网络实验手册

V1.0

南京大学软件学院

2017

目录

[第一章 交换机 4](#_Toc478741235)

[1.1 交换机概述 4](#_Toc478741236)

[1.1.1 交换机的工作原理 4](#_Toc478741237)

[1.1.2第二层交换技术 5](#_Toc478741238)

[1.1.3 转发和过滤数据包 6](#_Toc478741239)

[1.2交换机内部结构 7](#_Toc478741240)

[1.2.1构造及主要功能 7](#_Toc478741241)

[1.2.2内部结构 8](#_Toc478741242)

[1.3. 交换机配置 8](#_Toc478741243)

[1.3.1 切换命令行界面模式 8](#_Toc478741244)

[1.3.2 基本交换机配置 9](#_Toc478741245)

[1.3.3 验证交换机配置 11](#_Toc478741246)

[1.3.4基本交换机管理 11](#_Toc478741247)

[第二章 路由器 13](#_Toc478741248)

[2.1 路由器概述 13](#_Toc478741249)

[2.1.1 路由器的功能 13](#_Toc478741250)

[2.1.2 路由器的任务 13](#_Toc478741251)

[2.2 路由器的内部结构 14](#_Toc478741252)

[2.2.1 路由器的功能结构 14](#_Toc478741253)

[2.2.2路由器的系统组成 15](#_Toc478741254)

[2.3 路由器配置 16](#_Toc478741255)

[2.3.1 路由器配置途径 16](#_Toc478741256)

[2.3.2路由器状态以及配置模式 17](#_Toc478741257)

[2.3.3 路由器常用配置 17](#_Toc478741258)

[第三章 路由器基本命令 22](#_Toc478741259)

[3.1实验前准备 22](#_Toc478741260)

[3.2实验要求 22](#_Toc478741261)

[3.3实验拓扑 22](#_Toc478741262)

[3.4实验过程 23](#_Toc478741263)

[3.5实验命令列表 24](#_Toc478741264)

[3.6实验问题 24](#_Toc478741265)

[第四章 路由器密码恢复 25](#_Toc478741266)

[4.1实验前准备 25](#_Toc478741267)

[4.2实验要求 25](#_Toc478741268)

[4.3实验拓扑 25](#_Toc478741269)

[4.4实验过程 26](#_Toc478741270)

[4.5实验命令列表 27](#_Toc478741271)

[4.6实验问题 27](#_Toc478741272)

[第五章 路由器IOS备份 28](#_Toc478741273)

[5.1实验前准备 28](#_Toc478741274)

[5.2实验要求 28](#_Toc478741275)

[5.3实验拓扑 28](#_Toc478741276)

[5.4实验过程 28](#_Toc478741277)

[5.5实验命令列表 31](#_Toc478741278)

[5.6实验问题 31](#_Toc478741279)

[第六章 交换机基本命令 32](#_Toc478741280)

[6.1实验前准备 32](#_Toc478741281)

[6.2实验要求 32](#_Toc478741282)

[6.3实验拓扑 32](#_Toc478741283)

[6.4实验过程 33](#_Toc478741284)

[6.5实验命令列表 34](#_Toc478741285)

[6.6实验问题 34](#_Toc478741286)

[第七章 交换机端口安全 35](#_Toc478741287)

[7.1实验前准备 35](#_Toc478741288)

[7.2实验要求 35](#_Toc478741289)

[7.3实验拓扑 35](#_Toc478741290)

[7.4实验过程 35](#_Toc478741291)

[7.5实验命令列表 36](#_Toc478741292)

[7.6实验问题 37](#_Toc478741293)

[第八章 静态路由和简单组网 38](#_Toc478741294)

[8.1实验前准备 38](#_Toc478741295)

[8.2实验要求 38](#_Toc478741296)

[8.3实验拓扑 38](#_Toc478741297)

[8.4实验过程 38](#_Toc478741298)

[8.5实验命令列表 40](#_Toc478741299)

[8.6实验问题 40](#_Toc478741300)

[第九章 动态RIP 41](#_Toc478741301)

[9.1实验前准备 41](#_Toc478741302)

[9.2实验要求 41](#_Toc478741303)

[9.3实验拓扑 41](#_Toc478741304)

[9.4实验过程 41](#_Toc478741305)

[9.5实验命令列表 42](#_Toc478741306)

[9.6实验问题 42](#_Toc478741307)

[第十章 配置单域OSPF 43](#_Toc478741308)

[10.1实验前准备 43](#_Toc478741309)

[10.2实验要求 43](#_Toc478741310)

[10.3实验拓扑 43](#_Toc478741311)

[10.4实验过程 44](#_Toc478741312)

[10.5实验命令列表 46](#_Toc478741313)

[10.6实验问题 46](#_Toc478741314)

[第十一章 VLAN间路由 47](#_Toc478741315)

[11.1实验前准备 47](#_Toc478741316)

[11.2实验要求 47](#_Toc478741317)

[11.3实验拓扑 47](#_Toc478741318)

[11.4实验过程 48](#_Toc478741319)

[11.5实验命令列表 49](#_Toc478741320)

[第十二章 NAT网络地址转换 50](#_Toc478741321)

[12.1实验前准备 50](#_Toc478741322)

[12.2实验要求 50](#_Toc478741323)

[12.3实验拓扑 50](#_Toc478741324)

[12.4实验过程 51](#_Toc478741325)

[12.5实验命令列表 57](#_Toc478741326)

[12.6实验问题 58](#_Toc478741327)

[第十三章 ACL实验 59](#_Toc478741328)

[13.1实验前准备 59](#_Toc478741329)

[13.2实验要求 59](#_Toc478741330)

[13.3实验拓扑 59](#_Toc478741331)

[13.4实验过程 59](#_Toc478741332)

[13.5实验命令列表 63](#_Toc478741333)

[13.6实验问题 63](#_Toc478741334)

[第十四章 PPP验证实验 64](#_Toc478741335)

[14.1实验前准备 64](#_Toc478741336)

[14.2实验要求 64](#_Toc478741337)

[14.3实验拓扑 64](#_Toc478741338)

[14.4实验过程 64](#_Toc478741339)

[14.5实验命令列表 67](#_Toc478741340)

[14.6实验问题 67](#_Toc478741341)

[第十五章 帧中继实验 68](#_Toc478741342)

[15.1实验前准备 68](#_Toc478741343)

[15.2实验要求 68](#_Toc478741344)

[15.3实验拓扑 68](#_Toc478741345)

[15.4实验过程 68](#_Toc478741346)

[15.5实验命令列表 70](#_Toc478741347)

[15.6实验问题 71](#_Toc478741348)

[第十六章 DHCP欺诈保护 72](#_Toc478741349)

[16.1实验前准备 72](#_Toc478741350)

[16.2实验要求 72](#_Toc478741351)

[16.3实验拓扑 72](#_Toc478741352)

[16.4实验过程 72](#_Toc478741353)

[16.5实验命令列表 75](#_Toc478741354)

[16.6实验问题 75](#_Toc478741355)

# 第一章 交换机

## 1.1 交换机概述

交换机是一种基于MAC（网卡的硬件地址）识别，能完成封装转发数据包功能的网络设备，交换机正如它的名字一样采用的是交换的工作模式，它可以“学习”网络中各个终端的MAC 地址，并把其存放在内部的MAC 地址表中，通过在数据帧的始发者和目标接收者之间建立临时的交换路径，使数据帧直接由源地址到达目的地址。

交换机拥有一条高性能的背部总线和内部交换矩阵。交换机的所有端口均挂接在这条背部总线上，当控制电路接收到数据包后，处理端口会查找内存中的MAC 地址对照表以确定目的MAC 地址的网卡接在哪个端口上，通过内部交换矩阵直接将数据包传送到目的端口，而不是所有端口，交换机的这种工作方式较于集线器来说效率高，不浪费网络资源，因为它只是对目的地址传输数据，发送数据是其他节点很难侦听到所发送的信息。这也是交换机能很快取代集线器的重要原因之一。

交换机的另一个重要特点是它不像集线器一样每个端口共享带宽，它的每一个端口都是共享一部分交换机的总带宽，这样在速率上就对每个端口有个根本的保障。这样交换机就可以在同一时刻进行多个端口之间数据传输，每个端口都视为独立的网段，享有独立固定的带宽。无需同其他设备竞争使用。

交换机的目的是使得传输效率更高，它根据MAC 地址来进行判断，决定数据帧该送到目的地址的连接端口，而不打扰其他不相干的连接端口，如果内存中的地址表中不包含目的MAC 地址，交换机则会向所有端口广播这个数据包，找到后再将这个MAC地址加入到自己的MAC 地址表中，这样下次发送到这个地址时便不会发错。

### 1.1.1 交换机的工作原理

交换机的主要工作原理有以下几条：

1. 地址表：端口地址表记录了端口下包含主机的MAC地址。端口地址表是交换机上电后自动建立的，保存在RAM中，并且自动维护。交换机隔离冲突域的原理是根据其端口地址表和转发决策决定的。
2. 转发决策：交换机的转发决策有三种操作：丢弃、转发和扩散。丢弃：当本端口下的主机访问已知本端口下的主机时丢弃。转发：当某端口下的主机访问已知某端口下的主机时转发。扩散：当某端口下的主机访问未知端口下的主机时要扩散。每个操作都要记录下发包端的MAC地址，以备其它主机的访问。
3. 生存期：生存期是端口地址列表中表项的寿命。每个表项在建立后开始进行倒计时，每次发送数据都要刷新计时。对于长期不发送数据主机，其MAC地址的表项在生存期结束时删除。所以端口地址表记录的总是最活动的主机的MAC地址。
4. 三层路由：通常，普通的交换机只工作在数据链路层上，路由器则工作在网络层。而功能强大的三层交换机可同时工作在数据链路层和网络层，并根据 MAC地址或IP地址转发数据包。但是要注意到三层交换机并不能完全取代路由器，因为它主要是为了实现处于两个不同子网的Vlan进行通讯，而不是用来作数据传输的复杂路径选择。
5. 网管功能：一台交换机所支持的管理程度反映了该设备的可管理性与可操作性。带网管功能的交换机可对每个端口的流量进行监测，设置每个端口的速率，关闭/打开端口连接。通过对交换机端口进行监测，便于对网络业务流量的区分和迅速进行网络故障定义，提高了网络的可管理性。
6. 端口聚合：这是一种封装技术，它是一条点到点的链路，链路的两端可以都是交换机，也可以是交换机和路由器，还可以是主机和交换机或路由器。基于端口汇聚（Trunk）功能，允许交换机与交换机、交换机与路由器、主机与交换机或路由器之间通过两个或多个端口并行连接同时传输以提供更高带宽、更大吞吐量，大幅度提高整个网络能力。

### 1.1.2第二层交换技术

地址学习

以太网交换机通过学习地址来进行数据的转发操作。交换机开机启动后，会自动生成一张表，即MAC地址表，交换机关机后，MAC地址表中的内容会自动清空。

交换机MAC地址表用于记录连到交换机的所有设备的位置。交换机的目标是分割网上通信量，使发送到给定冲突域中主机的数据包不至于传播到另一个网段。这是由交换机的“学习”功能完成的，“学习”功能使交换机了解到主机位于哪里。交换机的学习和转发过程如下：

- 当一个交换机首次初始化时，交换机地址表是空的。

- 用一个空MAC地址表，基于地址的源过滤或转发决策是不可能的，因此交换机将每一帧转发给所有连接的端口，而不只是接受的端口。

- 转发一个帧到所有连接端口，称为“泛洪”。泛洪是一种通过交换机传输数据的低效方法，因为它将数据帧传输到了不需要的网段，浪费了带宽。

- 因为交换机能同时处理多个网段的通信量，交换机执行内存缓冲以致能独立接受、传输每个端口或网段的数据帧。

MAC地址表中的内容主要包括交换机端口、与交换机端口相连的主机MAC地址、VLAN标识。图1.1显示了不同网段间两个工作站之间的事务。MAC地址为0260.8c01.1111的站点A准备发送数据到MAC地址为0260.8c01.2222的站点C，交换机接受该帧，执行以下几个动作：

1. 从物理以太网接收该帧并且存储到临时缓冲区。

2. 因为交换机不知道哪个接口连到目的站点，它将帧泛洪给所有端口。

3. 当站点A泛洪该帧时，交换机在MAC地址表中记录发送数据包站点的源地址及与之相连的端口F0/1。

4. 如果该记录在一定时间内没有新的帧传到交换机来刷新，这个记录被废弃。

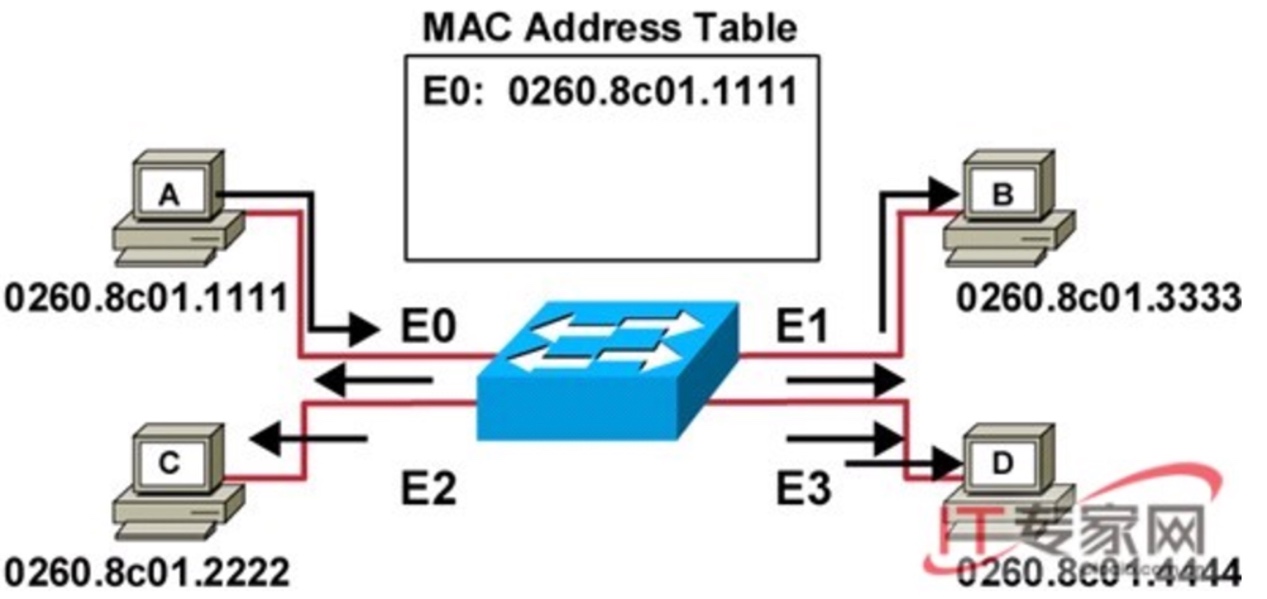


图1.1 泛洪数据包

当站点继续发送帧到另一个站点时，学习过程继续。MAC地址为0260.8c01.4444的站点D给MAC地址为0260.8c01.2222的站点C发送数据包，交换机采取以下几个动作：

1. 源地址0260.8c01.4444被加到MAC地址表中。

2. 将传输帧的目的MAC地址站点C与MAC地址表记录进行比较。

3. 当软件决定对这个目的地来说至此未见端口到MAC地址映射时，该帧被泛洪到交换机中所有的端口。

当站点A发送一帧到站点C时，交换机查询到站点C的MAC地址和端口F0/2，这里交换机将站点A发送的数据包直接转发给站点C，而不发送给B和D站点，如图1.2所示。只要在MAC地址表中记录生命周期内所有站点发送数据帧，就可以建立起完整的MAC地址表。

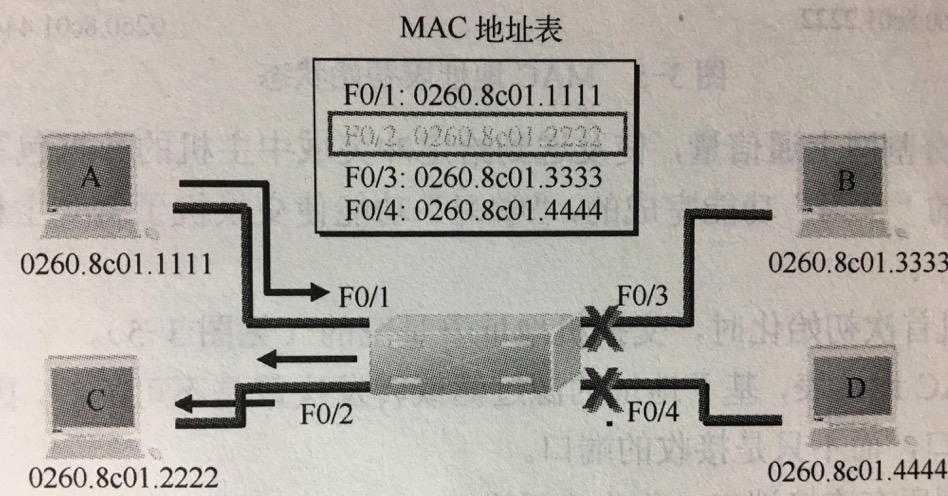


图1.2 站点应答

### 1.1.3 转发和过滤数据包

当一个帧带有一个已知目的地址到达时，它被转发到连接该站点而不是所有站点的端口。在图1.3中，站点A给站点C发送一帧。当目的MAC地址（站点C的MAC地址）已在MAC地址表中时，交换机只将帧传输到表中所列的这个端口。

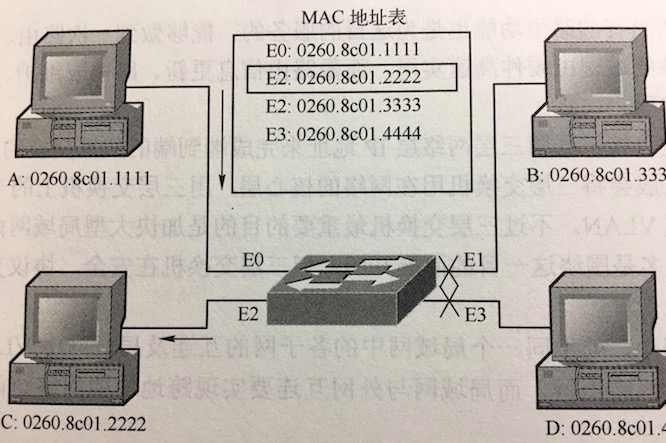


图1.3 交换机过滤决策

站点A发送帧给站点C的步骤如下：

1. 传输帧的目的MAC地址0260.8c01.2222与MAC地址表中项进行比较。

2. 当交换机决定目的地址经由E2端口可到达时，它将该帧传到该端口。

3. 交换机为了保护链路上的带宽没有将帧传到E1和E3口，这个动作称为“帧过滤”。

广播和组播是一个特殊情况。因为广播帧和组播帧可能所有站点都关心，交换机通常将广播帧和组播帧泛洪给发起端口外的所有端口。交换机从来不学习广播或组播地址，因为广播地址和组播地址不出现在帧的源地址中。所有站点接受广播帧的事实意味着所有交换网络中网段是在同一广播域中。

## 1.2交换机内部结构

### 1.2.1构造及主要功能

交换机是一种基于MAC地址识别，能完成封装转发数据包功能的网络设备。交换机可以“学习”MAC地址，并把其存放在内部地址表中，通过在数据帧的始发者和目标接收者之间建立临时的交换路径，使数据帧直接由源地址到达目的地址。

传统的交换机工作在OSI 模型中的第二层，可以将其看作为一台专用的特殊计算机，主要包括中央处理器(CPU)、随机存储器(RAM)和操作系统。它利用专门设计的芯片ASIC(Application Specific Integrated Circuits)使交换机以线路速率在所有的端口并行进行转发，因此，它比同在二层利用软件进行转发的网桥速度快的多。

交换机使用一种虚拟连接技术来连接通信的双方。所谓虚拟连接，就是指通信时通信双方建立一个逻辑上的专用连接，这个连接直到数据传送至目的节点后结束。虚拟连接是通过交换机的端口-地址表来实现的：交换机在工作过程中不断地建立和维护它本身的一个地址表，这个地址表标明了节点的MAC地址和交换机端口的对应关系。当交换机收到一个数据包，它便会去查看自身的地址表以验明数据包中的目的MAC地址究竟对应于哪个端口。一旦验证完毕，就将发送节点与该端口建立一个专用连接，发送方的数据仅发送到目的MAC 地址所对应的交换机端口。

### 1.2.2内部结构

局域网交换机卓越的性能表现，来源于其内部独特的技术结构。而不同的交换模式或不同的交换类型，也跟局域网交换机内部结构密不可分。目前局域网交换机采用的内部技术结构主要有以下几种。

1. 共享内存式结构

该结构依赖于中心局域网交换机引擎所提供的全端口的高性能连接，并由核心引擎完成检查每个输入包来决定连接路由。这种方式需要很大的内存带宽和很高的管理费用，尤其是随着局域网交换机端口的增加，需要内存容量更大，速度也更快，中央内存的价格就变得很高，从而使得局域网交换机内存成为性能实现的主要瓶颈。

1. 交叉总线式结构

交叉总线式结构可在端口间建立直接的点对点连接，这种结构对于简单的单点式（Unicast）信息传输来讲性能很好，但并不适合点对多点的广播式传输。由于实际网络应用环境中，广播和多播传输方式很常见，所以这种标准的交叉总线方式会带来一些传输问题。例如，当端口A向端口D传输数据时，端口B和端口C就只能等待。而当端口A向所有端口广播消息时，就可能会引起目标端口的排队等候。这样将会消耗掉系统大量带宽，从而影响局域网交换机传输性能。而且要连接N个端口，就需要N×（N+1）条交叉总线，因而实现成本也会随着端口数量的增加而急剧上升。

1. 混合交叉总线式结构

鉴于标准交叉总线存在的缺陷，一种混合交叉总线实现方式被提了出来。该方式的设计思路是将一体的交叉总线矩阵划分成小的交叉矩阵，中间通过一条高性能总线连接。该结构的优点是减少了交叉总线数，降低了成本，还减少了总线争用。但连接交叉矩阵的总线成为新的性能瓶颈。

1. 环形总线式结构

这种结构方式在一个环内最多可支持四个交换引擎，并且允许不同速度的交换矩阵互连，以及环与环间通过交换引擎连接。由于采用环形结构，所以很容易聚集带宽。当端口数增加的时候，带宽就相应增加了。与前述几种结构不同的是，该结构方式有独立的一条控制总线，用于搜集总线状态、处理路由、流量控制和清理数据总线。另外，在环形总线上可以加入管理模块，提供完整的SNMP管理特性。同时还可以根据需要选用第三层交换功能。这种结构的最大优点就是扩展能力强，实现成本低，而且有效地避免了系统扩展时造成的总线瓶颈。

## 1.3. 交换机配置

以Cisco交换机为例，说明交换机的一些常规配置。

### 1.3.1 切换命令行界面模式

作为一项安全功能，Cisco IOS软件将EXEC（执行）会话分成以下两种访问级别。

用户执行：只允许用户访问有限量的基本监视命令。用户执行模式是在从CLI登录到Cisco交换机后所进入的默认模式。用户执行模式由">"提示符标识。

特权执行：允许用户访问所有设备命令，如用于配置和管理的命令，特权执行模式可采用口令加以保护，使得只有获得授权的用户才能访问设备。特权执行模式由#提示符标识。

要从用户执行模式切换到特权执行模式，输入enable命令。要从特权执行模式切换到用户执行模式，输入disable命令。在实际网络中，交换机将提示输入口令。默认情况下未配置口令。表1.1中显示了用于在用户执行模式和特权执行模式之间来回切换的Cisco IOS命令。

表1.1 执行模式切换

|  |  |
| --- | --- |
| 说    明 | CLI |
| 从用户执行模式切换到特权执行模式 | Switch>enable |
| 如果已为特权执行模式设置口令，则系统将提示您现在输入口令 | Password:password |
| #提示符表示已处于特权执行模式 | Switch# |
| 从特权执行模式切换到用户执行模式 | Switch#disable |
| >提示符表示已处于用户执行模式 | Switch> |

在Cisco交换机上进入特权执行模式之后，就可以访问其他配置模式。Cisco IOS软件的命令模式结构采用分层的命令结构。

### 1.3.2 基本交换机配置

局域网中的第2层交换机的一些关键配置序列通常是在实施过程中进行的。这些配置序列包括配置交换机管理界面，默认网关，全双工和活动接口上的网速设置，对HTTP访问的支持和对MAC地址表的管理。

1. 管理接口

接入层交换机需要配置IP地址、子网掩码和默认网关。要使用TCP/IP来远程管理交换机，就需要为交换机分配IP地址。此IP地址将分配给称为虚拟LAN（VLAN）的虚拟接口，然后必须确保VLAN分配到计算机上的一个或多个特定端口。

要在交换机的管理VLAN上配置IP地址和子网掩码，必须处在VLAN接口配置模式下。先使用命令interface vlan 99，再输入ip address配置命令。必须使用no shutdown接口配置命令来使此第3层接口正常工作。当看到"interface VLAN x"时，这是指与VLAN x 关联的第3层接口。只有管理VLAN才有与之关联的"interface VLAN"。表1.2列出了Catalyst 2960交换机上的管理接口配置。

表1.2 管理接口配置

|  |  |
| --- | --- |
| 说    明 | 命    令 |
| 进入全局配置模式 | S1#configure terminal |
| 进入VLAN99接口的接口配置模式 | S1(config)#interface vlan 99 |
| 配置接口IP地址 | S1(config-if)#ip address 172.11.99.11 255.255.255.0 |
| 启动接口 | S1(config-if)#no shutdown |
| 返回特权执行模式 | S1(config-if)#end |
| 进入全局配置模式 | S1#configure terminal |
| 输入要分配VLAN的接口 | S1(config)#interface fastethernet 0/18 |
| 定义端口的VLAN成员模式 | S1(config-if)#switchport mode access |
| 将端口分配给VLAN | S1(config-if)#switchport access vlan 99 |
| 返回特权执行模式 | S1(config-if)#end |
| 将运行配置保存为交换机启动配置 | S1#copy running-config startup-config |

1. 默认网关

默认网关是用于将IP数据包转发到远程网络的机制。交换机将目的IP地址位于本地网络之外的IP数据包转发到默认网关。

使用ip default-gateway命令，为交换机配置默认网关。输入与需要配置默认网关的交换机直接相连的下一跳路由器接口的IP地址。

1. 双工和速度

可以使用duplex接口配置命令指定交换机端口的双工操作模式。可以手动设置交换机端口的双工模式和速度，以避免厂商间的自动协商问题。在将交换机端口双工设置配置为auto时可能出现问题，在图1.5中，S1和S2交换机有着相同的双工设置和速度。

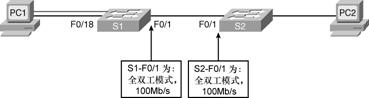


图1.5 双工和速度

在交换机S1上配置端口F0/1的具体步骤描述如图1.6所示。

S1# configure terminal

S1(config)# interface fastethernet 0/1

S1(config-if)# duplex auto

S1(config-if)# speed auto

S1(config-if)# end

图1.6 duplex和speed命令

1. HTTP访问

如图1.7是启用HTTP访问的基本配置，ip http authentication enable 是全局配置命令模式。

图1.7 启用HTTP访问的基本配置

S1# configure terminal

S1(config)# ip http authentication enable

S1(config)# ip http server

1. 管理MAC地址表

交换机使用MAC地址表来确定如何在端口间转发流量。这些MAC表包含动态地址和静态地址。用show mac-address-table命令显示MAC地址表，其输出包含静态和动态MAC地址。

使用mac-address-table static MAC-address vlan vlan-id interface interface-id命令可在MAC地址表中创建静态映射。使用no mac-address-table static MAC-address vlan vlan-id interface interface-id命令可移除MAC地址表中的静态映射。

### 1.3.3 验证交换机配置

执行初始交换机配置之后，可使用不同的show命令来验证交换机是否已正确配置。

show命令从特权执行模式下执行。表1.3列出了show命令的一些关键选项，它们可用于验证几乎所有可配置的交换机功能。

表1.3 show命令

|  |  |
| --- | --- |
| 说明 | 命令 |
| 显示交换机上单个或全部可用接口的接口状态和配置 | show interface {interface-id  ¦cr} |
| 显示启动配置的内容 | show startup-config |
| 显示当前运行配置 | show running-config |
| 显示关于flash：文件系统的信息 | show flash: |
| 显示系统硬件和软件状态 | show version |
| 显示会话命令历史记录 | show history |
| 显示IP信息 Interface选项显示IP接口状态和配置 http选项显示有关正在交换机上运行的Device Manager的HTTP信息 arp选项显示IP ARP表 | show ip {interface | http |arp }  //show ip http:显示不完整命令 |
| 显示MAC转发表 | show mac-address-table |

### 1.3.4基本交换机管理

交换机启动并运行之后，网络技术人员必须对交换机进行维护，这就包括备份和恢复交换机的配置文件，清除配置信息和删除配置文件。

1. 备份和恢复交换机配置文件

使用copy running-config startup-config特权执行命令备份了目前创建的配置。如果想在设备上保留多个不同的startup-config文件，则可以使用copy startup-config flash:filename命令将配置复制到不同文件名的多个文件中。存储多个startup-config版本可用于在配置出现问题时回滚到某个时间点。

恢复配置是一个简单的过程。只需用已存配置覆盖当前配置即可。例如，如果有名为config.bak1的已存配置，则输入Cisco IOS命令copy flash:config.bak1 startup-config即可覆盖现有start-config并恢复config.bak1的配置。当配置恢复到startup-config中后，可在特权执行模式下使用reload命令重新启动交换机，如表1.4所示，使计算机重新加载新的启动配置。reload命令将使系统停止。应在配置信息已输入到文件并保存到启动配置之后再使用reload命令。

表1.4恢复备份文件

|  |  |
| --- | --- |
| 说    明 | CLI |
| 将存储在闪存中的config.bak1文件复制到存储在闪存中的启动配置中。按Enter键接受，使用Ctrl+C组合键取消 | S1#copy flash:config.bak1 startup-config Destination filename [startup-config]? |
| 使Cisco IOS执行重新启动交换机。如果修改了运行配置文件，系统将询问是否保存。请按“y”或“n”确认。要确认重新装入，请按Enter键接受，使用Ctrl+C组合键取消 | S1#reload System configuration has been modified. Save?[yes/no]:n Proceed with reload?[confirm] |

1. 清除配置信息

交换机配置的清除使用erase nvram:或erase startup-config特权执行命令实现。当网络技术人员可能执行了一项很复杂的配置任务，并在闪存中存储了文件的很多备份副本，此时要从闪存中删除文件，要使用delete flash:filename特权执行命令。根据file prompt全局配置命令的设置，系统可能在技术人员删除文件之前提示确认。默认情况下，在删除文件时，交换机都会提示确认。抹除或删除配置之后，即可重新加载交换机以启动交换机的新配置。

# 第二章 路由器

## 2.1 路由器概述

### 2.1.1 路由器的功能

路由器是在网络层实现互联的设备。路由器实现网络层上数据包的存储转发，它具有路径选择功能，可依据网络当前的拓扑结构，选择“最佳”路径，把接收的数据包转发出去，从而实现网络负载平衡，减少网络拥塞路由器工作在网络层，用于连接不同的局域网和广域网，故称为“LAN网间互联设备”。一个路由器可以连接两个局域网、一个局域网和一个广域网，或两个广域网。

路由器的具体功能如下：

1.路由功能（寻径功能）——寻找并记录到达目的网段的最佳路径，体现在路由器上则包括路由表的建立、维护和查找

2.交换功能——路由器的交换功能与以太网交换机执行的交换功能不同，路由器的交换功能是指在网络之间转发分组数据的过程，涉及到从接收接口收到数据帧，解封装，对数据包做相应处理，根据目的网络查找路由表，决定转发接口，做新的数据链路层封装等过程

3.隔离广播、指定访问规则——路由器阻止广播的通过，并且可以设置访问控制列表(ACL)对流量进行控制

4.异种网络互连——支持不同的数据链路层协议，可以连接异种网络

5.子网间的速率匹配——路由器有多个接口，不同接口具有不同的速率，路由器需要利用缓存及流控协议进行速率适配

### 2.1.2 路由器的任务

路由器的主要任务是把通信引导到目的地网络，然后到达特定的节点站地址。后一个功能是通过网络地址分解完成的。例如，把网络地址部分的分配指定成网络、子网和区域的一组节点，其余的用来指明子网中的特别站。分层寻址允许路由器对有很多个节站的网络存储寻址信息。在广域网范围内的路由器按其转发报文的性能可以分为两种类型，即中间节点路由器和边界路由器。尽管在不断改进的各种路由协议中，对这两类路由器所使用的名称可能有很大的差别，但所发挥的作用却是一样的。中间节点路由器在网络中传输时，提供报文的存储和转发。同时根据当前的路由表所保持的路由信息情况，选择最好的路径传送报文。由多个互连的LAN组成的公司或企业网络一侧和外界广域网相连接的路由器，就是这个企业网络的连界路由器。它从外部广域网收集向本企业网络寻址的信息，转发到企业网络中有关的网络段；另一方面集中企业网络中各个LAN段向外部广域网发送的报文，对相关的报文确定最好的传输路径。

## 2.2 路由器的内部结构

### 2.2.1 路由器的功能结构

路由器结构从功能上可以分成两个部分：分组转发部分和路由选择部分。分组转发主要由三个部分组成：输入端口，输出端口，交换结构。路由选择部分也可以称作控制部分，其核心是路由选择处理机。

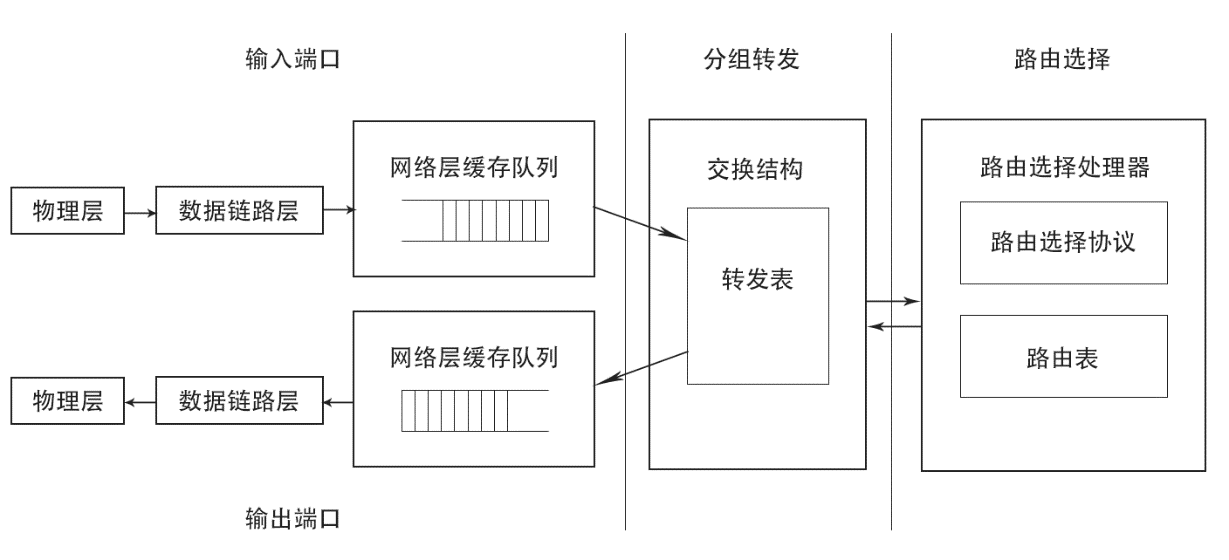


图2.1 路由器的功能结构示意图

1.输入端口：输入端口是物理链路和输入包的进口处。端口通常由线卡提供，一块线卡一般支持4、8或16个端口，一个输入端口具有许多功能。第一个功能是进行数据链路层的封装和解封装。第二个功能是在转发表中查找输入包目的地址从而决定目的端口（称为路由查找），路由查找可以使用一般的硬件来实现，或者通过在每块线卡上嵌入一个微处理器来完成。第三，为了提供QoS（服务质量），端口要对收到的包分成几个预定义的服务级别。第四，端口可能需要运行诸如SLIP（串行线网际协议）和PPP（点对点协议）这样的数据链路级协议或者诸如PPTP（点对点隧道协议）这样的网络级协议。一旦路由查找完成，必须用交换开关将包送到其输出端口。如果路由器是输入端加队列的，则有几个输入端共享同一个交换开关。这样输入端口的最后一项功能是参加对公共资源（如交换开关）的仲裁协议。

2.交换结构：交换结构可以使用多种不同的技术来实现。迄今为止使用最多的交换结构技术是总线、交叉开关和共享存贮器。最简单的开关使用一条总线来连接所有输入和输出端口，交换结构的缺点是其交换容量受限于总线的容量以及为共享总线仲裁所带来的额外开销。交叉开关通过开关提供多条数据通路，具有N×N个交叉点的交叉开关可以被认为具有2N条总线。如果一个交叉是闭合，输入总线上的数据在输出总线上可用，否则不可用。交叉点的闭合与打开由调度器来控制，因此，调度器限制了交换开关的速度。在共享存贮器路由器中，进来的包被存贮在共享存贮器中，所交换的仅是包的指针，这提高了交换容量，但是，开关的速度受限于存贮器的存取速度。尽管存贮器容量每18个月能够翻一番，但存贮器的存取时间每年仅降低5%，这是共享存贮器交换开关的一个固有限制。

3.输出端口：输出端口在包被发送到输出链路之前对包存贮，可以实现复杂的调度算法以支持优先级等要求。与输入端口一样，输出端口同样要能支持数据链路层的封装和解封装，以及许多较高级协议。

4.路由处理器：路由处理器计算转发表实现路由协议，并运行对路由器进行配置和管理的软件。同时，它还处理那些目的地址不在线卡转发表中的包。

### 2.2.2路由器的系统组成

以cisco路由器系统组成为例：

1. **CPU**

与计算机一样，路由器也包含了一个中央处理器（CPU）。不同系列和型号的路由器，其中的CPU也不尽相同。Cisco路由器一般采用Motorola 68030和Orion/R4600两种处理器。

路由器的CPU负责路由器的配置管理和数据包的转发工作，如维护路由器所需的各种表格以及路由运算等。路由器对数据包的处理速度很大程度上取决于CPU的类型和性能。

1. **存储器**

ROM:存储开机诊断程序，用于引导操作系统，类似于计算机的BIOS

RAM:路由器的主存储器，存放Running-config，路由器，ARP表，类似于计算机的内存。

FLASH:路由器的快闪存储器，用于存放路由器的IOS，类似于计算机硬盘。

NVRAM:非易失存储器，用于放置启动配置文件Startup-Config文件

1. **接口**

所有路由器都有接口（Interface），每个接口都有自己的名字和编号。一个接口的全名称由它的类型标志与数字编号构成，编号自0开始。

对于接口固定的路由器（如Cisco 2500系列）或采用模块化接口的路由器（如Cisco 4700系列），在接口的全名称中，只采用一个数字，并根据它们在路由器的物理顺序进行编号，例如Ethernet0表示第1个以太网接口，Serial1表示第2个串口。

对于支持“在线插拔和删除”或具有动态更改物理接口配置的路由器，其接口全名称中至少包含两个数字，中间用斜杠“/”分割。其中，第1个数字代表插槽编号，第2个数字代表接口卡内的端口编号。如Cisco 3600路由器中，serial3/0代表位于3号插槽上的第1个串口。

对于支持“万用接口处理器（VIP）”的路由器，其接口编号形式为“插槽/端口适配器/端口号”，如Cisco 7500系列路由器中，Ethernet4/0/1是指4号插槽上第1个端口适配器的第2个以太网接口。

1. 控制台端口

几乎所有路由器都在路由器背后安装了一个控制台端口。控制台端口提供了一个EIA/TIA—232(以前叫作RS—232)异步串行接口、使能与路由器通信。至于同控制台端口建立哪种形式的物理连接，则取决于路由器的型号。有些路由器采用一个DB25母连接(DB25F)，有些则用RJ45连接器。通常，较小的路由器采用RJ45控制台连接器，而较大路由器采用DB25控制台连接器。

1. 辅助端口

大多数Cisco路由器都配备了一个“辅助端口”(Auxiliary Port)。它和控制台端口类似，提供了一个EIA／TIA—232异步串行连接，能与路由器通信。辅助端口通常用来连接Modem，以实现对路由器的远程管理。远程通信链路通常并不用来传输平时的路由数据包，它的主要的作用是在网络路径或回路失效后访问一个路由器。

1. **IOS**

IOS为CISCO的专有操作系统，功能有连接多种网络，用于不同协议的路由和转换，实现流量控制、QoS服务质量控制、网络安全服务，网络拨号及VPN等。

有两种类型的IOS配置：

1)运行配置：有时也称作“活动配置”，驻留于RAM，包含了目前在路由器中“活动”的IOS配置命令。配置IOS时，就相当于更改路由器的运行配置。

2)启动配置：启动配置驻留在NVRAM中，包含了希望在路由器启动时执行的配置命令。有时也把启动配置称作“备份配置”。这是由于修改并认可了运行配置后，通常应将运行配置复制到NVRAM里，将作出的改动“备份”下来，以便路由器下次启动时调用。启动完成后，启动配置中的命令就变成了“运行配置”。

两者均以ASCII文本格式显示。所以，能够很方便地阅读与操作。一个路由器只能从这两种类型中选择一种。

## 2.3 路由器配置

### 2.3.1 路由器配置途径

**1.控制台**

将PC机的串口直接通过Rollover线与路由器控制台端口Console相连，在PC计算机上运行终端仿真软件，与路由器进行通信，完成路由器的配置。也可将PC与路由器辅助端口AUX直接相连，进行路由器的配置。

**2.虚拟终端(Telnet)**

如果路由器已有一些基本配置，至少有一个端口有效(如Ethernet口)，就可通过运行Telnet程序的计算机作为路由器的虚拟终端与路由器建立通信，完成路由器的配置。

**3.网络管理工作站**

路由器可通过运行网络管理软件的工作站配置，如Cisco的CiscoWorks、HP的OpenView等。

**4.CISCO　ConfigMaker**

ConfigMaker是一个由CISCO开发的免费的路由器配置工具。ConfigMaker采用图形化的方式对路由器进行配置，然后将所做的配置通过网络下载到路由器上。ConfigMaker要求路由器运行在IOS　11.2以上版本，可用Show Version命令查看路由器的版本信息。

5.TFTP(Trivial File Transfer Protocol)服务器

TFTP是一个TCP/IP简单文件传输协议，可将配置文件从路由器传送到TFTP服务器上，也可将配置文件从TFTP服务器传送到路由器上。TFTP不需要用户名和口令，使用非常简单。

注意：路由器的第一次设置必须通过第一种方式进行；这时终端的硬件设置为波特率：9600，数据位：8，停止位：1，无校验。

### 2.3.2路由器状态以及配置模式

路由器的配置模式是通过控制台连接路由器进入的模式，该模式下路由器有以下几个状态。

1. **用户命令状态**

前置符类似“Router>”，此时路由器处于用户命令状态，这时用户可以看路由器的连接状态，访问其它网络和主机，但不能看到和更改路由器的设置内容。

1. **特权命令状态**

前置符类似“Router#”，用户命令状态下输入“enable”即可进入，此时路由器处于特权命令状态，这时不但可以执行所有的用户命令，还可以看到和更改路由器的设置内容。

1. **全局设置状态**

前置符类似“Router(config)#”，特权命令状态下输入“configure terminal”即可进入，此时路由器处于全局设置状态，这时可以进行路由器端口以外的一些设置，如：路由协议，nat等。

1. **局部设置状态**

从全局设置状态进入，对某个功能的详细设置，这时可以设置路由器某个局部的参数。

1. **RXBOOT状态**

前置符为“>”，在开机后60秒内按ctrl-break可进入此状态，这时路由器不能完成正常的功能，只能进行软件升级和手工引导。

1. **设置对话状态**

这是一台新路由器开机时自动进入的状态，在特权命令状态使用SETUP命令也可进入此状态。这时可通过对话方式对路由器进行设置。

### 2.3.3 路由器常用配置

**[路由器使用注意事项]**

1. 须确认线路连接正确后才能打开路由器电源。
2. 绝对不允许热插拔flash卡（用于装载IOS），否则易造成flash卡烧毁。
3. 不允许频繁开关路由器。
4. **连接路由器**
5. 用console线（反转线，注意与网线的比较）把计算机的串口（com1, i.e. RS232）与路由器的console口直接相连。
6. 在win2000中打开“附件/通讯/超级终端”建立连接，在连接设置的波特率选择9600，其余为默认选项。
7. 开机，通常进入用户模式，使用Enable命令，进入特权模式。
8. **状态命令**

show version 这个命令可以查看IOS版本号，已启动时间，flash中的IOS的文件名，router里面共有什么的端口，寄存器的值等等。

show protocol 显示与IP有关的路由协议信息。各个端口的情况。

show flash 查看flash中的内容，IOS的长度，文件名，剩余空间，总共空间。

show running-config 查看路由器当前的配置信息。

show startup-config 查看nvram中的路由器配置信息。

show interface 查看路由器上的各个端口的状态信息。（很多重要信息）

show controller 查看接口控制器的状态，可看到连接的是DTE还是DCE

show history 查看history buffer 里面的命令列表

show controller s0 查看s0是DCE口还是DTE口

show ip route 查看路由器的路由配置情况

show hosts 查看IP host 表

terminal history size<size> 设置history buffer 里面保存命令的个数，最大允许为256

1. **修改系统时钟（按步骤体验一下？的作用）可以顺带提一提tab键的功能**
2. clock
3. clock ?
4. clock set ?
5. clock set 10:30:30 ?
6. clock set 10:30:30 20 oct ?
7. clock set 10:30:30 20 oct 2001 ?
8. enter
9. show clock
10. **使用组合键编辑**

输入一行命令（不执行它），然后操作下列组合键：

Ctrl+A：光标回到命令行的最开头

Ctrl+E：光标回到命令行的最后

Ctrl+B：光标向左移一字符位置

Ctrl+F：光标向右移一字符位置

执行刚刚输入的命令，然后操作下列组合键：

Ctrl+P（or 上箭头）：使用上一条用过的命令

Ctrl+N（or 下箭头）：使用下一条用过的命令

Ctrl+Z（在其他模式下）：保存设置并退出到特权模式

可以使用terminal no editing 命令来使组合键失效，要使组合键重新生效，可用terminal editing 命令。

1. **路由器中各种配置模式的转换**

路由器的几种配置模式：

1） 用户模式（user mode） 　 router>

2） 特权模式（privileged mode） router#

3） 全局配置模式（golbal configuration mode）:　　　　　　 router (config)#

4） Setup模式（setup mode）：

5） ROM Monitor 模式(ROM Monitor Mode): 　> 或rommon>。

6） RXBoot模式(RXBoot mode)： Router<boot>

（注：前3种模式是该实验需要用到的）

7） 用户模式（user mode）：该模式下只能查看路由器基本状态和普通命令，不能更改路由器配置。此时路由器名字后跟一个“>”符号，表明是在用户模式下。如：router>

8） 特权模式（privileged mode）：该模式下可查看各种路由器信息及修改路由器配置。在用户模式下以enable命令登陆，此时“>”将变成“#”，表明是在privileged mode .如：router#

9） 全局配置模式（global configuration mode）:该模式下可进行更高级的配置，并可由此模式进入各种配置子模式。其提示符如：router (config)#

10）Setup模式（setup mode）：该模式通常是在配置文件(configuration file)丢失的情况下进入的，以进行手动配置。在此模式下只保存着配置文件的最小子集，再以问答的形式由管理员选择配置。

11） ROM Monitor 模式(ROM Monitor Mode):当路由器启动时没有找到IOS时，自动进入该模式。提示符为> 或rommon>。

12）RXBoot模式(RXBoot mode)：该模式通常用于密码丢失时，要进行破密时进入。其提示符如：Router<boot>

Router>

Router>enable

Router#

Router#configure terminal

Router(config)#

Router(config)#int f0/0

Router(config-if)#

输入Ctrl+Z

Router#

1. **给路由器命名**

进入全局配置模式，用hostname <name>命令来设定路由器的名称。

1. **编辑路由器登录信息**

banner motd <message>

例：banner motd #

You are logging in C1600 Router

#

1. **给端口配IP地址**

在全局配置模式下，进入各端口配置模式配置IP地址。

1. 以太网口的配置

Router(config) # int f0/0

Router（config-if）# ip address <ipaddress><subnet marsk>

Router(config-if) # no shutdown

1. 串行线，根据串口是DTE还是DCE选择下面的配置

**DTE:**

Router(config)# int s0/1

Router(config-if) # ip address <ip address><subnet marsk>

Router(config-if)#bandwidth 64

Router(config-if)# no shutdown

**DCE:**

Router(config)# int s0/0

Router(config-if) # ip address <ip address><subnet marsk>

Router(config-if)#bandwidth 64

Router(config-if)#clock rate 56000

Router(config-if)# no shutdown

1. **Ping命令**

ping <ip address>

命令： Router# ping <ip address>

Router# ping <hostname>

分别从路由器和主机上使用ping命令

路由器间互ping：

Router1 <——> Router2

Router2 <——> Router3

Router1 <——> Router3

1. **CDP配置及查看**

sh cdp 注意输出的信息

conf t

cdp timer ?

cdp timer 90

sh cdp 注意输出的信息与上一次有何不同

sh cdp ?

sh cdp entry ?

sh cdp entry \*

sh cdp neighbors

1. **配置文件的复制与保存**
2. copy running-config startup-config
3. copy startup-config running-config
4. erase startup-config
5. show startup-config
6. **设置Telnet登陆用密码**

能进行telnet的前提：

1）主机能ping通路由器；

2）路由器设置了telnet密码；

3）路由器允许通过telnet登录；

4）如果需要进入特权模式，还需要配置enable密码。

配置命令： Router# telnet <ip address>

Router# telnet <hostname> // 要先配置IP host表，见 11.<选做实验>

启动telnet：

Router# config t

Router(config)# line vty 0 4 // 同时允许0-4共5个连接

Router(config-line)# login //登录

Router(config-line)# password cisco // 设置登录密码为cisco

设置enable密码：

Router(config)#enable password cisco

Router(config-line)#password cisco

# 第三章 路由器基本命令

## 3.1实验前准备

本次实验只涉及基础命令，暂无实验前准备

## 3.2实验要求

本次实验,主要完成以下几个基本命令的操作:

1.设置路由器系统时间：

系统时间是一个非常重要的参数，设备在运行过程中产生的每个日志信息都会有产生的时间作为参考，如果系统时间设置不正确，对于判断网络设备在某个时刻的状态是非常不利的，因此，设备在加电运行的时候,都会设置一个特定的时间,便于随时掌握设备的运行情况。

2.启动光标跟随服务：

网络管理员在对设备进行配置的时候，设备会不断的弹出控制台信息，告诉网络管理员设备的运行状态，但频繁的控制台信息会打断网络管理员正在输入的命令，给配置带来很大的不便，因此，可以打开光标跟随的功能。这样，即使弹出控制台信息,命令也不会被打断，该服务默认是关闭的。

3.设置路由器登陆界面：

对于一些商业机构，或者公众网络来说，部分网络设备必须暴露在互联网上，这样一来，除了合法的网络管理员，任何能接入互联网的用户都可以登陆到设备上来，因此，必须设置一个登陆界面，告知用户设备的归属方，该设备所起到的作用以及非法用户登陆所应该承担的法律责任，切忌在登陆界面上出现欢迎字样(内网设备除外)，以防止给入侵的黑客找到入侵的借口。

4.配置端口描述：

网络管理员在部署大规模的复杂网络时必须规划清楚各个端口连到对端的什么设备，什么端口，以及端口的作用。所以，在初始化配置的时候,端口描述成为必不可少的配置，尤其对于运营商来说,端口描述已经成为配置规范中不可缺少的一部分。

## 3.3实验拓扑

一台PC通过Console线接入设备，本次实验的所有配置都在这样的拓扑下完成。

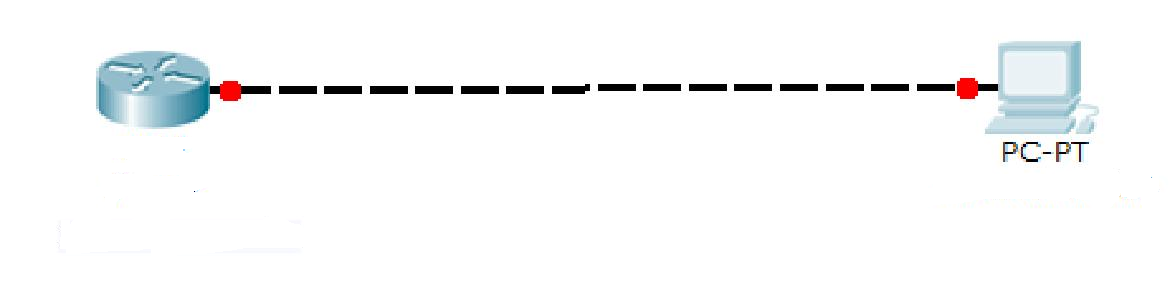


图3.1 拓扑图

## 3.4实验过程

1. 设置路由器时间，将设备的系统时间设置为2016年1月1日8点整

Router#clock set 08:00:00 1 jan 2016

1. 启动光标跟随功能

Router(config)#line con 0

Router(config-line)#logging synchronous

1. 设置路由器登陆界面

R1(config)#banner motd “Welcome to NJU”

测试结果如图3.2所示。

|  |
| --- |
| Welcome to NJU  User Access Verification  Password:  Router> |

图3.2 设置路由器登陆界面

1. 配置端口描述

R1(config)#int fa 0/0

R1(config-if)#description To ISP

1. 关闭思科设备的域名解析功能

对于思科的设备，如果在特权模式下，网络管理员不小心输入了错误的命令，那么思科设备会认为这条错误的命令是一个域名，它会做域名解析，如图3.3所示：

|  |
| --- |
| Router#fsdafasdf  Translation "fsdafasdf"...domain server (255.255.255.255)  (255.255.255.255)  Translating "fsdafasdf"...domain server (255.255.255.255)  Translating "fsdafasdf"...domain server (255.255.255.255)  % Unknown command or computer name, or unable to find computer address  #Router# |

图3.3 域名解析

在这个情况下，设备会卡在这里一段时间，这里千万不要按回车键，多按一次回车，域名就多解析一次。这里正确的做法是按 Ctrl+Shift+6，打断设备的域名解析，等设备退回到正常的情况下后，再输入下面的命令，关闭设备的域名解析。

R1(config)#no ip domain-lookup

测试结果如图3.4所示。

|  |
| --- |
| Router#configure terminal  Enter configuration commands, one per line. End with CNTL/Z  Router(config)#no ip domain-lookup  Router(config)#fsdafkjs  % Invalid input detected at ‘^’ marker |

图3.4 关闭域名解析

1. 将实验端口恢复到默认设置

对端口进行错误的设置之后，需要将其恢复为默认设置，这步操作会清空所有端口下做的配置。所以，在实际工作中，将端口恢复默认设置是一个风险操作，一定要小心谨慎。

|  |
| --- |
| Router(config)#default interface f0/0  Building configuration...  Interface FastEthernet0/0 set to default configuration |

图3.5 恢复默认设置

## 3.5实验命令列表

表3.1 实验命令表

|  |  |
| --- | --- |
| 设置系统时间 | clock set 时间 日期 月份 年份 |
| 设置登陆界面 | banner motd 欢迎语 |
| 配置端口描述 | description 描述信息 |
| 关闭域名解析 | no ip domain-lookup |
| 端口恢复默认设置 | default interface 端口 |

## 3.6实验问题

# 第四章 路由器密码恢复

## 4.1实验前准备

通常为了提高设备的安全性，都在设备里添加了密码。如果在工作交接时，没有明确交代每一台设备所设置的密码，就会造成无法进行管理的尴尬局面。这时就需要进行密码恢复的操作，但同时不对密码之外的配置造成影响。

## 4.2实验要求

本次实验,主要完成以下操作：

1. 设置设备名和密码并保存。实验环境中的设备都是无配置的，为了达到实验的要求，需要先自行设置密码。设置设备名的作用是便于辨认设备是否加载了配置。
2. 重启设备。路由器只有在特权模式进行命令reload来重启设备，和关闭电源两种方式重启设备。密码恢复需要在重启设备的时候进行操作。
3. 进ROMMON模式。有的电脑需要按住ctrl+break键在路由器刚开始重启的时候进入ROMMON模式，有的电脑需要按住ctrl+fn+break来进入ROMMON模式。
4. 修改寄存器值并且保存重启。路由器默认寄存器值为 0x2102，表示含义为路由器在启动过程中加载保存在nvram里的startup-config。密码保存在startup-config 里面，为了恢复密码必须跳过加载startup-config 的步骤，所以要将寄存器值更改为 0x2142，表示路由器在启动过程中不加载保存在nvram里的startup-config 文件。
5. 进入特权模式加载startup-config，删除密码。在ROMMON模式重启后，再进入配置模式时面对的是没有任何配置的路由器状态，因为没有加载 startup-config。从用户模式进入特权模式无需输入密码，在特权模式加载startup-config，此时的路由器拥有了配置，里面包含了之前配置的密码。进入全局配置模式删除密码。
6. 更改寄存器值并保存。此时寄存器值为 0x2142，在设备下次重启的时候还是不会读取startup-config，需要将寄存器值更改回0x2102，并且保存。

## 4.3实验拓扑

实验拓扑如图4.1所示：

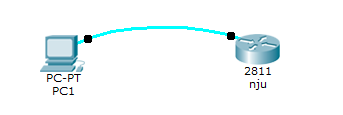


图4.1 实验拓扑

## 4.4实验过程

1. 将路由器用户名更改为nju，将密码更改为ccna

Router>enable

Router#configure terminal

Router(config)#hostname nju

nju(config)#enable password ccna

nju(config)#exit

nju#copy running-config startup-config

1. 重启设备

nju#reload

Proceed with reload? [confirm]

在实际工作中，接触一台设备时首先进入的是用户模式，因为不知道密码所以无法进入特权模式，这时需要关闭路由器的开关，再打开以达到重启的目的。

1. 进入ROMMON模式

在刚刚重启的时候，按住ctrl+break键进入 ROMMON模式，有的电脑需要按住ctrl+fn+break来进入ROMMON模式。

rommon 1 > confreg 0x2142

rommon 2 > reset

注：此为28系列路由器的设置方式

1. 进入特权模式加载 startup-config

等重新启动完成之后，面对的是没有配置的路由器。进入特权模式不需要输入密码，加载 startup-config。

Router>enable

Router#copy startup-config running-config

Destination filename [running-config]?

1. 删除密码并修改寄存器值

nju#config terminal

nju(config)#no enable password

nju(config)#config-register 0x2102

此时配置如图4.2所示：

|  |
| --- |
| Current configuration : 960 bytes  !  version 12.4  service timestamps debug datetime msec  Service timestamps log datetime msec  no service password-encryption  !  Hostname nju  !  boot-start-marker  boot-end-marker  ! |

图4.2 路由器配置

再检查寄存器值：

|  |
| --- |
| Cisco 2811 (revision 53.50) with 512000k/12288k bytes of memory.  Processor board ID FHK1408F1QE  2 FastEthernet interfaces  2 Low-speed serial(sync/async) interfaces  1 Virtual Private Network (VPN) Module  DRAM configuration is 64 bits wide with parity enabled.  239K bytes of non-volatile configuration memory.  126976K bytes of ATA CompactFlash (Read/Write)  Configuration register is 0x2142 (will be 0x2102 at next reload) |

图4.3 检查寄存器

此时寄存器值为0x2142,在重启后会变成0x2102。

注意：此时需要将当前运行配置拷贝为启动配置，因为此时启动配置仍然存在密码。

## 4.5实验命令列表

表4.1 实验命令列表

|  |  |
| --- | --- |
| 更改路由器名称 | hostname [name] |
| 更改路由器密码 | enable password [password] |
| 重启 | reload |
| 修改寄存器值 | confreg 0x2142 |
| 将配置拷贝到NVRAM，在下次启动时使用该配置 | copy startup-config running-config |

## 4.6实验问题

# 第五章 路由器IOS备份

## 5.1实验前准备

1. 交换机根据收到数据帧中的源MAC地址建立该地址同交换机端口的映射，并将其写入MAC地址表中。
2. 交换机将数据帧中的目的MAC地址同已建立的MAC地址表进行比较，以决定由哪个端口进行转发。
3. 如数据帧中的目的MAC地址不在MAC地址表中，则向所有端口转发。这一过程称为泛洪（flood）。

4.广播帧和组播帧会被转发到所有端口。

## 5.2实验要求

本次实验主要完成以下几个基本命令的操作:

1. 配置PC的IP地址为192.168.1.1/24
2. 配置路由器的以太网口的IP地址为192.168.1.254/24
3. 在PC上开启TFTP SERVER
4. 把路由器的IOS导入到PC上

## 5.3实验拓扑

拓扑如图所示：

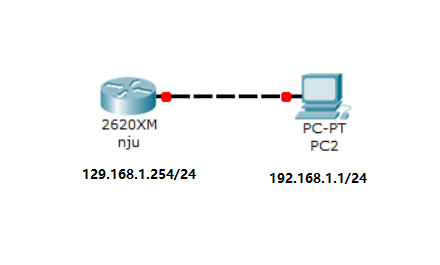


图5.1 拓扑图

## 5.4实验过程

nju上的配置

|  |
| --- |
| nju(config)#interface fastEthernet 0/1  nju(config-if)#ip address 192.168.1.254 255.255.255.0  nju(config-if)#no shutdown |

图5.2 配置路由器IP

配置PC端的IP地址为192.168.1.1，子网掩码为255.255.255.0。

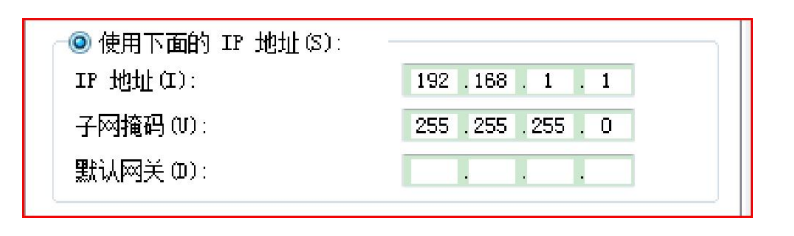


图5.3 配置电脑IP

用路由器ping主机，验证3层连通性。注意，此时要将主机的防火墙关闭，否则ping不通主机。

|  |
| --- |
| nju#ping 192.168.1.1  Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2 second:  !!!!!  Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms |

图5.4 Ping主机

在PC端开启TFTP软件，在桌面上点击以下图标。



修改upload/download的目录为D:\IOS。

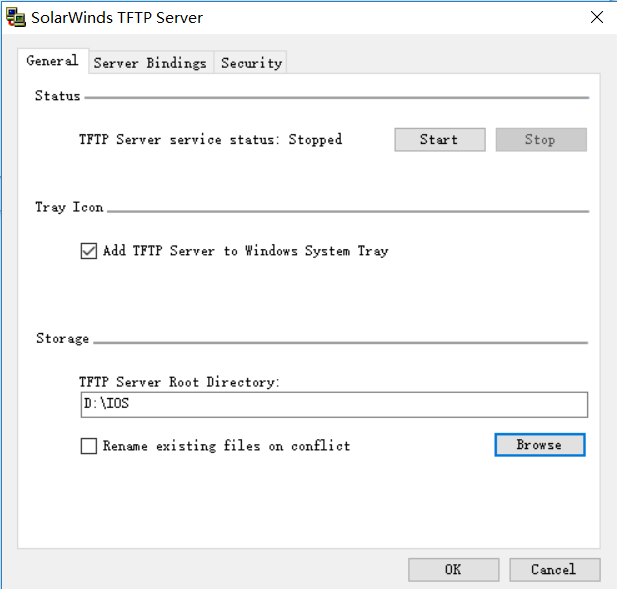


图5.5 设置TFTP保存路径

查看路由器的IOS名称。M7.bin 此文件就是路由器的IOS名称

|  |
| --- |
| nju#show flash:  -#- --length-- -----date/time------ path  1 58136556 Feb 22 1907 17:31:44 c2800nm.bin  70598656 byte available (58138624 bytes used) |

图5.6 查看IOS名称

使用下图命令来备份IOS。

|  |
| --- |
| nju#copy flash:c2800nm.bin tftp:  Address or name of remote host []? Ping 192.168.1.1  ?Invalid host address or name  %Error parsing filename (Invalid IP address or hostname)  nju#copy flash:c2800nm.bin tftp:  Address or name of remote host []? 192.168.1.1  Destination filename [c2800nm.bin]?  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!  58136556 bytes copied in 215.252 secs (270086 bytes/sec) |

图5.7

TFTP Server显示正在备份的过程。约过10分钟后，IOS备份成功，在PC的D盘的IOS文件夹是否存在备份好的IOS文件。

## 5.5实验命令列表

表5.1 实验命令列表

|  |  |
| --- | --- |
| 拷贝IOS | copy flash:[IOS名称] |

## 5.6实验问题

# 第六章 交换机基本命令

## 6.1实验前准备

交换机根据收到数据帧中的源MAC地址建立该地址同交换机端口的映射，并将其写入MAC地址表中。交换机将数据帧中的目的MAC地址同已建立的MAC地址表进行比较，以决定由哪个端口进行转发。如数据帧中的目的MAC地址不在MAC地址表中，则向所有端口转发。这一过程称为泛洪（flood）。广播帧和组播帧向所有的端口转发。

## 6.2实验要求

本次实验主要完成以下几个基本命令的操作:

将一台交换机的hostname 改成nju。

1 将交换机的特权密码设置为ccna。网管人员连接进入网络设备之后，首先进入的是用户模式，在这个模式下，能使用的命令很少，也无法对网络设备进行配置操作，因此，需要在用户模式下，输入enable 命令，进入特权模式，在这步操作时，可以设置密码，验证用户身份。增加设备的安全性。

2将交换机的vty 线路密码设置为ccnp。大多数情况下，网络设备并不在网络管理人员可以接触的地方，因此，有时需要远程登陆到网络设备上进行操作，远程登陆使用的是VTY 线路，因此，对VTY 线路设置密码，使得网络管理人员在远程登陆网络设备时需要被验证身份。增加设备的安全性。

3给交换机设置管理IP 地址和网关。路由器属于三层设备，可以通过接口设置IP 地址，进行远程登录管理设备。交换机需要通过设置管理IP 地址，使得网络管理人员通过这个地址远程登录管理交换机。

4给交换机静态绑定MAC 地址。交换机在转发数据帧时，通过查找MAC 地址表进行转发，通过静态绑定MAC 地址，减少交换机的泛洪的反应时间。

## 6.3实验拓扑

拓扑如图所示：

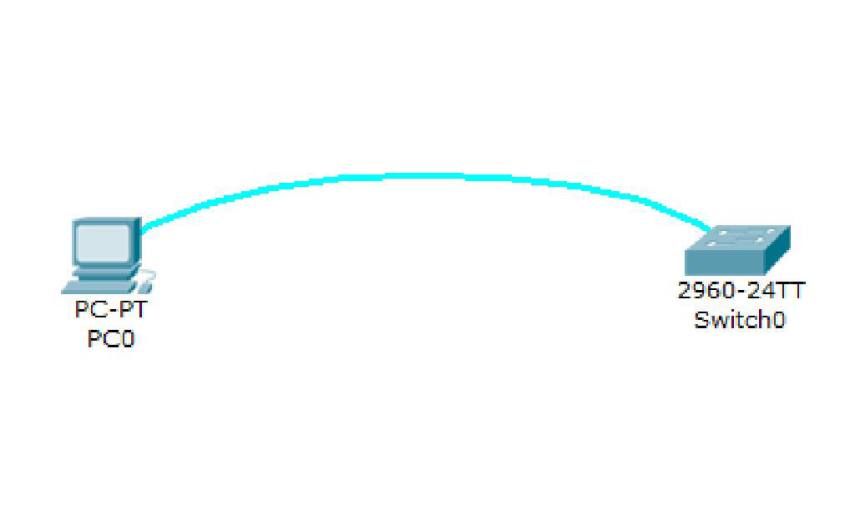


图6.1 交换机配置拓扑图

## 6.4实验过程

1将hostname改为nju

|  |
| --- |
| Switch>enable  Switch#conf t  Enter configuration commands, one per line. End with CNTL/Z.  Switch(config)#hostname nju  nju(config)# |

图6.2 修改hostname命令及结果图

2设置特权密码和vty线路密码

|  |
| --- |
| nju(config)#enable password ccna  nju(config)#line vty 0 4  nju(config-line)#password ccnp |

图6.3 设置密码及结果图

3设置管理ip地址

|  |
| --- |
| nju(config)#inter vlan1  nju(config-if)#ip add 192.168.1.1 255.255.255.0  nju(config-if)#no shut  nju(config-if)#exit  nju(config)#ip default-gateway 192.168.1.100 |

图6.4 设置ip地址及结果图

4验证实验

vlan1 默认关闭，需要手动打开，设置了管理ip 地址后，就可以通过远程登录来管理这台IP地址了，将PC 机的IP 地址设置为192.168.1.2，然后与交换机的fa0/1 相连，并打开“开始”菜单——“运行”——“cmd”,接着按照下面给出的DOS 所显示的信息来验实验。

telnet 登陆后，分别输入相应的vty 密码和特权密码，即可管理交换机。

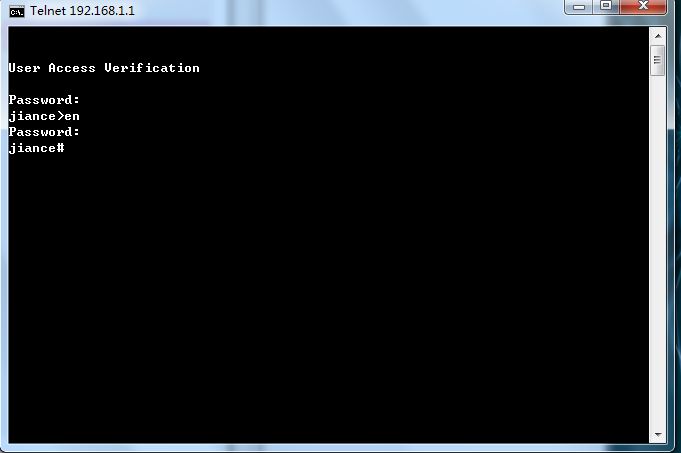


图6.5 cmd运行图

## 6.5实验命令列表

表6.1 实验命令列表

|  |  |
| --- | --- |
| 进入特权模式 | Enable |
| 配置主机名 | hostname [hostname] |
| 设置登陆台密码 | password [password] |
| 配置IP地址 | ip address [address] |
| 配置交换机网关 | switch(config)#ip default-gateway [address] |

## 6.6实验问题

# 第七章 交换机端口安全

## 7.1实验前准备

现实生活中，交换机的使用数量远远多于路由器的使用，交换机因为接口数目多，可以连接多个节点，为了保护交换机的安全性，实行了交换机的端口安全，将MAC地址进行绑定，提高安全性。

## 7.2实验要求

本次实验主要完成以下几项操作:

1. 启用端口安全措施

必须先开启端口安全功能，才能开始制定端口安全策略。

1. 限制fa0/23口最大允许访问量为1

通过限制访问量来保护设备安全。

1. 采用的安全措施为保护，限制或关闭

端口安全侦测到问题使用三种惩罚措施。

## 7.3实验拓扑

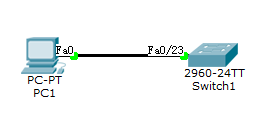


图7.1

## 7.4实验过程

1.惩罚措施为关闭

nju(config)#interface fastEthernet 0/23

nju(config-if)#switchport mode access

nju(config-if)#switchport port-security

nju(config-if)#switchport port-security mac-address aaaa.aaaa.aaaa

nju(config-if)#switchport port-security maximum 1

nju(config-if)#switchport port-security violation shutdown

注：惩罚措施有保护、限制和关闭。关闭：当新的计算机接入时，如果该接口的MAC地址条目超过了最大数目，则该接口将会被关闭，则这个新的计算机和原来的计算机都无法接入。

试验检测：用一根直通线将pc和交换机的0/23口相连，查看0/23接口的指示灯的变化情况。如果有橙色经过大约50秒的时间变为绿色再关闭，说明试验成功。

nju#:sh interface fastEthernet 0/23

在交换机上显示如图：

|  |
| --- |
| FastEthernet 0/23 isdown,line protocol is down(err-disabled)  Hardware is Fast Ethernet,address is ec44.767a.d519(bia ec44.767a.d519) |

图 7.2 端口被强制关闭

2.其他端口测试另外两种惩罚措施的现象

nju(config)#interface fastEthernet 0/22

nju(config-if)#switchport mode access

nju(config-if)#switchport port-security

nju(config-if)#switchport port-security mac-address aaaa.aaaa.aaab

nju(config-if)#switchport port-security maximum 1

nju(config-if)#switchport port-security violation protect

当新的计算机接入时，如果该接口的 MAC 地址条目超过了最大数目，则该端口将允许已知MAC地址发送的数据流但将抛弃未知MAC地址发送的数据流；

试验检测：用一根直通线将 pc 和交换机的 0/23 口相连，使用cmd发送ping命令发送报文至交换机端口ip：192.168.1.1，若mac地址不是之前设置的aaaa.aaaa.aaab，则无法成功。结果如图：

|  |
| --- |
| C：\Users\Administrator>ping 192.168.1.1  正在Ping 192.168.1.1具有32字节的数据：  请求超时。  请求超时。  请求超时。  请求超时。  192.168.1.1的Ping统计信息：  数据包：已发送=4，已接收=0，丢失=4（100%丢失） |

图 7.3 ping被限制导致失败

当新的计算机接入时，如果该接口的 MAC 地址条目超过了最大数目，则该端口将允许已知MAC地址发送的数据流但将抛弃未知MAC地址发送的数据流，但同时会发送一条讯息通知违规发生，大致过程与保护类似，不再赘述。

## 7.5实验命令列表

表7.1

|  |  |
| --- | --- |
| 选择交换机端口 | interface fastEthernet [端口号] |
| 将端口定义为主机端口 | switchport mode access |
| 开启交换机端口安全功能 | switchport port-security |
| 绑定mac地址到端口上 | switchport port-security mac-address [地址] |
| 设置安全访问的最大用户数 | switchport port-security maximum [数目] |
| 设置端口安全惩罚措施 | switchport port-security violation [措施] |

## 7.6实验问题

# 第八章 静态路由和简单组网

## 8.1实验前准备

本次实验需要准备三台路由器，并且熟悉路由器的相关配置命令。

## 8.2实验要求

本次实验主要完成以下几项操作:

1. 在Cisco26XX系列路由器上进行静态路由配置，通过使用静态路由将三台Cisco26XX路由器连接起来,组成一个小网络；

2. 练习在简单网络中查看网络和设备状态的各种指令。

## 8.3实验拓扑

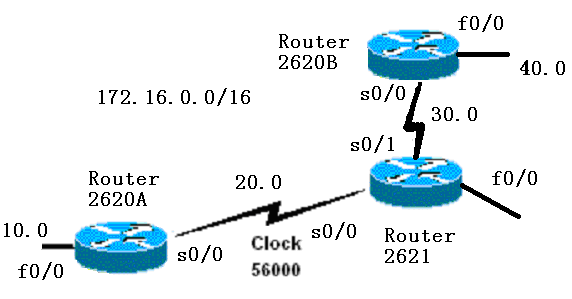


图8.1

## 8.4实验过程(需要区分DTE/DCE)

1.端口IP地址配置

配置Router2620A: （f0/0:192.168.10.1,s0/0:192.168.20.1）

Router>enable

Router#config terminal

Router#hostname Router2620A

Router2620A(config)#int f0/0

Router2620A(config-if)#ip address 192.168.10.1 255.255.255.0

Router2620A(config-if)#no shut

Router2620A(config-if)#int s0/0

Router2620A(config-if)#ip address 192.168.20.1 255.255.255.0

Router2620A(config-if)#no shut

配置Router2621: （s0/0:192.168.20.2,s0/1:192.168.30.1）

Router>enable

Router#config terminal

Router#hostname Router2621

Router2621(config)#int s0/1

Router2621(config-if)#ip address 192.168.30.1 255.255.255.0

Router2621(config-if)#no shut

Router2621(config-if)#int s0/0

Router2621(config-if)#ip address 192.168.20.2 255.255.255.0

Router2621(config-if)#clock rate 56000

Router2621(config-if)#no shut

配置Router2620B: （s0/0:192.168.30.2,f0/0:192.168.40.1）

Router>enable

Router#config terminal

Router#hostname Router2620B

Router2620B(config)#int s0/0

Router2620B(config-if)#ip address 192.168.30.2 255.255.255.0

Router2620B(config-if)#no shut

Router2620B(config-if)#int f0/0

Router2620B(config-if)#ip address 192.168.40.1 255.255.255.0

Router2620B(config-if)#no shut

用Ping命令测试各网段的连通性

2.路由表配置

格式: ip route <目标网段> <子网掩码> <下一跳路由器地址(IP地址)>

例如：

Router2620A(config)#ip route 192.168.40.0 255.255.255.0 192.168.20.2

将路由表配置完备后，用ping命令检查各个端口间是否已顺利接通

3. 配置默认路由

对于该实验的拓扑结构来说，只有Router1 和Router3 允许配置默认路由。

首先应该删除静态路由的配置，才配置默认路由。

以Router2620A为例：

Router2620A(config)# no ip route 192.168.40.0 255.255.255.0 192.168.20.2

Router2620A(config)# ip route 0.0.0.0 0.0.0.0 192.168.20.2

查看路由表 （命令：Router# show ip route）

注：有\*号表示默认路由

## 8.5实验命令列表

表8.1

|  |  |
| --- | --- |
| 路由表配置 | ip route [目标网段] [子网掩码] [下一跳路由器地址(IP地址)] |
| 删除静态路由的配置 | no ip route 192.168.40.0 255.255.255.0 192.168.20.2 |
| 配置默认路由 | ip route 0.0.0.0 0.0.0.0 192.168.20.2 |
| 查看路由表 | show ip route |
| 停止查看路由表 | no debug all |

## 8.6实验问题

假如只分配了一个网段：192.168.10.0/24，你该如何搭建上述拓扑？请设计并加以实现。

# 第九章 动态RIP

## 9.1实验前准备

本次实验需要准备三台路由器，并且熟悉路由器的相关配置命令。

## 9.2实验要求

本次实验主要完成以下操作:

1. 在Cisco26XX系列路由器上，通过使用动态RIP路由协议将三台Cisco26XX路由器组成一个小网络；

## 9.3实验拓扑

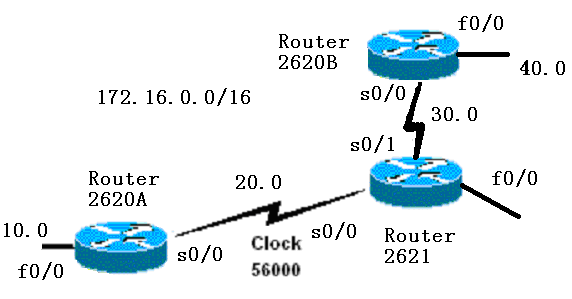


图9.1

## 9.4实验过程

1.使用show ip route查看路由表

检查现在能否从Router2620A 上ping 192.168.40.1。

2.配置动态路由

Router2620A:

Router2620A(config)#router rip

Router2620A(config)#network 192.168.10.0

Router2620A(config)#network 192.168.20.0

Router2621:

Router2621(config)# router rip

Router2621(config)#network 192.168.20.0

Router2621(config)#network 192.168.30.0

Router2620B:

Router2620B(config)#router rip

Router2620B(config)#network 192.168.30.0

Router2620B(config)#network 192.168.40.0

3.使用show ip route查看路由表

检查现在能否从Router2620B 上ping 192.168.40.1。

4.使用如下指令查看路由表更新(每30秒更新一次)

debug ip rip//开始查看

no debug all//停止查看

## 9.5实验命令列表

表9.1

|  |  |
| --- | --- |
| 路由表配置 | ip route [目标网段] [子网掩码] [下一跳路由器地址(IP地址)] |
| 查看路由表 | show ip route |
| 查看路由表更新 | debug ip rip |
| 停止查看路由表 | no debug all |

## 9.6实验问题

1. 在配置结束后用什么命令来查看具体的设置,请显示具体内容。
2. 在路由器的全局模式下用“show ip protocol”检查当前时间参数设置，所显示的时间值分别代表什么？
3. 观察网络路由路径的选择
4. 在路由器的全局模式下，“traceroute”命令可用来追踪数据包在网络上所经过的路由。可选择若干条有代表性的路径进行路由选择的跟踪，并将由源到目标的各路径的结果记录下来。下表可作为参考格式：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 路径编号 | 源IP | 中间节点1 | 中间节点2 | 中间节点3 | 中间节点4 | 目的IP |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# 第十章 配置单域OSPF

## 10.1实验前准备

OSPF是一种典型的链路状态路由协议。采用OSPF的路由器彼此交换并保存整个网络的链路信息，从而掌握全网的拓扑结构，独立计算路由。因为RIP路由协议不能服务于大型网络，所以，IETF的IGP工作组特别开发出链路状态协议——OSPF。目前广为使用的是OSPF第二版，最新标准为RFC2328。

OSPF作为一种内部网关协议，用于在同一个自治域（AS）中的路由器之间发布路由信息。区别于距离矢量协议（RIP），OSPF具有支持大型网络、路由收敛快、占用网络资源少等优点，在目前应用的路由协议中占有相当重要的地位。

OSPF路由器收集其所在网络区域上各路由器的连接状态信息，即链路状态信息（Link-State），生成链路状态数据库（Link-State Database）。路由器掌握了该区域上所有路由器的链路状态信息，也就等于了解了整个网络的拓扑状况。OSPF路由器利用“最短路径优先算法（ShortestPath First，SPF）”，独立地计算出到达任意目的地的路由。同时，OSPF协议还引入了“分层路由”的概念，将网络分割成一个“主干”连接的一组相互独立的部分，这些相互独立的部分被称为“区域”（Area），“主干”的部分称为“主干区域”。每个区域就如同一个独立的网络，该区域的OSPF路由器只保存该区域的链路状态。每个路由器的链路状态数据库都可以保持合理的大小，路由计算的时间、报文数量都不会过大。

## 10.2实验要求

本次实验主要完成以下几个基本命令的操作：

1根据拓扑组建和配置网络。配置好网络后，先不要配置OSPF，先用“ping”命令来核验工作，并测试以太网接口之间的连通性。

2为每台路由器配置一个环回接口。将环回接口（而不是物理接口）的地址用作路由器ID时，OSPF将更稳定，因为不同于物理接口，这种接口总是处于活动状态，为不会出现故障，因此再所有重要路由上，都应使用环回接口。

3配置OSPF。可结合使用命令router ospf和network area命令。

4查看OSPF运行情况。使用show ip protocols命令显示IP路由协议参数，包括定时器、过滤器、度量值、网络及路由器的其他信息。使用show ip ospf interface命令查看接口是否被加入到正确的区域中；该命令还显示各种定时器和邻接关系。

5调节OSPF的计时器。调节OSPF的计时器，以使这些核心路由器能更快地检测出失效的

情况，但这会导致额外的数据流量增加。

6设置OSPF认证。使用接口配置命令ip ospf message-digest-key key-id md5 key给采用OSPF

MD5身份验证的路由器指定要使用的密钥ID和密钥。

## 10.3实验拓扑

拓扑如图所示，此外需要为各台路由器配置环回地址RTA：10.0.0.1/32，RTB：10.0.0.2/32，RTC:10.0.0.3/32。以此为基础配置单区域的OSPF网络，即Area 0里OSPF的配置。

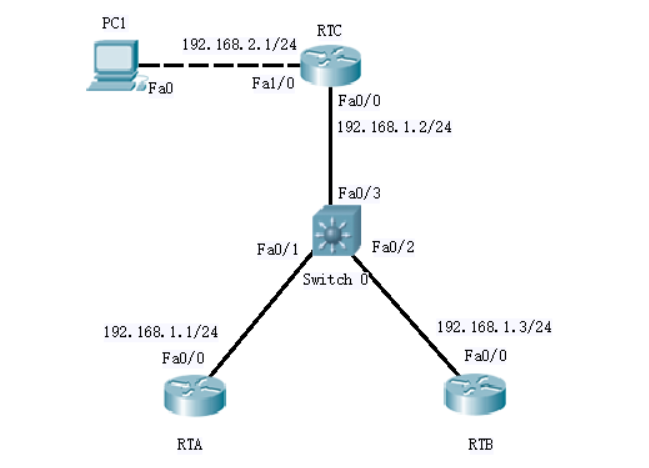


图10.1 OSPF配置实验拓扑图

## 10.4实验过程

1请按照前面几次实验练习的配置方法，根据给出的图示组建和配置网络

2在每台路由器上，用一个唯一的IP地址配置一个环回接口

RTA(config)#interface lo0

RTA(config-if)#ip address 10.0.0.1 255.255.255.255

RTB(config)#interface lo0

RTB(config-if)#ip address 10.0.0.2 255.255.255.255

RTC(config)#interface lo0

RTC(config-if)#ip address 10.0.0.3 255.255.255.255

3在配置了环回接口之后，可以开始配置OSPF了：

RTA(config)#router ospf 1

RTA(config-router)#network 192.168.1.0 0.0.0.255 area 0

RTB(config)#router ospf 1

RTB(config-router)#network 192.168.1.0 0.0.0.255 area 0

RTC(config)#router ospf 1

RTC(config-router)#network 192.168.1.0 0.0.0.255 area 0

RTC(config-router)#network 192.168.2.0 0.0.0.255 area 0

4用“show”命令来核验它的操作运行。

RTC#show ip protocols

注意，更新计时器被设置为0。路由更新不是在固定时间间隔上被发送的，它们是事件驱动的。下一步，用“show ip ospf”命令来获得有关OSPF进程的消息信息。

|  |
| --- |
| Routing protocol is "ospf 1"  Outgoing update filter list for all interface is not set  Incoming update filter list for all interface is not set  Router ID 10.0.0.3  Number of areas in this router is 1.1 normal 0 stub 0 nssa  Maximum path: 4  Routing for Networks:  192.168.1.0 .0.0.0.255 area 0  192.168.2.0 .0.0.0.255 area 0  Reference bandwidth unit is 100 mbps  Routing Information Sources:  Gateway Distance Last Update  Distance: (default is 110) |

图10.2 OSPF进程消息信息

查看DR/BDR：

|  |
| --- |
| RTB#show ip ospf interface  FastEthernet0/0 is uop, line protocol is up  Internet Address 192.168.1.3/24,Area 0  Process ID 1, Router ID 10.0.0.2, Network Type BROADCAST, Cost: 1  Transmit Delay is 1 sec, State BDR, Priority 1  Designated Router (ID) 10.0.0.2, Interface address 192.168.1.2  Backup Designated router (ID) 10.0.0.2, Interface address 192.168.1.3  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5  oob-resync timeout 40  Hello due in 00:00:02  Supports Link-local Signaling (LLS)  Cisco NSF helper support enabled  IETF NSF helper support enabled  Index1/1, flood queue length 0  Next 0x0(0)/0x0(0)  Last flood scan length is 0, maximum is 1  Last flood scan time is 0 ,sec, maximum is 0 msec  Neighbor Count is 2,Adjust neighbor count is 2  Adjust with neighbor 10.0.0.1  Adjust with neighbor 10.0.0.3 (Designated Router) |

图10.3 DR/BDR信息

可以看到DR为RTC，BDR为RTB

RTA(config)#interface f0/0

RTA(config-if)#ip ospf hello-interval 5

5调节OSPF的计时器

RTA(config-if)#ip ospf dead-interval 20

6设置OSPF认证

RTA(config-if)#ip ospf message-digest-key 1 md5 7 itsasecret

RTA(config-if)#router ospf 1

RTA(config-router)#area 0 authentication message-digest

## 10.5实验命令列表

表10.1 实验命令列表

|  |  |
| --- | --- |
| 全局配置命令 | router ospf [router-id] |
| 接口配置命令 | network [ipaddress] [wildcard-mask] area [area-id] |
| 显示ip路由协议参数 | show ip protocols |
| 显示接口的ospf状态 | show ip ospf interface |
| 修改hello间隔 | ip ospf hello-interval [time] |
| 修改dead时间 | ip ospf dead-interval [time] |
| 配置ospf MD5身份 | ip ospf message-digest-key [key-id] md5 [key] |

## 10.6实验问题

哪个路由器成为了DR？哪个路由器成为了BDR?为什么？

第十一章 VLAN间路由

11.1实验前准备

目前，很多中小型企业内部网络都是通过交换机互联而成，为了实现广播域的分割和广播包传播范围的控制，划分Vlan已成为网络架构中不可缺少的操作，通过划分Vlan，可以使得同一台交换机下的不同Vlan里的端口下连接的设备不能直接互相访问，这样有效的隔离了网络。虽然划分Vlan有效的地控制了广播包的传播范围，但是对于某些既希望隔离网络，也希望有些不同的Vlan能够通信的企业来讲，Vlan间路由就成为必要的技术，常常在中小型企业网中部署。为了完成Vlan间路由的实验，必须事先掌握Vlan的划分，VTP同步，把接口划分进相应的Vlan等交换机的基本操作，以及一个相对比较新的概念——子接口。

11.2实验要求

首先需要明确一点，不同的Vlan相互隔离广播域，因此，传统的以太网ARP 方式的通信机制在这里是不可用的，需要在网络中添加三层设备，这里的三层设备可以是路由器，也可以是Cisco三层交换机（例如Cisco 3550，Cisco3560）。

本次实验的目的，是让处于不同Vlan下的主机能够通信，因此用路由器充当上述的三层设备，需要用到的知识点有：

1. Vlan的划分
2. VTP同步
3. 将接口划分进Vlan
4. Trunk链路的封装类型
5. 子接口的配置

11.3实验拓扑

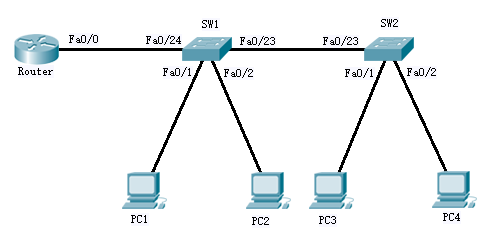


图11.1 实验拓扑

PC1和PC3属于Vlan 10，PC2和PC4属于Vlan 20，如果上述知识点能够配置正常，期望的现象应该是PC1和PC3能够 ping 通PC2和PC4，同样，PC2和PC4也能够 ping 通PC1和PC3。

11.4实验过程

首先，将SW1和SW2之间的链路设置为Trunk链路。

|  |
| --- |
| sw1(config)#interface fa 0/23  sw1(config-if)#switchport trunk encapsulation dot1q  sw1(config-if)#switchport mode trunk  sw2(config)#interface fa 0/23  sw2(config-if)#switchport trunk encapsulation dot1q  sw2(config-if)#switchport mode trunk |

图11.2 设置Trunk链路

划分两个Vlan，Vlan 10和Vlan 20

|  |
| --- |
| sw1(config)#vlan 10  sw2(config)#vlan 20 |

图11.3划分两个Vlan

分别将SW1和SW2的fa0/1口划分入 Vlan 10，fa0/2 口划分入 Vlan 20。

|  |
| --- |
| sw1(config)#interface fa 0/1  sw1(config-if)#switchport mode access  sw1(config-if)#swichport access vlan 10  sw1(config-if)#exit  sw1(config)#interface fa 0/2  sw1(config-if)#switchport mode access  sw1(config-if)#switchport access vlan 20 |

图11.4 划分接口

sw2上进行同样的操作，操作完成后，在 sw1和sw2上分别使用 show vlan brief 命令，查看对应接口是否在正确的vlan中。

然后将SW1 的fa0/24 接口设置为Trunk接口，与 Router互联。

|  |
| --- |
| sw1(config)#interface fa 0/24  sw1(config)#switchport trunk encapsulation dot1q  sw1(config-if)#switchport mode trunk |

图11.5 将SW1 的fa0/24 接口设置为Trunk接口

Router 的 fa0/0 口需要划分两个子接口，分别对应 Vlan10 和 Vlan20，作为它们的网关。

|  |
| --- |
| Router(config)#interface fa 0/0  Router(config-if)#no ip address  Router(config-if)#no shutdown  Router(config)#int fa 0/0.10  Router(config-if)#encapsulation dot1q 10  Router(config-if)#ip address 192.168.10.1 255.255.255.0  Router(config)#int fa 0/0.20  Router(config-if)#encapsulation dot1q 20  Router(config-if)#ip address 192.168.20.1 255.255.255.0 |

图11.6 划分router的两个子接口

测试：PC1的IP地址为 192.168.10.2，网关为192.168.10.1，PC2的IP地址为 192.168.20.2，网关为192.168.20.1。

配置正确，PC1 能够 ping 通 PC2。

|  |
| --- |
| Reply from 192.168.20.2: bytes=32 time=5ms TTL=127  Reply from 192.168.20.2: bytes=32 time=2ms TTL=127  Reply from 192.168.20.2: bytes=32 time=3ms TTL=127  Reply from 192.168.20.2: bytes=32 time=2ms TTL=127 |

图11.7 PC1 ping通PC2

如果没有看到上述现象，证明 PC1和PC2无法正常通信，请对照检查配置。

11.5实验命令列表

表11.1 实验命令列表

|  |  |
| --- | --- |
| 设置Trunk封装类型 | switchport trunk encapsulation [type] |
| 设置Trunk链路 | switchport mode trunk |
| 划分vlan | vlan [vlan name] |
| 将接口划分入vlan | swichport access vlan [vlan name] |
| 显示vlan简要信息 | show vlan brief |

11.6实验问题

将主机移动至其他 VLAN 上并且尝试 ping命令， 观察 ping 运行的结果。

第十二章 NAT网络地址转换

12.1实验前准备

任何位于内部网络和外部网络之间的设备都可以使用NAT（RFC3022对NAT 进行定义、讲解）。转换的地址不一定必须是私有地址，它可以是任何地址。

1.需要使用地址转换常见的原因：

1. 由于ISP没有分配足够的共有IPv4地址，不得不使用私有地址；
2. 使用了公有地址，但是更换了ISP，新的ISP不再支持这些公有地址；
3. 两家公司进行合并，他们使用了相同的地址空间；
4. 要将同一个IP地址分配给多台机器；

2.NAT 术语：

1. 内部本地地址：分配给位于内部网络主机的IPv4地址。内部本地地址可能不是由网络信息中心（NIC）或者服务提供商分配的IPv4地址。
2. 内部全局地址：由网络信息中心（NIC）或者服务提供商分配的合法IPv4 地址，他对外代表着一个或者多个内部本地IPv4地址。
3. 外部本地地址：外部主机显示给内网的IPv4地址。外部本地地址不一定是合法的地址，它是从可路由地址空间分配到内部网络的地址。
4. 外部全局地址：主机所有者分配给外部网络上某一主机的IPv4地址。外部全局地址从全局课路由地址或者空间中分配。

3.地址转换的类型：

1. 静态NAT：将未注册的IPv4地址映射到注册的IPv4地址（一对一）。在必须从网络外部访问设备时静态 NAT特别有用。
2. 动态NAT：将未注册的IPv4地址与某个注册的IPv4地址组中的注册的IPv4进行映射。
3. 过载NAT：使用不同的端口号将多个未注册IPv4地址映射到单个注册的 IPv4 地址（多对一）。过载也称PAT，是动态NAT的一种形式。

4.NAT 具有以下优势：

1. 不需要重新分配所有需要访问外部网络的主机的地址，从而节约时间和金钱。
2. 通过应用端口级的多路复用节约了地址。
3. 保护网络安全。

12.2实验要求

本次实验，希望通过地址转换，使拓扑图中左边内部网络中的内部本地地址分别通过三种方式转换成外部全局地址并成功的访问右边网络中的R3。

12.3实验拓扑

实验拓扑如图所示，R1和R2之间是192.168.1.0/24网段，R2 和R3之间是200.1.1.0/24网段。

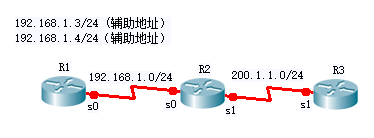


图12.1 实验拓扑

12.4实验过程

1.配置每个设备的名称和接口ip地址，确保彼此之间的三层连通性。

R1(config)#interface s0/0/0

R1(config-if)#ip address 192.168.1.1 255.255.255.0

R1(config-if)#ip address 192.168.1.3 255.255.255.0 secondary

R1(config-if)#ip address 192.168.1.4 255.255.255.0 secondary

R1(config-if)#no shutdown

R2(config)#interface s0/0/0

R2(config-if)#ip address 192.168.1.2 255.255.255.0

R2(config-if)#no shutdown

R2(config)#interface s0/0/1

R2(config-if)#ip address 200.1.1.1 255.255.255.0

R2(config-if)#no shutdown

R3(config)#interface s0/0/1

R3(config-if)#ip address 200.1.1.2 255.255.255.0

R3(config-if)#no shutdown

2.在R2上完成静态NAT的配置。

|  |
| --- |
| R2(config)#ip nat inside source static 192.168.1.1 200.1.1.254  \*Oct 27 09:04:33.599: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to  R2(config)#interface s0/0/0  R2(config-if)#ip nat inside  R2(config)#interface s0/0/1  R2(config-if)#ip nat outside  R2(config-if)#end  \*Oct 27 09:05:32.947: %SYS-5-CONFIG\_I: Configured from console by console  R2#debug ip nat  IP NAT debugging is on  R2#\_ |

图12.2 在R2上配置静态NAT

然后在R1上用本地地址192.168.1.1 Ping 200.1.1.2,结果没有ping通，为什么?

|  |
| --- |
| R1#ping 200.1.1.2  Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  .....  Success rate is 0 percent (0/5)  R1#\_ |

图12.3 在R1上用本地地址 192.168.1.1 Ping 200.1.1.2

查看R1上是否有地址转换的NAT表，转换表为空，说明没有发生地址转换，分析原因， R1去往200.1.1.0 网段，需要一条静态路由。

|  |
| --- |
| R1#show ip nat translations  R1#\_ |

图12.4 查看R1的NAT转换表

为R1加上去往R3的静态路由，现在R1可以ping通R3。

|  |
| --- |
| R1(config)#ip route 200.1.1.0 255.255.255.0 serial 0/0/0  R1(config)#end  R1#ping 200.1.1.2  Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  !!!!!  Success rate is 100 percent (5/5), round-trip min/avg/max = 40/42/44 ms  R1#\_ |

图12.5为R1加上去往R3的静态路由

在R1上使用扩展ping，发送50个数据包，默认情况下为5个数据包。

|  |
| --- |
| R1#ping  Protocol [ip]:  Target IP address: 200.1.1.2  Repeat count [5]: 50  Datagram size [100]:  Timeout in seconds [2]:  Extended commands [n]:  Sweep range of sizes [n]:  Type escape sequence to abort.  Sending 50, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!  Success rate is 100 percent (50/50), round-trip min/avg/max = 40/43/44 ms  R1#\_ |

图12.6在R1上使用扩展ping

快速切换到R2上，来查看具体的转换过程。

|  |
| --- |
| \*Oct 27 09:11:34.791: NAT\*: s=192.168.1.1->200.1.1.254, d=200.1.1.2 [5]  \*Oct 27 09:11:34.819: NAT\*: s=200.1.1.2, d=200.1.1.254->192.168.1.1 [5] |

图12.7 R2终端信息

查看R2的NAT转换表，R2建立NAT表，当有流量符合这个匹配规则时就会两个地址进行转换。

|  |
| --- |
| R2#show ip nat translations  Pro Inside global Inside local Outside local Outside global  --- 200.1.1.254 192.168.1.1 --- ---  R2#\_ |

图12.8查看R2的NAT转换表

3.在R2上完成动态NAT的配置。

将原来的静态NAT的条目删除，通过使用用户访问控制列表来定义本地地址池。

|  |
| --- |
| R2(config)#no ip nat inside source static 192.168.1.1 200.1.1.254  \*Oct 27 09:17:07.491: ipnat\_remove\_static\_cfg: id 1, flag  R2(config)#access-list 1 permit 192.168.1.0 0.0.0.255  R2(config)#ip nat pool nju 200.1.1.253 200.1.1.254 p 24  R2(config)#ip nat inside source list 1 pool nju  R2(config)#  \*Oct 27 09:19:45.703: ipnat\_add\_dynamic\_cfg\_common: id 1, flag 5, range 1  \*Oct 27 09:19:45.707: id 1, flags 0, domain 0, lookup 0, aclnum 1, aclname1,mapn  ame idb 0x00000000  \*Oct 27 09:19:45.707: poolstart 200.1.1.253 poolend 200.1.1.254  R2(config)# |

图12.9 在R2上完成动态NAT的配置

4.用 192.168.1.1 ping 200.1.1.2

|  |
| --- |
| R1#ping  Protocol [ip]:  Target IP address: 200.1.1.2  Repeat count [5]: 50  Datagram size [100]:  Timeout in seconds [2]:  Extended commands [n]:  Sweep range of sizes [n]:  Type escape sequence to abort.  Sending 50, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!  Success rate is 100 percent (50/50), round-trip min/avg/max = 40/43/44 ms  R1#\_ |

图12.10 ping 200.1.1.2

Ping通说明路由添加正确，查看R2的终端信息。

|  |
| --- |
| \*Oct 27 09:20:54.591: NAT\*: s=192.168.1.1->200.1.1.253, d=200.1.1.2 [60]  \*Oct 27 09:20:54.619: NAT\*: s=200.1.1.2, d=200.1.1.253->192.168.1.1 [60] |

图12.11 R2终端信息

5.在R1上用 192.168.1.3 ping 200.1.1.2

|  |
| --- |
| R1#ping  Protocol [ip]:  Target IP address: 200.1.1.2  Repeat count [5]: 20  Datagram size [100]:  Timeout in seconds [2]:  Extended commands [n]: y  Source address or interface: 192.168.1.3  Type of service [0]:  Set DF bit in IP header? [no]:  Validate reply data? [no]:  Data pattern [0xABCD]:  Loose, Strict, Record, Timestamp, Verbose[none]:  Sweep range of sizes [n]:  Type escape sequence to abort.  Sending 20, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  Packet sent with a source address of 192.168.1.3  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!  Success rate is 100 percent (20/20), round-trip min/avg/max = 40/43/44 ms  R1#\_ |

图12.12 ping 200.1.1.2

查看R2的终端信息以及NAT转换表，源地址192.168.1.2转换成200.1.1.254，很明显调用了第2公有地址。

|  |
| --- |
| \*Oct 27 09:26:10.339: NAT\*: s=192.168.1.3->200.1.1.254, d=200.1.1.2 [139]  \*Oct 27 09:26:10.367: NAT\*: s=200.1.1.2, d=200.1.1.254->192.168.1.3 [139] |

图12.13 R2终端信息

|  |
| --- |
| R2#show ip nat translations  Pro Inside global Inside local Outside local Outside global  --- 200.1.1.253 192.168.1.1 --- ---  --- 200.1.1.254 192.168.1.3 --- ---  R2#\_ |

图12.14查看R2的NAT转换表

6.在R1上用 192.168.1.4 ping 200.1.1.2

|  |
| --- |
| R1#ping  Protocol [ip]:  Target IP address: 200.1.1.2  Repeat count [5]: 20  Datagram size [100]:  Timeout in seconds [2]:  Extended commands [n]: y  Source address or interface: 192.168.1.4  Type of service [0]:  Set DF bit in IP header? [no]:  Validate reply data? [no]:  Data pattern [0xABCD]:  Loose, Strict, Record, Timestamp, Verbose[none]:  Sweep range of sizes [n]:  Type escape sequence to abort.  Sending 20, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  Packet sent with a source address of 192.168.1.4  ……………………………...  Success rate is 0 percent (0/20)  R1#\_ |

图12.15 ping 200.1.1.2

结果发现不能ping通到目的。查看R2的NAT转换表，发现没有192.168.1.4的条目。

|  |
| --- |
| R2#show ip nat translations  Pro Inside global Inside local Outside local Outside global  --- 200.1.1.253 192.168.1.1 --- ---  --- 200.1.1.254 192.168.1.3 --- ---  R2#\_ |

图12.16查看R2的NAT转换表

解决的方法：清除R2的NAT表中的条目，将公有地址池中的公有地址释放出来。

|  |
| --- |
| R2#clear ip nat translation \*  R2#show ip nat translations  R2#\_ |

图12.17清除R2的NAT表中的条目

在R1上重试。

|  |
| --- |
| R1#ping  Protocol [ip]:  Target IP address: 200.1.1.2  Repeat count [5]:  Datagram size [100]:  Timeout in seconds [2]:  Extended commands [n]: y  Source address or interface: 192.168.1.4  Type of service [0]:  Set DF bit in IP header? [no]:  Validate reply data? [no]:  Data pattern [0xABCD]:  Loose, Strict, Record, Timestamp, Verbose[none]:  Sweep range of sizes [n]:  Type escape sequence to abort.  Sending 20, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  Packet sent with a source address of 192.168.1.4  !!!!!!!!!!!  Success rate is 100 percent (5/5), round-trip min/avg/max = 44/44/44 ms  R1#\_ |

图12.18在R1上重试

R2终端上所显示的转换过程。

|  |
| --- |
| \*Oct 27 09:37:24.699: NAT\*: s=192.168.1.4->200.1.1.253, d=200.1.1.2 [170]  \*Oct 27 09:37:24.727: NAT\*: s=200.1.1.2, d=200.1.1.253->192.168.1.4 [170] |

图12.19 R2终端信息

再查看R2的NAT转换表。

|  |
| --- |
| R2#show ip nat translations  Pro Inside global Inside local Outside local Outside global  icmp 200.1.1.253:7 192.168.1.4:7 200.1.1.2:7 200.1.1.2:7  --- 200.1.1.253 192.168.1.4 --- ---  R2#\_ |

图12.20 R2的NAT转换表

7.配置 PAT

先删除转换语句，再删除之前建立的 pool，注意删除的顺序。

|  |
| --- |
| R2(config)#no ip nat inside source list 1 pool nju  Dynamic mapping in use, do you want to delete all entires? [no]: yes  R2(config)#no ip nat pool nju 200.1.1.253 200.1.1.254 prefix-length 24  R2(config)#ip nat pool nju 200.1.1.253 200.1.1.253 prefix-length 24  R2(config)#ip nat inside source list 1 pool nju overload  R2(config)#  \*Oct 27 09:42:38.571: ipnat\_add\_dynamic\_cfg\_common: id 2,flag 5, range 1  \*Oct 27 09:42:38.571: id 2, flags 0, domain 0, lookup 0, aclnum 1, aclname 1, map  name idb 0x00000000  \*Oct 27 09:42:38.571: poolstart 200.1.1.253 poolend 200.1.1.253 \_ |

图12.21 配置 PAT

8.在 R1 用 192.168.1.1 上 ping 200.1.1.2

|  |
| --- |
| R1#ping 200.1.1.2  Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  !!!!!  Success rate is 100 percent (5/5), round-trip min/avg/max =44/44/44 ms  R1#\_ |

图12.22 ping 200.1.1.2

查看R2的终端信息以及NAT转换表，随机产生端口号6。

|  |
| --- |
| \*Oct 27 09:44:05.283: NAT\*: s=192.168.1.1->200.1.1.253, d=200.1.1.2 [175]  \*Oct 27 09:44:05.311: NAT\*: s=200.1.1.2, d=200.1.1.253->192.168.1.1 [175] |

图12.23 R2终端信息

|  |
| --- |
| R2#show ip nat translations  Pro Inside global Inside local Outside local Outside global  icmp 200.1.1.253:6 192.168.1.1:6 200.1.1.2:6 200.1.1.2:6  R2#\_ |

图12.24查看R2的NAT转换表

R2约1分钟的时间释放地址转换的空间，此时查找NAT表中没有任何的转换条目。

|  |
| --- |
| R2#show ip nat translations  R2#\_ |

图12.25 查看R2的NAT转换表

9.在 R1用 192.168.1.3 ping 200.1.1.2

|  |
| --- |
| R1#ping  Protocol [ip]:  Target IP address: 200.1.1.2  Repeat count [5]:  Datagram size [100]:  Timeout in seconds [2]:  Extended commands [n]: y  Source address or interface: 192.168.1.3  Type of service [0]:  Set DF bit in IP header? [no]:  Validate reply data? [no]:  Data pattern [0xABCD]:  Loose, Strict, Record, Timestamp, Verbose[none]:  Sweep range of sizes [n]:  Type escape sequence to abort.  Sending 20, 100-byte ICMP Echos to 200.1.1.2, timeout is 2 seconds:  Packet sent with a source address of 192.168.1.3  ……………………………...  Success rate is 0 percent (5/5), round-trip min/avg/max = 40/43/44 ms  R1#\_ |

图12.26 ping 200.1.1.2

查看R2的终端信息。

|  |
| --- |
| \*Oct 27 09:47:40.827: NAT\*: s=192.168.1.1->200.1.1.253, d=200.1.1.2 [180]  \*Oct 27 09:47:40.855: NAT\*: s=200.1.1.2, d=200.1.1.253->192.168.1.3 [180] |

图12.27 R2的终端信息

端口号已改为9。

|  |
| --- |
| R2#show ip nat translations  Pro Inside global Inside local Outside local Outside global  icmp 200.1.1.253:9 192.168.1.1:9 200.1.1.2:9 200.1.1.2:9  R2#  \*Oct 27 09:48:41.467: NAT: expiring 200.1.1.253(192.168.1.3) icmp 9 (9)  R2# |

图12.28 查看R2的NAT转换表

12.5实验命令列表

表12.1 实验命令列表

|  |  |
| --- | --- |
| 配置静态NAT | ip nat inside source static [inside local ip address] [inside global ip address] |
| 删除静态 NAT条目 | no ip nat inside source static [inside local ip address] [inside global ip address] |
| 指定内部ip地址接口 | ip nat inside |
| 指定外部ip地址接口 | ip nat outside |
| 查看NAT转换表 | show ip nat translations |
| 清空NAT转换表 | clear ip nat translation \* |

12.6实验问题

# 第十三章 ACL实验

## 13.1实验前准备

ACL访问控制列表是路由器和交换机接口的指令列表，用来控制端口进出的数据包。ACL适用于所有的被路由协议。配置ACL后，可以限制网络流量，允许特定设备访问，指定转发特定端口数据包等。目前有两种主要的ACL：标准ACL和扩展ACL。扩展ACL相比标准ACL提供了更广泛的控制范围。

## 13.2实验要求

本次试验，两台路由器通过 s0/0/0口互联，hostname 分别为R1和R2，两设备采用192.168.1.0/24 网段。实现标准 ACL 和扩展 ACL 的实验，并对两种ACL进行比较。

## 13.3实验拓扑

拓扑如图13.1所示：

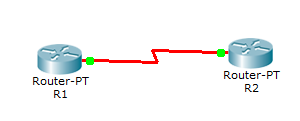


图13.1 实验拓扑图

## 13.4实验过程

首先为了区分两台路由器，分别将设备的hostname 改为 R1，R2，在相应接口下配置 IP 地址以及时钟速率，以保证 3 层连通性，具体配置如下：

Router>enable

Router#conf terminal  
Router(config)#hostname R1

R1(config)#interface serial 0/0/0

R1(config-if)#ip address 192.168.1.1 255.255.255.0

R1(config-if)#no shutdown

R1(config-if)#clock rate 64000

Router(config)#hostname R2

R2(config)#interface serial 0/0/0

R2(config-if)#ip address 192.168.1.2 255.255.255.0

R2(config-if)#no shutdown

一、使用扩展的ACL封杀R1到R2的PING命令

1. 验证3层连通性

|  |
| --- |
| R2#ping 192.168.1.1  Type escape sequence to abort.  Sending 5,100-byte ICMP Echos to 192.168.1.1, timeout is 2 seconds:  !!!!!  Success rate is 100 percent(5/5), round-trip min/avg/max = 28/28/28 ms |

图13.2 验证是否联通

1. 创建 ACL

R1(config)#access-list 100 deny icmp 192.168.1.1 0.0.0.0 192.168.1.2 0.0.0.0

R1(config)#access-list 100 permit ip any any

R1#show ip access-lists

Extended IP access list 100

deny icmp host 192.168.1.1 host 192.168.1.2

permit ip any any

1. 应用ACL到接口

R1(config)#interface serial 0/0/0

R1(config-if)#ip access-group 100 out

1. 验证效果

|  |
| --- |
| Enter configuration commands, one per line. End with CNTL/Z.  R1(config)#interface s0/0/0  R1(config-if)#ip access-group 100 out  R1(config-if)#^Z  R1#p  \*Oct 27 09:53:14.979:%SYS-5-CONFIG\_I: Configured from console by consoleing  .168.1.2  Type escape sequence to abort.  Sending 5,100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:  !!!!!  Success rate is 100 percent(5/5), round-trip min/avg/max = 28/28/32 ms  R1#ping 192.168.1.2  Type escape sequence to abort.  Sending 5,100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:  !!!!!  Success rate is 100 percent(5/5), round-trip min/avg/max = 28/28/32 ms |

图13.3 实验效果

对于ACL的放置位置，有以下的原则：扩展ACL放置在靠近源的位置，标准ACL 放置在靠近目的位置。那按照上述的原则，创建一个扩展的ACL，并放置在源端，并没有错误。

1. 排错

R1#show ip access-lists

Extended IP access list 100

deny icmp host 192.168.1.1 host 192.168.1.2

permit ip any any (15 matches)

问题分析：最后一条语句匹配到15个数据包。对于 ACL，有个非常重要的特性，他不能过滤本地数据流！也就是说，对于 R1 上发送的数据，设置在 R1 接口上的 ACL 并不能对它进行过滤。为了能对数据流进行过滤，需要把ACL 设置在对端的 R2 上 。

1. 在R2上设置并应用ACL

R2(config)#access-l 100 deny icmp host 192.168.1.1 host 192.168.1.2

R2(config)#access-l 100 permit icmp any any

R2(config)#interface serial 0/0/0

R2(config-if)#ip access-group 100 in

1. 检测效果

|  |
| --- |
| R1#ping 192.168.1.2  Type escape sequence to abort.  Sending 5,100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:  U.U.U  Success rate is 0 percent(0/5)  show ip access-lists  Extended IP access list 100  10 deny icmp host 192.168.1.1 host 192.168.1.2 (15matches)  20 permit icmp any any |

图13.4 ping测试

实验成功，ICMP包被拒绝，如图13.4所示。

二、使用 ACL禁止R1到R2的TELNET应用

注意：在进行第二部分实验前请将第一部分配置清除

路由器名为:R1、R2。R1的S0/0/0端口与R2的S0/0/0 端口相连，IP地址和第一部分相同,在R2上设置特权密码为nju，线路密码为cisco。从R1使用 PING命令测试到R2的连通性，结果可达，但却不可以TELNET到R2 。

有两种方法可以实现这样的操作。

方法一:使用扩展 ACL

1. 检测基本配置

R2(config)#enable secret nju

R2(config)#line vty 0 4

R2(config-line)#password cisco

R2(config-line)#login

|  |
| --- |
| R1#ping 192.168.1.2  Type escape sequence to abort.  Sending 5,100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:  !!!!!  Success rate is 100 percent(5/5), round-trip min/avg/max = 28/28/28 ms  R1#telnet 192.168.1.2  Trying 192.168.1.2 ... Open  User Access Verification  Password:  R2>enable  Password:  R2#exit |

图13.5 检验基本配置

1. 创建 ACL

R2(config)#access-list 101 deny tcp host 192.168.1.1 any eq 23

R2(config)#access-list 101 permit ip any any

R2(config)#int s0/0/0

R2(config-if)#ip access-group 101 in

|  |
| --- |
| R1#telnet 192.168.1.2  Trying 192.168.1.2 ...  % Destination unreachanle; gateway or host down  R1#ping 192.168.1.2  Type escape sequence to abort.  Sending 5,100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:  !!!!!  Success rate is 100 percent(5/5), round-trip min/avg/max = 28/28/32 ms |

图13.6 测试结果

telnet被拒绝但ping成功。

方法二:使用标准 ACL

1. 删除先前的配置

R2(config)#int s0/0/0

R2(config-if)#no ip access-group 101 in

1. 创建 ACL

R2(config)#access-list 1 deny host 192.168.1.1

R2(config)#access-list 1 permit any

1. 应用 ACL

R2(config)#line vty 0 4

R2(config-line)# access-class 1 in

实验结果如图13.7所示：

|  |
| --- |
| R1#telnet 192.168.1.2  Trying 192.168.1.2 ...  % Connection refused by remote host |

图13.7 使用标准ACL

问题分析：设置了2种ACL，但是得到的效果却不一样。如果是使用了扩展的ACL，那么它的提示是“% Destination unreachable; gateway or host down”,说明23号端口根本不可达。如果是使用了标准的 ACL 放置在VTY线路中，则提示“% Connection refused by remote host”，说明的确是到达了23号端口，只不过被拒绝了。在实际使用中最好使用扩展的ACL，减少23号端口的负担。

## 13.5实验命令列表

表13.1 实验命令列表

|  |  |
| --- | --- |
| 配置access list | access-list [list number] [permit|deny] [source address] [address] [wildcard mask] [log] |
| 将指定访问列表应用到相关接口，并指定ACL作用的方向 | ip access-group {[access-list-number]|[name]} [int|out] |
| 显示已设置的访问控制列表内容 | show ip access-lists |

## 13.6实验问题

第十四章 PPP验证实验

## 14.1实验前准备

在现实生活中，上网需要先向电信和联通这种运营商去缴费申请一个用户名和密码，然后通过登录连接到互联网。本章实验描述了如何通过PPP的验证使链路能够通畅。

## 14.2实验要求

本次实验主要完成以下两项操作:

PAP验证

1.为路由器指定唯一主机名，为了辨别设备，需要设置不同的主机名。

2.列出认证路由器时所使用的远程主机名称和口令，PAP为单项验证，所以被验证方需要正确的主机名和口令，方能使链路通畅。

3.WAN接口上完成PPP协议的封装。PAP验证基于PPP协议，必须封装后才能启用。

4.nju1为服务端，nju2为客户端，客户端主动向服务端发出认证请求，密码设置为ccna。

CHAP 验证

1.为路由器指定唯一主机名，为了辨别设备，需要设置不同的主机名

2.列出认证路由器时所使用的远端主机名称和口令，密码为ccna。CHAP 为双向验证，通过将对方发送的用户名和口令与本地的用户列表来对比确认。一致通过，不一致链路阻塞。WAN接口上完成PPP 协议的封装和CHAP 认证的配置PAP 验证基于PPP 协议，必须封装后才能启用。

## 14.3实验拓扑

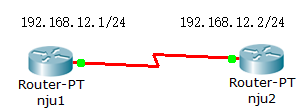


图14.1 拓扑图

## 14.4实验过程

#### 14.4.1 PAP验证

1.配置nju1的服务端设置

Router(config)#hostname nju1

nju1(config)#username nju password ccna

nju1(config)#interface serial 0/0/0

nju1(config-if)#ip address 192.168.12.1 255.255.255.0

nju1(config-if)#encapsulation ppp

nju1(config-if)#ppp authentication pap

nju1(config-if)#no shutdown

2.配置nju2的客户端设置

Router(config)#hostname nju2

nju2(config)#interface serial 0/0

nju2(config-if)#ip address 192.168.12.2 255.255.255.0

nju2(config-if)#clock rate 64000

nju2(config-if)#encapsulation ppp

nju2(config-if)#no shutdown

注：如果这时不设置被验证放所要放松的用户名和列表，接口不会UP

3.client 端发送用户名和密码

nju2(config-if)#ppp pap sent-username adsf password adsf

注：当用户名和口令中的任意一个和验证方的本地用户列表不同时，无法通信。

nju2#ping 192.168.12.1

测试结果如图：

|  |
| --- |
| Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:  .....  Success rate is 0 percent (0/5) |

图14.2 ping失败

4.设置正确的用户名和密码

nju2(config-if)#ppp pap sent-username nju password ccna

nju2(config-if)#end

nju2#ping 192.168.12.1

测试结果如图：

|  |
| --- |
| Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:  !!!!!  Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/32 ms |

图14.3 ping成功

#### 14.4.2 CHAP 验证

1.配置nju1

Router(config)#hostname nju1

nju1(config)#username nju2 password ccna

nju1(config)#interface serial 0/0

nju1(config-if)#ip address 192.168.12.1 255.255.255.0

nju1(config-if)#encapsulation ppp

nju1(config-if)#ppp authentication chap

nju1(config-if)#no shutdown

2.配置nju2

Router(config)#hostname nju2

nju2(config)#inter serial 0/0

nju2(config-if)#ip address 192.168.12.2 255.255.255.0

nju2(config-if)#clock rate 64000

nju2(config-if)#encapsulation ppp

nju2(config-if)#ppp authentication chap

nju2(config-if)#no shutdown

注：对端需要配置相同，因为chap 是双向认证，由于一端没有发送本地用户名和列表，导致链路不通。

3.设置nju2 上的用户名和密码

nju2(config)#username nju1 password ccnp

nju2#ping 192.168.12.1

测试结果如图：

|  |
| --- |
| Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:  .....  Success rate is 0 percent (0/5) |

图14.4 ping失败

4.设置正确的用户名和密码

nju2(config)#username nju1 password ccna

nju2#ping 192.168.12.1

测试结果如图：

|  |
| --- |
| Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:  !!!!!  Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/32 ms |

图14.5 ping成功

在验证通过的情况下，将任意一边的口令随意设置成一个非ccna 的口令，再测试连通性。

nju2(config)#username nju1 password ccnp

nju2#ping 192.168.12.1

测试结果如图：

|  |
| --- |
| Type escape sequence to abort.  Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:  !!!!!  Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/32 ms |

图14.6 ping仍然成功

注：因为当验证通过后会一直保存已经建立好的连接，解决方法是将接口关闭在启动。

## 14.5实验命令列表

表14.1

|  |  |
| --- | --- |
| 启用PPP封装协议 | encapsulation ppp |
| 启用PAP身份验证 | ppp authentication pap |
| 设置被验证发送的用户名和口令 | ppp pap sent-username [用户名] password [密码] |
| 启用CHAP认证协议 | ppp authentication chap |

## 14.6实验问题

# 第十五章 帧中继实验

## 15.1实验前准备

帧中继是由ITU-T标准化的高性能WAN协议，并在美国广泛应用。帧中继是一种面向连接的数据链路层技术，它定义了路由器与服务提供商的本地接入交换设备之间的互联过程。

连接到帧中继WAN的设备分为以下两类：

DTE：DTE设备通常位于客户所在地并且可能为客户所有，帧中继接入设备、路由器、网桥都属于 DTE设备。

DCE：运营商所拥有的网间设备，DCE设备的作用是在网络中提供时钟服务和交换服务，并通过WAN传输数据。

## 15.2实验要求

本次实验主要完成以下几个要求：

1配置帧中继交换机。帧中继是一种面向连接的数据链路层技术，它定义了路由器与服务提

供商的本地接入交换设备之间的互联过程。

2按照拓扑组建和配置路由器。

3进行验证实验。通过命令查看在帧中继交换机上虚电路交换的过程；通过手动配置DLCI

号与IP 地址的映射，在路由器上分别禁用逆向ARP 查询。

## 15.3实验拓扑

拓扑如图所示：

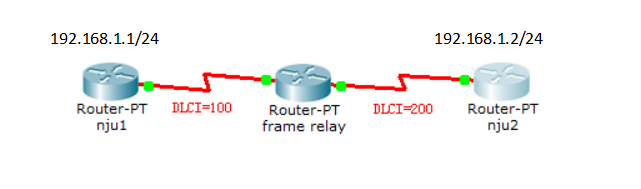


图15.1 帧中继实验拓扑图

## 15.4实验过程

1配置中间的帧中继交换机

Fr-sw(config)#frame-relay switching

Fr-sw(config)#interface serial 0/0/0

Fr-sw(config-if)#encapsulation frame-relay

Fr-sw(config-if)#frame-relay intf-type dce

Fr-sw(config-if)#clock rate 64000

Fr-sw(config-if)#frame-relay route 100 interface serial 0/0/1 200

Fr-sw(config-if)#no shutdown

Fr-sw(config-if)#exit

Fr-sw(config)#interface serial 0/0/1

Fr-sw(config-if)#encapsulaiton frame-relay

Fr-sw(config-if)#frame-relay intf-type dce

Fr-sw(config-if)#clock rate 64000

Fr-sw(config-if)#frame-relay route 200 interface serial 0/0/0 100

Fr-sw(config-if)#no shutdown

2配置 nju1

nju1(config)#interface serial 0/0/1

nju1(config-if)#ip address 192.168.1.1 255.255.255.0

nju1(config-if)#encapsulation frame-relay

nju1(config-if)#no shutdown

3配置 nju2

nju2(config)#interface serial 0/0/0

nju2(config-if)#ip address 192.168.1.2 255.255.255.0

nju2(config-if)#encapsulation frame-relay

nju2(config-if)#no shutdown

4验证实验

|  |
| --- |
| nju2#ping 192.168.1.1  Type escape sequence to abort.  Sending 5, 100 type percent (5/5), round-trip min/avg/max=56/56/60 ms |

图15.2 检测连通性

|  |
| --- |
| nju2#show frame-relay map  Serial0/0/1 (up): ip 192.168.1.1 dlci 100(0\*64,0\*1840), dynamic, |

图15.3 终端显示信息

通过命令可以查看在帧中继交换机上虚电路交换的过程。从接口s0/0/1的200虚电路交换到s0/0/0的100的虚电路。

|  |
| --- |
| Fr-sw#show frame-relay route  Input Intf Input Dlci Output Intf Output Dlci Status  Serial0/0/0 100 Serial0/0/1 200 active  Serial0/0/1 200 Serial0/0/0 100 active |

图15.4 终端显示信息

路由器的虚电路200在交换机s0/0/1上，这样从交换机s0/0/1过来的数据就会发送给路由器的s0/0/0上。

|  |
| --- |
| nju1#show frame-relay pvc  PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)  Active Inactive Deleted Static  Local 1 0 0 0  Switched 0 0 0 0  Unused 0 0 0 0  DLCI = 200, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0  input pkts 16 output pkts 16 in bytes 1594  out bytes 1594 dropped pkts 0 in pkts dropped 0  out pkts dropped 0 out bytes dropped 0  in FECN pkts 0 in BECN pkts 0 out FECN pkts 0  out BECN pkts 0 in DE pkts 0 out DE pkts 0  out va=casj=t okts 1 out bcast bytes 34  5 minute input rate 0 bits/sec, 0 packets/sec  5 minute output rate 0 bits/sec, 0 packets/sec  pvc create time 00:08:30, last time pvc status changed 00:08:20 |

图15.5 终端显示信息

也可以在nju1和nju2上分别禁用逆向ARP查询，手动配置DLCI号与IP地址的映射。

nju1的配置：

nju1(config-if)#no frame-relay inverse-arp

nju1(config-if)#frame-relay map ip 192.168.1.1 100 broadcast

nju2的配置：

nju2(config-if)#no frame-relay inverse-arp

nju2(config-if)#frame-relay map ip 192.168.1.2 200 broadcast

实验结果：

|  |
| --- |
| nju1#show frame-relay map  Serial0/0/0 (up) : ip 192.168.1.2 dlci 200 (0\*C8,0\*2080), static,  broadcast,  CISCO, status defined, active |

图15.6 终端显示信息

## 15.5实验命令列表

表15.1 实验命令列表

|  |  |
| --- | --- |
| 把路由器当成帧中继交换机 | frame-relay switching |
| 接口封装成帧中继 | encapsulation frame-relay |
| 配置接口是帧中继的DCE还是DTE | frame-relay intf-type dce |
| 配置帧中继交换表 | frame-relay route |
| 显示帧中继交换表 | show frame-relay route |
| 显示帧中继PVC状态 | show frame pvc |
| 查看帧中继映射 | show frame-relay map |
| 关闭帧中继自动映射 | no frame-relay inverse-arp |

## 15.6实验问题

# 第十六章 DHCP欺诈保护

## 16.1实验前准备

DHCP的主要作用是给网络中的其他设备动态分配IP地址，从而节约IP资源。

DHCP欺骗：攻击者可以通过伪造大量的IP请求包，消耗掉现有DHCP服务器的IP资源。当有计算机请求IP时，DHCP服务器就无法分配IP。此时，攻击者可以伪造一个DHCP服务器为计算机分配IP，并指定一个虚假的DNS服务器地址。这时，当用户访问网站的时候，就被虚假DNS服务器引导到错误的网站。

在交换机上开启DHCP snooping功能，绑定并过滤不信任的DHCP信息可以防止DHCP欺骗。对于信任端口收到的DHCP服务器报文，交换机不会丢弃而直接转发，来自非信任端口的DHCP报文则无法通过，从而有效的防止了DHCP欺骗。

## 16.2实验要求

本次实验要求在路由器上启用DHCP服务为两台计算机动态分配IP。此外，还需配置交换机的DHCP snooping功能防止DHCP欺骗。

## 16.3实验拓扑

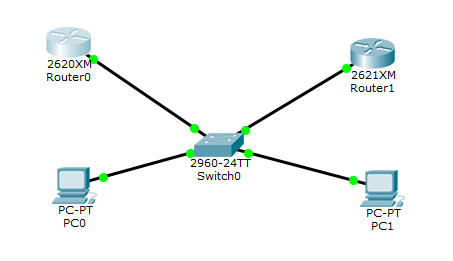


图16.1 实验拓扑

## 16.4实验过程

1．配置路由器Router1的DHCP功能。

Router1(config)#service dhcp

Router1(config)#ip dhcp pool nju1

Router1(dhcp-config)#network 10.1.1.0 255.255.255.0

Router1(dhcp-config)#default-router 10.1.1.1

Router1(dhcp-config)#dns-server 10.1.1.1

Router1(config)#ip dhcp excluded-address 10.1.1.1 10.1.1.10

2．设置计算机ip获取为DHCP

检查计算机IP并在Router1上查看地址分配，如图16.2所示。

|  |
| --- |
| Router#show ip dhcp binding  IP address Client-ID/ Lease expiration Type  Hardware address  10.1.1.21 00E0.A3A5.4929 - - Automatic  10.1.1.22 00D0.BAD4.D490 - - Automatic |

图16.2 dhcp地址分配信息

3．防止DHCP欺骗

按照步骤1配置Router0，将Router0的DHCP地址池设置为20.1.1.0

配置交换机snooping功能，将与Router1相连的f0/2端口设置为信任端口

Switch(config)#ip dhcp snooping

Switch(config)#ip dhcp snooping vlan 1

Switch(config)#int f0/2

Switch(config-if)#ip dhcp snooping trust

配置路由器Router1

Router1(config)#ip dhcp relay information trust-all

|  |
| --- |
| Switch#show ip dhcp snooping  Switch DHCP snooping is enabled  DHCP snooping is configured on following VLANs:  1  Insertion of option 82 is enabled  Option 82 on untrusted port is not allowed  Verification of hwaddr field is enabled  Interface Trusted Rate limit (pps)  ----------------------- ------- ----------------  FastEthernet0/1 no unlimited  FastEthernet0/2 yes unlimited  FastEthernet0/3 no unlimited  FastEthernet0/4 no unlimited |

图16.3 信任f0/2端口

此时，两台计算机IP地址均由Router1分配，IP地址如图16.4和16.5所示。

|  |
| --- |
| C:\>ipconfig  FastEthernet0 Connection:(default port)  Link-local IPv6 Address.........: FE80::2D0:BAFF:FED4:D490  IP Address......................: 10.1.1.22  Subnet Mask.....................: 255.255.255.0  Default Gateway.................: 10.1.1.1 |

图16.4 计算机0IP地址

|  |
| --- |
| C:\>ipconfig  FastEthernet0 Connection:(default port)  Link-local IPv6 Address.........: FE80::2E0:A3FF:FEA5:4929  IP Address......................: 10.1.1.21  Subnet Mask.....................: 255.255.255.0  Default Gateway.................: 10.1.1.1 |

图16.5 计算机1IP地址

将交换机f0/1设置为信任端口，f0/2设置为非信任端口。

Switch(config)#int f0/2

Switch(config-if)#no ip dhcp sn

Switch(config-if)#no ip dhcp snooping trust

Switch(config)#int f0/1

Switch(config-if)#ip dhcp snooping trust

Switch(config-if)#end

再次检查两台计算机的IP地址，如图16.6和16.7所示。

|  |
| --- |
| C:\>ipconfig  FastEthernet0 Connection:(default port)  Link-local IPv6 Address.........: FE80::2E0:A3FF:FEA5:4929  IP Address......................: 20.1.1.17  Subnet Mask.....................: 255.255.255.0  Default Gateway.................: 20.1.1.1 |

图16.6 计算机0IP地址

|  |
| --- |
| C:\>ipconfig  FastEthernet0 Connection:(default port)  Link-local IPv6 Address.........: FE80::2D0:BAFF:FED4:D490  IP Address......................: 20.1.1.18  Subnet Mask.....................: 255.255.255.0  Default Gateway.................: 20.1.1.1 |

图16.7 计算机1IP地址

可以看到，通过交换机snooping的配置，能够阻止非信任端口的DHCP报文传输，从而避免DHCP欺骗。

## 16.5实验命令列表

表16.1 实验命令列表

|  |  |
| --- | --- |
| 打开dhcp功能 | service dhcp |
| 配置dhcp地址池名称 | dhcp dhcp pool [pool name] |
| 配置要分配的网段 | network [address] netmask |
| 配置默认网关 | default-router [address] |
| 配置dns服务器 | dns-server [address] |
| 配置不分配地址 | ip dhcp excluded-address [address1] [address2] |
| 打开dhcp snooping功能 | ip dhcp snooping |
| 设置作用的vlan | ip dhcp snooping vlan *n* |
| 配置信任端口 | ip dhcp snooping trust |
| 配置dhcp中继代理的所有接口都作为dhcp中继信息选项的信任源 | ip dhcp relay information trust-all |

## 16.6实验问题