目 录

[第一章 eNSP及VRP基础操作 1](#_Toc21531)

[1.1 认识eNSP 1](#_Toc24012)

[1.2 熟悉VRP基本操作 16](#_Toc3644)

[1.3 熟悉常用的IP相关命令 22](#_Toc25305)

[1.4 配置通过Telnet登录系统 32](#_Toc2024)

[1.5 配置通过STelnet登录系统 37](#_Toc29173)

[1.6 配置通过FTP进行文件操作 44](#_Toc8787)

[第二章 交换机基础配置 50](#_Toc5913)

[2.1 交换机基础配置 50](#_Toc29401)

[2.2 理解ARP及Proxy ARP 54](#_Toc3665)

[第三章 VLAN 64](#_Toc22769)

[3.1 VLAN基础配置及Access接口 64](#_Toc26222)

[3.2 配置Trunk接口 69](#_Toc26768)

[3.3 理解Hybrid接口的应用 75](#_Toc19021)

[3.4 利用单臂路由实现VLAN间路由 88](#_Toc29912)

[3.5 利用三层交换机实现VLAN间路由 94](#_Toc11615)

[第四章 生成树 100](#_Toc9478)

[4.1 STP配置和选路规则 100](#_Toc15692)

[4.2 配置STP定时器 109](#_Toc3576)

[4.3 RSTP基础配置 117](#_Toc18927)

[4.4 MSTP基础配置 127](#_Toc23750)

[第五章 其他交换技术 139](#_Toc7681)

[5.1 GVRP基础配置 139](#_Toc17216)

[5.2 Smart-Link与Monitor-Link 148](#_Toc22047)

[5.3 配置Eth-Trunk链路聚合 154](#_Toc7513)

[第六章 静态路由 164](#_Toc28955)

[6.1 静态路由及默认路由基本配置 164](#_Toc16053)

[6.2 浮动静态路由及负载均衡 175](#_Toc27573)

[第七章 RIP 185](#_Toc4609)

[7.1 RIP路由协议基本配置 185](#_Toc2856)

[7.2 配置RIPv2的认证 190](#_Toc16321)

[7.3 RIP路由协议的汇总 198](#_Toc21820)

[7.4 配置RIP的版本兼容、定时器及协议优先级 205](#_Toc170)

[7.5 配置RIP抑制接口及单播更新 213](#_Toc27059)

[7.6 RIP与不连续子网 221](#_Toc8141)

[7.7 RIP的水平分割及触发更新 232](#_Toc20556)

[7.8 配置RIP路由附加度量值 239](#_Toc4893)

[7.9 RIP的故障处理 245](#_Toc10160)

[7.10 RIP的路由引入 260](#_Toc1298)

[第八章 OSPF 268](#_Toc17675)

[8.1 OSPF单区域配置 268](#_Toc16503)

[8.2 OSPF多区域配置 273](#_Toc4379)

[8.3 配置OSPF的认证 280](#_Toc14251)

[8.4 OSPF被动接口配置 287](#_Toc5876)

[8.5 理解OSPF Router-ID 293](#_Toc27353)

[8.6 OSPF的DR与BDR 300](#_Toc6619)

[8.7 OSPF开销值、协议优先级及计时器的修改 308](#_Toc19695)

[8.8 连接RIP与OSPF网络 315](#_Toc29399)

[8.9 使用RIP、OSPF发布默认路由 320](#_Toc1378)

[第九章 VRRP 327](#_Toc15340)

[9.1 VRRP基本配置 327](#_Toc7423)

[9.2 配置VRRP多备份组 333](#_Toc15905)

[9.3 配置VRRP的跟踪接口及认证 339](#_Toc22106)

[第十章 基础过滤工具 343](#_Toc189)

[10.1 配置基本的访问控制列表 343](#_Toc13979)

[10.2 配置高级的访问控制列表 343](#_Toc2910)

[10.3 配置前缀列表 343](#_Toc10672)

[第十一章 广域网 343](#_Toc14807)

[11.1 WAN接入配置 343](#_Toc17722)

[11.2 PPP的认证 343](#_Toc14332)

[11.3 帧中继基本配置 343](#_Toc1959)

[11.4 OSPF在帧中继网络中的配置 343](#_Toc11456)

[第十二章 DHCP 343](#_Toc5970)

[12.1 配置基于接口地址池的DHCP 343](#_Toc17356)

[12.2 配置基于全局地址池的DHCP 343](#_Toc13855)

[12.3 配置DHCP中继 343](#_Toc25363)

[第十三章 IPv6 343](#_Toc20193)

[13.1 IPv6基础配置 343](#_Toc30763)

[13.2 RIPng基础配置 343](#_Toc10272)

[13.3 OSPFv3基础配置 343](#_Toc32498)

[第十四章 其他特性 343](#_Toc18393)

[14.1 实现eNSP与真实PC桥接 343](#_Toc16578)

[14.2 SNMP基础配置 343](#_Toc26650)

[14.3 GRE 协议基础配置 343](#_Toc21765)

[14.4 配置NAT 343](#_Toc23007)

[附录一：资源列表 343](#_Toc19254)

1. eNSP及VRP基础操作

1.1 认识eNSP

# 原理概述

eNSP(Enterprise Network Simulation Platform)是一款由华为提供的免费的、可扩展的、图形化操作的网络仿真工具平台，主要对企业网络路由器、交换机进行软件仿真，完美呈现真实设备实景，支持大型网络模拟，让广大用户有机会在没有真实设备的情况下能够模拟演练，学习网络技术。

eNSP可模拟华为企业级路由器和交换机的大部分特性，可模拟PC终端、集线器、云、帧中继交换机等。通过仿真设备配置功能，用户可以快速学习华为命令行，可通过真实网卡实现与真实网络设备的对接，并且还可以模拟接口抓包，直观感受各种协议的报文交互过程。

eNSP使用图形化操作界面，支持拓扑创建、修改、删除、保存等操作。支持设备拖拽、接口连线操作，通过不同颜色直观反映设备与接口的运行状态。eNSP还预置了大量工程案例，可直接打开演练学习。

eNSP支持单机版本和多机版本，单机部署指只在一台主机上完成组网，多机部署指Server端分布式部署在多台服务器上，多机组网场景最大可模拟200台设备组网规模。

华为完全免费对外开放eNSP，直接下载安装即可使用，无需申请license。初学者、专业人员、学生、讲师、技术人员均能免费使用。

# 实验目的

* + 认识eNSP
  + 掌握eNSP的安装方法
  + 了解eNSP的各种功能
  + 掌握使用eNSP搭建网络拓扑并进行实验的操作方法

# 实验内容

本实验将全方面介绍eNSP，从安装eNSP开始，全方位了解eNSP的功能，包括熟悉eNSP软件的主界面，掌握工具栏的使用方法，了解eNSP所支持的网络设备以及如何灵活的使用这些设备搭建网络拓扑图，模拟现实组网。

# 实验步骤

## 安装eNSP

不管是学习网络知识还是用于复现现网问题或项目交付前的预模拟等，都需要模拟组网验证，但现实中往往缺少真实设备，而通过eNSP可以很方便的组建虚拟网络，模拟现实网络环境进行实验。

在华为官方网站可以下载到最新版本的eNSP安装包，由于eNSP上每台虚拟设备都要占用一定的内存资源，所以eNSP对系统的最低配置要求为CPU双核2.0GHz或以上，内存2GB，空闲磁盘空间2GB，操作系统为Windows XP，Windows Server2003或Windows 7，在最低配置的系统环境下组网设备最大数量为10台。

安装eNSP前请先检查系统配置，确认满足最低配置后进行安装，安装步骤如下。

步骤1：双击安装程序文件，打开安装向导。

步骤2：选择“中文（简体）”，并单击“确定”，进入欢迎界面。



图1-1

步骤3：单击“下一步（N）”。



图1-2

步骤4：设置安装的目录（整个目录路径都不能包含非英文字符），单击“下一步（N）”。

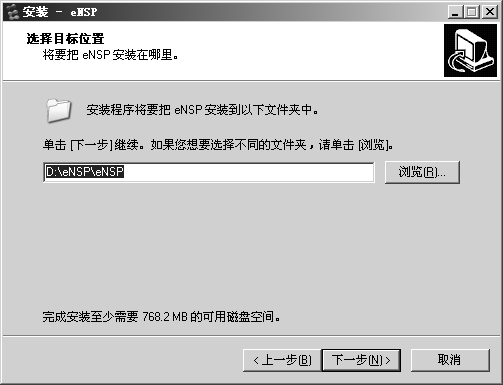


图1-3

步骤5：设置eNSP程序快捷方式在开始菜单中显示的名称，单击“下一步（N）”。

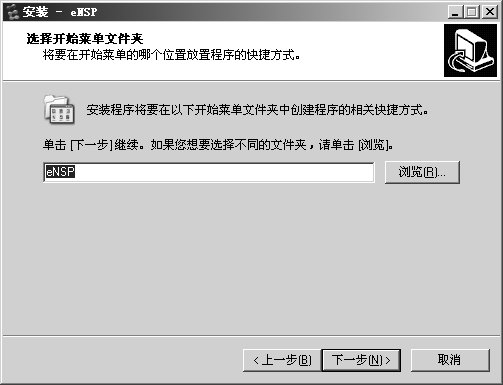


图1-4

步骤6：选择是否要在桌面创建快捷方式，单击“下一步（N）”。

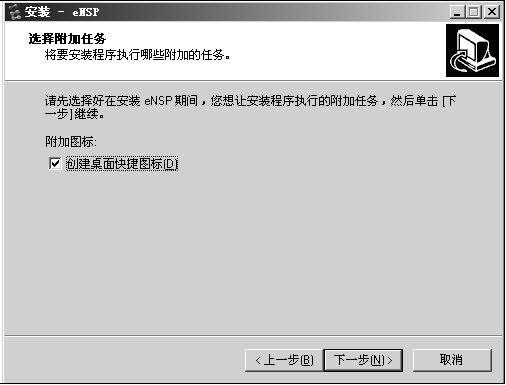


图1-5

步骤7：选择需安装的软件，注意首次安装请选择安装全部软件，单击“下一步（N）”。

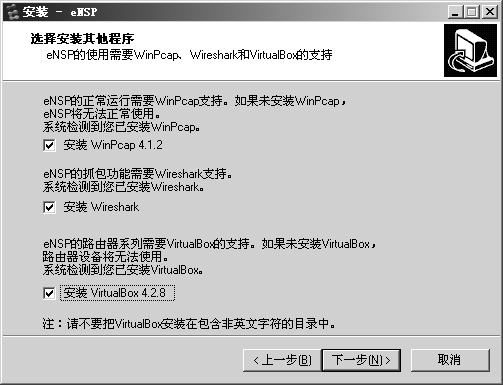


图1-6

步骤8：确认安装信息后，单击“安装（I）”开始安装。

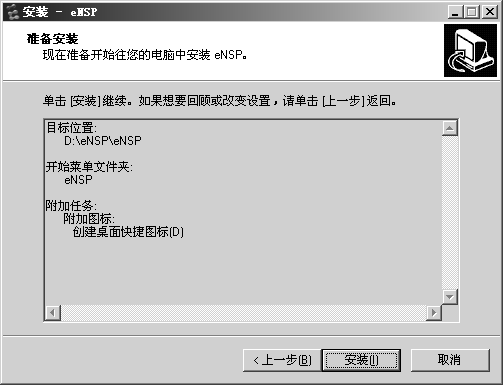


图1-7

步骤9：安装完成后，若不希望立刻打开程序，可不选择“运行 eNSP”。单击“完成（F）”结束安装。

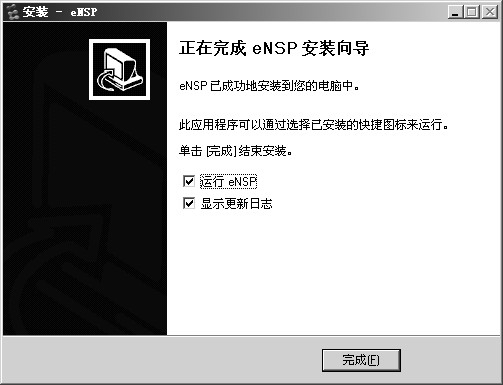


图1-8

## 熟悉eNSP界面

打开eNSP模拟器，可以看到如下主界面（如图1-9）。

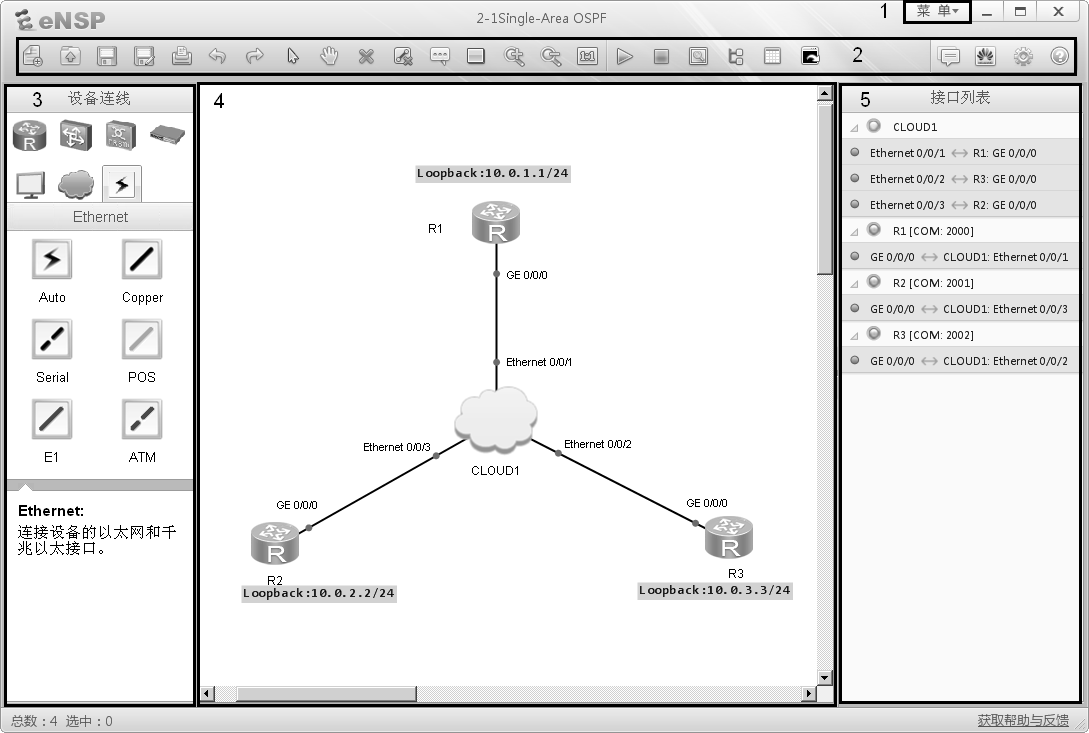


图1-9

主界面分为五大区域。区域1是主菜单，提供“文件”、“编辑”、“视图”、“工具”、“帮助”菜单：

文件菜单可进行拓扑图文件的打开、新建、保存、打印等操作；

编辑菜单可进行撤销、恢复、复制、粘贴等操作；

视图菜单可对拓扑图进行缩放和控制左右侧工具栏区的显示；工具菜单可打开调色板工具添加图形，启动或停止设备，进行数据抓包和各选项的设置；

帮助菜单可查看帮助文档，检测是否有可用更新，查看软件版本和版权信息。

进入工具菜单，点击“选项”，在弹出界面中设置软件的参数，如下图所示。

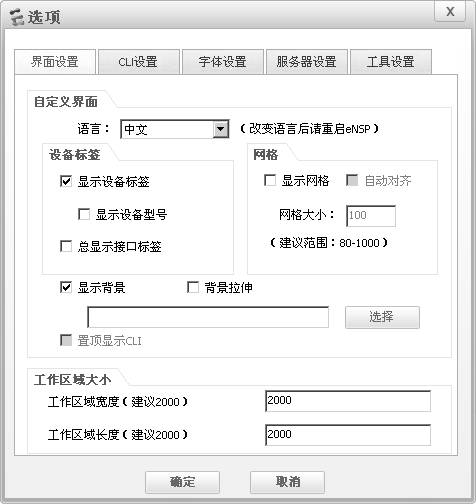


图1-10

在“界面设置”页面可以设置拓扑中的元素显示效果，比如是否显示设备标签和型号、是否显示背景图。在“工作区域大小”中可设置工作区的宽度和长度。

在“CLI 设置”页面设置命令行中信息保存方式。当选中“记录日志”时，设置命令行的显示行数和保存位置。当命令行界面内容行数超过“显示行数”中的设置值时，系统将自动保存超过行数的内容到“保存路径”中指定的位置。

在“字体设置”页面可以设置命令行界面和拓扑描述框的字体、字体颜色、背景色等参数。

在“服务器设置”页面可以设置服务器端参数，详细信息请参考[帮助](mk:@MSITStore:D:\\eNSP\\eNSP\\help\\eNSP%20帮助.chm::/ensp/ensp_help_026.html)文档。

在“工具设置”页面可以指定“引用工具”的具体路径。

区域2是工具栏，提供常用的工具，如新建拓扑、打印等。

|  |  |  |  |
| --- | --- | --- | --- |
| **工具** | **简要说明** | **工具** | **简要说明** |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\11.png | 新建拓扑 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\333.png | 添加文本 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\22.png | 打开拓扑 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\444.png | 调色板，可编辑添加各种图形 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\33.png | 保存拓扑 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\555.png | 放大 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\44.png | 另存为 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\666.png | 缩小 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\55.png | 打印拓扑 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\777.png | 恢复原大小 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\66.png | 撤销上次操作 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\888.png | 启动设备 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\77.png | 重复上次操作 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\999.png | 停止设备 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\88.png | 恢复鼠标 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\1111.png | 采集数据报文 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\99.png | 选定工作区，便于移动 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\2222.png | 显示/隐藏所有接口名称 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\111.png | 删除对象 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\3333.png | 显示网格 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\222.png | 删除所有连线 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\4444.png | 打开拓扑中所有路由器和交换机的命令行界面 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\5555.png | 华为论坛链接 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\6666.png | 华为官网链接 |
| C:\Documents and Settings\Administrator\桌面\eNSP实验图\7777.png | 选项设置 | C:\Documents and Settings\Administrator\桌面\eNSP实验图\8888.png | 帮助文档 |

在工具栏区域最右边有四个按钮，第一个是华为论坛的链接按钮，点击后可进入华为官方论坛，进行各种提问和参与讨论；第二个华为官网的链接按钮；第三个是设置按钮，可进行界面的设置，字体的设置等，与工具菜单中的选项一致；第四个是帮助文档按钮，详细介绍了当前版本的eNSP支持的所有设备特性，各种功能介绍，如何配置服务器和客户端等。

区域3是网络设备区，提供设备和网线，每种设备都有不同型号，比如点击路由器图标，设备型号区将提供AR1220、AR2220等各种路由器，供选择到工作区。

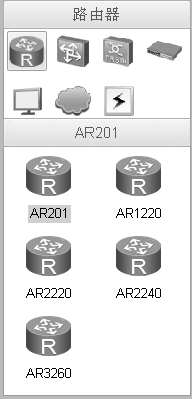


图1-11

区域4是工作区，在此区域可以灵活创建网络拓扑。

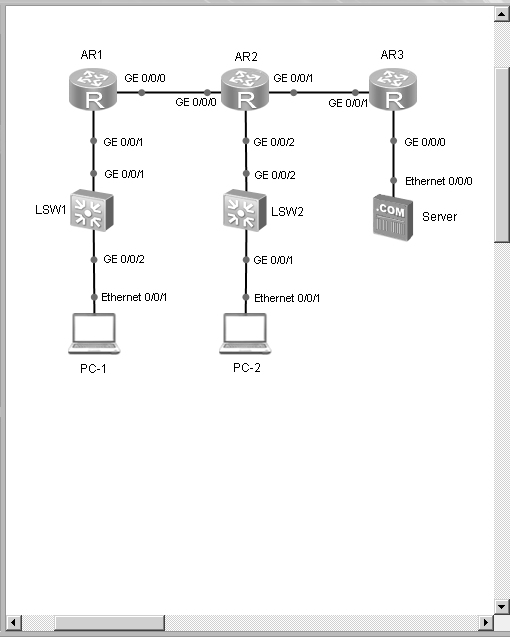


图1-12

区域5是设备接口区，显示拓扑中的设备和设备已连接的接口，可以通过观察指示灯了解接口运行状态。红色表示设备未启动或接口处于物理DOWN状态。绿色表示设备已启动或接口处于物理UP状态。蓝色表示接口正在采集报文。右键单击处于物理UP状态的接口名，可启动/停止接口报文采集。

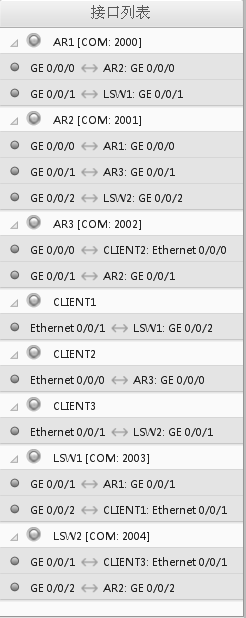


图1-13

## 网络设备配置

在eNSP中，可以利用图形化界面灵活的搭建自己需要的拓扑组网图。

(一) 选择设备。主界面左侧为可供选择的网络设备区，将需要的设备直接拖至工作区。每台设备带有默认名称，通过单击可以对其进行修改。还可以使用工具栏中文本按钮和调色板按钮的在拓扑中任意位置添加描述或图形标识。

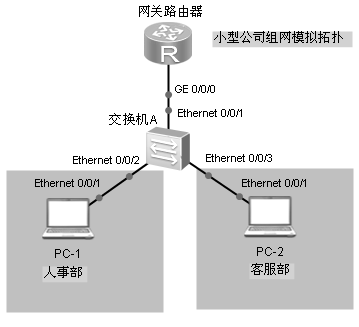


图1-14

(二) 配置设备。右键单击拓扑中的设备图标，选择“设置”，进入设备接口配置界面。

在“视图”选项卡中，可以查看到设备面板及可供使用的接口卡。如需为设备增加接口卡，可在“eNSP支持的接口卡”区域选择合适的接口卡，直接拖至上方的设备面板上相应槽位即可。如需删除某个接口卡，直接将设备面板上的接口卡拖回“eNSP支持的接口卡”区域即可。注意，只有在设备电源关闭的情况下才能进行增加或删除接口卡的操作。

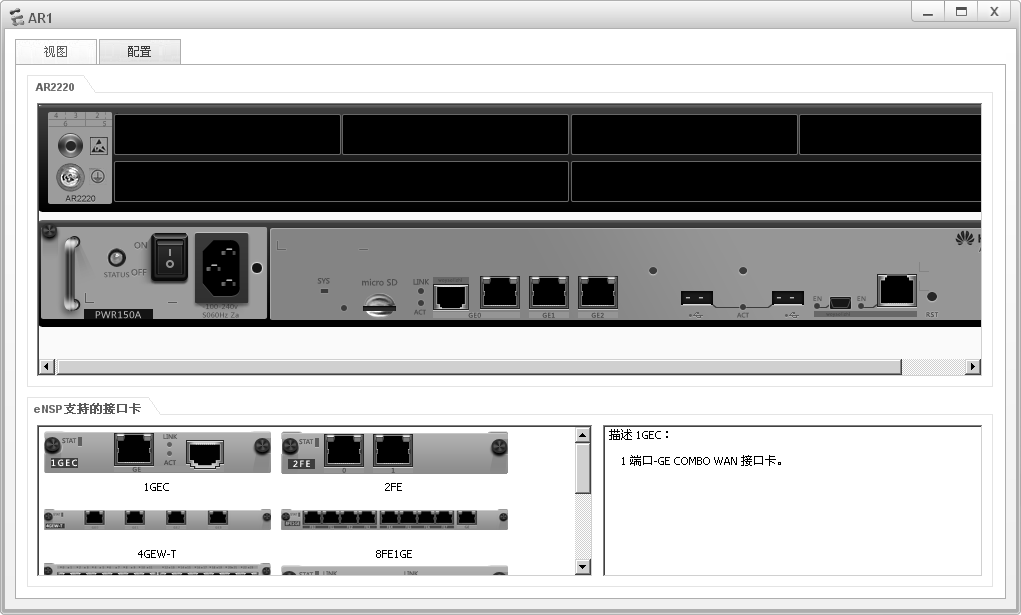


图1-15

在“配置”选项卡，可以设置设备的串口号，串口号范围在2000-65535之间，默认情况下从起始数字2000开始使用，可以自行更改串口号并点击“应用”生效。

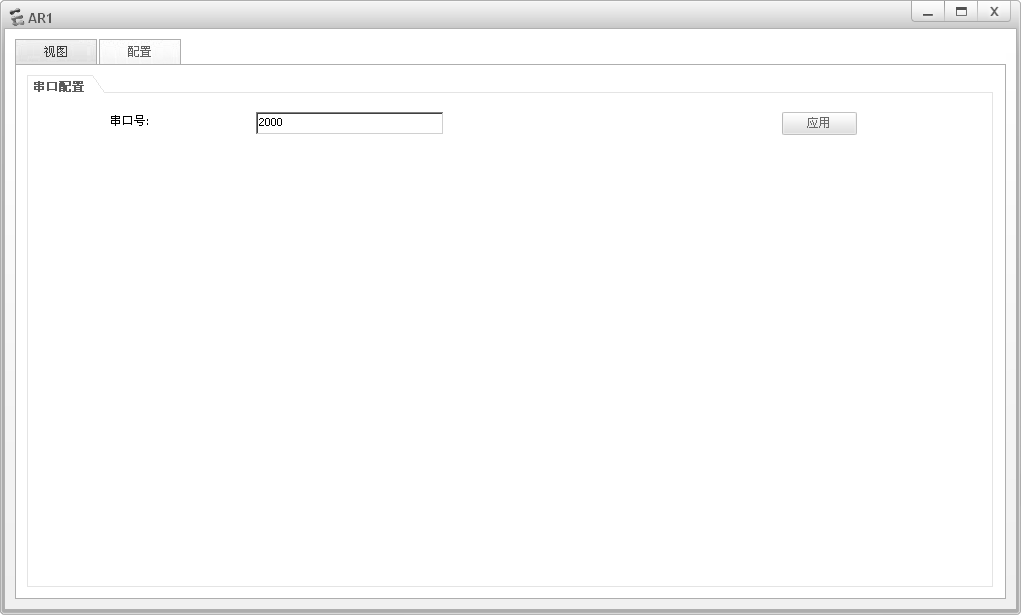


图1-16

右键点击PC，选择设置，在“基础配置”选项卡，配置设备的基础参数，如IP地址、子网掩码和MAC地址等。



图1-17

在“命令行”选项卡下可以输入**ping**命令，测试连通性。

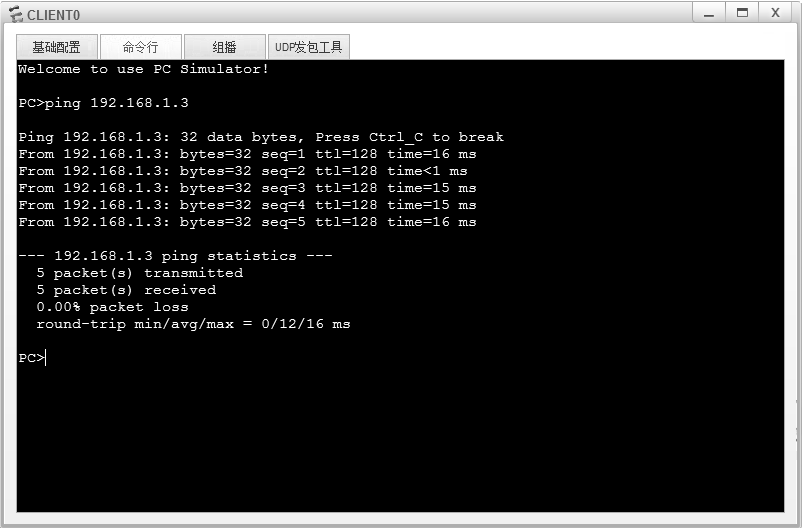


图1-18

(三) 设备连接。根据设备接口的不同可以灵活选择线缆的类型。当线缆仅一端连接了设备，而此时希望取消连接，在工作区右击或者按“Esc”键即可。选择“Auto”可以自动识别接口卡选择相应线缆。常见的如“Copper“为双绞线，”Serial“为串口线。

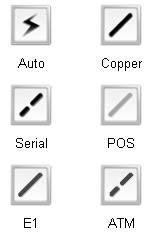


图1-19

(四) 配置导入。在设备未启动的状态下，右键单击设备，并选择“导入设备配置”，可以选择设备配置文件（.cfg或者.zip格式）并导入到设备中。

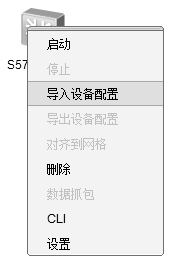


图1-20

(五) 设备启动。选中需要启动的设备后，可以通过点击工具栏中的启动设备按钮或者右键单击设备选择“启动”来启动设备。启动后，双击设备图标，通过的弹出CLI命令行界面进行配置。

(六) 设备和拓扑保存。完成配置后可以点击工具栏中的保存键保存拓扑图，并导出设备的配置文件。右键单击设备，并选择“导出设备配置”，输入设备配置文件的文件名，并将设备配置信息导出为.cfg文件。



图1-21

## 扩展功能介绍

(一) 样例加载。在工具栏中点击“打开”键，可弹出eNSP附带的实验样例，样例都是验证各种网络协议特性的典型实验案例。打开样例可以看到清晰的拓扑组网图如下。

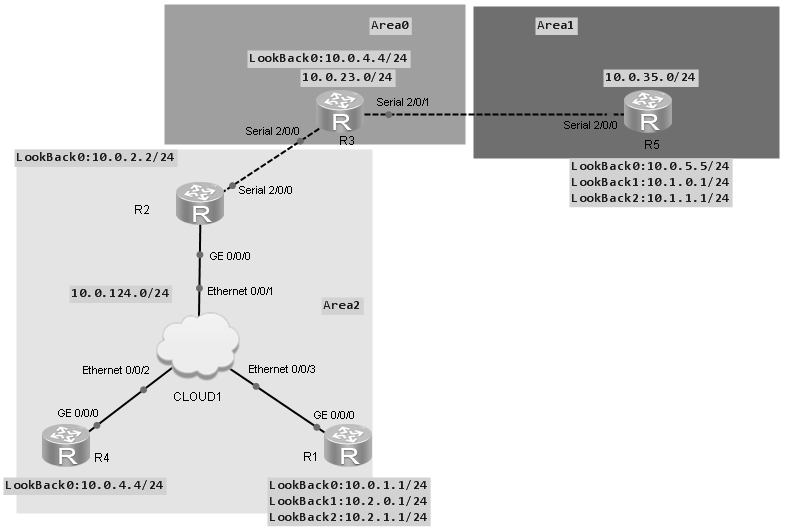


图1-22

每个样例都包括了具体的实验配置，就像是一个配置好的真实的网络环境，当运行拓扑图中所有设备后，设备会自动加载配置，用户可以在这套模拟环境中学习和验证理论知识。

(二) 数据抓包。在设备运行时，右键点击设备接口处，可选择数据抓包，通过抓包可直观感受到数据包的流动，更深刻的理解网络协议的原理，抓包相关操作在后续实验中详细介绍。

(三) 支持与真实PC进行桥接。通过虚拟设备接口与真实网卡的绑定，实现虚拟设备与真实设备的对接，后续将有专门实验介绍，或者参看帮助文档。

(四) 支持使用第三方软件登录eNSP模拟设备。在eNSP软件的接口视图中，设备名称后面将显示一个串口号。

C:\Documents and Settings\Administrator\桌面\eNSP实验图\端口号.png

图1-23

该端口号即使用第三方工具时需要设置的串口号。

1.2 熟悉VRP基本操作

# 原理概述

VRP（Versatile Routing Platform，通用路由平台）是华为公司数据通信产品的通用网络操作系统平台，拥有一致的网络界面、用户界面和管理界面。在VRP操作系统中用户通过命令行对设备下发各种命令来实现对设备的配置与日常维护操作。

用户登录到路由器后出现命令行提示符后，即进入命令行接口CLI（Command Line Interface），命令行接口是用户与路由器进行交互的常用工具。

当用户输入命令时，如果不记得此命令的关键字或参数，可以使用命令行的帮助获取全部或部分关键字和参数的提示。用户也可以通过使用系统快捷键完成对应命令的输入，简化操作。在首次登录设备时，用户可根据需要，完成设备的基本配置，如设备名称的修改，时钟的配置以及标题文本的设置。

# 实验目的

* + 熟悉VRP的基本操作
  + 掌握命令行视图的切换
  + 掌握命令行帮助和快捷键的使用
  + 掌握修改设备名称和设置时钟的方法
  + 掌握设置标题信息的方法
  + 掌握查看路由器基本信息的方法

# 实验内容

本实验模拟用户首次使用VRP操作系统，在登录路由器后使用命令行来配置设备，熟悉命令行视图的切换，命令行帮助和快捷键的使用，并完成设备的基本配置，包括修改路由器名称，配置路由器时钟，设置标题文本以及使用命令行查看路由器基本信息等。

# 实验拓扑



图1-24 熟悉VRP基本操作拓扑图

# 实验步骤

## 命令视图切换

启动设备，登录设备成功后，即进入用户视图。

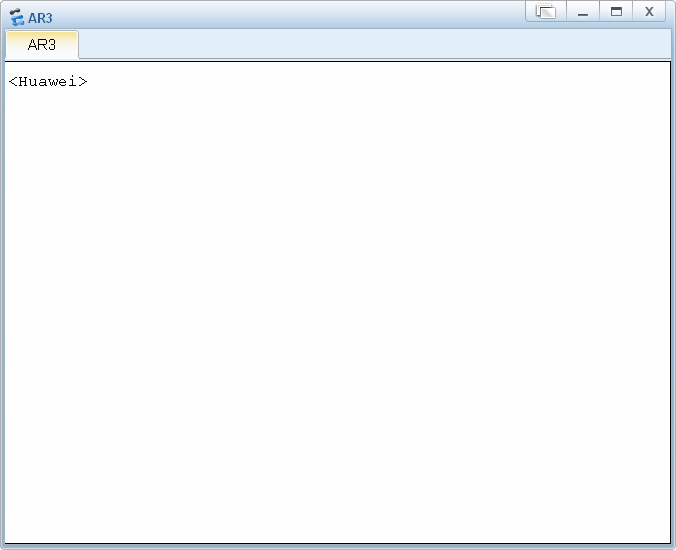


图1-25

用户视图下只能使用参观和监控级命令，如使用**display version**命令显示系统软件版本及硬件等信息。

<Huawei>display version

Huawei Versatile Routing Platform Software

VRP (R) software, Version 5.130 (AR2200 V200R001C01SPC300)

Copyright (C) 2011 HUAWEI TECH CO., LTD

Huawei AR2220 Router uptime is 0 week, 0 day, 0 hour, 2 minutes

BKP 0 version information:

……output omit……

可以观察到VRP操作系统的版本，设备的型号，启动时间等信息。

在用户视图下使用**system-view**命令可以切换到系统视图。在系统视图下可以配置接口、协议等。从系统视图使用**quit**命令可以切换到用户视图。

<Huawei>system-view

Enter system view, return user view with Ctrl+Z.

[Huawei]quit

<Huawei>

在系统视图下使用相应命令进入其他视图，如使用**interface**命令进入接口视图，在接口视图下可以使用**ip address**命令配置接口IP地址，子网掩码。为路由器的GE0/0/0接口配置IP地址，配置IP地址时可以使用子网掩码长度，也可以使用完整的子网掩码，如掩码为255.255.255.0，也可以使用24替代，上述配置以24替代。

<Huawei>system-view

Enter system view, return user view with Ctrl+Z

[Huawei]interface GigabitEthernet 0/0/0

[Huawei-GigabitEthernet0/0/0]ip address 10.1.1.1 24

配置完成后，可以使用命令**return**直接退回到用户视图。

[Huawei-GigabitEthernet0/0/0]return

<Huawei>

命令**return**可以使用户从任意非用户视图退回到用户视图，命令**return**的功能也可以用组合键<**Ctrl+Z**>完成。

## 命令行帮助

如果用户忘记命令的参数或关键字，可使用命令行在线帮助。命令行在线帮助分为完全帮助和部分帮助。

(一) 完全帮助，为在任一命令视图下，键入“？”获取该命令视图下所有的命令及其简单描述。

如在系统视图下，键入“？”获取该命令视图下所有的命令及其简单描述。

[Huawei]?

System view commands:

aaa <Group> aaa command group

aaa-authen-bypass Set remote authentication bypass

aaa-author-bypass Set remote authorization bypass

aaa-author-cmd-bypass Set remote command authorization bypass

access-user User access

acl Specify ACL configuration information

alarm Alarm

……

也可以键入一命令，后接以空格分隔的“？”，列出全部关键字或参数及其简单描述。

如在系统视图下，列出**interface**命令参数及其简单描述。

[Huawei]interface ?

Bridge-if Bridge-if interfac。

Cellular Cellular interface

Dialer Dialer interface

Eth-Trunk Ethernet-Trunk interface

GigabitEthernet GigabitEthernet interface

……

(二) 部分帮助，为键入一字符串，其后紧接“？”，列出以该字符串开头的所有关键字。

如在系统视图下列出以“rou” 字符串开头的所有命令及其简单描述。

[Huawei]rou?

route Routing Module

route-policy Route-policy

route-policy-change Specify route policy change parameter

router Configure router information

## 快捷键使用

命令行接口提供了基本的命令编辑功能，支持多行编辑，每条命令的最大长度为256个字符。各功能键详细描述如下所示：

退格键BackSpace表示删除光标位置的前一个字符；

左光标键←或<Ctrl+B> 表示光标向左移动一个字符位置；

右光标键→或<Ctrl+F>表示光标向右移动一个字符位置；

删除键Delete 表示删除光标位置字符；

上下光标键↑↓表示显示历史命令；

当用户输入不完整的关键字后按下Tab键，系统自动执行部分帮助，将命令补全。比如键入“dis”后，按键盘“Tab”键可以将命令补全为“display”。

<R1>dis

<R1>display

可以通过**display hotkey**命令用来查看已定义、未定义和系统保留的快捷键的情况。

## 修改路由器名称

当网络上有多个设备需要管理时，用户可以为每个设备设置特定的名称，以便于管理和识别。

在系统视图下，使用命令**sysname**修改当前路由器名称。如更改当前路由器的系统名称为R1。

[Huawei]sysname R1

[R1]

## 设置路由器时钟

为了保证网络中的设备有准确的时钟信号，用户需要准确设置设备的系统时钟。

使用命令**[clock datetime](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/clock_datetime.html)**，设置当前时间和日期，使用命令**[clock timezone](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/clock_timezone.html)**，设置所在的时区。

在用户视图下使用**clock datetime**命令修改系统日期和时间为2011年9月15日12时。

<R1>clock datetime 12:00:00 2011-09-15

在用户视图下使用命令**clock timezone**，设置所在的时区为北京。

<R1>clock timezone BJ add 08:00:00

注意，系统默认是伦敦时间，北京处于+8时区，时间偏移量增加了8，在配置时，需要加上偏移量8，才能得到预期的北京时区。

## 设置标题信息

如果需要对登录路由器的用户提供警示或说明信息，可以设置登录时或登录成功后的标题信息。

使用命令[**header**](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/header.html) **login**，设置登录时的标题文本为hello。使用命令**[header](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/header.html)** **shell**，设置登录成功后的标题文本信息为“Welcome to Huawei certification lab”。

[R1]header login information "hello"

[R1]header shell information "Welcome to Huawei certification lab"

其中login参数为当用户在登录路由器认证过程中，激活终端连接时显示的标题信息，是用户在连接到路由器并进行登录验证以及开始配置时，系统所显示的一段提示信息，当需要为用户登录提供明确的提示信息时，可以使用此配置。Shell参数为当用户成功登录到路由器上，已经建立了会话时显示的标题信息。

配置完成后，尝试退出路由器命令行界面重新登录。

[R1]quit

<R1>quit

Configuration console exit, please retry to log on

Password:

Welcome to Huawei certification lab

<R1>

可以观察到欢迎信息。

注意，通常情况下，登陆标语信息用于警告非法登陆或者说明信息。

## 查看路由器基本信息

使用相应命令查看路由器基本信息或运行状态。

使用命令**display version**查看路由器信息。

<R1>display version

Huawei Versatile Routing Platform Software

VRP (R) software, Version 5.130 (AR2200 V200R003C00)

Copyright (C) 2011-2012 HUAWEI TECH CO., LTD

Huawei AR2220 Router uptime is 0 week, 0 day, 0 hour, 17 minutes

BKP 0 version information:

……

可以观察到VRP操作系统的版本，设备的型号，启动时间等信息。

使用**display current-configuration**命令查看路由器当前配置。

<R1>display current-configuration

[V200R003C00]

#

sysname R1

header shell information "Welcome to Huawei certification lab"

header login information "hello"

#

snmp-agent local-engineid 800007DB03000000000000

snmp-agent

#

clock timezone BJ add 08:00:00

clock daylight-saving-time Day Light Saving Time repeating 12:32 9-1 12:32 11-23 00:00 2005 2005

#

……

#

Return

可以观察到所有路由器的已配置信息。

使用**display interface** **GigabitEthernet 0/0/0**命令查看路由器GE 0/0/0接口的状态信息。

[R1]display interface GigabitEthernet 0/0/0

GigabitEthernet0/0/0 current state : DOWN

Line protocol current state : DOWN

Description:HUAWEI, AR Series, GigabitEthernet0/0/0 Interface

Route Port,The Maximum Transmit Unit is 1500

Internet Address is 10.1.1.1/24

IP Sending Frames' Format is PKTFMT\_ETHNT\_2, Hardware address is 00e0-fc03-3a27

……

可以观察到该接口的物理状态、接口IP地址以及其它的统计信息。

1.3 熟悉常用的IP相关命令

# 原理概述

华为设备支持多种配置方式，包括web界面管理等。但是作为一名网络工程师，必须熟悉使用命令行的方式进行设备管理。在工作中，对路由器和交换机最常用的就是IP相关命令，如配置主机名，IP地址，测试IP数据包连通性等。这些命令是基本的配置和测试命令。

# 实验目的

* + 掌握路由器命名的方法
  + 掌握配置路由器IP地址方法
  + 掌握测试IP地址连通性的方法
  + 掌握查看设备配置的方法
  + 掌握抓包的方法

# 实验内容

本实验模拟简单的企业网络场景，某公司购买了新的路由器和交换机。交换机S1连接客服部PC-1，S2连接市场部PC-2，路由器R1连接S1和S2两台交换机。网络管理员需要首先熟悉设备的使用，包括基础的IP配置和查看命令。

# 实验拓扑

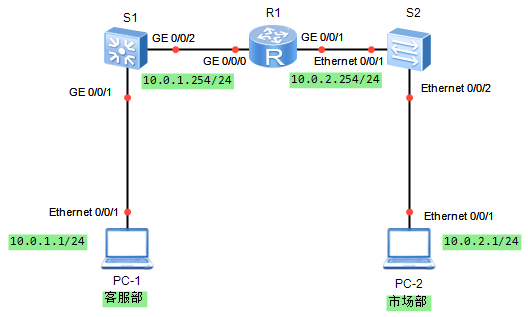


图1-26 熟悉常用的IP相关命令拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | 10.0.1.254 |
| PC-2 | Ethernet 0/0/1 | 10.0.2.1 | 255.255.255.0 | 10.0.2.254 |
| R1(AR2220) | GE 0/0/0 | 10.0.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.2.254 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表，使用图形化界面配置PC的IP地址，以客服部PC-1为例，如下图。

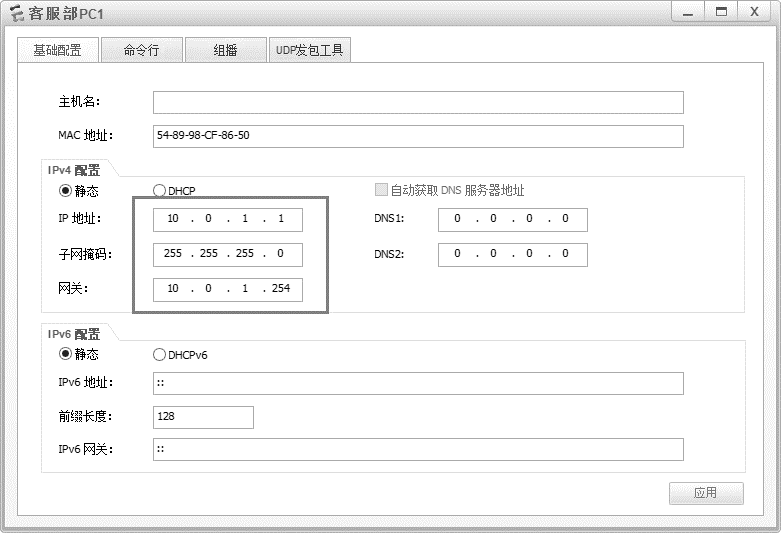


图1-27

PC-1设置完成后，继续设置市场部PC-2。



图1-28

配置路由器的主机名，打开R1的命令行界面，即进入用户视图，在用户视图下，用户可以完成查看运行状态和统计信息等功能。此时屏幕上显示如下。

<Huawei>

路由器主机名为默认的主机名Huawei。要更改主机名，必须进入系统视图模式，在系统视图下，用户可以配置系统参数以及通过该视图进入其他的功能配置视图。使用**system-view**命令，进入系统视图。

<Huawei>system-view

[Huawei]

这时图标由<Huawei>变成了[Huawei]，表示进入了系统视图模式。

在系统视图下，使用**sysname**命令修改设备主机名为R1。

[Huawei]sysname R1

可以观察到，主机名由原来的[Huawei]变成了[R1]，表示主机名修改成功。使用命令**quit**可以退出当前模式。

[R1]quit

输入**quit**命令后，可以观察到[R1]标志已经变成<R1>，表明已经成功退到用户视图。在用户视图下可以使用**save**命令保存当前配置。

<R1>save

这时会提示是否继续保存，输入**y**确认保存动作。

<R1>save

The current configuration will be written to the device.

Are you sure to continue? (y/n)[n]:y

It will take several minutes to save configuration file, please wait........

Configuration file had been saved successfully

Note: The configuration file will take effect after being activated

出现以上信息表示保存成功。

## 配置路由器接口IP地址

在系统视图下进入接口视图，在该视图下可以配置接口相关的物理属性、链路层特性及IP地址等重要参数。使用**interface** 命令进入路由器相应的接口视图GE 0/0/0 。

[R1]interface GigabitEthernet 0/0/0

在路由器的接口视图下配置路由器接口IP地址和掩码。注意，华为设备上的物理接口默认都处于开启状态。

[R1-GigabitEthernet0/0/0]ip address 10.0.1.254 255.255.255.0

配置完成后，使用**display ip interface brief**命令查看接口与IP相关摘要信息。

[R1-GigabitEthernet0/0/0]display ip interface brief

\*down: administratively down

^down: standby

(l): loopback

(s): spoofing

The number of interface that is UP in Physical is 3

The number of interface that is DOWN in Physical is 1

The number of interface that is UP in Protocol is 2

The number of interface that is DOWN in Protocol is 2

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 10.0.1.254/24 up up

GigabitEthernet0/0/1 unassigned up down

GigabitEthernet0/0/2 unassigned down down

NULL0 unassigned up up(s)

可以观察到，路由器接口GE 0/0/0的IP地址已经配置完成，“Physical”为UP，即接口的物理状态处于正常启动的状态，“Protocol”为UP，即接口的链路协议状态处于正常启动的状态。

同理配置路由器GE 0/0/1的IP地址。如果在配置过程中对命令非常熟悉，可以使用简写的方式配置。

<R1>system-view

[R1]int g0/0/1

[R1-GigabitEthernet0/0/1]ip add 10.0.2.254 24

注意，即便是简写，也要保证所输入的命令关键字是唯一的，否则不会成功。

如果忘记命令，可以输入“？”查看相关命令。如果输入命令首部部分，可以使用TAB键选择性补齐命令。

[Huawei]inter?

interface Specify the interface configuration view

[Huawei]inter

[Huawei]interface

配置完成后，再次确认接口与IP相关摘要信息。

<R1>display ip interface brief

\*down: administratively down

^down: standby

(l): loopback

(s): spoofing

The number of interface that is UP in Physical is 3

The number of interface that is DOWN in Physical is 1

The number of interface that is UP in Protocol is 3

The number of interface that is DOWN in Protocol is 1

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 10.0.1.254/24 up up

GigabitEthernet0/0/1 10.0.2.254/24 up up

GigabitEthernet0/0/2 unassigned down down

NULL0 unassigned up up(s)

可以观察到，路由器的GE 0/0/0与GE 0/0/1的接口IP地址已经配置完成。物理接口工作正常，“Physical”为UP，即接口的物理状态处于正常启动的状态，“Protocol”为UP，即接口的链路协议状态处于正常启动的状态。

## 查看路由器配置信息

经过以上步骤的配置，路由器接口的IP地址已经配置完成，可以使用命令**display ip routing-table**查看IPv4路由表的信息。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 Direct 0 0 D 10.0.1.254 GigabitEthernet0/0/0

10.0.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

10.0.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

10.0.2.0/24 Direct 0 0 D 10.0.2.254 GigabitEthernet0/0/1

10.0.2.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

10.0.2.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，路由器R1在GE 0/0/0接口上直连了一个10.0.1.0/24的网段，在G0/0/1接口上直连了一个10.0.2.0/24的网段。

其中“Route Flags”为路由标记，“R”表示该路由是迭代路由，“D”表示该路由下发到FIB表。“Routing Tables：Public”表示此路由表是公网路由表，如果是私网路由表，则显示私网的名称，如Routing Tables: ABC。“Destinations”表示显示目的网络/主机的总数。“Routes”表示显示路由的总数。“Destination/Mask”表示显示目的网络/主机的地址和掩码长度。“Proto”表示接收此路由的路由协议，“Direct”表示直连路由。“Pre”表示此路由的优先级。“Cost”表示此路由的路由开销值。

“Next Hop”表示此路由的下一跳地址。“Interface”表示此路由下一跳的出接口。

使用**Ping**命令测试路由器R1与PC间的连通性，以测试去往PC-1的连通性为例。

<R1>ping 10.0.1.1

PING 10.0.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.0.1.1: bytes=56 Sequence=1 ttl=128 time=200 ms

Reply from 10.0.1.1: bytes=56 Sequence=2 ttl=128 time=70 ms

Reply from 10.0.1.1: bytes=56 Sequence=3 ttl=128 time=40 ms

Reply from 10.0.1.1: bytes=56 Sequence=4 ttl=128 time=1 ms

Reply from 10.0.1.1: bytes=56 Sequence=5 ttl=128 time=10 ms

--- 10.0.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/64/200 ms

显示为上述结果，则表明测试连通性正常。

<R1>ping 10.0.1.1

PING 10.0.1.1: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

--- 10.0.1.1 ping statistics ---

5 packet(s) transmitted

0 packet(s) received

100.00% packet loss

显示为上述结果，则表示连通性测试失败，即R1与PC-1连通性异常。

直连网段连通性测试完毕后，测试非直连设备的连通性，即PC-1与PC-2的连通性。双击点开设备配置界面，选择命令行，此命令行如同PC的DOS窗口一样，可执行基本命令，如下图。

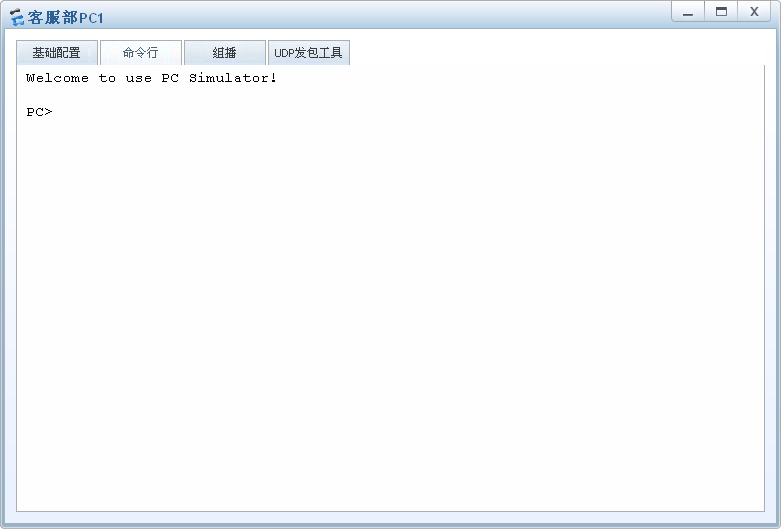


图1-29

测试PC-1到PC-2间的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.2.1: bytes=32 seq=1 ttl=127 time=47 ms

From 10.0.2.1: bytes=32 seq=2 ttl=127 time=47 ms

From 10.0.2.1: bytes=32 seq=3 ttl=127 time=47 ms

From 10.0.2.1: bytes=32 seq=4 ttl=127 time=62 ms

From 10.0.2.1: bytes=32 seq=5 ttl=127 time=32 ms

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 32/47/62 ms

可以观察到PC-1与PC-2之间能正常通信。

## 使用抓包工具

以抓取R1上GE 0/0/0接口的数据包为例，在R1与S1的直连链路上，将鼠标移动到接口GE 0/0/0上，点击鼠标右键选择开始抓包。

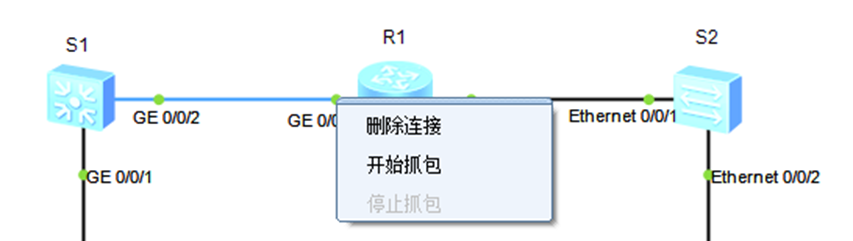


图1-30

这时会显示出解包的结果，如下图。



图1-31

双击数据包可查看详细的数据包内容。

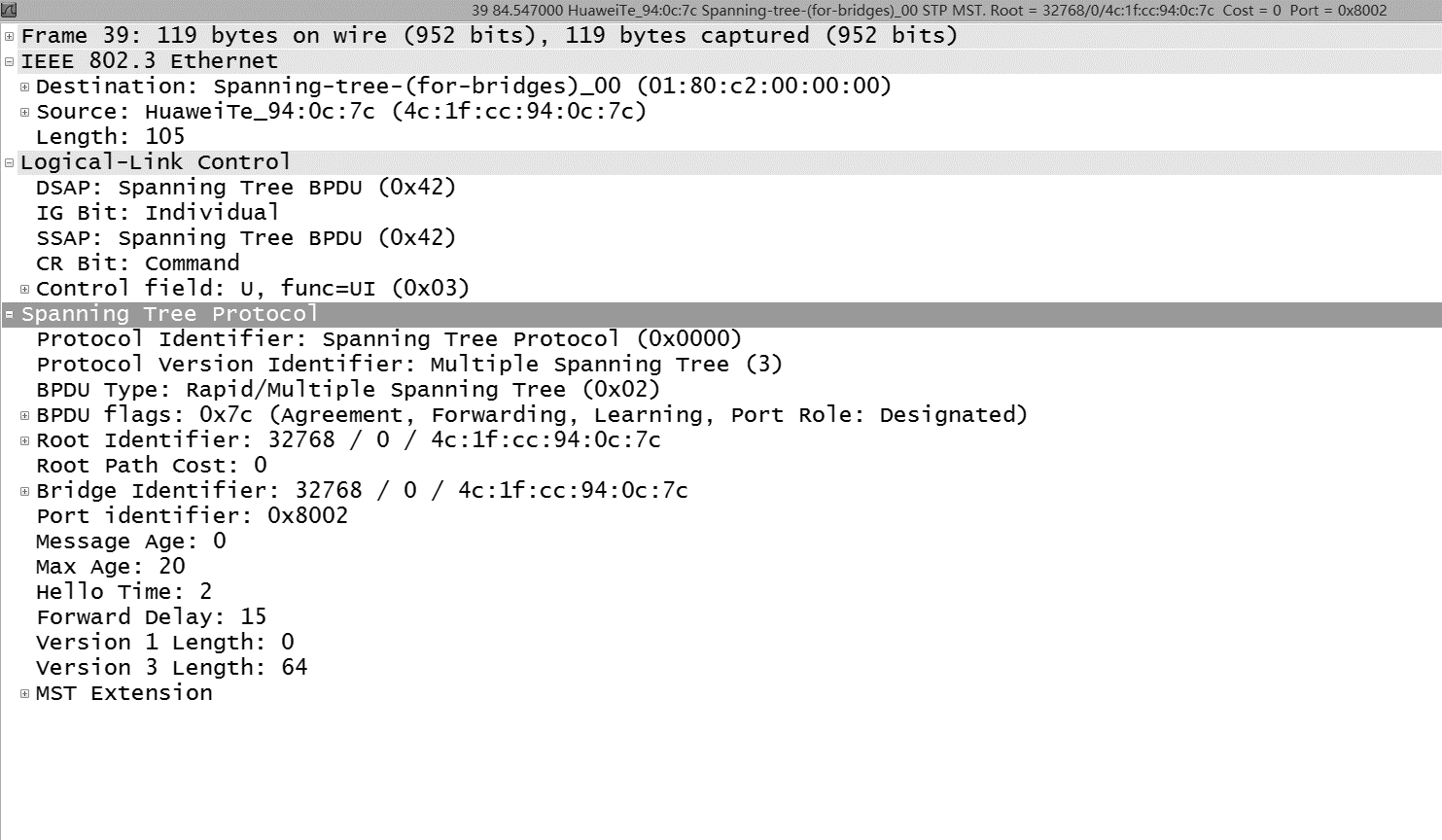


图1-32

如果不需要继续抓包，可点击停止抓包。

还可以采取另一种方式来抓包，效果一样。

点击上方快捷方式中的抓包按钮，这时会出现当前可抓取的接口列表。



图1-33

以抓取R1上GE 0/0/0接口的数据包为例，点击接口后选择开始抓包。

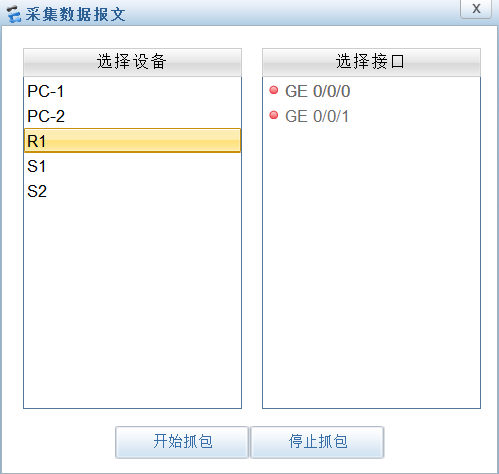


图1-34

这时会显示出解包的结果，如下图。



图1-35

双击数据包可查看详细的数据包内容。

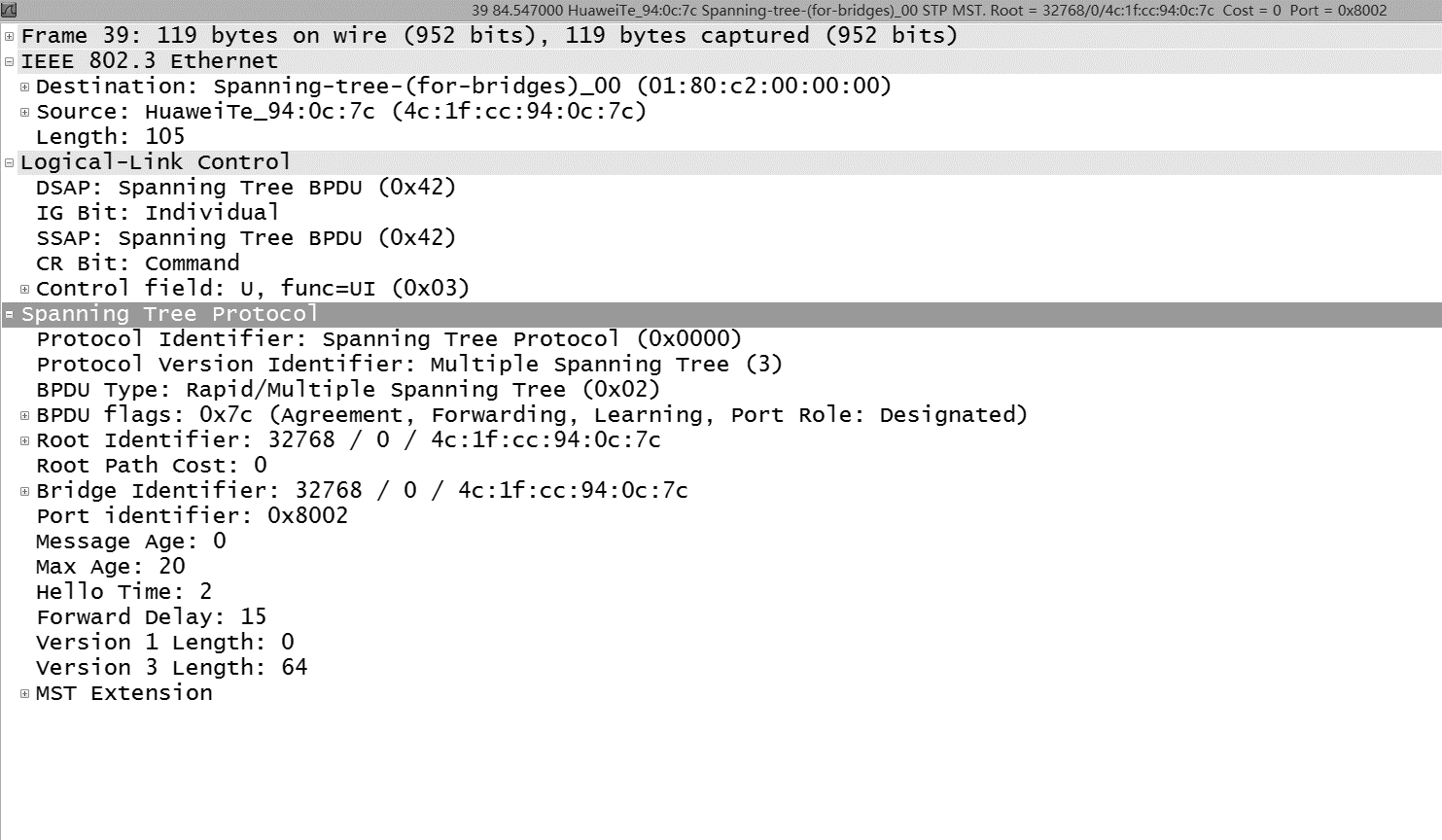


图1-36

如果不需要继续抓包，可点击“停止抓包”。

# 思考

管理员要经常在路由器上使用命令“display ip interface brief”查看接口状态，但该命令完整输入则较长，思考如何使用最简化且准确的方式输入这条命令？

1.4 配置通过Telnet登录系统

# 原理概述

Telnet（Telecommunication Network Protocol）起源于ARPANET，是一种最早的Internet应用，Telnet提供了一种通过终端远程登录到服务器的方式，呈现一个交互式操作界面。用户可以先登录到一台主机或设备，再通过Telnet的方式远程登录到网络上的其他主机或设备上去，然后便可对该设备进行配置和管理，而不需要为每一台设备都直接连接一个管理终端。

Telnet通常用在远程登录应用中，以便对本地或远端运行的网络设备进行配置、监控和维护。如网络中有多台设备需要配置和管理，用户无需为每一台设备都连接一个用户终端进行本地配置，可以通过Telnet方式在一台设备上对多台设备进行管理或配置。如果网络中需要管理或配置的设备不在本地时，也可以通过Telnet方式实现对网络中设备的远程维护，极大地提高了用户操作的灵活性。

# 实验目的

* + 理解Telnet的应用场景
  + 掌握Telnet的基本配置
  + 掌握Telnet密码验证的配置
  + 掌握Telnet用户级别的修改方法

# 实验内容

本实验模拟公司网络场景，路由器R1是公司机房的一台设备，公司员工的办公区与机房不在同一个楼层，路由器R2和R3模拟员工主机，通过交换机S1与机房设备相连。为了方便用户的管理，现需要在路由器R1上配置Telnet使用户能在办公区远程管理机房设备。为了提高安全性，Telnet时需要使用密码认证，只有网络管理员能对设备进行配置和管理，普通用户仅能监控设备。

# 实验拓扑

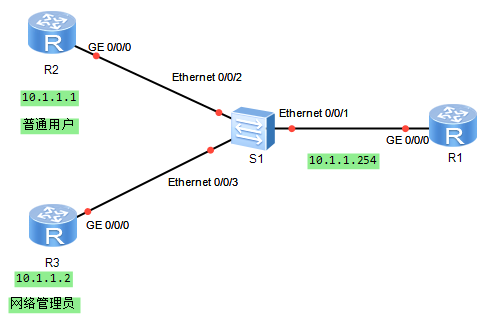


图1-37配置通过Telnet登录系统拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 10.1.1.254 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.1.1.1 | 255.255.255.0 | 10.1.1.254 |
| R3(AR2220) | GE 0/0/0 | 10.1.1.2 | 255.255.255.0 | 10.1.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

这里以用户主机和默认网关间的连通性为例。

R2>ping 10.1.1.254

Ping 10.1.1.254: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.254: bytes=32 seq=1 ttl=255 time=47 ms

From 10.1.1.254: bytes=32 seq=2 ttl=255 time=32 ms

From 10.1.1.254: bytes=32 seq=3 ttl=255 time=47 ms

From 10.1.1.254: bytes=32 seq=4 ttl=255 time=31 ms

From 10.1.1.254: bytes=32 seq=5 ttl=255 time=31 ms

--- 10.0.12.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/68/120 ms

## 配置Telnet的密码验证

为了方便公司员工对机房设备进行远程管理和维护，需要在路由器上配置Telnet功能。为了提高网络安全性，用户在Telnet时使用密码认证，只有通过认证的用户才有权限登录设备。

在R1上配置Telnet验证方式为密码验证方式，密码为huawei。并设置验证密码以密文方式存储，在配置文件中以加密的方式显示密码，能够使密码不容易被泄露。

[R1]user-interface vty 0 4 //使用Telnet登录的用户数



[R1-ui-vty0-4]authentication-mode password

Please configure the login password (maximum length 16):huawei

在用户设备R2和R3上使用Telnet连接R1。

<PC-1>telnet 10.1.1.254 //R2

Trying 10.1.1.254 ...

Press CTRL+K to abor

Connected to 10.1.1.254 ...

Login authentication

Password:

Info: The max number of VTY users is 10, and the number

of current VTY users on line is 1.

The current login time is 2013-06-25 13:56:34.

<R1>

<R3>telnet 10.1.1.254

Trying 10.1.1.254 ...

Press CTRL+K to abort

Connected to 10.1.1.254 ...

Login authentication

Password:

Info: The max number of VTY users is 10, and the number

of current VTY users on line is 1.

The current login time is 2013-06-25 19:08:01.

<R1>

可以观察到此时R2和R3在连接R1过程中，要求输入认证密码，当输入正确的密码后才能进入R1的用户界面。

登录成功后，可以继续使用命令**display users**查看已经登录的用户信息。

[R1]display users

User-Intf Delay Type Network Address AuthenStatus AuthorcmdFlag

+ 0 CON 0 00:00:00 no Username : Unspecified

34 VTY 0 00:01:26 TEL 10.1.1.2 pass no U

sername : Unspecified

35 VTY 1 00:00:30 TEL 10.1.1.1 pass no U

sername : Unspecified

## 配置Telnet区分不同用户的权限

为了进一步保证网络的安全性及稳定性，避免员工错误更改设备的配置，公司要求普通员工只能拥有设备的监控权限，只有网络管理员拥有设备的配置和管理权限。默认情况下，VTY用户界面的用户级别为0（参观级），只能使用**ping**、**tracert**等这些网络诊断命令。

在R1上配置Telnet的用户级别为1（监控级）。普通员工仅使用密码登录设备，只能使用**display**等命令监控设备。

[R1]user-interface vty 0 4

[R1-ui-vty0-4]authentication-mode password

[R1-ui-vty0-4]set authentication password cipher huawei

[R1-ui-vty0-4]user privilege level 1

配置完成后，将R2模拟成普通用户设备，测试到R1的Telnet连接。

<R2>telnet 10.1.1.254

Trying 10.1.1.254 ...

Press CTRL+K to abort

Connected to 10.1.1.254 ...

Login authentication

Password:

<R1>

<R1>system-view

^

Error: Unrecognized command found at '^' position.

可以观察到，此时输入正确的密码后即可进入R1的用户视图，但是试图进入R1的系统视图的时候被拒绝了，这是因为用户级别不够，所以无法执行更高一级的命令。

管理员使用自己单独的用户名和密码登录设备，拥有设备的配置和管理权限。这里要将VTY用户界面的认证模式修改成AAA认证，这样才能使用本地的用户名和密码进行认证，默认情况下设备的AAA认证功能是开启的，所以只需要为管理员在本地配置相应的用户名和密码即可。为管理员创建用户名为admin，密码为hello的本地用户，用户级别为3（管理级）。

进入AAA视图下配置本地用户名admin和密文密码hello，并且将该用户的用户级别修改为3。

[R1]aaa

[R1-aaa]local-user admin password cipher hello privilege level 3

配置该用户的接入类型为Telnet。

[R1-aaa]local-user admin service-type telnet

接下来进入VTY用户界面视图下，将认证模式改成AAA。

[R1]user-interface vty 0 4

[R1-ui-vty0-4]authentication-mode aaa

将R3模拟成管理员用户设备，测试到R1的Telnet连接。

<R3>telnet 10.1.1.254

Trying 10.1.1.254 ...

Press CTRL+K to abort

Connected to 10.1.1.254 ...

Login authentication

Username:admin

Password:

Info: The max number of VTY users is 10, and the number

of current VTY users on line is 1.

The current login time is 2013-06-24 12:28:38.

<R1>

<R1>system-view

Enter system view, return user view with Ctrl+Z.

[R1]

可以观察到，此时在连接R1时需要同时输入用户名和密码进入认证。输入正确的用户名和密码后即可进入R1的用户视图下，且可以使用命令**system-view**进入到R1的系统视图下，从而可以对R1进行所有相关的配置和管理操作。

# 思考

Telnet是基于TCP协议还是UDP协议的应用？为什么？

Telnet应用安全吗？为什么？

1.5 配置通过STelnet登录系统

# 原理概述

由于Telnet缺少安全的认证方式，而且传输过程采用TCP进行明文传输，存在很大的安全隐患，单纯提供Telnet服务容易招致主机IP地址欺骗、路由欺骗等恶意攻击。传统的Telnet和FTP等通过明文传送密码和数据的方式，已经慢慢不被接受。

SSH（Secure Shell）是一个网络安全协议，通过对网络数据的加密，使其能够在一个不安全的网络环境中，提供安全的远程登录和其他安全网络服务。SSH（Secure Shell）特性可以提供安全的信息保障和强大的认证功能，以保护路由器不受诸如IP地址欺诈、明文密码截取等攻击。由于SSH数据加密传输，认证机制更加安全，SSH已经越来越被广泛使用，成为了当前重要的网络协议之一。

SSH基于TCP协议22端口传输数据，支持Password认证。客户端向服务器发出password认证请求，将用户名和密码加密后发送给服务器；服务器将该信息解密后得到用户名和密码的明文，与设备上保存的用户名和密码进行比较，并返回认证成功或失败的消息。

STelnet是Secure Telnet的简称，在一个传统不安全的网络环境中，服务器通过对客户端的认证及双向的数据加密，为网络终端访问提供了安全的Telnet服务。

SFTP是SSH File Transfer Protocol的简称，在一个传统不安全的网络环境中，服务器通过对客户端的认证及双向的数据加密，为网络文件传输提供了安全的服务。

# 实验目的

* + 理解SSH的应用场景
  + 理解SSH协议的原理
  + 掌握配置SSH Password认证的方法
  + 掌握SFTP的配置

# 实验内容

在本实验中，使用路由器R1模拟PC，作为SSH的Client，路由器R2作为SSH的Server，模拟远程客户端R1通过SSH协议远程登录到路由器R2上进行各种配置。本实验将通过Password认证方式来实现。

# 实验逻辑拓扑

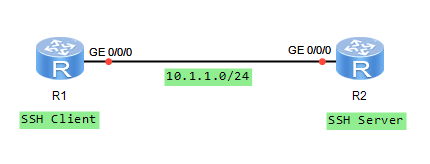


图1-38 配置通过STelnet登录系统拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 10.1.1.1 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.1.1.2 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

由于eNSP模拟软件自带PC没有SSH客户端，本实验采用2台路由器模拟实验，路由器R1作为SSH的Client，路由器R2作为SSH的Server。

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

[R2]ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=64 time=30 ms

Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=64 time=30 ms

Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=64 time=30 ms

Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=64 time=10 ms

Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=64 time=20 ms

--- 10.1.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/24/30 ms

## 配置SSH Server

相对于Telnet协议，SSH协议支持对报文加密传输，而非明文传送。因此，在跨越互联网的远程登录管理中，建议使用SSH协议。

生成本地RSA密钥对是完成SSH登录配置的首要操作。SSH用户使用Password方式验证，需要在SSH服务器端生成本地RSA密钥。

在R2上使用命令**rsa local-key-pair creat**来生成本地RSA主机密钥对。

[R2]rsa local-key-pair create

The key name will be: R2\_Host

The range of public key size is (512 ~ 2048).

NOTES: If the key modulus is greater than 512,

it will take a few minutes.

Input the bits in the modulus[default = 512]:

Generating keys...

....................++++++++++++

.................++++++++++++

.++++++++

.................++++++++

配置完成后，使用命令**display rsa local-key public**查看本地密钥对中的公钥部分信息。

[R2]display rsa local-key-pair public

=====================================================

Time of Key pair created: 2013-06-24 12:33:22-05:13

Key name: R2\_Host

Key type: RSA encryption Key

=====================================================

Key code:

3047

0240

B7C2165E 055CE5B2 ACB91781 18996572 05AF6068

F6B71A08 729D0494 84AD336D EAB8727C 2A8D4FB9

DC0E2AE8 FAD182F6 37BF685B 7D730889 173FA1CE

9621BF67

0203

010001

=====================================================

Time of Key pair created: 2013-06-24 12:33:27-05:13

Key name: Server

Key type: RSA encryption Key

=====================================================

Key code:

3067

0260

CEFA28DF E88B986F 8785B54E 035C0C4D 4671B975

C71871E3 6F069F1C C1D7ACA2 DE279ED9 368EC812

33E162D8 D03776C1 7757F05D A6D5F12E C5BBD88A

40EDD70F 071E2E99 B5D7330C 26E3D393 BDDB3B98

14E3086C 292F697D A973DC38 63C3570D

0203

010001

可以观察到，此时已经生成了本地RSA主机密钥对。“Time of Key pair created“描述公钥生成的时间，“Key name”描述公钥的名称，“Key type”描述公钥的类型。

在R2上配置VTY用户界面，设置用户的验证方式为AAA授权验证方式。

[R2]user-interface vty 0 4

[R2-ui-vty0-4]authentication-mode aaa

指定VTY类型用户界面只支持SSH协议，设备将自动禁止Telnet功能。

[R2-ui-vty0-4]protocol inbound ssh

[R2-ui-vty0-4]quit

使用**local-user**命令创建本地用户和用户口令，并以密文方式显示用户口令，指定用户名为huawei1，密码为huawei1。

[R2]aaa

[R2-aaa]local-user huawei1 password cipher huawei1

Info: Add a new user.

配置本地用户的接入类型为SSH。

[R2-aaa]local-user huawei1 service-type ssh

使用**ssh user**命令新建SSH用户，用户名为huawei1，指定SSH用户的认证方式为Password，即密码认证方式。

[R2]ssh user huawei1 authentication-type password

此处还可以继续使用命令**local-user huawei1 privilege level** 配置本地用户的优先级。取值范围为0-15，取值越大，代表用户的优先级越高。不同级别的用户登录后，只能使用等于或低于自身级别的命令，缺省值为3，代表管理级。

缺省情况下，设备的SSH服务器功能为关闭状态，只有开启了此功能后，客户端才能以SSH方式与设备建立连接。在R2上开启设备的SSH功能。

[R2]stelnet server enable

Info: Succeeded in starting the Stelnet server.

配置完成后，使用**display ssh user-information huawei1**命令在SSH服务器端查看SSH用户的配置信息。如果不在命令末尾指定SSH用户，则可以查看SSH服务器端所有的SSH用户配置信息。

[R2]display ssh user-information huawei1

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

Username Auth-type User-public-key-name

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

huawei1 password null

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

可以观察到所配置的SSH用户名，认证方式。

运行命令**display ssh server status**，可以查看SSH服务器全局配置信息。

[R2]display ssh server status

SSH version :1.99

SSH connection timeout :60 seconds

SSH server key generating interval :0 hours

SSH Authentication retries :3 times

SFTP Server :Disable

Stelnet server :Enable

可以观察到，此时R2上Stelnet Server服务器状态为启用状态。

## 配置SSH Client

当SSH客户端第一次登录SSH服务器时，客户端还没有保存SSH服务器的RSA公钥，会对服务器的RSA有效性公钥检查失败，从而导致登录服务器失败。因此当客户端R1首次登录时，需开启SSH客户端首次认证功能，不对SSH服务器的RSA公钥进行有效性检查。

[R1]ssh client first-time enable

在SSH客户端R1上使用命令**stelne**t连接SSH服务器。

[R1]stelnet 10.1.1.2

登录成功后，输入用户名huawei1。

Please input the username:huawei1

Trying 10.1.1.2 ...

Press CTRL+K to abort

Connected to 10.1.1.2 ...

The server is not authenticated. Continue to access it? (y/n)[n]:y

Jun 24 2013 13:14:46-05:13 R1 %%01SSH/4/CONTINUE\_KEYEXCHANGE(l)[0]:The server had not been authenticated in the process of exchanging keys. When deciding whether to continue, the user chose Y.

[R1]

Save the server's public key? (y/n)[n]:y

Jun 24 2013 1The server's public key will be saved with the name 10.1.1.2. Please wait...

3:14:50-05:13 R1 %%01SSH/4/SAVE\_PUBLICKEY(l)[1]:When deciding whether to save the server's public key 10.1.1.2, the user chose Y.

第一次登录时，由于开启了SSH客户端首次认证功能，在STelnet客户端第一次登录SSH服务器时，不对SSH服务器的RSA公钥进行有效性检查。登录后，系统将自动分配并保存RSA公钥，为下次登录时认证。

输入用户huawei1的密码huawei1。

Enter password:

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

User last login information: ---------------------------------------------------------------------------

Access Type: SSH

IP-Address : 10.1.1.1 ssh

Time : 2013-06-24 13:52:54-05:13 ---------------------------------------------------------------------------

<R2>

输入密码后，远程登录R2成功，使用命令**display ssh server session**查看SSH服务器端的当前会话连接信息。

[R2]display ssh server session

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

Conn Ver Encry State Auth-type Username

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

VTY 0 2.0 AES run password huawei1

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

可以观察到，用户huawei1已经成功通过VTY线路0远程登录上来，客户端已经成功连接到SSH服务器，可以进行各种配置。如果要退出登录，使用**quit**命令即可。

## 配置SFTP Server与Client

在R2上进入AAA视图，创建一个名称为huawei2的用户，并配置密码为huawei2，以密文方式显示。

[R2]aaa

[R2-aaa]local-user huawei2 password cipher huawei2

配置本地用户的接入类型为SSH。

[R2-aaa]local-user huawei2 service-type ssh

配置本地用户的优先级，不同级别的用户登录后，只能使用等于或低于自身级别的命令。取值范围为0~15，取值越大，用户的优先级越高。

[R2-aaa]local-user huawei2 privilege level 3

指定FTP用户的可访问目录。缺省为空，如果不配置，FTP用户将无法登录。

[R2-aaa]local-user huawei2 ftp-directory flash:

使用**ssh user**命令新建SSH用户，用户名为huawei2，指定SSH用户的认证方式为Password，即密码认证方式。

[R2]ssh user huawei2 authentication-type password

使用**sftp server enabe**命令开启SFTP服务器功能。

[R2]sftp server enable

配置完成后，查看SSH服务器的配置信息。

[R2]display ssh server status

SSH version :1.99

SSH connection timeout :60 seconds

SSH server key generating interval :0 hours

SSH Authentication retries :3 times

SFTP Server :Enable

Stelnet server :Enable

可以观察到，此时SFTP服务已经开启。

在R1上使用命令**sftp**连接SSH服务器，并输入用户名hauwei2和口令huawei2。

[R1]sftp 10.1.1.2

Please input the username:huawei2

Trying 10.1.1.2 ...

Press CTRL+K to abort

Enter password:

sftp-client>

可以观察到已经成功登录。

在R2上查看SSH会话连接信息。

[R2]display ssh server session

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

Conn Ver Encry State Auth-type Username

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

VTY 0 2.0 AES run password huawei2

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

可以观察到，用户huawei2已经成功通过VTY线路0远程登录上来，客户端已经成功连接到SSH服务器，可以进行各种配置。如果要退出登录，使用**quit**命令即可。

# 思考

开启SSH客户端首次认证功能有什么缺陷？如果不开启此功能如何成功在客户端远程登录？

1.6 配置通过FTP进行文件操作

# 原理概述

FTP（File Transfer Protocol）是在TCP/IP网络和INTERNET上最早使用的协议之一，在TCP/IP协议族中属于应用层协议，是文件传输的Internet标准。主要功能是向用户提供本地和远程主机之间的文件传输，尤其在进行版本升级、日志下载和配置保存等业务操作时。

FTP采用C/S（Client/Server）结构。FTP Server能够提供远程客户端访问和操作的功能，用户可以通过主机或者其他设备上的FTP客户端程序登录到服务器上，进行文件的上传、下载和目录访问等操作。

# 实验目的

* + 理解FTP的应用场景
  + 掌握操作FTP服务器的常见命令
  + 掌握保存文件到FTP的方法
  + 掌握获取FTP服务器文件到本地的方法
  + 掌握配置路由器为FTP服务器的方法

# 实验内容

本实验模拟企业网络，PC-1为FTP客户端设备，需要访问FTP Server，从服务器上下载或上传文件。处于安全角度考虑，防止服务器被病毒文件感染，不能允许客户端直接上传文件到Server。网络管理员在R1上做了限制使员工不能上传文件到Server，但是可以从Server下载文件。R1也需要作为客户端从Server下载更新文件，同时配置R1作为FTP服务器，员工可上传文件到R1上，经过管理员的检测后由R1再上传到FTP Server。

# 实验拓扑

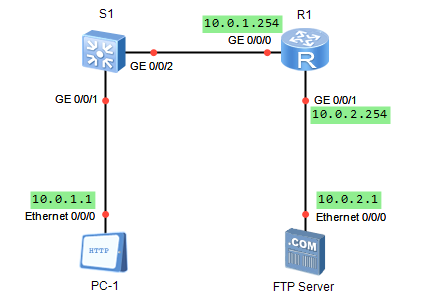


图1-39 配置通过FTP进行文件操作拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | 10.0.1.254 |
| FTP Server | Ethernet 0/0/1 | 10.0.2.1 | 255.255.255.0 | 10.0.2.254 |
| R1(AR2220) | GE 0/0/0 | 10.0.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.2.254 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 10.0.1.1

PING 10.0.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.0.1.1: bytes=56 Sequence=1 ttl=255 time=40 ms

Reply from 10.0.1.1: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 10.0.1.1: bytes=56 Sequence=3 ttl=255 time=10 ms

Reply from 10.0.1.1: bytes=56 Sequence=4 ttl=255 time=1 ms

Reply from 10.0.1.1: bytes=56 Sequence=5 ttl=255 time=20 ms

--- 10.0.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/16/40 ms

其余直连网段的连通性测试省略。

## 配置路由器为FTP Client

首先，在本地电脑上创建一个文件夹FTP-Huawei作为FTP服务器的文件夹，在该文件夹下再创建子文件夹Config，并创建测试文件test.txt。如图所示：

****

图1-40

创建完成后，设置FTP服务器的文件夹为刚才的主文件夹目录。

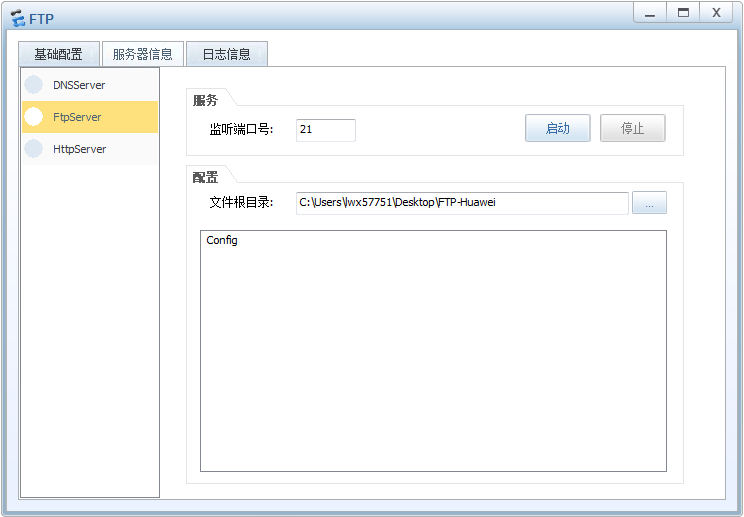


图1-41

设置完成之后，启动FTP Server。在R1上使用命令**ftp**连接FTP服务器。登录时默认需要输入用户名和密码，由于服务器上没有设置用户名和密码，每次在R1上输入时等同于创建该用户名和密码，本次使用用户名10.0.2.1，密码huawei。

<R1>ftp 10.0.2.1

Trying 10.0.2.1 ...

Press CTRL+K to abort

Connected to 10.0.2.1.

220 FtpServerTry FtpD for free

User(10.0.2.1:(none)):10.0.2.1

331 Password required for 10.0.2.1 .

Enter password:

230 User 10.0.2.1 logged in , proceed

[R1-ftp]

可以观察到，路由器进入FTP配置视图。

使用命令**ls**查看FTP服务器文件夹状态。

[R1-ftp]ls

200 Port command okay.

150 Opening ASCII NO-PRINT mode data connection for ls -l.

Config

226 Transfer finished successfully. Data connection closed.

FTP: 12 byte(s) received in 0.180 second(s) 66.66byte(s)/sec.

可以观察到，目前有文件夹Config。

使用命令**cd**进入文件夹。

[R1-ftp]cd Config

250 "/config" is current directory.

可以观察到，目前已进入该文件夹。

使用命令**dir**查看详细的文件属性。

[R1-ftp]dir

200 Port command okay.

150 Opening ASCII NO-PRINT mode data connection for ls -l.

drwxrwxrwx 1 10.0.2.1 nogroup 3 Aug 21 2013 test.txt

226 Transfer finished successfully. Data connection closed.

FTP: 66 byte(s) received in 0.050 second(s) 1.32Kbyte(s)/sec.

使用**get**命令下载test到本地路由器。

[R1-ftp]get test.txt

200 Port command okay.

150 Sending test.txt (3 bytes). Mode STREAM Type BINARY

226 Transfer finished successfully. Data connection closed.

FTP: 3 byte(s) received in 17.450 second(s) .17byte(s)/sec

可以观察到，下载文件成功。

使用**put**命令上传test到FTP服务器，命名为new.txt。

[R1-ftp]put test.txt new.txt

200 Port command okay.

150 Opening BINARY data connection for new.txt

226 Transfer finished successfully. Data connection closed.

FTP: 3 byte(s) sent in 0.070 second(s) 42.85byte(s)/sec.

可以观察到，上传文件成功。

## 配置路由器为FTP Server

在上面的步骤中，路由器作为FTP Client已经成功从FTP Server上获取和上传了文件。

现在将路由器配置为FTP服务器，可以使得路由器下行的客户端能够上传文件到路由器上，并可直接从Server上获取文件。

打开路由器R1的FTP服务器功能。

<R1>system-view

[R1]ftp server enable

设置FTP登录的用户名为ftp，密码为huawei，设置文件夹目录flash：。配置FTP用户可访问的目录为fash：，用户优先级为15，服务类型为ftp。

[R1]aaa

[R1-aaa]local-user ftp password cipher huawei

[R1-aaa]local-user ftp ftp-directory flash:

[R1-aaa]local-user ftp service-type ftp

[R1-aaa]local-user ftp privilege level 15

配置完成后，在本地创建测试文件test-user.txt，并设置客户端信息如下图。配置服务器地址为10.0.1.254，用户名为ftp，密码为huawei，然后点击“登录”按钮。

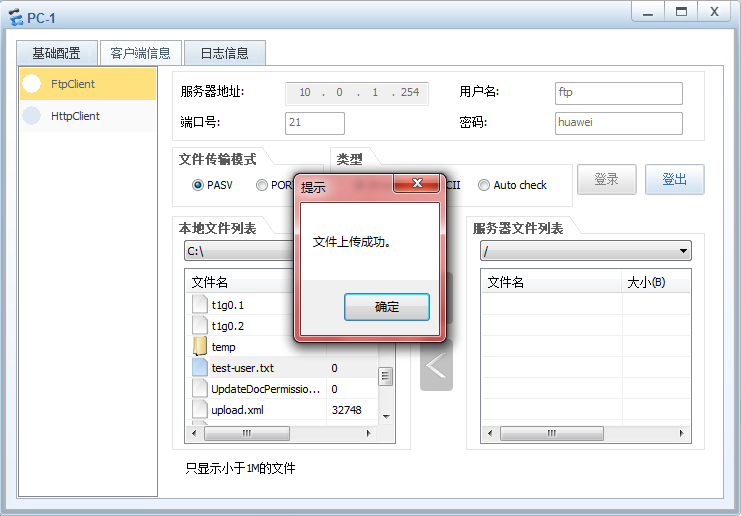


图1-42

登录成功后，可选择文件test-user，并点击向右箭头传送至FTP服务器，可观察到上传文件成功。

在R1上查看目录下的文件。

[R1]dir

Directory of flash:/

Idx Attr Size(Byte) Date Time(LMT) FileName

……

3 -rw- 0 Sep 09 2013 03:32:58 test-user.txt

4 -rw- 0 Sep 09 2013 03:25:47 test.txt

980,052 KB total (700,320 KB free)

可以观察到，已经将相应文件成功上传至FTP服务器R1。

# 思考

缺省情况下，FTP服务器端监听端口号是21，能否在路由器上变更此端口号，有什么好处

1. 交换机基础配置

2.1 交换机基础配置

# 原理概述

交换机之间通过以太网电接口对接时需要协商一些接口参数，比如速率、双工模式等。交换机的全双工是指交换机在发送数据的同时也能够接收数据，两者同时进行。就如平时打电话一样，说话的同时也能够听到对方的声音。而半双工指在同一时刻只能发送或只能接收数据，就像一条比较窄的路，只能先过一边的车，然后再过另一边的车，若两边一起过的话就会撞车。如果交换机两端接口协商模式不一致，会导致出现报文交互异常。

接口速率指交换机接口每秒钟传输数据的多少，在交换机上可根据需要调整以太网接口速率。缺省情况下，当以太网接口工作在非自协商模式时，它的速率为接口支持的最大速率。

# 实验目的

* + 理解双工模式和接口速率
  + 掌握更改双工模式的配置
  + 掌握更改接口速率的配置

# 实验内容

某公司刚成立，新组建网络，购置了三台交换机。其中S1和S2为接入层交换机，S3为汇聚层交换机。现在网络管理员需要对三台新交换机进行基本配置，保证交换机间的接口使用全双工模式，并根据需要配置接口速率。

# 实验拓扑

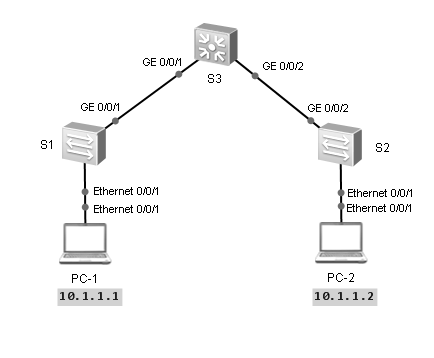


图2-1交换机基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

PC>ping 10.1.1.2

Ping 10.1.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: bytes=32 seq=1 ttl=128 time=62 ms

From 10.1.1.2: bytes=32 seq=2 ttl=128 time=63 ms

From 10.1.1.2: bytes=32 seq=3 ttl=128 time=31 ms

From 10.1.1.2: bytes=32 seq=4 ttl=128 time=47 ms

From 10.1.1.2: bytes=32 seq=5 ttl=128 time=78 ms

--- 10.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/56/78 ms

现在PC-1和PC-2能够正常通讯。

## 配置交换机双工模式

配置接口的双工模式可在自协商或者非自协商模式下进行。

在自协商模式下，接口的双工模式是和对端接口协商得到的，但协商得到的双工模式可能与实际要求不符。可通过配置双工模式的取值范围来控制协商的结果。例如互连的两个设备对应的接口都支持全/半双工，经自协商后工作在半双工模式，与实际要求的全双工模式不符，这时就可以执行命令**auto duplex full**使得接口的可协商双工模式为全双工模式。缺省情况下，以太网接口自协商双工模式范围为接口所支持的双工模式。

在非自协商模式下，可以根据实际需求手动配置接口的双工模式。

在S1，S2，S3三台交换机接口下先通过**undo negotiation auto**命令关掉自协商功能，再手工指定双工模式为全双工。

查看接口的基本状态信息

[S1]display interface GigabitEthernet 0/0/1

GigabitEthernet0/0/1 current state : UP

Line protocol current state : UP

Description:

Switch Port, PVID : 1, TPID : 8100(Hex), The Maximum Frame Length is 9216

IP Sending Frames' Format is PKTFMT\_ETHNT\_2, Hardware address is 4c1f-cc74-5560

Last physical up time : 2019-10-07 22:25:52 UTC-08:00

Last physical down time : 2019-10-07 22:25:51 UTC-08:00

Current system time: 2019-10-07 22:56:47-08:00

Hardware address is 4c1f-cc74-5560

Speed : 1000, Loopback: NONE

Duplex: FULL, Negotiation: ENABLE //模拟器看不到交换机接口速率，路由器可以

Last 300 seconds input rate 0 bytes/sec, 0 packets/sec

Last 300 seconds output rate 0 bytes/sec, 0 packets/sec

Input: 100033 bytes, 843 packets

Output: 906 bytes, 10 packets

Input:

Unicast: 6 packets, Multicast: 837 packets

Broadcast: 0 packets

Output:

Unicast: 5 packets, Multicast: 5 packets

Broadcast: 0 packets

Input bandwidth utilization : 0%

Output bandwidth utilization : 0%

[S1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]undo negotiation auto

[S1-GigabitEthernet0/0/1]duplex full

[S2]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]undo negotiation auto

[S2-GigabitEthernet0/0/2]duplex full

[S3]interface GigabitEthernet 0/0/1

[S3-GigabitEthernet0/0/1]undo negotiation auto

[S3-GigabitEthernet0/0/1]duplex full

[S3-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S3-GigabitEthernet0/0/2]undo negotiation auto

[S3-GigabitEthernet0/0/2]duplex full

## 配置接口速率

在自协商模式下，以太网接口的速率是和对端接口协商得到的。如果协商的速率与实际要求不符，可通过配置速率的取值范围来控制协商的结果。例如互连的两个设备对应的接口经自协商后的速率为10Mbit/s，与实际要求的100Mbit/s不符，可通过执行命令**auto speed 100**配置使得接口可协商的速率为100Mbit/s。缺省情况下，以太网接口自协商速率范围为接口支持的所有速率。在非自协商模式下，需手动配置接口速率，避免发生无法正常通讯的情况。

在非自协商模式时，缺省情况下，以太网接口的速率为接口支持的最大速率。

根据网络需要调整接口速率。由于网络用户较少，配置GE接口速率为100Mbit/s，Ethernet接口配置速率为10Mbit/s。

在三台交换机接口下配置速率。首先关闭接口自协商模式，并配置以太网接口的速率。

[S1]inter Ethernet 0/0/1

[S1-Ethernet0/0/1]undo negotiation auto

[S1-Ethernet0/0/1]speed 10

用同样的方法配置另外两台设备接口的速率。

[S2]interface ethernet 0/0/1

[S2-Ethernet0/0/1]undo negotiation auto

[S2-Ethernet0/0/1]speed 10

[S2-Ethernet0/0/1]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]undo negotiation auto

[S2-GigabitEthernet0/0/2]speed 100

[S3]interface GigabitEthernet 0/0/1

[S3-GigabitEthernet0/0/1]undo negotiation auto

[S3-GigabitEthernet0/0/1]speed 100

[S3-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S3-GigabitEthernet0/0/2]undo negotiation auto

[S3-GigabitEthernet0/0/2]speed 100

# 思考

在实际操作中，通常使用自动协商模式还是手动配置模式？为什么？

2.2 理解ARP及Proxy ARP

# 原理概述

ARP（Address Resolution Protocol）是用来将IP地址解析为MAC地址的协议。ARP表项可以分为动态和静态两种类型。动态ARP是利用ARP广播报文，动态执行并自动进行IP地址到以太网MAC地址的解析，无需网络管理员手工处理。静态ARP，建立IP地址和MAC地址之间固定的映射关系，在主机和路由器上不能动态调整此映射关系，需要网络管理员手工添加。设备上有一个ARP高速缓存（ARP cache），用来存放IP地址到MAC地址的映射表，利用ARP请求和应答报文来缓存映射表，以便能正确的把三层数据包封装成二层数据正，达到快速封装数据帧，正确转发数据的目的。另外ARP还有扩展应用功能，比如Proxy ARP功能。

Proxy ARP，即代理ARP，当主机上没有配置缺省网关地址（即不知道如何到达本地网络的网关设备），可以发送一个广播ARP请求（请求目的主机的MAC地址），使能Proxy ARP功能的路由器收到这样的请求后，在确认请求地址可达后，会使用自身的MAC地址作为该ARP请求的回应，使得处于不同物理网络的同一网段的主机之间可以正常的相互通信。

# 实验目的

* + 理解ARP工作原理
  + 掌握配置静态 ARP的方法
  + 理解Proxy ARP的工作原理
  + 掌握Proxy ARP的配置
  + 理解主机之间的通信过程

# 实验内容

本实验模拟公司网络场景，路由器R1是公司的出口网关，连接到外网。公司内所有员工使用10.1.0.0/16网段，通过交换机连接到网关路由器上。网络管理员通过配置静态ARP防止ARP欺骗攻击，保证通信安全。又由于公司内所有主机都没有配置网关，而又分属于不同广播域，造成无正常通信，网络管理员需要通过在路由器上配置ARP代理功能，实现网络内所有主机的通信。

# 实验拓扑



图2-2理解ARP及ARP proxy拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/1 | 10.1.1.254 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.1.2.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.0.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 10.1.1.2 | 255.255.0.0 | N/A |
| PC-3 | Ethernet 0/0/1 | 10.1.2.3 | 255.255.0.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

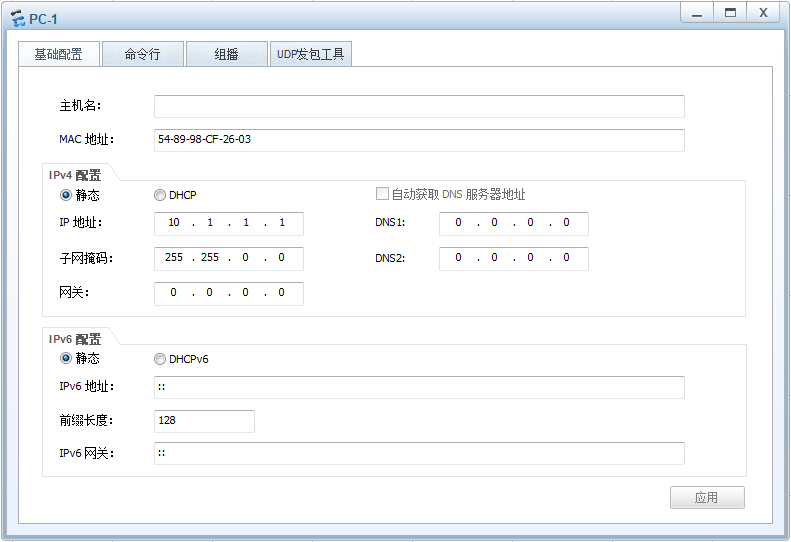


图2-3

根据实验编制表配置PC主机的IP地址及对应的掩码，掩码长度是16。配置完成后，在命令行下使用 **arp –a** 查看主机的ARP表。

PC>arp -a

Internet Address Physical Address Type

PC>

查看到ARP表项为空，没有任何条目在里面。

根据实验编制表配置路由器R1的接口IP地址，掩码长度为24。

[R1]interface GigabitEthernet0/0/1

[R1-GigabitEthernet0/0/1]ip address 10.1.1.254 255.255.255.0

[R1-GigabitEthernet0/0/1]interface GigabitEthernet0/0/2

[R1-GigabitEthernet0/0/2]ip address 10.1.2.254 255.255.255.0

配置完成后，使用**display arp all**命令查看R1的ARP表。

[R1]display arp all

IP ADDRESS MAC ADDRESS EXPIRE(M) TYPE INTERFACE VPN-INSTANCE

VLAN/CEVLAN PVC

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

10.1.1.254 5489-985c-8442 I - GE0/0/1

10.1.2.254 5489-985c-8542 I - GE0/0/2

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

Total:2 Dynamic:0 Static:0 Interface:2

ARP表中仅含有R1的两个接口IP地址及与其对应的MAC地址的ARP表项，没有其它条目。

在PC-1上使用**ping**命令测试到网关R1和PC-2的连通性。

PC>ping 10.1.1.254

Ping 10.1.1.254: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.254: bytes=32 seq=1 ttl=255 time=47 ms

From 10.1.1.254: bytes=32 seq=2 ttl=255 time=47 ms

From 10.1.1.254: bytes=32 seq=3 ttl=255 time=31 ms

From 10.1.1.254: bytes=32 seq=4 ttl=255 time=31 ms

From 10.1.1.254: bytes=32 seq=5 ttl=255 time=15 ms

--- 10.1.1.254 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/34/47 ms

PC>ping 10.1.1.2

Ping 10.1.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: bytes=32 seq=1 ttl=128 time=16 ms

From 10.1.1.2: bytes=32 seq=2 ttl=128 time<1 ms

From 10.1.1.2: bytes=32 seq=3 ttl=128 time=16 ms

From 10.1.1.2: bytes=32 seq=4 ttl=128 time=16 ms

From 10.1.1.2: bytes=32 seq=5 ttl=128 time=16 ms

--- 10.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/12/16 ms

在PC-3上，使用**ping**命令测试到网关R1的连通性。

PC>ping 10.1.2.254

Ping 10.1.2.254: 32 data bytes, Press Ctrl\_C to break

From 10.1.2.254: bytes=32 seq=1 ttl=255 time=47 ms

From 10.1.2.254: bytes=32 seq=2 ttl=255 time=46 ms

From 10.1.2.254: bytes=32 seq=3 ttl=255 time=31 ms

From 10.1.2.254: bytes=32 seq=4 ttl=255 time=16 ms

From 10.1.2.254: bytes=32 seq=5 ttl=255 time=15 ms

--- 10.1.2.254 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/31/47 ms

可以观察到，直连网络连通性正常。

当主机和网关之间有数据访问时，如果ARP表中没有目标IP地址与目标MAC地址的对应表项，ARP协议会被触发，向直连网段发送ARP广播请求包，请求目标IP地址所对应的MAC地址。下图是PC-1发送的ARP广播请求，请求目标IP 10.1.1.254的MAC地址。

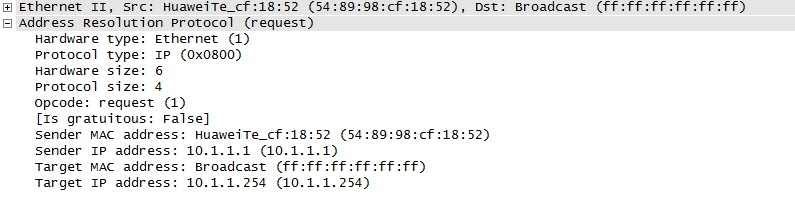


图2-4

网关收到广播请求后，回应单播的ARP响应，里面含有自身IP地址与MAC地址的对应关系，如下图所示。SRC-source DST:destination

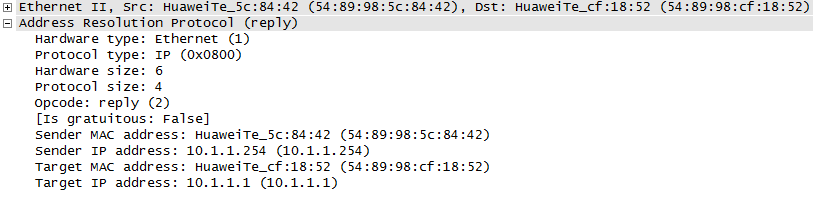


图2-5

PC和R1都会从这一对消息中知道对方的IP地址与MAC地址的对应关系，并将它写到各自的ARP表中。在PC-1上**ping**网关10.1.1.254后，在PC上使用**arp –a**命令查看。，在R1上使用命令**display arp all**去查看。

PC>arp -a

Internet Address Physical Address Type

10.1.1.254 54-89-98-5C-84-42 dynamic

<R1>display arp all

IP ADDRESS MAC ADDRESS EXPIRE(M) TYPE INTERFACE VPN-INSTANCE

VLAN/CEVLAN PVC

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

10.1.1.254 5489-985c-8442 I - GE0/0/1

10.1.1.1 5489-98cf-1852 20 D-0 GE0/0/1

10.1.2.254 5489-985c-8542 I - GE0/0/2

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

Total:3 Dynamic:1 Static:0 Interface:2

可以观察到，在PC-1上生成了网关IP地址10.1.1.254和与其对应的MAC地址的ARP表项，在R1上生成PC-1的IP地址10.1.1.1和与其对应的MAC地址的ARP表项。上述出现在PC机和R1里面的条目都是动态生成的。如果一段时间之后没有更新，便会从上述ARP表中删除。

## 配置静态ARP

上述ARP协议的工作行为往往被攻击者利用，发送伪造的ARP报文，而且报文里面所通告的IP地址和MAC地址的映射是错误的，主机或网关会把错误的映射更新到到ARP表里中。当主机要发送数据到指定的目标IP地址时，从ARP表里得到了不正确的硬件MAC地址，并用之封装数据帧，导致数据帧无法正确发送。

目前，由于公司内主机感染了这种ARP病毒，所以主机对网关R1进行ARP攻击，向网关R1通告含错误映射的ARP通告，导致网关路由器上使用不正确的动态ARP映射条目，造成其他主机无法与网关正常通信。

模拟ARP攻击发生时，网络的通讯受到了影响。在网关R1上，使用命令**arp static 10.1.1.1 1111-1111-1111**在路由器上静态添加一条关于PC-1的错误的ARP映射，假定此映射条目是通过一个ARP攻击报文所获得到的（静态的条目优于动态的条目），所以错误的映射将出现在ARP表中。

[R1]arp static 10.1.1.1 1111-1111-1111

使用命令**display arp all**查看ARP表，并使用**ping**测试PC-1和网关间的连通性。

[R1]display arp all

IP ADDRESS MAC ADDRESS EXPIRE(M) TYPE INTERFACE VPN-INSTANCE

VLAN/CEVLAN PVC

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

10.1.1.254 5489-985c-8442 I - GE0/0/1

10.1.2.254 5489-985c-8542 I - GE0/0/2

10.1.1.1 1111-1111-1111 S--

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

Total:3 Dynamic:0 Static:1 Interface:2

[R1]ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

……

PC>ping 10.1.1.254

Ping 10.1.1.254: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

可以观察到，在PC-1与网关间无法通信，因为在路由器R1上ARP的映射错误，导致路由器无法正确的发送数据包给PC-1。在R1的GE 0/0/1接口抓包观察。

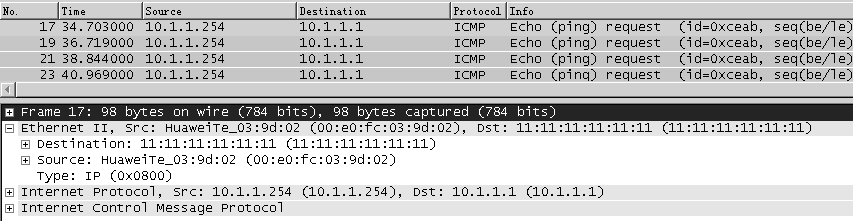


图2-6

可以观察到，由于配置了静态ARP，R1发往PC-1的ping包的二层头部，目的MAC地址被错误的封装为1111-1111-1111。

应对ARP欺骗攻击，防止其感染路由器的ARP表，可以通过配置静态ARP表项来防止。如果IP地址和一个MAC地址的静态映射已经出现在ARP表中，通过动态ARP方式学来的映射则无法进入ARP表。所以针对公司网络的现状，网络管理员在R1上手工配置三条关于PC-1、PC-2和PC-3的正确ARP映射，使用命令**arp static**，配置如下。

[R1]undo arp static 10.1.1.1 1111-1111-1111

[R1]arp static 10.1.1.1 5489-98cf-1852

[R1]arp static 10.1.1.2 5489-98cf-4f63

[R1]arp static 10.1.2.3 5489-98cf-5404

[R1]display arp all

IP ADDRESS MAC ADDRESS EXPIRE(M) TYPE INTERFACE VPN-INSTANCE

VLAN/CEVLAN PVC

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

10.1.1.254 5489-985c-8442 I - GE0/0/1

10.1.2.254 5489-985c-8542 I - GE0/0/2

10.1.1.1 5489-98cf-1852 S--

10.1.1.2 5489-98cf-4f63 S--

10.1.2.3 5489-98cf-5404 S--

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

Total:5 Dynamic:0 Static:3 Interface:2

配置完成后，在PC-1上测试。

PC>ping 10.1.1.254

Ping 10.1.1.254: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.254: bytes=32 seq=1 ttl=255 time=31 ms

From 10.1.1.254: bytes=32 seq=2 ttl=255 time=31 ms

From 10.1.1.254: bytes=32 seq=3 ttl=255 time=31 ms

From 10.1.1.254: bytes=32 seq=4 ttl=255 time=16 ms

From 10.1.1.254: bytes=32 seq=5 ttl=255 time=16 ms

--- 10.1.1.254 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 16/25/31 ms

可以观察到，配置后连通性恢复正常。

公司网络中出现ARP攻击比较常见。防御的办法之一是在ARP表里手工添加ARP映射，此种方法的优点是简单易作。不足之处是网络中每个网络设备都有ARP表，要全方位保护网络就要在尽可能多的三层设备上把全网的ARP映射手工写入到ARP表里，工作量过大，如果更换IP或MAC后，还需要手工更新ARP映射，远没有动态ARP协议维护ARP表方便。但如果公司网络规模不大或者网络设备不多的情况下，静态ARP方案还是具有一定优势的。

## 配置Proxy ARP

目前公司的网络被路由器R1分割为两个独立的广播域，每个路由器接口对应一个IP网络，分别是10.1.1.0/24和10.1.2.0/24，查看R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cos Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/1

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

10.1.2.0/24 Direct 0 0 D 10.1.2.254 GigabitEthernet0/0/2

10.1.2.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

在加入路由器之后，如果要主机在不改动原有配置的情况下，仍可正常的通讯，需要配置ARP代理功能。默认下，在路由器上ARP代理功能是关闭的。

如果R1保持ARP关闭的情况，PC-2和PC-3间不能互相通讯。在PC-2上，使用**ping**访问10.1.2.3，并在PC-2的E 0/0/1接口上抓包来观察。

PC>ping 10.1.2.3

Ping 10.1.2.3: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

……

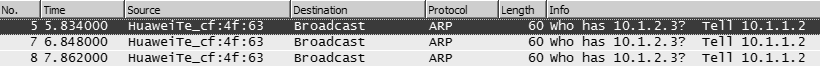


图2-7

可以观察到，PC-2发出了ARP广播，却一直没有收到ARP响应。原因是PC-2和PC-3分处在两个广播域内，PC-2发的ARP请求无法跨越中间的路由器，所以PC-3收不到PC-2的ARP请求，也无法知晓目标主机PC-3的硬件MAC地址而导致数据封装失败。

但R1如果开启ARP代理之后，看是否能够解决这个问题。配置**arp-proxy enable**命令在路由器的接口上来开启ARP代理功能。

[R1]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]arp-proxy enable

启用ARP代理之后，同样的测试，在PC-2上访问PC-3，并在PC-2的E0/0/1接口抓包观察。

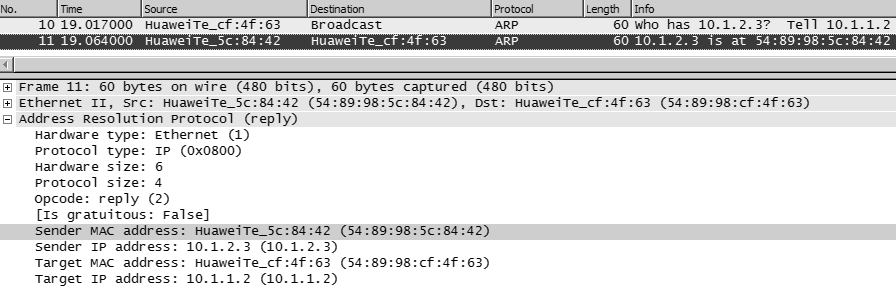


图2-8

可以观察到，PC-2发出了ARP请求并收到了ARP响应，但响应中10.1.2.3所对应的硬件MAC地址并非是PC-3的MAC地址，而是网关R1 的GE 0/0/1接口MAC地址。

在PC-2上查看ARP表。

PC>arp -a

Internet Address Physical Address Type

10.1.2.3 54-89-98-5C-84-42 dynamic

可以观察到，10.1.2.3所对应的的MAC是R1 GE 0/0/1接口的MAC地址，而不是真实PC-3的MAC地址。

开启ARP代理后，PC-2访问PC-3的工作过程如下。

R1的接口GE 0/0/1开启了ARP 代理后，收到PC-2的ARP广播请求报文后，R1根据ARP请求中的目标IP地址 10.1.2.3查看自身的路由表中是否有对应的目标网络，R1的GE 0/0/2接口就是10.1.2.0/24网络，所以，R1直接把自身的GE 0/0/1接口的MAC地址通过ARP响应返回给PC-2，PC-2接收到此ARP响应后使用该MAC作为目标硬件地址发送报文给R1，R1收到后再把报文转发给PC-3。

同理，PC-3要能访问R1连接的其他广播域的PC，也需要在R1的GE 0/0/2接口上开启ARP 代理功能。

[R1] interface GigabitEthernet 0/0/2

[R1-GigabitEthernet0/0/2]arp-proxy enable

配置完成后，测试PC-2与PC-3间的连通性。

PC>ping 10.1.2.3

Ping 10.1.2.3: 32 data bytes, Press Ctrl\_C to break

From 10.1.2.3: bytes=32 seq=1 ttl=127 time=47 ms

From 10.1.2.3: bytes=32 seq=2 ttl=127 time=63 ms

From 10.1.2.3: bytes=32 seq=3 ttl=127 time=47 ms

From 10.1.2.3: bytes=32 seq=4 ttl=127 time=62 ms

From 10.1.2.3: bytes=32 seq=5 ttl=127 time=62 ms

--- 10.1.2.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 47/56/63 ms

可以观察到，通信正常。

如果IP网络过大，广播对网络的影响也相应增大。在不改变网络主机配置的情况下，由管理员在网络中透明的插入一台路由器，靠路由器分割出多个广播域，降低了广播对网络的影响。尽管在当前的IP网络中，此种做法并不多见，但使用ARP代理的优点在于它不改变网络配置的情况下，透明的分割广播域。而缺点是主机间的通讯会因为引入额外的路由器而延迟增大，并存在着瓶颈问题，所以一般只作为临时解决方案使用。

# 思考

在ARP代理开启的情况下，如果在PC-2上，**ping 10.1.2.4**（10.1.2.4主机不存在），icmp echo报文是在PC-2还是R1路由器丢掉的？为什么？

1. VLAN

3.1 VLAN基础配置及Access接口

# 原理概述

早期的局域网技术是基于总线型结构的，总线型拓扑结构是由一根单电缆连接着所有主机，这种局域网技术存在着冲突域问题，即所有用户都在一个冲突域中，那么同一时间内只有一台主机能发送消息，从任意设备发出的消息都会被其他所有主机接收到，用户可能收到大量不需要的报文，而且所有主机共享一条传输通道，任意主机之间都可以直接互相访问，无法控制信息的安全。

为了避免冲突域，同时扩展传统局域网，以接入更多计算机，可以采用在局域网中使用二层交换机，交换机能有效隔离冲突域，但是由于所有计算机仍处于同一个广播域，任意设备都能接收到所有报文，不但降低了网络的效率，而且还降低了安全性，即广播域和信息安全问题依旧存在。为了能减少广播，提高局域网安全性，人们使用虚拟局域网即VLAN技术把一个物理的LAN在逻辑上划分成多个广播域。VLAN内的主机间可以直接通信，而VLAN间不能直接互通。这样，广播报文被限制在一个VLAN内，同时也提高了网络安全性。不同的VLAN使用不同的VLAN ID区分，VLAN ID的范围是0－4095，可配置的值为1－4094，0和4095为保留值。

Access接口是交换机上用来连接用户主机的接口。当Access接口从主机收到一个不带VLAN标签的数据帧时，会给该数据帧加上与PVID一致的VLAN 标签（PVID可手工配置，默认是1，即所有交换机上的接口默认都属于VLAN 1）。当Access接口要发送一个带VLAN标签的数据帧给主机时，首先检查该数据帧的VLAN ID是否与自己的PVID相同，若相同，则去掉VLAN标签后发送该数据帧给主机；若不相同，直接丢弃该数据帧。

# 实验目的

* + 理解VLAN的应用场景
  + 掌握VLAN的基本配置
  + 掌握Access接口的配置方法
  + 掌握Access接口加入相应VLAN的方法

# 实验内容

本实验模拟企业网络场景，公司内网是一个大的局域网，二层交换机S1放置在一楼，在一楼办公的部门有IT部和人事部，二层交换机S2放置在二楼，在二楼办公的部门有市场部和研发部。由于交换机组成的是一个广播网，交换机连接的所有主机都能互相通信，而公司策略是需要不同部门之间的主机不能互相通信，同一部门内的主机才可以互相访问。因此需要在交换机上划分不同的VLAN，并将连接主机的交换机接口配置成Access接口划分到相应VLAN内，实现公司不同部门之间不能互访的策略。

# 实验拓扑

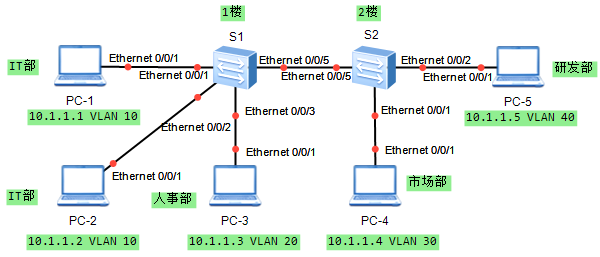


图3-1 VLAN基础配置及Access接口拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |
| PC-3 | Ethernet 0/0/1 | 10.1.1.3 | 255.255.255.0 | N/A |
| PC-4 | Ethernet 0/0/1 | 10.1.1.4 | 255.255.255.0 | N/A |
| PC-5 | Ethernet 0/0/1 | 10.1.1.5 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，在此步骤中不要为交换机创建任何的VLAN。

并使用**ping**命令检测各直连链路的连通性。所有的PC都能相互通信。

[PC]ping -c 1 10.1.1.2

PING 10.1.1.2: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=255 time=50 ms

--- 10.1.1.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 50/50/50 ms

其他主机间互相通信测试和上述相同，略过。

## 创建VLAN

除默认VLAN 1外，其余VLAN需要通过命令来手工创建。创建VLAN有两种方式，一种是使用命令**vlan**一次创建单个VLAN**，**另一种方式是使用命令**vlan batch**一次创建多个VLAN。

在S1上使用两条命令分别创建VLAN 10和VLAN 20。

[S1]vlan 10

[S1-vlan10]vlan 20

在S2上使用一条命令**vlan bath**创建VLAN 30和VLAN 40。

[S2]vlan batch 30 40

配置完成后，在S1和S2上使用**display vlan**命令查看VLAN的相关信息。

[S1]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(D) Eth0/0/2(D) Eth0/0/3(D) Eth0/0/4(D)

Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D) Eth0/0/8(D)

Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D) Eth0/0/12(D)

Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D) Eth0/0/16(D)

Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D) Eth0/0/20(D)

Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(D) GE0/0/2(D)

10 common

20 common

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(D) Eth0/0/2(D) Eth0/0/3(D) Eth0/0/4(D)

Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D) Eth0/0/8(D)

Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D) Eth0/0/12(D)

Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D) Eth0/0/16(D)

Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D) Eth0/0/20(D)

Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(D) GE0/0/2(D)

30 common

40 common

可以观察到，S1和S2都已经成功创建了相应VLAN，但目前没有任何接口加入所创建的VLAN 10与20中，默认情况下交换机上所有接口都属于VLAN 1。

## 配置Access接口

按照拓扑，使用命令**port link-type access**配置所有S1和S2交换机上连接PC的接口类型为Access类型接口，并使用命令**port default vlan**配置接口的缺省VLAN并同时加入相应VLAN中。缺省情况下，所有接口的缺省VLAN ID为1。

[S1]interface ethernet0/0/1

[S1-Ethernet0/0/1]port link-type access

[S1-Ethernet0/0/1]port default vlan 10

[S1-Ethernet0/0/1]interface ethernet0/0/2

[S1-Ethernet0/0/1]port link-type access

[S1-Ethernet0/0/1]port default vlan 10

[S1-Ethernet0/0/1]interface ethernet0/0/3

[S1-Ethernet0/0/1]port link-type access

[S1-Ethernet0/0/1]port default vlan 20

[S2]interface ethernet0/0/1

[S2-Ethernet0/0/1]port link-type access

[S2-Ethernet0/0/1]port default vlan 30

[S2-Ethernet0/0/1]interface ethernet0/0/2

[S2-Ethernet0/0/1]port link-type access

[S2-Ethernet0/0/1]port default vlan 40

配置完成后，查看S1与S2上的VLAN信息。

[S1]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

1 common UT:Eth0/0/4(D) Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D)

Eth0/0/8(D) Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D)

Eth0/0/12(D) Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D)

Eth0/0/16(D) Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D)

Eth0/0/20(D) Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(D)

GE0/0/2(D)

10 common UT:Eth0/0/1(D) Eth0/0/2(D)

20 common UT:Eth0/0/3(D)

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT: Eth0/0/3(D) Eth0/0/4(D) Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D) Eth0/0/8(D) Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D) Eth0/0/12(D) Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D) Eth0/0/16(D) Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D) Eth0/0/20(D) Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(D) GE0/0/2(D)

30 common UT:Eth0/0/1(D)

40 common UT:Eth0/0/2(D)

可以观察到，目前两台交换机上连接PC的接口都已经加入到相应所属部门的VLAN当中。

## 检查配置结果

在交换机上将不同接口加入各自不同的VLAN中后，属于相同VLAN的接口处于同一个广播域，相互之间可以直接通信。属于不同VLAN的接口是处于不同的广播域，相互之间不能直接通信。

在本实验环境中，只有同属于IT部门VLAN10的两台主机PC-1和PC-2之间可以互相通信。其他不同部门间的PC之间将无法通信。

在IT部门的终端PC-1上分别测试与同部门的终端PC-2，HR部门的PC-3间的连通性。

PC>ping -c 1 10.1.1.2

PING 10.0.1.2: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=255 time=50 ms

--- 10.1.1.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 50/50/50 ms

PC>ping 1 10.1.1.2

From 0.0.0.0: Destination host unreachable

From 0.0.0.0: Destination host unreachable

From 0.0.0.0: Destination host unreachable

From 0.0.0.0: Destination host unreachable

From 0.0.0.0: Destination host unreachable

……

可以观察到，相同VLAN内的PC可以互相通信，不同VLAN内的PC间无法通信。

# 思考

在本实验中，如果将S2的接口E 0/0/5配置为Access类型接口，并划入VLAN 30中，此时PC-1能否ping通PC-4？PC-1能否ping通PC-5？为什么？，

3.2 配置Trunk接口

# 原理概述

在以太网中，通过划分VLAN来隔离广播域和增强网络通讯的安全性。以太网通常由多台交换机组成，为了使VLAN的数据帧跨越多台交换机传递，交换机之间互连的链路需要配置为干道链路（Trunk Link）。和接入链路不同，干道链路是用来在不同的设备之间（如交换机和路由器之间、交换机和交换机之间）承载多个不同VLAN数据的，干道链路是不属于任何一个具体的VLAN的，干道链路可以承载所有的VLAN数据，也可以配置为只能传输指定VLAN的数据。

Trunk端口一般用于交换机之间连接的端口，Trunk端口可以属于多个VLAN，可以接收和发送多个VLAN的报文。

当Trunk端口收到数据帧时，如果该帧不包含802.1Q的VLAN标签，将打上该Trunk端口的PVID；如果该帧包含802.1Q的VLAN标签，则不改变。

当Trunk端口发送数据帧时，当该所发送帧的VLAN ID与端口的PVID不同时，检查是否允许该VLAN通过，若允许的话直接透传，不允许就直接丢弃；当该帧的VLAN ID与端口的PVID相同时，则剥离VLAN标签后转发。

# 实验目的

* + 理解干道链路的应用场景
  + 掌握Trunk端口的配置
  + 掌握Trunk端口允许所有VLAN通过的配置方法
  + 掌握Trunk端口允许特定VLAN通过的配置方法

# 实验内容

本实验模拟某公司网络场景，公司规模较大，员工200余名，内部网络是一个大的局域网。公司放置了多台接入交换机（如S1和S2）负责员工的网络接入。接入交换机之间通过汇聚交换机S3相连。公司通过划分VLAN来隔离广播域，由于员工较多，相同部门的员工通过不同交换机接入。为了保证在不同交换机下相同部门的员工能互相通信，需要配置交换机之间链路为干道模式，实现相同VLAN跨交换机通信。

# 实验拓扑

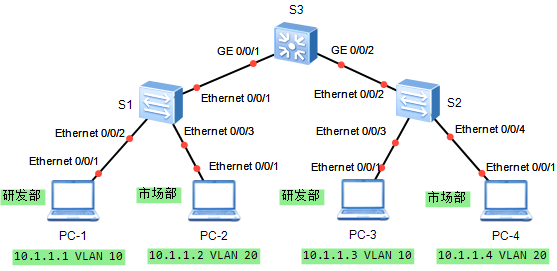


图3-2 跨交换机实现VLAN间通信拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet0/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet0/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |
| PC-3 | Ethernet0/0/1 | 10.1.1.3 | 255.255.255.0 | N/A |
| PC-4 | Ethernet0/0/1 | 10.1.1.4 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。在没有完成划分VLAN之前各PC之间都能互通（属于默认VLAN 1）。

这里以PC-1与PC-3的ping测试为例，其余省略。

PC>ping 10.1.1.3

Ping 10.1.1.3: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.3: bytes=32 seq=1 ttl=128 time=62 ms

From 10.1.1.3: bytes=32 seq=2 ttl=128 time=62 ms

From 10.1.1.3: bytes=32 seq=3 ttl=128 time=62 ms

From 10.1.1.3: bytes=32 seq=4 ttl=128 time=78 ms

From 10.1.1.3: bytes=32 seq=5 ttl=128 time=78 ms

--- 10.0.12.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/68/120 ms

## 创建VLAN，配置Access接口

公司内网需要通过VLAN的划分来隔离不同的部门。需要在三台交换机S1，S2，S3上都分别创建VLAN 10和VLAN 20，研发部员工属于VLAN 10，市场部员工属于VLAN 20。

[S1]vlan 10

[S1-vlan10]description R$D

[S1-vlan10]vlan 20

[S1-vlan20]description Market

[S2]vlan 10

[S2-vlan10]description R$D

[S2-vlan10]vlan 20

[S2-vlan20]description Market

[S3]vlan 10

[S3-vlan10]description R$D

[S3-vlan10]vlan 20

[S3-vlan20]description Market

配置完成后，使用命令**display vlan**查看所配置的VLAN信息，以S3为例。

<S3>display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 common

20 common

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable R$D

20 enable default enable disable Market

可以观察到相关的VLAN都已经配置好。也可以使用命令**display vlan summary**查看所配置VLAN的简要信息。

<S3>display vlan summary

static vlan:

Total 3 static vlan.

1 10 20

dynamic vlan:

Total 0 dynamic vlan.

reserved vlan:

Total 0 reserved vlan.

在S1上配置E 0/0/2和E 0/0/3为Access接口，并划分到相应的VLAN 10内。

[S1]interface Ethernet0/0/2

[S1-Ethernet0/0/2]port link-type access

[S1-Ethernet0/0/2]port default vlan 10

[S1]interface Ethernet0/0/3

[S1-Ethernet0/0/3]port link-type access

[S1-Ethernet0/0/3]port default vlan 20 直接输入正确vlan更正，无需undo

在S2上配置E 0/0/3和E 0/0/4为Access接口，并划分到相应的VLAN。

[S2]interface Ethernet0/0/3

[S2-Ethernet0/0/3]port link-type access

[S2-Ethernet0/0/3]port default vlan 10

[S2]interface Ethernet0/0/4

[S2-Ethernet0/0/4]port link-type access

[S2-Ethernet0/0/4]port default vlan 20

配置完成后，使用命令**display port vlan** 检查VLAN和接口配置情况。

[S1]display port vlan

Port Link Type PVID Trunk VLAN List

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

Ethernet0/0/1 hybrid 1 -

Ethernet0/0/2 access 10 -

Ethernet0/0/3 access 20 -

[S2]display port vlan

Port Link Type PVID Trunk VLAN List

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

Ethernet0/0/1 hybrid 1 -

Ethernet0/0/2 hybrid 1 -

Ethernet0/0/3 access 10 -

Ethernet0/0/4 access 20 -

可以观察到PC机所连接的交换机接口都已经被配置成Access模式，并且已经加入到了正确的VLAN中。

## 配置Trunk接口

在上个步骤中已经将PC机所连入的交换机接口划入到了相应的部门VLAN，现在测试相同部门中的PC是否能够通信。

测试PC-1与PC-3之间的连通性。

PC>ping 10.1.1.3

Ping 10.1.1.3: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

……

测试PC-2与PC-4之间的连通性。

PC>ping 10.1.1.4

Ping 10.1.1.4: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

From 10.1.1.2: Destination host unreachable

……

可以观察到此时同部门的PC机间不能通信。

目前在该跨交换机实现不同VLAN通信的二层组网拓扑中，虽然与PC端相连的交换机接口上创建并划分了VLAN信息，但是在交换机与交换机之间相连的接口上并没有相应的VLAN信息，不能够识别和发送跨越交换机的VLAN报文，此时VLAN只具有在每台交换机上的本地意义，无法实现相同VLAN的跨交换机通信。

为了使得交换机间能够识别和发送跨越交换机的VLAN报文，需要用到Trunk技术，将交换机间相连的接口配置成为Trunk接口。Trunk接口是交换机上用来与其他交换机连接的接口，能实现同时传递多个VLAN的流量。在配置的时候要明确配置被允许通过的VLAN，实现对VLAN流量传输的控制。

在S1上配置E 0/0/1为Trunk接口，允许VLAN 10和VLAN 20通过。

[S1]interface Ethernet0/0/1

[S1-Ethernet0/0/1]port link-type trunk

[S1-Ethernet0/0/1]port trunk allow-pass vlan 10 20

在S2上配置E 0/0/2为Trunk接口，允许VLAN 10和VLAN 20通过。

[S2]interface Ethernet0/0/2

[S2-Ethernet0/0/2]port link-type trunk

[S2-Ethernet0/0/2]port trunk allow-pass vlan 10 20

在S3上配置GE 0/0/1和GE 0/0/2为Trunk接口，允许所有VLAN通过。

[S3]interface GigabitEthernet 0/0/1

[S3-GigabitEthernet0/0/1]port link-type trunk

[S3-GigabitEthernet0/0/1]port trunk allow-pass vlan all

[S3]interface GigabitEthernet 0/0/2

[S3-GigabitEthernet0/0/2]port link-type trunk

[S3-GigabitEthernet0/0/2]port trunk allow-pass vlan all

配置完成后可以使用命令**display port vlan**来检查trunk的配置情况，这里以S3为例。

[S3]display port vlan

Port Link Type PVID Trunk VLAN List

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

GigabitEthernet0/0/1 trunk 1 1-4094

GigabitEthernet0/0/2 trunk 1 1-4094

可以观察到S3的GE 0/0/1和GE 0/0/2已被成功配置为Trunk接口，并且允许所有VLAN流量通过（VLAN 1-4094）。

再次验证不同交换机上的相同部门的PC间的连通性。

测试PC-1与PC-3之间的连通性。

PC>ping 10.1.1.3

Ping 10.1.1.3: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.3: bytes=32 seq=1 ttl=128 time=46 ms

From 10.1.1.3: bytes=32 seq=2 ttl=128 time=78 ms

From 10.1.1.3: bytes=32 seq=3 ttl=128 time=78 ms

From 10.1.1.3: bytes=32 seq=4 ttl=128 time=46 ms

From 10.1.1.3: bytes=32 seq=5 ttl=128 time=47 ms

-- 10.0.12.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/68/120 ms

测试PC-2与PC-4之间的连通性。

PC>ping 10.1.1.4

Ping 10.1.1.4: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.4: bytes=32 seq=1 ttl=128 time=46 ms

From 10.1.1.4: bytes=32 seq=2 ttl=128 time=47 ms

From 10.1.1.4: bytes=32 seq=3 ttl=128 time=46 ms

From 10.1.1.4: bytes=32 seq=4 ttl=128 time=78 ms

From 10.1.1.4: bytes=32 seq=5 ttl=128 time=63 ms

-- 10.0.12.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/68/120 ms

可以观察到此时同部门中的PC机已经能成功通信。

# 思考

连接PC机的交换机接口也可以配置成Trunk接口吗？为什么？

3.3 理解Hybrid接口的应用

# 原理概述

Hybrid接口既可以连接普通终端的接入链路又可以连接交换机间的干道链路。Hybrid接口允许多个VLAN的帧通过，并可以在出接口方向将某些VLAN帧的标签剥掉。

Hybrid接口处理VLAN帧的过程如下：

收到一个二层帧，判断是否有VLAN标签。没有标签，则标记上Hybrid接口的PVID，进行下一步处理。有标签，判断该Hybrid接口是否允许该VLAN的帧进入，允许则进行下一步处理，否则丢弃；

当数据帧从Hybrid接口发出时，交换机判断VLAN在本接口的属性是Untagged还是Tagged。如果是Untagged，先剥离帧的VLAN 标签，再发送；如果是Tagged，则直接发送帧。

通过配置Hybird接口，能够实现对VLAN标签的灵活控制，既能够实现Access接口的功能，又能够实现Trunk接口的功能。

# 实验目的

* + 掌握配置Hybrid接口的方法
* 理解Hybrid接口处理Untagged数据帧过程
* 理解Hybrid接口处理Tagged数据帧过程
* 理解Hybrid接口的应用场景

# 实验内容

某企业二层网络使用两台S3700交换机S1和S2，且两台设备在不同的楼层。网络管理员规划了三个不同VLAN，HR部门使用VLAN 10，市场部门使用VLAN 20，IT部门使用VLAN 30。现在需要处于不同楼层的HR部门和市场部门的员工能够实现各自部门内部通信，而两部门之间不允许互相通信，且IT部门可以访问任意部门，可以通过配置Hybird接口来实现较复杂的VLAN控制。

# 实验拓扑

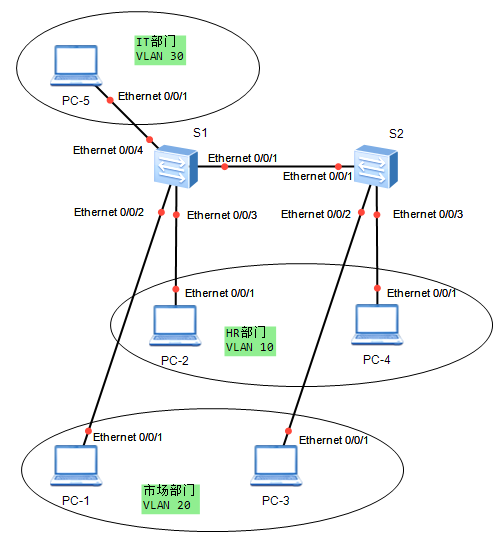


图3-3理解Hybird接口的应用拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 192.168.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 192.168.1.2 | 255.255.255.0 | N/A |
| PC-3 | Ethernet 0/0/1 | 192.168.1.3 | 255.255.255.0 | N/A |
| PC-4 | Ethernet 0/0/1 | 192.168.1.4 | 255.255.255.0 | N/A |
| PC-5 | Ethernet 0/0/1 | 192.168.1.100 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

按照实验编址表为PC机配置IP地址，如下的配置过程适用于所有终端。



图3-4

完成配置后，测试主机之间的连通性。

在PC-1上，使用**ping**命令。

PC>ping 192.168.1.2

Ping 192.168.1.2: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.2: bytes=32 seq=1 ttl=128 time=15 ms

From 192.168.1.2: bytes=32 seq=2 ttl=128 time<1 ms

From 192.168.1.2: bytes=32 seq=3 ttl=128 time<1 ms

From 192.168.1.2: bytes=32 seq=4 ttl=128 time=15 ms

From 192.168.1.2: bytes=32 seq=5 ttl=128 time<1 ms

--- 192.168.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/6/15 ms

可以观察到，此时PC-1访问其他主机通信正常，其他主机上测试过程省略。

在没有定义VLAN及接口类型之前，交换机上所有接口，默认的情况下，都是Hybrid类型，接口的PVID是VLAN 1，即所有接口收到没有标签的二层数据帧，都被转发到VLAN 1中，并继续以Untagged的方式把帧发送至同为VLAN 1的其它接口。所以，即使未做任何配置，主机之间默认仍然可以互相通信。

在S1上使用命令**display port vlan**查看接口的默认类型。

<S1>display port vlan

Port Link Type PVID Trunk VLAN List

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

Ethernet0/0/1 hybrid 1 -

Ethernet0/0/2 hybrid 1 -

Ethernet0/0/3 hybrid 1 -

Ethernet0/0/4 hybrid 1 -

Ethernet0/0/5 hybrid 1 -

Ethernet0/0/6 hybrid 1 -

……

可以观察到，接口默认是Hybrid类型，接口PVID是VLAN 1，其他接口也一样。

在交换机上使用命令**display vlan** 查看接口和所属VLAN的对应关系。

[S1]display vlan

The total number of vlans is : 1

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(U) Eth0/0/2(U) Eth0/0/3(U) Eth0/0/4(U)

Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D) Eth0/0/8(D)

Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D) Eth0/0/12(D)

Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D) Eth0/0/16(D)

Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D) Eth0/0/20(D)

Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(D) GE0/0/2(D)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

可以观察到，所有接口都默认属于VLAN 1，其他交换机也都一样，因此VLAN 1内所有的主机都可以直接访问。

## 实现组内通信，组间隔离

交换机接口的类型可以是Access类型，Trunk类型和Hybrid类型。Access类型的接口仅属于一个VLAN，只能接收、转发相应VLAN的帧。而Trunk类型接口则默认属于所有VLAN，任何Tagged帧都能经过Trunk接收和转发。Hybrid类型接口则介于二者之间，可自主定义端口上能接收和转发哪些VLAN Tag的帧，并可决定VLAN Tag是否继续携带或者剥离。Access和Trunk类型接口是Hybrid类型接口的两个特例，一个仅支持一个VLAN的传递，一个默认支持所有VLAN的传递，Access类型和Trunk类型的接口能做到的，Hybrid接口也能做到。

目前要求实现HR部门和市场部门内的员工终端可以进行部门内部通信，即VLAN 10内PC-2和PC-4之间可以自由访问，VLAN 20内PC-1和PC-3之间可以自由访问，而两个部门间的员工不能互相访问，即VLAN 10和VLAN 20之间不能互相访问。要实现此需求，可以使用Access和Trunk的配置方法达到要求，也可以仅使用Hybrid的配置方法实现同样的要求。

使用Trunk和Access类型接口的配置过程如下。

将S1上的E 0/0/2和S2上的E 0/0/2配置为Access类型，并将相应的接口加入到VLAN 20。同理，将S1上的 E 0/0/3和S2上的E 0/0/3也配置为Access类型接口，并加入到VLAN 10。而交换机之间的互连链路的两个E 0/0/1接口则配置为Trunk类型。

[S1]vlan 10

[S1-vlan10]vlan 20

[S1-vlan20]interface Ethernet0/0/3

[S1-Ethernet0/0/3]port link-type access

[S1-Ethernet0/0/3]port default vlan 10

[S1-Ethernet0/0/3]interface ethernet0/0/2

[S1-Ethernet0/0/2]port link-type access

[S1-Ethernet0/0/2]port default vlan 20

[S1-Ethernet0/0/2]interface Ethernet0/0/1

[S1-Ethernet0/0/1]port link-type trunk

[S1-Ethernet0/0/1]port trunk allow-pass vlan all

[S2]vlan 10

[S2-vlan10]vlan 20

[S2-vlan10]interface Ethernet0/0/3

[S2-Ethernet0/0/3]port link-type access

[S2-Ethernet0/0/3]port default vlan 10

[S2-Ethernet0/0/3]interface ethernet0/0/2

[S2-Ethernet0/0/2]port link-type access

[S2-Ethernet0/0/2]port default vlan 20

[S2-Ethernet0/0/2]interface ethernet0/0/1

[S2-Ethernet0/0/1]port link-type trunk

[S2-Ethernet0/0/1]port trunk allow-pass vlan all

配置完成后，查看接口和VLAN的对应关系。

[S1]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(U) Eth0/0/4(U) Eth0/0/5(D) Eth0/0/6(D)

Eth0/0/7(D) Eth0/0/8(D) Eth0/0/9(D) Eth0/0/10(D)

Eth0/0/11(D) Eth0/0/12(D) Eth0/0/13(D) Eth0/0/14(D)

Eth0/0/15(D) Eth0/0/16(D) Eth0/0/17(D) Eth0/0/18(D)

Eth0/0/19(D) Eth0/0/20(D) Eth0/0/21(D) Eth0/0/22(D)

GE0/0/1(D) GE0/0/2(D)

10 common UT:Eth0/0/3(U)

20 common UT:Eth0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(U) Eth0/0/4(D) Eth0/0/5(D) Eth0/0/6(D)

Eth0/0/7(D) Eth0/0/8(D) Eth0/0/9(D) Eth0/0/10(D)

Eth0/0/11(D) Eth0/0/12(D) Eth0/0/13(D) Eth0/0/14(D)

Eth0/0/15(D) Eth0/0/16(D) Eth0/0/17(D) Eth0/0/18(D)

Eth0/0/19(D) Eth0/0/20(D) Eth0/0/21(D) Eth0/0/22(D)

GE0/0/1(D) GE0/0/2(D)

10 common UT:Eth0/0/3(U)

20 common UT:Eth0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

可以观察到，配置已经生效。

在PC-1上测试与同VLAN 20的PC-3的连通性，及与VLAN 10内终端的连通性。

PC> ping 192.168.1.3

Ping 192.168.1.3: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.3: bytes=32 seq=1 ttl=128 time=47 ms

From 192.168.1.3: bytes=32 seq=2 ttl=128 time=31 ms

From 192.168.1.3: bytes=32 seq=3 ttl=128 time=46 ms

From 192.168.1.3: bytes=32 seq=4 ttl=128 time=31 ms

From 192.168.1.3: bytes=32 seq=5 ttl=128 time=31 ms

--- 192.168.1.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/37/47 ms

PC> ping 192.168.1.4

Ping 192.168.1.4: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

……

可以观察到，在单台交换机及跨交换机间的访问控制使用Trunk和Access类型接口实现了需求，但同样的需求使用Hybrid实现会更灵活。

S1的E 0/0/2接口连接PC-1主机，该接口收到的PC-1发送的Untagged的帧会被交换机转发到VLAN 20。同样，交换机从其他接口收到VLAN 20的发往PC-1的帧也会以Untagged的方式从E 0/0/2接口发送。S1的E 0/0/3接口连接PC-2主机，该接口收到Untagged的帧会被转发到VLAN 10。如果交换机收到的VLAN 10的发往PC-2的帧也会以Untagged的方式从接口E 0/0/3发送。VLAN 10和VLAN 20的帧也要经过交换机间链路发送至邻居交换机S2。反之，S1收到来自邻居交换机S2的Tagged的帧后，也会根据VLAN Tag转发到相应的VLAN。

在S1的E 0/0/2接口上使用**undo port default vlan**命令用来恢复接口默认VLAN。

[S1]interface ethernet0/0/2

[S1-Ethernet0/0/2]undo port default vlan

配置**port link-type hybrid** 命令修改接口类型为默认的Hybird类型。

[S1-Ethernet0/0/2]port link-type hybrid

配置**port hybrid untagged vlan 20**命令使得交换机在该接口转发VLAN 20的帧时，剥离掉相应的VLAN Tag 20，以Untagged的方式发送给PC。

[S1-Ethernet0/0/2]port hybrid untagged vlan 20

配置**port hybrid pvid vlan 20**命令设置Hybird类型接口的缺省VLAN ID，即使得该端口上接收到PC发来的未带VLAN Tag的帧时，加上VLAN Tag 20，并转发到VLAN 20。

[S1-Ethernet0/0/2]port hybrid pvid vlan 20

同样在连接另一台终端的E0/0/3接口做同样配置。

[S1]interface ethernet0/0/3

[S1-Ethernet0/0/3]undo port default vlan

[S1-Ethernet0/0/3]port link-type hybrid

[S1-Ethernet0/0/3]port hybrid untagged vlan 10

[S1-Ethernet0/0/3]port hybrid pvid vlan 10

在连接交换机S2的E 0/0/1接口上修改端口类型为默认的Hybird类型。并使用**port hybrid tagged vlan 10 20**命令设置该链路仅接收带有VLAN Tag 10和20的帧，而交换机也仅转发VLAN 10和VLAN 20的帧到该链路，一般该命令配置在交换机互连的链路接口之上。

[S1]interface ethernet0/0/1 // undo port trunk allow-pass vlan all

[S1-Ethernet0/0/1]port trunk allow-pass vlan 1

[S1-Ethernet0/0/1]port link-type hybrid

[S1-Ethernet0/0/1]port hybrid tagged vlan 10 20

S2交换机将在E 0/0/1接口接收到的Tagged帧，根据VLAN Tag标识，向接口E 0/0/2转发VLAN 20的帧，向接口E 0/0/3转发VLAN 30的帧。反之，接口E 0/0/2接收到PC发送的未带Tag的帧转发到VLAN 20，端口E 0/0/3接收的到未带Tag的帧会被转发到VLAN 10，并且这些的帧发送到邻居交换机S1时，会保留原有Tag。

S2上的配置如下，和S1类似，不再解释。

[S2]interface ethernet0/0/2

[S2-Ethernet0/0/2]undo port default vlan

[S2-Ethernet0/0/2]port link-type hybrid

[S2-Ethernet0/0/2]port hybrid untagged vlan 20

[S2-Ethernet0/0/2]port hybrid pvid vlan 20

[S2-Ethernet0/0/2]interface ethernet0/0/3

[S2-Ethernet0/0/3]undo port default vlan

[S2-Ethernet0/0/3]port link-type hybrid

[S2-Ethernet0/0/3]port hybrid untagged vlan 10

[S2-Ethernet0/0/3]port hybrid pvid vlan 10

[S2-Ethernet0/0/3]interface ethernet 0/0/1

[S2-Ethernet0/0/1]port trunk allow-pass vlan 1

[S2-Ethernet0/0/1]port link-type hybrid

[S2-Ethernet0/0/1]port hybrid tagged vlan 10 20

配置完成后，使用**display vlan**命令查看使用Hybrid配置下接口和VLAN的对应关系。

[S1]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(U) Eth0/0/2(U) Eth0/0/3(U) Eth0/0/4(U)

Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D) Eth0/0/8(D)

Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D) Eth0/0/12(D)

Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D) Eth0/0/16(D)

Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D) Eth0/0/20(D)

Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(D) GE0/0/2(D)

10 common UT:Eth0/0/3(U)

TG:Eth0/0/1(U)

20 common UT:Eth0/0/2(U)

TG:Eth0/0/1(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(U) Eth0/0/2(U) Eth0/0/3(U) Eth0/0/4(D)

Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D) Eth0/0/8(D)

Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D) Eth0/0/12(D)

Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D) Eth0/0/16(D)

Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D) Eth0/0/20(D)

Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(D) GE0/0/2(D)

10 common UT:Eth0/0/3(U)

TG:Eth0/0/1(U)

20 common UT:Eth0/0/2(U)

TG:Eth0/0/1(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

可以观察到，同样的需求，Hybrid和Access/Trunk都能实现，测试省略。但Hybrid的灵活性及解决复杂需求的能力是Access/Trunk做不到的。

## 实现网管员对所有网络的访问

在实现各部门内部终端可以互相访问，不同部门间的终端隔离访问后，要求网络管理员使用终端PC-5所在的IT部门能够实现对所有部门的访问。

即要求实现VLAN 30访问VLAN 10和VLAN 20，VLAN 10和VLAN 20之间仍然不允许互相访问。VLAN 30的终端要能访问VLAN 10的终端，如果S1的E0/0/2接口仍是Access类型且属于VLAN 10，是不能被其它VLAN访问的。需要修改接口的配置，使其既能被VLAN 10访问，又能被VLAN 30访问，则此接口同时要属于多个VLAN，且端口所连设备是PC，不能识别带VLAN Tag的帧，故此时只能使用Hybrid类型接口。Hybrid端口既能被加入多个VLAN中，又能够将其余VLAN的帧转发到此接口时，剥离掉相应的VLAN Tag。

配置S1交换机，E 0/0/4接口是网络管理员的PC终端，属于VLAN 30，该接口收到的PC发送的Untagged帧要能够发送至VLAN 30中，配置**port hybrid pvid vlan 30**命令设置Untagged帧加入至VLAN 30。

[S1]vlan 30

[S1-vlan30]interface Ethernet 0/0/4

[S1-Ethernet0/0/4]port hybrid pvid vlan 30

因为在华为交换上，默认所有接口都为Hybird类型接口，所以在该接口下不需要配置修改该接口为Hybird类型。

S1交换机收到VLAN 10，VLAN 20和VLAN 30的帧也要能够从该接口发送至PC，配置**port hybrid untagged vlan 10 20 30**命令使得上述三个VLAN的帧会以Untagged的方式从该接口发送出去。

[S1-Ethernet0/0/4]port hybrid untagged vlan 10 20 30

同理，端口E 0/0/2接PC-1，接口收到PC机的Untagged帧需要发送至VLAN 20，使用命令**port hybrid pvid vlan 20**。E 0/0/2接口同时也要能够被VLAN 30和VLAN 20的主机访问，即VLAN 20和30的帧能够从该接口发送出去，并以Untagged的方式发送至PC-1。

[S1]interface ethernet0/0/2

[S1-Ethernet0/0/2]port hybrid untagged vlan 20 30

接口E0/0/3收到Untagged的帧需发送至VLAN 10，同时VLAN 10和30的帧要能从该接口发送出去。

[S1]interface ethernet0/0/3

[S1-Ethernet0/0/3]port hybrid untagged vlan 10 30

VLAN10、20和30的帧要能够发送至邻居交换机S2，且要保留有原有的VLAN Tag，便于邻居交换机S2根据VLAN Tag继续转发到相应的VLAN。同样，邻居交换机S2发送过来的帧也会带有相应的VLAN Tag，所以S1与S2间互连的接口E 0/0/1配置如下。

[S1]interface ethernet 0/0/1

[S1-Ethernet0/0/1]port hybrid tagged vlan 10 20 30

[S2]interface ethernet 0/0/1

[S2-Ethernet0/0/1]port hybrid tagged vlan 10 20 30

同理在S2交换机上，E 0/0/1接口收到的带有相应VLAN Tag标记的帧，如果是VLAN 10的帧要能发送至接口E 0/0/3，如果是VLAN 20的帧要能发送至E 0/0/2。而如果是VLAN 30的帧要能发送至接口E 0/0/2和E 0/0/3。VLAN10，20和30的帧都是以Untagged的方式发送至接口E 0/0/2或E 0/0/3。反之，如果PC-3发出的Untagged的帧发送至接口E 0/0/2时会进入到Hybrid接口PVID所指明的VLAN 20中，PC-4发出的Untagged的帧发送至接口E 0/0/3时会进入到Hybrid接口PVID所指明的VLAN 10中，具体配置过程如下。

[S2]interface erhernet0/0/2

[S2-Ethernet0/0/2]port hybrid untagged vlan 20 30

[S2-Ethernet0/0/2]interface erhernet 0/0/3

[S2-Ethernet0/0/3]port hybrid untagged vlan 10 30

S1和S2上全部配置完成后，使用**ping**在IT部门的网络管理员的PC-5上测试与不同部门内的各台主机间的连通性，以PC-1为例。

PC>ping 192.168.1.1

Ping 192.168.1.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.1: bytes=32 seq=1 ttl=128 time=15 ms

From 192.168.1.1: bytes=32 seq=2 ttl=128 time<1 ms

From 192.168.1.1: bytes=32 seq=3 ttl=128 time<1 ms

From 192.168.1.1: bytes=32 seq=4 ttl=128 time=15 ms

From 192.168.1.1: bytes=32 seq=5 ttl=128 time=15 ms

--- 192.168.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/9/15 ms

可以观察到，PC-5所属网络管理员所在的VLAN 30，能够正常访问到所有其它部门的所有终端。

同理，选择市场部门所在VLAN 20内的主机PC-1，测试其访问与其它主机间的连通性。

测试PC-1与本部门内的主机PC-3间的连通性。

PC>ping 192.168.1.3

Ping 192.168.1.3: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.3: bytes=32 seq=1 ttl=128 time=31 ms

From 192.168.1.3: bytes=32 seq=2 ttl=128 time=62 ms

From 192.168.1.3: bytes=32 seq=3 ttl=128 time=46 ms

From 192.168.1.3: bytes=32 seq=4 ttl=128 time=31 ms

From 192.168.1.3: bytes=32 seq=5 ttl=128 time=47 ms

--- 192.168.1.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/43/62 ms

可以正常通信。

测试PC-1与外部门的主机PC-2和PC-4间的连通性。

PC>ping 192.168.1.2

Ping 192.168.1.2: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

……

PC>ping 192.168.1.4

Ping 192.168.1.4: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

……

不能正常通信，实现了设计要求。

测试PC-1与IT部门网络管理员主机PC-5间的连通性。

PC>ping 192.168.1.100

Ping 192.168.1.100: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.100: bytes=32 seq=1 ttl=128 time<1 ms

From 192.168.1.100: bytes=32 seq=2 ttl=128 time=16 ms

From 192.168.1.100: bytes=32 seq=3 ttl=128 time<1 ms

From 192.168.1.100: bytes=32 seq=4 ttl=128 time<1 ms

From 192.168.1.100: bytes=32 seq=5 ttl=128 time<1 ms

--- 192.168.1.100 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/3/16 ms

可以正常通信。

在交换机上可以定义多个VLAN，每个VLAN都可以看做是一个广播域，通常情况下每个VLAN都会分配一个独立的IP网络，根据需要把相应主机所在的接口划入到指定的VLAN中，并配置相应的网络IP地址，并通过VLAN间路由来实现VLAN间的互相访问，这是较为常用的方法。但是相比于基于端口的Hybrid配置，三层路由方式则不够灵活，原因在于VLAN之间的访问控制要借助于VLAN间的路由设备来实现。而控制VLAN访问使用Hybrid接口则极大的简化了配置的复杂性，它仅需在端口上自主定义基于VLAN Tag的过滤规则，来决定指定的VLAN的二层帧是否允许发送，它是通过二层来实现VLAN间的访问控制，既不需要每个VLAN定义单独的IP网段，更不需要在VLAN间引入路由设备，配置更为灵活方便。

# 思考

在本实验中，如果将PC-5所连交换机的接口E 0/0/4下的**port hybrid pvid 30**命令删除，PC-4所连的端口E 0/0/3下**port hybrid pvid 10**命令删除，其它端口配置则保持不变。此时在PC-5与PC-4间的连通性是否正常？报文经过S1和S2间端口时使用的VLAN Tag是哪个？为什么？

3.4 利用单臂路由实现VLAN间路由

# 原理概述

以太网中，通常会使用VLAN技术隔离二层广播域来减少广播的影响，并增强网络安全性和网络的可管理性，但同时也严格的隔离了不同VLAN之间的任何二层流量，分属于不同VLAN的用户不能直接互相通信。在现实中，经常会出现某些用户需要跨越VLAN实现通信的情况，单臂路由技术就是解决VLAN间通信的一种方法。

单臂路由的原理是通过一台路由器，使VLAN间互通数据通过路由器进行三层转发。如果在路由器上为每个VLAN分配一个单独的路由器物理接口，随着VLAN数量的增加，必然需要更多的接口，而路由器能提供的接口数量比较有限，所以在路由器的一个物理接口上通过配置子接口（即逻辑接口）的方式来实现以一当多的功能，将是一种非常好的方式，路由器同一物理接口的不同的子接口作为不同VLAN的缺省网关，当不同VLAN间的用户主机需要通信时，只需将数据包发送给网关，网关处理后再发送至目的主机所在VLAN，从而实现VLAN间通信。由于从拓扑结构图上看，在交换机与路由器之间，数据仅通过一条物理链路传输，故形象的称之为“单臂路由”。

# 实验目的

* + 理解单臂路由的应用场景
  + 掌握路由器子接口的配置方法
  + 掌握子接口封装VLAN的配置方法
  + 理解单臂路由的工作原理

# 实验内容

本实验模拟公司网络场景，路由器R1是公司的出口网关，员工PC通过接入层交换机（如S2和S3）接入公司网络，接入层交换机又通过汇聚交换机S1与路由器R1相连。公司内部网络通过划分不同的VLAN隔离了不同部门之间的二层通信，保证各部门间的信息安全，但是由于业务需要，经理、市场部和人事部之间需要能实现跨VLAN通信，网络管理员决定借助路由器的三层功能，通过配置单臂路由实现三个部门之间跨VLAN通信的需求。

# 实验拓扑

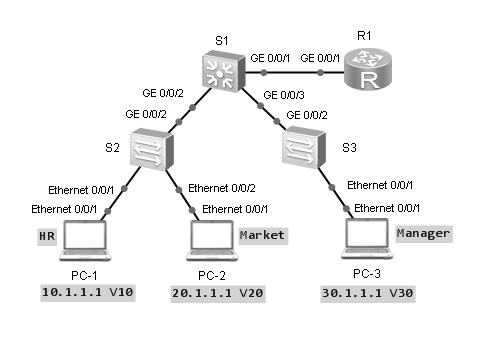


图3-5利用单臂路由实现Vlan间路由拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/1.1 | 10.1.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1.2 | 20.1.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1.3 | 30.1.1.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | 10.1.1.254 |
| PC-2 | Ethernet 0/0/1 | 20.1.1.1 | 255.255.255.0 | 20.1.1.254 |
| PC-3 | Ethernet 0/0/1 | 30.1.1.1 | 255.255.255.0 | 30.1.1.254 |

# 实验步骤

## 创建VLAN并配置Access、Trunk接口

公司为保证隔离不同部门间的二层通信，保障各部门的信息安全，规划不同部门的终端属于不同的VLAN。并为 PC配置相应IP地址。

在S2上创建VLAN 10和VLAN 20，把连接PC-1的E 0/0/1和连接PC-2的E 0/0/2接口配置为Access类型接口，并分别划分到相应的VLAN中。

[S2]vlan 10

[S2-vlan10]description HR

[S2-vlan10]vlan 20

[S2-vlan20]description Market

[S2-vlan20]interface Ethernet 0/0/1

[S2-Ethernet0/0/1]port link-type access

[S2-Ethernet0/0/1]port default vlan 10

[S2-Ethernet0/0/1]interface Ethernet 0/0/2

[S2-Ethernet0/0/2]port link-type access

[S2-Ethernet0/0/2]port default vlan 20

在S3上创建VLAN 30，把连接PC-3的E 0/0/1接口配置为Access类型接口，并划分到VLAN 30。

[S3]vlan 30

[S3-vlan30]description Manager

[S3-vlan30]interface Ethernet 0/0/1

[S3-Ethernet0/0/1]port link-type access

[S3-Ethernet0/0/1]port default vlan 30

交换机之间或交换机和路由器之间相连的接口需要传递多个VLAN信息，需要配置成为Trunk接口。

将S2和S3的GE 0/0/2接口配置成为Trunk类型接口，并允许所有VLAN通过。

[S2]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]port link-type trunk

[S2-GigabitEthernet0/0/2]port trunk allow-pass vlan all

[S3]interface GigabitEthernet 0/0/2

[S3-GigabitEthernet0/0/2]port link-type trunk

[S3-GigabitEthernet0/0/2]port trunk allow-pass vlan all

在S1上创建VLAN 10、VLAN 20和VLAN 30，并配置交换机和路由器相连的接口为Trunk，允许所有VLAN通过。

[S1]vlan 10

[S1-vlan10]vlan 20

[S1-vlan20]vlan 30

[S1-vlan30]interface GigabitEthernet 0/0/2

[S1-GigabitEthernet0/0/2]port link-type trunk

[S1-GigabitEthernet0/0/2] port trunk allow-pass vlan all

[S1-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/3

[S1-GigabitEthernet0/0/3]port link-type trunk

[S1-GigabitEthernet0/0/3] port trunk allow-pass vlan all

[S1-GigabitEthernet0/0/3]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]port link-type trunk

[S1-GigabitEthernet0/0/1] port trunk allow-pass vlan all

## 配置路由器子接口和IP地址

由于路由器R1只有一个实际的物理接口与交换机S1相连，可以在路由器上配置不同的逻辑子接口来作为不同VLAN的网关，从而达到节省路由器接口的目的。

在R1上创建子接口GE 0/0/1.1，配置IP地址10.1.1.254/24，作为人事部网关地址。

[R1]interface GigabitEthernet 0/0/1.1

[R1-GigabitEthernet0/0/1.1]ip address 10.1.1.254 24

在R1上创建子接口GE 0/0/1.2，配置IP地址20.1.1.254/24，作为市场部网关地址。

[R1]interface GigabitEthernet 0/0/1.2

[R1-GigabitEthernet0/0/1.2]ip address 20.1.1.254 24

在R1上创建子接口GE 0/0/1.3，配置IP地址30.1.1.254/24，作为经理的网关地址。

[R1]interface GigabitEthernet 0/0/1.3

[R1-GigabitEthernet0/0/1.3]ip address 30.1.1.254 24

在PC-1，PC-2和PC-3上配置IP和相应的网关地址后，在PC-1上测试与PC-2和PC-3间的连通性。

PC>ping 20.1.1.1

Ping 20.1.1.1: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

……

PC>ping 30.1.1.1

Ping 30.1.1.1: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

……

可以观察到，通信仍然无法建立。

## 配置路由器子接口封装VLAN

虽然目前已经创建了不同的子接口，并配置了相关IP地址，但是仍然无法通信。是由于处于不同VLAN下，不同网段的PC间要实现互相通信，数据包必须通过路由器进行中转。由S1发送到R1的数据都加上了VLAN标签，而路由器作为三层设备，默认无法处理带了VLAN标签的数据包。因此需要在路由器上的子接口下配置对应VLAN的封装，使路由器能够识别和处理VLAN标签，包括剥离和封装VLAN标签。

在R1的子接口GE 0/0/1.1上封装VLAN 10，在子接口GE 0/0/1.2上封装VLAN 20，在子接口GE 0/0/1.3上封装VLAN 30。并开启子接口的ARP广播功能。

使用**dot1q termination vid(802.1Q)**命令配置子接口对一层tag报文的终结功能。即配置该命令后，路由器子接口在接收带有VLAN tag的报文时，将剥掉tag进行三层转发，在发送报文时，会将与该子接口对应VLAN的VLAN tag添加到报文中。

[R1-GigabitEthernet0/0/1.1]dot1q termination vid 10

使用**arp broadcast enable**命令开启子接口的ARP广播功能。如果不配置命令，将会导致该子接口不发送携带有VLAN tag的报文(如果没有配置该命令，接口不会主动发送ARP请求报文，导致无法主动学习ARP，获得ARP表项。**arp broadcast enable**命令在V200R003C00及之前版本缺省处于未使能状态，在V200R003C01版本之后缺省处于使能状态。)。

[R1-GigabitEthernet0/0/1.1]arp broadcast enable

同理配置R1的子接口GE 0/0/1.2和GE 0/0/1/.3。

[R1]interface GigabitEthernet0/0/1.2

[R1-GigabitEthernet0/0/1.2]dot1q termination vid 20

[R1-GigabitEthernet0/0/1.2]arp broadcast enable

[R1-GigabitEthernet0/0/1.2]interface Ethernet0/0/1.3

[R1-GigabitEthernet0/0/1.3]dot1q termination vid 30

[R1-GigabitEthernet0/0/1.3]arp broadcast enable

配置完成后，在路由器R1上查看接口状态。

[R1]display ip interface brief

\*down: administratively down

……

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 unassigned down down

GigabitEthernet0/0/1 unassigned up down

GigabitEthernet0/0/1.1 10.1.1.254/24 up up

GigabitEthernet0/0/1.2 20.1.1.254/24 up up

GigabitEthernet0/0/1.3 30.1.1.254/24 up up

NULL0 unassigned up up(s)

可以观察到，三个子接口的物理状态和协议状态都正常。

查看路由器R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 13 Routes : 13

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/1.1

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1.1

10.1.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1.1

20.1.1.0/24 Direct 0 0 D 20.1.1.254 GigabitEthernet0/0/1.2

20.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1.2

20.1.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1.2

30.1.1.0/24 Direct 0 0 D 30.1.1.254 GigabitEthernet0/0/1.3

30.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1.3

30.1.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1.3

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，路由表中已经有了10.1.1.0/24、20.1.1.0/24、30.1.1.0/24的路由条目，并且都是路由器R1的直连路由，类似于路由器上的直连物理接口。

在PC-1上分别测试与网关地址10.1.1.254和PC-2间的连通性。

PC>ping 10.1.1.254

Ping 192.168.1.254: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.254: bytes=32 seq=1 ttl=255 time=47 ms

From 192.168.1.254: bytes=32 seq=2 ttl=255 time=47 ms

From 192.168.1.254: bytes=32 seq=3 ttl=255 time=47 ms

From 192.168.1.254: bytes=32 seq=4 ttl=255 time=16 ms

From 192.168.1.254: bytes=32 seq=5 ttl=255 time=16 ms

--- 192.168.1.254 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 16/34/47 ms

PC>ping 20.1.1.1

Ping 192.168.1.254: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.254: bytes=32 seq=1 ttl=255 time=47 ms

From 192.168.1.254: bytes=32 seq=2 ttl=255 time=47 ms

From 192.168.1.254: bytes=32 seq=3 ttl=255 time=47 ms

From 192.168.1.254: bytes=32 seq=4 ttl=255 time=16 ms

From 192.168.1.254: bytes=32 seq=5 ttl=255 time=16 ms

--- 192.168.1.254 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 16/34/47 ms

可以观察到，通信正常。在PC-1上Tracert PC-2。

PC>tracert 20.1.1.1

traceroute to 20.1.1.1, 8 hops max

(ICMP), press Ctrl+C to stop

1 10.1.1.254 62 ms 47 ms 31 ms

2 20.1.1.1 125 ms 94 ms 94 ms

可以观察到PC-1先把ping包发送给自身的网关10.1.1.254，然后再由网关发送到PC-2。

现以PC-1 ping PC-2为例，分析单臂路由的整个运作过程。

两台PC由于处于不同的网络中，这时PC-1会将数据包发往自己的网关，即路由器R1的子接口GE 0/0/1.1的地址10.1.1.254。

数据包到达路由器R1后，由于路由器的子接口GE 0/0/1.1已经配置了VLAN封装，当接收到PC-1发送的VLAN 10的数据帧时，发现数据帧的VLAN ID跟自身GE 0/0/1/1接口配置的VLAN ID一样，便会剥离掉数据帧的VLAN 标签后后通过三层路由转发。

通过查找路由表后，发现数据包中的目的地址20.1.1.1所属的20.1.1.0/24网段的路由条目，已经是路由器R1上的直连路由，且出接口为GE 0/0/1.2，便将该数据包发送至GE 0/0/1.2接口。

当GE 0/0/1.2接口接收到一个没有带VLAN标签的数据帧时，便会加上自身接口所配置的VLAN ID 20后再进行转发，然后通过交换机将数据帧顺利转发给PC-2。

以上便是通过配置单臂路由实现不同VLAN间的路由的整个过程。

# 思考

VLAN间的通信可以利用单臂路由的方式实现，那么利用单臂路由实现数据转发会存在哪些潜在问题？该如何解决？

3.5 利用三层交换机实现VLAN间路由

# 原理概述

VLAN将一个物理的LAN在逻辑上划分成多个广播域。VLAN内的主机间可以直接通信，而VLAN间不能直接互通。

在现实网络中，经常会遇到需要跨VLAN相互访问的情况，工程师通常会选择一些方法来实现不同VLAN间主机的相互访问，例如单臂路由。但是单臂路由技术中由于存在一些局限性，比如带宽、转发效率等，使得这项技术应用较少。

三层交换机在原有二层交换机的基础之上增加了路由功能，同时由于数据没有像单臂路由那样经过物理线路进行路由，很好的解决了带宽瓶颈的问题，为网络设计提供了一个灵活的解决方案。

# 实验目的

* + 掌握配置VLANIF接口的方法
  + 理解数据包跨VLAN路由的原理
  + 掌握测试多层交换网络连通性的方法

# 实验内容

本实验模拟企业网络场景，公司有两个部门销售部和客服部，分别规划使用VLAN 10和VLAN 20。其中销售部下有两台终端PC-1和PC-2，客服部下有一台终端PC-3。所有终端就都通过核心三层交换机S1相连。现需要该公司所有三台主机都能实现互相访问，网络管理员将通过配置三层交换机来实现。

# 实验拓扑

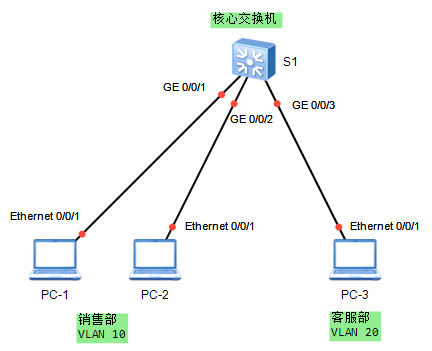


图3-6 利用三层交换机实现Vlan间路由拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 网关 |
| PC-1 | Ethernet 0/0/1 | 192.168.1.1 | 255.255.255.0 | 192.168.1.254 |
| PC-2 | Ethernet 0/0/1 | 192.168.1.2 | 255.255.255.0 | 192.168.1.254 |
| PC-3 | Ethernet 0/0/1 | 192.168.2.1 | 255.255.255.0 | 192.168.2.254 |
| S1(S5700) | VLANIF 10 | 192.168.1.254 | 255.255.255.0 | N/A |
| VLANIF 20 | 192.168.2.254 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表在PC上进行相应的基本IP地址配置，三层交换机S1上暂先不做配置。

配置完成后，测试销售部两台终端PC-1与PC-2间的连通性。

PC>ping 192.168.1.1

Ping 192.168.1.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.1: bytes=32 seq=1 ttl=128 time=31 ms

From 192.168.1.1: bytes=32 seq=2 ttl=128 time=15 ms

From 192.168.1.1: bytes=32 seq=3 ttl=128 time<1 ms

From 192.168.1.1: bytes=32 seq=4 ttl=128 time<1 ms

From 192.168.1.1: bytes=32 seq=5 ttl=128 time<1 ms

--- 192.168.1.100 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/9/31 ms

可以观察到，通信正常。

测试销售部PC-1与客服部PC-3间的连通性。

PC>ping 192.168.2.1

Ping 192.168.2.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

From 192.168.1.1: Destination host unreachable

……

PC-1与PC-3间无法正常通信，简要分析主机PC-1发出数据包，直至反馈目的无法到达的整个过程。

主机发出数据包前，将会查看数据包中的目的IP地址，如果目的IP地址和本机IP地址在同一个网段上，主机会直接发出一个ARP请求数据包来请求对方主机的MAC地址，封装数据包，继而发送该数据包。但如果目的IP地址与本机IP地址不在同一个网段，那么主机也会发出一个ARP数据包请求网关的MAC地址，收到网关ARP回复后，继而封装数据包后发送。

所以，销售部主机PC-1在访问192.168.2.1这个IP地址时发现这个目的IP地址与本机IP地址不在同一个IP地址段上，PC-1便会发出ARP数据包请求网关192.168.1.254的MAC地址。但由于交换机没有做任何IP配置，因此没有设备应答该ARP请求。导致销售部主机PC-1无法正常封装数据包，无法与客服部PC-3正常通信。

## 配置三层交换机实现VLAN间通信

通过在交换机上设置不同的VLAN使得主机实现相互隔离。在三层交换机S1上创建两个VLAN，VLAN 10 和VLAN 20，把销售部的主机全部划入VLAN 10中，客服部的主机划入VLAN 20中。

[S1]vlan 10

[S1-vlan10]vlan 20

[S1-vlan10]interface GigabitEthernet0/0/1

[S1-GigabitEthernet0/0/1]port link-type access

[S1-GigabitEthernet0/0/1]port default VLAN 10

[S1-GigabitEthernet0/0/1]interface GigabitEthernet0/0/2

[S1-GigabitEthernet0/0/2]port link-type access

[S1-GigabitEthernet0/0/2]port default VLAN 10

[S1-GigabitEthernet0/0/2]interface GigabitEthernet0/0/3

[S1-GigabitEthernet0/0/3]port link-type access

[S1-GigabitEthernet0/0/3]port default VLAN 20

现在需要通过VLAN间路由来实现通信。在三层交换机上配置VLANIF接口，VLANIF接口是基于网络层的接口，可以配置IP地址。借助VLANIF接口，三层交换机就能实现路由转发功能。

在S1上使用命令interface vlanif命令创建VLANIF接口，指定VLANIF接口所对应的VLAN ID为10，并进入VLANIF接口视图，在接口视图下配置IP地址192.168.1.254/24。再创建对应VLAN20的VLANIF接口，地址配置为192.168.2.254/24。

[S1]interface VLANif 10

[S1-VLANif10]ip address 192.168.1.254 24

[S1-VLANif10]interface VLANif 20

[S1-VLANif20]ip address 192.168.2.254 24

配置完成后，查看接口状态。

[S1]display ip interface brief

\*down: administratively down

……

Interface IP Address/Mask Physical Protocol

MEth0/0/1 unassigned down down

NULL0 unassigned up up(s)

VLANif1 unassigned down down

VLANif10 192.168.1.254/24 up up

VLANif20 192.168.2.254/24 up up

可以观察到，2个VLANIF接口已经生效。再次测试PC-1与PC-3间的连通性。

**PC>ping 192.168.2.1**

**Ping 192.168.2.1: 32 data bytes, Press Ctrl\_C to break**

**From 192.168.2.1: bytes=32 seq=1 ttl=127 time=-61765 ms**

**From 192.168.2.1: bytes=32 seq=2 ttl=127 time=-61781 ms**

**From 192.168.2.1: bytes=32 seq=3 ttl=127 time=-61766 ms**

**From 192.168.2.1: bytes=32 seq=4 ttl=127 time=-61766 ms**

**From 192.168.2.1: bytes=32 seq=5 ttl=127 time=-61781 ms**

**--- 192.168.2.1 ping statistics ---**

**5 packet(s) transmitted**

**5 packet(s) received**

**0.00% packet loss**

**round-trip min/avg/max = -61781/858931687/-61765 ms**

通信正常，实现了销售部终端与客服部终端间的通信，PC-2上的测试省略。

在PC-1上查看ARP信息。

PC>arp -a

Internet Address Physical Address Type

192.168.1.254 4C-1F-CC-63-AB-09 dynamic

可以观察到，目前PC上ARP解析到的地址只有交换机的VLANIF 10的地址，而没有对端的地址，PC-1先将数据包发送至网关，即对应的VLANIF 10接口，再由网关转发到对端。

# 思考

试问三层交换机与路由器实现三层功能的方式是否相同？为什么？

1. 生成树

4.1 STP配置和选路规则

# 原理概述

STP是用来避免数据链路层出现逻辑环路的协议。使用BPDU传递网络信息计算出一根无环的树状网络结构，并阻塞特定端口。在网络出现故障的时候，STP能快速发现链路故障，并尽快的找出另外一条路径用来进行数据传输。

交换机上运行的STP通过BPDU信息的交互，选举根交换机，然后每台非根交换机选择用来与根交换机通信的根端口，之后每个网段选择用来转发数据至根交换机的指定端口，最后剩余端口则被阻塞。

在STP工作过程中，根交换机的选举，根端口、指定端口的选举都非常重要。华为VRP提供了各种命令，用来调整STP的参数，用以优化网络。例如，交换机优先级、端口优先级、端口代价值等等。

# 实验目的

* + 理解STP的选举过程
  + 掌握修改交换机优先级的方法
  + 掌握修改端口开销值的方法

# 实验内容

公司购置了4台交换机，组建网络。考虑到网络的可靠性，将四台交换机如拓扑图搭建。由于默认情况下，交换机之间运行STP后，根交换机、根端口、指定端口的选择将基于交换机的MAC地址的大小，带来了不确定性，极可能由此产生隐患。

公司网络规划，需要S1作为主根交换机，S2作为S1的备份根交换机。同时对于S4交换机， E 0/0/1接口应该作为根端口。对于S2和S3之间的链路，应该保证S2的E 0/0/3接口作为指定端口。同时在交换机S3上，存在两个接口E 0/0/10－11连接到测试PC，测试PC经常上下线网络，需要将交换机S3与之相连的对应端口定义为边缘端口，避免测试电脑上下线对网络产生的影响。

# 实验拓扑

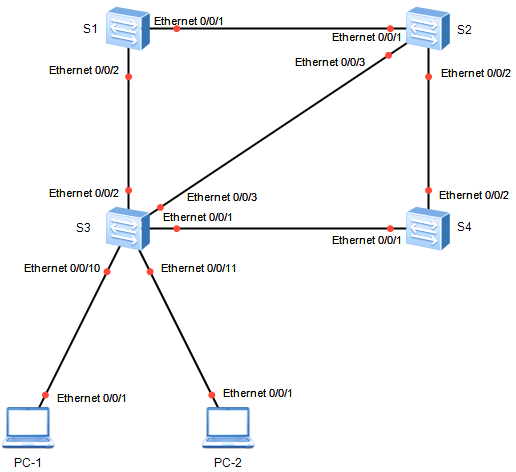


图4-1 STP配置及选举规则拓扑图

# MAC地址表

|  |  |
| --- | --- |
| 设备 | 全局MAC地址 |
| S1(S3700) | 4c1f-cceb-beac |
| S2(S3700) | 4c1f-ccbf-cbb5 |
| S3(S3700) | 4c1f-cc10-58df |
| S4(S3700) | 4c1f-ccac-3733 |

# 实验步骤

## 基本配置

根据实验拓扑图，在交换机上启用STP(华为交换机默认启用MSTP)，将交换机的STP模式更改为普通生成树STP。

[S1]stp enable

[S1]stp mode stp

[S2]stp enable

[S2]stp mode stp

[S3]stp enable

[S3]stp mode stp

[S4]stp enable

[S4]stp mode stp

配置完成后，缺省情况下需要等待30s生成树重新计算时间（15s Forward Delay加15s Learning状态时间），使用命令**display stp** 查看S1的生成树状态。 **CIST（Commonand Internal Spanning Tree）**

[S1]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :32768.4c1f-cceb-beac

……

Last TC occurred :Ethernet0/0/1

----[Port1(Ethernet0/0/1)][FORWARDING]----

Port Protocol :Enabled

Port Role :Root Port

Port Priority :128

Port Cost(Dot1T ) :Config=auto / Active=1

Designated Bridge/Port :32768.4c1f-ccbf-cbb5 / 128.1

……

BPDU Received :50

TCN: 0, Config: 50, RST: 0, MST: 0

----[Port2(Ethernet0/0/2)][DISCARDING]----

Port Protocol :Enabled

Port Role :Alternate Port

Port Priority :128

Port Cost(Dot1T ) :Config=auto / Active=1

Designated Bridge/Port :32768.4c1f-cceb-658f / 128.2

……

可以观察到S1的E 0/0/1端口为转发状态、端口角色为根端口，E 0/0/2端口为丢弃状态，端口角色Alternate，即替代端口。

还可以使用命令display stp brief在S2，S3，S4上仅查看摘要信息。

[S2]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

在交换机S2上面所有的端口为转发状态，观察到E0/0/1和E0/0/3端口角色为指定端口，E0/0/2为根端口。

[S3]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

……

在交换机S3上E 0/0/3端口角色为Alternate端口，且状态为丢弃状态，该端口将不会转发数据流量。

[S4]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

在交换机S4上所有的端口角色都为指定端口，且端口状态都为转发。

可以初步判断四台交换机中S4为根交换机，因为该交换机所有端口都为指定端口。通过命令 **display stp** 查看生成树详细信息。

[S4]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :32768.4c1f-ccac-3733

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :32768.4c1f-ccac-3733 / 0

CIST RegRoot/IRPC :32768.4c1f-ccac-3733 / 0

……

可以观察到“CIST Root”和“CIST Bridge”相同，即目前根交换机ID与自身的交换机ID相同，说明目前S4为根交换机。

生成树运算第一步就是通过比较每台交换机的交换机ID选举根交换机，交换机ID由交换机优先级和MAC地址组成。首先比较交换机优先级，数值最低的为根交换机，如果优先级一样，则比较MAC地址，同样数值最低的选举为根交换机。

目前在该公司的二层拓扑中，四台交换机的生成树都刚刚开始运行，交换机优先级都为默认值，即都相同，故根据每台交换机的MAC地址来选举，通过比较，最终S4为根交换机。

## 配置网络中的根交换机

根交换机的在网络当中的位置是非常重要的，如果选择了一台性能较差的交换机，或者是部署在接入层的交换机作为根交换机，会影响到整个网络的通信质量及数据传输。所以确定根交换机的位置极为重要，根交换机选举依据是根交换机ID（优先级与MAC地址组成），值越小越优先，交换机默认的优先级为32768，当然该值是可以修改的。

现在将S1配置为主根交换机，S2为备份根交换机，将S1的优先级改为0，S2的优先级改为4096。

[S1]stp priority 0

[S2]stp priority 4096

配置完成后查看S1和S2的STP状态信息。

[S1]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :0 .4c1f-cceb-beac

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cceb-beac / 0

CIST RegRoot/IRPC :0 .4c1f-cceb-beac / 0

……

[S2]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :4096 .4c1f-ccbf-cbb5

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cceb-beac / 1

CIST RegRoot/IRPC :4096 .4c1f-ccbf-cbb5 / 0

……

通过观察发现S1的优先级变为了0，为根交换机，而S2的优先级变为了4096，为备份根交换机。

也可以使用另外一种方式配置主根交换机和备份根交换机。（用同一种方式再次修改）。

删除在S1上所配置的优先级，使用命令**stp root primary**配置主根交换机。

[S1]undo stp priority

[S1]stp root primary

删除在S2上所配置的优先级，使用命令**stp root secondary**配置备份根交换机。

[S2]undo stp priority

[S2]stp root secondary

配置完成后查看STP的状态信息。

[S1]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :0 .4c1f-cceb-beac

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cceb-beac / 0

CIST RegRoot/IRPC :0 .4c1f-cceb-beac / 0

……

[S2]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :4096 .4c1f-ccbf-cbb5

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cceb-beac / 1

CIST RegRoot/IRPC :4096 .4c1f-ccbf-cbb5 / 0

……

可以看到此时S1自动更改优先级为0，而S2更改为4096。

## 理解根端口的选举

生成树在选举出根交换机之后，将在每台非根交换机上选举根端口。选举时首先比较该交换机上每个端口到达根交换机的根路径开销，路径开销最小的端口将成为根端口。如果根路径开销值相同，则比较每个端口所在链路上的上行交换机的交换机ID，如果该交换机ID也相同，则比较每个端口所在链路上的上行端口的端口ID。每台交换机上只能拥有一个根端口。

根据上一步骤，目前S1为主根交换机，而S2为备份根交换机，查看S4上生成树信息。

[S4]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ALTE DISCARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

可以观察到，现在S4的E 0/0/2为根端口，状态为转发状态。这是由于在S4选举根端口时，首先比较根路径开销，由于拓扑中所有链路都是相同的百兆以太网链路，S4经过S3到S1与经过S2到S1的开销值相同，接下来比较S4的两台上行链路的交换机S2和S3的交换机标识，S2目前的交换机优先级为4096，而S3为默认的32768，所以与S2连接S4的 E0/0/2接口被选为根端口。

查看S4的E 0/0/2接口开销值。

<S4>display stp interface Ethernet 0/0/2

----[Port2(Ethernet0/0/2)][FORWARDING]----

Port Protocol :Enabled

Port Role :Root Port

Port Priority :128

Port Cost(Dot1T ) :Config=auto / Active=1

Designated Bridge/Port :4096.4c1f-ccbf-cbb5 / 128.2

……

可以观察到，接口路径开销采用的是Dot1t的计算方法，Config是指手工配置的路径开销，Active是实际使用的路径开销，开销值为1。

配置S4的E 0/0/2接口的代价值为2000，即增加该接口默认的代价值。

[S4]interface ethernet0/0/2

[S4-Ethernet0/0/2]stp cost 2000

配置完成后再次查看S4的E 0/0/2接口开销值以及STP状态摘要信息。

<S4>display stp interface Ethernet 0/0/2

----[Port2(Ethernet0/0/2)][DISCARDING]----

Port Protocol :Enabled

Port Role :Alternate Port

Port Priority :128

Port Cost(Dot1T ) :Config=2000 / Active=2000

Designated Bridge/Port :4096.4c1f-ccbf-cbb5 / 128.2

……

[S4]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 ALTE DISCARDING NONE

发现此时E 0/0/1端口角色变成了根端口，而E 0/0/2变成了Alternate端口。这是由于将E 0/0/2接口的开销修改为2000之后，在选举根端口时，其到根路径开销大于E 0/0/1的根路径开销。

## 理解指定端口的选举

生成树协议在每台非根交换机选举出根端口之后，将在每个网段上选举指定端口，选举的比较规则和选举根端口类似。

现在网络管理员需要确保S2连接S3的E 0/0/3接口被选择为指定端口，可以通过修改端口开销值来实现。

为了模拟该场景，将S2的优先级恢复为默认的32768。

[S2]undo stp root (如果是根交换机，用undo stp priority 可能报错 )

配置完成后，查看S2的STP状态信息。

[S2]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :32768.4c1f-ccfa-a769

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

……

通过观察发现在删除之前所配置的优先级以后，S2的优先级已经变成了32768。

查看S2与S3的STP状态摘要信息。

[S2]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

[S3]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

0 Ethernet0/0/10 DESI FORWARDING NONE

0 Ethernet0/0/11 DESI FORWARDING NONE

通过观察发现在S2与S3间的链路上，选择了S3的E 0/0/3接口为指定端口，而S2的E 0/0/3接口为Alternate端口。这是由于在选举指定端口时，首先比较两个端口的根路径开销，目前都相同，接着比较上行交换机的交换机ID，此时S2和S3的交换机优先级相同，故比较MAC地址，最后通过比较MAC地址得出。

查看S2和S3的E 0/0/3接口信息。

<S2>display interface Ethernet 0/0/3

Ethernet0/0/3 current state : UP

……

Current system time: 2013-08-30 13:48:06-08:00

Hardware address is 4c1f-ccbf-cbb5

Last 300 seconds input rate 0 bytes/sec, 0 packets/sec

……

<S3>display interface Ethernet 0/0/3

Ethernet0/0/3 current state : UP

……

Current system time: 2013-08-30 13:49:15-08:00

Hardware address is 4c1f-cc00-58df

Last 300 seconds input rate 0 bytes/sec, 0 packets/sec

……

可以观察到，S2上E 0/0/3接口的MAC地址大于S3上E0/0/3接口的MAC地址，所以该网段上S3的E 0/0/3接口成为指定接口。

修改S3的E 0/0/2接口的开销值，将该值增大（默认为1），即增大该端口上的根路径开销，确保让S2的E 0/0/3接口成为指定端口。

[S3]interface Ethernet 0/0/2

[S3-Ethernet0/0/2]stp cost 2

配置完成后查看S2的STP状态摘要信息。

[S2]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

根据STP计算规则选择指定端口时，最终选择S2的E 0/0/3接口作为指定端口。

为了验证现在能够确保S2的E 0/0/3接口能成为指定端口，将S3的优先级调整为4096，并查看。

[S3]stp priority 4096

[S3]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :4096 .4c1f-cc00-58df

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

……

配置完成后观察到S3的优先级已调整为4096。

再次查看S2和S3的STP状态。

[S2]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

[S3]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

……

通过观察，即使修改了S3的优先级，目前比S2的优先级值更低，但是S2的E 0/0/3接口仍然为指定端口，而S3的E 0/0/3接口还是Alternate端口。验证了在选举指定端口时首先比较根路径开销的规则。

# 思考

在什么场景下，选举根端口，指定端口时会比较到端口ID？

4.2 配置STP定时器

# 原理概述

普通生成树STP不能实现快速收敛，但是在STP中诸如Hello Time定时器、Max Age定时器、Forward Delay定时器、未收到上游的BPDU就重新开始生成树计算的超时时间（3×hello time×time factor）等参数会影响其收敛速度，通过配置合适的系统参数，可以使STP实现最快的拓扑收敛。下面介绍一下在STP中定时器。

Hello Time定时器：Hello Time为周期发送BPDU来维护生成树的稳定的时间，默认为2秒。如果交换机在配置的超时时间内没有收到上游交换机发送的BPDU，则会重新进行生成树计算。在根交换机上配置的Hello Time将做为整个生成树内所有交换机的Hello Time。

Max Age定时器：BPDU的最大生存时间，默认为20秒，交换机通过比较从上游交换机收到的BPDU中携带的Message Age（配置BPDU的生存时间，如果配置BPDU是根桥发出的，则Message Age为0，每经过一台交换机增加1）和Max Age，来判断此BPDU是否超时。如果收到的BPDU超时，交换机将该BPDU老化，同时阻塞接收该BPDU的接口，并开始发出以自己为根桥的BPDU。这种老化机制可以有效的控制生成树的半径。在根交换机上配置的Max Age将做为整个生成树内所有交换机的Max Age。

Forward Delay定时器：此延迟时间为Forward Delay定时器的时间，默认为15秒。链路故障会引发网络重新进行生成树的计算，生成树的结构将发生相应的变化。不过重新计算得到的新配置消息无法立刻传遍整个网络，如果新选出的根端口和指定端口立刻就开始数据转发的话，可能会造成临时环路。为此，STP采用了一种端口状态迁移机制，新选出的根端口和指定端口要经过2倍的Forward Delay延时后才能进入转发状态，这个延时保证了新的配置消息传遍整个网络，使所有参与STP计算的交换都能正确知晓网络状态，从而防止了临时环路的产生。在华为交换机设备上，由于默认生成树模式为MSTP，当手工更改生成树模式为STP时，STP的端口状态同样只有Discarding、Learning、Forwarding三种。在根交换机上配置的延迟时间将做为整个生成树内所有交换机的延迟时间。

超时时间＝3×Hello Time×Timer Factor。如果交换机在配置的超时时间内没有收到上游发送的BPDU，就认为上游交换机已经出现故障，然后会重新进行生成树拓扑的计算。但是有时交换机在较长的时间内收不到上游发送的BPDU，是由于上游交换机的繁忙造成的，在这种情况下一般不应该重新进行生成树计算。因此，在稳定的网络中，应将超时时间配置的长一些，以减少网络资源的浪费。建议将Timer Factor的值设置为5－7，增强网络稳定性。

根交换机的Hello Time、Forward Delay以及Max Age三个时间参数之间取值应该满足如下公式，否则网络会频繁震荡。

2×( Forward Delay－1.0 second ) >= Max Age

Max Age >= 2×( Hello Time＋1.0 second )

建议使用[stp bridge-diameter](http://localhost:7890/pages/30001565/03/30001565/03/resources/s/stp_bridge-diameter.html)命令配置网络直径，交换机会自动根据网络直径计算出Hello Time、Forward Delay以及Max Age三个时间参数的最优值。不建议直接使用命令配置这三种定时器的时间。默认网络直径为7。

# 实验目的

* + 理解STP中定时器的作用
  + 掌握STP定时器的配置命令
  + 掌握查看STP定时器的生效方法
  + 理解STP定时器的最佳设置方法

# 实验内容

本实验模拟企业网络场景，公司内网是一个大的局域网，由四台交换机两两相连组成的一个环形网络。为了避免形成环路，每台交换机都运行了STP生成树协议，且配置S1为根交换机，S2为备份根交换机。现在为了优化网络，在网络变化时加快STP的收敛速度，需要在交换机上更改STP定时器的设置，将所有定时器调整到最优值，完成STP的加速收敛。

# 实验拓扑

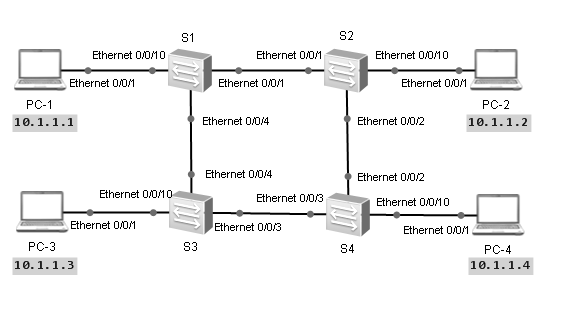


图4-2配置STP定时器拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 网关 |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |
| PC-3 | Ethernet 0/0/1 | 10.1.1.3 | 255.255.255.0 | N/A |
| PC-4 | Ethernet 0/0/1 | 10.1.1.4 | 255.255.255.0 | N/A |

# MAC地址表

|  |  |
| --- | --- |
| 设备 | 全局MAC地址 |
| S1(S3700) | 4c1f-cc73-c72d |
| S2(S3700) | 4c1f-ccf4-3d05 |
| S3(S3700) | 4c1f-cca5-443e |
| S4(S3700) | 4c1f-ccb9-8907 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

PC>ping 10.1.1.2

Ping 10.1.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: bytes=32 seq=1 ttl=128 time=47 ms

From 10.1.1.2: bytes=32 seq=2 ttl=128 time=32 ms

From 10.1.1.2: bytes=32 seq=3 ttl=128 e=78 ms

From 10.1.1.2: bytes=32 seq=4 ttl=128 time=79 ms

From 10.1.1.2: bytes=32 seq=5 ttl=128 time=46 ms

--- 10.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 32/56/79 ms

其余直连网段的连通性测试省略。

## 配置STP定时器

在四台交换机上配置使用STP，并配置S1为该二层网络中的根交换机，S2为备份根交换机。

<S1>system-view

[S1]stp enable

[S1]stp mode stp

[S1]stp root primary

<S2>system-view

[S2]stp enable

[S2]stp mode stp

[S2]stp root secondary

<S3>system-view

[S3]stp enable

[S3]stp mode stp

<S4>system-view

[S4]stp enable

[S4]stp mode stp

配置完成后，使用命令**display stp**查看各定时器的默认值。

<S1>display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :0 .4c1f-cc73-c72d

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cc73-c72d / 0

……

可以查看到在默认情况下，BPDU每2秒发送一次（hello），BPDU的最大老化时间为20秒（MaxAge），转发延迟为15秒（FwDly），最大传递跳数为20跳（MaxHop）。注意，Config Times标识的是当前设备配置的计时器，而Active Times标识的是正在生效的计时器，一般情况下他们二者是完全相同的。

在PC-4使用命令**ping -t**持续发送ICMP报文，进行连通性测试。

PC>ping 10.1.1.2 -t

Ping 10.1.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: bytes=32 seq=1 ttl=128 time=109 ms

From 10.1.1.2: bytes=32 seq=2 ttl=128 time=109 ms

From 10.1.1.2: bytes=32 seq=3 ttl=128 time=79 ms

From 10.1.1.2: bytes=32 seq=4 ttl=128 time=78 ms

From 10.1.1.2: bytes=32 seq=5 ttl=128 time=109 ms

……

可以观察到，此时网络稳定，没有出现任何丢包现象。

在S1上修改STP的Forwad Delay时间为2000厘秒，默认为1500厘秒。注意，只有在根交换机上进行该配置才会生效。

<S1>system-view

[S1]stp timer forward-delay 2000

配置完成后，交换机会弹出信息，提示配置已经被改变。

[S1]Jun 21 2013 05:57:28-08:00 S1 DS/4/DATASYNC\_CFGCHANGE:OID 1.3.6.1.4.1.2011.5.25.191.3.1 configurations have been changed.

使用命令**display stp**查看此时的定时器值。

<S1>display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :0 .4c1f-cc73-c72d

Config Times :Hello 2s MaxAge 20s FwDly 20s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 20s MaxHop 20

CIST Root/ERPC :0 .4c1f-cc73-c72d / 0

……

可以观察到，此时修改已经完成。如果在非根交换机上配置，那么“Config Times”配置值会发生改变，而“Active Timers”，实际运行值不会改变。

再回到PC-4上观察到PC-2的连通性测试结果。

From 10.1.1.2: bytes=32 seq=46 ttl=128 time=78 ms

From 10.1.1.2: bytes=32 seq=47 ttl=128 time=109 ms

From 10.1.1.2: bytes=32 seq=48 ttl=128 time=110 ms

From 10.1.1.2: bytes=32 seq=49 ttl=128 time=78 ms

Request timeout!

Request timeout!

Request timeout!

……

观察到出现大量丢包现象。更改STP的Hello Time时间及其他计时器也会出现相同的现象，这里不再赘述。

所以不建议使用命令直接修改定时器时间，建议使用[**stp bridge-diameter**](http://localhost:7890/pages/30001565/03/30001565/03/resources/s/stp_bridge-diameter.html)设置网络直径，交换机会根据网络直径自动计算出三个时间参数的最优值。注意，本命令需要在根交换机上配置才能生效。

在S1上使用命令**stp bridge-diameter 3**设置网络的直径为3。

<S1>system-view

[S1]stp bridge-diameter 3

配置完成后，观察STP计时器的改变情况。

[S1]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :0 .4c1f-cc73-c72d

Config Times :Hello 2s MaxAge 12s FwDly 9s MaxHop 20

Active Times :Hello 2s MaxAge 12s FwDly 9s MaxHop 20

……

可以观察到，此时最大老化时间被自动修改为12秒，转发延迟被自动修改为9秒。

同时对PC-4到PC-2连通性测试结果再次进行观察。

From 10.1.1.2: bytes=32 seq=18 ttl=128 time=63 ms

From 10.1.1.2: bytes=32 seq=19 ttl=128 time=78 ms

Request timeout!

Request timeout!

……

Request timeout!

Request timeout!

From 10.1.1.2: bytes=32 seq=30 ttl=128 time=78 ms

From 10.1.1.2: bytes=32 seq=31 ttl=128 time=78 ms

From 10.1.1.2: bytes=32 seq=32 ttl=128 time=62 ms

可以观察到，此时网络恢复了正常。

## 验证Forward Delay定时器

为了验证forward-delay时间对端口状态迁移的影响，仍然维持上一步骤中PC-4到PC-2的连通性测试。

在S1，S2，S3，S4上查看STP下的各个端口的状态。

<S1>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/4 DESI FORWARDING NONE

0 Ethernet0/0/10 DESI FORWARDING NONE

可以观察到，由于S1是根交换机，所以S1的所有端口都是DP端口即指定端口，所处状态都是转发状态。

同理观察其他交换机的STP接口状态。

<S2>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

0 Ethernet0/0/10 DESI FORWARDING NONE

<S3>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/3 DESI FORWARDING NONE

0 Ethernet0/0/4 ROOT FORWARDING NONE

0 Ethernet0/0/10 DESI FORWARDING NONE

<S4>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

0 Ethernet0/0/10 DESI FORWARDING NONE

可以观察到，此时S2和S3接口都属于转发状态，S4的E 0/0/2接口为根端口，E 0/0/3接口处于阻塞状态。

现在将S4的E 0/0/2接口关闭，使其E 0/0/3接口成为新的根端口。

请注意，在华为交换机上，当从MSTP模式切换到STP模式，运行STP协议的设备上端口支持的端口状态仍然保持和MSTP支持的端口状态一样，支持的状态仅包括Forwarding、Learning和Discarding。又由于华为交换机上默认的STP模式为MSTP，故本实验中，STP仅支持三个状态，S4的E 0/0/3接口会从Discarding状态，再经过Learning过渡状态，最终到Forwarding状态，只需经历一个Forward Delay的时间。

<S4>system-view

[S4]interface Ethernet 0/0/2

[S4-Ethernet0/0/2]shutdown

配置完成后，观察连通性测试结果。

From 10.1.1.2: bytes=32 seq=11 ttl=128 time=109 ms

From 10.1.1.2: bytes=32 seq=12 ttl=128 time=109 ms

From 10.1.1.2: bytes=32 seq=13 ttl=128 time=47 ms

From 10.1.1.2: bytes=32 seq=14 ttl=128 time=47 ms

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

From 10.1.1.2: bytes=32 seq=15 ttl=128 time=47 ms

From 10.1.1.2: bytes=32 seq=16 ttl=128 time=63 ms

From 10.1.1.2: bytes=32 seq=17 ttl=128 time=32 ms

可以观察到，此时丢失了9个数据包。这是因为根据上一步骤的配置结果，Forward Delay时间为9秒(发包间隔默认的是1000毫秒，也就是1秒。？？？)，即S4上该端口从Discarding状态经过Learning状态，最终到Forwarding状态需要一个Forwad Delay的时间间隔。

恢复S4的E 0/0/2接口，并在根交换机R1上更改网络直径为缺省值7。

<S4>system-view

[S4]interface Ethernet 0/0/2

[S4-Ethernet0/0/2]undo shutdown

<S1>system-view

[S1]stp bridge-diameter 7

配置完成后，查看S1上的STP信息。

[S1]display stp

-------[CIST Global Info][Mode STP]-------

CIST Bridge :0 .4c1f-cc73-c72d

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

……

可以观察到，Forward时间已被自动修改为15秒。

现在采用相同的方法，关闭S4上的E 0/0/2接口，测试丢包情况。

<S4>system-view

[S4]interface Ethernet 0/0/2

[S4-Ethernet0/0/2]shutdown

配置完成后，观察连通性测试结果。

From 10.1.1.2: bytes=32 seq=33 ttl=128 time=53 ms

From 10.1.1.2: bytes=32 seq=34 ttl=128 time=17 ms

From 10.1.1.2: bytes=32 seq=35 ttl=128 time=30 ms

From 10.1.1.2: bytes=32 seq=36 ttl=128 time=45 ms

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

From 10.1.1.2: bytes=32 seq=37 ttl=128 time=31 ms

From 10.1.1.2: bytes=32 seq=38 ttl=128 time=37 ms

From 10.1.1.2: bytes=32 seq=39 ttl=128 time=52 ms

可以观察到，丢包共17个。因为在Forward Delay时间变为15秒后，端口状态迁移从Discarding状态到Forwarding状态要经过一个Forward Delay的时间间隔。

## 思考

交换机端口在发生状态转换时，都有哪些状态会经历一个Forward Delay？

4.3 RSTP基础配置

# 原理概述

IEEE于2001年发布的802.1W标准定义了RSTP（Rapid Spanning-Tree Protocol，快速生成树协议），该协议基于STP协议，对原有的STP协议进行了更加细致的修改和补充。

STP协议虽然能够解决环路问题，但是STP也存在一些不足，比如STP没有细致区分端口状态和端口角色；其次STP端口状态共有五种，即Discarding、Blocking、Listening、Learning和Forwarding，收敛较慢。而且对于用户来说Listening、Learning和Blocking状态并没有区别，都不转发流量。根据STP的不足，RSTP做出了改进。

RSTP新增加了2种端口角色。RSTP的端口角色共有4种：根端口、指定端口、Alternate端口和Backup端口。根端口和指定端口的作用与STP协议中相同，Alternate端口和Backup端口的描述如下：

Alternate端口就是由于学习到其它网桥发送的配置BPDU报文而阻塞的端口，Alternate端口提供了从指定桥到根的另一条可切换路径，作为根端口的备份端口；

Backup端口就是由于学习到自身发送的配置BPDU报文而阻塞的端口，Backup端口作为指定端口的备份，提供了另一条从根桥到相应网段的备份通路。

RSTP把原来的5种状态缩减为3种。根据端口是否转发用户流量和学习MAC地址来划分：如果不转发用户流量也不学习MAC地址，那么端口状态就是Discarding状态。如果不转发用户流量但是学习MAC地址，那么端口状态就是Learning状态。如果既转发用户流量又学习MAC地址，那么端口状态就是Forwarding状态。

RSTP的快速收敛机制可分为以下三种：

Proposal/Agreement机制：当一个端口被选举成为指定端口之后，在STP中，该端口至少要等待一个Forward Delay（Learning）时间才会迁移到Forwarding状态。而在RSTP中，此端口会先进入Discarding状态，再通过Proposal/Agreement机制快速进入Forward状态。这种机制必须在点到点全双工链路上使用。

根端口快速切换机制：如果网络中一个根端口失效，那么网络中最优的Alternate端口将成为根端口，进入Forwarding状态。因为通过这个Alternate端口连接的网段上必然有个指定端口可以通往根桥。

边缘端口的引入：在RSTP里面，如果某一个指定端口位于整个网络的边缘，即不再与其他交换设备连接，而是直接与终端设备直连，这种端口叫做边缘端口。边缘端口不接收处理配置BPDU，不参与RSTP运算，可以由Disable直接转到Forwarding状态，且不经历时延，就像在端口上将STP禁用。但是一旦边缘端口收到配置BPDU，就丧失了边缘端口属性，成为普通STP端口，并重新进行生成树计算，从而引起网络震荡。

# 实验目的

* + 理解RSTP的应用场景
  + 掌握RSTP的基本配置
  + 掌握RSTP的边缘端口的应用
  + 理解RSTP备份端口

# 实验内容

本实验模拟公司网络场景，S3和S4是接入层交换机，负责用户的接入，S1和S2是汇聚层交换机，四台交换机组成一个环形网络。为了防止网络中出现环路，产生网络风暴，所有交换机上都需要运行生成树协议，同时为了加快网络收敛速度，网络管理员选择使用RSTP协议，且使得性能较好的S1为根交换机，S2为次根交换机，并配置边缘端口进一步优化公司网络。

# 实验拓扑

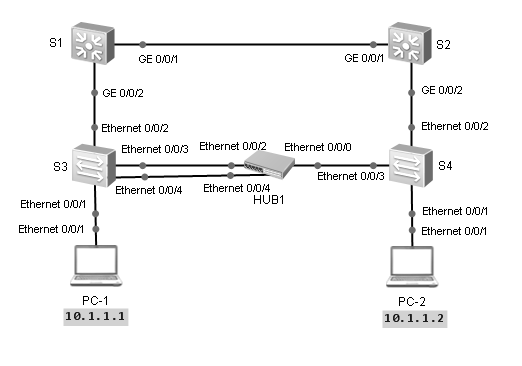


图4-3 RSTP基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |

# MAC地址表

|  |  |
| --- | --- |
| 设备 | 全局MAC地址 |
| S1(S3700) | 4c1f-cc33-9812 |
| S2(S3700) | 4c1f-cc4a-bea9 |
| S3(S3700) | 4c1f-cc32-b9d7 |
| S4(S3700) | 4c1f-cc10-279a |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。PC-1的配置如下图，PC-2的配置相同，省略。



图4-4

配置完成后，测试主机间的连通性。

PC>ping 10.1.1.2

Ping 10.1.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: bytes=32 seq=1 ttl=128 time=47 ms

From 10.1.1.2: bytes=32 seq=2 ttl=128 time=62 ms

From 10.1.1.2: bytes=32 seq=3 ttl=128 time=32 ms

From 10.1.1.2: bytes=32 seq=4 ttl=128 time=78 ms

From 10.1.1.2: bytes=32 seq=5 ttl=128 time=31 ms

--- 10.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/50/78 ms

可以观察到，连通性测试成功。

## 配置RSTP基本功能

RSTP协议能够消除网络环路，同时能加快网络的收敛速度。现在需要在公司所有交换机上都配置RSTP协议。

在汇聚层交换机S1、S2及接入层交换机S3、S4上，把生成树模式由默认的MSTP改为RSTP。由于华为交换机上默认即开启了MSTP，故只需修改生成树模式即可。

[S1]stp mode rstp

Info: This operation may take a few seconds. Please wait for a moment...done.

[S2]stp mode rstp

Info: This operation may take a few seconds. Please wait for a moment...done.

[S3]stp mode rstp

Info: This operation may take a few seconds. Please wait for a moment...done.

[S4]stp mode rstp

Info: This operation may take a few seconds. Please wait for a moment...done.

配置完成后，在交换机S1、S2、S3和S4上都使用命令**display stp**去查看生成树的模式及根交换机的位置。

<S1>display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :32768.4c1f-cc33-9812

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :32768.4c1f-cc10-279a / 2

CIST RegRoot/IRPC :32768.4c1f-cc33-9812 / 0

……

<S2>display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :32768.4c1f-cc4a-bea9

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :32768.4c1f-cc10-279a / 1

CIST RegRoot/IRPC :32768.4c1f-cc4a-bea9 / 0

……

<S3>display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :32768.4c1f-cc32-b9d7

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :32768.4c1f-cc10-279a / 1

CIST RegRoot/IRPC :32768.4c1f-cc32-b9d7 / 0

……

<S4>display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :32768.4c1f-cc10-279a

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :32768.4c1f-cc10-279a / 0

CIST RegRoot/IRPC :32768.4c1f-cc10-279a / 0

……

上述信息中，CIST Bridge是交换机自己的交换机ID，而CIST Root是根交换机的交换机ID。根交换机是交换机中交换机ID最小的交换机，所以，观察可知，S4是当前的根交换机。

在RSTP构建的树形拓扑中，网络管理员需要汇聚层主交换机S1为根交换机，而汇聚层交换机S2为备份根交换机。把S1设置成为根交换机，同时设置S2交换机为次根交换机。

[S1]stp root primary

[S2]stp root secondary

配置完成后，同样在S1上使用命令**display stp**去观察。

[S1]display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :0 .4c1f-cc33-9812

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cc33-9812 / 0

CIST RegRoot/IRPC :0 .4c1f-cc33-9812 / 0

CIST RootPortId :0.0

BPDU-Protection :Disabled

CIST Root Type :Primary root

TC or TCN received :33

……

可以观察到，**stp root primary**命令修改的是交换机ID中的交换机优先级，把默认的优先级由32768改为0，所以S1的交换机ID变为最小，是Primary root，即为根交换机。

在S2上使用命令**display stp**去观察。

[S2]display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :4096 .4c1f-cc4a-bea9

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cc33-9812 / 1

CIST RegRoot/IRPC :4096 .4c1f-cc4a-bea9 / 0

CIST RootPortId :128.1

BPDU-Protection :Disabled

CIST Root Type :Secondary root

TC or TCN received :23

……

可以观察到，**stp root secondary**命令修改的也是交换机ID中的交换机优先级，把默认的优先级由32768改为4096，使S2的桥ID变为次小，是Secondary root，即次根交换机。

在S3和S4上使用命令**display stp**去观察。

<S3>display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :32768.4c1f-cc32-b9d7

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cc33-9812 / 1

CIST RegRoot/IRPC :32768.4c1f-cc32-b9d7 / 0

……

[S4]display stp

-------[CIST Global Info][Mode RSTP]-------

CIST Bridge :32768.4c1f-cc10-279a

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :0 .4c1f-cc33-9812 / 2

CIST RegRoot/IRPC :32768.4c1f-cc10-279a / 0

……

可以观察到，S3和S4交换机的交换机优先级保持默认的32768，且都把S1当做根交换机。

继续使用**display stp brief**命令查看每台交换机上的端口角色及状态。

<S1>display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/1 DESI FORWARDING NONE

0 GigabitEthernet0/0/2 DESI FORWARDING NONE

根交换机S1上无根端口，所有端口都是指定端口。

<S2>display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/1 ROOT FORWARDING NONE

0 GigabitEthernet0/0/2 DESI FORWARDING NONE

交换机S2上的GE 0/0/1是根端口。

<S3>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

0 Ethernet0/0/4 BACK DISCARDING NONE

交换机S3上的E 0/0/2是根端口，E 0/0/3是指定端口，而E 0/0/4是备份端口。

<S4>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

交换机S4上的E 0/0/2是根端口，E 0/0/3是替代端口。

通过下面的操作，观察S2上端口的状态变化。

目前S2的GE 0/0/1端口是根端口，其它所有端口是指定端口。

[S2]dis stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/1 ROOT FORWARDING NONE

0 GigabitEthernet0/0/2 DESI FORWARDING NONE

如果S2的根端口down了，S2会选择把其它到达根交换机的端口置成根端口。RSTP协议的收敛比较快，端口GE 0/0/2会快速协商成为新的根端口，协商期间端口是discarding状态，协商结束后端口为forwarding状态，这个过程所需要的时间非常短，这就是RSTP收敛快的一个表现。

模拟根端口断掉的过程，把S2的GE 0/0/1端口使用**shutdown**关闭，同时，使用命令**display stp brief，**观察S2上其它端口的角色及状态的变化。

[S2]interface GigabitEthernet0/0/1

[S2-GigabitEthernet0/0/1]shutdown

Jun 26 2013 01:01:24-08:00 S2 %%01PHY/1/PHY(l)[2]: GigabitEthernet0/0/1: chang

e status to down

[S2-GigabitEthernet0/0/1]display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/2 DESI DISCARDING NONE

可以观察到，端口GE 0/0/2的角色还是指定端口，但状态是discarding。再次**display stp brief**时，就会观察到端口的角为根端口，且处于转发状态。

[S2-GigabitEthernet0/0/1]display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/2 ROOT FORWARDING NONE

观察结束之后，恢复端口。

[S2-GigabitEthernet0/0/1]undo shutdown

Jun 26 2013 01:01:47-08:00 S2 %%01PHY/1/PHY(l)[4]: GigabitEthernet0/0/1: chang

e statu

[S2-GigabitEthernet0/0/1]display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/1 ROOT FORWARDING NONE

0 GigabitEthernet0/0/2 DESI DISCARDING NONE

可以观察到，端口GE 0/0/2的角色是指定端口，状态是discarding。再次**display stp brief**时，就会观察到GE 0/0/2会经历discarding状态回到forwarding状态。

[S2-GigabitEthernet0/0/1]display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/1 ROOT FORWARDING NONE

0 GigabitEthernet0/0/2 DESI FORWARDING NONE

当拓扑发生变化时，RSTP使用P/A机制和根端口快速切换机制使端口状态立即从Discarding立即进入Forwarding状态，缩短了收敛的时间，减小对网络通信的影响。

## 配置边缘端口

生成树的计算主要发生在交换机互连的链路之上，而连接PC的端口没有必要参与生成树计算，为了优化网络，降低生成树计算对终端设备的影响，现网络管理员把交换机上连接PC的接口配置为边缘端口。

作为对比，在将S4上的E 0/0/1配置为边缘端口之前，先把端口关闭再开启，观察端口状态的变化。

[S4]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

[S4]interface Ethernet0/0/1

[S4-Ethernet0/0/1]shutdown

[S4-Ethernet0/0/1]undo shutdown

[S4-Ethernet0/0/1]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI DISCARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

可以观察到初始状态为discarding，15秒之后，接口将进入learning状态。

[S4-Ethernet0/0/1]dis stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI LEARNING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

保持在Learning状态15s后，接口最终进入到forwarding状态。

[S4-Ethernet0/0/1]dis stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

所以一个接口如果参与生成树计算，要经过discarding和learning状态，30s后，接口才最终进入转发状态。

配置S4上连接PC的端口为边缘端口，此时生成树计算工作依然进行，但端口进入转发状态无需等待30秒。

[S4]interface Ethernet0/0/1

[S4-Ethernet0/0/1]stp edged-port enable

在S4上，做同样的模拟过程，关闭掉E 0/0/1接口，再重新开启此端口，观察边缘端口E 0/0/1的状态变化。

[S4-Ethernet0/0/1]shutdown

[S4-Ethernet0/0/1]undo shutdown

[S4-Ethernet0/0/1]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

可以观察到，接口立刻进入到forwarding状态，没有30秒的延迟。

在使用RSTP的环境中，可以在交换机上把连接PC，路由器和防火墙的端口都配置为边缘端口，边缘端口能降低终端设备访问网络需要等待的时间，明显提高网络的可用性。

## 查看备份端口状态

网络管理员在S3与S4之间加了一台Hub设备，并将S3的E 0/0/4通过Hub与S4相连。

在S3上使用命令**display stp brief**。

[S3]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

0 Ethernet0/0/4 BACK DISCARDING NONE

可以观察到，S3的E 0/0/3接口为指定端口，而同交换机上的E 0/0/4为备份端口，两个接口接到同一台Hub上，当E 0/0/3接口down之后，E 0/0/4会成为新的指定端口。

在S3上关闭E 0/0/3接口，通过**display stp brief**命令查看备份端口的状态变化。

[S3]interface Ethernet 0/0/3

[S3-Ethernet0/0/3]shutdown

[S3-Ethernet0/0/3]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/4 BACK DISCARDING NONE

[S3-Ethernet0/0/3]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/4 DESI LEARNING NONE

[S3-Ethernet0/0/3]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/4 DESI FORWARDING NONE

[S3-Ethernet0/0/3]

可以观察到，S3上的指定接口断掉后，E 0/0/4接口角色发生变化，状态会由discarding，learning最终到forwarding状态，指定接口现在是E 0/0/4，指定交换机还是S3，S3仍然为Hub所在的网段提供访问其它交换机的数据访问路径。

相似的过程，在S4上，接口E 0/0/2是根端口，接口E 0/0/3是替代端口，discarding状态。当S4的根端口E 0/0/2 down之后，接口E 0/0/3会立即替代E 0/0/2成为新的根端口。

[S4]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 ALTE DISCARDING NONE

把S4上的根端口E 0/0/2关闭掉，观察替代端口E 0/0/3的状态及角色的变化。

[S4]interface ethernet0/0/2

[S4-Ethernet0/0/2]shutdown

[S4-Ethernet0/0/2]display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/3 ROOT FORWARDING NONE

RSTP协议收敛很快，所以替代端口立即成为根端口。

在RSTP中，Alternate端口和Backup端口角色所对应的最终端口状态都是discarding。区别是Alternate端口用于为根端口做备份，而Backup端口用于为本交换机上的指定端口做备份，所以当相应的根端口或指定端口断掉后，备份端口会立即承担原有的根端口或指定端口的角色，开始转发数据。

RSTP协议是对STP的升级，它重新划定端口的角色及状态，使用更快速的握手协商机制，降低了收敛时间。使它成为继STP协议后，首选的生成树协议，不足之处就是在同一网络内的交换机上所有的VLAN共用同样的拓扑，此时可以使用MSTP来优化。

# 思考

S4交换机的E 0/0/2接口down之后，E 0/0/3会成为新的根端口，如果此时S3交换机的指定端口E 0/0/3也down掉，S4交换机上会发生端口角色或状态的改变吗？如果边缘端口收到BPDU，此端口还是边缘端口吗？

4.4 MSTP基础配置

# 原理概述

RSTP在STP基础上进行了改进，实现了网络拓扑快速收敛。但RSTP和STP还存在同一个缺陷，即由于局域网内所有的VLAN共享一棵生成树，链路被阻塞后将不承载任何流量，造成带宽浪费，因此无法在VLAN间实现数据流量的负载均衡，还有可能造成部分VLAN的报文无法转发。

通过MSTP把一个交换网络划分成多个域，每个域内形成多棵生成树，生成树之间彼此独立。每个域叫做一个MST域（MST Region：Multiple Spanning Tree Region），每棵生成树叫做一个多生成树实例MSTI（Multiple Spanning Tree Instance）。

实例内可以包含多个VLAN。通过将多个VLAN映射到同一个实例内，可以节省通信开销和资源占用率。MSTP各个实例拓扑的生成树计算相互独立，通过这些实例可以实现负载均衡。把多个相同拓扑结构的VLAN映射到一个实例里，这些VLAN在端口上的转发状态取决于端口在对应MSTP实例的状态。

MSTP通过设置VLAN映射表（即VLAN和MSTI的对应关系表），把VLAN和MSTI联系起来。每个VLAN只能对应一个MSTI，即同一VLAN的数据只能在一个MSTI中传输，而一个MSTI可能对应多个VLAN。

# 实验目的

* + 掌握MSTP的基础配置
  + 掌握配置MSTP多实例的方法
  + 掌握配置MSTP实现流量分担的方法
  + 理解MSTP与STP，RSTP的区别

# 实验内容

某公司二层网络由三台交换机S1，S2，S3组成。交换机S1与S2在一个楼层，S3在另一楼层。PC-1与PC-2属于HR部门，划入VLAN 10，PC-3与PC-4属于IT部门，划入VLAN 20。当使用普通STP时，为防止环路产生，STP将会阻塞一条链路，导致该链路闲置。为了保证所有链路都能充分利用，使流量能够分担，网络管理员通过配置MSTP来实现。

# 实验拓扑

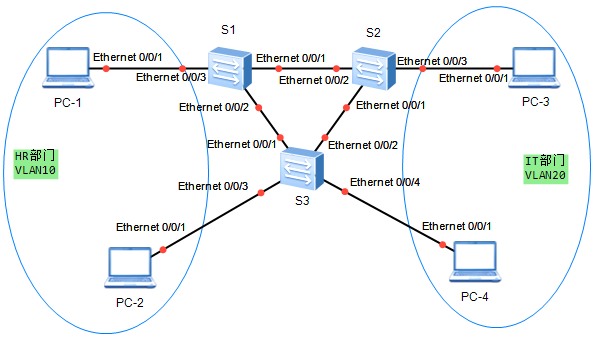


图4-5 MSTP基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 192.168.10.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 192.168.10.2 | 255.255.255.0 | N/A |
| PC-3 | Ethernet 0/0/1 | 192.168.20.1 | 255.255.255.0 | N/A |
| PC-4 | Ethernet 0/0/1 | 192.168.20.2 | 255.255.255.0 | N/A |

# MAC地址表

|  |  |
| --- | --- |
| 设备 | 全局MAC地址 |
| S1(S3700) | 4c1f-cc4d-0fcf |
| S2(S3700) | 4c1f-cc5e-3ea9 |
| S3(S3700) | 4c1f-ccf8-067c |

# 实验步骤

## 基础配置

根据编址表，在各台PC上配置IP 地址。

在交换机S1，S2，S3上创建VLAN 10与20，并将连接PC的端口配置成为Access类型接口，划入相应VLAN。交换机间的接口配置成为Trunk接口，允许所有VLAN 通过。

[S1]vlan batch 10 20

[S1]interface Ethernet0/0/3

[S1-Ethernet0/0/3]port link-type access

[S1-Ethernet0/0/3]port default vlan 10

[S1-Ethernet0/0/3]interface Ethernet0/0/1

[S1-Ethernet0/0/1]port link-type trunk

[S1-Ethernet0/0/1]port trunk allow-pass vlan all

[S1-Ethernet0/0/1]interface Ethernet0/0/2

[S1-Ethernet0/0/2]port trunk allow-pass vlan all

[S2]vlan batch 10 20

[S2]interface Ethernet0/0/3

[S2-Ethernet0/0/3]port link-type access

[S2-Ethernet0/0/3]port default vlan 20

[S2-Ethernet0/0/3]interface Ethernet0/0/2

[S2-Ethernet0/0/2]port link-type trunk

[S2-Ethernet0/0/2]port trunk allow-pass vlan all

[S2-Ethernet0/0/2]interface Ethernet0/0/1

[S2-Ethernet0/0/1]port link-type trunk

[S2-Ethernet0/0/1]port trunk allow-pass vlan all

[S3]vlan batch 10 20

[S3]interface Ethernet0/0/3

[S3-Ethernet0/0/3]port link-type access

[S3-Ethernet0/0/3]port default vlan 10

[S3-Ethernet0/0/3]interface Ethernet0/0/4

[S3-Ethernet0/0/4]port link-type access

[S3-Ethernet0/0/4]port default vlan 20

[S3-Ethernet0/0/4]interface Eth0/0/1

[S3-Ethernet0/0/1]port link-type trunk

[S3-Ethernet0/0/1]port trunk allow-pass vlan all

[S3-Ethernet0/0/1]interface Ethernet0/0/2

[S3-Ethernet0/0/2]port link-type trunk

[S3-Ethernet0/0/2]port trunk allow-pass vlan all

## 理解MSTP的运行机制及验证单实例

当网络管理员按照设计搭建完公司二层网络后，启动设备。在华为交换机上默认即运行MSTP协议。

在S1上使用命令**display stp**查看生成树的状态和统计信息。

<S1>display stp

-------[CIST Global Info][Mode MSTP]-------

CIST Bridge :32768.4c1f-cc4d-0fcf

Config Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

Active Times :Hello 2s MaxAge 20s FwDly 15s MaxHop 20

CIST Root/ERPC :32768.4c1f-cc4d-0fcf / 0

CIST RegRoot/IRPC :32768.4c1f-cc4d-0fcf / 0

CIST RootPortId :0.0

BPDU-Protection :Disabled

TC or TCN received :5

TC count per hello :0

STP Converge Mode :Normal

Time since last TC :0 days 0h:24m:48s

Number of TC :6

Last TC occurred :Ethernet0/0/1

----[Port1(Ethernet0/0/1)][FORWARDING]----

Port Protocol :Enabled

……

可以观察到，在CIST全局信息中，显示目前STP模式为MSTP，根交换机为S1自身，另外还有交换机各个接口上的STP信息。

使用命令**display stp brief**查看S1，S2，S3上生成树的状态和统计的摘要信息。

<S1>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

<S2>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

<S3>display stp brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 ALTE DISCARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

0 Ethernet0/0/4 DESI FORWARDING NONE

可以观察到，此时S1上的端口都为指定端口，且都处于转发状态，为根交换机。S3上的E 0/0/2为替代端口，处于丢弃状态。MSTID，即MSTP的实例ID，三台交换机上目前都为0，即在默认情况下，所有VLAN都处于MSTP实例0中。

假如网络管理员配置STP模式为RSTP，最终选举出来的根交换机及被阻塞的端口等结果将和目前MSTP的选举结果一致，即在MSTP的单个实例中，选举规则与RSTP一致，端口角色与状态与RSTP也一致。

在HR部门的PC-2上持续发送ping包至PC-1，在IT部门的PC-4上持续发送ping包至PC-3。

PC>ping 192.168.10.1 -t

Ping 192.168.10.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.10.1: bytes=32 seq=1 ttl=128 time=31 ms

From 192.168.10.1: bytes=32 seq=2 ttl=128 time=47 ms

……

PC>ping 192.168.20.1 -t

Ping 192.168.20.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.1: bytes=32 seq=1 ttl=128 time=47 ms

From 192.168.20.1: bytes=32 seq=2 ttl=128 time=62 ms

……

同时，在S3的E 0/0/1接口上抓包观察。

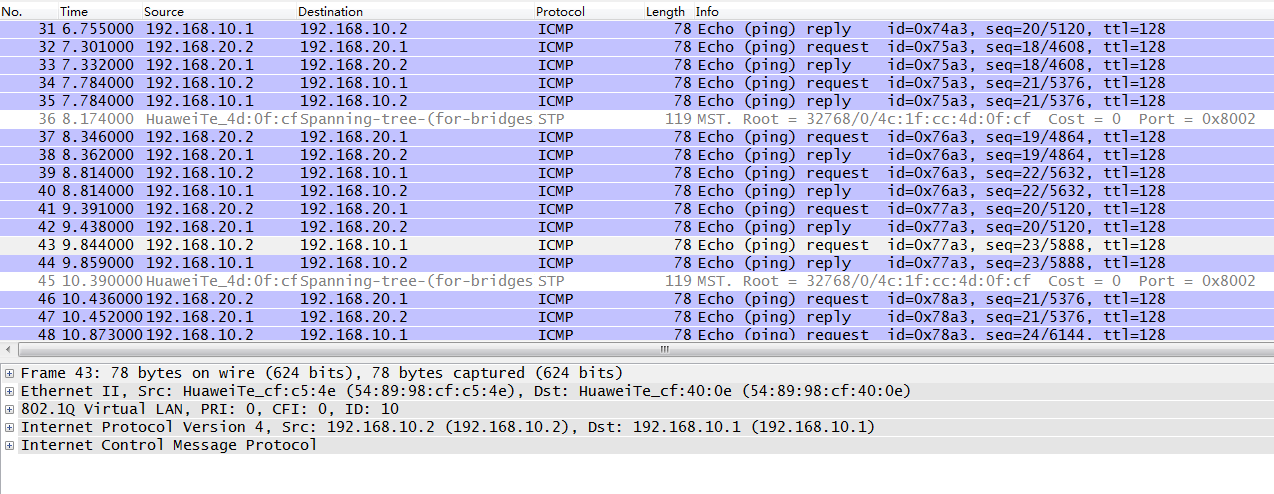


图4-6

可以观察到，目前VLAN 10和VLAN 20的数据包都从S2的接口E 0/0/1转发。

在S3的E 0/0/2接口上抓包观察。

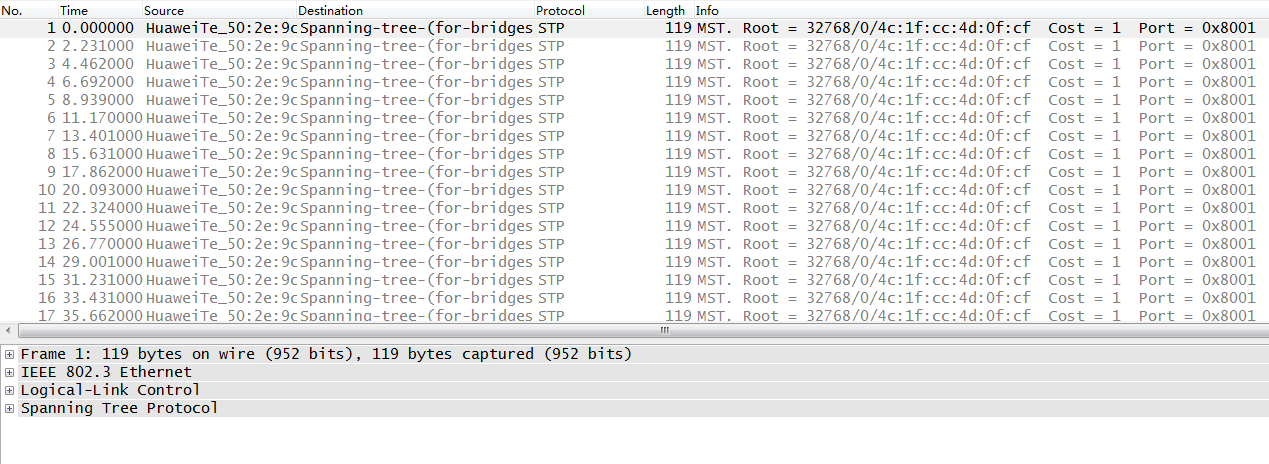


图4-7

可以观察到，在S3的E 0/0/2接口上，没有任何数据包转发，只接收到上行接口周期发送的BPDU。

此时S2与S3间的链路完全处于闲置状态，造成了资源的浪费，也导致了S1与S3间链路上数据转发任务繁重，易引起拥塞丢包。为了能够有效的利用链路资源，可以通过配置MSTP的多实例来实现。

关闭PC上的ping测试。

## 配置MSTP多实例

MSTP网络由一个或者多个MST域组成，每个MST域中可以包含一个或多个MSTI，即MST实例。

MST域中含有一张VLAN映射表，描述了VLAN与MSTI之间的映射关系，默认情况下所有VLAN都映射到MSTI 0中。MSTI之间彼此独立。MSTI可以与一个或者多个VLAN对应，但一个VLAN只能与一个MSTI对应。

在S1上配置MSTP的多实例。使用命令**stp region-configuration**进入MST域视图。

[S1]stp region-configuration

[S1-mst-region]

使用命令**region-name**配置MST域名为huawei。

[S1-mst-region]region-name huawei

使用命令**revision-level**配置MSTP的修订级别为1。

[S1-mst-region]revision-level 1

使用命令**instance**指定VLAN 10映射到MSTI 1，指定VLAN 20映射到MSTI 2。

[S1-mst-region]instance 1 vlan 10

[S1-mst-region]instance 2 vlan 20

使用命令**active region-configuration**激活MST域配置。

[S1-mst-region]active region-configuration

Info: This operation may take a few seconds. Please wait for a moment...done.

同理在S2，S3上做同样配置，但是注意，在同一MST域中，必须具有相同域名，修订级别，以及VLAN到MSTI的映射关系。

[S2]stp region-configuration

[S2-mst-region]region-name huawei

[S2-mst-region]revision-level 1

[S2-mst-region]instance 1 vlan 10

[S2-mst-region]instance 2 vlan 20

[S2-mst-region]active region-configuration

[S3]stp region-configuration

[S3-mst-region]region-name huawei

[S3-mst-region]revision-level 1

[S3-mst-region]instance 1 vlan 10

[S3-mst-region]instance 2 vlan 20

[S3-mst-region]active region-configuration

配置完成后，在S1，S2，S3上使用命令**display stp region-configuration**查看交换机上当前生效的MST域配置信息。

[S1]display stp region-configuration

Oper configuration

Format selector :0

Region name :huawei

Revision level :1

Instance VLANs Mapped

0 1 to 9, 11 to 19, 21 to 4094

1 10

2 20

[S2]display stp region-configuration

Oper configuration

Format selector :0

Region name :huawei

Revision level :1

Instance VLANs Mapped

0 1 to 9, 11 to 19, 21 to 4094

1 10

2 20

[S3]display stp region-configuration

Oper configuration

Format selector :0

Region name :huawei

Revision level :1

Instance VLANs Mapped

0 1 to 9, 11 to 19, 21 to 4094

1 10

2 20

可以观察到，所有交换机上的MST域名都为huawei，修订版本号都为1，及VLAN与实例间的映射关系相同，其中除VLAN 10与20之外，其余VLAN都属于实例0中。

MSTP多实例配置完成后，在HR部门的PC-2上持续发送ping包至PC-1，在IT部门的PC-4上持续发送ping包至PC-3。

PC>ping 192.168.10.1 -t

Ping 192.168.10.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.10.1: bytes=32 seq=1 ttl=128 time=31 ms

From 192.168.10.1: bytes=32 seq=2 ttl=128 time=47 ms

……

PC>ping 192.168.20.1 -t

Ping 192.168.20.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.1: bytes=32 seq=1 ttl=128 time=47 ms

From 192.168.20.1: bytes=32 seq=2 ttl=128 time=62 ms

……

同时，在S3的E 0/0/1接口上抓包观察。

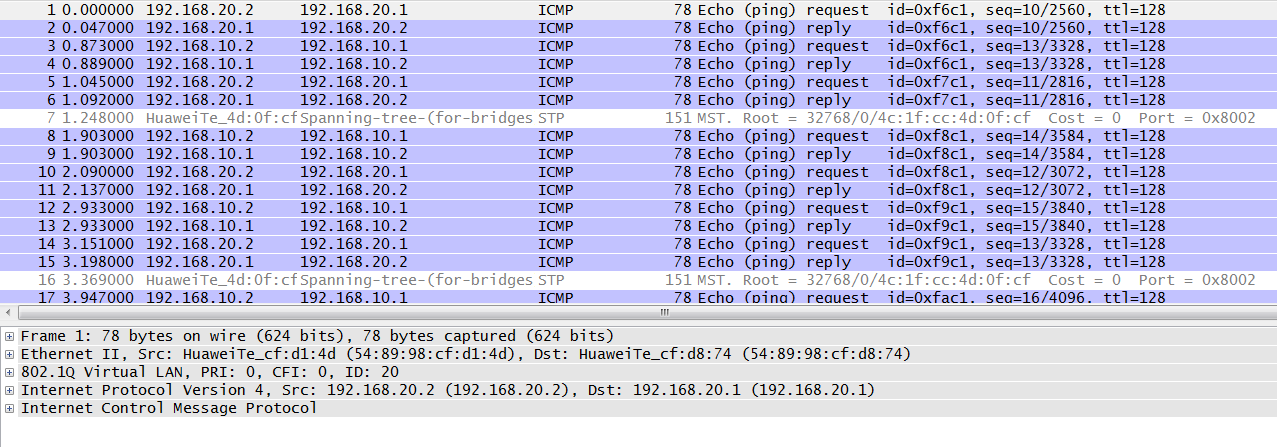


图4-8

可以观察到，目前VLAN 10和VLAN 20的数据包仍然从E 0/0/1转发。

在S3的E 0/0/2接口上抓包观察。

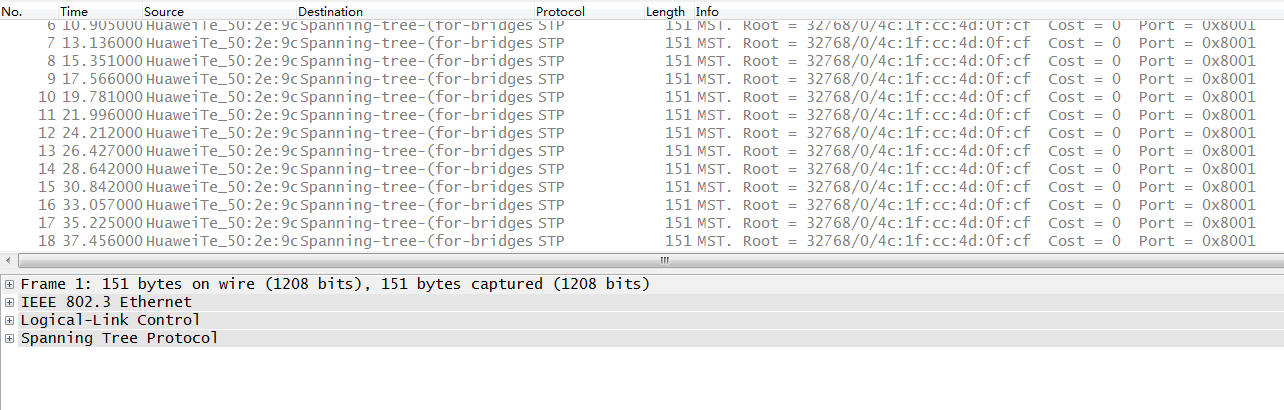


图4-9

可以观察到，在E 0/0/2接口上，仍然没有任何数据包转发，只有接收到的上行接口周期发送的BPDU。

关闭PC上的ping测试。

现在已经配置了MSTP多实例，但由于每个MSTP实例都是单独的一颗生成树，独立进行选举，所以在默认不变动任何生成树参数的情况下，其实每棵生成树的选举结果是一致的。

在S1，S2，S3上使用命令**display stp instance 0 brief**查看默认实例0中的生成树状态和统计的摘要信息。

<S1>display stp instance 0 brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 DESI FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

<S2>display stp instance 0 brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 DESI FORWARDING NONE

0 Ethernet0/0/2 ROOT FORWARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

<S3>display stp instance 0 brief

MSTID Port Role STP State Protection

0 Ethernet0/0/1 ROOT FORWARDING NONE

0 Ethernet0/0/2 ALTE DISCARDING NONE

0 Ethernet0/0/3 DESI FORWARDING NONE

0 Ethernet0/0/4 DESI FORWARDING NONE

在S1，S2，S3上使用命令**display stp instance 1 brief**查看实例1中的生成树状态和统计的摘要信息。

<S1>display stp instance 1 brief

MSTID Port Role STP State Protection

1 Ethernet0/0/1 DESI FORWARDING NONE

1 Ethernet0/0/2 DESI FORWARDING NONE

1 Ethernet0/0/3 DESI FORWARDING NONE

<S2>display stp instance 1 brief

MSTID Port Role STP State Protection

1 Ethernet0/0/1 DESI FORWARDING NONE

1 Ethernet0/0/2 ROOT FORWARDING NONE

<S3>display stp instance 1 brief

MSTID Port Role STP State Protection

1 Ethernet0/0/1 ROOT FORWARDING NONE

1 Ethernet0/0/2 ALTE DISCARDING NONE

1 Ethernet0/0/3 DESI FORWARDING NONE

在S1，S2，S3上使用命令**display stp instance 2 brief**查看实例2中的生成树状态和统计的摘要信息。

<S1>display stp instance 2 brief

MSTID Port Role STP State Protection

2 Ethernet0/0/1 DESI FORWARDING NONE

2 Ethernet0/0/2 DESI FORWARDING NONE

<S2>display stp instance 2 brief

MSTID Port Role STP State Protection

2 Ethernet0/0/1 DESI FORWARDING NONE

2 Ethernet0/0/2 ROOT FORWARDING NONE

2 Ethernet0/0/3 DESI FORWARDING NONE

<S3>display stp instance 2 brief

MSTID Port Role STP State Protection

2 Ethernet0/0/1 ROOT FORWARDING NONE

2 Ethernet0/0/2 ALTE DISCARDING NONE

2 Ethernet0/0/4 DESI DISCARDING NONE

可以观察到，在三个实例中，选举结果是一致的，都是S3的E 0/0/2接口处于Discarding状态。

现在要实现S2与S3间的链路被利用，可以在实例1中，保持目前生成树选举结果不变，即使得VLAN 10中的HR部门内的流量通过S1与S3间的链路转发。在实例2中，配置使得S2成为根交换机，阻塞S1与S3间的链路，即使得VLAN 20中的IT部门的流量通过S2与S3间的链路转发。

在S2上使用命令**stp instance priority**配置其在实例2中的优先级为0，成为实例2中的根交换机。

[S2]stp instance 2 priority 0

配置完成后，在S1，S2，S3上使用命令**display stp instance 2 brief**查看实例2中的生成树状态和统计的摘要信息。

<S1>display stp instance 2 brief

MSTID Port Role STP State Protection

2 Ethernet0/0/1 ROOT FORWARDING NONE

2 Ethernet0/0/2 DESI FORWARDING NONE

<S2>display stp instance 2 brief

MSTID Port Role STP State Protection

2 Ethernet0/0/1 DESI FORWARDING NONE

2 Ethernet0/0/2 DESI FORWARDING NONE

2 Ethernet0/0/3 DESI FORWARDING NONE

<S3>display stp instance 2 brief

MSTID Port Role STP State Protection

2 Ethernet0/0/1 ALTE DISCARDING NONE

2 Ethernet0/0/2 ROOT FORWARDING NONE

2 Ethernet0/0/4 DESI FORWARDING NONE

可以观察到，此时S2成为了实例2中的根交换机，所有端口都为指定端口，而S3的E0/0/1接口为替代端口，即S1与S3间的链路现已阻塞。

在HR部门的PC-2上持续发送ping包至PC-1，在IT部门的PC-4上持续发送ping包至PC-3。

PC>ping 192.168.10.1 -t

Ping 192.168.10.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.10.1: bytes=32 seq=1 ttl=128 time=31 ms

From 192.168.10.1: bytes=32 seq=2 ttl=128 time=47 ms

……

PC>ping 192.168.20.1 -t

Ping 192.168.20.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.1: bytes=32 seq=1 ttl=128 time=47 ms

From 192.168.20.1: bytes=32 seq=2 ttl=128 time=62 ms

……

同时，在S3的E 0/0/1接口上抓包观察。

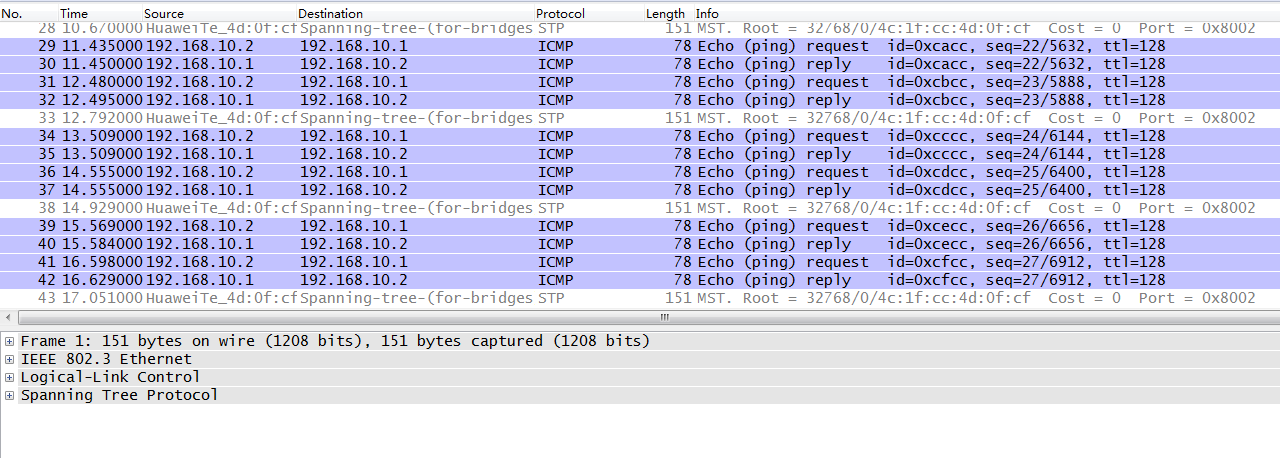


图4-10

可以观察到，目前VLAN 10的流量都从S3的E0/0/1接口转发。

在S3的E 0/0/2接口上抓包观察。

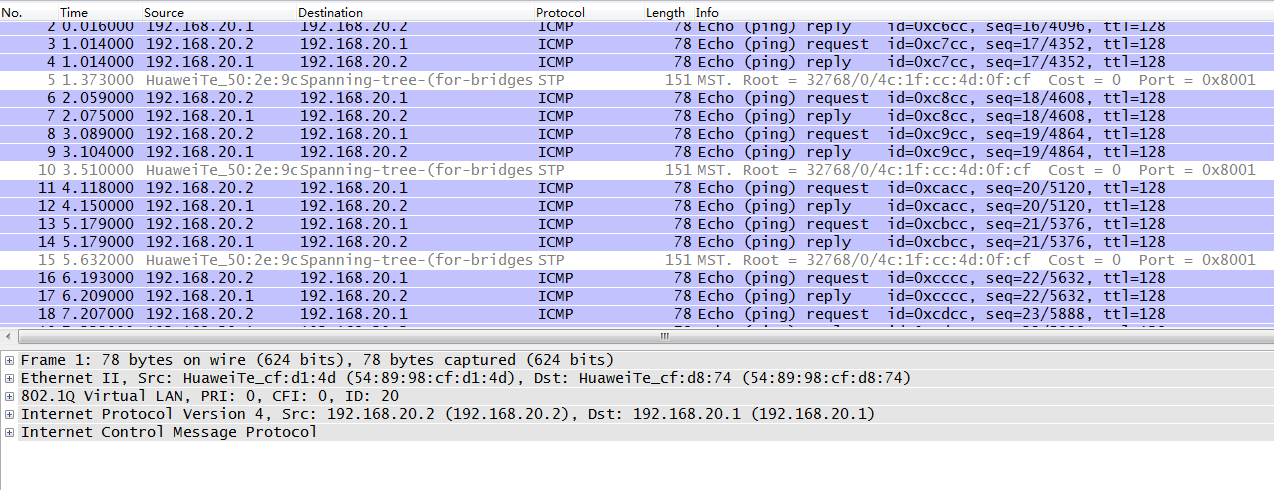


图4-11

可以观察到，目前VLAN 20的流量都从E0/0/2接口转发。

至此，完成了MSTP的多实例的配置，并达到了流量分担的目的，有效的利用了网络资源，也同时使得S3的两条上行可以互相备份。

# 思考

当MSTP和RSTP混合使用的时候，如何选举根桥？

1. 其他交换技术

5.1 GVRP基础配置

# 原理概述

GVRP（GARP VLAN Registration Protocol），中文名为GARP VLAN注册协议，是GARP（Generic Attribute Registration Protocol）通用属性注册协议的一种应用，用于注册和注销VLAN属性。使得交换机之间能够相互交换VLAN配置信息，动态创建和管理VLAN。用户只需要对少数交换机进行VLAN 配置即可，动态地传播VLAN信息。

手工配置的VLAN称为静态VLAN，通过GVRP协议创建的VLAN称为动态VLAN。GVRP有三种注册模式，不同的模式对静态VLAN和动态VLAN的处理方式也不同：

Normal模式：允许该接口动态注册、注销VLAN，传播动态VLAN以及静态VLAN信息；

Fixed模式：禁止该接口动态注册、注销VLAN，只传播静态VLAN信息。即被设置成为该模式下的Trunk接口，即使允许所有VLAN通过，实际通过的VLAN也只能是手动配置的那部分；

Forbidden模式：禁止该接口动态注册、注销VLAN，不传播任何除VLAN 1以外的任何VLAN信息。即被设置成为该模式下的Trunk接口，即使允许所有VLAN通过，实际通过的VLAN也只能是VLAN 1。

# 实验目的

* + 理解GVRP的应用场景
  + 掌握GVRP的配置
  + 理解GVRP不同注册模式的区别
  + 掌握GVRP配置不同注册模式的方法

# 实验内容

本实验模拟企业网络场景，S1和S4是接入层交换机，分别连接到汇聚层交换机S2和S3，公司不同部门员工通过接入层交换机连接到网络。现在需要在交换机上划分VLAN隔离不同部门，但考虑到部门较多，且随着发展，网络情况可能会越来越复杂，采用手工配置VLAN的方式工作量会非常大，而且容易导致配置错误。此时可以通过GVRP的VLAN自动注册功能完成VLAN的配置。

# 实验拓扑

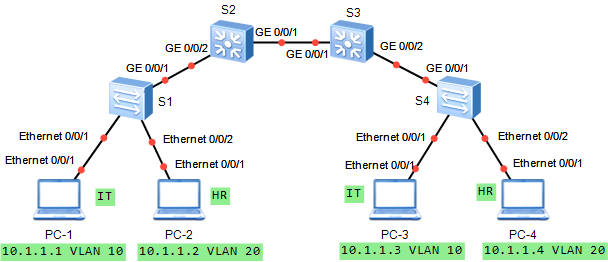


图5-1 GVRP基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |
| PC-3 | Ethernet 0/0/1 | 10.1.1.3 | 255.255.255.0 | N/A |
| PC-4 | Ethernet 0/0/1 | 10.1.1.4 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连

通性，在没有完成划分VLAN之前各PC都属于默认的VLAN 1，之间都能互通。测试PC-1与PC-2间的连通性。

PC>ping 10.1.1.2

Ping 10.1.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.2: bytes=32 seq=1 ttl=128 time=16 ms

From 10.1.1.2: bytes=32 seq=2 ttl=128 time=31 ms

From 10.1.1.2: bytes=32 seq=3 ttl=128 time<1 ms

From 10.1.1.2: bytes=32 seq=4 ttl=128 time=15 ms

From 10.1.1.2: bytes=32 seq=5 ttl=128 time=15 ms

--- 10.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/15/31 ms

其余主机间的连通性测试省略。

## 配置GVRP单向注册

在公司的二层网络中，有IT和HR两个不同的部门，需要将它们划分到不同的VLAN中去。如果按照常规配置方法，要手工在所有交换机上都创建相应的VLAN。后续如果有新的部门需要新增VLAN，或者二层网络整改，都要随之修改VLAN配置，配置量非常大且易出错，现网络管理员采用GVRP来完成VLAN配置。

将四台交换机之间所互连的接口（连接PC的接口除外）的接口类型都配置为Trunk，并允许所有VLAN通过。

[S1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]port link-type trunk

[S1-GigabitEthernet0/0/1]port trunk allow-pass vlan all

[S2]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]port link-type trunk

[S2-GigabitEthernet0/0/1]port trunk allow-pass vlan all

[S2-GigabitEthernet0/0/1]quit

[S2] interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]port link-type trunk

[S2-GigabitEthernet0/0/2]port trunk allow-pass vlan all

[S3]interface GigabitEthernet 0/0/1

[S3-GigabitEthernet0/0/1]port link-type trunk

[S3-GigabitEthernet0/0/1]port trunk allow-pass vlan all

[S3-GigabitEthernet0/0/1]quit

[S3] interface GigabitEthernet 0/0/2

[S3-GigabitEthernet0/0/2]port link-type trunk

[S3-GigabitEthernet0/0/2]port trunk allow-pass vlan all

[S4]interface GigabitEthernet 0/0/1

[S4-GigabitEthernet0/0/1]port link-type trunk

[S4-GigabitEthernet0/0/1]port trunk allow-pass vlan all

在S1上创建VLAN 10和VLAN 20，并把连接PC的接口类型配置为Access，划入到相应的VLAN中。

[S1]vlan 10

[S1-Vlan10]vlan 20

[S1-Vlan20]interface Ethernet0/0/1

[S1-Ethernet0/0/1]port link-type access

[S1-Ethernet0/0/1]port default vlan 10

[S1-Ethernet0/0/1]interface Ethernet0/0/2

[S1-Ethernet0/0/2]port link-type access

[S1-Ethernet0/0/2]port default vlan 20

在所有交换机上都开启GVRP功能，并在所有交换机两两互连的接口下也开启GVRP功能，GVRP注册模式缺省为Normal模式。

[S1]gvrp

[S1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]gvrp

[S2]gvrp

[S2]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]gvrp

[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]gvrp

[S3]gvrp

[S3]interface GigabitEthernet 0/0/1

[S3-GigabitEthernet0/0/1]gvrp

[S3-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S3-GigabitEthernet0/0/2]gvrp

[S4]gvrp

[S4]interface GigabitEthernet 0/0/1

[S4-GigabitEthernet0/0/1]gvrp

配置完成后，在S2，S3，S4上使用命令**display vlan**查看所有VLAN的相关信息。

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 dynamic TG:GE0/0/2(U)

20 dynamic TG:GE0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

[S3]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 dynamic TG:GE0/0/1(U)

20 dynamic TG:GE0/0/1(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

[S4]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:Eth0/0/1(U) Eth0/0/2(U) Eth0/0/3(D) Eth0/0/4(D)

Eth0/0/5(D) Eth0/0/6(D) Eth0/0/7(D) Eth0/0/8(D)

Eth0/0/9(D) Eth0/0/10(D) Eth0/0/11(D) Eth0/0/12(D)

Eth0/0/13(D) Eth0/0/14(D) Eth0/0/15(D) Eth0/0/16(D)

Eth0/0/17(D) Eth0/0/18(D) Eth0/0/19(D) Eth0/0/20(D)

Eth0/0/21(D) Eth0/0/22(D) GE0/0/1(U) GE0/0/2(D)

10 dynamic TG:GE0/0/1(U)

20 dynamic TG:GE0/0/1(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

可以观察到S2，S3，S4都动态获得了VLAN 10和VLAN 20。但是在S2上只有GE 0/0/2加入了这两个VLAN，同样在S3上只有GE 0/0/1加入这两个VLAN，在S4上只有GE 0/0/1加入了这两个VLAN。这是因为此时在S2，S3，S4上只有左侧的端口收到GVRP的注册消息，此时只完成了单向注册。

由于PC-1与PC-3同属于IT部门，即VLAN 10内，验证它们之间的连通性。

PC>ping 10.1.1.3

Ping 10.1.1.3: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

From 10.1.1.1: Destination host unreachable

……

发现无法通信，再次验证了此时只完成了单向注册。

## 配置GVRP双向注册

现需要在S4上也手工创建VLAN 10和VLAN 20，把连接PC的接口的模式配置为Access，划入到相应VLAN中。

<S4>system-view

[S4]vlan 10

[S4-Vlan10]vlan 20

[S4-Vlan20]interface Ethernet0/0/1

[S4-Ethernet0/0/1]port link-type access

[S4-Ethernet0/0/1]port default vlan 10

[S4-Ethernet0/0/1]interface Ethernet0/0/2

[S4-Ethernet0/0/2]port link-type access

[S4-Ethernet0/0/2]port default vlan 20

配置完成后，在S2，S3上再次使用**display vlan**查看。

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 dynamic TG:GE0/0/1(U) GE0/0/2(U)

20 dynamic TG:GE0/0/1(U) GE0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

[S3]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 dynamic TG:GE0/0/1(U) GE0/0/2(U)

20 dynamic TG:GE0/0/1(U) GE0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

可以观察到此时两台汇聚交换机的右侧接口也加入了VLAN 10和VLAN 20。因为从右侧收到了S4的GVRP注册消息，此时完成了双向注册。

在PC-1上验证与PC-3之间的连通性。

PC>ping 10.1.1.3

Ping 10.1.1.3: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.3: bytes=32 seq=1 ttl=128 time=78 ms

From 10.1.1.3: bytes=32 seq=2 ttl=128 time=109 ms

From 10.1.1.3: bytes=32 seq=3 ttl=128 time=63 ms

From 10.1.1.3: bytes=32 seq=4 ttl=128 time=62 ms

From 10.1.1.3: bytes=32 seq=5 ttl=128 time=31 ms

--- 10.1.1.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/68/109 ms

此时可以正常通信，PC-2与PC-4的连通性测试这里省略。

如果现在公司网络整改，需要添加新的汇聚交换机，或者替换新款设备，或者增删VLAN配置，都可以通过GVRP动态实现自动配置，不在需要手工配置。

## 配置GVRP的Fixed模式

现在S3的GE 0/0/1接口下将GVRP的注册模式修改为Fixed模式。

[S3]interface GigabitEthernet 0/0/1

[S3-GigabitEthernet0/0/1]gvrp registration fixed

在S3上使用**display vlan**命令查看。

[S3]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 dynamic TG:GE0/0/2(U)

20 dynamic TG:GE0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

发现在GE 0/0/1接口已经无法动态学习到VLAN信息，这是由于Fixed模式下不允许动态VLAN在接口上注册，相应同部门内跨交换机的两台PC就无法通信。

这时的解决办法有两种，一种是在S3上手工创建VLAN 10和VLAN 20，另一种是恢复GE 0/0/1接口下GVRP注册模式为Normal模式，做相应配置即可，具体配置和现象省略。

## 配置GVRP的Forbidden模式

在S2的GE 0/0/1接口下将GVRP的注册模式修改为Forbidden模式。

[S2]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]gvrp registration forbidden

在S2上使用display vlan命令查看。

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 common TG:GE0/0/2(U)

20 common TG:GE0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

可以观察到此时VLAN 10，VLAN 20中都没有GE 0/0/1接口加入。

在S2上手工创建VLAN 10和VLAN 20，并使用**display vlan**命令查看。

[S2]vlan batch 10 20

[S2]display vlan

The total number of vlans is : 3

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

U: Up; D: Down; TG: Tagged; UT: Untagged;

MP: Vlan-mapping; ST: Vlan-stacking;

#: ProtocolTransparent-vlan; \*: Management-vlan;

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

VID Type Ports

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

1 common UT:GE0/0/1(U) GE0/0/2(U) GE0/0/3(D) GE0/0/4(D)

GE0/0/5(D) GE0/0/6(D) GE0/0/7(D) GE0/0/8(D)

GE0/0/9(D) GE0/0/10(D) GE0/0/11(D) GE0/0/12(D)

GE0/0/13(D) GE0/0/14(D) GE0/0/15(D) GE0/0/16(D)

GE0/0/17(D) GE0/0/18(D) GE0/0/19(D) GE0/0/20(D)

GE0/0/21(D) GE0/0/22(D) GE0/0/23(D) GE0/0/24(D)

10 common TG:GE0/0/2(U)

20 common TG:GE0/0/2(U)

VID Status Property MAC-LRN Statistics Description

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

1 enable default enable disable VLAN 0001

10 enable default enable disable VLAN 0010

20 enable default enable disable VLAN 0020

结果还是一样，接口GE 0/0/1仍然没有加入到VLAN 10或20中。这是因为GE 0/0/1接口下注册模式配置成了Forbidden模式后，将只允许VLAN 1通过。

# 思考

GVRP能够应用在Hybrid类型的接口上吗？

5.2 Smart-Link与Monitor-Link

# 原理概述

在以太网络中，为了提高网络的可靠性，通常采用双归属上行方式进行组网，即一台交换机同时连接到两台上行交换机，但是在二层网络中可能会带来环路问题。为了解决环路问题，可以采用STP技术，但STP的收敛时间较长，当主用链路故障时，将流量切换到备用链路，只能是达到秒级的收敛速度，不适用于对收敛时间有很高要求的组网环境。

基于上述原因，华为针对双归属上行组网提出了Smart Link解决方案。网络中两条上行链路正常情况下，只有一条处于连通状态，而另一条处于阻塞状态，从而防止了环路引起的广播风暴。当主用链路发生故障后，流量会在毫秒级的时间内迅速切换到备用链路上，保证了数据的正常转发。缺省情况下，当原主用链路故障恢复时，将维持在阻塞状态，不进行抢占，从而保持网络稳定，可以手工配置回切功能使流量切换回原主用链路。Smart Link配置简单，便于操作和维护。

Smart Link虽然能够保证设备在本设备上行链路发生故障后能够快速进行倒换，但对于跨设备的链路故障不能提供有效保护，为此可以采用Monitor Link。Monitor Link用于扩展Smart Link的链路备份的范围，通过监控上游设备的上行链路，达到上行链路故障迅速传达给下游设备，从而触发Smart Link的主备链路切换，防止长时间因上行链路故障而出现网络中断，使Smart Link备份作用更为完善。

# 实验目的

* + 理解Smart Link的应用场景
  + 掌握Smart Link组的基本配置
  + 掌握Smart Link回切功能的配置
  + 掌握Monitor Link的基本配置

# 实验内容

本实验模拟公司网络场景，交换机S4作为公司出口设备连接外网，交换机S1是接入层交换机，负责员工终端接入，接入交换机通过两台交换机S2和S3双上行连接到S4。针对此双上行组网，为了实现主备链路冗余备份及故障后的快速迁移，部署使用Smart Link技术，且为了进一步扩展Smart Link的备份范围，使用Monitor Link联动方式监控上游设备的上行链路来完善Smart Link。

# 实验拓扑

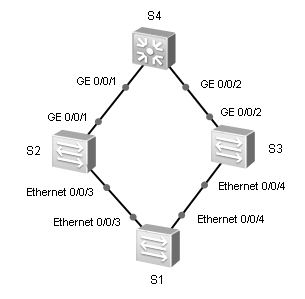


图5-2 Smart-Link与Monitor-Link拓扑图

# 实验步骤

## 配置Smart Link

公司接入层交换机S1通过S2和S3双上行链路连接到出口交换机S4，为了实现主备链路冗余备份及快速迁移，需要在S1上配置Smart Link。

在S1上进入到视图模式下创建Smart Link 组1，并开启Smart Link组功能。

<S1>system-view

[S1]smart-link group 1

[S1-smlk-group1]smart-link enable

配置Smart Link时，需要在相关运行Smart Link的接口下关闭生成树协议。由于华为交换机默认开启了生成树协议，因此需要关闭S1交换机上E 0/0/3和E 0/0/4接口下的生成树协议。

[S1]interface Ethernet 0/0/3

[S1-Ethernet0/0/3]stp disable

[S1-Ethernet0/0/3]interface Ethernet 0/0/4

[S1-Ethernet0/0/4]stp disable

注意，如果相应接口下生成树协议未关闭，在配置Smart Link组功能时会报错，将会出现下面的提示信息。

Error: Adding a port failed. The port is already enabled with STP

进入到Smart Link组1下，配置E 0/0/3为主接口，E 0/0/4为备份接口。

[S1]smart-link group 1

[S1-smlk-group1]port Ethernet 0/0/3 master

[S1-smlk-group1]port Ethernet 0/0/4 slave

配置完成后，使用命令**display smart-link group 1**查看主备状态。

[S1]display smart-link group 1

Smart Link group 1 information :

Smart Link group was enabled

There is no Load-Balance

There is no protected-vlan reference-instance

DeviceID: 4c1f-cc83-109c

Member Role State Flush Count Last-Flush-Time

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

Ethernet0/0/3 Master Active 0 0000/00/00 00:00:00 UTC+00:00

Ethernet0/0/4 Slave Inactive 0 0000/00/00 00:00:00 UTC+00:0

可以观察到，S1交换机的E 0/0/3为主接口，且状态为Active，E 0/0/4为备份接口，状态为Inactive。

## 配置回切功能

当S1上主接口E 0/0/3出现故障down掉时，备份接口会立刻切换为Active状态。并且缺省情况下，当原主接口恢复时，主接口不会自动回切到Active状态，需要手工配置回切功能。

将S2交换机E 0/0/3接口关闭，模拟故障发生，观察Smart Link组1的主备状态。

[S2]interface Ethernet 0/0/3

[S2-Ethernet0/0/3]shutdown

[S1]display smart-link group 1

Smart Link group 1 information :

Smart Link group was enabled

There is no Load-Balance

There is no protected-vlan reference-instance

DeviceID: 4c1f-cc83-109c

Member Role State Flush Count Last-Flush-Time

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

Ethernet0/0/3 Master Inactive 0 0000/00/00 00:00:00 UTC+00:00

Ethernet0/0/4 Slave Active 0 0000/00/00 00:00:00 UTC+00:00

可以观察到，S1交换机E 0/0/3仍然为主接口，但是状态处于Inactive，而Ethernet0/0/4状态此时为Active。

重新开启S2的E 0/0/3接口，再次观察Smart Link组1的主备状态。

[S2]interface Ethernet 0/0/3

[S2-Ethernet0/0/3]undo shutdown

[S1-Ethernet0/0/3]display smart-link group 1

Smart Link group 1 information :

Smart Link group was enabled

There is no Load-Balance

There is no protected-vlan reference-instance

DeviceID: 4c1f-cc83-109c

Member Role State Flush Count Last-Flush-Time

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

Ethernet0/0/3 Master Inactive 0 0000/00/00 00:00:00 UTC+00:00

Ethernet0/0/4 Slave Active 0 0000/00/00 00:00:00 UTC+00:00

可以观察到，接口的状态没有发生变化，E 0/0/3接口仍然处于Inactive状态，并没有抢占原来的Active状态。即当主链路出现故障后，会自动切换到备份链路；而当原主链路故障恢复后，为了保持网络稳定，它将维持在阻塞状态，不进行抢占。如果需要原主链路恢复为Active状态，可以通过配置Smart Link组回切功能，在回切定时器超时后会自动切换到主链路。

在S1上使用命令**restore enable**开启回切功能，并配置回切时间设置为30秒（默认为60秒）。

[S1]smart-link group 1

[S1-smlk-group1]restore enable

[S1-smlk-group1]timer wtr 30

等待30秒钟后S1上会弹出如下信息，即已经产生了状态的切换。

Jun 26 2013 18:53:09-08:00 S1 %%01SMLK/4/SMLK\_STATUS\_LOG(l)[5]:The state of Smart link group 1 changed to MASTER.

再次查看Smart Link组1的主备状态。

[S1]display smart-link group 1

Smart Link group 1 information :

Smart Link group was enabled

Wtr-time is: 30 sec.

There is no Load-Balance

There is no protected-vlan reference-instance

DeviceID: 4c1f-cc83-109c

Member Role State Flush Count Last-Flush-Time

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

Ethernet0/0/3 Master Active 0 0000/00/00 00:00:00 UTC+00:00

Ethernet0/0/4 Slave Inactive 0 0000/00/00 00:00:00 UTC+00:00

可以观察到，S1的E 0/0/3接口状态又重新恢复到Active状态，而E 0/0/4接口回到了Inactive状态。

## 配置Monitor Link

Monitor Link是对Smart Link进行补充而引入的接口联动方案，用于扩展Smart Link的链路备份的范围。通过监控上游设备的上行链路，而对下行链路进行同步设置，达到上游设备的上行链路故障迅速传达给下行设备，从而触发下游设备的Smart Link的主备链路切换，防止长时间因上行链路故障而出现网络故障。

正常情况下，S1与S2之间的链路为主链路，但是当S2的上行接口GE 0/0/1故障时， Smart Link无法感知故障，不会发生切换，导致网络中断。为了解决这一问题，需要在S2上配置Monitor Link监控上行接口，当GE 0/0/1故障时，使S1的Smart Link组切换切换。

为了模拟该场景，现将S2的GE 0/0/1接口关闭，并查看Smart Link组1的主备状态。

[S2]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]shutdown

[S1]display smart-link group 1

Smart Link group 1 information :

Smart Link group was enabled

Wtr-time is: 30 sec.

There is no Load-Balance

There is no protected-vlan reference-instance

DeviceID: 4c1f-cc83-109c

Member Role State Flush Count Last-Flush-Time

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

Ethernet0/0/3 Master Active 0 0000/00/00 00:00:00 UTC+00:00

Ethernet0/0/4 Slave Inactive 0 0000/00/00 00:00:00 UTC+00:00

可以观察到，当S2的上行GE 0/0/1接口出现故障以后，连接到下行链路的S1交换机无法感知到该故障，导致S1交换机的Smart Link无法进行切换，这样会导致连接到S1交换机仍然选择E 0/0/3接口转发数据，无法正常通信。

在S2上启用Monitor Link组1，配置上行接口为GE 0/0/1，下行接口为E 0/0/3。

[S2]monitor-link group 1

[S2-mtlk-group1]port GigabitEthernet 0/0/1 uplink

[S2-mtlk-group1]port Ethernet 0/0/3 downlink

配置完成后，再次查看S1的Smart Link组1的主备状态。

[S1]display smart-link group 1

Smart Link group 1 information :

Smart Link group was enabled

Wtr-time is: 30 sec.

There is no Load-Balance

There is no protected-vlan reference-instance

DeviceID: 4c1f-cc83-109c

Member Role State Flush Count Last-Flush-Time

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

Ethernet0/0/3 Master Inactive 0 0000/00/00 00:00:00 UTC+00:00

Ethernet0/0/4 Slave Active 0 0000/00/00 00:00:00 UTC+00:00

观察发现E 0/0/3接口状态已经变为Inactive，E 0/0/4接口状态成为了Active，流量已经被切换到E 0/0/4接口，保证了用户流量的正常转发。

修改Monitor Link组的回切时间为10秒（默认为3秒）。当S2的上行接口GE 0/0/1重新恢复以后，下行链路Smart Link组将在时间到期后，重新回切到主链路。

[S2-mtlk-group1]timer recover-time 10

重新开启S2的GE 0/0/1接口。

[S2]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]undo shutdown

等待40秒钟左右（加上步骤二中配置的Smart Link回切时间），查看S1的Smart Link组1的主备状态。

[S1]display smart-link group 1

Smart Link group 1 information :

Smart Link group was enabled

Wtr-time is: 30 sec.

There is no Load-Balance

There is no protected-vlan reference-instance

DeviceID: 4c1f-cc83-109c

Member Role State Flush Count Last-Flush-Time

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

Ethernet0/0/3 Master Active 0 0000/00/00 00:00:00 UTC+00:00

Ethernet0/0/4 Slave Inactive 0 0000/00/00 00:00:00 UTC+00:00

可以观察到，此时S1的E 0/0/3接口重新恢复到了Active状态。

# 思考

Smart Link和Monitor Link的联合使用可以确保链路出现故障后及时的切换，如果所有链路都正常，是否所有数据都只能通过主链路转发？

5.3 配置Eth-Trunk链路聚合

# 原理概述

在没有使用Eth-Trunk前，百兆以太网的双绞线在两个互连的网络设备间的带宽仅为100Mbit/s。若想达到更高的数据传输速率，则需要更换传输媒介，使用千兆光纤或升级成为千兆以太网。这样的解决方案成本较高。如果采用Eth-Trunk技术把多个接口捆绑在一起，则可以以较低的成本满足提高接口带宽的需求。例如，把3个100Mbit/s的全双工接口捆绑在一起，就可以达到300Mbit/s的最大带宽。

Eth-Trunk是一种捆绑技术，它将多个物理接口捆绑成一个逻辑接口，这个逻辑接口就称为Eth-Trunk接口，捆绑在一起的每个物理接口称为成员接口。Eth-Trunk只能由以太网链路构成。Trunk的优势在于：

负载分担，在一个Eth-Trunk接口内，可以实现流量负载分担；

提高可靠性，当某个成员接口连接的物理链路出现故障时，流量会切换到其他可用的链路上，从而提高整个Trunk链路的可靠性；

增加带宽，Trunk接口的总带宽是各成员接口带宽之和。

Eth-Trunk在逻辑上把多条物理链路捆绑等同于一条逻辑链路，对上层数据透明传输。所有Eth-Trunk中物理接口的参数必须一致，Eth-Trunk链路两端要求一致的物理参数有： Eth-Trunk链路两端相连的物理接口类型、物理接口数量、物理接口的速率、物理接口的双工方式以及物理接口的流控方式。

# 实验目的

* + 理解使用Eth-Trunk的应用场景
  + 掌握配置Eth-Trunk链路聚合的方法（手工负载分担模式）
  + 掌握配置Eth-Trunk链路聚合的方法（静态LACP模式）

# 实验内容

本实验模拟企业网络环境，S1和S2为企业核心交换机，PC-1属于A部门终端设备，PC-2属于B部门终端设备。根据企业规划，S1和S2之间线路原由1条光纤线路相连，但是由于带宽和冗余角度考虑需要对其进行升级，需要使用Eth-Trunk实现此需求。

# 实验拓扑

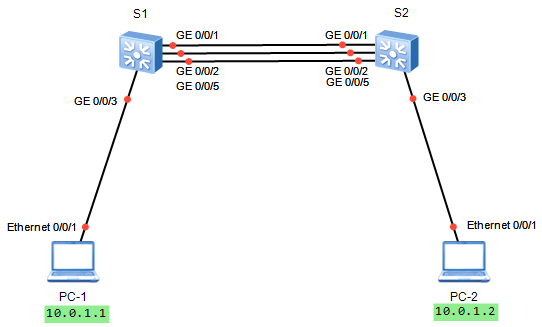


图5-3 配置Eth-Trunk链路聚合拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 10.0.1.2 | 255.255.255.0 | N/A |

# MAC地址表

|  |  |
| --- | --- |
| 设备 | 全局MAC地址 |
| S1(S5700) | 4c1f-cc55-b90f |
| S2(S5700) | 4c1f-cc71-68d4 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各PC之间的连通性**。**

PC>ping 10.0.1.2

Ping 10.0.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.0.1.2: bytes=32 seq=1 ttl=128 time=16 ms

From 10.0.1.2: bytes=32 seq=2 ttl=128 time=16 ms

From 10.0.1.2: bytes=32 seq=3 ttl=128 time<1 ms

From 10.0.1.2: bytes=32 seq=4 ttl=128 time=16 ms

From 10.0.1.2: bytes=32 seq=5 ttl=128 time=46 ms

--- 10.0.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/18/46 ms

其余PC的连通性测试省略。

由于本实验场景需要，请首先将S1与S2上互连的GE 0/0/2和GE 0/0/5接口关闭。

[S1]interface GigabitEthernet 0/0/2

[S1-GigabitEthernet0/0/2]shutdown

[S1-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5

[S1-GigabitEthernet0/0/5]shutdown

[S2]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]shutdown

[S2-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5

[S2-GigabitEthernet0/0/5]shutdown

## 未配置Eth-Trunk时的现象验证

在原有的网络环境中，公司在两台核心交换机间只部署了一条链路。但随着业务增长，数据量的增大，带宽出现了瓶颈，已经无法满足公司的业务需求，也无法实现冗余备份。考虑以上问题，公司网络管理员考虑通过增加链路的方式来提升带宽。在原有的网络基础上再增加一条链路来提高带宽，原链路只有一条，带宽为1Gb/s，再增加一条链路，将带宽增加到2Gb/s。

模拟链路增加，开启S1和S2上的GE 0/0/2接口。

[S1]interface GigabitEthernet 0/0/2

[S1-GigabitEthernet0/0/2]undo shutdown

[S2]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]undo shutdown

增加链路后，网络管理员考虑到，在该组网拓扑下，默认开启的STP协议一定会将其中一条链路阻塞掉。

查看S1和S2的STP状态信息。

[S1]display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/1 DESI FORWARDING NONE

0 GigabitEthernet0/0/2 DESI FORWARDING NONE

0 GigabitEthernet0/0/3 DESI FORWARDING NONE

[S2]display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/1 ROOT FORWARDING NONE

0 GigabitEthernet0/0/2 ALTE DISCARDING NONE

0 GigabitEthernet0/0/3 DESI FORWARDING NONE

可以观察到S2的GE 0/0/2接口处于丢弃状态。如果要实质性的增加S1和S2之间的带宽，显然单靠增加链路条数是不够的。生成树会阻塞多余接口，使得目前S1与S2之间的数据仍然仅通过GE 0/0/1接口传输。

## 配置Eth-Trunk实现链路聚合（手工负载分担模式）

通过上一步骤，发现仅靠简单增加互联的链路，不但无法解决目前带宽不够用的问题，还会在切换时带来断网的问题，显然是不合理的。此时网络管理员通过配置Eth-Trunk链路聚合来增加链路带宽，并可确保冗余链路。

Eth-Trunk工作模式可以分为两种：

手工负载分担模式。需要手动创建链路聚合组，并配置多个接口加入到所创建的Eth-trunk中；

静态LACP模式。该模式通过LACP协议协商Eth-trunk参数后并自主选择活动接口。

在S1和S2上配置链路聚合，创建Eth-Trunk 1接口，并指定为手工负载分担模式。

[S1]interface Eth-Trunk 1

[S1-Eth-Trunk1]mode manual load-balance

[S2]interface Eth-Trunk 1

[S2-Eth-Trunk1]mode manual load-balance

将S1和S2的GE 0/0/1和GE 0/0/2分别加入到Eth-Trunk 1接口。

[S1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]eth-trunk 1

[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S1-GigabitEthernet0/0/2]eth-trunk 1

[S2]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]eth-trunk 1

[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]eth-trunk 1

配置完成后，使用**display eth-trunk**命令查看S1和S2的Eth-Trunk 1 接口状态。

[S1]display eth-trunk 1

Eth-Trunk1's state information is:

WorkingMode: NORMAL Hash arithmetic: According to SIP-XOR-DIP

Least Active-linknumber: 1 Max Bandwidth-affected-linknumber: 8

Operate status: up Number Of Up Port In Trunk: 2

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

PortName Status Weight

GigabitEthernet0/0/1 Up 1

GigabitEthernet0/0/2 Up 1

[S2]display eth-trunk 1

Eth-Trunk1's state information is:

WorkingMode: NORMAL Hash arithmetic: According to SIP-XOR-DIP

Least Active-linknumber: 1 Max Bandwidth-affected-linknumber: 8

Operate status: up Number Of Up Port In Trunk: 2

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

PortName Status Weight

GigabitEthernet0/0/1 Up 1

GigabitEthernet0/0/2 Up 1

可以观察到，S1与S2的工作模式为NORMAL（手工负载分担方式），GE 0/0/1与GE 0/0/2接口已经添加到Eth-Trunk 1中，并且处于UP状态。

使用**display interface eth-trunk**命令查看S2的Eth-Trunk 1接口信息。

[S2]display interface Eth-Trunk 1

Eth-Trunk1 current state : UP

Line protocol current state : UP

Description:

Switch Port, PVID : 1, Hash arithmetic : According to SIP-XOR-DIP,Maximal BW: 4294967.29G, Current BW: 4294967.29G, The Maximum Frame Length is 9216

IP Sending Frames' Format is PKTFMT\_ETHNT\_2, Hardware address is 4c1f-cc71-68d4

Current system time: 2013-06-29 22:34:26-08:00

Input bandwidth utilization : 0%

Output bandwidth utilization : 0%

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

PortName Status Weight

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

GigabitEthernet0/0/1 UP 1

GigabitEthernet0/0/2 UP 1

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

The Number of Ports in Trunk : 2

The Number of UP Ports in Trunk : 2

可以观察到，目前该接口的总带宽，是GE 0/0/1和GE 0/0/2接口带宽之和。

查看S2接口的生成树状态。

[S2]display stp brief

MSTID Port Role STP State Protection

0 GigabitEthernet0/0/3 DESI FORWARDING NONE

0 Eth-Trunk1 ROOT FORWARDING NONE

可以观察到，S2的2个接口被捆绑成一个Eth-Trunk接口，并且该接口现在处于转发状态。

使用**ping**命令持续测试，同时将S2的GE 0/0/1或者GE 0/0/2接口关闭模拟故障发生。

PC>ping 10.0.1.2 -t

Ping 10.0.1.2: 32 data bytes, Press Ctrl\_C to break

From 10.0.1.2: bytes=32 seq=1 ttl=128 time=63 ms

From 10.0.1.2: bytes=32 seq=2 ttl=128 time=31 ms

From 10.0.1.2: bytes=32 seq=3 ttl=128 time=31 ms

From 10.0.1.2: bytes=32 seq=4 ttl=128 time=16 ms

From 10.0.1.2: bytes=32 seq=5 ttl=128 time=31 ms

Request timeout!

From 10.0.1.2: bytes=32 seq=7 ttl=128 time=47 ms

From 10.0.1.2: bytes=32 seq=8 ttl=128 time=15 ms

From 10.0.1.2: bytes=32 seq=9 ttl=128 time=31 ms

From 10.0.1.2: bytes=32 seq=10 ttl=128 time=16 ms

可以观察到，当链路故障发生时，链路立刻进行切换，数据包仅丢了一个，并且只要物理链路有一条是正常的，Eth-Trunk接口就不会断开，仍然可以保证数据的转发。Eth-Trunk在提高了带宽的情况下，也实现了链路冗余。模拟完成后将S2接口恢复。

## 配置Eth-Trunk实现链路聚合（静态LACP模式）

在上一步骤中，两条链路中，假设一条链路出现了故障，只有一条链路正常工作的情况下无法保证带宽。现网络管理员为公司再部署一条链路作为备份链路，并采用静态LACP模式配置Eth-Trunk实现两条链路同时转发，一条链路备份，当其中一条转发链路出现问题的时候，备份链路可立即进行数据转发。

开启S1与S2上的GE 0/0/5接口模拟增加了一条新链路。

[S1]interface GigabitEthernet 0/0/5

[S1-GigabitEthernet0/0/5]undo shutdown

[S2]interface GigabitEthernet 0/0/5

[S2-GigabitEthernet0/0/5]undo shutdown

在S1和S2上的Eth-Trunk 1接口下，将工作模式改为静态LACP模式。

[S1]interface Eth-Trunk 1

[S1-Eth-Trunk1]mode lacp-static

Error: Error in changing trunk working mode. There is(are) port(s) in the trunk.

[S2]interface Eth-Trunk 1

[S2-Eth-Trunk1]mode lacp-static

Error: Error in changing trunk working mode. There is(are) port(s) in the trunk.

发现报错，此时需要将先前已经加入到Eth-Trunk接口下的物理接口先删除。

[S1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]undo eth-trunk 1

[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S1-GigabitEthernet0/0/2]undo eth-trunk 1

[S2]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]undo eth-trunk 1

[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S2-GigabitEthernet0/0/2]undo eth-trunk 1

删除完成后，再在S1和S2上的Eth-Trunk 1接口下，将工作模式改为静态LACP模式，并将S1和S2的GE 0/0/1，GE 0/0/2和GE 0/0/5接口分别加入到Eth-Trunk 1接口。

[S1]interface Eth-Trunk 1

[S1-Eth-Trunk1]mode lacp-static

[S1-Eth-Trunk1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]eth-trunk 1

[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2 [S1-GigabitEthernet0/0/2]eth-trunk 1

[S1-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5

[S1-GigabitEthernet0/0/5]eth-trunk 1

[S2]interface Eth-Trunk 1

[S2-Eth-Trunk1]mode lacp-static

[S2-Eth-Trunk1]interface GigabitEthernet 0/0/1

[S2-GigabitEthernet0/0/1]eth-trunk 1

[S2-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2 [S2-GigabitEthernet0/0/2]eth-trunk 1

[S2-GigabitEthernet0/0/2]interface GigabitEthernet 0/0/5

[S2-GigabitEthernet0/0/5]eth-trunk 1

配置完成后，查看S1的Eth-Trunk 1 接口状态。

[S1]display eth-trunk 1

Eth-Trunk1's state information is:

Local:

LAG ID: 1 WorkingMode: STATIC

Preempt Delay: Disabled Hash arithmetic: According to SIP-XOR-DIP

System Priority: 32768 System ID: 4c1f-cc55-b90f

Least Active-linknumber: 1 Max Active-linknumber: 8

Operate status: up Number Of Up Port In Trunk: 3

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

ActorPortName Status PortType PortPri PortNo PortKey PortState Weight

GigabitEthernet0/0/2 Selected 1000TG 32768 3 401 10111100 1

GigabitEthernet0/0/1 Selected 1000TG 32768 2 401 10111100 1

GigabitEthernet0/0/5 Selected 1000TG 32768 6 401 10111100 1

Partner:

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

ActorPortName SysPri SystemID PortPri PortNo PortKey PortState

GigabitEthernet0/0/2 32768 4c1f-cc71-68d4 32768 3 401 10111100

GigabitEthernet0/0/1 32768 4c1f-cc71-68d4 32768 2 401 10111100

GigabitEthernet0/0/5 32768 4c1f-cc71-68d4 32768 6 401 10111100

可以观察到，三个接口默认都处于活动状态（Selected）。

将S1的系统优先级从默认的32768改为100，使其成为主动端（值越低优先级越高）。按照主动端设备的接口来选择活动接口。两端设备选出主动端后，两端都会以主动端的接口优先级来选择活动接口。两端设备选择了一致的活动接口，活动链路组便可以建立起来，从这些活动链路中以负载分担的方式转发数据。

[S1]lacp priority 100

配置完成后，查看S1的Eth-Trunk 1 接口状态。

[S1]display eth-trunk 1

Eth-Trunk1's state information is:

Local:

LAG ID: 1 WorkingMode: STATIC

Preempt Delay: Disabled Hash arithmetic: According to SIP-XOR-DIP

System Priority: 100 System ID: 4c1f-cc55-b90f

Least Active-linknumber: 1 Max Active-linknumber: 8

……

可以观察到，已经将S1的LACP系统优先级改为100，而S2没修改，仍为默认值。

在S1上配置活动接口上限阈值为2。

[S1]interface Eth-Trunk 1

[S1-Eth-Trunk1]max active-linknumber 2

在S1上配置接口的优先级确定活动链路。

[S1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]lacp priority 100

[S1-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/2

[S1-GigabitEthernet0/0/2]lacp priority 100

配置接口的活动优先级将默认的32768改为100，目的是为了使得GE 0/0/1和GE 0/0/2接口成为活动状态。

配置完成后，查看S1的Eth-Trunk 1 接口状态。

[S1]display eth-trunk 1

Eth-Trunk1's state information is:

Local:

LAG ID: 1 WorkingMode: STATIC

Preempt Delay: Disabled Hash arithmetic: According to SIP-XOR-DIP

System Priority: 100 System ID: 4c1f-cc55-b90f

Least Active-linknumber: 1 Max Active-linknumber: 2

Operate status: up Number Of Up Port In Trunk: 2

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

ActorPortName Status PortType PortPri PortNo PortKey PortState Weight

GigabitEthernet0/0/2 Selected 1000TG 100 3 401 10111100 1

GigabitEthernet0/0/1 Selected 1000TG 100 2 401 10111100 1

GigabitEthernet0/0/5 Unselect 1000TG 32768 6 401 10100000 1

Partner:

……

可以观察到，由于将接口的阈值改为2（默认活动接口最大阈值为8），该Eth-Trunk接口下将只有两个成员处于活动状态，并且具有负载分担能力。而GE 0/0/5接口已处于不活动状态（Unselect），该链路作为备份链路。当活动链路出现故障时，备份链路将会替代故障链路，保持数据传输的可靠性。

将S1的GE 0/0/1接口关闭，验证Eth-Trunk链路聚合信息。

[S1]interface GigabitEthernet 0/0/1

[S1-GigabitEthernet0/0/1]shutdown

[S1-GigabitEthernet0/0/1]display eth-trunk 1

Eth-Trunk1's state information is:

Local:

LAG ID: 1 WorkingMode: STATIC

Preempt Delay: Disabled Hash arithmetic: According to SIP-XOR-DIP

System Priority: 100 System ID: 4c1f-cc27-e139

Least Active-linknumber: 1 Max Active-linknumber: 2

Operate status: up Number Of Up Port In Trunk: 2

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

ActorPortName Status PortType PortPri PortNo PortKey PortState Weight

GigabitEthernet0/0/1 Unselect 1GE 100 2 305 10100010 1

GigabitEthernet0/0/2 Selected 1GE 100 3 305 10111100 1

GigabitEthernet0/0/5 Selected 1GE 32768 6 305 10111100 1

Partner:

……

可以观察到，S1的GE 0/0/1接口已经处于不活动状态，而GE 0/0/5接口为活动状态。将S1的GE 0/0/1接口开启后，又会恢复为活动状态，GE 0/0/5则为不活动状态，现象省略。

至此，完成了整个Eth-Trunk的部署。

# 思考

当接口数超出最大负载阈值时，剩余接口是否转发流量？

1. 静态路由

6.1 静态路由及默认路由基本配置

# 原理概述

静态路由是指用户或网络管理员手工配置的路由信息。当网络的拓扑结构或链路状态发生改变时，需要网络管理人员手工修改静态路由信息，静态路由一般适用于比较简单的网络环境。

优点：相比动态路由协议无需频繁的交换各自的路由表，配置简单，适合小型网络环境。

缺点：静态路由不适合大型和复杂的网络环境，当网络拓扑结构和链路状态发生变化时，网络管理员需要做大量的调整，且无法自动感知错误发生，不易排错。

默认路由是一种特殊的静态路由，当路由表中与数据包目的地址没有匹配的表项时，数据包将根据默认路由条目进行转发。默认路由在某些时候非常有效，如在末梢网络中，默认路由可以大大简化路由器配置，减轻网络管理员的工作负担。

# 实验目的

* + 掌握配置静态路由（指定接口）的方法
  + 掌握配置静态路由（指定下一跳IP地址）的方法
  + 掌握测试静态路由连通性的方法
  + 掌握配置默认路由的方法
  + 掌握测试默认路由的方法
  + 掌握在简单网络中部署静态路由时的故障排除方法
  + 掌握简单的网络优化方法

# 实验内容

在由三台路由器所组成的简单网络中，R1与R3各自连接着一台主机，现在要求能够实现主机PC-1与PC-2之间的正常通信，本实验将通过配置基本的静态路由和默认路由来实现。

# 实验拓扑

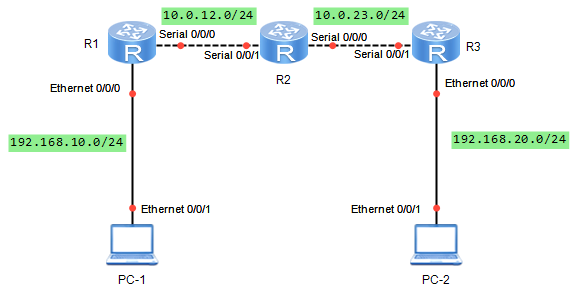


图6-1 静态路由及默认路由基本配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 192.168.10.10 | 255.255.255.0 | 192.168.10.1 |
| R1(AR2220) | Ethernet 0/0/0 | 192.168.10.1 | 255.255.255.0 | N/A |
| Serial 0/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| R2(AR2220) | Serial 0/0/1 | 10.0.12.2 | 255.255.255.0 | N/A |
| Serial 0/0/0 | 10.0.23.2 | 255.255.255.0 | N/A |
| R3(AR2220) | Serial 0/0/1 | 10.0.23.3 | 255.255.255.0 | N/A |
| Ethernet 0/0/0 | 192.168.20.3 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 192.168.20.20 | 255.255.255.0 | 192.168.20.3 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping -c 1 192.168.10.1 -c 1

PING 192.168.10.1: 56 data bytes, press CTRL\_C to break

Reply from 192.168.10.1: bytes=56 Sequence=1 ttl=255 time=510 ms

--- 192.168.10.1 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 510/510/510 ms

其余直连网段的连通性测试省略。

各直连链路间的IP连通性测试完成后，现尝试在主机PC-1上直接**ping**主机PC-2。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

发现无法ping通，这时需要思考是什么问题导致了它们之间无法通信。

首先假设主机PC-1与PC-2之间如果能够正常ping通，那么主机A将发送数据给其网关设备R1；R1收到后将根据数据包中的目的地址查看它的路由表，找到相应的目的网络的所在路由条目，并根据该条目中的下一跳和出接口信息将该数据转发给下一台路由器R2；R2采取同样的步骤将数据转发给R3；最后R3也采取同样的步骤将数据转发给与自己直连的主机PC-2；主机PC-2在收到数据后，与主机PC-1发送数据到PC-2的过程一样，再发送相应的回应消息给PC-1。

在保证基本配置没有错误的情况下，首先查看主机PC-1与其网关设备R1间能否正常通信。

PC>ping 192.168.10.1

Ping 192.168.10.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.10.1: bytes=32 seq=1 ttl=255 time=16 ms

From 192.168.10.1: bytes=32 seq=2 ttl=255 time=16 ms

From 192.168.10.1: bytes=32 seq=3 ttl=255 time=16 ms

From 192.168.10.1: bytes=32 seq=4 ttl=255 time=31 ms

From 192.168.10.1: bytes=32 seq=5 ttl=255 time<1 ms

--- 192.168.10.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/15/31 ms

主机与网关之间通信正常，接下来检查网关设备R1上的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

可以看到在R1的路由表上，没有任何有关于主机PC-2所在网段的信息。可以使用同样的方式查看R2与R3的路由表。

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Serial0/0/1

10.0.12.1/32 Direct 0 0 D 10.0.12.1 Serial0/0/1

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/1

10.0.23.0/24 Direct 0 0 D 10.0.23.2 Serial0/0/0

10.0.23.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.23.3/32 Direct 0 0 D 10.0.23.3 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

<R3>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.23.0/24 Direct 0 0 D 10.0.23.3 Serial0/0/1

10.0.23.2/32 Direct 0 0 D 10.0.23.2 Serial0/0/1

10.0.23.3/32 Direct 0 0 D 127.0.0.1 Serial0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.20.0/24 Direct 0 0 D 192.168.20.3 Ethernet0/0/0

192.168.20.3/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

可以看到在R2上没有任何有关于主机PC-1和PC-2所在网段的信息，R3上没有任何关于主机PC-1所在网段的信息，验证了初始情况下各路由器的路由表上仅包括了与自身直接相连的网段的路由信息。

现在主机PC-1与PC-2之间跨越了若干个不同网段，要实现它们之间的通信，只通过简单的IP地址等基本配置是无法实现的，必须在三台路由器上添加相应的路由信息，可以通过配置静态路由来实现。

静态路由的配置可以分成两种方式，一种是在配置中采取指定下一跳IP地址的方式，另一种是指定出接口的方式。

## 实现主机PC-1与PC-2之间的通信

在R1上配置目的网段为主机PC-2所在网段的静态路由，即目的IP地址为192.168.20.0，掩码为255.255.255.0。对于R1而言，要发送数据到主机PC-2，则必须先发送给R2，所以R2即为R1的下一跳路由器，R2与R1所在的直连链路上的物理接口的IP地址即为下一跳IP地址，即10.0.12.2。

[R1]ip route-static 192.168.20.0 255.255.255.0 10.0.12.2

配置完成后，查看R1上的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

192.168.20.0/24 Static 60 0 RD 10.0.12.2 Serial0/0/0

配置完成之后，可以在R1的路由表上查看到主机PC-2所在网段的路由信息。

采取同样的方式在R2上配置目的网段为主机PC-2所在网段的静态路由。

[R2]ip route-static 192.168.20.0 255.255.255.0 10.0.23.3

配置完成后，查看R2上的路由表。

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Serial0/0/1

10.0.12.1/32 Direct 0 0 D 10.0.12.1 Serial0/0/1

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/1

10.0.23.0/24 Direct 0 0 D 10.0.23.2 Serial0/0/0

10.0.23.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.23.3/32 Direct 0 0 D 10.0.23.3 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.20.0/24 Static 60 0 RD 10.0.23.3 Serial0/0/0

配置完成之后，可以在R2的路由表上查看到主机PC-2所在网段的路由信息。

此时在主机PC-1上**ping**主机PC-2。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

发现仍然无法ping通，并在主机PC-1的E0/0/1接口上进行数据抓包可以观察到如下现象。

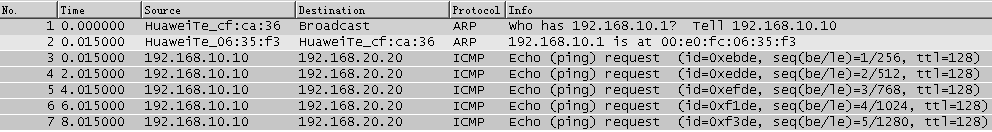


图6-2

此时主机PC-1仅发送了ICMP请求消息，并没有收到任何回应消息。

原因在于现在仅仅实现了PC-1能够通过路由将数据正常转发给PC-2，而PC-2仍然无法发送数据给PC-1，所以同样需要在R2和R3的路由表上添加PC-1所在网段的路由信息。

在R3上配置目的网段为PC-1所在网段的静态路由，即目的IP地址为192.168.10.0，目的地址的掩码除了可以采用点分十进制的格式表示外，还可以直接使用掩码长度，即24来表示。又对于R3而言，要发送数据到PC-1，则必须先发送给R2，所以R3与R2所在直连链路上的物理接口Serial 0/0/1即为数据转发接口，也称为出接口，在配置中指定该接口即可。

[R3]ip route-static 192.168.10.0 24 Serial 0/0/1

采取同样的方式在R2上配置目的网段为PC-1所在网段的静态路由。

[R2]ip route-static 192.168.10.0 24 Serial 0/0/1

配置完成后，查看R1，R2，R3上的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

192.168.20.0/24 Static 60 0 RD 10.0.12.2 Serial0/0/0

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Serial0/0/1

10.0.12.1/32 Direct 0 0 D 10.0.12.1 Serial0/0/1

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/1

10.0.23.0/24 Direct 0 0 D 10.0.23.2 Serial0/0/0

10.0.23.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.23.3/32 Direct 0 0 D 10.0.23.3 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Static 60 0 D 10.0.12.2 Serial0/0/1

192.168.20.0/24 Static 60 0 RD 10.0.23.3 Serial0/0/0

<R3>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.23.0/24 Direct 0 0 D 10.0.23.3 Serial0/0/1

10.0.23.2/32 Direct 0 0 D 10.0.23.2 Serial0/0/1

10.0.23.3/32 Direct 0 0 D 127.0.0.1 Serial0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Static 60 0 D 10.0.23.2 Serial0/0/1

192.168.20.0/24 Direct 0 0 D 192.168.20.3 Ethernet0/0/0

192.168.20.3/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

可以看到，现在每台路由器上都拥有了主机PC-1与PC-2所在网段的路由信息。再在主机PC-1上**ping**主机PC-2。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.20: bytes=32 seq=1 ttl=125 time=78 ms

From 192.168.20.20: bytes=32 seq=2 ttl=125 time=47 ms

From 192.168.20.20: bytes=32 seq=3 ttl=125 time=47 ms

From 192.168.20.20: bytes=32 seq=4 ttl=125 time=62 ms

From 192.168.20.20: bytes=32 seq=5 ttl=125 time=63 ms

--- 192.168.20.20 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 47/59/78 ms

可以ping通，即现在已经实现了主机PC-1与PC-2之间的正常通信。

## 实现全网全通来增强网络的可靠性

经过上面的步骤，主机PC-1与PC-2之间已经能够正常通信。

假设此时网络突然出现故障，主机PC-1侧的网络管理员发现无法与PC-2正常通信，于是先测试与网关设备R1间的连通性。

PC>ping 192.168.10.1

Ping 192.168.10.1: 32 data bytes, Press Ctrl\_C to break

From 192.168.10.1: bytes=32 seq=1 ttl=255 time<1 ms

From 192.168.10.1: bytes=32 seq=2 ttl=255 time=16 ms

From 192.168.10.1: bytes=32 seq=3 ttl=255 time=16 ms

From 192.168.10.1: bytes=32 seq=4 ttl=255 time=16 ms

From 192.168.10.1: bytes=32 seq=5 ttl=255 time=16 ms

--- 192.168.10.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/12/16 ms

发现与网关间的通信正常，再测试与主机PC-2的网关设备R3间的连通性。

PC>ping 10.0.23.3

Ping 10.0.23.3: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

发现无法与R3正常通信，这也意味着此时网络管理员将无法通过主机PC-1登录到R3上进行进一步的故障排除，由此可见，保证全网的连通性能够增强整网的可靠性，提高网络的可维护性，及健壮性。

因此有必要在R1的路由表中添加R2与R3间直连网段的路由信息，同样也应在R3的路由表中添加R1与R2间直连网段的路由信息，实现全网全通。

[R1]ip route-static 10.0.23.0 24 10.0.12.2

[R3]ip route-static 10.0.12.0 24 Serial 0/0/1

配置完成后，查看R1，R2，R3的路由表，注意观察新增的条目。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial0/0/0

10.0.23.0/24 Static 60 0 RD 10.0.12.2 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

192.168.20.0/24 Static 60 0 RD 10.0.12.2 Serial0/0/0

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Serial0/0/1

10.0.12.1/32 Direct 0 0 D 10.0.12.1 Serial0/0/1

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/1

10.0.23.0/24 Direct 0 0 D 10.0.23.2 Serial0/0/0

10.0.23.2/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.23.3/32 Direct 0 0 D 10.0.23.3 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Static 60 0 D 10.0.12.2 Serial0/0/1

192.168.20.0/24 Static 60 0 RD 10.0.23.3 Serial0/0/0

<R3>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Static 60 0 D 10.0.23.3 Serial0/0/1

10.0.23.0/24 Direct 0 0 D 10.0.23.3 Serial0/0/1

10.0.23.2/32 Direct 0 0 D 10.0.23.2 Serial0/0/1

10.0.23.3/32 Direct 0 0 D 127.0.0.1 Serial0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Static 60 0 D 10.0.23.3 Serial0/0/1

192.168.20.0/24 Direct 0 0 D 192.168.20.3 Ethernet0/0/0

192.168.20.3/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

此时再在主机PC-1上测试与R3间的连通性。

PC>ping 10.0.23.3

Ping 10.0.23.3: 32 data bytes, Press Ctrl\_C to break

From 10.0.23.3: bytes=32 seq=1 ttl=253 time=47 ms

From 10.0.23.3: bytes=32 seq=2 ttl=253 time=47 ms

From 10.0.23.3: bytes=32 seq=3 ttl=253 time=47 ms

From 10.0.23.3: bytes=32 seq=4 ttl=253 time=47 ms

From 10.0.23.3: bytes=32 seq=5 ttl=253 time=93 ms

--- 10.0.23.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 47/56/93 ms

测试成功，主机PC-1可以顺利与R3通信，同样主机PC-2此时也能够与R1进行通信，测试过程这里省略。

## 使用默认路由实现简单的网络优化

通过适当的减少设备上的配置工作量，能够帮助网络管理员在进行故障排除时更轻松的定位故障，且相对较少的配置量也能减少在配置时出错的可能，另一方面，也能够相对减少对设备本身硬件的负担。

默认路由是一种特殊的静态路由，使用默认路由可以简化路由器上的配置。

查看此时R1上的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial0/0/0

10.0.23.0/24 Static 60 0 RD 10.0.12.2 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

192.168.20.0/24 Static 60 0 RD 10.0.12.2 Serial0/0/0

此时R1上存在两条先前经过手动配置的静态路由条目，且它们的下一跳和出接口都一致。

现在在R1上配置一条默认路由，即目的网段和掩码为全0，表示任何网络，下一跳为10.0.12.2，并删除先前配置的两条静态路由。

[R1]ip route-static 0.0.0.0 0 10.0.12.2

[R1]undo ip route-static 10.0.23.0 255.255.255.0 10.0.12.2

[R1]undo ip route-static 192.168.20.0 255.255.255.0 10.0.12.2

配置完成后，查看R1的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 Static 60 0 RD 10.0.12.2 Serial0/0/0

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial0/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

再测试主机PC-1与PC-2间的通信。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.20: bytes=32 seq=1 ttl=125 time=63 ms

From 192.168.20.20: bytes=32 seq=2 ttl=125 time=47 ms

From 192.168.20.20: bytes=32 seq=3 ttl=125 time=31 ms

From 192.168.20.20: bytes=32 seq=4 ttl=125 time=47 ms

From 192.168.20.20: bytes=32 seq=5 ttl=125 time=47 ms

--- 192.168.20.20 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/47/63 ms

发现主机PC-1与PC-2间的通信正常，证明使用默认路由不但能够实现与静态路由同样的效果，而且还能够减少配置量。在R3上可以进行同样的配置。

[R3]ip route-static 0.0.0.0 0 Serial 0/0/1

[R3]undo ip route-static 10.0.12.0 255.255.255.0 Serial 0/0/1

[R3]undo ip route-static 192.168.10.0 255.255.255.0 Serial 0/0/1

再次测试主机PC-1与PC-2间的通信。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.20: bytes=32 seq=1 ttl=125 time=78 ms

From 192.168.20.20: bytes=32 seq=2 ttl=125 time=62 ms

From 192.168.20.20: bytes=32 seq=3 ttl=125 time=47 ms

From 192.168.20.20: bytes=32 seq=4 ttl=125 time=78 ms

From 192.168.20.20: bytes=32 seq=5 ttl=125 time=62 ms

--- 192.168.20.20 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 47/65/78 ms

主机PC-1与PC-2间的通信正常。

此外，请注意在上面的配置过程中，采取的配置顺序是先配置默认路由，再删除原有的静态路由配置，这样的操作可以避免网络出现通信中断，即要在配置过程当中注意操作的规范性与合理性。

# 思考

在静态路由配置当中，可以采取指定下一跳IP地址的方式，也可以采取指定出接口的方式，这两种方式存在着什么区别？

6.2 浮动静态路由及负载均衡

# 原理概述

浮动静态路由（Floating Static Route）是一种特殊的静态路由，通过配置去往相同的目的网段，但优先级不同的静态路由，以保证在网络中优先级较高的路由，即主路由失效的情况下，提供备份路由。正常情况下，备份路由不会出现在路由表中。

负载均衡（Load sharing），当数据有多条可选路径前往同一目的网络，可以通过配置相同优先级和开销的静态路由实现负载均衡，使得数据的传输均衡的分配到多条路径上，从而实现数据分流、减轻单条路径负载过重的效果。而当其中某一条路径失效时，其他路径仍然能够正常传输数据，也起到了冗余作用。

# 实验目的

* + 理解浮动静态路由的应用场景
  + 掌握配置浮动静态路由的方法
  + 掌握测试浮动静态路由的方法
  + 掌握配置静态路由负载均衡的方法
  + 掌握测试静态路由负载均衡的方法

# 实验内容

R2为某公司总部，R1与R3是2个分部，主机PC-1与PC-2的所在网段分别模拟两个分部中的办公网络。现需要总部与各个分部，分部与分部之间都要能够通信。且分部之间在通信时，之间的直连链路为主用链路，通过总部的链路为备用链路。本实验使用浮动静态路由实现需求，并再根据实际需求实现负载均衡来优化网络。

# 实验拓扑

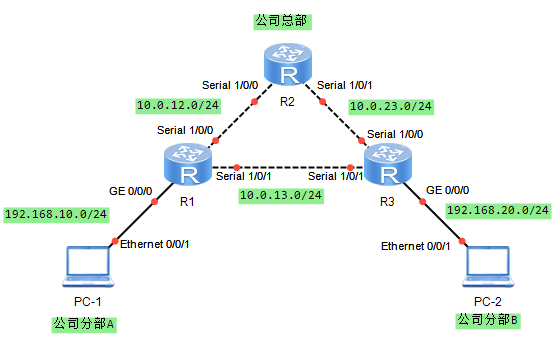


图6-3 浮动静态路由及负载均衡拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 192.168.10.10 | 255.255.255.0 | 192.168.10.1 |
| R1(AR2220) | GE 0/0/0 | 192.168.10.1 | 255.255.255.0 | N/A |
| Serial 1/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| Serial 1/0/1 | 10.0.13.1 | 255.255.255.0 | N/A |
| R2(AR2220) | Serial 1/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| Serial 1/0/1 | 10.0.23.2 | 255.255.255.0 | N/A |
| R3(AR2220) | Serial 1/0/0 | 10.0.23.3 | 255.255.255.0 | N/A |
| Serial 1/0/1 | 10.0.13.3 | 255.255.255.0 | N/A |
| GE 0/0/0 | 192.168.20.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 192.168.20.20 | 255.255.255.0 | 192.168.20.1 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping -c 1 10.1.12.1

PING 10.1.12.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.12.1: bytes=56 Sequence=1 ttl=255 time=510 ms

--- 192.168.10.1 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 510/510/510 ms

其余直连网段的连通性测试省略。

## 实现两分部间、总部与两分部间的通信

在R1上配置目的网段为主机PC-2所在网段的静态路由，在R3上配置目的网段为主机PC-1所在网段的静态路由，在R2上配置目的网段分别为主机PC-1和PC-2所在网段的静态路由。

[R1]ip route-static 192.168.20.0 24 10.0.13.3

[R2]ip route-static 192.168.20.0 24 10.0.23.3

[R2]ip route-static 192.168.10.0 24 10.0.12.1

[R3]ip route-static 192.168.10.0 24 10.0.13.1

配置完成后，在R1上查看路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.13.0/24 Direct 0 0 D 10.0.13.1 Serial1/0/1

10.0.13.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

10.0.13.3/32 Direct 0 0 D 10.0.13.3 Serial1/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.20.0/24 Static 60 0 RD 10.0.13.3 Serial1/0/1

可以观察到，在R1的路由表中存在以主机PC-2所在网段为目的网段的路由条目，且下一跳路由器为R3。

测试主机PC-1与主机PC-2之间的连通性。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.20: bytes=32 seq=1 ttl=126 time=31 ms

From 192.168.20.20: bytes=32 seq=2 ttl=126 time=32 ms

From 192.168.20.20: bytes=32 seq=3 ttl=126 time=31 ms

From 192.168.20.20: bytes=32 seq=4 ttl=126 time=62 ms

From 192.168.20.20: bytes=32 seq=5 ttl=126 time=47 ms

--- 192.168.20.20 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/40/62 ms

通信正常，这时可以通过在主机PC-1上使用**tracert**命令测试所经过的网关。

PC>tracert 192.168.20.20

traceroute to 192.168.20.20, 8 hops max

(ICMP), press Ctrl+C to stop

1 192.168.10.1 16 ms 15 ms 16 ms

2 10.0.13.3 47 ms 15 ms 31 ms

3 192.168.20.20 47 ms 47 ms 16 ms

发现数据包是经过R1和R3到达主机PC-2的。

同样在主机PC-2和R3上进行查看，首先在R3上查看路由表:

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 11 Routes : 11

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.13.0/24 Direct 0 0 D 10.0.13.3 Serial1/0/1

10.0.13.1/32 Direct 0 0 D 10.0.13.1 Serial1/0/1

10.0.13.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

10.0.23.0/24 Direct 0 0 D 10.0.23.3 Serial1/0/0

10.0.23.2/32 Direct 0 0 D 10.0.23.2 Serial1/0/0

10.0.23.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Static 60 0 RD 10.0.13.1 Serial0/0/1

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/0

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

在R3的路由表中存在以主机PC-1所在网段为目的网段的路由条目，且下一跳路由器为R1。

在主机PC-2上测试与主机PC-1的连通性。

PC>ping 192.168.10.10

Ping 192.168.10.10: 32 data bytes, Press Ctrl\_C to break

From 192.168.10.10: bytes=32 seq=1 ttl=126 time=32 ms

From 192.168.10.10: bytes=32 seq=2 ttl=126 time=31 ms

From 192.168.10.10: bytes=32 seq=3 ttl=126 time=46 ms

From 192.168.10.10: bytes=32 seq=4 ttl=126 time=31 ms

From 192.168.10.10: bytes=32 seq=5 ttl=126 time=16 ms

--- 192.168.10.10 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 16/31/46 ms

可以观察到通信正常。在主机PC-2上测试访问主机PC-1所经过的网关。

PC>tracert 192.168.10.10

traceroute to 192.168.10.10, 8 hops max

(ICMP), press Ctrl+C to stop

1 192.168.20.1 16 ms <1 ms 15 ms

2 10.0.13.1 47 ms 16 ms 31 ms

3 192.168.10.10 31 ms 31 ms 16 ms

可以验证数据包是经过R3和R1到达主机PC-1的。

在总部路由器R2上测试与分部的连通性。

[R2]ping 192.168.10.10

PING 192.168.10.10: 56 data bytes, press CTRL\_C to break

Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=127 time=50 ms

Reply from 192.168.10.10: bytes=56 Sequence=2 ttl=127 time=40 ms

Reply from 192.168.10.10: bytes=56 Sequence=3 ttl=127 time=30 ms

Reply from 192.168.10.10: bytes=56 Sequence=4 ttl=127 time=50 ms

Reply from 192.168.10.10: bytes=56 Sequence=5 ttl=127 time=60 ms

--- 192.168.10.10 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/46/60 ms

[R2]ping 192.168.20.20

PING 192.168.20.20: 56 data bytes, press CTRL\_C to break

Reply from 192.168.20.20: bytes=56 Sequence=1 ttl=127 time=40 ms

Reply from 192.168.20.20: bytes=56 Sequence=2 ttl=127 time=40 ms

Reply from 192.168.20.20: bytes=56 Sequence=3 ttl=127 time=50 ms

Reply from 192.168.20.20: bytes=56 Sequence=4 ttl=127 time=40 ms

Reply from 192.168.20.20: bytes=56 Sequence=5 ttl=127 time=10 ms

--- 192.168.20.20 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/36/50 ms

通过测试，总部路由器R2能够正常访问两个分部主机PC-1和主机PC-2的网络。

## 配置浮动静态路由实现路由备份

通过上一步骤的配置，现在网络搭建已经初步完成。现需要实现当两分部间通信时，直连链路为主用链路，通过总部的链路为备用链路，即当主用链路发生故障时，可以使用备用链路保障两分部网络间的通信，这里使用浮动静态路由实现网络冗余。

在R1上配置静态路由，目的网段为主机PC-2所在网段，掩码为24位，下一跳为R2，将路由优先级设置为100（默认是60）。

[R1]ip route-static 192.168.20.0 24 10.0.12.2 preference 100

配置完成后，查看路由器R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 12 Routes : 12

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial1/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial1/0/0

10.0.13.0/24 Direct 0 0 D 10.0.13.1 Serial1/0/1

10.0.13.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

10.0.13.3/32 Direct 0 0 D 10.0.13.3 Serial1/0/1

10.0.23.0/24 Static 60 0 RD 10.0.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.20.0/24 Static 60 0 RD 10.0.13.3 Serial0/0/1

发现路由表此时没有发生任何变化，使用命令**display ip routing-table protocol static**仅查看静态路由的路由信息。

[R1]display ip routing-table protocol static

Route Flags: R - relay, D - download to fib

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

Public routing table : Static

Destinations : 1 Routes : 2 Configured Routes : 2

Static routing table status : <Active>

Destinations : 1 Routes : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

192.168.20.0/24 Static 60 0 RD 10.0.13.3 Serial1/0/1

Static routing table status : <Inactive>

Destinations : 1 Routes : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

192.168.20.0/24 Static 100 0 R 10.0.12.2 Serial1/0/0

可以观察到目的地址为PC-2所在网段的两条优先级为100和60的静态路由条目都已经存在。

现在R1上去往相同的目的网段存在有两条不同路由条目，首先会比较它们的优先级，优先级高的，即对应的优先级数值较小的路由条目将被选为主用路由。通过比较，优先级数值为60的条目优先级更高，将被R1使用，放入路由表中，状态为Active，而另一条路由状态则为Inactive，作为备份，不会被放入路由表。只有当Active的路由条目失效时，优先级为100的路由条目才会被放入路由表。

在R3上做和R1同样的对称配置。

[R3]ip route-static 192.168.10.0 24 10.0.23.2 preference 100

接下来，将路由器R1的S1/0/1接口关闭，验证使用备份链路。

[R1]interface seria1/0/1

[R1-Serial0/0/1]shutdown

配置完成后，查看路由器R1的路由表，并使用命令**display ip routing-table protocol static**查看。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial1/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial1/0/0

10.0.23.0/24 Static 60 0 RD 10.0.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.20.0/24 Static 100 0 RD 10.0.12.2 Serial1/0/0

可以观察到，此时优先级为100的路由条目已经添加到路由表中。

[R1]display ip routing-table protocol static

Route Flags: R - relay, D - download to fib

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

Public routing table : Static

Destinations : 1 Routes : 2 Configured Routes : 2

Static routing table status : <Active>

Destinations : 1 Routes : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

192.168.20.0/24 Static 100 0 RD 10.0.12.2 Serial1/0/0

Static routing table status : <Inactive>

Destinations : 1 Routes : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

192.168.20.0/24 Static 60 0 10.0.13.3 Unknown

可以观察到，现在优先级为100的条目为Active状态，优先级为60的条目为Inactive状态。

测试主机PC-1与PC-2间的通信。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.20: bytes=32 seq=1 ttl=125 time=63 ms

From 192.168.20.20: bytes=32 seq=2 ttl=125 time=31 ms

From 192.168.20.20: bytes=32 seq=3 ttl=125 time=47 ms

From 192.168.20.20: bytes=32 seq=4 ttl=125 time=46 ms

From 192.168.20.20: bytes=32 seq=5 ttl=125 time=62 ms

--- 192.168.20.20 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/49/63 ms

通信正常，再使用**tracert**命令查看此时PC-1与PC-2通信时所经过的网关。

PC>tracert 192.168.20.20

traceroute to 192.168.20.20, 8 hops max

(ICMP), press Ctrl+C to stop

1 192.168.10.1 <1 ms <1 ms 31 ms

2 10.1.12.2 32 ms 31 ms 15 ms

3 10.1.23.3 62 ms 47 ms 47 ms

4 192.168.20.20 62 ms 47 ms 31 ms

再次验证了此时两分部之间通信时已经使用了备用链路。

## 通过负载均衡实现网络优化

公司网络管理员发现分部之间业务往来越来越多，网络流量剧增，主用链路压力非常大，而总部与两分部间的网络流量相对较少，即备用链路上的带宽多处在闲置状态。此时可以通过配置实现负载均衡，即同时利用主备两条链路来支撑两分部间的通信。

恢复R1上的S1/0/1接口，并配置目的网段为主机PC-2所在网段，掩码为24位，下一跳为R2，优先级不变。

[R1]interface seria1/0/1

[R1-Serial0/0/1]undo shutdown

[R1-Serial0/0/1]ip route-static 192.168.20.0 24 10.0.12.2

查看R1上的路由表，及使用命令**display ip routing-table** 查看。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 12 Routes : 13

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Serial1/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.0.12.2/32 Direct 0 0 D 10.0.12.2 Serial1/0/0

10.0.13.0/24 Direct 0 0 D 10.0.13.1 Serial1/0/1

10.0.13.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

10.0.13.3/32 Direct 0 0 D 10.0.13.3 Serial1/0/1

10.0.23.0/24 Static 60 0 RD 10.0.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.20.0/24 Static 60 0 RD 10.0.13.3 Serial1/0/1

Static 60 0 RD 10.0.12.2 Serial1/0/0

配置完成后，可以观察到现在去往192.168.20.0网段拥有两条下一跳不同的路由条目，即实现了负载均衡。

测试主机PC-1与PC-2间的通信。

PC>ping 192.168.20.20

Ping 192.168.20.20: 32 data bytes, Press Ctrl\_C to break

From 192.168.20.20: bytes=32 seq=1 ttl=126 time=31 ms

From 192.168.20.20: bytes=32 seq=2 ttl=126 time=31 ms

From 192.168.20.20: bytes=32 seq=3 ttl=126 time=32 ms

From 192.168.20.20: bytes=32 seq=4 ttl=126 time=47 ms

From 192.168.20.20: bytes=32 seq=5 ttl=126 time=31 ms

--- 192.168.20.20 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/34/47 ms

可以观察到，通信正常。

在R3上做和R1同样的对称配置。

[R3]ip route-static 192.168.10.0 24 10.0.23.2

配置完成后，能够在R3的路由表中观察到与R1路由表相同的情况，这里省略。 62

通过配置针对相同目的地址但优先级值不同的静态路由，可以在路由器上实现路径备份的功能。而通过配置针对相同目的地址且优先级值相同的静态路由，不仅互为备份还能实现负载均衡。

# 思考

在本实验的步骤3和4中，如果不在R3上做和R1同样的对称配置，会产生什么样的现象？为什么？

完成负载均衡的配置之后，可以在R1上的S1/0/0和S1/0/1两个接口上启用抓包工具，且在主机PC-1上ping主机PC-2，观察R1的两个接口上的现象，为什么会产生这样的现象？

1. RIP

7.1 RIP路由协议基本配置

# 原理概述

RIP（Routing Information Protocol）路由协议作为最早的距离矢量IP路由协议，也是最先得到广泛使用的一种路由协议，采用了Bellman-Ford算法，其最大的特点就是配置简单。

RIP协议要求网络中每一台路由器都要维护从自身到每一个目的网络的路由信息。RIP协议使用跳数来衡量网络间的“距离”。从一台路由器到其直连网络的跳数定义为1。从一台路由器到其非直连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允许路由的最大跳数为15，因此，16即为不可达。可见RIP协议只适用于小型网络。

目前RIP有两个版本，RIPv1和RIPv2，RIPv2针对 RIPv1进行扩充，能够携带更多的信息量，并增强了安全性能。RIPv1和RIPv2都是基于UDP的协议，使用UDP520号端口收发数据包。

# 实验内容

某小型公司组网拓扑很简单，只拥有两台路由器，因此可以采用RIP路由协议来完成网络的部署。本实验通过模拟简单的企业网络场景来描述RIP路由协议的基本配置，介绍一些基本的查看RIP信息的命令使用方法。

# 实验目的

* + 理解RIP的应用场景
  + 理解RIP的基本原理
  + 掌握RIPv1的基本配置
  + 掌握RIPv2的基本配置
  + 掌握测试RIP路由网络的连通性的方法
  + 掌握使用display与debug命令测试RIP
  + 了解RIPv1与RIPv2的区别

# 实验拓扑

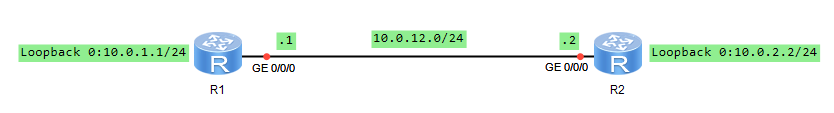


图7-1 RIP路由协议基本配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| Loopback 0 | 10.0.1.1 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| Loopback 0 | 10.0.2.2 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

[R1]interface loopback0

[R1-LoopBack0]ip address 10.0.1.1 24

[R2]interface loopback0

[R2-LoopBack0]ip address 10.0.2.2 24

根据实验编址表进行相应的基本配置，并使用**ping**命令检测直连链路的连通性。

[R1]ping -c 1 10.0.12.2

PING 10.0.12.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms

--- 10.0.12.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 50/50/50 ms

## 使用RIPv1搭建网络

在公司两台路由器R1和R2上配置RIP版本1。使用命令**rip**创建并开启协议进程，缺省情况进程号是1。使用**network**命令对指定网段接口使能RIP功能，注意必须是自然网段的地址。

[R1]rip

[R1-rip-1]network 10.0.0.0

[R2]rip

[R2-rip-1]network 10.0.0.0

配置完成后，使用**display ip routing-table**命令查看R1，R2的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 Direct 0 0 D 10.0.1.1 LoopBack0

10.0.1.1/32 Direct 0 0 D 127.0.0.1 LoopBack0

10.0.2.0/24 RIP 100 1 D 10.0.12.2 Ethernet0/0/0

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Ethernet0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 RIP 100 1 D 10.0.12.1 Ethernet0/0/0

10.0.2.0/24 Direct 0 0 D 10.0.2.2 LoopBack0

10.0.2.2/32 Direct 0 0 D 127.0.0.1 LoopBack0

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Ethernet0/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，两台路由器已经通过RIP协议学习到了对方环回接口所在网段的路由条目。

测试R1与R2环回接口间的连通性。

[R1]ping -a 10.0.1.1 10.0.2.2

PING 10.0.2.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.2.2: bytes=56 Sequence=1 ttl=255 time=1 ms

Reply from 10.0.2.2: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 10.0.2.2: bytes=56 Sequence=3 ttl=255 time=10 ms

Reply from 10.0.2.2: bytes=56 Sequence=4 ttl=255 time=20 ms

Reply from 10.0.2.2: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.0.2.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/10/20 ms

可以观察到通信正常。

使用**debug**命令查看RIP协议定期更新情况。

使用**debug**命令开启RIP调试功能。请注意，**debug**命令需要在用户视图下才能使用。并使用**terminal debugging**和 **terminal monitor**命令开启**debug**信息在屏幕上显示的功能。然后会看到电脑屏幕上显示出许多关于路由器之间RIP协议交互的信息。

<R1>debugging rip 1

<R1>terminal debugging

Info: Current terminal debugging is on.

<R1>terminal monitor

Info: Current terminal monitor is on.

<R1>

Dec 7 2012 11:20:22.530.1-08:00 R1 RIP/7/DBG: 6: 12176: RIP 1: Sending v1 response on Ethernet0/0/0 from 10.0.12.1 with 1 RTE

Dec 7 2012 11:20:22.530.2-08:00 R1 RIP/7/DBG: 6: 12227: RIP 1: Sending response

on interface Ethernet0/0/0 from 10.0.12.1 to 255.255.255.255

Dec 7 2012 11:20:22.530.3-08:00 R1 RIP/7/DBG: 6: 12247: Packet: Version 1, Cmd response, Length 24

Dec 7 2012 11:20:22.530.4-08:00 R1 RIP/7/DBG: 6: 12296: Dest 10.0.1.0, Cost 1

Dec 7 2012 11:20:24.510.1-08:00 R1 RIP/7/DBG: 6: 12185: RIP 1: Receiving v1 response on Ethernet0/0/0 from 10.0.12.2 with 1 RTE

Dec 7 2012 11:20:24.510.2-08:00 R1 RIP/7/DBG: 6: 12236: RIP 1: Receive response

from 10.0.12.2 on Ethernet0/0/0

Dec 7 2012 11:20:24.510.3-08:00 R1 RIP/7/DBG: 6: 12247: Packet: Version 1, Cmd response, Length 24

Dec 7 2012 11:20:24.510.4-08:00 R1 RIP/7/DBG: 6: 12296: Dest 10.0.2.0, Cost 1

可以观察到R1从连接R2的E 0/0/0接口周期性发送、接收v1的Response更新报文，包括目的地、数据包大小以及COST值。

可以使用**undo debug rip**或者**undo debug all**命令关闭debug调试功能。

<R1>undo debugging rip 1

也可以使用带更多参数的命令查看某类型的调试信息，如**debug rip 1 event**查看路由器发出和收到的定期更新事件。其它参数可以使用“？”获取帮助。

<R1>debugging rip 1 event

Dec 7 2012 11:21:18.690.1-08:00 R1 RIP/7/DBG: 25: 4379: RIP 1: Periodic timer expired for interface Ethernet0/0/0 (10.0.12.1) and its added to periodic update

queue

Dec 7 2012 11:21:18.690.2-08:00 R1 RIP/7/DBG: 25: 4707: RIP 1: Interface Ethern

et0/0/0 (10.0.12.1) is deleted from the periodic update queue

<R1>undo debugging all

Info: All possible debugging has been turned off.

警告：开启过多的**debug**功能会耗费大量路由器资源，甚至可能导致宕机。请慎重使用开启批量**debug**功能的命令，如**debug all**。

## 使用RIPv2搭建网络

基于前面的配置，只需在RIP子视图模式下配置**version 2**即可。

[R1]rip

[R1-rip-1]version 2

[R2]rip

[R2-rip-1]version 2

配置完成后使用**display ip routing-table**命令查看各路由器路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 Direct 0 0 D 10.0.1.1 LoopBack0

10.0.1.1/32 Direct 0 0 D 127.0.0.1 LoopBack0

10.0.2.0/24 RIP 100 1 D 10.0.12.2 Ethernet0/0/0

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Ethernet0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 RIP 100 1 D 10.0.12.1 Ethernet0/0/0

10.0.2.0/24 Direct 0 0 D 10.0.2.2 LoopBack0

10.0.2.2/32 Direct 0 0 D 127.0.0.1 LoopBack0

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Ethernet0/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Ethernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，两台路由器已经通过RIP协议学习到了对方环回接口所在网段的路由条目。

配置完成后，使用**ping**命令检测R1与R2之间直连链路的IP连通性。

[R1]ping -a 10.0.1.1 10.0.2.2

PING 10.0.2.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.2.2: bytes=56 Sequence=1 ttl=255 time=50 ms

Reply from 10.0.2.2: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 10.0.2.2: bytes=56 Sequence=3 ttl=255 time=40 ms

Reply from 10.0.2.2: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 10.0.2.2: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.0.2.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/28/50 ms

可以观察到通信正常。

使用**debug**命令查看RIPv2协议定期更新情况。

<R1>debugging rip 1

Dec 7 2012 11:37:47.620.3-08:00 R1 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd

response, Length 24

Dec 7 2012 11:37:47.620.4-08:00 R1 RIP/7/DBG: 6: 12315: Dest 10.0.2.0/24, Nexth

op 0.0.0.0, Cost 1, Tag 0

Dec 7 2012 11:37:48.470.1-08:00 R1 RIP/7/DBG: 6: 12176: RIP 1: Sending v2 respo

nse on Ethernet0/0/0 from 10.0.12.1 with 2 RTEs

Dec 7 2012 11:37:48.470.2-08:00 R1 RIP/7/DBG: 6: 12227: RIP 1: Sending response

on interface Ethernet0/0/0 from 10.0.12.1 to 224.0.0.9

<R1>undo debugging rip 1

与RIPv1中使用**debug**命令所查看的信息进行对比，可以明显区分出RIPv1和RIPv2的不同。

RIPv2的路由信息中携带了子网掩码。

RIPv2的路由信息中携带了下一跳地址，标识一个比通告路由器的地址更好的下一跳地址。换句话说，它指出的地址，其度量值（跳数）比在同一个子网上的通告路由器更靠近目的地。如果这个字段设置为全0（0.0.0.0），说明通告路由器的地址是最优的下一跳地址。

RIPv2默认采用组播方式发送报文，地址为224.0.0.9。

7.2 配置RIPv2的认证

# 原理概述

配置协议的认证可以降低设备接受非法路由选择更新消息的可能性，也可称为验证。非法的更新消息可能来自试图破坏网络的攻击者，或试图通过欺骗路由器发送数据到错误的目的地址的方法来捕获数据包。RIPv2协议能够通过更新消息所包含的口令来验证某个路由选择消息源的合法性，有简单和MD5密文两种验证方式。

简单验证是指在认证的消息当中所携带的认证口令是以明文传输的，可以通过抓包软件抓取到数据包中的密码。

MD5密文验证是一种单向消息摘要（message digest）算法或安全散列函数（secure hash function），由RSA Date Security, Inc提出。有时MD5也被作为一个加密校验和（cryptographic checksum），MD5算法是通过一个随意长度的明文消息(例如，一个RIPv2的更新消息)和口令计算出一个128位的hash值的。hash值类似“指纹”，这个“指纹”随同消息一起传送，拥有相同口令的接收者会计算它自己的hash值，如果消息的内容没有被更改，接收者的hash值应该和消息中发送者的hash值相匹配。

# 实验目的

* + 理解配置RIPv2认证的场景和意义
  + 掌握配置RIPv2简单验证的方法
  + 掌握测试RIPv2简单验证的配置结果的方法
  + 掌握配置RIPv2 MD5密文验证的方法
  + 掌握测试RIPv2 MD5密文验证配置结果的方法

# 实验内容

本实验模拟企业网络场景，某公司有两台路由器R1与R2，各自连接着一台主机，并且R1和R2之间配置RIPv2协议学习路由条目。R3模拟作为网络中的攻击者，窃取R1与R2间的路由信息，并发布了一些虚假路由，使R1和R2的相关路由的选路指向了R3，形成了路由欺骗。为了避免遭受攻击，提高网络安全性，网络管理员将通过配置RIPv2认证来实现。

# 实验拓扑

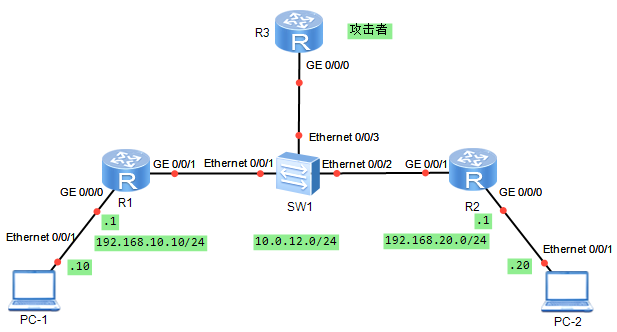


图7-2 配置RIPv2的认证拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 192.168.10.10 | 255.255.255.0 | 192.168.10.1 |
| R1(AR2220) | GE 0/0/0 | 192.168.10.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.12.1 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/1 | 10.0.12.2 | 255.255.255.0 | N/A |
| GE 0/0/0 | 192.168.20.1 | 255.255.255.0 | N/A |
| PC-2 | Ethernet 0/0/1 | 192.168.20.20 | 255.255.255.0 | 192.168.20.1 |
| R3(AR2220) | GE 0/0/0 | 10.0.12.3 | 255.255.255.0 | N/A |
| Loopback 0 | 192.168.10.1 | 255.255.255.0 | N/A |
| Loopback 1 | 192.168.20.1 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。其中，R3上的两个环回接口先不配置IP地址。

[R1]ping -c 1 10.0.12.2

PING 10.0.12.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms

--- 10.0.12.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 50/50/50 ms

[R1]ping -c 1 192.168.10.10

PING 192.168.10.10: 56 data bytes, press CTRL\_C to break

Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=128 time=30 ms

--- 192.168.10.10 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/30/30 ms

其余直连网段的连通性测试省略。

## 搭建RIP网络

配置公司路由器R1和R2的RIPv2协议，并添加需要通告的网段。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]network 192.168.10.0

[R1-rip-1]network 10.0.0.0

[R2]rip 1

[R2-rip-1]version 2

[R2-rip-1]network 192.168.20.0

[R2-rip-1]network 10.0.0.0

配置完成后，检查R1与R2的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.20.0/24 RIP 100 1 D 10.0.12.2 GigabitEthernet0/0/1

[R2-rip-1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 RIP 100 1 D 10.0.12.1 GigabitEthernet0/0/1

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/0

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

可以观察到，此时双方已经正常的获得了RIP路由条目。

## 模拟网络攻击

配置路由器R3作为攻击者，接入公司网络，在基本配置中已经将接口GE 0/0/0地址配置为10.0.12.3，与该公司路由器在同一网段，并配置RIPv2协议，通告该网段，配置完成后查看R3的路由表。

[R3]rip 1

[R3-rip-1]version 2

[R3-rip-1]network 10.0.0.0

[R3-rip-1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 5 Routes : 5

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 RIP 100 1 D 10.0.12.1 GigabitEthernet0/0/0

192.168.20.0/24 RIP 100 1 D 10.0.12.2 GigabitEthernet0/0/0

发现R3已经非法获取了R1和R2上用户终端所在的两个网段的路由信息。此时R3就可以向两个网段发送大量的ping包，导致网络链路拥塞，形成攻击。

下面是R3模拟攻击演示，发送十万个ping包给PC-1，导致攻击发生。可以按Ctrl+C键来终止该操作。

[R3]ping -c 100000 192.168.10.10

PING 192.168.20.1: 56 data bytes, press CTRL\_C to break

Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=255 time=480 ms

Reply from 192.168.10.10: bytes=56 Sequence=2 ttl=255 time=170 ms

Reply from 192.168.10.10: bytes=56 Sequence=3 ttl=255 time=130 ms

Reply from 192.168.10.10: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 192.168.10.10: bytes=56 Sequence=5 ttl=255 time=70 ms

Reply from 192.168.10.10: bytes=56 Sequence=6 ttl=255 time=60 ms

完成上述模拟后，在R3上分别配置两个用于欺骗的环回接口，地址分别为192.168.10.10/24和192.168.20.20/24，即与公司网络中两个用户的地址相同，并且在RIP协议中通告这两个欺骗的网段。

[R3]interface loopback 0

[R3-LoopBack0]ip address 192.168.10.10 255.255.255.0

[R3]interface loopback 1

[R3-LoopBack0]ip address 192.168.20.1 255.255.255.0

[R3]rip 1

[R3-rip-1]version 2

[R3-rip-1]network 192.168.10.0

[R3-rip-1]network 192.168.20.0

配置完成后，查看R1与R2的路由表。

[R1]display ip routing-table

Destination/Mask Proto Pre Cost Flags NextHop Interface

……

192.168.20.0/24 RIP 100 1 D 10.0.12.2 GigabitEthernet0/0/1

RIP 100 1 D 10.0.12.3 GigabitEthernet0/0/1

[R2]display ip routing-table

Destination/Mask Proto Pre Cost Flags NextHop Interface

……

192.168.10.0/24 RIP 100 1 D 10.0.12.1 GigabitEthernet0/0/1

RIP 100 1 D 10.0.12.3 GigabitEthernet0/0/1

可以观察到R3发过来的路由更新。因为R2和R3发送RIP更新的cost都是1跳，所以在R1的路由表中，目的为192.168.20.0网段形成了两条等价负载均衡的路径。下一跳分别是R2和R3。这样会导致去往192.168.20.0网段的数据包有部分转发给了欺骗路由器R3，R2的路由表变化和R1同理。

## 配置RIPv2简单验证

为了提升网络安全性，避免发生上述的攻击和路由欺骗，网络管理员在R1和R2上配置RIP的简单验证实现对网络的保护。

在路由器R1和R2的GE 0/0/1接口配置认证，使用简单验证方式，密码为huawei。注意，两点的密码必须保持一致，不一致将会导致认证失败，从而使得RIP协议无法正常运行。

[R1]interface GigabitEthernet0/0/1

[R1-GigabitEthernet0/0/1]rip authentication-mode simple huawei

[R2]interface GigabitEthernet0/0/1

[R2- GigabitEthernet0/0/1]rip authentication-mode simple huawei

配置完成后，等待一段时间，再次查看R1和R2的路由表。

[R1]display ip routing-table

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.20.0/24 RIP 100 1 D 10.0.12.2 GigabitEthernet0/0/1

[R2]display ip routing-table

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 RIP 100 1 D 10.0.12.1 GigabitEthernet0/0/1

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/0

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

可以观察到现在R1与R2的路由表恢复正常，R3发送的欺骗路由在路由表中消失。原因是R1和R2配置了RIP 认证，就要求在RIP更新报文中包含认证密码，如果密码错误或者不存在，将认为该路由非法并丢弃。

在路由器R1的GE 0/0/1接口上抓包。

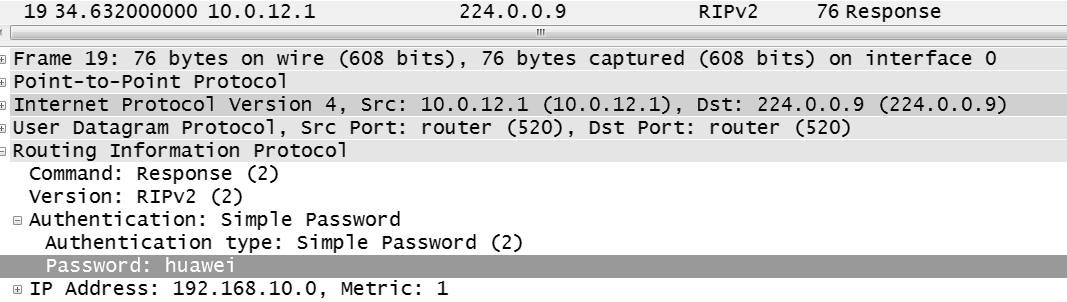


图7-3

可以观察到，此时R1与R2间发送的RIP报文中含authentication字段，并且密码是明文的huawei。

## 配置RIPv2 MD5密文验证

在上一步骤中，在R1和R2上配置了简单验证方式的认证后，成功的抵御了R3的路由欺骗和攻击，且主机PC-1与PC-2可以正常通信。但是通过抓包能够发现，简单验证方式下的认证安全性非常差，攻击者虽然无法直接攻击网络，但是可以抓取RIP协议数据包就能获得明文密码，因此建议使用MD5密文验证方式进行RIPv2的认证。

在R1和R2的GE 0/0/1接口上删除上一步骤中的简单验证配置，选择使用MD5密文验证方式配置。其中可以选择MD5密文验证方式的报文格式，配置usual参数表示使用通用报文格式，使用nonstanard参数表示使用非标准报文格式（IETF标准），但是必须保证两端的报文格式一致，这里选用通用标准格式。

[R1-GigabitEthernet0/0/1]undo rip authentication-mode

[R1-GigabitEthernet0/0/1]rip authentication-mode md5 usual huawei

[R2-GigabitEthernet0/0/1]undo rip authentication-mode

[R2-GigabitEthernet0/0/1]rip authentication-mode md5 usual huawei

配置完成后，查看R1和R2路由表。

[R1]display ip routing-table

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.20.0/24 RIP 100 1 D 10.0.12.2 GigabitEthernet0/0/1

[R2]display ip routing-table

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 RIP 100 1 D 10.0.12.1 GigabitEthernet0/0/1

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/0

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

可以观察到R1与R2的路由表正常，R3发送的欺骗路由在路由表中消失，与配置简单验证的效果一样。

继续抓取R1的GE 0/0/1接口上的报文。

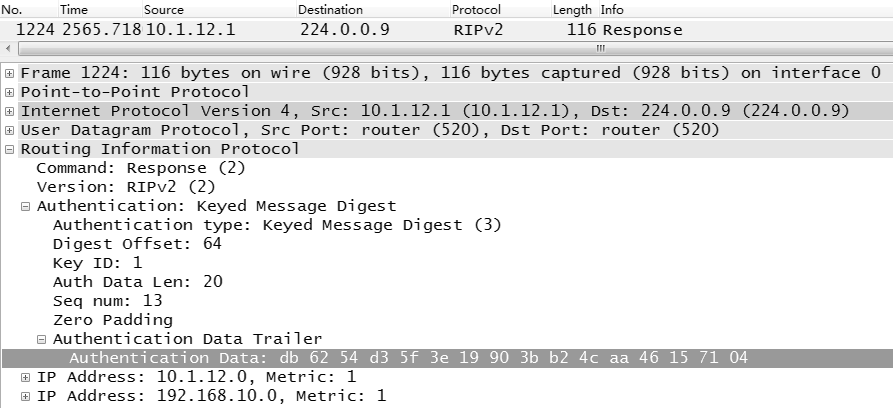


图7-4

抓包发现已经无法看到配置的认证密码，而看到的是一个128位的hash值，这是一种单向的散列值，难以破解，这样就能够进一步地保证网络的安全性。

# 思考

在本实验中，R1和R2上配置了认证，R3没有配置认证，根据分析，R1和R2不会再接收R3发送的不包含认证信息的RIP更新，那R3是否会接收R1和R2发送过来的带有认证信息的RIP更新呢？为什么？

7.3 RIP路由协议的汇总

# 原理概述

当网络中路由器的路由条目非常多时，可以通过路由汇总（又称路由汇聚或路由聚合）来减少路由条目数，加快路由收敛时间和增强网络稳定性。路由汇总的原理是，同一个自然网段内的不同子网的路由在向外（其它网段）发送时聚合成一个网段的路由发送。由于汇总后路由器将不会感知被汇总[子网](http://baike.baidu.com/view/65511.htm)有关的变化，从而提高了网络稳定性，减少了不必要的路由器更新。

RIPv1是有类别路由协议，它的协议报文中没有携带掩码信息，只能识别A、B、C类这样的自然网段的路由，因此RIPv1无法支持路由聚合，也不支持不连续子网，所有路由会被自动汇总为有类路由。

RIPv2是一种无分类路由协议，报文中携带掩码信息，支持手动路由汇总和自动路由汇总两种方式。

基于RIP进程的有类自动汇总。比如对于10.1.1.0/24(metric=2）和10.1.2.0/24(metric=3)这两条路由，聚合成自然网段路由10.0.0.0/8(metric=2）。自动汇总是按类聚合的，在华为设备上自动汇总是默认关闭的，可手动更改配置使自动汇总生效。

基于接口的手动汇总。用户可以指定聚合路由。比如，对于10.1.1.0/24(metric=2)和10.1.2.0/24(metric=3)这两条路由，可以在此接口上配置聚合路由10.1.0.0/16(metric=2)。

# 实验目的

* + 理解RIP路由协议汇总的应用场景
  + 理解RIPv1和RIPv2的自动汇总
  + 掌握配置和测试RIPv2手动汇总的方法

# 实验内容

在由三台路由器所组成的简单网络中， R3连接着多个网段，通过Loopback口来模拟多个网段，通过实验实现RIPv1自动汇总，RIPv2自动汇总以及RIPv2手工汇总。

# 实验拓扑

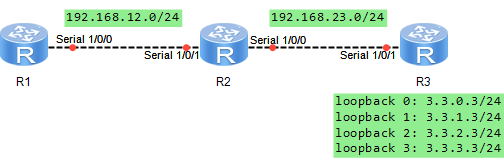


图7-5 RIP路由协议的汇总拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | Serial 1/0/0 | 192.168.12.1 | 255.255.255.0 | N/A |
| R2(AR1220) | Serial 1/0/1 | 192.168.12.2 | 255.255.255.0 | N/A |
| Serial 1/0/0 | 192.168.23.2 | 255.255.255.0 | N/A |
| R3(AR1220) | Serial 1/0/1 | 192.168.23.3 | 255.255.255.0 | N/A |
| Loopback 0 | 3.3.0.3 | 255.255.255.0 | N/A |
| Loopback 1 | 3.3.1.3 | 255.255.255.0 | N/A |
| Loopback 2 | 3.3.2.3 | 255.255.255.0 | N/A |
| Loopback 3 | 3.3.3.3 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping -c 1 192.168.12.2

PING 192.168.12.2: 56 data bytes, press CTRL\_C to break

Reply from 192.168.12.2: bytes=56 Sequence=1 ttl=255 time=30 ms

--- 192.168.12.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/30/30 ms

其余直连网段的连通性测试省略。

## 配置RIPv1协议

在路由器R1，R2，R3上配置RIPv1协议，通告相应网段。

[R1]rip 1

[R1-rip-1]network 192.168.12.0

[R2]rip 1

[R2-rip-1]network 192.168.12.0

[R2-rip-1]network 192.168.23.0

[R3]rip 1

[R3-rip-1]network 192.168.23.0

[R3-rip-1]network 3.0.0.0

配置完成后，查看R1与R2的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.0.0.0/8 RIP 100 2 D 192.168.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.12.0/24 Direct 0 0 D 192.168.12.1 Serial1/0/0

192.168.12.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

192.168.12.2/32 Direct 0 0 D 192.168.12.2 Serial1/0/0

192.168.23.0/24 RIP 100 1 D 192.168.12.2 Serial1/0/0

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.0.0.0/8 RIP 100 1 D 192.168.23.3 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.12.0/24 Direct 0 0 D 192.168.12.2 Serial1/0/1

192.168.12.1/32 Direct 0 0 D 192.168.12.1 Serial1/0/1

192.168.12.2/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

192.168.23.0/24 Direct 0 0 D 192.168.23.2 Serial1/0/0

192.168.23.2/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

192.168.23.3/32 Direct 0 0 D 192.168.23.3 Serial1/0/0

可以观察到R3发送过来的汇总路由条目3.0.0.0/8，没有任何明细路由条目。

在R3的S 1/0/1接口上抓包。

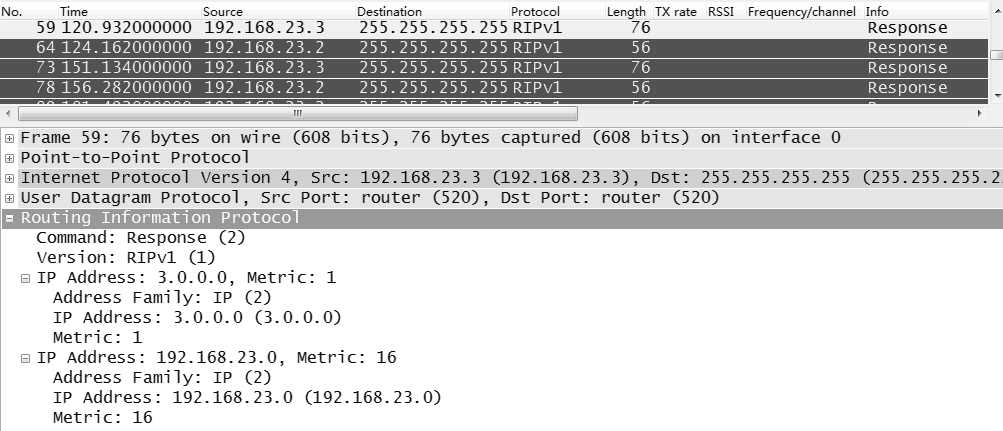


图7-6

可以观察到，RIPv1的协议报文中没有携带掩码信息，只有相应的网络号，以及Metric值，即RIPv1 只发布汇总后的有类路由。RIPv1默认开启自动汇总，且无法关闭，也不支持手动汇总。可以使用命令display default-parameter rip查看RIP缺省配置信息。

<R3>display default-parameter rip

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

Protocol Level Default Configurations

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

RIP version : 1

Preference : 100

Checkzero : Enabled

Default-cost : 0

Auto Summary : Enabled

Host-route : Enabled

……

可以观察到默认开启了自动汇总。

## 配置RIPv2自动汇总

在路由器R1，R2，R3上配置**version 2**命令，运行RIPv2协议。

[R1]rip 1

[R1-rip-1]version 2

[R2]rip 1

[R2-rip-1]version 2

[R3]rip 1

[R3-rip-1]version 2

配置完成后，在R3的S 1/0/1接口上抓包。

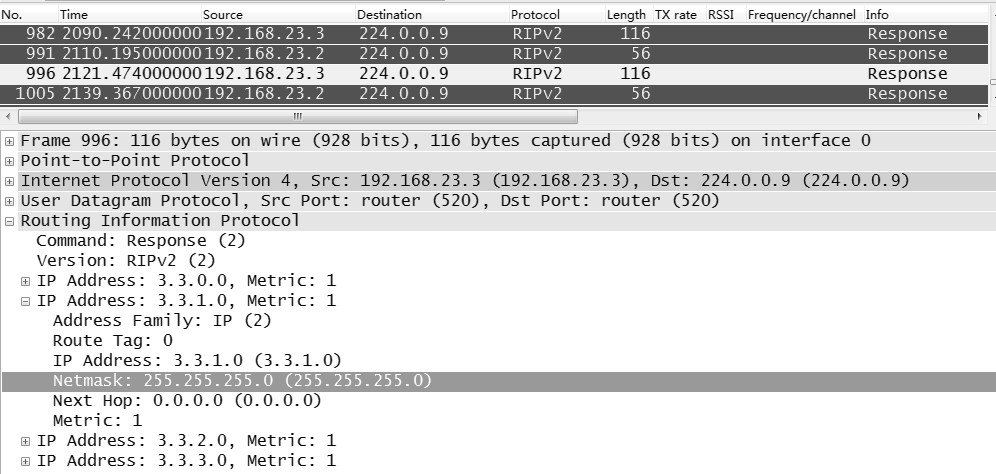


图7-7

可以观察到，RIPv2报文中携带了掩码信息。RIPv2支持自动汇总，默认是开启的，并且可以关闭。

查看R1与R2的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.3.0.0/24 RIP 100 2 D 192.168.12.2 Serial1/0/0

3.3.1.0/24 RIP 100 2 D 192.168.12.2 Serial1/0/0

3.3.2.0/24 RIP 100 2 D 192.168.12.2 Serial1/0/0

3.3.3.0/24 RIP 100 2 D 192.168.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

……

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 12 Routes : 12

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.3.0.0/24 RIP 100 1 D 192.168.23.3 Serial1/0/0

3.3.1.0/24 RIP 100 1 D 192.168.23.3 Serial1/0/0

3.3.2.0/24 RIP 100 1 D 192.168.23.3 Serial1/0/0

3.3.3.0/24 RIP 100 1 D 192.168.23.3 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

……

可以观察到，接收到的路由条目是具体的明细路由条目，而没有汇总路由，即此时RIPv2默认自动汇总并没有生效。

这是因为在华为设备上，以太网接口和串口都默认启用了水平分割功能。为了防止环路和不连续子网问题的产生，在启用了水平分割或毒性逆转的接口上，RIPv2的默认自动汇总就会失效，所以从R3通告过来的都是具体的明细路由条目。

要使RIPv2的默认自动汇总生效，有两种方法。

第一种方法，使用**summary always**命令。配置该命令后，不论水平分割是否启用，RIPv2的自动汇总都生效。

[R3]rip

[R3-rip-1]version 2

[R3-rip-1]summary always

第二种方法，关闭相应接口下的水平分割功能。

[R3]interface Serial 1/0/1

[R3-Serial1/0/1]undo rip split-horizon

使用以上的任一种方法后，查看R1与R2的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.0.0.0/8 RIP 100 2 D 192.168.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

……

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.0.0.0/8 RIP 100 1 D 192.168.23.3 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

……

可以观察到，此时RIPv2的自动汇总生效了。

## 配置RIPv2手动汇总

删除上一步骤中的使得RIPv2自动汇总功能生效的配置，具体视使用哪一种方式来定，这里省略。

在R3上使用命令**rip summary-address**配置手动汇总，配合需要汇总的本地网络IP地址为3.3.0.0，网络掩码为255.255.252.0。

[R3]interface Serial 1/0/1

[R3-Serial1/0/1]rip summary-address 3.3.0.0 255.255.252.0

配置完成后，查看R1与R2的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.3.0.0/22 RIP 100 2 D 192.168.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.12.0/24 Direct 0 0 D 192.168.12.1 Serial1/0/0

192.168.12.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

192.168.12.2/32 Direct 0 0 D 192.168.12.2 Serial1/0/0

192.168.23.0/24 RIP 100 1 D 192.168.12.2 Serial1/0/0

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.3.0.0/22 RIP 100 1 D 192.168.23.3 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.12.0/24 Direct 0 0 D 192.168.12.2 Serial1/0/1

192.168.12.1/32 Direct 0 0 D 192.168.12.1 Serial1/0/1

192.168.12.2/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

192.168.23.0/24 Direct 0 0 D 192.168.23.2 Serial1/0/0

192.168.23.2/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

192.168.23.3/32 Direct 0 0 D 192.168.23.3 Serial1/0/0

可以观察到，R1和R2上已经接收到了该汇总路由条目，且没有任何明细路由条目。

# 思考

华为设备默认开启了RIPv2的自动汇总，如果没有默认开启接口下的水平分割，即自动汇总生效的情况下，可能会导致出现环路以及不连续子网等问题。请设计一个相关场景，模拟在RIPv2开启了自动汇总且关闭了水平分割的情况下，导致路由环路或不连续子网问题的出现。

7.4 配置RIP的版本兼容、定时器及协议优先级

# 原理概述

RIP在IPv4中有v1和v2两个版本。在配置RIP时，如果不指定版本的情况下，接口默认情况下能接收v1和v2的报文，但只能发送v1的报文；在指定版本的情况下，RIPv1只能接收和发送v1的报文，RIPv2只能接收和发送v2 的报文。

RIP的定时器有三种：更新计时器，默认每30秒钟发送一次更新；超时计时器，默认时间180秒钟，如果在超时计时器内没有收到邻居发来的更新报文，则把该路由的度量值设置为16，并启动垃圾收集定时器；垃圾收集定时器，默认时间120秒钟，如果启动了该计时器，那么120秒钟超时以后，路由表中会删除该路由表项。

RIP默认协议优先级为100，可以手动修改。

# 实验内容

本实验中采用简单的场景介绍RIP各版本间的区别，及如何实现相互间的兼容，RIP三种定时器的作用及修改方法，RIP优先级的作用及修改方法。

# 实验目的

* + 掌握配置RIP版本的方法
  + 理解RIPv1和RIPv2的相互兼容性
  + 掌握RIP的三种定时器的配置
  + 掌握RIP的协议优先级的配置

# 实验拓扑

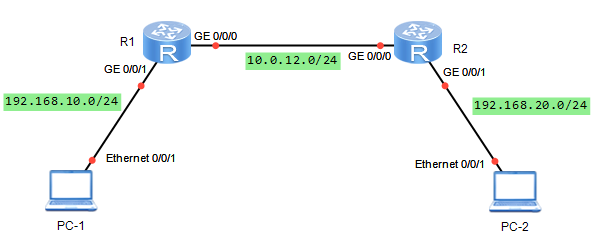


图7-8 配置RIP的版本兼容、定时器及协议优先级拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 192.168.10.1 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 192.168.20.1 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 192.168.10.10 | 255.255.255.0 | 192.168.10.1 |
| PC-2 | Ethernet 0/0/1 | 192.168.20.10 | 255.255.255.0 | 192.168.20.1 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping -c 1 10.0.12.2

PING 10.0.12.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms

--- 10.0.12.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 50/50/50 ms

其余直连网段的连通性测试省略。

## 配置RIP协议的版本兼容

按照拓扑图，分别在R1和R2上配置RIP协议，通告相应网段。但是在R1上，不指定RIP的版本，在R2上指定使用版本v2。

[R1]rip

[R1-rip-1]network 10.0.0.0

[R1-rip-1]network 192.168.10.0

[R2]rip

[R2-rip-1]network 10.0.0.0

[R2-rip-1]network 192.168.20.0

[R2-rip-1]version 2

配置完成后，使用**display ip routing-table**命令查看R1和R2的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 GigabitEthernet0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/1

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.20.0/24 RIP 100 1 D 10.0.12.2 GigabitEthernet0/0/0

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0 127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/1

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

可以观察到，在R1的路由表中存在PC-2所在网段的路由条目，在R2的路由表中没有发现PC-2所在网段的路由条目。

在R1的GE 0/0/0接口上抓取R1发送给R2，和从R2接收到的RIP报文。

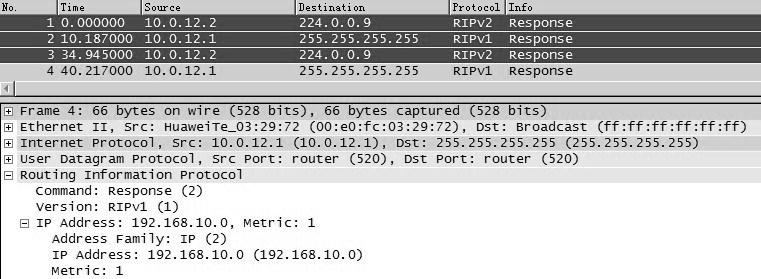


图7-9

可以观察到R1采用版本1，即广播方式来发送更新；而R2采用版本2，即组播方式发送更新。验证了R1在RIP协议进程中没有明确指定版本配置时，缺省是可以处理接收版本1和版本2的报文，但仅发送版本1的报文，而R2因在RIP协议进程中明确配置了版本 2，仅接收和发送版本2的报文。

因此，由于R1发送的是RIPv1报文，而R2不能正确处理接收，所以R2的路由表中没有PC-1所在网段的路由条目。而R2发送的RIPv2报文能够被R1 处理接收，所以在R1的路由表中存在PC-2所在网段的路由条目。

现在为了能够使得R2也能接收PC-1所在网段的路由条目，在R1上设置接口的RIP版本，使R1能够以广播发送RIPv2报文。

[R1]interface GigabitEthernet0/0/0

[R1- GigabitEthernet0/0/0]rip version 2 broadcast

配置完成后，查看R2的路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 RIP 100 1 D 10.0.12.1 GigabitEthernet0/0/0

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/1

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

发现路由表中已经存在R1发送过来的路由条目。同样也可以使用命令**rip version 2 mutlicast**，即使R1能够以组播方式发送RIPv2报文，效果一样，不再验证。

在配置RIP协议时建议路由器之间配置相同RIP版本，即所有路由器都配置RIPv1或者都配置RIP v2。以避免可能由于错误配置而导致RIP协议无法正常工作。

## 配置RIP的定时器

配置完RIP版本兼容后，再次在R1的GE 0/0/0接口上通过抓包分析R1和R2更新报文的发送情况。

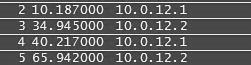


图7-10

可以观察到R1在10秒时发送了一次更新，R2在34秒发送了一次更新，R1在40秒时发送了下一次更新，R2在65秒发送了下一次更新。即默认情况下RIP协议会每隔30秒左右发送一次路由更新。

路由更新的有效期为超时定时器定义的时间180秒。即当在180秒内没有收到新的路由更新，则宣布该路由不可达，并从路由表中清除掉该路由条目。

为了验证效果，在R1的GE 0/0/0接口上配置停止发送RIP路由更新。

[R1]interface GigabitEthernet0/0/0

[R1- GigabitEthernet0/0/0]undo rip output

配置完成后，此时R1的G 0/0/0接口上已经无法发送任何RIP路由更新，此时立刻查看R2的路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 RIP 100 1 D 10.0.12.1 GigabitEthernet0/0/0

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/1

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

可以观察到，从R1接收到的路由条目依然存在，原因是RIP超时定时器没有到期，该路由条目依然被保存在路由表中。

使用命令**display rip database**检查R2的RIP发布数据库中的所有激活路由。

[R2]display rip 1 database

Advertisement State : [A] - Advertised

[I] - Not Advertised/Withdraw

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

10.0.0.0/8, cost 0, ClassfulSumm

10.0.12.0/24, cost 0, [A], Rip-interface

192.168.10.0/24, cost 1, ClassfulSumm

192.168.10.0/24, cost 1, [A], nexthop 10.0.12.1

192.168.20.0/24, cost 0, ClassfulSumm

192.168.20.0/24, cost 0, [A], Rip-interface

看到路由条目也没有发生变化，状态仍然为[A]，即仍被通告。在等待超时计时器到期定义的180秒钟以后。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 GigabitEthernet0/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.20.0/24 Direct 0 0 D 192.168.20.1 GigabitEthernet0/0/1

192.168.20.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

可以观察到，R2的路由表中已经无法看到R1发送过来的路由条目。原因是超时定时器已经到期，该路由条目被定义为失效，已经从路由表中清除了。

同时再次检查R2的路由表和发布数据库。

[R2]display rip 1 database

Advertisement State : [A] - Advertised

[I] - Not Advertised/Withdraw

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

10.0.0.0/8, cost 0, ClassfulSumm

10.0.12.0/24, cost 0, [A], Rip-interface

192.168.10.0/24, cost 16, ClassfulSumm

192.168.10.0/24, cost 16, [I], nexthop 10.0.12.1

192.168.20.0/24, cost 0, ClassfulSumm

192.168.20.0/24, cost 0, [A], Rip-interface

发现在数据库中可以看到该路由条目，但是该路由条目已经被标记为16跳，即不可达，并且状态标记为[I]，该路由将不能被通告出去。虽然该条目已失效，但是仍然存在于发布数据库中的原因是RIP垃圾收集定时器启动，且还没有到期，暂时不能从数据库中清除。

如果在默认120秒仍然没有收到更新报文，垃圾收集定时器超时后将删除该表项。经过120秒后再查看R2上的发布数据库。

[R2]display rip 1 database

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

Advertisement State : [A] - Advertised

[I] - Not Advertised/Withdraw

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

10.0.0.0/8, cost 0, ClassfulSumm

10.0.12.0/24, cost 0, [A], Rip-interface

192.168.20.0/24, cost 0, ClassfulSumm

192.168.20.0/24, cost 0, [A], Rip-interface

可以观察到，此时已经不存在任何R1发送过来的路由条目。

可以通过命令**timers rip**改变这几个定时器的默认值，来影响RIP的收敛速度。现将R1的更新报文的时间间隔修改为20秒，超时计时器的超时时间修改为120秒，垃圾收集计时器的超时时间为60秒。

[R1]rip 1

[R1-rip-1]timers rip 20 120 60

配置完成后，查看RIP的协议信息。

[R1]display rip

Public VPN-instance

RIP process : 1

RIP version : 2

Preference : 100

Checkzero : Enabled

Default-cost : 0

Summary : Enabled

Host-route : Enabled

Maximum number of balanced paths : 32

Update time : 20 sec Age time : 120 sec

Garbage-collect time : 60 sec

Graceful restart : Disabled

BFD : Disabled

……

可以观察到，RIP定时器的值在更改后立即生效。

如果三个定时器值设置不当，会引起网络不稳定。例如，如果更新时间大于失效时间，那么在更新时间内，可能在接收到路由更新之前，本地的路由条目已经失效了。定时器值的调整应考虑网络的规模和性能，并在所有运行的RIP路由器上进行统一配置。

## 配置RIP协议优先级

在实际网络中，去往相同目的网段的路由信息可以通过不同的路由协议获取到，比如同时通过静态路由和RIP协议获取到，此时就会先比较二者的协议优先级，通过具有较高优先级的路由协议所获取的路由信息将被优选放入至路由表中。

查看R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 GigabitEthernet0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/1

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.20.0/24 RIP 100 1 D 10.0.12.2 GigabitEthernet0/0/0

可以观察到RIP的路由优先级默认值为100。可以使用**preference**命令将R1的路由优先级调整为90，并查看R1的路由表。

[R1]rip

[R1-rip-1]preference 90

[R1-rip-1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 GigabitEthernet0/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 GigabitEthernet0/0/1

192.168.10.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.20.0/24 RIP 90 1 D 10.0.12.2 GigabitEthernet0/0/0

可以观察到此时已经完成了相应修改。注意优先级的数值越小，代表优先级越高。

# 思考

在此实验中，如果在R1上配置一条去往192.168.20.0网段的静态路由，再把RIP优先级修改为60，那么在R1的IP路由表中该网段路由来自RIP还是静态路由？为什么？

7.5 配置RIP抑制接口及单播更新

# 原理概述

RIP支持抑制接口的配置，即配置后禁止接口发送更新报文，但此接口所在网段的路由可以发布出去。可通过两种方法来实现，执行命令[silent-interface](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/silent-interface_rip.html)或在接口下配置[undo rip output](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_output.html)使其只接收报文，但不能发送RIP报文。[silent-interface](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/silent-interface_rip.html)的优先级大于在接口下配置的[undo rip output](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_output.html)，默认情况下为不抑制状态。还可以在接口下配置命令[undo rip input](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_input.html)，禁止接口接收RIP更新报文，这也是预防路由环路的一种方式。

单播更新是指RIP使用单播发送RIP报文。在默认情况下，RIP每隔30秒以广播或组播方式交换整个路由表的信息，这将耗费大量网络带宽，特别是在广域网中，可能出现严重性能问题。为了解决因RIP的广播报文而产生的网络性能问题，可以使用单播更新的方式来交换路由信息。当使用[silent-interface](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/silent-interface_rip.html)命令配置抑制接口后，再指定单播更新的目的地址后，单播更新有效；如果在接口下使用[undo rip output](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_output.html)命令来配置抑制接口，即使再指定单播更新的目的地址也是无法发送更新的路由条目的。

RIPv1和RIPv2对于抑制接口和单播更新的特性支持情况相同。

# 实验目的

* + 掌握RIP抑制接口的配置
  + 理解抑制接口的原理及应用场景
  + 掌握RIP中单播更新的配置
  + 理解单播更新的原理及应用场景

# 实验内容

本实验模拟企业网络场景，R1为该公司出口网关路由器，连接运营商网络；R2为公司IT部门路由器，通过交换机S1与网关相连；人事部员工直接通过交换机S1接入公司网络；R3为公司财务部门路由器，同样通过S1与网关相连。所有路由器运行路由协议RIP实现网络互通。由于交换机S1直连了大量PC用户，如果R1继续以广播（RIPv1）或组播（RIPv2）的方式发送更新的路由给R2和R3，处于同一广播网络中的S1下连接的PC也会收到这些对PC来说无用的更新，造成了带宽和资源的浪费。为了优化网络，现需在R1的GE 0/0/1口配置抑制接口来抑制广播或组播更新，为了使R2和R3能照常接收更新，还需要在R1上配置与R2、R3的单播更新；同时禁止其他部门访问财务部门，抑制R3的E 1/0/1接口，不发布任何RIP路由（单播更新也不行），仅可接收其他路由信息。

# 实验拓扑

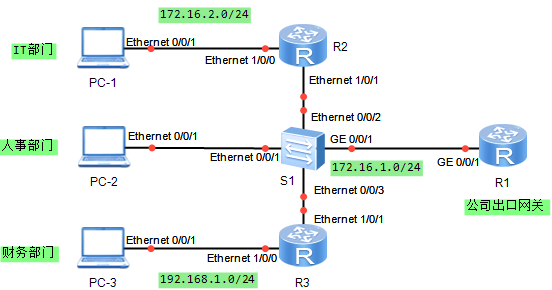


图7-11 配置RIP抑制接口及单播更新拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/1 | 172.16.1.254 | 255.255.255.0 | N/A |
| R2(AR2220) | Ethernet 1/0/1 | 172.16.1.100 | 255.255.255.0 | N/A |
| Ethernet 1/0/0 | 172.16.2.254 | 255.255.255.0 | N/A |
| R3(AR2220) | Ethernet 1/0/1 | 172.16.1.200 | 255.255.255.0 | N/A |
| Ethernet 1/0/0 | 192.168.1.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 172.16.2.1 | 255.255.255.0 | 172.16.2.254 |
| PC-2 | Ethernet 0/0/1 | 172.16.1.1 | 255.255.255.0 | 172.16.1.254 |
| PC-3 | Ethernet 0/0/1 | 192.168.1.1 | 255.255.255.0 | 192.168.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping -c 1 172.16.1.1

PING 172.16.1.1: 56 data bytes, press CTRL\_C to break

Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=128 time=50 ms

--- 172.16.1.1 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 50/50/50 ms

其余直连网段的连通性测试省略。

## 搭建基础的RIP网络

在公司各台路由器上都运行RIP路由协议，并通告相应网段。

[R1]rip 1

[R1-rip-1]network 172.16.0.0

[R2]rip 1

[R2-rip-1]network 172.16.0.0

[R3]rip 1

[R3-rip-1]network 172.16.0.0

[R3-rip-1]network 192.168.1.0

配置完成后，检查三台设备的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.254 GigabitEthernet0/0/1

172.16.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

172.16.2.0/24 RIP 100 1 D 172.16.1.100 GigabitEthernet0/0/1

192.168.1.0/24 RIP 100 1 D 172.16.1.200 GigabitEthernet0/0/1

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.100 Ethernet1/0/1

172.16.1.100/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

172.16.2.0/24 Direct 0 0 D 172.16.2.254 Ethernet1/0/0

172.16.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

192.168.1.0/24 RIP 100 1 D 172.16.1.200 Ethernet1/0/1

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.200 Ethernet1/0/1

172.16.1.200/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

172.16.2.0/24 RIP 100 1 D 172.16.1.100 Ethernet1/0/1

192.168.1.0/24 Direct 0 0 D 192.168.1.254 Ethernet1/0/0

192.168.1.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

可以观察到此时每台路由器上都已经拥有了所有网段的路由信息，连通性检查这里省略。

接下来网络管理员在PC-2的E 0/0/1接口上抓包。

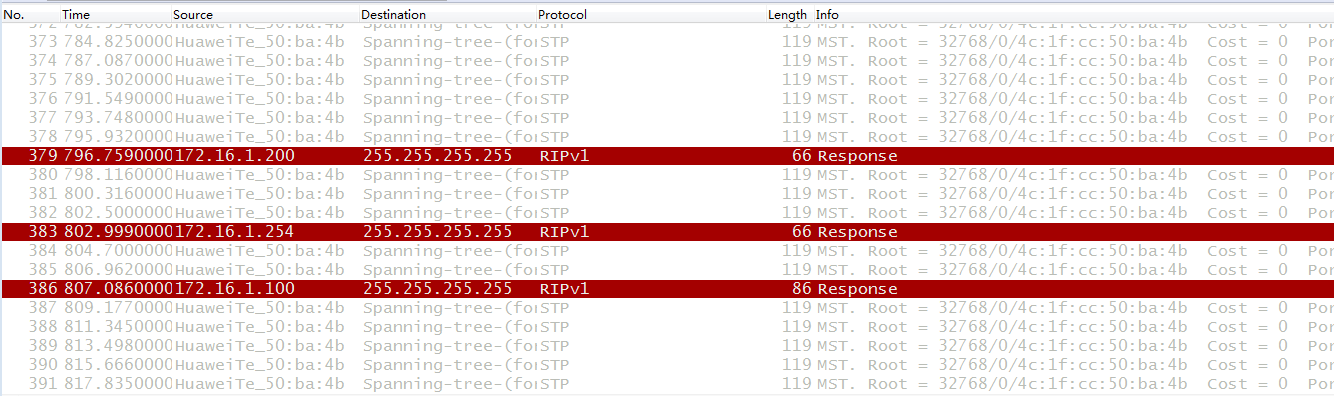


图7-12

可以观察到接收了许多对PC-2来说无用的RIP路由更新。

这是由于路由器R1、R2、R3以广播（RIPv1）或组播（RIPv2）的方式发送路由更新，而处于同一广播网中的S1下连接的PC也会收到这些对其而言无用的路由更新，造成了带宽和资源的浪费。

在PC-1的E 0/0/1接口下抓包。

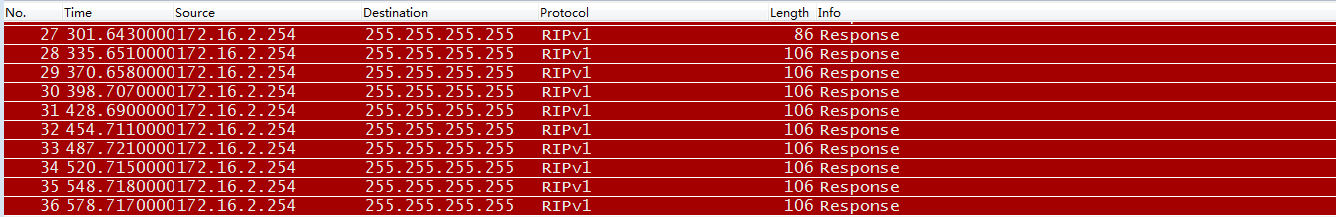


图7-13

同样可观察到PC-1上接收到许多对PC-1而言无用的RIP更新，这是由于在R2上172.16.2.0网段也被通告进了RIP协议中，即R2的E1/0/0接口运行在了RIP协议中，也会发送RIP路由信息。在PC-3上抓包可以看到同样的效果，原理和PC-1一样。

## 配置RIP抑制接口，优化公司网络

为了减少对带宽和资源的浪费，不让HR部门的PC，以及IT和财务部门的PC收到大量无关RIP报文，可以采用抑制接口来实现，使得该接口只接收RIP更新报文，而不发送更新报文。

在各路由器上使用命令**silent-interface**将相应接口配置成为抑制接口。

[R1]rip 1

[R1-rip-1]silent-interface GigabitEthernet 0/0/1

[R2]rip 1

[R2-rip-1]silent-interface Ethernet 1/0/1

[R2-rip-1]silent-interface Ethernet 1/0/0

[R3]rip 1

[R3-rip-1]silent-interface Ethernet 1/0/1

[R3-rip-1]silent-interface Ethernet 1/0/0

配置完成后，在R1上使用**display rip**命令查看相关配置信息。

[R1-rip-1]display rip 1

Public VPN-instance

RIP process : 1

……

BFD : Disabled

Silent-interfaces :

GigabitEthernet0/0/1

Default-route : Disabled

……

可以观察到，R1上配置已经成功，R2、R3的查看省略。

配置完成后，在PC-2的E 0/0/1接口上抓包,现象如下：

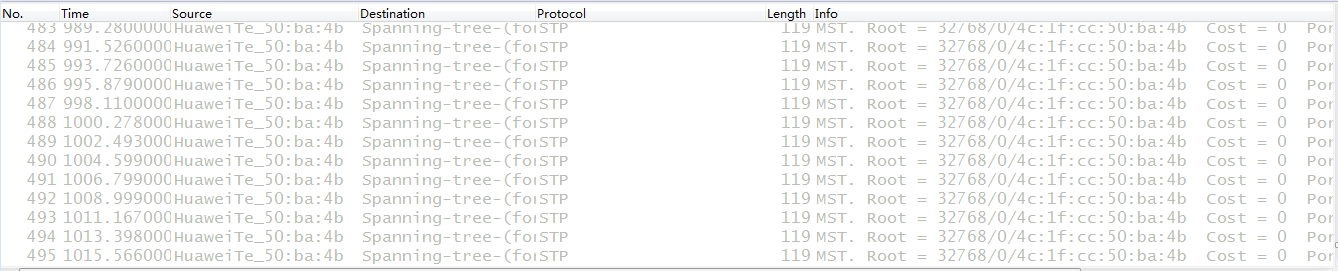


图7-14

可以观察到此时PC-2已接收不到RIP的路由更新

在PC-1和PC-3的E 0/0/1接口下抓包，效果也一样，此处省略。

## 配置RIP单播更新，恢复网络通信

在上一步骤中，网络管理员完成了对网络的优化，即使得所有的PC都不再接收与之无关的路由更新。而这时各部门员工却反映无法相互间访问，也无法访问外网。

查看R1的路由表。

[R1]display IP routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.254 GigabitEthernet0/0/1

172.16.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

172.16.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以发现此时R1获取不到其他网络的RIP路由信息，R2、R3路由表的查看省略，同样也接收不到RIP路由信息。

原因是在网络管理员将各路由器的相应接口配置成抑制接口后，接口将无法以广播或组播的方式发送RIP更新报文。

现在为了让RIP网络能够正常通信，可以通过增加RIP的单播更新配置来实现。通过该配置，RIP更新报文会以单播形式发送，而不采用正常的组播或广播的形式。

在R1上使用**peer**命令，后面跟上指定的邻居路由器IP地址，即R2和R3与R1相连的直连链路上的IP地址。

[R1]rip 1

[R1-rip-1]peer 172.16.1.100

[R1-rip-1]peer 172.16.1.200

配置完成后，使用命令**display rip**在R1上检查RIP协议的信息。

[R1-rip-1]display rip 1

Public VPN-instance

RIP process : 1

……

172.16.0.0

Configured peers :

172.16.1.200 172.16.1.100

Number of routes in database : 4

……

可以观察到邻居IP地址已经配置成功。

同样在R2，R3上使用**peer**命令，配置单播更新。

[R2]rip 1

[R2-rip-1]peer 172.16.1.254

[R2-rip-1]peer 172.16.1.200

[R3]rip 1

[R3-rip-1]peer 172.16.1.100

[R3-rip-1]peer 172.16.1.254

配置完成后，再次查看R1，R2，R3上的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.254 GigabitEthernet0/0/1

172.16.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

172.16.2.0/24 RIP 100 1 D 172.16.1.100 GigabitEthernet0/0/1

192.168.1.0/24 RIP 100 1 D 172.16.1.200 GigabitEthernet0/0/1

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.100 Ethernet1/0/1

172.16.1.100/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

172.16.2.0/24 Direct 0 0 D 172.16.2.254 Ethernet1/0/0

172.16.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

192.168.1.0/24 RIP 100 1 D 172.16.1.200 Ethernet1/0/1

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.200 Ethernet1/0/1

172.16.1.200/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

172.16.2.0/24 RIP 100 1 D 172.16.1.100 Ethernet1/0/1

192.168.1.0/24 Direct 0 0 D 192.168.1.254 Ethernet1/0/0

192.168.1.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

可以观察到此时每台路由器上都已经拥有了所有网段的路由信息，连通性检查这里省略。即现在网络通信恢复正常，并且所有PC也不会接收到任何RIP报文。

## 验证另一种抑制接口方式

RIP协议中还可以通过使用**undo rip output**命令来配置抑制接口，禁止接口发送RIP报文。

首先将R3上的现有抑制接口和单播更新的配置删除，然后在R3上的E 1/0/1上配置**undo rip output**命令, 禁止接口发送RIP报文。

[R3]rip 1

[R3-rip-1]undo silent-interface Ethernet 1/0/1

[R3-rip-1]undo silent-interface Ethernet 1/0/0

[R3-rip-1]undo peer 172.16.1.100

[R3-rip-1]undo peer 172.16.1.254

[R3-rip-1]interface Ethernet1/0/1

[R3-Ethernet1/0/1]undo rip output

配置完成后，等待一段时间，查看R1、R2路由器的路由表。

[R1-rip-1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 5 Routes : 5

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.254 GigabitEthernet0/0/1

172.16.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

172.16.2.0/24 RIP 100 1 D 172.16.1.100 GigabitEthernet0/0/1

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.100 Ethernet1/0/1

172.16.1.100/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

172.16.2.0/24 Direct 0 0 D 172.16.2.254 Ethernet1/0/0

172.16.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

可以观察到并没有R3上192.168.1.0所在直连网段的路由条目，说明R3上的**undo rip output**命令，已经生效，不再发送任何RIP路由更新。

在R3上配置与R1间的单播更新。

[R3]rip 1

[R3-rip-1]peer 172.16.1.254

配置完成后，等待一段时间，路由表收敛后，在R1上查看路由表。

[R1-rip-1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 5 Routes : 5

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.254 GigabitEthernet0/0/1

172.16.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

172.16.2.0/24 RIP 100 1 D 172.16.1.100 GigabitEthernet0/0/1

可以观察到R1上仍然没有192.168.1.0的路由条目。由此可以证明使用命令**undo rip output**来抑制接口，即使配置了单播更新也是无法再以单播的形式发送路由更新的。而在上一步骤当中，当使用**silent-interface**命令配置抑制接口后，再使用**peer**命令指定邻居IP的单播更新目的地址后，单播更新则生效。

在接口下可以使用**undo rip output**命令禁止该接口发送RIP报文，也可以使用**undo rip input**来禁止接口接收RIP报文，通过这两条命令可以灵活的控制接口下对RIP报文的发送和接收，默认情况下是可以接收和发送RIP报文。并且注意**silent-interface**命令的优先级大于**rip output**或**rip input**命令的优先级。

7.6 RIP与不连续子网

# 原理概述

RIP会在主网边界自动汇总，汇总发生时，汇总的子网路由在边界处被抑制掉，而仅通告主网路由。如果一台路由器上有两个接口，网段分别为10.1.1.0/24和172.16.1.0/24，那么在这两个网段的主网边界路由器就会自动将这两个网段汇总成10.0.0.0和172.16.0.0，并通告给其他路由器。如果主网的子网不连续，被其他主网所分隔，主网边界的自动汇总就会带来问题。

连续子网是指所相连的子网属于同一主网；不连续子网是指相同主网下的子网被另一主网分隔。

# 实验目的

* + 理解连续子网和不连续子网的概念
  + 掌握RIPv1中解决不连续子网问题的方法
  + 掌握RIPv2中解决不连续子网问题的方法
  + 理解RIPv1与RIPv2的区别

# 实验内容

在某公司的网络整改项目中，原先R1和R5属于同一主网络10.0.0.0/8，现被R2，R3，R4分离，整网采用了RIPv1协议，发现在该子网不连续的环境下通信出现了些问题，现需要通过额外的配置来解决这些问题，保证所有设备能够互通。

# 实验拓扑

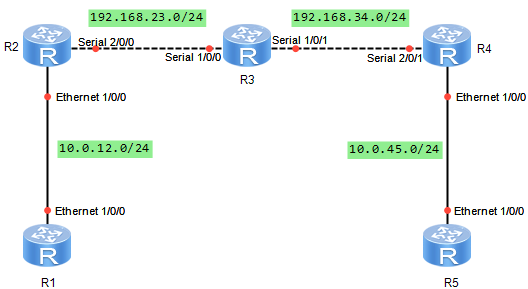


图7-15 RIP与不连续子网拓扑网

# 实验编址

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | Ethernet 1/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| R2(AR1220) | Ethernet 1/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| Serial 2/0/0 | 192.168.23.2 | 255.255.255.0 | N/A |
| R3(AR1220) | Serial 1/0/0 | 192.168.23.3 | 255.255.255.0 | N/A |
| Serial 1/0/1 | 192.168.34.3 | 255.255.255.0 | N/A |
| R4(AR1220) | Serial 2/0/1 | 192.168.34.4 | 255.255.255.0 | N/A |
| Ethernet 1/0/0 | 10.0.45.4 | 255.255.255.0 | N/A |
| R5(AR1220) | Ethernet 1/0/0 | 10.0.45.5 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP配置，并使用**ping**命令测试直连路由器间的连通性。

[R1]ping 10.0.12.2

PING 10.0.12.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=60 ms

Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=20 ms

Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=30 ms

Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=50 ms

--- 10.0.12.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 20/38/60 ms

其余直连网段的连通性测试省略。

## 组建基本的RIPv1网络

在路由器R1，R2，R3，R4，R5上配置RIPv1。

[R1]rip

[R1-rip-1]network 10.0.0.0

[R2]rip

[R2-rip-1]network 10.0.0.0

[R2-rip-1]network 192.168.23.0

[R3]rip

[R3-rip-1]network 192.168.23.0

[R3-rip-1]network 192.168.34.0

[R4]rip

[R4-rip-1]network 192.168.34.0

[R4-rip-1]network 10.0.0.0

[R5]rip

[R5-rip-1]network 10.0.0.0

配置完成后，查看R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Ethernet1/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 RIP 100 1 D 10.0.12.2 Ethernet1/0/0

192.168.34.0/24 RIP 100 2 D 10.0.12.2 Ethernet1/0/0

在R1的路由表中，存在192.168.23.0/24和192.168.34.0/24两条RIP路由条目，但并不存在R4和R5之间的10.0.45.0/24路由条目。

查看R2的路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Ethernet1/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 Direct 0 0 D 192.168.23.2 Serial2/0/0

192.168.23.2/32 Direct 0 0 D 127.0.0.1 Serial2/0/0

192.168.23.3/32 Direct 0 0 D 192.168.23.3 Serial2/0/0

192.168.34.0/24 RIP 100 1 D 192.168.23.3 Serial2/0/0

在R2的路由表中，除10.0.12.0/24和192.168.23.0/24为直连路由外，仅有192.168.34.0/24该条路由条目通过RIP接收，却没有R4和R5之间的10.0.45.0/24的路由条目。

查看R3的路由表。

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.0.0/8 RIP 100 1 D 192.168.23.2 Serial1/0/0

RIP 100 1 D 192.168.34.4 Serial1/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 Direct 0 0 D 192.168.23.3 Serial1/0/0

192.168.23.2/32 Direct 0 0 D 192.168.23.2 Serial1/0/0

192.168.23.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

192.168.34.0/24 Direct 0 0 D 192.168.34.3 Serial1/0/1

192.168.34.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

192.168.34.4/32 Direct 0 0 D 192.168.34.4 Serial1/0/1

在R3的路由表中，除了192.168.23.0/24和192.168.34.0/24这两个子网是直连之外，分别通过R2和R4接收到了两条相同的10.0.0.0/8的主网路由条目，而并非现网拓扑中的10.0.12.0/24和10.0.45.0/24两条子网路由。

导致这种情况的原因是：由于采用了RIPv1，在R2和R4分别接收到10.0.12.0/24和10.0.45.0/24的路由条目时，且默认打开了自动有类汇总功能，所以在主网边界向外发送路由信息的时候都汇总成了10.0.0.0/8，发送给R3，最终在R3上由于接收到了两条目的网段相同，代价值也相同的路由条目。

那么既然此时在R3的路由表中存在有10.0.0.0/8的路由，在R3上测试到R1和R5的连通性。

[R3]ping 10.0.45.5

PING 10.0.45.5: 56 data bytes, press CTRL\_C to break

Reply from 10.0.45.5: bytes=56 Sequence=1 ttl=254 time=80 ms

Reply from 10.0.45.5: bytes=56 Sequence=2 ttl=254 time=40 ms

Reply from 10.0.45.5: bytes=56 Sequence=3 ttl=254 time=80 ms

Reply from 10.0.45.5: bytes=56 Sequence=4 ttl=254 time=80 ms

Reply from 10.0.45.5: bytes=56 Sequence=5 ttl=254 time=80 ms

--- 10.0.45.5 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 40/72/80 ms

[R3]ping 10.0.12.1

PING 10.0.12.1: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

--- 10.0.12.1 ping statistics ---

5 packet(s) transmitted

0 packet(s) received

100.00% packet loss

发现此时呈现出有一端无法通信的现象。在R3发送ICMP报文的时候，会根据路由表进行匹配，即匹配10.0.0.0/8，那么最终报文流量可能会出现R3将本该要发送给R1的ICMP报文错误的转发给了R4导致。

现在每台设备上的路由表中没有清晰的反馈出拓扑中的真实子网信息，这是由于在RIPv1默认自动汇总开启的情况下，设计网络时没有遵循主网的子网应该连续这一要求所致，解决的办法视路由器所使用的RIP版本是版本v1还是v2有所不同。

## RIPv1中解决不连续子网问题

路由器上所运行RIP协议的默认版本是v1，自动汇总无法关闭，所以上面不连续子网所带来的问题，不能通过关闭自动汇总来解决。但如果把不连续的子网转变成连续的子网，问题就可以解决，办法是给接口配置第二个IP地址，IP地址取10.0.0.0/8主网的子网。

在路由器R2上的S 2/0/0接口上配置从IP地址，只要在常规配置IP地址的命令之后加上sub参数即可。

[R2]interface serial 2/0/0

[R2-Serial2/0/0]ip address 10.0.23.2 24 sub

同理，在R3和R4上也做相应配置。并在R3的RIP进程中添加10.0.0.0网段。

[R3]interface serial 1/0/0

[R3-Serial1/0/0]ip address 10.0.23.3 24 sub

[R3-Serial1/0/0]interface s0/0/1

[R3-Serial1/0/1]ip address 10.0.34.3 24 sub

[R3-Serial1/0/1]rip

[R3-rip-1]network 10.0.0.0

[R4]interface serial 2/0/1

[R4-Serial2/0/1]ip address 10.0.34.4 24 sub

经过这样的配置之后，相当于原先在整网拓扑中被孤立的两个不连续子网10.0.12.0/24和10.0.45.0/24网段被新添加的子网10.0.23.0/24和10.0.34.0/24网段连接了起来，即现在已经构成了一个连续的子网。

配置完成后，观察每台路由器的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.0.0/8 RIP 100 1 D 10.0.12.2 Ethernet1/0/0

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Ethernet1/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

10.0.23.0/24 RIP 100 1 D 10.0.12.2 Ethernet1/0/0

10.0.34.0/24 RIP 100 2 D 10.0.12.2 Ethernet1/0/0

10.0.45.0/24 RIP 100 3 D 10.0.12.2 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 RIP 100 1 D 10.0.12.2 Ethernet1/0/0

192.168.34.0/24 RIP 100 2 D 10.0.12.2 Ethernet1/0/0

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 13 Routes : 15

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.0.0/8 RIP 100 1 D 10.0.23.3 Serial2/0/0

RIP 100 1 D 192.168.23.3 Serial2/0/0

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Ethernet1/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

10.0.23.0/24 Direct 0 0 D 10.0.23.2 Serial2/0/0

10.0.23.2/32 Direct 0 0 D 127.0.0.1 Serial2/0/0

10.0.34.0/24 RIP 100 1 D 10.0.23.3 Serial2/0/0

10.0.45.0/24 RIP 100 2 D 10.0.23.3 Serial2/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 Direct 0 0 D 192.168.23.2 Serial2/0/0

192.168.23.2/32 Direct 0 0 D 127.0.0.1 Serial2/0/0

192.168.23.3/32 Direct 0 0 D 192.168.23.3 Serial2/0/0

192.168.34.0/24 RIP 100 1 D 192.168.23.3 Serial2/0/0

RIP 100 1 D 10.0.23.3 Serial2/0/0

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 15 Routes : 16

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.0.0/8 RIP 100 1 D 192.168.23.2 Serial1/0/0

RIP 100 1 D 192.168.34.4 Serial1/0/1

10.0.12.0/24 RIP 100 1 D 10.0.23.2 Serial1/0/0

10.0.23.0/24 Direct 0 0 D 10.0.23.3 Serial1/0/0

10.0.23.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.0.34.0/24 Direct 0 0 D 10.0.34.3 Serial1/0/1

10.0.34.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

10.0.45.0/24 RIP 100 1 D 10.0.34.4 Serial1/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 Direct 0 0 D 192.168.23.3 Serial1/0/0

192.168.23.2/32 Direct 0 0 D 192.168.23.2 Serial1/0/0

192.168.23.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

192.168.34.0/24 Direct 0 0 D 192.168.34.3 Serial1/0/1

192.168.34.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

192.168.34.4/32 Direct 0 0 D 192.168.34.4 Serial1/0/1

[R4]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 13 Routes : 15

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.0.0/8 RIP 100 1 D 192.168.34.3 Serial2/0/1

RIP 100 1 D 10.0.34.3 Serial2/0/1

10.0.12.0/24 RIP 100 2 D 10.0.34.3 Serial2/0/1

10.0.23.0/24 RIP 100 1 D 10.0.34.3 Serial2/0/1

10.0.34.0/24 Direct 0 0 D 10.0.34.4 Serial2/0/1

10.0.34.4/32 Direct 0 0 D 127.0.0.1 Serial2/0/1

10.0.45.0/24 Direct 0 0 D 10.0.45.4 Ethernet1/0/0

10.0.45.4/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 RIP 100 1 D 192.168.34.3 Serial2/0/1

RIP 100 1 D 10.0.34.3 Serial2/0/1

192.168.34.0/24 Direct 0 0 D 192.168.34.4 Serial2/0/1

192.168.34.3/32 Direct 0 0 D 192.168.34.3 Serial2/0/1

192.168.34.4/32 Direct 0 0 D 127.0.0.1 Serial2/0/1

[R5]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.0.0/8 RIP 100 1 D 10.0.45.4 Ethernet1/0/0

10.0.12.0/24 RIP 100 3 D 10.0.45.4 Ethernet1/0/0

10.0.23.0/24 RIP 100 2 D 10.0.45.4 Ethernet1/0/0

10.0.34.0/24 RIP 100 1 D 10.0.45.4 Ethernet1/0/0

10.0.45.0/24 Direct 0 0 D 10.0.45.5 Ethernet1/0/0

10.0.45.5/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 RIP 100 2 D 10.0.45.4 Ethernet1/0/0

192.168.34.0/24 RIP 100 1 D 10.0.45.4 Ethernet1/0/0

此时发现在每台路由器的路由表中都拥有了所有的子网信息。其中在R2上，由于R3的S 1/0/0接口配置了第二IP地址，所以下一跳为R3的S1/0/0接口的路由条目会出现两个下一跳地址。

在R1上测试与R5之间的连通性。

[R1]ping 10.0.45.5

PING 10.0.45.5: 56 data bytes, press CTRL\_C to break

Reply from 10.0.45.5: bytes=56 Sequence=1 ttl=252 time=110 ms

Reply from 10.0.45.5: bytes=56 Sequence=2 ttl=252 time=140 ms

Reply from 10.0.45.5: bytes=56 Sequence=3 ttl=252 time=80 ms

Reply from 10.0.45.5: bytes=56 Sequence=4 ttl=252 time=100 ms

Reply from 10.0.45.5: bytes=56 Sequence=5 ttl=252 time=90 ms

--- 10.0.45.5 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 80/104/140 ms

连通性测试成功。

上述做法通过在不连续的子网之间的链路上配置相同主网的子网IP地址，即采用配置从IP地址的方式来实现子网的连续性，解决了因为自动汇总发生后，子网路由被抑制掉而导致的子网不可达。此种做法优点是RIPv1在不做大的拓扑结构调整的前提下，仅靠配置第二个IP地址就解决了不连续子网问题；不足之处是需要配置第二个IP地址，要消耗掉多个子网网段。

## RIPv2中解决不连续子网问题

如果路由器运行的是RIPv2，则可以直接关闭自动汇总，子网是否连续就不重要了，因为RIPv2会直接通告相应的子网路由。

删除上一步骤中的从IP地址配置命令，并在所有路由器中将RIP的版本配置为2，且关闭自动汇总。

[R1]rip

[R1-rip-1]version 2

[R1-rip-1]undo summary

[R2]interface serial2/0/0

[R2-Serial2/0/0]undo ip address 10.0.23.2 24 sub

[R2-Serial2/0/0]rip

[R2-rip-1]version 2

[R2-rip-1]undo summary

[R3]interface serial1/0/0

[R3-Serial1/0/0]undo ip address 10.0.23.3 24 sub

[R3-Serial1/0/0]interface serial0/0/1

[R3-Serial1/0/1]undo ip address 10.0.34.3 24 sub

[R3-Serial1/0/1]rip

[R3-rip-1]version 2

[R3-rip-1]undo summary

[R4]interface serial 2/0/1

[R4-Serial2/0/1]undo ip address 10.0.34.4 24 sub

[R4-Serial2/0/1]rip

[R4-rip-1]version 2

[R4-rip-1]undo summary

[R5]rip

[R5-rip-1]version 2

[R5-rip-1]undo summary

配置完成后，观察每台路由器上的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.1 Ethernet1/0/0

10.0.12.1/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

10.0.45.0/24 RIP 100 3 D 10.0.12.2 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 RIP 100 1 D 10.0.12.2 Ethernet1/0/0

192.168.34.0/24 RIP 100 2 D 10.0.12.2 Ethernet1/0/0

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 Direct 0 0 D 10.0.12.2 Ethernet1/0/0

10.0.12.2/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

10.0.45.0/24 RIP 100 2 D 192.168.23.3 Serial2/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 Direct 0 0 D 192.168.23.2 Serial2/0/0

192.168.23.2/32 Direct 0 0 D 127.0.0.1 Serial2/0/0

192.168.23.3/32 Direct 0 0 D 192.168.23.3 Serial2/0/0

192.168.34.0/24 RIP 100 1 D 192.168.23.3 Serial2/0/0

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 RIP 100 1 D 192.168.23.2 Serial1/0/0

10.0.45.0/24 RIP 100 1 D 192.168.34.4 Serial1/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 Direct 0 0 D 192.168.23.3 Serial1/0/0

192.168.23.2/32 Direct 0 0 D 192.168.23.2 Serial1/0/0

192.168.23.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

192.168.34.0/24 Direct 0 0 D 192.168.34.3 Serial1/0/1

192.168.34.3/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

192.168.34.4/32 Direct 0 0 D 192.168.34.4 Serial1/0/1

[R4]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 RIP 100 2 D 192.168.34.3 Serial2/0/1

10.0.45.0/24 Direct 0 0 D 10.0.45.4 Ethernet1/0/0

10.0.45.4/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 RIP 100 1 D 192.168.34.3 Serial2/0/1

192.168.34.0/24 Direct 0 0 D 192.168.34.4 Serial2/0/1

192.168.34.3/32 Direct 0 0 D 192.168.34.3 Serial2/0/1

192.168.34.4/32 Direct 0 0 D 127.0.0.1 Serial2/0/1

[R5]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.12.0/24 RIP 100 3 D 10.0.45.4 Ethernet1/0/0

10.0.45.0/24 Direct 0 0 D 10.0.45.5 Ethernet1/0/0

10.0.45.5/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

192.168.23.0/24 RIP 100 2 D 10.0.45.4 Ethernet1/0/0

192.168.34.0/24 RIP 100 1 D 10.0.45.4 Ethernet1/0/0

可以看到在所有路由表中都没有汇总路由10.0.0.0/8，并且10.0.45.0/24和10.0.12.0/24子网出现在所有的路由表中。

测试R1，R5间的连通性。

[R1]ping 10.0.45.5

PING 10.0.45.5: 56 data bytes, press CTRL\_C to break

Reply from 10.0.45.5: bytes=56 Sequence=1 ttl=252 time=90 ms

Reply from 10.0.45.5: bytes=56 Sequence=2 ttl=252 time=130 ms

Reply from 10.0.45.5: bytes=56 Sequence=3 ttl=252 time=90 ms

Reply from 10.0.45.5: bytes=56 Sequence=4 ttl=252 time=110 ms

Reply from 10.0.45.5: bytes=56 Sequence=5 ttl=252 time=90 ms

--- 10.0.45.5 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 90/102/130 ms

在RIPv2的环境中，因为默认情况下自动汇总是开启的，所以网络在设计时，应尽量不要出现同主网的子网被其它主网分隔的情况。如果出现了，关闭自动汇总是最佳的做法，不足之处是路由表中路由条目会有所增多。

# 思考

在使用RIPv1的环境中，R2、R3和R4都配置了第二个IP地址，10.0.0.0/8的子网已经连续，如果R2是主网边界，为什么R3还能看到10.0.12.0/24的子网？如果R2不是主网边界，为什么在R3的路由表里能看到10.0.0.0/8的汇总路由？

7.7 RIP的水平分割及触发更新

# 原理概述

水平分割（Split Horizon）指的是RIP从某个接口接收到的路由信息，不会从该接口再发回给邻居设备。这样不但减少了带宽消耗，还可以防止路由环路。在华为设备上，水平分割功能缺省情况下是开启的。

触发更新（Triggered Updates）的原理是，当路由信息发生变化时，运行RIP的设备会立即向邻居设备发送更新报文，而不必等待定时更新，缩短了网络收敛时间。在华为设备上，没有相关命令能主动关闭触发更新的功能。

毒性逆转（Poison Reverse）指的是RIP从某个接口接收到路由信息后，将该路由的开销设置为16（即该路由不可达），并从原接口发回邻居设备。利用这种方式，可以清除对方路由表中的无用路由。如果毒性逆转和水平分割都配置了，水平分割行为会被毒性逆转行为代替。在华为设备上，毒性逆转功能缺省情况下是关闭的，需要手动打开此功能。

毒性逆转可以快速清除无用的路由而不必等待老化时间，另外，在帧中继和X.25等非广播多路访问网络中，如果开启了水平分割功能，会造成有的路由器无法接收到更新路由的情况，因此在这种网络中，水平分割是缺省禁止的，我们需要手动开启毒性逆转防止路由环路。

RIPv1和RIPv2都支持水平分割、触发更新和毒性逆转功能。

# 实验目的

* + 掌握RIP触发更新的原理
  + 掌握RIP中触发更新与等待老化时间的现象差别
  + 掌握RIP水平分割的原理和配置
  + 掌握开启水平分割与关掉水平分割时路由条目的变化
  + 掌握毒性逆转原理和配置

# 实验内容

本实验模拟企业网络场景，R1为该公司出口网关路由器，连接运营商网络，R2为公司IT部门路由器，通过交换机S2与网关相连；同时公司人力资源部路由器R3也通过交换机S1与网关相连，所有路由器运行路由协议RIPv2实现公司整网互通。当R3与S1之间链路down掉时R1不会触发更新，网络收敛较慢，而当R1与S1之间链路down掉时，R1会触发更新，网络收敛快；华为路由器默认开启RIP水平分割功能，当主动关掉路由器上水平分割功能时，检查路由器发送RIP路由条目的变化。手动开启毒性逆转功能，检查路由器发送RIP路由条目的变化。

# 实验拓扑

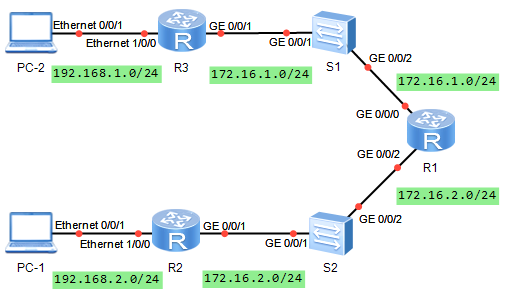


图7-16 RIP的水平分割及触发更新拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 172.16.1.1 | 255.255.255.0 | N/A |
| GE 0/0/2 | 172.16.2.1 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/1 | 172.16.2.2 | 255.255.255.0 | N/A |
| Ethernet 1/0/0 | 192.168.2.254 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/1 | 172.16.1.2 | 255.255.255.0 | N/A |
| Ethernet 1/0/0 | 192.168.1.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 192.168.2.1 | 255.255.255.0 | 192.168.2.254 |
| PC-2 | Ethernet 0/0/1 | 192.168.1.1 | 255.255.255.0 | 192.168.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping -c 1 172.16.1.2

PING 172.16.1.2: 56 data bytes, press CTRL\_C to break

Reply from 172.16.1.2: bytes=56 Sequence=1 ttl=255 time=80 ms

--- 172.16.1.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 80/80/80 ms

其余直连网段的连通性测试省略。

## 搭建RIP网络

根据公司组网拓扑图，在R1，R2，R3上配置RIP协议，并将相应网段通告进RIP协议中。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]network 172.16.0.0

[R2]rip 1

[R2-rip-1]version 2

[R2-rip-1]network 192.168.2.0

[R2-rip-1]network 172.16.0.0

[R3]rip 1

[R3-rip-1]version 2

[R3-rip-1]network 172.16.0.0

[R3-rip-1]network 192.168.1.0

配置完成后，查看R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.1 GigabitEthernet0/0/0

172.16.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

172.16.2.0/24 Direct 0 0 D 172.16.2.1 GigabitEthernet0/0/2

172.16.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

192.168.1.0/24 RIP 100 1 D 172.16.1.2 GigabitEthernet0/0/0

192.168.2.0/24 RIP 100 1 D 172.16.2.2 GigabitEthernet0/0/2

可以观察到，R1已经正常获取到了PC-1与PC-2所在网段的RIP路由信息。R2，R3上的查看省略。

## 验证触发更新

断掉R3与S1之间的链路（在模拟器上直接删除该链路）。

查看R2的路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 RIP 100 1 D 172.16.2.1 GigabitEthernet0/0/1

172.16.2.0/24 Direct 0 0 D 172.16.2.2 GigabitEthernet0/0/1

172.16.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.1.0/24 RIP 100 2 D 172.16.2.1 GigabitEthernet0/0/1

192.168.2.0/24 Direct 0 0 D 192.168.2.254 Ethernet1/0/0

192.168.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

可以观察到192.168.1.0网段的路由信息仍然存在；这是因为down掉的不是R1的直连接口，R1此时无法直接感知到故障的发生，路由条目需要等180秒的老化计时器超时后，此路由条目才会在路由表中删除。

等待180秒老化时间过后，查看R2的路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 RIP 100 1 D 172.16.2.1 GigabitEthernet0/0/1

172.16.2.0/24 Direct 0 0 D 172.16.2.2 GigabitEthernet0/0/1

172.16.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.2.0/24 Direct 0 0 D 192.168.2.254 Ethernet1/0/0

192.168.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

可以观察到此时192.168.1.0网段的路由信息已经从路由表中删除。

恢复R3与S1之间的链路（在模拟器上重新连线），并在R2的路由表正常后，断掉R1与S1之间的链路，查看R2路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.2.0/24 Direct 0 0 D 172.16.2.2 GigabitEthernet0/0/1

172.16.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.2.0/24 Direct 0 0 D 192.168.2.254 Ethernet1/0/0

192.168.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

可以观察到192.168.1.0网段的路由信息已经不存在，因为down掉的是R1的直连接口，所以R1能够直接感知到链路发生故障，在路由表删除192.168.1.0的路由同时，并会触发更新，使得R2上的路由表为最新状态。

## 验证水平分割

在R2上使用命令**debugging rip 1 send GigabitEthernet 0/0/1**打开debug功能，再用命令**terminal monitor、terminal debugging**可查看R2发送给R1的路由条目。

<R2>debugging rip 1 send GigabitEthernet 0/0/1

<R2>terminal monitor

<R2>terminal debugging

<R2>

Jun 19 2013 19:17:59.640.1-08:00 R2 RIP/7/DBG: 6: 12227: RIP 1: Sending response

on interface GigabitEthernet0/0/1 from 172.16.2.2 to 224.0.0.9

Jun 19 2013 19:17:59.640.2-08:00 R2 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd response, Length 24

Jun 19 2013 19:17:59.640.3-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0

从debug的信息中可以观察到R2发送给R1的路由条目中没有包含192.168.1.0网段的路由信息，因为该路由条目是从R1始发过来。

在华为设备上RIP的水平分割是默认开启的，某个接口获得的路由，不会再从该接口再发回给邻居设备。这样不但减少了带宽消耗，还可以防止路由环路。

关闭debug，并在R2的GE 0/0/1和R1的GE 0/0/2接口下使用命令**undo rip split-horizon**关闭水平分割功能。

<R2>undo debugging all

<R2>system-view

[R2]interface GigabitEthernet0/0/1

[R2-GigabitEthernet0/0/1]undo rip split-horizon

[R1]interface GigabitEthernet0/0/2

[R1-GigabitEthernet0/0/2]undo rip split-horizon

配置完成后，用相同命令的查看debug信息。

<R2>debugging rip 1 send GigabitEthernet 0/0/1

<R2>terminal monitor

<R2>terminal debugging

Jun 19 2013 19:21:53.910.4-08:00 R2 RIP/7/DBG: 6: 12315: Dest 172.16.0.0/16, Next

hop 0.0.0.0, Cost 1, Tag 0

Jun 19 2013 19:21:53.910.5-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.1.0/24, Nexthop 0.0.0.0, Cost 3, Tag 0

Jun 19 2013 19:21:53.910.6-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0

从debug信息中可以观察R2发送给R1的路由条目中包含有192.168.1.0网段，此时接口上的水平分割功能不生效。

## 验证毒性逆转

在华为设备上，毒性逆转功能是默认关闭的，需要手动打开此功能。

关闭debug，并在R2的GE 0/0/1接口下恢复水平分割功能。

<R2>undo debugging all

<R2>system-view

[R2]interface GigabitEthernet0/0/1

[R2-GigabitEthernet0/0/1]rip split-horizon

配置完成后，在R2上开启debug功能。

<R2>debugging rip 1 send GigabitEthernet 0/0/1

<R2>terminal monitor

<R2>terminal debugging

<R2>

Jun 19 2013 19:28:06.720.1-08:00 R2 RIP/7/DBG: 6: 12227: RIP 1: Sending response

on interface GigabitEthernet0/0/1 from 172.16.2.2 to 224.0.0.9

Jun 19 2013 19:28:06.720.2-08:00 R2 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd response, Length 24

Jun 19 2013 19:28:06.720.3-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0

通过debug信息可以观察到，此时开启了水平分割后，R2发送给R1的路由条目中没有包含192.168.1.0网段。

关闭debug，并在R2的GE 0/0/1接口下开启毒性逆转功能。

<R2>undo debugging all

<R2>system-view

[R2]interface GigabitEthernet0/0/1

[R2-GigabitEthernet0/0/1]rip poison-reverse

配置完成后，查看debug信息。

<R2>debugging rip 1 send GigabitEthernet 0/0/1

<R2>terminal monitor

<R2>terminal debugging

Jun 19 2013 19:30:00.520.2-08:00 R2 RIP/7/DBG: 6: 12247: Packet: Version 2, Cmd r·esponse, Length 64

Jun 19 2013 19:30:00.520.4-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.1.0/24, Nexthop 172.16.2.1, Cost 16, Tag 0

Jun 19 2013 19:30:00.520.5-08:00 R2 RIP/7/DBG: 6: 12315: Dest 192.168.2.0/24, Nexthop 0.0.0.0, Cost 1, Tag 0

从debug信息可以观察到，R2发送给R1的路由条目中包含了192.168.1.0网段，但是cost值为16。

说明在毒性逆转和水平分割同时开启的情况下，简单的水平分割行为（从某接口学到的路由再从这个接口发布时将被抑制）会被毒性逆转行为代替，并且在华为设备上RIP的毒性逆转功能是默认关闭的。

# 思考

水平分割可以防止环路，那为什么RIP协议还需要其它防环机制？水平分割的局限性在哪？

7.8 配置RIP路由附加度量值

# 原理概述

附加路由度量值是在RIP路由原来度量值的基础上所增加或减少的度量值（跳数）。对于RIP接收和发布路由，可通过不同的命令配置附加度量值更加灵活的控制RIP的路由选择。

[**rip metricin**](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_metricin.html)用于在接收到路由后，给其增加一个附加度量值，再加入路由表中，使得路由表中的度量值发生变化。运行该命令会影响到本地设备和其他设备的路由选择。

[**rip metricout**](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_metricout.html)用于自身路由的发布，发布时增加一个附加的度量值，但本地路由表中的度量值不会发生变化。运行该命令不会影响本地设备的路由选择，但是会影响其他设备的路由选择。

# 实验目的

* + 理解RIP路由附加度量值应用场景
  + 掌握使用**metricin**方式附加度量值的方法
  + 掌握使用**metricout**方式附加度量值的方法

# 实验内容

本实验模拟公司网络场景，路由器R1左侧连接的是公司市场部，路由器R4右侧连接的是公司财务部，R1与R4之间通过R2、R3双链路连接，所有路由器运行RIP协议，R1与R4之间互访的流量通过两条链路负载分担。现在网络管理员在R2上做了流量控制，要求所有市场部访问财务部的流量都必须经过R2，同时为了减轻R2的负担，由财务部去往市场部的流量都由R3来转发。网络管理员可通过两种RIP路由附加度量值的方式修改相应路由度量值，灵活控制RIP的路由选择来达到公司的流量控制。

# 实验拓扑

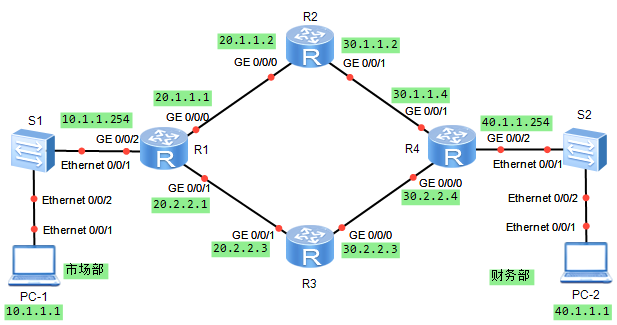


图7-17 配置RIP路由附加度量值拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 20.1.1.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.2.2.1 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.1.1.254 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 20.1.1.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 30.1.1.2 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 30.2.2.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.2.2.3 | 255.255.255.0 | N/A |
| R4(AR2220) | GE 0/0/0 | 30.2.2.4 | 255.255.255.0 | N/A |
| GE 0/0/1 | 30.1.1.4 | 255.255.255.0 | N/A |
| GE 0/0/2 | 40.1.1.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | 10.1.1.254 |
| PC-2 | Ethernet 0/0/1 | 40.1.1.1 | 255.255.255.128 | 40.1.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping 20.1.1.2

PING 20.1.1.2: 56 data bytes, press CTRL\_C to break

Reply from 20.1.1.2: bytes=56 Sequence=1 ttl=255 time=30 ms

Reply from 20.1.1.2: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 20.1.1.2: bytes=56 Sequence=3 ttl=255 time=50 ms

Reply from 20.1.1.2: bytes=56 Sequence=4 ttl=255 time=10 ms

Reply from 20.1.1.2: bytes=56 Sequence=5 ttl=255 time=20 ms

--- 20.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/24/50 ms

其他直连网段的连通性检测省略。

## 搭建RIP网络

公司内部网络使用RIP协议。首先配置R1、R2，R3和R4运行RIP协议，通告所有网段，使公司网络互通。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]undo summary

[R1-rip-1]network 10.0.0.0

[R1-rip-1]network 20.0.0.0

其他路由器配置省略。配置完成后查看R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 11 Routes : 12

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/2

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

20.1.1.0/24 Direct 0 0 D 20.1.1.1 GigabitEthernet0/0/0

20.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.2.2.0/24 Direct 0 0 D 20.2.2.1 GigabitEthernet0/0/1

20.2.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

30.1.1.0/24 RIP 100 1 D 20.1.1.2 GigabitEthernet0/0/0

30.2.2.0/24 RIP 100 1 D 20.2.2.3 GigabitEthernet0/0/1

40.1.1.0/24 RIP 100 2 D 20.2.2.3 GigabitEthernet0/0/1

RIP 100 2 D 20.1.1.2 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到在R1存在有两条通过RIP协议接收到的去往财务部所在网段40.1.1.1/24的路由条目。

同样在R4上查看路由表。

<R4>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 11 Routes : 12

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 2 D 30.2.2.3 GigabitEthernet0/0/0

RIP 100 2 D 30.1.1.2 GigabitEthernet0/0/1

20.1.1.0/24 RIP 100 1 D 30.1.1.2 GigabitEthernet0/0/1

20.2.2.0/24 RIP 100 1 D 30.2.2.3 GigabitEthernet0/0/0

30.1.1.0/24 Direct 0 0 D 30.1.1.4 GigabitEthernet0/0/1

30.1.1.4/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

30.2.2.0/24 Direct 0 0 D 30.2.2.4 GigabitEthernet0/0/0

30.2.2.4/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

40.1.1.0/24 Direct 0 0 D 40.1.1.254 GigabitEthernet0/0/2

40.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

同样可以观察到在R4存在有两条通过RIP协议接收到的去往市场部所在网段10.1.1.0/24的路由条目，且有两个下一跳，呈现负载分担。

## 配置[RIP Metricin](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_metricin.html)

在R1的GE 0/0/1接口下配置**[rip metricin](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_metricin.html) 2**，设置R1在接收R3发送来的路由条目时增加度量值2。这样由R3发给R1的路由条目的度量值将大于R2发给R1的路由，R1会优选R2发来的RIP路由条目，并加入路由表中。

[R1]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]rip metricin 2

配置完成后，查看路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 11 Routes : 11

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/2

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

20.1.1.0/24 Direct 0 0 D 20.1.1.1 GigabitEthernet0/0/0

20.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.2.2.0/24 Direct 0 0 D 20.2.2.1 GigabitEthernet0/0/1

20.2.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

30.1.1.0/24 RIP 100 1 D 20.1.1.2 GigabitEthernet0/0/0

30.2.2.0/24 RIP 100 2 D 20.1.1.2 GigabitEthernet0/0/0

40.1.1.0/24 RIP 100 2 D 20.1.1.2 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，此时在R1访问财务部网段只有一个下一跳R2，在R1上查看RIP的数据库。

[R1]display rip 1 database

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

Advertisement State : [A] - Advertised

[I] - Not Advertised/Withdraw

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

10.0.0.0/8, cost 0, ClassfulSumm

10.1.1.0/24, cost 0, [A], Rip-interface

20.0.0.0/8, cost 0, ClassfulSumm

20.1.1.0/24, cost 0, [A], Rip-interface

20.2.2.0/24, cost 0, [A], Rip-interface

30.0.0.0/8, cost 1, ClassfulSumm

30.1.1.0/24, cost 1, [A], nexthop 20.1.1.2

30.2.2.0/24, cost 2, [A], nexthop 20.1.1.2

30.2.2.0/24, cost 16, [I], nexthop 20.2.2.3

40.0.0.0/8, cost 2, ClassfulSumm

40.1.1.0/24, cost 2, [A], nexthop 20.1.1.2

40.1.1.0/24, cost 16, [I], nexthop 20.2.2.3

可以观察到在RIP的数据库中，还是存在下一跳为R3的去往40.1.1.0网段的路由，但COST值被设为16不可达。

在PC-1上测试访问PC-2所经过的网关设备。

PC>tracert 40.1.1.1

traceroute to 40.1.1.1, 8 hops max

(ICMP), press Ctrl+C to stop

1 10.1.1.254 31 ms 15 ms 47 ms

2 20.1.1.2 78 ms 62 ms 47 ms

3 30.1.1.4 156 ms 78 ms 109 ms

4 40.1.1.1 156 ms 78 ms 63 ms

可以观察到，数据包此时是经过R2转发至PC-2。

## 配置[RIP Metric](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_metricin.html)out

为了减轻R2的负担，所有由财务部去往市场部的流量都由R3来转发。

在R2上的GE 0/0/1接口下配置**[rip metricout](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_metricin.html) 2**，设置R2在向R4发送路由条目时增加度量值2。这样R4收到来自R2的路由的度量值就会大于来自R3的路由，R4会优选来自R3的RIP路由条目，并加入到路由表中。

[R2]interface GigabitEthernet0/0/1

[R2-GigabitEthernet0/0/1]rip metricout 2

配置完成后，查看路由表。

<R4>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 11 Routes : 11

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 2 D 30.2.2.3 GigabitEthernet0/0/0

20.1.1.0/24 RIP 100 2 D 30.2.2.3 GigabitEthernet0/0/0

20.2.2.0/24 RIP 100 1 D 30.2.2.3 GigabitEthernet0/0/0

30.1.1.0/24 Direct 0 0 D 30.1.1.4 GigabitEthernet0/0/1

30.1.1.4/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

30.2.2.0/24 Direct 0 0 D 30.2.2.4 GigabitEthernet0/0/0

30.2.2.4/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

40.1.1.0/24 Direct 0 0 D 40.1.1.254 GigabitEthernet0/0/2

40.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，此时在R4访问市场部网段只有一个下一跳R3，在PC-1上测试访问PC-2所经过的网关设备。

PC>tracert 40.1.1.1

traceroute to 40.1.1.1, 8 hops max

(ICMP), press Ctrl+C to stop

1 10.1.1.254 31 ms 15 ms 47 ms

2 20.1.1.2 78 ms 62 ms 47 ms

3 30.1.1.4 156 ms 78 ms 109 ms

4 40.1.1.1 156 ms 78 ms 63 ms

可以观察到，数据包此时是经过R3转发至PC-1。

# 思考

无论是配置metricin还是metricout都会将所有RIP路由条目的度量值增加，如何在完成对财务部路由附加度量值配置的同时不影响其他RIP路由的Metric值？

7.9 RIP的故障处理

# 原理概述

在实际的小型网络中，设计者可能会选用RIP作为网络路由协议，但在实际操作配置中，往往会出现人为的疏忽、错误配置等种种问题造成网络不能正常通信。这便需要我们有一定的故障处理能力，去排除网络中的故障来恢复网络通信。

下面为RIP中的常用排障流程：

1．接口状态不是UP；

2．RIP进程下没有对该网段做network配置；

3．对端RIP协议报文的版本号和本地接收的RIP协议报文版本号不一致；

4．接口上配置了禁止接收RIP报文或禁止发送RIP报文的命令；

5．在RIP中配置了策略，过滤掉收到的RIP路由或不允许发送RIP路由；

6．接口上没有开启水平分割；

7．链路两端的接口认证方式不匹配；

8．路由表中存在从其它协议获得的相同路由条目；

9．收到的路由度量值大于16。

可以根据以上的常见问题，制定出合理的故障排除步骤。比如检查第一项时，首先使用相关的配置命令查看接口状态是否UP，这同时就需要熟悉用于检查各个状态的配置命令。若第一项没有问题，则顺序的查看下一项，直到排查出错误，恢复网络的正常。

# 实验目的

* + 掌握RIP故障的常见原因
  + 掌握RIP故障诊断流程
  + 掌握RIP故障处理步骤
  + 掌握RIP故障排除的常用命令

# 实验内容

本实验模拟企业网络场景，R1为该公司出口网关路由器，连接运营商网络，R2为公司HR部门路由器与网关相连；同时公司人力资源部路由器R2也与网关相连，由于公司的网络规模比较小，所以选择使用RIPv2来作为动态路由协议实现公司整网互通。现在公司IT部门员工发现自己用PC-2无法与HR部门的PC-1通信，作为公司的网络管理员，现需对此网络故障进行排查，恢复网络，两台PC都确认了IP地址和网关地址设置正确，现给出公司网络拓扑图如下以及三台路由器的配置，请用模拟器搭建如下网络并把已经给出的配置拷贝入对应路由器中，再进行故障处理。本实验较全面的介绍了RIP的排障流程，适合大部分RIP网络。

# 实验拓扑

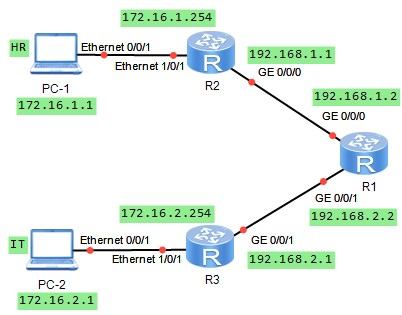


图7-18 RIP的故障排除拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 192.168.1.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 192.168.2.2 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 192.168.1.1 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.1.254 | 255.255.255.0 | N/A |
| R3(AR1220) | GE 0/0/1 | 192.168.2.1 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.2.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 172.16.1.1 | 255.255.255.0 | 172.16.1.254 |
| PC-2 | Ethernet 0/0/1 | 172.16.2.1 | 255.255.255.0 | 172.16.2.254 |

# 实验步骤

## 导入设备预配置

本实验中设置了如下故障点：

R3缺少**network** 192.168.2.0命令；

在R3的GE 0/0/1接口下配置**[undo rip input](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_input.html)**命令；

关闭 R1上的GE 0/0/1接口；

在R1的GE 0/0/0接口下配置**rip metricin15**命令；

在R2上的GE 0/0/0接口下配置RIP认证，方式为明文认证，密码huawei。

下面即为三台路由器R1，R2，R3的初始配置，直接使用即可。

system-view

sysname R1

interface GigabitEthernet0/0/0

ip address 192.168.1.2 255.255.255.0

rip metricin 15

interface GigabitEthernet0/0/1

ip address 192.168.2.2 255.255.255.0

shutdown

rip 1

version 2

network 192.168.1.0

network 192.168.2.0

system-view

sysname R2

interface Ethernet1/0/1

ip address 172.16.1.254 255.255.255.0

interface GigabitEthernet0/0/0

ip address 192.168.1.1 255.255.255.0

rip authentication-mode simple huawei

rip 1

version 2

network 172.16.0.0

network 192.168.1.0

system-view

sysname R3

interface Ethernet1/0/1

ip address 172.16.2.254 255.255.255.0

interface GigabitEthernet0/0/1

ip address 192.168.2.1 255.255.255.0

undo rip input

rip 1

version 2

network 172.16.0.0

在接下来的步骤中网络管理员需要根据和逻辑的思路，并借助必要的查看命令来进行排障。

## 排除R1与R2间的故障

网络管理员发现公司网络此时出现故障，IT部门与人力资源部门的终端无法正常互访。

测试IT部门PC-1与人力资源部门PC-2间的连通性。

PC>ping 172.16.2.1

Ping 172.16.2.1: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

--- 172.16.2.1 ping statistics ---

5 packet(s) transmitted

0 packet(s) received

100.00% packet loss

可以观察到无法正常通信。即网络中存在故障，但此时网络管理员并不能确定故障位置，首先在PC-1上测试与网关设备R2间的连通性。

PC>ping 172.16.1.254

Ping 172.16.1.254: 32 data bytes, Press Ctrl\_C to break

From 172.16.1.254: bytes=32 seq=1 ttl=255 time=15 ms

From 172.16.1.254: bytes=32 seq=2 ttl=255 time<1 ms

From 172.16.1.254: bytes=32 seq=3 ttl=255 time<1 ms

From 172.16.1.254: bytes=32 seq=4 ttl=255 time=16 ms

From 172.16.1.254: bytes=32 seq=5 ttl=255 time=16 ms

--- 172.16.1.254 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/9/16 ms

通信正常，表明PC-1跟网关设备R2间的链路没有问题。

路由器是根据路由表来进行数据转发，所以在R2上使用**display ip routing-table**命令查看是否有PC-2所在网段的路由条目。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.254 Ethernet1/0/1

172.16.1.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

192.168.1.0/24 Direct 0 0 D 192.168.1.1 GigabitEthernet0/0/0

192.168.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

可以观察到，R2上没有任何通过RIP协议接收的路由信息，说明R1与R2间的RIP路由信息通告不正常，即接下来需要在R1与R2之间排障。

第一步，检查R1与R2所在直连链路上的物理接口状态是否正常。

[R2]display ip interface brief GigabitEthernet 0/0/0

\*down: administratively down

!down: FIB overload down

^down: standby

(l): loopback

(s): spoofing

(d): Dampening Suppressed

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 192.168.1.1/24 up up

[R1]display ip interface brief GigabitEthernet 0/0/0

\*down: administratively down

!down: FIB overload down

^down: standby

(l): loopback

(s): spoofing

(d): Dampening Suppressed

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 192.168.1.2/24 up up

可以观察到，物理接口工作正常。“Physical”为UP，即接口的物理状态处于正常启动的状态，“Protocol”为UP，即接口的链路协议状态处于正常启动的状态。

在R2上测试与R1间直连链路的连通性。

[R2]ping -c 1 192.168.1.2

PING 192.168.1.2: 56 data bytes, press CTRL\_C to break

Reply from 192.168.1.2: bytes=56 Sequence=1 ttl=255 time=70 ms

--- 192.168.1.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 70/70/70 ms

可以观察到，连通性没有问题，继续下一步排查。

第二步，检查直连链路上的接口所在网段是否在RIP中通告。

[R2]display rip 1

Public VPN-instance

RIP process : 1

……

Verify-source : Enabled

Networks :

192.168.1.0 172.16.0.0

Configured peers : None

……

[R1]display rip 1

Public VPN-instance

RIP process : 1

……

Verify-source : Enabled

Networks :

192.168.2.0 192.168.1.0

Configured peers : None

……

可以观察到，接口的网段都已经在RIP中通告，继续下一步排查。

第三步，检查R1，R2上的RIP协议发送版本号和本地接口接收的版本号是否匹配。

在R1、R2上查看相应运行在RIP协议下的接口的详细信息。

[R1]display rip 1 interface GigabitEthernet 0/0/0 verbose

GigabitEthernet0/0/0(192.168.1.2)

State : UP MTU : 500

Metricin : 15

Metricout : 1

Input : Enabled Output : Enabled

Protocol : RIPv2 Multicast

Send version : RIPv2 Multicast Packets

Receive version : RIPv2 Multicast and Broadcast Packets

Poison-reverse : Disabled

Split-Horizon : Enabled

Authentication type : Simple

Replay Protection : Disabled

[R2]display rip 1 interface GigabitEthernet 0/0/0 verbose

GigabitEthernet0/0/0(192.168.1.1)

State : UP MTU : 500

Metricin : 0

Metricout : 1

Input : Enabled Output : Enabled

Protocol : RIPv2 Multicast

Send version : RIPv2 Multicast Packets

Receive version : RIPv2 Multicast and Broadcast Packets

Poison-reverse : Disabled

Split-Horizon : Enabled

Authentication type : None

Replay Protection : Disabled

可以观察到，双方的发送版本号和本地接口接收的版本号匹配，继续下一步排查。

第四步，由于目前在R2上没有接收到R1发送过来的路由信息，所以在R2上的入接口检查是否配置了**[undo rip input](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_input.html)**、**[silent-interface](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/silent-interface_rip.html)**命令。

查看R2入接口GE 0/0/0的配置信息。

[R2]display current-configuration interface GigabitEthernet 0/0/0

#

interface GigabitEthernet0/0/0

ip address 192.168.1.1 255.255.255.0

rip authentication-mode simple cipher %$%$B""x"|=aa$$Ys\*OM$OA&\*pSf%$%$

#

Return

可以观察到，目前R2的入接口GE 0/0/0上并没有配置**undo rip input**、**silent-interface**命令。

第五步，检查是否在RIP进程中配置了filter-policy策略，过滤掉收到的RIP路由或不允许发送RIP路由。

[R2]rip

[R2-rip-1]display this

#

rip 1

version 2

network 172.16.0.0

network 192.168.1.0

[R1]rip

[R1-rip-1]display this

#

rip 1

version 2

network 192.168.1.0

network 192.168.2.0

可以观察到，并没有配置策略，继续下一步排查。

第六步，检查接口上是否已经开启水平分割，水平分割为默认开启。查看R1和R2相应接口上的RIP详细信息。

[R2]display rip 1 interface GigabitEthernet 0/0/0 verbose

GigabitEthernet0/0/0(192.168.1.1)

……

Poison-reverse : Disabled

Split-Horizon : Enabled

Authentication type : None

Replay Protection : Disabled

[R1]display rip 1 interface GigabitEthernet 0/0/0 verbose

GigabitEthernet0/0/0(192.168.1.2)

……

Poison-reverse : Disabled

Split-Horizon : Enabled

Authentication type : None

Replay Protection : Disabled

可以观察到，接口下没有关闭水平分割，继续下一步排查。

第七步，检查链路两端的接口认证方式是否匹配。使用**display rip 1 statistics interface**命令查看两端RIP接口上的统计信息。

[R1]display rip 1 statistics interface GigabitEthernet 0/0/0

GigabitEthernet0/0/0(192.168.1.2)

Statistical information Last min Last 5 min Total

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

Periodic updates sent 1 6 361

……

Bad routes received 0 0 0

Packet authentication failed 0 3 168

Packet send failed 0 0 0

[R2]display rip 1 statistics interface GigabitEthernet 0/0/0

GigabitEthernet0/0/0(192.168.1.1)

Statistical information Last min Last 5 min Total

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

Periodic updates sent 0 4 361

……

Bad routes received 0 0 0

Packet authentication failed 0 4 328

Packet send failed 0 0 0

观察到有认证失败的RIP报文，说明两端的RIP认证方式有问题。在双方路由器上执行**display current-configuration**查看配置信息。

[R2]display current-configuration

#

interface GigabitEthernet0/0/0

ip address 192.168.1.1 255.255.255.0

rip authentication-mode simple cipher %$%$B""x"|=aa$$Ys\*OM$OA&\*pSf%$%$

[R1]display current-configuration

#

interface GigabitEthernet0/0/0

ip address 192.168.1.2 255.255.255.0

发现R2的GE 0/0/0接口下配置了RIP认证，而R1的GE 0/0/0接口下没有配置认证。

进入R2的GE 0/0/0接口删除该认证命令

[R2]interface GigabitEthernet0/0/0

[R2-GigabitEthernet0/0/0]undo rip authentication-mode

配置完成后，使用**display ip routing-table protocol rip**，检查双方现在是否能够正常收发RIP路由。

[R2]display ip routing-table protocol rip

[R2]

[R1]display ip routing-table protocol rip

[R1]

发现仍然没有相关的RIP路由条目，继续下一步排查。

第八步，查看路由表中是否存在从其它协议获得的相同路由，查看R2的路由表信息。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.254 Ethernet1/0/1

172.16.1.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

192.168.1.0/24 Direct 0 0 D 192.168.1.1 GigabitEthernet0/0/0

192.168.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

可以观察到，并没从有其它路由协议获得相同的路由，继续下一步排查。

第九步，检查收到的路由度量值是否大于16。使用**display rip 1 route**命令查看。

[R1]display rip 1 route

Route Flags : R - RIP

A - Aging, G - Garbage-collect

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

Peer 192.168.1.1 on GigabitEthernet0/0/0

Destination/Mask Nexthop Cost Tag Flags Sec

172.16.1.0/24 192.168.1.1 16 0 RG 16

发现从R1接收到的172.16.1.0网段的路由条目度量值是16，该路由不可达，所以不会将该路由加到路由表中。注意，如果操作到该步骤时距离导入预配置的时间间隔过久，将会无法查看到该结果。

在R1上使用**display current-configuration | include rip**命令，查看包含字符串“rip”的所有配置信息。

[R1]display current-configuration | include rip

rip metricin 15

rip 1

发现R1上配置了**rip metricin 15**命令，将GE 0/0/0接口接收到的路由都加上15的度量值，再放入路由表中，导致172.16.1.0网段的路由条目的度量值是16。

进入GE 0/0/0接口，删除该命令。

[R1]interface GigabitEthernet0/0/0

[R1-GigabitEthernet0/0/0]undo rip metricin

配置完成后，在R1上使用**display ip routing-table**，检查路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 RIP 100 1 D 192.168.1.1 GigabitEthernet0/0/0

192.168.1.0/24 Direct 0 0 D 192.168.1.2 GigabitEthernet0/0/0

192.168.1.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以发现R1已经收到172.16.1.0网段的路由信息，接收到了R2通告的路由条目，即R1与R2之间的故障已排除完毕。

但是此时R1仍然没有R3上172.16.2.0网段的路由信息，说明R1与R3间的RIP路由信息通告不正常，即接下来需要在R1与R3之间排障。

## 排除R1与R3间的故障

第一步，检查R1与R3所在直连链路上的物理接口状态是否正常。

[R1]display ip interface brief

\*down: administratively down

^down: standby

(l): loopback

(s): spoofing

The number of interface that is UP in Physical is 2

The number of interface that is DOWN in Physical is 2

The number of interface that is UP in Protocol is 2

The number of interface that is DOWN in Protocol is 2

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 192.168.1.2/24 up up

GigabitEthernet0/0/1 192.168.2.2/24 \*down down

NULL0 unassigned up up(s)

可以观察到，物理接口工作不正常。接口的物理状态和链路协议状态都不正常，并且“Physical”状态为“\*down”，表示网络管理员在该接口执行了**shutdown**命令。

进入R1的GE 0/0/1接口，通过**display this**命令查看接口配置

[R1]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]display this

[V200R003C00]

#

interface GigabitEthernet0/0/1

shutdown

ip address 192.168.2.2 255.255.255.0

果然发现接口下配置了shutdown。在该接口配置undo shutdown命令。

[R1-GigabitEthernet0/0/1]undo shutdown

配置完成后，测试连通性。

[R1]ping 192.168.2.1

PING 192.168.2.1: 56 data bytes, press CTRL\_C to break

Reply from 192.168.2.1: bytes=56 Sequence=1 ttl=255 time=210 ms

Reply from 192.168.2.1: bytes=56 Sequence=2 ttl=255 time=60 ms

Reply from 192.168.2.1: bytes=56 Sequence=3 ttl=255 time=50 ms

Reply from 192.168.2.1: bytes=56 Sequence=4 ttl=255 time=20 ms

Reply from 192.168.2.1: bytes=56 Sequence=5 ttl=255 time=20 ms

--- 192.168.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 20/72/210 ms

现在连通性恢复正常。检查R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 RIP 100 1 D 192.168.1.1 GigabitEthernet0/0/0

192.168.1.0/24 Direct 0 0 D 192.168.1.2 GigabitEthernet0/0/0

192.168.1.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

发现没有收到相关的RIP路由条目，继续下一步排查。

第二步，检查直连链路上的接口所在网段是否在RIP中通告。

[R1]display rip 1

Public VPN-instance

RIP process : 1

……

Verify-source : Enabled

Networks :

192.168.2.0 192.168.1.0

Configured peers : None

……

[R3]display rip 1

Public VPN-instance

RIP process : 1

……

Verify-source : Enabled

Networks :

172.16.0.0

Configured peers : None

……

发现R3上没有通告 192.168.2.0网段。

在R3上添加**network 192.168.2.0**命令，通告接口GE 0/0/1所在网段。

[R3]rip 1

[R3-rip-1]network 192.168.2.0

配置完成后，检查双方是否能够收发RIP路由条目。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 12 Routes : 12

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 RIP 100 1 D 192.168.1.1 GigabitEthernet0/0/0

172.16.2.0/24 RIP 100 1 D 192.168.2.1 GigabitEthernet0/0/1

192.168.1.0/24 Direct 0 0 D 192.168.1.2 GigabitEthernet0/0/0

192.168.1.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.2.0/24 Direct 0 0 D 192.168.2.2 GigabitEthernet0/0/1

192.168.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.2.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.2.0/24 Direct 0 0 D 172.16.2.254 Ethernet1/0/1

172.16.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1 172.16.2.255/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

255.255.255.255/32Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，R1此时能够正常接收到R3的路由，但是R3上接收不到R1的路由。

第三步，检查R1，R2上的RIP协议发送版本号和本地接口接收的版本号是否匹配。

在R1、R3上查看相应RIP接口GE 0/0/0的详细信息。

[R1]display rip 1 interface GigabitEthernet 0/0/1 verbose

GigabitEthernet0/0/1(192.168.2.2)

State : DOWN MTU : 500

Metricin : 0

Metricout : 1

Input : Enabled Output : Enabled

Protocol : RIPv2 Multicast

Send version : RIPv2 Multicast Packets

Receive version : RIPv2 Multicast and Broadcast Packets

Poison-reverse : Disabled

Split-Horizon : Enabled

Authentication type : None

Replay Protection : Disabled

[R3]display rip 1 interface GigabitEthernet 0/0/1 verbose

GigabitEthernet0/0/1(192.168.2.1)

State : DOWN MTU : 500

Metricin : 0

Metricout : 1

Input : Disabled Output : Enabled

Protocol : RIPv2 Multicast

Send version : RIPv2 Multicast Packets

Receive version : RIPv2 Multicast and Broadcast Packets

Poison-reverse : Disabled

Split-Horizon : Enabled

Authentication type : None

Replay Protection : Disabled

可以观察到，双方的发送版本号和本地接口接收的版本号匹配，继续下一步排查。

第四步，由于目前在R3上没有接收到R1发送过来的路由信息，所以在R3上的入接口检查是否配置了**[undo rip input](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/rip_input.html)**、**[silent-interface](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/silent-interface_rip.html)**命令。

查看R3入接口GE 0/0/0的配置信息。

[R3]interface GigabitEthernet 1/0/1

[R3-GigabitEthernet0/0/1]display this

[V200R003C00]

#

interface GigabitEthernet1/0/1

ip address 192.168.2.1 255.255.255.0

undo rip input

发现R3的GE 0/0/1接口下配置**undo rip input**命令禁止了RIP报文的接收。

配置rip input命令使接口GE 0/0/1可以接收RIP报文。

[R3-GigabitEthernet0/0/1]rip input

配置完成后，检查双方是否能够正常收发RIP路由条目。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 12 Routes : 12

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 RIP 100 1 D 192.168.1.1 GigabitEthernet0/0/0 172.16.2.0/24 RIP 100 1 D 192.168.2.1 GigabitEthernet0/0/1

192.168.1.0/24 Direct 0 0 D 192.168.1.2 GigabitEthernet0/0/0

192.168.1.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.1.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

192.168.2.0/24 Direct 0 0 D 192.168.2.2 GigabitEthernet0/0/1

192.168.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.2.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 12 Routes : 12

Destination/Mask Proto Pre Cost Flags NextHop Interface

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

127.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 RIP 100 2 D 192.168.2.2 GigabitEthernet0/0/1

172.16.2.0/24 Direct 0 0 D 172.16.2.254 Ethernet1/0/1

172.16.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

172.16.2.255/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

192.168.1.0/24 RIP 100 1 D 192.168.2.2 GigabitEthernet0/0/1

192.168.2.0/24 Direct 0 0 D 192.168.2.1 GigabitEthernet0/0/1 192.168.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

192.168.2.255/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

255.255.255.255/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，双方能正常接收到对方RIP路由。

最后在PC-1上测试与 PC-2间的连通性。

PC>ping 172.16.2.1

Ping 172.16.2.1: 32 data bytes, Press Ctrl\_C to break

From 172.16.2.1: bytes=32 seq=1 ttl=125 time=93 ms

From 172.16.2.1: bytes=32 seq=2 ttl=125 time=31 ms

From 172.16.2.1: bytes=32 seq=3 ttl=125 time=16 ms

From 172.16.2.1: bytes=32 seq=4 ttl=125 time=16 ms

From 172.16.2.1: bytes=32 seq=5 ttl=125 time=15 ms

--- 172.16.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/34/93 ms

通信正常，至此故障排除完成。

# 思考

如果采用debug或者抓包的方式排错，与采用查看命令进行排错相比有什么优劣？

7.10 RIP的路由引入

# 原理概述

设计者在进行网络规划或设计时，一般都设计成仅运行一种路由协议，以降低网络复杂性，易于维护。但是，如果在网络升级、扩展或合并时，就可能造成在网络中同时运行几种不同的路由协议，这时就需要部署路由协议间的引入，使路由信息能够在不同协议间传递。

RIP支持不同路由协议的引入，包括直连路由、静态路由或其他动态路由协议。由于RIP的度量值是跳数且最大值不能超过16，所以在将其他路由协议引入至RIP时需要注意设置度量值，避免引入的路由度量值超过16。默认情况下，引入另一种协议或引入同种协议的不同进程时往往是把该协议或该进程的所有路由一起引入，可以在引入的同时通过设置策略来控制和过滤特定的路由信息。

# 实验目的

* + 掌握RIP路由引入的应用场景
  + 掌握在RIP中引入直连路由的配置
  + 掌握在RIP中引入静态路由的配置
  + 理解RIP抑制接口的使用场景

# 实验内容

A和B两家公司，R4是公司A网关，左侧连接的公司A内网；R1是公司B的网关路由器，右侧是公司B的内网。内网中R2连接财务部门，R3连接的研发部门，三台路由器运行RIP协议。财务部门和研发部门不希望接收到大量RIP的更新报文，通过把它们的网段当作外部网络引入到RIP中来实现。而在优化完公司B的RIP网络之后，要求公司B与公司A能够互相通信，现需要使用静态路由和路由引入技术使两家公司的网络能够互访。

# 实验拓扑

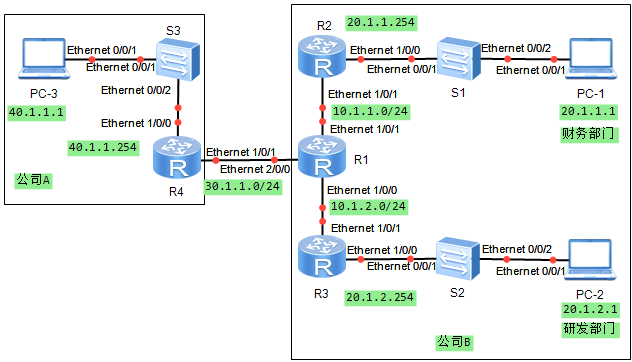


图7-19 RIP的路由引入拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | Ethernet 2/0/0 | 30.1.1.1 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| Ethernet 1/0/0 | 10.1.2.1 | 255.255.255.0 | N/A |
| R2(AR1220) | Ethernet 1/0/0 | 20.1.1.254 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |
| R3(AR1220) | Ethernet 1/0/0 | 20.1.2.254 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 10.1.2.2 | 255.255.255.0 | N/A |
| R4(AR1220) | Ethernet 1/0/1 | 30.1.1.2 | 255.255.255.0 | N/A |
| Ethernet 1/0/0 | 40.1.1.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 20.1.1.1 | 255.255.255.0 | 20.1.1.254 |
| PC-2 | Ethernet 0/0/1 | 20.1.2.1 | 255.255.255.0 | 20.1.2.254 |
| PC-3 | Ethernet 0/0/1 | 40.1.1.1 | 255.255.255.0 | 40.1.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping 30.1.1.2

PING 30.1.1.2: 56 data bytes, press CTRL\_C to break

Reply from 30.1.1.2: bytes=56 Sequence=1 ttl=255 time=30 ms

Reply from 30.1.1.2: bytes=56 Sequence=2 ttl=255 time=40 ms

Reply from 30.1.1.2: bytes=56 Sequence=3 ttl=255 time=30 ms

Reply from 30.1.1.2: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 30.1.1.2: bytes=56 Sequence=5 ttl=255 time=60 ms

--- 30.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/38/60 ms

其余直连网段的连通性测试省略。

## 搭建公司B的RIP网络

在公司B的路由器R1、R2和R3上配置RIPv2协议， 通告所有公司B内部网段。

[R1]rip 1

[R1-rip-1]undo summary

[R1-rip-1]version 2

[R1-rip-1]network 10.0.0.0

[R2]rip 1

[R2-rip-1]undo summary

[R2-rip-1]version 2

[R2-rip-1]network 10.0.0.0

[R2-rip-1]network 20.0.0.0

[R3-rip-1]rip 1

[R3-rip-1]undo summary

[R3-rip-1]version 2

[R3-rip-1]network 10.0.0.0

[R3-rip-1]network 20.0.0.0

配置完成后，查看R1路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.1 Ethernet1/0/1

10.1.1.1/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

10.1.2.0/24 Direct 0 0 D 10.1.2.1 Ethernet1/0/0

10.1.2.1/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

20.1.2.0/24 RIP 100 1 D 10.1.2.2 Ethernet1/0/0

20.1.1.0/24 RIP 100 1 D 10.1.1.2 Ethernet1/0/1

30.1.1.0/24 Direct 0 0 D 30.1.1.1 Ethernet2/0/0

30.1.1.1/32 Direct 0 0 D 127.0.0.1 Ethernet2/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0E

可以观察到，此时公司B的网关路由器R1已经成功接收到了内网中财务部20.1.2.0/24和研发部门20.1.1.0/24网段的路由条目。

## 优化公司B的RIP网络

公司B网络搭建完成后，网络管理员对网络进行维护。在R2的E0/0/0接口抓取数据包。

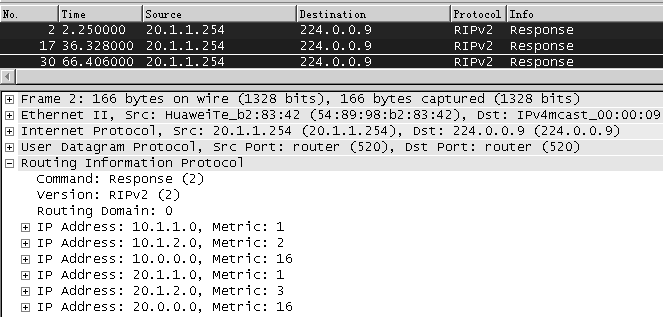


图7-20

可以观察到，此时R2上连接财务部门终端一侧的接口上会通告RIP路由信息，而这些RIP报文对终端PC而言是毫无用处的。

原因是使用**network**命令通告财务部门所在网段后，R2的该E 0/0/0接口就会收发RIP协议报文，不管对端设备是否利用。

为了使财务部门的终端不接收这些无用RIP更新报文，可以在R2的RIP进程中不使用**network**命令通告该网段，而采用引入直连路由的方式来代替，将财务部门的网段作为外部路由发布到公司RIP网络中。

在R2上使用命令**import-route**配置路由引入，指定引入的源路由协议为直连路由。

注意，在一台设备上配置路由引入时，需要保证被引入的路由条目已经存在于当前设备的路由表中。

[R2]rip 1

[R2-rip-1]undo network 20.0.0.0

[R2-rip-1]import-route direct

配置完成后，查看R1路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.1 Ethernet1/0/1

10.1.1.1/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

10.1.2.0/24 Direct 0 0 D 10.1.2.1 Ethernet1/0/0

10.1.2.1/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

20.1.1.0/24 RIP 100 1 D 10.1.1.2 Ethernet1/0/1

20.1.2.0/24 RIP 100 1 D 10.1.2.2 Ethernet1/0/0

30.1.1.0/24 Direct 0 0 D 30.1.1.1 Ethernet2/0/0

30.1.1.1/32 Direct 0 0 D 127.0.0.1 Ethernet2/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，R1上接收到了R2引入的20.1.1.0网段的路由信息。

再次在R2的E 1/0/0接口下抓包。

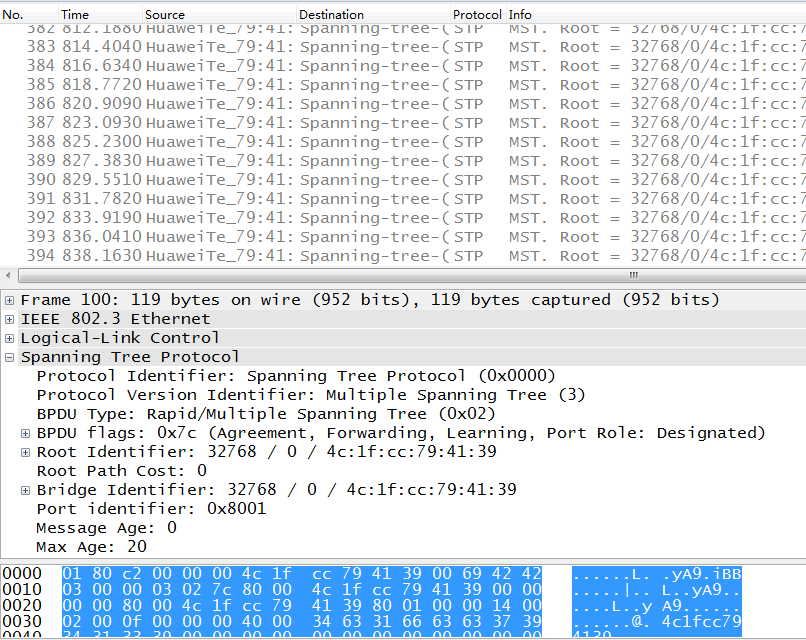


图7-21

可以观察到，现在该接口上没有发送任何RIP更新报文，。即此时已经完成优化，财务部门的终端不再收到与其无关的RIP更新报文。

可以在R2的E 1/0/1接口上抓取数据包观察区别。

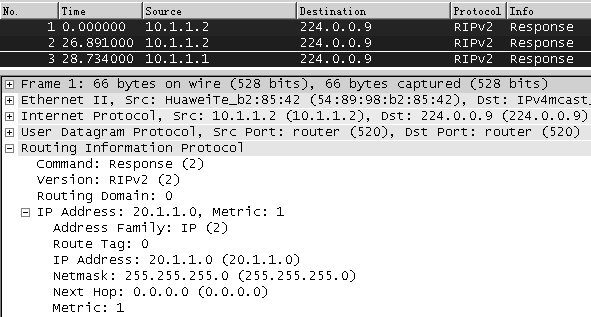


图7-22

可以观察到，在该E 1/0/1的接口上仍然正常发送RIP更新报文，将引入后的20.1.1.0/24网段通告出去。

同样在研发部门也会有相同的问题，收到对用户无用的RIP报文， 同样采用引入直连路由的方式来解决，此处省略。

## 连接公司A与公司B的网络

由于业务需要，需要公司A与B的网络能够互相访问。

在公司B的网关设备R1上配置目的为40.1.1.0网段的静态路由，并在RIP进程中引入该条静态路由，引入后公司B中RIP网络内的所有路由器会通过RIP协议自动学习到该路由。

[R1]ip route-static 40.1.1.0 255.255.255.0 30.1.1.2

[R1]rip 1

[R1-rip-1]import-route static

配置完成后，查看公司B设备R2，R3路由表。

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.2 Ethernet1/0/1

10.1.1.2/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

10.1.2.0/24 RIP 100 1 D 10.1.1.1 Ethernet1/0/1

20.1.1.0/24 Direct 0 0 D 20.1.1.254 Ethernet1/0/0

20.1.1.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

20.1.2.0/24 RIP 100 2 D 10.1.1.1 Ethernet1/0/1

40.1.1.0/24 RIP 100 1 D 10.1.1.1 Ethernet1/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

<R3>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 1 D 10.1.2.1 Ethernet1/0/1

10.1.2.0/24 Direct 0 0 D 10.1.2.2 Ethernet1/0/1

10.1.2.2/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

20.1.1.0/24 RIP 100 2 D 10.1.2.1 Ethernet1/0/1

20.1.2.0/24 Direct 0 0 D 20.1.2.254 Ethernet1/0/0

20.1.2.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

40.1.1.0/24 RIP 100 1 D 10.1.2.1 Ethernet1/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，此时公司B的内部路由器R2和R3上能够正常获得公司A内部网段的路由信息。但是此时公司A的路由器上仍然还没有公司B的任何路由信息。

在R4上配置一条默认路由，下一跳为R1。

[R4]ip route-static 0.0.0.0 0.0.0.0 30.1.1.1

配置完成后，查看R4路由表。

[R4]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 Static 60 0 D 30.1.1.1 Ethernet1/0/1

30.1.1.0/24 Direct 0 0 D 30.1.1.2 Ethernet1/0/1

30.1.1.2/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

40.1.1.0/24 Direct 0 0 D 40.1.1.254 Ethernet1/0/0

40.1.1.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，静态路由配置成功。

在PC-1上测试与PC-3间的连通性。

PC>ping 40.1.1.1

Ping 40.1.1.1: 32 data bytes, Press Ctrl\_C to break

From 40.1.1.1: bytes=32 seq=1 ttl=125 time=94 ms

From 40.1.1.1: bytes=32 seq=2 ttl=125 time=94 ms

From 40.1.1.1: bytes=32 seq=3 ttl=125 time=141 ms

From 40.1.1.1: bytes=32 seq=4 ttl=125 time=93 ms

From 40.1.1.1: bytes=32 seq=5 ttl=125 time=125 ms

--- 40.1.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 93/109/141 ms

可以观察到，PC-1与PC-3通信正常。至此，公司A和B之间可以正常通信。

# 思考

使用**network**命令方式通告路由，与路由引入的方式通告路由有什么区别？

1. OSPF

8.1 OSPF单区域配置

# 原理概述

为了弥补距离矢量路由协议的不足，IETF组织于20世纪80年代末开发了一种基于链路状态的内部网关协议，OSPF（Open Shortest Path First）。

最初的OSPF规范体现在RFC 113中，这个第1版( OSPFv1 )很快被进行了重大改进的版本所代替，新版本体现在RFC 1247文档中，称为OSPFv2，版本2在稳定性和功能性方面的做出了很大的改进。现在IPv4网络中所使用的都是OSPFv2。

OSPF作为基于链路状态的协议，具有收敛快、路由无环、扩展性好等优点，被快速接受并广泛使用。链路状态算法路由协议互相通告的是链路状态信息，每台路由器都将自己的链路状态信息（包含接口的IP地址和[子网掩码](http://baike.baidu.com/view/878.htm" \t "_blank)、[网络类型](http://baike.baidu.com/view/206578.htm)、该链路的开销等）发送给其他路由器，并在网络中泛洪，当每台路由器收集到网络内所有链路状态信息后，就能拥有整个网络的拓扑情况，然后根据整网拓扑情况运行SPF算法，得出到所有网段的最短路径。

OSPF支持区域的划分，区域是从逻辑上将路由器划分为不同的组，每个组用区域号（Area ID）来标识。一个网段（链路）只能属于一个区域，或者说每个运行OSPF的接口必须指明属于哪一个区域。区域0为骨干区域，骨干区域负责在非骨干区域之间发布区域间的路由信息，在一个OSPF区域中只能有一个骨干区域。

# 实验目的

* + 掌握OSPF单区域的配置方法
  + 理解OSPF单区域的应用场景
  + 掌握查看OSPF邻居状态的方法

# 实验内容

本实验模拟企业网络场景，该公司有三大办公区，每个办公区放置了一台路由器，R1放在办公区A，A区经理的PC-1直接连接R1；R2放在办公区B，B区经理的PC-2直接连接到R2；R3放在办公区C，C区经理的PC-3直接连接到R3；三台路由器都互相直连，为了能使整个公司网络互相通信，需要在所有路由器上部署路由协议，考虑到公司未来的发展（部门的增加和分公司的成立），为了适应不断扩展的网络的需求，公司在所有路由器上部署OSPF协议，且现在所有路由器都属于骨干区域。

# 实验拓扑



图8-1 OSPF单区域配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 172.16.10.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 172.16.20.1 | 255.255.255.0 | N/A |
| GE 0/0/2 | 172.16.1.254 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 172.16.10.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 172.16.30.2 | 255.255.255.0 | N/A |
| GE 0/0/2 | 172.16.2.254 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 172.16.20.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 172.16.30.3 | 255.255.255.0 | N/A |
| GE 0/0/2 | 172.16.3.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 172.16.1.1 | 255.255.255.0 | 172.16.1.254 |
| PC-2 | Ethernet 0/0/1 | 172.16.2.1 | 255.255.255.0 | 172.16.2.254 |
| PC-3 | Ethernet 0/0/1 | 172.16.3.1 | 255.255.255.0 | 172.16.3.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 172.16.20.3

PING 172.16.20.3: 56 data bytes, press CTRL\_C to break

Reply from 172.16.20.3: bytes=56 Sequence=1 ttl=255 time=20 ms

Reply from 172.16.20.3: bytes=56 Sequence=2 ttl=255 time=20 ms

Reply from 172.16.20.3: bytes=56 Sequence=3 ttl=255 time=10 ms

Reply from 172.16.20.3: bytes=56 Sequence=4 ttl=255 time=10 ms

Reply from 172.16.20.3: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 172.16.20.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/14/20 ms

其余直连网段的连通性测试省略。

## 部署单区域OSPF网络

首先使用命令**ospf**创建并运行OSPF。

<R1>system-view

[R1]ospf 1

其中，1代表的是进程号，如果没有写明进程号，默认是1。

接着使用命令**area**创建区域并进入OS

PF区域视图，输入要创建的区域ID。由于本实验为OSPF单区域配置，所以使用骨干区域，即区域0即可。

[R1-ospf-1]area 0

再使用**network**命令来指定运行OSPF协议的接口和接口所属的区域。本实验中R1上的三个物理接口都需要指定。配置中需注意，尽量精确匹配所通告的网段。

[R1-ospf-1-area-0.0.0.0]network 172.16.10.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 172.16.20.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

配置完成后使用命令**display ospf interface**检查OSPF接口通告是否正确。

[R1]display ospf interface

OSPF Process 1 with Router-ID 172.16.1.254

Interfaces

Area: 0.0.0.0 (MPLS TE not enabled)

IP Address Type State Cost Pri DR BDR

172.16.1.254 Broadcast DR 1 1 172.16.1.254 0.0.0.0

172.16.10.1 Broadcast DR 1 1 172.16.10.1 0.0.0.0

172.16.20.1 Broadcast DR 1 1 172.16.20.1 0.0.0.0

可以观察到本地OSPF进程使用的Router-ID是172.16.1.254。在此进程下，有三个接口加入了OSPF进程。网络类型为以太网默认的广播网络类型。State为该接口当前的状态，显示为DR状态，即表示为这三个接口在它们所在的网段中都被选举为DR。

接下来在R2和R3上做相应配置，配置方法和R1相同，不再赘述。

<R2>system-view

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 172.16.10.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 172.16.30.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

<R3>system-view

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 172.16.20.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 172.16.30.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

## 检查OSPF单区域的配置结果

以R1为例使用**display ospf peer**命令查看OSPF邻居状态。

<R1>display ospf peer

OSPF Process 1 with Router-ID 172.16.1.254

Neighbors

Area 0.0.0.0 interface 172.16.10.1(GigabitEthernet0/0/0)'s neighbors

Router-ID: 172.16.2.254 Address: 172.16.10.2

State: Full Mode:Nbr is Master Priority: 1

DR: 172.16.10.1 BDR: 172.16.10.2 MTU: 0

Dead timer due in 35 sec

Retrans timer interval: 5

Neighbor is up for 00:07:38

Authentication Sequence: [ 0 ]

Neighbors

Area 0.0.0.0 interface 172.16.20.1(GigabitEthernet0/0/1)'s neighbors

Router-ID: 172.16.3.254 Address: 172.16.20.3

State: Full Mode:Nbr is Master Priority: 1

DR: 172.16.20.1 BDR: 172.16.20.3 MTU: 0

Dead timer due in 29 sec

Retrans timer interval: 5

Neighbor is up for 00:04:14

Authentication Sequence: [ 0 ]

通过这条命令，可以查看很多内容。例如通过Router-ID可以查看邻居的路由器标识，通过Address可以查看邻居的OSPF接口IP地址，通过State可以查看目前与该路由器的OSPF邻居状态，通过Priority可以查看当前该邻居OSPF接口的DR优先级等等。

使用**display ip routing-table protocol ospf**命令查看R1上的OSPF路由表。

<R1>display ip routing-table protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 3 Routes : 4

OSPF routing table status : <Active>

Destinations : 3 Routes : 4

Destination/Mask Proto Pre Cost Flags NextHop Interface

172.16.2.0/24 OSPF 10 2 D 172.16.10.2 GigabitEthernet0/0/0

172.16.3.0/24 OSPF 10 2 D 172.16.20.3 GigabitEthernet0/0/1

172.16.30.0/24 OSPF 10 2 D 172.16.10.2 GigabitEthernet0/0/0

OSPF 10 2 D 172.16.20.3 GigabitEthernet0/0/1

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

通过此命令可以观察到，“Destination/Mask”标识了目的网段的前缀及掩码，“Proto”标识了此路由信息是通过OSPF协议获取到的，“Pre”标识了路由优先级，“Cost”标识了开销值，“NextHop”标识了下一跳地址，“Interface”标识了此前缀的出接口。

此时R1的路由表中已经拥有了去往网络中所有其他网段的路由条目。R2与R3上的现象一样，此处不再赘述。

在PC-1上用**ping**命令测试与PC-3间的连通性。

PC>ping 172.16.3.1

Ping 172.16.3.1: 32 data bytes, Press Ctrl\_C to break

From 172.16.3.1: bytes=32 seq=1 ttl=126 time=31 ms

From 172.16.3.1: bytes=32 seq=2 ttl=126 time=32 ms

From 172.16.3.1: bytes=32 seq=3 ttl=126 time=15 ms

From 172.16.3.1: bytes=32 seq=4 ttl=126 time=16 ms

From 172.16.3.1: bytes=32 seq=5 ttl=126 time=16 ms

--- 172.16.3.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/22/32 ms

通信正常，其他测试省略。

# 思考

请列举链路状态协议与距离矢量路由协议的相同点与不同点。

8.2 OSPF多区域配置

# 原理概述

在OSPF单区域中，每台路由器都需要收集其他所有路由器的链路状态信息，如果网络规模不断扩大，链路状态信息也会随之不断的增多，这将使得单台路由器上链路状态数据库非常庞大，导致路由器负担加重，也不便于维护管理。为了解决上述问题，OSPF协议可以将整个自治系统划分为不同的区域（Area），就像一个国家的国土面积很大时，会把整个国家划分为不同的省份来管理一样。

区域是从逻辑上将路由器划分为不同的组，每个组用区域号（Area ID）来标识，区域0为骨干区域，就像一个国家必须有首都一样，OSPF必须有骨干区域，且只能有一个，其他区域为非骨干区。一台路由器的不同接口可以属于不同的区域，但是同一网段（链路）必须属于同一区域。

链路状态信息只在区域内部泛洪，区域之间传递的只是路由条目而非链路状态信息，因此大大减小了路由器的负担。当一台路由器属于不同区域时称它为区域边界路由器（Area Border Router），负责传递区域间路由信息。区域间的路由信息传递类似距离矢量算法，为了防止区域间产生环路，所有非骨干区域之间的路由信息必须经过骨干区域，也就是非骨干区域必须和骨干区域相连，且非骨干区域之间不能直接进行路由信息交互。

# 实验目的

* + 理解配置OSPF多区域的使用场景
  + 掌握配置OSPF多区域的方法
  + 理解OSPF 区域边界路由器（ABR）的工作特点

# 实验内容

本实验模拟企业网络场景，R1，R2，R3，R4为企业总部核心区域设备，属于区域0，R5属于新增分支机构A的网关设备，R6属于新增分支机构B的网关设备。PC-1和PC-2分别属于分支机构A和B，PC-3和PC-4属于总部管理员登设备，用于管理网络。

在该网络中，如果设计方案采用单区域配置，则会导致单一区域LSA数目过于庞大，导致路由器开销过高，SPF算法运算过于频繁。因此网络管理员选择配置多区域方案进行网络配置，将两个新分支运行在不同的OSPF区域中，R5属于区域1，R6属于区域2。

# 实验拓扑



图8-2 OSPF多区域配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | 10.0.1.254 |
| PC-2 | Ethernet 0/0/1 | 10.0.2.1 | 255.255.255.0 | 10.0.2.254 |
| PC-3 | Ethernet 0/0/1 | 10.0.3.1 | 255.255.255.0 | 10.0.3.254 |
| PC-4 | Ethernet 0/0/1 | 10.0.4.1 | 255.255.255.0 | 10.0.4.254 |
| R1(AR2240) | GE 0/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.13.1 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.15.1 | 255.255.255.0 | N/A |
| R2(AR2240) | GE 0/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.24.2 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.26.2 | 255.255.255.0 | N/A |
| R3(AR2240) | GE 0/0/0 | 10.0.34.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.13.3 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.35.3 | 255.255.255.0 | N/A |
| Ethernet 4/0/0 | 10.0.3.254 | 255.255.255.0 | N/A |
| R4(AR2240) | GE 0/0/0 | 10.0.34.4 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.24.4 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.46.4 | 255.255.255.0 | N/A |
| Ethernet 4/0/0 | 10.0.4.254 | 255.255.255.0 | N/A |
| R5(AR2240) | GE 0/0/0 | 10.0.15.5 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.35.5 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.1.254 | 255.255.255.0 | N/A |
| R6(AR2240) | GE 0/0/0 | 10.0.26.6 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.46.6 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.2.254 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用ping命令检测各直连链路的连通性。

<R1>ping 10.0.15.5

PING 10.0.15.5: 56 data bytes, press CTRL\_C to break

Reply from 10.0.15.5: bytes=56 Sequence=1 ttl=255 time=210 ms

Reply from 10.0.15.5: bytes=56 Sequence=2 ttl=255 time=60 ms

Reply from 10.0.15.5: bytes=56 Sequence=3 ttl=255 time=60 ms

Reply from 10.0.15.5: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 10.0.15.5: bytes=56 Sequence=5 ttl=255 time=30 ms

--- 10.0.15.5 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/78/210 ms

测试通过，其余直连网段的连通性测试省略。

## 配置骨干区域路由器

在公司总部路由器R1，R2，R3，R4上创建OSPF进程，并在骨干区域0视图下通告总部各网段。

<R1>system-view

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

<R2>system-view

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

<R3>system-view

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.3.0 0.0.0.255

<R4>system-view

[R4]ospf 1

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R4-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

[R4-ospf-1-area-0.0.0.0]network 10.0.4.0 0.0.0.255

配置完成后，测试总部内两台PC间的连通性 。

PC>ping 10.0.4.1

Ping 10.0.4.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.4.1: bytes=32 seq=1 ttl=124 time=15 ms

From 10.0.4.1: bytes=32 seq=2 ttl=124 time=16 ms

From 10.0.4.1: bytes=32 seq=3 ttl=124 time=31 ms

From 10.0.4.1: bytes=32 seq=4 ttl=124 time=15 ms

From 10.0.4.1: bytes=32 seq=5 ttl=124 time=32 ms

--- 10.0.4.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/21/32 ms

已经可以正常通信，骨干区域路由器配置完成。

## 配置非骨干区域路由器

在分支A的路由器R5上创建OSPF进程，创建并进入区域1，并通告分支A的相应网段。

<R5>system-view

[R5]ospf 1

[R5-ospf-1]area 1

[R5-ospf-1-area-0.0.0.1]network 10.0.15.0 0.0.0.255

[R5-ospf-1-area-0.0.0.1]network 10.0.35.0 0.0.0.255

[R5-ospf-1-area-0.0.0.1]network 10.0.1.0 0.0.0.255

在R1和R3上也创建并进入区域1视图，将与R5相连的接口进行通告。

<R1>system-view

[R1]ospf 1

[R1-ospf-1]area 1

[R1-ospf-1-area-0.0.0.1]network 10.0.15.0 0.0.0.255

<R3>system-view

[R3]ospf 1

[R3-ospf-1]area 1

[R3-ospf-1-area-0.0.0.1]network 10.0.15.0 0.0.0.255

配置完成后，查看OSPF邻居状态。

[R5]display ospf peer

OSPF Process 1 with Router-ID 10.0.15.5

Neighbors

Area 0.0.0.1 interface 10.0.15.5(GigabitEthernet0/0/0)'s neighbors

Router-ID: 10.0.12.1 Address: 10.0.15.1

State: Full Mode:Nbr is Slave Priority: 1

DR: 10.0.15.5 BDR: 10.0.15.1 MTU: 0

Dead timer due in 34 sec

Retrans timer interval: 5

Neighbor is up for 00:07:46

Authentication Sequence: [ 0 ]

Neighbors

Area 0.0.0.1 interface 10.0.35.5(GigabitEthernet0/0/1)'s neighbors

Router-ID: 10.0.34.3 Address: 10.0.35.3

State: Full Mode:Nbr is Master Priority: 1

DR: 10.0.35.5 BDR: 10.0.35.3 MTU: 0

Dead timer due in 36 sec

Retrans timer interval: 5

Neighbor is up for 00:05:33

Authentication Sequence: [ 0 ]

可以观察到，现在R5与R1和R3的OSPF邻居关系建立正常，都为Full状态。

使用命令**display ip routing-table protocol ospf**查看R5路由表中的OSPF路由条目。

[R5]display ip routing-table protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 6 Routes : 8

OSPF routing table status : <Active>

Destinations : 6 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.3.0/24 OSPF 10 2 D 10.0.35.3 GigabitEthernet0/0/1

10.0.4.0/24 OSPF 10 3 D 10.0.35.3 GigabitEthernet0/0/1

10.0.12.0/24 OSPF 10 2 D 10.0.15.1 GigabitEthernet0/0/0

10.0.13.0/24 OSPF 10 2 D 10.0.15.1 GigabitEthernet0/0/0

OSPF 10 2 D 10.0.35.3 GigabitEthernet0/0/1

10.0.24.0/24 OSPF 10 3 D 10.0.15.1 GigabitEthernet0/0/0

OSPF 10 3 D 10.0.35.3 GigabitEthernet0/0/1

10.0.34.0/24 OSPF 10 2 D 10.0.35.3 GigabitEthernet0/0/1

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

可以观察到，除OSPF区域2内的路由外，相关OSPF路由条目都已经获得。在拓扑中，R1和R3这两台连接不同区域的路由器称为ABR，即区域边界路由器，该类路由器设备可以同时属于两个以上的区域，但其中至少一个端口必须在骨干区域内。ABR是用来连接骨干区域和非骨干区域的，其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。

使用命令**display ospf lsdb**查看R5的OSPF链路状态数据库信息。

<R5>display ospf lsdb

OSPF Process 1 with Router-ID 10.0.15.5

Link State Database

Area: 0.0.0.1

Type LinkState ID AdvRouter Age Len Sequence Metric

Router 10.0.12.1 10.0.12.1 1151 36 80000007 1

Router 10.0.34.3 10.0.34.3 1140 36 80000007 1

Router 10.0.15.5 10.0.15.5 1124 60 8000000C 1

Network 10.0.35.3 10.0.34.3 1140 32 80000004 0

Network 10.0.15.5 10.0.15.5 1145 32 80000005 0

Sum-Net 10.0.34.0 10.0.34.3 1137 28 80000005 1

Sum-Net 10.0.34.0 10.0.12.1 595 28 80000001 2

Sum-Net 10.0.13.0 10.0.12.1 1168 28 80000004 1

Sum-Net 10.0.13.0 10.0.34.3 1181 28 80000004 1

Sum-Net 10.0.12.0 10.0.12.1 1180 28 80000004 1

Sum-Net 10.0.12.0 10.0.34.3 516 28 80000001 2

Sum-Net 10.0.3.0 10.0.34.3 1179 28 80000004 1

Sum-Net 10.0.3.0 10.0.12.1 1125 28 80000004 2

可以观察到，关于其他区域的路由条目都是通过“Sum-Net”这类LSA获得，而这类LSA是不参与本区域的SPF算法运算的。

对公司另一分部B的路由器R6，和相应ABR设备R2，R4也做同样的配置。

<R6>system-view

[R6]ospf 1

[R6-ospf-1]area 2

[R6-ospf-1-area-0.0.0.1]network 10.0.26.0 0.0.0.255

[R6-ospf-1-area-0.0.0.1]network 10.0.46.0 0.0.0.255

[R6-ospf-1-area-0.0.0.1]network 10.0.2.0 0.0.0.255

<R2>system-view

[R2]ospf 1

[R2-ospf-1]area 2

[R2-ospf-1-area-0.0.0.1]network 10.0.26.0 0.0.0.255

<R4>system-view

[R4]ospf 1

[R4-ospf-1]area 2

[R4-ospf-1-area-0.0.0.1]network 10.0.46.0 0.0.0.255

配置完成，查看R6 的OSPF路由条目。

<R6>display ip routing-table protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 9 Routes : 12

OSPF routing table status : <Active>

Destinations : 8 Routes : 11

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 OSPF 10 4 D 10.0.46.4 GigabitEthernet0/0/1

OSPF 10 4 D 10.0.26.2 GigabitEthernet0/0/0

10.0.3.0/24 OSPF 10 3 D 10.0.46.4 GigabitEthernet0/0/1

10.0.12.0/24 OSPF 10 2 D 10.0.26.2 GigabitEthernet0/0/0

10.0.13.0/24 OSPF 10 3 D 10.0.46.4 GigabitEthernet0/0/1

OSPF 10 3 D 10.0.26.2 GigabitEthernet0/0/0

10.0.15.0/24 OSPF 10 3 D 10.0.26.2 GigabitEthernet0/0/0

10.0.24.0/24 OSPF 10 2 D 10.0.26.2 GigabitEthernet0/0/0

OSPF 10 2 D 10.0.46.4 GigabitEthernet0/0/1

10.0.34.0/24 OSPF 10 2 D 10.0.46.4 GigabitEthernet0/0/1

10.0.35.0/24 OSPF 10 3 D 10.0.46.4 GigabitEthernet0/0/1

OSPF routing table status : <Inactive>

Destinations : 1 Routes : 1

可以观察到，可以正常接收到所有OSPF路由信息。

测试分支A和分支B的两台PC-1和PC-2连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.2.1: bytes=32 seq=1 ttl=124 time=32 ms

From 10.0.2.1: bytes=32 seq=2 ttl=124 time=47 ms

From 10.0.2.1: bytes=32 seq=3 ttl=124 time=31 ms

From 10.0.2.1: bytes=32 seq=4 ttl=124 time=31 ms

From 10.0.2.1: bytes=32 seq=5 ttl=124 time=31 ms

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/34/47 ms

可以观察到，现在通信正常。至此，OSPF多区域配置完成。

# 思考

在本实验中，如果现在公司总部配置的区域不是骨干区域0，而是其他非骨干区域，会有什么现象？

8.3 配置OSPF的认证

# 原理概述

OSPF支持报文验证功能，只有通过验证的报文才能接收，否则将不能正常建立邻居关系。OSPF协议支持2种认证方式，区域认证和链路认证。使用区域认证时，一个区域中所有的路由器在该区域下的认证模式和口令必须一致。OSPF链路认证相比区域认证更加灵活，可专门针对某个邻居设置单独的认证模式和密码。如果同时配置了接口认证和区域认证时，优先使用接口认证建立OSPF邻居。

每种认证方式又分为简单验证模式，MD5验证模式，和Keychain验证模式。

简单验证模式在数据传递过程中，认证密钥和密钥ID都是明文传输，很容易被截获。MD5验证模式下的密钥是经过MD5加密传输，相比简单验证模式更为安全。key-chain验证模式可以同时配置多个密钥，不同密钥可单独设置生效周期等。

# 实验目的

* + 理解OSPF认证的应用场景
  + 理解OSPF区域认证和链路认证的区别
  + 掌握配置OSPF区域认证的方法
  + 掌握配置OSPF链路认证的方法

# 实验内容

本实验模拟企业网络环境，R3，R5，R6属于公司总部骨干区域路由器，R2为ABR。公司分部路由器R1和R4都属于区域1，但分属不通部门，R1作为市场部门网关，R4作为财务部门网关。网络管理员在区域0和区域1上配置OSPF区域认证。其中区域0开启密文认证，区域1开启明文认证。为进一步提高该OSPF网络安全性，R2和R4上单独设置密钥，配置OSPF链路认证。

# 实验拓扑

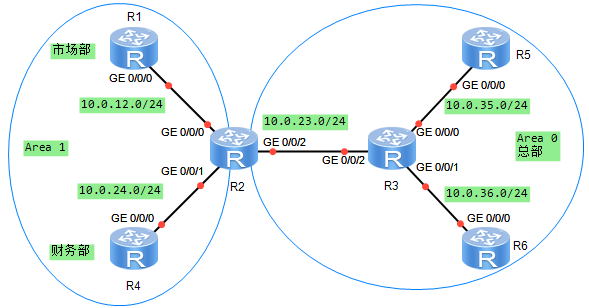


图8-3 配置OSPF的认证拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | Loopback 0 | 1.1.1.1 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| R2(AR2220) | Loopback 0 | 2.2.2.2 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.24.2 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.23.2 | 255.255.255.0 | N/A |
| R3(AR2220) | Loopback 0 | 3.3.3.3 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.35.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.36.3 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.23.3 | 255.255.255.0 | N/A |
| R4(AR2220) | Loopback 0 | 4.4.4.4 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.24.4 | 255.255.255.0 | N/A |
| R5(AR2220) | Loopback 0 | 5.5.5.5 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.35.5 | 255.255.255.0 | N/A |
| R6(AR2220) | Loopback 0 | 6.6.6.6 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.36.6 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 10.0.12.2

PING 10.0.12.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=120 ms

Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=90 ms

Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=60 ms

Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=40 ms

Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=30 ms

--- 10.0.12.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/68/120 ms

其余直连网段的连通性测试省略。

## 搭建OSPF网络

在公司总部和分部各台路由器上进行相关OSPF多区域配置。

<R1>system-view

[R1]ospf 1

[R1-ospf-1]area 1

[R1-ospf-1-area-0.0.0.1]network 10.0.12.0 0.0.0.255

[R1-ospf-1-area-0.0.0.1]network 1.1.1.1 0.0.0.0

<R2>system-view

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 2.2.2.2 0.0.0.0

[R2-ospf-1]area 1

[R2-ospf-1-area-0.0.0.1]network 10.0.12.0 0.0.0.255

[R2-ospf-1-area-0.0.0.1]network 10.0.24.0 0.0.0.255

<R3>system-view

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.35.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.36.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 3.3.3.3 0.0.0.0

<R4>system-view

[R4]ospf 1

[R4-ospf-1]area 1

[R4-ospf-1-area-0.0.0.1]network 10.0.24.0 0.0.0.255

[R4-ospf-1-area-0.0.0.1]network 4.4.4.4 0.0.0.0

<R5>system-view

[R5]ospf 1

[R5-ospf-1]area 0

[R5-ospf-1-area-0.0.0.0]network 10.0.35.0 0.0.0.255

[R5-ospf-1-area-0.0.0.0]network 5.5.5.5 0.0.0.0

<R6>system-view

[R6]ospf 1

[R6-ospf-1]area 0

[R6-ospf-1-area-0.0.0.0]network 10.0.36.0 0.0.0.255

[R6-ospf-1-area-0.0.0.0]network 6.6.6.6 0.0.0.0

其中每台设备上的环回口地址是为了后续实验中方便测试使用，所以需要通告到其所在区域。

配置完成后测试各设备上环回口的连通性。

<R1>ping 6.6.6.6

PING 6.6.6.6: 56 data bytes， press CTRL\_C to break

Reply from 6.6.6.6: bytes=56 Sequence=1 ttl=252 time=30 ms

Reply from 6.6.6.6: bytes=56 Sequence=2 ttl=252 time=30 ms

Reply from 6.6.6.6: bytes=56 Sequence=3 ttl=252 time=40 ms

Reply from 6.6.6.6: bytes=56 Sequence=4 ttl=252 time=30 ms

Reply from 6.6.6.6: bytes=56 Sequence=5 ttl=252 time=30 ms

--- 6.6.6.6 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/32/40 ms

可以正常通信，其他环回口的测试省略。

## 配置公司分部OSPF区域明文认证

网络管理员在公司分部的OSPF区域1中配置区域明文认证。

在R1上OSPF的区域1视图下使用命令**authentication-mode**指定该区域使用认证模式为simple，即简单验证模式，配置口令为huawei1，并配置**plain**参数。

配置**plain**参数后，可以使得在查看配置文件时，口令均已明文方式显示。如果不设置该参数的话，在查看配置文件时，默认会以密文方式显示口令，即无法查看到所配置的口令原文，这可以使非管理员用户在登录设备后无法查看到口令原文，从而提高安全性。

[R1]ospf 1

[R1-ospf-1]area 1

[R1-ospf-1-area-0.0.0.1]authentication-mode simple plain huawei1

[R1-ospf-1-area-0.0.0.1]display this

[V200R003C00]

#

area 0.0.0.1

authentication-mode simple plain huawei1

network 1.1.1.1 0.0.0.0

network 10.0.12.0 0.0.0.255

#

Return

可以观察到，此时以明文方式显示口令。

在R1上重新配置区域认证命令，并查看配置。

[R1-ospf-1-area-0.0.0.1]authentication-mode simple huawei1

[R1-ospf-1-area-0.0.0.1]display this

[V200R003C00]

#

area 0.0.0.1

authentication-mode simple cipher %$%$P!/9Ac}\_uERifpO}I^OJN<+@%$%$

network 1.1.1.1 0.0.0.0

network 10.0.12.0 0.0.0.255

#

Return

可以观察到，默认情况下，口令以密文形式显示。

配置完成，等待OSPF网络收敛之后，查看R1与R2的OSPF邻居。

<R1>display ospf peer brief

OSPF Process 1 with Router-ID 1

Peer Statistic Information

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

Area Id Interface Neighbor id State

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

可以观察到，现在R1与R2邻居关系中断了，原因是目前仅仅在R1上配置了认证，导致R1和R2间的OSPF认证不匹配。

继续配置该区域的另一台设备R2，必须要保证验证模式一致，口令也一致。

[R2]ospf 1

[R2-ospf-1]area 1

[R2-ospf-1-area-0.0.0.1]authentication-mode simple huawei1

配置完成后，等待一段时间，再次观察两者的邻居关系。

<R1>display ospf peer brief

OSPF Process 1 with Router-ID 10.0.12.1

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.1 GigabitEthernet0/0/0 10.0.12.2 Full

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

可以观察到，现在R1与R2的邻居关系状态恢复正常。

同理在R4上也做相同配置。

[R4]ospf 1

[R4-ospf-1]area 1

[R4-ospf-1-area-0.0.0.1]authentication-mode simple huawei1

配置完成后，在R2上查看OSPF邻居关系。

[R2]display ospf peer brief

OSPF Process 1 with Router-ID 10.0.12.2

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/2 10.0.23.3 Full

0.0.0.1 GigabitEthernet0/0/0 10.0.12.1 Full

0.0.0.1 GigabitEthernet0/0/1 10.0.24.4 Full

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

可以观察到，现在区域1中的邻居关系都建立正常。

## 配置公司总部OSPF区域密文认证

根据设计，网络管理员在公司总部OSPF区域0中配置区域密文认证。

在R2上配置OSPF Area 0区域认证，使用验证模式为**md5**，即MD5验证模式，验证字标识符为1，配置口令为huawei3。

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.3]authentication-mode md5 1 huawei3

继续在其他骨干路由器上做相同配置。注意，密文认证必须保证验证字标识符和口令完全一致认证才可以通过。

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]authentication-mode md5 1 huawei3

[R5]ospf 1

[R5-ospf-1]area 0

[R5-ospf-1-area-0.0.0.0]authentication-mode md5 1 huawei3

[R6]ospf 1

[R6-ospf-1]area 0

[R6-ospf-1-area-0.0.0.0]authentication-mode md5 1 huawei3

配置完成后，查看R3的OSPF邻居状态。

<R3>display ospf peer brief

OSPF Process 1 with Router-ID 10.0.23.3

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/2 10.0.12.2 Full

0.0.0.0 GigabitEthernet0/0/0 10.0.35.5 Full

0.0.0.0 GigabitEthernet0/0/1 10.0.36.6 Full

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

可以观察到，OSPF邻居状态建立正常，其他设备上的查看省略。

## 配置OSPF链路认证

在上面两个步骤中，使用了OSPF的区域认证方式配置了OSPF认证，使用链路认证方式配置可以达到同样的效果。如果采用链路认证的方式，就需要在同一OSPF的链路接口下都配置链路认证的命令，设置验证模式和口令等参数，而采用区域认证的方式时，在同一区域中，仅需在OSPF进程下的相应区域视图下配置一条命令来设置验证模式和口令即可，大大节省了配置量，所以在同一区域中如果有多台OSPF设备需要配置认证，建议选用区域认证的方式进行配置。

目前公司分部的OSPF区域中配置了简单模式的区域认证，为了进一步提升R2与R4之间的OSPF网络安全性，网络管理员需要在两台设备之间部署MD5验证模式的OSPF链路认证。

在R2的GE 0/0/1接口下使用命令**ospf authentication-mode**配置链路认证，配置使用MD5验证模式，验证字标识符为1，口令为huawei5。

[R2]interface GigabitEthernet 0/0/1

[R2-GigabitEthernet0/0/1]ospf authentication-mode md5 1 huawei5

配置完成后，等待一段时间，查看R2上的简要OSPF邻居信息。

[R2]display ospf peer brief

OSPF Process 1 with Router-ID 10.0.12.2

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/2 10.0.23.3 Full

0.0.0.1 GigabitEthernet0/0/0 10.0.12.1 Full

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

发现R2与R4间的OSPF邻居关系已经消失。虽然已经配置好区域认证，但是如果同时配置了接口认证和区域认证时，会优先使用接口验证建立OSPF邻居。所以R4在没有配置链路认证之前，R2与R4的邻居关系会因认证不匹配而无法建立。

同样在R4上配置链路，注意验证模式，标识符，口令都需要保持一致。

[R4]interface GigabitEthernet 0/0/0

[R4-GigabitEthernet0/0/1]ospf authentication-mode md5 1 huawei5

配置完成后，等待一段时间，再次查看R4的OSPF邻居信息。

<R4>display ospf peer brief

OSPF Process 1 with Router-ID 10.0.24.4

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.1 GigabitEthernet0/0/0 10.0.12.2 Full

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

可以观察到，邻居关系已经恢复正常。至此，OSPF链路认证配置完成。

# 思考

OSPF认证如果采用MD5验证模式，有没办法可以获取到其密钥内容？

8.4 OSPF被动接口配置

# 原理概述

OSPF被动接口也称抑制接口，成为被动接口后，将不会接收和发送OSPF报文。如果要使OSPF路由信息不被某一网络中的路由器获得且使本地路由器不接收网络中其他路由器发布的路由更新信息，即已运行在OSPF协议进程中的接口不与本链路上其余路由器建立邻居关系时，可通过配置被动接口来禁止此接口接收和发送OSPF报文。

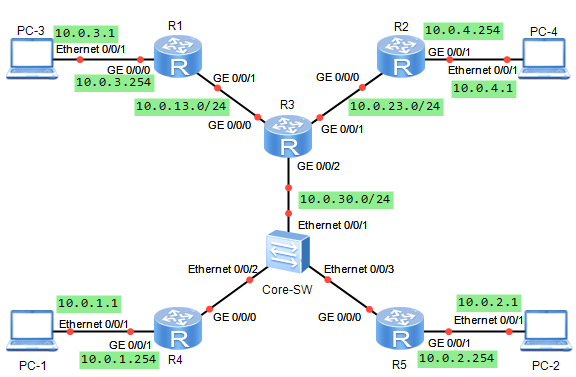
# 实验目的

* + 理解OSPF被动接口的应用场景
  + 掌握OSPF被动接口的配置方法
  + 理解OSPF被动接口的作用原理

# 实验内容

本实验模拟企业网络场景，有路由器R1，R2，R4与R5分属不同部门的网关设备，每台设备都连接着各部门的员工终端，公司整网运行OSPF协议，并都处于区域0中。员工终端上经常收到路由器发送的OSPF数据报文，而该报文对终端而言毫无用处，还占用了一定的链路带宽资源，并有可能引起安全风险，比如非法接入路由器做路由欺骗。现通告配置被动接口来实现阻隔OSPF报文，优化公司网络。

# 实验拓扑

图8-4 OSPF的被动接口拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | 10.0.1.254 |
| PC-2 | Ethernet 0/0/1 | 10.0.2.1 | 255.255.255.0 | 10.0.2.254 |
| PC-3 | Ethernet 0/0/1 | 10.0.3.1 | 255.255.255.0 | 10.0.3.254 |
| PC-4 | Ethernet 0/0/1 | 10.0.4.1 | 255.255.255.0 | 10.0.4.254 |
| R1(AR2220) | GE 0/0/0 | 10.0.3.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.13.1 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.0.23.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.4.254 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 10.0.13.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.23.3 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.30.3 | 255.255.255.0 | N/A |
| R4(AR2220) | GE 0/0/0 | 10.0.30.4 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.1.254 | 255.255.255.0 | N/A |
| R5(AR2220) | GE 0/0/0 | 10.0.30.5 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.2.254 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 10.0.3.1

PING 10.0.3.1: 56 data bytes， press CTRL\_C to break

Reply from 10.0.3.1: bytes=56 Sequence=1 ttl=128 time=30 ms

Reply from 10.0.3.1: bytes=56 Sequence=2 ttl=128 time=10 ms

Reply from 10.0.3.1: bytes=56 Sequence=3 ttl=128 time=1 ms

Reply from 10.0.3.1: bytes=56 Sequence=4 ttl=128 time=1 ms

Reply from 10.0.3.1: bytes=56 Sequence=5 ttl=128 time=10 ms

--- 10.0.3.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/10/30 ms

其余直连网段的连通性测试省略。

## 搭建OSPF网络

配置基本的OSPF，所有路由器的接口都运行在区域0内。

<R1>system-view

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 10.0.3.0 0.0.0.255

<R2>system-view

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.4.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

<R3>system-view

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.30.0 0.0.0.255

<R4>system-view

[R4]ospf 1

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 10.0.30.0 0.0.0.255

[R4-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

<R5>system-view

[R5]ospf 1

[R5-ospf-1]area 0

[R5-ospf-1-area-0.0.0.0]network 10.0.30.0 0.0.0.255

[R5-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255

配置完成之后，在PC上测试各网段间的连通性，以PC-3到PC-4为例。

PC>ping 10.0.4.1

Ping 10.0.4.1: 32 data bytes， Press Ctrl\_C to break

From 10.0.4.1: bytes=32 seq=1 ttl=125 time=16 ms

From 10.0.4.1: bytes=32 seq=2 ttl=125 time=16 ms

From 10.0.4.1: bytes=32 seq=3 ttl=125 time=15 ms

From 10.0.4.1: bytes=32 seq=4 ttl=125 time=32 ms

From 10.0.4.1: bytes=32 seq=5 ttl=125 time=15 ms

--- 10.0.4.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/18/32 ms

可以观察到，通信正常建立，其他测试省略。

在PC-1的接口E 0/0/1上抓包。

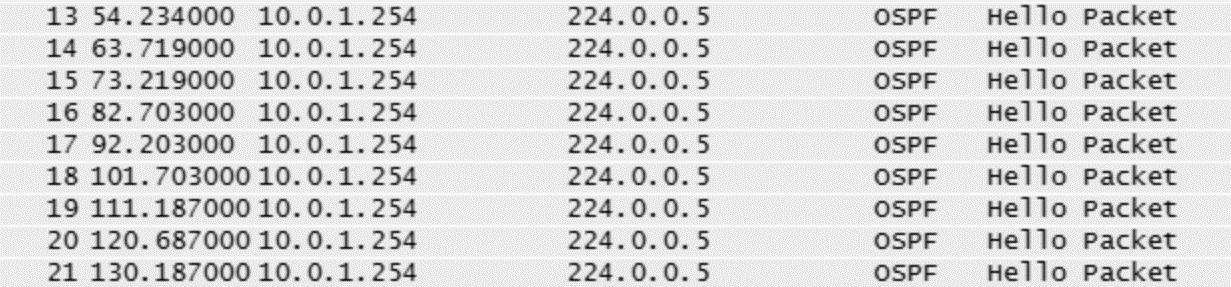


图8-5

发现该PC所在部门的网关路由器R4在不停的向这条线路发出OSPF的Hello报文尝试发现邻居，而对于PC而言，该报文是毫无用处的，同时也是不安全的，在OSPF的Hello报文中含有很多OSPF网络的重要信息，如果被恶意截取，容易出现安全隐患。

## 配置被动接口

现在网络管理员通过配置被动接口来优化连接终端的网络，使终端不再收到任何OSPF报文。

在R4的OSPF进程中，使用命令**silent-interface**禁止接口接收和发送OSPF报文。

[R4]ospf 1

[R4-ospf-1]silent-interface GigabitEthernet 0/0/1

配置完成后，再次观察抓包结果。

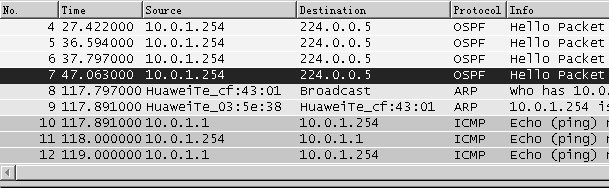


图8-6

发现OSPF的报文在47时刻之后已经不再周期性发送任何OSPF的Hello报文。

如果R4上有多个接口需要设置为被动接口，只有GE 0/0/1接口保持活动状态，可以通过以下命令简化配置。

[R4]ospf 1

[R4-ospf-1]silent-interface all

[R4-ospf-1]undo silent-interface GigabitEthernet 0/0/1

这两种方法都可以实现对GE 0/0/1接口进行被动接口的操作，区别在于第一种方法只是单独对某一个接口进行被动操作。而第二种是在对所有接口配置为被动接口后，再排除不需要配置为被动接口的接口。

同样在其他部门的网关路由器上进行相应配置，使得所有部门的终端都不再收到无关的OSPF报文。

[R1]ospf 1

[R1-ospf-1]silent-interface GigabitEthernet 0/0/0

[R2]ospf 1

[R2-ospf-1]silent-interface GigabitEthernet 0/0/1

[R5]ospf 1

[R5-ospf-1]silent-interface GigabitEthernet 0/0/1

## 验证被动接口

配置被动接口，该接口会禁止接收和发送OSPF报文，故假使在两台路由器间OSPF链路的接口上也做该配置，会导致OSPF邻居的无法建立。

以R5为例，将其GE 0/0/0接口配置为被动接口。

[R5]ospf 1

[R5-ospf-1]silent-interface GigabitEthernet 0/0/0

配置完成后，查看R5的OSPF邻居关系状态。

<R5>display ospf peer

OSPF Process 1 with Router-ID 10.0.30.5

可以观察到，此时R5的OSPF邻居全部消失。

查看R5上的OSPF路由条目。

<R5>display ip routing-table protocol ospf

可以观察到，所有的OSPF 路由条目都丢失。即验证了配置了被动接口后，OSPF报文不再转发，包括建立邻居和维护邻居的Hello报文。

在上一步骤中，R4的GE 0/0/1接口已经被配置了被动接口，那么配置该被动接口上的相关网段的路由信息能否正常被其他邻居路由器收到？

在R1上查看R4被动接口GE 0/0/1上所连网段的路由条目 10.0.1.0/24。

<R1>display ip routing-table 10.0.1.1

Route Flags: R - relay， D - download to fib

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

Routing Table : Public

Summary Count : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 OSPF 10 3 D 10.0.13.3 GigabitEthernet0/0/1

可以观察到，此时其他邻居路由器仍然可以收到该网段的路由条目。

被动接口特性为只是不再收发任何OSPF协议报文，但是被动接口所在网段的直连路由条目如果已经在OSPF中通告，那么也会被其他的OSPF邻居路由器接收到。

在PC-1上，测试与PC-4之间的连通性。

PC>ping 10.0.4.1

Ping 10.0.4.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.4.1: bytes=32 seq=1 ttl=125 time=32 ms

From 10.0.4.1: bytes=32 seq=2 ttl=125 time=46 ms

From 10.0.4.1: bytes=32 seq=3 ttl=125 time=47 ms

From 10.0.4.1: bytes=32 seq=4 ttl=125 time=32 ms

From 10.0.4.1: bytes=32 seq=5 ttl=125 time=31 ms

--- 10.0.4.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/37/47 ms

可以观察到，通信正常，完全不受影响。

# 思考

在本实验中，通过配置被动接口可以禁止OSPF收发Hello报文，是否还有其他办法也能实现？

8.5 理解OSPF Router-ID

# 原理概述

一些动态路由协议要求使用Router-ID作为路由器的身份标示，如果在启动这些路由协议时没有指定Router-ID，则缺省使用路由器全局下的路由管理Router-ID。

Router-ID选举规则为，如果通过命令Router-ID配置了Router-ID，则按照配置结果设置。在没有配置Router-ID的情况下，如果存在配置了IP地址的Loopback接口，则选择Loopback接口地址中最大的地址作为Router-ID。如果没有已配置IP地址的Loopback接口，则从其他接口的IP地址中选择最大的地址作为Router-ID（不考虑接口的UP/DOWN状态）。

当且仅当被选为Router-ID的接口IP地址被删除/修改，才触发重新选择过程，其他情况（例如接口处于DOWN状态；已经选取了一个非Loopback接口地址后又配置了一个Loopback接口地址；配置了一个更大的接口地址等）不触发重新选择的过程。

Router-ID改变之后，各协议需要通过手工执行reset命令才会重新选取新的Router-ID。

# 实验目的

* + 理解Router-ID的选举规则
  + 掌握OSPF手动配置Router-ID的方法
  + 理解OSPF中Router-ID必须唯一的意义

# 实验内容

本实验模拟企业网络环境，R1为部门A的网关设备，R3为部门B的网关设备，R4为部门C的网关设备，R2为企业核心路由器。现网络中运行OSPF协议实现全网互通，所有路由器运行在区域0内。网络管理员需要正确配置Router-ID以避免产生不必要的问题。

# 实验拓扑

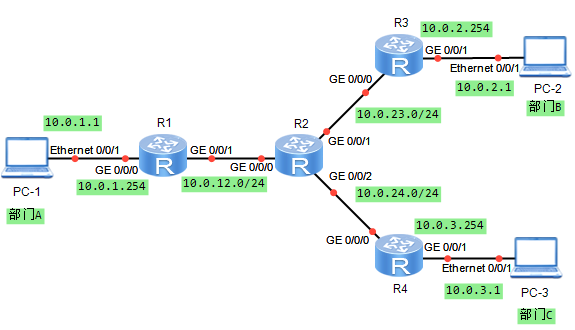


图8-7 理解OSPF的Router-ID拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | 10.0.1.254 |
| PC-2 | Ethernet 0/0/1 | 10.0.2.1 | 255.255.255.0 | 10.0.2.254 |
| PC-3 | Ethernet 0/0/1 | 10.0.3.1 | 255.255.255.0 | 10.0.3.254 |
| R1(AR2220) | Loopback 0 | 1.1.1.1 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.12.1 | 255.255.255.0 | N/A |
| R2(AR2220) | Loopback 0 | 2.2.2.2 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.23.2 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.24.2 | 255.255.255.0 | N/A |
| R3(AR2220) | Loopback 0 | 3.3.3.3 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.23.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.2.254 | 255.255.255.0 | N/A |
| R4(AR2220) | Loopback 0 | 4.4.4.4 | 255.255.255.255 | N/A |
| GE 0/0/0 | 10.0.24.4 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.3.254 | 255.255.255.0 | N/A |

# 实验步骤

## 验证Router-ID选举规则

在进行基本配置之前，在R1上使用命令**display router id**来查看当前设备上的Router-ID。

[R1]display Router ID

Router-ID:0.0.0.0

可以观察到，在设备没有配置任何接口时，Router-ID为0.0.0.0。

根据实验编址表，在R1的GE 0/0/1接口上配置IP地址10.0.12.1，GE 0/0/0接口配置IP地址10.0.1.254，配置环回接口0的地址1.1.1.1。

[R1]interface gigabitethernet 0/0/1

[R1-GigabitEthernet0/0/1]ip addres 10.0.12.1 24

[R1-GigabitEthernet0/0/1]interface gigabitethernet 0/0/0

[R1-GigabitEthernet0/0/0]ip addres 10.0.1.254 24

[R1-GigabitEthernet0/0/0]interface loopback 0

[R1-LoopBack0]ip addres 1.1.1.1 32

配置完成后，在R1上查看所有接口信息。

<R1>display ip interface brief

\*down: administratively down

^down: standby

(l): loopback

(s): spoofing

The number of interface that is UP in Physical is 4

The number of interface that is DOWN in Physical is 1

The number of interface that is UP in Protocol is 4

The number of interface that is DOWN in Protocol is 1

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 10.0.1.254/24 up up

GigabitEthernet0/0/1 10.0.12.1/24 up up

GigabitEthernet0/0/2 unassigned down down

LoopBack0 1.1.1.1/32 up up(s)

NULL0 unassigned up up(s)

可以观察到，目前所配置的接口及IP地址信息。

查看当前设备上的Router-ID。

[R1]display Router-ID

Router-ID:10.0.12.1

可以观察到当前设备上的全局Router-ID为10.0.12.1，而不是环回接口地址1.1.1.1，这是为什么？

原因是接口配置顺序会影响Router-ID的选举，因为设备上第一次配置的是物理接口的地址，该动作便会触发Router-ID的选举。而此刻，设备上也有且仅有该物理地址，所以该地址便会被Router-ID所使用，后续即使再配置了环回接口地址也不会使用。同理，如果第一次配置的是其他物理接口的地址，或者是环回接口的地址，都会被Router-ID所使用。

在R1上删除接口GE 0/0/1的IP地址，并再次查看此时设备的Router-ID。

[R1]interface gigabitethernet 0/0/1

[R1-GigabitEthernet0/0/1]undo ip address

[R1]display Router-ID

Router-ID:1.1.1.1

可以观察到，当删除当前Router-ID所使用的IP地址时，便会触发重新选举，按照环回接口优先的规则选择使用1.1.1.1作为Router-ID。

可以采用手动配置的方式强制指定R1的Router-ID为1.1.1.1。这样配置的优点是，即使该地址现在已经不是R1的任何接口的地址，也可以修改成为Router-ID（删除该环回接口也不会触发重新选举，验证省略）。

<R1>system-view

[R1]Router-ID 1.1.1.1

配置完成后，马上弹出以下信息。

Info: Router-ID has been modified, please reset the relative protocols manually to update the Router-ID.

该信息表示Router-ID已经被修改，请重启相应的路由协议进行更新。即当前全局配置的Router-ID已经被更新，如果目前设备上已经运行了OSPF协议，需要重置OSPF协议进程或者重启整台路由器才可以使得OSPF协议中的Router-ID也同步更新使用该新的全局Router-ID。需要使用命令**reset ospf process**来重置OSPF协议进程。

## 基本配置

根据实验编址表进行完成剩余基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 10.0.1.1

PING 10.0.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.0.1.1: bytes=56 Sequence=1 ttl=128 time=170 ms

Reply from 10.0.1.1: bytes=56 Sequence=2 ttl=128 time=70 ms

Reply from 10.0.1.1: bytes=56 Sequence=3 ttl=128 time=30 ms

Reply from 10.0.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms

Reply from 10.0.1.1: bytes=56 Sequence=5 ttl=128 time=10 ms

--- 10.0.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/62/170 ms

其余直连网段的连通性测试省略。

## 理解OSPF的Router-ID

在所有路由器上配置OSPF协议，并都运行在区域0内。使用**ospf router-id**命令来配置OSPF协议的私有Router-ID，如果不配置，则默认使用全局下的Router-ID。

注意区分设备全局下的Router-ID和路由协议的Router-ID的概念。如果在路由协议中没有配置Router-ID，就会默认使用路由器的全局Router-ID。如果配置，则可以和全局Router-ID不一致。

一般建议采用环回接口地址作为路由协议的Router-ID，优点为环回接口是逻辑接口，比物理接口更加稳定。在对网络操作时，网络管理员有可能误操作导致物理接口地址删除，或者改动，而环回接口则一般不会去改动。

[R1]ospf 1 router-id 1.1.1.1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

[R2]ospf 1 router-id 2.2.2.2

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

[R3]ospf 1 router-id 3.3.3.3

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255

[R4]ospf 1 router-id 4.4.4.4

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

[R4-ospf-1-area-0.0.0.0]network 10.0.3.0 0.0.0.255

配置完成后测试PC-1和PC-2间的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.2.1: bytes=32 seq=1 ttl=124 time=31 ms

From 10.0.2.1: bytes=32 seq=2 ttl=124 time=47 ms

From 10.0.2.1: bytes=32 seq=3 ttl=124 time=31 ms

From 10.0.2.1: bytes=32 seq=4 ttl=124 time=32 ms

From 10.0.2.1: bytes=32 seq=5 ttl=124 time=31 ms

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/34/47 ms

可以观察到，目前通信正常，其余PC间连通性测试省略。

现在修改R2的Router-ID为3.3.3.3，即R3的Router-ID，使R3和R2的Router-ID重叠，并重置协议进程使该配置生效。

<R2>ospf 1 router-id 3.3.3.3

<R2>reset ospf process

Warning: The OSPF process will be reset. Continue? [Y/N]:y

待协议收敛后，再次查看R2 的OSPF邻居信息。

<R2>display ospf peer

OSPF Process 1 with Router-ID 3.3.3.3

Neighbors

Area 0.0.0.0 interface 10.0.12.2(GigabitEthernet0/0/0)'s neighbors

Router-ID: 1.1.1.1 Address: 10.0.12.1

State: Full Mode:Nbr is Slave Priority: 1

DR: 10.0.12.1 BDR: 10.0.12.2 MTU: 0

Dead timer due in 40 sec

Retrans timer interval: 5

Neighbor is up for 00:01:36

Authentication Sequence: [ 0 ]

Neighbors

Area 0.0.0.0 interface 10.0.24.2(GigabitEthernet0/0/2)'s neighbors

Router ID: 4.4.4.4 Address: 10.0.24.4

State: Full Mode:Nbr is Master Priority: 1

DR: 10.0.24.4 BDR: 10.0.24.2 MTU: 0

Dead timer due in 40 sec

Retrans timer interval: 5

Neighbor is up for 00:00:40

Authentication Sequence: [ 0 ]

可以观察到到R2与R3的邻居关系消失。

测试PC-1与PC-2的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

0 packet(s) received

100.00% packet loss

网络已经发生故障，无法正常通信。验证了OSPF建立直连邻居关系时，Router-ID一定不能重叠。那么如果OSPF非直连邻居的Router-ID重叠会产生什么现象？

还原R2之前的配置，调整R4的Router-ID为3.3.3.3，与R3重叠。

<R2>ospf 1 router-id 2.2.2.2

<R2>reset ospf process

Warning: The OSPF process will be reset. Continue? [Y/N]:y

<R4>ospf 1 router-id 3.3.3.3

<R4>reset ospf process

Warning: The OSPF process will be reset. Continue? [Y/N]:y

配置完成后，查看R2的OSPF邻居状态。

<R2>display ospf peer brief

OSPF Process 1 with Router-ID 2.2.2.2

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/0 1.1.1.1 Full

0.0.0.0 GigabitEthernet0/0/1 3.3.3.3 Full

0.0.0.0 GigabitEthernet0/0/2 3.3.3.3 Full

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

发现R2有两个3.3.3.3的邻居，查看R2的路由表。

<R2>display ip routing-table protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 2 Routes : 2

OSPF routing table status : <Active>

Destinations : 2 Routes : 2

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 OSPF 10 2 D 10.0.12.1 GigabitEthernet0/0/0

10.0.3.0/24 OSPF 10 2 D 10.0.24.4 GigabitEthernet0/0/2

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

可以观察到，此时R2没有接收到R3上10.0.2.0/24网段的路由条目，即使路由器邻居关系建立正常，但也无法正常获取路由条目。

测试PC-1与PC-2的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

可以观察到，通信无法正常进行。这是因为R2认为是同一个OSPF邻居，但是LSA又不一致，造成链路状态数据库发送错误，无法计算出正确的路由信息。

综上所述，OSPF协议的Router-ID务必要在整个路由选择域内保持唯一。

# 思考

试问如果不同区域中的OSPF路由器的Router-ID重叠又会导致什么问题的产生？

8.6 OSPF的DR与BDR

# 原理概述

在OSPF的广播类型网络和NBMA类型网络中，如果网络中有n台路由器，若任意两台路由器之间都要建立邻接关系，则需要建立n\*(n-1)/2个邻接关系，即当路由器很多时，则需要建立和维护的邻接关系就很多，两两之间需要发送的报文也就很多，这会造成很多内容重复的报文在网络中传递，浪费了设备的带宽资源。因此在广播和NBMA类型网络中，OSPF协议定义了指定路由器DR（Designated Router），即所有其他路由器都只将各自的链路状态信息发送给DR，再由DR以组播方式发送至所有路由器，大大减少了OSPF数据包的发送。

但是如果DR由于某种故障而失效，此时网络中必须重新选举DR，并同步链路状态信息，这需要较长的时间。为了能够缩短这个过程，OSPF协议又定义了BDR（Backup Designated Router）的概念，作为DR路由器的备份，当DR路由器失效时，BDR成为DR，并再选择新的BDR路由器。其他非DR/BDR路由器都称为DR Other路由器。

每一个含有至少两个路由器的广播类型网络或NBMA类型网络都会选举一个DR和BDR。选举规则是首先比较DR优先级，优先级高者成为DR，次高的成为BDR。如果优先级相等，则Router-ID数值高的成为DR，次高的成为BDR。如果一台路由器的DR优先级为0，则不参与选举。需要注意的是：

DR是在某个广播或者NBMA网段内进行选举的，是针对路由器的接口而言的。某台路由器在一个接口上可能是DR，在另一个接口上有可能是BDR，或者是DR Other。

若DR、BDR已经选举完毕，人为修改任何一台路由器的DR优先级值为最大，也不会抢占成为新的DR或BDR，即OSPF的DR/BDR选举是非抢占的。

# 实验目的

* + 理解OSPF在哪种网络类型中会选举DR/BDR
  + 掌握OSPF DR/BDR的选举规则
  + 掌握如何更改设备接口上的DR优先级
  + 理解OSPF DR/BDR选举的非抢占特性

# 实验内容

某公司有四个部门，路由器R1连接的总经理办公室，路由器R2连接到人事部，R3连接的是开发部，R4连接的是市场部。四台路由器通过交换机S1互联，每台路由器都运行了OSPF路由协议，都运行在区域0内，使得公司内部各部门网络能够互相通信。由于路由器通过广播网络互连，OSPF会选举DR和BDR，现网络管理员要配置使得性能较好的R1成为DR，性能次之的R2成为BDR，而性能最差的R4不能参加DR和BDR的选举，由此来完成网络的优化。

# 实验拓扑

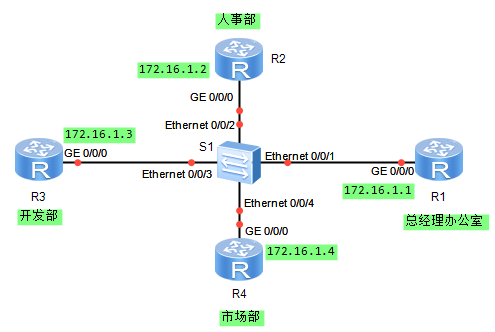


图8-8 理解OSPF的DR与BDR拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 172.16.1.1 | 255.255.255.0 | N/A |
| Loopback 0 | 1.1.1.1 | 255.255.255.255 | N/A |
| R2(AR2220) | GE 0/0/0 | 172.16.1.2 | 255.255.255.0 | N/A |
| Loopback 0 | 2.2.2.2 | 255.255.255.255 | N/A |
| R3(AR2220) | GE 0/0/0 | 172.16.1.3 | 255.255.255.0 | N/A |
| Loopback 0 | 3.3.3.3 | 255.255.255255 | N/A |
| R4(AR2220) | GE 0/0/0 | 172.16.1.4 | 255.255.255.0 | N/A |
| Loopback 0 | 4.4.4.4 | 255.255.255.255 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 172.16.1.4

PING 172.16.1.4: 56 data bytes, press CTRL\_C to break

Reply from 172.16.1.4: bytes=56 Sequence=1 ttl=255 time=100 ms

Reply from 172.16.1.4: bytes=56 Sequence=2 ttl=255 time=70 ms

Reply from 172.16.1.4: bytes=56 Sequence=3 ttl=255 time=70 ms

Reply from 172.16.1.4: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 172.16.1.4: bytes=56 Sequence=5 ttl=255 time=50 ms

--- 172.16.1.4 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/64/100 ms

其余直连网段的连通性测试省略。

## 搭建基本的OSPF网络

在公司网络中的四台路由器R1，R2，R3，R4上配置基础的OSPF网络配置。每台路由器使用各自的环回接口地址作为Router-ID，并且都运行在区域0内。

<R1>system-view

[R1]router id 1.1.1.1

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

<R2>system-view

[R2]router id 2.2.2.2

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

<R3>system-view

[R3]router id 3.3.3.3

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

<R4>system-view

[R4]router id 4.4.4.4

[R4]ospf 1

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

配置完成后，同时重启四台路由器上的OSPF进程，或者直接同时重启设备。

<R1>reset ospf process

<R2>reset ospf process

<R3>reset ospf process

<R4>reset ospf process

重置后再次检查OSPF邻居建立情况。使用命令**display ospf peer brief**进行查看。

<R1>display ospf peer brief

OSPF Process 1 with Router-ID 1.1.1.1

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/0 2.2.2.2 Full

0.0.0.0 GigabitEthernet0/0/0 3.3.3.3 Full

0.0.0.0 GigabitEthernet0/0/0 4.4.4.4 Full

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

可以观察到，R1此时已经和其他路由器成功建立起OSPF邻居关系。其他设备上的查看省略。

## 查看缺省情况下的DR/BDR状态

使用**display ospf peer**命令查看此时缺省情况下OSPF网络中的DR/BDR选举情况。

[R1]display ospf peer

OSPF Process 1 with Router-ID 1.1.1.1

Neighbors

Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors

Router-ID: 2.2.2.2 Address: 172.16.1.2

State: 2-Way Mode:Nbr is Master Priority: 1

DR: 172.16.1.4 BDR: 172.16.1.3 MTU: 0

Dead timer due in 35 sec

Retrans timer interval: 0

Neighbor is up for 00:00:00

Authentication Sequence: [ 0 ]

Router-ID: 3.3.3.3 Address: 172.16.1.3

State: Full Mode:Nbr is Master Priority: 1

DR: 172.16.1.4 BDR: 172.16.1.3 MTU: 0

Dead timer due in 34 sec

Retrans timer interval: 5

Neighbor is up for 00:00:48

Authentication Sequence: [ 0 ]

Router-ID: 4.4.4.4 Address: 172.16.1.4

State: Full Mode:Nbr is Master Priority: 1

DR: 172.16.1.4 BDR: 172.16.1.3 MTU: 0

Dead timer due in 34 sec

Retrans timer interval: 0

Neighbor is up for 00:00:46

Authentication Sequence: [ 0 ]

可以观察到在该广播网络中，此时R4为OSPF网络中的DR，R3为BDR。这是由于在缺省情况下，每台路由器上的DR优先级都为1，此时通过Router-ID的数值高低进行比较。

接下来在每台设备上的相关接口下使用命令**ospf network-type p2mp**修改OSPF的网络类型为点到多点。

<R1>system-view

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ospf network-type p2mp

<R2>system-view

[R2]interface GigabitEthernet 0/0/0

[R2-GigabitEthernet0/0/0]ospf network-type p2mp

<R3>system-view

[R3]interface GigabitEthernet 0/0/0

[R3-GigabitEthernet0/0/0]ospf network-type p2mp

<R4>system-view

[R4]interface GigabitEthernet 0/0/0

[R4-GigabitEthernet0/0/0]ospf network-type p2mp

配置完成后，在R1上再次观察此时OSPF的DR/BDR选举情况。

[R1]display ospf peer

OSPF Process 1 with Router-ID 1.1.1.1

Neighbors

Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors

Router-ID: 2.2.2.2 Address: 172.16.1.2

State: Full Mode:Nbr is Master Priority: 1

DR: None BDR: None MTU: 0

Dead timer due in 109 sec

Retrans timer interval: 0

Neighbor is up for 00:01:16

Authentication Sequence: [ 0 ]

Router-ID: 3.3.3.3 Address: 172.16.1.3

State: Full Mode:Nbr is Master Priority: 1

DR: None BDR: None MTU: 0

Dead timer due in 103 sec

Retrans timer interval: 0

Neighbor is up for 00:01:02

Authentication Sequence: [ 0 ]

Router-ID: 4.4.4.4 Address: 172.16.1.4

State: Full Mode:Nbr is Master Priority: 1

DR: None BDR: None MTU: 0

Dead timer due in 113 sec

Retrans timer interval: 0

Neighbor is up for 00:00:57

Authentication Sequence: [ 0 ]

可以观察到，DR/BDR都为None，验证了在点到多点的网络类型中不选举DR/BDR，同样在点到点网络中也是，这里不再赘述。

## 根据现网需求影响DR/BDR选举

现在根据需求，网络管理员要使得性能较好，处理能力较强的R1成为DR，性能次之的R2成为BDR，而性能最差的R4不能参加DR和BDR的选举，由此来完成网络的优化。

首先将OSPF网络类型还原为默认的广播网络类型。

<R1>system-view

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ospf network-type broadcast

<R2>system-view

[R2]interface GigabitEthernet 0/0/0

[R2-GigabitEthernet0/0/0]ospf network-type broadcast

<R3>system-view

[R3]interface GigabitEthernet 0/0/0

[R3-GigabitEthernet0/0/0]ospf network-type broadcast

<R4>system-view

[R4]interface GigabitEthernet 0/0/0

[R4-GigabitEthernet0/0/0]ospf network-type broadcast

配置完成后，修改R1上GE 0/0/0接口的DR优先级为100，R2为50，R4为0，R3保持缺省不变。

<R1>system-view

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ospf dr-priority 100

<R2>system-view

[R2]interface GigabitEthernet 0/0/0

[R2-GigabitEthernet0/0/0]ospf dr-priority 50

<R4>system-view

[R4]interface GigabitEthernet 0/0/0

[R4-GigabitEthernet0/0/0]ospf dr-priority 0

配置完成后，查看各路由器的DR/BDR选举情况。

[R1]display ospf peer

OSPF Process 1 with Router-ID 1.1.1.1

Neighbors

Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors

Router-ID: 2.2.2.2 Address: 172.16.1.2

State: 2-Way Mode:Nbr is Master Priority: 1

DR: 172.16.1.4 BDR: 172.16.1.3 MTU: 0

Dead timer due in 35 sec

Retrans timer interval: 0

Neighbor is up for 00:00:00

Authentication Sequence: [ 0 ]

Router-ID: 3.3.3.3 Address: 172.16.1.3

State: Full Mode:Nbr is Master Priority: 1

DR: 172.16.1.4 BDR: 172.16.1.3 MTU: 0

Dead timer due in 34 sec

Retrans timer interval: 5

Neighbor is up for 00:00:48

Authentication Sequence: [ 0 ]

Router-ID: 4.4.4.4 Address: 172.16.1.4

State: Full Mode:Nbr is Master Priority: 1

DR: 172.16.1.4 BDR: 172.16.1.3 MTU: 0

Dead timer due in 34 sec

Retrans timer interval: 0

Neighbor is up for 00:00:46

Authentication Sequence: [ 0 ]

发现此时的DR与BDR都没有改变，即验证了OSPF的DR/BDR选举是非抢占的。必须要在四台路由器上同时重启OSPF进程，或者重启路由器才能使得其重新正确选举。

同时重启四台路由器的OSPF进程，或直接同时重启设备。

重置后再次查看各路由器的DR/BDR选举状态。

<R1>display ospf peer

OSPF Process 1 with Router-ID 1.1.1.1

Neighbors

Area 0.0.0.0 interface 172.16.1.1(GigabitEthernet0/0/0)'s neighbors

Router-ID: 2.2.2.2 Address: 172.16.1.2

State: Full Mode:Nbr is Master Priority: 50

DR: 172.16.1.1 BDR: 172.16.1.2 MTU: 0

Dead timer due in 28 sec

Retrans timer interval: 5

Neighbor is up for 00:00:19

Authentication Sequence: [ 0 ]

Router-ID: 3.3.3.3 Address: 172.16.1.3

State: Full Mode:Nbr is Master Priority: 1

DR: 172.16.1.1 BDR: 172.16.1.2 MTU: 0

Dead timer due in 32 sec

Retrans timer interval: 5

Neighbor is up for 00:00:04

Authentication Sequence: [ 0 ]

Router-ID: 4.4.4.4 Address: 172.16.1.4

State: Full Mode:Nbr is Master Priority: 0

DR: 172.16.1.1 BDR: 172.16.1.2 MTU: 0

Dead timer due in 29 sec

Retrans timer interval: 5

Neighbor is up for 00:00:28

Authentication Sequence: [ 0 ]

此时发现在该广播网络中，R1为DR，R2为BDR，实现了网络的需求。

# 思考

在本实验步骤二中，基础的OSPF网络配置完毕后，为什么要同时重启下四台路由器上的OSPF进程？

8.7 OSPF开销值、协议优先级及计时器的修改

# 原理概述

由于路由器上可能同时运行多种动态路由协议，就存在各个路由协议之间路由信息共享和选择的问题。系统为每一种路由协议设置了不同的默认优先级，当在不同协议中发现同一条路由时，协议优先级高的将被优选。

如果没有直接配置OSPF接口的开销值，OSPF会根据该接口的带宽自动计算其开销值。计算公式为，接口开销=带宽参考值/接口带宽，取计算结果的整数部分作为接口开销值（当结果小于1时取1）。通过改变带宽参考值可以间接改变接口的开销值。

OSPF常见的计时器包括hello timer 和dead timer，分别决定了OSPF发送hello报文的间隔和保持邻居关系的计时器。缺省情况下，P2P、Broadcast类型接口发送Hello报文的时间间隔为10秒，邻居失效时间为40秒；P2MP、NBMA类型接口发送Hello报文的时间间隔为30秒；邻居失效时间为120秒。

# 实验目的

* + 掌握配置OSPF协议优先级的方法
  + 掌握配置OSPF开销的方法
  + 掌握配置OSPF hello timer的方法
  + 掌握配置OSPF dead timer的方法

# 实验内容

本实验模拟企业两个分支机构之间通过两条路径实现互联互通。R1为分支机构A网关设备，R4为分支机构B的网关设备。公司原网络为分支A与分支B通过R2进行通信，设备之间运行的是OSPF协议，都属于区域0。后因带宽需要增大，两机构之间决定新增一条带宽更大路径，通过R3相连，运行RIP协议，并设置为主用路径，以前的链路为备用路径。当后期运营商B设备升级之后，可支持OSPF时需要将网络割接到OSPF协议以便于管理。

# 实验拓扑

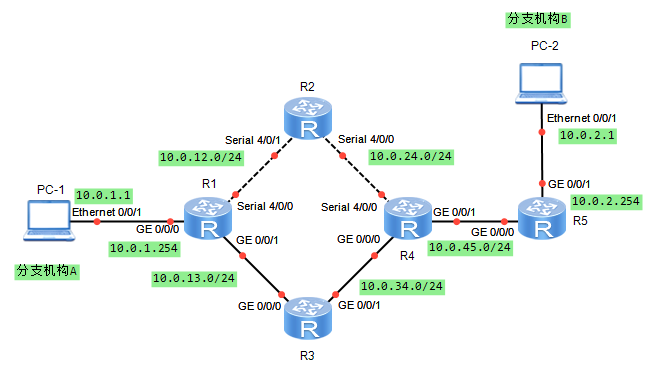


图8-9 OSPF开销值、协议优先级及计时器的修改拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | 10.0.1.254 |
| PC-2 | Ethernet 0/0/1 | 10.0.2.1 | 255.255.255.0 | 10.0.2.254 |
| R1(AR2220) | GE 0/0/0 | 10.0.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.13.1 | 255.255.255.0 | N/A |
| Serial 4/0/0 | 10.0.12.1 | 255.255.255.0 | N/A |
| R2(AR2220) | Serial 4/0/0 | 10.0.12.2 | 255.255.255.0 | N/A |
| Serial 4/0/1 | 10.0.24.2 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 10.0.13.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.34.3 | 255.255.255.0 | N/A |
| R4(AR2220) | GE 0/0/0 | 10.0.34.4 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.45.4 | 255.255.255.0 | N/A |
| Serial 4/0/0 | 10.0.24.4 | 255.255.255.0 | N/A |
| R5(AR2220) | GE 0/0/0 | 10.0.45.5 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.2.254 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 10.0.12.2

PING 10.0.12.2: 56 data bytes， press CTRL\_C to break

Reply from 10.0.12.2: bytes=56 Sequence=1 ttl=255 time=50 ms

Reply from 10.0.12.2: bytes=56 Sequence=2 ttl=255 time=40 ms

Reply from 10.0.12.2: bytes=56 Sequence=3 ttl=255 time=20 ms

Reply from 10.0.12.2: bytes=56 Sequence=4 ttl=255 time=10 ms

Reply from 10.0.12.2: bytes=56 Sequence=5 ttl=255 time=40 ms

--- 10.0.12.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/32/50 ms

其余直连网段的连通性测试省略。

## 配置协议优先级

部署OSPF网络，实现分支A和分支B之间通过R2实现通信。

在路由器R1，R2，R4上部署OSPF网络，通告相关网段属于区域0。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.12.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

[R4]ospf 1

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 10.0.24.0 0.0.0.255

[R4-ospf-1-area-0.0.0.0]network 10.0.45.0 0.0.0.255

[R5]ospf 1

[R5-ospf-1]area 0

[R5-ospf-1-area-0.0.0.0]network 10.0.45.0 0.0.0.255

[R5-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255

部署完成后，测试两分支间PC-1与PC-2的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes， Press Ctrl\_C to break

From 10.0.2.1: bytes=32 seq=1 ttl=124 time=47 ms

From 10.0.2.1: bytes=32 seq=2 ttl=124 time=31 ms

From 10.0.2.1: bytes=32 seq=3 ttl=124 time=47 ms

From 10.0.2.1: bytes=32 seq=4 ttl=124 time=31 ms

From 10.0.2.1: bytes=32 seq=5 ttl=124 time=47 ms

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/40/47 ms

通信正常，即目前通过R2的线路正常。

现网络管理员开始实施网络升级方案，部署使用经过R3的线路，运行RIP协议。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]undo summary

[R1-rip-1]network 10.0.0.0

[R3]rip 1

[R3-rip-1]version 2

[R3-rip-1]undo summary

[R3-rip-1]network 10.0.0.0

[R4]rip 1

[R4-rip-1]version 2

[R4-rip-1]undo summary

[R4-rip-1]network 10.0.0.0

[R5]rip 1

[R5-rip-1]version 2

[R5-rip-1]undo summary

[R5-rip-1]network 10.0.0.0

配置完成后，在分支A的网关设备R1上查看路由表中关于分支B网段的10.0.2.0的条目。

[R1]display ip routing-table 10.0.2.0

Route Flags: R - relay， D - download to fib

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

Routing Table : Public

Summary Count : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.2.0/24 OSPF 10 98 D 10.0.24.2 Serial4/0/0

发现分支B网段的路由条目现在仍然通过OSPF协议获得，即两分支间的数据仍然通过R2转发。新接入的R3，带宽更大的路径没有参与数据转发，升级不成功。可以使用**tracert**命令在设备和PC上进行验证，此处省略。

导致不成功的原因是该路由条目可以同时从OSPF协议和RIP协议获得，当同一路由条目可以通过不同的路由协议获得时，首先比较两协议的优先级，路由器将优选优先级高的路由协议。OSPF的默认协议优先级为10，而RIP为100，优先级值越低表示优先级越高，故而选择了从OSPF协议获得的路由条目。

但是根据实际需求，经过R2使用的OSPF线路是广域网线路，带宽很低，而经过R3使用的RIP线路是以太网线路，带宽高，所以现在一定要选择RIP条目进行转发。通过修改OSPF协议优先级即可。

在R1，R4，R5的进程下使用命令**preference**修改OSPF协议优先级的值为110，大于RIP的100。

[R1]ospf 1

[R1-ospf-1]preference 110

[R4]ospf 1

[R4-ospf-1]preference 110

[R5]ospf 1

[R5-ospf-1]preference 110

配置完成后，在分支A的网关设备R1上查看路由表中关于分支B网段的10.0.2.0的条目。

<R1>display ip routing-table 10.0.2.0

Route Flags: R - relay， D - download to fib

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

Routing Table : Public

Summary Count : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.2.0/24 RIP 100 3 D 10.0.13.3 GigabitEthernet0/0/1

可以观察到，现在已经使用经过R3的线路。

在分支B的网关设备R4上查看路由表中关于分支B网段的10.0.1.0的条目。

<R4>display ip routing-table 10.0.1.1

Route Flags: R - relay， D - download to fib

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

Routing Table : Public

Summary Count : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.1.0/24 　　　　 RIP 100 2 D 10.0.34.3 GigabitEthernet0/0/0

R4也采用经过R3的线路，往返路径一致。可以进一步使用**tracert**命令测试，这里省略。

## 配置OSPF开销值

由于网络中运行不同路由协议将会导致管理不便，现需要更改R3的配置，使其运行OSPF协议。

在网络调整过程中最重要的就是尽量确保能够使其对用户通信所造成的影响程度降至最小，并且一般选择在用户网络使用率较少的深夜进行。经过对网络的分析后发现，在R3上直接部署OSPF协议属于区域0中，即和R2一样都运行OSPF协议，那么在相同OSPF协议下，路由的选择首先比较链路的开销值，而经过R2的线路为广域网链路，开销值明显高于经过R3的以太网链路，即仍然维持通过R3来转发公司两支间的流量，风险较小。故网络管理员将直接在经过R3的线路上部署OSPF协议。

在R1，R3，R4上配置OSPF协议，通告相关网段。

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R4]ospf 1

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

配置完成后，在分支A的网关设备R1上查看路由表中关于分支B网段的10.0.2.0的条目。

<R1>display ip routing-table 10.0.2.0

Route Flags: R - relay， D - download to fib

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

Routing Table : Public

Summary Count : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.2.0/24 OSPF 110 4 D 10.0.13.3 GigabitEthernet0/0/1

可以观察到，网络配置调整完成后，仍然维持使用R3的线路转发。注意，最后还要删除RIP协议的相关配置，以免造成不必要的隐患，删除步骤此处省略。

现要求分支机构A能够每月定期检查备用路径是否正常可用，那么就要求流量能够通过R2转发，但是由于目前经过R2的线路的开销值远大于经过R3的线路而导致无法测试，可以通过手动修改OSPF开销值的方法来实现路径选择。

在R1的GE 0/0/1接口上使用命令**ospf cost**配置运行OSPF协议所需的开销值。

[R1]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]ospf cost 1000

配置完成后，在分支A的网关设备R1上查看路由表中关于分支B网段的10.0.2.0的条目。

<R1>display ip routing-table 10.0.2.0

Route Flags: R - relay， D - download to fib

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

Routing Table : Public

Summary Count : 1

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.0.2.0/24 OSPF 110 98 D 10.0.24.2 Serial4/0/0

可以观察到，现在发送至分支B的流量已经通过R2来转发，经过R2的路径的路由开销为值98，远小于R3上配置的路由开销1000。

注意，OSPF链路开销值是基于接口修改的，一定要在路由更新的入方向接口修改才生效。

## 配置OSPF计时器

网络管理员在日常网络巡检中发现，经过R3的线路是以太网，在OSPF协议中的网络类型为广播网络类型，即默认Hello计时器和Dead计时器是10秒和40秒。这样OSPF数据的Hello报文发送过于频繁。

现修改R1上Hello计时器和Dead计时器为20s和80s。

[R1]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]ospf timer hello 20

[R1-GigabitEthernet0/0/1]ospf timer dead 80

稍等片刻，会发现R1与R3的邻居关系中断，这是因为Hello计时器和Dead计时器在OSPF广播网络中建立邻居关系时要进行校验，校验一致才能够建立邻居。

同样修改R3的2个计时器，和R1保持一致。

[R3]interface GigabitEthernet 0/0/0

[R3-GigabitEthernet0/0/1]ospf timer hello 20

[R3-GigabitEthernet0/0/1]ospf timer dead 80

配置完成后，查看R1邻居状态。

<R1>display ospf peer

OSPF Process 1 with Router-ID 10.0.1.254

Neighbors

Area 0.0.0.0 interface 10.0.13.1(GigabitEthernet0/0/1)'s neighbors

Router-ID: 10.0.13.3 Address: 10.0.13.3

State: Full Mode:Nbr is Master Priority: 1

DR: 10.0.13.3 BDR: None MTU: 0

Dead timer due in 79 sec

Retrans timer interval: 0

Neighbor is up for 00:00:18

Authentication Sequence: [ 0 ]

可以观察到，邻居恢复正常。

# 思考

OSPF的Dead计时器时长默认为什么要保持是Hello计时器的4倍？一定要保持4倍关系吗？

8.8 连接RIP与OSPF网络

# 原理概述

不同的网络会根据自身的实际情况来选用路由协议。比如有些网络规模很小，为了管理简单，部署了RIP，而有些网络很复杂，可以部署OSPF。不同路由协议之间不能直接共享各自的路由信息，需要依靠配置路由的引入来实现。

获得路由信息一般有三种途径，直连网段、静态配置和路由协议。可以将通过这三种途径获得的路由信息引入到路由协议中。例如，把直连网段引入到OSPF中，叫做“引入直连”，把静态路由引入OSPF，叫做“引入静态路由”， 把RIP引入OSPF叫做“引入RIP”。当把这些路由信息引入到路由协议进程以后，这些路由信息就可以在路由协议进程中进行通告了，也就是说通过配置引入，一种路由协议可以自动获得所有来自另一种协议的所有路由信息。

不同的路由协议计算路由开销的依据是不同的，开销值的大小和范围都是不同的。OSPF的开销值基于带宽，而且值的范围很大，RIP的开销基于跳数，范围很小，所以当配置OSPF和RIP的相互引入时一定要小心（幸运的是，在华为VRP平台上，当引入OSPF路由至RIP时，如不指定Cost值，开销值将默认设为1。尽管如此，网络管理员还是应该手工配置开销值以反映网络的真实情况）。

# 实验目的

* + 理解路由引入的应用场景
  + 掌握RIP中引入其他协议的配置
  + 掌握OSPF中引入其他协议的配置
  + 掌握路由引入时修改开销值的方法

# 实验内容

本实验模拟真实网络场景，路由器R1分别连接两家公司网络，R1左侧公司A内部网络运行RIP协议，公司B内部网络运行OSPF协议。由于业务发展需要，两家公司需要能够互相通信。但由于两家公司使用不同的路由协议，现需要在路由器R1上配置双向路由引入。

# 实验拓扑

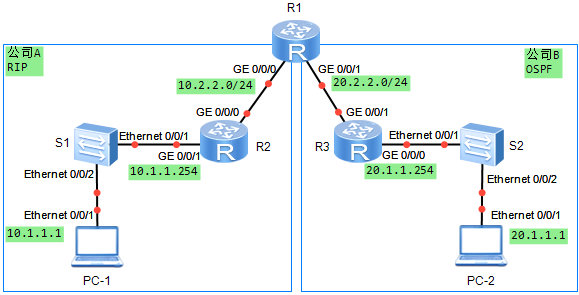


图8-10 连接RIP与OSPF网络拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 10.2.2.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.2.2.1 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 10.2.2.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.1.1.254 | 255.255.255.0 | N/A |
| R3(AR1220) | GE 0/0/0 | 20.1.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.2.2.3 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | 10.1.1.254 |
| PC-2 | Ethernet 0/0/1 | 20.1.1.1 | 255.255.255.0 | 20.1.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R2]ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=128 time=50 ms

Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=128 time=60 ms

Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=128 time=80 ms

Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms

Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=128 time=60 ms

--- 10.1.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/56/80 ms

其余直连网段的连通性测试省略。

## 搭建RIP和OSPF网络

根据实验拓扑图配置路由协议，公司A内部运行RIP协议。在R1和R2上配置RIP，进程号为1，启用RIP v2版本、关闭自动汇总，通告各自接口所在网段，R1在RIP中仅通告GE 0/0/0接口所在网段。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]undo summary

[R1-rip-1]network 10.0.0.0

[R2]rip 1

[R2-rip-1]version 2

[R2-rip-1]undo summary

[R2-rip-1]network 10.0.0.0

公司B内部运行OSPF协议。在R1和R3上配置OSPF，使用进程号1，所有网段都属于区域0，R1在OSPF中仅通告GE 0/0/1接口所在网段。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 20.2.2.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1] network 20.1.1.0 0.0.0.255

[R3-ospf-1] network 20.2.2.0 0.0.0.255

配置完成后查看R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 1 D 10.2.2.2 GigabitEthernet0/0/0

10.2.2.0/24 Direct 0 0 D 10.2.2.1 GigabitEthernet0/0/0

10.2.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.1.1.0/24 OSPF 10 2 D 20.2.2.3 GigabitEthernet0/0/1

20.2.2.0/24 Direct 0 0 D 20.2.2.1 GigabitEthernet0/0/1

20.2.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

由于R1上同时运行了RIP协议和OSPF协议，可以观察到R1同时拥有公司A和公司B的路由信息。

## 配置双向路由引入

为了使两个公司网络能够互相访问，需要把公司A的RIP协议的路由引入到公司B的OSPF协议中，同样把公司B的OSPF协议的路由引入到公司A的RIP协议中。

在R1的OSPF进程中使用命令**import-route rip**引入RIP路由。

[R1]ospf 1

[R1-ospf-1]import-route rip 1

配置完成后，查看R3的路由表。

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 O\_ASE 150 1 D 20.2.2.1 GigabitEthernet0/0/1

10.2.2.0/24 O\_ASE 150 1 D 20.2.2.1 GigabitEthernet0/0/1

20.1.1.0/24 Direct 0 0 D 20.1.1.254 GigabitEthernet0/0/0

20.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.2.2.0/24 Direct 0 0 D 20.2.2.3 GigabitEthernet0/0/1

20.2.2.3/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到R3上现在拥有来自公司A的路由信息。

在R1的RIP进程中使用命令**import-route ospf**引入OSPF路由。

[R1]rip 1

[R1-rip-1]import-route ospf 1

配置完成后，查看R2的路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/1

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

10.2.2.0/24 Direct 0 0 D 10.2.2.2 GigabitEthernet0/0/0

10.2.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.1.1.0/24 RIP 100 1 D 10.2.2.1 GigabitEthernet0/0/0

20.2.2.0/24 RIP 100 1 D 10.2.2.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到R2上现在拥有来自公司B的路由信息，且来自公司B的路由的开销值默认都为1。

当配置路由引入后双方可以互相获得对方的路由信息，但是在各自的路由表中，开销都为默认值1。

## 手工配置引入时的开销值

为了能够反应真实的网络拓扑情况，更好的进行路由控制。网络管理员在将OSPF引入进RIP时手工配置路由开销值，例如在R1的RIP进程中使用命令**import-route ospf cost 3**修改开销值为3。

[R1]rip 1

[R1-rip-1]import-route ospf 1 cost 3

配置完成后，在R2上查看cost值的变化情况。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/1

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

10.2.2.0/24 Direct 0 0 D 10.2.2.2 GigabitEthernet0/0/0

10.2.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.1.1.0/24 RIP 100 4 D 10.2.2.1 GigabitEthernet0/0/0

20.2.2.0/24 RIP 100 4 D 10.2.2.1 GigabitEthernet0/0/0

24.1.1.0/24 Direct 0 0 D 24.1.1.2 GigabitEthernet0/0/2

24.1.1.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，在R2的路由表中两条路由的Cost值已经变为4，这是因为还加上了R2接口上的Cost值1。

# 思考

关于在路由引入时手工修改路由的Cost值，这么做还有其它的用处吗？

8.9 使用RIP、OSPF发布默认路由

# 原理概述

默认路由是指目的地址和掩码都是0的路由条目。当路由器无精确匹配的路由时，就可以通过默认路由进行报文转发。如果报文的目的地址不在路由表的默认路由中，那么该报文将被路由器丢弃，并向源端返回一个ICMP报文，报告该目的地址或网络不可达。

合理使用默认路由，可以很大程度减小本地路由表的大小，节约设备资源。默认路由可以在路由器上手工配置，也可以由路由协议自动发布。

RIP和OSPF这两种路由协议都可以通过配置使路由器对协议邻居发布默认路由，并且可以设置该路由的度量值。

# 实验目的

* + 理解默认路由的应用场景
  + 掌握RIP发布默认路由的配置
  + 掌握OSPF发布默认路由的配置

# 实验内容

本实验模拟真实网络场景，路由器R1分别连接两家公司网络，R1左侧公司A内部网络运行RIP协议，公司B内部网络运行OSPF协议。由于业务发展需要，两家公司人员需要能够互相通信，但是为了保护自身网络的私密性，双方都不愿意对方知道自己网络的明细路由。这种情况下需要配置路由协议自动发布默认路由的方式来完成此需求。

# 实验拓扑

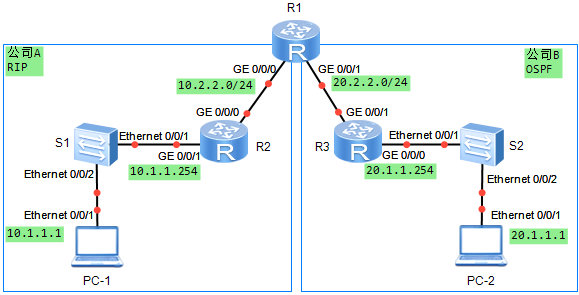


图8-11使用RIP、OSPF发布默认路由拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 10.2.2.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.2.2.1 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 10.2.2.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.1.1.254 | 255.255.255.0 | N/A |
| R3(AR1220) | GE 0/0/0 | 20.1.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.2.2.3 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | 10.1.1.254 |
| PC-2 | Ethernet 0/0/1 | 20.1.1.1 | 255.255.255.0 | 20.1.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R2]ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=128 time=50 ms

Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=128 time=60 ms

Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=128 time=80 ms

Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms

Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=128 time=60 ms

--- 10.1.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/56/80 ms

其余直连网段的连通性测试省略。

## 配置RIP和OSPF路由协议

根据实验拓扑图配置路由协议，公司A内部运行RIP协议。在R1和R2上配置RIP，进程号为1，启用RIP v2版本、关闭自动汇总，通告各自接口所在网段，R1在RIP中仅通告GE 0/0/0接口所在网段。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]undo summary

[R1-rip-1]network 10.0.0.0

[R2]rip 1

[R2-rip-1]version 2

[R2-rip-1]undo summary

[R2-rip-1]network 10.0.0.0

公司B内部运行OSPF协议。在R1和R3上配置OSPF，使用进程号1，所有网段都属于区域0，R1在OSPF中仅通告GE 0/0/1接口所在网段。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 20.2.2.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1] network 20.1.1.0 0.0.0.255

[R3-ospf-1] network 20.2.2.0 0.0.0.255

配置完成后查看R1的路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 1 D 10.2.2.2 GigabitEthernet0/0/0

10.2.2.0/24 Direct 0 0 D 10.2.2.1 GigabitEthernet0/0/0

10.2.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.1.1.0/24 OSPF 10 2 D 20.2.2.3 GigabitEthernet0/0/1

20.2.2.0/24 Direct 0 0 D 20.2.2.1 GigabitEthernet0/0/1

20.2.2.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

由于R1上同时运行了RIP协议和OSPF协议，可以观察到R1同时拥有公司A和公司B的路由信息。

查看R2和R3的路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/1

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

10.2.2.0/24 Direct 0 0 D 10.2.2.2 GigabitEthernet0/0/0

10.2.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

20.1.1.0/24 Direct 0 0 D 20.1.1.254 GigabitEthernet0/0/0

20.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.2.2.0/24 Direct 0 0 D 20.2.2.3 GigabitEthernet0/0/1

20.2.2.3/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，此时在R2和R3上都只拥有本公司的路由信息。测试PC-1与PC-2间的连通性。

PC>ping 20.1.1.1

Ping 20.1.1.1: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

可以观察到，公司A和公司B之间的PC也无法进行通信。

## 配置RIP发布默认路由

公司A需要能访问公司B的网络，而公司B为了保护自身网络私密性，不希望公司A获知自身内部网络的明细路由，这时可以在R1的RIP协议进程中发布默认路由，使公司A能在没有公司B的明细路由的情况下访问公司B的网络。

在R1的RIP进程中，使用命令**[default-route originate](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/default-route_originate.html)**发布默认路由。

[R1]rip 1

[R1-rip-1]default-route originate

配置完成后，在R2上查看路由表。

[R2]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 RIP 100 1 D 10.2.2.1 GigabitEthernet0/0/0

10.1.1.0/24 Direct 0 0 D 10.1.1.254 GigabitEthernet0/0/1

10.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

10.2.2.0/24 Direct 0 0 D 10.2.2.2 GigabitEthernet0/0/0

10.2.2.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到R2上有一条从RIP协议获取来的默认路由，通过这条默认路由，公司A可以访问公司B的网络。

## 配置OSPF发布默认路由

为了能够实现通信，公司B也需要能访问公司A的网络，而同样公司A同样为了保护自身网络私密性，不希望公司B获知自身内部网络的明细路由。这时可以在R1的OSPF协议进程中发布默认路由，使公司B能在没有公司A的明细路由的情况下能够访问公司A的网络。

在R1的OSPF进程中，使用命令**[default-route advertise](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/default-route_originate.html) always**发布默认路由。

[R1]ospf 1

[R1-ospf-1]default-route-advertise always

配置完成后，在R3上查看路由表。

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 O\_ASE 150 1 D 20.2.2.1 GigabitEthernet0/0/1

20.1.1.0/24 Direct 0 0 D 20.1.1.254 GigabitEthernet0/0/0

20.1.1.254/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

20.2.2.0/24 Direct 0 0 D 20.2.2.3 GigabitEthernet0/0/1

20.2.2.3/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到R3上有一条通过OSPF协议获得来的默认路由，通过这条默认路由，公司B可以访问公司A的网络。

再次验证PC-1与PC-2之间的连通性。

PC>ping 20.1.1.1

Ping 20.1.1.1: 32 data bytes, Press Ctrl\_C to break

From 20.1.1.1: bytes=32 seq=1 ttl=125 time=109 ms

From 20.1.1.1: bytes=32 seq=2 ttl=125 time=78 ms

From 20.1.1.1: bytes=32 seq=3 ttl=125 time=78 ms

From 20.1.1.1: bytes=32 seq=4 ttl=125 time=94 ms

From 20.1.1.1: bytes=32 seq=5 ttl=125 time=62 ms

--- 20.1.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 62/84/109 ms

通过观察可以看到此时PC之间能够正常的通信。

通过配置看到在RIP和OSPF中都可以为各自路由协议发布默认路由。配置默认路由在可以保证网络的可达性的情况下，不仅可以保护网络的私密性，同时能够有效减少路由表中路由条目的数量，使得路由器不需要维护大量的路由信息，同时其配置和维护相对简单。

# 思考

在本实验的步骤四中，OSPF中发布默认路由时使用到了命令**[default-route advertise](http://localhost:7890/pages/30002176/02/30002176/02/resources/ar/default-route_originate.html) always**，如果末尾不加always参数，会出现什么情况？如何解决？

1. VRRP

9.1 VRRP基本配置

# 原理概述

随着Internet的发展，人们对网络可靠性的要求越来越高。对于用户来说，能够时刻与外部网络保持通信非常重要，但内部网络中的所有主机通常只能设置一个网关IP地址，通过该出口网关实现主机与外部网络的通信。若此时出口网关设备发生故障，主机与外部网络的通信就会中断，所以配置多个出口网关是提高网络可靠性的常用方法。为此IETF组织推出了VRRP协议，使得在多个出口网关的情况下主机仅需配置一个虚拟网关IP地址作为出口网关即可，解决了局域网主机访问外部网络的可靠性问题。

VRRP（Virtual Router Redundancy Protocol）全称是虚拟路由冗余协议，它是一种容错协议。该协议通过把几台路由设备联合组成一台虚拟的路由设备，该虚拟路由器在本地局域网拥有唯一的一个虚拟ID和虚拟IP地址。实际上该虚拟路由器是由一个Master设备和若干Backup设备组成。正常情况下，业务全部由Master承担，所有客户端仅需设置此虚拟IP为网关地址。当Master出现故障时，Backup接替工作，及时将业务切换到备份路由器，从而保持通讯的连续性和可靠性。而客户端无需做任何配置更改，对故障无感知。

VRRP的Master选举基于优先级，优先级取值范围是0－255，缺省情况下，配置优先级为100，在接口上可以配置优先级的大小来手工选择Master设备。

# 实验目的

* + 理解VRRP的应用场景
  + 掌握VRRP虚拟路由器的配置
  + 掌握修改VRRP优先级的方法
  + 掌握查看VRRP主备状态的方法

# 实验内容

本实验模拟企业网络场景，公司内员工如PC-1、PC-2通过交换机LSW1连接到公司网络，LSW1连接到公司出口网关路由器。为了提高网络的可靠性，公司使用两台路由器R2与R3作为双出口连接到外网路由器R1。R1、R2、R3之间运行OSPF协议。在双网关的情况下，如果在PC上配置R2或R3的真实IP地址作为网关，当其中一台路由器故障时，就需要手动更改PC的网关IP，若网络中有大量PC则需要耗费大量时间和人力去更改配置，且会带来一定时间的断网影响。为了能够使故障所造成的断网影响达到最小化，增强网络的可靠性，网络管理员在R2与R3之间部署VRRP协议，可以使得当任一网关发生故障时能自动切换而无需更改PC的网关IP地址。

# 实验拓扑

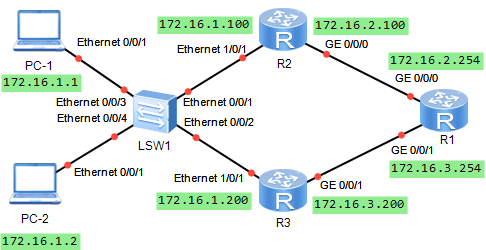


图9-1 VRRP基本配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 172.16.2.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 172.16.3.254 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 172.16.2.100 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.1.100 | 255.255.255.0 | N/A |
| R3(AR1220) | GE 0/0/1 | 172.16.3.200 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.1.200 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 172.16.1.1 | 255.255.255.0 | 172.16.1.254 |
| PC-2 | Ethernet 0/0/1 | 172.16.1.2 | 255.255.255.0 | 172.16.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping 172.16.2.100

PING 172.16.2.100: 56 data bytes, press CTRL\_C to break

Reply from 172.16.2.100: bytes=56 Sequence=1 ttl=255 time=110 ms

Reply from 172.16.2.100: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 172.16.2.100: bytes=56 Sequence=3 ttl=255 time=1 ms

Reply from 172.16.2.100: bytes=56 Sequence=4 ttl=255 time=20 ms

Reply from 172.16.2.100: bytes=56 Sequence=5 ttl=255 time=50 ms

--- 172.16.2.100 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/38/110 ms

其余直连网段的连通性测试省略。

## 部署OSPF网络

在公司的出口网关路由器R1，R2，和外网路由器R3上配置OSPF协议，使用进程号1，且所有网段均通告进区域0中。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

配置完成后，在R1上检查OSPF邻居建立情况。

[R1]display ospf peer brief

OSPF Process 1 with Router-ID 172.16.2.254

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/0 172.16.1.100 Full

0.0.0.0 GigabitEthernet0/0/1 172.16.1.200 Full

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

可以观察到，此时R1已经于R2，R3成功建立起了OSPF邻居关系。

## 配置VRRP协议

为了提高网络的可靠性，公司采用双出口的方式连接到外网。现网络管理员想针对两台出口网关路由器实现主备备份，即正常情况下，只有主网关工作，当其发生故障时能够自动切换到备份网关。现在通过配置VRRP协议来实现这样的需求。

在R2和R3上配置VRRP协议，使用命令**vrrp vrid virtual-ip**创建VRRP备份组，指定即R1和R2处于同一个VRRP备份组内，VRRP备份组号为1，配置虚拟IP为172.16.1.254。注意虚拟IP地址必须和当前接口在同一网段。

[R2]interface ethernet 1/0/1

[R2-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254

[R3]interface ethernet 1/0/1

[R3-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254

经过这样的配置后，PC将使用虚拟路由器IP地址作为默认网关。

在VRRP协议中，优先级决定路由器在备份组中的角色，优先级高者成为Master。如果优先级相同，比较接口的IP地址大小，较大的成为Master。优先级值缺省为100，0被系统保留，255保留给IP地址拥有者使用。

现在配置R2的优先级为120，R3的优先级保持默认100不变，这将使得R2成为Master，R3为Backup。

[R2-Ethernet1/0/1]vrrp vrid 1 priority 120

配置完成后，在R2和R3上使用**display vrrp**查看VRRP信息。

[R2]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Master

Virtual IP : 172.16.1.254

Master IP : 172.16.1.100

PriorityRun : 120

PriorityConfig : 120

MasterPriority : 120

Preempt : YES Delay Time : 0 s

……

[R3]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Backup

Virtual IP : 172.16.1.254

Master IP : 172.16.1.100

PriorityRun : 100

PriorityConfig : 100

MasterPriority : 120

Preempt : YES Delay Time : 0 s

……

可以观察到现在R2的VRRP状态是Master，R3是Backup。两者都处在VRRP备份组1中，且都是E 0/0/1接口运行在VRRP协议中。输出信息中的PriorityRun表示设备当前的运行优先级，PriorityConfig表示为该设备配置的优先级，MasterPriority为该备份组中Master的优先级。一般配置优先级就是运行优先级，但个别情况下可能运行优先级和配置优先级会不一样，在后续实验中会进行讨论。

也可以使用命令**display vrrp brief** 或 **display vrrp interface**来显示VRRP的工作状态，以R2为例。

[R2]display vrrp brief

VRID State Interface Type Virtual IP

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

1 Master Eth1/0/1 Normal 172.16.1.254

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

Total:1 Master:1 Backup:0 Non-active:0

[R2]display vrrp interface ethernet1/0/1

Ethernet1/0/1 | Virtual Router 1

State : Master

Virtual IP : 172.16.1.254

Master IP : 172.16.1.100

PriorityRun : 120

PriorityConfig : 120

MasterPriority : 120

Preempt : YES Delay Time : 0 s

……

测试PC机访问公网时的数据包转发路径。

PC>tracert 172.16.2.254

traceroute to 172.16.2.254, 8 hops max

(ICMP), press Ctrl+C to stop

1 172.16.1.100 32 ms 31 ms 15 ms

2 172.16.2.254 63 ms 62 ms 47 ms

可以观察此时都是通过R2转发。

## 验证VRRP主备切换

现在手动模拟网络出现故障，将LSW1的E 0/0/1接口关闭。

[LSW1]interface ethernet 0/0/1

[LSW1-Ethernet0/0/1]shutdown

经过3秒钟左右后，使用**display vrrp**查看R3的VRRP信息。

[R3]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Master

Virtual IP : 172.16.1.254

Master IP : 172.16.1.200

PriorityRun : 100

PriorityConfig : 100

MasterPriority : 100

Preempt : YES Delay Time : 0 s

……

可以观察到R3切换成为了Master，从而能够确保用户对公网的访问，几乎感知不到故障的发生。

测试PC机访问公网时的数据包转发路径。

PC>tracert 172.16.2.254

traceroute to 172.16.2.254, 8 hops max

(ICMP), press Ctrl+C to stop

1 172.16.1.200 63 ms 15 ms 32 ms

2 172.16.2.254 62 ms 62 ms 32 ms

发现数据包发送路径已经切换到R3。

如果R2从故障中恢复，手动开启LSW1的E 0/0/1接口。

[LWS1]interface ethernet 1/0/1

[LSW1-Ethernet1/0/1]undo shutdown

查看R2和R3的VRRP工作状态。

[R2]display vrrp brief

Total:1 Master:1 Backup:0 Non-active:0

VRID State Interface Type Virtual IP

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

1 Master Eth1/0/1 Normal 172.16.1.254

[R3]display vrrp brief

Total:1 Master:1 Backup:0 Non-active:0

VRID State Interface Type Virtual IP

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

1 Backup Eth1/0/1 Normal 172.16.1.254

可以观察到Master设备又立刻重新切换回至R2。

测试PC机访问公网时的数据包转发路径。

PC>tracert 172.16.2.254

traceroute to 172.16.2.254, 8 hops max

(ICMP), press Ctrl+C to stop

1 172.16.1.100 47 ms 47 ms 46 ms

2 172.16.2.254 78 ms 78 ms 62 ms

可以验证也切换回R2转发。而这整个过程对于用户来说是透明的。

# 思考

如果主路由器出现故障，比如断电停机了，备份路由器是通过什么机制检测到的？

9.2 配置VRRP多备份组

# 原理概述

当VRRP配置为单备份组时，业务全部由Master设备承担，而Backup设备完全处于空闲状态，没有得到充分利用。VRRP可以通过配置多备份组来实现负载分担，有效的解决了这一问题。

VRRP允许同一台设备的同一个接口加入多个VRRP备份组，在不同备份组中有不同的优先级，使得各备份组中的Master设备不同，也就是建立多个虚拟网关路由器。各主机可以使用不同的虚拟组路由器作为网关出口，这样可以达到分担数据流而又相互备份的目的，充分利用了每一台设备的资源。

VRRP的优先级取值范围中，255是保留给IP地址拥有者使用的，当一个VRRP路由器的物理端口IP地址和虚拟路由器的虚拟IP地址相同，这台路由器称为虚拟IP地址拥有者，VRRP优先级自动设置为255；优先级0也是特殊值，当Master设备删除VRRP配置停止运行VRRP时，会发送优先级为0的VRRP报文通知Backup设备，当Backup收到该消息后，立刻从Backup状态转为Master状态。

# 实验目的

* + 理解VRRP多备份组的应用场景
  + 掌握VRRP多备份组的配置方法
  + 理解VRRP的运行优先级和配置优先级
  + 理解VRRP虚拟地址拥有者的应用

# 实验内容

本实验模拟企业网络场景，该公司使用两台路由器R2和R3作为出口网关连接到外网R1，R2和R3运行VRRP协议，两台路由器在同一个虚拟组。当R2为主路由器时，所有业务流量都由R2承担，高峰期时会造成网络阻塞，而R3一直处于空闲状态，这样就造成了一台路由器资源的浪费。现在为了优化公司网络，增加设备利用率，需要在R2和R3之间部署双备份组VRRP，使得R2，R3分别为两个备份组的Master，保证设备的利用率。

# 实验拓扑

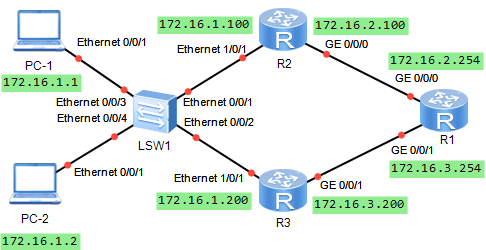


图9-2 配置VRRP多备份组拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 172.16.2.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 172.16.3.254 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 172.16.2.100 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.1.100 | 255.255.255.0 | N/A |
| R3(AR1220) | GE 0/0/1 | 172.16.3.200 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.1.200 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 172.16.1.1 | 255.255.255.0 | 172.16.1.254 |
| PC-2 | Ethernet 0/0/1 | 172.16.1.2 | 255.255.255.0 | 172.16.1.253 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

PC>ping 172.16.1.2

Ping 172.16.1.2: 32 data bytes, Press Ctrl\_C to break

From 172.16.1.2: bytes=32 seq=1 ttl=128 time=31 ms

From 172.16.1.2: bytes=32 seq=2 ttl=128 time=32 ms

From 172.16.1.2: bytes=32 seq=3 ttl=128 time=31 ms

From 172.16.1.2: bytes=32 seq=4 ttl=128 time=15 ms

From 172.16.1.2: bytes=32 seq=5 ttl=128 time=16 ms

--- 172.16.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/25/32 ms

其余直连网段的连通性测试省略。

## 部署OSPF网络

在公司的出口网关路由器R2，R3，和外网路由器R1上配置OSPF协议，使用进程号1，且所有网段均通告进区域0中。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

配置完成后，在R1上检查OSPF邻居建立情况。

[R1]display ospf peer brief

OSPF Process 1 with Router-ID 172.16.2.254

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/0 172.16.1.100 Full

0.0.0.0 GigabitEthernet0/0/1 172.16.1.200 Full

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

可以观察到，此时R1已经于R2，R3成功建立起了OSPF邻居关系。

## 配置VRRP双备份组

为了提高网络的可靠性，公司采用双出口的形式连接到外网。但是如果采用普通的VRRP单备份组配置，就会有一台设备处在空闲状态。为了提高设备的利用率，网络管理员决定采用双备份组的配置，使得不同的设备成为不同备份组中的Master，一起承担网络流量。

在R2和R3上创建VRRP虚拟组1，虚拟IP为172.16.1.254，指定R2的优先级为120，R3的优先级保持默认优先级不变。

[R2]interface Ethernet 1/0/1

[R2-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254

[R2-Ethernet1/0/1]vrrp vrid 1 priority 120

[R3]interface Ethernet 1/0/1

[R3-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254

配置完成后，分别查看R2和R3的VRRP信息。

[R2]display vrrp brief

Total:1 Master:1 Backup:0 Non-active:0

VRID State Interface Type Virtual IP

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

1 Master Eth1/0/1 Normal 172.16.1.254

[R3]display vrrp brief

Total:1 Master:0 Backup:1 Non-active:0

VRID State Interface Type Virtual IP

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

1 Backup Eth1/0/1 Normal 172.16.1.254

可以观察到，R2为组1的Master，R3为Backup。

在R2和R3上创建VRRP虚拟组2，虚拟IP为172.16.1.253，指定R3的优先级为120，R2的优先级保持默认优先级不变。

[R2]interface Ethernet 1/0/1

[R2-Ethernet1/0/1]vrrp vrid 2 virtual-ip 172.16.1.253

[R3]interface Ethernet 1/0/1

[R3-Ethernet1/0/1]vrrp vrid 2 virtual-ip 172.16.1.253

[R3-Ethernet1/0/1]vrrp vrid 2 priority 120

配置完成后，分别查看R2和R3的VRRP信息。

[R2]display vrrp brief

Total:2 Master:1 Backup:1 Non-active:0

VRID State Interface Type Virtual IP

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

1 Master Eth1/0/1 Normal 172.16.1.254

2 Backup Eth1/0/1 Normal 172.16.1.253

[R3]display vrrp brief

Total:2 Master:1 Backup:1 Non-active:0

VRID State Interface Type Virtual IP

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

1 Backup Eth1/0/1 Normal 172.16.1.254

2 Master Eth1/0/1 Normal 172.16.1.253

可以观察到，R3为组2的Master，R2为Backup。

在PC-1上设置网关地址为172.16.1.254，PC-2上设置网关地址为172.16.1.253，并在两台PC-1上**tracert** 172.16.2.254，PC-2上**tracert** 172.16.3.254。

PC>tracert 172.16.2.254

traceroute to 172.16.2.254, 8 hops max

(ICMP), press Ctrl+C to stop

1 172.16.1.100 47 ms 15 ms 47 ms

2 172.16.2.254 62 ms 63 ms 31 ms

PC>tracert 172.16.3.254

traceroute to 172.16.3.254, 8 hops max

(ICMP), press Ctrl+C to stop

1 172.16.1.200 46 ms 47 ms 31 ms

2 172.16.3.254 63 ms 62 ms 63 ms

观察发现PC-1现在是通过R2访问外网，PC-2现在是通过R3访问外网，实现了网络优化的需求。

## 验证VRRP抢占特性

在虚拟组2中R3为Master路由器，优先级为120。现在虚拟组2中修改R2的抢占模式为非抢占方式，默认是抢占方式。并将优先级改为200，即大于R3的优先级。

[R2]interface Ethernet 1/0/1

[R2-Ethernet1/0/1]vrrp vrid 2 preempt-mode disable

[R2-Ethernet1/0/1]vrrp vrid 2 priority 200

配置完成后，在R2上查看虚拟组2的信息。

[R2-Ethernet1/0/1]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Master

……

Ethernet1/0/1 | Virtual Router 2

State : Backup

Virtual IP : 172.16.1.253

Master IP : 172.16.1.200

PriorityRun : 200

PriorityConfig : 200

MasterPriority : 120

Preempt : NO

TimerRun : 1 s

……

可以观察到，尽管R2的配置优先级大于R3，并且最终运行优先级也大于R3，但是由于R2是非抢占模式，R2不会成抢占成为Master。

## 配置虚拟IP拥有者

在虚拟组1中，R2的配置优先级为120，R3的配置优先级为默认的100，R2暂时是虚拟组1的Master路由器。现在网络管理员为了保证R2在虚拟组1始终是Master，在R2的E 0/0/1接口上修改IP地址为172.16.1.254/24，这样R2就成为了该虚拟组的虚拟IP地址拥有者。

[R2]interface Ethernet 1/0/1

[R2-Ethernet1/0/1]ip address 172.16.1.254 24

配置完成后，更改R3在虚拟组1的配置优先级为可配的最大值254，这样R3的配置优先级就大于现在R2的配置优先级。

[R3]interface Ethernet 1/0/1

[R3-Ethernet1/0/1]vrrp vrid 1 priority 254

配置完成后，使用**display vrrp** **brief**查看主备状态。

[R3]display vrrp brief

VRID State Interface Type Virtual IP

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

1 Backup Eth1/0/1 Normal 172.16.1.254

2 Master Eth1/0/1 Normal 172.16.1.253

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

Total:2 Master:1 Backup:1 Non-active:0

观察发现R3无法抢占成为虚拟组1的Master。

查看R2上的VRRP信息。

[R2]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Master

Virtual IP : 172.16.1.254

Master IP : 172.16.1.254

PriorityRun : 255

PriorityConfig : 120

MasterPriority : 255

Preempt : YES Delay Time : 0 s

……

可以观察到，虽然R2在虚拟组1的配置优先级为120，但是在成为了虚拟IP地址拥有者之后，其运行优先级为255，高于R3的优先级254，所以R3无法抢占成为该组的Master。也再次验证了Master的选举及抢占都是比较运行优先级。

# 思考

在步骤四中，如果将R2配置成为虚拟组2的虚拟IP拥有者，试问此时R2能否抢占成为Master？

9.3 配置VRRP的跟踪接口及认证

# 原理概述

当VRRP的Master设备的上行接口出现问题，而Master设备一直保持Active状态，那么就会导致网络出现中断，所以必须要使得VRRP的运行状态和上行接口能够关联。在配置了VRRP冗余的网络中，为了进一步提高网络可靠性，需要在Master设备上配置上行接口监视，也就是监视连接了外网的出接口。即当此接口down掉时，自动减小优先级一定的数值（该数值由人为配置），使减小后的优先级小于Backup设备的优先级，这样Backup设备就会抢占Master角色接替工作。

VRRP支持报文的认证。缺省情况下，设备对要发送和接收的VRRP报文不进行任何认证处理，认为收到的都是真实的、合法的VRRP报文。为了使VRRP运行更加安全和稳定，可以配置VRRP的认证。VRRP支持简单字符（Simple）认证方式和MD5认证方式，用户可根据安全需要选择认证方式。

# 实验目的

* + 理解VRRP监视接口的应用场景
  + 掌握VRRP监视接口的配置方法
  + 掌握VRRP认证的配置方法

# 实验内容

本实验模拟企业网络场景，该公司使用两台路由器R2和R3作为出口网关连接到外网路由器R1，且R2和R3运行VRRP协议，两台路由器在同一个虚拟组1，R2为Master路由器。一次公司网络故障，突然所有主机都不能访问外网了，经检查发现是R2与外网路由器R1之间的链路down掉了，而VRRP的Master角色并没有发生切换，所有流量仍发送给R2，导致无法访问外网。现在需对此网络进行优化，进一步提高可靠性和安全性，需要在Master设备做VRRP的上行接口监视，当上行接口故障时，自动降低VRRP优先级使Backup设备能抢占Master角色，接替工作，当链路恢复时，R2又能自动切换回Master设备，并且在Master与Backup设备之间配置VRRP认证，提高安全性。

# 实验拓扑

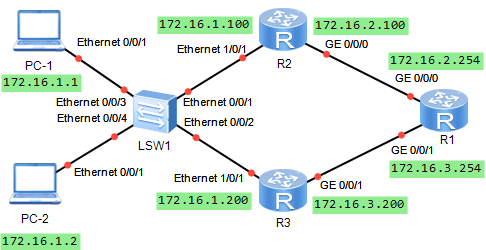


图9-3 配置VRRP的接口监视及认证拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 172.16.2.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 172.16.3.254 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 172.16.2.100 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.1.100 | 255.255.255.0 | N/A |
| R3(AR1220) | GE 0/0/1 | 172.16.3.200 | 255.255.255.0 | N/A |
| Ethernet 1/0/1 | 172.16.1.200 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 172.16.1.1 | 255.255.255.0 | 172.16.1.254 |
| PC-2 | Ethernet 0/0/1 | 172.16.1.2 | 255.255.255.0 | 172.16.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

[R2]ping 172.16.1.1

PING 172.16.1.1: 56 data bytes, press CTRL\_C to break

Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=128 time=80 ms

Reply from 172.16.1.1: bytes=56 Sequence=2 ttl=128 time=50 ms

Reply from 172.16.1.1: bytes=56 Sequence=3 ttl=128 time=40 ms

Reply from 172.16.1.1: bytes=56 Sequence=4 ttl=128 time=30 ms

Reply from 172.16.1.1: bytes=56 Sequence=5 ttl=128 time=20 ms

--- 172.16.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 20/44/80 ms

其余直连网段的连通性测试省略。

## 部署OSPF网络

在公司的出口网关路由器R2，R3，和外网路由器R1上配置OSPF协议，使用进程号1，且所有网段均通告进区域0中。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 172.16.2.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 172.16.1.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 172.16.3.0 0.0.0.255

配置完成后，在R1上检查OSPF邻居建立情况。

[R1]display ospf peer brief

OSPF Process 1 with Router-ID 172.16.2.254

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 GigabitEthernet0/0/0 172.16.1.100 Full

0.0.0.0 GigabitEthernet0/0/1 172.16.1.200 Full

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

可以观察到，此时R1已经于R2，R3成功建立起了OSPF邻居关系。

## VRRP基本配置

为了提高网络可靠性，公司采用双出口组网，并采用VRRP协议实现网关设备冗余。在R2与R3上创建同一个虚拟组，虚拟ID为1，虚拟IP为172.16.1.254，其中调整R2优先级为120，使R2成为Master，R3上的优先级不变。

[R2]interface ethernet 1/0/1

[R2-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254

[R2-Ethernet1/0/1]vrrp vrid 1 priority 120

[R3]interface ethernet 1/0/1

[R3-Ethernet1/0/1]vrrp vrid 1 virtual-ip 172.16.1.254

配置完成后，查看R2，R3上的VRRP信息。

[R2]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Master

Virtual IP : 172.16.1.254

Master IP : 172.16.1.100

PriorityRun : 120

PriorityConfig : 120

MasterPriority : 120

……

[R3]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Backup

Virtual IP : 172.16.1.254

Master IP : 172.16.1.100

PriorityRun : 100

PriorityConfig : 100

MasterPriority : 120

……

可以观察到R2现为Master，R3为Backup。

此时公司网络发生故障，R2与外网路由器R1之间的链路出现问题，无故断掉。关掉R1的GE 0/0/0接口模拟该故障。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]shutdown

配置完成后，查看主备的切换情况。

[R2]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Master

Virtual IP : 172.16.1.254

……

观察到路由器R2的Master角色并没有发生切换，所有流量仍发送给R2，导致此时用户无法访问外网，连通性测试此处省略。即VRRP无法感知上行接口发生故障来完成主备设备切换。

## 配置上行接口监视

为了进一步提高网络的可靠性和安全性，需要在Master设备R2上配置VRRP的上行接口监视。当R2的上行接口发生故障时，将自动降低优先级使得Backup设备能抢占Master角色，接替工作，使网络中断所造成的影响最小化。

在R1上恢复GE 0/0/0接口，并R2上配置上行接口监视。监视上行接口G0/0/0，当此接口down掉时，裁减优先级50，使优先级变为70，小于R3的优先级100。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]undo shutdown

[R2]interface Ethernet 1/0/1

[R2-Ethernet1/0/1]vrrp vrid 1 track interface GigabitEthernet 0/0/0 reduced 50

配置完成后，关闭R1的G0/0/0接口模拟故障发生，并使用**display vrrp**命令查看主备的切换情况。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]shutdown

[R2]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Backup

Virtual IP : 172.16.1.254

Master IP : 172.16.1.200

PriorityRun : 70

PriorityConfig : 120

MasterPriority : 100

Preempt : YES Delay Time : 0 s

TimerRun : 1 s

TimerConfig : 1 s

Auth type : NONE

Virtual MAC : 0000-5e00-0101

Check TTL : YES

Config type : normal-vrrp

Backup-forward : disabled

Track IF : GigabitEthernet0/0/0 Priority reduced : 50

IF state : DOWN

Create time : 2013-06-23 18:53:52 UTC-05:13

Last change time : 2013-06-23 19:21:15 UTC-05:13

[R3]display vrrp

Ethernet1/0/1 | Virtual Router 1

State : Master

Virtual IP : 172.16.1.254

Master IP : 172.16.1.200

PriorityRun : 100

PriorityConfig : 100

MasterPriority : 100

Preempt : YES Delay Time : 0 s

……

可以观察到，当R2上监视指定接口的状态为DOWN时，VRRP优先级被裁减掉优先级50，变成70，小于路由器R3的优先级100，由于R3的VRRP默认为抢占模式，从而变成了Backup，由R3成为新的Master并继续网络的转发。缺省情况下，当被监视的接口变为Down时，VRRP优先级的数值降低10。

## 在R2和R3上配置VRRP认证

缺省情况下，设备对要发送的VRRP报文不进行任何认证处理，收到VRRP报文的设备也不进行任何检测，认为收到的都是真实的、合法的VRRP报文。可以通过配置更改VRRP的认证模式，使VRRP对报文进行验证，从而增强安全性。

在R2和R3上对VRRP虚拟组1配置接口认证，认证方式为MD5，密码为huawei。

[R2-Ethernet1/0/1]vrrp vrid 1 authentication-mode md5 huawei

[R3-Ethernet1/0/1]vrrp vrid 1 authentication-mode md5 huawei

注意在配置VRRP报文认证时，同一VRRP备份组的认证方式必须相同，否则Master设备和Backup设备无法协商成功。

配置完成后，使用**display vrrp**命令查看。

[R2]display vrrp

Ethernet1/0/1 | Virtual Router 1

……

Auth type : MD5 Auth key : %$%$!B56J6".AW`Os:5nOIM96GU"%$%$

Virtual MAC : 0000-5e00-0101

……

[R3]display vrrp

Ethernet1/0/1 | Virtual Router 1

……

Auth type : MD5 Auth key : %$%$xASELV]Z77V(rDFgUna@6FBd%$%$

Virtual MAC : 0000-5e00-0101

……

在以上显示信息中，可以观察到“Auth Type”字段显示为“MD5”，“Auth key”字段显示为“%$%$xASELV]Z77V(rDFgUna@6FBd%$%$”，即VRRP备份组1的报文认证方式为MD5认证，以密文为“%$%$xASELV]Z77V(rDFgUna@6FBd%$%$”形式显示。

# 思考

在本实验中，可以通过配置监视上行接口来提高VRRP的可靠性，但是监视上行接口仍然等同于监视直连链路，如果非直连链路出现故障，VRRP协议能否感知？

1. 基础过滤工具

10.1 配置基本的访问控制列表

# 原理概述

访问控制列表ACL（Access Control List）是由permit或deny语句组成的一系列有顺序的规则集合，这些规则根据数据包的源地址、目的地址、源端口、目的端口等信息来描述。ACL规则通过匹配报文中的信息对数据包进行分类，路由设备根据这些规则判断哪些数据包可以通过，哪些数据包需要拒绝。

按照访问控制列表的用途，可以分为基本的访问控制列表和高级的访问控制列表，基本ACL可使用报文的源IP地址、时间段信息来定义规则，编号范围为2000～2999。

一个ACL可以由多条“deny | permit”语句组成，每一条语句描述一条规则，每条规则有一个Rule-ID，Rule-ID可以由用户进行配置，也可以由系统自动根据步长生成，缺省步长为5，Rule-ID默认按照配置先后顺序分配0、5、10、15等，匹配顺序按照ACL的Rule-ID的顺序，从小到大进行匹配。

# 实验目的

* + 掌握配置基本访问控制列表的方法
  + 掌握标准访问控制列表的执行顺序
  + 掌握标准访问控制列表的语法规则
  + 掌握标准访问控制列表的应用场景

# 实验内容

本实验模拟企业网络环境，R1为分支机构A管理员所在IT部门网关，R2为分支机构A用户部门网关，R3为分支机构A去往总部出口网关设备，R4为总部核心路由器设备。整网运行OSPF协议，并在区域0内。企业原始设计思路想要通过远程方式管理核心网路由器R4，要求只能由R1所连的某1台PC（本实验使用环回接口模拟）。地址为1.1.1.1的设备访问R4，其他设备均不能访问。

# 实验拓扑

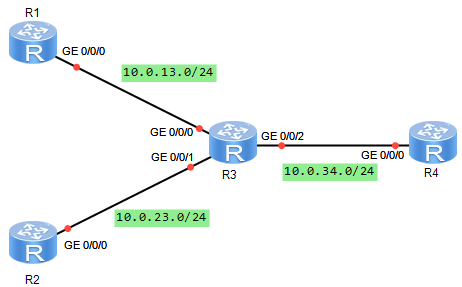


图10-1 配置基本的访问控制列表拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 10.0.13.1 | 255.255.255.0 | N/A |
| Loopback 0 | 1.1.1.1 | 255.255.255.255 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.0.23.2 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 10.0.13.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.23.3 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.34.3 | 255.255.255.0 | N/A |
| Loopback 0 | 3.3.3.3 | 255.255.255.255 | N/A |
| R4(AR2220) | GE 0/0/0 | 10.0.34.4 | 255.255.255.0 | N/A |
| Loopback 0 | 4.4.4.4 | 255.255.255.255 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping 10.0.13.3

PING 10.0.13.3: 56 data bytes, press CTRL\_C to break

Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=130 ms

Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=60 ms

Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=40 ms

Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.0.13.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/54/130 ms

测试通过，其余直连网段的连通性测试省略。

## 搭建OSPF网络

在所有路由器上运行OSPF协议，通告相应网段。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 1.1.1.1 0.0.0.0

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 3.3.3.3 0.0.0.0

[R4]ospf 1

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R4-ospf-1-area-0.0.0.0]network 4.4.4.4 0.0.0.0

配置完成之后，在R1的路由表上查看OSPF路由信息。

<R1>display ip routing-table protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 4 Routes : 4

OSPF routing table status : <Active>

Destinations : 4 Routes : 4

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.3.3.3/32 OSPF 10 1 D 10.0.13.3 GigabitEthernet0/0/0

4.4.4.4/32 OSPF 10 2 D 10.0.13.3 GigabitEthernet0/0/0

10.0.23.0/24 OSPF 10 2 D 10.0.13.3 GigabitEthernet0/0/0

10.0.34.0/24 OSPF 10 2 D 10.0.13.3 GigabitEthernet0/0/0

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

路由器R1已经学习到了相关网段的路由条目，测试R1的环回口与R4的环回口间的连通性。

<R1>ping -a 1.1.1.1 4.4.4.4

PING 4.4.4.4: 56 data bytes, press CTRL\_C to break

Reply from 4.4.4.4: bytes=56 Sequence=1 ttl=254 time=20 ms

Reply from 4.4.4.4: bytes=56 Sequence=2 ttl=254 time=20 ms

Reply from 4.4.4.4: bytes=56 Sequence=3 ttl=254 time=10 ms

Reply from 4.4.4.4: bytes=56 Sequence=4 ttl=254 time=20 ms

Reply from 4.4.4.4: bytes=56 Sequence=5 ttl=254 time=20 ms

--- 4.4.4.4 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/18/20 ms

通信正常，其他路由器之间测试省略。

## 配置基本ACL控制访问

在总部核心路由器R4上配置Telnet相关配置，配置用户密码为huawei。

<R4>system-view

[R4]user-interface vty 0 4

[R4-ui-vty0-4]authentication-mode password

Please configure the login password (maximum length 16):huawei

配置完成后，尝试在IT部门网关设备R1上建立telnet连接。

<R1>telnet 4.4.4.4

Press CTRL\_] to quit telnet mode

Trying 4.4.4.4 ...

Connected to 4.4.4.4 ...

Login authentication

Password:

<R4>

可以观察到，R1已经可以成功登陆至R4。

再尝试在普通员工部门网关设备R2上建立连接。

<R2>telnet 4.4.4.4

Press CTRL\_] to quit telnet mode

Trying 4.4.4.4 ...

Connected to 4.4.4.4 ...

Login authentication

Password:

<R4>

这时发现，只要是路由可达的设备，并且拥有Telnet的密码，都可以成功访问核心设备R4。这显然是极为不安全的。网络管理员通过配置标准ACL来实现访问过滤，禁止普通员工设备登陆。

标准的ACL可以针对数据包的源IP地址进行过滤，在R4上使用命令**acl**创建一个编号型ACL，基本ACL的范围是2000-2999。

[R4]acl 2000

接下来在ACL视图中，使用命令**rule**配置ACL规则，指定规则ID为5，允许数据包源地址为1.1.1.1的报文通过，反掩码为全0，即精确匹配。

[R4-acl-basic-2000]rule 5 permit source 1.1.1.1 0

使用命令**rule**配置第二条ACL规则，指定规则ID为10，拒绝任意源地址的数据包通过。

[R4-acl-basic-2000]rule 10 deny source any

另外在上面的ACL配置中，第一条规则的规则ID定义为5，并不是1，第二条定义为10，也不与5连续，这样配置的好处是能够方便后续的修改或插入新的条目。并且在配置的时候也可以不采用手工方式指定规则ID，ACL会自动分配规则ID，第一条为5，第二条为10，第三条为15，以此类推，即默认步长为5，该步长参数也是可以修改的。

ACL配置完成后，在VTY中调用。使用inbound参数，即在R4的数据入方向上调用。

[R4]user-interface vty 0 4

[R4-ui-vty0-4]acl 2000 inbound

配置完成后，使用R1的环回口地址1.1.1.1测试访问4.4.4.4的连通性。

<R1>telnet -a 1.1.1.1 4.4.4.4

Press CTRL\_] to quit telnet mode

Trying 4.4.4.4 ...

Connected to 4.4.4.4 ...

Login authentication

Password:

<R4>

发现没有问题，然后尝试在R2上访问R4。

<R2>telnet 4.4.4.4

Press CTRL\_] to quit telnet mode

Trying 4.4.4.4 ...

Error: Can't connect to the remote host

<R2>

可以观察到，此时R2已经无法访问4.4.4.4，即上述ACL配置已经生效。

## 基本ACL的语法规则

ACL的执行是有顺序性的，如果规则ID小的规则已经被命中，并且执行了允许或者拒绝的动作，那么后续的规则就不再继续匹配。

在R4上使用命令**display acl all**查看设备上所有的访问控制列表。

[R4]display acl all

Total quantity of nonempty ACL number is 1

Basic ACL 2000, 2 rules

Acl's step is 5

rule 5 permit source 1.1.1.1 0

rule 10 deny

可以观察到，目前ACL的所有配置信息。根据上一步骤中的配置，R4中存在一个基本ACL，有2个规则**rule 5 permit source 1.1.1.1 0**和**rule 10 deny source any**，且根据这两个规则已经将R2的访问过滤掉了。现在出现新的需求，需要R3能够使用其环回口3.3.3.3访问R4。

首先尝试使用规则ID 15来添加允许3.3.3.3访问的规则。

[R4]acl 2000

[R4-acl-basic-2000]rule 15 permit source 3.3.3.3 0

配置完成后，尝试使用R3的3.3.3.3访问R4。

<R3>telnet -a 3.3.3.3 4.4.4.4

Press CTRL\_] to quit telnet mode

Trying 4.4.4.4 ...

Error: Can't connect to the remote host

<R3>

发现无法访问。按照ACL匹配顺序，这是由于规则为10的条目是拒绝所有行为，后续所有的允许规则都不会被匹配。所以如果想要此规则生效，必须添加在拒绝所有的规则ID之前。

在R4上修改ACL2000，将规则ID修改为8。

[R4]acl 2000

[R4-acl-basic-2000]undo rule 15

[R4-acl-basic-2000]rule 8 permit source 3.3.3.3 0

配置完成后，再次尝试使用R3的环回口访问R4。

<R3>telnet -a 3.3.3.3 4.4.4.4

Press CTRL\_] to quit telnet mode

Trying 4.4.4.4 ...

Connected to 4.4.4.4 ...

Login authentication

Password:

<R4>

此时访问成功，证明配置已经生效。

# 思考

在本实验的OSPF网络中，如果ACL不配置在R4上，那么该如何设置？有什么优缺点？

10.2 配置高级的访问控制列表

# 原理概述

基本的访问控制列表(ACL)只能用于匹配源IP地址，而在实际应用当中往往需要针对数据包的其他参数进行匹配，比如目的IP地址、协议号、端口号等，所以基本的ACL由于匹配的局限性而无法实现更多的功能，所以就需要使用高级的访问控制列表，高级的访问控制列表在匹配项上做了扩展。

高级的访问控制列表编号范围为3000～3999。既可使用报文的源IP地址，也可使用目的地址、IP优先级、IP协议类型、ICMP类型、TCP源端口/目的端口、UDP源端口/目的端口号等信息来定义规则。

高级访问控制列表可以定义比基本访问控制列表更准确、更丰富、更灵活的规则，也因此得到更加广泛的应用。

# 实验目的

* + 理解高级访问控制列表的应用场景
  + 掌握配置高级访问控制列表的方法
  + 理解高级访问控制列表与基本访问控制列表的区别

# 实验内容

本实验模拟企业网络环境，R1为分支机构A管理员所在IT部门网关，R2为分支机构A用户部门网关，R3为分支机构A去往总部出口网关设备，R4为总部核心路由器设备。企业原始设计思路想要通过远程方式管理核心网路由器R4，要求由R1所连的某1台PC（本实验使用环回接口模拟），地址为1.1.1.1的设备可以访问R4，其他设备均不能访问。同时又要求只能管理R4上的4.4.4.4这台服务器（本实验使用环回口0模拟），另一台同样直连R4的服务器40.40.40.40（本实验使用环回口1模拟）不能被管理。

# 实验拓扑

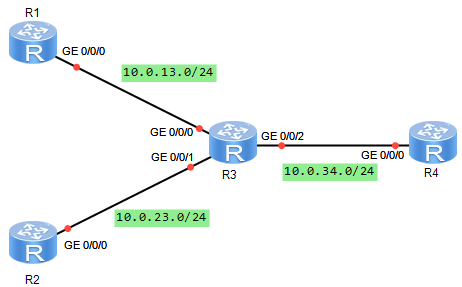


图10-2 配置高级的访问控制列表拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 10.0.13.1 | 255.255.255.0 | N/A |
| Loopback 0 | 1.1.1.1 | 255.255.255.255 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.0.23.2 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 10.0.13.3 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.23.3 | 255.255.255.0 | N/A |
| GE 0/0/2 | 10.0.34.3 | 255.255.255.0 | N/A |
| Loopback 0 | 3.3.3.3 | 255.255.255.255 | N/A |
| R4(AR2220) | GE 0/0/0 | 10.0.34.4 | 255.255.255.0 | N/A |
| Loopback 0 | 4.4.4.4 | 255.255.255.255 | N/A |
| Loopback 1 | 40.40.40.40 | 255.255.255.255 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping 10.0.13.3

PING 10.0.13.3: 56 data bytes, press CTRL\_C to break

Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=130 ms

Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=60 ms

Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=40 ms

Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.0.13.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/54/130 ms

测试通过，其余直连网段的连通性测试省略。

## 搭建OSPF网络

在所有路由器上运行OSPF协议，通告相应网段，所有设备都宣告到区域0中。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 1.1.1.1 0.0.0.0

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 3.3.3.3 0.0.0.0

[R4]ospf 1

[R4-ospf-1]area 0

[R4-ospf-1-area-0.0.0.0]network 10.0.34.0 0.0.0.255

[R4-ospf-1-area-0.0.0.0]network 4.4.4.4 0.0.0.0

[R4-ospf-1-area-0.0.0.0]network 40.40.40.40 0.0.0.0

配置完成之后，在R1的路由表上查看OSPF路由信息。

<R1>display ip routing-table protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 5 Routes : 5

OSPF routing table status : <Active>

Destinations : 5 Routes : 5

Destination/Mask Proto Pre Cost Flags NextHop Interface

3.3.3.3/32 OSPF 10 1 D 10.0.13.3 GigabitEthernet0/0/0

4.4.4.4/32 OSPF 10 2 D 10.0.13.3 GigabitEthernet0/0/0

40.40.40.40/32 OSPF 10 2 D 10.0.13.3 GigabitEthernet0/0/0

10.0.23.0/24 OSPF 10 2 D 10.0.13.3 GigabitEthernet0/0/0

10.0.34.0/24 OSPF 10 2 D 10.0.13.3 GigabitEthernet0/0/0

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

路由器R1已经学习到了相关网段的路由条目。

## 配置基本ACL控制访问

在总部核心路由器R4上配置Telnet相关配置，配置用户密码为huawei。

<R4>system-view

[R4]user-interface vty 0 4

[R4-ui-vty0-4]authentication-mode password

Please configure the login password (maximum length 16):huawei

配置完成后，尝试在IT部门网关设备R1上建立与R4的环回接口0的IP地址的telnet连接。

<R1>telnet –a 1.1.1.1 4.4.4.4

Press CTRL\_] to quit telnet mode

Trying 4.4.4.4 ...

Connected to 4.4.4.4 ...

Login authentication

Password:

<R4>

可以观察到，R1已经可以成功登陆至R4。

再尝试在R1上建立与R4的环回接口1的IP地址的telnet连接。

<R1>telnet –a 1.1.1.1 40.40.40.40

Press CTRL\_] to quit telnet mode

Trying 40.40.40.40 ...

Connected to 40.40.40.40 ...

Login authentication

Password:

<R4>

这时发现，只要是路由可达的设备，并且拥有Telnet的密码，都可以成功正常登录。

## 配置高级ACL控制访问

根据设计要求，R1的环回接口只能通过R4上的4.4.4.4进行telnet访问，但是不能通过40.40.40.40访问。

如果要R1只能通过访问R4的环回口0地址登录设备，即同时匹配数据包的源地址和目的地址实现过滤，此时通过标准ACL是无法实现的，ACL只能通过匹配源地址实现过滤，需要使用到高级ACL。

在R4上使用命令**acl**创建一个高级ACL 3000，3000-3999是高级ACL的范围。

[R4]acl 3000

接下来在高级ACL视图中，使用命令**rule**配置ACL规则，**ip**为协议类型，允许源地址为1.1.1.1，目的地址为4.4.4.4的数据包通过。

[R4-acl-adv-3000]rule permit ip source 1.1.1.1 0 destination 4.4.4.4 0

配置完成后，查看ACL配置信息。

[R4-acl-adv-3000]dis acl all

Total quantity of nonempty ACL number is 1

Advanced ACL 3000, 1 rule

Acl's step is 5

rule 5 permit ip source 1.1.1.1 0 destination 4.4.4.4 0

可以观察到相关配置信息，在不指定规则ID的情况下，默认步长为5，第一条规则的规则ID即为5。

将ACL 3000调用在VTY下，使用inbound参数，即在R4的数据入方向上调用。

[R4]user-interface vty 0 4

[R4-ui-vty0-4]acl 3000 inbound

配置完成后，分别在R1上使用环回口地址尝试访问40.40.40.40。

<R1>telnet -a 1.1.1.1 40.40.40.40

Press CTRL\_] to quit telnet mode

Trying 40.40.40.40 ...

Error: Can't connect to the remote host

<R1>

可以观察到，此时过滤已经实现，R1不能使用环回口地址访问40.40.40.40。

此外高级ACL还可以实现对源、目的端口，协议号等信息的匹配，功能非常强大。

# 思考

路由器能否通过ACL过滤自身产生的数据包？

10.3 配置前缀列表

# 原理概述

前缀列表即IP-Prefix List，它可以将与所定义的前缀列表相匹配的路由，根据定义的匹配模式进行过滤。前缀列表中的匹配条目由IP地址和掩码组成，IP地址可以是网段地址或者主机地址，掩码长度的配置范围为0～32，可以进行精确匹配或者在一定掩码长度范围内匹配，也可以通过配置关键字greater-equal和less-equal指定待匹配的前缀掩码长度范围。前缀列表能同时匹配前缀号和前缀长度，主要用于路由的匹配和控制，不能用于数据包的过滤。

# 实验目的

* + 理解前缀列表的应用场景
  + 掌握前缀列表过滤掩码长度的配置
  + 掌握前缀列表通配地址的应用
  + 理解前缀列表与ACL的区别

# 实验内容

本实验模拟公司网络场景，公司分部A网络使用10.1.1.0/24网段，通过路由器R2和骨干路由器R1相连，网络运行RIPv2协议。现在公司新成立一个分部B，新分部B路由器R3连接R1加入该RIPv2网络，由于新分部B的网络管理员不熟悉公司内网IP地址规划，在新分部B中使用了10.1.1.0/25网段，这样导致从总部发往分部A的部分数据包在R1上都会由于路由掩码最长匹配从而错误的发往分部B。而整个新分部B整改IP地址需要一定时间，公司当务之急是需要恢复总部与分部A的通信，可以通过在R1上使用前缀列表过滤掉这些错误的路由。

# 实验拓扑

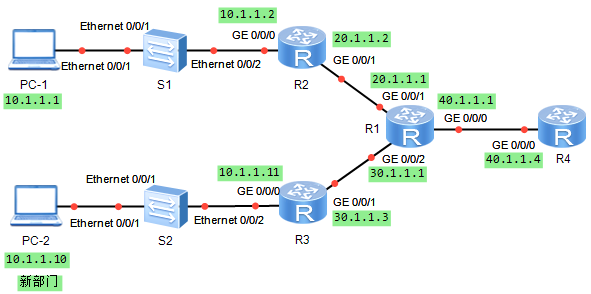


图10-3配置前缀列表拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 40.1.1.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.1.1.1 | 255.255.255.0 | N/A |
| GE 0/0/2 | 30.1.1.1 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.1.1.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 20.1.1.2 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 10.1.1.11 | 255.255.255.128 | N/A |
| GE 0/0/1 | 30.1.1.3 | 255.255.255.0 | N/A |
| R4(AR2220) | GE 0/0/0 | 40.1.1.4 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | 10.1.1.2 |
| PC-2 | Ethernet 0/0/1 | 10.1.1.10 | 255.255.255.128 | 10.1.1.11 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 20.1.1.2

PING 20.1.1.2: 56 data bytes, press CTRL\_C to break

Reply from 20.1.1.2: bytes=56 Sequence=1 ttl=255 time=60 ms

Reply from 20.1.1.2: bytes=56 Sequence=2 ttl=255 time=50 ms

Reply from 20.1.1.2: bytes=56 Sequence=3 ttl=255 time=30 ms

Reply from 20.1.1.2: bytes=56 Sequence=4 ttl=255 time=50 ms

Reply from 20.1.1.2: bytes=56 Sequence=5 ttl=255 time=40 ms

--- 172.16.2.100 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/38/110 ms

其它设备间的连通性测试略。

## 搭建RIP网络

公司内部网络使用RIPv2协议。首先配置R1、R2和R4运行RIPv2协议，在总部路由器R4上能访问分部A的PC。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]undo summary

[R1-rip-1]network 20.0.0.0

[R1-rip-1]network 30.0.0.0

[R1-rip-1]network 40.0.0.0

[R2]rip 1

[R2-rip-1]version 2

[R2-rip-1]undo summary //令用来关闭RIP-2的路由聚合功能

[R2-rip-1]network 10.0.0.0

[R2-rip-1]network 20.0.0.0

[R4]rip 1

[R4-rip-1]version 2

[R4-rip-1]undo summary

[R4-rip-1]network 40.0.0.0

配置完成后，查看总部R4的路由表。

[R4]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 2 D 40.1.1.1 GigabitEthernet0/0/0

20.1.1.0/24 RIP 100 1 D 40.1.1.1 GigabitEthernet0/0/0

30.1.1.0/24 RIP 100 1 D 40.1.1.1 GigabitEthernet0/0/0

40.1.1.0/24 Direct 0 0 D 40.1.1.4 GigabitEthernet0/0/0

40.1.1.4/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

观察发现R4的路由表中已经获得了10.1.1.0/24的路由，测试一下R4与公司总部A中PC-1间的连通性。

[R4]ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=126 time=150 ms

Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=126 time=100 ms

Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=126 time=100 ms

Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=126 time=90 ms

Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=126 time=100 ms

……

现在新分部B加入公司网络，在R3上配置RIPv2协议。

[R3]rip 1

[R3-rip-1]version 2

[R3-rip-1]undo summary

[R3-rip-1]network 10.0.0.0

[R3-rip-1]network 30.0.0.0

配置完成后，再一次查看R4的路由表。

[R4]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 2 D 40.1.1.1 GigabitEthernet0/0/0

10.1.1.0/25 RIP 100 2 D 40.1.1.1 GigabitEthernet0/0/0

20.1.1.0/24 RIP 100 1 D 40.1.1.1 GigabitEthernet0/0/0

30.1.1.0/24 RIP 100 1 D 40.1.1.1 GigabitEthernet0/0/0

40.1.1.0/24 Direct 0 0 D 40.1.1.4 GigabitEthernet0/0/0

40.1.1.4/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，此时R4接收到了公司分部A的10.1.1.0/24路由条目，和新分部B的10.1.1.0/25路由条目，同样R1也会接收到这两条路由条目，这样会造成什么后果？

根据路由器转发数据的原理，在转发数据包时路由器会根据最长匹配的原则去匹配路由条目，即R4向分部A的终端PC-1发送数据时，当数据包到达R1后，根据包头的目的IP地址与路由表中的路由条目进行匹配，发现10.1.1.0/25该条目匹配更精确，这会使得数据包都根据这条路由条目进行转发，即将原本要发往PC-1的数据包都错误的发往R3，造成总部与分部A异常通信。

在R4上测试与PC-1间的连通性。

<R4>ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

……

可以观察到此时无法正常通信。

在R4上使用命令**tracert**测试发往PC-1的数据包所经过的网关。

<R4>tracert 10.1.1.1

traceroute to 10.1.1.1(10.1.1.1), max hops: 30 ,packet length: 40,press CTRL\_C to break

1 40.1.1.1 70 ms 50 ms 50 ms

2 30.1.1.3 60 ms 80 ms 60 ms

3 \* \* \*

4 \* \* \*

可以观察到，确实此时R4发往分公司A的PC-1的数据包都已错误的发往R3。

## 配置ACL过滤路由

由于业务需要，现公司急需恢复总部与分部A间的通信。但是重新规划并配置整个分部B的IP地址需要一定时间，此时网络管理员尝试使用ACL来配置路由过滤，即在R1上过滤掉10.1.1.0/25这条路由。

在R1上创建基本的ACL，拒绝10.1.1.0 这个目的网段的路由。

[R1]acl number 2000

[R1-acl-basic-2000]rule 5 deny source 10.1.1.0 0.0.0.0

[R1-acl-basic-2000]rule 10 permit source any

接下来在RIP视图下，配置过滤策略（filter-policy)，该策略通过调用之前配置好的ACL来达到过滤路由的目的，并且在R1的RIP路由进程中的接收方向应用此路由过滤策略。

[R1]rip 1

[R1-rip-1]filter-policy 2000 import

配置完成后，查看R1的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

20.1.1.0/24 Direct 0 0 D 20.1.1.1 GigabitEthernet0/0/1

20.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

30.1.1.0/24 Direct 0 0 D 30.1.1.1 GigabitEthernet0/0/2

30.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

40.1.1.0/24 Direct 0 0 D 40.1.1.1 GigabitEthernet0/0/0

40.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

观察发现R1的路由表中10.1.1.0/24和10.1.1.0/25这两条路由都被过滤。这是因为ACL无法实现对掩码长度进行精确匹配，而分部A的网络位和分部B的网络位相同，都是10.1.1.0，就会导致把分部A的路由也同时过滤。

## 配置前缀列表过滤路由 // 前缀列表优先

为了能够精确匹配掩码长度，仅过滤掉10.1.1.0/25这条新分部B的路由，可以在R1上配置前缀列表实现。

在R1上配置前缀列表，同时精确匹配网络位和掩码长度。

[R1]ip ip-prefix 1 deny 10.1.1.0 25 greater-equal 25 less-equal 25

[R1]ip ip-prefix 1 permit 0.0.0.0 0 less-equal 32

第二条配置表示放行所有其它的路由，这是因为前缀列表也会有一条隐含的拒绝所有的规则，所以如果要放行其它所有路由的话，一定要显式增加一条允许所有的规则。而第一条配置也可以使用下面的方式简写。

[R1]ip ip-prefix 1 deny 10.1.1.0 25 ///R1路由信息

将该前缀列表应用到过滤策略下。

[R1-rip-1]filter-policy ip-prefix 1 import

配置完成后，查看R1的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 1 D 20.1.1.2 GigabitEthernet0/0/1

20.1.1.0/24 Direct 0 0 D 20.1.1.1 GigabitEthernet0/0/1

20.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

30.1.1.0/24 Direct 0 0 D 30.1.1.1 GigabitEthernet0/0/2

30.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

40.1.1.0/24 Direct 0 0 D 40.1.1.1 GigabitEthernet0/0/0

40.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，此时R1的路由表中仅存在10.1.1.0/24的分部A的路由条目，这样就恢复了总部与分部A的通信。

测试总部与分部A中PC-1间的连通性。

<R4>ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=126 time=120 ms

Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=126 time=50 ms

Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=126 time=130 ms

Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=126 time=60 ms

Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=126 time=110 ms

--- 172.16.2.100 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/38/110 ms

可以观察到，此时通信恢复正常。

## 恢复新分部网络

为了新分部B能正确接入到现有网络中，网络管理员需要重新规划IP地址，使分部B能与公司总部及分部A通信。

规划分部B网络使用10.2.2.0/24网段，更改PC2的IP地址为10.2.2.1/24，R3的G0/0/0接口IP地址为10.2.2.3/24。

[R3]interface GigabitEthernet 0/0/0

[R3-GigabitEthernet0/0/0]ip address 10.2.2.3 24

配置完成后，查看R1的路由表。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 9 Routes : 9

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 RIP 100 1 D 20.1.1.2 GigabitEthernet0/0/1

10.2.2.0/24 RIP 100 1 D 20.1.1.2 GigabitEthernet0/0/1

20.1.1.0/24 Direct 0 0 D 20.1.1.1 GigabitEthernet0/0/1

20.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

30.1.1.0/24 Direct 0 0 D 30.1.1.1 GigabitEthernet0/0/2

30.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/2

40.1.1.0/24 Direct 0 0 D 40.1.1.1 GigabitEthernet0/0/0

40.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到，R1的路由表中现有新分部B所在10.2.2.0/24网段的路由条目，也有分部A的路由。使用**ping**命令测试总部与分部A，新分部B间的连通性。

<R4>ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=126 time=110 ms

Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=126 time=70 ms

Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=126 time=70 ms

Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=126 time=80 ms

Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=126 time=100 ms

--- 172.16.2.100 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/38/110 ms

<R4>ping 10.2.2.1

PING 10.2.2.1: 56 data bytes, press CTRL\_C to break

Reply from 10.2.2.1: bytes=56 Sequence=1 ttl=126 time=120 ms

Reply from 10.2.2.1: bytes=56 Sequence=2 ttl=126 time=100 ms

Reply from 10.2.2.1: bytes=56 Sequence=3 ttl=126 time=80 ms

Reply from 10.2.2.1: bytes=56 Sequence=4 ttl=126 time=80 ms

Reply from 10.2.2.1: bytes=56 Sequence=5 ttl=126 time=110 ms

--- 172.16.2.100 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/38/110 ms

可以观察到，通信正常。

# 思考

如果将前缀列表中已配置好的语句顺序打乱会对实验结果产生影响吗？

1. 广域网

11.1 WAN接入配置

# 原理概述

高级数据链路控制HDLC（High-level Data Link Control）是一种链路层协议，运行在同步串行链路之上。HDLC最大的特点是不需要规定数据必须是字符集，对任何一种比特流，均可以实现透明的传输。

HDLC是由国际标准化组织ISO制定的，是通信领域曾应用十分广泛的一个数据链路层协议。但是随着技术的进步，目前通信信道的可靠性比过去已经有了非常大的改进。我们已经没有必要在数据链路层使用很复杂的协议（包括编号、检错重传等技术）来实现数据的可靠传输。作为窄带通信协议的HDLC，在公网的应用逐渐消失，应用范围逐渐减小，在部分专网中主要利用HDLC协议封装透传业务数据。

PPP（Point-to-Point Protocol）协议是一种数据链路层协议，主要用来在全双工的同异步链路上进行点到点之间的数据传输。PPP最初是设计为两个对等节点之间的IP流量提供一种封装协议，是在串行线IP协议SLIP（Serial Line IP）的基础上发展而来的。由于SLIP协议只支持异步传输方式，无协商过程，只能承载IP一种网络报文等问题，在发展过程中，逐步被PPP协议所替代。PPP与HDLC的主要区别是HDLC是面向位的，而PPP是面向字节的。PPP是一种多协议成帧机制，适用于调制解调器。

串行链路是指信息的各位数据被逐位按顺序传送的线路，适用于远距离通信，但速度较慢，与之相对的是并行链路，能够在同一时刻传送一个8位数据。同步和异步是广域网的串行链路的两种传输模式，同步模式要求通信双方以相同的时钟频率进行，通过共享单个时钟或定时脉冲源保证发送方和接收方的准确同步，效率较高；异步模式不要求双方同步，收发方可以采用各自的时钟源，双方遵循异步的通信协议，以字符为数据传输单位，发送方传送字符的时间间隔不确定，发送效率比同步模式低。

# 实验目的

* + 掌握PPP的基本配置
  + 掌握HDLC的基本配置
  + 理解PPP与HDLC的异同

# 实验内容

本实验模拟企业网络场景，某公司有开发部门PC-1，通过部门路由器R2连接到公司出口网关R1；有市场部门PC-2，直连到公司出口网关；有IT部门PC-3，通过部门路由器R3连接到公司出口网关。R2与R1之间链路为串行链路，封装PPP协议，R3与R1之间链路为串行链路，封装HDLC协议，R2与R3分别设置默认路由指向R1，使各部门之间能互相访问。

# 实验拓扑

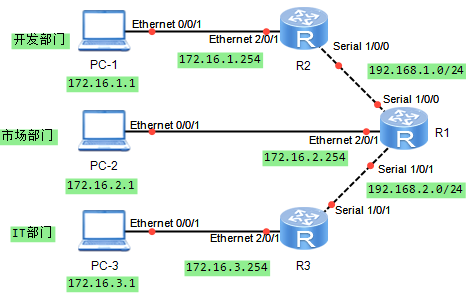


图11-1 WAN接入拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | Serial 1/0/1 | 192.168.2.2 | 255.255.255.0 | N/A |
| Serial 1/0/0 | 192.168.1.2 | 255.255.255.0 | N/A |
| Ethernet 2/0/1 | 172.16.2.254 | 255.255.255.0 | N/A |
| R2(AR1220) | Serial 1/0/0 | 192.168.1.1 | 255.255.255.0 | 192.168.1.2 |
| Ethernet 2/0/1 | 172.16.1.254 | 255.255.255.0 | 192.168.1.2 |
| R3(AR1220) | Serial 1/0/1 | 192.168.2.1 | 255.255.255.0 | 192.168.2.2 |
| Ethernet 2/0/1 | 172.16.3.254 | 255.255.255.0 | 192.168.2.2 |
| PC-1 | Ethernet 0/0/1 | 172.16.1.1 | 255.255.255.0 | 172.16.1.254 |
| PC-2 | Ethernet 0/0/1 | 172.16.2.1 | 255.255.255.0 | 172.16.2.254 |
| PC-3 | Ethernet 0/0/1 | 172.16.3.1 | 255.255.255.0 | 172.16.3.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用ping命令检测各直连链路的连通性。

<R1>ping -c 1 172.16.2.1

PING 172.16.2.1: 56 data bytes, press CTRL\_C to break

Reply from 172.16.2.1: bytes=56 Sequence=1 ttl=255 time=510 ms

--- 172.16.2.1 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 510/510/510 ms

其余直连网段的连通性测试省略。

## 配置PPP

缺省情况下，串行接口封装的链路层协议即为PPP，可以直接在R1上使用命令**display interface serial 1/0/0**进行查看。

[R1]display interface serial1/0/0

Serial1/0/0 current state : UP

……

Internet Address is 192.168.1.2/24

Link layer protocol is PPP

LCP opened, IPCP opened

……

在R2上配置默认路由指向出口网关R1，并在R1上配置目的网段为PC-1所在网络的静态路由，下一跳路由器为R2。

[R2]ip route-static 0.0.0.0 0.0.0.0 192.168.1.2

[R1]ip route-static 172.16.1.0 255.255.255.0 192.168.1.1

配置完成后，在PC-1上测试与R1间的连通性。

PC>ping 192.168.1.2

Ping 192.168.1.2: 32 data bytes, Press Ctrl\_C to break

From 192.168.1.2: bytes=32 seq=1 ttl=254 time=16 ms

From 192.168.1.2: bytes=32 seq=2 ttl=254 time=31 ms

From 192.168.1.2: bytes=32 seq=3 ttl=254 time=32 ms

From 192.168.1.2: bytes=32 seq=4 ttl=254 time=31 ms

From 192.168.1.2: bytes=32 seq=5 ttl=254 time=31 ms

--- 192.168.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 16/28/32 ms

可以正常通信。

## 配置HDLC

在R1和R3的S 1/0/1接口上分别使用**link-protocol**命令配置链路层协议为HDLC。

[R1]interface Serial 1/0/1

[R1-Serial1/0/1]link-protocol hdlc

Warning: The encapsulation protocol of the link will be changed.

Continue? [Y/N]:y

[R3]interface Serial 1/0/1

[R3-Serial1/0/1]link-protocol hdlc

Warning: The encapsulation protocol of the link will be changed.

Continue? [Y/N]:y

在R3上配置默认路由指向出口网关R1，并在R1上配置目的网段为PC-3所在网络的静态路由，下一跳路由器为R3连接R1的S 1/0/1接口。

[R3]ip route-static 0.0.0.0 0.0.0.0 serial 1/0/1

[R1]ip route-static 172.16.3.0 255.255.255.0 serial 1/0/1

配置完成后，在PC-3上测试与路由器R1间连通性。

PC>ping 192.168.2.2

Ping 192.168.2.2: 32 data bytes, Press Ctrl\_C to break

From 192.168.2.2: bytes=32 seq=1 ttl=254 time=15 ms

From 192.168.2.2: bytes=32 seq=2 ttl=254 time=46 ms

From 192.168.2.2: bytes=32 seq=3 ttl=254 time=31 ms

From 192.168.2.2: bytes=32 seq=4 ttl=254 time=16 ms

From 192.168.2.2: bytes=32 seq=5 ttl=254 time=31 ms

--- 192.168.2.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 15/27/46 ms

可以正常通信。

# 思考

假设在某一直连链路上，一端接口的链路层协议为PPP，另一端为HDLC，此时能否正常通信？

11.2 PPP的认证

# 原理概述

在网络日益发展的今天，人们对网络安全性的要求越来越高，而PPP协议之所以能成为广域网中应用较为广泛的协议，原因之一就是因为其能提供验证协议CHAP（Challenge- Handshake Authentication Protocol）、PAP（Password Authentication Protocol），更好的保证了网络安全性。

PAP验证协议为两次握手验证，口令为明文。验证过程仅在链路初始建立阶段进行。当链路建立阶段结束后，用户名和密码将由被验证方重复地在链路上发送给验证方，直到验证通过或者中止连接。PAP不是一种安全的验证协议，当验证时，口令以明文方式在链路上发送，并且由于完成PPP链路建立后，被验证方会不停地在链路上反复发送用户名和口令，直到身份验证过程结束，用户名和口令很容易被截获，所以不安全。

CHAP是挑战握手验证协议，它是三次握手验证协议，只在网络上传输用户名，而并不传输用户密码，因此安全性要比PAP高。CHAP协议是在链路建立的开始就完成的。在链路建立完成后的任何时间都可以进行再次验证。当链路建立阶段完成后，验证方发送一个“challenge“报文给被验证方。被验证方经过一次哈希算法后，给验证方返回一个值。验证方把自己经过哈希算法生成的值和被验证方返回的值进行比较。如果两者匹配，那么验证通过。否则验证不通过，连接被终止。

# 实验目的

* + 掌握配置PPP PAP认证的方法
  + 掌握配置PPP CHAP认证的方法
  + 理解PPP PAP认证与CHAP认证的区别

# 实验内容

本实验模拟企业网络环境，R1为分支机构接入端网关设备，PC-1为企业分支机构终端，

R2为企业总部接入终端网关设备，PC-2为企业总部终端，R3为企业总部核心路由器。出于安全角度考虑，网络管理员在分支机构访问总部时部署PPP认证，R1是被认证方路由器，R3是认证方路由器，只有认证通过才能建立PPP连接进行正常访问。

# 实验拓扑

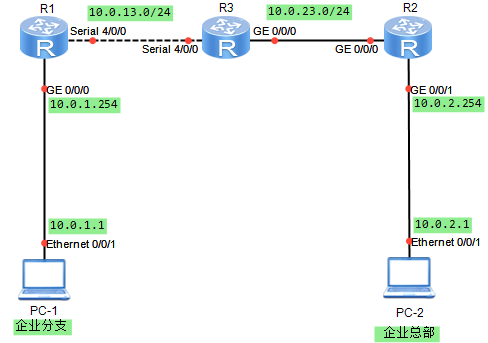


图11-2 PPP的认证拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| PC-1 | Ethernet 0/0/1 | 10.0.1.1 | 255.255.255.0 | 10.0.1.254 |
| PC-2 | Ethernet 0/0/1 | 10.0.2.1 | 255.255.255.0 | 10.0.2.254 |
| R1(AR2220) | GE 0/0/0 | 10.0.1.254 | 255.255.255.0 | N/A |
| Serial 4/0/0 | 10.0.13.1 | 255.255.255.0 | N/A |
| R2(AR2220) | GE 0/0/0 | 10.0.23.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.0.2.254 | 255.255.255.0 | N/A |
| R3(AR2220) | GE 0/0/0 | 10.0.23.3 | 255.255.255.0 | N/A |
| Serial 4/0/0 | 10.0.13.3 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 10.0.1.254

PING 10.0.1.254: 56 data bytes, press CTRL\_C to break

Reply from 10.0.1.254: bytes=56 Sequence=1 ttl=255 time=90 ms

Reply from 10.0.1.254: bytes=56 Sequence=2 ttl=255 time=50 ms

Reply from 10.0.1.254: bytes=56 Sequence=3 ttl=255 time=1 ms

Reply from 10.0.1.254: bytes=56 Sequence=4 ttl=255 time=1 ms

Reply from 10.0.1.254: bytes=56 Sequence=5 ttl=255 time=1 ms

--- 10.0.1.254 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 1/28/90 ms

测试通过，其余直连网段的连通性测试省略。

## 搭建OSPF网络

在每台路由器上配置OSPF协议，并通告相应网段到区域0内。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 10.0.1.0 0.0.0.255

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 10.0.2.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.23.0 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.0.13.0 0.0.0.255

配置完成后测试总部与分支终端间的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.2.1: bytes=32 seq=1 ttl=252 time=78 ms

From 10.0.2.1: bytes=32 seq=2 ttl=252 time=78 ms

From 10.0.2.1: bytes=32 seq=3 ttl=252 time=63 ms

From 10.0.2.1: bytes=32 seq=4 ttl=252 time=31 ms

From 10.0.2.1: bytes=32 seq=5 ttl=252 time=31 ms

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/56/78 ms

可以观察到，通信正常。

## 配置PPP的PAP认证

现在为了提升分支机构与总部通信时的安全性，在分支网关设备R1与公司核心设备R3上部署PPP的PAP认证。R3作为认证路由器，R1作为被认证路由器。

由于在华为路由器上，广域网串行接口默认链路层协议即为PPP，可以直接配置PPP认证即可。在总部设备R3上使用命令**ppp authentication-mode**命令设置本端的PPP协议对对端设备的认证方式为PAP，认证采用的域名为huawei???(*域名保持前后一致*)。

[R3]interface Serial 4/0/0

[R3-Serial4/0/0]ppp authentication-mode pap domain huawei

配置认证路由器R3的本地认证信息。执行命令**aaa**，进入AAA视图。

[R3]aaa

使用命令**authentication-scheme**创建认证方案huawei\_1，并进入认证方案视图。

[R3-aaa]authentication-scheme huawei\_1

使用命令**authentication-mode**配置认证模式为本地认证。

[R3-aaa-authen-huawei\_1]authentication-mode local

使用命令**domain**创建域huaweiyu，并进入域视图。

[R3-aaa]domain huaweiyu

使用命令**authentication-scheme**配置域的认证方案为huawei\_1，注意必须和创建的认证方案一致。

[R3-aaa-domain-huaweiyu]authentication-scheme huawei\_1

退回到AAA视图，使用命令**local-user**配置存储在本地，为对端认证方所使用的用户名为R1@huaweiyu，密码为Huawei。

[R3-aaa]local-user R1@huaweiyu password cipher Huawei

[R3-aaa]local-user R1@huaweiyu service-type ppp

配置完成后，关闭R1与R3相连接口，再打开使得R1与R3间的链路重新协商，并检查链路状态，和连通性。

[R3]interface Serial 4/0/0

[R3-Serial4/0/0]shutdown

[R3-Serial4/0/0]undo shutdown

<R1>display ip interface brief

\*down: administratively down

……

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 10.0.1.254/24 up up

GigabitEthernet0/0/1 unassigned down down

GigabitEthernet0/0/2 unassigned down down

NULL0 unassigned up up(s)

Serial4/0/0 10.0.13.1/24 up down

Serial4/0/1 unassigned down down

<R3>ping 10.0.13.1

PING 10.0.13.1: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

……

可以观察到，现在R1与R3间无法正常通信，链路物理状态正常，但是链路层协议状态不正常。这是因为此时PPP链路上的PAP认证不通过，现在仅仅配置了被认证方设备R3，还需要再认证方R1配置相关PAP认证参数。

在R1上的S 4/0/0接口下，使用命令**ppp pap local-user**配置本端被对端以PAP方式验证时本地发送的PAP用户名和密码。

[R1]interface Serial 4/0/0

[R1-Serial4/0/0]ppp pap local-user R1@huaweiyu password cipher Huawei

配置完成后，再次查看链路状态，并测试连通性。

[R1]display ip interface brief

\*down: administratively down

……

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 10.0.1.254/24 up up

GigabitEthernet0/0/1 unassigned down down

GigabitEthernet0/0/2 unassigned down down

NULL0 unassigned up up(s)

Serial4/0/0 10.0.13.1/24 up up

Serial4/0/1 unassigned down down

<R1>ping 10.0.13.3

PING 10.0.13.3: 56 data bytes, press CTRL\_C to break

Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=70 ms

Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=30 ms

Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=50 ms

Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=40 ms

Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.0.13.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/40/70 ms

可以观察到，现在R1与R3间的链路层协议状态正常，并且可以正常通信。

测试PC-1与PC-2的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.2.1: bytes=32 seq=1 ttl=125 time=31 ms

From 10.0.2.1: bytes=32 seq=2 ttl=125 time=63 ms

From 10.0.2.1: bytes=32 seq=3 ttl=125 time=47 ms

From 10.0.2.1: bytes=32 seq=4 ttl=125 time=31 ms

From 10.0.2.1: bytes=32 seq=5 ttl=125 time=31 ms

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/40/63 ms

总部与分支间终端通信正常。

## 配置PPP的CHAP认证

公司网络管理员在日常网络维护中发现，分部公司频繁遭受攻击，PPP认证密码经常被盗用，遂对网络状况进行分析。抓取R1的Serial 4/0/0数据包进行分析。

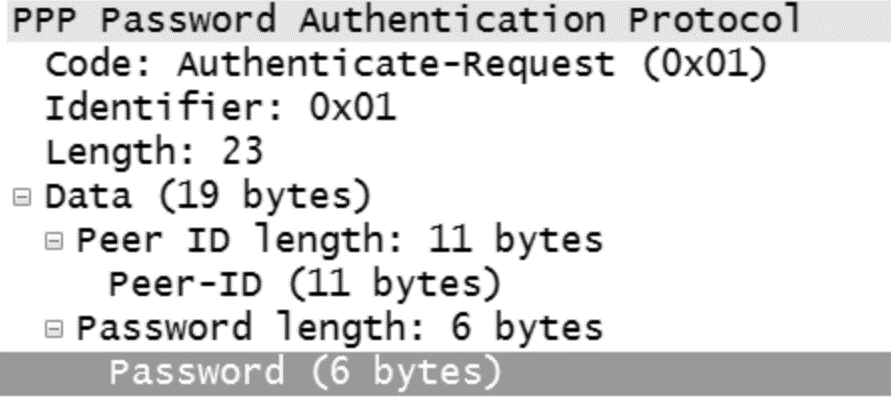


图11-3

可以观察到，在数据包中很容易找到所配置的用户名和密码。Peer-ID显示内容为用户名，Password显示内容为密码，可以查看具体内容。

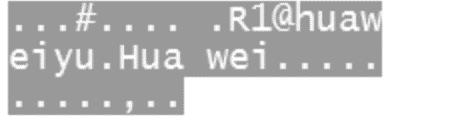


图11-4

可以观察到，能够找到用户名为R1@huaweiyu，密码为Huawei。验证了使用PAP认证时，口令将以明文方式在链路上传送，并且由于完成PPP链路建立后，被认证方会不停地在链路上反复发送用户名和口令，直到身份认证过程结束，所以不能防止攻击。而使用CHAP认证时，口令用MD5算法加密后在链路上发送，能有效的防止攻击。因此为了进一步提高链路安全性，网络管理员重新部署PPP的CHAP认证解决此问题。

首先删除原有的PAP认证配置。域名保持不变。

[R3]interface Serial 4/0/0

[R3-Serial4/0/0]undo ppp authentication-mode

[R1]interface Serial 4/0/0

[R1-Serial4/0/0]undo ppp pap local-user

删除后，在认证设备R3的S 4/0/0接口下配置PPP的认证方式为CHAP。

[R3]interface Serial 4/0/0

[R3-Serial4/0/0]ppp authentication-mode CHAP

配置存储在本地，为对端认证方所使用的用户名为R1，密码为huawei。

[R3]aaa

[R3-aaa]local-user R1 password cipher huawei

[R3-aaa]local-user R1 service-type ppp

其余认证方案和域的配置保持不变。

配置完成后，关闭R1与R3相连接口后，再打开使得链路重新协商，并查看链路状态及测试连通性。

[R3]interface s4/0/0

[R3-Serial4/0/0]shutdown

[R3-Serial4/0/0]undo shutdown

<R3>display ip interface brief

\*down: administratively down

……

Interface IP Address/Mask Physical Protocol

GigabitEthernet0/0/0 10.0.23.3/24 up up

GigabitEthernet0/0/1 unassigned down down

GigabitEthernet0/0/2 unassigned down down

NULL0 unassigned up up(s)

Serial4/0/0 10.0.13.3/24 up down

Serial4/0/1 unassigned down down

<R3>ping 10.0.13.1

PING 10.0.13.1: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

--- 10.0.13.1 ping statistics ---

5 packet(s) transmitted

0 packet(s) received

100.00% packet loss

可以观察到，目前R1与R3间的链路层协议状态不正常，无法正常通信。这是由于此时被认证方R1上还没有配置用户名和密码。

在R1的S 4/0/0接口下配置CHAP认证的用户名和密码。

[R1]interface Serial 4/0/0



[R1-Serial4/0/0]ppp chap user R1



[R1-Serial4/0/0]ppp chap password huawei

配置完成，测试R1与R3的连通性。

<R1>ping 10.0.13.3

PING 10.0.13.3: 56 data bytes, press CTRL\_C to break

Reply from 10.0.13.3: bytes=56 Sequence=1 ttl=255 time=50 ms

Reply from 10.0.13.3: bytes=56 Sequence=2 ttl=255 time=30 ms

Reply from 10.0.13.3: bytes=56 Sequence=3 ttl=255 time=40 ms

Reply from 10.0.13.3: bytes=56 Sequence=4 ttl=255 time=40 ms

Reply from 10.0.13.3: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.0.13.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/30/50 ms

认证通过，R1与R3间通正常，再测试分支PC-1和总部PC-2间的连通性。

PC>ping 10.0.2.1

Ping 10.0.2.1: 32 data bytes, Press Ctrl\_C to break

From 10.0.2.1: bytes=32 seq=1 ttl=125 time=31 ms

From 10.0.2.1: bytes=32 seq=2 ttl=125 time=32 ms

From 10.0.2.1: bytes=32 seq=3 ttl=125 time=47 ms

From 10.0.2.1: bytes=32 seq=4 ttl=125 time=46 ms

From 10.0.2.1: bytes=32 seq=5 ttl=125 time=32 ms

--- 10.0.2.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/37/47 ms

并在R1的S 4/0/0接口下再次抓取数据包查看。

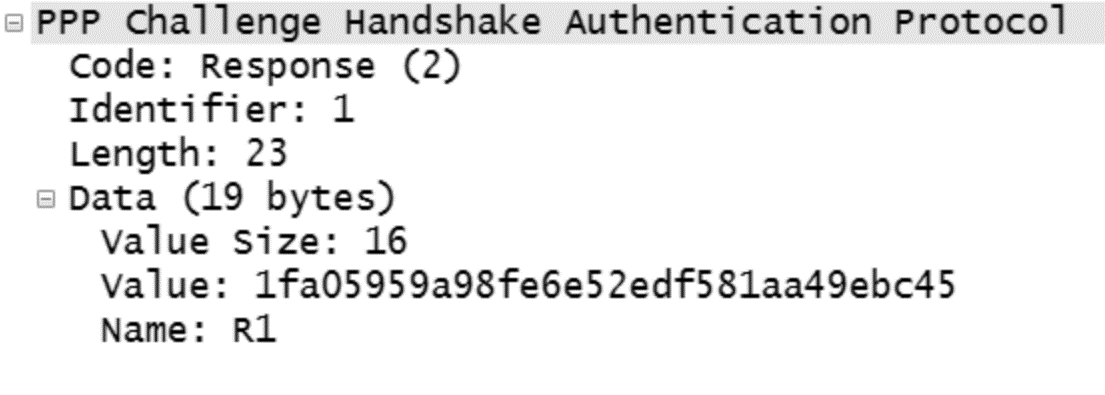


图11-5

可以观察到，现在数据包内容已经为加密方式发送，无法截获认证密码，安全性得到了提升。

# 思考

当PPP链路UP后，在PPP链路一端加上认证配置而另一端不加，为什么一定要重启端口后认证才能生效？

11.3 帧中继基本配置

# 原理概述

帧中继（frame relay)是一种面向连接的数据链路层技术。它主要用在公共或专用网上的局域网互联以及广域网连接。

帧中继协议是一种简化X.25的广域网协议，它在控制层面上提供虚电路的管理，带宽管理和防止阻塞等功能。在传送数据时使用的传输链路是逻辑连接，而不是物理连接。在一个物理连接上可以复用多个逻辑连接，可以实现带宽的复用和动态分配。帧透明传输和错误检测，但不提供重传操作。与传统的电路交换相比，帧中继网络有利于多用户、多速率数据的传输，充分利用了网络资源。

帧中继是用虚电路来连接帧中继网络两端的设备。每条虚电路是用数据链路连接标识符定义的一条帧中继连接通道，提供了用户设备（如路由器和主机等）之间进行数据通信的能力，下面是帧中继网络中一些相关述语如下：

DTE（Data Terminal Equipment）数据终端设备：通常指的是用户侧的主机或终端等。

DCE（Data Circuit-terminating Equipment）数据电路终结设备：为用户设备提供接入的设备，属于网络设备，如帧中继交换机。

DLCI（Data Link Connection Identifier）数据链路连接标识：虚链路接口的标识，帧中继能够在单一物理传输线路上提供多条虚电路，虚电路通过DLCI来区分。

PVC（Permanent Virtual Circuit）永久虚电路：永久虚电路是指给用户提供固定的虚电路，该电路一旦建立，则链路永远生效，除非管理员手动删除。PVC用于两端之间频繁的、流量稳定的数据传输。

逆向地址解析协议（Inverse ARP）的主要功能是求解每条虚电路连接的对端设备的IP地址。如果知道了某条虚电路连接的对端设备的IP协议地址，在本地就可以生成对端IP地址与DLCI的映射（MAP），从而避免手工配置地址映射。

# 实验目的

* + 掌握帧中继交换机的配置
  + 掌握动态映射的配置
  + 掌握静态映射的配置
  + 掌握子接口和DLCI的映射配置

# 实验内容

本实验模拟企业网络场景，公司A的总部和分部分别设在不同地方，总部路由器R1和分部路由器R2通过帧中继网络相连，总部与分部之间申请了一条PVC。由于业务的发展，公司A与公司B有了密切的业务来往，公司B路由器R3也采用帧中继并使用动态映射方式与公司A相连，即只能与公司A总部直接通信。现需要采用帧中继子接口配置和静态路由使R3能通过的R1访问R2，实现全网全通。

# 实验拓扑



图11-6 帧中继基本配置拓扑图

# 实验编址表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 | DLCI |
| R1(AR1220) | Serial 1/0/0 | 10.1.1.1 | 255.255.255.0 | N/A | 102 |
| Serial 1/0/0.1 | 20.1.1.1 | 255.255.255.0 | N/A | 103 |
| R2(AR1220) | Serial 1/0/0 | 10.1.1.2 | 255.255.255.0 | N/A | 201 |
| R3(AR1220) | Serial 1/0/0 | 20.1.1.3 | 255.255.255.0 | N/A | 301 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置。

在帧中继交换机上配置两条PVC，R1和R2一条，R1和R3一条。

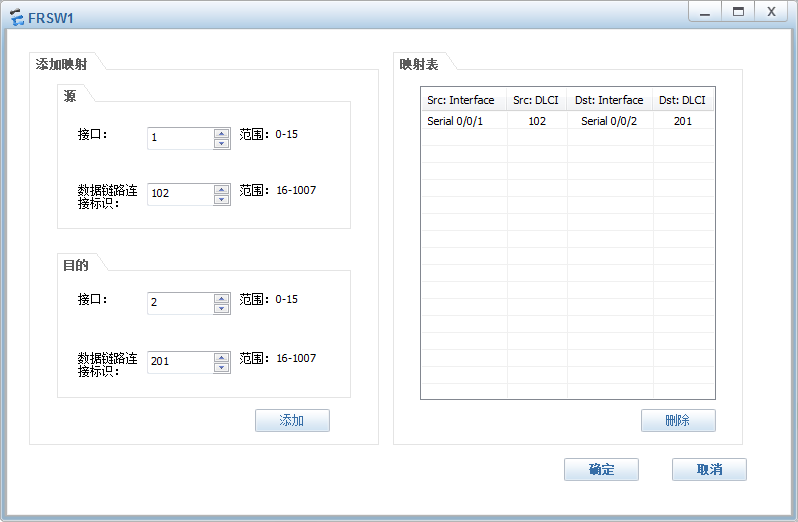


图11-7

如上图所示，在帧中继交换机上建立一条PVC，这条PVC在S 0/0/1接口上分配DLCI为102，在S 0/0/2接口分配DLCI 201，二者同属于一条PVC。

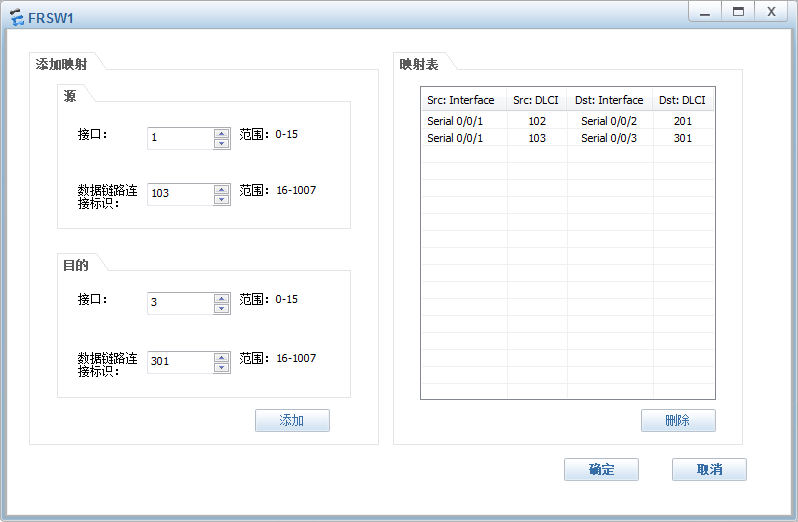


图11-8

如上图所示，在帧中继交换机上建立另一条PVC，这条PVC在S 0/0/1接口上分配DLCI为103，在S 0/0/3接口分配DLCI 301，二者同属于另一条PVC。

## 静态与动态映射的配置

帧中继接口在转发数据包时必须查找帧中继地址映射表来确定下一跳的DLCI。地址映射表中存放对端IP地址和下一跳的的DLCI的映射关系。只有找到相应的映射表项，才能完成二层帧中继报头的封装，这个机制类似于以太网中的ARP机制。该地址映射表可以手动配置（静态），也可以使用Inverse ARP协议来自动建立（动态）。

公司A总部使用动态映射，在R1的S 1/0/0接口配置链路层协议为FR，并使用命令**fr inarp**允许帧中继逆向地址解析功能自动生成地址映射表。

[R1]interface Serial 1/0/0

[R1-Serial1/0/0]link-protocol fr

Warning: The encapsulation protocol of the link will be changed. Continue? [Y/N]:y

[R1-Serial1/0/0]fr inarp

注意，默认情况下，串行接口使用的链路层协议为PPP协议，当试图改变接口默认的封装方式的时候，路由器会弹出一个警告，选择y即可。此外，帧中继接口的逆向地址解析功能默认是开启的，所以**fr inarp**命令可以不配置。

公司A分部由于只需要与总部通信即可，使用静态映射，在R2的S 1/0/0接口下配置链路层协议为FR，关闭逆向解析功能，使用命令**fr map ip**手工配置与R1的IP地址与DLCI的静态映射。

[R2]interface Serial 1/0/0

[R2-Serial1/0/0]link-protocol fr

Warning: The encapsulation protocol of the link will be changed. Continue? [Y/N]:y

[R2-Serial1/0/0]undo fr inarp

[R2-Serial1/0/0]fr map ip 10.1.1.1 201

将R1的IP地址与R2本端DLCI 201配置为一条静态地址映射，即R2通过下一跳DLCI 201的来访问R1。

默认情况下，帧中继不支持广播或组播数据的转发。如果需要在帧中继上运行一些动态路由协议，比如OSPF协议，需要在静态映射后面添加broadcast参数，从而使PVC能够正常发送来自路由协议的广播或组播流量。

[R2]interface Serial 1/0/0

[R2-Serial1/0/0]fr map ip 10.1.1.1 201 broadcast

配置完成后，在R1和R2上使用命令**display fr pvc-info**查看PVC的建立情况。

<R1>display fr pvc-info

PVC statistics for interface Serial1/0/0 (DTE, physical UP)

DLCI = 102, USAGE = UNUSED (00000000), Serial1/0/0

create time = 2013/06/28 07:11:45, status = ACTIVE

InARP = Enable, PVC-GROUP = NONE

in packets = 0, in bytes = 0

out packets = 50, out bytes = 1500

DLCI = 103, USAGE = UNUSED (00000000), Serial1/0/0

create time = 2013/06/28 07:11:45, status = ACTIVE

InARP = Enable, PVC-GROUP = NONE

in packets = 0, in bytes = 0

out packets = 45, out bytes = 1350

<R2>display fr pvc-info

PVC statistics for interface Serial1/0/0 (DTE, physical UP)

DLCI = 201, USAGE = LOCAL (00000100), Serial1/0/0

create time = 2013/06/28 07:11:45, status = ACTIVE

InARP = Disable, PVC-GROUP = NONE

in packets = 0, in bytes = 0

out packets = 50, out bytes = 1500

可以观察到，R1上有两条PVC，而且都为激活状态。R2上的PVC也为激活状态。

使用**ping**命令测试R1与R2之间的连通性。

<R1>ping 10.1.1.2

PING 10.1.1.2: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=255 time=540 ms

Reply from 10.1.1.2: bytes=56 Sequence=2 ttl=255 time=60 ms

Reply from 10.1.1.2: bytes=56 Sequence=3 ttl=255 time=70 ms

Reply from 10.1.1.2: bytes=56 Sequence=4 ttl=255 time=20 ms

Reply from 10.1.1.2: bytes=56 Sequence=5 ttl=255 time=50 ms

……

此时R1和R2已经正常通信。

## 子接口配置和静态路由

由于业务需要，公司B需要能和公司A互相通信。

公司B和公司A总部之间互连IP网段使用20.1.1.0/24。在R3的S 1/0/0接口配置链路层协议为FR，并维持默认开启的逆向地址解析功能。

[R3]interface s1/0/0

[R3-Serial1/0/0]ip address 20.1.1.3 24

[R3-Serial1/0/0]link-protocol fr

配置完成后，在R3上使用命令**display fr pvc-info**查看PVC建立的情况。

<R3>display fr pvc-info

PVC statistics for interface Serial1/0/0 (DTE, physical UP)

DLCI = 301, USAGE = UNUSED (00000000), Serial1/0/0

create time = 2013/06/28 08:07:06, status = ACTIVE

InARP = Enable, PVC-GROUP = NONE

in packets = 0, in bytes = 0

out packets = 65, out bytes = 2298

可以观察到，此时R3的PVC已经激活。

为实现与R3的互相通信，需要在R1上创建子接口S 1/0/0.1，配置与R3同网段的IP地址，并手工指定本地DLCI配置虚电路。

[R1]interface Serial 1/0/0.1

[R1-Serial1/0/0.1]ip address 20.1.1.1 24

[R1-Serial1/0/0.1]fr dlci 103

默认情况下，帧中继交换机分配的DLCI都关联到用户设备的物理接口上，而子接口关联的DLCI需要手动指定。

配置完成后，测试R1与R3间能否正常通信。

<R1>ping 20.1.1.3

PING 20.1.1.3: 56 data bytes, press CTRL\_C to break

Reply from 20.1.1.3: bytes=56 Sequence=1 ttl=255 time=50 ms

Reply from 20.1.1.3: bytes=56 Sequence=2 ttl=255 time=40 ms

Reply from 20.1.1.3: bytes=56 Sequence=3 ttl=255 time=50 ms

Reply from 20.1.1.3: bytes=56 Sequence=4 ttl=255 time=40 ms

Reply from 20.1.1.3: bytes=56 Sequence=5 ttl=255 time=30 ms

……

可以观察到，R1和R3已经通信正常，测试R2与R3间能否正常通信。

<R3>ping 10.1.1.2

PING 10.1.1.2: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

……

无法正常通信，这是因为R2与R3不在同一个网段上，需要有到达对方的路由才能连通。

为了使R2和R3能够互相通信，在R3上配置静态路由，目的地址为R2，下一跳为R1的子接口地址，同样在R2上也需要配置静态路由，目的地址为R3，下一跳为R1的S 1/0/0接口地址。

[R3]ip route-static 10.1.1.2 32 20.1.1.1

[R2]ip route-static 20.1.1.3 32 10.1.1.1

由此使R2与R3之间可以通过R1来通信，使用**ping**命令检查它们间的连通性。

<R3>ping 10.1.1.2

PING 10.1.1.2: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=254 time=110 ms

Reply from 10.1.1.2: bytes=56 Sequence=2 ttl=254 time=90 ms

Reply from 10.1.1.2: bytes=56 Sequence=3 ttl=254 time=110 ms

Reply from 10.1.1.2: bytes=56 Sequence=4 ttl=254 time=90 ms

Reply from 10.1.1.2: bytes=56 Sequence=5 ttl=254 time=90 ms

……

可以观察到，R2和R3间已经能够正常通信。使用**tracert** 命令查看它们之间的路径。

<R3>tracert 10.1.1.2

traceroute to 10.1.1.2(10.1.1.2), max hops: 30 ,packet length: 40,press CTRL\_C to break

1 20.1.1.1 50 ms 40 ms 40 ms

2 10.1.1.2 140 ms 90 ms 70 ms

可以观察到，R3去往R2的流量经过了R1。至此公司A与B所有的设备间都能正常通信。

# 思考

帧中继中动态映射的过程是怎么样的？它和ARP机制的区别在哪里？

11.4 OSPF在帧中继网络中的配置

# 原理概述

OSPF将网络分为四种不同的类型，即Point-to-Point、Broadcast、NBMA及Point-to-MultiPoint，不同网络类型下OSPF的工作机制不一样。比如在Broadcast网络中OSPF能够直接建立邻居邻接关系，在NBMA网络中默认必须手工指定邻居等。在实际网络中，可通过配置接口的网络类型来强制改变默认的接口的网络类型。在帧中继的环境中，OSPF默认的网络类型是NBMA。

# 实验目的

* + 掌握OSPF在帧中继网络中的配置方法
  + 理解Hub-Spoke组网架构
  + 掌握在帧中继网络中排除OSPF故障的方法

# 场景

某公司的网络使用OSPF协议，该公司由一个总部和两个分支机构组成。如图所示，R1为总部路由器，R2和R3分别是两个分支机构的出口路由器。两分支机构都是通过租用运营商的帧中继虚电路来与总部通信。但为了节省成本，两个分支机构间没有直接互联的虚电路，即典型的Hub-Spoke组网架构，R1为Hub端设备，R2，R3为Spoke端设备。

# 实验拓扑

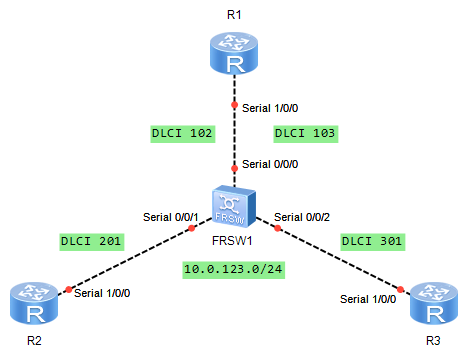


图11-9 配置OSPF帧中继网络拓扑图

# 实验编址表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 | DLCI |
| R1(AR2220) | Loopback 0 | 10.1.1.1 | 255.255.255.255 | N/A | N/A |
| Serial 1/0/0 | 10.0.123.1 | 255.255.255.0 | N/A | 102/103 |
| R2(AR2220) | Loopback 0 | 10.1.2.2 | 255.255.255.255 | N/A | N/A |
| Serial 1/0/0 | 10.0.123.2 | 255.255.255.0 | N/A | 201 |
| R3(AR2220) | Loopback 0 | 10.1.3.3 | 255.255.255.255 | N/A | N/A |
| Serial 1/0/0 | 10.0.123.3 | 255.255.255.0 | N/A | 301 |

# 实验步骤

## 基本配置

在公司总部路由器R1，和两个分部的路由器R2和R3上配置帧中继接口，关闭帧中继逆向地址解析功能，根据实验编址表进行相应的基本IP地址配置，配置帧中继静态地址映射。环回接口的掩码为32位，用来模拟公司总部和分部的主机。注意将R1设置为DR，调整其DR优先级为100。

[R1]interface Serial 1/0/0

[R1-Serial1/0/0]link-protocol fr

[R1-Serial1/0/0]ip address 10.0.123.1 24

[R1-Serial1/0/0]undo fr inarp

[R1-Serial1/0/0]fr map ip 10.0.123.2 102

[R1-Serial1/0/0]fr map ip 10.0.123.3 103

[R1-Serial1/0/0]ospf dr-priority 100

[R1-Serial1/0/0]interface loopback 0

[R1-LoopBack0]ip address 10.1.1.1 32

[R2]interface Serial 1/0/0

[R2-Serial1/0/0]link-protocol fr

[R2-Serial1/0/0]ip address 10.0.123.2 24

[R2-Serial1/0/0]undo fr inarp

[R2-Serial1/0/0]fr map ip 10.0.123.1 201

[R2-Serial1/0/0]interface LoopBack 0

[R2-LoopBack0]ip address 10.1.2.2 32

[R3]interface Serial 1/0/0

[R3-Serial1/0/0]link-protocol fr

[R3-Serial1/0/0]ip address 10.0.123.3 24

[R3-Serial1/0/0]undo fr inarp

[R3-Serial1/0/0]fr map ip 10.0.123.1 301

[R3-Serial1/0/0]interface LoopBack 0

[R3-LoopBack0]ip address 10.1.3.3 32

配置完成后，检查帧中继的虚电路状态和映射表。

<R1>display fr pvc-info

PVC statistics for interface Serial1/0/0 (DTE, physical UP)

DLCI = 102, USAGE = LOCAL (00000100), Serial1/0/0

create time = 2013/05/30 20:35:14, status = ACTIVE

InARP = Disable, PVC-GROUP = NONE

in BECN = 0, in FECN = 0

in packets = 1, in bytes = 30

out packets = 2, out bytes = 60

DLCI = 103, USAGE = LOCAL (00000100), Serial1/0/0

create time = 2013/05/30 20:35:14, status = ACTIVE

InARP = Disable, PVC-GROUP = NONE

in BECN = 0, in FECN = 0

in packets = 5, in bytes = 440

out packets = 7, out bytes = 500

可以观察到，PVC处于ACTIVE状态表示正常。

<R1>display fr map-info

Map Statistics for interface Serial1/0/0 (DTE)

DLCI = 102, IP 10.0.123.2, Serial1/0/0

create time = 2013/05/30 20:35:41, status = ACTIVE

encapsulation = ietf, vlink = 1

DLCI = 103, IP 10.0.123.3, Serial1/0/0

create time = 2013/05/30 20:35:52, status = ACTIVE

encapsulation = ietf, vlink = 2

检查R1与R2，R1与R3间的网络连通性。

<R1>ping 10.0.123.2

PING 10.0.123.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.123.2: bytes=56 Sequence=1 ttl=255 time=60 ms

Reply from 10.0.123.2: bytes=56 Sequence=2 ttl=255 time=20 ms

Reply from 10.0.123.2: bytes=56 Sequence=3 ttl=255 time=40 ms

Reply from 10.0.123.2: bytes=56 Sequence=4 ttl=255 time=30 ms

Reply from 10.0.123.2: bytes=56 Sequence=5 ttl=255 time=30 ms

--- 10.0.123.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 20/36/60 ms

<R1>ping 10.0.123.3

PING 10.0.123.3: 56 data bytes, press CTRL\_C to break

Reply from 10.0.123.3: bytes=56 Sequence=1 ttl=255 time=30 ms

Reply from 10.0.123.3: bytes=56 Sequence=2 ttl=255 time=30 ms

Reply from 10.0.123.3: bytes=56 Sequence=3 ttl=255 time=50 ms

Reply from 10.0.123.3: bytes=56 Sequence=4 ttl=255 time=40 ms

Reply from 10.0.123.3: bytes=56 Sequence=5 ttl=255 time=40 ms

--- 10.0.123.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/38/50 ms

此时通信正常。

## 在帧中继上搭建OSPF网络

在R1、R2和R3上配置OSPF协议。由于网络拓扑简单，采用OSPF的单区域配置即可，指定它们各自的环回接口地址作为Router-ID，所有网段都属于区域0内。

[R1]ospf 1 router-id 10.1.1.1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 10.0.123.1 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 10.1.1.1 0.0.0.0

[R2]ospf 1 router-id 10.1.2.2

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 10.0.123.2 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 10.1.2.2 0.0.0.0

[R3]ospf 1 router-id 10.1.3.3

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 10.0.123.3 0.0.0.255

[R3-ospf-1-area-0.0.0.0]network 10.1.3.3 0.0.0.0

配置完成后，查看OSPF的邻居建立情况。

<R1>display ospf peer

OSPF Process 1 with Router-ID 10.1.1.1

发现无法正常建立邻居，这是明显的网络故障，现在网络管理员需要立刻进行分析排除故障。排障的时候需要注意遵循从底层逐步往上层排查的顺序，即先物理层线缆是否正常，然后检查二层链路的连通性，再检查三层路由协议的运行情况，最后检查高层相关应用是否正常。

物理层检查这里省略，首先测试直连线路的连通性。

[R1]ping 10.0.123.2

PING 10.0.123.2: 56 data bytes, press CTRL\_C to break

Reply from 10.0.123.2: bytes=56 Sequence=1 ttl=255 time=100 ms

Reply from 10.0.123.2: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 10.0.123.2: bytes=56 Sequence=3 ttl=255 time=10 ms

Reply from 10.0.123.2: bytes=56 Sequence=4 ttl=255 time=10 ms

Reply from 10.0.123.2: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.0.123.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/28/100 ms

直连链路连通性没有问题。再查看三层路由协议，即相应接口否被通告到OSPF进程中。

[R1]display ospf interface

OSPF Process 1 with Router-ID 10.1.1.1

Interfaces

Area: 0.0.0.0 (MPLS TE not enabled)

IP Address Type State Cost Pri DR BDR

10.0.123.1 NBMA DR 48 100 10.0.123.1 0.0.0.0

10.1.1.1 P2P P-2-P 0 1 0.0.0.0 0.0.0.0

观察到所有接口已经被通告进入OSPF进程。

此时可以对R1的S 1/0/0接口进行抓包分析，查看协议的运行情况。



图11-10

发现R1始终没有向外发送OSPF数据包。这是由于OSPF在帧中继上默认的网络类型为NBMA，即非广播多路访问。这种网络类型的特点是不支持广播和组播的数据包，而OSPF协议默认是采用组播方式发送报文，所以设备的OSPF报文无法在帧中继链路上进行发送，导致没有成功建立邻居关系。

这时可以采用命令**peer**手工指定OSPF邻居，采用单播方式发送报文。

[R1]ospf 1

[R1-ospf-1]peer 10.0.123.2

[R1-ospf-1]peer 10.0.123.3

[R2]ospf 1

[R2-ospf-1]peer 10.0.123.1

[R3]ospf 1

[R3-ospf-1]peer 10.0.123.1

配置完成后，再次检查OSPF的邻居关系状态。

<R1>display ospf peer brief

OSPF Process 1 with Router-ID 10.0.1.1

Peer Statistic Information

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

Area Id Interface Neighbor id State

0.0.0.0 Serial1/0/0 10.1.2.2 Full

0.0.0.0 Serial1/0/0 10.1.3.3 Full

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

可以观察到，这时R1与R2、R3都建立了完全的邻接关系。再查看R1，R2，R3的路由表。

<R1>display ip routing protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 2 Routes : 2

OSPF routing table status : <Active>

Destinations : 2 Routes : 2

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.2.2/32 OSPF 10 48 D 10.0.123.2 Serial1/0/0

10.1.3.3/32 OSPF 10 48 D 10.0.123.3 Serial1/0/0

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

<R2>display ip routing protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 2 Routes : 2

OSPF routing table status : <Active>

Destinations : 2 Routes : 2

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.1/32 OSPF 10 48 D 10.0.123.1 Serial1/0/0

10.1.3.3/32 OSPF 10 48 D 10.0.123.3 Serial1/0/0

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

<R3>display ip routing-table protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 2 Routes : 2

OSPF routing table status : <Active>

Destinations : 2 Routes : 2

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.1/32 OSPF 10 48 D 10.0.123.1 Serial1/0/0

10.1.2.2/32 OSPF 10 48 D 10.0.123.2 Serial1/0/0

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

可以观察到此时的R1、R2、R3路由表中都互相接收到了各自环回口所在网段的路由条目。测试环回口之间的连通性。

<R1> ping -a 10.1.1.1 10.1.2.2

PING 10.1.2.2: 56 data bytes, press CTRL\_C to break

Reply from 10.1.2.2: bytes=56 Sequence=1 ttl=255 time=10 ms

Reply from 10.1.2.2: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 10.1.2.2: bytes=56 Sequence=3 ttl=255 time=10 ms

Reply from 10.1.2.2: bytes=56 Sequence=4 ttl=255 time=20 ms

Reply from 10.1.2.2: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.1.2.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/12/20 ms

<R1>ping -a 10.1.1.1 10.1.3.3

PING 10.1.3.3: 56 data bytes, press CTRL\_C to break

Reply from 10.1.3.3: bytes=56 Sequence=1 ttl=255 time=20 ms

Reply from 10.1.3.3: bytes=56 Sequence=2 ttl=255 time=10 ms

Reply from 10.1.3.3: bytes=56 Sequence=3 ttl=255 time=10 ms

Reply from 10.1.3.3: bytes=56 Sequence=4 ttl=255 time=10 ms

Reply from 10.1.3.3: bytes=56 Sequence=5 ttl=255 time=10 ms

--- 10.1.3.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 10/12/20 ms

此时R1与R2，R1与R3之间的环回口通信正常。测试R2与R3环回口之间的通信情况。

<R2>ping -a 10.1.2.2 10.1.3.3

PING 10.1.3.3: 56 data bytes, press CTRL\_C to break

Request time out

Request time out

Request time out

Request time out

Request time out

……

发现R2却无法ping通R3的环回口。此时网络管理员需要再次进行排障。物理链路和二层链路的连通性测试这里省略。首先查看R2上的OSPF路由条目。

<R2>display ip routing protocol ospf

Route Flags: R - relay, D - download to fib

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

Public routing table : OSPF

Destinations : 2 Routes : 2

OSPF routing table status : <Active>

Destinations : 2 Routes : 2

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.1/32 OSPF 10 48 D 10.0.123.1 Serial1/0/0

10.1.3.3/32 OSPF 10 48 D 10.0.123.3 Serial1/0/0

OSPF routing table status : <Inactive>

Destinations : 0 Routes : 0

观察到去往10.1.3.3的网段下一跳地址是10.0.123.3。

然后在R2上查看帧中继映射关系。

<R2>display fr map-info

Map Statistics for interface Serial1/0/0 (DTE)

DLCI = 201, IP 10.0.123.1, Serial1/0/0

create time = 2013/06/23 20:06:07, status = ACTIVE

encapsulation = ietf, vlink = 1

可以观察到，此时没有关于10.0.123.3的映射，如果R2要发送数据包至下一跳10.0.123.3，但无法知晓该从哪条PVC上进行发送和封装。可以使用PVC复用技术解决此问题，需要在R1的S 1/0/0接口下添加一条帧中继静态映射， 通过R1与R2的PVC去往10.0.123.3。

[R2]interface Serial 1/0/0

[R2-Serial1/0/0]fr map ip 10.0.123.3 201

同样需要在R3上也添加关于10.0.123.2的相关映射。

[R3]interface Serial 1/0/0

[R3-Serial1/0/0]fr map ip 10.0.123.2 301

配置完成后，再在R2上查看帧中继映射关系。

<R2>display fr map-info

Map Statistics for interface Serial1/0/0 (DTE)

DLCI = 201, IP 10.0.123.1, Serial1/0/0

create time = 2013/06/23 20:06:07, status = ACTIVE

encapsulation = ietf, vlink = 1

DLCI = 201, IP 10.0.123.3, Serial1/0/0

create time = 2013/06/23 20:06:07, status = ACTIVE

encapsulation = ietf, vlink = 2

此时可以观察到已经添加上了相应映射。

再次测试R2到R3环回口的连通性。

[R2]ping -a 10.1.2.2 10.1.3.3

PING 10.1.3.3: 56 data bytes, press CTRL\_C to break

Reply from 10.1.3.3: bytes=56 Sequence=1 ttl=254 time=50 ms

Reply from 10.1.3.3: bytes=56 Sequence=2 ttl=254 time=20 ms

Reply from 10.1.3.3: bytes=56 Sequence=3 ttl=254 time=30 ms

Reply from 10.1.3.3: bytes=56 Sequence=4 ttl=254 time=20 ms

Reply from 10.1.3.3: bytes=56 Sequence=5 ttl=254 time=30 ms

--- 10.1.3.3 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 20/30/50 ms

此时通信正常，所有问题解决。

# 思考

在上文中将R1的DR优先级设置成了100，为什么要这么做？

1. DHCP

12.1 配置基于接口地址池的DHCP

# 原理概述

随着网络规模的扩大和网络复杂程度的提高，经常出现计算机位置变化（如便携机或无线网络）和计算机数量超过可分配的IP地址的情况。DHCP（Dynamic Host Configuration Protocol，动态主机配置协议）就是为满足这些需求而发展起来的。DHCP协议采用客户端/服务器（Client/Server）方式工作，DHCP Client向DHCP Server动态地请求配置信息，DHCP Server根据策略返回相应的配置信息（如IP地址等）。

DHCP客户端首次登录网络时，主要通过四个阶段与DHCP服务器建立联系。

发现阶段，即DHCP客户端寻找DHCP服务器的阶段。客户端以广播方式发送DHCP\_Discover报文，只有DHCP服务器才会进行响应；

提供阶段，即DHCP服务器提供IP地址的阶段。DHCP服务器接收到客户端的DHCP\_Discover报文后，从IP地址池中挑选一个尚未分配的IP地址分配给客户端，向该客户端发送包含出租IP地址和其它设置的DHCP\_Offer报文；

选择阶段，即DHCP客户端选择IP地址的阶段。如果有多台DHCP服务器向该客户端发来DHCP\_Offer报文，客户端只接受第一个收到的DHCP\_Offer报文，然后以广播方式向各DHCP服务器回应DHCP\_Request报文。

确认阶段，即DHCP服务器确认所提供IP地址的阶段。当DHCP服务器收到DHCP客户端回答的DHCP\_Request报文后，便向客户端发送包含它所提供的IP地址和其它设置的DHCP\_ACK确认报文。

# 实验目的

* + 掌握DHCP Server配置方法
  + 掌握基于接口地址池的DHCP Server配置方法
  + 掌握配置DHCP租期/不参与自动分配地址/DNS服务器地址方法
  + 掌握配置和检测DHCP客户端的方法

# 实验内容

本实验将路由器R1模拟成为公司DHCP Server，该公司两个部门市场部和财务部下的PC通过DHCP的方式自动配置IP地址。网络管理员配置客户端PC通过接口地址池的方式自动获取IP地址。

# 实验拓扑

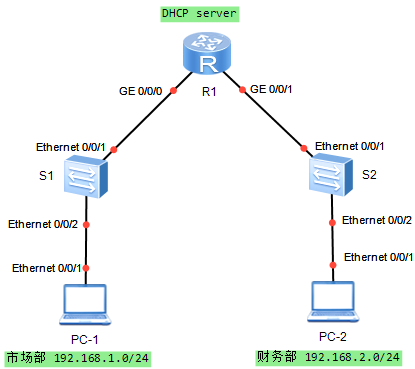


图12-1 配置基于接口地址池的DHCP拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 192.168.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 192.168.2.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet0/0/1 | DHCP获取 | DHCP获取 | DHCP获取 |
| PC-2 | Ethernet0/0/1 | DHCP获取 | DHCP获取 | DHCP获取 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，由于PC是通过DHCP自动获取地址，暂时无法测试连通性。交换机为二层设备，无需配置IP地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ip address 192.168.1.254 24

[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]ip address 192.168.2.254 24

## 基于接口配置DHCP Server功能

在R1上开启DHCP功能。

[R1]dhcp enable

在R1的GE 0/0/0和GE 0/0/1接口，配置命令**dhcp select interface**开启接口的DHCP服务功能，指定从接口地址池分配地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]dhcp select interface

[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]dhcp select interface

接口地址池可动态分配IP地址，范围就是接口的IP地址所在网段，且只在此接口下有效。当DHCP服务器接收到DHCP客户端的请求报文后，DHCP服务器将会使用该接口的地址网段给客户端分配地址。

## 配置基于接口的DHCP Server租期/DNS服务器地址

在R1的GE 0/0/0接口上使用命令**dhcp server lease**配置DHCP服务器接口地址池中IP地址的租用有效期限为2天，GE 0/0/1接口不修改，使用默认值1天，超过租期后该地址将会重新分配。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]dhcp server lease day 2

在GE 0/0/0接口上使用命令**dhcp server excluded-ip-address**配置接口地址池中不参与自动分配的IP地址范围为192.168.1.1到192.168.1.10。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]dhcp server excluded-ip-address 192.168.1.1 192.168.1.10

有些地址需要分配给其他的服务，如DNS服务器或HTTP服务器等需要手工静态配置的IP地址，就不能再动态分配给客户端使用，可以执行该命令配置地址池中不参与自动分配的IP地址（默认该地址池所有地址参与自动分配，此命令作为可选命令）。

当DHCP服务器收到客户端的DHCP请求时，DPCP服务器将会选择地址池中空闲的IP地址分配给客户端。GE 0/0/0接口地址池中192.168.1.1-192.168.1.10不参与分配，而GE 0/0/1接口没有配置该命令，因此可以分配的IP地址范围192.168.2.1-192.168.2.253（不包括本接口地址）。

在GE 0/0/1接口上使用命令**dhcp server dns-list**指定接口地址池下的DNS服务器，为PC-2自动分配DNS服务器地址为8.8.8.8。

[R1-GigabitEthernet0/0/1]dhcp server dns-list 8.8.8.8

## 配置DHCP Client

设置PC使用DHCP配置地址。

打开PC-1的基础配置界面，在IPv4配置栏中选择DHCP，然后点击右下角“应用”。

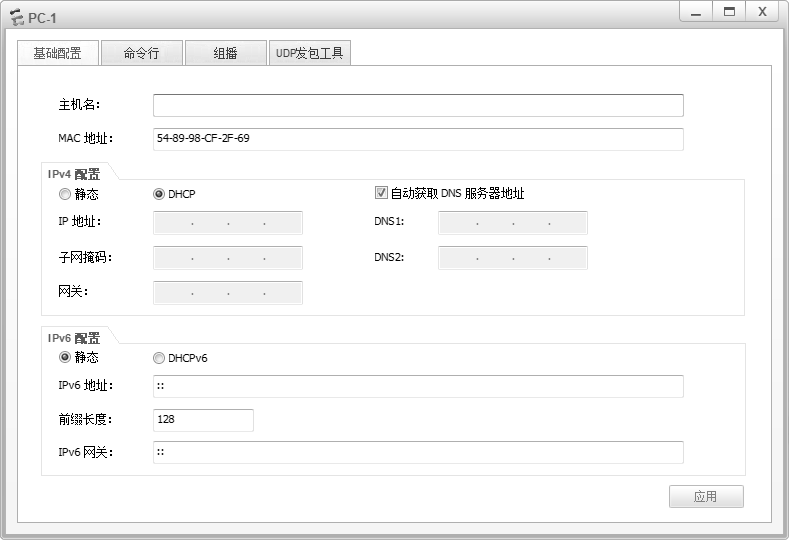


图12-2

选择PC-1中的“命令行”界面，输入命令**ipconfig**来查看接口的IP地址。



图12-3

通过观察发现PC-1已经通过DHCP Server获取到一个IPv4地址192.168.1.253，网关地址为路由器的接口地址192.168.1.254。

在R1上使用命令**display ip pool**查看DHCP地址池当中的地址分配情况。

[R1]display ip pool

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

Pool-name : GigabitEthernet0/0/0

Pool-No : 0

Position : Interface Status : Unlocked

Gateway-0 : 192.168.1.254

Mask : 255.255.255.0

VPN instance : --

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

Pool-name : GigabitEthernet0/0/1

Pool-No : 1

Position : Interface Status : Unlocked

Gateway-0 : 192.168.2.254

Mask : 255.255.255.0

VPN instance : --

IP address Statistic

Total :506

Used :1 Idle :495

Expired :0 Conflict :0 Disable :10

通过该命令观察到目前为基于接口的地址池，由于有两个接口启用DHCP功能，所以地址池也有两个，Pool-name分别为GE 0/0/0，GE 0/0/1。在DHCP Server地址池中，网关为192.168.1.254，掩码为255.255.255.0，IP地址池总共可以分配506个地址（除了路由器接口地址），已经使用了一个，空闲地址为495个，其中地址池中有10个地址是不参与分配的。

配置PC-2时参考配置PC-1的方法，选择通过DHCP配置地址。

选择PC-2中的“命令行”界面，输入命令**ipconfig**来查看接口的IP地址。



图12-4

通过观察发现PC-2以通过DHCP Server获取到一个IPv4地址 192.168.2.253，网关地址为路由器的接口地址192.168.2.254，DNS服务器地址为8.8.8.8。DHCP地址池中的地址分配情况查看省略。

# 思考

DHCP Server 从地址池分配IP的顺序如何，是按顺序还是随机的？DHCP Server如何防范地址冲突的问题？

12.2 配置基于全局地址池的DHCP

# 原理概述

基于接口地址池的DHCP服务器，连接这个接口网段的用户都从该接口地址池中获取IP地址等配置信息，由于地址池绑定在特定的接口上，可以限制用户的使用条件，在保障了安全性的同时也存在一定局限性。当用户从不同接口接入DHCP服务器且需要从同一个地址池里获取IP地址时，就需要配置基于全局地址池的DHCP。

配置基于全局地址池的DHCP服务器，从所有接口上连接的用户都可以选择该地址池中的地址，也就是说全局地址池是一个公共地址池。在DHCP服务器上创建地址池并配置相关属性（包括地址范围、地址租期、不参与自动分配的IP地址等），在配置接口工作在全局地址池模式。路由器支持工作在全局地址池模式的接口有三层接口及其子接口、三层Ethernet接口及其子接口、三层Eth-trunk接口及其子接口和VLANIF接口。

# 实验目的

* + 掌握DHCP Server配置方法
  + 掌握基于全局地址池的DHCP Server配置方法
  + 掌握配置DHCP租期/网关地址/不参与自动分配地址方法
  + 掌握配置和检测DHCP客户端的方法

# 场景

本实验将路由器R1模拟成为公司DHCP Server，配置全局地址池，该公司两个部门市场部和财务部下的PC通过DHCP的方式自动配置IP地址。

# 实验拓扑

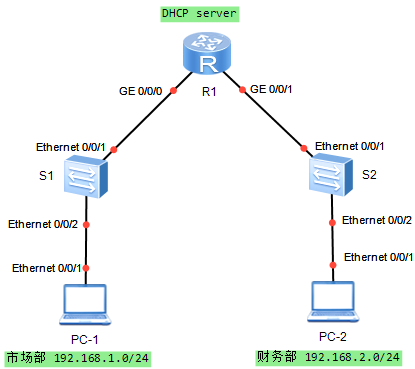


图12-5 配置基于全局地址池的DHCP拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 192.168.1.254 | 255.255.255.0 | N/A |
| GE 0/0/1 | 192.168.2.254 | 255.255.255.0 | N/A |
| PC-1 | Ethernet0/0/1 | DHCP获取 | DHCP获取 | DHCP获取 |
| PC-2 | Ethernet0/0/1 | DHCP获取 | DHCP获取 | DHCP获取 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，由于PC是通过DHCP自动获取地址，暂时无法测试连通性。交换机为二层设备，无需配置IP地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ip address 192.168.1.254 24

[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]ip address 192.168.2.254 24

## 配置基于全局地址池的DHCP Server

在R1上开启DHCP功能。

[R1]dhcp enable

使用命令**ip pool**创建一个全局地址池，地址池名称为huawei1。缺省情况下，设备上没有创建任何全局地址池。

[R1]ip pool huawei1

使用命令**network**配置全局地址池huawei1可动态分配的网段范围为192.168.1.0，如果不指定掩码，默认使用自然掩码，即24位掩码。该网段必需与路由器接口GE 0/0/0的IP地址为同一网段。

[R1-ip-pool-huawei1]network 192.168.1.0

使用命令**lease day**配置DHCP全局地址池下的地址租期。缺省情况下，IP地址租期为1天，对于不同的地址池，DHCP服务器可以指定不同的地址租用期限，但是同一地址池中的地址具有相同的租期。

[R1-ip-pool-huawei1]lease day 2

配置DHCP客户端的出口网关地址。

[R1-ip-pool-huawei]gateway-list 192.168.1.254

配置地址池中192.168.1.250到192.168.1.253这些地址不参与自动分配。

[R1-ip-pool-huawei1]excluded-ip-address 192.168.1.250 192.168.1.253

由于地址192.168.1.250到192.168.1.253不参与自动分配，而网关地址也不参与自动分配。DHCP服务器将会从地址池中由192.168.1.249开始往前分配。

配置DNS服务器地址。

[R1-ip-pool-huawei1]dns-list 8.8.8.8

开启接口的DHCP功能。使用该命令配置设备指定接口采用全局地址池为客户端分配IP地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]dhcp select global

由于路由器需要为两个不同部门分配IP地址，即需要两个全局地址池。

为财务部配置的全局地址池名称为：huawei2，IP网段为192.168.2.0，网关地址为192.168.2.254，DNS服务器地址为8.8.8.8，配置完成后在GE 0/0/1接口下面启用全局地址池的DHCP服务器模式。

[R1]ip pool huawei2

[R1-ip-pool-huawei2]network 192.168.2.0

[R1-ip-pool-huawei2]lease day 2

[R1-ip-pool-huawei2]gateway-list 192.168.2.254

[R1-ip-pool-huawei2]dns-list 8.8.8.8

[R1-ip-pool-huawei2]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]dhcp select global

配置完成后，查看IP地址池信息。

[R1]display ip pool

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

Pool-name : huawei1

Pool-No : 0

Position : Local Status : Unlocked

Gateway-0 : 192.168.1.254

Mask : 255.255.255.0

VPN instance : --

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

Pool-name : huawei2

Pool-No : 1

Position : Local Status : Unlocked

Gateway-0 : 192.168.2.254

Mask : 255.255.255.0

VPN instance : --

IP address Statistic

Total :506

Used :0 Idle :502

Expired :0 Conflict :0 Disable :4

通过命令观察到有两个地址池，其中一个地址池为huawei1，另外一个地址池为huawei2，地址池的总数为506个，使用了0个，空闲502个，有4个地址不参与分配。

## 配置DHCP Client

设置PC-1使用DHCP配置地址。

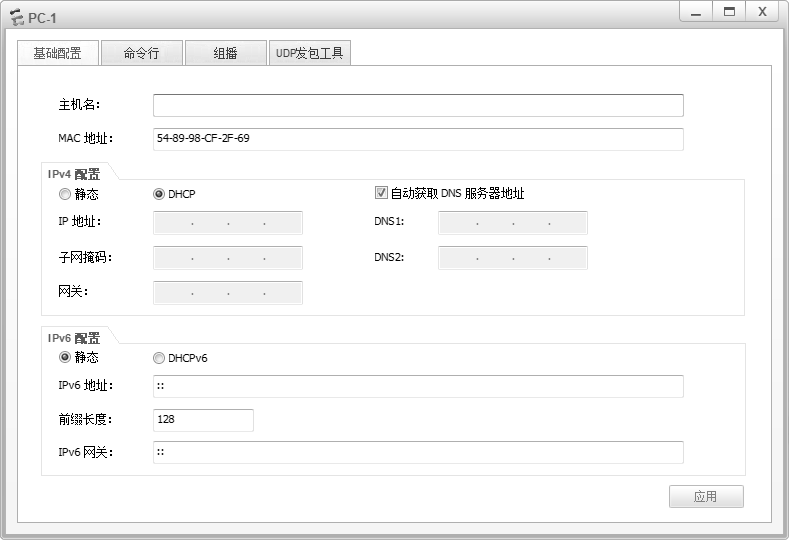


图12-6

打开PC-1的基础配置界面，在IPV4配置栏中选择DHCP，然后点击右下角“应用”。

选择PC-1中的“命令行”界面，输入命令：ipconfig来查看接口的IP地址。



图12-7

通过观察发现PC-1已经通过DHCP Server获取到一个IPv4地址 192.168.1.249，网关地址为192.168.1.254，DNS server地址为8.8.8.8。

验证路由器与PC的连通性。

[R1]ping 192.168.1.249

PING 192.168.1.249: 56 data bytes, press CTRL\_C to break

Reply from 192.168.1.249: bytes=56 Sequence=1 ttl=128 time=500 ms

Reply from 192.168.1.249: bytes=56 Sequence=2 ttl=128 time=180 ms

Reply from 192.168.1.249: bytes=56 Sequence=3 ttl=128 time=130 ms

Reply from 192.168.1.249: bytes=56 Sequence=4 ttl=128 time=90 ms

Reply from 192.168.1.249: bytes=56 Sequence=5 ttl=128 time=110 ms

--- 192.168.1.249 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 90/202/500 ms

通过观察路由器与PC之间可以正常通信。

设置PC-2使用DHCP配置地址



图12-8

打开PC-2的基础配置界面，在IPv4配置栏中选择DHCP，然后点击右下角“应用”。

选择PC-2中的“命令行”界面，输入命令：ipconfig来查看接口的IP地址。



图12-9

通过观察发现PC-2已经通过DHCP Server获取到一个IPv4地址 192.168.2.253，网关地址为192.168.2.254，DNS server地址为8.8.8.8。

[R1]ping 192.168.2.253

PING 192.168.2.253: 56 data bytes, press CTRL\_C to break

Reply from 192.168.2.253: bytes=56 Sequence=1 ttl=128 time=500 ms

Reply from 192.168.2.253: bytes=56 Sequence=2 ttl=128 time=180 ms

Reply from 192.168.2.253: bytes=56 Sequence=3 ttl=128 time=130 ms

Reply from 192.168.2.253: bytes=56 Sequence=4 ttl=128 time=90 ms

Reply from 192.168.2.253: bytes=56 Sequence=5 ttl=128 time=110 ms

--- 192.168.2.253 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 90/202/500 ms

通过观察路由器与PC之间可以正常通信。

# 思考

请问DHCP服务器在分配地址的时候是从该网段中最小的地址进行分配还是最大的地址进行分配，这样有什么好处？

12.3 配置DHCP中继

# 原理概述

由于在IP地址动态获取的过程中，客户端采用广播方式发送请求报文，而广播报文不能跨网段传送，因此DHCP只适用于DHCP客户端和服务器处于同一个网段内的情况。当多个网段都需要进行动态IP地址分配时，就需要在所有网段上都设置一个DHCP服务器，这显然是不易管理和维护的。

DHCP中继使客户端可以通过DHCP中继与其他网段的DHCP服务器通信，最终获取到IP地址。解决了DHCP客户端不能跨网段向服务器动态获取IP地址的问题，这样，在多个不同网络上的DHCP客户端可以使用同一个DHCP服务器，这样既节省了成本，又便于进行集中管理和维护。

路由器或三层交换机都可以充当DHCP中继设备。DHCP中继设备可以转发客户端和服务器间的DHCP报文，完成对客户端IP地址的动态分配。

# 实验目的

* + 理解DHCP中继的应用场景
  + 掌握DHCP中继的配置

# 实验内容

本实验模拟企业网络场景，某公司分部的网络由交换机S1和网关路由器R1组成，员工终端PC-1和PC-2都连接在S1上。公司要求分部内所有员工主机的IP地址都通过总部的DHCP服务器自动获取。分部网关路由器R1通过公网路由器R2访问公司总部的DHCP服务器R3。由于公司分部与总部不在同一个子网，需要在R1上配置DHCP中继，使分部内主机能跨网段从总部的DHCP服务器自动获取IP地址。

# 实验拓扑

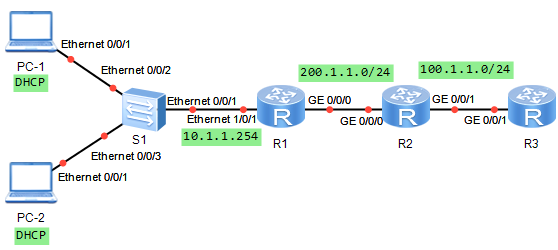


图12-10 配置DHCP中继拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | Ethernet 1/0/1 | 10.1.1.254 | 255.255.255.0 | N/A |
| GE 0/0/0 | 200.1.1.1 | 255.255.255.0 | N/A |
| R2(AR1220) | GE 0/0/0 | 200.1.1.2 | 255.255.255.0 | N/A |
| GE 0/0/1 | 100.1.1.2 | 255.255.255.0 | N/A |
| R3(AR1220) | GE 0/0/1 | 100.1.1.1 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IP地址配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping 200.1.1.2

PING 200.1.1.2: 56 data bytes, press CTRL\_C to break

Reply from 200.1.1.2: bytes=56 Sequence=1 ttl=255 time=40 ms

Reply from 200.1.1.2: bytes=56 Sequence=2 ttl=255 time=30 ms

Reply from 200.1.1.2: bytes=56 Sequence=3 ttl=255 time=50 ms

Reply from 200.1.1.2: bytes=56 Sequence=4 ttl=255 time=20 ms

Reply from 200.1.1.2: bytes=56 Sequence=5 ttl=255 time=50 ms

--- 200.1.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 20/38/50 ms

其余直连网段的连通性测试省略。

## 搭建OSPF网络

在公司路由器R1，R2，R3上都配置运行OSPF协议，所有网段都发布到区域0中。

[R1]ospf 1

[R1-ospf-1]area 0

[R1-ospf-1-area-0.0.0.0]network 200.1.1.0 0.0.0.255

[R1-ospf-1-area-0.0.0.0]network 10.1.1.0 0.0.0.255

[R2]ospf 1

[R2-ospf-1]area 0

[R2-ospf-1-area-0.0.0.0]network 200.1.1.0 0.0.0.255

[R2-ospf-1-area-0.0.0.0]network 100.1.1.0 0.0.0.255

[R3]ospf 1

[R3-ospf-1]area 0

[R3-ospf-1-area-0.0.0.0]network 100.1.1.0 0.0.0.255

配置完成后，查看路由表信息。

<R1>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 Direct 0 0 D 10.1.1.254 Ethernet1/0/1

10.1.1.254/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/1

100.1.1.0/24 OSPF 10 D 200.1.1.2 GigabitEthernet0/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

200.1.1.0/24 Direct 0 0 D 200.1.1.1 GigabitEthernet0/0/0

200.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

<R2>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 7 Routes : 7

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 OSPF 10 2 D 200.1.1.1 GigabitEthernet0/0/0

100.1.1.0/24 Direct 0 0 D 100.1.1.2 GigabitEthernet0/0/1

100.1.1.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

200.1.1.0/24 Direct 0 0 D 200.1.1.2 GigabitEthernet0/0/1

200.1.1.2/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/0

<R3>display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 6 Routes : 6

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.1.0/24 OSPF 10 3 D 100.1.1.2 GigabitEthernet0/0/1

100.1.1.0/24 Direct 0 0 D 100.1.1.1 GigabitEthernet0/0/1

100.1.1.1/32 Direct 0 0 D 127.0.0.1 GigabitEthernet0/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

200.1.1.0/24 OSPF 10 2 D 100.1.1.2 GigabitEthernet0/0/1

可以观察到，目前每台设备都可以正常获得路由信息，连通性测试省略。

## 配置DHCP服务器

总部路由器R3配置为DHCP服务器，负责为分部的网络分配IP地址。在R3上使用命令**dhcp enable**开启DHCP功能，创建全局地址池dhcp-pool，可分配IP地址范围为10.1.1.0/24，出口网关地址为10.1.1.254。并在面向DHCP中继设备的接口上开启DHCP服务功能，指定从全局地址池分配地址。

[R3]dhcp enable

Info: The operation may take a few seconds. Please wait for a moment.done.

[R3]ip pool dhcp-pool

Info:It's successful to create an IP address pool.

[R3-ip-pool-dhcp-pool]network 10.1.1.0 mask 255.255.255.0

[R3-ip-pool-dhcp-pool]gateway-list 10.1.1.254

[R3-ip-pool-dhcp-pool]interface GigabitEthernet0/0/1

[R3-GigabitEthernet0/0/1]dhcp select global

配置完成后，使用命令**display ip pool**查看IP地址池配置情况。

[R3]display ip pool

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

Pool-name : dhcp-pool

Pool-No : 0

Position : Local Status : Unlocked

Gateway-0 : 10.1.1.254

Mask : 255.255.255.0

VPN instance : --

IP address Statistic

Total :253

Used :0 Idle :253

Expired :0 Conflict :0 Disable :0

可以观察到，当前可用的地址除去网关IP以外还剩下253个可用，目前还没有PC动态申请IP地址。

## 配置DHCP中继

配置R1为DHCP中继设备，指定DHCP服务器为R3。

这时如果R1从E 0/0/1接口收到PC的DHCP广播请求包，R1作为DHCP中继设备会以单播形式转发请求包到中继所指明的DHCP服务器R3，服务器R3收到DHCP请求包后，会把分配的IP地址通过单播包返回给DHCP中继设备R1，中继设备R1再把地址信息发送给PC。所以在DHCP中继上需要指定所代理的DHCP服务器的地址。

配置指定DHCP服务器有两种方式，一种方式是在面向PC的接口下直接配置DHCP服务器IP地址。另外一种方式是在面向PC的接口下调用全局定义的DHCP服务器组。

第一种配置方法。直接在R1的E 0/0/1接口下开启DHCP中继功能，并直接指定DHCP服务器IP地址为100.1.1.1。

[R1]dhcp enable

[R1]interface Ethernet1/0/1

[R1-Ethernet1/0/1]dhcp select relay

[R1-Ethernet1/0/1]dhcp relay server-ip 100.1.1.1

第二种配置方法。在R1上创建DHCP服务器组，指定组名为dhcp-group，并使用**dhcp-server**命令添加远端的DHCP服务器地址。接着在E 0/0/1接口下开启DHCP中继功能并配置所对应的DHCP服务器组。

[R1]dhcp server group dhcp-group

Info:It's successful to create a DHCP server group.

[R1-dhcp-server-group-dhcp-group]dhcp-server 100.1.1.1

[R1-dhcp-server-group-dhcp-group]interface Ethernet1/0/1

[R1-Ethernet1/0/1]dhcp select relay

[R1-Ethernet1/0/1]dhcp relay server-select dhcp-group

两种方式均能达到同样的配置要求。相比而言，在接口下直接指定DHCP服务器IP地址的方式较简单，但如果中继设备上有多个接口需要配置DHCP中继功能，则要在所有接口上重复同样的配置，产生的配置量较大。而如果使用服务器组的方式，则可以仅在全局定义一次，在每个接口重复调用，尤其是当有多个DHCP服务器或者服务器IP地址需要更改的时候尤为方便。

## 配置PC获取地址方式为DHCP

DHCP中继设备R1和DHCP服务器R3配置完成后，并且中间链路的连通性也正常的情况下，配置PC机使用DHCP获取IP地址。

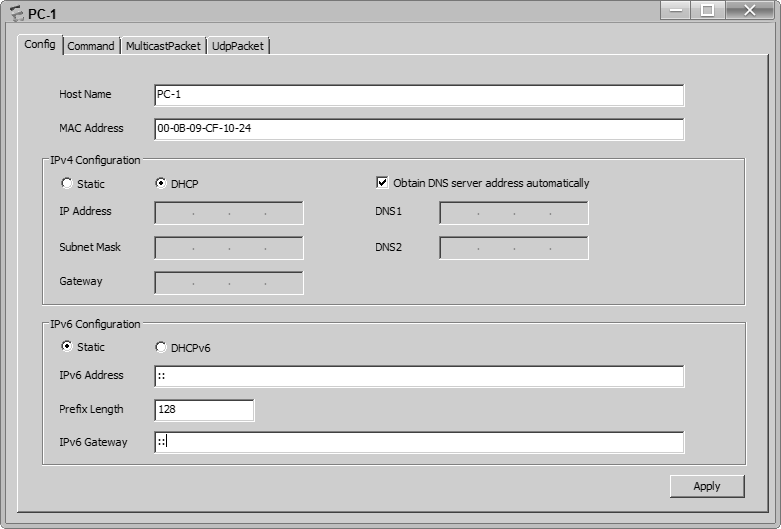


图12-11

PC-2也使用同样的方式配置使用DHCP。配置完成后，在PC-1上，使用命令**ipconfig**去查看地址获得情况。



在PC-2上，使用命令**ipconfig**去查看地址获得情况。



现在两台PC机都从总部DHCP服务器获得IP地址，测试两台PC的连通性。

PC>ping 10.1.1.253

Ping 10.1.1.253: 32 data bytes, Press Ctrl\_C to break

From 10.1.1.253: bytes=32 seq=1 ttl=128 time=32 ms

From 10.1.1.253: bytes=32 seq=2 ttl=128 time<1 ms

From 10.1.1.253: bytes=32 seq=3 ttl=128 time=16 ms

From 10.1.1.253: bytes=32 seq=4 ttl=128 time<1 ms

From 10.1.1.253: bytes=32 seq=5 ttl=128 time=15 ms

--- 10.1.1.253 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/12/32 ms

测试成功，PC机通过DHCP中继设备成功从DHCP服务器获得IP地址，并能使用该地址相互通信。

整个配置过程，仅在网关路由器R1上开启DHCP中继功能，在分部没有其它过多的DHCP配置。在网络设计和管理中灵活使用DHCP中继功能能够使网络运行更加高效和方便。

# 思考

在R1充当DHCP中继代理时，客户的DHCP请求包经DHCP中继R1到达DHCP服务器R3后，如果R3上定义有不同网段的多个IP地址池，R3是如何知道该从哪个地址池分配地址给PC-1和PC-2 ？

1. IPv6

13.1 IPv6基础配置

# 原理概述

以IPv4为核心技术的Internet获得巨大成功，促使IP技术得到广泛应用，丰富了人们的沟通和联系。然而，随着因特网的迅猛发展，IPv4技术的不足也日益明显，特别是地址空间的不足直接限制了IP技术应用的进一步发展。IPv6（Internet Protocol Version 6）是网络层协议的第二代标准协议，也被称为IPng（IP Next Generation）。它是IETF设计的一套规范。IPv6和IPv4之间最显著的区别就是IP地址长度从原来的32位变为128位，地址空间大的惊人。有夸张的说法是：可以做到地球上的每一粒沙子都有一个IPv6地址。IPv6以其简化的报文头格式、充足的地址空间、层次化的地址结构、灵活的扩展头、增强的邻居发现机制将在未来的市场竞争中充满活力。

128位的IPv6地址被分为8组，每组的16位用4个十六进制字符（0～9，A～F）来表示，组和组之间用冒号隔开。比如2031:0000:130F:0000:0000:09C0:876A:130B，为了书写方便，每组中的前导“0”都可以省略。地址中包含的连续两个或多个均为0的组，可以用双冒号“::”来代替，这样可以压缩IPv6地址书写时的长度。但是在一个IPv6地址中只能使用一次双冒号“::”，否则当计算机将压缩后的地址恢复成128位时，无法确定每段中0的个数。所以上述地址可以简写为：2031:0:130F::9C0:876A:130B。

一个IPv6地址可以分为两部分，比如2001:A304:6101:1:0000:E0:F726:4E58 /64，前64位是网络前缀，相当于IPv4地址中的网络ID，后64位相当于IPv4地址中的主机ID。

# 实验目的

* + 理解IPv6的地址格式
  + 掌握IPv6手工配置IP地址的方法
  + 掌握EUI-64方式配置IPv6地址的方法
  + 掌握IPv6静态路由和缺省路由的配置方法

# 实验内容

某公司在新建网络时部署IPv6，R1和R2分别为公司IT部门和人事部门路由器，两个部门通过交换机S1相连，IT部门的员工终端PC-1手工配置IPv6地址，并在R1与R2上配置IPv6静态路由，使两个部门的终端能够互相通信。

# 实验拓扑



图13-1 IPv6基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IPv6地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 2001:3:FD::/64 eui-64 | 64 | N/A |
| GE 0/0/1 | 2031:0:130F::1 | 64 | N/A |
| R2(AR1220) | GE 0/0/0 | 2002:3:DE::/64 eui-64 | 64 | N/A |
| GE 0/0/1 | 2031:0:130F::2 | 64 | N/A |
| PC-1 | Ethernet 0/0/1 | 2001:3:FD::2 | 64 | R1 |
| PC-2 | Ethernet 0/0/1 | 2002:3:DE::2 | 64 | R2 |

# 实验步骤

## 配置IPv6单播地址

根据实验编址表在PC上配置相应的IPv6地址。模拟器中的PC上已经默认开启了IPv6功能，即已经自动生成了链路本地地址。

在路由器系统视图模式上全局开启IPv6功能。

<R1>system-view

[R1]ipv6

在R1上的GE 0/0/0接口下使用命令**ipv6 enable**开启IPv6功能。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ipv6 enable

在R1的GE 0/0/0接口上配置自动生成的链路本地地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ipv6 address auto link-local

配置完成后，在R1上查看GE 0/0/0接口所配置的自动生成的链路本地地址。

[R1]display ipv6 interface

GigabitEthernet0/0/0 current state : UP

IPv6 protocol current state : UP

IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:19AB

No global unicast address configured

……

可以观察到当前GE 0/0/0接口下的链路本地地址为FE80::2E0:FCFF:FE03:19AB。

在PC-1上测试与R1链路本地地址间的连通性。

PC>ping FE80::2E0:FCFF:FE03:19AB

Ping fe80::2e0:fcff:fe03:19ab: 32 data bytes, Press Ctrl\_C to break

From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=1 hop limit=64 time=141 ms

From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=2 hop limit=64 time=47 ms

From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=3 hop limit=64 time=47 ms

From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=4 hop limit=64 time=31 ms

From fe80::2e0:fcff:fe03:19ab: bytes=32 seq=5 hop limit=64 time=16 ms

--- fe80::2e0:fcff:fe03:19ab ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 16/56/141 ms

可以观察到连通性正常，同理配置R2 的GE 0/0/0接口。

<R2>system-view

[R2]ipv6

[R2]interface GigabitEthernet 0/0/0

[R2-GigabitEthernet0/0/0]ipv6 enable

[R2-GigabitEthernet0/0/0]ipv6 address auto link-local

配置完成后，测试PC-2到R2的连通性。

PC>ping FE80::2E0:FCFF:FE03:3B30

Ping fe80::2e0:fcff:fe03:3b30: 32 data bytes, Press Ctrl\_C to break

From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=1 hop limit=64 time=109 ms

From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=2 hop limit=64 time=16 ms

From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=3 hop limit=64 time<1 ms

From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=4 hop limit=64 time<1 ms

From fe80::2e0:fcff:fe03:3b30: bytes=32 seq=5 hop limit=64 time=16 ms

--- fe80::2e0:fcff:fe03:3b30 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 0/28/109 ms

可以观察到，连通性正常。

在R1和R2的GE 0/0/1接口上手工静态配置全球单播地址。在配置IPv4地址时，新地址会替换老地址，而在配置IPv6地址时，新地址会被添加，老地址不受影响。使用**ipv6 address**命令可以为接口直接添加IPv6地址，2031:0:130F::1为需要配置的IPv6地址，64为此地址的前缀长度。

[R1]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]ipv6 enable

[R1-GigabitEthernet0/0/1]ipv6 address 2031:0:130F::1 64

[R2]interface GigabitEthernet 0/0/1

[R2-GigabitEthernet0/0/1]ipv6 enable

[R2-GigabitEthernet0/0/1]ipv6 address 2031:0:130F::2 64

配置完成后在R1和R2上查看所配置的全局地址。

[R1]display ipv6 interface GigabitEthernet 0/0/1

GigabitEthernet0/0/1 current state : UP

IPv6 protocol current state : UP

IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:19AC

Global unicast address(es):

2031:0:130F::1, subnet is 2031:0:130F::/64

Joined group address(es):

……

[R2]display ipv6 interface GigabitEthernet 0/0/1

GigabitEthernet0/0/1 current state : UP

IPv6 protocol current state : UP

IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:3B31

Global unicast address(es):

2031:0:130F::2, subnet is 2031:0:130F::/64

Joined group address(es):

……

可以观察到，IPv6全球单播地址的配置已经生效。

测试R1与R2的全球单播地址间的连通性。同时请注意区分全球单播地址和链路本地地址。

<R2>ping ipv6 2031:0:130F::1

PING 2031:0:130F::1 : 56 data bytes, press CTRL\_C to break

Reply from 2031:0:130F::1

bytes=56 Sequence=1 hop limit=64 time = 340 ms

Reply from 2031:0:130F::1

bytes=56 Sequence=2 hop limit=64 time = 70 ms

Reply from 2031:0:130F::1

bytes=56 Sequence=3 hop limit=64 time = 50 ms

Reply from 2031:0:130F::1

bytes=56 Sequence=4 hop limit=64 time = 50 ms

Reply from 2031:0:130F::1

bytes=56 Sequence=5 hop limit=64 time = 30 ms

--- 2031:0:130F::1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 30/108/340 ms

可以观察到，连通性正常。

## 用EUI－64方式配置IPv6地址

在R1的GE 0/0/0接口使用命令**ipv6 address 2001:3:FD::/64 eui-64**配置EUI-64地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ipv6 address 2001:3:FD:: 64 eui-64

配置完成后，查看配置结果。

[R1]display ipv6 interface brief

\*down: administratively down

(l): loopback

(s): spoofing

Interface Physical Protocol

GigabitEthernet0/0/0 up up

[IPv6 Address] 2001:3:FD:0:2E0:FCFF:FE03:19AB

GigabitEthernet0/0/1 up up

[IPv6 Address] 2031:0:130F::1

可以观察到，R1的GE 0/0/0接口此时已经生成了有效的EUI-64地址

继续在R2上的GE 0/0/0接口使用命令**ipv6 address 2002:3:DE::/64 eui-64**配置EUI-64地址。

[R2]interface GigabitEthernet 0/0/0

[R2-GigabitEthernet0/0/0]ipv6 address 2002:3:DE:: 64 eui-64

配置完成后，查看配置结果。

[R2]display ipv6 interface brief

\*down: administratively down

(l): loopback

(s): spoofing

Interface Physical Protocol

GigabitEthernet0/0/0 up up

[IPv6 Address] 2002:3:DE:0:2E0:FCFF:FE03:3B30

GigabitEthernet0/0/1 up up

[IPv6 Address] 2031:0:130F::2

地址生成后，在PC-1上配置R1的GE 0/0/0接口地址为网关地址，在PC-2上配置R2的GE 0/0/0接口地址为网关地址。

## 配置IPv6静态路由和缺省路由

在R1上使用ipv6 route-static命令配置IPv6静态路由，目的网段为PC-2所在的IPv6网段，下一跳为R2的GE0/0/1接口的IPv6全球单播地址。

[R1]ipv6 route-static 2002:3:DE:: 64 2031:0:130F::2

在R2上配置IPv6缺省路由，下一跳为R1的GE 0/0/1接口的IPv6全球单播地址。

[R2]ipv6 route-static :: 0 2031:0:130F::1

配置完成后在PC-1上测试与PC-2间的连通性。

PC>ping 2002:3:DE::2

Ping 2002:3:de::2: 32 data bytes, Press Ctrl\_C to break

From 2002:3:de::2: bytes=32 seq=1 hop limit=253 time=31 ms

From 2002:3:de::2: bytes=32 seq=2 hop limit=253 time=47 ms

From 2002:3:de::2: bytes=32 seq=3 hop limit=253 time=47 ms

From 2002:3:de::2: bytes=32 seq=4 hop limit=253 time=47 ms

From 2002:3:de::2: bytes=32 seq=5 hop limit=253 time=32 ms

--- 2002:3:de::2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/40/47 ms

可以观察到，连通性正常。

# 思考

如果路由器的某一接口下配置了多个IPv6地址，互相之间是否会产生影响？

13.2 RIPng基础配置

# 原理概述

RIPng（RIP next generation，下一代RIP协议）是IPv4中RIP version 2协议在IPv6网络上的扩展，多数RIPv2的原理都可以适用于RIPng。RIPng协议同样是基于距离矢量算法的路由协议，用跳数来衡量到达目的主机的距离（也称为度量值或开销）。在RIPng协议中，当跳数大于或等于16时，目的网络或主机就被定义为不可达。

为了能在IPv6网络中应用，RIPng对原有的RIP协议进行了修改：

UDP端口号：使用UDP的521端口（RIP使用520端口）发送和接收路由信息；

组播地址：使用FF02::9作为链路本地范围内的RIPng路由器组播地址；

目的地址和下一跳地址：使用128比特的IPv6地址，并使用前缀长度来代替子网掩码。

RIPng协议路由算法和RIPv2一样，同样支持水平分割、毒性逆转和触发更新功能，用来防止环路。缺省情况下，启用水平分割功能和触发更新，不启用毒性逆转功能。

# 实验目的

* + 理解RIPng的应用场景
  + 掌握RIPng的配置
  + 理解RIPng配置与RIP配置的区别

# 实验内容

某公司内部网络是个小型的IPv6网络，公司内IT部门通过路由器R2与公司出口网关R1相连，人事部门通过路由器R3与网关R1相连。由于公司网络是个简单的网络，本实验通过配置RIPng协议使各IPv6网络互通。

# 实验拓扑

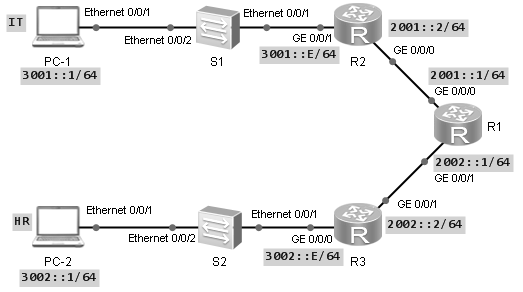


图13-2 RIPng基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IPv6地址 | 子网掩码 | 默认网关 |
| R1(AR2220) | GE 0/0/0 | 2001::1 | 64 | N/A |
| GE 0/0/1 | 2002::1 | 64 | N/A |
| R2(AR2220) | GE 0/0/1 | 3001::E | 64 | N/A |
| GE 0/0/0 | 2001::2 | 64 | N/A |
| R3(AR2220) | GE 0/0/0 | 3002::E | 64 | N/A |
| GE 0/0/1 | 2002::2 | 64 | N/A |
| PC-1 | Ethernet 0/0/1 | 3001::1 | 64 | 3001::E |
| PC-2 | Ethernet 0/0/1 | 3002::1 | 64 | 3002::E |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本IPv6地址配置，配置各接口的IPv6地址，并使用**ping**命令检测各直连链路的连通性。

[R1]ping ipv6 2002::2

PING 2002::2 : 56 data bytes, press CTRL\_C to break

Reply from 2002::2

bytes=56 Sequence=1 hop limit=64 time = 140 ms

Reply from 2002::2

bytes=56 Sequence=2 hop limit=64 time = 50 ms

Reply from 2002::2

bytes=56 Sequence=3 hop limit=64 time = 70 ms

Reply from 2002::2

bytes=56 Sequence=4 hop limit=64 time = 30 ms

Reply from 2002::2

bytes=56 Sequence=5 hop limit=64 time = 20 ms

--- 2002::2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 20/62/140 ms

其余直连网段的连通性测试省略。

## 配置RIPng

根据公司组网拓扑图，在R1，R2，R3上配置RIPng协议。

在R1、R2、R3上，创建RIPng路由进程1。

[R1]ripng 1

[R2]ripng 1

[R3]ripng 1

配置完成后，在路由器各相应接口下配置RIPng。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ripng 1 enable

[R1-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]ripng 1 enable

[R2]interface GigabitEthernet 0/0/0

[R2-GigabitEthernet0/0/0]ripng 1 enable

[R2-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1

[R2-GigabitEthernet0/0/1]ripng 1 enable

[R3]interface GigabitEthernet 0/0/0

[R3-GigabitEthernet0/0/0]ripng 1 enable

[R3-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1

[R3-GigabitEthernet0/0/1]ripng 1 enable

## 检查RIPng的路由表

配置完成后，查看每台路由器RIPng的路由表。

[R1]display ipv6 routing-table

Routing Table : Public

Destinations : 8 Routes : 10

……

Destination : 3001:: PrefixLength : 64

NextHop : FE80::5689:98FF:FE76:832B Preference : 100

Cost : 1 Protocol : RIPng

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/0 Flags : D

……

Destination : 3002:: PrefixLength : 64

NextHop : FE80::5689:98FF:FE76:8224 Preference : 100

Cost : 1 Protocol : RIPng

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/1 Flags : D

……

[R2]display ipv6 routing-table

Routing Table : Public

Destinations : 8 Routes : 12

……

Destination : 2002:: PrefixLength : 64

NextHop : FE80::5689:98FF:FE76:8223 Preference : 100

Cost : 1 Protocol : RIPng

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/0 Flags : D

……

Destination : 3002:: PrefixLength : 64

NextHop : FE80::5689:98FF:FE76:8223 Preference : 100

Cost : 1 Protocol : RIPng

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/0 Flags : D

……

[R3]display ipv6 routing-table

Routing Table : Public

Destinations : 8 Routes : 12

……

Destination : 2001:: PrefixLength : 64

NextHop : FE80::5689:98FF:FE76:830C Preference : 100

Cost : 1 Protocol : RIPng

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/1 Flags : D

……

Destination : 3001:: PrefixLength : 64

NextHop : FE80::5689:98FF:FE76:832C Preference : 100

Cost : 1 Protocol : RIPng

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/1 Flags : D

……

可以观察到各个路由器都获取到了相应的RIPng路由信息。

在路由器R1、R2、R3使用**display ripng 1 route**查看RIPng1的路由信息，同样也可以观察到每台路由器上获取到的RIPng路由信息，以R1为例。

[R1]display ripng 1 route

Route Flags: R - RIPng

A - Aging, G - Garbage-collect

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

Peer FE80::5689:98FF:FE76:832B on GigabitEthernet0/0/0

Dest 3001::/64,

via FE80::5689:98FF:FE76:832B, cost 1, tag 0, RA, 29 Sec

Peer FE80::5689:98FF:FE76:832C on GigabitEthernet0/0/1

Dest 3001::/64,

via FE80::5689:98FF:FE76:832C, cost 1, tag 0, RA, 14 Sec

Peer FE80::5689:98FF:FE76:8223 on GigabitEthernet0/0/0

Dest 3002::/64,

via FE80::5689:98FF:FE76:8223, cost 1, tag 0, RA, 15 Sec

Peer FE80::5689:98FF:FE76:8224 on GigabitEthernet0/0/1

Dest 3002::/64,

via FE80::5689:98FF:FE76:8224, cost 1, tag 0, RA, 13 Sec

在PC-1上测试与PC-2间的连通性

PC>ping 3002::1

Ping 3002::1: 32 data bytes, Press Ctrl\_C to break

From 3002::1: bytes=32 seq=1 hop limit=64 time=172 ms

From 3002::1: bytes=32 seq=2 hop limit=64 time=46 ms

From 3002::1: bytes=32 seq=3 hop limit=64 time=93 ms

From 3002::1: bytes=32 seq=4 hop limit=64 time=16 ms

From 3002::1: bytes=32 seq=5 hop limit=64 time=16 ms

--- 3002::1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 16/68/172 ms

可以观察到通信正常，RIPng协议已经使全网互通。

# 思考

RIPng支持认证吗？为什么？

13.3 OSPFv3基础配置

# 原理概述

OSPF（Open Shortest Path First）是IETF组织开发的一个基于链路状态的内部网关协议，针对IPv4协议使用的是OSPFv2，针对IPv6协议使用OSPFv3，即OSPF Version 3的简称。OSPFv3在OSPFv2基础上进行了增强，是一种运行在IPv6网络之上的路由协议。

OSPFv2是基于IPv4子网运行的，同一链路上的所有节点同处于一个IPv4子网或网络内，邻居关系建立的前提之一是相连接口必须处于同一IPv4子网内，每一条路由的下一跳地址都是和路由器接口处于同一网段的IPv4地址。OSPFv3是基于链路运行的，同一链路上的两个节点不必具有相同的前缀也可以直接通信，这一点极大地改变了OSPF的行为，使它独立于网络协议，容易扩展适应各种协议。

OSPFv3的Router-ID， Area ID仍然保留类似IPv4地址长度的32位的格式。实际上这些字段既不是IPv4地址，也不是IPv6地址，而只是一个编号。

另外在OSPFv2中，对Broadcast和NBMA网络类型，邻居路由器是以IP地址作为标识的。而在OSPFv3中，邻居路由器总是以Router-ID作为标识的，所以DR和BDR也总是用其Router-ID来标识的。

OSPFv3不再直接提供验证功能，转而依赖IPv6所提供的IP AH(Authentication Header)和IP ESP（Encapsulating Security Payload）协议进行验证，以确保路由信息的可信性、完整性和机密性。

# 实验目的

* + 理解OSPFv3的应用场景
  + 掌握OSPFv3的配置
  + 理解OSPFv3配置与OSPFv2配置的区别
  + 理解OSPFv3基于链路运行的特点

# 实验内容

公司内部网络是个中型的IPv6网络，R1和R2是公司两台核心路由器，R3是IT部门路由器，与核心层路由器R1直连。R4为人事部门路由器，与核心层路由器R2直连。为了使公司内网所有部门网络能互相通信，需要在此网络中配置支持IPv6的动态路由协议。考虑到公司网络较大和网络的扩展，部署OSPFv3。核心层路由器之间为区域0，整个IT部门在区域1中，人事部在区域2中。

# 实验拓扑

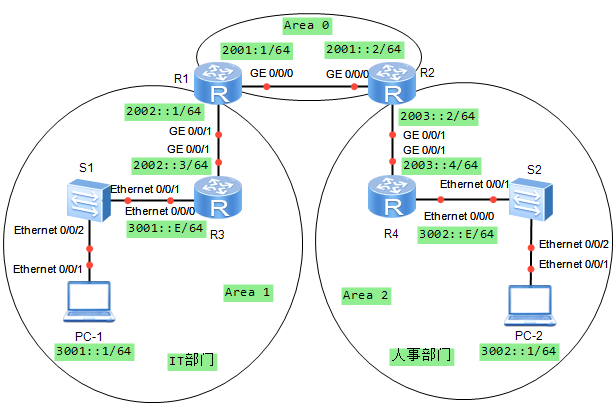


图13-3 OSPFv3基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IPv6地址 | 子网掩码 | 默认网关 |
| R1(AR1220) | GE 0/0/0 | 2001::1 | 64 | N/A |
| GE 0/0/1 | 2002::1 | 64 | N/A |
| R2(AR1220) | GE 0/0/0 | 2001::2 | 64 | N/A |
| GE 0/0/1 | 2003::2 | 64 | N/A |
| R3(AR1220) | GE 0/0/1 | 2002::3 | 64 | N/A |
| Ethernet 0/0/0 | 3001::E | 64 | N/A |
| R4(AR1220) | GE 0/0/1 | 2003::4 | 64 | N/A |
| Ethernet 0/0/0 | 3002::E | 64 | N/A |
| PC-1 | Ethernet 0/0/1 | 3001::1 | 64 | 3001::E |
| PC-2 | Ethernet 0/0/1 | 3002::1 | 64 | 3002::E |

# 实验步骤

## 基本配置

在各台设备上开启ipv6功能，根据实验编址表进行相应的基本IPv6地址配置，并使用**ping**命令检测各直连链路的连通性。

[R1]ping ipv6 2001::2

PING 2001::2 : 56 data bytes, press CTRL\_C to break

Reply from 2001::2

bytes=56 Sequence=1 hop limit=64 time = 270 ms

Reply from 2001::2

bytes=56 Sequence=2 hop limit=64 time = 70 ms

Reply from 2001::2

bytes=56 Sequence=3 hop limit=64 time = 50 ms

Reply from 2001::2

bytes=56 Sequence=4 hop limit=64 time = 40 ms

Reply from 2001::2

bytes=56 Sequence=5 hop limit=64 time = 40 ms

--- 2001::2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 40/94/270 ms

其余直连网段的连通性测试省略。

## 搭建OSPFv3网络

在各路由器上创建OSPFv3进程1，并在OSPFv3进程中配置每台路由器的OSPF Router-ID。R1，R2，R3，R4的Router-ID分别为1.1.1.1，2.2.2.2，3.3.3.3，4.4.4.4。

[R1]ospfv3 1

[R1-ospfv3-1]router-id 1.1.1.1

[R2]ospfv3 1

[R2-ospfv3-1]router-id 2.2.2.2

[R3]ospfv3 1

[R3-ospfv3-1]router-id 3.3.3.3

[R4]ospfv3 1

[R4-ospfv3-1]router-id 4.4.4.4

如果不配置Router-ID，OSPFv3的邻居就无法建立。因为在OSPFv3中，路由器是以Router-ID作为标识的，而不是用接口地址来标识。

在接口下配置OSPFv3，区域按照拓扑图划分的区域进行配置，IT部属于到区域1，HR部属于到区域2，区域1和区域2通过骨干区域0相连。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ospfv3 1 area 0

[R1]interface GigabitEthernet 0/0/1

[R1-GigabitEthernet0/0/1]ospfv3 1 area 1

[R2]interface GigabitEthernet 0/0/0

[R2-GigabitEthernet0/0/0]ospfv3 1 area 0

[R2]interface GigabitEthernet 0/0/1

[R2-GigabitEthernet0/0/1]ospfv3 1 area 2

[R3]interface GigabitEthernet 0/0/1

[R3-GigabitEthernet0/0/1]ospfv3 1 area 1

[R3]interface Ethernet 0/0/0

[R3-Ethernet0/0/0] ospfv3 1 area 1

[R4]interface GigabitEthernet 0/0/1

[R4-GigabitEthernet0/0/1]ospfv3 1 area 2

[R4]interface Ethernet 0/0/0

[R4-Ethernet0/0/0] ospfv3 1 area 2

配置完成后，查看每台路由器上的OSPFv3邻居状态。

[R1]display ospfv3 peer

OSPFv3 Process (1)

OSPFv3 Area (0.0.0.0)

Neighbor ID Pri State Dead Time Interface Instance ID

2.2.2.2 1 Full/DR 00:00:37 GE0/0/0 0

OSPFv3 Area (0.0.0.1)

Neighbor ID Pri State Dead Time Interface Instance ID

3.3.3.3 1 Full/DR 00:00:36 GE0/0/1 0

[R2]display ospfv3 peer

OSPFv3 Process (1)

OSPFv3 Area (0.0.0.0)

Neighbor ID Pri State Dead Time Interface Instance ID

1.1.1.1 1 Full/Backup 00:00:32 GE0/0/0 0

OSPFv3 Area (0.0.0.2)

Neighbor ID Pri State Dead Time Interface Instance ID

4.4.4.4 1 Full/DR 00:00:34 GE0/0/1 0

[R3]display ospfv3 peer

OSPFv3 Process (1)

OSPFv3 Area (0.0.0.1)

Neighbor ID Pri State Dead Time Interface Instance ID

1.1.1.1 1 Full/Backup 00:00:34 GE0/0/1 0

[R4]display ospfv3 peer

OSPFv3 Process (1)

OSPFv3 Area (0.0.0.2)

Neighbor ID Pri State Dead Time Interface Instance ID

2.2.2.2 1 Full/Backup 00:00:32 GE0/0/1 0

可以观察到，所有路由器都成功建立起了OSPFv3邻居。

查看每台路由器的IPv6路由表。

[R1]display ipv6 routing-table

Routing Table : Public

Destinations : 9 Routes : 9

……

Destination : 3001:: PrefixLength : 64

NextHop : FE80::2E0:FCFF:FE03:1B68 Preference : 10

Cost : 2 Protocol : OSPFv3

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/1 Flags : D

Destination : 3002:: PrefixLength : 64

NextHop : FE80::2E0:FCFF:FE03:56F0 Preference : 10

Cost : 3 Protocol : OSPFv3

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/0 Flags : D

……

[R2]display ipv6 routing-table

Routing Table : Public

Destinations : 9 Routes : 9

……

Destination : 3001:: PrefixLength : 64

NextHop : FE80::2E0:FCFF:FE03:F661 Preference : 10

Cost : 3 Protocol : OSPFv3

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/0 Flags : D

Destination : 3002:: PrefixLength : 64

NextHop : FE80::2E0:FCFF:FE03:2906 Preference : 10

Cost : 2 Protocol : OSPFv3

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/1 Flags : D

……

[R3]display ipv6 routing-table

Routing Table : Public

Destinations : 9 Routes : 9

……

Destination : 3002:: PrefixLength : 64

NextHop : FE80::2E0:FCFF:FE03:F662 Preference : 10

Cost : 4 Protocol : OSPFv3

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/1 Flags : D

……

[R4]display ipv6 routing-table

Routing Table : Public

Destinations : 9 Routes : 9

……

Destination : 3001:: PrefixLength : 64

NextHop : FE80::2E0:FCFF:FE03:56F1 Preference : 10

Cost : 4 Protocol : OSPFv3

RelayNextHop : :: TunnelID : 0x0

Interface : GigabitEthernet0/0/1 Flags : D

……

可以观察到，各个路由器之间相互接收到了添加进OSPFv3进程接口所在网段的路由条目。

在PC-1上测试与PC-2间的连通性。

PC>ping 3002::1

Ping 3002::1: 32 data bytes, Press Ctrl\_C to break

From 3002::1: bytes=32 seq=1 hop limit=64 time=47 ms

From 3002::1: bytes=32 seq=2 hop limit=64 time=32 ms

From 3002::1: bytes=32 seq=3 hop limit=64 time=31 ms

From 3002::1: bytes=32 seq=4 hop limit=64 time=31 ms

From 3002::1: bytes=32 seq=5 hop limit=64 time=31 ms

--- 3002::1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 31/34/47 ms

此时主机PC-1和PC-2可以通过OSPFv3路由协议进行通讯。

## 验证OSPFv3建立邻居的特性

在R1的GE 0/0/0接口下删除之前配置的IPv6地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]undo ipv6 address

配置完成后，在R1上使用**display ospfv3 peer**命令查看邻居关系。

[R1]display ospfv3 peer

OSPFv3 Process (1)

OSPFv3 Area (0.0.0.1)

Neighbor ID Pri State Dead Time Interface Instance ID

3.3.3.3 1 Full/DR 00:00:36 GE0/0/1 0

可以观察到，此时R1上已经没有与R2间的邻居关系。

在R1的GE 0/0/0接口上配置与R2直连接口所在IPv6网段不同前缀的IPv6地址2009::1/64。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]ipv6 address 2009::1

配置完成后，在R1上使用**display ospfv3 peer**命令查看邻居关系。

[R1]display ospfv3 peer

OSPFv3 Process (1)

OSPFv3 Area (0.0.0.0)

Neighbor ID Pri State Dead Time Interface Instance ID

2.2.2.2 1 Full/DR 00:00:31 GE0/0/0 0

OSPFv3 Area (0.0.0.1)

Neighbor ID Pri State Dead Time Interface Instance ID

3.3.3.3 1 Full/DR 00:00:34 GE0/0/1 0

可以观察到R1与R2仍然能建立邻居关系。这是因为OSPFv3的邻居关系建立是通过Link-Local地址来实现的，即链路上两端的IPv6地址即使拥有不同的前缀也可以建立邻居关系，而OSPFv2在同一链路上必须要使用相同网段的IPv4地址才能建立邻居关系。

在R1上查看GE 0/0/0接口上的Link-Local地址。

[R1]display ipv6 interface GigabitEthernet 0/0/0

GigabitEthernet0/0/0 current state : UP

IPv6 protocol current state : UP

IPv6 is enabled, link-local address is FE80::2E0:FCFF:FE03:330F

Global unicast address(es):

……

当通过改变接口的IPv6地址时，Link-Local地址是不会改变的，所以R1与R2的OSPFv3邻居关系仍然可以建立起来。

# 思考

在本实验中，OSPFv3协议修改了全球单播地址，邻居关系未受影响，请问各网络间的通信是否正常？从此处能够得到什么启示？

1. 其他特性

14.1 实现eNSP与真实PC桥接

# 原理概述

eNSP(Enterprise Network Simulation Platform)是一款由华为提供的免费的、可扩展的、图形化网络仿真工具平台，主要对企业网路由器、交换机进行软件仿真，完美呈现真实设备实景，支持大型网络模拟，让用户有机会在没有真实设备的情况下也能够实验测试和学习网络技术。eNSP不仅支持单机部署，同时还支持Server端分布式部署在多台服务器上，分布式部署环境下能够支持更多设备组成复杂的大型网络。也**可与真实设备对接，**通过虚拟设备接口与真实网卡的绑定，实现虚拟设备与真实设备的对接，进而实现虚拟网络与真实网络的互连互通。

# 实验内容

本实验将介绍如何使用eNSP中云设备实现模拟器与真实电脑的桥接，实现与真实电脑或其他设备间的正常通信。

# 实验目的

* + 掌握在eNSP中使用云设备与虚拟PC连接的方法
  + 掌握在eNSP中使用云设备与真实电脑上的物理网卡桥接的方法

# 实验步骤

## 配置云设备

双击eNSP图标，打开模拟器，选择左侧栏目中“云设备”图标，拖进拓扑图中。

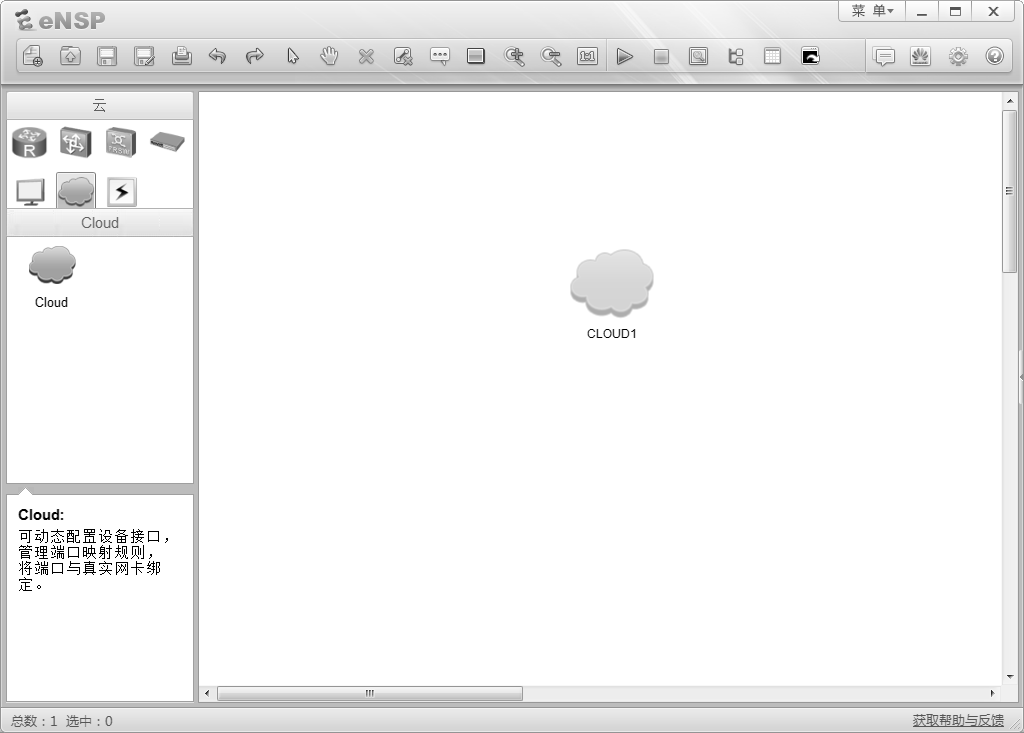


图14-1

将鼠标移到云设备图标上，点击右键选择“设置"。

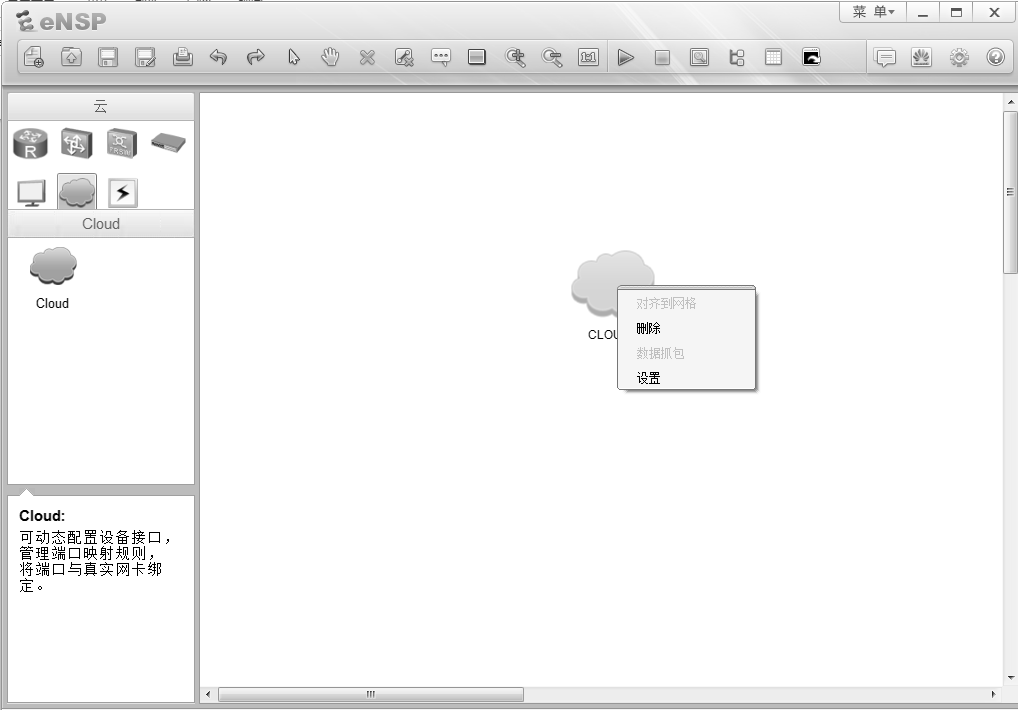


图14-2

进入到云设置界面后，创建一个端口，绑定信息栏选择“UDP"，端口类型选择“Ethernet"，点击“增加”按钮，关于端口的信息将会出现在端口信息表中，序列号为1。

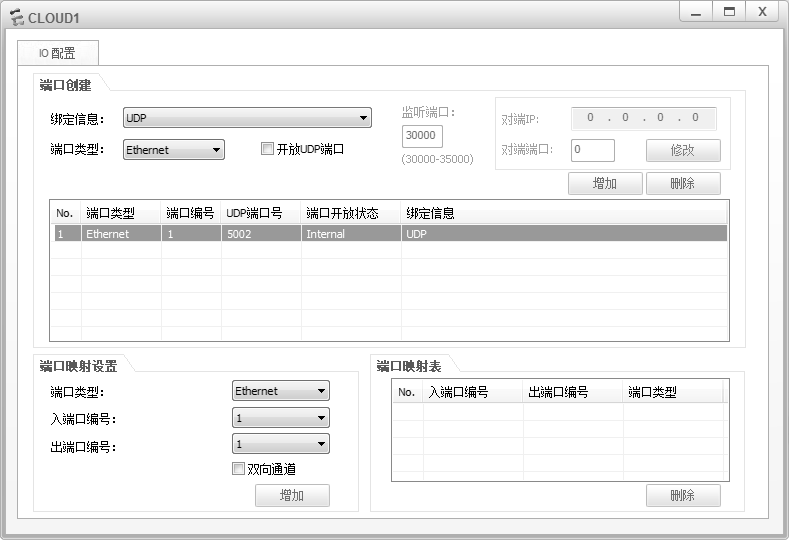


图14-3

创建另外一个端口，在绑定信息栏中下拉选择真实PC中任意一个网卡地址，这里选择了PC中的无线网卡，IP地址为：192.168.6.33。端口类型选择“Ethernet"



图14-4

点击“增加”按钮，端口表中会创建另外一个端口信息。

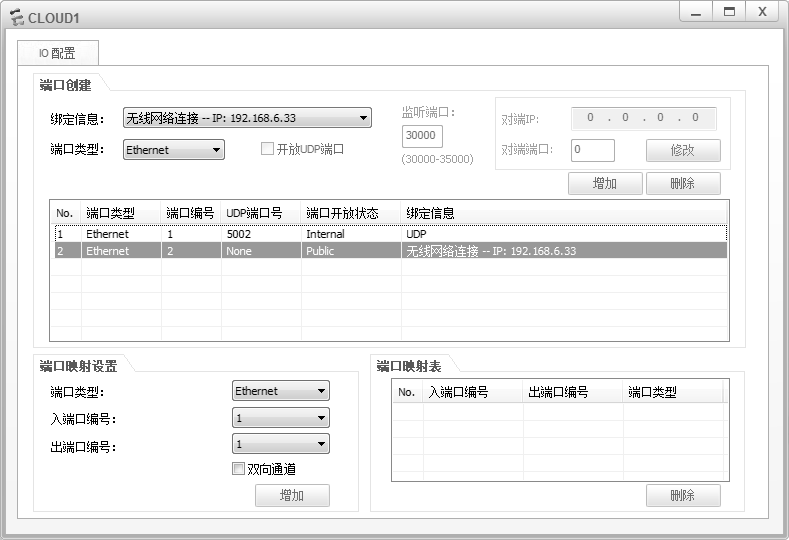


图14-5

创建端口的映射关系。在下方端口映射设置中，将“入端口编号”选择“2”，也就是刚才所创建的对应真实PC上无线网卡的端口，将“出端口编号”选择为“1”，勾选“双向通道”，点击“增加”按钮，即可添加到端口映射表。



图14-6

端口的映射关系表说明了模拟器的设备如何与真实的PC之间的通信连接，指明了数据从哪个接口发送，从哪个接口接收。

## 为模拟器添加设备，实现与真实设备的桥接

在eNSP模拟器中添加一台虚拟PC，使用线缆连接到云设备的E 0/0/1接口，并启动设备。

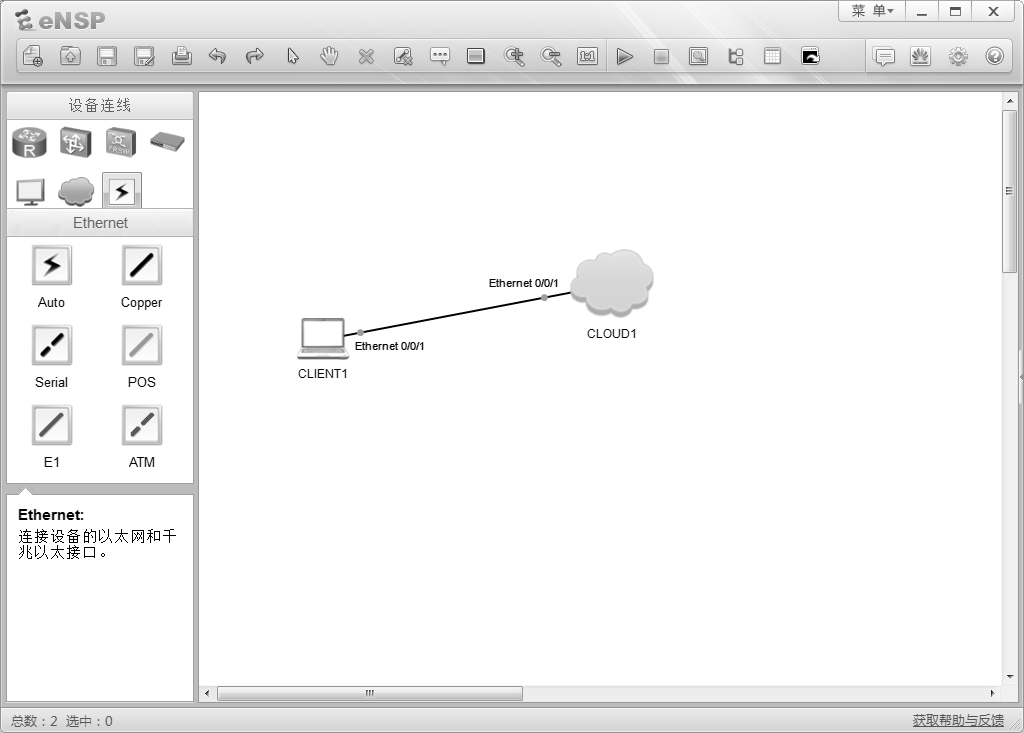


图14-7

为eNSP模拟器中的PC设置IP地址为192.168.6.1，子网掩码：255.255.255.0（该地址要配置成与真实电脑的IP地址为同一网段）。

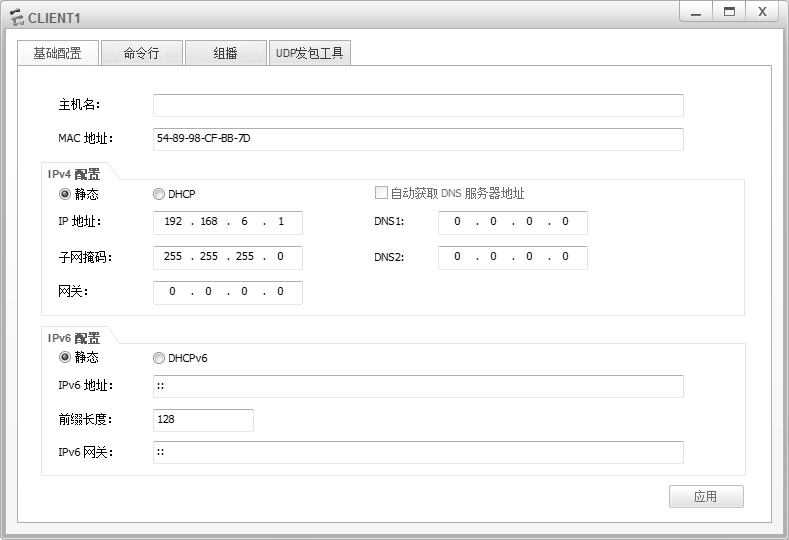


图14-8

## 验证模拟器中的PC与真实电脑之间的连通性

在真实电脑上使用**ping**命令，验证与模拟器中虚拟PC间的连通性，观察到可以PING通。



图14-9

在模拟器中使用**ping**命令，验证与真实电脑间的连通性，观察到可以PING通。

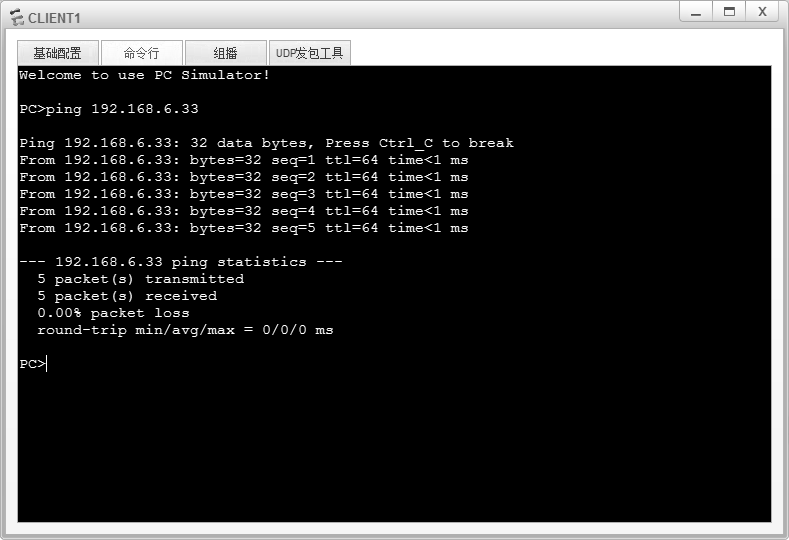


图14-10

至此，成功实现了真实电脑通过eNSP中云设备的桥接功能实现了与虚拟PC间的通信。

14.2 SNMP基础配置

# 原理概述

随着网络规模的日益发展，现有的网络中，设备数量日益庞大。当这些设备发生故障时，由于设备无法主动上报故障，导致网络管理员无法及时感知、定位和排除故障，从而导致网络的维护效率降低，维护工作量大大增加。

为了解决这个问题，设备制造商已经在一些设备中提供了网络管理的功能，这样网管就可以远程查看设备的状态，同样设备能够在特定类型的事件发生时向网络管理工作站发出警告。SNMP(Simple Network Management Protocol，[简单网络管理协议](http://baike.baidu.com/view/875800.htm" \t "_blank))就是规定网管站和设备之间如何传递管理信息的应用层协议。SNMP定义了网管管理设备的几种操作，以及设备故障时能向网管主动发送告警。网络管理使用SNMP协议时存在网络管理站，代理进程（Agent），和被管理设备三个角色。

网络管理站（NMS）：向被管理设备发送各种查询报文，以及接收被管理设备发送的告警；

代理进程（Agent）：是被管理设备上的一个代理进程。用于维护被管理设备的信息数据并响应来自NMS的请求，把管理数据汇报给发送请求的NMS。Agent的作用为接收、解析来自网管站的查询报文；根据报文类型对管理变量进行Read或Write操作，并生成响应报文，返回给网管站；根据各协议模块对告警触发条件的定义，在达到触发条件后，如进入、退出系统视图或设备重新启动等，相应的模块通过Agent主动向网管站发送告警，报告所发生的事件；

被管理设备：接受网管的管理，产生和主动上报告警。

# 实验内容

本实验模拟真实网络场景，在网络当中分别部署了两台管理站设备（NMS）和一台代理站设备（Agent）。分别使用两台PC来模拟（NMS）；一台路由器来模拟代理站（Agent）。本实验将介绍如何配置Agent设备、版本，配置管理站及用户权限，了解SNMP协议的作用及管理方法。被管理的设备可以是路由器、服务器、交换机、主机等设备，一般与Agent部署在同一网络。

# 实验目的

* + 理解SNMP 应用场景
  + 掌握配置SNMP Agent的方法
  + 掌握配置SNMP 版本的方法
  + 掌握配置管理站，用户权限的方法

# 实验拓扑

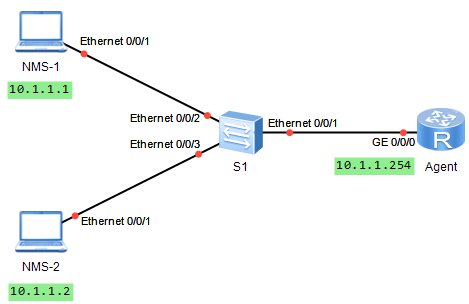


图14-11 SNMP基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| NMS-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | N/A |
| NMS-2 | Ethernet 0/0/1 | 10.1.1.2 | 255.255.255.0 | N/A |
| Agent(AR2220) | GE 0/0/0 | 10.1.1.254 | 255.255.255.0 | N/A |

# 实验步骤

## 基本配置

将两台PC模拟成为管理站（NMS），分别为NMS-1、NMS-2，根据实验编址表配置设备名称及IP地址，并验证连通性。

[Agent]ping 10.1.1.1

PING 10.1.1.1: 56 data bytes, press CTRL\_C to break

Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=128 time=550 ms

Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=128 time=200 ms

Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=128 time=150 ms

Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=128 time=90 ms

Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=128 time=120 ms

--- 10.1.1.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 90/222/550 ms

其余直连网段的连通性测试省略。

## 开启Agent服务

代理站设备（Agent）负责维护被管理设备的信息数据，并响应来自管理站的请求，把管理数据汇报给发送请求的管理站。

将路由器模拟为代理站设备，需要为路由器启动SNMP Agent服务。

[Agent]snmp-agent

配置完成后使用命令**display snmp-agent sys-info**查看SNMP信息。

[Agent]display snmp-agent sys-info

The contact person for this managed node:

R&D Shenzhen, Huawei Technologies Co., Ltd.

The physical location of this node:

Shenzhen China

SNMP version running in the system:

SNMPv1 SNMPv2c SNMPv3

显示当前默认情况下所运行的SNMP版本为v1，v2c，v3。

## 配置SNMP版本

SNMP一共有三个版本，分别为v1、v2c、v3，开启SNMP服务后默认同时兼容三个版本。当网络规模较小且安全性较高时，在规划时可配置SNMPv1版本与网管进行通信。当网络规模较大，安全性较高时，但运行的业务较为繁忙，在规划时配置使用SNMPv2c版本与网管进行通信。当网络规模较大且安全性较低时，在规划时配置设备使用SNMPv3版本与网管进行通信，并配置认证和加密功能保证安全性。

可以根据实际网络需求，使用命令**snmp-agent sys-info**配置SNMP版本，本例中使用SNMPv3版本。

[Agent]snmp-agent sys-info version v3

配置完后查看Agent信息。

[Agent]display snmp-agent sys-info version

SNMP version running in the system:

SNMPv3

显示当前运行的SNMP版本为v3。

由于SNMPv3版本适用于大型网络规模，且可以配置认证加密，在实际工作环境下通常使用该版本。

## 配置NMS管理权限

如果网络中不止一个管理站用户，可以根据业务需要，为不同管理站用户设置不同的访问权限。本实验中有两个管理站用户，现仅允许NMS-2可以管理设备。

配置基本ACL，限制NMS-2可以管理设备，NMS-1不允许管理设备。

[Agent]acl 2000

[Agent-acl-basic-2000]rule 5 permit source 10.1.1.2 0.0.0.255

[Agent-acl-basic-2000]rule 10 deny source 10.1.1.1 0.0.0.255

配置用户组为group，用户名为user，指定使用访问控制列表2000。

[Agent]snmp-agent usm-user v3 user group acl 2000

配置完成后，查看SNMPv3的用户信息。

[Agent]display snmp-agent usm-user

User name: user

Engine ID: 800007DB03000000000000

Group name: group

Authentication mode: No authentication mode, Privacy mode: No privacy mode

Storage type: nonVolatile

User status: active

Acl: 2000

Total number is 1

可以观察到，配置已经生效。

## 配置向SNMP Agent输出Trap信息

网络管理员需要查看被管理者产生的Trap信息，以便监控设备运行情况及定位故障信息。

配置代理站发送Trap消息。用于接收该Trap消息的网管名为adnminNMS2，目标地址为10.1.1.2，且指定接收该消息使用UDP端口为9991。Trap消息的发送参数信息列表名称为trapNMS2。

[Agent]snmp-agent target-host trap-hostname adminNMS2 address 10.1.1.2 udp-port 9991 trap-paramsname trapNMS2

开启设备的告警开关。只有将该开关打开以后，Agent才会向网管站发送告警消息。

[Agent]snmp-agent trap enable

Info: All switches of SNMP trap/notification will be open. Continue? [Y/N]:y

设置告警消息的队列长度为200（默认值为100）。如果某个时间段trap报文消息很多，为防止丢包，可以设置增加消息队列长度以便减少丢包的情况发生。

[Agent]snmp-agent trap queue-size 200

设置报文消息的保存时间为240秒（默认值为120）。该值是Trap报文消息的生存时间，如果超过该时间报文将会被丢弃，不再发送，也不再保存。

[Agent]snmp-agent trap life 240

为了便于维护，配置管理员的联系方式，电话为：400-822-9999，地址为中国深圳。

[Agent]snmp-agent sys-info contact call admin 400-822-9999

[Agent]snmp-agent sys-info location ShenZhen China

配置完后使用命令**display snmp-agent sys-info**查看到相关的系统信息。

[Agent]display snmp-agent sys-info

The contact person for this managed node:

call admin 400-822-9999

The physical location of this node:

ShenZhen China

SNMP version running in the system:

SNMPv3

查看SNMP Agent输出网管的信息。

[Agent]display snmp-agent target-host

Traphost list:

Target host name: adminNMS2

Traphost address: 10.1.1.2

Traphost portnumber: 9991

Target host parameter: trapNMS2

Total number is 1

Parameter list trap target host:

Total number is 0

配置完成后通过命令可以观察到Trap目标主机名为adnminNMS2，主机地址10.1.1.2，主机端口为9991，目标主机参数列表名为trapNMS2。

14.3 GRE 协议基础配置

# 原理概述

GRE（Generic Routing Encapsulation）通用路由封装协议，提供了将一种协议的报文封装在另一种协议报文中的机制，使报文能够在异种网络中（如IPv4网络）传输，而异种报文传输的通道称为Tunnel。

GRE协议也可以作为VPN的第三层隧道协议，为VPN数据提供透明传输通道。GRE是VPN的第三层隧道（Tunnel）协议。Tunnel是一个虚拟的点对点的连接，可以看成仅支持点对点连接的虚拟接口，这个接口提供了一条通路，使封装的数据报能够在这个通路上传输，并在一个Tunnel的两端分别对数据报进行封装及解封装。

# 实验目的

* + 理解GRE协议的使用场景
  + 掌握配置GRE隧道的方法
  + 掌握配置基于GRE接口的动态路由协议的方法

# 实验内容

本实验模拟企业网络场景，R1为企业总部的网关设备，并且内部有一台服务器，R3连接着企业分公司网关设备，R2为公网ISP设备。一般情况下，运营商只会维护自身的公网路由信息，而不会维护企业内部私网的路由信息，即运营商设备上的路由表中不会出现任何企业内部私网的路由条目。通过配置GRE实现公司总部和分部间私网路由信息的透传及数据通信。

# 实验拓扑

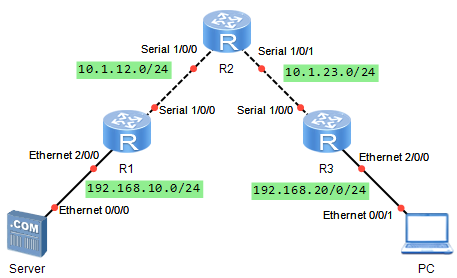


图14-12 GRE协议基础配置拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| Server | Ethernet 0/0/0 | 192.168.10.10 | 255.255.255.0 | 192.168.10.1 |
| R1(AR2220) | Ethernet 2/0/0 | 192.168.10.1 | 255.255.255.0 | N/A |
| Serial 1/0/0 | 10.1.12.1 | 255.255.255.0 | N/A |
| R2(AR2220) | Serial 1/0/0 | 10.1.12.2 | 255.255.255.0 | N/A |
| Serial 1/0/1 | 10.1.23.2 | 255.255.255.0 | N/A |
| R3(AR2220) | Serial 1/0/0 | 10.1.23.1 | 255.255.255.0 | N/A |
| Ethernet 2/0/0 | 192.168.20.1 | 255.255.255.0 | N/A |
| PC | Ethernet 0/0/1 | 192.168.20.20 | 255.255.255.0 | 192.168.20.1 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping -c 1 192.168.10.10

PING 192.168.10.10: 56 data bytes, press CTRL\_C to break

Reply from 192.168.10.10: bytes=56 Sequence=1 ttl=255 time=510 ms

--- 192.168.10.10 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 510/510/510 ms

其余直连网段的连通性测试省略。

在R1和R3上分别配置访问公网路由器R2的默认路由。

[R1]ip route-static 0.0.0.0 0.0.0.0 10.1.12.2

[R3]ip route-static 0.0.0.0 0.0.0.0 10.1.23.2

配置完成后在PC上测试与总部服务器间的连通性。

PC>ping 192.168.10.10

Ping 192.168.10.10: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

可以观察到，跨越了互联网的两个私网网段之间默认是无法直接通信的。此时可以通过GRE协议来实现跨越了互联网的两个私网网段之间的通信。

## 配置GRE Tunnel

在路由器R1和R3上配置GRE Tunnel，使用命令interface tunnel创建隧道接口，指定隧道模式为GRE。配置R1 Tunnel接口的源地址为其S 1/0/0接口IP地址，目的地址为R3的S 1/0/0接口IP地址；配置R3 Tunnel接口源地址为其S 1/0/0接口IP地址，目的地址为R1的S 1/0/0接口IP地址。还要使用命令**ip addres**s配置Tunnel接口的IP地址，注意要在同一网段。

[R1]interface tunnel 0/0/0

[R1-Tunnel0/0/0]tunnel-protocol gre

[R1-Tunnel0/0/0]source 10.1.12.1

[R1-Tunnel0/0/0]destination 10.1.23.1

[R1-Tunnel0/0/0]ip address 172.16.1.1 24

[R3]interface tunnel 0/0/0

[R3-Tunnel0/0/0]tunnel-protocol gre

[R3-Tunnel0/0/0]source 10.1.23.1

[R3-Tunnel0/0/0]destination 10.1.12.1

[R3-Tunnel0/0/0]ip address 172.16.1.2 24

配置完成后，在R1上测试本端隧道接口地址与目的端口隧道接口地址的连通性。

[R1]ping -a 172.16.1.1 172.16.1.2

PING 172.16.1.2: 56 data bytes, press CTRL\_C to break

Reply from 172.16.1.2: bytes=56 Sequence=1 ttl=255 time=60 ms

Reply from 172.16.1.2: bytes=56 Sequence=2 ttl=255 time=60 ms

Reply from 172.16.1.2: bytes=56 Sequence=3 ttl=255 time=50 ms

Reply from 172.16.1.2: bytes=56 Sequence=4 ttl=255 time=50 ms

Reply from 172.16.1.2: bytes=56 Sequence=5 ttl=255 time=60 ms

--- 172.16.1.2 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 50/56/60 ms

可以观察到，通信正常。

在R1和R3上分别执行命令**display interface tunnel**查看隧道接口状态。

[R1]display interface Tunnel 0/0/0

Tunnel0/0/0 current state : UP

Line protocol current state : UP

Last line protocol up time : 2013-05-29 15:36:53 UTC-08:00

Description:

Route Port,The Maximum Transmit Unit is 1500

Internet Address is 172.16.1.1/24

Encapsulation is TUNNEL, loopback not set

Tunnel source 10.1.12.1 (Serial1/0/0), destination 10.1.23.1

Tunnel protocol/transport GRE/IP, key disabled

keepalive disabled

……

[R3]display interface Tunnel 0/0/0

Tunnel0/0/0 current state : UP

Line protocol current state : UP

Last line protocol up time : 2013-05-29 16:06:09 UTC-08:00

Description:

Route Port,The Maximum Transmit Unit is 1500

Internet Address is 172.16.1.2/24

Encapsulation is TUNNEL, loopback not set

Tunnel source 10.1.23.1 (Serial1/0/0), destination 10.1.12.1

Tunnel protocol/transport GRE/IP, key disabled

keepalive disabled

……

可以观察到，当前隧道接口的物理层状态为正常启动状态，链路层协议状态为正常运行状态，隧道封装协议为GRE协议，Tunnel的IP地址，及所配置的隧道源和目的地址分别为R1和R3的S1/0/0接口地址。

在R1和R2上执行命令**display ip routing-table**查看路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 Static 60 0 RD 10.1.12.2 Serial1/0/0

10.1.12.0/24 Direct 0 0 D 10.1.12.1 Serial1/0/0

10.1.12.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.1.12.2/32 Direct 0 0 D 10.1.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.1 Tunnel0/0/0

172.16.1.1/32 Direct 0 0 D 127.0.0.1 Tunnel0/0/0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet2/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet2/0/0

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 10 Routes : 10

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 Static 60 0 RD 10.1.23.2 Serial1/0/0

10.1.23.0/24 Direct 0 0 D 10.1.23.1 Serial1/0/0

10.1.23.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.1.23.2/32 Direct 0 0 D 10.1.23.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.2 Tunnel0/0/0

172.16.1.2/32 Direct 0 0 D 127.0.0.1 Tunnel0/0/0

192.168.20.0/24 Direct 0 0 D 192.168.20.1 Ethernet2/0/0

192.168.20.1/32 Direct 0 0 D 127.0.0.1 Ethernet2/0/0

可以观察到，R1和R3的路由表中已经有所配置隧道接口的路由条目。即R1和R3间已经形成了类似点到点直连的逻辑链路，但没有互相接收到对方的私网路由信息。

## 配置基于GRE接口的动态路由协议

经过上面的步骤，R1和R3之间GRE隧道已经建立，测试分部PC与总部服务器间的连通性。

PC>ping 192.168.10.10

Ping 192.168.10.10: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

可以观察到仍然无法正常通信，原因是目前还没有配置基于GRE隧道接口的路由协议。GRE协议支持组播数据的传输，因此可以支持一些动态路由协议的运行，动态路由协议通过GRE协议形成的逻辑隧道在R1和R3间传递路由信息。

在R1和R3上配置RIPv2协议，通告相应的私网网段和Tunnel接口所在网络。

[R1]rip 1

[R1-rip-1]version 2

[R1-rip-1]network 192.168.10.0

[R1-rip-1]network 172.16.0.0

[R3]rip 1

[R3-rip-1]version 2

[R3-rip-1]network 192.168.20.0

[R3-rip-1]network 172.16.0.0

分别在R1与R3上查看RIP邻居。

[R1]display rip 1 neighbor

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

IP Address Interface Type Last-Heard-Time

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

172.16.1.2 Tunnel0/0/0 RIP 0:0:7

Number of RIP routes : 1

[R3]display rip 1 neighbor

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

IP Address Interface Type Last-Heard-Time

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

172.16.1.1 Tunnel0/0/0 RIP 0:0:12

Number of RIP routes : 1

可以观察到，此时双方都已经通过隧道接口建立了RIP邻居关系。

查看路由器R1和R3路由表。

[R1]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 11 Routes : 11

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 Static 60 0 RD 10.1.12.2 Serial1/0/0

10.1.12.0/24 Direct 0 0 D 10.1.12.1 Serial1/0/0

10.1.12.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.1.12.2/32 Direct 0 0 D 10.1.12.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.1 Tunnel0/0/0

172.16.1.1/32 Direct 0 0 D 127.0.0.1 Tunnel0/0/0

192.168.10.0/24 Direct 0 0 D 192.168.10.1 Ethernet2/0/0

192.168.10.1/32 Direct 0 0 D 127.0.0.1 Ethernet2/0/0

192.168.20.0/24 RIP 100 1 D 172.16.1.2 Tunnel0/0/0

[R3]display ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 11 Routes : 11

Destination/Mask Proto Pre Cost Flags NextHop Interface

0.0.0.0/0 Static 60 0 RD 10.1.23.2 Serial1/0/0

10.1.23.0/24 Direct 0 0 D 10.1.23.1 Serial1/0/0

10.1.23.1/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.1.23.2/32 Direct 0 0 D 10.1.23.2 Serial1/0/0

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

172.16.1.0/24 Direct 0 0 D 172.16.1.2 Tunnel0/0/0

172.16.1.2/32 Direct 0 0 D 127.0.0.1 Tunnel0/0/0

192.168.10.0/24 RIP 100 1 D 172.16.1.1 Tunnel0/0/0

192.168.20.0/24 Direct 0 0 D 192.168.20.1 Ethernet2/0/0

192.168.20.1/32 Direct 0 0 D 127.0.0.1 Ethernet2/0/0

可以观察到，双方都能接收到各自内部私有网络发送过来的路由更新。

在分公司PC上测试与总公司Server间的连通性。

PC>ping 192.168.10.10

Ping 192.168.10.10: 32 data bytes, Press Ctrl\_C to break

From 192.168.10.10: bytes=32 seq=1 ttl=126 time=47 ms

From 192.168.10.10: bytes=32 seq=2 ttl=126 time=46 ms

From 192.168.10.10: bytes=32 seq=3 ttl=126 time=47 ms

From 192.168.10.10: bytes=32 seq=4 ttl=126 time=62 ms

From 192.168.10.10: bytes=32 seq=5 ttl=126 time=47 ms

--- 192.168.10.10 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 46/49/62 ms

可以观察到，通信正常。

查看路由器R2的路由表。

[R2]dis ip routing-table

Route Flags: R - relay, D - download to fib

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

Routing Tables: Public

Destinations : 8 Routes : 8

Destination/Mask Proto Pre Cost Flags NextHop Interface

10.1.12.0/24 Direct 0 0 D 10.1.12.2 Serial1/0/0

10.1.12.1/32 Direct 0 0 D 10.1.12.1 Serial1/0/0

10.1.12.2/32 Direct 0 0 D 127.0.0.1 Serial1/0/0

10.1.23.0/24 Direct 0 0 D 10.1.23.2 Serial1/0/1

10.1.23.1/32 Direct 0 0 D 10.1.23.1 Serial1/0/1

10.1.23.2/32 Direct 0 0 D 127.0.0.1 Serial1/0/1

127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0

127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0

可以观察到R2上没有任何R1和R3的各自私网网段的路由信息。说明通过GRE协议建立起来的隧道，能够跨公网传递各个内部私有网络的路由信息，实现了两个私有网络间的跨公网通信。

# 思考

GRE是一种三层的隧道协议，可以形成逻辑的点到点直连隧道，支持组播数据的传输，但是它的安全性能较差，不能实现隧道中所传输数据的加密，那GRE应采用何种方式实现组播数据跨互联网的加密传输？

14.4 配置NAT

# 原理概述

早在上世纪90年代初，有关RFC文档就提出IP地址耗尽的可能性。IPv6技术的提出虽然可以从根本上解决地址短缺的问题，但是也无法立刻替换现有成熟且广泛应用的IPv4网络。既然不能立即过渡到IPv6网络，那么必须使用一些技术手段来延长IPv4的寿命，其中广泛使用的技术之一就是网络地址转换（NAT， Network Address Translation）。

NAT是将IP数据报文报头中的IP地址转换为另一个IP地址的过程，主要用于实现内部网络（私有IP地址）访问外部网络（公有IP地址）的功能。NAT转换设备（实现NAT功能的网络设备）都维护着地址转换表，所有经过NAT转换设备并且需要进行地址转换的报文，都会通过该表做相应转换。

NAT转换设备处于内部网络和外部网络的连接处，常见的NAT转换设备有路由器、防火墙等。

# 实验目的

* + 理解NAT的应用场景
  + 掌握静态NAT的配置
  + 掌握NAT Outbound的配置
  + 掌握NAT Easy-IP的配置
  + 掌握NAT Server的配置

# 实验内容

本实验模拟企业网络场景，R1是公司的出口网关路由器，公司内员工和服务器都通过交换机S1或S2连接到R1上，R2模拟外网设备与R1直连。由于公司内网都使用的是私网IP地址，为了实现公司内部分员工可以访问外网，服务器可以供外网用户访问，网络管理员根据公司需求，需要在路由器R1上配置NAT，使用静态NAT和NAT Outbound技术使部分员工可以访问外网，使用NAT Server技术使服务器可以供外网用户访问。

# 实验拓扑

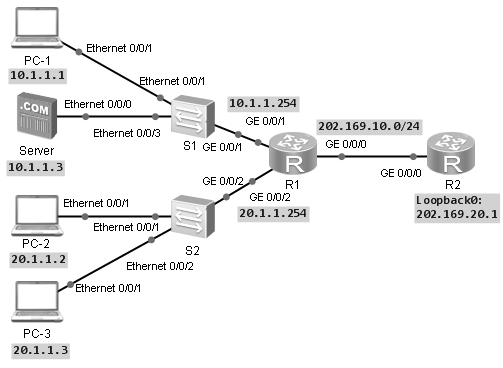


图14-13 配置NAT拓扑图

# 实验编址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 设备 | 接口 | IP地址 | 子网掩码 | 默认网关 |
| R1(AR2200) | GE 0/0/0 | 202.169.10.1 | 255.255.255.0 | N/A |
| GE 0/0/1 | 10.1.1.254 | 255.255.255.0 | N/A |
| GE 0/0/2 | 20.1.1.254 | 255.255.255.0 | N/A |
| R2(AR2200) | GE 0/0/0 | 202.169.10.2 | 255.255.255.0 | N/A |
| Loopback 0 | 202.169.20.1 | 255.255.255.0 | N/A |
| PC-1 | Ethernet 0/0/1 | 10.1.1.1 | 255.255.255.0 | 10.1.1.254 |
| PC-2 | Ethernet 0/0/1 | 20.1.1.2 | 255.255.255.0 | 20.1.1.254 |
| PC-3 | Ethernet 0/0/1 | 20.1.1.3 | 255.255.255.0 | 20.1.1.254 |
| Server | Ethernet 0/0/0 | 10.1.1.3 | 255.255.255.0 | 10.1.1.254 |

# 实验步骤

## 基本配置

根据实验编址表进行相应的基本配置，并使用**ping**命令检测各直连链路的连通性。

<R1>ping -c 1 202.169.10.2

PING 202.169.10.2: 56 data bytes, press CTRL\_C to break

Reply from 202.169.10.2: bytes=56 Sequence=1 ttl=255 time=140 ms

--- 202.169.10.2 ping statistics ---

1 packet(s) transmitted

1 packet(s) received

0.00% packet loss

round-trip min/avg/max = 140/140/140 ms

其余直连网段的连通性测试省略。

## 配置静态NAT

公司在网关路由器R1上配置访问外网的默认路由。

[R1]ip route-static 0.0.0.0 0.0.0.0 202.169.10.2

但由于内网使用的都是私有IP地址，员工无法直接访问公网。现需要在网关路由器R1上配置NAT地址转换，将私网地址转换为公网地址。

PC-1为公司客户经理使用的终端，不仅需要自身能访问外网，还需要外网用户也能够直接访问他，因此网络管理员分配了一个公网IP地址202.169.10.5给PC-1做静态NAT地址转换。在R1的GE 0/0/0接口下使用命令**nat static**配置内部地址到外部地址的一对一转换。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]nat static global 202.169.10.5 inside 10.1.1.1

配置完成后，在R1上查看NAT静态配置信息，并在PC-1上使用ping命令测试与外网连通性。

<R1>display nat static

Static Nat Information:

Interface : GigabitEthernet0/0/0

Global IP/Port : 202.169.10.5/----

Inside IP/Port : 10.1.1.1/----

Protocol : ----

……

PC-1>ping 202.169.20.1

Ping 202.169.20.1: 32 data bytes, Press Ctrl\_C to break

Request timeout!

Request timeout!

Request timeout!

Request timeout!

Request timeout!

……

可以观察到PC-1还是无法与外网通信，在路由器R1的GE 0/0/0接口上抓包查看NAT地址转换是否成功。

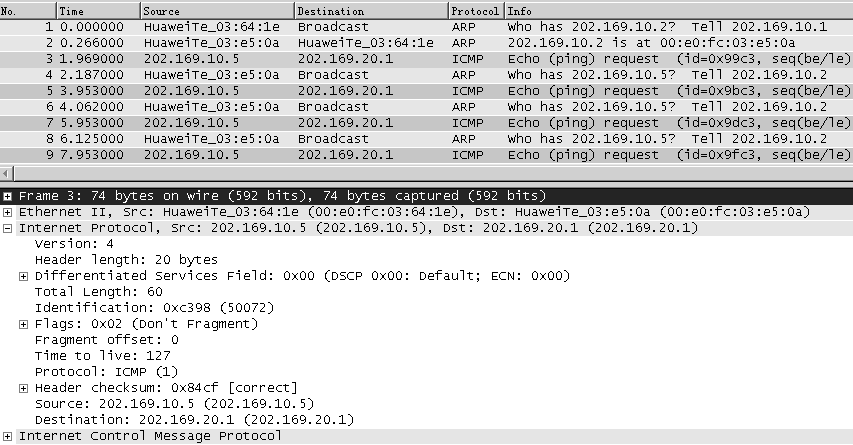


图14-14

可以观察到R1已经成功把来自PC-1的ICMP报文的源地址10.1.1.1转换成公网地址202.169.10.5，但是当R2回复报文的时候却不能通过ARP查找到202.169.10.5的MAC地址，因为R1上并没有地址为202.169.10.5的物理接口，因此没有相应的MAC地址回应。

此时需要在R1的GE 0/0/0接口开启ARP代理功能，使得R1以自身的GE 0/0/0接口的MAC地址作为202.169.10.5的MAC地址回应给公网设备。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]arp-proxy enable

配置完成后，再次测试PC-1与外网连通性。

PC>ping 202.169.20.1

Ping 202.169.20.1: 32 data bytes, Press Ctrl\_C to break

From 202.169.20.1: bytes=32 seq=1 ttl=254 time=422 ms

From 202.169.20.1: bytes=32 seq=2 ttl=254 time=156 ms

From 202.169.20.1: bytes=32 seq=3 ttl=254 time=203 ms

From 202.169.20.1: bytes=32 seq=4 ttl=254 time=47 ms

From 202.169.20.1: bytes=32 seq=5 ttl=254 time=63 ms

--- 202.169.20.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 47/178/422 ms

可以观察到，PC-1通过静态NAT地址转换已经可以成功访问外网，在R2使用环回口Loopback 0模拟外网用户访问PC-1，并在PC-1的E 0/0/1接口上抓包观察。

<R2>ping -a 202.169.20.1 202.169.10.5

PING 202.169.10.5: 56 data bytes, press CTRL\_C to break

Reply from 202.169.10.5: bytes=56 Sequence=1 ttl=127 time=260 ms

Reply from 202.169.10.5: bytes=56 Sequence=2 ttl=127 time=140 ms

Reply from 202.169.10.5: bytes=56 Sequence=3 ttl=127 time=110 ms

Reply from 202.169.10.5: bytes=56 Sequence=4 ttl=127 time=130 ms

Reply from 202.169.10.5: bytes=56 Sequence=5 ttl=127 time=40 ms

--- 202.169.10.5 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 40/136/260 ms

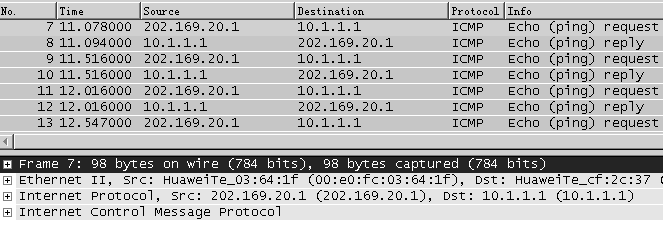


图14-15

可以观察到由于PC-1的私网地址被转换为唯一的公网地址，外网用户也能主动访问PC-1，且数据包在经过R1进入内网的时候，R1把目的IP转换为与公网地址202.169.10.5对应的私网地址10.1.1.1发给PC-1。

## 配置NAT Outbound

公司内市场部的员工都需要能够访问外网，市场部使用私网IP地址20.1.1.0/24网段，网络管理员使用公网地址池202.169.10.50－202.169.10.60为市场部员工做NAT转换。

在R1上使用命令nat addres-group配置NAT地址池，设置起始和结束地址分别为202.169.10.50和202.169.10.60。

[R1]nat address-group 1 202.169.10.50 202.169.10.60

**创建基本ACL 2000，匹配20.1.1.0，掩码为24位的地址段。**

[R1]acl 2001

[R1-acl-basic-2001]rule 5 permit source 20.1.1.0 0.0.0.255

在GE 0/0/0接口下使用命令**nat outband**将ACL 2000与地址池相关联，使得ACL中规定的地址可以使用地址池进行地址转换。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]nat outbound 2001 address-group 1 no-pat

配置完成后，在R1上查看NAT Outbound信息。

[R1]display nat outbound

NAT Outbound Information:

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

Interface Acl Address-group/IP/Interface Type

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

GigabitEthernet0/0/0 2001 1 no-pat

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

Total : 1

可以观察到R1上的NAT Outband配置信息。使用PC-2测试与外网连通性，并在R1的接口GE 0/0/0上抓包观察地址转换情况。

PC>ping 202.169.20.1

Ping 202.169.20.1: 32 data bytes, Press Ctrl\_C to break

From 202.169.20.1: bytes=32 seq=1 ttl=254 time=219 ms

From 202.169.20.1: bytes=32 seq=2 ttl=254 time=141 ms

From 202.169.20.1: bytes=32 seq=3 ttl=254 time=94 ms

From 202.169.20.1: bytes=32 seq=4 ttl=254 time=47 ms

From 202.169.20.1: bytes=32 seq=5 ttl=254 time=94 ms

--- 202.169.20.1 ping statistics ---

5 packet(s) transmitted

5 packet(s) received

0.00% packet loss

round-trip min/avg/max = 47/119/219 ms

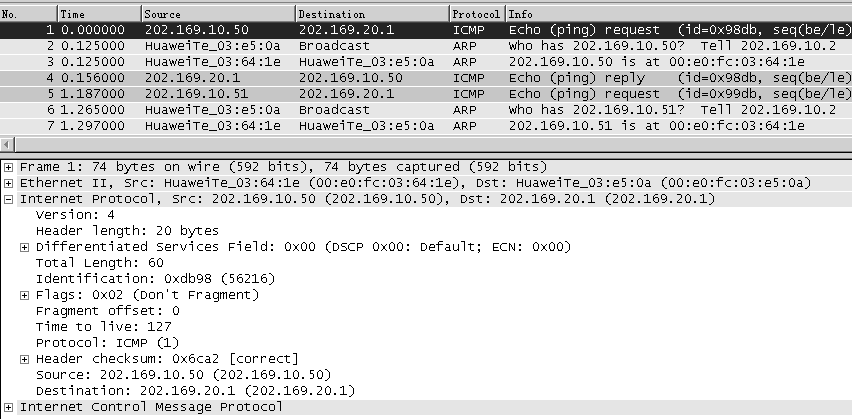


图14-16

可以观察到PC-2可以成功访问外网，且通过抓包可以观察到，来自PC-2的ICMP数据包在R1上的GE 0/0/0接口上源地址20.1.1.2被替换为地址池中第一个地址202.169.10.50。

## 配置NAT Easy－IP

由于公司发展人员扩招，若继续使用多对多的NAT转换方式，就必须增加公网地址池的地址数。为了节约公网地址，网络管理员使用多对一的Easy-IP转换方式实现市场部员工访问外网的需求。

Easy-IP是NAPT的一种方式，直接借用路由器出接口IP地址作为公网地址，将不同的内部地址映射到同一公有地址的不同端口号上，实现多对一地址转换。网络管理员配置路由器R1的GE 0/0/0接口为Easy-IP接口。

在R1的GE 0/0/0接口上删除NAT Outband配置，并使用命令**nat outband**配置Easy-IP特性，直接使用接口IP地址作为NAT转换后的地址。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]undo nat outbound 2001 address-group 1 no-pat

[R1-GigabitEthernet0/0/0]nat outbound 2001

配置完成后，在PC-2和PC-3使用UDP发包工具发送UDP数据包到公网地址202.169.20.1，配置好目的IP和UDP源、目的端口号后，输入字符串数据后点击发送。



图14-17



图14-18

在PC-2和PC-3发送UDP数据包后，在R1上查看NAT Session的详细信息。

<R1>display nat session protocol udp verbose

NAT Session Table Information:

Protocol : UDP(17)

SrcAddr Port Vpn : 20.1.1.2 2560

DestAddr Port Vpn : 202.169.20.1 2560

Time To Live : 120 s

NAT-Info

New SrcAddr : 202.169.10.1

New SrcPort : 10255

New DestAddr : ----

New DestPort : ----

Protocol : UDP(17)

SrcAddr Port Vpn : 20.1.1.3 2560

DestAddr Port Vpn : 202.169.20.1 2560

Time To Live : 120 s

NAT-Info

New SrcAddr : 202.169.10.1

New SrcPort : 10256

New DestAddr : ----

New DestPort : ----

Total : 2

可以观察到，源地址为20.1.1.2的UDP数据包被新源地址202.169.10.1和新源端口号10255替换，源地址为20.1.1.3的UDP数据包被新源地址202.169.10.1和新源端口号10256替换。R1借用自身GE 0/0/0接口的公网IP地址为所有私网地址做NAT转换，使用不同的端口号区分不同私网数据。此方式不需要创建地址池，大大节省了地址空间。

## 配置NAT Server

公司内Server提供FTP服务供外网用户访问，配置NAT Server并使用公网IP地址202.169.10.6对外公布服务器地址，然后开启NAT ALG功能，因为对于封装在IP数据报文中的应用层协议报文，正常的NAT转换会导致错误，在开启某应用协议的NAT ALG功能后，该应用协议报文可以正常进行NAT转换，否则该应用协议不能正常工作。

在R1的GE 0/0/0接口上，使用命令nat server定义内部服务器的映射表，指定服务器通讯协议类型为TCP，配置服务器使用的公网IP地址为202.169.10.6，服务器内网地址为10.1.1.3，指定端口号为21，该常用端口号可以直接使用关键字ftp代替。

[R1]interface GigabitEthernet 0/0/0

[R1-GigabitEthernet0/0/0]nat server protocol tcp global 202.169.10.6 ftp inside 10.1.1.3 ftp

[R1-GigabitEthernet0/0/0]quit

[R1]nat alg ftp enable

配置完成后，在R1上查看NAT Server信息。

<R1>display nat server

Nat Server Information:

Interface : GigabitEthernet0/0/0

Global IP/Port : 202.169.10.6/21(ftp)

Inside IP/Port : 10.1.1.3/21(ftp)

Protocol : 6(tcp)

VPN instance-name : ----

Acl number : ----

Description : ----

Total : 1

可以观察到，配置已经生效，并开启服务器的FTP功能

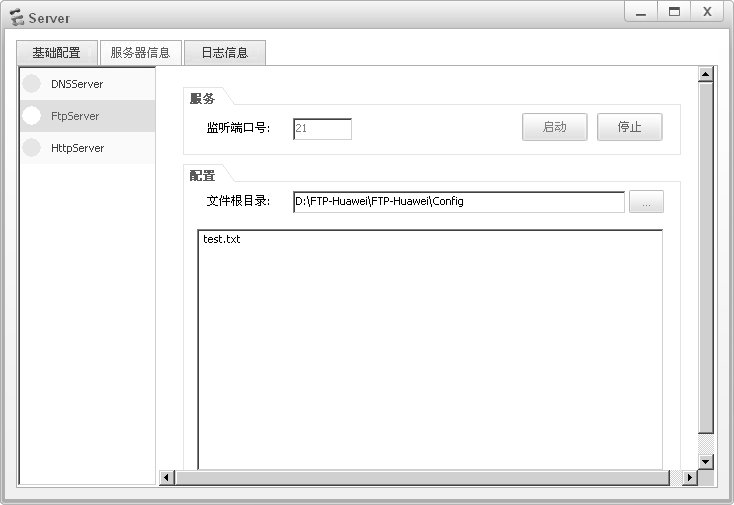


图14-19

设置完服务器后，在R2上模拟公网用户访问该私网服务器。

<R2>ftp 202.169.10.6

Trying 202.169.10.6 ...

Press CTRL+K to abort

Connected to 202.169.10.6.

220 FtpServerTry FtpD for free

User(202.169.10.6:(none)):huawei

331 Password required for huawei .

Enter password:

230 User huawei logged in , proceed

[R2-ftp]ls

200 Port command okay.

150 Opening ASCII NO-PRINT mode data connection for ls -l.

test.txt

226 Transfer finished successfully. Data connection closed.

FTP: 10 byte(s) received in 0.160 second(s) 62.50byte(s)/sec.

可以观察到，公网用户可以成功登录至公司内的私网FTP服务器。

# 思考

什么情况下需要使用到NAT的双向转换？

附录一：资源列表