目录

1 前言

声明

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

版本说明

本手册对应产品版本为：通用网络操作系统 SWITCH 1.0

使用范围

本书简介

章节安排如下：

- **系统管理** 介绍系统信息，访问配置、SNMP 配置等
- **接口管理** 介绍物理端口，风暴抑制，端口限速等
- **业务管理** 介绍 vlan 配置，MAC 配置，MSTP 配置，二层组播配置等
- **路由管理** 介绍三层接口，查看路由、Static 管理等
- **安全管理** 介绍访问控制，防攻击设置，ACL 配置，流量监控，告警配置等
- **系统维护** 介绍日志配置，诊断测试，NTP，重启设备等
- **附录 A 缩略语** 列举了本手册中出现的缩略语

读者对象

本书适合下列人员阅读：

- 网络工程师
- 网络管理人员
- 具备网络基础知识的用户

本书约定

1.1 通用格式约定

格式	意义
宋体	正文采用宋体表示。
黑体	除一级标题采用宋体加粗以外，其余各级标题均采用黑体。
楷体	警告、提示等内容一律用楷体，并且在内容前后增加线条与正文隔离。
“Terminal Display”格式	自定义的“Terminal Display”格式（英文 Courier New； 中文 宋体； 文字大小 8.5）表示屏幕输出信息。此外，屏幕输出信息中夹杂的用户从终端输入的信息采用加粗字体表示。

1.2 命令行格式约定

格式	意义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用加粗字体表示。
斜体	命令行参数 （命令中必须由实际值进行替代的部分） 采用斜体表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
(x | y | ...)	表示从两个或多个选项中选取一个。
[x | y | ...]	表示从两个或多个选项中选取一个或者不选。
#	由“#”开始的行表示为注释行。

1. 各类标志

本书还采用各种醒目标志来表示在操作过程中应该特别注意的地方，这些标志的意义如下：

注意：提醒操作中应注意的事项。

说明：对操作内容的描述进行必要的补充和说明。

2 系统管理

2.1 系统信息

2.1.1 write 配置保存

在特权视图下，使用 write 命令来保存配置

命令	说明
write	保存当前运行配置，作为启动配置

#示例

```
switch# write
Building Configuration...
Integrated configuration saved [OK]
switch#
```

2.1.2 show running-config 查看运行配置

在所有视图下，使用该命令来查看当前运行配置

命令	说明
show running-config	查看当前运行配置

#示例

```
switch# show  running-config
Building configuration...

Current configuration:
!
!
username aa password aa
!
vlan 1
!
interface ge1/1
!
interface ge1/2
!
interface ge1/3
!
interface ge1/4
!
interface ge1/5
!
interface ge1/6
!
interface ge1/7
!
interface ge1/8
!
interface ge1/9
!
interface ge1/10
!
interface ge1/11
!
interface ge1/12
!
interface ge1/13!
--More--
```

2.1.3 show version 查看版本

在特权视图下，使用该命令查看 SWITCH 运行版本

命令	说明
show version	查看SWITCH版本。

#示例

```
switch# show version

CNS 3.1.171-rc build 669
GIT 0e52899 SDK 6.5.7
Compiled Aug 26 2019 15:02:45 by root
Copyright (C) 2017-2018.

Product     : S8500
SN          : 000000037102
System Mac  : 00:00:70:00:00:00
Up time     : 0 days, 0 hours, 25 minutes
Temperature : 33 degree celsius

switch#
```

2.1.4 hostname 设置系统名

在全局配置视图下，使用该命令设置 hostname

命令	说明
hostname	设置设备名称
no hostname	恢复设备名称为默认值switch

#示例，以下将当前配置的设备名称设置为 sw1

```
switch(config)# hostname  sw1
sw1(config)#
```

2.1.5 username 用户配置

在全局配置视图下，使用该命令添加/删除用户

命令	说明
username *NAME* **password** *PASSWD*	添加用户： username kemyond password Kemyond@123
no username *NAME*	删除用户

#示例，以下命令添加一个用户，用户名为 Aaaaaaaa8，密码为 Bbbbbbb8

```
switch(config)# username Aaaaaaaa8 password Bbbbbbb8
switch(config)#
```

2.1.6 clock 时间配置

在特权视图下，使用该命令添加/删除用户

命令	说明
clock set {date | time}	设置系统时间，日期格式为：YYYY-MM-DD，时间格式为：HH-MM-SS
show clock	显示系统时间

#示例，以下命令修改系统日期为 2010-01-03

```
switch# clock  set  2010-1-3
Sun Jan  3 00:00:00 CST 2010
switch#
```

#以下命令显示当前系统时间

```
switch# show  clock
Sun Nov 10 21:09:45 CST 2013
switch#
```

2.2 访问配置

2.2.1 HTTP 配置

2.2.1.1 使能 http/https 服务

默认情况下 http 服务是开启的。

在全局配置态下使用下面的命令使能 http 服务：

命令	说明
ip http-server (all | http | https)	使能http|https服务

#示例，以下命令开启 http 服务

```
switch(config)# ip http-server http
switch(config)#
```

2.2.1.2 改变 http 服务的端口号

默认情况下 http 服务的监听端口是 80。

在全局配置态下使用下面的命令进行改变 http 服务的端口号：

命令	说明
ip http *port-number*	改变http服务的端口号

#示例，以下命令开启 http 服务端口号为 8080

```
switch(config)# ip http-server port 8080
switch(config)#
```

2.2.2 telnet 配置

2.2.2.1 使能和关闭 telnet 服务

默认情况下 telnet 服务是关闭的。

在全局配置态下使用下面的命令使能/关闭 telnet 服务：

命令	说明
ip telnet-server	使能telnet服务
no ip telnet-server	关闭telnet服务

#示例，以下命令开启 telnet 服务

```
switch(config)# ip telnet-server
switch(config)#
```

#示例，以下命令关闭 telnet 服务

```
switch(config)# no ip telnet-server
switch(config)#
```

2.2.2.2 改变 telnet 服务器的端口号

默认情况下 telnet 服务的监听端口是 23。

在全局配置态下使用下面的命令进行改变 telnet 服务的端口号：

命令	说明
ip telnet-server port *port-number*	改变telnet服务的端口号

#示例，以下命令开启 telnet 服务端口号为 9998

```
switch(config)# ip telnet-server port 9998
switch(config)#
```

2.2.3 SSH 配置

2.2.3.1 使能和关闭 SSH 服务

默认情况下 ssh 服务是关闭的。

在全局配置态下使用下面的命令使能/关闭 ssh 服务：

命令	说明
ip ssh-server	使能ssh服务
no ip ssh-server	关闭ssh服务

```
#示例，以下命令开启 ssh 服务
switch# configure terminal
switch(config)# ip ssh-server
switch(config)#

#示例，以下命令关闭 ssh 服务
switch(config)# no ip ssh-server
switch(config)#
```

2.2.3.2 改变 SSH 服务器的端口号

默认情况下 ssh 服务的监听端口是 22。

在全局配置态下使用下面的命令进行改变 ssh 服务的端口号：

命令	说明
ip ssh-server port *port-number*	改变ssh服务的端口号

```
#示例，以下命令开启 ssh 服务端口号为 9999
switch(config)# ip ssh-server port 9999
switch(config)#
```

2.3 SNMP 配置

2.3.1 简介

2.3.1.1 概述

简单网络管理协议 SNMP（Simple Network Management Protocol）是被广泛接受并使用的网络标准。SNMP 是应用层协议。它提供了在 SNMP 管理端和代理之间进行通信的报文格式。

SNMP 保证管理信息在任意两点间传送，便于网络管理员在网络上的任何节点检索信息，进行修改，寻找故障，完成故障诊断，容量规划和报告生成。

2.3.1.2 SNMP 结构

SNMP 系统包括下面 3 个部分：

- SNMP 管理端 (NMS)
- SNMP 代理 (AGENT)
- 管理信息库 （MIB）

SNMP 管理端可以是网络管理系统（NMS，如 CiscoWorks）的一部分。代理和 MIB 驻留在

系统上。配置系统上的 SNMP，需要定义管理端和代理间的关系。

SNMP 代理包含 MIB 变量，SNMP 管理端可以查询或改变这些变量的值。管理端可以从代理处得到变量值，或者把变量值存储到代理处。代理从 MIB 收集数据。MIB 是设备参数 和网络数据的信息库。代理也能响应管理端的读取或设置数据的请求。SNMP 代理可以主动向管理端发送陷阱（trap）。陷阱是针对网络的某一条件而向 SNMP 管理端报警的消息。陷阱能指出不正确的用户认证、重启、链路状态（启动或关闭）、TCP 连接的关闭、与邻近系统连接的丢失或其它重要的事件。

1）Agent 是驻留在被管理设备上的一个进程。

Agent 实现以下功能：

- 接收、处理来自网管站的 Request 报文。
- 根据报文类型对管理变量进行 Read 或 Write 操作，并生成 Response 报文，返回给 NMS。
- 在设备发生冷启动或热启动等异常情况时，主动向 NMS 发送 Trap 报文，报告所发生的事件。

2）网管站 NMS

网管站 NMS 是运行客户端程序的工作站，目前常用的网管平台有 Sun NetManager 和 IBM NetView。

网管站实现以下功能：

- 向网络设备发送各种查询报文。
- 接收来自被管理设备的响应报文及 Trap 报文，并显示结果

2.3.1.3 SNMP 通告

特殊事件发生时系统能向 SNMP 管理端发送通知(inform)。例如，当代理系统遭遇一个错误条件时，它可能向管理端发送一个消息。

SNMP 通告可以作为陷阱（trap）或通知请求（inform request）来发送。由于接收方收到一个陷阱时不发送任何应答，导致发送方不能确定是否陷阱已经被接收，所以陷阱不可靠。与此相对的是，接收通知请求的 SNMP 管理端用 SNMP 响应 PDU 作为这个消息的应 答。如果管理端没有收到一个通知请求，也不会发送响应。如果发送方没有收到应答， 那 么 可以重新发送通知请求。这样，通告更可能到达它们计划中的目的地。

因为通知请求更加可靠，所以它们消耗了系统和网络的更多的资源。陷阱只要一发出便被丢弃。与此不同的是，通知请求必须保留在内存中，直到收到响应或者请求超时。另外，陷阱只发送一次，而通知请求可以重新发送多次。重新发送增加了网络通信量并在网络上产生更多的负荷。因此，陷阱和通知请求在可靠性和资源间提供了平衡。如果 SNMP 管理端非常需要收到每个通知，可使用通知请求；如果关心网络的通信量或系统的内存，并且不必收到每个通知，可使用陷阱。

2.3.1.4 SNMP 的版本

我司支持下面的 SNMP 版本：

- SNMPv1---简单网络管理协议，一个完全的 Internet 标准，在 RFC1157 中定义。

- SNMPv2C--- SNMPv2 的基于团体的管理框架， Internet 试验协议，在 RFC1901 中定义。
- SNMPv3--- 简单网络管理协议版本 3，在 RFC3410 中定义。

SNMPv1 使用基于团体的安全形式。能访问代理 MIB 的管理端团体用 IP 地址访问控制列表和口令来定义。

SNMPv3 通过对 SNMP 报文进行认证和加密操作来提供对设备访问的安全性。

SNMPv3 提供了下列安全特性：

- 消息完整性：保证消息在传输过程中没有被篡改
- 认证：确保消息的来源地的合法性
- 加密：对消息题进行加密，未经认证的主机即使窃取到消息也无法解密阅读

SNMPv3 提供安全模型和安全级别。安全模型是指一种认证策略，通过配置用户名和该用户所属的组来实现。安全级别是指安全模型中支持的不同的认证方式。SNMPv3 基于用户的安全模型支持三种安全级别，按照从高到低的顺序排列分别是认证并加密、认证不加密和不认证；通过使用 MD5 或 SHA 散列算法计算认证密钥的摘要值在网络间传送并在 SNMP 引擎比对来实现密码不被泄漏，使用 DES 加密算法对报文进行加密保证设备不被第三者窃听。通过配置用户/密码对和用户所属的组来实现管理者对设备的身份认证，通过配置组和视图决定组内的用户不同操作对于管理信息库的访问权限；组同时限制了组内用户的最低安全级别。

必须把 SNMP 代理配置为管理工作站支持的 SNMP 版本。代理才能与多个管理端通信。

2.3.1.5 所支持的 MIB

支持所有的 MIB II 变量（在 RFC 1213 中讲述）和 SNMP 陷阱（RFC 1215 中讲述）。

SWITCH 为每个系统提供了自己私有的 MIB 扩充。

2.3.2 SNMP 配置

2.3.2.1 为 SNMP 创建或修改视图

在全局配置视图下使用下面的命令来配置视图：

命令	说明
snmp mib-view name *WORD1* **(included | excluded) oid** *WORD2*	定义视图。 *WORD1 WORD <1-32>* *WORD2 STRING <1-255>*

#示例，添加一个视图用户为 test，oid 为 1 的视图

```
switch(config)# snmp mib-view name test1 included oid 1
switch(config)#
```

2.3.2.1 为 SNMP 团体创建或修改访问控制

使用 SNMP 团体字符串定义 SNMP 管理端和代理的关系。团体字符串类似于允许访问系统上代理的口令。可选的是，可以指定下面一个或多个与团体字符串相关联的特性：

允许使用团体字符串获得代理访问权的 SNMP 管理端的 IP 地址访问列表。

定义对指定团体有访问权的所有 MIB 对象子集的 MIB 视图。

指定团体对有访问权的 MIB 对象的读写权限。

在全局配置视图下使用下面的命令来配置团体字符串：

命令	说明
snmp group name *WORD1* **(v1 | v2c | v3) read-view (***WORD2* **| none) write-view (***WORD3* **| none) notify-view (***WORD4* **| none)**	定义团体访问字符串 *WORD1 STRING <1-255>* *WORD2 STRING <1-255>* *WORD3 STRING <1-255>* *WORD4 STRING <1-255>*

#示例，添加一个团体用户为 test，版本为 1 的团体

```
switch(config)#  switch(config)#  snmp  group  name  test  v1  read-view defaultView write-view defaultView notify-view defaultView
switch(config)#
```

2.3.2.2 使能 SNMP

默认不使能 SNMP

命令	说明
[no] SNMP	使能/禁用 SNMP

#示例，开启 SNMP 和关闭 SNMP

```
switch(config)# snmp
switch(config)# no snmp
```

2.3.2.3 设置该系统管理员的联系方式和系统所在的位子

sysName、sysContact 和 sysLocation 都是 MIB 中 system 组中的管理变量，分别定义了被管理该节点（系统）的联系人标识和实际位置。这些信息可以通过配置文件进行访问。

在全局配置模式下使用下面的一个或多个命令：

命令	说明
snmp-server sysname *text*	设置节点系统名称字符串。
snmp-server syscontact *text*	设置节点联系人字符串。
snmp-server syslocation *text*	设置节点位置字符串。

2.3.2.4 查看 SNMP 配置

在全局配置模式下使用下　面的命令，查看 SNMP 配置。

命令	说明
show snmp （group | mib-view | user）	查看 SNMP 配置

2.3.2.5 配置 SNMP 陷阱

在全局模式下面的命令配置系统发送 SNMP 陷阱：

命令	说明
snmp trap ip *A.B.C.D* **(v1 | v2c)**	指定陷阱消息的接受者
snmp trap link	开启发送端口状态变化时发送 trap 信息
snmp trap start	开启系统启动时发送 trap 信息
snmp trap traffic	开启发送端口流量超出阈值时发送 trap 信息
snmp trap user	开启用户登陆/离开/添加用户/修改用户等操作时发送 trap 信息
snmp trap operation	开启用户进行 WEB 配置时发送 trap 信息
snmp trap mac	开启 MAC 地址静态绑定时发送 trap 信息

#示例，开启 link、traffic、user、operation、mac 等 trap,且 host 为 192.0.2.30

```
switch(config)# snmp trap link
switch(config)# snmp trap traffic
switch(config)# snmp trap user
switch(config)# snmp trap operation
switch(config)# snmp trap mac
switch(config)# snmp trap ip 192.0.2.30 v1
switch(config)#
```

2.3.2.6 配置 SNMP 用户

通过下面的命令配置一个本地用户，管理者访问设备时，必须使用设备上配置的用户名和密码进行访问。用户的安全级别不能低于用户所属的组的安全级别，否则用户不能通过认证

命令	说明
snmp user *text1* **group-name** *text2* **(auth_priv|auth_no_priv | no_auth_no_priv)**	配置一个本地 SNMPv3 用户

authentication-mode (md5 | sha) *test3* **privacy-mode (aes | des)** *test4*	

配置 SNMPV3 用户为：

```
switch(config)#snmp user test3 group-name test3 auth_priv authentication-mode md5 test3 privacy-mode des test3
switch(config)#
```

3 接口管理

3.1 物理端口

3.1.1 开启和关闭端口

一个接口可被禁止使用，从而禁止使用在指定接口上的所有功能，并且在所有监控命令显示上将此接口标记为不可用接口。

在接口配置态中使用如下命令来关闭/使能接口。

命令	说明
shutdown	停用接口。
no shutdown	重新启用接口。

默认所有端口使能。

想检查一个接口是否被停用，可以使用命令 show interface 和 show running-config。在 show interface 命令显示中，一个已被停用的接口显示为“administratively down”。

#关闭接口示例：

```
switch(config)# int ge1/3
switch(config-ge1/3)# shutdown
switch(config-ge1/3)#
```

3.1.2 配置速率

以太网的速率既可以通过自协商实现，也可以在接口视图下配置。

命令	说明
speed (10 | 100 | 1000 | auto)	设置快速以太网的速率为 10M,100M,1000M 或自协商

注：光接口的 speed 是固定的。如果在光接口 speed 命令后面有 auto 参数，则表示该接口可以打开自动协商功能，否则，该接口是强制的不能协商。以太网电接口支持多种速率：

- FE 电接口支持 10Mbit/s、100Mbit/s 两种速率 。
- GE 电接口支持 10Mbit/s、100Mbit/s、1000Mbit/s 三种速率。
- XE 电接口支持 10GMbit/s。

因此，只需对以太网电接口进行配置，而光接口不需要配置。 用户可以强制指定接口工作速率，但指定的速率值应与对端设备接口的速率相同

#设置端口速率示例：

```
switch(config-ge1/3)# speed  100
switch(config-ge1/3)#
```

3.1.3 配置端口的双工模式

缺省时，以太网接口可以自动协商是半双工或全双工。在千兆端口上，没有该命令。千兆端口的双工模式总是 auto。

命令	说明
duplex (full | half | auto)	设置以太网的双工模式。

默认所有端口自协商。

注：一般设备工作在双工模式；与 Hub 相连时，以太网电接口应选择工作在半双工方式下（因为 Hub 只能工作在半双工方式下）；配置时注意与对端设备的模式相同。

#配置双工模式示例

```
switch# configure terminal
switch(config)# int ge1/1
switch(config-ge1/1)# duplex full
```

3.1.4 配置接口流量模式

在接口为全双工模式时，流量控制通过 802.3X 定义的 PAUSE 帧实现；在接口为半双工模式时，通过背压实现。

命令	说明
flowctrl {both | rx | tx}	打开接口流控。
flowctrl disable	关闭接口流控。

#配置流控示例

```
switch(config)# int ge1/2
switch(config-ge1/2)# flowctrl both    //使能流控
switch(config-ge1/2)# flowctrl disable //禁用流控
switch(config-ge1/2)#
```

3.1.5 配置最大帧长

在接口视图下，使用下面命令，配置端口 max frame size。默认最大帧长，与具体产品型号相关，各产品可能不一致。

命令	说明
max-frame *size*	设置最大帧长。
no max-frame	恢复默认最大帧长

#配置最大帧长为 2000

```
switch(config)# int ge1/2
switch(config-ge1/2)# max-frame  2000
switch(config-ge1/2)# ex
```

```
switch(config)#
```

3.1.6 查看端口信息

在特权视图下，使用该命令查看端口详细信息

命令	说明
show interface *portname*	查看端口的详细信息

#示例

```
switch# show interface ge1/1

ge1/1 is down
  Hardware address is 00:00:70:00:00:01
  Media type is MEDIUM_COPPER
  Link ups:       0  last: (never)
  Link downs:     0  last: (never)
  Autonegotiation enable, Pause tx is on, rx is off
  Speed: 0M, Duplex-full, Max frame size: 1518
  Ifindex: 0x2010001
  PVID is 1, Discard none
    Tag vid  :
    Untag vid:
   0 packets input, 0 bytes
   0 broadcast, 0 multicast
   0 jabber, 0 pause
   0 input errors, 0 CRC, 0 drops
   0 packets output, 0 bytes
   0 broadcast, 0 multicast
   0 output errors, 0 drops
   0 late collision, 0 pause
switch#
```

3.2 风暴抑制

交换机端口可能受到持续的、异常的单播（MAC 地址查找失败）、组播或者广播报文的冲击，造成交换机端口甚至整个交换机的瘫痪。为此，必须提供一种机制来抑制这种现象。

命令	说明
storm-control {broadcast | multicast | unicast} kbps *value*	对广播、多播或者单播报文进行风暴控制
no storm-control {broadcast | multicast | unicast}	不进行风暴抑制。

#以下配置 ge1/5 风暴抑制为 100kbps

```
switch# con t
```

```
switch(config)# int ge1/5
switch(config-ge1/5)# storm-control  broadcast  kbps  100
switch(config-ge1/5)#
```

3.3 端口限速

通过配置可以限制端口进出口的流量速率。进入接口视图下使用下面的命令限制端口的流量速率。

命令	说明
rate-limit *rate burst* **(ingress | egress)**	配置端口的流量速率限制。
no rate-limit (ingress | egress)	转发所有报文。

```
#以下配置 ge1/5 入口带宽为 64kbps
switch# con t
switch(config)# int ge1/5
switch(config-ge1/5)# rate-limit 64 64 ingress
switch(config-ge1/5)#
```

4.4 端口镜像

在全局模式以下命令来设置端口镜像功能

命令	说明
monitor session *value1* **{ both | egress | ingress} destination** *value2* **source** *value3*	设置端口镜像；*value1* 为会话 ID；*value2* 为镜像目的端口名称；*value3* 为镜像源端口名称；
no monitor session *value1*	取消端口镜像。*value1* 为会话 ID

```
#以下配置将端口 ge1/1 的双向数据全部镜像到 ge1/2
switch(config)# monitor session 1 both destination ge1/2 source ge1/1
switch(config)#
#以下配置取消上面设置的端口镜像功能
switch(config)# no monitor session 1
switch(config)#
```

3.5 链路聚合

3.5.1 简介

3.5.1.1 LACP 概述

LACP，基于 IEEE802.3ad 标准的 LACP（Link Aggregation Control Protocol，链路汇

聚控制协议）是一种实现链路动态汇聚的协议。LACP 协议通过 LACPDU（Link Aggregation Control Protocol Data Unit，链路汇聚控制协议数据单元）与对端交互信息。

启用某端口的 LACP 协议后，该端口将通过发送 LACPDU 向对端通告自己的系统优先级、系统 MAC 地址、端口优先级、端口号和操作 Key。对端接收到这些信息后，将这些信息与其它端口所保存的信息比较以选择能够汇聚的端口，从而双方可以对端口加入或退出某个动态汇聚组达成一致。

操作 Key 是在端口汇聚时，LACP 协议根据端口的配置（即速率、双工、基本配置、管理 Key）生成的一个配置组合。

动态汇聚端口在启用 LACP 协议后，其管理 Key 缺省为零。静态汇聚端口在启用 LACP 后，端口的管理 Key 与汇聚组 ID 相同。

对于动态汇聚组而言，同组成员一定有相同的操作 Key，而手工和静态汇聚组中，处于 Active 的端口具有相同的操作 Key。

端口汇聚是将多个端口汇聚在一起形成一个汇聚组，以实现出/入负荷在汇聚组中各个成员端口中的分担，同时也提供了更高的连接可靠性

3.5.1.2 静态汇聚

静态 lacp 汇聚由用户手工配置，不允许系统自动添加或删除汇聚组中的端口。汇聚组中必须至少包含一个端口。当汇聚组只有一个端口时，只能通过删除汇聚组的方式将该端口从汇聚组中删除。

静态汇聚端口的 lacp 协议为激活状态，当一个静态汇聚组被删除时，其成员端口将形成一个或多个动态 lacp 汇聚，并保持 lacp 的被激活。禁止用户关闭静态汇聚端口的 lacp 协议。

在静态汇聚组中，端口可能处于两种状态：selected 或 standby。selected 端口和 standby 端口都能收发 lacp 协议，但 standby 端口不能转发用户报文。

在静态汇聚组中，系统按照以下原则设置端口处于 selected 或者 standby 状态：

系统按照端口全双工/高速率、全双工/低速率、半双工/高速率、半双工/低速率的优先次序，选择优先次序最高的端口处于 selected 状态，其他端口则处于 standby 状态。

与处于 selected 状态的最小端口所连接的对端设备不同，或者连接的是同一个对端设备但端口在不同的汇聚组内的端口将处于 standby 状态。

端口因存在硬件限制（如不能跨板汇聚）无法汇聚在一起，而无法与处于 selected 状态的最小端口汇聚的端口将处于 standby 状态。

与处于 selected 状态的最小端口的基本配置不同的端口将处于 standby 状态。

由于设备所能支持的汇聚组中的 selected 端口数有限制，如果当前的成员端口数超过了设备所能支持的最大 selected 端口数，系统将按照端口号从小到大的顺序选择一些端口为 selected 端口，其他则为 standby 端口

3.5.1.3 动态汇聚

动态 lacp 汇聚是一种系统自动创建/删除的汇聚，不允许用户增加或删除动态 lacp 汇聚中的成员端口。只有速率和双工属性相同、连接到同一个设备、有相同基本配置的端口才能被动态汇聚在一起。即使只有一个端口也可以创建动态汇聚，此时为单端口汇聚。动态汇聚中，端口的 lacp 协议处于使能状态。

在动态汇聚组中，端口可能处于两种状态：selected 或 standby。selected 端口和 standby 端口都能收发 lacp 协议，但 standby 端口不能转发用户报文。

由于设备所能支持的汇聚组中的最大端口数有限制，如果当前的成员端口数量超过了最大端口数的限制，则本端系统和对端系统会进行协商，根据设备 id 优的一端的端口 id 的大小，来决定端口的状态。具体协商步骤如下：

比较设备 id（系统优先级+系统 mac 地址）。先比较系统优先级，如果相同再比较系统 mac 地址。设备 id 小的一端被认为优。

比较端口 id（端口优先级+端口号）。对于设备 id 优的一端的各个端口，首先比较端口优先级，如果优先级相同再比较端口号。端口 id 小的端口为 selected 端口，剩余端口为 standby 端口。

在一个汇聚组中，处于 selected 状态且端口号最小的端口为汇聚组的主端口，其他处于 selected 状态的端口为汇聚组的成员端口。

3.5.1.4 工作模式

启动 LACP 的端口可以有两种工作模式，passive，和 active。

passive：被动模式，该模式下端口不会主动发送 LACPDU 报文，在接收到对端发送的 LACP 报文后，该端口进入协议计算状态。

Active：主动模式，该模式下端口会主动向对端发送 LACPDU 报文，进行 LACP 协议的计算。

3.5.1.5 使用场合

1）在带宽比较紧张的情况下，可以通过逻辑聚合可以扩展带宽到原链路的 n 倍

2）在需要对链路进行动态备份的情况下，可以通过配置链路聚合实现同一聚合组各个成员端口之间彼此动态备份

3.5.2 链路聚合配置

3.5.2.1 配置静态汇聚

全局配置视图创建静态汇聚组使用如下命令配置：

命令	说明
interface *trunkID*	创建静态汇聚组 trunkID:trunk1~trunk32

静态汇聚组接口视图设置静态汇聚组负载模式：

命令	说明
load-balance（both-mac| dst-mac| src-mac）	配置静态汇聚组的负载模式；默认both-mac。

#配置示例，静态 LACP 汇聚，设置负载模式为源 MAC 模式

```
switch(config)# interface trunk2
switch(config-trunk2)# load-balance src-mac
switch(config-trunk2)# exit
switch(config)# interface  ge1/3
switch(config-ge1/3)# trunk  2
switch(config-ge1/3)# exit
switch(config)# interface ge1/4
switch(config-ge1/4)# trunk 2
switch(config-ge1/4)# exit
switch(config)#
```

3.5.2.2 配置动态汇聚

接口视图下如下命令配置：

命令	说明
lacp	开启lacp

全局视图设置 lacp 汇聚组负载模式：

命令	说明
lacp load-balance （ both-mac| dst-mac| src-mac）	配置lacp的负载模式；默认both-mac。

#配置示例，动态 LACP 汇聚

```
switch(config)# interface ge1/5
switch(config-ge1/5)# lacp
switch(config-ge1/5)# exit
switch(config)# interface ge1/6
switch(config-ge1/6)# lacp
switch(config-ge1/6)# exit
switch(config)#
```

3.5.2.3 配置 lacp 工作模式

端口可以支持 LACP 的两种工作模式。

在接口配置视图下使用如下命令配置 LACP 的工作模式：

命令	说明
lacp (active | passive)	配置LACP 主动模式/被动模式

#配置示例，动态 LACP 汇聚

```
switch(config)# interface ge1/5
switch(config-ge1/5)# la
switch(config-ge1/5)# lacp pa
switch(config-ge1/5)# lacp passive
```

```
switch(config-ge1/5)#
```

3.5.2.4 配置 lacp 优先级

系统可支持配置系统和端口 LACP 优先级

在全局配置视图下使用下面的命令可以配置系统的 LACP 优先级：

命令	说明
lacp system-priority 2	配置系统的LACP优先级，取值范围：<0-32768>

在接口配置视图下使用下面的命令可以配置端口的 LACP 优先级：

命令	说明
lacp port-priority 0	配置端口的LACP优先级，取值范围：<0-32768>

#配置示例，动态 LACP 优先级

系统优先级：

```
switch(config)# interface ge1/5
switch(config-ge1/5)# lacp port-priority 6
```

端口优先级：

```
switch(config)# lacp system-priority 8
```

3.5.3 查看与调试

3.5.3.1 LACP 状态查询

通过显示命令可以观察 LACP 相关的信息，LACP 状态和端口状态信息。

在全局视图下，使用下面显示命令：

命令	说明
show trunk [verbose]	显示LACP的相关信息，加参数verbose可以看到详细的信息。

#操作示例，查看 LACP 相关信息

```
switch# show trunk

Group  Status   Psc         Ports
-------------------------------------
1      DOWN     src-mac
3      DOWN     both-mac    ge1/6

switch#
```

3.5.3.2 LACP 调试

使用下面的命令，可以显示 LACP 调试信息

命令	说明
debug lacp all	监视LACP所有信息
debug lacp events	监视LACP事件
debug lacp interface	监视LACP接口信息
debug lacp packet	监视LACP报文

#操作示例，显示 LACP 调试信息

```
switch# debug lacp all
switch#
```

3.6 端口隔离

在正常情况下，交换机的不同端口间的数据包能够自由的转发。在某些情况下，需要禁止端口之间的数据流，端口隔离功能就是提供这种控制的，设置隔离功能的端口之间不能够再有数据包通信，其它没有隔离的端口之间以及隔离端口和未隔离端口之间的数据包仍然能够正常转发。

命令	说明
isolate *[IFNAME]*	设置端口隔离。
no isolate *[IFNAME]*	取消端口隔离。

#配置示例：以下配置端口 ge1/3 和 ge1/4 端口隔离，不能互通

```
switch> en
switch# con t
switch(config)# int ge1/3
switch(config-ge1/3)# isolate ge1/4
switch(config-ge1/3)# exit
switch(config)# int ge1/4
switch(config-ge1/4)# isolate ge1/3
switch(config-ge1/4)# exit
switch(config)#
```

3.7 端口统计

3.7.1 查看端口统计

在特权视图下，使用下面的命令，清除端口统计。

命令	说明
show interface counters	查看全部端口的统计信息

show interface counters detail *[IFNAME]*	查看端口详细的统计信息

#示例

```
switch# show interface counters

Interface      InOctets    InUcastPkts  InMcastPkts  InBcastPkts

ge1/1          888         6            0            0
ge1/2          0           0            0            0
ge1/3          0           0            0            0
ge1/4          0           0            0            0
ge1/5          0           0            0            0
ge1/6          0           0            0            0
ge1/7          0           0            0            0
ge1/8          0           0            0            0
ge1/9          0           0            0            0
ge1/10         0           0            0            0
ge1/11         0           0            0            0
ge1/12         0           0            0            0
ge1/13         0           0            0            0
ge1/14         0           0            0            0
ge1/15         0           0            0            0
ge1/16         0           0            0            0
ge1/17         0           0            0            0
ge1/18         0           0            0            0
ge1/19         0           0            0            0
ge1/20         0           0            0            0
ge1/21         0           0            0            0
ge1/22         0           0            0            0
ge1/23         0           0            0            0
ge1/24         0           0            0            0
ge1/25         0           0            0            0
ge1/26         0           0            0            0
ge1/27         0           0            0            0
ge1/28         0           0            0            0

switch#
```

3.7.2 清除端口统计

在特权视图下，使用下面的命令，清除端口统计。

命令	说明
clear counters *[IFNAME]*	清除端口统计。

```
#示例
switch# clear  counters  ge1/1
switch#
```

4 业务管理

4.1 VLAN 配置

4.1.1 VLAN 简介

VLAN(Virtual Local Area Network)，即虚拟局域网，是一种通过将局域网内的设备逻辑地而不是物理地划分的交换网络。IEEE 于 1999 年颁布了用以标准化 VLAN 实现方案的 IEEE 802.1Q 协议标准草案。VLAN 技术允许将一个物理的 LAN 逻辑地划分成不同的广播域（VLAN），每一个 VLAN 都包含一组有着相同需求的设备，与物理上形成的 LAN 有着相同的属性。

由于它是从逻辑上划分，而不是从物理上划分，所以同一个 VLAN 内的各个工作站没有限制在同一个物理范围中，即这些工作站可以在不同物理 LAN 网段。由 VLAN 的特点可知，一个 VLAN 内部的广播和单播流量都不会转发到其他 VLAN 中，从而有助于控制流量、减少设备投资、简化网络管理、提高网络的安全性。

- 支持基于端口的 VLAN
- 端口支持 802.1Q 的中继模式
- 支持访问型端口

基于端口的 VLAN，就是将端口归属到交换机支持的 VLAN 的一个子集中。如果这个 VLAN 子集只有一个 VLAN，那么该端口就是访问模式（access）端口；如果这个 VLAN 子集中有多个 VLAN，该端口为中继（trunk）端口，其中有一个默认的 VLAN，它是该端口的 native VLAN，该 VLAN ID 就是 Port VLAN ID（PVID）。

4.1.2 VLAN 配置

4.1.2.1 创建/删除 VLAN

VLAN 是一个按不同功能、不同项目组或不同的应用进行逻辑划分的交换网络，不区分这些用户的物理位置。 VLAN 拥有与物理局域网相类似的属性，利用 VLAN 可以将不在同一物理局域网的终端划分到一个 VLAN 组中。一个 VLAN 可以拥有多个端口，而所有的单播、组播、广播数据报文只能在同一个 vlan 中进行转发、扩散到终端。每一个 VLAN 就是一个逻辑上的网络，一个数据报文要到达另外一个 VLAN，则必须通过路由或桥转发。

使用下面的命令进行 vlan 的配置：

命令	说明
vlan *〈1-4094〉*	进入到 VLAN 的配置模式

name string	vlan 别名

Vlan 也可以通过 VLAN 管理协议 GVRP 进行动态的添加删除。

#示例

```
switch(config)# vlan 5
switch(config-vlan5)#
```

4.1.2.2 配置 PVID

每一个端口都有一个默认的 VLAN，即 PVID，端口下接收到的所有的没有 VLAN 标签的数据都是属于该 VLAN 的数据报文。

默认 pvid 为 1。

命令	说明
switchport pvid *<1-4094>*	配置交换机端口的 PVID。

#以下将端口 ge1/10 的 pvid 设置为 2

```
switch(config)# int ge1/10
switch(config-ge1/10)# switchport  pvid  2
switch(config-ge1/10)#
```

4.1.2.3 端口添加 VLAN

中继模式可以将端口归属于多个 VLAN，同时也可以配置转发何种报文和所属的 VLAN 的数量，即端口发送的报文是 Tagged 还是 unTagged，以及端口所属的 VLAN list。

使用如下的命令可以配置交换机端口所属 vlan：

命令	说明
switchport trunk (tag | untag) *vlan-list*	配置交换机端口 VLAN 范围。

#以下示例，将端口 ge1/35 配置成 trunk 模式，并将该端口加入到 vlan 2,4,9-10,12

```
switch(config)# interface ge1/35
switch(config-ge1/35)# switchport  trunk  tag  2 4 9-10 12
switch(config-ge1/35)#
```

4.1.2.4 端口过滤模式设置

端口过滤模式可以将不符合要求的以太网帧过滤掉。

使用如下的命令可以配置交换机端口过滤模式属性：

命令	说明
switchport vlan-filter (both | egress | ingress | none)	配置交换机端口过滤模式，默认 egress

#以下示例，将端口 ge1/2 配置成 ingress 模式.

```
switch# con t
switch(config)# int ge1/2
switch(config-ge1/2)# switchport vlan-filter ingress
```

```
switch(config-ge1/2)# ex
switch(config)#
```

4.1.2.5 端口入口丢弃模式设置

端口丢弃模式可以将不符合要求的以太网帧丢弃掉。

使用如下的命令可以配置交换机端口过滤模式属性：

命令	说明
switchport discard (all | none | tag | untag)	配置交换机端口 vlan 丢弃过模式，默认 none

#以下示例，将端口 ge1/2 配置成丢弃 tag 模式.

```
switch# con t
switch(config)# int ge1/2
switch(config-ge1/2)# switchport discard tag
switch(config-ge1/2)# ex
switch(config)#
```

4.1.2.6 基于 MAC 的 VLAN 配置

使用如下的命令可以配置交换机端口过滤模式属性：

命令	说明
mac-vlan mac-address *value1*	设置基于 MAC 的 VLAN

#以下示例，将 MAC 地址：00:01:02:03:04:06 绑定在 VLAN2。

```
switch(config)# vlan 2
switch(config-vlan2)# mac-vlan mac-address  00:01:02:03:04:06
switch(config-vlan2)#
```

4.1.2.7 基于协议的 VLAN 配置

使用如下的命令可以配置交换机端口过滤模式属性：

命令	说明
protocol-vlan frame-type (802d3 | ether2 | llc) ether-type (802d1q | 802d1x | arp | ip | ipv6) vid *value1*	设置基于协议的 VLAN

#以下示例，将 IP 协议绑定在 VLAN2。

```
switch(config)# vlan 2
switch(config-ge1/2)# protocol-vlan frame-type ether2 ether-type ip vid 2
switch(config-ge1/2)#
```

4.1.2.8 VLAN 监控与维护

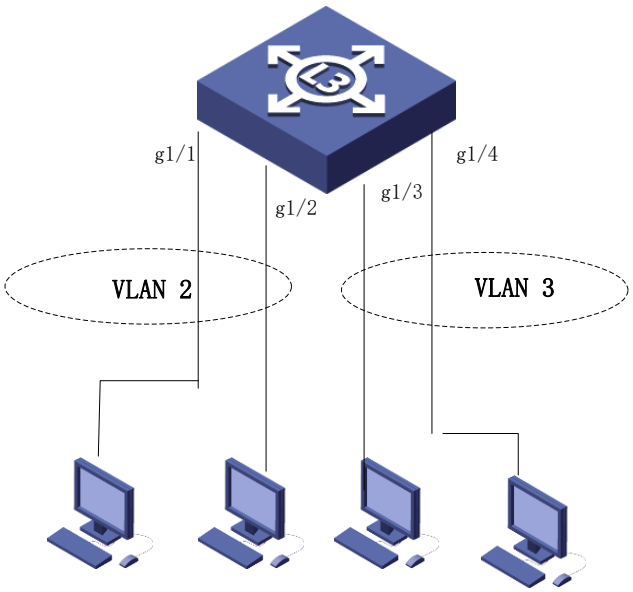
为了监控 VLAN 的配置和状态，可以在所有视图下使用下面的命令：

命令	说明
show vlan *[id]*	显示 VLAN。

#以下示例，显示所有 vlan

```
switch(config)# show  vlan
Vid  Status Name      Ports
-------------------------------------
1     static vlan1        ge1/1 ge1/2 ge1/3 ge1/4 ge1/5 ge1/6 ge1/7 ge1/8
ge1/9 ge1/10 ge1/11 ge1/12 ge1/13 ge1/14 ge1/15 ge1/16 ge1/17 ge1/18 ge1/19
ge1/20 ge1/21 ge1/22 ge1/23 ge1/24 ge1/25 ge1/26 ge1/27 ge1/28 ge1/29 ge1/30
ge1/31 ge1/32 ge1/33 ge1/34 ge1/35 ge1/36 ge1/37 ge1/38 ge1/39 ge1/40 ge1/41
ge1/42 ge1/43 ge1/44 ge1/45 ge1/46 ge1/47 ge1/48 xe1/49 xe1/50 xe1/51 xe1/52
2    static vlan2     ge1/2
4    static vlan4     ge1/2
9    static vlan9     ge1/2
10   static vlan10    ge1/2
12   static vlan12    ge1/2
switch(config)#
```

4.1.3 配置示例



#如图，ge1/1 和 ge1/2 加入 vlan2，ge1/3 和 ge1/4 加入 vlan3

```
switch(config)# int ge1/1
switch(config-ge1/1)# switchport  pvid  2
switch(config-ge1/1)# ex
```

```
switch(config)# int ge1/2
switch(config-ge1/2)# switchport  pvid  2
switch(config-ge1/2)# ex
switch(config)# int ge1/3
switch(config-ge1/3)# switchport  pvid  3
switch(config-ge1/3)# ex
switch(config)# int ge1/4
switch(config-ge1/4)# switchport  pvid  4
switch(config-ge1/4)#
```

#Vlan 2 内的主机能互相 ping 通，vlan2 内主机无法 ping 通 vlan3 内主机，说明已实现 vlan 隔离

4.2 MAC 配置

4.2.1 简介

4.2.1.1 概述

MAC(Medium/Media Access Control)地址，用来表示互联网上每一个站点的标识符，采用十六进制数表示，共六个字节（48 位）。其中，前三个字节是由 IEEE 的注册管理机构 RA 负责给不同厂家分配的代码(高位 24 位),也称为“编制上唯一的标识符”(Organizationally Unique Identifier)，后三个字节(低位 24 位)由各厂家自行指派给生产的适配器接口，称为扩展标识符（唯一性）。一个地址块可以生成 2^24 个不同的地址。

路由器端口所连的各个工作站或服务器都有自己唯一的 MAC 地址，为此，对应路由器的每个端口都有一个 MAC 地址表，包含该端口下所有设备的 MAC 地址。 一般情况下，路由器根据所连设备，通过源地址学习自动建立 MAC 地址表。为了提高端口安全性，网络管理员可手工在表中加入特定的 MAC 和端口的对应关系，将设备与端口绑定，防止非法用户使用假冒身份通过网络中其它交换机接入。

4.2.1.2 MAC 地址分类

MAC 地址表的表项分为动态、静态。

- 动态表项：由设备学习存储。表项会被老化。系统复位，表项将丢失。
- 静态表项：由用户配置。表项不被老化。系统复位，表项不会丢失。

4.2.2 MAC 地址配置

4.2.2.1 配置静态 MAC 地址

静态 MAC 地址表项指的是那些不能够被交换机老化掉的 MAC 地址表项，它只能被人工的删除掉。根据交换机使用过程中的实际需要，可以添加和删除静态 MAC 地址。 进入全局配置视图下，按下列步骤来添加和删除一个静态 MAC 地址。

命令	说明
mac-address static *mac-addr* **vlan** *<1-4094>* **interface** *IFNAME*	添加一个静态MAC地址表项。
no mac address-table static *mac-addr* **vlan** *vlan-id*	删除一个静态MAC地址表项。

#配置示例

```
switch(config)# mac-address static 0000-1111-2222 vlan 1 interface  ge1/2
switch(config)#
```

4.2.2.2 配置 MAC 地址老化时间

当一个动态的 MAC 地址在指定的老化时间内没有被使用时，交换机/路由器将把该 MAC 地址从 MAC 地址表中删除。

MAC 地址的老化时间可以根据需要进行配置，默认的老化时间为 300 秒。

进入全局配置视图下，按下列步骤来配置 MAC 地址的老化时间

命令	说明
mac-address aging-time <10-1000000>	配置 MAC 老化时间。
no mac-address aging-time	恢复 MAC 老化时间为默认值。

#以下配置 mac 地址老化时间为 600 秒

```
switch# con t
switch(config)# mac-address  aging-time  600
switch(config)#
```

4.2.2.3 清除动态 MAC 地址

某些情况下需要清除掉设备已经学习到的 MAC 地址。

进入特权视图下使用下面的命令删除地址

命令	说明
clear mac-address *[interface interface-id /static /vlan vlan-id]*	清除 mac 地址

#示例：

```
switch#
switch# clear mac-address interface ge1/40
switch#
```

4.2.2.4 端口的 MAC 地址限制

某些情况下需要限制端口学习的 MAC 地址最大数量。

在接口视图下，使用下面的命令限制 MAC 地址数量

命令	说明
mac-limit *count* **(broadcast | cpu | drop)**	配置限制 MAC 地址数量
no mac-limit	不限制 MAC 地址数量

#端口 g1/36 学习到的 MAC 地址数量，最多为 100

```
switch(config)# interface ge1/36
switch(config-ge1/36)# mac-limit 100
switch(config-ge1/36)#
```

4.2.2.5 vlan 的 MAC 地址限制

某些情况下需要限制 vlan 学习的 MAC 地址最大数量。

在 vlan 视图下，使用下面的命令限制 MAC 地址数量

命令	说明
mac-limit *count* **(broadcast | cpu | drop)**	配置限制 MAC 地址数量
no mac-limit	不限制 MAC 地址数量

#端口 vlan 2 学习到的 MAC 地址数量，最多为 100，未知 MAC 将被丢弃

```
switch(config)# vlan 2
switch(config-vlan2)# mac-limit 100 drop
switch(config-vlan2)#
```

4.2.3 调试和维护 MAC 地址表

在使用交换机的过程中，由于调试或管理的需要，我们想要知道交换机 MAC 地址表的信息。通过 show 命令可以把交换机 MAC 地址表的内容显示出来。

命令	说明
show mac-address [dynamic | interface *IFNAME* **| vlan** *vlan-id* **| static]**	显示 mac 地址

#以下显示 MAC 地址信息

```
switch# show  mac-address  static
Mac            Vid  Interface         Type
-----------------------------------------
0000-1111-2222  1    ge1/2             static
total: 1
switch#
```

4.3 MSTP 配置

4.3.1 MSTP 简介

4.3.1.1 STP

交换机的端口在 STP 环境中共有 5 种状态：阻塞 blocking、监听 listening、学习 learning、转发 forwarding、关闭(disable)。

- Blocking:处于这个状态的端口不能够参与转发数据报文,但是可以接收配置消息,并交给 CPU 进行处理。 不过不能发送配置消息，也不进行地址学习。
- Listening：处于这个状态的端口也不参与数据转发，不进行地址学习；但是可以接收并发送配置消息。
- Learning：处于这个状态的端口同样不能转发数据，但是开始地址学习，并可以接收、处理和发送配置消息。
- Forwarding：一旦端口进入该状态，就可以转发任何数据了，同时也进行地址学习和配置消息的接收、处理和发送。

交换机上一个原来被阻塞掉的端口由于在最大老化时间内没有收到 BPDU，从阻塞状态转变为倾听状态，倾听状态经过一个转发延迟（15 秒）到达学习状态，经过一个转发延迟时间的 MAC 地址学习过程后进入转发状态。

如果到达倾听状态后发现本端口在新的生成树中不应该由此端口转发数据则直接回到阻塞状态。

当拓扑发生变化，新的配置消息要经过一定的时延才能传播到整个网络，这个时延称为转发延迟（Forward Delay），协议默认值是 15 秒。

在所有网桥收到这个变化的消息之前,若旧拓扑结构中处于转发的端口还没有发现自己应该在新的拓扑中停止转发，则可能存在临时环路。为了解决临时环路的问题，生成树使用了一种定时器策略，即在端口从阻塞状态到转发状态中间加上一个只学习 MAC 地址但不参与转发的中间状态，两次状态切换的时间长度都是 Forward Delay，这样就可以保证在拓扑变化的时候不会产生临时环路

4.3.1.2 RSTP

1、RSTP 端口状态

STP 定义了 5 种不同的端口状态,关闭(disable),监听(Listening),学习(Learning),阻断（Blocking）和转发（Forwarding），其端口状态表现为在网络拓扑中端口状态混合（阻断或转发），在拓扑中的角色（根端口、指定端口等等）。在操作上看，阻断状态和监听状态没有区别，都是丢弃数据帧而且不学习 MAC 地址，在转发状态下，无法知道该端口是根端口还是指定端口。

在 RSTP 中只有三种端口状态，Discarding、Learning 和 Forwarding。802.1D 中的禁止端口，监听端口，阻塞端口在 802.1W 中统一合并为禁止端口。

2、RSTP 有五种端口类型

根端口和指定端口这两个角色在 RSTP 中被保留，阻断端口分成备份和替换端口角色。生成树算法(STA)使用 BPDU 来决定端口的角色，端口类型也是通过比较端口中保存的 BPDUB 来确定哪个比其他的更优先。

- 根端口。非根桥收到最优的 BPDU 配置信息的端口为根端口，即到根桥开销最小的端口，这点和 STP 一样。请注意图 8-16 上方的交换机，根桥没有根端口。按照 STP 的选择根端口的原则，SW-1 和 SW-2 和根连接的端口为根端口。
- 指定端口。与 STP 一样，每个以太网网段段内必须有一个指定端口。假设 SW-1 的 BID 比 SW-2 优先，而且 SW-1 的 P1 口端口 ID 比 P2 优先级高，那么 P1 为指定端口。
- 替换端口。如果一个端口收到另外一个网桥的更好的 BPDU，但不是最好的，那么这个端口成为替换端口。
- 备份端口。如果一个端口收到同一个网桥的更好 BPDU，那么这个端口成为备份端。当两个端口被一个点到点链路的一个环路连在一起时，或者当一个交换机有两个或多个到共享局域网段的连接时，一个备份端口才能存在。

禁用端口。在快速生成树协议应用的网络运行中不担当任何角色

3、RSTP 改进

- STP 没有明确区分端口状态与端口角色，收敛时主要依赖于端口状态的切换。RSTP 比较明确的区分了端口状态与端口角色，且其收敛时更多的是依赖于端口角色的切换。
- STP 端口状态的切换必须被动的等待时间的超时。而 RSTP 端口状态的切换却是一种主动的协商。
- STP 中的非根网桥只能被动的中继 BPDU。而 RSTP 中的非根网桥对 BPDU 的中继具有一定的主动性

4.3.1.3 MSTP 概述

MSTP（Multiple Spanning Tree Protocol）多生成树协议，用来在桥接局域网中建立简单而完整的拓扑结构。MSTP 可以与早期的 STP（Spanning Tree Protocol）和 RSTP（Rapid Spanning Tree Protocol）相兼容。

STP 和 RSTP 都只能在网络中建立单独的生成树拓扑，所有 VLAN 的报文沿着唯一的生成树进行转发。STP 收敛过慢，RSTP 通过握手机制保证网络拓扑快速稳定。

MSTP 继承了 RSTP 的快速握手机制，同时，在保证网络拓扑快速建立的基础上，MSTP 允许将不同的 VLAN 划分到不同的生成树中去，从而在网络中建立多个树状拓扑。在 MSTP 建立的网络中，属于不同 VLAN 的帧可以在不同的路径上转发，实现了 VLAN 数据的负载均衡。

与按 VLAN 划分 STP（per-VLAN Spanning Tree，PVST）不同的是，MSTP 还允许将多个 VLAN 划分到同一个生成树拓扑中去，这样可以有效减少支持大量 VLAN 所需的生成树的数目。

4.3.1.4 MSTP 工作原理

RSTP 和 MSTP 都能够与传统生成树协议互操作。但是，当与传统网桥交互时，IEEE 802.1w 的快速融合优势就会失去。为保留与基于 IEEE 802.1d 网桥的向后兼容性，IEEE 802.1s 协议网桥在其端口上接听 IEEE 802.1d 格式的 BPDU（网桥协议数据单元）。如果收到了 IEEE

802.1d BPDU，端口会采用标准 IEEE 802.1d 行为，以确保兼容性。

IEEE802.1s 引入了 IST（Single Spanning Tree，单生成树）概念和 MSTP 实例。IST 是一种 RSTP 实例，它扩展了 MSTP 区域内的 802.1D 单一生成树。IST 连接所有 MSTP 网桥，并从边界端口发出、作为贯穿整个网桥域的虚拟网桥。MSTP 实例（MSTI）是一种仅存在于区域内部的 RSTP 实例。它可以默认运行 RSTP，无须额外配置。不同于 IST 的是，MSTI 在区域外既不与 BPDU 交互，也不发送 BPDU。MSTP 可以与传统的 PVST+交换机互操作。

采用 MSTP 技术后，可以建立多个生成树，关联 VLANs 到相关的生成树进程，而且每个生成树进程具有独立于其他进程的拓扑结构。MSTP 还提供了多个数据转发路径和负载均衡，提高了网络容错能力，因为一个进程（转发路径）的故障不会影响其他进程（转发路径）。

每台运行 MSTP 的交换机都拥有单一配置，包括一个字母数字式配置名、一个配置修订号和一个 4096 部件表，与潜在支持某个实例的各 4096 VLAN 相关联。作为公共 MSTP 区域的一部分，一组交换机必须共享相同的配置属性。重要的是要记住，配置属性不同的交换机会被视为位于不同的区域。

在大型网络的不同网络部分，通过 MSTP 来定位不同 VLANs 和生成树进程的分配可以更容易地管理网络和使用冗余路径；一个生成树进程只能存在于具有一致的 VLAN 进程分配的桥中，必须用同样的 MSTP 配置信息来配置一组桥，这使得这些桥能参与到一组生成树进程中，具有同样的 MSTP 配置信息的互连的桥构成多生成树（MST）区。

为确保一致的 VLAN 实例映射，协议需要识别区域的边界。因此，区域的特征都包括在 BPDU 中。交换机必须了解它们是否像邻居一样位于同一区域，因此会发送一份 VLAN 实例映射表摘要，以及修订号和名称。当交换机接收到 BPDU 后，它会提取摘要，并将其与自身的计算结果进行比较。为避免出现生成树环路，如果两台交换机在 BPDU 中所接收的参数不一致，负责接收 BPDU 的端口就会被宣布为边界端口

与 STP 和 RSTP 不同，MSTP 协议不使用 BPDU 配置消息中的消息生存期（Message Age）和最大生存期（Max Age）来计算网络拓扑，而是使用了跳数属性（Hop Count）。为了防止旧的信息在网络中无休止的循环而影响新信息的传输，MSTP 使传输的信息在每个生成树中都与一个跳数属性相关联。BPDU 的跳数属性由 CIST 区域内根桥或者 MSTI 区域内根桥指定，并在每个接收端口处被减小。如果跳数在端口处变为了 0，该信息会被丢弃同时该端口会成为一个指派端口。

4.3.1.5 MSTP 区域

MSTP 中，VLAN 与生成树的对应关系是通过一个多生成树配置表来描述的。多生成树配置表，连同一个配置名称和一个配置修订号，共同构成了多生成树配置标识（MST Configuration Identifier）。

在网络上，相互连接且具有相同多生成树配置标识的网桥被认为是在同一个多生成树区域（MST Region）中。同一个多生成树区域中的网桥通常也具有相同的 VLAN 配置，从而保证这些 VLAN 的帧只在区域内部流动。

- CIST，Common and Internal Spanning Tree，公共与内部生成树。是指网络中所有单个交换机及其连接的局域网构成的生成树。这些交换机可能分属不同的多生成树区域，也可能是运行传统 STP 或者 RSTP 协议的交换机，运行这两种协议的交换机在多生成树网络中被认为是处在仅由其自身组成的区域中。网络拓扑稳定后，整个 CIST 会选出一个 CIST 根桥。每个区域内部也会选出 CIST 区域内根桥，作为从区域内部到达 CIST 根的最短路径。

- CST，Common Spanning Tree，公共生成树。如果把每个多生成树区域看作是一个单独的交换机，CST 就是连接着所有这些“单独交换机”的生成树。
- IST，Internal Spanning Tree，内部生成树。是指 CIST 在某个多生成树区域以内的部分。也可以理解为 IST 与 CST 共同构成了 CIST。
- MSTI，Multiple Spanning Tree Instance，多生成树实例。MSTP 协议允许将不同的 VLAN 划分到不同的生成树中，从而就建立起多个生成树实例。通常情况下，编号为 0 的生成树实例是指 CIST，它可以扩展到整个网络，而从 1 开始所指的生成树实例，都处在某个区域的内部。每个生成树实例中都可以被分配多个 VLAN，初始情况下，所有的 VLAN 都被分配在 CIST 中。多生成树区域中所有的 MSTI 都是相互独立的，它们可以选出不同的交换机作为各自的根。

4.3.1.6 MSTP 端口角色

MSTP 协议具有与 RSTP 相似的端口角色分配。

- 根端口（Root Port）

根端口表示当前交换机到网络根桥的路径，该路径具有最小的根路径开销。

- 预备端口（Alternate Port）

预备端口作为当前交换机到网络根桥路径的备份，当根端口连接失效时，预备端口可以立即转为新的根端口开始工作。

- 指派端口（Designated Port）

指派端口可以连接着下游的交换机或者局域网，作为该局域网到达网络根桥的路径。

- 备份端口（Backup Port）

当交换机的两个端口直接相连或者连接到同一个局域网时，优先级较低的端口会成为备份端口（较高的成为指派端口）。如果指派端口失效，则备份端口转为指派端口开始工作。

- Master 端口

Master 端口作为多生成树区域连接 CIST 根桥的最短路径。Master 端口也就是 CIST 区域内根桥的根端口。

- 边界端口（Boundary Port）

边界端口的概念在 CIST 中与在每个 MSTI 中稍有不同。在 CIST 中，边界端口表示连接着另一个多生成树区域的端口；而在 MSTI 中，边界端口角色表示该生成树实例在这个端口处不再扩展。

- 边缘端口（Edge Port）

在 RSTP 和 MSTP 协议中，边缘端口表示直接连接到网络主机的端口，这些端口不需要经过等待既可以进入转发状态，且不会在网络上造成环路。在初始情况下，MSTP（包括 RSTP）协议会认为所有的端口都是边缘端口，从而可以保证网络拓扑的快速建立。此时如果一个端口收到了来自其它交换机的 BPDU，该端口就会从边缘状态恢复为普通状态，如果收到的是 802.1D STP BPDU，那么该端口需要等待 2 倍的 Forward Delay 时间才能进入转发。

4.3.1.7 MSTP BPDU

与 STP 和 RSTP 协议相同，运行 MSTP 协议的交换机之间通过 BPDU（Bridge Protocol Data Unit）交互信息，CIST 以及所有 MSTI 中的配置信息都可以由 BPDU 携带。表 2.1 和表 2.2 列出了 MSTP 协议使用的 BPDU 结构。

字段名	字节数
Protocol Identifier	1 – 2
Protocol Version Identifier	3
BPDU Type	4
CIST Flags	5
CIST Root Identifier	6 – 13
CIST External Root Path Cost	14 – 17
CIST Regional Root Identifier	18 – 25
CIST Port Identifier	26 – 27
Message Age	28 – 29
Max Age	30 – 31
Hello Time	32 – 33
Forward Delay	34 – 35
Version 1 Length	36
Version 3 Length	37 – 38
Format Selector	39
Configuration Name	40 – 71
Revision	72 – 73
Configuration Digest	74 – 89
CIST Internal Root Path Cost	90 – 93
CIST Bridge Identifier	94 – 101
CIST Remaining Hops	102
MSTI Configuration Messages	103 ~

4.3.1.8 MSTP 稳定状态

MSTP 交换机根据接收到的 BPDU 执行计算和比较操作，最终可以使网络达到如下的稳定状态：

- 一台交换机被选为整个网络的 CIST 根（CIST Root）；

- 每个交换机和局域网段都会决定出到 CIST 根的具有最小开销的路径，以保证连接的完整性并防止环路；
- 每个区域内部都会选出一台交换机作为 CIST 区域内根（CIST Regional Root），该交换机具有到达 CIST 根的开销最小的路径；
- 每个 MSTI 都会独立的选择出一台交换机作为 MSTI 区域内根；
- 区域内部的每台交换机和局域网段都会确定出到达所在 MSTI 的根的开销最小的路径；
- CIST 根端口（Root Port）提供经过 CIST 区域内根（如果该交换机不是 CIST 区域内根）到达 CIST 根（如果该交换机不是 CIST 根）的具有最小开销的路径；
- CIST 指派端口（Designated Port）为所连接的局域网提供到达 CIST 根的最小开销路径；
- Alternate 和 Backup 端口在交换机、端口或局域网失效或被移除时提供连接；
- MSTI 根端口（Root Port）提供到达 MSTI 区域内根的最小开销路径（如果该交换机不是 MSTI 区域内根桥）；
- MSTI 指派端口（Designated Port）为所连接局域网提供到达 MSTI 区域内根的最小开销路径；

一个主端口（Master Port）提供区域与区域外 CIST 根桥的连接。在区域内部，CIST 区域内根桥的 CIST 根端口会作为区域内所有 MSTI 的 Master 端口。

4.3.1.9 MSTP 兼容性

MSTP 协议允许交换机通过一种协议转换机制与传统的 STP 交换机协同工作。如果交换机的一个端口接收到 STP 的配置信息，那么该端口就会转为仅发送 STP 报文。同时，接受到 STP 信息的端口也会被认为是一个边界端口。

请注意：一个端口转入 STP 兼容状态之后，即使不再接受到 STP 报文，该端口也不会自动恢复为 MSTP 状态。这种情况下，可以使用 spanning-tree mstp migration-check 命令清除端口学习到的生成树协议信息，使之恢复为 MSTP 状态。

运行 RSTP 协议的交换机可以识别并处理 MSTP 报文，因此与 RSTP 交换机协同工作时 MSTP 交换机不需要发生协议转换。

4.3.2 MSTP 全局配置

4.3.2.1 MSTP 默认配置

属性	默认设置
生成树协议模式	RSTP（PVST，SSTP和MSTP没有启动）。
区域名称	交换机MAC地址的字符串形式。
区域修订级别	0

多生成树配置表	所有VLAN都映射在CIST（MST00）中。
生成树优先级（CIST和所有MSTI）	32768
生成树端口优先级（CIST和所有MSTI）	128
生成树端口路径开销（CIST和所有MSTI）	1000 Mbps：20000 100 Mbps：200000 10 Mbps：2000000
Hello Time	2 秒
Forward Delay	15 秒
Maximum-aging Time	20 秒
最大跳数	20

#示例：

查看 STP 参数信息：

```
switch# sh spanning-tree instance
  bridge id      8.000.AA:6F:29:60:00:00
  designated root 8.000.AA:6F:29:60:00:00
  regional root   8.000.AA:6F:29:60:00:00
  root port      none 0
  path cost     0         internal path cost   0
  max age      20        bridge max age      20
  forward delay 15        bridge forward delay 15
  tx hold count 6         max hops            20
  hello time    2         ageing time         300
  force protocol version     mstp
  time since topology change 75690
  topology change count      0
  topology change           no
  topology change port       None
  last topology change port  None
switch#
```

4.3.2.2 启动和停止多生成树协议

生成树协议在默认情况下会以 MSTP 模式启动，当不需要运行 spanning-tree 时可以停止其运行。默认启用生成树，并且是 MSTP 模式

使用下面的命令将生成树协议设置为 MSTP 模式：

命令	说明
[no] spanning-tree	启动/禁用生成树协议。
spanning-tree mode (stp | rstp | mstp)	配置生成树模式。

#示例开启生成树

```
switch> enable
switch# configure t
switch# configure terminal
switch(config)# spanning-tree
switch(config)#

#示例关闭生成树
switch> enable
switch# configure t
switch# configure terminal
switch(config)# no spanning-tree
switch(config)#

#示例生成树模式选择 RSTP
switch> enable
switch# configure t
switch# configure terminal
switch(config)# spanning-tree mode rstp
switch(config)#
```

4.3.2.3 配置多生成树区域

交换机所处的多生成树区域，由配置名称、修订号以及 VLAN 与 MSTI 映射关系这三项属性决定，通过区域配置命令可以分别对其进行设置。需要注意的是，三项属性中任何一项的变化都会导致交换机所处区域的变化。

在初始情况下，多生成树配置名称等于交换机 MAC 地址的字符串形式，修订号为 0，并且所有的 VLAN 都被映射在 CIST（MST00）中。由于不同交换机的 MAC 地址都是不同的，因此初始情况下所有运行多生成树协议的交换机都是处在不同的区域中。通过执行 spanning-tree mstp instance instance-id vlan vlan-list 命令，可以创建一个新的 MSTI，并将指定的 VLAN 映射给它；如果该 MSTI 被删除，这些 VLAN 会被重新映射到 CIST 中。

使用下面的命令设置多生成树的区域信息：

命令	说明
spanning-tree mstp name *string*	设置多生成树配置名称。 string表示配置名称字符串，最多可包含32个字符，大小写敏感。默认值为交换机MAC地址的字符串形式。
no spanning-tree mstp name	设置多生成树配置名称为默认值。
spanning-tree mstp revision *value*	设置多生成树配置修订号。 value表示修订号，范围：0－65535，默认值0。
no spanning-tree mstp revision	设置多生成树修订号为默认值。

spanning-tree instance *instance-id* **vid** *vlan-list*	将VLAN映射到MSTI。 instance-id：生成树实例号，表示一个MSTI。范围1 – 15。 vlan-list：映射到该生成树的VLAN列表。范围 1 – 4094。
no spanning-tree instance *instance-id*	取消MSTI的VLAN映射，停止生成树实例。 instance-id：生成树实例号，范围1 – 15。

#示例：

设置生成树配置名称：

```
switch(config)# spanning-tree mstp name 3
switch(config)#
```

映射 VLAN 到 MSTI：

```
switch(config)# spanning-tree instance 1 vid 9
switch(config)#
```

4.3.2.4 配置网桥优先级

配置网桥优先级在某些情况下可以更直接的将交换机设置为网络的根，而不需通过 root 子命令。交换机在每个生成树实例中的优先级值是相互独立的，可以独立配置。

使用下面的命令设置生成树的优先级：

命令	说明
spanning-tree *instance-id* **priority** *value*	设置交换机的优先级值。 instance-id：生成树实例号，范围：0 – 15； value：网桥优先级，可为下列值之一： 0，4096，8192，12288，16384，20480，24576，28672,32768，36864，40960，45056, 49152, 53248, 57344, 61440。
no spanning-tree *instance-id* **priority**	恢复交换机的网桥优先级为默认值。 instance-id：生成树实例号，范围：0 – 15。

#示例：

设置交换机的优先级值：

```
switch# con t
switch(config)# spanning-tree instance 1 priority 8192
switch(config)#
```

4.3.2.5 配置生成树协议的时间参数

生成树协议的时间参数包括下列几项：

- Hello Time：交换机作为网络根桥时向指派端口发送配置信息的时间间隔；
- Forward Delay：STP 模式下，端口从 Blocking 状态到 Learning 状态，以及从 Learning 状态到 Forwarding 状态经历的时间；
- Max Age：生成树配置信息的最大生存期。

为了减少网络拓扑震荡，时间参数之间应该符合以下条件的要求：

- 2 x (fwd_delay - 1.0) >= max_age
- max_age >= (hello_time + 1) x 2

使用下面的命令配置多生成树协议的时间参数：

命令	说明
spanning-tree hello-time *seconds*	设置Hello Time参数。 seconds：范围：1 – 10秒，默认值2秒。
no spanning-tree hello-time	恢复Hello Time参数为默认值。
spanning-tree forward-time *seconds*	设置Forward Delay参数。 seconds：范围：4 – 30秒，默认值15秒。
no spanning-tree forward-time	恢复Forward Delay参数为默认值。
spanning-tree max-age *seconds*	设置Max Age参数。 seconds：范围：6 – 40秒，默认值20秒。
no spanning-tree max-age	恢复Max Age参数为默认值。

建议：通过设置根桥或者设置网络直径的方法来修改生成树协议的时间参数，以保证其合理性。配置时请注意控制台的提示。

#示例：

设置交换机的 Forward Delay 参数：

```
switch# con t
switch(config)# spanning-tree forward-delay 9
%Configured Bridge Times don't meet
 2 * (Bridge Foward Delay - 1 second) >= Bridge Max Age
switch(config)#
```

4.3.2.6 配置最大跳数

使用下面的命令配置最大跳数：

命令	说明
spanning-tree max-hops *hop-count*	设置最大跳数。 hop-count：范围：1 – 40，默认值为20。
no spanning-tree *hop-count*	恢复最大跳数为默认值。

#示例：

设置交换机的最大跳数：

```
switch# con t
switch(config)# spanning-tree max-hop 25
switch(config)#
```

4.3.3 MSTP 端口配置

4.3.3.1 配置端口优先级

如果交换机的两个端口之间形成环路，优先级较高的端口会进入 Forwarding 状态，较低的则被阻塞。如果所有端口的优先级都相同，那么端口号较小的端口将优先进入 Forwarding 状态。

在端口配置模式下，使用下面的命令设置多生成树协议端口的优先级：

命令	说明
spanning-tree mstp *instance-id* **port-priority** *priority*	设置端口优先级。 instance-id：生成树实例号，范围0 – 15； priority：端口优先级，为下列值之一： 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, 240。
no spanning-tree mstp *instance-id* **port-priority**	恢复端口优先级为默认值。

#示例：

设置交换机端口 45 的优先级：

```
switch# con t
switch(config)#
switch(config)# interface ge1/45
switch(config-ge1/45)# spanning-tree mstp 0 port-priority 64
switch(config-ge1/45)#
```

4.3.3.2 配置端口路径开销

MSTP 协议中端口路径开销的默认值是根据端口的连接速率计算出来的。如果两台交换机之间形成环路，路径开销较小的端口会进入 Forwarding 状态，路径开销越小表示端口的速率越高。如果所有端口的路径开销都相同，端口号较小的端口会优先进入转发状态。

在端口配置模式下，使用下面的命令设置端口的路径开销：

命令	说明
spanning-tree mstp *instance-id* **cost** *cost*	设置端口路径开销。 instance-id：生成树实例号，范围0 – 15； cost：路径开销值，范围1 – 200000000。
no spanning-tree mstp *instance-id* **cost**	恢复端口路径开销为默认值。

#示例：

设置交换机的端口 45 路径开销：

```
switch# con t
switch(config)#
switch(config)# interface ge1/45
switch(config-ge1/45)# spanning-tree mstp 0 cost 5000
switch(config-ge1/45)#
```

4.3.3.3 配置边缘端口

边缘端口表示该端口连接着网络上的终端设备，一个强制的边缘端口在 Link Up 之后会立刻进入转发状态。在端口配置模式下，通过下面的命令配置 MSTP 的边缘端口：

命令	说明
spanning-tree mstp edge force-true	强制端口为边缘端口。
spanning-tree mstp edge force-false	强制端口为非边缘端口。
spanning-tree mstp edge auto	自动检测边缘端口（默认）。

#示例：

设置交换机的端口 45 路径开销：

```
switch# con t
switch(config)#
switch(config)# interface ge1/45
switch(config-ge1/45)# spanning-tree mstp edge force-true
switch(config-ge1/45)#
```

4.3.3.4 配置端口连接类型

运行 MSTP 协议的交换机之间如果是点到点的直接连接，它们可以通过握手机制快速建立拓扑。配置端口连接类型功能允许将端口的连接设置为点到点。

在默认情况下，协议会根据端口的双工属性决定其是不是使用了点到点的连接。如果端口工作在全双工模式，协议就会认为它是点到点的连接；如果端口工作在半双工模式，协议会认为它的连接是共享的。

如果确认端口所连的交换机运行着 RSTP 或者 MSTP 协议，可以将端口的连接类型设置为点到点，保证快速握手的进行。

在端口配置模式下，使用下面的命令设置端口的连接类型：

命令	说明
spanning-tree mstp point-to-point force-true	设置端口连接方式为点到点。
spanning-tree mstp point-to-point force-false	设置端口连接方式为共享。
spanning-tree mstp point-to-point auto	自动检测端口的连接方式（默认）。

#示例：

设置交换机的端口 45 路径开销：

```
switch# con t
switch(config)#
switch(config)# interface ge1/45
switch(config-ge1/45)# spanning-tree mstp edge force-true
switch(config-ge1/45)#
```

4.3.3.5 重启协议转换检查

MSTP 协议允许交换机通过一种协议转换机制与传统的 STP 交换机或不同类型的 MST 交换机协同工作。如果交换机的一个端口接收到 STP 的配置信息，那么该端口就会转为仅发送 STP 报文。接受到 STP 信息的端口也会被认为是一个边界端口。同样，在 MST 兼容模式下，如果一个端口接收到兼容模式的 BPDU，该端口也会转为发送兼容模式的 BPDU。

注意：一个端口转入 STP 兼容状态或 MST 兼容状态之后，即使不再接收到相应模式的报文，该端口也不会自动恢复为 MSTP 状态。这种情况下，可以使用 spanning-tree mstp migration-check 命令清除端口学习到的生成树协议信息，使之恢复为 MSTP 状态。

运行 RSTP 协议的交换机可以识别并处理 MSTP 报文，因此与 RSTP 交换机协同工作时 MSTP 交换机不需要发生协议转换。

在端口配置模式下，使用下面的命令清除端口检测到的生成树协议信息：

命令	说明
spanning-tree mstp migration-check	清除端口检测到的生成树协议信息。

#示例：

```
switch(config-ge1/45)# spanning-tree mstp migration-check
switch(config-ge1/45)#
```

4.3.4 监视与维护 MSTP

4.3.4.1 查看多生成树协议信息

在监控模式、全局配置模式以及端口配置模式下，使用下面的命令查看多生成树协议的各种信息：

命令	说明
show spanning-tree	查看生成树协议信息。
show spanning-tree interface *[IFNAME* **|** **brief]**	查看生成树协议端口信息。
show spanning-tree instance *instance-id*	查看某一个多生成树实例信息。

#示例：

```
switch# sh spanning-tree interface ge1/45
CIST info
  enabled          no                  role              Disabled
  port id          4.02D               state              discarding
  external port cost 5000                admin external cost  5000
```

```
 internal port cost 200000000              admin internal cost  0
 designated root    8.000.AA:6F:29:60:00:00 dsgn external cost   0
 dsgn regional root 8.000.AA:6F:29:60:00:00 dsgn internal cost   0
 designated bridge  8.000.AA:6F:29:60:00:00 designated port     0.000
 admin edge port    no                      auto edge port      yes
 oper edge port     no                      topology change ack  no
 point-to-point     no                      admin point-to-point auto
 restricted role    no                      restricted TCN      no
 port hello time    2                       disputed            no
 bpdu guard port    no                      bpdu guard error    no
 network port       no                      BA inconsistent     no
 Num TX BPDU        0                       Num TX TCN          0
 Num RX BPDU        0                       Num RX TCN          0
 Num Transition FWD 0                        Num Transition BLK   1
switch#
```

4.3.4.2 调试多生成树协议信息

在全局视图视图下，使用下面的命令打印多生成树协议的各种信息：

命令	说明
debug mstp (all | event | interface | packet)	打开mstp调试开关

在出现 MSTP 运行故障时，使用 debug 命令对 MSTP 进行调试，查看调试信息，定位故障并分析故障原因

注意：打开调试开关将影响系统的性能。调试完毕后，应及时执行 no debug all 命令关闭调试开关。

#示例：

```
switch# debug mstp all
switch#
```

4.4 二层组播配置

4.4.1 简介

4.4.1.1 概述

IGMP（Internet Group Management Protocol）作为因特网组管理协议，是 TCP/IP 协议族中负责 IP 组播成员管理的协议，它用来在 IP 主机和与其直接相邻的组播路由器之间建立、维护组播组成员关系。

IGMP-snooping，是对 IGMP 报文的侦听，形成二层组播表项。IP 主机通过发送 IGMP 报文宣布加入某组播组；本地组播路由器通过周期性的发送 IGMP 报文轮询本地网络上的主机，

确定本地组播组成员信息。到目前为止，IGMP 有三个版本：IGMPv1 版本（由 RFC1112 定义）、IGMPv2 版本（由 RFC2236 定义）和 IGMPv3（由 RFC3376 定义）版本。

4.4.1.2 定义

- 路由器端口（router port）：交换机上连接组播路由器的端口，而不是指路由器设备上的端口。
- 组播成员端口：以太网交换机上与组播组成员相连的端口。此处的组播组成员是加入某个组播组的主机。
- mac 组播组：以太网交换机维护的以 mac 组播地址标识的组播组。

4.4.1.3 原理

当二层以太网交换机收到主机和路由器之间传递的 igmp 报文时，igmp snooping 分析 igmp 报文所带的信息。当监听到主机发出的 igmp 主机报告报文时，交换机就将该主机加入到相应的组播表中；当监听到主机发出的 igmp 离开报文时，交换机就将删除与该主机对应的组播表项。通过不断地监听 igmp 报文，交换机就可以在二层建立和维护 mac 组播地址表。之后，交换机就可以根据 mac 组播地址表转发从路由器下发的组播报文。

4.4.1.4 Igmp 报文

- Igmp report 报文：igmp 报告报文是主机向组播路由器发送的报告报文，用于申请加入某个组播组或者应答 igmp 查询报文。
- Igmp leave 报文：是组播组成员向组播路由器发送的报文，用于告知路由器主机离开了某个组播组
- Igmp query 报文：特定组查询报文是组播路由器向组播组成员发送的报文，用于查询特定组播组是否存在成员；igmp 通用查询报文是组播路由器向组播组成员发送的报文，用于查询哪些组播组存在成员。

4.4.2 配置 IGMP-snooping

4.4.2.1 使能 IGMP-snooping

默认 igmp-snooping 关闭，在全局视图下，通过以下命令开启

命令	说明
[no] igmp-snooping	启用igmp-snooping。

#配置示例

```
switch(config)# igmp-snooping
switch(config)#
```

4.4.2.2 配置主机老化时间

当一个端口加入组播组的时，组播组端口成员老化时间就是该定时器设置的时间。如果在此定时器超时后还没有收到 igmp 报告报文，那么以太网交换机就向该端口发送 igmp 特定组查询报文，如果还是接收不到 igmp 报告报文，则删除该表项。

在全局配置视图下，使用下面命令配置老化时间。

命令	说明
igmp-snooping host-age-time *〈200-1000〉*	配置主机老化时间。

#配置示例

```
switch(config)# igmp-snooping host-age-time 300
switch(config)#
```

4.4.2.3 添加基于 IP 的静态组播

某些情况下需要静态添加二层组播表项，在接口视图下，使用下面的命令。

命令	说明
igmp-snooping static-group *A.B.C.D* **[source** *A.B.C.D***] vlan** *〈1-4094〉*	添加静态组播

no 命令为删除。

#以下显示 MAC 地址信息

```
switch(config-ge1/2)# igmp-snooping  static-group  225.1.1.1 vlan  2
switch(config-ge1/2)#
```

4.4.2.4 添加基于 MAC 的组播静态组播

某些情况下需要静态添加二层组播表项，在接口视图下，使用下面的命令。

命令	说明
mcast static *MM:MM:MM:MM:MM:MM* **vlan** *〈1-4094〉* **interface** *[IFNAME]*	添加静态组播

no 命令为删除。

#以下显示 MAC 地址信息

```
switch(config)# mcast static 01:00:00:00:00:02 vlan 1 interface ge1/1
switch(config)#
```

4.4.2.5 丢弃未知组播

缺省情况下，对未知组播数据报文进行广播。在 vlan 视图下，使用下面的命令配置未知组播行为。

命令	说明
unknown-multicast drop	丢弃未知组播

unknown-multicast flood (all|unknown)	广播未知组播。 All：广播所有 Unknown：只广播未知组播

#在 vlan2 上，丢弃未知组播

```
switch(config)# vlan 2
switch(config-vlan2)# unknown-multicast  drop
switch(config-vlan2)# exit
switch(config)#
```

4.4.3 调试和维护 MAC 地址表

4.4.3.1 显示组播 MAC

在使用交换机的过程中，由于调试或管理的需要，我们想要知道交换机二层组播表的信息。

命令	说明
show igmp-snooping group [vlan *<1-4094>*]	显示 mac 地址

#以下显示组播 MAC 地址信息

```
switch# show igmp-snooping group

Vid  Source          Group           Interface  Type    Timeout(s)
2    0.0.0.0         225.1.1.1        ge1/45    static  -

switch#
```

4.4.3.2 调试 igmp-snooping

在全局视图视图下，使用下面的命令调试 igmp-snooping 信息：

命令	说明
debug igmp-snooping (all | event | interface | packet | timer)	打开igmps调试开关

在出现 igmp-snooping 运行故障时，使用 debug 命令对 IGMP-snooping 进行调试，查看调试信息，定位故障并分析故障原因。

```
switch# debug igmp-snooping all
switch#
```

4.5 QOS 配置

4.5.1 简介

4.5.1.1 概述

QoS（Quality of Service）服务质量，是网络的一种安全机制， 是用来解决网络延迟和阻塞等问题的一种技术。在正常情况下，如果网络只用于特定的无时间限制的应用系统，并不需要 QoS，比如 Web 应用，或 E-mail 设置等。但是对关键应用和多媒体应用就十分必要。当网络过载或拥塞时，QoS 能确保重要业务量不受延迟或丢弃，同时保证网络的高效运行。

4.5.1.2 服务模型

通常 QoS 提供以下三种服务模型：

1）Best-Effort service（尽力而为服务模型）

Best-Effort 服务模型是一个单一的服务模型，也是最简单的服务模型。对 Best-Effort 服务模型，网络尽最大的可能性来发送报文。但对延时、可靠性等性能不提供任何保证。

Best-Effort 服务模型是网络的缺省服务模型，通过 FIFO（first in first out 先入先出）队列来实现。它适用于绝大多数网络应用，如 FTP、E-Mail 等。

2）Integrated service（综合服务模型，简称 Int-Serv）

Int-Serv 服务模型 Int-Serv 是一个综合服务模型，它可以满足多种 QoS 需求。该模型使用资源预留协议（RSVP），RSVP 运行在从源端到目的端的每个设备上，可以监视每个流，以防止其消耗资源过多。这种体系能够明确区分并保证每一个业务流的服务质量，为网络提供最细粒度化的服务质量区分。

但是，Inter-Serv 模型对设备的要求很高，当网络中的数据流数量很大时，设备的存储和处理能力会遇到很大的压力。Inter-Serv 模型可扩展性很差，难以在 Internet 核心网络实施。

3）Differentiated service（区分服务模型，简称 Diff-Serv）

Diff-Serv 服务模型 Diff-Serv 是一个多服务模型，它可以满足不同的 QoS 需求。与 Int-Serv 不同，它不需要通知网络为每个业务预留资源。区分服务实现简单，扩展性较好。

4.1.1.3 流量管理

流分类、流量监管、流量整形、拥塞管理和拥塞避免是构造有区别地实施服务的基石，它们主要完成如下功能：

- 流分类： 依据一定的匹配规则识别出对象。 流分类是有区别地实施服务的前提。
- 流量监管：对进入路由器的特定流量的规格进行监管。当流量超出规格时，可以采取限制或惩罚措施，以保护运营商的商业利益和网络资源不受损害。

- 流量整形：一种主动调整流的输出速率的流控措施，通常是为了使流量适配下游路由器可供给的网络资源，避免不必要的报文丢弃和拥塞。
- 拥塞管理：网络拥塞时必须采取的解决资源竞争的措施。通常是将报文放入队列中缓存，并采取某种调度算法安排报文的转发次序。
- 拥塞避免：过度的拥塞会对网络资源造成损害。拥塞避免监督网络资源的使用情况，当发现拥塞有加剧的趋势时采取主动丢弃报文的策略，通过调整流量来解除网络的过载。

在这些流量管理技术中，流分类是基础，它依据一定的匹配规则识别出报文，是有区别地实施服务的前提；而流量监管、流量整形、拥塞管理和拥塞避免从不同方面对网络流量及其分配的资源实施控制，是有区别地提供服务思想的具体体现。

4.5.2 配置 QOS

4.5.2.1 配置缺省 CoS

配置缺省 CoS 值命令如下：

命令	说明
cos default *cos*	缺省 CoS 值，范围 0—7。

no 命令为恢复默认配置，默认的缺省 COS 值为 0。

#将 ge1/1 端口收到的无标签帧的 CoS 值设为 3：

```
switch(config)# int ge1/1
switch(config-ge1/1)# cos default 4
switch(config-ge1/1)#
```

4.5.2.2 配置 CoS 调度策略

CoS 优先级队列调度策略默认为 SP，在全局配置视图下，使用以下命令设置 CoS 优先级队列调度策略。

命令	说明
scheduler policy (sp | wrr)	设置调度策略。
no scheduler policy	恢复默认调度策略。

#设置调度模式为 wrr：

```
switch(config)# scheduler policy wrr 1 2 3 4 5 6 7 8
```

4.5.2.3 设置优先级队列

IEEE802.1p 定义的 CoS 值，0—7，CoS 优先级队列 0-7。默认映射如下表所示。

CoS 值	优先级队列
0	0
1	1

2	2
3	3
4	4
5	5
6	6
7	7

设置 CoS 优先级队列，使用如下命令。

命令	说明
cos map *queue cos1 .. cosn*	设置调度策略。

在全局配置模式下使用该命令，将影响所有端口 CoS 优先级队列；而在二层端口下面配置该命令，将只影响该端口的 CoS 优先级队列。

```
#将 CoS 0 映射到优先级队列 1，将 CoS 3 映射到优先级队列 2。
switch(config-ge1/1)# cos map 0 1
switch(config-ge1/1)# cos map 3 2
```

4.6 LLDP

4.6.1 简介

4.6.1.1 LLDP 概述

802.1AB 链接层发现协议（Link Layer Discovery Protocol），能够使企业网络的故障查找变得更加容易，并加强网络管理工具在多厂商环境中发现和保持精确网络拓扑结构的能力。它可使邻近设备向其他设备发出其状态信息的通知，并且所有设备的每个端口上都存储着定义自己的信息，如果需要还可以向与它们直接连接的近邻设备发送更新的信息，近邻的设备会将信息存储在标准的 SNMP MIBs。网络管理系统可从 MIB 处查询出当前第二层的连接情况。LLDP 不会配置也不会控制网络元素或流量，它只是报告第二层的配置。

简单说来，LLDP 是一种邻近发现协议。它为以太网网络设备，如交换机、路由器和无线局域网接入点定义了一种标准的方法，使其可以向网络中其他节点公告自身的存在，并保存各个邻近设备的发现信息。例如设备配置和设备识别等详细信息都可以用该协议进行公告。具体来说，LLDP 定义了一个通用公告信息集、一个传输公告的协议和一种用来存储所收到的公告信息的方法。要公告自身信息的设备可以将多条公告信息放在一个局域网数据包内传输，传输的形式为类型长度值（TLV）域。

LLDP 是单向协议，一个 LLDP 代理能够通过与之关联的 MSAP 发送自己系统状态和自身功能，也可接收邻接设备的当前系统状态和功能。但是，LLDP 代理不能通过此协议请求对方任何信息。LLDP 代理其发送和接收信息互不影响，可以只配置实现发送或接收功能，或两者都有。

4.6.1.2 协议初始化

本地 LLDP 代理可能配置为只接收帧，只发送帧，即接收又发送帧，所以对帧接收和发

送需要独立的协议初始化处理。在没有配置说明只接收或只发送的情况下，LLDP 代理默认为可接收可发送模型。

4.6.1.3 LLDP 发送模型初始化

在接口模式下配置接口是否为可发送帧模型。当配置为可发送帧模型时，在本地系统中一个或多个信息元素（管理对象）状态或值发生变化和发送计时器超时两种情况下，自动发送 LLPD 信息；当为不可发送帧时，接口不向邻居发送 LLDP 报文。

4.6.1.4 LLDP 接收模型初始化

在接口模式下配置接口是否为可接收帧模型。当配置为可接收帧模型时，能够接收周围邻居发送的 LLDP 报文，并将其中的 tlv 内容存入远端 MIB 中；当为不可接收帧时，接口接收到邻居发送 LLDP 报文后直接丢弃。

4.6.1.5 LLDP 报文结构说明

LLDP 应该按照顺序包含三个必须的 TLV，后面为一个或多个可选 TLV，最后为结束 TLV。

- 三个必须的 TLVs 应该在 LLDPDU 的开始依次出现，其顺序为：

Chassis ID TLV
Port ID TLV
Time To Live TLV

- 由网络管理选择可选的 TLV，顺序任意。
- 结束 TLV 应该为 LLDPDU 中最后一个 TLV。

4.6.2 LLDP 配置

4.6.2.1 禁止/启动 LLDP

默认情况下 LLDP 关闭，当需要运行 LLDP 时可以开启 lldp 让其运行。 开启 LLDP 功能后，本地端口定期向外发送 lldp 帧以通知对端本地端口的信息。

在全局配置视图下使用如下命令配置 LLDP：

命令	说明
lldp (enable | disable)	开启/关闭LLDP

注意：只有开启了 lldp 功能才能对接收到的 lldp 报文进行处理，否则 lldp 帧将被直接转发。

#配置示例

使能 LLDP：

```
switch(config)# lldp enable
switch(config)#
```

4.6.2.2 配置 holdtime

正常情况下，MIB 中存储的远端信息在老化前都会更新，但由于更新帧发送过程中可能丢失，引起 MIB 中信息老化。为了防止此情况，设置 TTL 值使在老化时间内，多次发送更新 LLDP 帧。可以通过更改交换机的 holdtime 来控制发送 lldp 报文的 ttl 超时时间。

在全局配置视图下使用下面的命令可以配置 lldp 的 holdtime：

命令	说明
lldp holdtime *time*	配置lldp的超时时间，取值范围：<0-65535>，默认为120秒

恢复超时时间的默认配置：

命令	说明
no lldp holdtime	将超时时间恢复成默认值，默认为120秒

注意：超时时间应该比 lldp 报文发送的间隔时间要长，这样才能够保证在收到下一个 lldp 帧时，前面的邻居信息没有老化丢失。

#配置 LLDP 的 holdtime：

```
switch(config)# lldp holdtime 100
switch(config)#
```

4.6.2.3 配置 timer

通过配置 lldp 的 timer 可以控制交换机发送报文的间隔时间.

在全局配置视图下使用下面的命令可以配置 lldp 的 timer：

命令	说明
lldp timer *time*	配置lldp帧发送间隔。取值范围：<5-65534>，默认为30秒

恢复发送间隔的默认值：

命令	说明
no lldp timer	恢复为默认的间隔时间，默认为30秒

#配置 LLDP 的发送周期：

```
switch(config)# lldp timer 580
switch(config)#
```

4.6.2.4 配置 reinit

在本地系统中一个或多个信息元素（管理对象）状态或值发生变化和发送计时器超时两种情况下，自动发送 LLPD 信息。由于单个信息变化即需要发送 LLDP 帧，可能连续的一系列信息改变触发许多 LLDP 帧发送，每个帧中只报告一个变化，为了避免这种情况，网络管理定义了两个连续发送 LLDP 帧间的等待时间。通过配置 lldp 的 reinit 可以控制连续两个

lldp 报文发送的间隔时间

在全局配置视图下使用下面的命令可以配置 lldp 的 reinit：

命令	说明
lldp reinit *time*	配置lldp连续发送报文的间隔时间。取值范围：<2-5>，默认为2秒

恢复 reinit 的默认值：

命令	说明
no lldp reinit	恢复为默认的连续报文发送间隔，默认为2秒

#配置 LLDP 的发送间隔：

```
switch(config)# lldp reinit 3
switch(config)#
```

4.6.2.5 配置选择需要发送的 tlv

通过配置 lldp 的 tlv-select 可以选择需要发送的 tlv 进行发送. 默认情况下全部 tlv 都被发送。

在全局配置视图下使用下面的命令可以配置添加要发送的 tlv：

命令	说明
[no] lldp tlv-select management-address	发送802.3组织自定义tlv，包含以下内容： a）物理层具有的比特率和通信模式(duplex)； b）目前的duplex，和设置的比特率； c） 表明设置是连接初始化阶段自动协商的结果还是手动强制行为。
[no] lldp tlv-select port-description	发送管理地址tlv，管理地址应该方便管理使用，一般为三层IP地址。
[no] lldp tlv-select system-capabilities	发送系统性能tlv，系统性能是指发送报文的系统是交换机/路由器或是其它。
[no] lldp tlv-select system-description	发送系统描述tlv，系统描述由数字字母组成的网络实体的文本描述。系统描述应该包括系统的全名，系统硬件类型的版本定义，软件操作系统，网络软件。
[no] lldp tlv-select system-name	发送系统名称tlv。系统名域为由数字字母组成的系统管理者指定的名称。系统名应该为系统管理者名称。即交换机名称。

#配置发送系统描述 tlv：

```
switch(config)# lldp tlv-select system-description
switch(config)#
```

4.6.2.6 发送 / 接收状态配置

Lldp 系统可工作在以下几种模式下：仅可发送模式，仅可接收模式和可发送可接收模式. 默认情况下为可发送接收模式，通过下面的命令可改变 lldp 的工作模式.

在接口视图下使用下面的命令配置 lldp 的工作模式为可发送接收模式：

命令	说明
[no] lldp transmit	配置端口为可发送lldp帧模式
[no] lldp receive	配置端口为可接收lldp帧模式

#配置发送系统描述 tlv：

```
switch(config)# interface ge1/42
switch(config-ge1/42)# no lldp receive
switch(config-ge1/42)#
```

4.6.3 LLDP 监控与维护

4.6.3.1 LLDP 查看

通过显示命令可以观察 lldp 模块接收到的邻居信息，各种统计和端口状态信息。

在全局视图下，使用下面显示命令：

命令	说明
show lldp interface *interface-name*	显示端口的状态信息，即发送和接收模式.
show lldp neighbors	显示接收邻居的简略信息
show lldp neighbors detail	显示接收邻居的详细信息
show lldp traffic	显示发送和接收的各种统计信息

#示例：

显示接受邻居的简略信息：

```
switch# sh lldp neighbors
Capability Codes:
  (R)Router,(B)Bridge,(C)DOCSIS Cable Device,(T)Telephone
  (W)WLAN Access Point,(P)Repeater,(S)Station,(O)Other

Device-ID                  Chassis-ID          Mgm-IP              Local-
Intf     Hldtme   Port-ID        Capability
                           MAC: 484d7eb6ebf0                       ge1/1
3601     HM~

switch#
```

4.6.3.2 LLDP 调试

使用下面的命令，可以显示 LLDP 调试信息

命令	说明
debug lldp all	监视LLDP所有信息
debug lldp events	监视LLDP事件
debug lldp interface	监视LLDP接口信息
debug lldp msap	监视LLDP msap信息
debug lldp packet	监视LLDP报文

#示例：

```
switch# debug lldp all
switch#
```

4.6.4 配置示例

#在 SWA 上使能 lldp：

```
SWA# con t
SWA(config)# lldp  enable
SWA(config)#
```

#在 SWB 上使能 lldp：

```
SWB# con t
SWB(config)# lldp  enable
SWB(config)#
```

4.7 DHCP Server 配置

4.7.1 简介

4.7.1.1 概述

DHCP（Dynamic Host Configuration Protocol）协议为 Internet 上的主机提供了部分

网络配置参数。DHCP 在 RFC 2131 中讲述。DHCP 最主要的一项功能是分配接口上的 IP 地址。DHCP 协议支持三种机制的 IP 地址分配机制：

- 自动分配

DHCP 服务器自动分配一个永久性的 IP 地址给某一客户端使用。

- 动态分配

DHCP 服务器分配一个 IP 地址给某一客户端使用一定的时间，或者直到该客户主动放弃该地址的使用权。

- 手工分配

DHCP 服务器管理员手工指定一个 IP 地址且通过 DHCP 协议传送给客户端使用。

4.7.1.2 DHCP 原理

DHCP 统一使用两个 IANA 分配的端口作为 BOOTP：服务器端使用 67/udp，客户端使用 68/udp。

DHCP 运行分为四个基本过程，分别为请求 IP 租约、提供 IP 租约、选择 IP 租约和确认 IP 租约。

客户在获得了一个 IP 地址以后，就可以发送一个 ARP 请求来避免由于 DHCP 服务器地址池重叠而引发的 IP 冲突

4.7.1.3 DHCP 的优点

DHCP 有几种应用。当存在以下需求时，可以使用 DHCP 协议：

如果需要为某一个以太网接口分配 IP 地址、网段及相关资源（如相应的网关），可以通过配置 DHCP 客户端来实现。

交换机上接有多个主机，而交换机能够访问到 DHCP 时，可以通过 DHCP 中继，从 DHCP 服务器上获得一个 IP 地址，将该地址再分配该主机。

该功能的使用可以提供以下优点：

- 减少配置时间
- 减少配置错误
- 通过 DHCP 服务器集中控制设备部分接口的 IP 地址

4.7.1.4 DHCP 术语

DHCP 协议本身是基于 Server/Client 结构的，所以在 DHCP 运行环境中，存在 DHCP-Server 和 DHCP-Client：

- DHCP-Server

用来发放、收回 DHCP 协议所涉及资源（如 IP 地址、租用时间等）的设备。

- DHCP-Client

从 DHCP-Server 处获取 IP 地址等信息，并且用于本地系统的设备。

如上所述，对于 DHCP 信息动态分配的过程中，存在租用时间的概念：某个 IP 地址资源从分配开始计时的一段有效期，在该段时间之后，相应的 IP 地址资源将被 DHCP-Server 收回，若要继续使用，DHCP-Client 需要重新申请。

4.7.3 配置 DHCP Server

4.7.3.1 打开 DHCP Server 服务

打开 DHCP Server 服务，为 DHCP Client 分配 IP 地址等参数，在全局配置视图下执行下列命令（此时，DHCP 服务器也支持 relay 操作，对于自身不能分配地址请求，配置了 ip helper-address 的端口将转发 DHCP 请求）：

命令	说明
dhcp-server (enable | disable)	打开/关闭DHCP Server服务

#示例，使能 DHCP Server 服务：

```
switch(config)# dhcp-server enable
switch(config)#
```

4.7.3.2 配置 DHCP Server 地址池

添加 DHCP Server 地址池，在全局配置视图下执行下列命令：

命令	说明
[no] dhcp-server pool *name*	添加DHCP Server地址池，并进入DHCP地址池配置态。

#示例，配置 DHCP Server 地址池：

```
switch(config)# dhcp-server pool 2
switch(config-dhcp-2)#
```

4.7.3.3 配置 DHCP Server 地址池参数

在 DHCP 地址池配置态下，可以执行以下命令来配置相关参数

命令	说明
network *ip-addr netsubnet*	配置用于自动分配的地址池的网络地址。
default-router *ip-addr*	配置分配给客户机的缺省路由。
dns-server *ip-addr*	配置分配给客户机的DNS服务器地址。
domain-name *name*	配置分配给客户机的域名。

lease (*days* [*hours*][*minutes*] | *infinite*)	配置分配给客户机的地址的时间期限。
netbios-name-server *ip-addr*	配置分配给客户机的netbios名字服务器地址。
static *ip-address hardware-address*	为mac地址为“hardware-address”的主机分配ip-address地址
port-bind *[IFNAME] ip-addr*	为端口绑定一个分配的IP

#配置示例

```
switch(config)#
switch(config)# dhcp-server  pool  aa
switch(config-dhcp)# network  11.1.1.0/24
switch(config-dhcp)# default-router  11.1.1.1
switch(config-dhcp)# dns-server  202.96.209.133
```

4.7.3.4 监视 DHCP Server

查看 DHCP Server 当前配置信息，在全局视图下执行下列命令：

命令	说明
show dhcp-server	显示DHCP Server信息

#配置示例：

```
switch# show dhcp-server

Dhcp server global is enable
-----------------------------------------------------
Dhcp pool 2
 network 11.1.1.0/24

switch#
```

5 路由管理

5.1 三层接口

5.1.1 配置主 IP 地址

一个接口只能拥有一个主 IP 地址。要配置网络接口的主 IP 地址和网络掩码，在接口配置态使用下列命令：

命令	说明
[no] ip address *A.B.C.D/M*	删除/配置接口的主 IP 地址。 A.B.C.D 为接口 IP

	M 为掩码，范围 1-32

#以下配置 vlanif2 的 IP 地址为 10.1.1.1/24

```
switch# con t
switch(config)# int vlanif2
switch(config-vlanif2)# ip address  10.1.1.1/24
switch(config-vlanif2)#
```

5.1.2 配置从 IP 地址

每个接口可以拥有多个 IP 地址，包括一个主 IP 地址和任意个从属 IP 地址。在以下几种情况下，需要配置从属 IP 地址：

当一个特定网段中没有足够的 IP 地址时。例如，某个逻辑子网中最多只有 254 个有效 IP 地址，但是需要在实际的物理网络中连接 300 台主机。在路由交换机或者是访问服务器上配置从属 IP 地址，可以使两个逻辑子网使用同一个物理子网。

许多较早期的网络是基于第二层网桥，而不是被划分成多个子网。正确使用从属 IP 地址可以把这样的网络改造成基于路由的多个子网。在网络中的路由交换机，通过配置的从属 IP 地址，可以了解同样连接在这个物理网络中的多个子网。

当一个网络的两个子网，被另一个网络在物理上分隔开。这时，可以把这个网络的地址作为从属 IP 地址，从而可以把一个逻辑网络中的两个在物理上被分隔开的网络在逻辑上连接在一起。

注意：

如果一个网段上的任意一台路由交换机配置了一个从属地址,则相同网段上的所有其它路由交换机也需要配置同样网段的从属 IP 地址。

在网络接口配置多个地址，在接口配置态使用下列命令：

命令	说明
[no] ip address *A.B.C.D/M* label *LINE*	删除/配置接口的从 IP 地址。 A.B.C.D 为接口 IP M 为掩码，范围 1-32 LINE：描述该从 IP

#以下配置 vlanif2 的从 IP 地址为 10.1.2.1/24

```
switch# con t
switch(config)# int vlanif2
switch(config-vlanif2)# ip address  10.1.2.1/24
switch(config-vlanif2)#
```

5.2 查看路由

在所有视图下使用下面命令，查看路由相关信息

命令	说明
show ip route	显示所有路由信息。
show ip route (kernel | connected | static | rip | ospf | isis | bgp | babel)	显示指定模块路由。

show ip route A.B.C.D **show ip route A.B.C.D/M**	显示指定网段路由。

#查看路由示例

```
switch#  show  ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

K>* 0.0.0.0/0 via 10.1.6.254, eth0
C>* 10.1.6.0/24 is directly connected, eth0
switch#
```

5.3 Static 配置

5.3.1 静态路由简介

静态路由是一种特殊的路由，它由管理员手工配置。下表列出了本章所包含的内容。当网络结构比较简单时，只需配置静态路由就可以使网络正常工作。仔细设置和使

用静态路由可以改进网络的性能，并可为重要的应用保证带宽。

静态路由的缺点在于：当网络发生故障或者拓扑发生变化后，静态路由不会自动改变，必须有管理员的介入。

5.3.2 缺省路由简介

缺省路由是另外一种特殊的路由。通常情况下，管理员可以通过手工方式配置缺省路由；但有些时候，也可以使动态路由协议生成缺省路由，如 OSPF 和 IS-IS。 简单来说，缺省路由是在没有找到匹配的路由表入口项时才使用的路由。在路由表中，缺省路由以到网络 0.0.0.0（掩码也为 0.0.0.0）的路由形式出现。可通过命令

display ip routing-table 查看当前是否设置了缺省路由。 如果报文的目的地址不能与路由表的任何入口项相匹配， 那么该报文将选取缺省路由。如果没有缺省路由且报文的目的地不在路由表中，那么该报文将被丢弃，并向源端返回一个 ICMP 报文，报告该目的地址或网络不可达。

5.3.3 出接口和下一跳

在配置静态路由时，可指定出接口 interface-type interface-name，也可指定下一跳地址 nexthop-address， 是指定发送接口还是指定下一跳地址要视具体情况而定。 实际上，所有的路由项都必须明确下一跳地址。在发送报文时，首先根据报文的目的地址寻找路由表中与之匹配的路由。只有指定了下一跳地址，链路层才能找到对应的链路层地址，并转发报文。

注意：

建议不要指定为以太网接口。因为以太网接口是广播类型的接口，会导致出现多个下一跳，无法唯一确定下一跳。在应用中，如果必须指定广播接口（如以太网接口）为发送接口，应同时指定通过该接口发送时对应的下一跳地址。

5.3.4 配置静态路由

在全局配置视图下，使用下面命令添加/删除静态路由

命令	说明
ip route *A.B.C.D A.B.C.D (A.B.C.D/INTERFACE)* **(reject | blackhole)** **ip route** *A.B.C.D A.B.C.D (A.B.C.D/INTERFACE/nu110)* **ip route** *A.B.C.D A.B.C.D* **(reject | blackhole)** **ip route** *A.B.C.D/M (A.B.C.D/INTERFACE)* **(reject | blackhole)** **ip route** *A.B.C.D/M (A.B.C.D/INTERFACE/nu110)* **ip route** *A.B.C.D/M* **(reject | blackhole)**	添加静态路由
ip route *A.B.C.D/M (A.B.C.D/INTERFACE/nu110) <1-255>* **ip route** *A.B.C.D A.B.C.D* **(reject | blackhole)** *<1-255>* **ip route** *A.B.C.D/M (A.B.C.D/INTERFACE)* **(reject | blackhole)** *<1-255>* **ip route** *A.B.C.D A.B.C.D (A.B.C.D/INTERFACE/nu110) <1-255>* **ip route** *A.B.C.D/M* **(reject | blackhole)** *<1-255>*	添加静态路由， 并指定该路由 Distance
no ip route *A.B.C.D A.B.C.D (A.B.C.D/INTERFACE)* **(reject | blackhole)** **no ip route A.***B.C.D A.B.C.D (A.B.C.D/INTERFACE/nu110)* **no ip route** *A.B.C.D A.B.C.D* **(reject | blackhole)** **no ip route** *A.B.C.D A.B.C.D* **(reject | blackhole)** *<1-255>* **no ip route** *A.B.C.D/M (A.B.C.D/INTERFACE)* **(reject | blackhole)** **no ip route** *A.B.C.D/M (A.B.C.D/INTERFACE/nu110)* **no ip route** *A.B.C.D/M (A.B.C.D/INTERFACE/nu110) <1-255>* **no ip route** *A.B.C.D/M* **(reject | blackhole)** **no ip route** *A.B.C.D/M* **(reject | blackhole)** *<1-255>*	删除静态路由

#以下配置 10.1.1.0/24 网段路由，下一跳为 11.1.1.2

```
switch# con t
switch(config)# ip ro 10.1.1.0/24 11.1.1.2
switch(config)#
```

5.4 RIP 配置

5.4.1 RIP 简介

5.4.1.1 概述

路由信息协议 RIP 是一个相对过时但仍然普遍使用的内部网关协议（IGP），主要应用于规模较小的同质型网络。RIP 是一个经典的距离向量路由协议，出现在 RFC 1058 中，RFC1388 中提出了改进的 RIP-2，并在 RFC 1723 和 RFC 2453 中进行了修订。

RIP 采用贝尔曼—福德（Bellman-Ford）算法，目前 RIP ipv4 有两个版本 RIPv1 和 RIPv2。RIP 有以下一些主要特性：

- RIP 属于典型的距离矢量路由选择协议。
- RIP 消息通过广播地址 255.255.255.255 进行发送，RIPv2 使用组播地址 224.0.0.9 发送消息，两者都使用 UDP 协议的 520 端口。
- RIP 以到目的网络的最小跳数作为路由选择度量标准，而不是在链路的带宽和延迟的基础上进行选择。
- RIP 是为小型网络设计的。它的跳数计数限制为 15 跳，16 跳为不可到达。
- RIP-1 是一种有类路由协议，不支持不连续子网设计。
- RIP-2 支持 CIDR 及 VLSM 可变长子网掩码，使其支持不连续子网设计。
- RIP 周期性进行完全路由更新，将路由表广播给邻居路由器，广播周期缺省为 30 秒。
- RIP 的协议管理距离为 120。

对于小型网络，RIP 就所占带宽而言开销小，易于配置、管理和实现，并且 RIP 还在大量使用中。但 RIP 也有明显的不足，即当有多个网络时会出现环路问题。为了解决环路问题，IETF 提出了水平分割法，在这个接口收到的路由信息不会再从该接口出去（split-Horizon）。分割范围解决了两个路由器之间的路由环路问题，但不能防止因网络规模较大、主要由延迟因素产生的环路。触发更新要求路由器在链路发生变化时立即传输它的路由表。这加速了网络的聚合，但容易产生广播泛滥。总之，环路问题的解决需要消耗一定的时间和带宽。若采用 RIP 协议，其网络内部所经过的链路数不能超过 15，这使得 RIP 协议不适于大型网络。

5.4.1.2 RIP 工作原理

RIP 是一种分布式的基于距离向量的路由选择协议，是因特网的标准协议，其最大的优点就是简单。RIP 协议要求网络中每一个路由器都要维护从它自己到其他每一个目的网 络的距离记录。RIP 协议将“距离”定义为：从一路由器到直接连接的网络的距离定义为 1。从一路由器到非直接连接的网络的距离定义为每经过一个路由器则距离加 1。“距离”也称为“跳数”。RIP 允许一条路径最多只能包含 15 个路由器，因此，距离等于 16 时即为不可达。可见 RIP 协议只适用于小型互联网。

RIP 2 由 RIP 而来，属于 RIP 协议的补充协议，主要用于扩大装载的有用信息的数

量，同时增加其安全性能。RIPv1 和 RIPv2 都是基于 UDP 的协议。在 RIP2 下，每台主机或路由器通过路由选择进程发送和接受来自 UDP 端口 520 的数据包。RIP 协议默认的路由更新周期是 30S。

5.4.1.3 RIP2 特性

RIP-2 是一种无类别路由协议（Classless Routing Protocol）。

RIP-2 协议报文中携带掩码信息，支持 VLSM（可变长子网掩码）和 CIDR。

RIP-2 支持以组播方式发送路由更新报文，组播地址为 224.0.0.9，减少网络与系统资源消耗。

RIP-2 支持对协议报文进行验证，并提供明文验证和 MD5 验证两种方式，增强安全性。

RIP-2 能够支持 VLSM。

5.4.1.4 RIP 防环机制

- 记数最大值（maximum hop count）：定义最大跳数（最大为 15 跳），当跳数为 16 跳时,目标为不可达。
- 水平分割（split horizon）：从一个接口学习到的路由不会再广播回该接口。cisco 可以对每个接口关闭水平分割功能。
- 毒性逆转（poison reverse）：从一个接口学习的路由会发送回该接口，但是已经被毒化，跳数设置为 16 跳，不可达。
- 触发更新（trigger update）：一旦检测到路由崩溃，立即广播路由刷新报文，而不等到下一刷新周期。
- 抑制计时器（holddown timer）：防止路由表频繁翻动，增加了网络的稳定性。

5.4.1.5 RIP 相关 RFC

RFC1058：Routing Information Protocol
RFC1723：RIP Version 2 - Carrying Additional Information
RFC1721：RIP Version 2 Protocol Analysis
RFC1722：RIP Version 2 Protocol Applicability Statement
RFC1724：RIP Version 2 MIB Extension
RFC2082：RIP-2 MD5 Authentication

5.4.2 配置 RIP

5.4.2.1 启动 RIP

要激活 RIP，进入全局配置视图，按以下步骤进行：

命令	说明
router rip	激活RIP路由进程，进入路由交换机配置模式。
network *network-number <network-mask>*	指定与RIP路由进程相关的网络号。

#配置示例，发布 10.1.1.0/24 网段

```
switch(config)# router rip
switch(config-rip)# network  10.1.1.0/24
switch(config-rip)#
```

5.4.2.2 配置邻居

RIP 通常是一个广播型协议，如果要使 RIP 路由更新能够到达非广播型网络，你必须对路由交换机进行配置以允许路由信息的交换。要达到这样的目的，在路由交换机配置模式中使用如下命令：

命令	说明
neighbor *ip-address*	配置邻居，用来交换路由信息。

#以下配置邻居（对端设备的接口 IP 为 10.1.1.2）

```
switch(config)# router rip
switch(config-rip)# neighbor  10.1.1.2
```

5.4.2.3 配置 RIP 缺省的度量值

缺省的度量值为 1，有效值为 1 到 16，要定义 RIP 缺省的度量值，在 RIP 视图下，使用如下命令：

命令	说明
default-metric *<1-16>*	配置默认度量值。
no default-metric	恢复度量值为默认值1

#配置缺省度量值为 2

```
switch# con t
switch(config)# router rip
switch(config-rip)# default-metric  2
switch(config-rip)#
```

5.4.2.4 配置路由更新频率

RIP 协议默认的路由更新周期是 30 秒。要调整计时器，在 RIP 视图下，使用如下命令：

命令	说明
timers basic *<5-2147483647> <5-*	配置路由更新的时间、路由信息超时时间、

2147483647> <5-2147483647>	garbage collection时间，单位：秒。
no timers basic	恢复路由更新的时间、路由信息超时时间、garbage collection时间为默认值

5.4.2.5 指定 RIP 版本号

缺省情况下，路由交换机接收 RIP-1 和 RIP-2 的分组，但只发送 RIP-1 的分组。通过配置，可以使路由交换机仅发送和接收 RIP-1 的分组，或者仅发送和接收 RIP-2 的分组。要达到这样的目的，在 RIP 视图下，使用如下命令：

命令	说明
version (1 | 2)	配置路由交换机仅仅发送和接受RIP-1或RIP-2的分组。

要控制接口发送 RIP-1 分组还是 RIP-2 分组，在接口视图下使用如下命令：

命令	说明
ip rip send version 1	配置接口仅仅发送RIP-1的分组。
ip rip send version 2	配置接口仅仅发送RIP-2的分组。

要控制接口接收 RIP-1 分组还是 RIP-2 分组，在接口视图下，使用如下命令：

命令	说明
ip rip receive version 1	配置接口仅仅接受RIP-1的分组。
ip rip receive version 2	配置接口仅仅接受RIP-2的分组。

5.4.2.6 配置 RIP 认证

RIP-1 不支持认证。如果你在发送和接收 RIP-2 的分组，你可以在接口上激活 RIP 认证。

在 RIP 激活的接口上，我们支持两种认证模式：明文认证和 MD5 认证。每个 RIP-2 分组中缺省时使用明文认证。

注意：如果处于安全的目的，不要在 RIP 分组中使用明文验证，这是因为未经加密的认证密钥发送在每个 RIP-2 分组中。如果在不涉及安全问题的情况下（例如：要保证配置错误的主机不能参与路由），可以使用明文验证。

要配置认证，在接口视图下，按如下步骤进行：

命令	说明
ip rip authentication mode (md5 | text)	配置接口认证为MD5或明文。
ip rip authentication string *LINE*	配置认证密钥

5.4.2.7 配置发布缺省路由

配置当前交换机/路由器向 RIP 邻居发布一条缺省路由，使用如下命令：

命令	说明
default-information originate	发布缺省路由。
no default-information originate	不发布缺省路由。

#配置发布缺省路由

```
switch(config)# router rip
switch(config-rip)# default-information originate
switch(config-rip)#
```

5.4.2.8 配置被动接口

配置借口为被动接口，使用如下命令，默认不是被动接口

命令	说明
passive-interface *IFNAME*	设置指定接口为被动接口。
passive-interface default	设置所有接口为被动接口。

#配置示例，配置所有接口都为被动接口

```
switch(config)# router rip
switch(config-rip)# passive-interface default
switch(config-rip)#
```

5.4.2.9 配置 RIP 引入外部路由信息

由于 RIP 要发布的路由信息中，有可能是引入的其他路由协议的路由信息，所以可通过指定 protocol 参数来对这些特定的路由信息进行过滤。如果没有指定 protocol 参数，则对所有要发布的路由信息进行过滤，包括引入的路由和本地 RIP 路由（相当于直连路由）。

命令	说明
redistribute *{bgp | connected | isis | kernel | ospf | static}*	引入指定外部路由
no redistribute *{bgp | connected | isis | kernel | ospf | static}*	不引入指定的外部路由

#配置示例，引入静态路由

```
switch(config)# router rip
switch(config-rip)# redistribute static
switch(config-rip)#
```

5.4.2.10 激活或禁止水平分割

正常情况下，与广播型 IP 网络相连并使用距离向量路由协议的路由交换机采用水平分割机制来减小路由环路的可能性。水平分割阻塞路由信息向接收到该路由信息的接口进行宣告。这样做可以优化多个路由交换机间的通信（尤其是在环路打破的时候）。但是，对于非

广播型网络（例如帧中继），情况并非那么理想。此时，你可能要禁止水平分割。

如果一个接口配置了辅助的 IP 地址并且激活了水平分割，路由更新的信源 IP 地址可能不包括每一个辅助地址。一条路由更新中的信源 IP 地址只包括一个网络号（除非水平分割被禁止）。

要激活或禁止水平分割，在接口视图下使用如下命令：

命令	说明
ip rip split-horizon	激活水平分割。
no ip rip split-horizon	禁止水平分割。

在缺省情况下，对于点对点接口，水平分割是激活的；对于点对多点接口，水平分割是禁止的。

注意：在一般情况下，推荐你不要改变缺省状态，除非你能肯定你的应用程序需要状态的改变才能正确地宣告路由。

```
#配置关闭 vlanif1 的水平分割
switch(config)# interface vlanif1
switch(config-vlanif1)# no ip rip split-horizon
switch(config-vlanif1)#

#配置开启 vlanif1 的水平分割
switch(config)# interface vlanif1
switch(config-vlanif1)#ip rip split-horizon
switch(config-vlanif1)#
```

5.4.2.11 对路由权值应用偏移量

偏移量列表是用来对那些由 RIP 学习到的入站和出站路由增加一个偏移量。这就提供了一个本地的机制来增加路由权值。另外，你还可以使用访问列表或接口来限制偏移量列表。要增加路由权值，在路由交换机配置模式中使用如下命令

命令	说明
offset-list *[access-list-name]* **{in | out}** *offset [IFNAME]*	对路由权值增加一个偏移量。 access-list-name：访问列表 offset：偏移量范围0-16 IFNAME：指定接口接口

5.4.3 监视和维护 RIP

监视和维护 RIP，可以显示网络的统计信息，如：RIP 协议参数配置、使用网络、网络通信实时跟踪等。这些信息能帮助你判断网络资源的利用，解决网络问题。能了解网络节点的可达性。

在特权视图下，使用下面的命令，可以显示 RIP 路由统计信息：

命令	说明

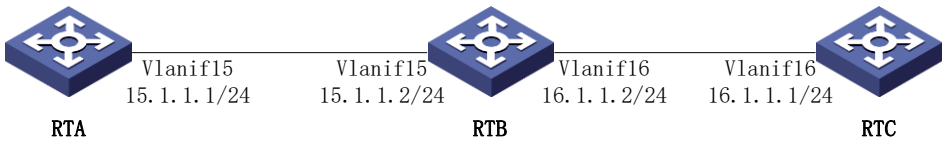
show ip rip	显示RIP协议当前状态。
show ip route rip	显示RIP所有路由。
show ip rip status	显示RIP协议统计相关信息。

在特权视图下，使用下面的命令，可以跟踪路由协议信息。

命令	说明
debug rip events	跟踪RIP路由加入路由表、从路由表中删除路由、路由改变等过程信息。
debug rip packet	跟踪RIP协议报文。
debug rip zebra	跟踪RIP与zebra路由控制模块间的通讯

5.4.4 RIP 配置举例

三台设备，组网如下。要求三台都启用 RIP，其中 RTC 引入静态路由。



三台设备配置如下：

RTA 配置：

```
interface vlanif15
 ip address 15.1.1.1/24
!
router rip
 network 15.1.1.0/24
 neighbor 15.1.1.2
 !
```

RTB 配置：

```
interface vlanif15
 ip address 15.1.1.2/24
!
interface vlanif16
 ip address 16.1.1.2/24
!
router rip
 network 15.1.1.0/24
 network 16.1.1.0/24
 neighbor 15.1.1.1
 neighbor 16.1.1.1
!
```

RTC 配置：

```
interface vlanif16
```

```
 ip address 16.1.1.1/24
!
router rip
 network 16.1.1.0/24
 neighbor 16.1.1.2
!
```

5.5 OSPF 配置

5.5.1 简介

5.5.1.1 概述

OSPF(Open Shortest Path First 开放式最短路径优先)是一个内部网关协议(Interior Gateway Protocol，简称 IGP)，用于在单一自治系统（autonomous system,AS）内决策路由。是对链路状态路由协议的一种实现，隶属内部网关协议(IGP)，故运作于自治系统内部。采用迪克斯加算法计算最短路径。

OSPF 是 IETF 的 OSPF 工作组的开发的 IGP 路由协议。 为 IP 网络设计的 OSPF 支持 IP 子网和外部路由信息标记，也允许报文的认证以及支持 IP 多播。

5.5.1.2 OSPF 基本概念

1、ROUTER-ID

每一台 OSPF 路由器只有一个 Router-ID，Router-ID 使用 IP 地址的形式来表示，确定 Router-ID 的方法为手工指定 Router-ID。

路由器上活动 Loopback 接口中 IP 地址最大的，也就是数字最大的，如 C 类地址优先于 B 类地址，一个非活动的接口的 IP 地址是不能被选为 Router-ID 的。如果没有活动的 Loopback 接口，则选择活动物理接口 IP 地址最大的。

注：如果一台路由器收到一条链路状态，无法到达该 Router-ID 的位置，就无法到达链路状态中的目标网络。

2、COST

OSPF 会自动计算接口上的 Cost 值，但也可以通过手工指定该接口的 Cost 值，手工指定的优先于自动计算的值。计算的 Cost，和接口带宽成反比，带宽越高，Cost 值越小。到达目标相同 Cost 值的路径，可以执行负载均衡，最多 6 条链路同时执行负载均衡。

3. OSPF 的协议报文

OSPF 有五种类型的协议报文：

- Hello 报文：周期性发送，用来发现和维持 OSPF 邻居关系。
- DD 报文（Database Description packet）：描述了本地 LSDB 的摘要信息，用于两台路由器进行数据库同步。

- LSR 报文（Link State Request packet）：向对方请求所需的 LSA。只有在双方成功交换 DD 报文后才会向对方发出 LSR 报文。
- LSU 报文（Link State Update packet） ：向对方发送其所需要的 LSA。
- LSAck 报文(Link State Acknowledgment packet)：用来对收到的 LSA 进行确认。
- AS External LSA（Type5）：由 ASBR 产生，描述到 AS 外部的路由，通告到所有的区域（除了 Stub 区域和 NSSA 区域）。
- NSSA LSA（Type7）：由 ASBR 产生，描述到 AS 外部的路由，仅在 NSSA 区域内传播

4. LSA 的类型

OSPF 中对路由信息的描述都是封装在 LSA 中发布出去的，常用的 LSA 有以下几种类型：

- Router LSA（Type1）：每个路由器都会产生，描述了路由器的链路状态和花费，在所属的区域内传播。
- Network LSA（Type2）：由 DR 产生，描述本网段的链路状态，在所属的区域内传播。
- Network Summary LSA（Type3）：由 ABR 产生，描述区域内某个网段的路由，并通告给其他区域。

ASBR Summary LSA（Type4）：由 ABR 产生，描述到 ASBR 的路由，通告给相关区域。

5.5.1.3 OSPF 区域

1、区域划分

OSPF 区域是基于路由器的接口划分的，而不是基于整台路由器划分的，一台路由器可以属于单个区域，也可以属于多个区域。

OSPF 网络分为以下 2 个级别的层次：

- 骨干区域 (backbone or area 0)
- 非骨干区域 (nonbackbone areas)

在一个 OSPF 区域中只能有一个骨干区域，可以有多个非骨干区域，骨干区域的区域号为 0。为了避免回环的产生，各非骨干区域间是不可以交换 LSA 信息的，他们只有与骨干区域相连，通过骨干区域相互交换信息。

非骨干区域和骨干区域之间相连的路由叫边界路由（ABRs-Area Border Routers），只有 ABRs 记载了接入各区域的所有路由信息。各非骨干区域内的非 ABRs 只记载了本区域内的路由表，若要与外部区域中的路由相连，只能通过本区域的 ABRs，由 ABRs 连到骨干区域的 BR，再由骨干区域的 BR 连到要到达的区域。

2、路由器类型

- Internal Router：内部路由器

该类路由器的所有接口都属于同一个 OSPF 区域。

- ABR(Area Border Router)：区域边界路由器

该类路由器可以同时属于两个以上的区域，但其中一个必须是骨干区域。ABR 用来连接骨干区域和非骨干区域，它与骨干区域之间既可以是物理连接，也可以是逻

辑上的连接。

- Backbone Router(BR)：骨干路由器

该类路由器至少有一个接口属于骨干区域。因此，所有的 ABR 和位于 Area0 的内部路由器都是骨干路由器。

- ASBR(Autonomous System Boundary Router)：自治系统边界路由器.

与其他 AS 交换路由信息的路由器称为 ASBR。ASBR 并不一定位于 AS 的边界，它可能是区域内路由器，也可能是 ABR。只要一台 OSPF 路由器引入了外部路由的信息，它就成为 ASBR。

3、虚链接

以下 2 中情况需要使用到虚链路：

- 通过一个非骨干区域连接到一个骨干区域.
- 通过一个非骨干区域连接一个分段的骨干区域两边的部分区域.

虚链接是一个逻辑的隧道（Tunnel），配置虚链接的一些规则：

- 虚链接必须配置在 2 个 ABR 之间.
- 虚链接所经过的区域叫 Transit Area，它必须拥有完整的路由信息.
- Transit Area 不能是 Stub Area.

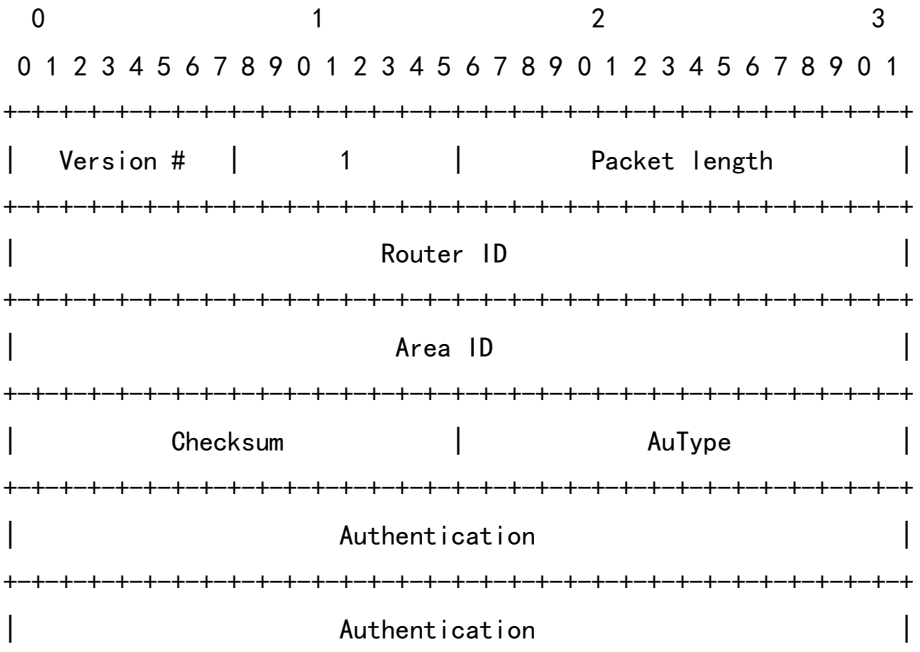
尽可能的避免使用虚链接，它增加了网络的复杂程度和加大了排错的难度

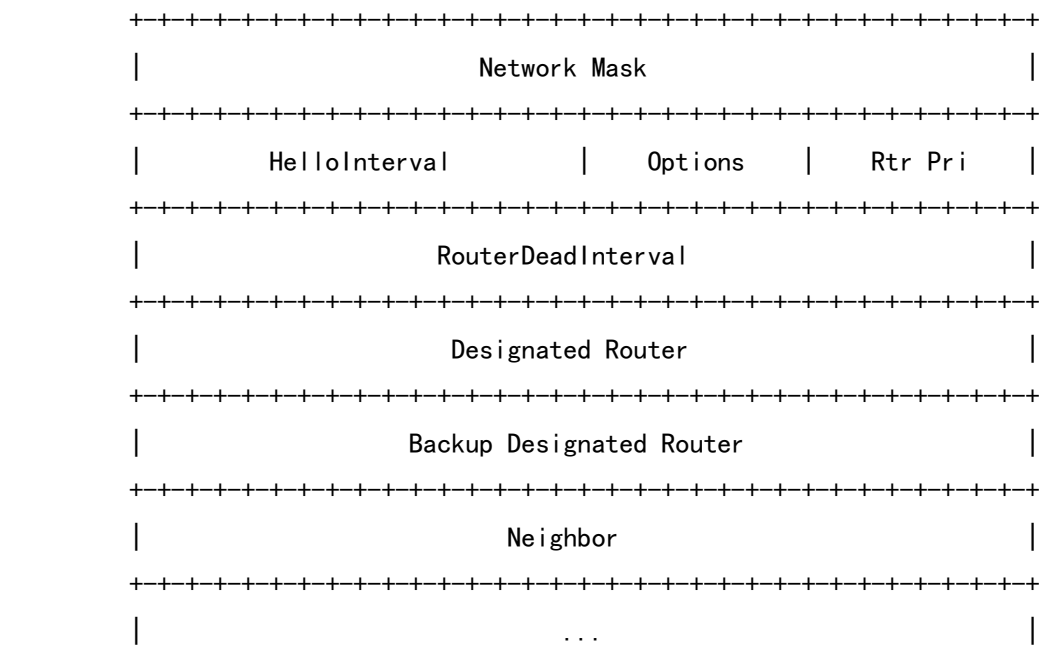
5.5.1.4 OSPF 报文格式

- Hello 报文

Hello 包的 OSPF 包类型为 1。这些包被周期性的从各个接口（包括虚链路）发出，用来建立和维护邻居关系。另外，在支持组播或广播的物理网络上，Hello 包使用组播地址发送。

Hello 包的格式如下：

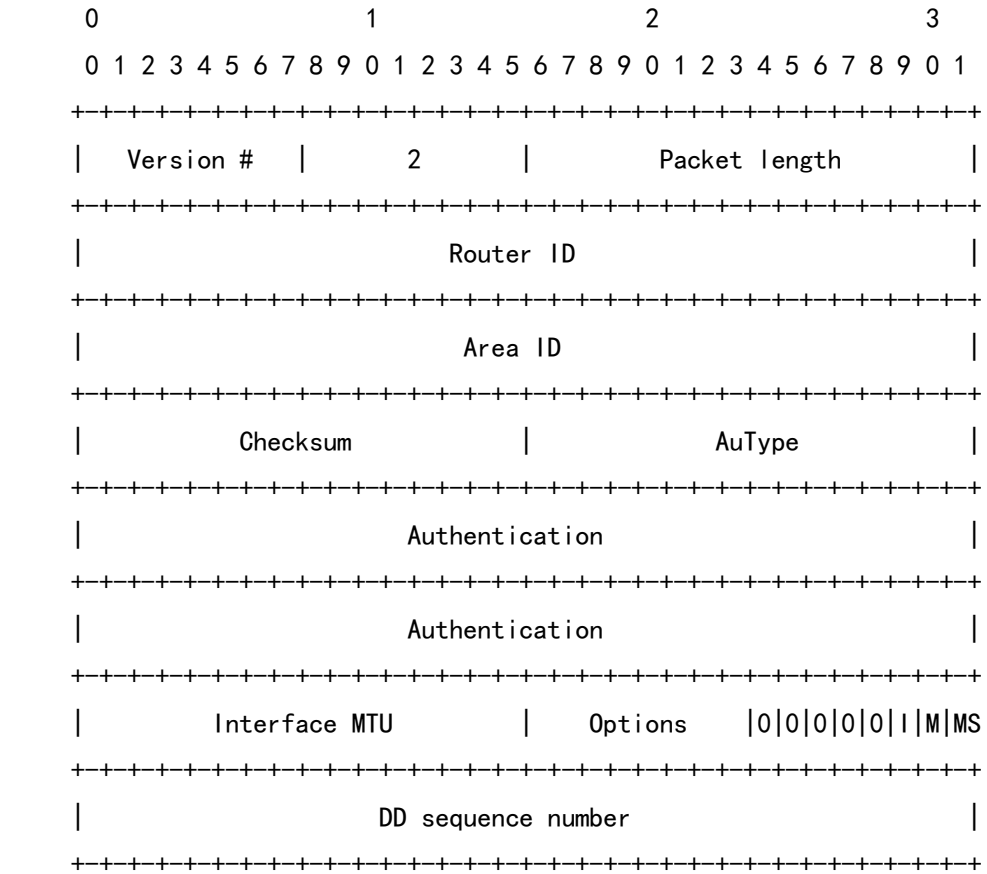


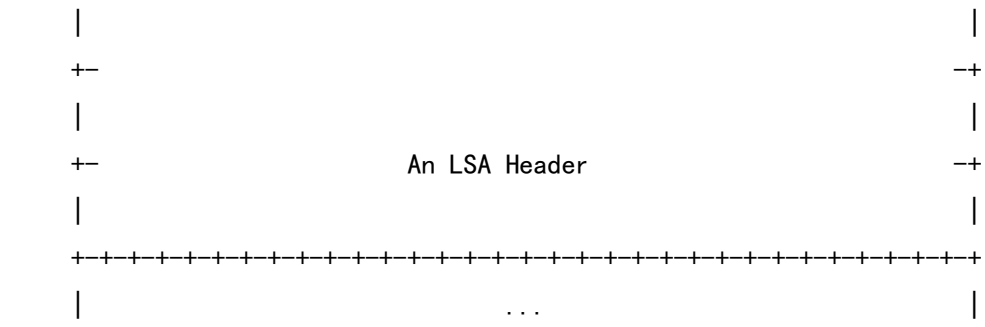


- 数据库描述包

数据库描述包（Database Description）的 OSPF 包类型为 2。当邻接关系初始化后，便开始交换这些数据包。它们描述了链路状态数据库的摘要信息（只包含 LSA 的头部信息）。

数据库描述包的格式如下：

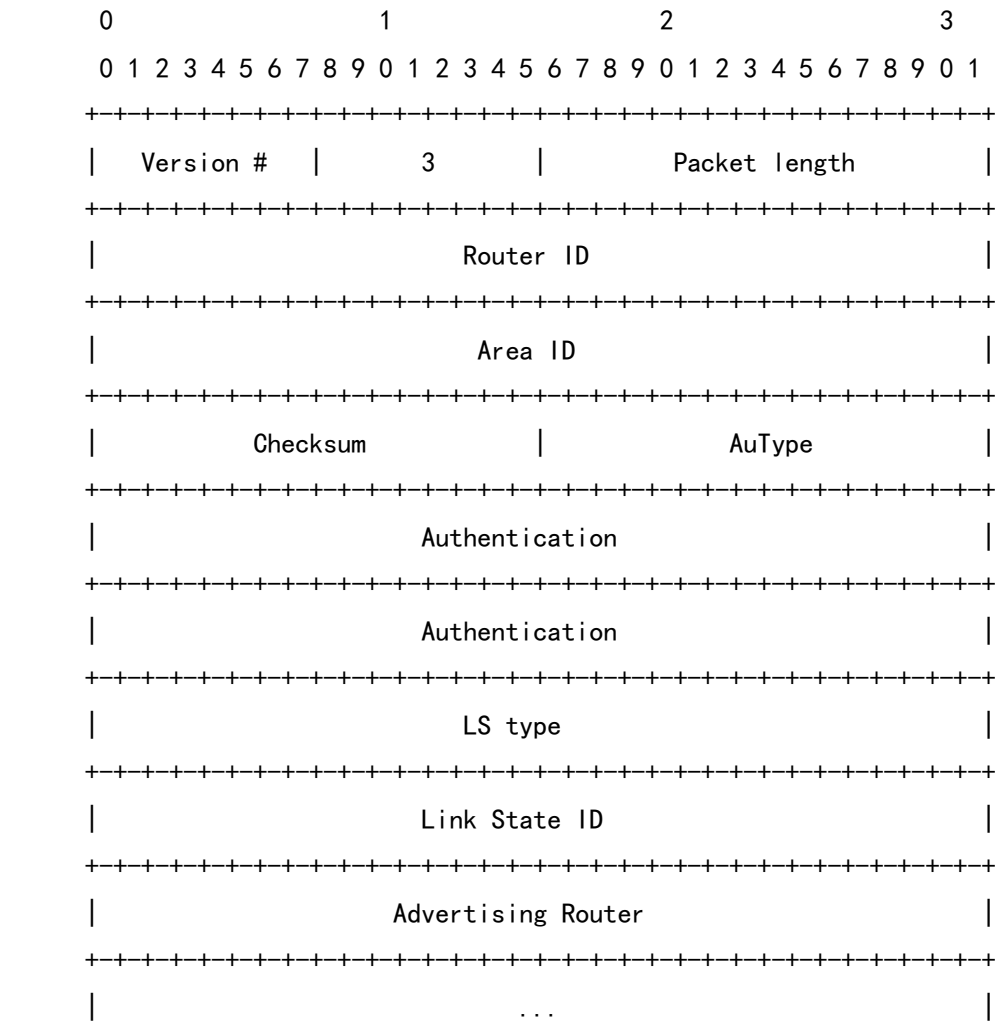




- 链路状态请求包

链路状态请求包（Link State Request）的 OSPF 包类型为 3。在交换数据库描述包之后，路由器便知道其自身链路状态数据库缺少哪些 LSA，以及哪些 LSA 是过期的。这时就可以发送链路状态请求包来请求对方发送缺少的 LSA 和最新的 LSA。

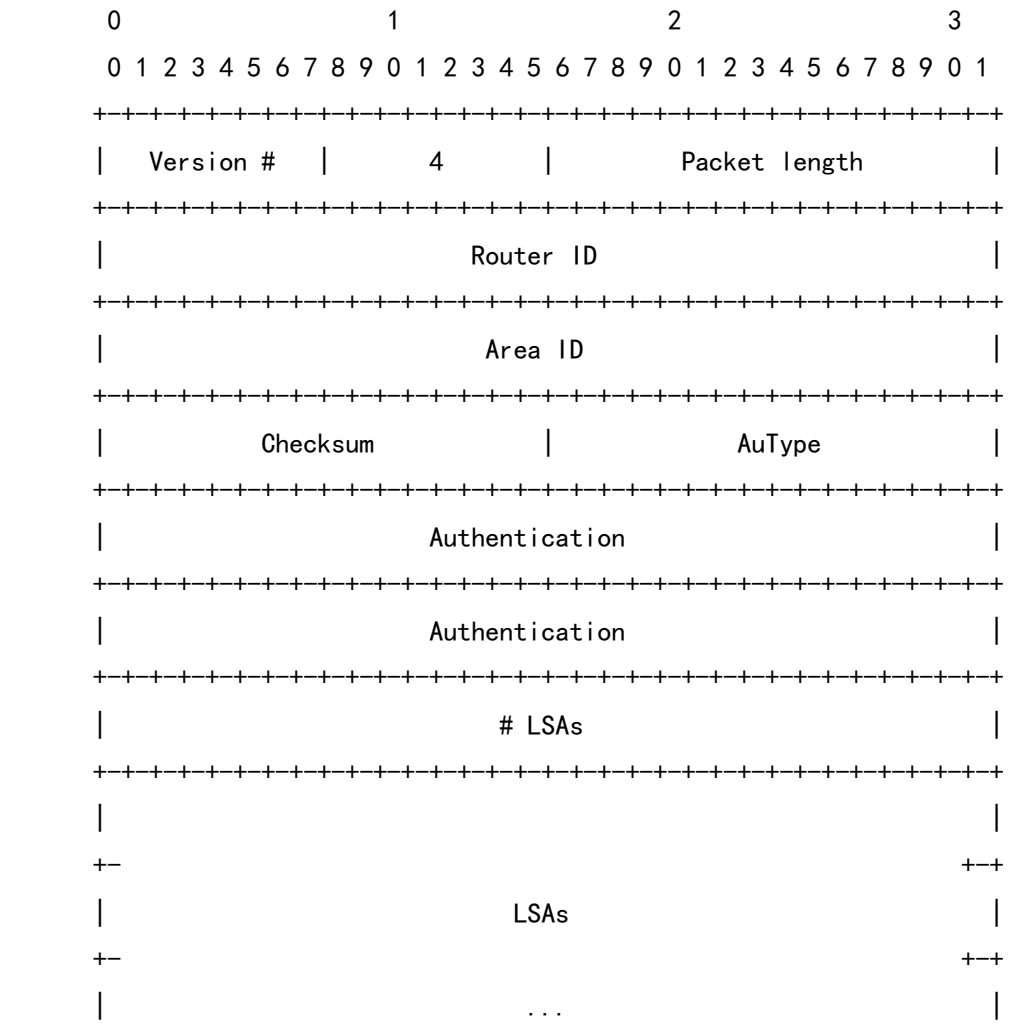
链路状态请求包格式如下：



- 链路状态更新包

链路状态更新包（Link State Update）的 OSPF 包类型为 4。LSA 的洪泛就是由此类型的包实现的。每一个链路状态更新包可能包含多条 LSA 信息。这里的 LSA 信息是完整的，而不像数据库描述包只包含 LSA 的头部信息。

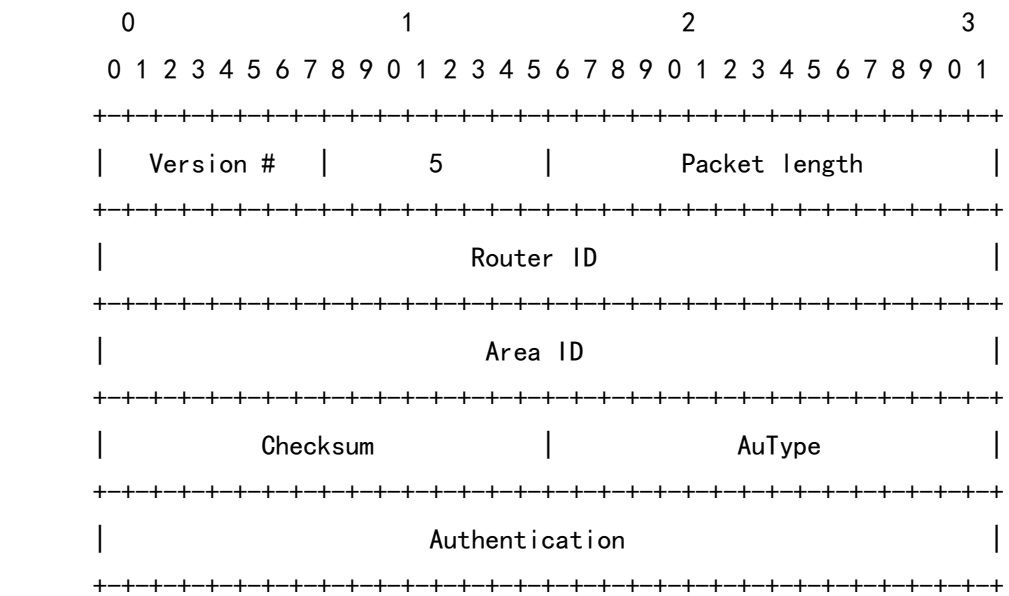
链路状态更新包的格式如下：

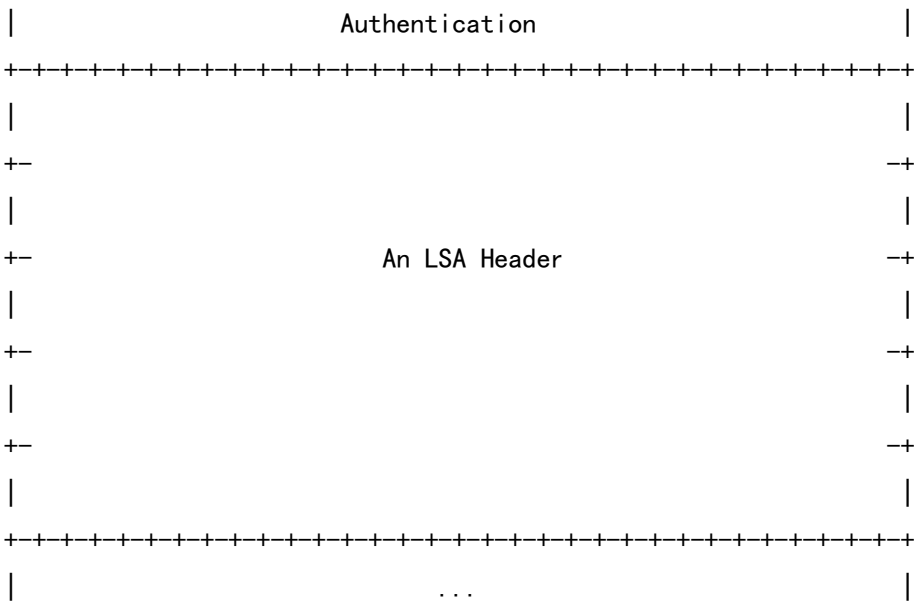


- 链路状态确认包

链路状态确认包（Link State Acknowledgement）的 OSPF 包类型为 5。为了确保 LSA 的洪泛是可靠的，LSA 信息必须被显式的确认。

链路状态确认包的格式如下：





5.5.2 OSPF 配置任务

5.5.2.1 配置区域所包含的网段

该处的网段是指运行 OSPF 协议接口的 IP 地址所在的网段。一个网段只能属于一个区域，或者说每个运行 OSPF 协议的接口必须指明属于某一个特定的区域。

命令	说明
network A.B.C.D/M **area** (A.B.C.D | <0-4294967295>)	配置区域所包含的网段
no network A.B.C.D/M **area** (A.B.C.D | <0-4294967295>)	删除区域包含网段

#配置示例，配置 10.1.1.0/24 属于区域 0

```
switch# con t
switch(config)# router ospf
switch(config-router)# network 10.1.1.0/24 area  0
switch(config-router)#
```

5.5.2.2 配置 OSPF 的接口参数

在 OSPF 实现中，允许按照需要修改接口有关的 OSPF 参数。并不需要改变任何一个参数，但必须保证某些参数在相连网络的所有路由交换机上的保持一致。

在接口视图下，使用下面的命令配置接口参数，在相关命令前加 no，恢复默认值：

命令	说明
ip ospf cost *<1-65535>*	配置OSPF 接口发送包的权值。
ip ospf retransmit-interval *<3-65535>*	属于同一个OSPF接口的邻居之间重传LSA

	的秒数。（单位：秒）
ip ospf transmit-delay *<1-65535>*	配置在一个OSPF接口传输LSA的估计时间（单位：秒）。
ip ospf priority *<0-255>*	配置路由交换机成为OSPF DR路由交换机的优先值。
ip ospf hello-interval <1-65535>	配置在OSPF接口发送hello 包的时间间隔。
ip ospf dead-interval *<1-65535>*	在这个规定的时间间隔内，未收到邻居的hello包，则认为邻居路由交换机已关机。
ip ospf authentication-key *key*	设置一个网段内的邻接路由的认证口令。它使用ospf简单认证
ip ospf authentication (|null | message-digest)	要求OSPF在该接口使用简单、null、MD5 认证。
ip ospf message-digest-key <1-255> md5 *key*	设置md5加密口令

5.5.2.3 配置 OSPF 接口网络类型

不管网络的物理媒体类型，你都可以配置你的网络或者为广播网或者非广播、多访问网络。使用这个特性，你能灵活配置网络，可以将物理上的广播网络配置成非广播、多访问网络；也能配置非广播网络（X.25, Frame Relay, 与 SMDS）成为广播网络。这个特征也减少对邻居的配置，具体参见为非广播网络配置 OSPF 相关内容。

配置非广播、多访问网络为广播网络或者非广播网络，即假设从每一个路由交换机到其他路由交换机都存在虚链路，或假设为一个全网状网络。由于花费的限制，这常常是不现实的；或者有一个部分网状网。这种情形下，你可以配置成点到多点网络。不相邻的路由器、交换机之间可以通过虚链路交换路由信息。

在接口视图下，用下面的命令配置 OSPF 的网络类型。

命令	说明
ip ospf network [broadcast	non-broadcast	point-to-multipoint	point-to-point]	配置OSPF的网络类型。

交换机的网络类型是广播类型。

5.5.2.4 配置 OSPF 被动接口

在 OSPF 视图下，用下面的命令配置 OSPF 不发送 ospf 报文。

命令	说明
passive-interface *IFNAME* **passive-interface** *IFNAME A.B.C.D* **passive-interface default**	配置OSPF接口为被动。抑制该接口更新路由信息

no passive-interface *IFNAME* **no passive-interface** *IFNAME A.B.C.D* **no passive-interface default**	不抑制该接口更新路由信息

```
#以下配置 vlanif2 为被动接口
switch(config)# router ospf
switch(config-ospf)# passive-interface  vlanif2
switch(config-ospf)#
```

5.5.2.5 配置 OSPF 域参数

可以配置的区域参数有：认证、指定 Stub 区、为默认汇总路由指定权值。认证采用基于口令保护。

Stub 区域即不分发外部路由到该区的区域。取而代之的是,在 ABR 生成一条默认外部路由进入 stub 区域，使它能到达自治区域的外部网络。为了利用 OSPF Stub 支持的特性，在 Stub 区域必须使用默认路由，为了进一步减少发送进入 Stub 区域的 LSA 数，你能在 ABR 禁止汇总（No Summary）来减少发送汇总 LSA(类型 3)进入 Stub 区域。

在 OSPF 视图下，使用下面的命令设定区域参数：

命令	说明
area *area-id* **authentication**	使用简单口令认证。
area *area-id* **authentication message-digest**	使用MD5进行认证。
area *area-id* **stub** [no-summary]	定义一个stub区。
area *area-id* **nssa** [no-summary]	定义一个nssa区。
area *area-id* **default-cost** *cost*	为Stub区域的默认路由设定权值。

OSPF 划分区域后，可以减少网络中 LSA 的数量，OSPF 的扩展性也得以增强。对于位于 AS 边缘的一些非骨干区域，为了更多的缩减其路由表规模和降低 LSA 的数量，可以将它们配置为 Stub 区域。

Stub 区域不能引入外部路由，为此又产生了 NSSA 区域的概念。NSSA 区域中允许 Type7 LSA 的传播。Type7 LSA 由 NSSA 区域的 ASBR 产生，当它到达 NSSA 的 ABR 时，就会转换成 AS-External LSA，并通告到其他区域。

在划分区域之后，非骨干区域之间的 OSPF 路由更新是通过骨干区域来交换完成的。对此，OSPF 要求所有非骨干区域必须与骨干区域保持连通，并且骨干区域自身也要保持连通。

5.5.2.6 配置 OSPF 域内路由的汇总

这个特性使得 ABR 广播一条汇总路由到其他区域。在 OSPF 中，ABR 将广播每一个网络到其他区域。如果网络号按照某种方式分配，使得它们连续，你能配置 ABR 广播一条汇总路由到其他区。汇总路由能覆盖一定范围的所有网络。

在 OSPF 视图下，使用下面的命令设定地址范围：

命令	目的

area *area-id* **range** *address mask*	设定汇总路由的地址范围。
no area *area-id* **range** *address mask*	取消汇总路由的地址范围

详细可配参数如下：

```
area (A.B.C.D|<0-4294967295>) range A.B.C.D/M
area (A.B.C.D|<0-4294967295>) range A.B.C.D/M advertise
area (A.B.C.D|<0-4294967295>) range A.B.C.D/M advertise cost <0-16777215>
area (A.B.C.D|<0-4294967295>) range A.B.C.D/M cost <0-16777215>
area (A.B.C.D|<0-4294967295>) range A.B.C.D/M not-advertise
area (A.B.C.D|<0-4294967295>) range A.B.C.D/M substitute A.B.C.D/M
no area (A.B.C.D|<0-4294967295>) range A.B.C.D/M
no area (A.B.C.D|<0-4294967295>) range A.B.C.D/M (advertise|not-advertise)
no area (A.B.C.D|<0-4294967295>) range A.B.C.D/M advertise cost <0-16777215>
no area (A.B.C.D|<0-4294967295>) range A.B.C.D/M cost <0-16777215>
no area (A.B.C.D|<0-4294967295>) range A.B.C.D/M substitute A.B.C.D/M
```

5.5.2.7 生成默认路由

能要求 ASBR 生成一条默认路由进入 OSPF 路由域。无论何时，你配置路由交换机分发路由进入 OSPF 路由域，该路由自动变成 ASBR。然而，ASBR 默认并不生成默认路由进入 OSPF 路由域。

在 OSPF 视图下，使用下面的命令，强制 ASBR 生成默认路由：

命令	目的
default-information originate	强制ASBR生成默认路由进入OSPF路由域。
no default-information originate	不生成默认路由

详细可配参数如下：

```
default-information originate always
default-information originate always metric <0-16777214>
default-information originate always metric <0-16777214> metric-type (1|2)
default-information originate always metric <0-16777214> metric-type (1|2)
route-map WORD
default-information originate always metric <0-16777214> route-map WORD
default-information originate always metric-type (1|2)
default-information originate always metric-type (1|2) metric <0-16777214>
default-information originate always metric-type (1|2) metric <0-16777214>
route-map WORD
default-information originate always metric-type (1|2) route-map WORD
default-information originate always route-map WORD
default-information originate metric <0-16777214>
default-information originate metric <0-16777214> metric-type (1|2)
default-information originate metric <0-16777214> metric-type (1|2) route-
map WORD
default-information originate metric <0-16777214> route-map WORD
```

```
default-information originate metric-type (1|2)
default-information originate metric-type (1|2) metric <0-16777214>
default-information originate metric-type (1|2) metric <0-16777214> route-map WORD
default-information originate metric-type (1|2) route-map WORD
default-information originate route-map WORD
```

5.5.2.8 配置 OSPF 的管理距离

管理距离是路由信息源的信任等级，如单个路由交换机或一组路由交换机。一般来说，管理距离是 0—255 之间的整数，值越大，信任级别越低。如果管理距离为 255，则路由信息源不被信任且应当被忽略。

OSPF 使用三类不同的管理距离：域间、域内和外部。在一个区域内的路由是域内；到其他区域的路由是区域间；其他路由协议域分发来的路由为外部。每种类型路由的默认值为 110。

在 OSPF 视图下，使用下面的命令，配置 OSPF 的距离值：

命令	说明
distance *<1-255>*	改变OSPF 域内、域间以及外部路由管理距离值。
no distance *<1-255>* **no distance ospf**	恢复默认配置

详细可配参数如下：

```
distance <1-255>
distance ospf external <1-255>
distance ospf external <1-255> inter-area <1-255>
distance ospf external <1-255> inter-area <1-255> intra-area <1-255>
distance ospf external <1-255> intra-area <1-255>
distance ospf external <1-255> intra-area <1-255> inter-area <1-255>
distance ospf inter-area <1-255>
distance ospf inter-area <1-255> external <1-255>
distance ospf inter-area <1-255> external <1-255> intra-area <1-255>
distance ospf inter-area <1-255> intra-area <1-255>
distance ospf inter-area <1-255> intra-area <1-255> external <1-255>
distance ospf intra-area <1-255>
distance ospf intra-area <1-255> external <1-255>
distance ospf intra-area <1-255> external <1-255> inter-area <1-255>
distance ospf intra-area <1-255> inter-area <1-255>
distance ospf intra-area <1-255> inter-area <1-255> external <1-255>
```

5.5.2.9 配置路由计算的计时器

你能配置 OSPF 收到拓扑变化消息与开始计算 SPF 之间的时延。也能配置连续两次计算

SPF 之间的间隔。在路由交换机配置模式，使用下面的命令进行配置：

命令	说明
timers throttle spf *delaytime inittime maxtime*	设定在一个域中路由计算的时间延迟，初始时间间隔，最大时间间隔
no timers throttle spf	恢复默认配置

5.5.2.10 配置 OSPF 引入外部路由

由于 OSPF 要发布的路由信息中，有可能是引入的其他路由协议的路由信息，所以可通过指定 protocol 参数来对这些特定的路由信息进行过滤。如果没有指定 protocol 参数，则对所有要发布的路由信息进行过滤，包括引入的路由和本地 RIP 路由（相当于直连路由）。

命令	说明
redistribute *{bgp | connected | isis | kernel | rip | static}*	引入指定外部路由
no redistribute *{bgp | connected | isis | kernel | rip | static}*	不引入指定的外部路由

#配置示例，引入静态路由

```
switch(config)# router ospf
switch(config-ospf)# redistribute  static
switch(config-ospf)#
```

5.5.2.11 配置 OSPF 路由 ID

OSPF 路由 ID 配置命令如下。

命令	说明
ospf router-id *A.B.C.D*	设置ospf的路由ID

#配置示例，设置 ospf 的路由 ID 为 1.1.1.1

```
switch> enable
switch# configure terminal
switch(config)# router ospf
switch(config-router)# ospf router-id 1.1.1.1
switch(config-router)#
```

5.5.3 监视和维护 OSPF

5.5.3.1 OSPF 查看

能显示网络的统计信息，如：OSPF 路由表的内容、缓冲和数据库等数据。这些信息能帮

助你判断网络资源的利用，解决网络问题。能了解网络节点的可达性，发现网络数据包经过网络的路由。

命令	说明
show ip ospf	显示OSPF 路由进程的一般信息。
show ip ospf database	显示OSPF数据库的相关信息。
show ip ospf border-routers	显示ABR与ASBR的内部路由表项。
show ip ospf interface	显示有关OSPF接口的信息。
show ip ospf neighbor	显示OSPF的邻居信息。
show ip ospf route	显示ospf路由

详细可带参数如下

```
show ip ospf border-routers
show ip ospf database
show ip ospf database
(asbr-summary|external|network|router|summary|nssa-external|opaque-link|opaque-area|opaque-as) (self-originate|)
show ip ospf database
(asbr-summary|external|network|router|summary|nssa-external|opaque-link|opaque-area|opaque-as) [A.B.C.D]
show ip ospf database
(asbr-summary|external|network|router|summary|nssa-external|opaque-link|opaque-area|opaque-as) A.B.C.D adv-router A.B.C.D
show ip ospf database
(asbr-summary|external|network|router|summary|nssa-external|opaque-link|opaque-area|opaque-as) adv-router A.B.C.D
show ip ospf database
(asbr-summary|external|network|router|summary|nssa-external|opaque-link|opaque-area|opaque-as|max-age|self-originate)
show ip ospf interface [INTERFACE]
show ip ospf neighbor
show ip ospf neighbor A.B.C.D
show ip ospf neighbor IFNAME
show ip ospf neighbor IFNAME detail
show ip ospf neighbor all
show ip ospf neighbor detail
show ip ospf neighbor detail all
show ip ospf route
```

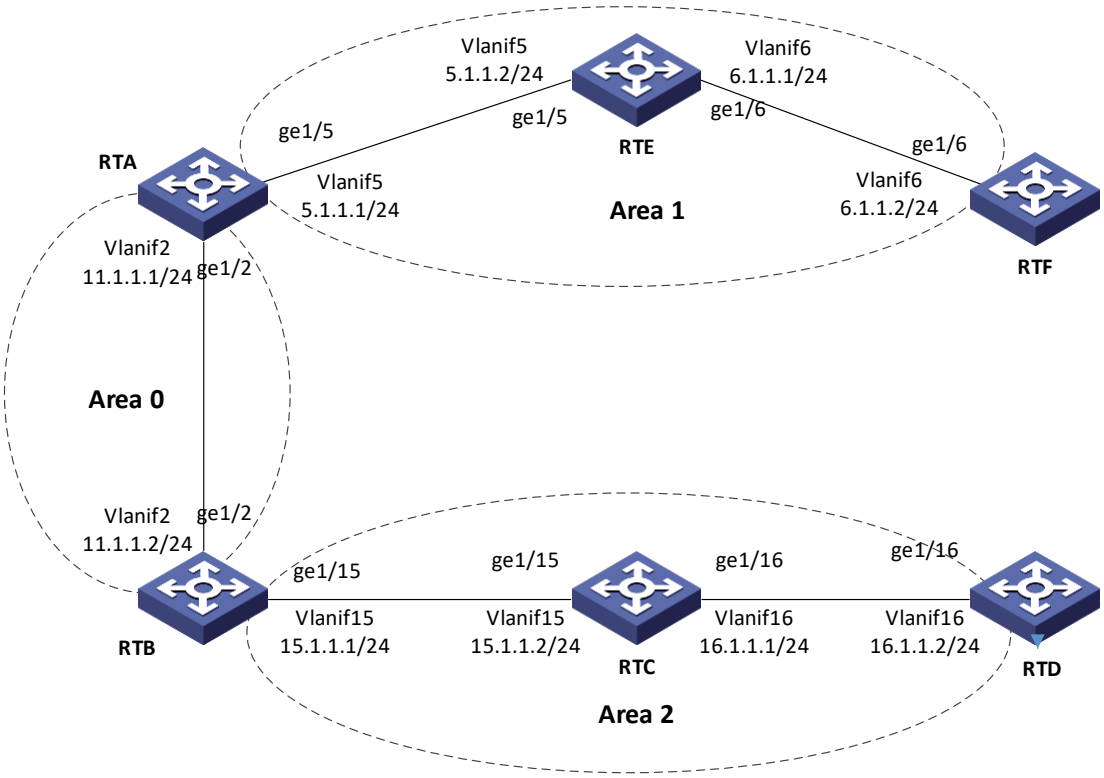
5.5.3.2 OSPF 调试

使用下面的命令，可以显示 OSPF 调试信息：

命令	说明
debug ospf {event | ism | isa | nsm | nssa | packet | zebra }	打开ospf调试开关

5.5.4 OSPF 配置举例

组网图如下，所有的路由器都运行 OSPF，整个自治系统划分为 3 个区域。其中 RTA 和 RTB 作为 ABR 来转发区域之间的路由， RTD 作为 ASBR 引入了外部路由（静态路由）。要求将 Area1 配置为 Stub 区域，减少通告到此区域内的 LSA 数量，但不影响路由的可达性。



1、配置 IP

划分正确的 vlan，在所有路由器的 vlanif 接口配置相应 IP 地址，举例说明 RTA 配置，其他参考 RTA

```
RTA:
RTA> enable
RTA# configure terminal
RTA(config)# vlan 2
RTA(config-vlan2)# exit
RTA(config)# vlan 5
RTA(config-vlan5)# exit
RTA(config)# interface vlanif2
RTA(config-vlanif2)# ip address 11.1.1.1/24
RTA(config-vlanif2)# exit
RTA(config)# interface vlanif5
RTA(config-vlanif5)# ip address 5.1.1.1/24
```

```
RTA(config-vlanif5)# exit
RTA(config)# interface ge1/2
RTA(config-ge1/2)# switchport pvid 2
RTA(config-ge1/2)# exit
RTA(config)# interface ge1/5
RTA(config-ge1/5)# switchport pvid 5
RTA(config-ge1/5)# exit
RTA(config)#
```

2、配置 OSPF

在所有路由器配置相应 OSPF，并划到各自区域，举例说明 RTA 配置，其他参考 RTA

RTA:

```
RTA(config)# router ospf
RTA(config-router)# network 11.1.1.0/24 area 0
RTA(config-router)# network 5.1.1.0/24 area 1
RTA(config-router)# exit
```

3、查看配置

RTA:

```
RTA# show ip route
```

5.6 VRRP 配置

5.6.1 简介

VRRP（Virtual Router Redundancy Protocol，虚拟路由冗余协议）是一种容错协议。通常，一个网络内的所有主机都设置一条缺省路由，这样，主机发出的目的地址不在本网段的报文将被通过缺省路由发往路由器 RouterA，从而实现了主机与外部网络的通信。当路由器 RouterA 坏掉时，本网段内所有以 RouterA 为缺省路由下一跳的主机将断掉与外部的通信产生单点故障。VRRP 就是为解决上述问题而提出的，它为具有多播组播或广播能力的局域网（如：以太网）设计。

VRRP 将局域网的一组路由器（包括一个 Master 即活动路由器和若干个 Backup 即备份路由器）组织成一个虚拟路由器，称之为一个备份组。这个虚拟的路由器拥有自己的 IP 地址 10.100.10.1（这个 IP 地址可以和备份组内的某个路由器的接口地址相同，相同的则称为 ip 拥有者），备份组内的路由器也有自己的 IP 地址（如 Master 的 IP 地址为 10.100.10.2，Backup 的 IP 地址为 10.100.10.3）。局域网内的主机仅仅知道这个虚拟路由器的 IP 地址 10.100.10.1，而并不知道具体的 Master 路由器的 IP 地址 10.100.10.2 以及 Backup 路由器的 IP 地址 10.100.10.3。[1]它们将自己的缺省路由下一跳地址设置为该虚拟路由器的 IP 地址 10.100.10.1。于是，网络内的主机就通过这个虚拟的路由器来与其它网络进行通信。如果备份组内的 Master 路由器坏掉，Backup 路由器将会通过选举策略选出一个新的 Master 路由器，继续向网络内的主机提供路由服务。从而实现网络内的主机不间断地与外部网络进行通信

5.6.1.1 概述

5.6.1.2 原理

一个 VRRP 路由器有唯一的标识:VRID,范围为 0—255。该路由器对外表现为唯一的虚拟 MAC 地址,地址的格式为 00-00-5E-00-01-[VRID]。主控路由器负责对 ARP 请求用该 MAC 地址做应答。这样,无论如何切换,保证给终端设备的是唯一一致的 IP 和 MAC 地址,减少了切换对终端设备的影响。

VRRP 控制报文只有一种:VRRP 通告(advertisement)。它使用 IP 多播数据包进行封装,组地址为 224.0.0.18,发布范围只限于同一局域网内。这保证了 VRID 在不同网络中可以重复使用。为了减少网络带宽消耗只有主控路由器才可以周期性的发送 VRRP 通告报文。备份路由器在连续三个通告间隔内收不到 VRRP 或收到优先级为 0 的通告后启动新的一轮 VRRP 选举。

在 VRRP 路由器组中,按优先级选举主控路由器,VRRP 协议中优先级范围是 0—255。若 VRRP 路由器的 IP 地址和虚拟路由器的接口 IP 地址相同,则称该虚拟路由器作 VRRP 组中的 IP 地址所有者;IP 地址所有者自动具有最高优先级:255。优先级 0 一般用在 IP 地址所有者主动放弃主控者角色时使用。可配置的优先级范围为 1—254。优先级的配置原则可以依据链路的速度和成本、路由器性能和可靠性以及其它管理策略设定。主控路由器的选举中,高优先级的虚拟路由器获胜,因此,如果在 VRRP 组中有 IP 地址所有者,则它总是作为主控路由的角色出现。对于相同优先级的候选路由器,按照 IP 地址大小顺序选举。VRRP 还提供了优先级抢占策略,如果配置了该策略,高优先级的备份路由器便会剥夺当前低优先级的主控路由器而成为新的主控路由器。

为了保证 VRRP 协议的安全性,提供了两种安全认证措施:明文认证和 IP 头认证。明文认证方式要求:在加入一个 VRRP 路由器组时,必须同时提供相同的 VRID 和明文密码。适合于避免在局域网内的配置错误,但不能防止通过网络监听方式获得密码。IP 头认证的方式提供了更高的安全性,能够防止报文重放和修改等攻击。

5.6.2 VRRP 配置任务

5.6.2.1 创建/删除 VRRP 组

配置 VRRP 的虚拟地址后就开启了该虚拟路由交换机，虚拟地址必须和该端口的 primary IP 地址在同一网段，否则虚拟路由交换机将一直停在 Init 状态，虚拟地址不用配置掩码，虚拟路由交换机使用端口 primary mask 作为自己的掩码，当虚拟地址和端口 primary IP 地址一致时，系统将自动提升该虚拟路由交换机的优先级为 255。

在接口视图下，使用下列命令配置 VRRP 组：

命令	说明
vrrp *<1-255>* **ip** *A.B.C.D*	创建VRRP组。
no vrrp *<1-255>*	删除VRRP组。

#配置示例，创建 VRRP 组 1，及虚拟地址 10.1.1.2

```
switch(config)# int vlan5
switch(config-vlanif5)# ip address  10.1.1.1/24
switch(config-vlanif5)# vrrp  1 ip 10.1.1.2
switch(config-vlanif5)#
```

5.6.2.2 配置 VRRP 优先级抢占

优先级抢占只对处于 Backup 下的路由交换机有效，当收到由 master 路由交换机发送来的 announce 报文后，当检测到 master 的优先级没有本地配置的优先级高时，如果 Backup 路由交换机配置了优先级抢占，Backup 路由交换机将从 Backup 状态跃迁到 master 状态，并向外发送 announce 报文。反之，则继续停留在 Backup 状态。

缺省方式是优先级抢占。

在接口下进行下列配置。

命令	说明
vrrp *<1-255>* **preempt** **vrrp** *<1-255>* **preempt** *<1-1000>*	配置VRRP优先级抢占。 配置抢占时延
no vrrp *<1-255>* **preempt**	恢复VRRP优先级抢占缺省方式。

5.6.2.3 配置 VRRP 优先级

当虚拟地址和端口地址一致时，VRRP 会自动提升优先级为 255，当虚拟地址或者端口地址发生变化后，优先级值会自动恢复到原来配置的值。

缺省值为 100。

在端口配置模式下进行下列配置。

命令	说明

vrrp *<1-255>* **priority** *<1-254>*	配置VRRP优先级
no vrrp *<1-255>* **priority**	恢复VRRP优先级为缺省值。

5.6.2.4 配置 VRRP 时钟值

时钟值将决定虚拟路由交换机从故障中恢复的最短时间，当 master 路由交换机 down 了后，Backup 路由交换机将在 3*advertisement + skew_time 间隔后跃迁为 master 路由交换机，asvertisement 时钟太长显然不利于故障恢复，推荐采用缺省值。

缺省值为 1 second。

在接口视图下进行下列配置：

命令	说明
vrrp *<1-255>* **advertisement** *<1-10>*	配置VRRP宣告报文的间隔
no vrrp *<1-255>* **advertisement**	恢复VRRP宣告报文的间隔为缺省值。

5.6.3 VRPP 的监控与维护

5.6.3.1 VRRP 查看

在全局视图下，使用下列命令查看 VRRP 信息：

命令	说明
show vrrp *[<1-255>]*	显示VRRP信息。

#配置示例，显示 VRRP 组：

```
switch# show  vrrp
Virtual Router ID: 1
 State: MASTER
 Interface: vlanif1
 Gratuitous arp delay: 5
 Base Priority: 100
 Effective Priority: 100
 Advert interval: 1 secs
 Virtual ip: 10.1.1.2
switch#
```

5.6.3.2 VRRP 调试

使用下面的命令，可以显示 VRRP 调试信息：

命令	说明

debug vrrp events	监视VRRP的事件
debug vrrp packet	监视VRRP的报文。
debug vrrp interface	监视VRRP的接口信息

5.6.4 VRRP 配置示例

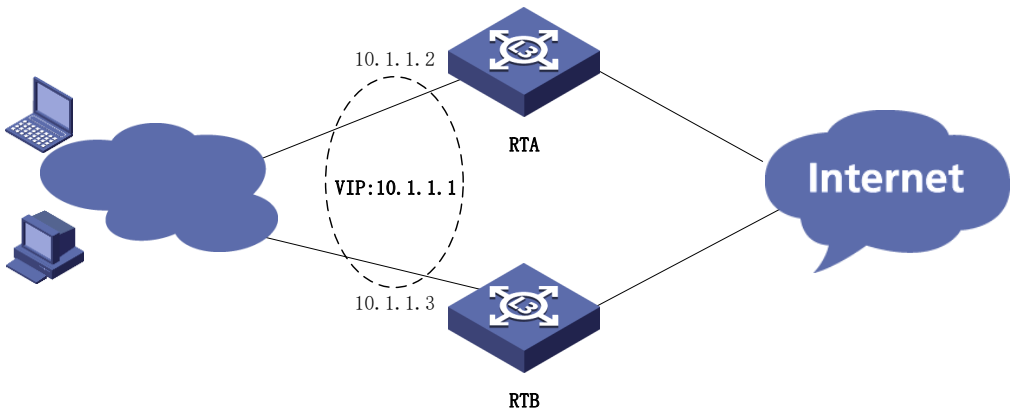
VRRP 协议的工作机理与 CISCO 公司的 HSRP（Hot Standby Routing Protocol）有许多相似之处。但二者主要的区别是在 CISCO 的 HSRP 中，需要单独配置一个 IP 地址作为虚拟路由器对外体现的地址，这个地址不能是组中任何一个成员的接口地址。

使用 VRRP 协议，不用改造网络结构，最大限度保护了投资，只需最少的管理费用，却大大提升了网络性能，具有重大的应用价值。

最典型的 VRRP 应用：RTA、RTB 组成一个 VRRP 路由器组，假设 RTB 的处理能力高于 RTA，则将 RTB 配置成 IP 地址所有者，H1、H2、H3 的默认网关设定为 RTB。则 RTB 成为主控路由器，负责 ICMP 重定向、ARP 应答和 IP 报文的转发；一旦 RTB 失败，RTA 立即启动切换，成为主控，从而保证了对客户透明的安全切换。

在 VRRP 应用中，RTB 在线时 RTA 只是作为后备，不参与转发工作，闲置了路由器 RTA 和链路 L1。通过合理的网络设计，可以达到备份和负载分担双重效果。让 RTA、RTB 同时属于互为备份的两个 VRRP 组：在组 1 中 RTA 为 IP 地址所有者；组 2 中 RTB 为 IP 地址所有者。将 H1 的默认网关设定为 RTA；H2、H3 的默认网关设定为 RTB。这样，既分担了设备负载和网络流量，又提高了网络可靠性。

示例图如下所示：



1. 配置交换机 Switch A

```
RTA(config)# int vlan2
RTA(config-vlanif2)# ip address  10.1.1.2/24
RTA(config-vlanif2)# vrrp 1 ip 10.1.1.1
RTA(config-vlanif2)# exit
RTA(config)# show  vrrp
Virtual Router ID: 1
 State: MASTER
 Interface: vlanif2
 Gratuitous arp delay: 5
```

```
 Base Priority: 100
 Effective Priority: 100
 Advert interval: 1 secs
 Virtual ip: 10.1.1.1
RTA(config)#
```

2. 配置交换机 Switch B

```
RTB(config)# int vlan2
RTB(config-vlanif2)# ip address  10.1.1.3/24
RTB(config-vlanif2)# vrrp 1 ip 10.1.1.1
RTB(config-vlanif2)# exit
RTB(config)# show  vrrp
Virtual Router ID: 1
 State: BACKUP
 Interface: vlanif2
 Gratuitous arp delay: 5
 Base Priority: 100
 Effective Priority: 100
 Advert interval: 1 secs
 Virtual ip: 10.1.1.1
RTB(config)#
```

5.7 ARP 配置

5.7.1 ARP 简介

地址解析协议（Address Resolution Protocol，ARP）是在仅知道主机的 IP 地址时确定地址解析协议定其物理地址的一种协议。因 IPv4 和以太网的广泛应用，其主要作用是通过已知 IP 地址，获取对应物理地址的一种协议。但其也能在 ATM(异步传输模式)和 FDDIIP(Fiber Distributed Data Interface 光纤分布式数据接口)网络中使用。从 IP 地址到物理地址的映射有两种方式:表格方式和非表格方式。ARP 具体说来就是将网络层(IP 层，也就是相当于 OSI 的第三层）地址解析为数据链路层（MAC 层，也就是相 当于 OSI 的第二层）的 MAC 地址。

OSI 模式把网络工作分为七层，IP 地址在第三层,.MAC 地址在第二层。协议在发送数据包时，得先封装第三层（IP 地址）和第二层（MAC 地址）的报头，但协议只知道目的节点的 IP 地址，不知道其 MAC 地址，又不能跨第二、三层，所以得用 ARP 的服务。

5.7.2 ARP 配置

5.7.2.1 配置静态 ARP

在全局配置态，使用下面命令配置 IP 与 MAC 的静态映射：

命令	说明
arp static *A.B.C.D MMMM-MMMM-MMMM*	添加静态 ARP。
no arp static *A.B.C.D*	删除静态 ARP。

#以下示例，配置静态 ARP

```
switch(config)# arp  static  10.1.1.2 0001-0002-0003
switch(config)#
```

5.7.2.2 配置 ARP 老化时间

ARP 老化时间默认 300 秒。

在子接口配置视图下，使用下面命令设置 ARP 老化时间

命令	说明
arp timeout *< 30-2147483647>*	配置 ARP 老化时间。
no arp timeout	恢复 ARP 老化时间为默认值。

#以下示例，配置 arp 老化时间为 100 秒

```
switch(config)# interface vlanif 2
switch(config-vlanif2)# arp timeout 300
switch(config-vlanif2)#
```

5.7.2.3 清除 ARP 缓存

在特权模式视图下使用下面命令清除 ARP 缓存中所有的表项。

命令	说明
clear arp [interface *IFNAME*]	清除 ARP 缓存。

#以下示例，清除所有动态 arp 缓存

```
switch# clear arp
switch#
```

5.7.3 检视和维护 ARP

系统可以显示特定的 arp 信息。

在所有视图下使用下面命令，查看 arp 表项

命令	说明
show arp [dynamic | static | *A.B.C.D*]	显示 ARP 缓存

注意：

ARP 欺骗可以导致目标计算机与网关通信失败；更可怕的是会导致通信重定向，所有的数据都会通过攻击者的机器，因此存在极大的安全隐患。基于端到端的 IP-MAC 双向绑定可以解决 ARP 欺骗。

但是对于不支持 IP-MAC 双向绑定的设备，可以用绑定端口-MAC 的交换来预防 ARP 欺骗。

#以下示例，显示 ARP 缓存

```
switch# show arp dynamic

Ip              Mac               Interface     Type      Timeout(s)
192.0.2.133       48:4d:7e:b6:eb:f0  vlanif1       dynamic   555

Total: 1

switch#
```

6 安全管理

6.1 访问控制

6.1.1 system access-policy 设置过滤规则

在全局配置视图下，使用 system access-policy 命令配置交换机的过滤规则

命令	说明
system access-policy deny/permit	设置过滤规则为禁止或者允许

#示例

```
switch# configure terminal
switch(config)# system access-policy permit
switch(config)#
```

6.1.1 system access-rule 设置访问规则

在全局配置视图下，使用 system access-policy 命令配置交换机的过滤规则

命令	说明
System access-rule A.B.C.D/M all/http/ssh/telnet	设置过滤规则为禁止或者允许

#示例

```
switch# configure terminal
switch(config)# system access-rule 192.168.3.1/24 all
switch(config)#
```

6.2 防攻击设置

6.2.1 简介

6.2.1.1 概述

通常的网络攻击，一般是侵入或破坏网上的服务器（主机），盗取服务器的敏感数据或干扰破坏服务器对外提供的服务。也有直接破坏网络设备的网络攻击，这种破坏影响较大，会导致网络服务异常，甚至中断。

SWITCH 提供的攻击防范功能能够检测出多种类型的网络攻击，并能采取相应的措施保护交换机/路由器免受恶意攻击，保证内部网络及系统的正常运行。

6.2.1.2 攻击种类

攻击类型有以下几类：

- 拒绝服务型攻击。拒绝服务型 DoS（Denial of Service）攻击是使用大量的数据包攻击系统，使系统无法接受正常用户的请求， 或者挂起不能正常的工作。 主要 DoS 攻击有 SYN Flood、Fraggle 等。拒绝服务攻击和其他类型的攻击不同之处在于：攻击者并不是去寻找进入内部网络的入口，而是阻止合法用户访问资源或路由器。
- 畸形报文攻击。畸形报文攻击是通过向目标系统发送有缺陷的 IP 报文，使得目标系统在处理这样的 IP 包时会出现崩溃，给目标系统带来损失。主要的畸形报文攻击有 Ping of Death、Teardrop 等。
- 扫描窥探攻击。扫描窥探攻击是利用 ping 扫射（包括 ICMP 和 TCP）来标识网络上存活着的系统，从而准确的指出潜在的目标。利用 TCP 和 UDP 等进行端口扫描，就能检测出操作系统的种类和潜在的服务种类。攻击者通过扫描窥探就能大致了解目标系统提供的服务种类和潜在的安全漏洞，为进一步侵入系统做好准备。

6.2.2 配置

6.2.2.1 禁止 ping 本设备

缺省允许 ping 本设备。在全局配置视图下，通过以下命令，使本设备不回应 ping 报文：

命令	说明
[no] system ignore icmp-echo	配置忽略 ping 包。

no 命令为恢复允许 ping 本设备。

#忽略对本设备的 ping 报文

```
switch(config)# system  ignore  icmp-echo
```

```
switch(config)#
```

6.2.2.2 防止对本设备的 syn 攻击

通过以下命令，防止对本设备的 syn 攻击。

命令	说明
[no] system protection syn-ack	配置防止 syn 攻击

```
#设置防 syn 攻击
switch(config)# system  protection  syn-ack
switch(config)#
```

6.2.2.3 设置上送设备报文速率

协议报文、目的为本设备的报文，会上送给设备处理，以下命令配置上送给设备的速率，单位 pps。

命令	说明
system rate-limit *pps*	配置报文上送速度

```
#配置上送给设备处理的报文速率为 1000
switch(config)# system  rate-limit  1000
switch(config)#
```

6.3 ACL 配置

6.3.1 简介

交换机为了过滤数据包，需要配置一系列的规则，以决定什么样的数据包能够通过，这些规则就是通过访问控制列表 ACL(Access Control List)定义的。访问控制列表是由 permit | deny 语句组成的一系列有顺序的规则，这些规则根据数据包的源地址、目的地址、协议号等来描述。

ACL 通过这些规则对数据包进行分类，这些规则应用到交换机接口上，交换机根据这些规则判断哪些数据包可以接收，哪些数据包需要拒绝。

6.3.2 配置 MAC 访问列表

6.3.2.1 创建 MAC 访问列表

要想在端口上应用 MAC 访问列表，必须首先创建 MAC 访问列表，当成功创建一个 MAC 访问列表后，就进入 MAC 访问列表配置模式，在该模式下可以配置 MAC 访问列表的条目。

进入全局配置视图下，使用下面命令配置 MAC 访问列表

命令	说明
[no] mac acl *name*	添加/删除一个 MAC 访问列表。 *name* 为 MAC 访问列表的名称

#配置添加一个 MAC 访问列表：

```
switch(config)# mac acl 2
switch(config-acl-mac-2)#
```

6.3.2.2 配置 MAC 访问列表规则

使用 permit/deny 命令来配置 MAC 访问列表允许/拒绝的条目，一个 MAC 访问列表可以配置多个允许/拒绝的条目。

同样的条目在一个访问列表中只允许配置一次。

进入 MAC 访问列表配置视图下，配置 MAC 访问列表规则

命令	说明
rule *rule-id* {**deny** | **permit**} {**any** | **host** *src-mac-addr*} {**any** | **host** *dst-mac-addr*}[ethertype]	添加一个MAC访问列表条目，可以重复该命令来添加多个MAC访问列表条目。 any表示匹配任何MAC地址； src-mac-addr为源MAC地址； dst-mac-addr为目的MAC地址； ethertype为匹配的以太网数据包的类型。
no rule *rule-id*	删除一个MAC访问列表条目，可以重复该命令来添加/删除多个MAC访问列表条目。

#访问列表配置举例

#创建 mac 访问列表

```
switch(config)# mac acl 1
```

#添加 mac 访问列表规则

```
switch(config-acl-mac-2)# rule 2 permit 0001-0002-0003 any
```

6.3.2.3 应用 MAC 访问列表

可以在任何物理端口上应用已经创建的 MAC 访问列表，但是每一个端口只能应用一个 MAC 访问列表,，同一个 MAC 访问列表可以被应用到多个端口上。

进入接口视图下应用 MAC 访问列表。

命令	说明
[no] mac access-group *list number*	在端口上应用已经创建的MAC访问列表或者删除已经应用到端口的MAC访问列表： *list number*为mac访问列表的序号。

#以下配置，首先创建访问列表 2，并在接口 ge1/2 上应用 mac 访问列表 2

```
switch(config)# mac acl  2
switch(config-acl-mac)# rule 1 permit  0001-2222-3333 any
```

```
switch(config-acl-mac)# exit
switch(config)# interface ge1/45
switch(config-ge1/45)# mac access-group 2
```

6.3.3 配置 IP 访问列表

6.3.3.1 创建 IP 访问列表

在全局配置试图下，执行下列命令配置 IP 访问列表。

命令	说明
[no] ip acl *list number*	创建/删除IP访问列表。

#配置添加一个 MAC 访问列表：

```
switch(config)# ip acl 100
switch(config-acl-ip-100)#
```

6.3.3.2 配置 IP 访问列表规则

为创建访问列表规则，在 IP 访问列表视图下执行下列命令：

命令	说明
rule *rule-id* {**deny	permit**} *source source-mask* [source-port] *destination destination-mask [dest-port]*	在扩展访问列表配置模式下，指定一个或多个允许或不允许条件。这决定该包通过还是不通过。

注意:在初始建立访问列表后,任何后续的增加部分(可能从终端键入)都放入表的尾部。当建立访问列表时，记住缺省时访问列表的结尾包含隐含的 deny 语句。

#配置添加一个 MAC 访问列表：

```
switch(config)# ip acl 100
switch(config-acl-ip-100)# rule 6 deny 192.0.2.6 255.255.255.0 0.0.0.0 0.0.0.0
switch(config-acl-ip-100)#
```

6.3.3.3 应用访问列表

当建立了访问列表后，可以将它应用到一个或多个端口上，可以应用到入口。

在端口配置态使用以下命令。

命令	说明
ip access-group *list number*	将访问列表应用到端口。

#示例：

```
switch(config)# ip acl 100
switch(config-acl-ip-100)# rule 6 deny 192.0.2.6 255.255.255.0 0.0.0.0
```

```
0.0.0.0
    switch(config-acl-ip-100)#ex
    switch(config)# interface ge1/45
    switch(config-ge1/45)# ip access-group 100
    switch(config-ge1/45)#
```

6.3.3.4 IP 访问列表示例

#在以下例子中，禁止端口 g1/2 通过目的为 10.1.0.0/16，源为 22.1.1.0/24 的报文通过。

```
    switch(config)#
    switch(config)# ip acl
      <100-999>  Acl name
    switch(config)# ip acl  200
    switch(config-acl-ip-200)# rule  3 deny  22.1.1.0 255.255.255.0 10.1.0.0
255.255.0.0
    switch(config-acl-ip-200)# exit
    switch(config)# int g1/2
    switch(config-ge1/2)# ip access-group  200
    switch(config-ge1/2)#
```

6.4 流量监控

在全局配置视图下，使用 traffic monitor 命令配置交换机的端口流量监控功能

命令	说明
traffic monitor rx/tx (0-1073741824)	使能交换机端口的入口/出口流量监控

#示例

```
switch# configure terminal
switch(config)# interface ge1/5
switch(config-ge1/5)# traffic monitor rx 200
switch(config-ge1/5)#
```

6.5 告警配置

6.5.1 系统告警

6.5.1 alarm 系统告警使能

在全局配置视图下，使用 alarm 命令配置交换机的告警使能

命令	说明

alarm	使能告警

#示例

```
switch(config)# alarm
switch(config)#
```

6.5.2 alarm 系统告警参数配置

在全局配置视图下，使用 alarm 命令配置告警参数

命令	说明
alarm cpu-threshold (30-100)	设置CPU告警阈值
alarm mem-threshold (30-100)	设置内存告警阈值

注：范围：30-100,0表示不告警；当超过配置阀值，将会产生告警。

#示例

```
switch(config)# alarm mem-threshold 50
switch(config)#
```

6.5.2 链路告警

在全局配置视图下，使用 alarm link 命令来配置端口的告警

命令	说明
alarm link	配置端口告警

#示例

```
switch# configure terminal
switch(config)# interface ge1/5
switch(config-ge1/5)# alarm link
switch(config-ge1/5)#
```

7 系统维护

7.1 日志配置

7.1.1 show log 查看日志信息

在全局配置视图下，使用 show log 命令来查看交换机日志

命令	说明
show log	查看交换机日志信息

#示例

```
switch# show log
1970/01/01 00:00:06 GSM: Before set rlimit CORE dump current is:0, max is:-
1

1970/01/01 08:13:24 VTYSH: @console: ena
1970/01/01 08:13:29 VTYSH: @console: reboot
1970/01/01 08:14:01 VTYSH: @console: con t
1970/01/01 08:15:11 ZEBRA: vlanif1 changes to UP
1970/01/01 08:15:11 GSM: ge1/1 is up
1970/01/01 08:15:20 WEB: admin login from 192.0.2.133
1970/01/01 09:01:36 WEB:  logout from 192.0.2.133
1970/01/01 09:20:17 WEB: admin login from 192.0.2.133
1970/01/01 09:23:50 VTYSH: @console: ntp unicast-server 202.108.6.95
1970/01/01 09:45:25 VTYSH: @console: log monitor
1970/01/01 09:51:49 VTYSH: @console: exit
1970/01/01 09:52:00 VTYSH: @console: show lo
1970/01/01 09:52:04 VTYSH: @console: show log

switch#
```

7.2 诊断测试

7.2.1 PING

在全局视图下，使用 ping 命令来测试网络连通状态

命令	说明
ping *ip_address*	目的IP地址。

示例

```
switch# ping 11.1.1.1
PING 11.1.1.1 (11.1.1.1) 56(84) bytes of data.
64 bytes from 11.1.1.1: icmp_seq=1 ttl=64 time=0.022 ms
64 bytes from 11.1.1.1: icmp_seq=2 ttl=64 time=0.030 ms
64 bytes from 11.1.1.1: icmp_seq=3 ttl=64 time=0.030 ms
64 bytes from 11.1.1.1: icmp_seq=4 ttl=64 time=0.031 ms
64 bytes from 11.1.1.1: icmp_seq=5 ttl=64 time=0.029 ms
64 bytes from 11.1.1.1: icmp_seq=6 ttl=64 time=0.024 ms
^C
--- 11.1.1.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5799ms
rtt min/avg/max/mdev = 0.022/0.027/0.031/0.007 ms
```

7.2.2 traceroute

Traceroute 通过发送小的数据包到目的设备直到其返回，来测量其需要多长时间。一条路径上的每个设备 Traceroute 要测 3 次。输出结果中包括每次测试的时间(ms)和设备的名称（如有的话）及其 IP 地址

在全局视图下，使用 traceroute 命令来检测数据包路径

命令	说明
Traceroute {ip | ipv6} *ip_address*	目的IP地址。

示例：

```
switch# traceroute  ip 123.125.114.144
traceroute to 123.125.114.144 (123.125.114.144), 30 hops max, 60 byte packets
 1  * * *
 2  c-24-0-0-1.hsd1.nj.comcast.net (24.0.0.1)  0.414 ms  0.465 ms  0.566 ms
 3  116.238.184.1 (116.238.184.1)  17.506 ms  19.330 ms  21.195 ms
 4  124.74.34.1 (124.74.34.1)  34.916 ms  35.033 ms  34.962 ms
 5  124.74.215.181 (124.74.215.181)  34.925 ms  36.009 ms  38.158 ms
 6  61.152.86.46 (61.152.86.46)  48.895 ms * *
 7  * 202.97.46.50 (202.97.46.50)  32.083 ms  30.852 ms
 8  219.158.32.169 (219.158.32.169)  51.713 ms  49.593 ms  55.515 ms
 9  219.158.4.201 (219.158.4.201)  123.260 ms  127.119 ms  126.632 ms
10  123.126.0.66 (123.126.0.66)  59.877 ms  52.931 ms  52.865 ms
11  123.126.6.166 (123.126.6.166)  56.330 ms  37.875 ms  48.043 ms
12  202.106.43.174 (202.106.43.174)  64.906 ms 202.106.43.66 (202.106.43.66) 42.564 ms 202.106.43.174 (202.106.43.174)  58.588 ms
switch#
```

7.3 NTP 设置

7.3.1 NTP 客户端配置

在全局配置视图下，使用 ntp unicast-server 命令配置 NTP 服务器

命令	说明
ntp unicast-server A.B.C.D	配置NTP服务器

#示例

```
switch(config)# ntp unicast-server 202.108.6.95
switch(config)#
```

7.3.2 NTP 服务器配置

7.3.2.1 NTP 服务器使能

在全局配置视图下，使用 ntp unicast-server 命令来使能 NTP 服务器

命令	说明
ntp server enable/disable	使能/禁用NTP服务器功能

#示例

```
switch(config)# ntp server enable
switch(config)#
```

7.3.2.2 NTP 服务器参数配置

在全局配置视图下，使用 ntp refclock-master 命令配置 NTP 服务器参数

命令	说明
ntp refclock-master (1-15)	配置ntp服务器参数

#示例

```
switch(config)# ntp refclock-master 2
switch(config)#
```

7.4 重启设备

7.4.1 reboot 重启

在特权视图下，使用 reboot 命令来重启设备

命令	说明
reboot	重起系统

#示例

```
switch# reboot
Are you sure reboot [y/n]
```

输入字符 y 将重启

7.5 在线升级

7.5.1 upgrade 在线升级

在特权视图下，使用 upgrade 命令来保存配置

命令	说明
upgrade software tftp A.B.C.D Filename	使用tftp方式升级
upgrade software tftp A.B.C.D Filename USERNAME PASSWORD	使用ftp方式升级

#示例

tftp 方式升级

```
switch# upgrade software tftp 192.0.2.133 x53322s_patch.bin
```

ftp 方式升级

```
switch# upgrade software tftp 192.0.2.133 x53322s_patch.bin admin admin
```

8 保存配置

8.1 write 保存配置

在特权视图下，使用 write 命令来保存配置

命令	说明
write	保存当前运行配置，作为启动配置

#示例

```
switch# write
Building Configuration...
Integrated configuration saved to switch.conf
[OK]
switch#
```

附录 缩略语表

缩略语	英文全称	中文
ARP	Address Resolution Protocol	地址解析协议
BPDU	Bridge Protocol Data Unit	网桥协议数据单元
CRC	Cyclic Redundancy Check	循环冗余校验
DHCP	Dynamic Host Configuration Protocol	动态主机配置协议
DSCP	Differentiated Services CodePoint	差分服务编码点
HTTP	Hyper Text Transport Protocol	超级文本传送协议
IGMP	Internet Group Management Protocol	因特网组管理协议
IGMP Snooping	Internet Group Management Protocol Snooping	互联网组管理协议窥探
LLDP	Link Layer Discovery Protocol	链路层发现协议
MAC	Media Access Control	媒体访问控制
MIB	Management Information Base	管理信息库
NMS	Network Management Station	网络管理站
OID	Object Identifier	对象标识符
QoS	Quality of Service	服务质量
RMON	Remote Network Monitoring	远程网络监控
RSTP	Rapid Spanning Tree Protocol	快速生成树协议
SNMP	Simple Network Management Protocol	简单网络管理协议
SNTP	Simple Network Time Protocol	简单网络时间协议
SP	Strict Priority	严格优先级
STP	Spanning Tree Protocol	生成树协议
TCP	Transmission Control Protocol	传输控制协议
ToS	Type of Service	服务类型
UDP	User Datagram Protocol	用户数据报协议
VLAN	Virtual Local Area Network	虚拟局域网
WRR	Weighted Round Robin	加权轮询调度