3 Create a dhcp server

```
Switch(config)# dhcp-server 1 4.4.4.1
```

step 4 Enable DHCP server and option82 for the interface

```
Switch(config)# interface eth-0-1
Switch(config-if)# dhcp relay information trusted
Switch(config-if)# dhcp-server 1
Switch(config-if)# exit
```

step 5 Enable DHCP server and DHCP relay globally

```
Switch(config)# service dhcp enable
Switch(config)# dhcp relay
```

step 6 Validation

Check the interface configuration

```
Switch# show running-config interface eth-0-12
!
interface eth-0-12
 no switchport
 ip address 4.4.4.2/24
!
Switch# show running-config interface eth-0-1
!
interface eth-0-1
 no switchport
 dhcp relay information trusted
 dhcp-server 1
 ip address 5.5.5.2/24
!
```

Check the dhcp service status

```
Switch# show services
Networking services configuration:
Service Name        Status
============================================================
dhcp                enable
```

Check the dhcp server group configuration

```
Switch# show dhcp-server
DHCP server group information:
============================================================
group 1 ip address list:
[1] 4.4.4.1
```

Check the dhcp relay statistics

```
Switch# show dhcp relay statistics
DHCP relay packet statistics:
============================================================
Client relayed packets:  20
Server relayed packets:  20
Client error packets:    20
Server error packets:    0
Bogus GIADDR drops:      0
Bad circuit ID packets:  0
Corrupted agent options:  0
Missing agent options:   0
Missing circuit IDs:     0
```

Check your computer ip address from DHCP server

```
Ipconfig /all
Dhcp Enabled. . . . . . . . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IP Address. . . . . . . . . . . . : 5.5.5.1
Subnet Mask . . . . . . . . . . . : 255.255.255.0
```

```
Default Gateway . . . . . : 5.5.5.2
DHCP Server . . . . . : 4.4.4.1
DNS Servers . . . . . : 4.4.4.1
```

4.4.3 Application Cases

N/A

4.5 Configuring DHCP server

4.5.1 Overview

Function Introduction

A DHCP server is an Internet host that returns configuration parameters to DHCP clients 。 DHCP server can provide IP address and network configuration for DHCP client by DHCP. For provide DHCP service, DHCP server need to be configured first. For example, IP address pool need be create , default gateway should be set in a pool, and some network parameters for DHCP client should be set before DHCP working. After DHCP server start to work, it will find a valid IP address from pool for DHCP client when receiving client's request. Meantime it also send network configuration parameters to client. The IP address assigned by DHCP server have a period of validity(lease), so DHCP client need to renew its lease before the lease expired for reserving current IP address by sending DHCP REQUEST message.

If DHCP server was in the same subnet with client,it can normal work after connect to subnet. Otherwise DHCP relay was needed for server providing DHCP service ,which can help to forward DHCP message between server and client.

Main options supported by DHCP server include bootfile-name, dns-server, domain-name, gateway, netbios-name-server, netbios-node-type, tftp-server-address. Besides these, some raw options were also be supported ,which were set with option code.

Principle Description

N/A

4.5.2 Configuration

Configuring DHCP server

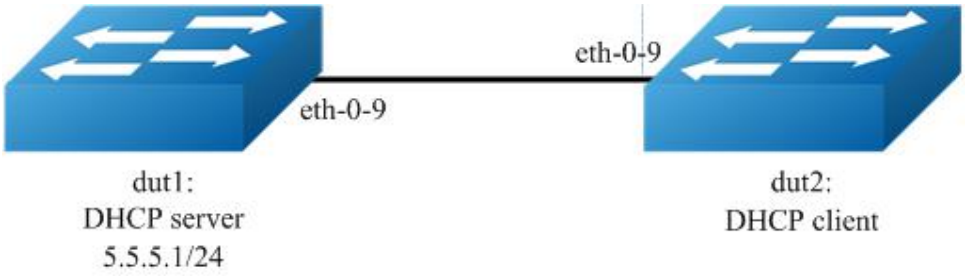


Figure 1-26 DHCP server

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable DHCP server globally, configure the ip address pool

Configure on DUT1:

```
Switch(config)#service dhcp enable
Switch(config)#dhcp server
Switch(config)#dhcp pool pool5
Switch(dhcp-config)#network 5.5.5.0/24
Switch(dhcp-config)#gateway 5.5.5.1
Switch(dhcp-config)#exit
```

step 3 Enter the interface configure mode, set the attributes and ip address

Configure on DUT1:

```
Switch(config)#interface eth-0-9
Switch (config-if)#no switchport
Switch (config-if)# no shutdown
Switch (config-if)# ip address 5.5.5.1/24
Switch (config-if)# dhcp server enable
Switch (config-if)#exit
```

Configure on DUT2:

```
Switch#configure terminal
Switch(config)#interface eth-0-9
Switch (config-if)#no switchport
Switch (config-if)# no shutdown
Switch (config-if)# ip address dhcp
Switch (config-if)#exit
```

step 4 Validation

Check DHCP Server(dut1) configuration:

```
Switch# show running-config
!
service dhcp enable
!
interface eth-0-9
 no switchport
 dhcp server enable
 ip address 5.5.5.1/24!
!
dhcp server
dhcp pool pool5
 network 5.5.5.0/24
 gateway 5.5.5.1
```

Check DHCP client status on DHCP Server(dut1):

```
Switch# show dhcp client verbose
DHCP client informations:
============================================================
eth-0-9 DHCP client information:
  Current state: BOUND
  Allocated IP: 5.5.5.2 255.255.255.0
  Lease/renewal/rebinding: 1194/546/1044 seconds
  Lease from 2012-02-04 07:40:12 to 2012-02-04 08:00:12
  Will Renewal in 0 days 0 hours 9 minutes 6 seconds
  DHCP server: 5.5.5.1
  Transaction ID: 0x45b0b27b
  Default router: 5.5.5.1
  Classless static route:
    Destination: 5.5.4.0, mask: 255.255.255.0, Nexthop: 5.5.5.1
  TFTP server addresses: 5.5.5.3
Client ID: switch-6e6e.361f.8400-eth-0-9
```

Check DHCP server statistics on DHCP Server(dut1):

```
Switch# show dhcp server statistics
DHCP server packet statistics:
============================================================
Message Received:
BOOTREQUEST:  0
DHCPDISCOVER:  1
DHCPREQUEST:  1
DHCPDECLINE:  0
DHCPRELEASE:  0
DHCPINFORM:  0
Message Sent:
```

```
BOOTREPLY: 0
DHCPOFFER: 1
DHCPACK: 1
DHCPNAK: 0
```

Check DHCP server addresses and interfaces on DHCP Server(dut1):

```
Switch# show dhcp server binding all
 IP address     Client-ID/          Lease expiration          Type
                Hardware address
 5.5.5.2        6e:6e:36:1f:84:00   Sat 2012.02.04 08:00:12   Dynamic
Switch# show dhcp server interfaces
List of DHCP server enabled interface(s):
DHCP server service status: enabled
Interface Name
============================================================
eth-0-9
```

Configuring DHCP server with relay

Figure 1-27 DHCP relay

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable DHCP server globally, configure the ip address pool and DHCP relay

Configure on DUT1:

```
Switch(config)#service dhcp enable
Switch(config)#dhcp server
Switch(dhcp-config)#dhcp pool pool4
Switch(dhcp-config)#network 4.4.4.0/24
Switch(dhcp-config)#gateway 4.4.4.1
Switch(dhcp-config)#exit
```

Configure on DUT2:

```
Switch(config)#service dhcp enable
Switch(config)#dhcp relay
Switch(config)#dhcp-server 1 5.5.5.1
```

step 2 Add a ip route

Configure on DUT1:

```
Switch(config)#ip route 4.4.4.0/24 5.5.5.2
```

step 4 Enter the interface configure mode, set the attributes and ip address

Configure on DUT1:

```
Switch(config)#interface eth-0-9
Switch (config-if)#no switchport
Switch (config-if)# no shutdown
Switch (config-if)# ip address 5.5.5.1/24
Switch (config-if)# dhcp server enable
Switch (config-if)#exit
```

Configure on DUT2:

```
Switch(config)#interface eth-0-17
Switch (config-if)#no switchport
Switch (config-if)# no shutdown
Switch (config-if)# ip address 4.4.4.1/24
Switch (config-if)# dhcp-server 1

Switch (config-if)#interface eth-0-9
Switch (config-if)#no switchport
Switch (config-if)# no shutdown
Switch (config-if)# ip address 5.5.5.2/24
Switch (config-if)#exit
```

Configure on DUT3:

```
Switch(config)#interface eth-0-17
Switch (config-if)#no switchport
Switch (config-if)# no shutdown
Switch (config-if)# ip address dhcp
Switch (config-if)#exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Check DHCP Server(dut1) configuration:

```
Switch# show running-config
!
service dhcp enable
!
interface eth-0-9
 no switchport
 dhcp server enable
 ip address 5.5.5.1/24!
!
ip route 4.4.4.0/24 5.5.5.2
!
dhcp server
dhcp pool pool4
 network 4.4.4.0/24
 gateway 4.4.4.1
```

Check DHCP client status on DHCP Server(dut1):

```
Switch# show dhcp client verbose
DHCP client informations:
============================================================
eth-0-17 DHCP client information:
  Current state: BOUND
  Allocated IP: 4.4.4.5 255.255.255.0
  Lease/renewal/rebinding: 1199/517/1049 seconds
  Lease from 2012-02-06 05:23:09 to 2012-02-06 05:43:09
  Will Renewal in 0 days 0 hours 8 minutes 37 seconds
  DHCP server: 5.5.5.1
  Transaction ID: 0x192a4f7d
  Default router: 4.4.4.1
  Classless static route:
    Destination: 5.5.4.0, mask: 255.255.255.0, Nexthop: 4.4.4.1
  TFTP server addresses: 5.5.5.3
  Client ID: switch-3c9a.b29a.ba00-eth-0-17
```

Check DHCP server statistics on DHCP Server(dut1):

```
Switch# show dhcp server statistics
DHCP server packet statistics:
```

```
============================================================
Message Received:
BOOTREQUEST: 0
DHCPDISCOVER: 1
DHCPREQUEST: 1
DHCPDECLINE: 0
DHCPRELEASE: 0
DHCPINFORM: 0
Message Sent:
BOOTREPLY: 0
DHCPOFFER: 1
DHCPACK: 1
DHCPNAK: 0
```

Check DHCP server addresses and interfaces on DHCP Server(dut1):

```
Switch# show dhcp server binding all
IP address     Client-ID/          Lease expiration          Type
               Hardware address
 4.4.4.5       3c:9a:b2:9a:ba:00   Mon 2012.02.06 05:43:09   Dynamic
Switch# show dhcp server interfaces
List of DHCP server enabled interface(s):
DHCP server service status: enabled
Interface Name
============================================================
eth-0-9
```

4.5.3 Application Cases

N/A

4.6 Configuring DNS

4.6.1 Overview

Function Introduction

The DNS protocol controls the Domain Name System (DNS), a distributed database with which you can map hostnames to IP addresses. When you configure DNS on your switch, you can substitute the hostname for the IP address with all IP commands, such as ping, telnet, connect, and related Telnet support operations. IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. To keep track of domain names, IP has defined the concept of a domain name server, which holds a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the hostnames, specify the name server that is present on your network, and enable the DNS.

Principle Description

N/A

4.6.2 Configuration

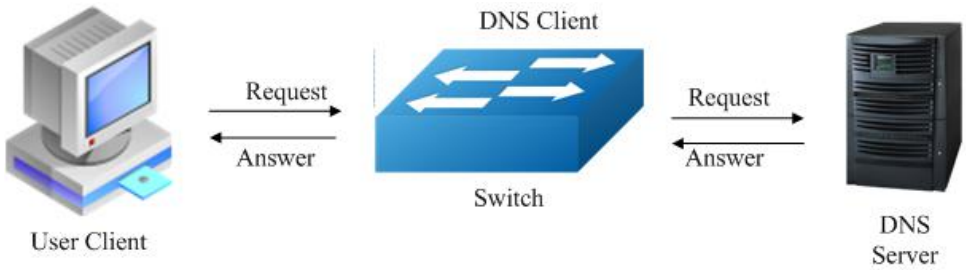


Figure 1-28 DNS

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the dns domain name and dns server address

```
Switch(config)#dns domain server1
Switch(config)#dns server 202.100.10.20
```

step 3 Set static hostname-to-address mappings (optional)

```
Switch(config)# ip host www.example1.com 192.0.2.141
```

step 4 Validation

```
Switch# show dns server
Current DNS name server configuration:
   Server              IP Address
------------------------------------------------------------------
1   nameserver          202.100.10.20
```

4.6.3 Application Cases

N/A

Chapter 5 IP Routing Configuration Guide

5.1 Configuring IP Unicast-Routing

5.1.1 Overview

Function Introduction

Static routing is a concept describing one way of configuring path selection of routers in computer networks. It is the type of routing characterized by the absence of communication between routers regarding the current topology of the network. This is achieved by manually adding routes to the routing table. The opposite of static routing is dynamic routing, sometimes also referred to as adaptive routing.

In these systems, routes through a data network are described by fixed paths (statically). These routes are usually entered into the router by the system administrator. An entire network can be configured using static routes, but this type of configuration is not fault tolerant. When there is a change in the network or a failure occurs between two statically defined nodes, traffic will not be rerouted. This means that anything that wishes to take an affected path will either have to wait for the failure to be repaired or the static route to be updated by the administrator before restarting its journey. Most requests will time out (ultimately failing) before these repairs can be made. There are, however, times when static routes can improve the performance of a network. Some of these include stub networks and default routes.

Principle Description

N/A

5.1.2 Configuration

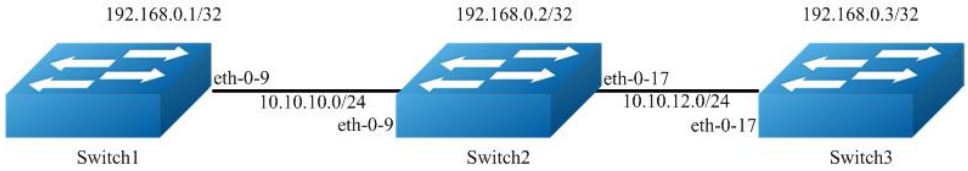


Figure 1-29 ip unicast routing

This example shows how to enable static route in a simple network topology.

There are 3 static routes on Switch1, one is to achieve remote network 10.10.12.0/24, the other two are to achieve the loopback addresses on Switch2 and Switch3. There is a default static route on Switch3, that is, static routes use same gateway or nexthop address. There are 2 static routes on swithc2, both of them are to achieve the remote switch's loopback address.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, set the attributes and ip address

Configure on Switch1:

```
Switch(config)# interface eth-0-9
Switch(config-if)# no shutdown
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1/24
Switch(config-if)# exit

Switch(config)# interface loopback 0
Switch(config-if)# ip address 192.168.0.1/32
Switch(config-if)# exit
```

Configure on Switch2:

```
Switch(config)# interface eth-0-9
Switch(config-if)# no shutdown
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.2/24
```

```
Switch(config-if)# exit

Switch(config)# interface eth-0-17
Switch(config-if)# no shutdown
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.12.2/24
Switch(config-if)# exit

Switch(config)# interface loopback 0
Switch(config-if)# ip address 192.168.0.2/32
Switch(config-if)# exit
```

Configure on Switch3:

```
Switch(config)# interface eth-0-17
Switch(config-if)# no shutdown
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.12.3/24
Switch(config-if)# exit

Switch(config)# interface loopback 0
Switch(config-if)# ip add 192.168.0.3/32
Switch(config-if)# exit
```

step 3 Configuring static route

Configure on Switch1:

Note:Specify the destination prefix and mask for the network for which a gateway is required, for example, 10.10.12.0/24. Add a gateway for each of them (in this case 10.10.10.2 for all). Since R2 is the only next hop available, you can configure a default route instead of configuring the same static route for individual addresses.

```
Switch(config)# ip route 10.10.12.0/24 10.10.10.2
Switch(config)# ip route 192.168.0.2/32 10.10.10.2
Switch(config)# ip route 192.168.0.3/32 10.10.10.2
```

Configure on Switch2:

```
Switch(config)# ip route 192.168.0.1/32 10.10.10.1
Switch(config)# ip route 192.168.0.3/32 10.10.12.3
```

Configure on Switch3:

Note:Specify 10.10.12.2 as a default gateway to reach any network. Since 10.10.12.2 is the only route available you can specify it as the default gateway instead of specifying it as the gateway for individual network or host addresses.

```
Switch(config)# ip route 0.0.0.0/0 10.10.12.2
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

Use the following command to display the route information on Switch1:

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C      10.10.10.0/24 is directly connected, eth-0-9
C      10.10.10.1/32 is in local loopback, eth-0-9
S      10.10.12.0/24 [1/0] via 10.10.10.2, eth-0-9
C      192.168.0.1/32 is directly connected, loopback0
S      192.168.0.2/32 [1/0] via 10.10.10.2, eth-0-9
S      192.168.0.3/32 [1/0] via 10.10.10.2, eth-0-9
```

Use the following command to display the route information on Switch2:

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C      10.10.10.0/24 is directly connected, eth-0-9
C      10.10.10.2/32 is in local loopback, eth-0-9
C      10.10.12.0/24 is directly connected, eth-0-17
C      10.10.12.2/32 is in local loopback, eth-0-17S      192.168.0.1/32 [1/0] via 10.10.10.1, eth-0-9
C      192.168.0.2/32 is directly connected, loopback0
S      192.168.0.3/32 [1/0] via 10.10.12.3, eth-0-17
```

Use the following command to display the route information on Switch3:

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
Gateway of last resort is 10.10.12.2 to network 0.0.0.0
S*     0.0.0.0/0 [1/0] via 10.10.12.2, eth-0-17
C      10.10.12.0/24 is directly connected, eth-0-17
C      10.10.12.3/32 is in local loopback, eth-0-17
C      192.168.0.3/32 is directly connected, loopback0
```

5.1.3 Application Cases

N/A

5.2 Configuring RIP

5.2.1 Overview

Function Introduction

Routing Information Protocol (RIP) is an IP route exchange protocol that uses a distance vector (a number representing distance) to measure the cost of a given route. The cost is a distance vector because the cost is often equivalent to the number of router hops between the source and the destination networks. RIP can receive multiple paths to a destination. The system evaluates the paths, selects the best path, and saves the path in the IP route table as the route to the destination. Typically, the best path is the path with the fewest hops. A hop is another router through which packets must travel to reach the destination. If RIP receives a RIP update from another router that contains a path with fewer hops than the path stored in the route table, the system replaces the older route with the newer one. The system then includes the new path in the updates it sends to other RIP routers. RIP routers also can modify a route's cost, generally by adding to it, to bias the selection of a route for a given destination. In this case, the actual number of router hops may be the same, but the route has an administratively higher cost and is thus less likely to be used than other, lower-cost routes. A RIP route can have a maximum cost of 15. Any destination with a higher cost is considered unreachable. Although limiting to larger networks, the low maximum hop count prevents endless loops in the network.

This chapter contains basic RIP configuration examples. To see details on the commands used in these examples, or to see the outputs of the Validation commands, refer to the RIP Command Reference. To avoid repetition, some Common commands, like configure terminal, have not been listed under the Commands Used section.

Principle Description

Reference to RFC 2453

5.2.2 Configuration

Enabling RIP

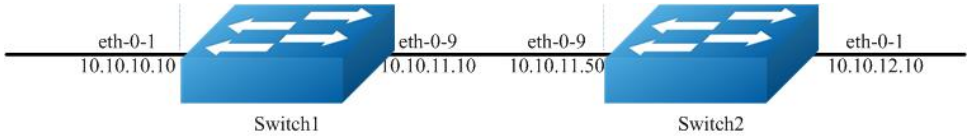


Figure 1-30 enable rip

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, set the attributes and ip address

Configure on Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.10.10/24
Switch(config-if)# exit

Switch(config)# interface eth-0-9
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.11.10/24
Switch(config-if)# exit
```

Configure on Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.12.10/24
Switch(config-if)# exit

Switch(config)# interface eth-0-9
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.11.50/24
Switch(config-if)# exit
```

step 3 Enable RIP routing process and associate networks

Configure on Switch1:

```
Switch(config)# router rip
Switch(config-router)#network 10.10.10.0/24
Switch(config-router)#network 10.10.11.0/24
Switch(config-router)# exit
```

Configure on Switch2:

```
Switch(config)# router rip
Switch(config-router)#network 10.10.11.0/24
Switch(config-router)#network 10.10.12.0/24
Switch(config-router)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Use the following command to display the database of rip on Switch1:

```
Switch# show ip rip database
Codes:  R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
      C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
  Network          Next Hop       Metric From          If      Time
Rc 10.10.10.0/24                    1                 eth-0-1
Rc 10.10.11.0/24                    1                 eth-0-9
R  10.10.12.0/24     10.10.11.50      2 10.10.11.50     eth-0-9  00: 02: 52
```

Use the following command to display the protocol state of rip process on Switch1:

```
Switch# show ip protocols rip
Routing protocol is "rip"
 Sending updates every 30 seconds with +/-5 seconds, next due in 17 seconds
 Timeout after 180 seconds, Garbage collect after 120 seconds
 Outgoing update filter list for all interface is not set
 Incoming update filter list for all interface is not set
 Default redistribution metric is 1
 Redistributing:
 Default version control:  send version 2, receive version 2
   Interface       Send        Recv   Key-chain
   eth-0-1         2           2
   eth-0-9         2           2
 Routing for Networks:
   10.10.10.0/24
   10.10.11.0/24
 Routing Information Sources:
   Gateway         Distance  Last Update  Bad Packets  Bad Routes
   10.10.11.50          120  00: 00: 22             0           0
 Number of routes (including connected):  3
 Distance:  (default is 120)
```

Use the following command to display the interface of rip on Switch1:

```
Switch# show ip rip interface
eth-0-1 is up, line protocol is up
 Routing Protocol:  RIP
   Receive RIP packets
   Send RIP packets
   Passive interface:  Disabled
   Split horizon:  Enabled with Poisoned Reversed
   IP interface address:
     10.10.10.10/24
eth-0-9 is up, line protocol is up
 Routing Protocol:  RIP
   Receive RIP packets
   Send RIP packets
   Passive interface:  Disabled
   Split horizon:  Enabled with Poisoned Reversed
   IP interface address:
     10.10.11.10/24
```

Use the following command to display routes on Switch1:

```
Switch# show ip route
Codes:  K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      [*] - [AD/Metric]
      * - candidate default
C     10.10.10.0/24 is directly connected, eth-0-1
C     10.10.10.10/32 is in local loopback, eth-0-1
C     10.10.11.0/24 is directly connected, eth-0-9
C     10.10.11.10/32 is in local loopback, eth-0-9
R     10.10.12.0/24 [120/2] via 10.10.11.50, eth-0-9, 00: 25: 50
```

Configuring The RIP Version

Figure 1-31 rip version

Configure the receive and send specific versions of packets on an interface .

In this example, Switch2 is configured to receive and send RIP version 1 and 2 on eth-0-9 and eth-0-20.

step 1 Enter the configure mode

The following commands operate on Switch2:

```
Switch# configure terminal
```

step 2 Enable RIP routing process

```
Switch(config)# router rip
Switch(config-router)# exit
```

step 3 Enter the interface configure mode and set the version for sending and receiving rip packets

```
Switch(config)# interface eth-0-9
Switch(config-if)# ip rip send version 1 2
Switch(config-if)# ip rip receive version 1 2
Switch(config-if)# quit

Switch(config)# interface eth-0-20
Switch(config-if)# ip rip send version 1 2
Switch(config-if)# ip rip receive version 1 2
Switch(config-if)# quit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

Use the following command to display the configuration on Switch1:

```
Switch# show running-config
interface eth-0-9
 no switchport
 ip address 10.10.11.10/24
!
router rip
 network 10.10.11.0/24
```

Use the following command to display the database of rip on Switch2:

```
Switch# show ip rip database
Codes:  R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
     C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
  Network         Next Hop      Metric From         If     Time
R  10.0.0.0/8                        1            eth-0-9
Rc 10.10.11.0/24                       1            eth-0-9
Rc 10.10.12.0/24                       1            eth-0-20
```

Use the following command to display the protocol state of rip process on Switch2:

```
Switch# show ip protocols rip
Routing protocol is "rip"
```

```
 Sending updates every 30 seconds with +/-5 seconds, next due in 1 seconds
 Timeout after 180 seconds, Garbage collect after 120 seconds
 Outgoing update filter list for all interface is not set
 Incoming update filter list for all interface is not set
 Default redistribution metric is 1
 Redistributing:
 Default version control:  send version 2, receive version 2
   Interface       Send        Recv   Key-chain
   eth-0-9         1 2         1 2
   eth-0-20        1 2         1 2
 Routing for Networks:
   10.10.11.0/24
   10.10.12.0/24
 Routing Information Sources:
   Gateway         Distance  Last Update  Bad Packets  Bad Routes
   10.10.11.10          120  00: 00: 22             0         0
   10.10.12.50          120  00: 00: 27             0         0
 Number of routes (including connected):  3
 Distance:  (default is 120)
```

Use the following command to display the interface of rip on Switch2:

```
Switch# show ip rip interface
eth-0-9 is up, line protocol is up
  Routing Protocol:  RIP
    Receive RIPv1 and RIPv2 packets
    Send RIPv1 and RIPv2 packets
    Passive interface:  Disabled
    Split horizon:  Enabled with Poisoned Reversed
    IP interface address:
      10.10.11.50/24
eth-0-20 is up, line protocol is up
  Routing Protocol:  RIP
    Receive RIPv1 and RIPv2 packets
    Send RIPv1 and RIPv2 packets
    Passive interface:  Disabled
    Split horizon:  Enabled with Poisoned Reversed
    IP interface address:
      10.10.12.10/24
```

Use the following command to display the configuration on Switch2:

```
Switch# show run
interface eth-0-9
 no switchport
 ip address 10.10.11.50/24
 ip rip send version 1 2
 ip rip receive version 1 2
!
interface eth-0-20
 no switchport
 ip address 10.10.12.10/24
 ip rip send version 1 2
 ip rip receive version 1 2
!
router rip
 network 10.10.11.0/24
 network 10.10.12.0/24
```

Use the following command to display the configuration on Switch3:

```
Switch# show running-config
interface eth-0-20
 no switchport
 ip address 10.10.12.50/24
!
router rip
 network 10.10.12.0/24
```

Configuring Metric Parameters

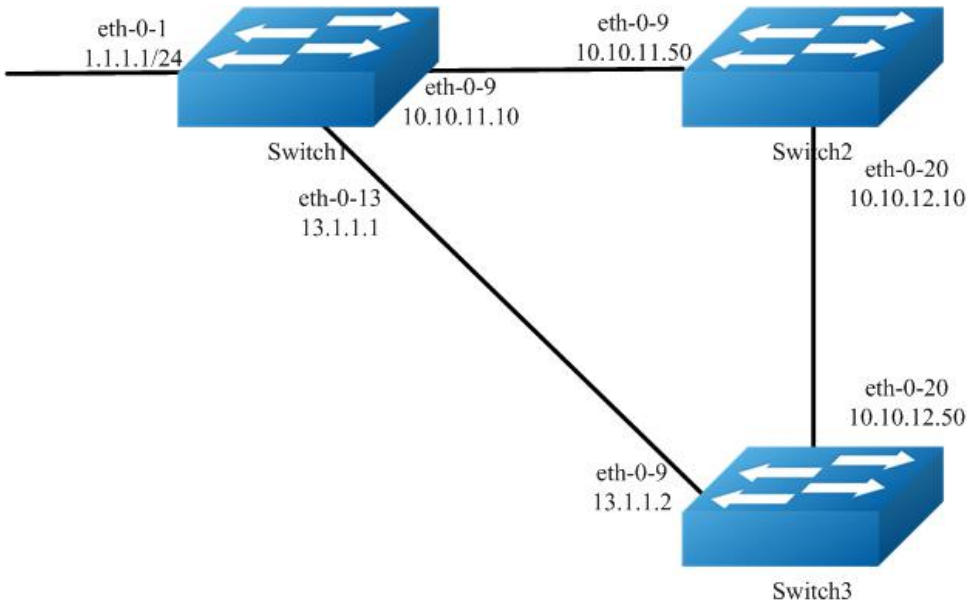


Figure 1-32 rip metric

A RIP offset list allows you to add to the metric of specific inbound or outbound routes learned or advertised by RIP. RIP offset lists provide a simple method for adding to the cost of specific routes and therefore biasing the router's route selection away from those routes. An offset list consists of the following parameters:

- An ACL that specifies the routes to which to add the metric. The direction:
- In: applies to routes the router learns from RIP neighbors.
- Out: applies to routes the router is advertising to its RIP neighbors.
- The offset value that will be added to the routing metric of the routes that match the ACL.
- The interface that the offset list applies (optional).

If a route matches both a global offset list (without specified interface) and an interface-based offset list, the interface-based offset list takes precedence. The interface-based offset list's metric is added to the route in this case.

This example Switch1 will advertise route 1.1.1.0 out of int eth-0-13 with metric 3.

step 1 precondition

Switch1

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
interface eth-0-13
no switchport
ip address 13.1.1.1/24
!
router rip
network 1.1.1.0/24
network 10.10.11.0/24
network 13.1.1.0/24
```

Switch2

```
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
interface eth-0-20
no switchport
ip address 10.10.12.10/24
!
router rip
network 10.10.11.0/24
network 10.10.12.0/24
```

Switch3

```
interface eth-0-13
no switchport
ip address 13.1.1.2/24
!
interface eth-0-20
no switchport
ip address 10.10.12.50/24
!
router rip
network 10.10.12.0/24
network 13.1.1.0/24
```

Display the routes on Switch3:

```
Switch# show ip route rip
R      1.1.1.0/24 [120/2] via 13.1.1.1, eth-0-13, 00: 07: 46
R      10.10.11.0/24 [120/2] via 13.1.1.1, eth-0-13, 00: 07: 39
              [120/2] via 10.10.12.10, eth-0-20, 00: 07: 39
Change router 1.1.1.0/24 via 10.10.12.10
```

step 2 Enter the configure mode

The following commands operate on Switch1:

```
Switch# configure terminal
```

step 3 Configuring access list

```
Switch(config)#ip access-list ripoffset
Switch(config-ip-acl)#permit any 1.1.1.0 0.0.0.255 any
```

step 4 Enable RIP routing process and set offset list and offset value for an interface

```
Switch(config-ip-acl)# router rip
Switch(config-router)# offset-list ripoffset out 3 eth-0-13
```

step 5 Exit the configure mode

```
Switch(config-router)# end
```

step 6 Validation

Display the routes on Switch3. The metric for the route which distributed by Switch1 is 3 now.

```
Switch# show ip route rip
R      1.1.1.0/24 [120/3] via 10.10.12.10, eth-0-20, 00: 00: 02
R      10.10.11.0/24 [120/2] via 13.1.1.1, eth-0-13, 00: 11: 40
              [120/2] via 10.10.12.10, eth-0-20, 00: 11: 40
```

Configuring the Administrative Distance

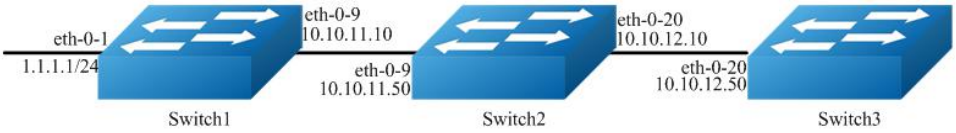


Figure 1-33 rip distance

By default, RIP assigns the default RIP administrative distance (120) to RIP routes. When comparing routes based on administrative distance, the router selects the route with the lower distance. You can change the administrative distance for RIP routes.

This example all Switches have two router protocols, RIP and OSPF, OSPF route has higher priority, Switch3 will change route 1.1.1.0 with administrative distance 100.

step 1 precondition

Switch1

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router ospf
network 1.1.1.0/24 area 0
network 10.10.11.0/24 area 0
!
router rip
network 1.1.1.0/24
network 10.10.11.0/24
```

Switch2

```
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
interface eth-0-20
no switchport
ip address 10.10.12.10/24
!
router ospf
network 10.10.11.0/24 area 0
network 10.10.12.0/24 area 0
!
router rip
network 10.10.11.0/24
network 10.10.12.0/24
```

Switch3

```
interface eth-0-20
no switchport
ip address 10.10.12.50/24
!
router ospf
network 10.10.12.0/24 area 0
!
router rip
network 10.10.12.0/24
```

Display the routes on Switch3:

```
Switch# show ip route
Codes:  K - kernel, C - connected, S - static, R - RIP, B - BGP
    O - OSPF, IA - OSPF inter area
    N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
    E1 - OSPF external type 1, E2 - OSPF external type 2
    i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
    [*] - [AD/Metric]
    * - candidate default
O     1.1.1.0/24 [110/3] via 10.10.12.10, eth-0-20, 01: 05: 49
O     10.10.11.0/24 [110/2] via 10.10.12.10, eth-0-20, 01: 05: 49
C     10.10.12.0/24 is directly connected, eth-0-20
C     10.10.12.50/32 is in local loopback, eth-0-20
```

step 2 Enter the configure mode

The following commands operate on Switch3:

```
Switch# configure terminal
```

step 3 Configuring access list

```
Switch(config)#ip access-list ripdistancelist
Switch(config-ip-acl)#permit any 1.1.1.0 0.0.0.255 any
```

step 4 Enable RIP routing process and set administrative distance

```
Switch(config-ip-acl)# router rip
Switch(config-router)# distance 100 0.0.0.0/0 ripdistancelist
```

step 5 Exit the configure mode

```
Switch(config-router)# end
```

step 6 Validation

Display the routes on Switch3. The distance for the rip route is 100 now.

```
Switch# show ip route
Codes:  K - kernel, C - connected, S - static, R - RIP, B - BGP
    O - OSPF, IA - OSPF inter area
    N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
    E1 - OSPF external type 1, E2 - OSPF external type 2
    i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
    [*] - [AD/Metric]
    * - candidate default
R     1.1.1.0/24 [100/3] via 10.10.12.10, eth-0-20, 00: 00: 02
O     10.10.11.0/24 [110/2] via 10.10.12.10, eth-0-20, 01: 10: 42
C     10.10.12.0/24 is directly connected, eth-0-20
C     10.10.12.50/32 is in local loopback, eth-0-20
```

Configuring Redistribution

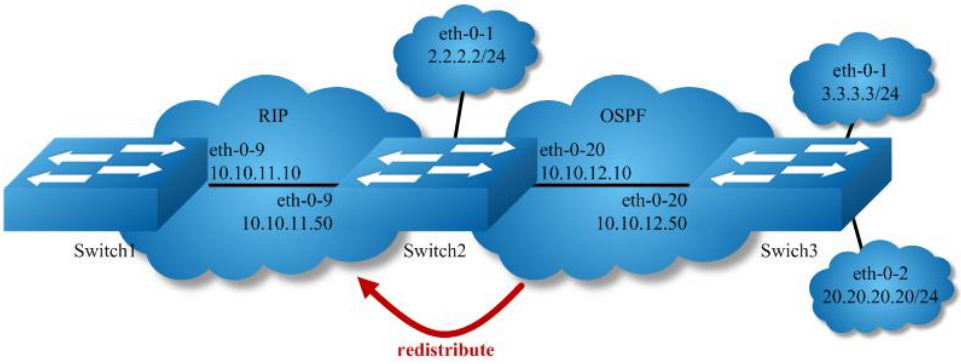


Figure 1-34 rip redistribute

You can configure the router to redistribute static routes, direct connected routes or routes learned through Open Shortest Path First (OSPF) into RIP. When you redistribute a route from one of these other protocols into RIP, the router can use RIP to advertise the route to its RIP neighbors.

Change the default redistribution metric (optional). The router assigns a RIP metric of 1 to each redistributed route by default. You can change the default metric to a value up to 16.

Enable specified routes to redistribute with default or specified metric. This example the router will set the default metric to 2 for redistributed routes and redistributes static routes and direct connected routes to RIP with default metric 2, redistributes OSPF routes with specified metric 5.

step 1 precondition

Switch1

```
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router rip
network 10.10.11.0/24
```

Switch2

```
interface eth-0-1
no switchport
ip address 2.2.2.2/24
!
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
interface eth-0-20
no switchport
ip address 10.10.12.10/24
!
router ospf
network 10.10.12.0/24 area 0
!
router rip
network 10.10.11.0/24
!
ip route 20.20.20.0/24 10.10.12.50
```

Switch3

```
interface eth-0-1
no switchport
ip address 3.3.3.3/24
!
```

```
interface eth-0-2
no switchport
ip address 20.20.20.20/24
!
interface eth-0-20
no switchport
ip address 10.10.12.50/24
!
router ospf
network 3.3.3.0/24 area 0
network 10.10.12.0/24 area 0
```

Display the routes on Switch1:

```
Switch# show ip route
Codes:  K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      [*] - [AD/Metric]
      * - candidate default
C      10.10.11.0/24 is directly connected, eth-0-9
C      10.10.11.10/32 is in local loopback, eth-0-9
```

Display the routes on Switch2:

```
Switch# show ip route
Codes:  K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      [*] - [AD/Metric]
      * - candidate default
C      2.2.2.0/24 is directly connected, eth-0-1
C      2.2.2.02/32 is in local loopback, eth-0-1
O      3.3.3.0/24 [110/2] via 10.10.12.50, eth-0-20, 01: 05: 41
C      10.10.11.0/24 is directly connected, eth-0-9
C      10.10.11.50/32 is in local loopback, eth-0-9
C      10.10.12.0/24 is directly connected, eth-0-20
C      10.10.12.10/24 is in local loopback, eth-0-20
S      20.20.20.0/24 [1/0] via 10.10.12.50, eth-0-20
```

step 2 Enter the configure mode

The following commands operate on Switch2:

```
Switch# configure terminal
```

step 3 Enable RIP routing process and set metric and enable redistribute

```
Switch(config)# router rip
Switch(config-router)# default-metric 2
Switch(config-router)# redistribute static
Switch(config-router)# redistribute connected
Switch(config-router)# redistribute ospf metric 5
```

redistribute connected routes by ospf (optional)

```
Switch(config)# router ospf
Switch(config-router)# redistribute connected
```

step 4 Exit the configure mode

```
Switch(config-router)# end
```

step 5 Validation

Display the routes on Switch1:

```
Switch# show ip route
Codes:  K - kernel, C - connected, S - static, R - RIP, B - BGP
    O - OSPF, IA - OSPF inter area
    N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
    E1 - OSPF external type 1, E2 - OSPF external type 2
    i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
    [*] - [AD/Metric]
    * - candidate default
R     2.2.2.0/24 [120/3] via 10.10.11.50, eth-0-9, 00: 02: 36
R     3.3.3.0/24 [120/6] via 10.10.11.50, eth-0-9, 00: 02: 26
C     10.10.11.0/24 is directly connected, eth-0-9
C     10.10.11.10/32 is in local loopback eth-0-9
R     10.10.12.0/24 [120/3] via 10.10.11.50, eth-0-9, 00: 02: 36
R     20.20.20.0/24 [120/3] via 10.10.11.50, eth-0-9, 00: 02: 41
```

Configuring Split-horizon Parameters

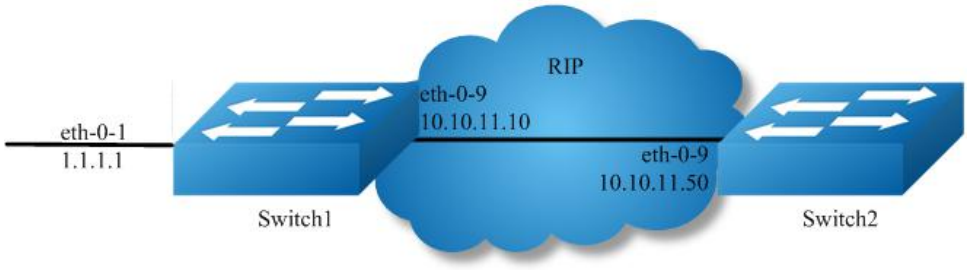


Figure 1-35 rip split-horizon

Normally, routers that are connected to broadcast-type IP networks and that use distance-vector routing protocols employ the split horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router out of any interface from which that information originated. This behavior usually optimizes communications among multiple routers, particularly when links are broken. However, with non-broadcast networks (such as Frame Relay), situations can arise for which this behavior is less than ideal. For these situations, you might want to disable split horizon for RIP.

You can avoid including routes in updates sent to the same gateway from which they were learned. Using the split horizon command omits routes learned from one neighbor, in updates sent to that neighbor. Using the poisoned parameter with this command includes such routes in updates, but sets their metrics to infinity. Thus, advertising these routes means that they are not reachable.

step 1 precondition

Switch1

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router rip
network 10.10.11.0/24
redistribute connected
```

Switch2

```
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
router rip
network 10.10.11.0/24
```

step 2 Enabling debug on Switch2 (optional)

```
Switch# debug rip packet send detail
Switch# terminal monitor
```

step 3 Enter the configure mode

The following commands operate on Switch2:

```
Switch# configure terminal
```

step 4 Enter the interface configure mode and set split-horizon

Disable Split-horizon:

```
Switch(config)#interface eth-0-9
Switch(config-if)# no ip rip split-horizon
```

If debug is enabled, the following messages will be shown:

```
Apr  8 06: 24: 25 Switch RIP4-7:  SEND[eth-0-9]:  Send to 224.0.0.9: 520
Apr  8 06: 24: 25 Switch RIP4-7:  SEND[eth-0-9]:  RESPONSE version 2 packet size 44
Apr  8 06: 24: 25 Switch RIP4-7:  1.1.1.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
Apr  8 06: 24: 25 Switch RIP4-7:  10.10.11.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
```

Enable Split-horizon and poisoned:

```
Switch(config-if)# ip rip split-horizon
Switch(config-if)# ip rip split-horizon poisoned
```

If debug is enabled, the following messages will be shown:

```
Apr  8 06: 38: 35 Switch RIP4-7:  SEND[eth-0-9]:  Send to 224.0.0.9: 520
Apr  8 06: 38: 35 Switch RIP4-7:  SEND[eth-0-9]:  RESPONSE version 2 packet size 44
Apr  8 06: 38: 35 Switch RIP4-7:  1.1.1.0/24 -> 0.0.0.0 family 2 tag 0 metric 16
Apr  8 06: 38: 35 Switch RIP4-7:  10.10.11.0/24 -> 0.0.0.0 family 2 tag 0 metric 16
```

step 5 Exit the configure mode

```
 Switch(config-router)# end
```

step 6 Validation

Use the following command to display the configuration:

```
Switch# show running-config
interface eth-0-9
 no switchport
 ip address 10.10.11.50/24
!
router rip
 network 10.10.11.0/24
!
```

Use the following command to display the interface of rip:

```
Switch# show ip rip interface
eth-0-9 is up, line protocol is up
  Routing Protocol:  RIP
    Receive RIP packets
    Send RIP packets
    Passive interface:  Disabled
    Split horizon:  Enabled with Poisoned Reversed
    IP interface address:
      10.10.11.50/24
```

Configuring Timers

RIP use several timers that determine such variables as the frequency of routing updates, the length of time before a route becomes invalid, and other parameters. You can adjust these timers to tune RIP performance to better suit your internet-work needs. You can make the following timer adjustments:

- The rate (time in seconds between updates) at which routing updates are sent.
- The interval of time (in seconds) after which a route is declared invalid.
- The amount of time (in seconds) that must pass before a route is removed from the routing table.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable RIP routing process and set the timers

Specify the routing table update timer in 10 seconds. Specifies the routing information timeout timer in 180 seconds. Specifies the routing garbage collection timer in 120 seconds:

```
Switch(config)# router rip
Switch(config-router)# timers basic 10 180 120
```

step 3 Exit the configure mode

```
Switch(config-router)# end
```

step 4 Validation

Use the following command to display the protocol state of rip process:

```
Switch# show ip protocols rip
Routing protocol is "rip"
 Sending updates every 10 seconds with +/-5 seconds, next due in 2 seconds
 Timeout after 180 seconds, Garbage collect after 120 seconds
 Outgoing update filter list for all interface is not set
 Incoming update filter list for all interface is not set
 Default redistribution metric is 1
 Redistributing:
 Default version control:  send version 2, receive version 2
  Interface       Send        Recv   Key-chain
  eth-0-9         2           2
 Routing for Networks:
  10.10.11.0/24
 Routing Information Sources:
  Gateway         Distance  Last Update  Bad Packets  Bad Routes
  10.10.11.50          120  00: 00: 02             0           0
 Number of routes (including connected):  5
 Distance:  (default is 120)
```

Configuring RIP Route Distribute Filters

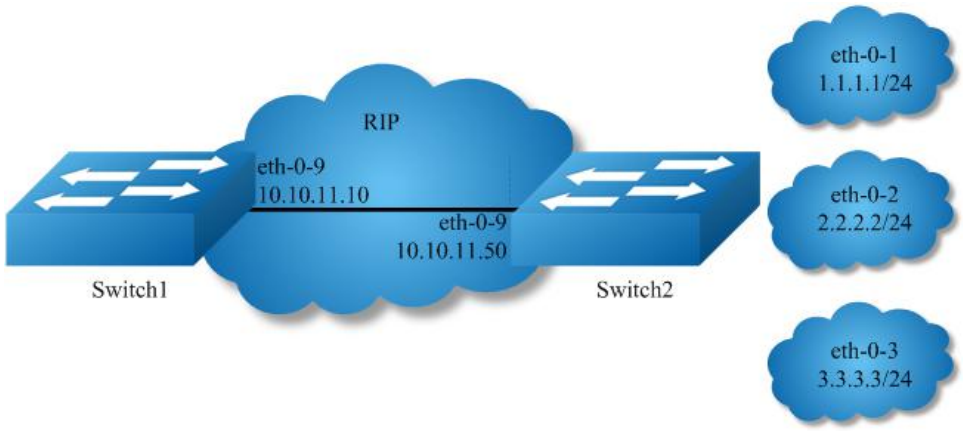


Figure 1-36 rip filter list

A RIP distribute list allows you to permit or deny learning or advertising of specific routes. A distribute list consists of the following parameters:

- An ACL or a prefix list that filter the routes.
- The direction:

  In: filter applies to learned routes.

  Out: filter applies to advertised routes
- The interface that the filer applies (optional).

step 1 precondition

Switch1

```
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router rip
network 10.10.11.0/24
```

Switch2

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-2
no switchport
ip address 2.2.2.2/24
!
interface eth-0-3
no switchport
ip address 3.3.3.3/24
!
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
router rip
network 1.1.1.0/24
network 2.2.2.0/24
network 3.3.3.0/24
network 10.10.11.0/24
```

Display the routes on Switch1:

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
R      1.1.1.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:01:50
R      2.2.2.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:01:50
R      3.3.3.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:01:50
C      10.10.11.0/24 is directly connected, eth-0-9
C      10.10.11.10/32 is in local loopback, eth-0-9
```

step 2 Enter the configure mode

The following commands operate on Switch2:

```
Switch# configure terminal
```

step 3 Configuring prefix list

```
Switch(config)# ip prefix-list 1 deny 1.1.1.0/24
Switch(config)# ip prefix-list 1 permit any
```

step 4 Apply prefix list

```
Switch(config)# router rip
Switch(config-router)# distribute-list prefix 1 out
```

step 5 Exit the configure mode

```
Switch(config-router)# end
```

step 6 Validation

Display the routes on Switch1:

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
R      2.2.2.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:00:08
R      3.3.3.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:00:08
C      10.10.11.0/24 is directly connected, eth-0-9
C      10.10.11.10/32 is in local loopback, eth-0-9
```

Configuring RIPv2 authentication (single key)

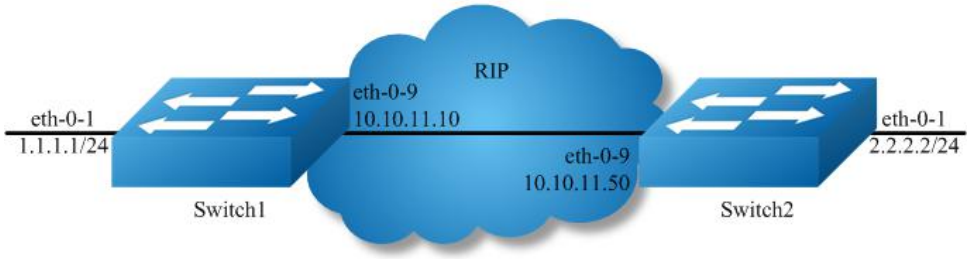


Figure 1-37 rip authentication

RIPv2 supports 2 authentication methods: plaintext and MD5 encryption.

The following example shows how to enable plaintext authentication.

To using this feature, the following steps are required:

- Specify an interface and set the authentication string
- Specify the authentication mode as "text"

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, set the attributes and ip address

Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 1.1.1.1/24
Switch(config-if)# exit

Switch(config-if)# interface eth-0-9
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.11.10/24
Switch(config-if)# exit
```

Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 2.2.2.2/24
Switch(config-if)# exit

Switch(config-if)# interface eth-0-9
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.11.50/24
Switch(config-if)# exit
```

step 3 Enable RIP routing process and set the parameters

```
Switch(config)# router rip
Switch(config-router)# network 10.10.11.0/24
Switch(config-router)# redistribute connected
Switch(config-router)# exit
```

step 4 Specify the authentication string and mode

```
Switch(config)# interface eth-0-9
Switch(config-if)# ip rip authentication string Auth1
Switch(config-if)# ip rip authentication mode text
```

step 5 Exit the configure mode

```
Switch(config-if)# end
```

step 6 Validation

Use the following command to display the database of rip:

```
Switch# show ip rip database

Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
    C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
```

```
  Network         Next Hop      Metric From          If     Time
R  2.2.2.0/24      10.10.11.50       2 10.10.11.50    eth-0-9  00:02:52
Rc 10.10.11.0/24
```

Use the following command to display the protocol state of rip process:

```
Switch# show ip protocols rip
Routing protocol is "rip"
 Sending updates every 30 seconds with +/-5 seconds, next due in 23 seconds
 Timeout after 180 seconds, Garbage collect after 120 seconds
 Outgoing update filter list for all interface is not set
 Incoming update filter list for all interface is not set
 Default redistribution metric is 1
 Redistributing:
  connected  metric default
 Default version control: send version 2, receive version 2
  Interface      Send        Recv   Key-chain
  eth-0-9        2           2
 Routing for Networks:
  10.10.11.0/24
 Routing Information Sources:
  Gateway         Distance  Last Update  Bad Packets  Bad Routes
  10.10.11.50          120  00:00:45             1         0
 Number of routes (including connected): 2
 Distance: (default is 120)



Switch# show ip rip interface
eth-0-9 is up, line protocol is up
 Routing Protocol: RIP
  Receive RIP packets
  Send RIP packets
  Passive interface: Disabled
  Split horizon: Enabled with Poisoned Reversed
  IP interface address:
   10.10.11.10/24
```

Use the following command to display the interface of rip:

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       Dc - DHCP Client
       [*] - [AD/Metric]
       * - candidate default

R       2.2.2.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:02:28
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.10/32 is in local loopback, eth-0-9
```

Configuring RIPv2 MD5 authentication (multiple keys)

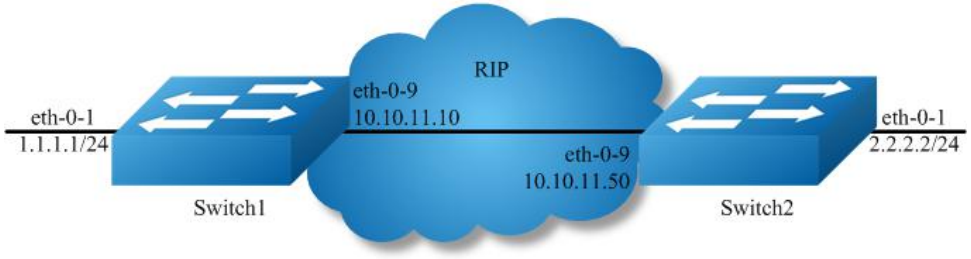


Figure 1-38 rip authentication

This example illustrates the md5 authentication of the routing information exchange process for RIP using multiple keys. Switch1 and B are running RIP and exchange routing updates. To configure authentication on Switch1, define a key chain, specify keys in the key chain and then define the authentication string or passwords to be used by the keys. Then set the time period during which it is valid to receive or send the authentication key by specifying the accept and send lifetimes.[optional].After defining the key string, specify the key chain (or the set of keys) that will be used for authentication on the interface and the authentication mode to be used. Configure Switch1 and B to have the same key ID and key string as Switch1 for the time that updates need to be exchanged.

In md5 authentication, both the key ID and key string are matched for authentication. R1 will receive only packets that match both the key ID and the key string in the specified key chain (within the accept lifetime) on that interface In the following example, Switch2 has the same key ID and key string as Switch1. For additional security, the accept lifetime and send lifetime are configured such that every fifth day the key ID and key string changes. To maintain continuity, the accept lifetimes should be configured to overlap; however, the send lifetime should not be overlapping.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, set the attributes and ip address

Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 1.1.1.1/24
Switch(config-if)# exit

Switch(config-if)# interface eth-0-9
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.11.10/24
Switch(config-if)# exit
```

Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 2.2.2.2/24
Switch(config-if)# exit

Switch(config-if)# interface eth-0-9
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 10.10.11.50/24
Switch(config-if)# exit
```

step 3 Enable RIP routing process and set the parameters

```
Switch(config)# router rip
Switch(config-router)# network 10.10.11.0/24
Switch(config-router)# redistribute connected
Switch(config-router)# exit
```

step 4 Create a key chain, and set the key string and lifetime

```
Switch(config)# key chain SUN
Switch(config-keychain)# key 1
Switch(config-keychain-key)# key-string key1
Switch(config-keychain-key)# accept-lifetime 12:00:00 Mar 2 2012 14:00:00 Mar 7 2012
Switch(config-keychain-key)# send-lifetime 12:00:00 Mar 2 2012 12:00:00 Mar 7 2012
Switch(config-keychain-key)# exit
```

Another key (optional):

```
Switch(config-keychain)# key 2
Switch(config-keychain-key)# key-string Earth
Switch(config-keychain-key)# accept-lifetime 12:00:00 Mar 7 2012 14:00:00 Mar 12 2012
```

```
Switch(config-keychain-key)# send-lifetime 12:00:00 Mar 7 2012 12:00:00 Mar 12 2012
Switch(config-keychain-key)# exit
```

Exit the keychain configure mode:

```
Switch(config-keychain)# exit
```

step 5 Specify the authentication string and mode

```
Switch(config)# interface eth-0-9
Switch(config-if)# ip rip authentication key-chain SUN
Switch(config-if)# ip rip authentication mode md5
```

step 6 Exit the configure mode

```
Switch(config-if)# end
```

step 7 Validation

Use the following command to display the database of rip:

```
Switch# show ip rip database

Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
      C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP

  Network          Next Hop       Metric From          If      Time
R  2.2.2.0/24      10.10.11.50         2 10.10.11.50    eth-0-9  00:01:10
Rc 10.10.11.0/24                       1              eth-0-9
```

Use the following command to display the protocol state of rip process:

```
Switch# show ip protocols rip
Routing protocol is "rip"
 Sending updates every 30 seconds with +/-5 seconds, next due in 17 seconds
 Timeout after 180 seconds, Garbage collect after 120 seconds
 Outgoing update filter list for all interface is not set
 Incoming update filter list for all interface is not set
 Default redistribution metric is 1
 Redistributing:
  connected  metric default
 Default version control: send version 2, receive version 2
  Interface       Send        Recv   Key-chain
  eth-0-9         2           2      SUN
 Routing for Networks:
  10.10.11.0/24
 Routing Information Sources:
  Gateway         Distance  Last Update  Bad Packets  Bad Routes
 Number of routes (including connected): 2
 Distance: (default is 120)
```

Use the following command to display the interface of rip:

```
Switch# show ip rip interface
eth-0-9 is up, line protocol is up
 Routing Protocol: RIP
  Receive RIP packets
  Send RIP packets
  Passive interface: Disabled
  Split horizon: Enabled with Poisoned Reversed
  IP interface address:
   10.10.11.10/24
```

Use the following command to display routes on the device:

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
      O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
```

```
    E1 - OSPF external type 1, E2 - OSPF external type 2
    i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
    Dc - DHCP Client
    [*] - [AD/Metric]
    * - candidate default

C       1.1.1.0/24 is directly connected, eth-0-1
C       1.1.1.1/32 is in local loopback, eth-0-1
R       2.2.2.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:02:27
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.10/32 is in local loopback, eth-0-9
```

Use the following command to display key chain:

```
Switch# show key chain
key chain SUN:
  key 1 -- text "key1"
    accept-lifetime <12:00:00 Mar 02 2012> - <14:00:00 Mar 07 2012>
    send-lifetime <12:00:00 Mar 02 2012> - < 12:00:00 Mar 07 2012>
  key 2 -- text "Earth"
    accept-lifetime <12:00:00 Mar 07 2012> - <14:00:00 Mar 12 2012>
    send-lifetime <12:00:00 Mar 07 2012> - < 12:00:00 Mar 12 2012>
Switch#
```

5.2.3 Application Cases

N/A

5.3 Configuring Prefix-list

5.3.1 Overview

Function Introduction

Routing Policy is the technology for modifying route information to change traffic route. Prefix list is a kind of route policies that used to control and modify routing information. A prefix list is identified by list name and contains one or more ordered entries which are processed sequentially. Each entry provides a matched range for network prefix and has a unique sequence number in the list. In the matching process, switch will check entries orderly. If a entry matches conditions, this process would finish.

Principle Description

N/A

5.3.2 Configuration

Basic Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create a prefix-list

Note: Create a prefix-list. If the sequence of the rule is not specified, system should automatically assign an sequence number for it. Support different actions such as permit and deny. Support to add description string for a prefix-list.

```
Switch(config)# ip prefix-list test seq 1 deny 35.0.0.0/8 le 16
Switch(config)# ip prefix-list test permit any
Switch(config)# ip prefix-list test description this prefix list is fot test
Switch(config)# ip prefix-list test permit 36.0.0.0/24
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display the prefix-list:

```
Switch# show ip prefix-list detail
Prefix-list list number: 1
Prefix-list entry number: 3
Prefix-list with the last deletion/insertion: test
ip prefix-list test:
  Description: this prefix list is fot test
  count: 3, range entries: 0, sequences: 1 - 10
    seq 1 deny 35.0.0.0/8 le 16 (hit count: 0, refcount: 0)
    seq 5 permit any (hit count: 0, refcount: 0)
    seq 10 permit 36.0.0.0/24 (hit count: 0, refcount: 0)
```

Used by rip

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create a prefix-list

```
Switch(config)# ip prefix-list aa seq 11 deny 35.0.0.0/8 le 16
Switch(config)# ip prefix-list aa permit any
```

step 3 Apply the prefix-list under the router rip configure mode

```
Switch(config)# router rip
Switch(config-router)# distribute-list prefix aa out
Switch(config-router)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Use the following command to display the prefix-list:

```
Switch# show ip prefix-list
ip prefix-list aa: 2 entries
    seq 11 deny 35.0.0.0/8 le 16
    seq 15 permit any
```

Use the following command to display the configuration of the device:

```
Switch# show running-config
Building configuration...
…
ip prefix-list aa seq 11 deny 35.0.0.0/8 le 16
ip prefix-list aa seq 15 permit any
…
router rip
 distribute-list prefix aa out
```

Used by Route-map

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create a prefix-list

```
Switch(config)# ip prefix-list aa seq 11 deny 3.3.3.0/8 le 24
Switch(config)# ip prefix-list aa permit any
```

step 3 create a route map to match the prefix-list

```
Switch(config)# route-map abc permit
Switch(config-route-map)# match ip address prefix-list aa
Switch(config-route-map)# set local-preference 200
Switch(config-route-map)# exit

Switch(config)# route-map abc permit 20
Switch(config-route-map)# exit
```

step 4 Apply the route under the router bgp configure mode

```
Switch(config)# router bgp 1
Switch(config-router)# neighbor 1.1.1.2 remote-as 1
Switch(config-router)# neighbor 1.1.1.2 route-map abc out
Switch(config-router)# network 2.2.2.2/32
Switch(config-router)# network 3.3.3.3/32
```

step 5 Exit the configure mode

```
Switch(config-router)# end
```

step 6 Validation

Use the following command to display the route map:

```
Switch # show route-map
route-map abc, permit, sequence 10
  Match clauses:
    ip address prefix-list aa
  Set clauses:
    local-preference 200
route-map abc, permit, sequence 20
  Match clauses:
Set clauses:
```

Use the following command to display the configuration of the device:

```
Switch # show running-config
Building configuration...
…
ip prefix-list aa seq 11 deny 3.3.3.0/8 le 24
ip prefix-list aa seq 15 permit any
!
!
route-map abc permit 10
 match ip address prefix-list aa
 set local-preference 200
!
route-map abc permit 20
…
router bgp 1
 neighbor 1.1.1.2 remote-as 1
 !
 address-family ipv4
 no synchronization
 network 2.2.2.2 mask 255.255.255.255
 network 3.3.3.3 mask 255.255.255.255
 neighbor 1.1.1.2 activate
 neighbor 1.1.1.2 route-map abc out
 exit-address-family
 !
 address-family vpnv4 unicast
 no synchronization
 exit-address-family
```

5.3.3 Application Cases

N/A

5.4 Configuring Route-map

5.4.1 Overview

Function Introduction

Route-map is used to control and modify routing information. The route-map command allows redistribution of routes. It has a list of match and set commands associated with it. The match commands specify the conditions under which redistribution is allowed, and the set commands specify the particular redistribution actions to be performed if the criteria enforced by match commands are met. Route maps are used for detailed control over route distribution between routing processes. Route maps also allow policy routing, and might route packets to a different route than the obvious shortest path.

If the permit parameter is specified, and the match criteria are met, the route is redistributed as specified by set actions. If the match criteria are not met, the next route map with the same tag is tested. If the deny parameter is specified, and the match criteria are met, the route is not redistributed, and any other route maps with the same map tag are not examined. Routes are checked from line to line looking for a match. If there is no match and the bottom of the route map is reached, then the router denies the route from being redistributed. There is always an implicit deny at the end of a route map.

Specify the sequence parameter to indicate the position a new route map is to have in the list of route maps already configured with the same name.

Principle Description

N/A

5.4.2 Configuration

Configuring Route-map for OSPF

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create route map and set the rule and action

NOTE:
The name of route-map is up to 20 characters, in this example the name is “abc”. Two actions “permit” and “deny” are supported; the default action is “permit”. The valid range for sequence number is 1-65535. If the sequence number is not specified when creating first rule of the route-map, system assigns number 10 by default.

```
Switch(config)# route-map abc permit
Switch(config-route-map)# match metric 20
Switch(config-route-map)# set tag 2
Switch(config-route-map)# exit

Switch(config)# route-map abc permit 20
Switch(config-route-map)# exit
```

step 3 Enter the router ospf configure mode, redistribute rip routes and apply the route map

```
Switch(config)# router ospf 100
Switch(config-router)# redistribute rip route-map abc
Switch(config-router)# exit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch# show route-map
route-map abc, permit, sequence 10
  Match clauses:
    metric 20
  Set clauses:
    tag 2
route-map abc, permit, sequence 20
```

```
 Match clauses:
 Set clauses:
```

Configuring Route-map for BGP

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create ip access list

```
Switch(config)# ip access-list acl1
Switch(config-ip-acl)# permit any 3.3.3.0 0.0.0.255 any
Switch(config-ip-acl)# exit
```

step 3 Create route map to match the access list and set the rule and action

```
Switch(config)# route-map abc permit
Switch(config-route-map)# match ip address acl1
Switch(config-route-map)# set local-preference 200
Switch(config-route-map)# exit

Switch(config)# route-map abc permit 20
Switch(config-route-map)# exit
```

step 4 Enter the router bgp configure mode, and apply the route map

```
Switch(config)# router bgp 1
Switch(config-router)# neighbor 1.1.1.2 remote-as 1
Switch(config-router)# neighbor 1.1.1.2 route-map abc out
Switch(config-router)# network 2.2.2.2/32
Switch(config-router)# network 3.3.3.3/32
Switch(config-router)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
DUT1# show route-map
route-map abc, permit, sequence 10
 Match clauses:
   ip address acl1
 Set clauses:
   local-preference 200
route-map abc, permit, sequence 20
 Match clauses:
 Set clauses:
DUT2# show ip bgp
BGP table version is 6, local router ID is 1.1.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
*>i2.2.2.2/32       1.1.1.1                  0    100      0 i
*>i3.3.3.3/32       1.1.1.1                  0    200      0 i
```

5.4.3 Application Cases

N/A

Chapter 6 Multicast Configuration Guide

6.1 Configuring IGMP Snooping

6.1.1 Overview

Function Introduction

Layer 2 switches can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded only to those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN switch to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the switch receives an IGMP report from a host for a particular multicast group, the switch adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also deletes entries per entry if it does not receive IGMP membership reports from the multicast clients. The multicast router sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send report and are added to the forwarding table entry. The switch forwards only one report per IP multicast group to the multicast router. It creates one entry per VLAN in the Layer 2 forwarding table for each MAC group from which it receives an IGMP report.

Layer 2 multicast groups learned through IGMP snooping are dynamic. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings

Limitations And Notice:

VRRP, RIP and OSPF used multicast IP address, so you need to avoid use such multicast IP addresses, which have same multicast MAC address with multicast IP address reserved by VRRP, RIP and OSPF.

VRRP used multicast group address 224.0.0.18, so when igmp snooping and VRRP are working, you need to avoid using multicast group address that matched same mac address with group address 224.0.0.18.

OSPF used multicast group address 224.0.0.5, so when igmp snooping and OSFP are working, you need to avoid using multicast group address that matched same mac address with group address 224.0.0.18.

RIP used multicast group address 224.0.0.9, so when igmp snooping and RIP are working, you need to avoid using multicast group address that matched same mac address with group address 224.0.0.9.

Principle Description

N/A

6.1.2 Configuration

Enable Globally Or Per Vlan

IGMP Snooping can be enabled globally or per vlan. If IGMP Snooping is disabled globally, it can't be active on any vlan even it is enabled on the vlan. If IGMP snooping is enabled globally, it can be disabled on a vlan. On the other hand, the global configuration can overwrite the per vlan configuration. By default, IGMP snooping is enabled globally and per vlan.

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Enable igmp snooping globally and per-vlan

```
Switch(config)# ip igmp snooping
Switch(config)# ip igmp snooping vlan 1
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

Use the following command to display igmp snooping of a vlan:

```
Switch # show ip igmp snooping vlan 1
Global Igmp Snooping Configuration
----------------------------------------------------
Igmp Snooping                    :Enabled
Igmp Snooping Fast-Leave         :Disabled
Igmp Snooping Version            :2
Igmp Snooping Robustness Variable    :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Report-Suppression     :Enabled
Vlan 1
-----------
Igmp Snooping                    :Enabled
Igmp Snooping Fast-Leave         :Disabled
Igmp Snooping Report-Suppression     :Enabled
Igmp Snooping Version            :2
Igmp Snooping Robustness Variable    :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Group Access-list      :N/A
Igmp Snooping Mrouter Port           :
Igmp Snooping Mrouter Port Aging Interval(sec) :255
```

Configuring Fast Leave

When IGMP Snooping fast leave is enabled, the igmp snooping group will be removed at once upon receiving a corresponding igmp report. Otherwise the switch will send out specified igmp specific query, if it doesn't get response in specified period, it will remove the group. By default, igmp snooping fast-leave is disabled globally and per vlan.

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Enable Fast Leave globally and per-vlan

```
Switch(config)#ip igmp snooping fast-leave
Switch(config)#ip igmp snooping vlan 1 fast-leave
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch # show ip igmp snooping vlan 1
Global Igmp Snooping Configuration
----------------------------------------------------
Igmp Snooping                    :Enabled
Igmp Snooping Fast-Leave         :Enabled
Igmp Snooping Version            :2
Igmp Snooping Robustness Variable    :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Report-Suppression     :Enabled
Vlan 1
-----------
Igmp Snooping                    :Enabled
Igmp Snooping Fast-Leave         :Enabled
Igmp Snooping Report-Suppression     :Enabled
Igmp Snooping Version            :2
Igmp Snooping Robustness Variable    :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Group Access-list      :N/A
Igmp Snooping Mrouter Port           :
Igmp Snooping Mrouter Port Aging Interval(sec) :255
```

Configuring Querior Parameters

In order for IGMP, and thus IGMP snooping, to function, an multicast router must exist on the network and generate IGMP queries. The tables created for snooping (holding the member ports for a each multicast group) are associated with the querier. Without a querier the tables are not created and snooping will not work.

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Set the global attributes of igmp snooping

```
Switch(config)# ip igmp snooping query-interval 100
Switch(config)# ip igmp snooping query-max-response-time 5
Switch(config)# ip igmp snooping last-member-query-interval 2000
Switch(config)# ip igmp snooping discard-unknown
```

step 3 Set the per-vlan attributes of igmp snooping

```
Switch(config)# ip igmp snooping vlan 1 querier address 10.10.10.1
Switch(config)# ip igmp snooping vlan 1 querier
Switch(config)# ip igmp snooping vlan 1 query-interval 200
Switch(config)# ip igmp snooping vlan 1 query-max-response-time 5
Switch(config)# ip igmp snooping vlan 1 querier-timeout 100
Switch(config)# ip igmp snooping vlan 1 last-member-query-interval 2000
Switch(config)# ip igmp snooping vlan 1 discard-unknown
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch # show ip igmp snooping querier
Global Igmp Snooping Querier Configuration
-------------------------------------------------
Version                    :2
Last-Member-Query-Interval (msec) :2000
Last-Member-Query-Count         :2
Max-Query-Response-Time (sec)    :5
Query-Interval (sec)           :100
Global Source-Address          :0.0.0.0
TCN Query Count                :2
TCN Query Interval (sec)       :10
TCN Query Max Respose Time (sec)  :5
Vlan 1:  IGMP snooping querier status
-------------------------------------------
Elected querier is : 0.0.0.0
-------------------------------------------
Admin state                  :Enabled
Admin version                :2
Operational state            :Non-Querier
Querier operational address     :10.10.10.1
Querier configure address       :10.10.10.1
Last-Member-Query-Interval (msec) :2000
Last-Member-Query-Count         :2
Max-Query-Response-Time (sec)    :5
Query-Interval (sec)           :200
Querier-Timeout (sec)          :100
```

Configuring Mrouter Port

An IGMP Snooping mrouter port is a switch port which is assumed to connect a multicast router. The mrouter port is configured on the vlan or learnt dynamic. When IGMP general query packet or PIMv2 hello packet is received on port of speficified VLAN, this port becomes mrouter port of this vlan. All the igmp queries received on this port will be flooded on the belonged vlan. All the igmp reports and leaves received on this vlan will be forwarded to the mrouter port, directly or aggregated, depending on the report-suppression configuration. In addition, all the multicast traffic on this vlan will be forwarded to this mrouter port.

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Enable igmp snooping report suppression globally

```
Switch(config)# ip igmp snooping report-suppression
```

step 3 Configure mrouter port, Enable igmp snooping report suppression, and set igmp snooping dynamic mrouter port aging interval for a vlan

```
Switch(config)# ip igmp snooping vlan 1 mrouter interface eth-0-1
Switch(config)# ip igmp snooping vlan 1 report-suppression
Switch(config)# ip igmp snooping vlan 1 mrouter-aging-interval 200
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch# show ip igmp snooping vlan 1
Global Igmp Snooping Configuration
---------------------------------------------------
Igmp Snooping                       :Enabled
Igmp Snooping Fast-Leave            :Disabled
Igmp Snooping Version               :2
Igmp Snooping Robustness Variable   :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Report-Suppression     :Enabled
Vlan 1
-----------
Igmp Snooping                       :Enabled
Igmp Snooping Fast-Leave            :Disabled
Igmp Snooping Report-Suppression     :Enabled
Igmp Snooping Version               :2
Igmp Snooping Robustness Variable   :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Group Access-list      :N/A
Igmp Snooping Mrouter Port           :eth-0-1
Igmp Snooping Mrouter Port Aging Interval(sec) :200
```

Configuring Querier TCN

System supports to adapt the multicast router learning and updating after STP convergence by configuring the TCN querier count and querier interval.

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Configuring the TCN querier count and querier interval

```
Switch(config)# ip igmp snooping querier tcn query-count 5
Switch(config)# ip igmp snooping querier tcn query-interval 20
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch # show ip igmp snooping querier
Global Igmp Snooping Querier Configuration
---------------------------------------------------
Version                 :2
```

```
Last-Member-Query-Interval (msec) :1000
Max-Query-Response-Time (sec)    :10
Query-Interval (sec)             :125
Global Source-Address            :0.0.0.0
TCN Query Count                  :5
TCN Query Interval (sec)         :20
Vlan 1:  IGMP snooping querier status
---------------------------------------------
Elected querier is : 0.0.0.0
---------------------------------------------
Admin state                      :Disabled
Admin version                    :2
Operational state                :Non-Querier
Querier operational address      :0.0.0.0
Querier configure address        :N/A
Last-Member-Query-Interval (msec) :1000
Max-Query-Response-Time (sec)    :10
Query-Interval (sec)             :125
Querier-Timeout (sec)            :255
```

Configuring Report Suppression

The switch uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP router suppression is enabled (the default), the switch sends the first IGMP report from all hosts for a group to all the multicast routers. The switch does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Enable Report Suppression globally and per-vlan

```
Switch(config)# ip igmp snooping report-suppression
Switch(config)# ip igmp snooping vlan 1 report-suppression
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch # show ip igmp snooping
Global Igmp Snooping Configuration
------------------------------------------------
Igmp Snooping                    :Enabled
Igmp Snooping Fast-Leave         :Disabled
Igmp Snooping Version            :2
Igmp Snooping Robustness Variable    :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Report-Suppression     :Enabled
Vlan 1
-----------
Igmp Snooping                    :Enabled
Igmp Snooping Fast-Leave         :Disabled
Igmp Snooping Report-Suppression     :Enabled
Igmp Snooping Version            :2
Igmp Snooping Robustness Variable    :2
Igmp Snooping Max-Member-Number      :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Group Access-list      :N/A
Igmp Snooping Mrouter Port           :
Igmp Snooping Mrouter Port Aging Interval(sec) :255
```

Configuring Static group

The switch can build IGMP Snooping Group when receiving IGMP report packet on Layer 2 port of specified VLAN. We also support configure static IGMP Snooping Group by specifying IGMP group, Layer 2 port and VLAN.

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Configure static group

```
Switch(config)# ip igmp snooping vlan 1 static-group 229.1.1.1 interface eth-0-2
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show ip igmp snooping groups
VLAN  Interface    Group-Address    Uptime    Expires-time
1     eth-0-2      229.1.1.1        00:01:08  stopped
```

6.1.3 Application Cases

N/A

Chapter 7 Security Configuration Guide

7.1 Configuring Port Security

7.1.1 Overview

Function Introduction

Port security feature is used to limit the number of "secure" MAC addresses learnt on a particular interface. The interface will forward packets only with source MAC addresses that match these secure addresses. The secure MAC addresses can be created manually, or learnt automatically. After the number of secure MAC addresses reaches the limit for the number of secure MAC addresses, new MAC address can't be learnt or configured on the interface. if the interface then receives a packet with a source MAC address that is different with any of the secure addresses, it is considered as a security violation and should be discarded.

Port security feature also binds a MAC to a port so that the port does not forward packets with source addresses that are outside of defined addresses. If a MAC addresses configured or learnt on a secure port attempts to access another port, this is also considered as a security violation.

Two types of secure MAC addresses are supported:

- Static secure MAC addresses: These are manually configured by the interface configuration command "switchport port-security mac-address".
- Dynamic secure MAC addresses: These are dynamically learnt.
- If a security violation occurs, the packets to be forwarded will be dropped. User can configure the action by command "switchport port-security violation". There are three actions can be chosen:
- errdisable: discard the packet and set the port to errdisable status. Please reference to Ethernet configuration guide, chapter errdisable.
- protect: discard only.
- restrict: discard and record the event in log.

Principle Description

N/A

7.1.2 Configuration

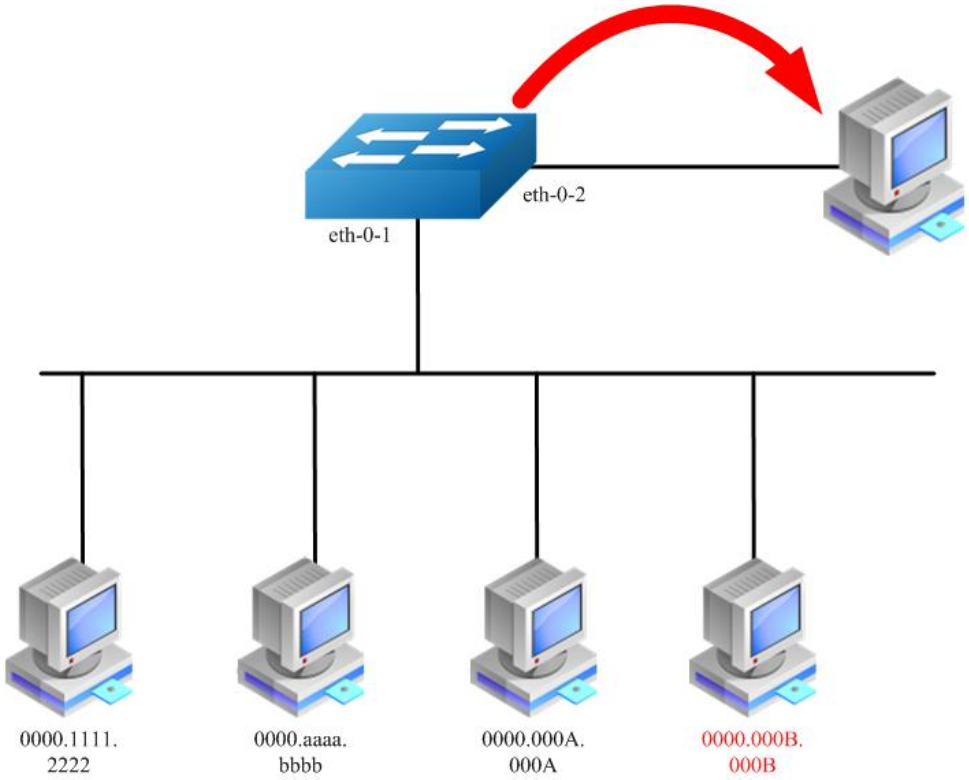


Figure 1-39 Port Security

According to the topology above, only receive three Mac entries and discard source mac 0000.000B.000B after the following configuration:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode, set the attributes, and enable pim-sm

```
Switch(config)# interface eth-0-1
Switch(config-if)# switchport
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 3
Switch(config-if)# switchport port-security mac-address 0000.1111.2222 vlan 1
Switch(config-if)# switchport port-security mac-address 0000.aaaa.bbbb vlan 1
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolationMode
            (Count)      (Count)
---------------------------------------------------------------
 eth-0-1         3          2     restrict

Switch# show port-security address-table
       Secure MAC address table
-----------------------------------------------------------
Vlan   Mac Address      Type             Ports
----   -----------      -------          -------
```

```
1     0000.1111.2222   SecureConfigured    eth-0-1
1     0000.aaaa.bbbb   SecureConfigured    eth-0-1

Switch# show port-security interface eth-0-1
Port security               : enabled
Violation mode              : discard packet and log
Maximum MAC addresses       : 3
Total MAC addresses         : 2
Static configured MAC addresses    : 2
```

7.1.3 Application Cases

N/A

7.2 Configuring Vlan Security

7.2.1 Overview

Function Introduction

Vlan security feature is used to limit the total number of MAC addresses learnt in a particular vlan. The MAC addresses can be added manually, or learnt automatically. After the device reaches the limit for the number of MAC addresses on the vlan, if the vlan receives a packet with an unknown source MAC address, the configured action will take effect.

Two types of MAC addresses are supported:

- Static MAC addresses: These are manually configured by users.
- Dynamic MAC addresses: These are dynamically learnt.
- User can set the action for unknown source MAC packets after the MAC address table count exceed max by using command line "vlan X mac-limit action". Three types of actions are supported:
- Discard: Packet with an unknown source MAC address from the vlan will be discarded and its source MAC address will not be learnt.
- Warn: Packet with an unknown source MAC address from the vlan will be discarded, its source MAC address will not be learnt, but warning log will be printed in syslog.
- Forward: Packets from the vlan will be forwarded without MAC learning or warning log.

MAC address learning feature can be enabled or disabled per-VLAN.

Principle Description

N/A

7.2.2 Configuration

Configuring vlan mac-limit

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan, set the the maximum of MAC addresses and the action at exceeding

```
Switch# configure terminal
Switch(config)# vlan database
Switch(config)# vlan 2
Switch(config-vlan)# vlan 2 mac-limit maximum 100
Switch(config-vlan)# vlan 2 mac-limit action discard
Switch(config-vlan)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show vlan-security
 Vlan  learning-en  max-mac-count  cur-mac-count  action
 ------------------------------------------------------------
2    Enable     100          0          Discard
```

Configuring vlan mac learning

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan, set the mac learning states

```
Switch(config)# vlan database
Switch(config)# vlan 2
Switch(config-vlan)# vlan 2 mac learning disable
Switch(config-vlan)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show vlan-security
 Vlan  learning-en  max-mac-count  cur-mac-count  action
 ------------------------------------------------------------
2    Disable    100          0          Discard
```

7.2.3 Application Cases

N/A

7.3 Configuring Time-Range

7.3.1 Overview

Function Introduction

A time range is created that defines specific absolute times or periodic times of the day and week in order to implement time-based function, such as ACLs. The time range is identified by a name and then referenced by a function, which by itself has no relevance. Therefore, the time restriction is imposed on the function itself. The time range relies on the system clock.

Principle Description

N/A

7.3.2 Configuration

Create an absolute time range

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create a time-range and set absolute time

```
Switch(config)# time-range test-absolute
Switch(config-tm-range)# absolute start 1:1:2 jan 1 2012 end 1:1:3 jan 7 2012
Switch(config-tm-range)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
DUT1# show time-range
time-range test-absolute
  absolute start 01:01:02 Jan 01 2012 end 01:01:03 Jan 07 2012
```

Create a periodic time range

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create a time-range and set periodic time

```
Switch(config)# time-range test-periodic
Switch(config-tm-range)# periodic 1:1 mon to 1:1 wed
Switch(config-tm-range)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
DUT1# show time-range
time-range test-periodic
  periodic 01:01 Mon to 01:01 Wed
```

7.3.3 Application Cases

N/A

7.4 Configuring ACL

7.4.1 Overview

Function Introduction

Access control lists (ACLs) classify traffic with the same characteristics. The ACL can have multiple access control entries (ACEs), which are commands that match fields against the contents of the packet. ACLs can filter packets received on interface by many fields such as ip address, mac address and deny or permit the packets.

Principle Description

- The following terms and concepts are used to describe ACL:
- Access control entry (ACE): Each ACE includes an action element (permit or deny) and a series of filter element based on criteria such as source address, destination address, protocol, and protocol-specific parameters.
- MAC ACL: MAC ACL can filter packet by mac-sa and mac-da, and the mac-address can be masked, or configured as host id, or configured as any to filter all MAC addresses. MAC ACL can also filter other L2 fields such as COS, VLAN-ID, INNER-COS, INNER-VLAN-ID, L2 type, L3 type.

- IPv4 ACL: IPv4 ACL can filter packet by ip-sa and ip-da, and ip-address can be masked, or configured as host id, or configured as any to filter all IPv4 address. IPv4 ACL can also filter other L3 fields such as DSCP, L4 protocol and L4 fields such as TCP port, UDP port, and so on.

- Time Range: Time range can define a period of time only between which the ACE can be valid if the ACE is associated to the time range.

7.4.2 Configuration

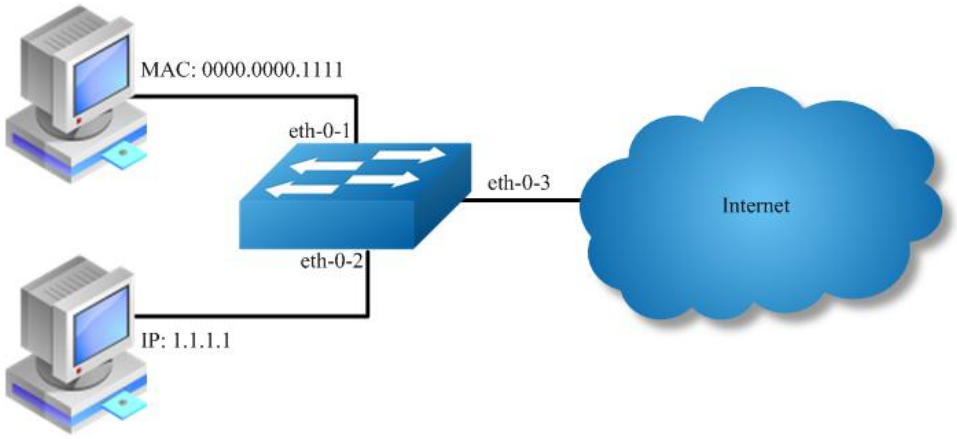


Figure 1-40 acl

In this example, use MAC ACL on interface eth-0-1, to permit packets with source mac 0000.0000.1111 and deny any other packets. Use IPv4 ACL on interface eth-0-2, to permit packets with source ip 1.1.1.1/24 and deny any other packets.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create access list

mac access list:

```
Switch(config)# mac access-list mac
Switch(config-mac-acl)# permit src-mac host 0000.0000.1111 dest-mac any
Switch(config-mac-acl)# deny src-mac any dest-mac any
Switch(config-mac-acl)# exit
```

ip access list:

```
Switch(config)# ip access-list ipv4
Switch(config-ip-acl)# permit any 1.1.1.1 0.0.0.255 any
Switch(config-ip-acl)# deny any any any
Switch(config-ip-acl)# exit
```

step 3 Create class-map, and bind the access list

```
Switch(config)# class-map cmap1
Switch(config-cmap)# match access-group mac
Switch(config-cmap)# exit

Switch(config)# class-map cmap2
Switch(config-cmap)# match access-group ipv4
Switch(config-cmap)# exit
```

step 4 Create policy-map and bind the class map

```
Switch(config)# policy-map pmap1
Switch(config-pmap)# class cmap1
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

```
Switch(config)# policy-map pmap2
Switch(config-pmap)# class cmap2
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

step 5 Apply the policy to the interface

```
Switch(config)# interface eth-0-1
Switch(config-if)# service-policy input pmap1
Switch(config-if)# exit

Switch(config-if)# interface eth-0-2
Switch(config-if)# service-policy input pmap2
Switch(config-if)# exit
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

The result of show running-config is as follows:

```
Switch# show running-config
mac access-list mac
 10 permit src-mac host 0000.0000.1111 dest-mac any
 20 deny src-mac any dest-mac any
!
ip access-list ipv4
 10 permit any 1.1.1.0 0.0.0.255 any
 20 deny any any any
!
class-map match-any cmap1
 match access-group mac
!
class-map match-any cmap2
 match access-group ipv4
!
policy-map pmap1
 class cmap1
!
policy-map pmap2
 class cmap2
!
interface eth-0-1
 service-policy input pmap1
!
interface eth-0-2
 service-policy input pmap2
!
```

7.4.3 Application Cases

N/A

7.5 Configuring Extern ACL

7.5.1 Overview

Function Introduction

Extend IPv4 ACL combines MAC filters with IP filters in one access list. Different from MAC and IP ACL, extend ACL can access-control all packets (IP packets and non-IP packets). Extend ACL supported extend IPv4 ACL.

Principle Description

Following is a brief description of terms and concepts used to describe the extend ACL:

- Extend IPv4 ACL: Extend IPv4 ACL takes advantages of MAC ACL and IPv4 ACL, which combines MAC ACE with IPv4 ACE in an ACL to provide more powerful function of access-controlling traverse packets.
- MAC ACE: Filter packets by mac-sa and mac-da, and the mac-address can be masked, or configured as host id, or configured as any to filter all MAC addresses. Other L2 fields, such as COS, VLAN-ID, INNER-COS, INNER-VLAN-ID, L2 type, L3 type, can also be filtered by MAC ACE.
- IPv4 ACE: Filter packets by ip-sa and ip-da, and ip-address can be masked, or configured as host id, or configured as any to filter all IPv4 address. Other L3 fields such as DSCP, L4 protocol and L4 fields, such as TCP port, UDP port, can also be filtered by IPv4 ACE.

The MAC ACE and IPv4 ACE in an extend IPv4 ACL can be configured alternately in arbitrary order which is completely specified by user.

7.5.2 Configuration

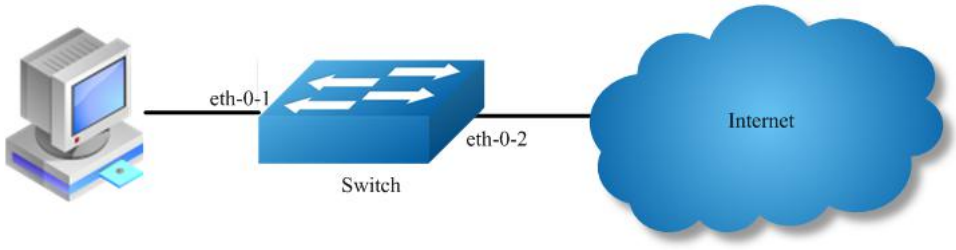


Figure 1-41 extern acl

In this example, use extend IPv4 ACL on interface eth-0-1, to permit packets with source mac 0000.0000.1111 and cos value of 2, permit all TCP packets, and deny any other packets.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create access list

```
Switch(config)# ip access-list ipxacl extend
Switch(config-ex-ip-acl)# permit src-mac host 0000.0000.1111 dest-mac any cos 2
Switch(config-ex-ip-acl)# permit tcp any any
Switch(config-ex-ip-acl)# deny src-mac any dest-mac any
Switch(config-ex-ip-acl)# end
```

step 3 Create class-map, and bind the access list

```
Switch(config)# class-map cmap
Switch(config-cmap)# match access-group ipxacl
Switch(config-cmap)# exit
```

step 4 Create policy-map and bind the class map

```
Switch(config)# policy-map pmap
Switch(config-pmap)# class cmap
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

step 5 Apply the policy to the interface

```
Switch(config)# interface eth-0-1
Switch(config-if)# service-policy input pmap
Switch(config-if)# exit
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

The result of show running-config is as follows:

```
Switch# show running-config
ip access-list ipxacl extend
 10 permit src-mac host 0000.0000.1111 dest-mac any cos 2
 20 permit tcp any any
 30 deny src-mac any dest-mac any
!
class-map match-any cmap
 match access-group ipxacl
!
policy-map pmap
 class cmap
!
interface eth-0-1
 service-policy input pmap
!
Switch# show access-list ip
ip access-list ipxacl extend
 10 permit src-mac host 0000.0000.1111 dest-mac any cos 2
 20 permit tcp any any
 30 deny src-mac any dest-mac any
```

7.5.3 Application Cases

N/A

7.6 Configuring dot1x

7.6.1 Overview

Function Introduction

IEEE 802 Local Area Networks are often deployed in environments that permit unauthorized devices to be physically attached to the LAN infrastructure, or Permit unauthorized users to attempt to access the LAN through equipment already attached.

Port-based network access control makes use of the physical access characteristics of IEEE 802 LAN infrastructures in order to provide a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics, and of preventing access to that port in cases in which the authentication and authorization process fails.

With 802.1X port-based authentication, the devices in the network have specific roles:

- Client: the device (PC) that requests access to the LAN and switch services and responds to requests from the switch. The client software with support the follow the 802.1X standard should run on the PC. For linux system, we recommend the application which named "xsupplicant".
- Authentication server: performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch whether or not the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the client. In this release, the Remote Authentication Dial-In User Service (RADIUS) security system with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server. RADIUS operates in a client/server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.
- Switch (edge switch or wireless access point): controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client. The switch includes the RADIUS client, which is responsible for encapsulating and decapsulation the EAP frames and Interacting with the authentication server. When the switch receives EAPOL frames and relays them to the authentication server, the Ethernet header is stripped and the remaining EAP frame is re-encapsulated in the RADIUS format. The EAP Frames are not modified or examined during encapsulation, and the authentication server must support EAP within the native frame format. When the switch receives frames from the authentication server, the server's frame header is removed, leaving the EAP frame, which is then encapsulated for Ethernet and sent to the client. We can enable dot1x on routed port and access port.

Principle Description

Reference to IEEE Std 802.1X- 2004

7. 6. 2 Configuration

Basic dot1x configuration

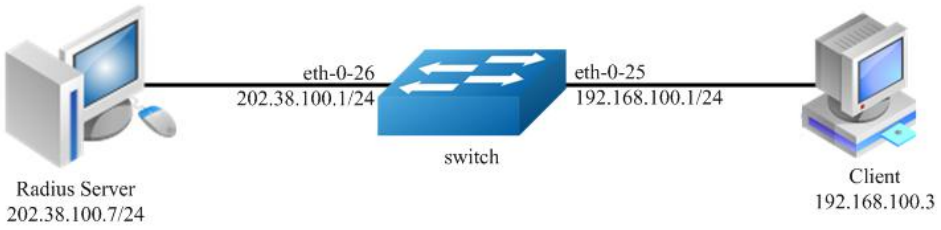


Figure 1-42 dot1x

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable dot1x globally

```
Switch(config)# dot1x system-auth-ctrl
```

step 3 Enter the interface configure mode, set the attributes of the interface and enable dot1x

```
Switch(config)# interface eth-0-25
Switch(config-if)# switchport mode access
Switch(config-if)# dot1x port-control auto
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface vlan 1
Switch(config-if)# ip address 192.168.100.1/24
Switch(config-if)# exit
```

step 4 Set the attributes of Layer 3 interface and set the Radius server

```
Switch(config)# interface eth-0-26
Switch(config-if)# no switchport
Switch(config-if)# ip address 202.38.100.1/24
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# radius-server host 202.38.100.7
Switch(config)# radius-server host 2001:1000::1
Switch(config)# radius-server key test
Switch(config)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show dot1x
802.1X Port-Based Authentication Enabled
 RADIUS server address: 2001:1000::1:1812
 Next radius message ID: 0
 RADIUS server address: 202.38.100.7:1812
 Next radius message ID: 0
Switch# show dot1x interface eth-0-25
802.1X info for interface eth-0-25
 portEnabled          : true
 portControl          : Auto
 portMode             : Port based
 portStatus           : Authorized
 Mac Auth bypass      : disabled
```

```
 reAuthenticate       : disabled
 reAuthPeriod         : 3600
 Max user number      : 255
 Current session number : 1
 Accept user number   : 1
 Reject user number   : 0
 Guest VLAN           : N/A
 Assign VLAN          : N/A
 QuietPeriod          : 60
 ReqMax               : 2
 TxPeriod             : 30
 SuppTimeout          : 30
 ServerTimeout        : 30
 CD: adminControlledDirections : in
 CD: operControlledDirections  : in
 CD: bridgeDetected           : false
======================================

session 1: 1 - 0011.0100.0001
----------------------------------------
 user name : admin
 abort:F fail:F start:F timeout:F success:T
 PAE: state: Authenticated - portMode: Auto
 PAE: reAuthCount: 0 - rxRespId: 0
 BE: state: Idle - reqCount: 0 - idFromServer: 5
```

Enable dot1x on routed port

The example above describes how to enable dot1x on access port. This function can also enable on routed port. The following example shows how to change eth-0-25 to a routed port and enable dot1x.

```
Switch(config)# interface eth-0-25
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.168.100.1/24
Switch(config-if)# dot1x port-control auto
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Using force mode

Dot1x port control mode can be force-authorized or force-unauthorized.

force-authorized:

```
Switch(config)# interface eth-0-25
Switch(config-if)# dot1x port-control force-authorized
Switch(config-if)# exit
```

force-unauthorized:

```
Switch(config)# interface eth-0-25
Switch(config-if)# dot1x port-control force-unauthorized
Switch(config-if)# exit
```

User can choose port control mode as force-authorized,force-unauthorized or auto. The final configuration should over write the previous one.

dot1x optional parameter

Timer for Radius server: Set the wait time for re-activating RADIUS server; Set the maximum failed RADIUS requests sent to server; Set the timeout value for no response from RADIUS server.

```
Switch(config)# radius-server deadtime 10
Switch(config)# radius-server retransmit 5
Switch(config)# radius-server timeout 10
```

Interface attributes: Specify the number of reauthentication attempts before becoming unauthorized; Set the protocol version; Specify the quiet period in the HELD state; Enable reauthentication on a port; Specify the seconds between reauthorization attempts; Specify the

authentication server response timeout; Specify the supplicant response timeout; Specify the Seconds between successive request ID attempts.

```
Switch(config)# interface eth-0-25
Switch(config-if)# dot1x  max-req  5
Switch(config-if)# dot1x protocol-version 1
Switch(config-if)# dot1x quiet-period 120
Switch(config-if)# dot1x reauthentication
Switch(config-if)# dot1x timeout re-authperiod 1800
Switch(config-if)# dot1x timeout server-timeout 60
Switch(config-if)# dot1x timeout supp-timeout 60
Switch(config-if)# dot1x timeout tx-period 60
Switch(config-if)# exit
```

7.6.3 Application Cases

Radius server configuration (Using WinRadius for example)

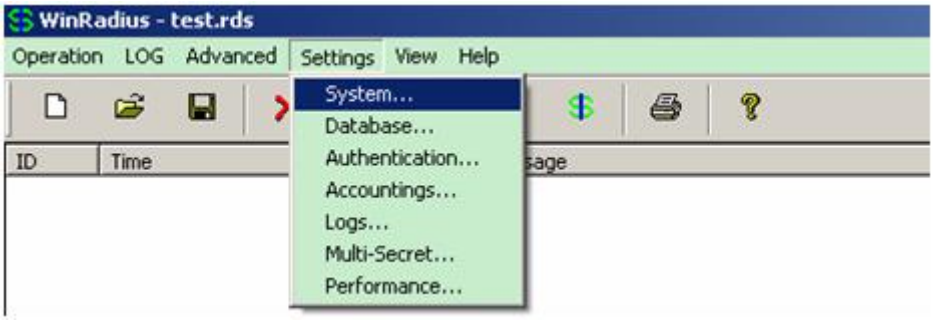


Figure 1-43 Select "Setting-> System"

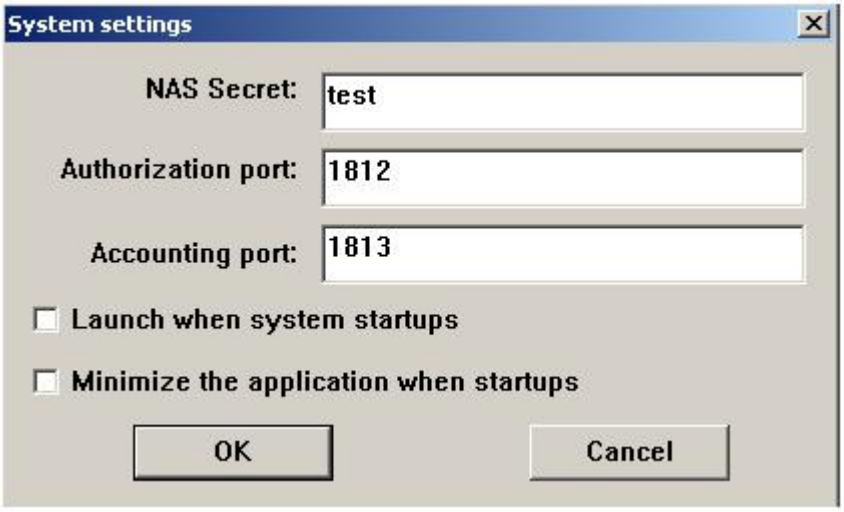


Figure 1-44 Configure the shared-key, authorization port and account port

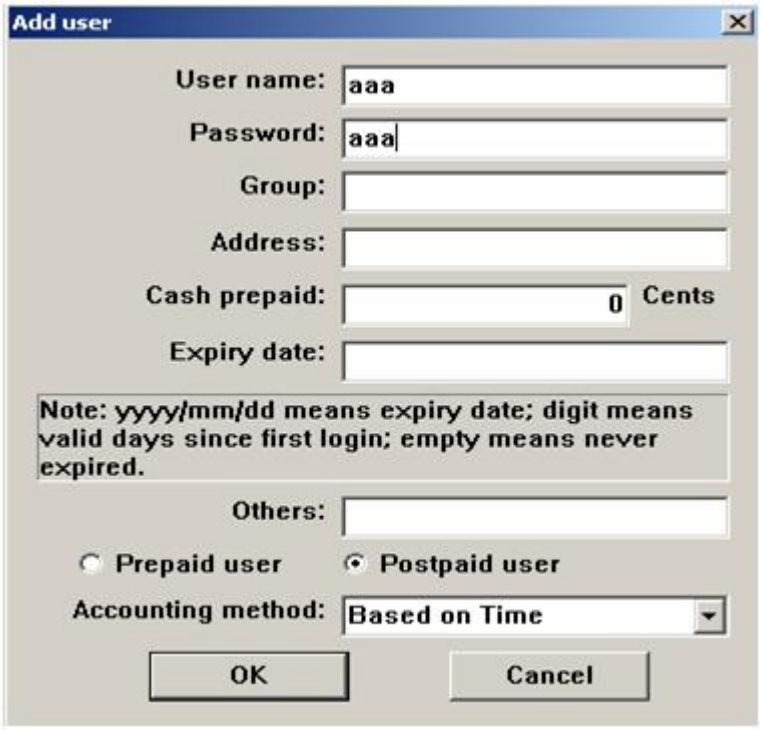


Figure 1-45 Add user name and password on the server

7.7 Configuring Guest VLAN

7.7.1 Overview

Function Introduction

You can configure a guest VLAN for each 802.1x port on the switch to provide limited services to clients (for example, how to download the 802.1x client). These clients might be upgrading their system for 802.1x authentication, and some hosts, such as Windows 98 systems, might not be 802.1x-capable.

When the authentication server does not receive a response to its EAPOL request/identity frame, clients that are not 802.1x-capable are put into the guest VLAN for the port, if one is configured. However, the server does not grant 802.1x-capable clients that fail authentication access to the network. Any number of hosts is allowed access when the switch port is moved to the guest VLAN.

The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

NOTE: Guest VLAN is supported on access port, and not supported on routed port or trunk port.

Principle Description

NA

7. 7. 2 Configuration

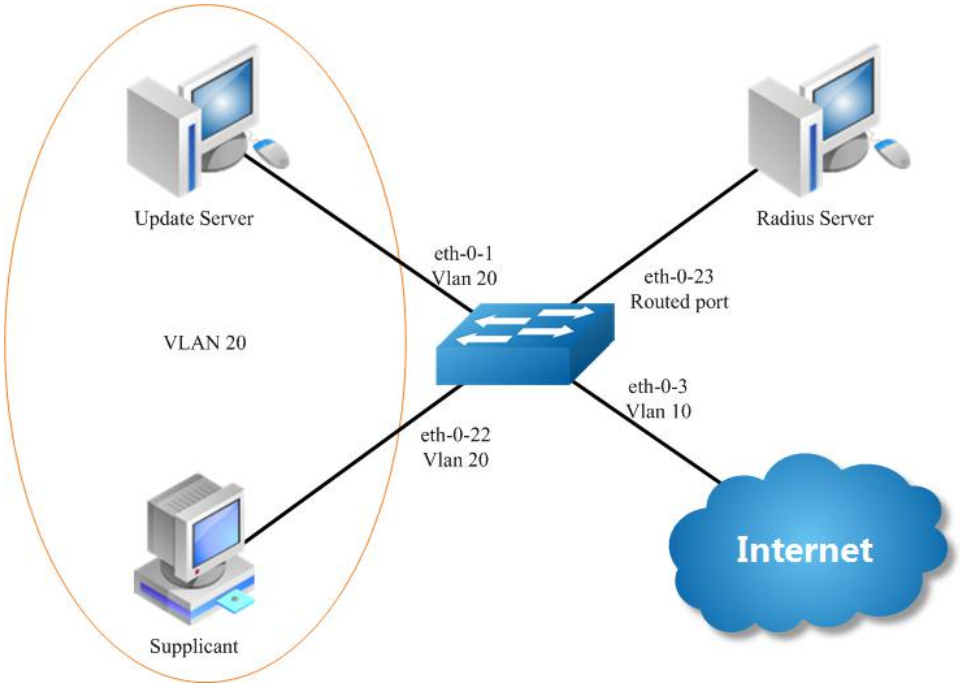


Figure 1-46 Guest vlan: before authenticated

In the above topology, eth-0-22 is an IEEE 802.1X enabled port, and it is in the native VLAN 10, the configured guest VLAN for this port is VLAN 20. So clients that are not 802.1X capable will be put into VLAN 20 after the authenticator had send max EAPOL request/identity frame but got no response.

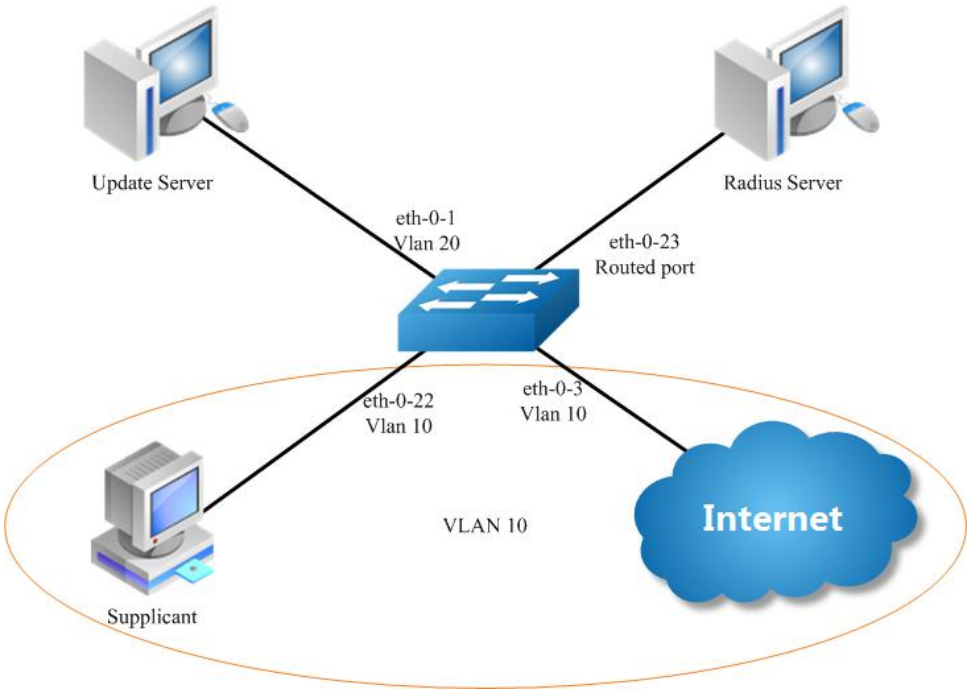


Figure 1-47 Guest vlan: after authenticated

We use remote linux Radius server as authenticate server, the server's address is 202.38.100.7, and the IP address for the connected routed port eth-0-23 is 202.38.100.1. When the client is authenticated by the radius server, then it can access the public internet which is also in VLAN 10.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10
Switch(config-vlan)# vlan 20
Switch(config-vlan)# exit
```

step 3 Enable dot1x globally

```
Switch(config)# dot1x system-auth-ctrl
```

step 4 Enter the interface configure mode, set the attributes of the interface and enable dot1x and set guest vlan

```
Switch(config)# interface eth-0-22
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# dot1x port-control auto
Switch( config-if)# no shutdown
Switch(config-if)# dot1x guest vlan 20
Switch(config-if)# exit
```

step 5 Set the attributes of Layer 3 interface and set the Radius server

```
Switch(config)# interface eth-0-23
Switch(config-if)# no switchport
Switch(config-if)# ip address 202.38.100.1/24
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# radius-server host 202.38.100.7
Switch(config)# radius-server key test
Switch(config)#end
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

Init state:

```
Switch# show running-config
dot1x system-auth-ctrl
radius-server host 202.38.100.7 key test
vlan database
 vlan 10,20
!
interface eth-0-22
 switchport access vlan 10
 dot1x port-control auto
 dot1x guest-vlan 20
!
interface eth-0-23
 no switchport
 ip address 202.38.100.1/24
!

Switch# show dot1x interface eth-0-22
802.1X info for interface eth-0-22
  portEnabled          : true
  portControl          : Auto
  portMode             : Port based
  portStatus           : Unauthorized
  Mac Auth bypass      : disabled
  reAuthenticate       : disabled
```

```
 reAuthPeriod         : 3600
 Max user number      : 255
 Current session number : 0
 Accept user number   : 0
 Reject user number   : 0
 Guest VLAN           : 20
 Assign VLAN          : N/A
 QuietPeriod          : 60
 ReqMax               : 2
 TxPeriod             : 30
 SuppTimeout          : 30
 ServerTimeout        : 30
 CD: adminControlledDirections : in
 CD: operControlledDirections  : in
 CD: bridgeDetected           : false
==========================================

Switch# show vlan brief
VLAN ID  Name          State  STP ID  DSCP    Member ports
                                    (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ======= ========================
1     default        ACTIVE  0     Disable eth-0-1(u) eth-0-2(u)
                                    eth-0-3(u) eth-0-4(u)
                                    eth-0-5(u) eth-0-6(u)
                                    eth-0-7(u) eth-0-8(u)
                                    eth-0-9(u) eth-0-10(u)
                                    eth-0-11(u) eth-0-12(u)
                                    eth-0-13(u) eth-0-14(u)
                                    eth-0-15(u) eth-0-16(u)
                                    eth-0-17(u) eth-0-18(u)
                                    eth-0-19(u) eth-0-20(u)
                                    eth-0-21(u) eth-0-24(u)
                                    eth-0-25(u) eth-0-26(u)
                                    eth-0-27(u) eth-0-28(u)
                                    eth-0-29(u) eth-0-30(u)
                                    eth-0-31(u) eth-0-32(u)
                                    eth-0-33(u) eth-0-34(u)
                                    eth-0-35(u) eth-0-36(u)
                                    eth-0-37(u) eth-0-38(u)
                                    eth-0-39(u) eth-0-40(u)
                                    eth-0-41(u) eth-0-42(u)
                                    eth-0-43(u) eth-0-44(u)
                                    eth-0-45(u) eth-0-46(u)
                                    eth-0-47(u) eth-0-48(u)
10    VLAN0010       ACTIVE  0     Disable eth-0-22(u)
20    VLAN0020       ACTIVE  0     Disable
```

After configure the guest vlan:

unauthorized:

```
Switch# show dot1x interface eth-0-22
802.1X info for interface eth-0-22
 portEnabled          : true
 portControl          : Auto
 portMode             : Port based
 portStatus           : Unauthorized
 Mac Auth bypass      : disabled
 reAuthenticate       : disabled
 reAuthPeriod         : 3600
 Max user number      : 255
 Current session number : 1
 Accept user number   : 0
 Reject user number   : 1
 Guest VLAN           : 20(Port Authorized by guest vlan)
 Assign VLAN          : N/A
 QuietPeriod          : 60
 ReqMax               : 2
 TxPeriod             : 30
 SuppTimeout          : 30
 ServerTimeout        : 30
 CD: adminControlledDirections : in
```

```
 CD: operControlledDirections  : in
 CD: bridgeDetected            : false
==========================================

session 1: 1 - 0011.0100.0001
------------------------------------------
 user name : admin
 abort:F fail:T start:F timeout:F success:F
 PAE: state: Held - portMode: Auto
 PAE: reAuthCount: 1 - rxRespId: 0
 BE: state: Idle - reqCount: 0 - idFromServer: 92

Switch# show vlan brief
VLAN ID  Name          State   STP ID  DSCP    Member ports
                                 (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ========================
1      default        ACTIVE  0      Disable eth-0-1(u) eth-0-2(u)
                                 eth-0-3(u) eth-0-4(u)
                                 eth-0-5(u) eth-0-6(u)
                                 eth-0-7(u) eth-0-8(u)
                                 eth-0-9(u) eth-0-10(u)
                                 eth-0-11(u) eth-0-12(u)
                                 eth-0-13(u) eth-0-14(u)
                                 eth-0-15(u) eth-0-16(u)
                                 eth-0-17(u) eth-0-18(u)
                                 eth-0-19(u) eth-0-20(u)
                                 eth-0-21(u) eth-0-24(u)
                                 eth-0-25(u) eth-0-26(u)
                                 eth-0-27(u) eth-0-28(u)
                                 eth-0-29(u) eth-0-30(u)
                                 eth-0-31(u) eth-0-32(u)
                                 eth-0-33(u) eth-0-34(u)
                                 eth-0-35(u) eth-0-36(u)
                                 eth-0-37(u) eth-0-38(u)
                                 eth-0-39(u) eth-0-40(u)
                                 eth-0-41(u) eth-0-42(u)
                                 eth-0-43(u) eth-0-44(u)
                                 eth-0-45(u) eth-0-46(u)
                                 eth-0-47(u) eth-0-48(u)
10     VLAN0010       ACTIVE  0      Disable
20     VLAN0020       ACTIVE  0      Disable eth-0-22(u)
  Client is authenticated
```

authorized:

```
Switch# show dot1x interface eth-0-22
802.1X info for interface eth-0-22
 portEnabled          : true
 portControl          : Auto
 portMode             : Port based
 portStatus           : Authorized
 Mac Auth bypass      : disabled
 reAuthenticate       : disabled
 reAuthPeriod         : 3600
 Max user number      : 255
 Current session number : 1
 Accept user number   : 1
 Reject user number   : 0
 Guest VLAN           : 20
 Assign VLAN          : N/A
 QuietPeriod          : 60
 ReqMax               : 2
 TxPeriod             : 30
 SuppTimeout          : 30
 ServerTimeout        : 30
 CD: adminControlledDirections : in
 CD: operControlledDirections  : in
 CD: bridgeDetected            : false
==========================================

session 1: 1 - 0011.0100.0001
```

```
----------------------------------------
 user name : admin
  abort:F fail:F start:F timeout:F success:T
  PAE: state: Authenticated - portMode: Auto
  PAE: reAuthCount: 0 - rxRespId: 0
  BE: state: Idle - reqCount: 0 - idFromServer: 207

Switch# show vlan brief
VLAN ID  Name         State  STP ID  DSCP   Member ports
                             (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ======= ========================
1     default       ACTIVE  0     Disable eth-0-1(u) eth-0-2(u)
                              eth-0-3(u) eth-0-4(u)
                              eth-0-5(u) eth-0-6(u)
                              eth-0-7(u) eth-0-8(u)
                              eth-0-9(u) eth-0-10(u)
                              eth-0-11(u) eth-0-12(u)
                              eth-0-13(u) eth-0-14(u)
                              eth-0-15(u) eth-0-16(u)
                              eth-0-17(u) eth-0-18(u)
                              eth-0-19(u) eth-0-20(u)
                              eth-0-21(u) eth-0-24(u)
                              eth-0-25(u) eth-0-26(u)
                              eth-0-27(u) eth-0-28(u)
                              eth-0-29(u) eth-0-30(u)
                              eth-0-31(u) eth-0-32(u)
                              eth-0-33(u) eth-0-34(u)
                              eth-0-35(u) eth-0-36(u)
                              eth-0-37(u) eth-0-38(u)
                              eth-0-39(u) eth-0-40(u)
                              eth-0-41(u) eth-0-42(u)
                              eth-0-43(u) eth-0-44(u)
                              eth-0-45(u) eth-0-46(u)
                              eth-0-47(u) eth-0-48(u)
10    VLAN0010      ACTIVE  0     Disable eth-0-22(u)
20    VLAN0020      ACTIVE  0     Disable

Switch# show dot1x
802.1X Port-Based Authentication Enabled
  RADIUS server address: 202.38.100.7:1812
  Next radius message ID: 0
Switch# show dot1x statistics
=====================================
802.1X statistics for interface eth-0-22
  EAPOL Frames Rx: 52 - EAPOL Frames Tx: 4270
  EAPOL Start Frames Rx: 18 - EAPOL Logoff Frames Rx: 2
  EAP Rsp/Id Frames Rx: 29 - EAP Response Frames Rx: 3
  EAP Req/Id Frames Tx: 3196 - EAP Request Frames Tx: 3
  Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
  EAPOL Last Frame Version Rx: 2 - EAPOL Last Frame Src: ae38.3288.f046
```

7.7.3 Application Cases

N/A

7.8 Configuring ARP Inspection

7.8.1 Overview

Function Introduction

ARP inspection is a security feature that validates ARP packets in a network. ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from some man-in-the-middle attacks. ARP inspection ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

Intercept all ARP requests and responses on untrusted ports.

Verify that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination.

Drop invalid ARP packets.

ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On entrusted interfaces, the switch forwards the packet only if it is valid.

Principle Description

Following is a brief description of terms and concepts used to describe the ARP Inspection:

- DHCP Snooping: DHCP snooping is a security feature that acts like a firewall between untrusted hosts and trusted DHCP servers. This feature builds and maintains the DHCP snooping binding database, which contains information about untrusted hosts with leased IP addresses.
- Address Resolution Protocol (ARP): ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A , but it does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address.

7.8.2 Configuration

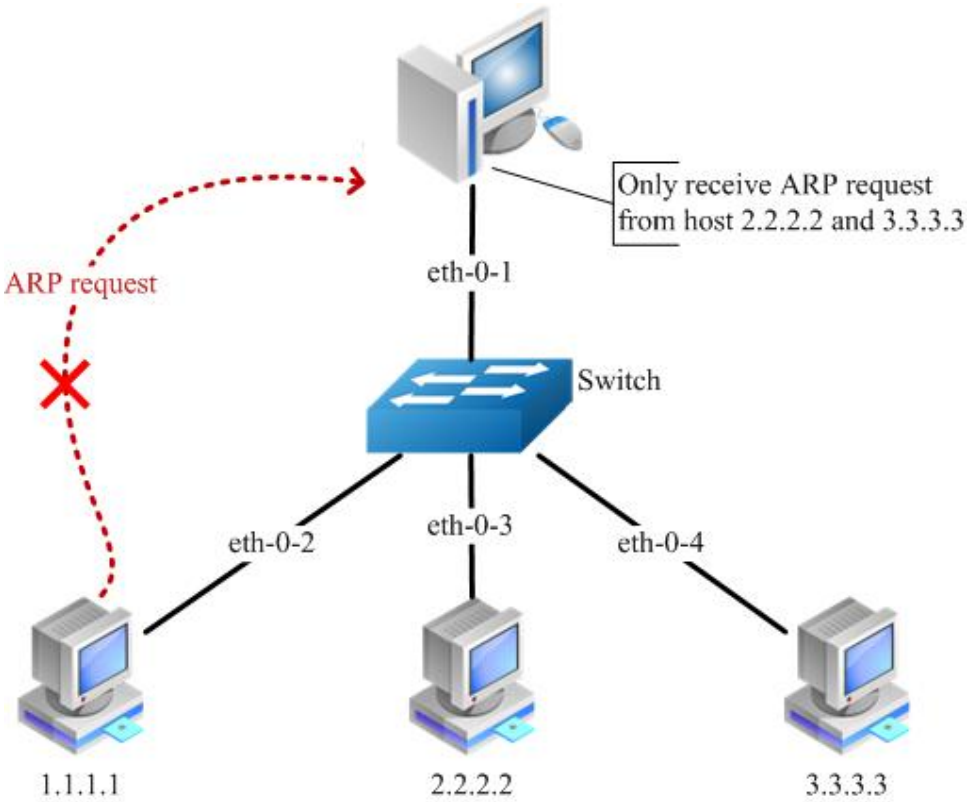


Figure 1-48 arp inspection

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 2
Switch(config-vlan)# exit
Switch(config)# exit
```

step 3 Enter the interface configure mode, add the interface into the vlan

```
Switch(config)# interface eth-0-1
Switch(config-if)# switchport access vlan 2
Switch(config-if)# exit

Switch(config)# interface eth-0-2
Switch(config-if)# switchport access vlan 2
Switch(config-if)# exit

Switch(config)# interface eth-0-3
Switch(config-if)# switchport access vlan 2
Switch(config-if)# exit

Switch(config)# interface eth-0-4
Switch(config-if)# switchport access vlan 2
Switch(config-if)# exit
```

step 4 Configure arp inspection

```
Switch(config)# interface eth-0-1
Switch(config-if)# ip arp inspection trust
Switch(config-if)# exit
Switch(config)# ip arp inspection vlan 2
Switch(config)# ip arp inspection validate src-mac ip dst-mac
```

step 5 Configure arp access list

```
Switch(config)# arp access-list test
Switch(config-arp-acl)# deny request ip host 1.1.1.1 mac any
Switch(config-arp-acl)# exit
Switch(config)# ip arp inspection filter test vlan 2
```

step 6 Exit the configure mode

```
Switch(config)# exit
```

step 7 Validation

Check the configuration of ARP Inspection on switch:

```
Switch# show ip arp inspection
Source Mac Validation      : Enabled
Destination Mac Validation  : Enabled
IP Address Validation      : Enabled
Vlan    Configuration    ACL Match    Static ACL
================================================================
2       enabled          test
Vlan    ACL Logging    DHCP Logging
================================================================
2       deny           deny
Vlan    Forwarded     Dropped    DHCP Drops    ACL Drops
================================================================
2       0             0          0             0
Vlan    DHCP Permits    ACL Permits    Source MAC Failures
================================================================
2       0               0              0
Vlan   Dest MAC Failures   IP Validation Failures   Invalid Protocol Data
================================================================
2      0                 0                    0
```

Show the log information of ARP Inspection on switch:

```
Switch# show ip arp inspection log
Total Log Buffer Size : 32
Syslog rate : 5 entries per 1 seconds.
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
```

```
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
```

7.8.3 Application Cases

N/A

7.9 Configuring DHCP Snooping

7.9.1 Overview

Function Introduction

DHCP snooping is a security feature that acts like a firewall between untrusted hosts and trusted DHCP servers.

The DHCP snooping feature performs the following activities:

- Validate DHCP messages received from untrusted sources and filters out invalid messages.
- Build and maintain the DHCP snooping binding database, which contains information about untrusted hosts with leased IP addresses.
- Utilize the DHCP snooping binding database to validate subsequent requests from untrusted hosts.

Other security features, such as dynamic ARP inspection (DAI), also use information stored in the DHCP snooping binding database. DHCP snooping is enabled on a per-VLAN basis. By default, the feature is inactive on all VLANs. You can enable the feature on a single VLAN or a range of VLANs. The DHCP snooping feature is implemented in software basis. All DHCP messages are intercepted in the BAY and directed to the CPU for processing.

Principle Description

N/A

7.9.2 Configuration

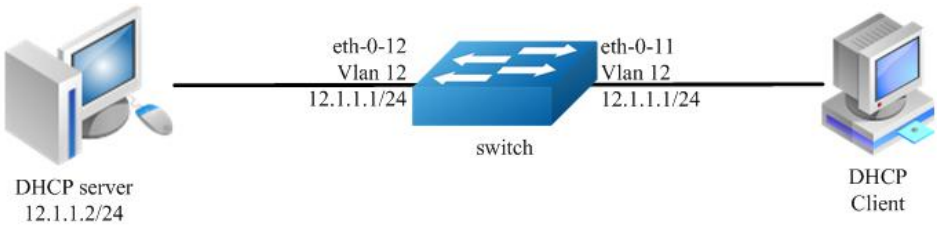


Figure 1-49 dhcp snooping

This figure is the networking topology for testing DHCP snooping functions. We need two Linux boxes and one switch to construct the test bed.

- Computer A is used as a DHCP server.
- Computer B is used as a DHCP client.
- Switch is used as a DHCP Snooping box.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 12
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode, add the interface into the vlan

```
Switch(config)# interface eth-0-12
Switch(config-if)# switchport
Switch(config-if)# switchport access vlan 12
Switch(config-if)# dhcp snooping trust
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-11
Switch(config-if)# switchport
Switch(config-if)# switchport access vlan 12
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface vlan 12
Switch(config-if)# ip address 12.1.1.1/24
Switch(config-if)# exit
```

step 4 Set DHCP attributes

```
Switch(config)# dhcp snooping verify mac-address
Switch(config)# service dhcp enable
Switch(config)# dhcp snooping
Switch(config)# dhcp snooping vlan 12
```

step 5 Exit the configure mode

```
Switch(config)# exit
```

step 6 Validation

Check the interface configuration.

```
Switch(config)# show running-config interface eth-0-12
!
interface eth-0-12
 dhcp snooping trust
 switchport access vlan 12
!

Switch(config)# show running-config interface eth-0-11
!
interface eth-0-11
 switchport access vlan 12
!
```

Check the dhcp service status.

```
Switch# show services
Networking services configuration:
Service Name       Status
============================================================
dhcp               enable
```

Print dhcp snooping configuration to check current configuration.

```
Switch# show dhcp snooping config
dhcp snooping service: enabled
dhcp snooping switch: enabled
Verification of hwaddr field: enabled
Insertion of relay agent information (option 82): disable
```

```
Relay agent information (option 82) on untrusted port: not allowed
dhcp snooping vlan 12
```

Show dhcp snooping statistics.

```
Switch# show dhcp snooping statistics
DHCP snooping statistics:
============================================================
DHCP packets                 17
BOOTP packets                 0
Packets forwarded              30
Packets invalid              0
Packets MAC address verify failed    0
Packets dropped               0
```

Show dhcp snooping binding information.

```
Switch# show dhcp snooping binding all
DHCP snooping binding table:
VLAN MAC Address   Interface  Lease(s)  IP Address
============================================================
12  0016.76a1.7ed9 eth-0-11  691190    12.1.1.65
```

7.9.3 Application Cases

N/A

7.10 Configuring IP source guard

7.10.1 Overview

Function Introduction

IP source guard prevents IP spoofing by allowing only the IP addresses that are obtained through DHCP snooping on a particular port. Initially, all IP traffic on the port is blocked except for the DHCP packets that are captured by DHCP snooping. When a client receives a valid IP address from the DHCP server, an access control list (ACL) is installed on the port that permits the traffic from the IP address. This process restricts the client IP traffic to those source IP addresses that are obtained from the DHCP server; any IP traffic with a source IP address other than that in the ACL's permit list is filtered out. This filtering limits the ability of a host to attack the network by claiming a neighbor host's IP address.

IP source guard uses source IP address filtering, which filters the IP traffic that is based on its source IP address. Only the IP traffic with a source IP address that matches the IP source binding entry is permitted. A port's IP source address filter is changed when a new DHCP-snooping binding entry for a port is created or deleted. The port ACL is modified and reapplied in the hardware to reflect the IP source binding change. By default, if you enable IP source guard without any DHCP-snooping bindings on the port, a default ACL that denies all IP traffic is installed on the port. When you disable IP source guard, any IP source filter ACL is removed from the port.

Also IP source guard can use source IP and MAC address Filtering. When IP source guard is enabled with this option, IP traffic is filtered based on the source IP and Mac addresses. The switch forwards traffic only when the source IP and MAC addresses match an entry in the IP source binding table. If not, the switch drops all other types of packets except DHCP packet.

The switch also supports to have IP, MAC and VLAN Filtering. When IP source guard is enabled with this option, IP traffic is filtered cased on the source IP and MAC addresses. The switch forwards traffic only when the source IP, MAC addresses and VLAN match an entry in the IP source binding table.

Principle Description

The following terms and concepts are used to describe the IP source guard:

- Dynamic Host Configuration Protocol (DHCP): Dynamic Host Configuration Protocol (DHCP) is a client/server protocol that automatically provides an Internet Protocol (IP) host with its IP address and other related configuration information such as the subnet mask and default gateway.
- DHCP Snooping: DHCP snooping is a security feature that acts like a firewall between untrusted hosts and trusted DHCP servers. This feature builds and maintains the DHCP snooping binding database, which contains information about untrusted hosts with leased IP addresses.
- ACL: Access control list.

7. 10. 2 Configuration

Configure ip source guard

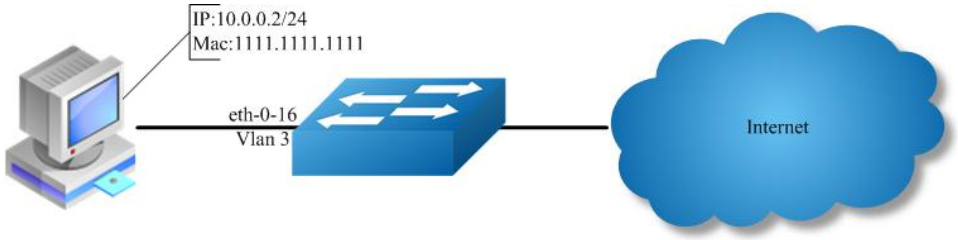


Figure 1-50 ip source guard

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 3
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode and set the attributes

```
Switch(config)# interface eth-0-16
Switch(config-if)# switchport
Switch(config-if)# no shutdown
Switch(config-if)# switchport access vlan 3
Switch(config-if)# exit
```

step 4 Add IP source guard entries

```
Switch(config)# ip source maximal binding number per-port 15
Switch(config)# ip source binding mac 1111.1111.1111 vlan 3 ip 10.0.0.2 interface eth-0-16
```

step 5 Enable IP source guard on the interface

```
Switch(config)# interface eth-0-16
Switch(config-if)# ip verify source ip
Switch(config-if)# exit
```

step 6 Exit the configure mode

```
Switch(config)# exit
```

step 7 Validation

```
Switch#show running-config interface eth-0-16
!
interface eth-0-16
 ip verify source ip
 switchport access vlan 3
```

Remove ip source guard entries

Remove by entry:

```
Switch(config)# no ip source binding mac 1111.1111.1111 vlan 3 ip 10.0.0.2 interface eth-0-16
```

Remove by interface:

```
Switch(config)# no ip source binding entries interface eth-0-16
```

Remove by vlan:

```
Switch(config)# no ip source binding entries vlan 3
```

Remove all:

```
Switch(config)# no ip source binding entries
```

7.10.3 Application Cases

N/A

7.11 Configuring Private-vlan

7.11.1 Overview

Function Introduction

Private-vlan a security feature which is used to prevent from direct l2 communication among a set of ports in a vlan.

It can provide a safer and more flexible network solutions by isolating the ports which in the same VLAN.

Principle Description

N/A

7.11.2 Configuration

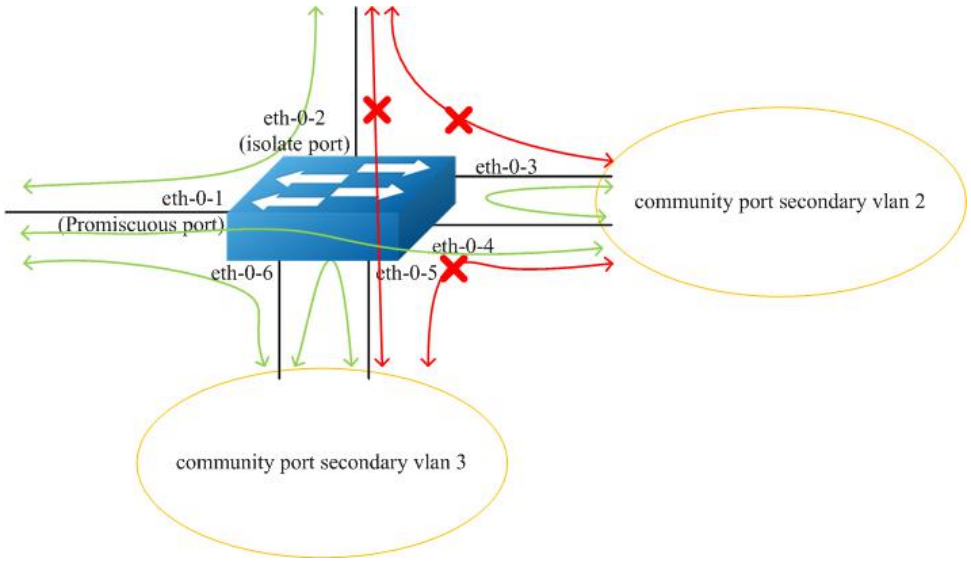


Figure 1-51 private vlan

As the figure above shows:

- All ports are in a same primary vlan.
- Port 1 is promiscuous port; it can communicate with all other ports.
- Port 2 is isolate port; it cannot communicate with all other ports except for the promiscuous port (port 1).
- Port 3 and port 4 are community ports in secondary vlan 2; they can communicate with each other. They cannot communicate with all other ports except for the promiscuous port.
- Port 5 and port6 are community ports in secondary vlan 3; they can communicate with each other. They cannot communicate with all other ports except for the promiscuous port.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create vlan

```
Switch (config)# vlan database
Switch (config-vlan)# vlan 2
Switch (config-vlan)# quit
```

step 3 Enter the interface configure mode and set the attributes

Promiscuous port: promiscuous port in pvlan can communicate with any other ports in this pvlan

```
Switch (config)# interface eth-0-1
Switch (config-if)# switchport mode private-vlan promiscuous
Switch (config-if)# switchport private-vlan 2
Switch (config-if)# quit
```

Isolate port: isolate port in pvlan can only communicate with promiscuous port in this pvlan

```
Switch (config)# interface eth-0-2
Switch (config-if)# switchport mode private-vlan host
Switch (config-if)# switchport private-vlan 2 isolate
Switch (config-if)# quit
```

Community port: community port in pvlan can communicate with promiscuous port and community ports with same community-vlan id in this pvlan

```
Switch (config)# interface eth-0-3
Switch (config-if)# switchport mode private-vlan host
Switch (config-if)# switchport private-vlan 2 community-vlan 2
Switch (config-if)# quit

Switch (config)# interface eth-0-4
Switch (config-if)# switchport mode private-vlan host
Switch (config-if)# switchport private-vlan 2 community-vlan 2
Switch (config-if)# quit

Switch (config)# interface eth-0-5
Switch (config-if)# switchport mode private-vlan host
Switch (config-if)# switchport private-vlan 2 community-vlan 3
Switch (config-if)# quit

Switch (config)# interface eth-0-6
Switch (config-if)# switchport mode private-vlan host
Switch (config-if)# switchport private-vlan 2 community-vlan 3
Switch (config-if)# quit
```

step 4 Exit the configure mode

```
Switch(config)# exit
```

step 5 Validation

The result of show private-vlan is as follows:

```
switch # show private-vlan
Primary  Secondary Type        Ports
------------------------------------------------------------------------------
2      N/A     promiscuous   eth-0-1
2      N/A     isloate       eth-0-2
2      2       community     eth-0-3  eth-0-4
2      3       community     eth-0-5  eth-0-6
```

7.11.3 Application Cases

N/A

7.12 Configuring AAA

7.12.1 Overview

Function Introduction

Authentication verifies users before they are allowed access to the network and network services. System can use AAA authentication methods and Non-AAA authentication methods. RADIUS Authentication is one of AAA authentication methods. RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS is widely used protocol in network environments. It is commonly used for embedded network devices such as routers, modem servers, switches, etc. RADIUS clients run on support routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information.

Principle Description

N/A

7.12.2 Configuration

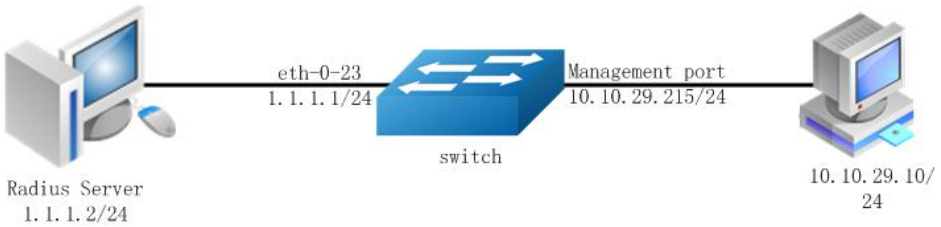


Figure 1-52 private vlan

The figure above is the networking topology for RADIUS authentication functions. We need one Switch and two computers for this test.

One computer as RADIUS server, it ip address of the eth0 interface is 1.1.1.2/24.

Switch has RADIUS authentication function. The ip address of interface eth-0-23 is 1.1.1.1/24. The management ip address of switch is 10.10.29.215, management port is connected the PC for test login, PC's ip address is 10.10.29.10.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable AAA

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication login radius-login radius local
```

step 3 Configure Radius server

```
Switch(config)# radius-server host 1.1.1.2 auth-port 1819 key keyname
Switch(config)# radius-server host 2001:1000::1 auth-port 1819 key keyname
```

step 4 Configure a layer 3 interface and set ip address

```
Switch(config)# interface eth-0-23
Switch(config-if)# no switchport
Switch(config-if)# ip address 1.1.1.1/24
Switch(config-if)# quit
```

step 5 set authentication mode

```
Switch(config)# line vty 0 7
Switch(config-line)#login authentication radius-login
Switch(config-line)#privilege level 4
Switch(config-line)#no line-password
```

step 6 Exit the configure mode

```
Switch(config-line)# end
```

step 7 Validation

You can use command show authentication status in switch:

```
Switch# show aaa status
aaa status:
   Authentication enable
```

You can use command show keys in switch:

```
Switch# show aaa method-lists authentication
authen queue=AAA_ML_AUTHEN_LOGIN
   Name = default  state = ALIVE :  local
Name = radius-login  state = ALIVE :  radius  local
```

Telnet output:

Figure 1-53 Telnet connecting test

NOTE: Don't forget to turn RADIUS authentication feature on.

Make sure the cables is linked correctly You can use command to check log messages if Switch can't do RADIUS authentication:

```
Switch# show logging buffer
```

7.12.3 Application Cases

Radius server configuration (Using WinRadius for example)

Set ip address for PC:

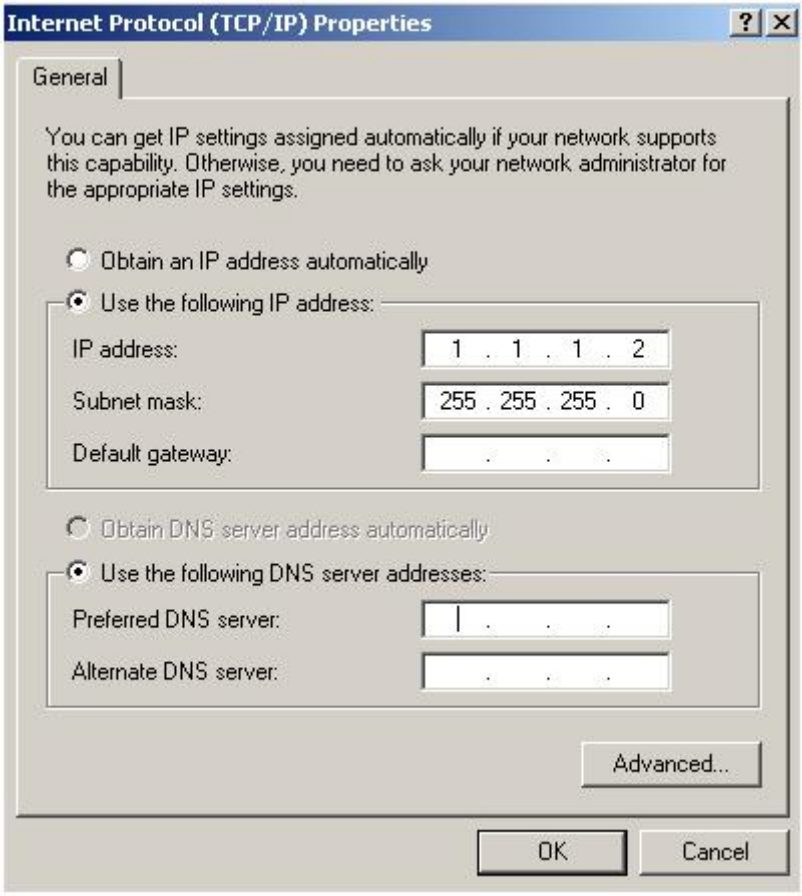


Figure 1-54 Set IP address for PC

Connectivity test between server and switch:

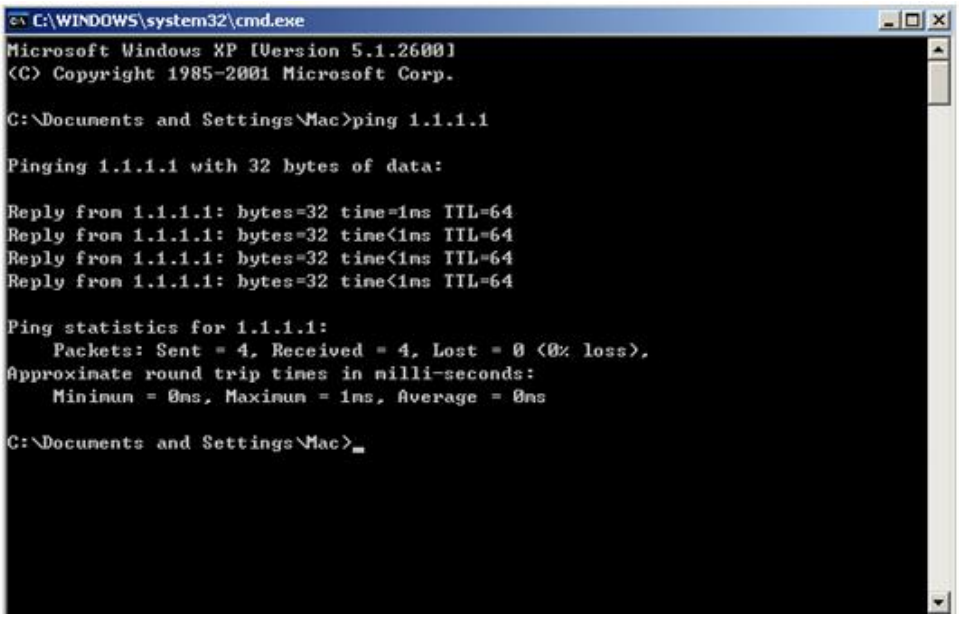


Figure 1-55 Connectivity test

Open winRadius:

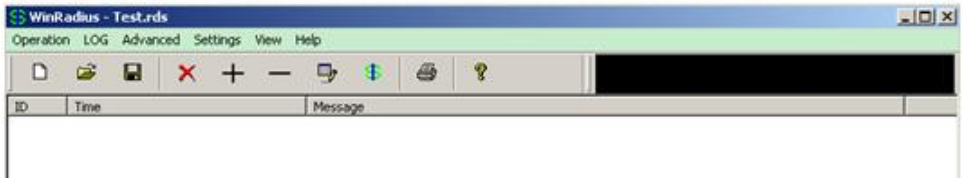


Figure 1-56 WinRadius

Configurations for winRadius:

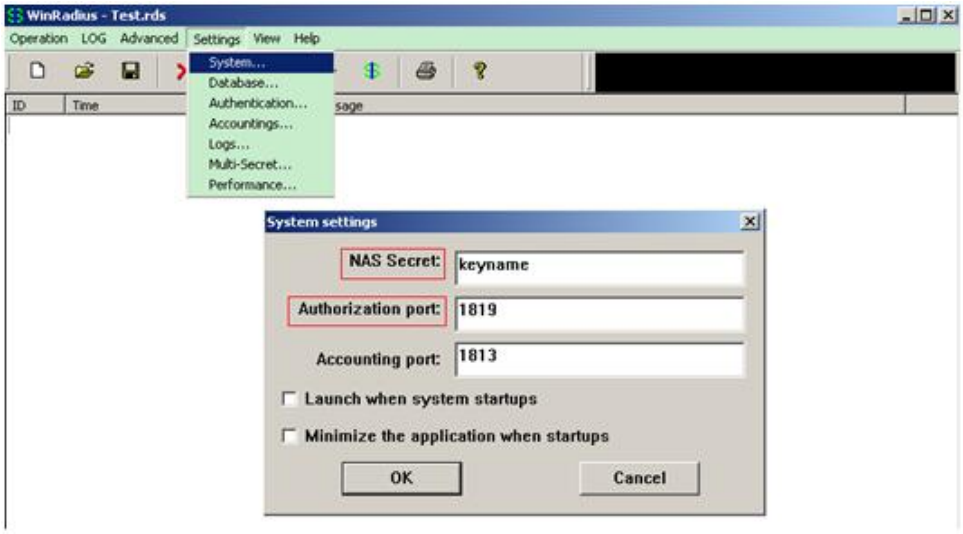


Figure 1-57 WinRadius

Add user and password:

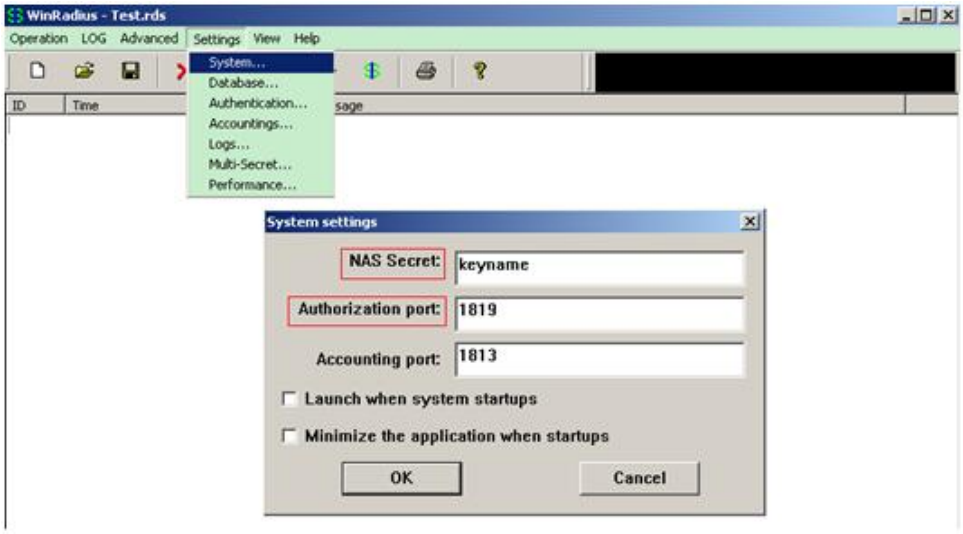


Figure 1-58 Add user and password

Connectivity test between client and switch:

```
C:\Documents and Settings\mac>ping 10.10.29.215

Pinging 10.10.29.215 with 32 bytes of data:

Reply from 10.10.29.215: bytes=32 time<1ms TTL=63
Reply from 10.10.29.215: bytes=32 time<1ms TTL=63
Reply from 10.10.29.215: bytes=32 time<1ms TTL=63
Reply from 10.10.29.215: bytes=32 time<1ms TTL=63

Ping statistics for 10.10.29.215:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figure 1-59 Connectivity test

7.13 Configuring TACACS+

7.13.1 Overview

Function Introduction

Authentication verifies users before they are allowed access to the network and network services. System can use AAA authentication methods and Non-AAA authentication methods. TACACS+ Authentication is one of AAA authentication methods. TACACS+ is a distributed client/server system that secures networks against unauthorized access. TACACS+ is widely used protocol in network environments. It is commonly used for embedded network devices such as routers, modem servers, switches, etc. TACACS+ clients run on support routers and switches. Clients send authentication requests to a central TACACS+ server, which contains all user authentication and network service access information.

Principle Description

N/A

7.13.2 Configuration

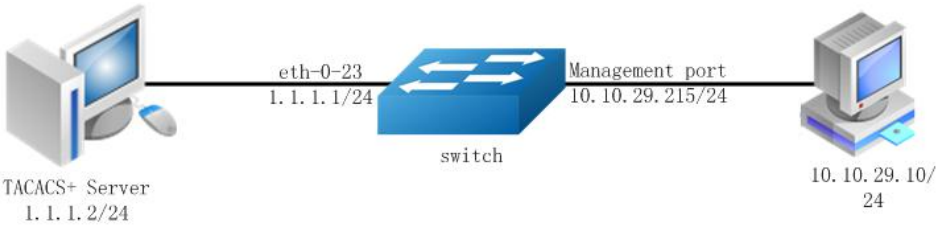


Figure 1-60 TACACS+

The figure above is the networking topology for TACACS+ authentication functions. We need one Switch and two computers for this test. One computer as TACACS+ server, it ip address of the eth0 interface is 1.1.1.2/24. Switch has TACACS+ authentication function. The ip address of interface eth-0-23 is 1.1.1.1/24. The management ip address of switch is 10.10.29.215, management port (only in-band management port) is connected the PC for test login, PC's ip address is 10.10.29.10

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable AAA

```
Switch# configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa authentication login tac-login tacacs-plus local
Switch(config)# aaa authorization exec default tacacs-plus
Switch(config)# aaa accounting exec default start-stop tacacs-plus
Switch(config)# aaa accounting commands default tacacs-plus
```

step 3 Configure tacacs+ server

```
Switch(config)# tacacs-server host 1.1.1.2 port 123 key keyname
```

step 4 Configure a layer 3 interface and set ip address

```
Switch(config)# interface eth-0-23
Switch(config-if)# no switchport
Switch(config-if)# ip address 1.1.1.1/24
Switch(config-if)# quit
```

step 5 set authentication mode

```
Switch(config)# line vty 0 7
Switch(config-line)#login authentication tac-login
Switch(config-line)#privilege level 4
Switch(config-line)#no line-password
```

step 6 Exit the configure mode

```
Switch(config-line)# end
```

step 7 Validation

You can use command show authentication status in switch:

```
Switch# show aaa status
aaa stats:
   Authentication enable
```

You can use command show keys in switch:

```
Switch# show aaa method-lists authentication
authen queue=AAA_ML_AUTHEN_LOGIN
  Name = default  state = ALIVE :  local
Name = tac-login  state = ALIVE :  tacacs-plus local
```

Telnet output:

Figure 1-61 Telnet connecting test

7.13.3 Application Cases

Radius server configuration

Download TACACS+ server code, DEVEL.201105261843.tar.bz2.

Build the TACACS+ server.

Add username and password in configure file.

```
#!../obj.linux-2.6.9-89.29.1.elsmp-x86_64/tac_plus
id = spawnd {
     listen = { port = 49 }
```

```
    spawn = {
        instances min = 1
        instances max = 10
    }
    background = no
}
    user = aaa {
        password = clear bbb
        member = guest
    }
```

Run TACACS+ server:

```
[disciple: ~]$ ./tac_plus ./tac_plus.cfg.in -d 1
```

Use Ping command for test on PC:

Figure 1-62 Connectivity test

7.14 Configuring Port Isolate

7.14.1 Overview

Function Introduction

Port-isolation a security feature which is used to prevent from direct l2/l3 communication among a set of ports.

It can provide a safer and more flexible network solutions by isolating the ports which in the same VLAN.

Generally, it's used as an access device for user isolation.

Principle Description

N/A

7. 14. 2 Configuration

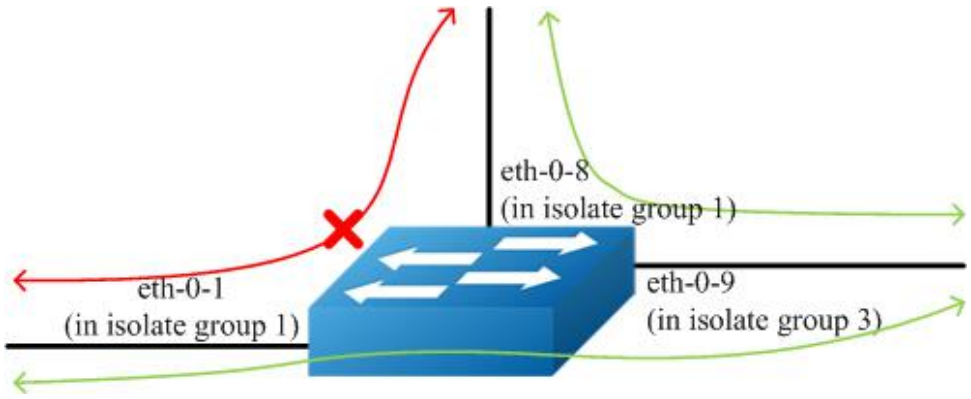


Figure 1-63 Port Isolate

The figure above is the basic topology for port-isolate.

Port 1 and port 8 are in the same isolate group 1, they are isolated. So port1 can not communicate with port 8. Port 9 is in a different isolate group 3, so port 9 can communicate with port 1 and port 8.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the port isolate mode globally

The mode "l2" means only layer 2 packets are isolated. The mode "all" means all packet are isolated include the packets forward according to layer 3 routes.

```
Switch(config)# port-isolate mode l2
```

step 3 Enter the interface configure mode and set isolate group

```
Switch(config-if)# interface eth-0-1
Switch(config-if)# port-isolate group 1
Switch(config-if)# exit

Switch(config)# interface eth-0-8
Switch(config-if)# port-isolate group 1
Switch(config-if)# exit

Switch(config)# interface eth-0-9
Switch(config-if)# port-isolate group 3
Switch(config-if)# exit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

Use the following command to display the port isolate groups:

```
switch# show port-isolate
------------------------------------------------------------------
 Port Isolate Groups:
------------------------------------------------------------------
 Groups ID: 1
eth-0-1, eth-0-8
------------------------------------------------------------------
```

```
Groups ID: 3
eth-0-9
```

7.14.3 Application Cases

N/A

7.15 Configuring DDoS

7.15.1 Overview

Function Introduction

A denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a computer resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the concerted efforts of a person or people to prevent an Internet site or service from functioning efficiently or at all, temporarily or indefinitely. Perpetrators of DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root name servers. The term is generally used with regards to computer networks, but is not limited to this field, for example, it is also used in reference to CPU resource management.

DDoS prevent is a feature which can protect our switch from follow kinds of denial-of-service attack and intercept the attack packets.

The flowing types are supported:

- ICMP flood: attackers overwhelm the victim with ICMP packets.
- Smurf attack: attackers flood a target system via spoofed broadcast ping messages.
- SYN flood: attackers send a succession of SYN requests to a target's system.
- UDP flood: attackers send a large number of UDP packets to random ports on a remote host.
- Fraggle attack:attackers send a large number of UDP echo traffic to IP broadcast addresses, all fake source address.
- Small-packet: attackers send a large number of small packets to the system utill the resource exhaust.
- bad mac intercept: attackers send packets with same source and destination MAC address.
- bad ip equal: attackers send packets with same source and destination IP address.

Principle Description

N/A

7.15.2 Configuration

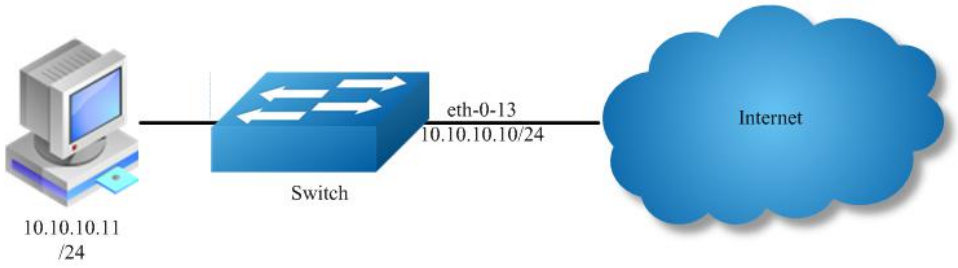


Figure 1-64 Topology for DDoS test

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set DDoS

Enable ICMP flood intercept and set the max received ICMP packet rate 100 packets per-second

```
Switch(config)# ip icmp intercept maxcount 100
```

Enable UDP flood intercept and set the max received UDP packet rate 100 packets per-second

```
Switch(config)# ip udp intercept maxcount 100
```

Enable Smurf attack intercept

```
Switch(config)# ip smurf intercept
```

Enable SYN flood intercept and set the max received SYN packet rate 100 packets per-second

```
Switch(config)# ip tcp intercept maxcount 100
```

Enable Fraggle attack intercept

```
Switch(config)# ip fraggle intercept
```

Enable Small-packet attack intercept and set the received packet length is be more than or equal to 32

```
Switch(config)# ip small-packet intercept maxlength 32
```

Enable packet source IP equals destination IP intercept

```
Switch(config)# ip ipeq intercept
```

Enable packet source MAC equals destination MAC intercept

```
Switch(config)# ip maceq intercept
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show ip-intercept config
Current DDoS Prevent configuration:
============================================================
ICMP Flood Intercept          :Enable  Maxcount:500
UDP Flood Intercept           :Enable  Maxcount:500
SYN Flood Intercept           :Enable  Maxcount:500
Small-packet Attack Intercept   :Enable  Packet Length:45
Smurf Attack Intercept        :Enable
Fraggle Attack Intercept      :Enable
MAC Equal Intercept           :Enable
IP Equal Intercept            :Enable
Switch# show ip-intercept statistics
Current DDoS Prevent statistics:
============================================================
Resist Small-packet Attack packets number   : 1730
Resist ICMP Flood packets number          : 0
Resist SYN Flood packets number           : 0
Resist Fraggle Attack packets number      : 0
Resist UDP Flood packets number           : 0

Current DDoS Prevent mgmt-if statistics:
============================================================
Resist ICMP Flood packets number          : 0
Resist SYN Flood packets number           : 0
Resist Fraggle Attack packets number      : 0
Resist UDP Flood packets number           : 0
```

7.15.3 Application Cases

N/A

7.16 Configuring Key Chain

7.16.1 Overview

Function Introduction

Keychain is a common method of authentication to configure shared secrets on all the entities, which exchange secrets such as keys before establishing trust with each other. Routing protocols and network applications often use this authentication to enhance security while communicating with peers.

The keychain by itself has no relevance; therefore, it must be used by an application that needs to communicate by using the keys (for authentication) with its peers. The keychain provides a secure mechanism to handle the keys and rollover based on the lifetime.

If you are using keys as the security method, you must specify the lifetime for the keys and change the keys on a regular basis when they expire. To maintain stability, each party must be able to store and use more than one key for an application at the same time. A keychain is a sequence of keys that are collectively managed for authenticating the same peer, peer group, or both. Keychain groups a sequence of keys together under a keychain and associates each key in the keychain with a lifetime.

Principle Description

N/A

7.16.2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create key chain and set key

```
Switch(config)# key chain test
Switch(config-keychain)# key 1
Switch(config-keychain-key)# key-string ##test_keystring_1##
Switch(config-keychain-key)# accept-lifetime 0:0:1 1 jan 2012 infinite
Switch(config-keychain)# key 2
Switch(config-keychain-key)# key-string ##test_keystring_2##
Switch(config-keychain-key)# send-lifetime 0:0:1 2 jan 2012 infinite
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

To display the keychain configuration, use the command show key chain in the privileged EXEC mode"

```
Switch # show key chain
key chain test:
  key 1 -- text "key-string ##test_keystring_1##"
    accept-lifetime <00:00:01 Jan 01 2012> - <infinite>
    send-lifetime <always valid> - <always valid> [valid now]
  key 2 -- text "key-string ##test_keystring_2##"
    accept-lifetime <always valid> - <always valid> [valid now]
    send-lifetime <00:00:01 Jan 02 2012> - <infinite>
```

7.16.3 Application Cases

N/A

7.17 Configuring Port-Block

7.17.1 Overview

Function Introduction

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or unprotected) from flooding unknown unicast or multicast packets to other ports.

Principle Description

N/A

7.17.2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and block unknown unicast

```
Switch(config)# interface eth-0-1
Switch(config-if)# port-block unknown-unicast
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

To display the port-block configuration, use the command show port-block in the privileged EXEC mode:

```
Switch # show port-block interface eth-0-1
 Known unicast blocked: Enabled
 Known multicast blocked: Disabled
 Unknown unicast blocked: Disabled
 Unknown multicast blocked: Disabled
 Broadcast blocked: Disabled
```

7.17.3 Application Cases

N/A

Chapter 8 Device Management Configuration Guide

8.1 Configuring STM

8.1.1 Overview

Function Introduction

Switch Table Management (STM) is used to configure system resources in the switch to optimize support for specific features, depending on how the switch is used in the network.

You can select a profile to provide maximum system usage for some functions; for example, use the default profile to balance resources and use vlan profile to obtain max MAC entries.

To allocate ternary content addressable memory (TCAM) resources for different usages, the switch STM profile prioritize system resources to optimize support for certain features. You can select STM templates to optimize these features:

- layer2: The VLAN template supports the maximum number of unicast MAC addresses. It would typically be selected for a Layer 2 switch.
- layer3: The routing template maximizes system resources for unicast routing, typically required for a router or aggregator in the center of a network.
- ipv6: The ipv6 template, support the ipv6 functions.
- default: The default template gives balance to all functions.

NOTE: When users configured a profile mode which is not exist in the next reboot image, then default hardware configure will be used when system up with the next image. The hardware configure may be different from the default profile.

Principle Description

N/A

8.1.2 Configuration

Follow these guidelines when selecting and configuring STM profiles.

You must reload the switch for the configuration to take effect.

Use the "stm prefer layer2" global configuration command only on switches intended for Layer 2 switching with no routing.

Do not use the layer3 profile if you do not have routing enabled on your switch. The stm prefer layer3 global configuration command prevents other features from using the memory allocated to IPv4 unicast routing in the routing profile.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set STM profile(use layer3 for example)

```
Switch(config)# stm prefer layer3
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

This is an example of an output display for route template:

```
Switch# show stm prefer
Current profile is :default
  number of vlan instance              : 1/4094
  number of unicast mac address        : 0/65536
  number of multicast mac address      : 0/2048
  number of blackhole mac address      : 0/128
```

```
number of max applied vlan mapping         : 0/1024
number of bfd sessions                     : 0/128
number of CFM loacl&remote MEPs            : 0/1024
number of CFM lm                           : 0/256
number of CFM lck                          : 0/24
number of G8031 groups                     : 0/256
number of G8032 rings                      : 0/256
number of G8032 member ports               : 0/256
number of mac based vlan class             : 0/512
number of ipv4 based vlan class            : 0/512
number of ipv6 based vlan class            : 0/0
number of dot1x mac based                  : 0/2048
number of unicast ipv4 host routes         : 0/4096
number of unicast ipv4 indirect routes     : 0/8192
number of unicast ipv4 policy based routes  : 0/16
number of unicast ipv6 host routes         : 0/0
number of unicast ipv6 indirect routes     : 0/0
number of unicast ecmp groups              : 0/240
number of unicast ip tunnel peers          : 0/8
number of multicast ipv4 routes            : 0/1023
number of mvr entries                      : 0/511
number of mvr6 entries                     : 0/0
number of multicast ipv6 routes            : 0/0
number of ipv4 source guard entries        : 0/1024
number of ingress port acl flow entries    : 0/2035
number of ingress vlan acl flow entries    : 0/255
number of egress port acl flow entries     : 0/255
number of ingress port qos flow entries    : 9/2043
number of ingress port acl ipv6 flow entries : 0/0
number of ingress vlan acl ipv6 flow entries : 0/0
number of egress port acl ipv6 flow entries  : 0/0
number of ingress port qos ipv6 flow entries : 0/0
number of link aggregation (static & lacp)   : 0/55
number of ipfix cache                      : 0/16384
```

The profile stored for use after the next reload is the layer3 profile.

step 5 Reboot the device

```
Switch# reload
```

8.1.3 Application Cases

N/A

8.2 Configuring syslog

8.2.1 Overview

Function Introduction

The system message logging software can save messages in a log file or direct the messages to other devices. The system message logging facility has these features:

- Provides you with logging information for monitoring and troubleshooting.
- Allows you to select the types of logging information that is captured.
- Allows you to select the destination of the captured logging information.

By default, the switch logs normal but significant system messages to its internal buffer and sends these messages to the system console. You can specify which system messages should be saved based on the type of the severity level. The messages are time-stamped to enhance real-time debugging and management.

You can access the logged system messages using the switch command-line interface (CLI) or by saving them to a properly configured log server. The switch software saves the log messages in an internal buffer that can store up to 1000 messages. You can monitor the system messages remotely by accessing the switch through Telnet or the console port, or by viewing the logs on a log server.

Principle Description

Terminology:

Terminology	Description
Logging	Current logging configuration
Show	Show logging configuration
Levels	Severity level information
Enable	Enable write log to local file
Disable	Disable write log to local file

System Message Log Facility Types:

Facility Name	Definition
kern	kernel messages
user	random user-level messages
mail	mail system
daemon	system daemons
auth	security/authorization messages
syslog	messages generated internally by syslogd
lpr	line printer subsystem
news	network news subsystem
uucp	UUCP subsystem
cron	clock daemon
authpriv	security/authorization messages (private)
ftp	ftp daemon

Severity Level Definitions:

Severity Level	Definition
emergency	system is unusable
alert	action must be taken immediately
critical	critical conditions
error	error conditions
warning	warning conditions
notice	normal but significant condition
information	Informational
debug	debug-level messages

8.2.2 Configuration

Configuring Logging server

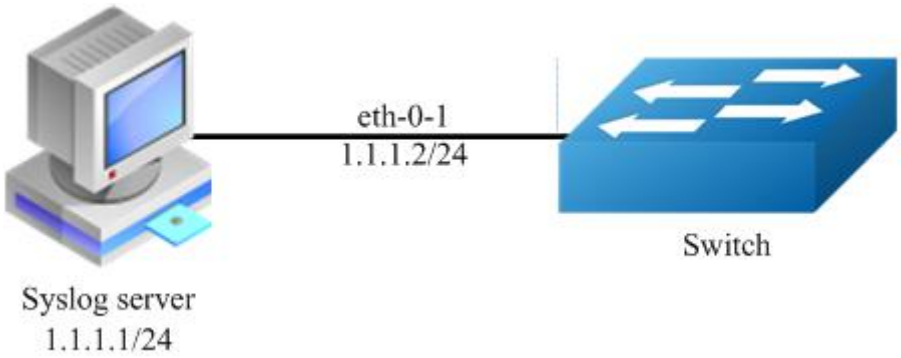


Figure 1-65 syslog server

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable logging server and set the attributes

```
Switch(config)# logging server enable
Switch(config)# logging server address 1.1.1.1
Switch(config)# logging server address 2001:1000::2
Switch(config)# logging server severity debug
Switch(config)# logging server facility mail
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show logging
Current logging configuration:
============================================================
logging buffer 500
logging timestamp bsd
logging file enable
logging level file warning
logging level module debug
logging server enable
logging server severity debug
logging server facility mail
logging server address 1.1.1.1
logging server address 2001:1000::2
logging alarm-trap enable
logging alarm-trap level middle
logging merge enable
logging merge fifo-size 1024
logging merge timeout 10
logging operate disable
```

Configuring Logging Buffer Size

By default, the number of messages to log to the logging buffer is 500. If desired, you can set the number between 10 and 1000.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the logging Buffer Size

```
Switch(config)# logging buffer 700
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show logging
Current logging configuration:
============================================================
logging buffer 700
logging timestamp bsd
logging file enable
logging level file warning
logging level module debug
logging server enable
logging server severity debug
logging server facility mail
logging server address 1.1.1.1
logging alarm-trap enable
logging alarm-trap level middle
logging merge enable
logging merge fifo-size 1024
logging merge timeout 10
logging operate disable
```

The following is the information of logging server:

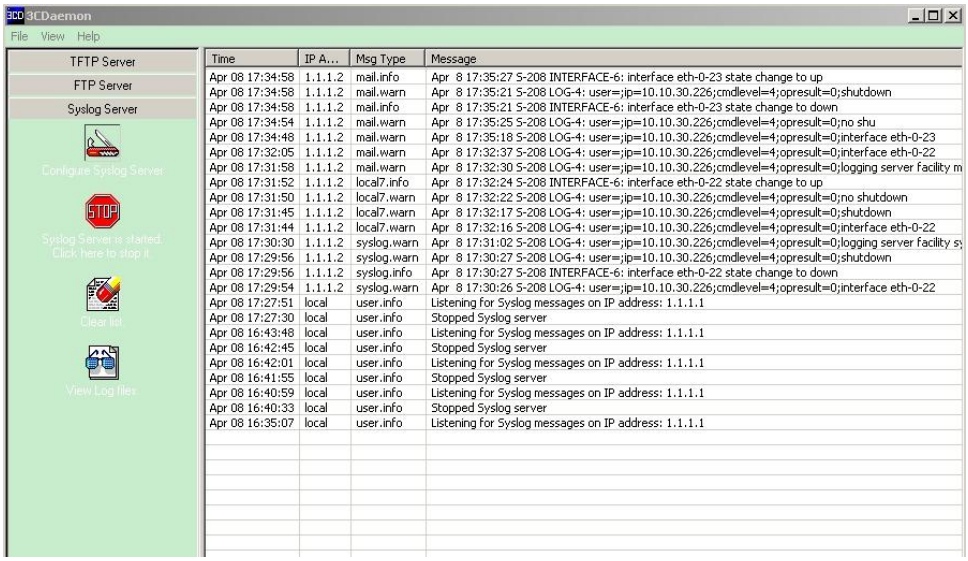


Figure 1-66 syslog on server

NOTE: You can use command to check showing Logging Information. When configuring the syslog Servers, make sure the cables is linked correctly and two computers can ping each other. Before you can send the system log messages to a log server, you must configure Syslog Software, at the end you can see the log from your software.

8. 2. 3 **Application Cases**

N/A

8.3 Configuring mirror

8.3.1 Overview

Function Introduction

Mirror function can send one or more copies of packets which are passing through the ports/vlans or sending and receiving by CPU to one or more specified destination ports. It can also send the copies to the CPU and keep in memory or flash files.

The copies of the packets are used for network analyze. The mirror function does not affect the original network traffic.

Principle Description

The following describes concepts and terminology associated with mirror configuration:

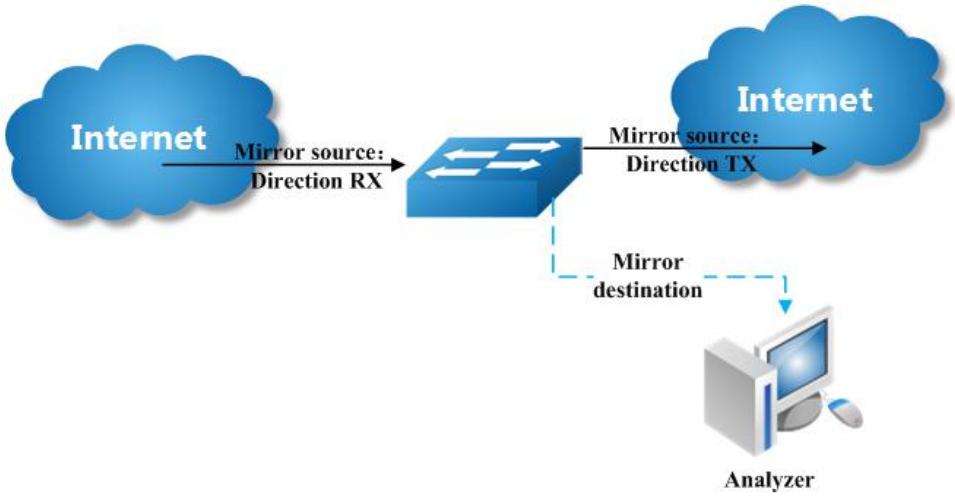


Figure 1-67 Mirror

1.Mirror session

A mirror session is an association of a mirror destination with one or more mirror source. The mirror destination and mirror source will describe later.

The device supports up to 3 mirror sessions.

Mirror sessions do not interfere with the normal operation of the switch. However, an oversubscribed mirror destination, for example, a 10-Gbps port monitoring a 100-Gbps port, results in dropped or lost packets.

2.Mirror direction

The device supports to set the direction of the mirror source, there are 3 options for choose: TX/RX/BOTH.

Receive (RX) mirror: The goal of receive (or ingress) mirror is to monitor as much as possible packets received by the source interface or VLAN before any modification or processing is performed by the switch. A copy of each packet received (except these packets: BPDU, LACPDU, BMGPDU, packets have been discarded by IP-MAC binding check for Vlan_based mirror, CRC error packets for both Port_based and vlan_based mirror) by the source is sent to the destination port for that mirror session. You can monitor a series or range of ingress ports or VLANs in a mirror session. Packets that are modified because of routing are copied without modification; that is, the original packet is copied. Packets that are modified because of quality of service (QoS)—for example, modified Differentiated Services Code Point (DSCP)—are copied with modification. Packets that are modified because of VLAN translation or VLAN classification is copied with the modification. Some features that can cause a packet to be dropped during receive processing have no effect on mirror, the destination port can receive a copy of the packet even if the actual incoming packet is dropped. These features include ingress ACL, VLAN's ingress filter, MAC filter, STP, VLAN tag control, port security, unknown routing packets.

Transmit (TX) mirror: The goal of transmit (or egress) mirror is to monitor as much as possible packets sent by the source interface after all modification and processing is performed by the switch. A copy of each packet (except these packets: packets from CPU port for Vlan_based mirror, mirroring packets for both Port_based and vlan_based mirror) sent by the source is sent to the destination port for that mirror session. Some features that can cause a packet to be dropped during transmit processing might have affect on mirror.

Both: In a mirror session, you can monitor a single port for both received and sent packets.

3.Mirror source

The Mirror source is the original traffic of the network. The types of source are described as following:

Source port: A source port is a layer2 or layer 2 interface which need to be monitored. A physical port or link agg port can be a source port. The member of link agg port is not supported to be a mirror source.

Source VLAN: A source vlan is a vlan which need to be monitored. User should create a vlan interface before set a vlan as mirror source.

CPU:User can set CPU as mirror source to monitor the packets send to or receive from the CPU. The copies of packets send to the mirror destination are before cpu-traffic-limit process. Only session 1 support CPU as mirror source currently.

4.Mirror destination

Mirror function will copy the packets and sent the copies to the mirror destination.

The types of destination are described as following:

Local destination port: The destination port should be a physical port or link agg port, member of link agg port is not supported. The destination port has these characteristics:

- It must reside on the same switch as the source port.
- It should not be in "shutdown" state
- It can participate in only one mirror session at a time (a destination port in one mirror session cannot be a destination port for a second mirror session).
- It cannot be a source port.
- The port does not transmit any traffic except that required for the mirror session.
- It does not participate in spanning tree while the mirror session is active.
- When it is a destination port, all other normal system function of this port should not work until mirror destination configure disabled on this port.
- No address learning occurs on the destination port.
- The real statues of the speed/duplex might not coincide with the values which are displayed.

Multi-destination: The device supports to use a group of destination ports to receive several copies of the traffic. The characteristics of each member in the group of destination ports are same as single destination port.

Remote destination: A remote mirror destination is a remote destination vlan, which has a specified out-going port. The copies of the packets should send to the specified port and add the tag of the remote vlan. A remote destination has these characteristics:

- It is a vlan with a specified out going port.
- The remote VLAN range should be 2 to 4094. If the VLAN isn't created in system, user can not configure this VLAN as mirror remote vlan.
- The out going port should be a physical port. User should manually check if the out going port can transfer mirrored packets.
- Monitor traffic packets are inserted a tag with the remote VLAN ID and directed over the specified out going port to the mirror destination session device.
- It is recommended to configure remote mirror's destination port as switch port. Users should add the destination port to the remote vlan otherwise the mirrored packet can not be transmitted out.

CPU destination: send the copies of packet to the CPU of current device. If there is no analyzer available, user can use CPU as mirror destination and save the result for user or developers analyze packets.

You can analyze network traffic passing through ports or vlans by using mirror function to send a copy of the traffic to another port on the switch that has been connected to a Switch Probe device or other Remote Monitoring (RMON) probe or security device. However, when there is no other monitoring device for capturing packets, normal mirror destination to ports doesn't work. So we can set CPU as mirror destination to send a copy of the traffic to CPU for storing packets. It supports the cli to display the packets of mirror CPU and write the packets in a text file. It is a very functional debug tool. Mirror does not affect the switching of network traffic on source ports or source vlans; a copy of the packets received or sent by the source interfaces are sent to the destination CPU. The cpu-traffic-limit rate can be configured. CPU can participate as a destination in only one mirror session.

8.3.2 Configuration

Configuring Local port mirror

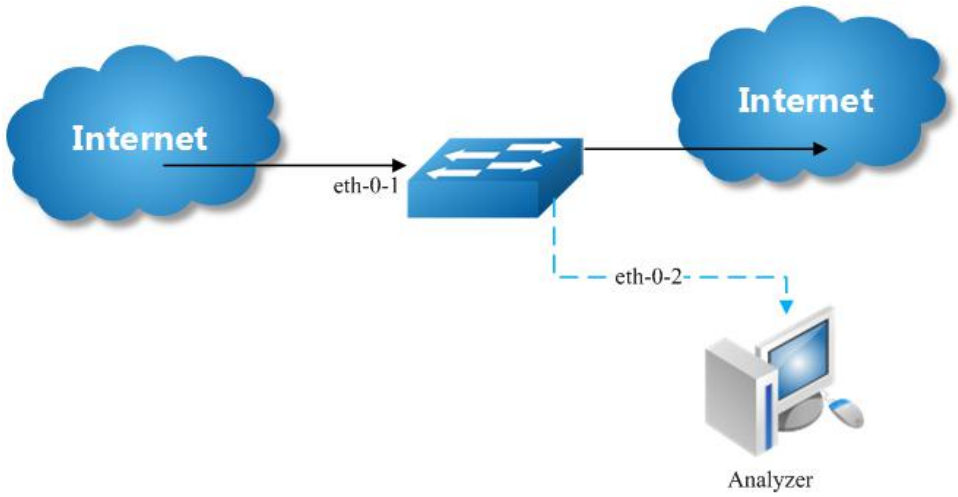


Figure 1-68 port Mirror

Copy the packets of eth-0-1 and send them to eth-0-2

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the destination of mirror

```
Switch(config)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# monitor session 1 destination interface eth-0-2
```

step 3 Set the source of mirror

```
Switch(config)# monitor session 1 source interface eth-0-1 both
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch# show monitor session 1
Session  1
----------
Status          : Valid
Type            : Local Session
Source Ports    :
 Receive Only   :
 Transmit Only  :
 Both           : eth-0-1
Source VLANs    :
 Receive Only   :
 Transmit Only  :
 Both           :
Destination Port  : eth-0-2
```

Configuring local vlan mirror

Copy the packets from vlan 10 and send them to eth-0-2

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the destination of mirror

```
Switch(config)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# monitor session 1 destination interface eth-0-2
```

step 3 Enter the vlan configure mode and create a vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10
Switch(config-vlan)# exit
```

step 4 Create a vlan interface

```
Switch(config)# interface vlan10
Switch(config-if)# exit
```

step 5 Set the source of mirror

```
Switch(config)# monitor session 1 source vlan 10 rx
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

```
Switch# show monitor session 1
Session   1
----------
Status          : Valid
Type            : Local Session
Source Ports    :
 Receive Only   :
 Transmit Only  :
 Both           :
Source VLANs    :
 Receive Only   : 10
 Transmit Only  :
 Both           :
Destination Port  : eth-0-2
```

Configuring CPU as mirror source

Copy the packets from or to CPU and send them to eth-0-2

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the destination of mirror

```
Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# monitor session 1 destination interface eth-0-2
```

step 3 Set the source of mirror

```
Switch(config)# monitor session 1 source cpu both
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
DUT1# show monitor session 1
Session  1
----------
Status          : Valid
Type            : Cpu Session
Source Ports    :
 Receive Only   :
 Transmit Only  :
 Both           : cpu
Source VLANs    :
 Receive Only   :
 Transmit Only  :
 Both           :
Destination Port  :eth-0-1
```

Configuring Multi-destination Mirror

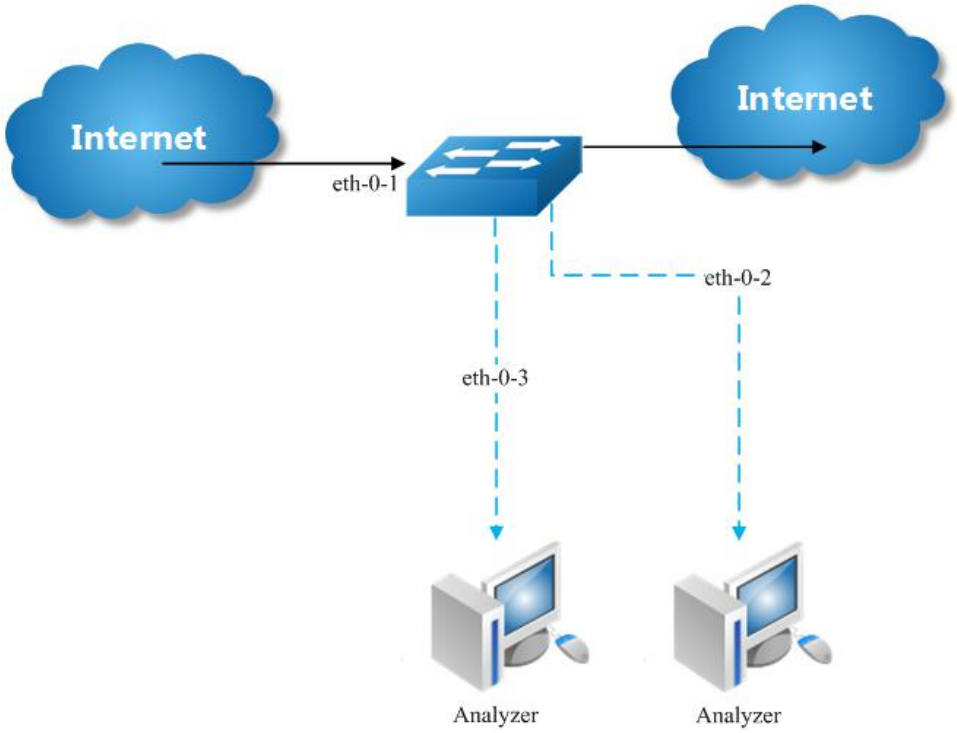


Figure 1-69 Multi-destination Mirror

Copy the packets of eth-0-1 and send them to eth-0-2 and eth-0-3

The rules of mirror source are same as single destination port. The following case use source port for example.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the destination group of mirror

```
Switch(config)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface eth-0-3
```

```
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# monitor session 1 destination group 1
Switch(config-monitor-d-group)# member eth-0-2
Switch(config-monitor-d-group)# member eth-0-3
Switch(config-monitor-d-group)# exit
```

step 3 Set the source of mirror

```
Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# monitor session 1 source interface eth-0-1
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Session  1
----------
Status          : Valid
Type            : Local Session
Source Ports    :
 Receive Only   :
 Transmit Only  :
 Both           : eth-0-1
Source VLANs    :
 Receive Only   :
 Transmit Only  :
 Both           :
Destination Port  : eth-0-2 eth-0-3
```

Configuring Remote Mirror

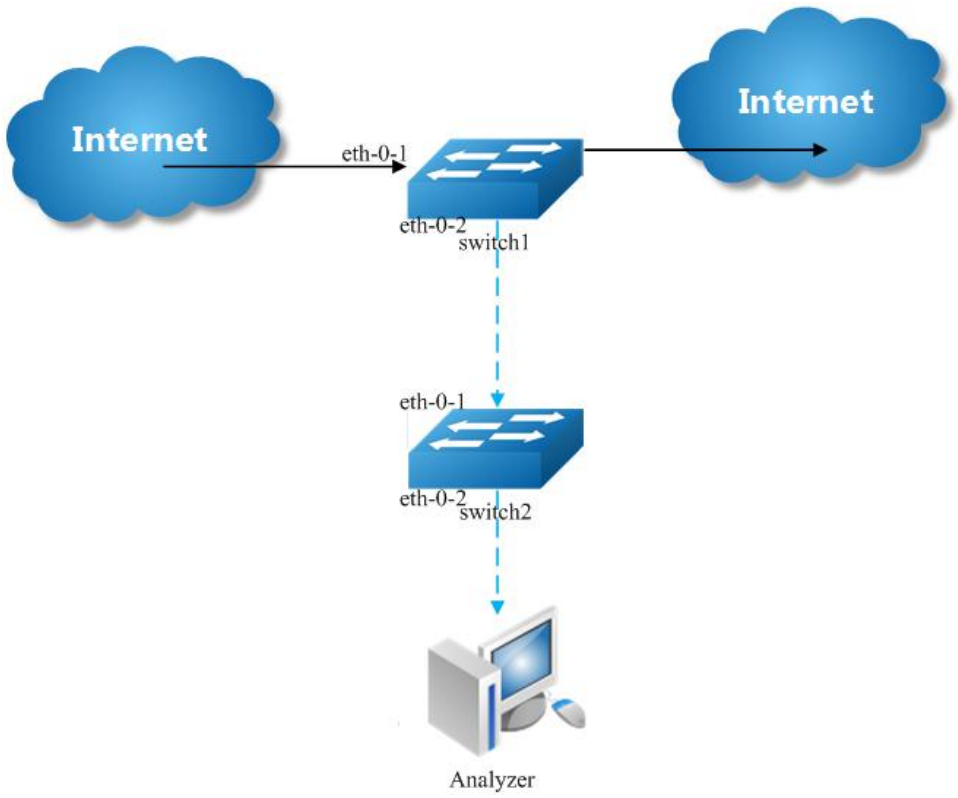


Figure 1-70 Remote Mirror

If local device cannot connect to an analyzer directly, User can choose remote mirror to send the copies of packets with specified vlan tag.

The remote device can pick out the packets with this vlan for analyze.

The following example copies the packets form Switch1's eth-0-1, and send them to Switch2 via Switch1's eth-0-2. Switch2 sends these packets to the analyzer.

The configuration of Switch1:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the destination of mirror

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 15
Switch(config-vlan)# exit
Switch(config-if)# exit

Switch(config)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 15
Switch(config-if)# exit

Switch(config)# monitor session 1 destination remote vlan 15 interface eth-0-2
```

step 3 Set the source of mirror

```
Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config)# monitor session 1 source interface eth-0-1 both
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
SwitchA# show monitor session 1
Session   1
----------
Status          : Valid
Type            : Remote Session
Source Ports    :
 Receive Only   :
 Transmit Only  :
 Both           : eth-0-1
Source VLANs    :
 Receive Only   :
 Transmit Only  :
 Both           :
Destination Port  : eth-0-2
Destination remote VLAN : 15
```

The configuration of Switch2:

Use these methods on Switch2 to send packets to analyzer via eth-0-2

method 1: use vlan 15 as mirror source, eth-0-2 as mirror destination

```
Switch # configure terminal
Switch (config)# vlan database
Switch (config-vlan)# vlan 15
Switch (config-vlan)# exit

Switch (config)# interface vlan15
Switch (config-if)# exit
```

```
Switch (config)# interface eth-0-2
Switch (config-if)# no shutdown

Switch (config)# interface eth-0-1
Switch (config-if)# no shutdown
Switch (config-if)# switchport mode trunk
Switch (config-if)# switchport trunk allowed vlan add 15
Switch (config-if)# exit

Switch (config)# monitor session 1 destination interface eth-0-2
Switch (config)# monitor session 1 source vlan 15 rx
Switch (config)# end
```

method 2: add both ports in to the same vlan (15), and make the packet flood in this vlan

```
Switch# configure terminal

Switch(config)# no spanning-tree enable

Switch(config)# vlan database
Switch(config-vlan)# vlan 15
Switch(config-vlan)# exit

Switch(config)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 15

Switch(config)# interface eth-0-1
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 15
Switch(config-if)# exit
```

NOTE: In this configuration vlan tag is stripped because eth-0-2 is access port.

method 3: flood in vlan and keep vlan tag 15

If user needs to keep the vlan tag 15, eth-0-2 should be trunk port: (other configurations are same as method 2)

```
Switch(config)# interface eth-0-2
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 15
```

Configuring CPU Mirror Dest

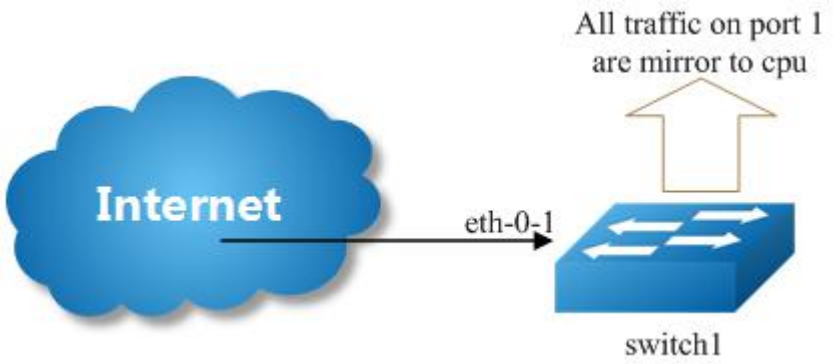


Figure 1-71 Mirror to cpu

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the destination of mirror

```
Switch(config)# monitor session 1 destination cpu
```

Set the buffer size and to cpu rate:

```
Switch(config)# monitor cpu set packet buffer 100
Switch(config)# cpu-traffic-limit reason mirror-to-cpu rate 128
```

step 3 Set the source of mirror

```
Switch(config)# monitor session 1 source interface eth-0-1 both
```

step 4 Exit the configure mode

```
Switch(config)# end
```

Optional steps

Enable or disable to write the packets in to the flash files.

```
Switch#  monitor cpu capture packet start
Switch#  monitor cpu capture packet stop
```

Exchange the files from *.txt to *.pcap

```
Switch# pcap convert flash:/mirror/MirCpuPkt-2016-02-05-18-31-13.txt flash:/MirCpuPkt-2016-02-05.pcap
```

Set the action after the packet buffer is exceeded: "drop" means discard the latest packet; "replace" means discard the oldest packet.

```
Switch(config)# monitor cpu capture strategy drop
Switch(config)# monitor cpu capture strategy replace
```

step 5 Validation

This example shows how to set up a mirror session, session 1, for monitoring source port traffic to a destination cpu. You can use show monitor session to see the configuration.

```
Switch# show monitor session 1
DUT1# show monitor session 1
Session   1
----------
Status          : Valid
Type            : Cpu Session
Source Ports    :
 Receive Only   :
 Transmit Only  :
 Both           : eth-0-1
Source VLANs    :
 Receive Only   :
 Transmit Only  :
 Both           :
Destination Port  : cpu
```

This example shows how to display the mirror cpu packets

```
Switch# show monitor cpu packet all
-----------------show all mirror to cpu packet info-----------------
packet: 1
Source port: eth-0-1
MACDA:264e.ad52.d800, MACSA:0000.0000.1111
vlan tag:100
IPv4 Packet, IP Protocol is 0
IPDA:3.3.3.3, IPSA: 10.0.0.2
Data length: 47
Data:
 264e ad52 d800 0000 0000 1111 8100 0064
```

```
0800 4500 001d 0001 0000 4000 6ad9 0a00
0002 0303 0303 6365 6e74 6563 796f  75
```

This example shows how to display the mirror buffer size:

```
Switch# show monitor cpu packet buffer
--------------------show packet buffer size ---------------------
The mirror-to-cpu packet buffer size of user set is: 100
```

This example shows how to display the mirror cpu traffic-limit rate:

```
Switch# show cpu traffic-limit | include mirror-to-cpu
mirror-to-cpu            128         0
```

This example shows how to display the files of the flash:

```
Switch# ls flash:/mirror
Directory of flash:/mirror

total 8
-rw-r-----  1 2287 Dec 23 01:16 MirCpuPkt-2016-12-23-01-15-54.txt
-rw-r-----  1 2568 Jan  3 11:41 MirCpuPkt-2017-01-03-11-41-33.txt
14.8T bytes total (7.9T bytes free)

Switch# more flash:/mirror/ MirCpuPkt-2017-01-03-11-41-33.txt
sequence  srcPort
1         eth-0-1
++++++++1483443444:648884
8c 1d cd 93 51 00 00 00 00 00 11 11 08 00 45 00
00 26 00 01 00 00 40 00 72 d0 01 01 01 01 03 03
03 03 63 65 6e 74 65 63 79 6f 75 63 65 6e 74 65
63 79 6f 75
--------
sequence  srcPort
2         eth-0-1
++++++++1483443445:546440
8c 1d cd 93 51 00 00 00 00 00 11 11 08 00 45 00
00 26 00 01 00 00 40 00 72 d0 01 01 01 01 03 03
03 03 63 65 6e 74 65 63 79 6f 75 63 65 6e 74 65
63 79 6f 75
```

This example shows how to display the files of the flash. *.pcap files can open with packets analyzer applications such as wireshark. Please referenc to the “ftp” and “tftp” part to download the files.

```
Switch#ls flash:/mirror
Directory of flash:/mirror

total 12
-rw-r-----  1 2287 Dec 23 01:16 MirCpuPkt-2016-12-23-01-15-54.txt
-rw-r-----  1 2568 Jan  3 11:41 MirCpuPkt-2017-01-03-11-41-33.txt
-rw-r--r--  1  704 Jan  3 13:07 test.pcap
14.8T bytes total (7.9T bytes free)
```

This example shows how to display the actions after the buffer is full

```
Switch# show monitor cpu capture strategy
The capture strategy of cpu mirror is: replace (add new packet and remove oldest
 packet when buffer is full)
```

8.3.3 Application Cases

N/A

8.4 Configuring Device Management

8.4.1 Overview

Function Introduction

User can manage the switch through the management port. The switch has two management ports: an Ethernet port and a console port.

Principle Description

N/A

8.4.2 Configuration

Configuring console port for management

The default console parameters of switch are:

- Baud rate default is 115200.
- Data bits default is 8.
- Stop bits default is 1.
- Parity settings default is none.

Before you can assign switch information, make sure you have connected a PC or terminal to the console port, and configured the PC or terminal software parameters to match the default console port parameters. After login in the switch, you can modify the console parameters.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter line configuration mode and set the console speed

```
Switch(config)# line console 0
Switch(config-line)# speed 19200
```

step 3 Exit the configure mode

```
Switch(config-line)# end
```

step 4 Validation

After the above setting, console port parameter has been changed, and the PC or terminal can't configure the switch by console port. You must update PC or terminal console speed from 115200 to 19200 to match the new console parameter and can continue configure the switch by console port.

Configuring out band Ethernet port for management

In order to manage device by out band Ethernet port, you should configure management ip address first by console port.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Configure switch management address

IPv4 & IPv6 are both supported, for example:

```
Switch(config)# management ip address 10.10.38.106/24
Switch(config)# management ipv6 address 2001:1000::1/96
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show management ip address
 Management IP address is:  10.10.38.106/24
 Gateway: 0.0.0.0

Switch # show management ipv6 address
 Management IPv6 address is:  2001:1000::1/96
 Gateway: ::
```

Configuring Temperature

The switch supports temperature alarm management. You can configure three temperature thresholds: low, high and critical. When switch temperature is lower than low threshold or higher than higher threshold, the switch will be alarm. If the switch temperature is higher than critical threshold, the switch will cut off its power automatically.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Configuring temperature threshold

5℃ for low; 70℃ for high; 90℃ for critical.

```
Switch(config)# temperature 5 70 90
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show environment
---------------------------------------------------------------
Sensor status (Degree Centigrade):
Index Temperature  Lower_alarm  Upper_alarm Critical_limit
1    50         5          70         90
```

Configuring Fan

The switch supports to manage fan automatically. If the fan is fail or the fan tray is absent, the switch will be alarm. And if the fan tray supports speed-adjust, the switch can adjust the fan speed depending on the real-time temperature. The switch has three temperature thresholds: Tlow=50, Thigh=65 and Tcrit=80 Celsius scales. If Temperature<Tlow, the fan will stall; if Tlow<=Temperature<Thigh, the fan will run on 30% speed rate; if Thigh<=Temperature<Tcrit, the fan will run on 70% speed rate; if Tcrit>=Temperature, the fan will run on 100% speed rate. And there has a temperature hysteresis Thyst=2 Celsius scales. Assuming temperature has previously crossed above Tlow, Thigh or Tcrit, then the temperature must drop below the points corresponding Thyst(Tlow-Thyst, Thigh-Thyst or Tcrit-Thyst) in order for the condition to drive fan speed rate to lower level. For example:

- temperature is 58 Celsius scales, the fan speed rate is 30%; (Tlow<58<Thigh)
- temperature increases to 65 Celsius scales, the fan speed rate is 70%;(Thigh=65)
- temperature decreases to 63 Celsius scales, the fan speed rate is still 70%;(Thigh-Thyst =63)
- temperature decreases to 62 Celsius scales, the fan speed rate is 30%;(62<Thigh-Thyst)

The Tlow, Thigh, Tcrit, Thyst and fan speed rate for each temperature threshold are hard code, and couldn't be modified.

```
Switch# show environment
Fan tray status:
Index    Status
1        PRESENT
FanIndex   Status SpeedRate Mode
1-1        OK     30%     Auto
1-2        OK     30%     Auto
```

```
1-3      OK    30%    Auto
1-4      OK    30%    Auto
-------------------------------------------------------------
```

Configuring Power

The switch supports to manage power status automatically. If the power is failed or the fan in power is failed, the switch will be alarm. If power is removed or inserted, the switch will notice user also.

User can show the power status to verify the power status.

```
Switch# show environment
-------------------------------------------------------------
Power status:
Index    Status   Power    Type     Fans     Control
1        PRESENT  OK       AC       -        -
2        ABSENT   -        -        -        -
3        PRESENT  OK       DC(PoE)  -        -
-------------------------------------------------------------
```

Configuring Transceiver

The switch supports manage the transceiver information, and the transceiver information includes basic information and diagnostic information. The basic information includes transceiver type, vendor name, PN, S/N, wavelength and link length for supported type. The diagnostic information includes real-time temperature, voltage, current, optical transmit power, optical receive power and the threshold about these parameters. If the transceiver is inserted or removed, the real-time parameter is out of threshold, the switch will notice the users.

User can show the transceiver information to verify this function.

```
Switch# show transceiver detail
Port eth-1-2 transceiver info:
Transceiver Type: 10G Base-SR
Transceiver Vendor Name : OEM
Transceiver PN          : SFP-10GB-SR
Transceiver S/N         : 201033PST1077C
Transceiver Output Wavelength: 850 nm
Supported Link Type and Length:
    Link Length for 50/125um multi-mode fiber: 80 m
    Link Length for 62.5/125um multi-mode fiber: 30 m
-----------------------------------------------------------------------
Transceiver is internally calibrated.
mA: milliamperes, dBm: decibels (milliwatts), NA or N/A: not applicable.
++ : high alarm, +  : high warning, -  : low warning, -- : low alarm.
The threshold values are calibrated.
-----------------------------------------------------------------------
          High Alarm   High Warn   Low Warn    Low Alarm
Temperature  Threshold   Threshold   Threshold    Threshold
Port  (Celsius)    (Celsius)   (Celsius)   (Celsius)    (Celsius)
--------- ----------------- ---------- ---------- ---------- ----------
eth-1-2  25.92      95.00       90.00     -20.00      -25.00
-----------------------------------------------------------------------------------------------
              High Alarm   High Warn   Low Warn    Low Alarm
     Voltage    Threshold    Threshold  Threshold    Threshold
Port    (Volts)     (Volts)     (Volts)     (Volts)     (Volts)
--------- ----------------- ----------------- ----------------- ---------------- ----------------
eth-1-2  3.32       3.80       3.70       2.90        2.80
-----------------------------------------------------------------------------------------------
              High Alarm   High Warn   Low Warn    Low Alarm
      Current    Threshold    Threshold  Threshold    Threshold
Port  (milliamperes)  (mA)       (mA)      (mA)      (mA)
--------- ----------------- ------ ---------- ---------- ----------------  -------------
eth-1-2   6.41       20.00      18.00      1.00         0.50
-----------------------------------------------------------------------------------------------
      Optical     High Alarm   High Warn   Low Warn    Low Alarm
   Transmit Power  Threshold   Threshold   Threshold    Threshold
Port  (dBm)       (dBm)     (dBm)      (dBm)     (dBm)
--------- ----------------- ---------- ---------- ----------------  ----------------
eth-1-2   -2.41       2.01       1.00       -6.99       -7.96
-----------------------------------------------------------------------------------------------
      Optical     High Alarm   High Warn   Low Warn    Low Alarm
```

```
   Receive Power   Threshold   Threshold    Threshold    Threshold
Port   (dBm)       (dBm)      (dBm)       (dBm)        (dBm)
--------- ----------------- --------------- ----------------- ----------------- -----------------
eth-1-2    -12     -    1.00       0.00       -19.00      -20.00
-------------------------------------------------------------------------------------------------
```

Upgrade bootrom

The switch supports to upgrade the bootrom image when system is running. And after upgrading, you must reboot the switch to take effect.

step 1 Copy bootrom image file to the flash

```
Switch# copy mgmt-if tftp://10.10.38.160/bootrom.bin flash:/boot/
```

step 2 Enter the configure mode

```
Switch# configure terminal
```

step 3 Upgrade the bootrom

```
Switch(config)# update bootrom flash:/boot/bootrom.bin
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Reboot the system

```
Switch# reboot
```

step 6 Validation

After the above setting, you can show uboot version information of platform:

```
Switch# show version
……
EPLD Version is 1
BootRom Version is 3.0.2
```

Upgrade EPLD

The switch supports to upgrade the EPLD image when system is running. And after upgrading, you must reboot the switch to take effect.

step 1 Copy epld image file to the flash

```
Switch# copy mgmt-if tftp://10.10.38.160/vme_v1.0 flash:/boot/vme_v1.0
```

step 2 Enter the configure mode

```
Switch# configure terminal
```

step 3 Upgrade the epld

```
Switch(config)# update epld flash:/boot/vme_v1.0
```

step 4 Exit the configure mode

```
Switch(config)# exit
```

step 5 Reboot the system

```
Switch# reboot
```

step 6 Validation

After the above setting, then power off and restart the device, you can show epld version information with command:

```
Switch# show version
……
EPLD Version is 1
BootRom Version is 3.0.2
```

8.4.3 Application Cases

N/A

8.5 Configuring Bootrom

8.5.1 Overview

Function Introduction

The main function of Bootrom is to initialize the board simply and load the system image to boot. You can use some necessary commands in bootrom mode.

Bootrom can load the system image both from TFTP server and persistent storage like flash. Then you can configure the Switch and TFTP server IP address as environment variables in Bootrom mode for boot the system image.

Principle Description

N/A

8.5.2 Configuration

Configuring Boot from TFTP Server

Method 1: Boot the system from TFTP server

Save the configuration and reboot the system:

```
bootrom:> setenv bootcmd boot_tftp OS-ms-v3.1.9.it.r.bin
bootrom:> saveenv
bootrom:> reset
```

Method 2: Method 1:Boot the system from TFTP server without password

Save the configuration and reboot the system:

```
bootrom:> setenv bootcmd boot_tftp_nopass OS-ms-v3.1.9.it.r.bin
bootrom:> saveenv
bootrom:> reset
```

Method 3: Boot the system from TFTP server and reboot automatically

```
bootrom:> boot_tftp OS-ms-v3.1.9.it.r.bin
```

Method 4: Boot the system from TFTP server and reboot automatically without password

```
bootrom:> boot_tftp_nopass OS-ms-v3.1.9.it.r.bin
```

Validation

After the above setting, you can get show information:

```
bootrom:> reset
……..
TFTP from server 10.10.29.160; our IP address is 10.10.29.118
Filename 'OS-ms-v3.1.9.it.r.bin'.
```

```
Load address: 0xaa00000
Loading: octeth0: Up 100 Mbps Full duplex (port  0)
#################################################################
      ####################
done
Bytes transferred = 12314539 (bbe7ab hex), 1829 Kbytes/sec
```

Configuring Boot from FLASH

Boot the system from FLASH

Save the configuration and reboot the system:

```
bootrom:> setenv bootcmd boot_flash OS-ms-v3.1.9.it.r.bin
bootrom:> saveenv
bootrom:> reset
```

Boot the system from without password

Save the configuration and reboot the system:

```
bootrom:> setenv bootcmd boot_flash_nopass OS-ms-v3.1.9.it.r.bin
bootrom:> saveenv
bootrom:> reset
Do you want to revert to the default config file ? [Y|N|E]:Y
```

Boot the system from FLASH and reboot automatically

```
bootrom:> boot_flash OS-ms-v3.1.9.it.r.bin
```

Boot the system from FLASH and reboot automatically without password

```
bootrom:> boot_flash_nopass OS-ms-v3.1.9.it.r.bin
Do you want to  revert  to the default config file ? [Y|N|E]:Y
```

Validation

After the above setting, you can get show information:

```
bootrom:> reset
……
Do you want to  revert  to the default config file ? [Y|N|E]:Y
### JFFS2 loading '/boot/OS-ms-v3.1.9.it.r.bin' to 0xaa00000
Scanning JFFS2 FS: . done.
### JFFS2 load complete: 12314539 bytes loaded to 0xaa00000
## Booting image at 0aa00000 ...
  Verifying Checksum ... OK
  Uncompressing Kernel Image ... OK
……
```

Set boot IP

step 1 Set Switch IP address , details information as follows

```
bootrom:> setenv ipaddr 10.10.29.101
bootrom:> saveenv
```

step 2 Set TFTP server IP address , details information as follows

```
bootrom:> setenv ipserver 10.10.29.160
bootrom:> saveenv
```

step 3 validation

After the above setting, you can get show information:

```
bootrom:> printenv
printenv
bootdelay=5
baudrate=9600
download_baudrate=9600
……
stderr=serial
ipaddr=10.10.29.101
ipserver=10.10.29.160
Environment size: 856/2044 bytes
```

Upgrade bootrom

step 1 upgrade the Bootrom image from TFTP server

```
bootrom:> upgrade_uboot bootrom.bin
```

step 2 validation

After the above setting, you can get show information:

```
bootrom:> version
version
Bootrom 3.0.3 (Development build) (Build time: Aug  4 2011 - 11:47:06)
```

Set gateway IP

step 1 Set Switch gateway IP address , details information as follows

```
bootrom:> setenv gatewayip 10.10.37.1
bootrom:> saveenv
```

step 2 Set network mask , details information as follows

```
bootrom:> setenv netmask 255.255.255.0
bootrom:> saveenv
```

step 3 validation

After the above setting, you can get show information:

```
bootrom:> printenv
printenv
bootdelay=5
baudrate=9600
download_baudrate=9600
……
stderr=serial
gatewayip=10.10.38.1
netmask=255.255.255.0
Environment size: 856/2044 bytes
```

8.5.3 Application Cases

N/A

8.6 Configuring Bootup Diagnostic

8.6.1 Overview

Function Introduction

Bootup diagnostic is used to help user diagnose whether the hardware component of Switch is working normally, after the Switch is already bootup. The diagnostic item includes EPLD, EEPROM, PHY, MAC, etc.

Principle Description

N/A

8.6.2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the bootup diagnotic level

```
Switch(config)# diagnostic bootup level minimal
```

step 3 Exit the configure mode

```
Switch(config)# exit
```

step 4 Validation

Use this command to display the diagnostic bootup level for current and next.

```
Switch# show diagnostic bootup level
The current running is no diagnostic bootup level
The next running bootup diag level is minimal
```

step 5 Reboot the system

```
Switch# reboot
```

step 6 Validation

```
Switch# show diagnostic bootup result detail
########################################################
Item Name                Attribute Result Time(usec)
1   EPLD TEST             C       Pass  57
2   EEPROM0 TEST          C       Pass  101262
3   PHY TEST              C       Pass  1161
4   FAN TEST              C       Pass  4668
5   SENSOR TEST           C       Pass  5472
6   PSU TEST              C       Pass  1370
7   L2 UCAST FUNC TEST    C       Pass  40126
```

8.6.3 Application Cases

N/A

8.7 Configuring SmartConfig

8.7.1 Overview

Function Introduction

SmartConfig is a smart method of switch initial configuration. After enabling SmartConfig, switch will start to download configuration file or image file from tftp server , if not finding startup-config file at startup. Then switch will install these file , and it will reboot itself if had downloaded image file.

Note that we use deploy file to control the configuration file and image file downloaded by switch. Switch fetch these file according the deploy file, which is a XML-formatted file. The deploy file named smartdeploy.xml , while its content like below:

```
<SmartDeploy>
<ftype>init</ftype>
<hostprefix>Bruce</hostprefix>
<defItem>
<option>enable</option>
<image>def.bin</image>
<config>def.cfg</config>
```

```
</defItem>
<groups>
<Item>
<type>MAC</type>
<value>001e.0808.9100</value>
<image>switchOs.bin</image>
<config>startup.cfg</config>
</Item>
<Item>
<type>productid</type>
<value>09SWITCH-E48-10</value>
<image>productid.bin</image>
<config>productid.cfg</config>
</Item>
<Item>
<type>SN</type>
<value>E054GD116004</value>
<image>sn.bin</image>
<config>sn.cfg</config>
</Item>
</groups>
</SmartDeploy>
```

There are three kind of item used by switch to find out image file and configuration file fit itself. Switch will search fit item according sequence like MAC, SN , product-id。We just specify the file name in the deploy file, and place all these file on tftp server.

Principle Description

N/A

8.7.2 Configuration

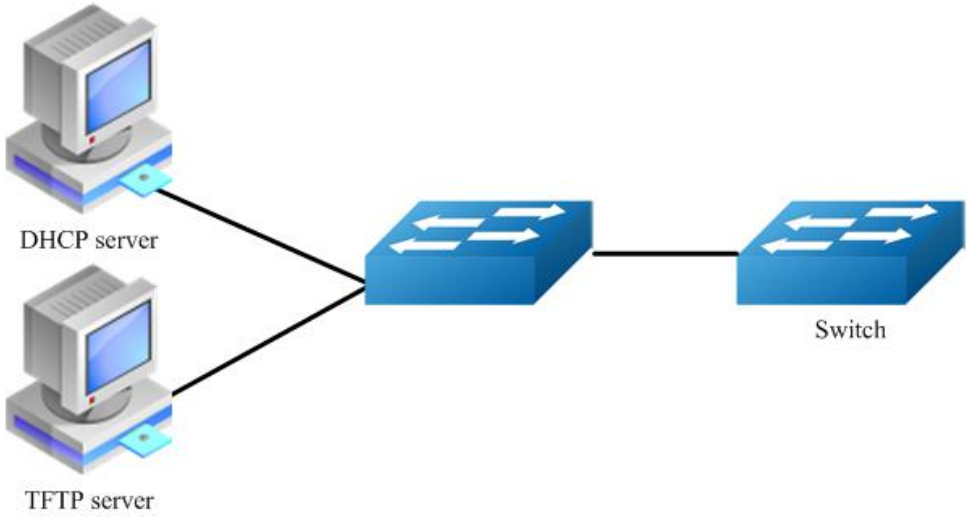


Figure 1-72 smart config

This figure is the network topology of testing SmartConfig function, We need two switches and two linux boxes to construct the test bed。"switch" in the figure is the switch we enable SmartCofng on. Note that the address of TFTP server provided by DHCP server can be used by switch to connect to TFTP server directly or via routes.

Enable smartConfig

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Enable smartConfige

```
Switch(config)#smart-config initial-switch-deployment
```

step 3 Exit the configure mode

```
Switch (config)#exit
```

step 4 Validation

Use this command to check the smart-config settings:

```
Switch# show smart-config config
Smart-Config config:
 initial-switch-deployment: on
 hostname-prefix: on
 Send log message to console: on
```

Using smartConfig

SmartConfig was enable default, so we just make sure there is no startup-config.conf file. Then switch will start SmartConfig next boot. And we can delete startup-config.conf manually, so that Smartconfig will work after reboot. Procedure of configure SmartConfig as fallow:

step 1:

Configure smartdeploy.xml file, and place it with image file, configuration file to tftp server. The directory must be like this (Configuration files should be in conf directory and images should be in images directory.) :

```
 /--
 |--smartconfig/
 |--conf/
 |--images/
 |--smartdeploy.xml
```

step 2:

Configure DHCP server, tftp server address option must be set；

step 3:

Make sure there is no startup-config.conf file;

step 4:

boot or reboot the system.

8.7.3 Application Cases

N/A

8.8 Reboot Logs

8.8.1 Overview

Function Introduction

Switch support display reboot logs. Depend on these logs, user can judge the reboot reasons of a switch. The reboot reasons include Manual Reboot, Power Off or Other Reasons. Also, user can clear the reboot logs through a command.

NOTE: User can find no more than ten reboot logs through this command, to find more reboot logs, can refer to the following file: flash:/reboot-info/reboot_info.log

Detail about the show result as following:

Reboot Type	Description
POWER	Power outages
MANUAL	Cli "reboot/reload" is used

Reboot Type	Description
HIGH-TMPR	Reboot for abnormal high temperature
BHMDOG BHM	watchdog, monitor functional module
LCMDOG LCM	watchdog, monitor each LC
SCHEDULE	Schedule reboot
SNMP-RELOAD	SNMP reboot
HALFAIL	Reboot for HAGT communicate with HSRV failed, need stack enable
ABNORMAL	Unusual reboot, include reboot under shell
CTCINTR	Button reboot
LCATTACH	Reboot for LC attach CHSM failed
OTHER	Other reboot

Principle Description

N/A

8.8.2 Configuration

Reboot logs are enabled by default. User can display and clear the logs as the following examples:

step 1 Display the logs

```
Switch# show reboot-info
Times     Reboot Type     Reboot Time(DST)
1         MANUAL          2000/01/01 01:21:35
2         MANUAL          2000/01/01 02:07:52
3         MANUAL          2000/01/01 02:24:59
4         MANUAL          2000/01/01 03:28:58
5         MANUAL          2000/01/01 03:43:02
6         MANUAL          2000/01/01 03:49:51
7         MANUAL          2000/01/01 04:01:23
8         MANUAL          2000/01/01 04:42:40
9         MANUAL          2000/01/01 04:49:27
10        MANUAL          2000/01/01 20:59:20
```

step 2 Clear the logs(optional)

```
Switch(config)# reset reboot-info
```

8.8.3 Application Cases

N/A

Chapter 9 Network Management Configuration Guide

9.1 Configuring Network Diagnosis

9.1.1 Overview

Function Introduction

Ping is a computer network administration utility used to test the reachability of a host on an Internet Protocol (IP) network and to measure the round-trip time for messages sent from the originating host to a destination computer. The name comes from active sonar terminology.

Ping operates by sending Internet Control Message Protocol (ICMP) echo request packets to the target host and waiting for an ICMP response. In the process it measures the time from transmission to reception (round-trip time) [1] and records any packet loss. The results of the test are printed in form of a statistical summary of the response packets received, including the minimum, maximum, and the mean round-trip times, and sometimes the standard deviation of the mean.

Traceroute is a computer network tool for measuring the route path and transit times of packets across an Internet Protocol (IP) network.

Traceroute sends a sequence of Internet Control Message Protocol (ICMP) packets addressed to a destination host. Tracing the intermediate routers traversed involves control of the time-to-live (TTL) Internet Protocol parameter. Routers decrement this parameter and discard a packet when the TTL value has reached zero, returning an ICMP error message (ICMP Time Exceeded) to the sender.

Principle Description

N/A

9.1.2 Configuration

Ping IP with in-band port

```
Switch# ping 10.10.29.247
Switch# ping ipv6 2001:1000::1
```

Ping IP with management port

```
Switch# ping mgmt-if 10.10.29.247
Switch# ping mgmt-if ipv6 2001:1000::1
```

Ping IP with VRF instance

```
Switch# ping vrf vrf1 10.10.10.1
```

Traceroute IP with inner port

```
Switch# traceroute 1.1.1.2
Switch# traceroute ipv6 2001:1000::1
```

9.1.3 Application Cases

Example for Ping

```
Switch # ping mgmt-if 192.168.100.101
PING 192.168.100.101 (192.168.100.101) 56(84) bytes of data.
64 bytes from 192.168.100.101: icmp_seq=0 ttl=64 time=0.092 ms
64 bytes from 192.168.100.101: icmp_seq=1 ttl=64 time=0.081 ms
64 bytes from 192.168.100.101: icmp_seq=2 ttl=64 time=0.693 ms
64 bytes from 192.168.100.101: icmp_seq=3 ttl=64 time=0.071 ms
64 bytes from 192.168.100.101: icmp_seq=4 ttl=64 time=1.10 ms
--- 192.168.100.101 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4054ms
rtt min/avg/max/mdev = 0.071/0.408/1.104/0.421 ms, pipe 2
```

Example for traceroute

```
Switch# traceroute 1.1.1.2
traceroute to 1.1.1.2 (1.1.1.2), 30 hops max, 38 byte packets
 1  1.1.1.2 (1.1.1.2)  112.465 ms  102.257 ms  131.948 ms
Switch # ping mgmt-if ipv6 2001:1000::1
PING 2001:1000::1(2001:1000::1) 56 data bytes
64 bytes from 2001:1000::1: icmp_seq=1 ttl=64 time=0.291 ms
64 bytes from 2001:1000::1: icmp_seq=2 ttl=64 time=0.262 ms
64 bytes from 2001:1000::1: icmp_seq=3 ttl=64 time=0.264 ms
64 bytes from 2001:1000::1: icmp_seq=4 ttl=64 time=0.270 ms
64 bytes from 2001:1000::1: icmp_seq=5 ttl=64 time=0.274 ms
--- 2001:1000::1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3997ms
rtt min/avg/max/mdev = 0.262/0.272/0.291/0.014 ms
```

9.2 Configuring NTP

9.2.1 Overview

Function Introduction

NTP is a tiered time distribution system with redundancy capability. NTP measures delays within the network and within the algorithms on the machine on which it is running. Using these tools and techniques, it is able to synchronize clocks to within milliseconds of each other when connected on a Local Area Network and within hundreds of milliseconds of each other when connected to a Wide Area Network. The tiered nature of the NTP time distribution tree enables a user to choose the accuracy needed by selecting a level (stratum) within the tree for machine placement. A time server placed higher in the tree (lower stratum number), provides a higher likelihood of agreement with the UTC time standard.

Some of the hosts act as time servers, that is, they provide what they believe is the correct time to other hosts. Other hosts act as clients, that is, they find out what time it is by querying a time server. Some hosts act as both clients and time servers, because these hosts are links in a chain over which the correct time is forwarded from one host to the next. As part of this chain, a host acts first as a client to get the correct time from another host that is a time server. It then turns around and functions as a time server when other hosts, acting as clients, send requests to it for the correct time.

Principle Description

N/A

9.2.2 Configuration

Configuring Client/Server mode connecting with in-band interface

Before configuring NTP client, make sure that NTP service is enabled on Server.

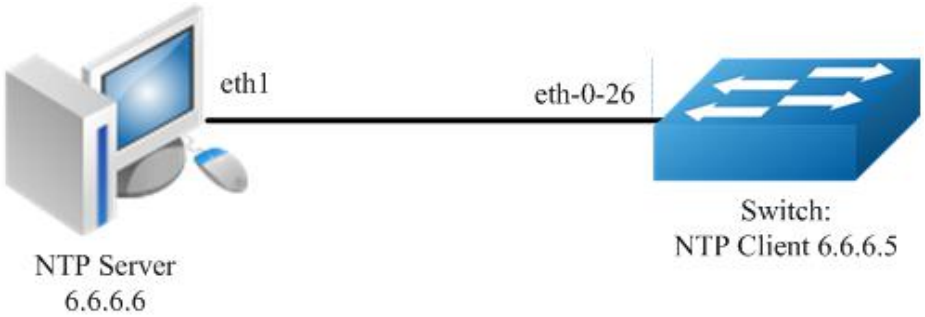


Figure 1-73 NTP

step 1 Enter the configure mode

```
Switch#configure terminal
```

step 2 Enter the vlan configure mode and create a vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 10
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode and join the vlan

```
Switch(config)# interface eth-0-26
Switch(config-if)# switch access vlan 10
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 4 create a vlan interface and set the IP address

```
Switch(config)# interface vlan10
Switch(config-if)# ip address 6.6.6.5/24
Switch(config-if)# exit
```

step 5 Set the attributes of NTP client

Enable a trustedkey; Configure the IP address of the NTP server; Enable authentication; Once you have enabled authentication, the client switch sends the time-of-day requests to the trusted NTP servers only; Configure ntp ace.

```
Switch(config)# ntp key 1 serverkey
Switch(config)# ntp server 6.6.6.6 key 1
Switch(config)# ntp authentication enable
Switch(config)# ntp trustedkey 1
Switch(config)# ntp ace 6.6.6.6 none
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

```
Switch# show ntp
Current NTP configuration:
============================================================
NTP access control list:
 6.6.6.6 none
Unicast peer:
Unicast server:
 6.6.6.6 key 1
Authentication: enabled
Local reference clock:
Disable management interface

Switch# show ntp status
Current NTP status:
============================================================
clock is synchronized
stratum:       7
reference clock: 6.6.6.6
frequency:     17.365 ppm
precision:     2**20
reference time:  d14797dd.70b196a2  ( 1:54:37.440 UTC   Thu Apr  7 2011)
root delay:    0.787 ms
root dispersion: 23.993 ms
peer dispersion: 57.717 ms
clock offset:   -0.231 ms
stability:     6.222 ppm
Switch# show ntp associations
Current NTP associations:
  remote         refid     st   when poll reach   delay   offset  disp
================================================================================
*6.6.6.6       127.127.1.0   6   50  128   37     0.778   -0.234  71.945
synchronized, + candidate, # selected, x falsetick, . excess, - outlier
```

Configuring Client/Server mode connecting with management interface

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable ntp management interface

```
Switch(config)# ntp mgmt-if only
```

Note1: Use the following command to enable both in-band and management interface

```
Switch(config)# ntp mgmt-if enable
```

Note: Use the following command to disable management interface

```
Switch(config)# no ntp mgmt-if
```

step 3 Set the attributes of NTP client

```
Switch(config)# ntp key 1 serverkey
Switch(config)# ntp server 192.168.100.101 key 1
Switch(config)# ntp authentication enable
Switch(config)# ntp trustedkey 1
Switch(config)# ntp ace 192.168.100.101 none
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch# show ntp
Current NTP configuration:
============================================================
NTP access control list:
 192.168.100.101 none
Unicast peer:
Unicast server:
 192.168.100.101 key 1
Authentication: enabled
Local reference clock:
Only management interface

Switch# show ntp associations
Current NTP associations:
  remote         refid     st when poll reach   delay   offset  disp
==============================================================================
*192.168.100.101 127.127.1.0     3  27   64    1    1.328    2.033 433.075

* sys.peer, + candidate, # selected, x falsetick, . excess, - outlyer
```

9.2.3 Application Cases

Configuring NTP Server (Use the ntpd of linux system for example）

Step 1 Display eth1 ip address

```
[root@localhost octeon]# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:08:C7:89:4B:AA
          inet addr:6.6.6.6  Bcast:6.6.6.255  Mask:255.255.255.0
          inet6 addr: fe80::208:c7ff:fe89:4baa/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3453 errors:1 dropped:0 overruns:0 frame:1
          TX packets:3459 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:368070 (359.4 KiB)  TX bytes:318042 (310.5 KiB)
```

Step 2 Check networks via Ping

```
[root@localhost octeon]# ping 6.6.6.5
PING 6.6.6.5 (6.6.6.5) 56(84) bytes of data.
64 bytes from 6.6.6.5: icmp_seq=0 ttl=64 time=0.951 ms
64 bytes from 6.6.6.5: icmp_seq=1 ttl=64 time=0.811 ms
64 bytes from 6.6.6.5: icmp_seq=2 ttl=64 time=0.790 ms
```

Step 3 Configure ntp.conf

```
[root@localhost octeon]# vi /etc/ntp.conf
server  127.127.1.0 # local clock
fudge   127.127.1.0 stratum 5
#
# Drift file.  Put this in a directory which the daemon can write to.
# No symbolic links allowed, either, since the daemon updates the file
# by creating a temporary in the same directory and then rename()'ing
# it to the file.
#
driftfile /var/lib/ntp/drift
broadcastdelay  0.008
broadcast 6.6.6.255
#
# PLEASE DO NOT USE THE DEFAULT VALUES HERE. Pick your own, or remote
# systems might be able to reset your clock at will. Note also that
# ntpd is started with a -A flag, disabling authentication, that
# will have to be removed as well.
#
#disable auth
keys       /etc/ntp/keys
trustedkey 1
```

Step 4 Configure keys

```
[root@localhost octeon]# vi /etc/ntp/keys
#
# PLEASE DO NOT USE THE DEFAULT VALUES HERE. Pick your own, or remote
# systems might be able to reset your clock at will. Note also that
# ntpd is started with a -A flag, disabling authentication, that
# will have to be removed as well.
#
1 M serverkey
```

Step 5 Start ntpd service

```
[root@localhost octeon]# ntpd
```

9.3 Configuring Phy Loopback

9.3.1 Overview

Function Introduction

Phy loopback is a proprietary based loopback. There are 2 types of phy loopback: phy(including internal and external) level loopback and port level loopback.

- If a physical port is configured as "external phy loopback", all packets coming into this port should be loopback back from the port itself at phy level.
- If a physical port is configured as "internal phy loopback", all packets expected out from this port should be looped back to specified physical port.
- If a physical port is configured as "port loopback", all packets coming into this port should be looped back from the port itself, and whether to swap the SMAC with the DMAC should be selectable by users. And if the MAC is swapped, the CRC should be recalculated.

Principle Description

N/A

9.3.2 Configuration

Configuring external phy loopback

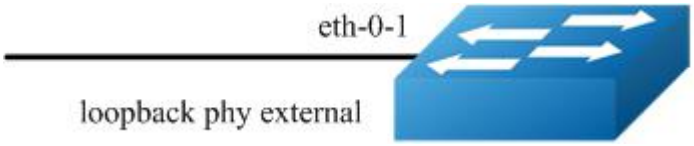


Figure 1-74 external phy topology

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set loopback phy external

```
Switch (config)# interface eth-0-1
Switch (config-if)# no shutdown
Switch (config-if)# loopback phy external
```

step 3 Exit the configure mode

```
Switch (config-if)# end
```

step 4 Validation

```
Switch# show phy loopback
 Interface  Type      DestIntf  SwapMac
 ------------------------------------------------
eth-0-1   external  -        -
 ------------------------------------------------
```

Configuring internal phy loopback

Figure 1-75 Internal phy topology

step 1 Enter the configure mode

```
Switch # configure terminal
```

step 2 Enter the interface configure mode and set loopback phy internal and specify the destination interface

```
Switch (config)# interface eth-0-2
Switch (config-if)# no shutdown
Switch (config-if)# exit

Switch (config)# interface eth-0-1
Switch (config-if)# no shutdown
Switch (config-if)# loopback phy internal eth-0-2
```

step 3 Exit the configure mode

```
Switch (config-if)# end
```

step 4 Validation

```
Switch# show phy loopback
 Interface  Type      DestIntf   SwapMac
 ------------------------------------------------
eth-0-1    internal   eth-0-2    -
 ------------------------------------------------
```

Configuring port level loopback

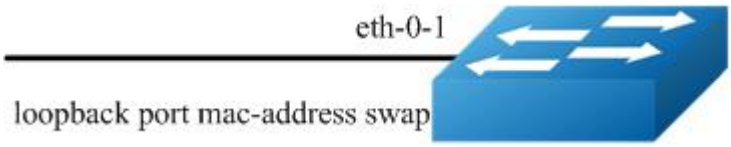


Figure 1-76 Port level topology

step 1 Enter the configure mode

```
Switch # configure terminal
```

step 2 Enter the interface configure mode and set loopback phy mac-address swap

```
Switch (config)# interface eth-0-1
Switch (config-if)# no shutdown
Switch (config-if)# loopback port mac-address swap
```

step 3 Exit the configure mode

```
Switch (config-if)# end
```

step 4 Validation

```
Switch# show phy loopback
 Interface  Type      DestIntf   SwapMac
 ------------------------------------------------
eth-0-1    port      -          yes
 ------------------------------------------------
```

9. 3. 3 Application Cases

N/A

9. 3. 4 Configuring L2 ping

9. 3. 5 Overview

Function Introduction

The tool L2 ping is a useful application which's purpose is detecting the connection between two switches. The L2 ping tool is not same with the well-known 'ping IP-ADDRESS' in the WINDOWS system. The normal "ping" is realized by the protocol ICMP which is dependent on the IP layer, so it may be inapplicable if the destination device is only Layer 2 switch. But the protocol used by L2 ping is only relying on Layer 2 ethernet packets.

When L2 ping is started, the L2 ping protocol packet (with ether type '36873(0x9009)') is sent from a specified physical port to another specified destination port. At the destination end, the L2 ping protocol will be sent back via non 802.1ag loopback, or via a configuration "l2 ping response". The device which is pinging, will receive the ping response packet, and print the ping result.

Principle Description

N/A

9. 3. 6 Configuration

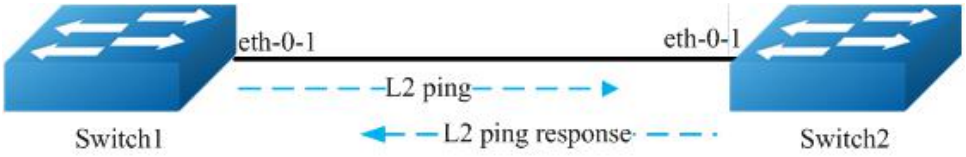


Figure 1-77 ping a switch port

The configurations are almost same on Switch1 and Switch2, except the parts which are specially pointed out.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and turn up the interface

```
Switch (config)# interface eth-0-1
Switch (config-if)# no shutdown
```

step 3 Enable the L2 ping response function

Configure on Switch2:

```
Switch (config-if)# l2 ping response enable
```

step 4 Exit the configure mode

```
Switch (config-if)# end
```

step 5 Using L2 ping

Operate on Switch1:

```
Switch1# l2 ping 001e.0808.58f1 interface eth-0-1 count 10 interval 1000 timeout 2000

 Sending 10 L2 ping message(s):
64 bytes from 001e.0808.58f1: sequence = 0, time = 10ms
64 bytes from 001e.0808.58f1: sequence = 1, time = 15ms
64 bytes from 001e.0808.58f1: sequence = 2, time = 13ms
64 bytes from 001e.0808.58f1: sequence = 3, time = 12ms
64 bytes from 001e.0808.58f1: sequence = 4, time = 20ms
64 bytes from 001e.0808.58f1: sequence = 5, time = 21ms
64 bytes from 001e.0808.58f1: sequence = 6, time = 12ms
64 bytes from 001e.0808.58f1: sequence = 7, time = 16ms
64 bytes from 001e.0808.58f1: sequence = 8, time = 14ms
64 bytes from 001e.0808.58f1: sequence = 9, time = 17ms
L2 ping completed.
 ---------------------------------
10 packet(s) transmitted, 10 received, 0 % packet loss
```

001e.0808.58f1 is the MAC address of the interface on Switch2. It can be gained by command "show interface eth-0-1" on Switch2.

9. 3. 7 Application Cases

N/A

9.4 Configuring RMON

9.4.1 Overview

Function Introduction

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. You can use the RMON feature with the Simple Network Management Protocol (SNMP) agent in the switch to monitor all the traffic flowing among switched on all connected LAN segments.

RMON is a standard monitoring specification that defines a set of statistics and functions that can be exchanged between RMON-compliant console systems and network probes RMON provides you with comprehensive network-fault diagnosis, planning, and performance-tuning information.

Principle Description

N/A

9.4.2 Configuration

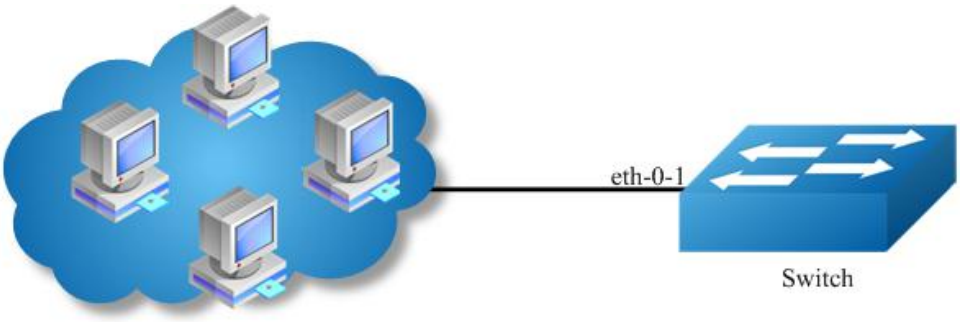


Figure 1-78 rmon

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and create a stats and a history

```
Switch(config)# interface eth-0-1
Switch(config-if)# rmon collection stats 1 owner test
Switch(config-if)# rmon collection history 1 buckets 100 interval 1000 owner test
Switch(config-if)# exit
```

step 3 Create an event with log and trap both set.

```
Switch(config)# rmon event 1 log trap public description test_event owner test
```

step 4 Create a alarm using event 1 we created before and monitor the alarm on ETHERSTATSBROADCASTPKTS on eth-0-1

```
Switch(config)# rmon alarm 1 etherStatsEntry.6.1 interval 1000 delta rising-threshold 1000 event 1 falling-threshold 1 event 1 owner test
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show rmon statistics
 Rmon collection index 1
     Statistics ifindex = 1, Owner: test
     Input packets 0, octets 0, dropped 0
     Broadcast packets 0, multicast packets 0, CRC alignment errors 0, collisions 0
     Undersized packets 0, oversized packets 0, fragments 0, jabbers 0
```

```
    # of packets received of length (in octets):
    64: 0,  65-127: 0, 128-255: 0
    256-511: 0,  512-1023: 0, 1024-max: 0

Switch# show rmon history
 History index = 1
    Data source ifindex = 1
    Buckets requested = 100
    Buckets granted = 100
    Interval = 1000
    Owner: test

Switch# show rmon event
 Event Index = 1
    Description: test_event
    Event type Log & Trap
    Event community name: public
    Last Time Sent = 00:00:00
    Owner: test

Switch# show rmon alarm
 Alarm Index = 1
 Alarm status = VALID
 Alarm Interval = 1000
 Alarm Type is Delta
 Alarm Value = 00
 Alarm Rising Threshold = 1000
 Alarm Rising Event = 1
 Alarm Falling Threshold = 1
 Alarm Falling Event = 1
 Alarm Owner is test
```

9.4.3 Application Cases

N/A

9.5 Configuring SNMP

9.5.1 Overview

Function Introduction

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a MIB. The SNMP manager can be part of a network management system (NMS). The agent and MIB reside on the switch. To configure SNMP on the switch, you define the relationship between the manager and the agent. The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data. An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Error user authentication, restarts, link status (up or down), MAC address tracking, closing of a Transmission Control Protocol (TCP) connection, loss of connection to a neighbor, or other significant events may send a trap.

Principle Description

SNMP module is based on the following RFC draft:

- SNMPv1: Defined in RFC 1157.
- SNMPv2C: Defined in RFC 1901.
- SNMPv3: Defined in RFC 2273 to 2275.
- Following is a brief description of terms and concepts used to describe the SNMP protocol:
- Agent: A network-management software module, an agent has local knowledge of management information and translates that information into a form compatible with SNMP.
- Management Information Base (MIB): Management Information Base, collection of information is organized hierarchically.

- Engine ID: A unique ID for a network's node.
- Trap: Used by managed devices to asynchronously report events to the NMS.

9.5.2 Configuration

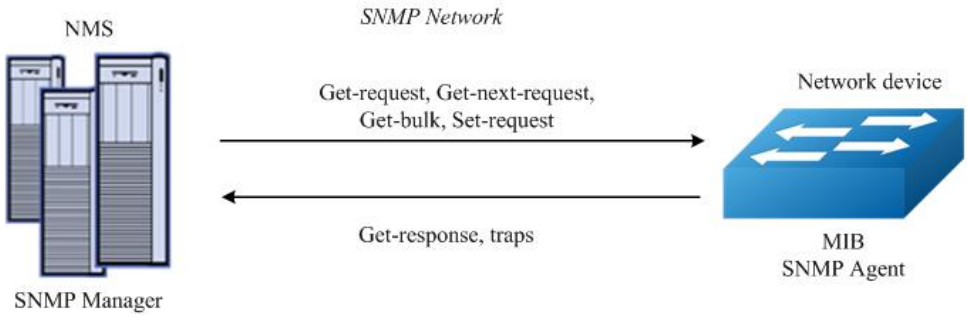


Figure 1-79 snmp

As shown in the figure SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in get-request, get-next-request, and set-request format.

Enable SNMP

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable SNMP globally

```
Switch(config)# snmp-server enable
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show running-config
snmp-server enable
```

Configuring community string

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the switch. Optionally, you can specify one or more of these characteristics associated with the string:

- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Beginning in privileged EXEC mode, follow these steps to configure a community string on the switch.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Configuring community string

Configure a view named "DUT"(optional); Configure a community named "public" with read access and view "DUT".

```
Switch(config)# snmp-server view DUT included 1
Switch(config)# snmp-server community public read-write (view DUT)
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show running-config
snmp-server enable
snmp-server view DUT included .1
snmp-server community public read-only view DUT
```

Configuring SNMPv3 Groups, Users and Accesses

You can specify an identification name (engine ID) for the local SNMP server engine on the switch. You can configure an SNMP server group that maps SNMP users to SNMP views, you can add new users to the SNMP group, and you can add access for the SNMP group.

Beginning in privileged EXEC mode, follow these steps to configure SNMP on the switch.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the globle configurations for SNMP

Set engineID; Set the user name, password, and authentication type; Create SNMP server; Set the authority for the group member.

```
Switch(config)# snmp-server engineID 8000123456
Switch(config)# snmp-server usm-user usr1 authentication md5 mypassword privacy des yourpassword
Switch(config)# snmp-server group grp1 user usr1 security-model usm
Switch(config)# snmp-server access grp1 security-model usm noauth
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show running-config
snmp-server engineID 8000123456
snmp-server usm-user usr1 authentication md5 mypassword privacy des yourpassword
snmp-server group grp1 user usr1 security-model usm
snmp-server access grp1 security-model usm noauth
```

SNMPv1 and SNMPv2 notifications configure

Beginning in privileged EXEC mode, follow these steps to configure SNMP on the switch.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the global configurations for SNMP

Enable all supported traps; Configure a remote trap manager which IP is “10.0.0.2”; Configure a remote trap manager which IPv6 address is “2001:1000::1”.

```
Switch(config)# snmp-server trap enable all
Switch(config)# snmp-server trap target-address 10.0.0.2 community public
Switch(config)# snmp-server trap target-address 2001:1000::1 community public
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show running-config
snmp-server trap target-address 10.0.0.2 community public
snmp-server trap target-address 2001:1000::1 community public
snmp-server trap enable vrrp
snmp-server trap enable igmp snooping
snmp-server trap enable ospf
snmp-server trap enable pim
snmp-server trap enable stp
snmp-server trap enable system
snmp-server trap enable coldstart
snmp-server trap enable warmstart
snmp-server trap enable linkdown
snmp-server trap enable linkup
```

Configuring SNMPv3 notifications

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the global configurations for SNMP

Enable all supported traps; Configure a trap notify item for SNMPv3; Configure a remote trap manager's IP address; Configure a remote trap manager's IPv6 address; Add a local user to SNMPv3 notifications.

```
Switch(config)# snmp-server trap enable all
Switch(config)# snmp-server notify notif1 tag tmptag trap
Switch(config)# snmp-server target-address targ1 param parm1 10.0.0.2 taglist tmptag
Switch(config)# snmp-server target-address t1 param p1 2001:1000::1 taglist tag1
Switch(config)# snmp-server target-params parm1 user usr1 security-model v3 message-processing v3 noauth
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show running-config
snmp-server notify notif1 tag tmptag trap
snmp-server target-address t1 param p1  2001:1000::1 taglist tag1
snmp-server target-address targ1 param parm1  10.0.0.2 taglist tmptag
snmp-server target-params parm1 user usr1 security-model v3 message-processing v3 noauth
snmp-server trap enable vrrp
snmp-server trap enable igmp snooping
snmp-server trap enable ospf
snmp-server trap enable pim
snmp-server trap enable stp
snmp-server trap enable system
snmp-server trap enable coldstart
snmp-server trap enable warmstart
snmp-server trap enable linkdown
snmp-server trap enable linkup
```

9.5.3 Application Cases

N/A

9.6 Configuring SFLOW

9.6.1 Overview

Function Introduction

sFlow is a technology for monitoring traffic in data networks containing switches and routers. In particular, it defines the sampling mechanisms implemented in a sFlow Agent for monitoring traffic, and the format of sample data used by the sFlow Agent when forwarding data to a central data collector.

The architecture and sampling techniques used in the sFlow monitoring system are designed to provide continuous site-wide (and network-wide) traffic monitoring for high speed switched and routed networks.

The sFlow Agent uses two forms of sampling: statistical packet-based sampling of switched flows, and time-based sampling of network interface statistics.

Default Configuration for sflow:

Feature	Default Setting
global sflow	disabled
sflow on port	disable
collector udp port	6343
counter interval time	20 seconds

Principle Description

N/A

9.6.2 Configuration

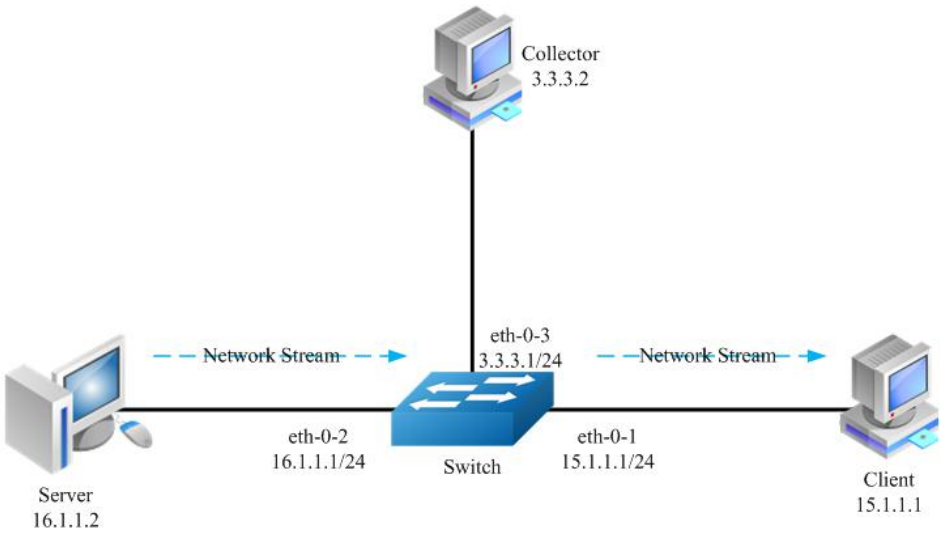


Figure 1-80 sflow

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable sflow globally

```
Switch(config)# sflow enable
```

step 3 Set the global attribute for sflow

Set the agent IP address, set the collector IP address and udp port. If the udp port is not specified, it means default port 6364.

```
Switch(config)# sflow agent ip 3.3.3.1
Switch(config)# sflow collector 3.3.3.2 6342
```

Set the agent and collector with IPv6:

```
Switch(config)# sflow agent ipv6 2001:2000::2
Switch(config)# sflow collector 2001:2000::1
```

NOTE: At list one Agent and one collector must be configured for sflow. User can use IPv4 or IPv6.

Set the interval to send interface counter information (optional):

```
Switch(config)# sflow counter interval 15
```

step 4 Enter the interface configure mode and set the attributes of the interfaces

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 15.1.1.1/24
Switch(config-if)# exit

Switch(config)# interface eth-0-2
Switch(config-if)#no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 16.1.1.1/24
Switch(config-if)# exit

Switch(config)# interface eth-0-3
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 3.1.1.1/24
Switch(config-if)# exit
```

step 5 Enable sflow for input packets on eth-0-1

```
Switch(config)# interface eth-0-1
Switch(config-if)# sflow flow-sampling rate 8192
Switch(config-if)# sflow flow-sampling enable input
Switch(config-if)# sflow counter-sampling enable
Switch(config-if)# exit
```

step 6 Validation

To display the sflow configuration, use following command:

```
Switch# show sflow
sFlow Version: 5
sFlow Global Information:
 Agent IPv4 address          : 3.3.3.1
 Agent IPv6 address          : 2001:1000::2
 Counter Sampling Interval    : 15 seconds
 Collector 1:
  IPv4 Address: 3.3.3.2
  Port: 6342
 Collector 2:
  IPv6 Address: 2001:1000::1
  Port: 6343

sFlow Port Information:
                        Flow-Sample   Flow-Sample
 Port      Counter  Flow     Direction    Rate
 ---------------------------------------------------------
 eth-0-1    Enable   Enable   Input       8192
```

9.6.3 Application Cases

N/A

9.7 Configuring LLDP

9.7.1 Overview

Function Introduction

LLDP（Link Layer Discovery Protocol）is the discovery protocol on link layer defined as standard in IEEE 802.1ab. Discovery on Layer 2 can locate interfaces attached to the devices exactly with connection information on layer 2, such as VLAN attribute of port and protocols supported, and present paths among client, switch, router, application servers and other network servers. This detailed description is helpful to get useful information for diagnosing network fast, like topology of devices attached, conflict configuration between devices, and reason of network failure.

Principle Description

N/A

9.7.2 Configuration

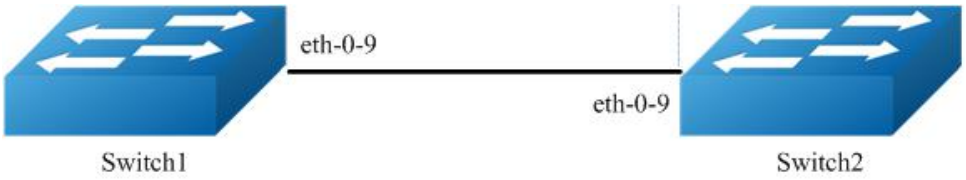


Figure 1-81 lldp

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable SNMP globally

```
Switch(config)# lldp enable
```

step 3 Enter the interface configure mode and set the attributes of LLDP on the interface

```
Switch(config)# interface eth-0-9
Switch(config)# no shutdown
Switch(config-if)# no lldp tlv 8021-org-specific vlan-name
Switch(config-if)# lldp tlv med location-id ecs-elin 1234567890
Switch(config-if)# lldp enable txrx
Switch(config-if)# exit
```

step 4 Set LLDP timers (optional)

Configure the transmitting interval of LLDP packet to 40 seconds; Configure the transmitting delay of LLDP packet to 3 seconds; Configure the reinit delay of LLDP function to 1 second.

```
Switch(config)# lldp timer msg-tx-interval 40
Switch(config)# lldp timer tx-delay 3
Switch(config)# lldp timer reinitDelay 1
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

To display the LLDP configuration, use following command:

```
Switch# show lldp local config
LLDP global configuration:
============================================================
LLDP function global enabled : YES
LLDP msgTxHold    : 4
LLDP msgTxInterval : 40
LLDP reinitDelay  : 1
LLDP txDelay      : 3
Switch# show lldp local config interface eth-0-9
LLDP configuration on interface eth-0-9 :
============================================================
LLDP admin status : TXRX
Basic optional TLV Enabled:
   Port Description TLV
   System Name TLV
   System Description TLV
   System Capabilities TLV
   Management Address TLV
IEEE 802.1 TLV Enabled:
   Port Vlan ID TLV
   Port and Protocol Vlan ID TLV
   Protocol Identity TLV
IEEE 802.3 TLV Enabled:
   MAC/PHY Configuration/Status TLV
   Power Via MDI TLV
   Link Aggregation TLV
   Maximum Frame Size TLV
LLDP-MED TLV Enabled:
   Med Capabilities TLV
   Network Policy TLV
   Location Identification TLV
   Extended Power-via-MDI TLV
   Inventory TLV
Switch# show running-config
!
lldp enable
lldp timer msg-tx-interval 40
lldp timer reinit-delay 1
lldp timer tx-delay 3
!
interface eth-0-9
lldp enable txrx
 no lldp tlv 8021-org-specific vlan-name
 lldp tlv med location-id ecs-elin 1234567890
!
Switch# show lldp neighbor
Local Port eth-0-1 has 0 neighbor(s)

Local Port eth-0-2 has 0 neighbor(s)
…
Local Port eth-0-9 has 2 neighbor(s)

Remote LLDP Information of port eth-0-9
============================================================
Neighbor Index : 1
Chassis ID type: Mac address
Chassis ID    : 48:16:be:a4:d7:09
Port ID type   : Interface Name
Port ID       : eth-0-9
TTL : 160
Expired time: 134
…
Location Identification :
ECS ELIN: 1234567890
```

Chapter 10 Traffic Managemant Configuration Guide

10. 1 Configuring QoS

10. 1. 1 Overview

Function Introduction

Quality of Service (QoS) can be used to give certain traffic priority over other traffic. Without QoS, all traffic in a network has the same priority and chance of being delivered on time. If congestion occurs, all traffic has the same chance of being dropped. With QoS, specific network traffic can be prioritized to receive preferential treatment. In turn, a network performs more predictably, and utilizes bandwidth more effectively.

Classification information can be carried in the Layer-3 IP packet header or the Layer-2 frame. IP packet headers carry the information using 6 bits or 3 bits from the deprecated IP type of service (TOS) field. Layer-2 802.1Q frames carry the information using a 2-byte Tag Control Information field.

All switches and routers accessing the Internet depend on class information to give the same forwarding treatment to packets with the same class information, and give different treatment to packets with different class information. A packet can be assigned class information, as follows:

- End hosts or switches along a path, based on a configured policy
- Detailed packet examination, expected to occur nearer to the network edge, to prevent overloading core switches and routers
- A combination of the above two techniques

Class information can be used by switches and routers along a path to limit the amount of allotted resources per traffic class.

Per-hop behavior is an individual device's behavior when handling traffic in the DiffServ architecture. An end-to-end QoS solution can be created if all devices along a path have consistent per-hop behavior.

Principle Description

Following is a brief description of terms and concepts used to describe QoS:

ACL

Access control lists (ACLs) classify traffic with the same characteristics. IP traffic is classified using IP ACLs, and non-IP traffic is classified using MAC ACLs. The ACL can have multiple access control entries (ACEs), which are commands that match fields against the contents of the packet.

CoS Value

Class of Service (CoS) is a 3-bit value used to classify the priority of Layer-2 frames upon entry into a network.

QoS classifies frames by assigning priority-indexed CoS values to them, and gives preference to higher-priority traffic.

Layer-2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS values in the 3 most significant bits, called the User Priority bits. On interfaces configured as Layer-2 802.1Q trunks, all traffic is in 802.1Q frames, except for traffic in the native VLAN.

Other frame types cannot carry Layer-2 CoS values. CoS values range from 0 to 7.

DSCP Value

Differentiated Services Code Point (DSCP) is a 6-bit value used to classify the priority of Layer-3 packets upon entry into a network.

DSCP values range from 0 to 63.

IP-Precedence Value

IP-Precedence is a 3-bit value used to classify the priority of Layer-3 packets upon entry into a network.

IP-Precedence values range from 0 to 7.

EXP Value

EXP value is a 3-bit value used to classify the priority of MPLS packets upon entry into a network.

MPLS EXP values range from 0 to 7.

Classification

Classification distinguishes one kind of traffic from another by examining the fields in the packet. The process generates an internal priority for a packet, which identifies all future QoS actions to be taken on the packet.

Each packet is classified upon entry into the network. At the ingress, the packet is inspected, and the priority is determined based on ACLs or the configuration. The Layer-2 CoS value is then mapped to a priority value.

The classification is carried in the IP packet header using 6 bits or 3 bits from the deprecated IP TOS field to carry the classification information. Classification can also occur in the Layer-2 frame.

Classification is enabled only if QoS is globally enabled on the switch. By default, QoS is globally disabled, thus, no classification occurs.

Classification occurs on an ingress physical port, but not at the switch virtual interface level.

Classification can be based on CoS/inner-CoS/DSCP/IP-Precedence, default port cos, or class maps and policy maps.

Shaping

Shaping is to change the rate of incoming traffic flow to regulate the rate in such a way that the outgoing traffic flow behaves more smoothly. If the incoming traffic is highly bursty, it needs to be buffered so that the output of the buffer is less bursty and smoother.

Shaping has the following attributes:

- Shaping can be deployed base on physical port.
- Shaping can be deployed on queues of egress interface.

Policing

Policing determines whether a packet is in or out of profile by comparing the internal priority to the configured policer.

The policer limits the bandwidth consumed by a traffic flow. The result is given to the marker.

There are two types of policers:

- Individual: QoS applies the bandwidth limits specified in the policer, separately, to each matched traffic class. An individual policer is configured within a policy map.
- Aggregate: QoS applies the bandwidth limits specified in an aggregate policer, cumulatively, to all matched traffic flows. An aggregate policer is configured by specifying the policer name within a policy map. The bandwidth limits of the policer are specified. In this way, the aggregate policer is shared by multiple classes of traffic within one or multiple policy map.

Marking

Marking determines how to handle a packet when it is out of profile. It assesses the policer and the configuration information to determine the action required for the packet, and then handles the packet using one of the following methods:

- Let the packet through and mark color down
- Drop the packet

Marking can occur on ingress and egress interfaces.

Queuing

Queuing maps packets to a queue. Each egress port can accommodate up to 8 unicast queues, 4 multicast queues and 1 SPAN queue.

The packet internal priority can be mapped to one of the egress queues. The unit of queue depth is buffer cell. Buffer cell is the granularity, which is 288 bytes, for packet storing.

After the packets are mapped to a queue, they are scheduled.

Tail Drop

Tail drop is the default congestion-avoidance technique on the interface. With tail drop, packets are queued until the thresholds are exceeded. The packets with different priority and color are assigned to different drop precedence. The mapping between priority and color to queue and drop precedence is configurable. You can modify the three tail-drop threshold to every egress queue by using the queue threshold interface configuration command. Each threshold value is packet buffer cell, which ranges from 0 to 16383.

WRED

Weighted Random Early Detection (WRED) differs from other congestion-avoidance techniques because it attempts to anticipate and avoid congestion, rather than controlling congestion when it occurs.

WRED reduces the chances of tail drop by selectively dropping packets when the output interface begins to show signs of congestion. By dropping some packets early rather than waiting until the queue is full, WRED avoids dropping large numbers of packets at once. Thus, WRED allows the transmission line to be fully used at all times. WRED also drops more packets from large users than small. Therefore, sources that generate the most traffic are more likely to be slowed down versus sources that generate little traffic.

You can enable WRED and configure the two thresholds for a drop-precedence assigned to every egress queues. The WRED's color drop precedence map is the same as tail-drop's. Each min-threshold represents where WRED starts to randomly drop packets. After min-threshold is exceeded, WRED randomly begins to drop packets assigned to this threshold. As the queue max-threshold is approached, WRED continues to drop packets randomly with the rate of drop-probability. When the max-threshold is reached, WRED drops all packets assigned to the threshold. By default, WRED is disabled.

Scheduling

Scheduling forwards conditions packets using combination of WDRR and SP. Every queue belongs to a class. The class range from 0 to 7, and 7 is the highest priority. Several queues can be in a same class, or non queue in some class. Packets are scheduled by SP between classes and WDRR between queues in a class.

- Strict Priority-Based (SP), in which any high-priority packets are first transmitted. Lower-priority packets are transmitted only when the higher-priority queues are empty. A problem may occur when too many lower-priority packets are not transmitted.
- Weighted Deficit Round Robin (WDRR), in which each queue is assigned a weight to control the number of packets relatively sent from each queue.

Class Map

A class map names and isolates specific traffic from other traffic. The class map defines the criteria used to match against a specific traffic flow to further classify it. The criteria can match several access groups defined by the ACL.

If there is more than one type of traffic to be classified, another class map can be created under a different name. After a packet is matched against the class-map criteria, it is further classified using a policy map.

Policy Map

A policy map specifies on which traffic class to act. This can be implemented as follows:

- Set a specific priority and color in the traffic class.
- Set a specific trust policy to map priority and color.
- Specify the traffic bandwidth limitations for each matched traffic class (policer) and the action to take (marking) when the traffic is out of profile.
- Redirect the matched traffic class to a specific physical interface.
- Mirror the matched traffic class to a specific monitor session, which's destination is defined in mirror module(please refer to the "monitor session destination" command).
- Enable statistics of matching each ace or each class-map(if the class-map operator is match-any).
- Policy maps have the following attributes:
- A policy map can contain multiple class statements, each with different match criteria and action.
- A separate policy-map class can exist for each type of traffic received through an interface.
- There can be only one policy map per interface per direction. The same policy map can be applied to multiple interfaces and directions.
- Before a policy map can be effective, it must be attached to an interface.
- A policy map can be applied on physical interface(not link agg member), link agg interface, or vlan interface.

Mapping Tables

During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with an internal priority value:

- During classification, QoS uses configurable mapping tables to derive the internal priority (a 6-bit value) from received CoS, EXP(3-bit), DSCP or IP precedence (3-bit) values. These maps include the CoS-to-priority-color/COS-to-PHB map, EXP-to-priority-color/EXP-to-PHB map, DSCP-to-priority-color/DSCP-to-PHB map and the IP-precedence-to- priority-color/IP-PREC-to-PHB map.
- During policing, QoS can assign another priority and color to an IP or non-IP packet (if the packet matches the class-map). This configurable map is called the policed-priority-color map.
- Before the traffic reaches the scheduling stage, and replace CoS or DSCP is set, QoS uses the configurable priority-color-to-CoS or priority-color-to-DSCP map to derive a CoS or DSCP value from the internal priority color.
- Each QoS domain has an independent set of map tables mentioned above.

Time-range

By using time-range, the aces in the class-map can be applied based on the time of day or week. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when adding an ace. You can use the time-range to define when the aces in the class-map are in effect, for example, during a specified time period or on specified days of the week.

These are some of the many possible benefits of using time-range:

- You can control over permitting or denying a user access to resources, such as an application, which is identified by an IP address and a port number.
- You can obtain the traffic statistics during appointed time.
- You can define when the action of a traffic class is in effect.

SRTCM

Single Rate Three Color Marker

TRTCM

Two Rate Three Color Marker

CIR

Committed Information Rate

CBS

Committed Burst Size

EIR

Excess Information Rate

EBS

Excess Burst Size

PIR

Peak Information Rate

PBS

Peak Burst Size

10. 1. 2 Configuration

The following provides information to consider before configuring QoS:

- QoS policing cannot be configured on Linkagg interface.
- Traffic can be only classified per ingress port.

- There can be multiple ACLs per class map. An ACL can have multiple access control entries that match fields against the packet contents.
- QoS policing cannot be configured on virtual interface.
- When queeu shaping is configured, the total CIR of all queues on the port cannot neither port speed nor port shaping speed.

Enable QoS

By default, QoS is disabled on the switch, which means that the switch offers best-effort service to each packet regardless of the packet contents or size. All the packets map to egress queue 0 with both tail-drop thresholds set to 100 percent of the total queue size. When the buffer is full, packets are dropped.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS

```
Switch(config)# qos enable
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show qos
 Enable
```

Configuring Tail Drop

Tail drop is the default congestion-avoidance technique on every egress queue. With tail drop, packets are queued until the thresholds are exceeded. The following shows configuring tail drop threshold for different drop-precedence. Follow these steps from Privileged Exec mode.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and configure the tail drop

queue <QID> tail-drop threshold THRESHOLD0 THRESHOLD1 THRESHOLD2 to set threshold for different drop precedence. THRESHOLD0 = threshold for drop precedence 0 packets, range is 0-12284. THRESHOLD1 = threshold for drop precedence 1 packets, range is 0-12285. THRESHOLD2 = threshold for drop precedence 2 packet, range is 0-12286.

Configure drop precedence0 packet drop threshold is 2000, drop precedence1 packet drop threshold is 3000, and drop precedence2 packet drop threshold is 4000

```
Switch(config)# interface eth-0-1
Switch(config-if)# queue 3 tail-drop threshold 2000 3000 4000
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show qos interface eth-0-1
 Interface QoS domain: 0
 Interface trust state: cos
 Interface default CoS value: 0

  Schedule mode: SP(between Class), WDRR(between queue in the same Class)
  The number of class on interface: 8
  Strict priority class ID: 7 6 5 4 3 2 1 0
  The number of egress queue: 8
```

```
  Queue 0 class 0, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 1 class 1, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 2 class 2, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 3 class 3, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
  Queue 4 class 4, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 5, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 7, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

Configuring WRED

WRED reduces the chances of tail drop by selectively dropping packets when the output interface detects congestion. By dropping some packets early rather than waiting until the queue is full, WRED avoids TCP synchronization dropping and thereafter improves the overall network throughput. The following shows configuring WRED threshold for different color. Follow these steps from Privileged Exec mode.

The following example shows configuring WRED threshold for queue 1. In this example, the min-threshold for drop precedence 0, drop precedence 1, and drop precedence 2 packet is 32, 48, and 64, respectively; the max-threshold is 596, 612, and 628, respectively. If buffered packets exceed min-threshold, the subsequent packet will be dropped randomly with rate of 1024/65535 by default.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and enable WRED and set the threshold

```
Switch(config)# queue 1 random-detect
Switch(config-if)# queue 1 random-detect max-threshold 596 612 628
Switch(config-if)# queue 1 random-detect min-threshold 32 48 64
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show qos interface eth-0-1
 Interface QoS domain: 0
 Interface trust state: cos
 Interface default CoS value: 0

  Schedule mode: SP(between Class), WDRR(between queue in the same Class)
  The number of class on interface: 8
  Strict priority class ID: 7 6 5 4 3 2 1 0
  The number of egress queue: 8
    Queue 0 class 0, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 1 class 1, DRR weight 1
      WRED drop mode
      WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
      Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
      Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
```

```
    Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
  Queue 2 class 2, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 3 class 3, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
  Queue 4 class 4, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 5, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 7, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

Configuring Schedule

Packets are scheduled by SP between different classes and WDRR between queues in the same class.

The following shows mapping queue to different classes and configuring WDRR weight. Follow these steps from Privileged Exec mode.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and configure the schedule

```
Switch(config)# interface eth-0-1
Switch(config-if)# queue 6 class 6
Switch(config-if)# queue 5 class 6
Switch(config-if)# queue 6 drr-weight 20
Switch(config-if)# queue 5 drr-weight 30
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show qos interface eth-0-1
 Interface QoS domain: 0
 Interface trust state: cos
 Interface default CoS value: 0

  Schedule mode: SP(between Class), WDRR(between queue in the same Class)
  The number of class on interface: 8
  Strict priority class ID: 7 6 5 4 3 2 1 0
  The number of egress queue: 8
    Queue 0 class 0, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 1 class 1, DRR weight 1
      WRED drop mode
      WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
      Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
      Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
      Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
    Queue 2 class 2, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 3 class 3, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
```

```
  Queue 4 class 4, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 6, DRR weight 20
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 20
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 6, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

Configuring Port policing

All traffic received or transmitted in the physical interface can be limited rate, and all the exceeding traffic will be dropped.

The following shows creating a port-policer to limit bandwidth. Follow these steps from Privileged Exec mode.

The no port-policier input|output command deletes a port policer. The following example shows creating an ingress port policer. In this example, if the received traffic exceeds a 48000-kbps average traffic rate, it is dropped.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and configure the policing

Configure 48000-kbps average traffic rate to be limited

```
Switch(config)# interface eth-0-1
Switch(config-if)# port-policer input mode rfc2697 color-blind cir 48000 cbs 10000 ebs 128000 drop-color red
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show qos interface eth-0-1
 Input port policer:
   mode rfc2697, CIR 48000 kbps, CBS 10000 bytes, EBS 128000 bytes, color blind mode, drop color is red

 Interface QoS domain: 0
 Interface trust state: cos
 Interface default CoS value: 0

   Schedule mode: SP(between Class), WDRR(between queue in the same Class)
   The number of class on interface: 8
   Strict priority class ID: 7 6 5 4 3 2 1 0
   The number of egress queue: 8
     Queue 0 class 0, DRR weight 1
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
     Queue 1 class 1, DRR weight 1
       WRED drop mode
       WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
       Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
       Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
       Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
     Queue 2 class 2, DRR weight 1
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
     Queue 3 class 3, DRR weight 1
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
     Queue 4 class 4, DRR weight 1
       Tail drop mode
```

```
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 6, DRR weight 20
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 20
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 6, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

Configuring Shaping

All traffic transmitted in the physical interface can be shaped, and all the exceeding traffic will be buffered. If no buffer, it is dropped.

The following shows creating a port shaping to shape traffic. Follow these steps from Privileged Exec mode.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and configure the Shaping

Configure the received traffic exceeds a 50 percent of the whole interface bandwidth, it will be buffered

```
Switch(config)# interface eth-0-1
Switch(config-if)# shape average percent 50
Switch(config-if)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show qos interface eth-0-1
 Input port policer:
   mode rfc2697, CIR 48000 kbps, CBS 10000 bytes, EBS 16000 bytes, color blind mode, drop color is red

 Interface QoS domain: 0
 Interface trust state: cos
 Interface default CoS value: 0

   Schedule mode: SP(between Class), WDRR(between queue in the same Class)
   The number of class on interface: 8
   Strict priority class ID: 7 6 5 4 3 2 1 0
   Shape with rate 500000 kbps
   The number of egress queue: 8
     Queue 0 class 0, DRR weight 1
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
     Queue 1 class 1, DRR weight 1
       WRED drop mode
       WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
       Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
       Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
       Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
     Queue 2 class 2, DRR weight 1
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
     Queue 3 class 3, DRR weight 1
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
     Queue 4 class 4, DRR weight 1
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
     Queue 5 class 6, DRR weight 20
       Tail drop mode
       Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

```
  Queue 6 class 6, DRR weight 20
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 6, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

Configuring Policy

To configure a QoS policy, the following is usually required:

- Categorize traffic into classes.
- Configure policies to apply to the traffic classes.
- Attach policies to interfaces.

Classify Traffic Using ACLs

IP traffic can be classified using IP ACLs.

The following shows creating an IP ACL for IP traffic. Follow these steps from Privileged Exec mode.

The no ip access-list command deletes an access list.

The following example shows allowing access only for hosts on three specified networks. Wildcard bits correspond to the network address host portions. If a host has a source address that does not match the access list statements, it is rejected.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Configure the ACL and ACEs

```
Switch(config)# ip access-list ip-acl
Switch(config-ip-acl)# permit any 128.88.12.0 0.0.0.255 any
Switch(config-ip-acl)# permit any 28.88.0.0 0.0.255.255 any
Switch(config-ip-acl)# permit any 11.0.0.0 0.255.255.255 any
Switch(config-ip-acl)# exit
```

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch# show access-list ip ip-acl
ip access-list ip-acl
 10 permit any 128.88.12.0 0.0.0.255 any
 20 permit any 28.88.0.0 0.0.255.255 any
 30 permit any 11.0.0.0 0.255.255.255 any
```

Create class-map

The following shows classifying IP traffic on a physical-port basis using class maps. This involves creating a class map, and defining the match criterion.

If neither the match-any or match-all keyword is specified, the default is match-any. match access-group NAME to define the match criterion. NAME = name of the ACL created using the ip access-list command.

The no class-map command deletes an existing class-map. The following example shows configuring a class map named cmap1 with 1 match criterion: IP access list ip-acl, which allows traffic from any source to any destination.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Configure the ACL and ACEs

```
Switch(config)# ip access-list ip-acl
Switch(config-ip-acl)# permit any any any
Switch(config-ip-acl)# quit
```

step 3 Create and enter into class-map cmap1 mode, Configure ip-acl into cmap1

class-map (match-any|match-all) NAME to create a class map. match-any = Use the match-any keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. match-all = Use the match-all keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. NAME = name of the class map.

```
Switch(config)# class-map cmap1
Switch (config-cmap)# match access-group ip-acl
Switch (config-cmap)# quit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch# show class-map cmap1
  CLASS-MAP-NAME: cmap1 (match-any)
    match access-group: ip-acl
```

Create Policy Map

The following shows creating a policy map to classify, policer, and mark traffic.

There can be only one policy map per interface per direction. The no policy-map command deletes an existing policy-map. The no set priority color command removes a specified priority color value. The no policer command removes an existing policer. The no trust command removes trust policy. The no service-policy input|output command removes a policy map from interface.

The following example shows creating a policy map, and attaching it to an ingress interface. In this example, the IP ACL allows traffic from network 10.1.0.0. If the matched traffic exceeds a 48000-kbps average traffic rate, it is dropped.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS

```
Switch(config)# qos enable
```

step 3 Configure the ACL and ACEs

```
Switch(config)# ip access-list ip-acl
Switch(config-ip-acl)# permit any 10.1.0.0 0.0.255.255 any
Switch(config-ip-acl)# quit
```

step 4 Create and enter into class-map cmap1 mode, Configure ip-acl into cmap1

```
Switch(config)# class-map cmap1
Switch(config-cmap)# match access-group ip-acl
Switch(config-cmap)# quit
```

step 5 Create and enter into policy-map pmap1 mode, Attach class-map cmap1 into policy-map pmap1, Configure 48000-kbps average traffic rate to be limited

```
Switch(config)# policy-map pmap1
Switch(config-pmap)# class cmap1
Switch(config-pmap-c)# policer mode rfc2697 color-blind cir 48000 cbs 10000 ebs 128000 drop-color red
Switch(config-pmap-c)# quit
Switch(config-pmap)# quit
```

step 6 Enter the interface configure mode and Attach policy-map pmap1 to interface

```
Switch(config)# interface eth-0-1
Switch(config-if)# service-policy input pmap1
Switch(config-if)# exit
```

step 7 Exit the configure mode

```
Switch(config)# end
```

step 8 Validation

```
Switch# show policy-map pmap1
 POLICY-MAP-NAME: pmap1
  State: attached

  CLASS-MAP-NAME: cmap1
   match access-group: ip-acl
     mode rfc2697, CIR 48000 kbps, CBS 10000 bytes, EBS 128000 bytes, color blind mode, drop color is red
```

Create Aggregate Policer

The following shows creating an aggregate policer to classify, police, and mark traffic.

There can be only one policy map per interface per direction.

The no policer-aggregate command deletes an aggregate policer from a policy map. The no qos aggregate-policer command deletes an aggregate policer.

The following example shows creating an aggregate policer, and attaching it to multiple classes within a policy map. In this example, the IP ACLs allow traffic from network 10.1.0.0 and host 11.3.1.1. The traffic rate from network 10.1.0.0 and host 11.3.1.1 is policed. If the traffic exceeds a 48000-kbps average traffic rate and an 8000-byte normal burst size, it is considered out of profile, and is dropped. The policy map is attached to an ingress interface.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS

```
Switch(config)# qos enable
```

step 3 Configure the ACLs and ACEs

```
Switch(config)# ip access-list ip-acl1
Switch(config-ip-acl)# permit any 10.1.0.0 0.0.255.255 any
Switch(config-ip-acl)# exit
Switch(config)# ip access-list ip-acl2
Switch(config-ip-acl)# permit any host 11.3.1.1 any
Switch(config-ip-acl)# exit
```

step 4 Configure 48000-kbps average traffic rate to be limited

```
Switch(config)# qos aggregate-policer transmit1 mode rfc2697 color-blind cir 48000 cbs 8000 ebs 10000 drop-color red
```

step 5 Create and enter into class-map cmap1 mode, Configure ip-acl into cmap1

```
Switch(config)# class-map cmap1
Switch(config-cmap)# match access-group ip-acl1
Switch(config-cmap)# exit
Switch(config)# class-map cmap2
Switch(config-cmap)# match access-group ip-acl2
Switch(config-cmap)# exit
```

step 6 Create and enter into policy-map mode, Attach class-map into policy-map, Set cmap1 as policer-aggregate transmit1

```
Switch(config-pmap-c)# policer-aggregate transmit1
Switch(config-pmap-c)# exit
```

```
Switch(config-pmap)# class cmap2
Switch(config-pmap-c)# set priority 56 color green
Switch(config-pmap-c)# policer-aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

step 7 Enter the interface configure mode and Attach policy-map to interface

```
Switch(config)# interface eth-0-1
Switch(config-if)# service-policy input aggflow1
Switch(config-if)# exit
```

step 8 Exit the configure mode

```
Switch(config)# end
```

step 8 Validation

```
Switch# show qos aggregator-policer
AGGREGATOR-POLICER-NAME: transmit1
  mode rfc2697, CIR 48000 kbps, CBS 8000 bytes, EBS 10000 bytes, color blind mode, drop color is red
```

Configuring QoS Mapping tables

CoS-Priority-Color Map:

The following shows modifying a CoS-Priority-Color map. This map is used to generate an internal priority color value from CoS during classification; this value determines the QoS action in the DUT, such as selecting one of the eight egress queues, etc. The CoS value can also came from the inner cos of incoming packets, if the port trusts inner cos.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS globally

```
Switch(config)# qos enable
```

step 3 Configure mapping cos 1 to priority 63 color green for QoS domain 1

```
Switch(config)# qos domain 1 map cos-pri-color cos 1 to 63 green
```

step 4 Enter the interface configure mode and configure interface eth-0-1 to QoS domain 1, configure to trust cos

```
Switch(config)# interface eth-0-1
Switch(config-if)# qos domain 1
Switch(config-if)# trust cos
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show qos domain 1 map-table ingress cos-priority-color running
QoS DOMAIN 1, CFI disable, COS map to PRIORITY & COLOR:
-----------------------------------------------
 COS    : 0    1    2    3    4    5    6    7
 priority: 0    63    16    24    32    40    48    56
 color  : green  green  green  green  green  green  green  green

Switch# show qos interface eth-0-1
 Input port policer:
   mode rfc2697, CIR 48000 kbps, CBS 10000 bytes, EBS 16000 bytes, color blind mode, drop color is red

 Interface QoS domain: 1
```

```
Interface trust state: cos
Interface default CoS value: 0

 Schedule mode: SP(between Class), WDRR(between queue in the same Class)
 The number of class on interface: 8
 Strict priority class ID: 7 6 5 4 3 2 1 0
 Shape with rate 500000 kbps
 The number of egress queue: 8
  Queue 0 class 0, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 1 class 1, DRR weight 1
   WRED drop mode
   WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
   Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
   Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
   Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
  Queue 2 class 2, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 3 class 3, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
   Queue shape with CIR 100000 kbps, PIR 100000 kbps
  Queue 4 class 4, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 6, DRR weight 20
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 20
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 6, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

IP-Precedence-Priority-Color Map:

The following shows modifying an IP-Precedence-Priority-Color map. This map is used to generate an internal priority color value from IP-Precedence during classification; this value determines the QoS action in the DUT, such as selecting one of the eight egress queues, etc.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS globally

```
Switch(config)# qos enable
```

step 3 Configure mapping ip-prec 1 to priority 63 color green for QoS domain 1

```
Switch(config)# qos domain 1 map ip-prec-pri-color ip-prec 1 to 63 green
```

step 4 Enter the interface configure mode and configure interface eth-0-1 to QoS domain 1, configure to trust ip-prec

```
Switch(config)# interface eth-0-1
Switch(config-if)# qos domain 1
Switch(config-if)# trust ip-prec
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show qos domain 1 map-table ingress ip-prec-priority-color running
QoS DOMAIN 1, IP PRECEDENCE map to PRIORITY & COLOR:
```

```
-----------------------------------------------
IP-prec : 0    1    2    3    4    5    6    7
priority: 0    63   16   24   32   40   48   56
color   : green green green green green green green green
```

?
Switch# show qos domain 1 map-table ingress ip-prec-priority-color default QoS DOMAIN 1, IP PRECEDENCE map to PRIORITY & COLOR: ——————————————————- IP-prec : 0 1 2 3 4 5 6 7 priority: 0 8 16 24 32 40 48 56 color : green green green green green green green green

?
Switch# show qos interface eth-0-1 Input port policer: mode rfc2697, CIR 48000 kbps, CBS 10000 bytes, EBS 16000 bytes, color blind mode, drop color is red

```
Interface QoS domain: 1
Interface trust state: ip-prec
Interface default CoS value: 0

  Schedule mode: SP(between Class), WDRR(between queue in the same Class)
  The number of class on interface: 8
  Strict priority class ID: 7 6 5 4 3 2 1 0
  Shape with rate 500000 kbps
  The number of egress queue: 8
    Queue 0 class 0, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 1 class 1, DRR weight 1
      WRED drop mode
      WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
      Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
      Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
      Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
    Queue 2 class 2, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 3 class 3, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
      Queue shape with CIR 100000 kbps, PIR 100000 kbps
    Queue 4 class 4, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 5 class 6, DRR weight 20
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 6 class 6, DRR weight 20
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 7 class 6, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

EXP-Priority-Color Map:

The following shows modifying an EXP-Priority-Color map. This map is used to generate an internal priority color value from MPLS EXP during classification; this value determines the QoS action in the DUT, such as selecting one of the eight egress queues, etc.

The following example shows mapping EXP 1 to priority 63 color green for QoS domain 1, and configure interface eth-0-1 to QoS domain 1 with trust dscp.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS globally

```
Switch(config)# qos enable
```

step 3 Configure mapping EXP 1 to priority 63 color green for QoS domain 1

```
Switch(config)# qos domain 1 map exp-pri-color exp 1 to 63 green
```

step 4 Enter the interface configure mode and configure interface eth-0-1 to QoS domain 1, configure to trust dscp, trust dscp for ip packets and trust exp for mpls packets

```
Switch(config)# interface eth-0-1
Switch(config-if)# qos domain 1
Switch(config-if)# trust dscp-exp
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show qos domain 1 map-table ingress exp-priority-color running
QoS DOMAIN 1, EXP map to PRIORITY & COLOR:
-----------------------------------------
 EXP    : 0    1    2    3    4    5    6    7
 priority: 0    63   16   24   32   40   48   56
 color  : green green green green green green green green
```

?
Switch# show qos domain 1 map-table ingress exp-priority-color default QoS DOMAIN 1, EXP map to PRIORITY & COLOR: ———————————————————— EXP : 0 1 2 3 4 5 6 7 priority: 0 8 16 24 32 40 48 56
color : green green green green green green green green

?
Switch# show qos interface eth-0-1 Input port policer: mode rfc2697, CIR 48000 kbps, CBS 10000 bytes, EBS 16000 bytes, color blind mode, drop color is red

```
 Interface QoS domain: 1
 Interface trust state: dscp-exp
 Interface default CoS value: 0

  Schedule mode: SP(between Class), WDRR(between queue in the same Class)
  The number of class on interface: 8
  Strict priority class ID: 7 6 5 4 3 2 1 0
  Shape with rate 500000 kbps
  The number of egress queue: 8
    Queue 0 class 0, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 1 class 1, DRR weight 1
      WRED drop mode
      WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
      Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
      Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
      Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
    Queue 2 class 2, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 3 class 3, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
      Queue shape with CIR 100000 kbps, PIR 100000 kbps
    Queue 4 class 4, DRR weight 1
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 5 class 6, DRR weight 20
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 6 class 6, DRR weight 20
      Tail drop mode
      Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
    Queue 7 class 6, DRR weight 1
```

```
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

DSCP-Priority-Color Map:

The following shows modifying a DSCP-Priority-Color map. This map is used to generate an internal priority color value from DSCP during classification; this value determines the QoS action in the DUT, such as selecting one of the eight egress queues, etc.

The following example shows mapping DSCP 34 to priority 63 color green for QoS domain 1, and configure interface eth-0-1 to QoS domain 1 with trust dscp.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS globally

```
Switch(config)# qos enable
```

step 3 Configure mapping DSCP 34 to priority 63 color green for QoS domain 1

```
Switch(config)# qos domain 1 map dscp-pri-color 34 to 63 green
```

step 4 Enter the interface configure mode and configure interface eth-0-1 to QoS domain 1, Configure to trust dscp. trust dscp for ip packets and trust exp for mpls packets

```
Switch(config)# interface eth-0-1
Switch(config-if)# qos domain 1
Switch(config-if)# trust dscp-exp
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show qos domain 1 map-table ingress dscp-priority-color running
QoS DOMAIN 1, DSCP map to PRIORITY & COLOR:
-----------------------------------------------
 DSCP   : 0    1    2    3    4    5    6    7
 priority: 0    1    2    3    4    5    6    7
 color  : green green green green green green green green
-----------------------------------------------
 DSCP   : 8    9    10   11   12   13   14   15
 priority: 8    9    10   11   12   13   14   15
 color  : green green green green green green green green
-----------------------------------------------
 DSCP   : 16   17   18   19   20   21   22   23
 priority: 16   17   18   19   20   21   22   23
 color  : green green green green green green green green
-----------------------------------------------
 DSCP   : 24   25   26   27   28   29   30   31
 priority: 24   25   26   27   28   29   30   31
 color  : green green green green green green green green
-----------------------------------------------
 DSCP   : 32   33   34   35   36   37   38   39
 priority: 32   33   63   35   36   37   38   39
 color  : green green green green green green green green
-----------------------------------------------
 DSCP   : 40   41   42   43   44   45   46   47
 priority: 40   41   42   43   44   45   46   47
 color  : green green green green green green green green
-----------------------------------------------
 DSCP   : 48   49   50   51   52   53   54   55
 priority: 48   49   50   51   52   53   54   55
 color  : green green green green green green green green
-----------------------------------------------
 DSCP   : 56   57   58   59   60   61   62   63
```

```
priority: 56    57    58    59    60    61    62    63
color   : green  green  green  green  green  green  green  green
```

?
?
?
Switch# show qos domain 1 map-table ingress dscp-priority-color default QoS DOMAIN 1, DSCP map to PRIORITY & COLOR: ———————
—————————- DSCP : 0 1 2 3 4 5 6 7
priority: 0 1 2 3 4 5 6 7
color : green green green green green green green green
———————————————- DSCP : 8 9 10 11 12 13 14 15
priority: 8 9 10 11 12 13 14 15
color : green green green green green green green green
———————————————- DSCP : 16 17 18 19 20 21 22 23
priority: 16 17 18 19 20 21 22 23
color : green green green green green green green green
———————————————- DSCP : 24 25 26 27 28 29 30 31
priority: 24 25 26 27 28 29 30 31
color : green green green green green green green green
———————————————- DSCP : 32 33 34 35 36 37 38 39
priority: 32 33 34 35 36 37 38 39
color : green green green green green green green green
———————————————- DSCP : 40 41 42 43 44 45 46 47
priority: 40 41 42 43 44 45 46 47
color : green green green green green green green green
———————————————- DSCP : 48 49 50 51 52 53 54 55
priority: 48 49 50 51 52 53 54 55
color : green green green green green green green green
———————————————- DSCP : 56 57 58 59 60 61 62 63
priority: 56 57 58 59 60 61 62 63
color : green green green green green green green green

?
Switch# show qos interface eth-0-1 Input port policer: mode rfc2697, CIR 48000 kbps, CBS 10000 bytes, EBS 16000 bytes, color blind mode, drop color is red

```
Interface QoS domain: 1
Interface trust state: dscp-exp
Interface default CoS value: 0

 Schedule mode: SP(between Class), WDRR(between queue in the same Class)
 The number of class on interface: 8
 Strict priority class ID: 7 6 5 4 3 2 1 0
 Shape with rate 500000 kbps
 The number of egress queue: 8
  Queue 0 class 0, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 1 class 1, DRR weight 1
   WRED drop mode
   WRED Exponential-Weighted-Moving-Average (EWMA) factor: 9
   Max threshold(Tresh0 Tresh1 Tresh2): 596  612  628
   Min threshold(Tresh0 Tresh1 Tresh2): 32  48  64
   Drop probability(Tresh0 Tresh1 Tresh2): 1024/65536  1024/65536  1024/65536
  Queue 2 class 2, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 3 class 3, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 2000  3000  4000
   Queue shape with CIR 100000 kbps, PIR 100000 kbps
  Queue 4 class 4, DRR weight 1
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 6, DRR weight 20
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 20
   Tail drop mode
   Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 6, DRR weight 1
```

```
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

Priority-Color-CoS Map:

The following shows modifying a Priority-Color-CoS map. This map is used to generate a new CoS from the internal priority color value in egress; This map is used if two domains have different CoS definitions; this map translates a set of one domain's CoS values to match the other domain's definition.

The following example shows mapping priority 63 color green to CoS 6, and replace CoS in the interface eth-0-1 egress.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS globally

```
Switch(config)# qos enable
```

step 3 Configure mapping priority 63 color green to CoS 6

```
Switch(config)# qos domain 1 map pri-color-cos 63 green to 6
```

step 4 Enter the interface configure mode and configure interface eth-0-1 to QoS domain 1, Configure to replace cos

```
Switch(config)# interface eth-0-1
Switch(config-if)# qos domain 1
Switch(config-if)# replace cos
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show qos domain 1 map-table egress priority-color-cos running
QoS DOMAIN 1, CFI disable, PRIORITY & COLOR map to COS:
          | COLOR:
          | red    yellow  green
-----------------------------------------
 PRIORITY: 0  | 0      0      0
        1  | 0      0      0
        2  | 0      0      0
        3  | 0      0      0
        4  | 0      0      0
        5  | 0      0      0
        6  | 0      0      0
        7  | 0      0      0
        8  | 1      1      1
        9  | 1      1      1
        10 | 1      1      1
        11 | 1      1      1
        12 | 1      1      1
        13 | 1      1      1
        14 | 1      1      1
        15 | 1      1      1
        16 | 2      2      2
        17 | 2      2      2
        18 | 2      2      2
        19 | 2      2      2
        20 | 2      2      2
        21 | 2      2      2
        22 | 2      2      2
        23 | 2      2      2
        24 | 3      3      3
        25 | 3      3      3
        26 | 3      3      3
        27 | 3      3      3
```

```
     28 | 3    3    3
     29 | 3    3    3
     30 | 3    3    3
     31 | 3    3    3
     32 | 4    4    4
     33 | 4    4    4
     34 | 4    4    4
     35 | 4    4    4
     36 | 4    4    4
     37 | 4    4    4
     38 | 4    4    4
     39 | 4    4    4
     40 | 5    5    5
     41 | 5    5    5
     42 | 5    5    5
     43 | 5    5    5
     44 | 5    5    5
     45 | 5    5    5
     46 | 5    5    5
     47 | 5    5    5
     48 | 6    6    6
     49 | 6    6    6
     50 | 6    6    6
     51 | 6    6    6
     52 | 6    6    6
     53 | 6    6    6
     54 | 6    6    6
     55 | 6    6    6
     56 | 7    7    7
     57 | 7    7    7
     58 | 7    7    7
     59 | 7    7    7
     60 | 7    7    7
     61 | 7    7    7
     62 | 7    7    7
     63 | 7    7    6     COS value
```

?
?

Switch# show qos domain 1 map-table egress priority-color-cos default QoS DOMAIN 1, CFI disable, PRIORITY & COLOR map to COS:

COLOR:	red	yellow	green
PRIORITY:	0	0	0	0
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
6	0	0	0
7	0	0	0
8	1	1	1
9	1	1	1
10	1	1	1
11	1	1	1
12	1	1	1
13	1	1	1
14	1	1	1
15	1	1	1
16	2	2	2
17	2	2	2
18	2	2	2
19	2	2	2
20	2	2	2
21	2	2	2
22	2	2	2
23	2	2	2
24	3	3	3
25	3	3	3
26	3	3	3
27	3	3	3
28	3	3	3
29	3	3	3

30	3	3	3
31	3	3	3
32	4	4	4
33	4	4	4
34	4	4	4
35	4	4	4
36	4	4	4
37	4	4	4
38	4	4	4
39	4	4	4
40	5	5	5
41	5	5	5
42	5	5	5
43	5	5	5
44	5	5	5
45	5	5	5
46	5	5	5
47	5	5	5
48	6	6	6
49	6	6	6
50	6	6	6
51	6	6	6
52	6	6	6
53	6	6	6
54	6	6	6
55	6	6	6
56	7	7	7
57	7	7	7
58	7	7	7
59	7	7	7
60	7	7	7
61	7	7	7
62	7	7	7

63 | 7 7 7 COS value

?

Switch# show qos interface eth-0-1 Interface QoS domain: 1 Interface trust state: cos Interface default CoS value: 0 Enable replace CoS

```
 Schedule mode: SP(between Class), WDRR(between queue in the same Class)
The number of class on interface: 8
Strict priority class ID: 7 6 5 4 3 2 1 0
The number of egress queue: 8
  Queue 0 class 0, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 1 class 1, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 2 class 2, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 3 class 3, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 4 class 4, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 5, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 7, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

Priority-Color-DSCP Map:

The following shows modifying a Priority-Color-DSCP map. This map is used to generate a new DSCP from the internal priority color value in egress; This map is used if two domains have different DSCP definitions; this map translates a set of one domain's DSCP values to match the other domain's definition.

The following example shows mapping priority 63 color green to DSCP 60, and replace DSCP in the interface eth-0-1 egress.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable QoS globally

```
Switch(config)# qos enable
```

step 3 Configure mapping priority 63 color green to DSCP 60

```
Switch(config)# qos domain 1 map pri-color-dscp 63 green to 60
```

step 4 Enter the interface configure mode and configure interface eth-0-1 to QoS domain 1, Configure to replace dscp

```
Switch(config)# interface eth-0-1
Switch(config-if)# qos domain 1
Switch(config-if)# replace dscp-exp
Switch(config-if)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show qos domain 1 map-table egress priority-color-dscp running
QoS DOMAIN 1, PRIORITY & COLOR map to DSCP:
          | COLOR:
          | red    yellow  green
-----------------------------------------------
 PRIORITY: 0 | 0      0      0
        1 | 1      1      1
        2 | 2      2      2
        3 | 3      3      3
        4 | 4      4      4
        5 | 5      5      5
        6 | 6      6      6
        7 | 7      7      7
        8 | 8      8      8
        9 | 9      9      9
        10 | 10     10     10
        11 | 11     11     11
        12 | 12     12     12
        13 | 13     13     13
        14 | 14     14     14
        15 | 15     15     15
        16 | 16     16     16
        17 | 17     17     17
        18 | 18     18     18
        19 | 19     19     19
        20 | 20     20     20
        21 | 21     21     21
        22 | 22     22     22
        23 | 23     23     23
        24 | 24     24     24
        25 | 25     25     25
        26 | 26     26     26
        27 | 27     27     27
        28 | 28     28     28
        29 | 29     29     29
        30 | 30     30     30
        31 | 31     31     31
        32 | 32     32     32
```

```
     33 | 33    33    33
     34 | 34    34    34
     35 | 35    35    35
     36 | 36    36    36
     37 | 37    37    37
     38 | 38    38    38
     39 | 39    39    39
     40 | 40    40    40
     41 | 41    41    41
     42 | 42    42    42
     43 | 43    43    43
     44 | 44    44    44
     45 | 45    45    45
     46 | 46    46    46
     47 | 47    47    47
     48 | 48    48    48
     49 | 49    49    49
     50 | 50    50    50
     51 | 51    51    51
     52 | 52    52    52
     53 | 53    53    53
     54 | 54    54    54
     55 | 55    55    55
     56 | 56    56    56
     57 | 57    57    57
     58 | 58    58    58
     59 | 59    59    59
     60 | 60    60    60
     61 | 61    61    61
     62 | 62    62    62
     63 | 63    63    60     DSCP value
```

?

Switch# show qos domain 1 map-table egress priority-color-dscp default QoS DOMAIN 1, PRIORITY & COLOR map to DSCP: | COLOR: | red yellow green

PRIORITY:	0	0	0	0
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5
6	6	6	6	6
7	7	7	7	7
8	8	8	8	8
9	9	9	9	9
10	10	10	10	10
11	11	11	11	11
12	12	12	12	12
13	13	13	13	13
14	14	14	14	14
15	15	15	15	15
16	16	16	16	16
17	17	17	17	17
18	18	18	18	18
19	19	19	19	19
20	20	20	20	20
21	21	21	21	21
22	22	22	22	22
23	23	23	23	23
24	24	24	24	24
25	25	25	25	25
26	26	26	26	26
27	27	27	27	27
28	28	28	28	28
29	29	29	29	29
30	30	30	30	30
31	31	31	31	31
32	32	32	32	32
33	33	33	33	33
34	34	34	34	34
35	35	35	35	35

36	36	36	36
37	37	37	37
38	38	38	38
39	39	39	39
40	40	40	40
41	41	41	41
42	42	42	42
43	43	43	43
44	44	44	44
45	45	45	45
46	46	46	46
47	47	47	47
48	48	48	48
49	49	49	49
50	50	50	50
51	51	51	51
52	52	52	52
53	53	53	53
54	54	54	54
55	55	55	55
56	56	56	56
57	57	57	57
58	58	58	58
59	59	59	59
60	60	60	60
61	61	61	61
62	62	62	62
63 | 63 63 63 DSCP value			

?
Switch# show qos interface eth-0-1 Interface QoS domain: 1 Interface trust state: cos Interface default CoS value: 0 Enable replace DSCP Enable replace CoS

```
 Schedule mode: SP(between Class), WDRR(between queue in the same Class)
The number of class on interface: 8
Strict priority class ID: 7 6 5 4 3 2 1 0
The number of egress queue: 8
  Queue 0 class 0, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 1 class 1, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 2 class 2, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 3 class 3, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 4 class 4, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 5 class 5, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 6 class 6, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
  Queue 7 class 7, DRR weight 1
    Tail drop mode
    Tail drop threshold(Tresh0 Tresh1 Tresh2): 224  240  256
```

10. 1. 3 **Application Cases**

N/A

Chapter 11 VPN Configuration Guide

11. 1 Configuring IPv4 GRE Tunnel

11. 1. 1 Overview

Function Introduction

Tunneling is an encapsulation technology, which uses one network protocol to encapsulate packet of another network protocol and transfer them over a virtual point to point connection. The virtual connection is called a tunnel. Tunneling refers to the whole process from data encapsulation to data transfer to data de-encapsulation.

Principle Description

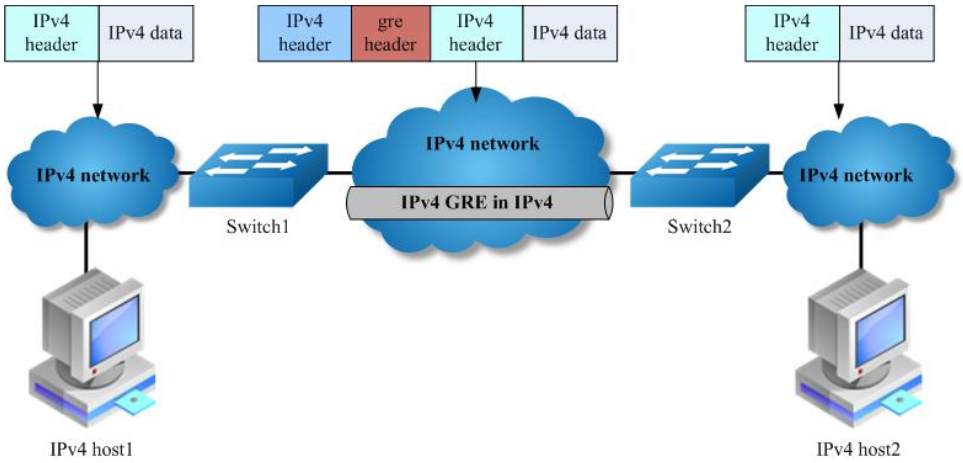


Figure 1-82 IPv4 gre over IPv4

When it is required to communicate with isolated IPv4 networks, you should create a tunnel mechanism between them. The tunnel with transmit protocol of gre connected with two isolated IPv4 island is called IPv4 gre tunnel, which is that IPv4 packets are encapsulated by gre protocol over outer IPv4 packets. Gre tunnel would add gre head in encapsulated packets, including key, sequence, checksum and so on. In order to make an implement of gre tunnel, both tunnel endpoints must support the IPv4 protocol stacks.

IPv4 gre tunnel processes packets in the following ways:

- A host in the IPv4 network sends an IPv4 packet to Switch1 at the tunnel source.
- After determining according to the routing table that the packet needs to be forwarded through the tunnel, Switch1 encapsulates the IPv4 packet with an IPv4 header and forwards it through the physical interface of the tunnel.
- Upon receiving the packet, Switch2 de-encapsulates the packet.
- Switch2 forwards the packet according to the destination address in the de-encapsulated IPv4 packet. If the destination address is the device itself, Switch2 forwards the IPv4 packet to the upper-layer protocol for processing. In the process of de-encapsulation, it would check gre key, only the matched key of packet can be processed, otherwise discarded.

The ip address of tunnel source and tunnel destination is manually assigned, and it provides point-to-point connection. By using overlay tunnels, you can communicate with isolated IPv4 networks without upgrading the IPv4 infrastructure between them. Overlay tunnels can be configured between border routers or between border routers and a host.

The primary use is for stable connections that require regular secure communication between two edge routers or between an end system and an edge router, or for connection to remote IPv4 networks, gre key is alternative configuration.

11. 1. 2 Configuration

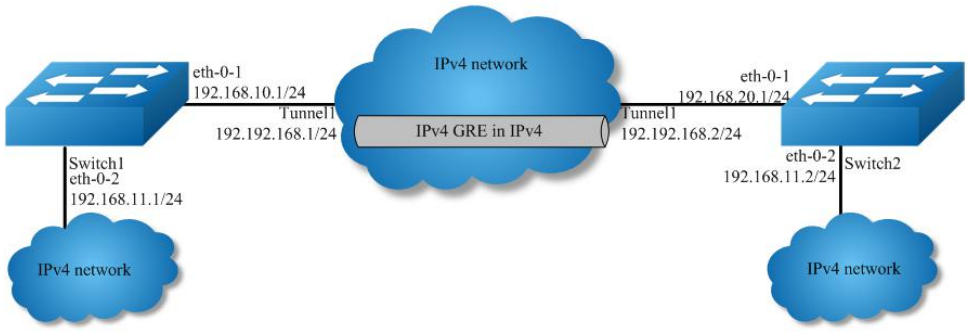


Figure 1-83 IPv4 gre Tunnel

As the topology shows, two IPv4 networks connect to the network via Switch1 and Switch2. An Ipv4 gre tunnel is required between Switch1 and Switch2, in order to connect two networks.

NOTE: A reachable Ipv4 route is necessary for forwarding tunnel packet. Ipv4 address must be configured on tunnel interface; otherwise the route via this tunnel interface is invalid.

The following configuration should be operated on all switches if the switch ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.10.1/24
Switch(config-if)# tunnel enable
Switch(config-if)# exit

Switch(config)# interface eth-0-2
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.11.1/24
Switch(config-if)# exit
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.20.1/24
Switch(config-if)# tunnel enable
Switch(config-if)# exit

Switch(config)# interface eth-0-2
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.11.2/24
Switch(config-if)# exit
```

step 3 Configure the tunnel interface

Tunnel interface configuration for Switch1:

```
Switch(config)# interface tunnel1
Switch(config-if)# tunnel mode gre
Switch(config-if)# tunnel source eth-0-1
```

```
Switch(config-if)# tunnel destination 192.168.20.1
Switch(config-if)# tunnel gre key 100
Switch(config-if)# ip address 192.192.168.1/24
Switch(config-if)# keepalive 5 3
Switch(config-if)# exit
```

Tunnel interface configuration for Switch2:

```
Switch(config)# interface tunnel1
Switch(config-if)# tunnel mode gre
Switch(config-if)# tunnel source eth-0-1
Switch(config-if)# tunnel destination 192.168.10.1
Switch(config-if)# tunnel gre key 100
Switch(config-if)# ip address 192.192.168.2/24
Switch(config-if)# keepalive 5 3
Switch(config-if)# exit
```

step 4 Configure the static route and arp

Configuring Switch1:

```
Switch(config)# ip route 192.168.20.0/24 192.168.10.2
Switch(config)# arp 192.168.10.2 0.0.2222

Switch(config)# ip route 3.3.3.3/24 tunnel1
```

Configuring Switch2:

```
Switch(config)# ip route 192.168.10.0/24 192.168.20.2
Switch(config)# arp 192.168.20.2 0.0.1111

Switch(config)# ip route 4.4.4.4/24 tunnel1
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on Switch1:

```
Switch# show interface tunnel1
Interface tunnel1
 Interface current state: UP
 Hardware is Tunnel
 Index 8193 , Metric 1 , Encapsulation TUNNEL
 VRF binding: not bound
 Internet primary address:
   192.192.168.1/24 pointopoint 192.192.168.255
 Tunnel protocol/transport GRE/IP, Status Valid
 Tunnel source 192.168.10.1(eth-0-1), destination 192.168.20.1
 Tunnel DSCP inherit,  Tunnel TTL 255
 Tunnel GRE key enable: 100
 Tunnel GRE keepalive enable, Send period: 5, Retry times: 3
 0 packets input, 0 bytes
 0 packets output, 0 bytes
```

Display the result on Switch2:

```
Switch# show interface tunnel1
Interface tunnel1
 Interface current state: UP
 Hardware is Tunnel
 Index 8193 , Metric 1 , Encapsulation TUNNEL
 VRF binding: not bound
 Internet primary address:
   192.192.168.2/24 pointopoint 192.192.168.255
 Tunnel protocol/transport GRE/IP, Status Valid
 Tunnel source 192.168.20.1(eth-0-1), destination 192.168.10.1
```

```
Tunnel DSCP inherit,  Tunnel TTL 255
Tunnel GRE key enable: 100
Tunnel GRE keepalive enable, Send period: 5, Retry times: 3
0 packets input, 0 bytes
0 packets output, 0 bytes
```

11. 1. 3 **Application Cases**

N/A

Chapter 12 Reliability Configuration Guide

12. 1 Configuring BHM

12. 1. 1 Overview

Function Introduction

BHM is a module which is used to monitor other Processes. When a monitored Process is uncontrolled, the BHM module will take measures, such as printing warning on screen, shutting all ports, or restarting the system, to help or remind users to recover the system.

The monitored Processes include RIP, RIPNG, OSPF, OSPF6, BGP, LDP, RSVP, PIM, PIM6, 802.1X, LACP MSTP, DHCP-RELAY, DHCP-RELAY6, RMON, OAM, ONM, SSH, SNMP, PTP, SSM. In addition, some system procedures are also monitored, including NSM, IMI, CHSM, HSRVD. There are three activations of BHM, including "reload system", including "reload system","warning", "shutdown port".

Principle Description

N/A

12. 1. 2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enable system monitor and heart-beat-monitor globally

```
Switch(config)# sysmon enable
Switch(config)# heart-beat-monitor enable
```

step 3 Reload system if a monitored PM is uncontrolled

```
Switch(config)# heart-beat-monitor reactivate reload system
```

NOTE: There are three activations of BHM, including "reload system","warning", "shutdown port".

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch# show heart-beat-monitor
heart-beat-monitor enable.
heart-beat-monitor reactivation: restart system.
```

12. 1. 3 Application Cases

N/A

12. 2 Configuring CPU Traffic

12. 2. 1 Overview

Function Introduction

CPU traffic limit is a useful mechanism for protecting CPU from malicious flows by injecting huge volume of PDUs into switches.

CPU traffic limit provides two-level protection for CPU.

- The low-level traffic limit is performed for each reason, which is realized by queue shaping of each type of PDU.
- The high-level traffic limit is performed for all reasons, which is realized by channel shaping at CPU channel.

With this two-level protection, each PDU-to-CPU rate is limited and the overall PDU-to-CPU rate is also limited.

NOTE: The word "reason", means this type of packets will be sent to cpu for further processing.

The description of all reason is as following.

Reason	Description
arp	Address Resolution Protocol
bpdu	Bridge Protocol Data Unit
dhcp	Dynamic Host Configuration Protocol
eapol	Extensible Authentication Protocol Over Lan
erps	Ethernet Ring Protection Switching
fwd-to-cpu	Packets forwarding to cpu
icmp-redirect	ICMP Redirect
igmp	IGMP Snooping Protocol
ip-option	Packets with IP Option
ipda	IP Destination to Router-self
ssh	SSH protocol packet
telnet	Telnet protocol packet
mlag	MLAG protocol packet
tcp	TCP protocol packet
ldp	Label Distribution Protocol
macsa-mismatch	Port Security for source mac learned
mcast-rpf-fail	Multicast with rpf fail or first multicast packet
mpls-ttl-fail	Mpls Packets with ttl fail
ip-mtu-fail	IP packet with mtu fail
ospf	Open Shortest Path First
pim	Protocol Independent Multicast
port-security-discard	Port Security for exceeding fdb maxnum
rip	Routing Information Protocol
sflow-egress	Sampled flow at egress direction
sflow-ingress	Sampled flow at ingress direction
slow-protocol	Slow Protocol (including EFM, LACP and SYNCE)
smart-link	Smart Link Protocol
ucast-ttl-fail	Unicast Packets with ttl fail
udld	Unidirectional Link Detection Protocol
vlan-security-discard	Vlan Security for exceeding fdb maxnum
vrrp	Virtual Router Redundancy Protocol
bfd-learning	BFD learning packets
dot1x-mac-bypass	Mac auth bypass packets
bgp	Border gateway protocol packet

Reason	Description
egress-ttl-fail	Egress ttl fail packet
icmpv6	ICMPv6 packet
l2protocol-tunnel	Layer2 protocol tunnel packet
loopback-detection	lLoopback detection packet
mirror-to-cpu	Mirror to cpu packet
ndp	Neighbor discovery protocol packet
tunnel-gre-keepalive	Tunnel gre keepalive reply packet

The default rate and class configuration for all reason is as following.

Reason	Rate(pps)	Class
arp	256	1
bpdu	64	3
dhcp	128	0
eapol	128	0
erps	128	3
fwd-to-cpu	64	0
icmp-redirect	128	0
igmp	128	2
ip-option	512	0
ipda	1000	0
ssh	64	3
telnet	64	3
mlag	1000	1
tcp	64	2
ldp	512	1
macsa-mismatch	128	0
mcast-rpf-fail	128	1
mpls-ttl-fail	64	0
ip-mtu-fail	64	0
ospf	256	1
pim	128	1
port-security-discard	128	0
rip	64	1
sflow-egress	128	0
sflow-ingress	128	0
slow-protocol	256	1
smart-link	128	2

Reason	Rate(pps)	Class
ucast-ttl-fail	64	0
udld	128	3
vlan-security-discard	128	0
vrrp	512	1
bfd-learning	128	1
dot1x-mac-bypass	64	2
bgp	256	1
egress-ttl-fail	64	0
icmpv6	64	2
l2protocol-tunnel	1000	0
loopback-detection	64	3
mirror-to-cpu	1000	0
ndp	64	2
tunnel-gre-keepalive	64	0

Principle Description

Terminology

- PDU: Protocol Data Unit

12.2.2 Configuration

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Set the total rate

The default value of total rate is 2000, the unit is pps (packet-per-second)

```
Switch(config)# cpu-traffic-limit total rate 3000
```

step 3 Set the saparate rate

Use RIP packets for example:

```
Switch(config)# cpu-traffic-limit reason rip rate 500
```

step 4 Set the reason class

```
Switch(config)# cpu-traffic-limit reason rip class 3
```

NOTE: The valid range of reason class is 0-3. The larger number indicates the higher priority.

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

To display the CPU Traffic Limit configuration, use following privileged EXEC commands.

```
Switch# show cpu traffic-limit
reason                rate (pps)  class
dot1x-mac-bypass      64          2
bpdu                  64          3
slow-protocol         256         1
eapol                 128         0
erps                  128         3
smart-link            128         2
udld                  128         3
loopback-detection    64          3
arp                   256         1
dhcp                  128         0
rip                   500         3
ldp                   512         1
ospf                  256         1
pim                   128         1
bgp                   256         1
vrrp                  512         1
ndp                   64          2
icmpv6                64          2
ssh                   64          3
telnet                64          3
mlag                  1000        1
tcp                   64          2
ipda                  1000        0
icmp-redirect         128         0
mcast-rpf-fail        128         1
macsa-mismatch        128         0
port-security-discard 128         0
vlan-security-discard 128         0
egress-ttl-fail       64          0
ip-mtu-fail           64          0
bfd-learning          128         1
ptp                   512         2
ip-option             512         0
tunnel-gre-keepalive  64          0
ucast-ttl-fail        64          0
mpls-ttl-fail         64          0
igmp                  128         2
sflow-ingress         128         0
sflow-egress          128         0
fwd-to-cpu            64          0
l2protocol-tunnel     1000        0
mirror-to-cpu         1000        0
Total rate:     3000 (pps)
```

To display the CPU Traffic statistics information, use following privileged EXEC commands.

```
Switch# show cpu traffic-statistics receive all
statistics rate time is 5 second(s)
reason          count(packets)   rate(pps)
dot1x-mac-bypass    0           0
bpdu                0           0
slow-protocol       0           0
eapol               0           0
erps                0           0
smart-link          0           0
udld                0           0
loopback-detection  0           0
arp                 0           0
dhcp                0           0
rip                 0           0
ldp                 0           0
ospf                0           0
pim                 0           0
bgp                 0           0
vrrp                0           0
rsvp                0           0
ndp                 0           0
icmpv6              0           0
ssh                 0           0
```

```
telnet              0          0
mlag                0          0
tcp                 0          0
ipda                0          0
icmp-redirect       0          0
mcast-rpf-fail      0          0
macsa-mismatch      0          0
port-security-discard 0        0
vlan-security-discard 0        0
egress-ttl-fail     0          0
ip-mtu-fail         0          0
bfd-learning        0          0
ptp                 0          0
ip-option           0          0
tunnel-gre-keepalive  0        0
ucast-ttl-fail      0          0
mpls-ttl-fail       0          0
igmp                0          0
sflow-ingress       0          0
sflow-egress        0          0
fwd-to-cpu          0          0
l2protocol-tunnel   0          0
mirror-to-cpu       0          0
mpls-tp-pwoam       0          0
other               0          0
Total               0          0
```

12.2.3 Application Cases

N/A

12.3 Configuring UDLD

12.3.1 Overview

Function Introduction

The Unidirectional Link Detection protocol is a light-weight protocol that can be used to detect and disable one-way connections before they create dangerous situations such as Spanning Tree loops or other protocol malfunctions.

Principle Description

N/A

12.3.2 Configuration

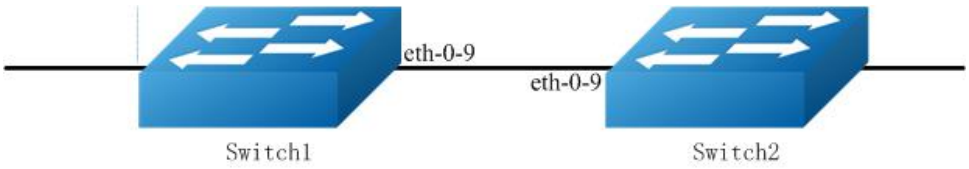


Figure 1-84 UDLD

The following configurations are same on Switch1 and Switch2.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and enable udld

```
Switch(config)# interface eth-0-9
Switch(config-if)# no shutdown
Switch(config-if)# udld port
Switch(config-if)# exit
```

step 3 Enable udld globally

```
Switch(config)# udld enable
```

step 4 Set the message interval (optional)

If the message is not specified, use the default value: 15 seconds.

```
Switch(config)# udld message interval 10
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on Switch1.

```
Switch# show udld eth-0-9
Interface eth-0-9
---
UDLD mode       : normal
Operation state  : Bidirectional
Message interval : 10
Message timeout   : 3
 Neighbor 1
 ---
 Device ID      : 4c7b.8510.ab00
 Port ID        : eth-0-9
 Device Name    : Switch
 Message interval: 10
 Message timeout : 3
 Link Status    : bidirectional
 Expiration time : 29
```

Display the result on Switch2.

```
Switch# show udld eth-0-9
Interface eth-0-9
---
UDLD mode      : normal
Operation state : Bidirectional
Message interval: 10
Message timeout : 3
 Neighbor 1
 ---
 Device ID      : 28bc.83db.8400
 Port ID        : eth-0-9
 Device Name    : Switch
 Message interval: 10
 Message timeout : 3
 Link Status    : bidirectional
 Expiration time : 23
```

12. 3. 3 Application Cases

N/A

12. 4 Configuring ERPS

12. 4. 1 Overview

Function Introduction

ERPS technology increases the availability and robustness of Ethernet rings. In the event that a fiber cut occurs, ERPS converges in less than one second, often in less than 50 milliseconds.

The main idea is described as the following. ERPS operates by declaring an ERPS domain on a single ring. On that ring domain, one switch, or node, is designated the master node, while all other nodes are designated as transit nodes. One port of the master node is designated as the master node's primary port to the ring; another port is designated as the master node's secondary port to the ring. In normal operation, the master node blocks the secondary port for all non-ERPS traffic belonging to this ERPS domain, thereby avoiding a loop in the ring. Keep-alive messages are sent by the master node in a pre-set time interval. Transit nodes in the ring domain will forward the ERPS messages. Once a link failure event occurs, the master node will detect this either by receiving the link-down message sent by the node adjacent to the failed link or by the timeout of the keep-alive message. After link failure is detected, master node will open the secondary port for data traffic to re-route the traffic.

Principle Description

Reference: RFC 3619

12.4.2 Configuration

ERPS is a soft-state protocol. The main requirement is to enable ERPS on desired devices, and configure the ERPS information correctly for various network topologies.

This section provides ERPS configuration examples for their typical network topologies.

Configuring ERPS for a Single-Ring Topology

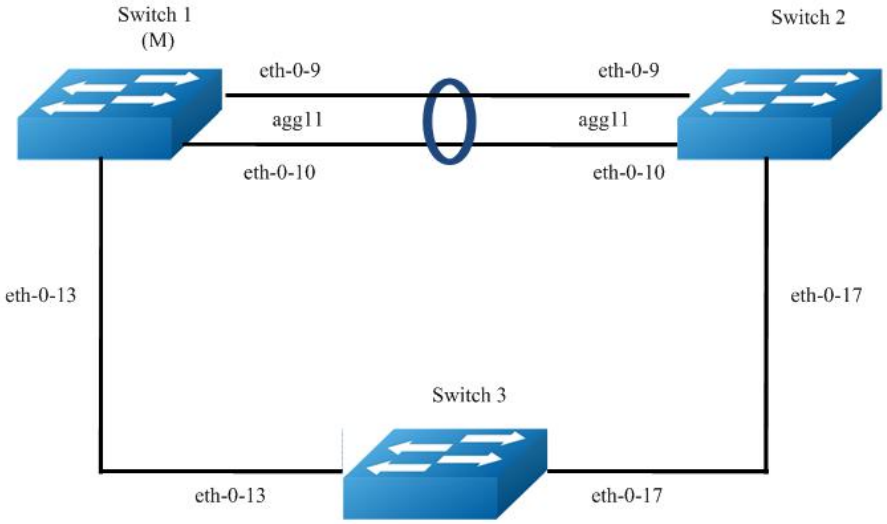


Figure 1-85 ERPS

Configure same ERPS domain and ring at Switch1, Switch2 and Switch3. Switch1 is configured as ERPS master node and other two switches are configured as ERPS transit nodes. Interface agg11, which has two members called eth-0-9 and eth-0-10, is configured as primary interface at Switch1 and eth-0-13 is configured as secondary interface.

NOTE: The ports accessing an ERPS ring must be configured as trunk ports, permitting the traffic of data VLANs to pass through. If the switch is enabled stacking, the port of ERPS ring should not on slave stacking device.

- The ports accessing an ERPS ring must be configured as the members of the control VLAN, allowing the ERPS packets to be sent and received.
- STP on ports accessing ERPS rings must be disabled.
- Only one node can be configured as master node.
- Control VLAN must not be configured as Layer 3 interface.
- VLAN mapping must not be enabled on the ERPS ports.
- Native VLAN of a port accessing an ERPS ring must not be set as the primary control VLAN or the secondary control VLAN.

The following configuration should be operated on all switches if the switch ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create the vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 15
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode and set the attributes of the interface

As the topology shows, eth-0-9 and eth-0-10 of Switch1 and Switch2 join agg 11 and connect to each other directly. eth-0-13 of Switch1 and Switch3 connect to each other directly. eth-0-17 of Switch2 and Switch3 connect to each other directly.

Interface agg 11 configuration for Switch1 and Switch2:

```
Switch(config)# interface eth-0-9
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 15
Switch(config-if)# static-channel-group 11
Switch(config-if)# exit

Switch(config)# interface eth-0-10
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 15
Switch(config-if)# static-channel-group 11
Switch(config-if)# exit

Switch(config)# interface agg11
Switch(config-if)# spanning-tree port disable
```

Interface eth-0-13 configuration for Switch1 and Switch3:

```
Switch(config)# interface eth-0-13
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 15
Switch(config-if)# spanning-tree port disable
Switch(config-if)# exit
```

Interface eth-0-17 configuration for Switch2 and Switch3:

```
Switch(config)# interface eth-0-17
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 15
Switch(config-if)# spanning-tree port disable
Switch(config-vlan)# exit
```

step 4 Create and enable ERPS domain.

ERPS domain for Switch1:

```
Switch(config)# erps 11
Switch(config)# erps 11 primary control vlan 15
Switch(config)# erps 11 mstp instance 0
Switch(config)# erps 11 ring 1 level primary
Switch(config)# erps 11 ring 1 mode master
Switch(config)# erps 11 ring 1 primary interface agg11
Switch(config)# erps 11 ring 1 secondary interface eth-0-13
Switch(config)# erps 11 ring 1 enable
Switch(config)# erps 11 enable
```

ERPS domain for Switch2:

```
Switch(config)# erps 11
Switch(config)# erps 11 primary control vlan 15
Switch(config)# erps 11 mstp instance 0
Switch(config)# erps 11 ring 1 level primary
Switch(config)# erps 11 ring 1 mode transit
Switch(config)# erps 11 ring 1 primary interface agg11
```

```
Switch(config)# erps 11 ring 1 secondary interface eth-0-17
Switch(config)# erps 11 ring 1 enable
Switch(config)# erps 11 enable
```

ERPS domain for Switch3:

```
Switch(config)# erps 11
Switch(config)# erps 11 primary control vlan 15
Switch(config)# erps 11 mstp instance 0
Switch(config)# erps 11 ring 1 level primary
Switch(config)# erps 11 ring 1 mode transit
Switch(config)# erps 11 ring 1 primary interface eth-0-17
Switch(config)# erps 11 ring 1 secondary interface eth-0-13
Switch(config)# erps 11 ring 1 enable
Switch(config)# erps 11 enable
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on Switch1.

```
Switch# show erps 11
ERPS domain ID: 11
ERPS domain name: ERPS0011
ERPS domain mode: normal
ERPS domain primary control VLAN ID: 15
ERPS domain sub control VLAN ID: 0
ERPS domain hello timer interval: 1 second(s)
ERPS domain fail timer interval: 3 second(s)
ERPS domain protected mstp instance: 0
ERPS ring ID: 1
ERPS ring level: primary
ERPS ring 1 node mode: master
ERPS ring 1 node state: complete
ERPS ring 1 primary interface name: agg11        state:unblock
ERPS ring 1 secondary interface name: eth-0-13    state:block
ERPS ring 1 stats:
Sent:
 total packets:51
 hello packets:47              ring-up-flush-fdb packets:2
 ring-down-flush-fdb packets:2     link-down packets:0
 edge-hello packets:0            major-fault packets:0
Received:
 total packets:21
 hello packets:21              ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0     link-down packets:0
 edge-hello packets:0            major-fault packets:0
```

Display the result on Switch2.

```
Switch# show erps 11
ERPS domain ID: 11
ERPS domain name: ERPS0011
ERPS domain mode: normal
ERPS domain primary control VLAN ID: 15
ERPS domain sub control VLAN ID: 0
ERPS domain hello timer interval: 1 second(s)
ERPS domain fail timer interval: 3 second(s)
ERPS domain protected mstp instance: 0
ERPS ring ID: 1
ERPS ring level: primary
ERPS ring 1 node mode: transit
ERPS ring 1 node state: link up
ERPS ring 1 primary interface name: agg11        state:unblock
ERPS ring 1 secondary interface name: eth-0-17    state:unblock
ERPS ring 1 stats:
Sent:
```

```
 total packets:0
 hello packets:0              ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0    link-down packets:0
 edge-hello packets:0          major-fault packets:0
Received:
 total packets:114
 hello packets:113            ring-up-flush-fdb packets:1
 ring-down-flush-fdb packets:0    link-down packets:0
 edge-hello packets:0          major-fault packets:0
```

Display the result on Switch3.

```
Switch# show erps 11
ERPS domain ID: 11
ERPS domain name: ERPS0011
ERPS domain mode: normal
ERPS domain primary control VLAN ID: 15
ERPS domain sub control VLAN ID: 0
ERPS domain hello timer interval: 1 second(s)
ERPS domain fail timer interval: 3 second(s)
ERPS domain protected mstp instance: 0
ERPS ring ID: 1
ERPS ring level: primary
ERPS ring 1 node mode: transit
ERPS ring 1 node state: link up
ERPS ring 1 primary interface name: eth-0-17    state:unblock
ERPS ring 1 secondary interface name: eth-0-13    state:unblock
ERPS ring 1 stats:
Sent:
 total packets:0
 hello packets:0              ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0    link-down packets:0
 edge-hello packets:0          major-fault packets:0
Received:
 total packets:130
 hello packets:129            ring-up-flush-fdb packets:1
 ring-down-flush-fdb packets:0    link-down packets:0
 edge-hello packets:0          major-fault packets:0
```

Configuring a Intersecting-Ring Topology

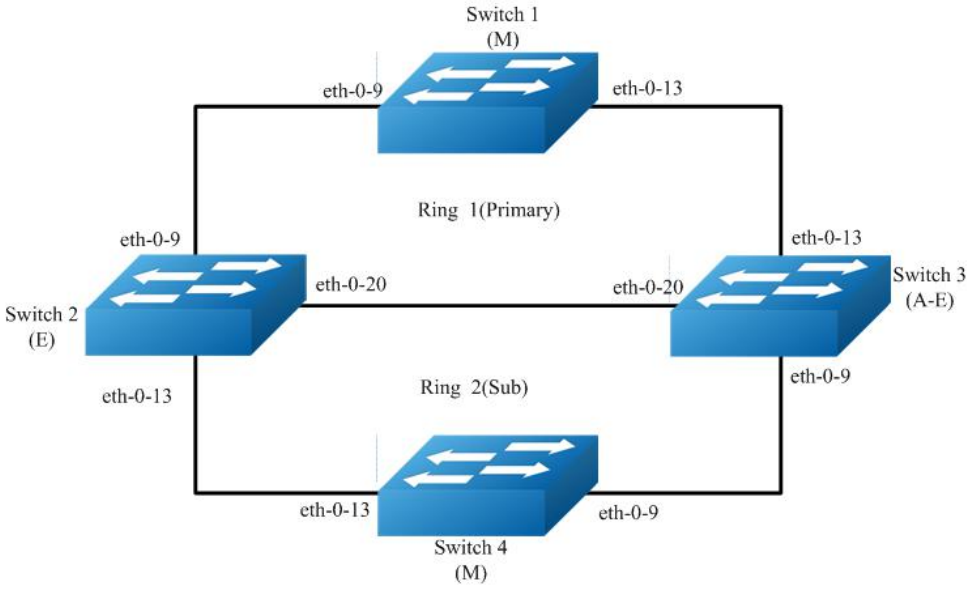


Figure 1-86 ERPS

Configure same ERPS domain at Switch1, Switch2, Switch3 and Switch4. Switch1, Switch2 and Switch3 consist of ERPS primary ring 1 while Switch2, Switch3 and Switch4 consist of ERPS sub ring 2. Switch1 is configured as ERPS ring 1 master node and other two switches are

configured as ERPS transit nodes while Switch4 is configured as ERPS ring 2 master node. In addition Switch2 is configured as edge node and Switch3 is configured as assistant-edge node.

The ports accessing an ERPS ring must be configured as trunk ports, permitting the traffic of data VLANs to pass through.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create the vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 11,12
Switch(config-vlan)# exit
```

step 3 Enter the interface configure mode and set the attributes of the interface

```
Switch(config)# interface eth-0-9
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 11,12
Switch(config-if)# spanning-tree port disable
Switch(config-if)# exit

Switch(config)# interface eth-0-13
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 11,12
Switch(config-if)# spanning-tree port disable
Switch(config-if)# exit
```

Interface eth-0-20 configuration for Switch2 and Switch3:

```
Switch(config)# interface eth-0-20
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan add 11,12
Switch(config-if)# spanning-tree port disable
Switch(config-if)# exit
```

step 4 Create and enable ERPS domain.

ERPS domain for Switch1:

```
Switch(config)# erps 1
Switch(config)# erps 1 primary control vlan 11
Switch(config)# erps 1 sub control vlan 12
Switch(config)# erps 1 mstp instance 0
Switch(config)# erps 1 ring 1 level primary
Switch(config)# erps 1 ring 1 mode master
Switch(config)# erps 1 ring 1 primary interface eth-0-9
Switch(config)# erps 1 ring 1 secondary interface eth-0-13
Switch(config)# erps 1 ring 1 enable
Switch(config)# erps 1 enable
```

ERPS domain for Switch2:

```
Switch(config)# erps 1
Switch(config)# erps 1 primary control vlan 11
Switch(config)# erps 1 sub control vlan 12
Switch(config)# erps 1 mstp instance 0
Switch(config)# erps 1 ring 1 level primary
Switch(config)# erps 1 ring 1 mode transit
Switch(config)# erps 1 ring 1 primary interface eth-0-9
Switch(config)# erps 1 ring 1 secondary interface eth-0-20
Switch(config)# erps 1 ring 1 enable

Switch(config)# erps 1 ring 2 level sub
Switch(config)# erps 1 ring 2 edge-mode edge
```

```
Switch(config)# erps 1 ring 2 edge interface eth-0-13
Switch(config)# erps 1 ring 2 common interface eth-0-20
Switch(config)# erps 1 ring 2 srpt disable
Switch(config)# erps 1 ring 2 enable
Switch(config)# erps 1 enable
```

ERPS domain for Switch3:

```
Switch(config)# erps 1
Switch(config)# erps 1 primary control vlan 11
Switch(config)# erps 1 sub control vlan 12
Switch(config)# erps 1 mstp instance 0
Switch(config)# erps 1 ring 1 level primary
Switch(config)# erps 1 ring 1 mode transit
Switch(config)# erps 1 ring 1 primary interface eth-0-13
Switch(config)# erps 1 ring 1 secondary interface eth-0-20
Switch(config)# erps 1 ring 1 enable

Switch(config)# erps 1 ring 2 level sub
Switch(config)# erps 1 ring 2 edge-mode assistant-edge
Switch(config)# erps 1 ring 2 edge interface eth-0-9
Switch(config)# erps 1 ring 2 common interface eth-0-20
Switch(config)# erps 1 ring 2 enable
Switch(config)# erps 1 enable
```

ERPS domain for Switch4:

```
Switch(config)# erps 1
Switch(config)# erps 1 sub control vlan 12
Switch(config)# erps 1 mstp instance 0
Switch(config)# erps 1 ring 2 level sub
Switch(config)# erps 1 ring 2 mode master
Switch(config)# erps 1 ring 2 primary interface eth-0-9
Switch(config)# erps 1 ring 2 secondary interface eth-0-13
Switch(config)# erps 1 ring 2 enable
Switch(config)# erps 1 enable
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on Switch1.

```
Switch# show erps 1
ERPS domain ID: 1
ERPS domain name: ERPS001
ERPS domain mode: normal
ERPS domain primary control VLAN ID: 11
ERPS domain sub control VLAN ID: 12
ERPS domain hello timer interval: 1 second(s)
ERPS domain fail timer interval: 3 second(s)
ERPS domain protected mstp instance: 0
ERPS ring ID: 1
ERPS ring level: primary
ERPS ring 1 node mode: master
ERPS ring 1 node state: complete
ERPS ring 1 primary interface name: eth-0-9       state:unblock
ERPS ring 1 secondary interface name: eth-0-13    state:block
ERPS ring 1 stats:
Sent:
 total packets:1310
 hello packets:1303              ring-up-flush-fdb packets:3
 ring-down-flush-fdb packets:4   link-down packets:0
 edge-hello packets:0            major-fault packets:0
Received:
 total packets:921
 hello packets:921               ring-up-flush-fdb packets:0
```

```
 ring-down-flush-fdb packets:0      link-down packets:0
 edge-hello packets:0               major-fault packets:0
```

Display the result on Switch2.

```
Switch# show erps 1
ERPS domain ID: 1
ERPS domain name: ERPS001
ERPS domain mode: normal
ERPS domain primary control VLAN ID: 11
ERPS domain sub control VLAN ID: 12
ERPS domain hello timer interval: 1 second(s)
ERPS domain fail timer interval: 3 second(s)
ERPS domain protected mstp instance: 0
ERPS ring ID: 1
ERPS ring level: primary
ERPS ring 1 node mode: transit
ERPS ring 1 node state: link up
ERPS ring 1 primary interface name: eth-0-9       state:unblock
ERPS ring 1 secondary interface name: eth-0-20    state:unblock
ERPS ring 1 stats:
Sent:
 total packets:0
 hello packets:0                ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0      link-down packets:0
 edge-hello packets:0               major-fault packets:0
Received:
 total packets:988
 hello packets:985              ring-up-flush-fdb packets:2
 ring-down-flush-fdb packets:1      link-down packets:0
 edge-hello packets:0               major-fault packets:0
ERPS ring ID: 2
ERPS ring level: sub
ERPS ring 2 node mode: transit
ERPS ring 2 edge node mode: edge
ERPS ring 2 node state: link up
ERPS ring 2 edge interface name: eth-0-13      state: unblock
ERPS ring 2 common interface name: eth-0-20    state: unblock
EPRS ring 2 SRPT is disabled
ERPS ring 2 stats:
Sent:
 total packets:0
 hello packets:0                ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0      link-down packets:0
 edge-hello packets:0               major-fault packets:0
Received:
 total packets:858
 hello packets:856              ring-up-flush-fdb packets:1
 ring-down-flush-fdb packets:1      link-down packets:0
 edge-hello packets:0               major-fault packets:0
```

Display the result on Switch3.

```
Switch# show erps 1
ERPS domain ID: 1
ERPS domain name: ERPS001
ERPS domain mode: normal
ERPS domain primary control VLAN ID: 11
ERPS domain sub control VLAN ID: 12
ERPS domain hello timer interval: 1 second(s)
ERPS domain fail timer interval: 3 second(s)
ERPS domain protected mstp instance: 0
ERPS ring ID: 1
ERPS ring level: primary
ERPS ring 1 node mode: transit
ERPS ring 1 node state: link up
ERPS ring 1 primary interface name: eth-0-13      state:unblock
ERPS ring 1 secondary interface name: eth-0-20    state:unblock
ERPS ring 1 stats:
Sent:
 total packets:0
```

```
 hello packets:0                ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0  link-down packets:0
 edge-hello packets:0           major-fault packets:0
Received:
 total packets:645
 hello packets:644              ring-up-flush-fdb packets:1
 ring-down-flush-fdb packets:0  link-down packets:0
 edge-hello packets:0           major-fault packets:0
ERPS ring ID: 2
ERPS ring level: sub
ERPS ring 2 node mode: transit
ERPS ring 2 edge node mode: assistant edge
ERPS ring 2 node state: link up
ERPS ring 2 edge interface name: eth-0-9      state: unblock
ERPS ring 2 common interface name: eth-0-20   state: unblock
ERPS ring 2 stats:
Sent:
 total packets:0
 hello packets:0                ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0  link-down packets:0
 edge-hello packets:0           major-fault packets:0
Received:
 total packets:645
 hello packets:644              ring-up-flush-fdb packets:1
 ring-down-flush-fdb packets:0  link-down packets:0
 edge-hello packets:0           major-fault packets:0
```

Display the result on Switch4.

```
Switch# show erps 1
ERPS domain ID: 1
ERPS domain name: ERPS001
ERPS domain mode: normal
ERPS domain primary control VLAN ID: 0
ERPS domain sub control VLAN ID: 12
ERPS domain hello timer interval: 1 second(s)
ERPS domain fail timer interval: 3 second(s)
ERPS domain protected mstp instance: 0
ERPS ring ID: 2
ERPS ring level: sub
ERPS ring 2 node mode: master
ERPS ring 2 node state: complete
ERPS ring 2 primary interface name: eth-0-9      state:unblock
ERPS ring 2 secondary interface name: eth-0-13   state:block
ERPS ring 2 stats:
Sent:
 total packets:814
 hello packets:810              ring-up-flush-fdb packets:2
 ring-down-flush-fdb packets:2  link-down packets:0
 edge-hello packets:0           major-fault packets:0
Received:
 total packets:774
 hello packets:774              ring-up-flush-fdb packets:0
 ring-down-flush-fdb packets:0  link-down packets:0
 edge-hello packets:0           major-fault packets:0
Switch#
```

12.4.3 Application Cases

N/A

12.5 Configuring Smart Link

12.5.1 Overview

Function Introduction

The Smart Link is a simple but practical technology of fast link protection. It is a solution specific to dual uplink networking to fulfill redundancy and fast migration of active and standby links.

Every smart-link group is included a pair of a layer 2 interfaces where one interface is configured to act as a standby to the other. The feature provides an alternative solution to the STP. Users can disable STP and still retain basic link redundancy. The feature also support load-balancing so than both interfaces simultaneously forward the traffic.

Principle Description

N/A

12.5.2 Configuration

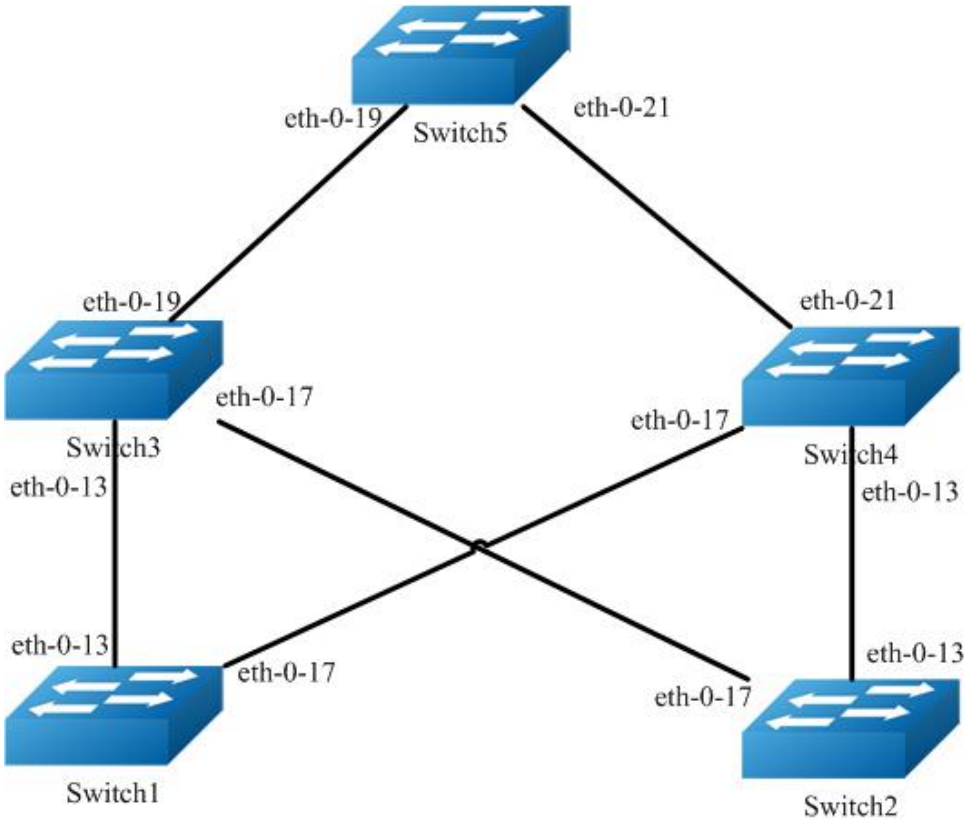


Figure 1-87 Smart-Link Typical Topology

The figure above is a typical smart-link application. The Switch1 and Switch2 are configured smart-link group. Switch3, Switch4 and Switch5 are configured smart-link flush receiver.

To configure smart-link group, some configuration should be configured before it.

- VLANs should be configured.
- MSTP instance should be configured.
- Spanning-tree should be disabled in the interface.
- About above configurations, please see the related references.

The following configuration should be operated on all switches if the switch ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create the vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 2-20
Switch(config-vlan)# exit
```

step 3 Set the spanning tree mode and create mstp instance

Create the mstp instance on Switch1 and Switch2:

```
Switch(config)# spanning-tree mode mstp
Switch(config)# spanning-tree mst configuration
Switch(config-mst)# instance 1 vlan 1
Switch(config-mst)# instance 2 vlan 2
Switch(config-mst)# instance 3 vlan 3
Switch(config-mst)# exit
```

step 4 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1 and Switch2:

```
Switch(config)# interface eth-0-13
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# spanning-tree port disable
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-17
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# spanning-tree port disable
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Interface configuration for Switch3 and Switch4:

```
Switch(config)# interface eth-0-13
Switch(config-if)# switchport mode trunk
Switch(config-if)# no shutdown
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# smart-link flush receive control-vlan 10 password simple test
Switch(config-if)# exit

Switch(config)# interface eth-0-17
Switch(config-if)# no shutdown
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# smart-link flush receive control-vlan 10 password simple test
Switch (config-if)# exit
```

Interface eth-0-19 configuration for Switch3:

```
Switch(config)# interface eth-0-19
Switch(config-if)# switchport mode trunk
Switch(config-if)# no shutdown
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# exit
```

Interface eth-0-21 configuration for Switch4:

```
Switch(config)# interface eth-0-21
Switch(config-if)# switchport mode trunk
Switch(config-if)# no shutdown
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# exit
```

Interface configuration for Switch5:

```
Switch(config)# interface eth-0-19
Switch(config-if)# switchport mode trunk
Switch(config-if)# no shutdown
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# smart-link flush receive control-vlan 10 password simple test
Switch(config-if)# exit

Switch(config)# interface eth-0-21
Switch(config-if)# switchport mode trunk
Switch(config-if)# no shutdown
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# smart-ink flush receive control-vlan 10 password simple test
Switch(config-if)# exit
```

step 5 Create smart link group and set the attributes of the group

Create smart link group on Switch1 and Switch2:

```
Switch(config)# smart-link group 1
Switch(config-smlk-group)# interface eth-0-13 master
Switch(config-smlk-group)# interface eth-0-17 slave
Switch(config-smlk-group)# protected mstp instance 1
Switch(config-smlk-group)# protected mstp instance 2
Switch(config-smlk-group)# protected mstp instance 3
Switch(config-smlk-group)# load-balance instance 3
Switch(config-smlk-group)# restore time 40
Switch(config-smlk-group)# restore enable
Switch(config-smlk-group)# flush send control-vlan 10 password simple test
Switch(config-smlk-group)# group enable
Switch(config-smlk-group)# exit
```

step 6 Disable the smart link relay function

Configure on Switch5:

```
Switch(config)# no smart-link relay enable
```

step 7 Exit the configure mode

```
Switch(config)# end
```

step 8 Validation

Display the result on Switch1.

```
Switch1# show smart-link group 1
Smart-link group 1 information:
The smart-link group was enabled.
============================================================
Auto-restore:
 state     time         count   Last-time
 enabled    40           0    N/A
============================================================
Protected instance: 1  2  3
Load balance instance: 3
Flush sender , Control-vlan ID: 10   Password:test
============================================================
INTERFACE:
Role   Member   DownCount Last-Down-Time  FlushCount Last-Flush-Time
MASTER eth-0-13   0     N/A              0    N/A
SLAVE  eth-0-17   0     N/A              0    N/A
============================================================
Instance states in the member interfaces:
 A - ACTIVE ,  B -BLOCK , D-The interface is link-down
Map-instance-ID   MASTER(eth-0-13)   SLAVE(eth-0-17)
    1           A              B
    2           A              B
    3           B              A
```

Display the result on Switch2.

```
Switch# show smart-link group 1
Smart-link group 1 information:
 The smart-link group was enabled.
 ============================================================
 Auto-restore:
  state      time          count    Last-time
  enabled     40            0     N/A
 ============================================================
 Protected instance: 1  2  3
 Load balance instance: 3
 Flush sender , Control-vlan ID: 10    Password:test
 ============================================================
 INTERFACE:
 Role   Member    DownCount Last-Down-Time  FlushCount Last-Flush-Time
 MASTER eth-0-13   0      N/A               0    N/A
 SLAVE  eth-0-17   0      N/A               0    N/A
 ============================================================
 Instance states in the member interfaces:
  A - ACTIVE ,   B -BLOCK  , D-The interface is link-down
 Map-instance-ID   MASTER(eth-0-13)    SLAVE(eth-0-17)
     1             A                B
     2             A                B
     3             B                A
```

Display the result on Switch3.

```
Switch# show smart-link
 Relay smart-link flush packet is enabled
 Smart-link flush receiver interface:
   eth-0-13   control-vlan:10   password:test
   eth-0-17   control-vlan:10   password:test
Smart-link received flush packet number:0
Smart-link processed flush packet number:0
Smart link Group Number is 0.
```

Display the result on Switch4.

```
Switch# show smart-link
 Relay smart-link flush packet is enabled
 Smart-link flush receiver interface:
   eth-0-13   control-vlan:10   password:test
   eth-0-17   control-vlan:10   password:test
Smart-link received flush packet number:0
Smart-link processed flush packet number:0
Smart link Group Number is 0.
```

Display the result on Switch5.

```
Switch# show smart-link
 Relay smart-link flush packet is disabled
 Smart-link flush receiver interface:
   eth-0-21   control-vlan:10   password: test
   eth-0-19   control-vlan:10   password:test
Smart-link received flush packet number:0
Smart-link processed flush packet number:0
 Smart link Group Number is 0.
```

12. 5. 3 **Application Cases**

N/A

12. 6 Configuring Multi-Link

12. 6. 1 Overview

Function Introduction

The Multi-Link is a simple but practical technology of fast link protection. It is a solution specific to multi-uplink networking to fulfill redundancy and fast migration of between links.

The feature is like smart link, but links extend to four instead of two.

Principle Description

N/A

12. 6. 2 Configuration

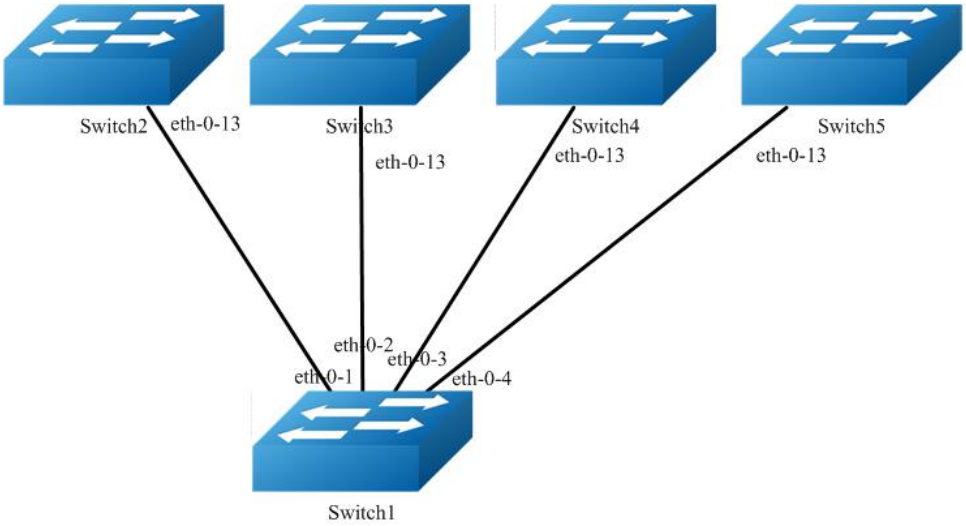


Figure 1-88 Multi-Link Typical Topology

The figure above is a typical multi-link application. The Switch1 are configured multi-link group. Switch2, Switch3, Switch4 and Switch5 are configured multi-link flush receiver.

To configure Multi-link group, some configuration should be configured before it.

- VLANs should be configured.
- MSTP instance should be configured.
- Spanning-tree should be disabled in the interface.
- About above configurations, please see the related references.
- The following configuration should be operated on all switches if the switch ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create the vlan

```
Switch(config)# vlan database
Switch(config- vlan)# vlan 2-10
Switch(config- vlan)# exit
```

step 3 Set the spanning tree mode and create mstp instance

```
Switch(config)# spanning-tree mode mstp
Switch(config)# spanning-tree mst configuration
Switch(config-mst)# instance 1 vlan 1
Switch(config-mst)# instance 2 vlan 2
Switch(config-mst)# instance 3 vlan 3
Switch(config-mst)# instance 4 vlan 4-10
Switch(config-mst)# exit
```

step 4 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface range eth-0-1 - 4
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# spanning-tree port disable
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Interface configuration for Switch1 ~ 5:

```
Switch(config)# interface eth-0-13
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# multi-link flush receive control-vlan 10 password simple test
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 5 Create multi link group and set the attributes of the group

Create multi link group on Switch1:

```
Switch(config)# multi-link group 1
Switch(config-multilk-group)# interface eth-0-1 priority 1
Switch(config-multilk-group)# interface eth-0-2 priority 2
Switch(config-multilk-group)# interface eth-0-3 priority 3
Switch(config-multilk-group)# interface eth-0-4 priority 4
Switch(config-multilk-group)# protected mstp instance 1
Switch(config-multilk-group)# protected mstp instance 2
Switch(config-multilk-group)# protected mstp instance 3
Switch(config-multilk-group)# protected mstp instance 4
Switch(config-multilk-group)# load-balance instance 2 priority 2
Switch(config-multilk-group)# load-balance instance 3 priority 3
Switch(config-multilk-group)# load-balance instance 4 priority 4
Switch(config-multilk-group)# restore time 40
Switch(config-multilk-group)# restore enable
Switch(config-multilk-group)# flush send control-vlan 10 password simple test
Switch(config-multilk-group)# group enable
Switch(config-multilk-group)# exit
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

Display the result on Switch1.

```
Switch# show multi-link group 1
Multi-link group 1 information:
 The multi-link group was enabled.
 ============================================================
 Auto-restore:
  state     time         count   Last-time
  enabled    40            0     N/A
 ============================================================
 Protected instance: 1  2  3  4
```

```
Load balance instance: 2(to P2)  3(to P3)  4(to P4)
Flush sender , Control-vlan ID: 10   Password:test
============================================================
INTERFACE:
Role  Member   DownCount Last-Down-Time     FlushCount Last-Flush-Time
PRI1  eth-0-1  0       N/A             1       2016/09/05,07:13:24
PRI2  eth-0-2  0       N/A             1       2016/09/05,07:13:24
PRI3  eth-0-3  0       N/A             1       2016/09/05,07:13:24
PRI4  eth-0-4  0       N/A             1       2016/09/05,07:13:24
============================================================
Instance states in the member interfaces:
 A - ACTIVE ,  B -BLOCK , D-The interface is link-down
Map-instance-ID  P1(eth-0-1 )  P2(eth-0-2 )  P3(eth-0-3 )  P4(eth-0-4 )
1           A         B         B         B
2           B         A         B         B
3           B         B         A         B
4           B         B         B         A
```

Display the result on Switch2~5.

```
Switch# show multi-link
Relay multi-link flush packet is enabled
 Multi-link flush receiver interface:
   eth-0-13   control-vlan:10   password:test
 Multi-link received flush packet number:0
 Multi-link processed flush packet number:0
 Multi-link tcn is disabled
 Multi-link tcn query count    :2
 Multi-link tcn query interval :10
 Multi-link Group Number is 0.
```

12.6.3 Application Cases

Configuring Multi-Link Enhance

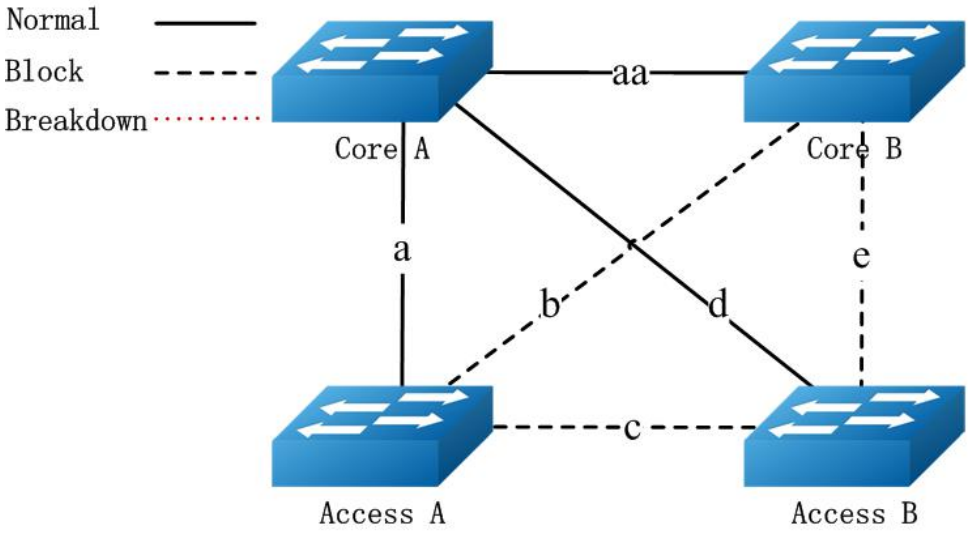
There is an enhanced method to improve the ability of multi-link to protect link. When all the interfaces of multi-link group are down, you can enable another interface to send the enhance packet to peer which makes the instance state of one interface to change from block to active. It would avoid the switch being the state of islet.

When 2 multi-link group on different switches backup for each other, multi-link members on one switch is blocked and can not protect the traffic.

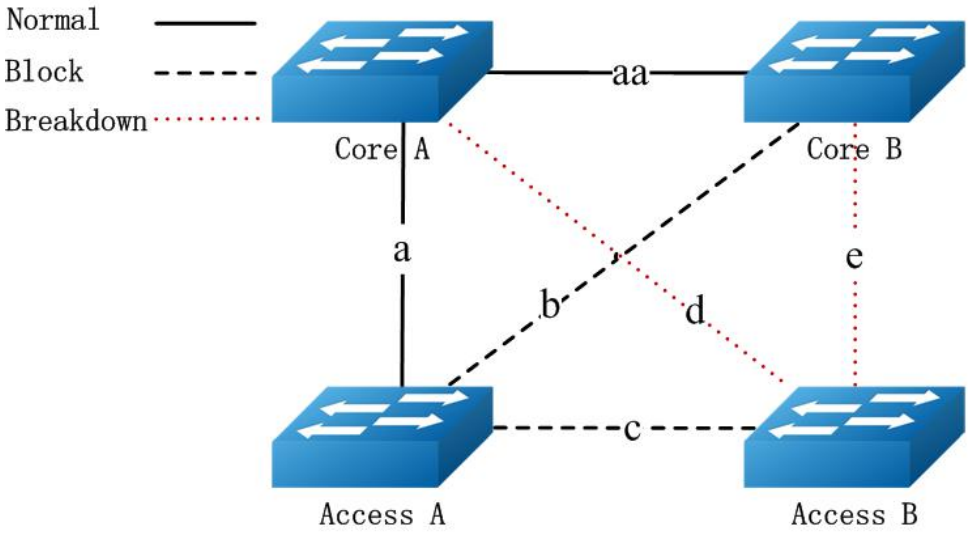
In this example:

- Core switch A and B, Access switch A and B, make up a full-match topology.
- Enable multi-link on Access switch A, the priority for link a/b/c is 1/2/3.
- Enable multi-link on Access switch B, the priority for link d/e is 1/2.

In normal condition, link b/c/e are block, link a/d are active. As the following figure shows:



When link d/e are break down, the only out going link for Access switch B is link c, which is between Access switch A and Access switch B.



Because link c is blocked, the Access switch B is the state of islet. As the following figure shows:

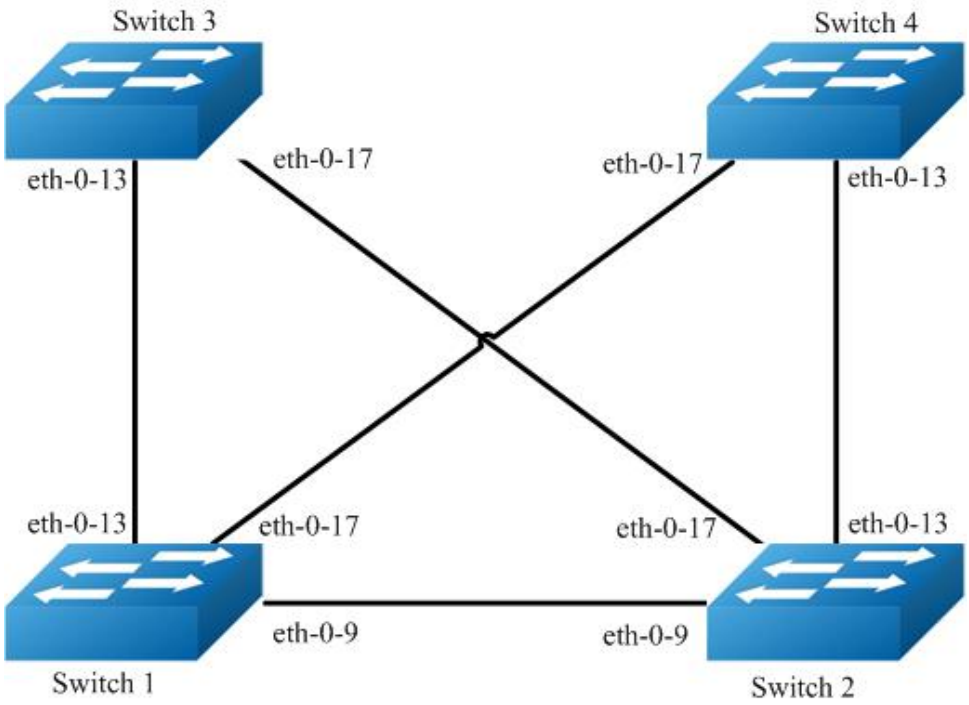


Figure 1-89 Multilink-enhance Typical Topology

The figure above is a typical multi-link application. The Switch1, 2 are configured multi-link group. Switch1 has the interface which receives the multilink-enhance packets. And , Switch2 has the interface which sends the multilink-enhance packets.

To configure multi-link group, some configuration should be configured before it.

- VLANs should be configured.
- MSTP instance should be configured.
- Spanning-tree should be disabled in the interface.
- About above configurations, please see the related references.
- It should configure the control vlan and password of flush sending before setting the multilink-enhance interface.

The following configuration should be operated on all switches if the switch ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the vlan configure mode and create the vlan

```
Switch(config)# vlan database
Switch(config- vlan)# vlan 10
Switch(config- vlan)# vlan 20
Switch(config- vlan)# vlan 30
Switch(config- vlan)# vlan 40
Switch(config- vlan)# exit
```

step 3 Set the spanning tree mode and create mstp instance

```
Switch(config)# spanning-tree mode mstp
Switch(config)# spanning-tree mst configuration
Switch(config-mst)# instance 1 vlan 10
Switch(config-mst)# instance 1 vlan 30
Switch(config-mst)# instance 2 vlan 20
Switch(config-mst)# instance 2 vlan 40
Switch(config-mst)# exit
```

step 4 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch1(config)# interface range eth-0-9
Switch1(config-if)# switchport mode trunk
Switch1(config-if)# switchport trunk allowed vlan all
Switch1(config-if)# spanning-tree port disable
Switch1(config-if)# no shutdown
Switch1(config-if)# exit

Switch1(config)# interface range eth-0-13
Switch1(config-if)# switchport mode trunk
Switch1(config-if)# switchport trunk allowed vlan all
Switch1(config-if)# spanning-tree port disable
Switch1(config-if)# no shutdown
Switch1(config-if)# exit

Switch1(config)# interface range eth-0-17
Switch1(config-if)# switchport mode trunk
Switch1(config-if)# switchport trunk allowed vlan all
Switch1(config-if)# spanning-tree port disable
Switch1(config-if)# no shutdown
Switch1(config-if)# exit
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-13
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-17
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan all
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-9
Switch(config-if)# multi-link flush receive control-vlan 30 password simple a
Switch(config-if)#exit
```

Interface configuration for Switch3:

```
Switch(config)# interface eth-0-13
Switch(config-if)# multi-link flush receive control-vlan 30 password simple a
Switch(config-if)#exit

Switch(config)# interface eth-0-17
Switch(config-if)# multi-link flush receive control-vlan 30 password simple b
Switch(config-if)#exit
```

Interface configuration for Switch4:

```
Switch(config)# interface eth-0-13
Switch(config-if)# multi-link flush receive control-vlan 30 password simple b
Switch(config-if)#exit

Switch(config)# interface eth-0-17
Switch(config-if)# multi-link flush receive control-vlan 30 password simple a
Switch(config-if)#exit
```

step 5 Create multi link group and set the attributes of the group

Create multi link group on Switch1:

```
Switch(config)# multi-link group 1
Switch(config-multilk-group)# interface eth-0-13 priority 1
Switch(config-multilk-group)# interface eth-0-17 priority 2
```

```
Switch(config-multilk-group)# interface eth-0-9 priority 3
Switch(config-multilk-group)# protected mstp instance 1
Switch(config-multilk-group)# protected mstp instance 2
Switch(config-multilk-group)# flush send control-vlan 30 password simple a
Switch(config-multilk-group)# multilink-enhance receive control-vlan 10 password b interface eth-0-9
Switch(config-multilk-group)# group enable
Switch(config-multilk-group)# end
```

Create multi link group on Switch2:

```
Switch(config)# multi-link group 1
Switch(config-multilk-group)# interface eth-0-13 priority 1
Switch(config-multilk-group)# interface eth-0-17 priority 2
Switch(config-multilk-group)# protected mstp instance 1
Switch(config-multilk-group)# protected mstp instance 2
Switch(config-multilk-group)# flush send control-vlan 10 password simple b
Switch(config-multilk-group)# multilink-enhance interface eth-0-9
Switch(config-multilk-group)# group enable
Switch(config-multilk-group)# exit
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

Display the result on Switch1.

```
Switch1# show multi-link group 1
Multi-link group 1 information:
 The multi-link group was enabled.
 ============================================================
 Auto-restore:
  state      time          count    Last-time
  disabled   60            0        N/A
 ============================================================
 Protected instance: 1  2
 Load balance instance:
 Flush sender , Control-vlan ID: 30    Password: a
 ============================================================
 INTERFACE:
 Role   Member    DownCount Last-Down-Time      FlushCount Last-Flush-Time
 PRI1   eth-0-13  0         N/A                5          2017/05/15,07:50:11
 PRI2   eth-0-17  0         N/A                0          N/A
 PRI3   eth-0-9   1         2017/05/15,07:48:46 5         2017/05/15,07:50:11
 PRI4   N/A       0         N/A                0          N/A
 ============================================================
 Instance states in the member interfaces:
  A-ACTIVE ,   B-BLOCK , A(E)-ENHANCE_ACTIVE   D-The interface is link-down
 Map-instance-ID  P1(eth-0-13)   P2(eth-0-17)   P3(eth-0-9)    P4(N/A)
 1                A              B              B              D
 2                A              B              B              D

Switch# show multi-link
Relay multi-link flush packet is enabled
 Multi-link enhance receiver interface:
   eth-0-9    control-vlan:10   password:b
 Multi-link received flush packet number : 0
 Multi-link processed flush packet number: 0
 Multi-link received enhance packet number : 4
 Multi-link processed enhance packet number: 4
 Multi-link tcn is disabled
 Multi-link tcn query count    : 2
 Multi-link tcn query interval : 10
 Multi-link Group Number is 1.
  Group-ID   State      Pri-1     Pri-2     Pri-3     Pri-4
  1          enabled    eth-0-13  eth-0-17  eth-0-9   N/A
```

Display the result on Switch2.

```
Switch# show multi-link group1
Multi-link group 1 information:
 The multi-link group was enabled.
 ============================================================
 Auto-restore:
  state      time         count    Last-time
  disabled   60           0        N/A
 ============================================================
 Protected instance: 1  2
 Load balance instance:
 Flush sender , Control-vlan ID: 10   Password: b
 Multilk enhance interface: eth-0-9, Control-vlan ID: 10  Password: b
 ============================================================
 INTERFACE:
 Role   Member    DownCount Last-Down-Time     FlushCount Last-Flush-Time
 PRI1   eth-0-13  1         2017/05/15,07:49:15 0         N/A
 PRI2   eth-0-17  2         2017/05/15,07:50:03 3         2017/05/15,07:50:11
 PRI3   N/A       0         N/A                 0         N/A
 PRI4   N/A       0         N/A                 0         N/A
 ============================================================
ENHANCE INTERFACE:
 Role   Member    DownCount Last-Down-Time     EnhanceCount Last-SendEnhance-Ti
me
 M-En   eth-0-9   0         N/A                 0          N/A

 ============================================================
 Instance states in the member interfaces:
   A-ACTIVE ,   B-BLOCK , A(E)-ENHANCE_ACTIVE   D-The interface is link-down
 Map-instance-ID  P1(eth-0-13)   P2(eth-0-17)   P3(N/A)       P4(N/A)
 1                A              B              D             D
 2                A              B              D             D

Switch# show multi-link
Relay multi-link flush packet is enabled
 Multi-link received flush packet number : 0
 Multi-link processed flush packet number: 0
 Multi-link received enhance packet number : 0
 Multi-link processed enhance packet number: 0
 Multi-link tcn is disabled
 Multi-link tcn query count    : 2
 Multi-link tcn query interval : 10
 Multi-link Group Number is 1.
  Group-ID   State      Pri-1     Pri-2     Pri-3     Pri-4
  1          enabled    eth-0-13  eth-0-17  N/A       N/A
```

12.7 Configuring Monitor Link

12.7.1 Overview

Function Introduction

Monitor Link is a port collaboration function. Monitor Link usually works together with Layer 2 topology protocols. The idea is to monitor the states of uplink ports and adapt the up/down state of downlink ports to the up/down state of uplink ports, triggering link switchover on the downstream switch in time.

Principle Description

N/A

12. 7. 2 Configuration

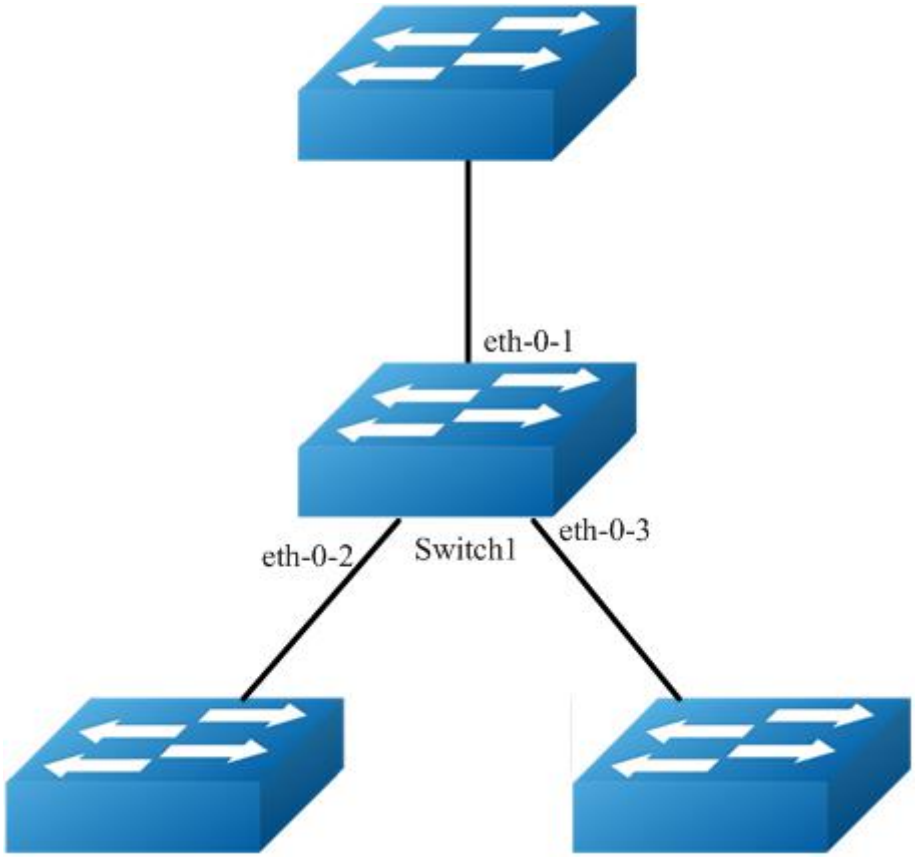


Figure 1-90 monitor link

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and turn on the interface

```
Switch(config)# interface range eth-0-1 - 3
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

step 3 Create multi link group and set the attributes of the group

```
Switch(config)# monitor-link group 1
Switch(config-mtlk-group)# monitor-link uplink interface eth-0-1
Switch(config-mtlk-group)# monitor-link downlink interface eth-0-2
Switch(config-mtlk-group)# monitor-link downlink interface eth-0-3
Switch(config-mtlk-group)# exit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

```
Switch# show monitor-link group
Group Id: 1
Monitor link status: UP
Role    Member  Last-up-time    Last-down-time       upcount  downcount
UpLk 1  eth-0-1  2011/07/15,02:07:31  2011/07/15,02:07:31    2       1
DwLk 1  eth-0-2  2011/07/15,02:07:34  2011/07/15,02:07:31    1       1
DwLk 2  eth-0-3  N/A                 N/A                    0       0
```

12.7.3 Application Cases

N/A

12.8 Configuring VRRP

12.8.1 Overview

Function Introduction

This chapter provides an overview of Virtual Router Redundancy Protocol (VRRP) and its implementation. VRRP eliminates the risk of a single point of failure inherent in a static default routing environment. It specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VRRP routers on a LAN. One of the major advantages of VRRP is that it makes default path available without requiring configuration of dynamic routing on every end-host.

NOTE: MD5 authentication is not yet supported for VRRP.

Principle Description

The VRRP module is based on: RFC 3768 (VRRP): Knight, S., et.al "Virtual Router Redundancy Protocol (VRRP)"

Terminology

- Backup Router: VRRP router that back up an IP address. It assumes forwarding responsibility for the virtual IP address if the Master fails.
- Critical IP: The IP address that the VRRP router send/receive messages on for a particular session.
- IP Address Owner: The VRRP Router that has the virtual router's IP address (es) as real interface address (es). This is the router that, when up, will respond to packets addressed to one of these IP addresses for ICMP pings, TCP connections, etc.
- Master Router: The VRRP router that owns the IP address (i.e., is being backed up), and which is the default router for forwarding for that IP address.
- Virtual IP: The IP address back up by a VRRP session.
- Virtual Router: A router managed by VRRP that acts as a default router for hosts on a shared LAN. It consists of a Virtual Router Identifier and a set of associated IP addresses across a common LAN. A VRRP Router might backup one or more virtual routers.
- VRRP Router: A router runs the Virtual Router Redundancy Protocol. It might participate in one or more virtual routers.

Typically, end hosts are connected to the enterprise network through a single router (first hop router) that is in the same Local Area Network (LAN) segment. The most popular method of configuration for the end hosts is to statically configure this router as their default gateway. This minimizes configuration and processing overhead. The main problem with this configuration method is that it produces a single point of failure if this first hop router fails.

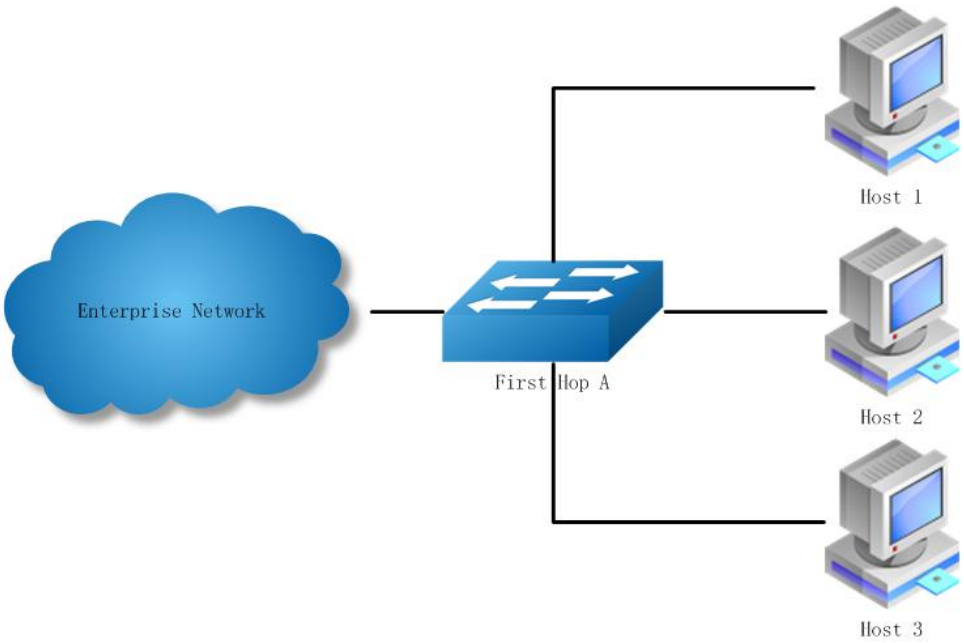


Figure 1-91 Without VRRP

The Virtual Router Redundancy Protocol attempts to solve this problem by introducing the concept of a virtual router, composed of two or more VRRP routers on the same subnet. The concept of a virtual IP address is also introduced, which is the address that end hosts configure as their default gateway. Only one router (called the master) forward packets on the behalf of this IP address. In the event that the Master router fails, one of the other routers (Backup) assumes forwarding responsibility for it.

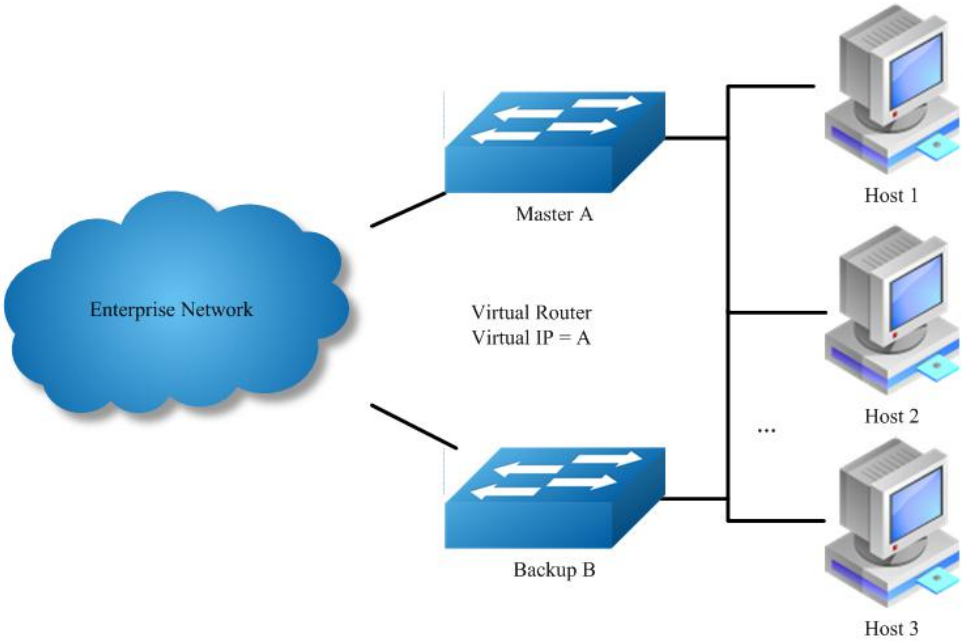


Figure 1-92 With VRRP

At first glance, the configuration outlined in might not seem very useful, as it doubles the cost and leaves one router idle at all times. This, however, can be avoided by creating two virtual routers and splitting the traffic between them.

12. 8. 2 Configuration

Configuring VRRP (One Virtual Router)

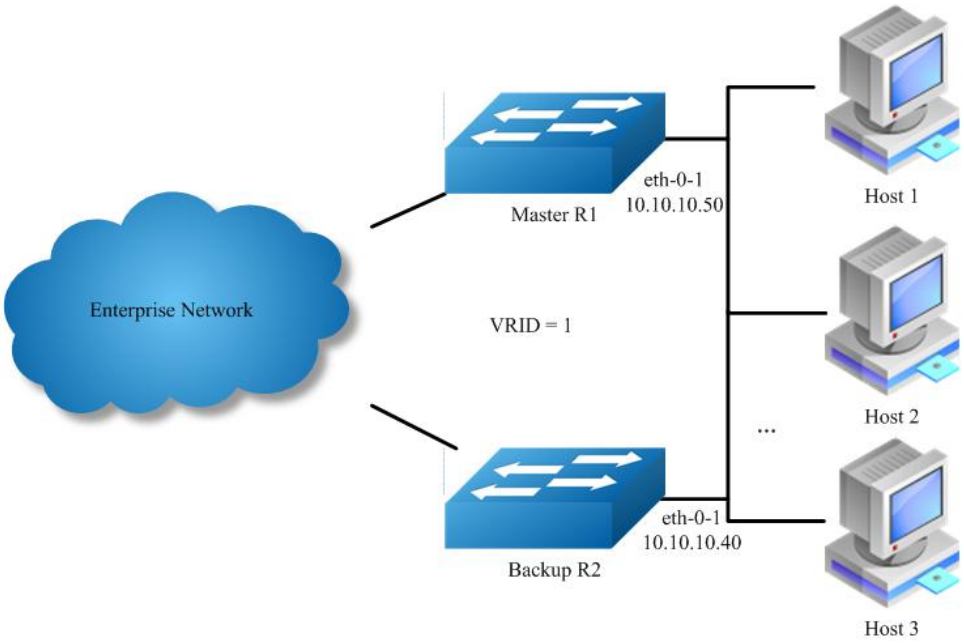


Figure 1-93 VRRP with one virtual router

In this configuration the end-hosts install a default route to the IP address of virtual router 1(VRID = 1) and both routers R1 and R2 run VRRP. R1 is configured to be the Master for virtual router 1 (VRID = 1) and R2 as a Backup for virtual router 1. If R1 fails, R2 will take over virtual router 1 and its IP addresses, and provide uninterrupted service for the hosts. Configuring only one virtual router, doubles the cost and leaves R2 idle at all times.

The following configuration should be operated on all devices if the device ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for R1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.50/24
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Interface configuration for R2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.40/24
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 3 Create an instance of vrrp

```
Switch(config)# router vrrp 1
Switch(config-router)# virtual-ip 10.10.10.60
Switch(config-router)# interface eth-0-1
Switch(config-router)# preempt-mode true
Switch(config-router)# advertisement-interval 5
```

step 4 Set the priority (optional)

Set the priority on R2. R1 use the default value if the priority is not configured.

```
Switch(config-router)# priority 200
```

step 5 Set bfd session (optional)

Configuring R1:

```
Switch (config-router)# bfd 10.10.10.40
```

Configuring R2:

```
Switch (config-router)# bfd 10.10.10.50
```

step 6 Enable vrrp and Exit the vrrp configure mode

```
Switch(config-router)# enable
Switch(config-router)# exit
```

step 7 Exit the configure mode

```
Switch(config)# end
```

step 8 Validation

Display the result on R1.

```
Switch# show vrrp
vrrp session count: 1
VRID <1>
 State           : Backup
 Virtual IP      : 10.10.10.60(Not IP owner)
 Interface       : eth-0-1
 VMAC            : 0000.5e00.0101
 VRF             : Default
 Advt timer      : 5 second(s)
 Preempt mode    : TRUE
 Conf pri        : Unset       Run pri  : 100
 Increased pri   : 0
 Master router ip  : 10.10.10.40
 Master priority   : 200
 Master advt timer  : 5 second(s)
 Master down timer  : 16 second(s)
 Preempt delay     : 0 second(s)
 Learn master mode  : FALSE
 BFD session state  : UP
 BFD local discr   : 8192
 BFD state change   : 1
```

Display the result on R2.

```
Switch# show vrrp
vrrp session count: 1
VRID <1>
 State           : Master
 Virtual IP      : 10.10.10.60(Not IP owner)
 Interface       : eth-0-1
 VMAC            : 0000.5e00.0101
 VRF             : Default
 Advt timer      : 5 second(s)
 Preempt mode    : TRUE
 Conf pri        : 200         Run pri  : 200
 Increased pri   : 0
 Master router ip  : 10.10.10.40
 Master priority   : 200
 Master advt timer  : 5 second(s)
 Master down timer  : 15 second(s)
```

```
Preempt delay     : 0 second(s)
Learn master mode  : FALSE
BFD session state  : UP
BFD local discr    : 8192
BFD state change   : 1
```

Configuring VRRP (Two Virtual Router)

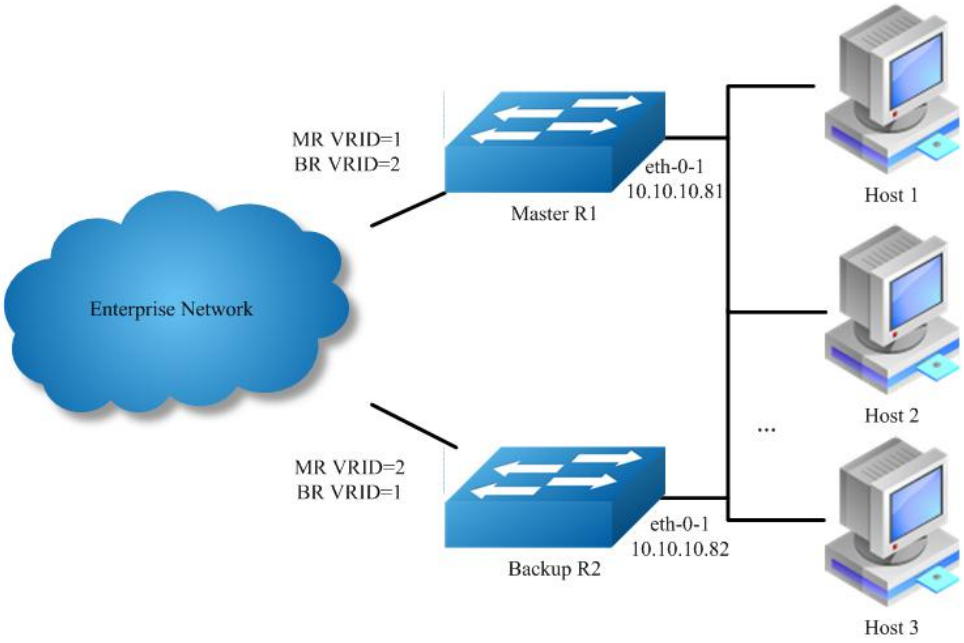


Figure 1-94 VRRP with two virtual router

In the one virtual router example earlier, R2 is not backed up by R1. This example illustrates how to backup R2 by configuring a second virtual router.

In this configuration, R1 and R2 are two virtual routers and the hosts split their traffic between R1 and R2. R1 and R2 function as backups for each other.

The following configuration should be operated on all devices if the device ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for R1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.81/24
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Interface configuration for R2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.82/24
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 3 Create an instance of vrrp

Configuring R1:

```
Switch(config)# router vrrp 1
Switch(config-router)# virtual-ip 10.10.10.81
Switch(config-router)# interface eth-0-1
Switch(config-router)# preempt-mode true
Switch(config-router)# advertisement-interval 5
Switch(config-router)# enable
Switch(config-router)# exit

Switch(config)# router vrrp 2
Switch(config-router)# virtual-ip 10.10.10.82
Switch(config-router)# interface eth-0-1
Switch(config-router)# priority 200
Switch(config-router)# preempt-mode true
Switch(config-router)# advertisement-interval 5
Switch(config-router)# enable
Switch(config-router)# exit
```

Configuring R2:

```
Switch(config)# router vrrp 1
Switch(config-router)# virtual-ip 10.10.10.81
Switch(config-router)# interface eth-0-1
Switch(config-router)# priority 200
Switch(config-router)# preempt-mode true
Switch(config-router)# advertisement-interval 5
Switch(config-router)# enable
Switch(config-router)# exit

Switch(config)# router vrrp 2
Switch(config-router)# virtual-ip 10.10.10.82
Switch(config-router)# interface eth-0-1
Switch(config-router)# preempt-mode true
Switch(config-router)# advertisement-interval 5
Switch(config-router)# enable
Switch(config-router)# exit
```

step 4 Exit the configure mode

```
Switch(config)# end
```

step 5 Validation

Display the result on R1.

```
Switch# show vrrp
vrrp session count: 2
VRID <1>
 State           : Master
 Virtual IP      : 10.10.10.81(IP owner)
 Interface       : eth-0-9
 VMAC            : 0000.5e00.0101
 VRF             : Default
 Advt timer      : 5 second(s)
 Preempt mode    : TRUE
 Conf pri        : Unset       Run pri   : 255
 Increased pri   : 0
 Master router ip   : 10.10.10.81
 Master priority    : 255
 Master advt timer  : 5 second(s)
 Master down timer  : 15 second(s)
 Preempt delay      : 0 second(s)
 Learn master mode  : FALSE
 BFD session state  : UNSET
VRID <2>
 State           : Backup
 Virtual IP      : 10.10.10.82(Not IP owner)
```

```
Interface         : eth-0-9
VMAC              : 0000.5e00.0102
VRF               : Default
Advt timer        : 5 second(s)
Preempt mode      : TRUE
Conf pri          : 200         Run pri   : 200
Increased pri     : 0
Master router ip  : 10.10.10.82
Master priority   : 255
Master advt timer : 5 second(s)
Master down timer : 15 second(s)
Preempt delay     : 0 second(s)
Learn master mode : FALSE
BFD session state : UNSET
```

Display the result on R2.

```
Switch# show vrrp
vrrp session count: 2
VRID <1>
 State            : Backup
 Virtual IP       : 10.10.10.81(Not IP owner)
 Interface        : eth-0-9
 VMAC             : 0000.5e00.0101
 VRF              : Default
 Advt timer       : 5 second(s)
 Preempt mode     : TRUE
 Conf pri         : 200         Run pri   : 200
 Increased pri    : 0
 Master router ip : 10.10.10.81
 Master priority  : 255
 Master advt timer : 5 second(s)
 Master down timer : 15 second(s)
 Preempt delay    : 0 second(s)
 Learn master mode : FALSE
 BFD session state : UNSET
VRID <2>
 State            : Master
 Virtual IP       : 10.10.10.82(IP owner)
 Interface        : eth-0-9
 VMAC             : 0000.5e00.0102
 VRF              : Default
 Advt timer       : 5 second(s)
 Preempt mode     : TRUE
 Conf pri         : Unset       Run pri   : 255
 Increased pri    : 0
 Master router ip : 10.10.10.82
 Master priority  : 255
 Master advt timer : 5 second(s)
 Master down timer : 15 second(s)
 Preempt delay    : 0 second(s)
 Learn master mode : FALSE
 BFD session state : UNSET
```

VRRP Circuit Failover

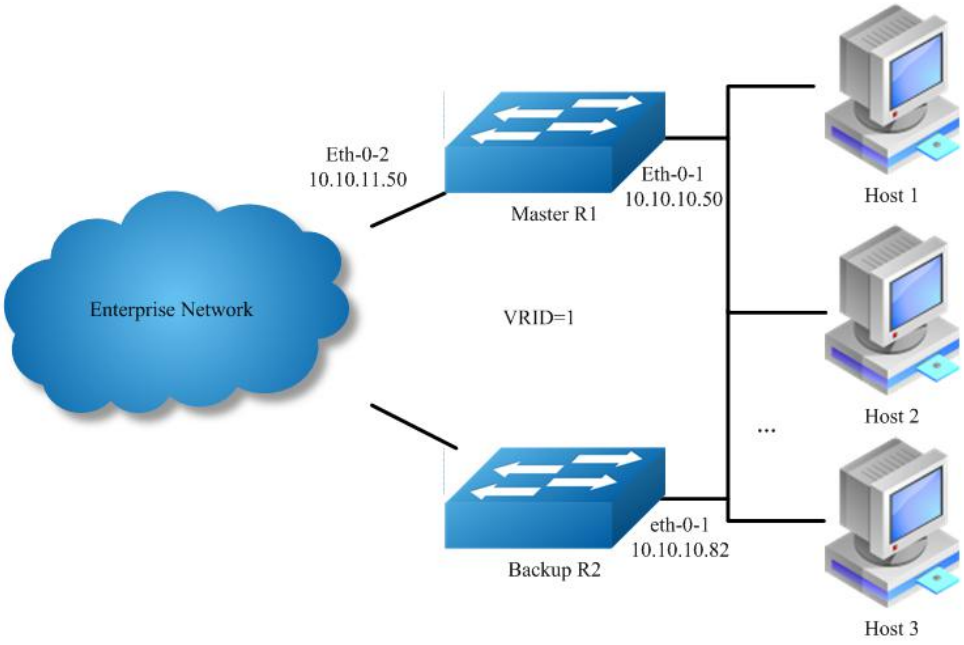


Figure 1-95 VRRP Circuit Failover

The need for VRRP Circuit Failover arose because VRRPv2 was unable to track the gateway interface status. The VRRP Circuit Failover feature provides a dynamic failover of an entire circuit in the event that one member of the group fails. It introduces the concept of a circuit, where two or more Virtual Routers on a single system can be grouped. In the event that a failure occurs and one of the Virtual Routers performs the Master to Backup transition, the other Virtual Routers in the group are notified and are forced into the Master to Backup transition, so that both incoming and outgoing packets are routed through the same gateway router, eliminating the problem for Firewall/NAT environments. The following scenario explains this feature.

To configure VRRP Circuit Failover, each circuit is configured to have a corresponding priority-delta value, which is passed to VRRP when a failure occurs. The priority of each Virtual Router on the circuit is decremented by the priority delta value causing the VR Master to VR Backup transition.

In this example, two routers R1 and R2 are configured as backup routers with different priorities. The priority-delta value is configured to be greater than the difference of both the priorities. R1 is configured to have a priority of 100 and R2 has a priority of 90. R1 with a greater priority is the Virtual Router Master. The priority-delta value is 20, greater than 10 (100 minus 90). On R1 when the external interface eth1 fails, the priority of R1 becomes 80 (100 minus 20). Since R2 has a greater priority (90) than R1, R2 becomes the VR Master and routing of packages continues without interruption.

When this VR Backup (R1) is up again, it regains its original priority (100) and becomes the VR Master again.

The following configuration should be operated on all devices if the device ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for R1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.50/24
Switch(config-if)# no shutdown
Switch(config-if)# exit

Switch(config)# interface eth-0-2
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.11.50/24
```

```
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Interface configuration for R2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.40/24
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 3 Create an track object to monitor the link state

Configuring R1:

```
Switch(config)# track 10 interface eth-0-2 linkstate
```

To get more information about track, please reference to the "Configuring Track" chapter.

step 4 Create an instance of vrrp

Configuring R1:

```
Switch(config)# router vrrp 1
Switch(config-router)# virtual-ip 10.10.10.1
Switch(config-router)# interface eth-0-1
Switch(config-router)# preempt-mode true
Switch(config-router)# advertisement-interval 5
Switch(config-router)# priority 100
Switch(config-router)# track 10 decrement 20
Switch(config-router)# enable
```

Configuring R2:

```
Switch(config)# router vrrp 1
Switch(config-router)# virtual-ip 10.10.10.1
Switch(config-router)# interface eth-0-1
Switch(config-router)# preempt-mode true
Switch(config-router)# advertisement-interval 5
Switch(config-router)# priority 90
Switch(config-router)# enable
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on R1.

```
Switch# show vrrp
vrrp session count: 1
VRID <1>
 State           : Master
 Virtual IP      : 10.10.10.1(Not IP owner)
 Interface       : eth-0-9
 VMAC            : 0000.5e00.0101
 VRF             : Default
 Advt timer      : 5 second(s)
 Preempt mode    : TRUE
 Conf pri        : 100        Run pri   : 100
 Increased pri   : 0
 Track Object    : 10
 Decre pri       : 20
 Master router ip   : 10.10.10.50
 Master priority    : 100
 Master advt timer  : 5 second(s)
 Master down timer  : 16 second(s)
 Preempt delay      : 0 second(s)
```

```
Learn master mode   : FALSE
BFD session state   : UNSET
```

Display the result on R2.

```
Switch# show vrrp
vrrp session count: 1
VRID <1>
 State           : Backup
 Virtual IP      : 10.10.10.1(Not IP owner)
 Interface       : eth-0-9
 VMAC            : 0000.5e00.0101
 VRF             : Default
 Advt timer      : 5 second(s)
 Preempt mode    : TRUE
 Conf pri        : 90          Run pri  : 90
 Increased pri   : 0
 Master router ip   : 10.10.10.50
 Master priority    : 100
 Master advt timer  : 5 second(s)
 Master down timer  : 16 second(s)
 Preempt delay      : 0 second(s)
 Learn master mode  : FALSE
 BFD session state  : UNSET
```

12.8.3 Application Cases

N/A

12.9 Configuring Track

12.9.1 Overview

Function Introduction

Track is used for link the functional modules and monitor modules. Track builds a system structure with 3 levels: “functional modules – Track – monitor modules”.

Track can shield the difference of the monitor modules and provide an unitized API for the functional modules.

The following monitor modules are supported:

- IP SLA
- interface states
- bfd states
- The following functional modules are supported:
- Static route
- VRRP

Track makes a communication for the functional modules and monitor modules. When link states or network performance is changed, the monitor modules can detect the event and notify the track module; therefore track will change its owner states and notify the related functional modules.

Principle Description

N/A

12. 9. 2 Configuration

Configuring IP SLA for interfaces in the VRF

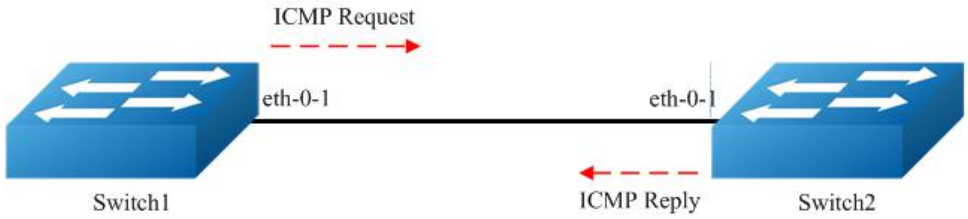


Figure 1-96 IP SLA

IP SLA (Service Level Agreement) is a network performance measurement and diagnostics tool that uses active monitoring. Active monitoring is the generation of traffic in a reliable and predictable manner to measure network performance. Every IP SLA operation maintains an operation return-code value. This return code is interpreted by the tracking process. The return code can return OK, Over Threshold, and several other return codes. Different operations can have different return-code values, so only values common to all operation types are used. In IP SLA, use icmp echo to check state or reachability of a route.

The following configuration should be operated on all switches if the switch ID is not specified.

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create a vrf instance

```
Switch(config)# ip vrf vpn1
Switch(config-vrf)# exit
```

step 3 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip vrf forwarding vpn1
Switch(config-if)# ip address 192.168.0.2/24
Switch(config-if)# exit
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip vrf forwarding vpn1
Switch(config-if)# ip address 192.168.0.1/24
Switch(config-if)# exit
```

step 4 Create ip sla and set the attributes

Configuring Switch1:

```
Switch(config)# ip sla monitor 1
Switch(config-ipsla)# type icmp-echo 192.168.0.1
Switch(config-ipsla)# frequency 35
Switch(config-ipsla)# timeout 6
Switch(config-ipsla)# threshold 3000
Switch(config-ipsla)# ttl 65
Switch(config-ipsla)# tos 1
Switch(config-ipsla)# data-size 29
Switch(config-ipsla)# data-pattern abababab
Switch(config-ipsla)# fail-percent 90
Switch(config-ipsla)# packets-per-test 4
```

```
Switch(config-ipsla)# interval 9
Switch(config-ipsla)# statistics packet 10
Switch(config-ipsla)# statistics test 3
Switch(config-ipsla)# vrf vpn1
Switch(config-ipsla)# exit
```

NOTE: Parameters for ip sla:

- frequency:Time between 2 probes. Valid range is 1-4800 second, default value is 60 seconds.
- timeout:Timeout value for icmp reply. Valid range is 1-4800 second, default value is 5 seconds.
- threshold: Timeout value for icmp threshold. Valid range is 1-4800000 millisecond, default value is 5000 millisecond.
- packets-per-test:Packet number for each probe. Valid range is 1-10, default value is 3.
- interval:Time between 2 packets. Valid range is 1-4800 second, default value is 6 seconds.
- statistics packet:Packet number for statistics. Valid range is 0-1000, default value is 50.
- statistics test probe number for statistics. Valid range is 0-10, default value is 5

step 5 Enable ip sla

Configuring Switch1:

```
Switch(config)# ip sla monitor schedule 1
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

Display the result on Switch1.

```
Switch# sho ip sla monitor 1
Entry 1
Type            : Echo
Admin state     : Disable
Destination address  : 192.168.0.1
Frequency       : 35s
Timeout         : 6s
Threshold       : 3000ms
Interval        : 9s
Packet per test    : 4
TTL             : 65
TOS             : 1
Data Size       : 29 bytes
Fail Percent    : 90%
Packet Item Cnt    : 10
Test Item Cnt      : 3
Vrf             : vpn1
Return code     : Unknown
```

Configuring IP SLA for Layer3 interfaces

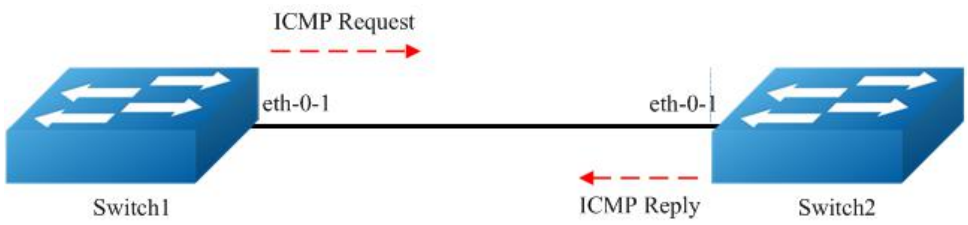


Figure 1-97 IP SLA

The following configuration should be operated on all switches if the switch ID is not specified.:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.2/24
Switch(config-if)# exit
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.1/24
Switch(config-if)# exit
```

step 3 Create ip sla and set the attributes

Configuring Switch1:

```
Switch(config)# ip sla monitor 1
Switch(config-ipsla)# type icmp-echo  192.168.0.1
Switch(config-ipsla)# frequency 10
Switch(config-ipsla)# timeout 5
Switch(config-ipsla)# exit
```

step 4 Enable ip sla

Configuring Switch1:

```
Switch(config)# ip sla monitor schedule 1
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on Switch1.

```
Switch# show ip sla monitor
Entry 1
  Type                  : Echo
  Admin state           : Enable
  Destination address   : 192.168.0.1
  Frequency             : 10 seconds
  Timeout               : 5 seconds
  Threshold             : 5 seconds
  Running Frequency     : 8 seconds
Return code             : OK

Switch# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=0.846 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=64 time=0.643 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=64 time=0.978 ms
64 bytes from 192.168.0.1: icmp_seq=4 ttl=64 time=0.640 ms
64 bytes from 192.168.0.1: icmp_seq=5 ttl=64 time=0.704 ms
```

Shutdown the interface eth-0-1 on Switch2.

```
Switch(config)# interface eth-0-1
Switch(config-if)# shutdown
```

Display the result on Switch1 again.

```
Switch# show ip sla monitor
Entry 1
  Type                    : Echo
  Admin state             : Enable
  Destination address     : 192.168.0.1
  Frequency               : 10 seconds
  Timeout                 : 5 seconds
  Threshold               : 5 seconds
  Running Frequency       : 9 seconds
  Running Timeout         : 4 seconds
  Running Threshold       : 4 seconds
Return code               : Timeout
```

Configuring IP SLA for outgongin interface of static route

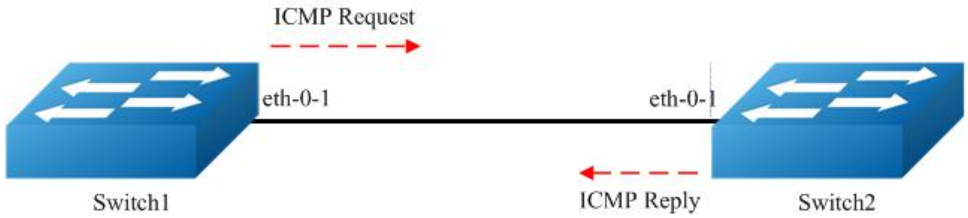


Figure 1-98 IP SLA

The following configuration should be operated on all switches if the switch ID is not specified.:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.2/24n
Switch(config-if)# exit
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.1/24
Switch(config-if)# exit

Switch(config)# interface loopback 1
Switch(config-if)# ip address 1.1.1.1/32
Switch(config-if)# exit
```

step 3 Create ip sla and set the attributes

Configuring Switch1:

```
Switch(config)# ip sla monitor 2
Switch(config-ipsla)# type icmp-echo 1.1.1.1
Switch(config-ipsla)# frequency 10
Switch(config-ipsla)# timeout 5
Switch(config-ipsla)# exit
```

step 4 Enable ip sla

Configuring Switch1:

```
Switch(config)# ip sla monitor schedule 2
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on Switch1.

```
Switch# show ip sla monitor 2
Entry 2
  Type                  : Echo
  Admin state           : Enable
  Destination address   : 1.1.1.1
  Frequency             : 10 seconds
  Timeout               : 5 seconds
  Threshold             : 5 seconds
  Running Frequency     : 1 seconds
  Return code           : Unreachable
Switch# ping 1.1.1.1
connect: Network is unreachable
```

Create a static route on Switch1

```
Switch#configure terminal
Switch(config)# ip route 1.1.1.1/32 192.168.0.1
Switch(config)# end
```

Display the result on Switch1 again.

```
Switch# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=64 time=1.03 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=64 time=1.63 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=64 time=0.661 ms
64 bytes from 1.1.1.1: icmp_seq=4 ttl=64 time=0.762 ms
64 bytes from 1.1.1.1: icmp_seq=5 ttl=64 time=0.942 ms


Entry 2
  Type                  : Echo
  Admin state           : Enable
  Destination address   : 1.1.1.1
  Frequency             : 10 seconds
  Timeout               : 5 seconds
  Threshold             : 5 seconds
  Running Frequency     : 8 seconds
  Return code           : OK
```

Configuring track interface linkstate

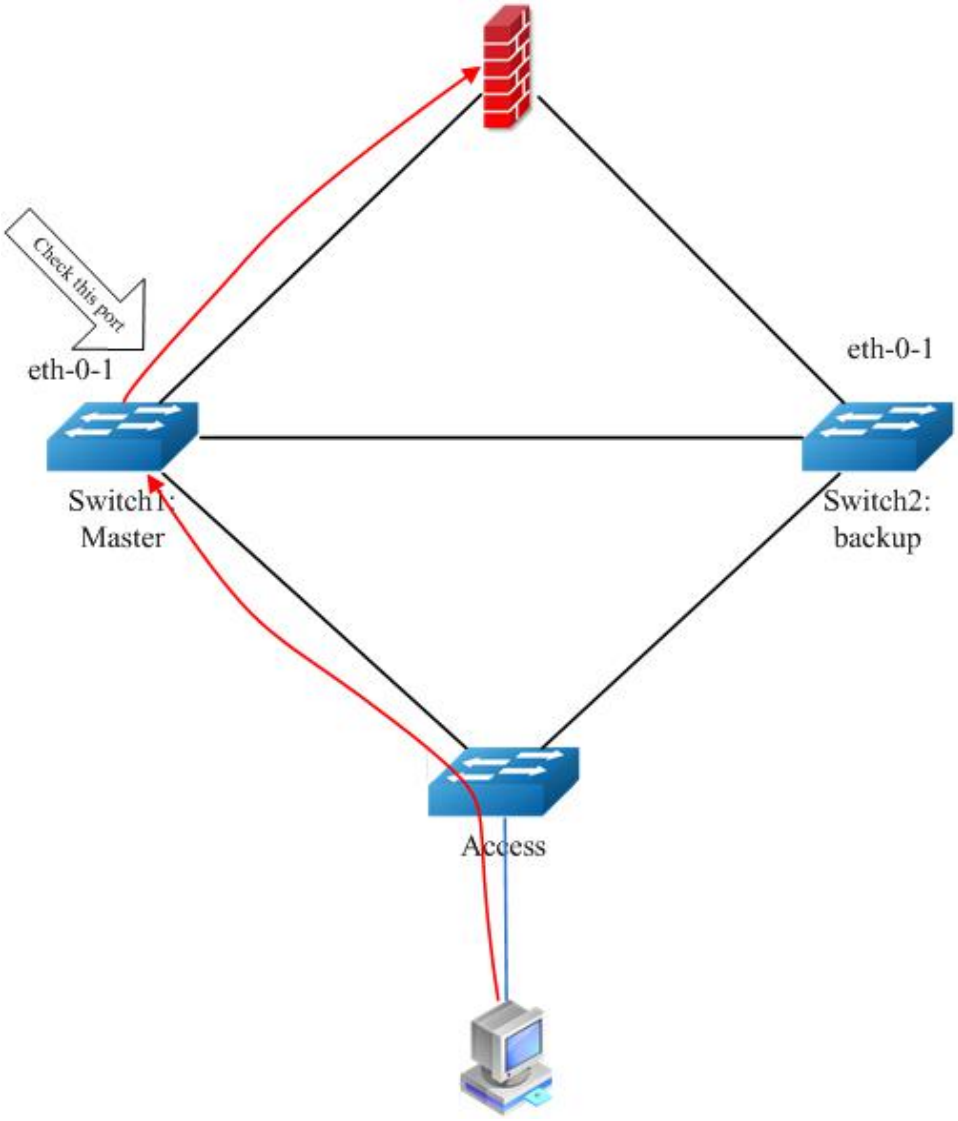


Figure 1-99 Track interface

Before the introduction of track feature, the VRRP had a simple tracking mechanism that allowed you to track the interface link state only. If the link state of the interface went down, the VRRP priority of the router was reduced, allowing another VRRP router with a higher priority to become active. The Track feature separates the tracking mechanism from VRRP and creates a separate standalone tracking process that can be used by other processes in future. This feature allows tracking of other objects in addition to the interface link state. VRRP can now register its interest in tracking objects and then be notified when the tracked object changes state. TRACK is a separate standalone tracking process that can be used by other processes as well as VRRP. This feature allows tracking of other objects in addition to the interface link state.

Configuring Switch1:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Create track and set the attributes

```
Switch(config)# track 1 interface eth-0-1 linkstate
Switch(config-track)# delay up 30
Switch(config-track)# delay down 30
Switch(config-track)# exit
```

Parameters for track:

- delay up: After the interface states is up, the track will wait for a cycle before restore the states. Valid range is 1-180 second. The default configuration is restore without delay.
- delay down: After the interface states is down, the track will wait for a cycle before change the states. Valid range is 1-180 second. The default configuration is change without delay.

NOTE: If the track is using bfd or ip sla, the "delay up" and "delay down" is similar as using interface states.

step 3 Exit the configure mode

```
Switch(config)# end
```

step 4 Validation

```
Switch#show track
Track 2
  Type            : Interface Link state
  Interface       : eth-0-1
  State           : down
  Delay up        : 30 seconds
Delay down        : 30 seconds
```

Configuring track ip sla reachability

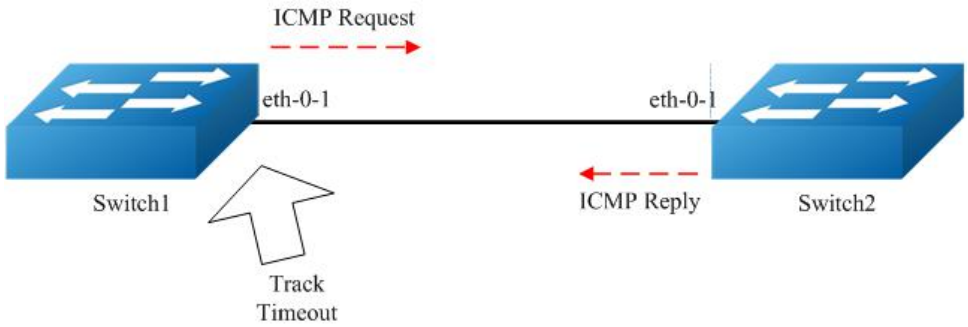


Figure 1-100 Track ip sla

The following configuration should be operated on all switches if the switch ID is not specified.:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.2/24
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.1/24
```

step 3 Create ip sla and enable it

Configuring Switch1:

```
Switch(config)# ip sla monitor 1
Switch(config-ipsla)# type icmp-echo  192.168.0.1
Switch(config-ipsla)# frequency 10
Switch(config-ipsla)# timeout 5
Switch(config-ipsla)# threshold 1
Switch(config-ipsla)# exit
Switch(config)# ip sla monitor schedule 1
```

step 4 Create track and set the attributes

Configuring Switch1:

```
Switch(config)# track 1 rtr 1 reachability
Switch(config-track)# delay up 30
Switch(config-track)# delay down 30
Switch(config-track)#exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch#show track
Track 1
  Type              : Response Time Reporter(RTR) Reachability
  RTR entry number  : 1
  State             : up
  Delay up          : 30 seconds
Delay down          : 30 seconds
```

Configuring track ip sla state

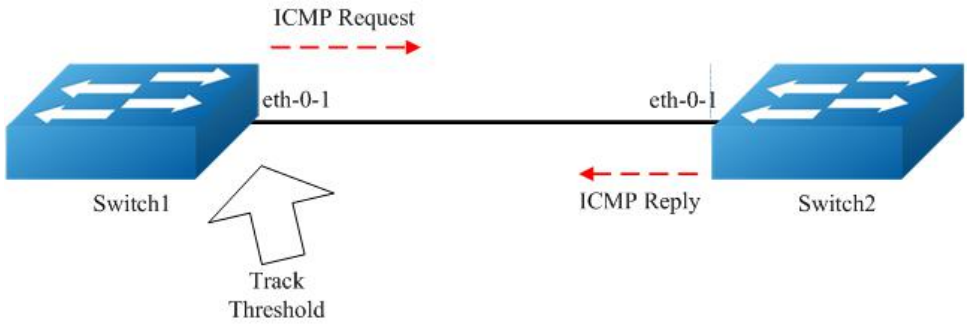


Figure 1-101 Track ip sla

The following configuration should be operated on all switches if the switch ID is not specified.:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.2/24
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
```

```
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.0.1/24
```

step 3 Create ip sla and enable it

Configuring Switch1:

```
Switch(config)# ip sla monitor 1
Switch(config-ipsla)# type icmp-echo  192.168.0.1
Switch(config-ipsla)# frequency 10
Switch(config-ipsla)# timeout 5
Switch(config-ipsla)# threshold 1
Switch(config-ipsla)# exit
Switch(config)# ip sla monitor schedule 1
```

step 4 Create track and set the attributes

Configuring Switch1:

```
Switch(config)# track 1 rtr 1 state
Switch(config-track)# delay up 30
Switch(config-track)# delay down 30
Switch(config-track)#exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

```
Switch# show track
Track 1
  Type              : Response Time Reporter(RTR) State
  RTR entry number  : 1
  State             : up
  Delay up          : 30 seconds
Delay down          : 30 seconds
```

Configuring track bfd

Figure 1-102 Track bfd

The following configuration should be operated on all switches if the switch ID is not specified.:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
```

```
Switch(config-if)# ip address 9.9.9.1/24
Switch(config-if)# quit
```

Interface configuration for Switch2:

```
Switch(config)# interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 9.9.9.2/24
Switch(config-if)# quit
```

step 4 Create track and set the attributes

Configuring Switch1:

```
Switch(config)# track 1 bfd source interface eth-0-1 destination 9.9.9.2
Switch(config-track)# delay up 30
Switch(config-track)# delay down 30
Switch(config-track)# exit
```

Configuring Switch2:

```
Switch(config)# track 1 bfd source interface eth-0-1 destination 9.9.9.1
Switch(config-track)# delay up 30
Switch(config-track)# delay down 30
Switch(config-track)# exit
```

step 5 Exit the configure mode

```
Switch(config)# end
```

step 6 Validation

Display the result on Switch1.

```
Switch #show track
Track 1
  Type            : BFD state
  Source interface   : eth-0-1
  Destination IP     : 9.9.9.2
  BFD Local discr    : 1
  State           : up
```

Display the result on Switch2.

```
Switch # show track
Track 1
  Type            : BFD state
  Source interface   : eth-0-1
  Destination IP     : 9.9.9.1
  BFD Local discr    : 1
  State           : up
```

Configuring track for vrrp

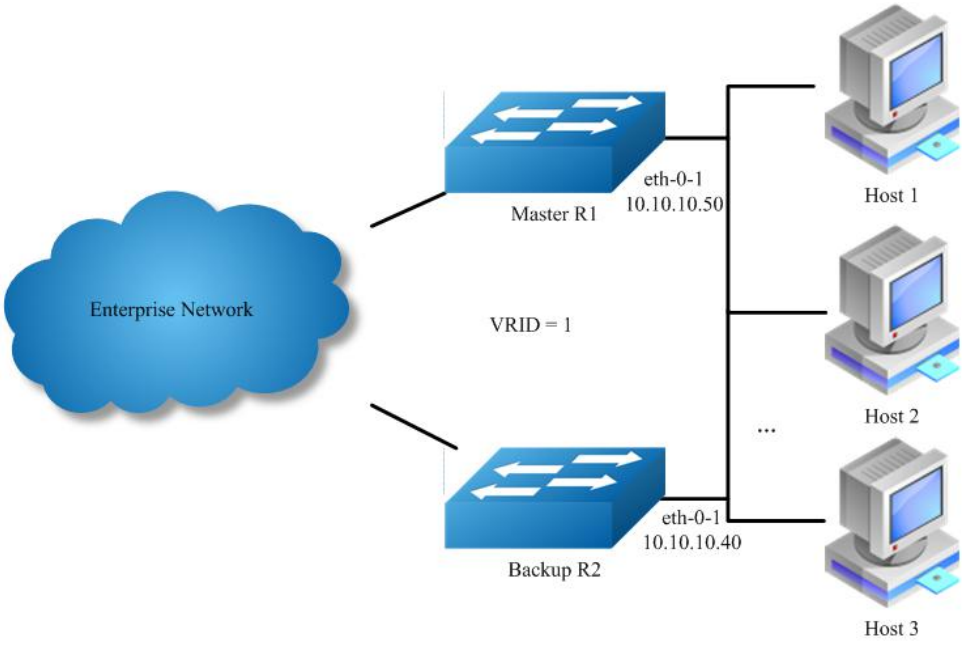


Figure 1-103 VRRP Track

step 1 Check current configuration

Reference to chapter "Configuring VRRP" - "Configuring VRRP (One Virtual Router)"

Display the configuration on R1.

```
interface eth-0-1
 no switchport
 ip address 10.10.10.50/24
!
router vrrp 1
 interface eth-0-1
 virtual-ip 10.10.10.60
 advertisement-interval 5
 enable
```

Display the configuration on R2.

```
interface eth-0-1
 no switchport
 ip address 10.10.10.40/24
!
router vrrp 1
 interface eth-0-1
 priority 200
 virtual-ip 10.10.10.60
 advertisement-interval 5
 enable
```

step 2 Create track and set the attributes

Create track on Switch1

```
Switch(config)# track 1 interface eth-0-1 linkstate
Switch(config-track)# exit
```

step 3 Apply track for vrrp

Apply track on Switch1

```
Switch(config)# router vrrp 1
Switch(config-router)# disable
Switch(config-router)# track 1 decrement 30
Switch(config-router)# enable
```

step 4 Validation

Display the result on Switch1.

```
Switch# show vrrp
vrrp session count: 1
VRID <1>
 State           : Backup
 Virtual IP      : 10.10.10.60(Not IP owner)
 Interface       : eth-0-9
 VMAC            : 0000.5e00.0101
 VRF             : Default
 Advt timer      : 5 second(s)
 Preempt mode    : TRUE
 Conf pri        : Unset       Run pri  : 100
 Increased pri   : 0
 Track Object    : 1
 Decre pri       : 30
 Master router ip  : 10.10.10.40
 Master priority   : 200
 Master advt timer  : 5 second(s)
 Master down timer  : 16 second(s)
 Preempt delay     : 0 second(s)
 Learn master mode  : FALSE
 BFD session state  : UNSET
```

Configuring track for static route

Figure 1-104 Static Route Track

The following configuration should be operated on all switches if the switch ID is not specified.:

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 2 Enter the interface configure mode and set the attributes of the interface

Interface configuration for Switch1:

```
Switch(config)#interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.1.10/24
Switch(config-if)# exit
```

Interface configuration for Switch2:

```
Switch(config)#interface eth-0-1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# ip address 192.168.1.11/24
Switch(config-if)# exit
```

step 3 Create ip sla and enable it

Configuring Switch1:

```
Switch(config)# ip sla monitor 1
Switch(config-ipsla)# type icmp-echo 192.168.1.11
Switch(config-ipsla)# exit
Switch(config)# ip sla monitor schedule 1
```

step 4 Create track and set the attributes

Configuring Switch1:

```
Switch(config)# track 1 rtr 1 reachability
Switch(config-track)# exit
```

step 5 Apply track for static route

```
Switch(config)#ip route 10.10.10.0/24 192.168.1.11 track 1
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

Display the result on Switch1.

```
Switch# show ip sla monitor 1
Entry 1
  Type             : Echo
  Admin state      : Enable
  Destination address  : 192.168.1.11
  Frequency        : 60 seconds
  Timeout          : 5 seconds
  Threshold        : 5 seconds
  Running Frequency   : 49 seconds
Return code      : OK

Switch# show track 1
Track 1
  Type             : Response Time Reporter(RTR) Reachability
  RTR entry number   : 1
  State            : up
Switch# show ip route static
S      10.10.10.0/24 [1/0] via 192.168.1.11, eth-0-1
```

Shutdown the interface eth-0-1 on Switch2.

```
Switch(config)# interface eth-0-1
Switch(config-if)# shutdown
```

Display the result on Switch1 again.

```
Switch# show ip sla monitor 1
Entry 1
  Type             : Echo
  Admin state      : Enable
  Destination address  : 192.168.1.11
  Frequency        : 60 seconds
  Timeout          : 5 seconds
  Threshold        : 5 seconds
  Running Frequency   : 8 seconds
Return code      : Timeout
Switch# show track 1
Track 1
  Type             : Response Time Reporter(RTR) Reachability
  RTR entry number   : 1
  State            : down
Switch# show ip route static
Switch#
```

12. 9. 3 Application Cases

N/A

12. 10 Configuring VARP

12. 10. 1 Overview

Function Introduction

Virtual ARP (VARP) allows multiple switches to simultaneously route packets with the same destination MAC address. Each switch is configured with the same virtual MAC address for the the L3 interfaces configured with a virtual IP address. In MLAG configurations, VARP is preferred over VRRP because VARP working on active-active mode without traffic traverse peer link.

For ARP and GARP requests to virtual IP address, VARP will use the virtual MAC address to reply. The virtual MAC address is only used in the destination field of inbound packets and never used in the source field of outbound packets.Topology

Principle Description

N/A

12. 10. 2 Configuration

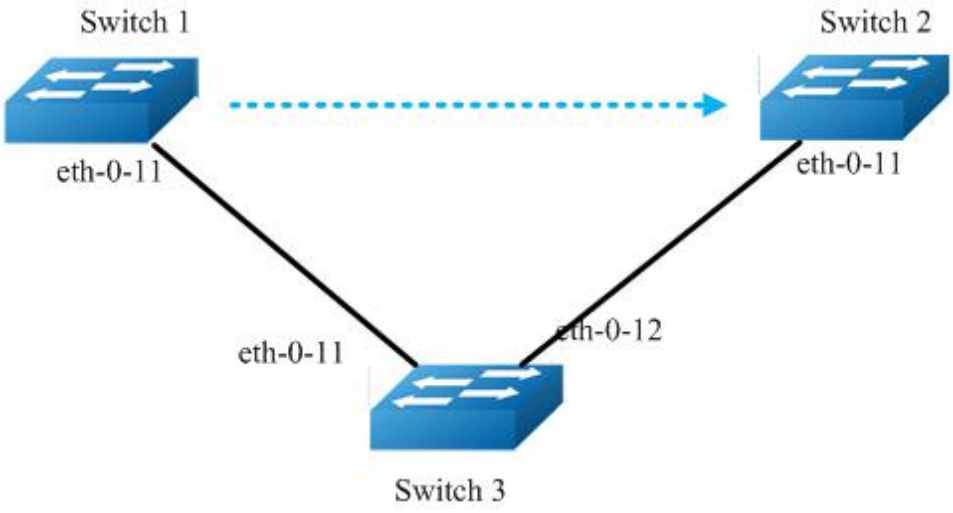


Figure 1-105 VARP with MALG

The following configuration should be operated on all devices if the device ID is not specified.

step 2 Set the virtual-router mac address

```
Switch(config)# ip virtual-router mac a.a.a
```

step 3 Enter the vlan configure mode and create the vlan

```
Switch(config)# vlan database
Switch(config-vlan)# vlan 2
Switch(config-vlan)# exit
```

step 1 Enter the configure mode

```
Switch# configure terminal
```

step 4 Enter the interface configure mode and set the attributes of the interface

```
Switch(config)# interface eth-0-11
Switch(config-if)# switchport access vlan 2
```

```
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

step 5 Create the vlan interface and set ip and virtual router ip

Configuring Switch1:

```
Switch(config)# interface vlan 2
Switch(config-if)# ip address 10.10.10.1/24
Switch(config-if)# ip virtual-router address 10.10.10.254
Switch(config-if)# exit
```

Configuring Switch2:

```
Switch2(config-if)# interface vlan 2
Switch2(config-if)# ip address 10.10.10.2/24
Switch2(config-if)# ip virtual-router address 10.10.10.254
Switch(config-if)# exit
```

step 6 Exit the configure mode

```
Switch(config)# end
```

step 7 Validation

Display the result on Switch1.

```
Switch# show ip arp
 Protocol   Address        Age (min)  Hardware Addr   Interface
 Internet   10.10.10.1          -  cef0.12da.8100  vlan2
 Internet   10.10.10.254        -  000a.000a.000a  vlan2
```

Display the result on Switch2.

```
Switch#  show ip arp
 Protocol   Address        Age (min)  Hardware Addr   Interface
 Internet   10.10.10.2          -  66d1.4c26.e100  vlan2
 Internet   10.10.10.254        -  000a.000a.000a  vlan2
```

12.10.3 Application Cases

N/A

https://www.fs.com

The information in this document is subject to change without notice. FS has made all efforts to ensure the accuracy of the information, but all information in this document does not constitute any kind of warranty.