**实验三 路由协议实验**

**报告成绩：\_\_\_\_\_\_\_\_\_ 指导教师：\_\_\_\_\_\_\_\_\_**

**姓名：\_\_黄耀增\_\_ 学号：\_\_20231302231\_\_**

**班级：\_\_计算机科学与技术23-1\_\_**

1. **实验目的**
2. 掌握静态路由的配置方法；
3. 掌握测试静态路由连通性的方法；
4. 掌握RIPv1的配置方法；
5. 查看RIP路由的更新过程；
6. 掌握测试RIP网络连通性的方法。
7. **实验环境（15分）**
8. 系统环境：Windows 10 专业版；eNSP仿真平台； Wireshark。
9. 图示

   AI 生成的内容可能不正确。网络环境：构建的网络拓扑结构

【练习一网络拓扑图及IP地址分配】

图示

AI 生成的内容可能不正确。【练习二网络拓扑图及IP地址分配】

1. **实验内容及结果分析（55分）**
2. 练习一 配置静态路由：
3. 配置好PC1~PC4的网络参数；
4. 为路由器AR1、AR2和AR3配置端口IP地址；
5. 使得PC1/PC2与 PC3相互ping通，按顺序记录操作（命令），请对配置过程抓包结果进行说明与分析；
6. 使得PC3与 PC4相互ping通，按顺序记录操作（命令），请对配置过程抓包结果进行说明与分析。

**实验过程：**

在本练习中，首先按照实验要求新建拓扑结构，构建如实验环境图“【练习一网络拓扑图及IP地址分配】”所示的拓扑结构，然后依次点击PC设备，配置了PC1~PC4的网络参数，如下“表1.1 PC设备网络参数设置”所示：

**表1.1 PC设备网络参数设置**

|  |  |  |  |
| --- | --- | --- | --- |
| PC编号 | IP地址 | 子网掩码 | 网关 |
| PC1 | 192.168.1.2 | 255.255.255.0 | 192.168.1.1 |
| PC2 | 192.168.1.3 | 255.255.255.0 | 192.168.1.1 |
| PC3 | 192.168.2.2 | 255.255.255.0 | 192.168.2.1 |
| PC4 | 192.168.3.2 | 255.255.255.0 | 192.168.3.1 |

文本

AI 生成的内容可能不正确。 接下来，选中所有设备并点击启动，依次设置每一个路由器，按照实验要求配置相应的端口IP地址，如下“图1.1 AR1端口配置命令行”、“图1.2 AR2端口配置命令行”、“图1.3 AR3端口配置命令行”所示：

配置AR1的0/0/2接口

配置AR1的0/0/0接口

文本

AI 生成的内容可能不正确。**图1.1 AR1端口配置命令行**

配置AR2的0/0/2接口

配置AR2的0/0/1接口

配置AR2的0/0/0接口

**图1.2 AR2端口配置命令行**

文本

AI 生成的内容可能不正确。**图1.3 AR3端口配置命令行**

配置AR3的0/0/1接口

配置AR3的0/0/0接口

下面进行了三个不同主机之间的通信，包括同路由下的PC1与PC2通信、设置静态路由下的PC1与PC3通信、设置静态路由下的PC3与PC4通信。

文本

AI 生成的内容可能不正确。 （1）对于同路由下的PC1与PC2之间的通信，不用设置什么其他的东西，下面就简单展示一下效果，如下“图1.4 PC1 ping PC2 的命令行”、“图1.5 PC1 ping PC2 的抓包呈现”所示：

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。**图1.4 PC1 ping PC2 的命令行**

**图1.5 PC1 ping PC2 的抓包呈现**

（2）对于设置静态路由下的PC1与PC3通信，下面呈现详细过程：

文本

AI 生成的内容可能不正确。 首先打开PC1的命令行，尝试ping一下PC3，查看是否可以ping通，如下“图1.6 PC1 ping PC2 命令行”所示：

**图1.6 PC1 ping PC2 命令行**

可以观察到，在PC1的命令行输入**ping 192.168.2.2**后，命令行报错输出**Request timeout！**，这个输出表明，PC1没法直接和PC3通信，原因是PC1的路由表中没有有关于PC3的路由信息，为了实现PC1与PC3之间的通信，必须要对PC1与PC3的路由器分别设置静态路由。

文本

AI 生成的内容可能不正确。 对于PC1的路由器，打开AR1的命令行，输入**sys**进入系统模式，然后再输入**ip route-static 192.168.2.0 24 192.168.0.2**设置静态路由，详细呈现效果如下“图1.7 AR1设置静态路由”所示：

**图1.7 AR1设置静态路由**

文本

AI 生成的内容可能不正确。 由图1.7 AR1设置静态路由可以观察到没有任何输出的反馈，具体如何知道是否设置静态路由成功，需要在终端输入**display ip routing-table**来查看路由表，下面呈现了设置静态路由前后AR1路由表的变化，如下“图1.8 AR1设置静态路由前的路由表”、“图1.9 AR1设置静态路由后的路由表”所示：

文本

AI 生成的内容可能不正确。**图1.8 AR1设置静态路由前的路由表**

**图1.9 AR1设置静态路由后的路由表**

观察前后两张图，可以看到路由表中增加了一条Proto为Static的路由信息，对应的就是AR2的IP地址和下一跳路由器地址，从而可以知道设置静态路由成功。

电脑屏幕截图

AI 生成的内容可能不正确。 同理再设置一下AR2的静态路由，由于详细过程相似，这里简单呈现一下效果，打开AR2的命令行，输入如下“图1.10 AR2设置静态路由”所示的命令，呈现效果如下：

AR2配置静态路由后路由表中新增的路由数据。

AR2配置静态路由

输入命令ip route-static 192.168.1.0 24 192.168.0.1

AR2配置静态路由前的路由表

AR2配置静态路由后的路由表

**图1.10 AR2设置静态路由**

图形用户界面, 文本, 应用程序, 电子邮件

AI 生成的内容可能不正确。 最后再次打开PC1的命令行，输入**ping 192.168.2.2**，发现成功ping通，打开PC1的WireShark进行抓包，随机打开一个数据包，如下“图1.11 PC1 ping PC3抓包”所示：

PC1与PC3 ping通，抓包成功

**图1.11 PC1 ping PC3抓包**

观察上图1.11 PC1 ping PC3抓包，可以发现由于打开的是一个**reply**的回复包，源地址和目的地址分别是PC3（192.168.2.2）和PC1（192.168.1.2），是PC3对于PC1的请求进行一个答复，可见配置静态路由后，AR1和AR2的路由表中存在了双方的目的IP地址和下一跳地址，PC1和PC3可以成功进行互ping通信了。

（3）对于设置静态路由下的PC3与PC4通信，总体配置过程其实与PC1和PC3之间的通信并无很大区别，只是相应的对路由器静态路由设置的目的IP地址和下一跳地址有相应的改变。由上述PC1与PC3之间的通信可以知道在路由器没有设置静态路由前，直接实行互ping是没法ping通的，所以同理PC3与PC4的路由器在没有设置静态路由前也无法ping通。于是下面直接开始对PC3与PC4相对应的路由器AR2和AR3进行静态路由配置，分别打开AR2和AR3的命令行，分别执行**ip route-static 192.168.3.0 24 192.168.4.2**和**ip route-static 192.168.2.0 24 192.168.4.1**设置静态路由，如下“图1.12 AR2设置静态路由”、“图1.13 AR3设置静态路由”所示：

文本

AI 生成的内容可能不正确。文本

AI 生成的内容可能不正确。图形用户界面, 文本

AI 生成的内容可能不正确。图形用户界面, 文本

AI 生成的内容可能不正确。**图1.12 AR2设置静态路由**

AR3配置静态路由后的路由表

AR3配置静态路由

AR2配置静态路由

AR2配置静态路由后的路由表

**图1.13 AR3设置静态路由**

观察上“图1.12 AR2设置静态路由”、“图1.13 AR3设置静态路由”所呈现的效果，基本和PC1与PC3通信所呈现的一样，都是路由表中会多一条Proto为Static的路由数据，目的IP地址和下一跳路由地址分别对应的就是PC3和PC4通信所需的地址。

文本

AI 生成的内容可能不正确。然后打开PC3的命令行，执行PC3 ping PC 4，输入**ping 192.168.3.2**的命令，如下“图1.14 PC3 ping PC4命令行”所示：

**图1.14 PC3 ping PC4命令行**

文本

AI 生成的内容可能不正确。 同时打开PC4的WireShark抓包工具，启动抓包，打开一个数据包，如下“图1.15 PC3 ping PC4抓包”所示：

**图1.15 PC3 ping PC4抓包**

可以观察到图1.15 PC3 ping PC4抓包的数据中，PC3和PC4成功通信，同理，在AR2和AR3的路由表中，由于设置静态路由的缘故，包含了PC3与PC4之间通信的地址，所以PC3和PC4才能够成功通信，综上实验就是完成了。

1. 练习二 配置RIPv1动态路由并分析协议数据包：
2. 配置好PC1~PC3的网络参数；
3. 为路由器AR1、AR2和AR3配置端口IP地址；
4. 为路由器AR1、AR2和AR3配置RIP；
5. 使得PC1、PC2与 PC3相互ping通，按顺序记录操作（命令），请对配置过程抓包结果进行说明与分析。

**实验过程：**

在本练习中，首先按照实验要求新建拓扑结构，构建如实验环境图“【练习二网络拓扑图及IP地址分配】”所示的拓扑结构，然后依次点击PC设备，配置了PC1~PC4的网络参数，如下“表2.1 PC设备网络参数设置”所示：

**表2.1 PC设备网络参数设置**

|  |  |  |  |
| --- | --- | --- | --- |
| PC编号 | IP地址 | 子网掩码 | 网关 |
| PC1 | 223.1.1.1 | 255.255.255.0 | 223.1.1.254 |
| PC2 | 223.1.2.1 | 255.255.255.0 | 223.1.2.254 |
| PC3 | 223.1.3.1 | 255.255.255.0 | 223.1.3.254 |

文本

AI 生成的内容可能不正确。文本

AI 生成的内容可能不正确。文本

AI 生成的内容可能不正确。 接下来，选中所有设备并点击启动，依次设置每一个路由器，按照实验要求配置相应的端口IP地址，如下“图2.1 AR1端口配置命令行”、“图2.2 AR2端口配置命令行”、“图2.3 AR3端口配置命令行”所示：

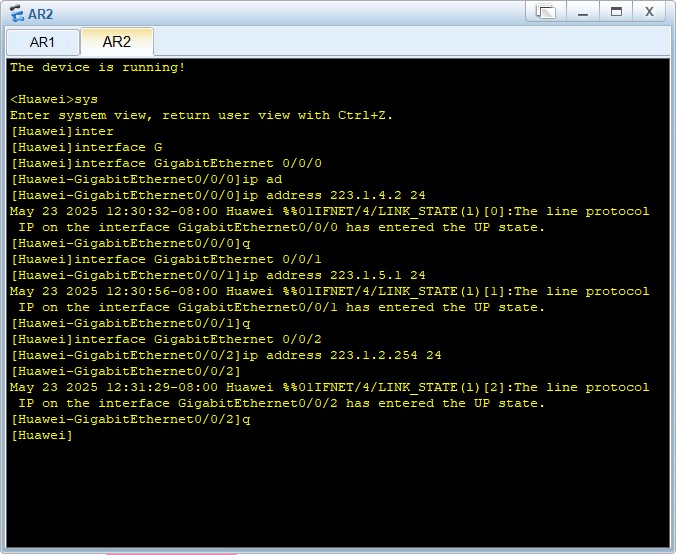
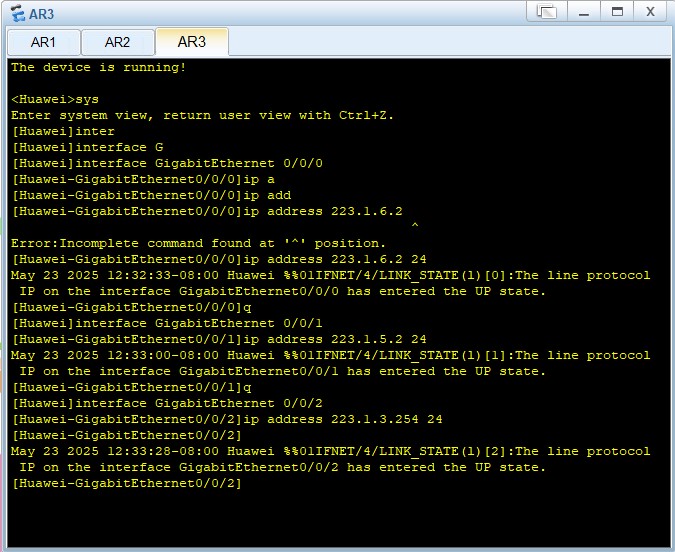
配置AR1的0/0/2接口

配置AR1的0/0/1接口

配置AR1的0/0/0接口

**图2.1 AR1端口配置命令行**

文本

AI 生成的内容可能不正确。**图2.2 AR2端口配置命令行**

配置AR3的0/0/2接口

配置AR3的0/0/1接口

配置AR3的0/0/0接口

配置AR2的0/0/2接口

配置AR2的0/0/1接口

配置AR2的0/0/0接口

**图2.3 AR3端口配置命令行**

下面继续按照实验要求位每一个路由器AR1、AR2、AR3配置RIP，按照网络的拓扑结构，设置相应的地址，如下“图2.4 AR1配置RIP”、“图2.5 AR2配置RIP”、“图2.6 AR3配置RIP”所示：

文本

AI 生成的内容可能不正确。文本

AI 生成的内容可能不正确。图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。**图2.4 AR1配置RIP**

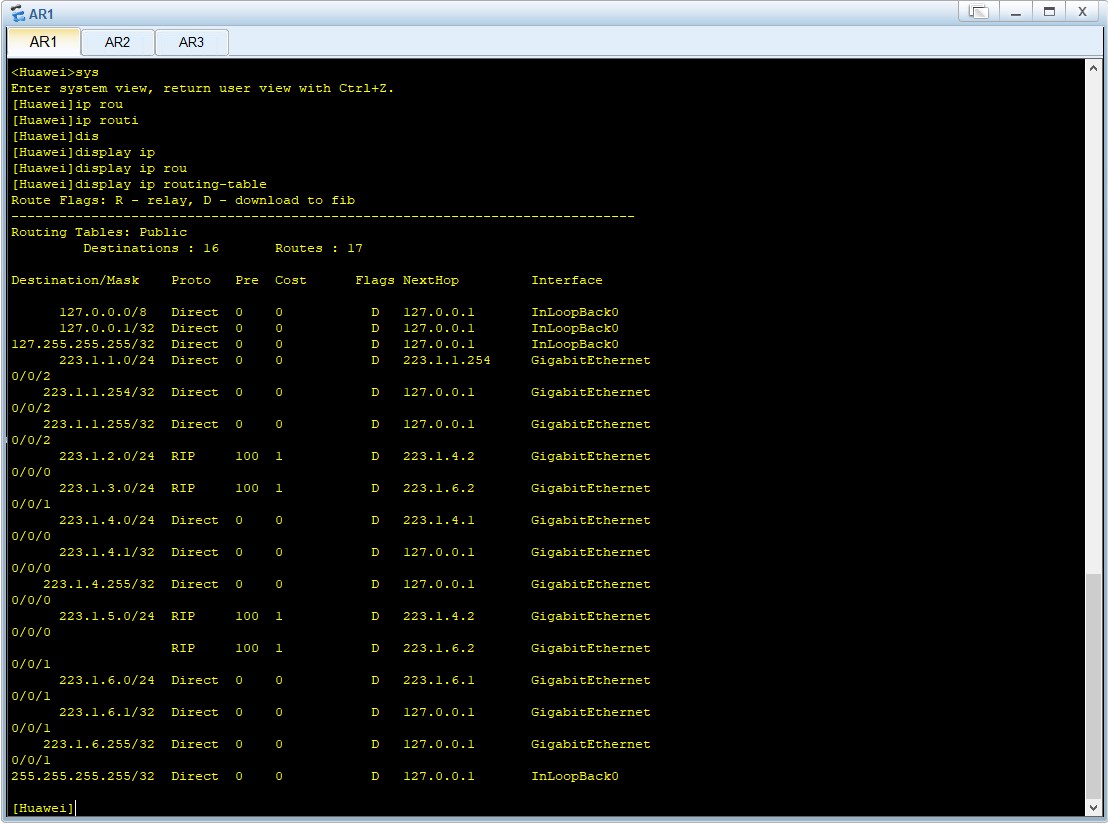
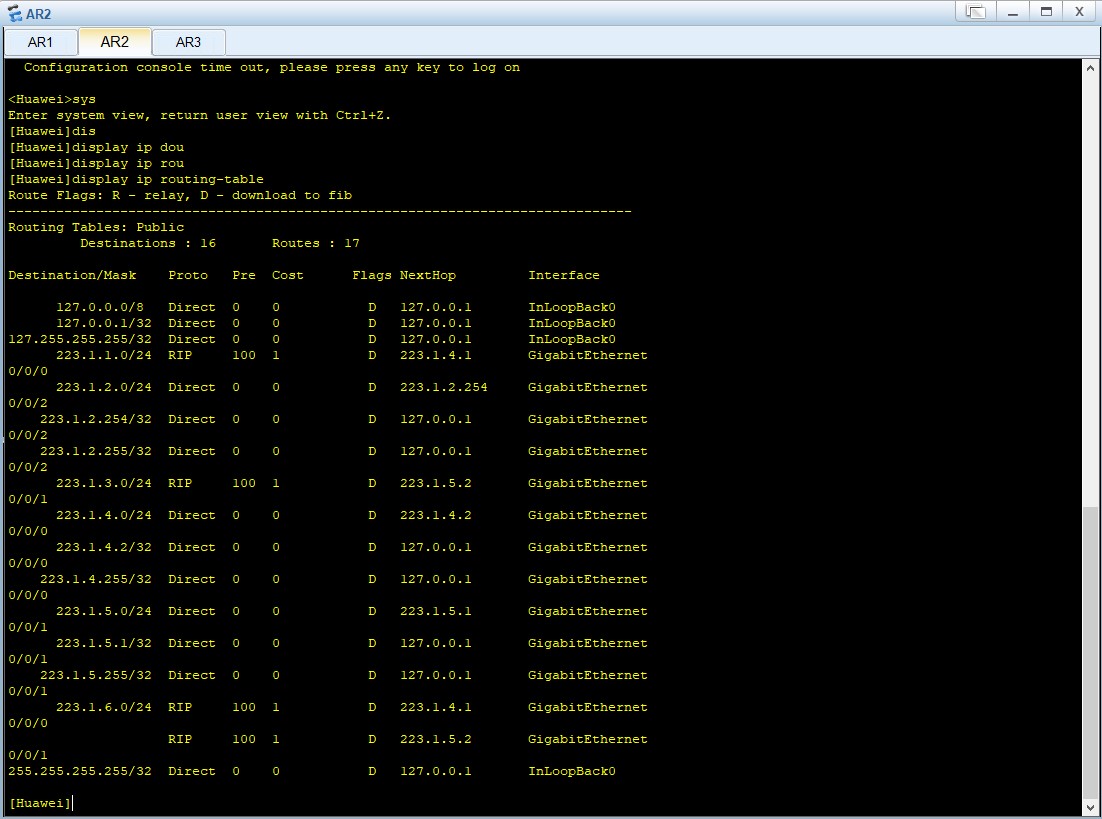
文本

AI 生成的内容可能不正确。文本

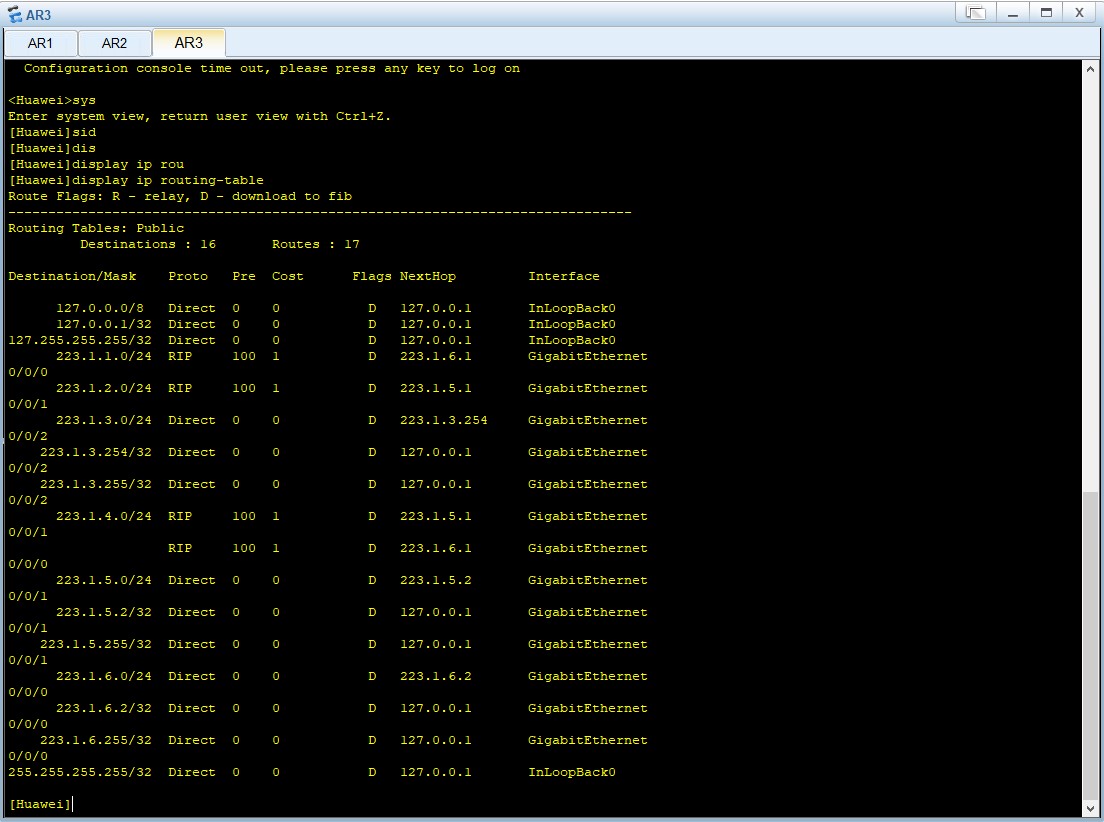
AI 生成的内容可能不正确。**图2.5 AR2配置RIP**

**图2.6 AR3配置RIP**

为了知道是否对路由器设置RIP成功，需要查看路由器的路由表，在每个路由器的命令行输入**display ip routing-table**来查看路由表，如下“图2.7 AR1查看路由表”、“图2.8 AR2查看路由表”、“图2.9 AR3查看路由表”所示：

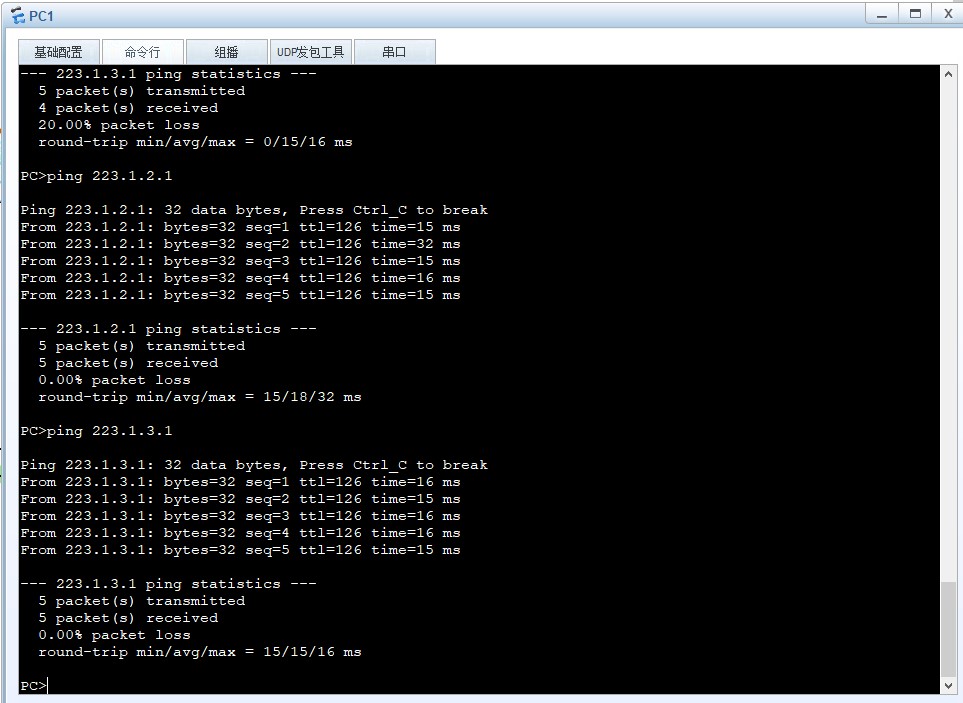
**图2.7 AR1查看路由表**

**图2.8 AR2查看路由表**

**图2.9 AR3查看路由表**

可以观察到每个路由器的路由表中存在刚刚设置的RIP信息，表明每个路由器的RIP设置成功，且对于这些RIP设置信息，包含到达路径，以及相应的度量值，对于上述这些RIP设置，度量值均为1。

接下来，已经成功设置完RIP，可以测试一下PC之间的通信连通性，下面以PC1 ping PC 2和PC1 ping PC3两个例子来验证连通性。

 首先打开PC1的命令行，在命令行分别输入ping 223.1.2.1（PC2的IP地址）和ping 223.1.3.1（PC3的IP地址），如下“图2.10 PC1 ping PC 2和PC3命令行”所示：

PC1 ping PC3

PC1 ping PC2

**图2.10 PC1 ping PC 2和PC3命令行**

图片包含 表格

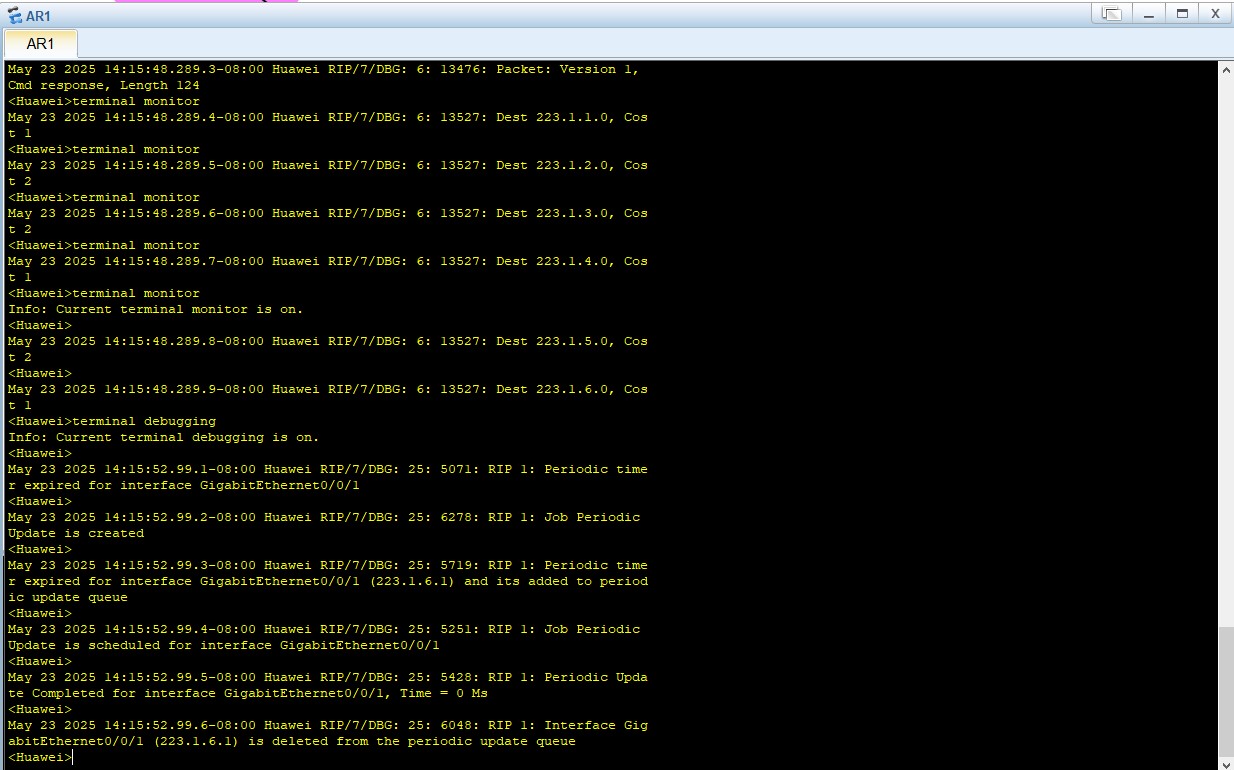
AI 生成的内容可能不正确。 分别打开PC2和PC3的WireShark抓包工具，开始捕获，可以观察到成功通信连通，如下“图2.11 PC2抓包数据”、“图2.12 PC3抓包数据”所示：

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。**图2.11 PC2抓包数据**

**图2.12 PC3抓包数据**

由上述抓包数据可以观察到，会出现红色和黑色的数据包，是有关RIP相关信息的，随机打开一个淡粉色的数据包，可以观察到，由于打开的都是**reply**类型的数据包，源地址变成了被访问主机的网关地址，目的地址变成了255.255.255.255的广播地址，这个就是因为配置了RIP才引起了这个变化。

 最后使用debug命令来开启RIP协议调试功能，并查看RIP协议的更新情况，打开路由器AR1的命令行，依次输入**debugging rip 1**、**terminal debugging**、**terminal monitor**，可以看到命令行不断地输出RIP协议的更新信息，连续不断，如下“图2.13 使用debug命令查看AR1的RIP更新情况”所示：

**图2.13 使用debug命令查看AR1的RIP更新情况**

可以观察很多信息，包含实时更新时间、设备名称、接口信息、延时信息等等，综上实验就是完成了。

1. **思考题（20分）**
2. 练习1：

* 如果想让 PC1（192.168.1.2）能够 ping 通 PC3（192.168.2.2），需要在 AR1 和 AR2 上分别配置什么静态路由？请写出具体命令。

AR1 需要配置：\_**ip route-static 192.168.2.0 255.255.255.0 192.168.0.2**\_

AR2 需要配置：\_**ip route-static 192.168.1.0 255.255.255.0 192.168.0.1**\_

* 为什么 PC1 不能直接访问 PC3，必须配置静态路由？

答：PC1 和 PC3 属于不同的子网（192.168.1.0/24 和 192.168.2.0/24），且未直接连接。路由器默认仅知晓直连网络的路由信息，需手动添加静态路由以告知如何转发跨子网的数据包。

1. 练习2：

* 配置 RIPv1 后，发现 PC2 无法访问 PC3。请列出 2 以上可能的原因：

原因1：\_路由器未在所有接口启用 RIPv1，导致路由信息未正确广播或接收\_

原因2：\_子网掩码不一致（如部分网络使用非标准掩码），RIPv1 无法处理变长子网掩码，导致路由汇总错误\_

原因3：\_网络中存在物理链路故障或接口未激活，导致路由更新无法传递\_

1. **实验总结（10分）**

本次实验通过配置静态路由和 RIPv1 动态路由，深入理解了路由协议的工作原理及实际应用。在静态路由实验中，手动配置路由条目验证了跨子网通信的必要性；而在 RIPv1 实验中，动态路由协议自动更新路由表的特点显著提升了网络管理的效率。

实验中遇到的问题主要包括路由配置遗漏、子网掩码错误等，通过检查路由表、抓包分析数据流，逐步定位并修正配置。例如，静态路由未添加导致通信失败，动态路由中未启用 RIP 协议接口导致路由信息缺失。

本次实验不仅巩固了路由配置的实践技能，还加深了对静态路由与动态路由优缺点的理解：静态路由适用于小型网络，配置简单但维护繁琐；动态路由（如 RIP）适用于复杂网络，自动化程度高但需注意协议限制（如跳数限制、不支持变长掩码）。