

**计算机网络**

**课程实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | IP数据报的的转发及收发 | | | | | |
| 姓名 | 常添 | | 院系 | 计算学部 | | |
| 班级 | 2203102 | | 学号 | 2022111699 | | |
| 任课教师 | 聂兰顺 | | 指导教师 | 聂兰顺 | | |
| 实验地点 | G001 | | 实验时间 | 2024.11.4 | | |
| 实验课表现 | 出勤、表现得分(10) |  | 实验报告  得分(40) |  | 实验总分 |  |
| 操作结果得分(50) |  |
| 教师评语 | | | | | | |
|  | | | | | | |

****

|  |
| --- |
| 实验目的： |
| 本次实验的主要目的。  1、了解原始套接字的基本概念和使用方法。 2、掌握路由器进行 IP 数据报转发的基本原理。 3、实现于原始套接字的 IP 数据报的发送和接收 4、实现基于原始套接字的 IP 数据报转发，包括 AF\_INET 和 AF\_PACKET 原始套接字的 应用。 |
| 实验环境： |
| 1. 硬件环境  计算机或虚拟机：  至少三台计算机/虚拟机：一台源主机、一台充当路由器的主机、一台目的主机。  每台主机都需要具备网络接口卡（NIC），路由器主机需要有两个网卡或虚拟网卡。  网线或虚拟网络连接：用于物理连接或虚拟环境中网络适配器的桥接。  2. 软件环境  操作系统：  使用基于 Linux 的操作系统，例如 Ubuntu、Debian、CentOS 等，这些系统更易于进行网络配置和原始套接字编程。确保内核支持原始套接字和 IP 转发。  3. 编译器和开发工具：  GCC：用于编译 C 语言程序。  make：用于管理编译和构建程序的过程。  4. 网络工具：  Wireshark 或 tcpdump：用于抓包和分析网络流量。  net-tools（可选）：提供 ifconfig、arp 等网络命令。  iproute2：提供 ip 命令，用于查看和修改路由表。 |
| 实验内容： |
| (1) 使用虚拟机实现多主机间的 UDP 数据报收发及转发 利用虚拟机搭建实验环境，掌握 Linux 下的 Socket 网络编程。 选做 1：改进程序，示例程序只实现了一个数据包（携带 1 条消息）的发、转、收过程，要求实现每条消息由控制台输入，并且不限制发送消息的数目。 (2) 基于单网口主机的 IP 数据转发及收发 在局域网中，模拟 IP 数据报的路由转发过程。通过原始套接字实现了完整的数据封装过程，实现了 UDP 头部、IP 头部、MAC 帧头部的构造。 选做 2：扩展实验的网络规模，由原始方案中 3 台主机增加到不少于 5 台主机，共同完成 IP 数据报转发及收发过程，要求采用转发表改进示例程序，增加程序通用性。 (3) 基于双网口主机的路由转发 构造了静态路由表，并实现了不同子网间的 IP 数据报查表转发过程。 选做 3：通过完善路由表，改进示例程序实现双向传输。 |
| 实验结果分析： |
| **实验4.1：用虚拟机实现多主机间的 UDP 数据报收发及转发**  三个虚拟机均为桥接模式，IP均使用DHCP协议自动分配，三台 Linux 虚拟机（客户机）与物理主机上处于同一子网中，网络拓扑结构如下：    **发送端：**    发送端IP为：172.29.172.121/20，共发送了两个数据报，内容分别为123456及qwerty  **转发端：**    转发端IP为172.29.174.244/20，运行完forward.c程序后可见：  第一条信息内容为：123456  Received time为2024-11-04 20:07:27  Source IP为172.29.172.121  第二条信息内容为：qwerty  Received time为2024-11-04 20:07:30  Source IP为172.29.172.121  **接收端：**    接受端的IP为172.29.168.254/20  由结果可知已经接收到两条信息，并有接收时间  **4.2 基于单网口主机的 IP 数据报转发及收发**  实验环境与4.1相同，其中第二台虚拟机作为有一个接口的路由器，网络拓扑结构如下：    这个实验使用了一个特殊的处理技巧，因为源主机（192.168.1.1）发送的 IP 数据包的目的地址为接收主机（192.18.1.3），由于二者处于一个子网中，这个 IP 数据报将直接交付给目的主机，因而正常不会经由转发主机（192.18.1.2）来进行转发，鉴于此种情况，我们将直接处理 IP 数据报的以太网帧封装，将 IP 数据报交给转发主机。  在发送主机上，首先构造出 IP 数据报，其头部源 IP 地址为 192.168.1.1，目的 IP 地址为192.168.1.3，然后构造用于封装该 IP 数据报的以太网帧，该帧头部的源 MAC 地址为aa.aa.aa.aa.aa.aa，但其目的 MAC 地址为转发主机的 MAC 地址 bb.bb.bb.bb.bb.bb（而不是接收主机的 MAC 地址 cc.cc.cc.cc.cc.cc），通过这种方法，将去往接收主机(192.168.1.3)的IP 数据报交给转发主机(192.168.1.2)进行转发。  实际上路由器转发的过程与此类似，在将 IP 数据报封装到链路层的 MAC 帧时，通过源MAC 地址和目的 MAC 地址的变化，来实现 IP 数据报从源主机到目的主机及传输路径上各路由器间的跳转传输。  这里仅展示五台虚拟机的转发结果： |
| 实验总结： |
| **发送端：**    数据包发送时间为：2024-11-04 20:00:11  源MAC地址：00:0c:29:ac:08:d4  源IP地址：172.29.172.121  目的MAC地址：00:0c:29:ef:69:fc（非目的主机的MAC地址而是下一台主机的 MAC地址）  目的IP地址：172.29.168.254  TTL:255  数据包的长度81bytes  **转发端1：**    **转发端2：**    **转发端3：**    这是三台中间虚拟机作为转发端的结果，可见源MAC地址以及目的MAC地址在变化，但是IP地址并没有改变，这是符合MAC协议以及IP协议的结果的；TTL是数据包的生存时间，可见每经过一台虚拟机TTL都会减一。  接收端：    这是接收端的结果，可见已经成功接收到了数据包并打印了数据包中的信息。  **4.3 基于双网口主机的路由转发**    先展示rev端机器的IP信息及网络配置信息：  7a9b00fef20cc4a579318845944d789  347a8ef4cca8de8cfeaf6c602b7815e  以下是src端的IP信息及网络配置：  2de50130361e6e6f3e4483591a0f230  d601614a561b845087934f4bfd40bb3  以下为route端主机的IP信息及路由配置：  2bc5399ffab3544110687fcf3c7fe9f  但是出现了源主机和目的主机ping不通的问题，原因是路由主机的路由功能未开启，以下为解决方法：  在路由主机运行sudo nano /etc/sysctl.conf，修改配置文件以启用ipv4转发：  67e7218f8a5f7e1da5b760159ad220e  可以看到net.ipv4.ip\_forward=1的注释被解除，运行sudo sysctl -p 以应用配置  运行cat /proc/sys/net/ipv4/ip\_forward显示转发情况：  152b66bd457ccacdb4534b03674dd13  rev主机ping source主机：  81e7dea0a32251c0810f17881165419  source主机ping rev主机：  7005137ee71c18916e61999db5c9356  以下为结果截图：  源主机：  087d888f5ebe07ec940234480713f3f  路由主机：  8a24ddf590497dd2ab239f5299e47b4  目的主机： |
| 参考文献：  《计算机网络：自顶向下方法》  《实验四FAQ》  《实验四讲解》  《实验四 IP数据报转发及收发-new》 |
| 结合实验过程和结果给出实验的体会和收获。   * 在实验中，通过实际配置网络设备和实现数据包的转发、捕获和处理，能够深刻理解网络协议的工作原理，如 IP 转发、路由表的作用以及数据包的传输路径。理论知识通过实践得以验证和巩固，使抽象的概念变得更加直观和具体。 * 学习了如何通过网络工具（如 ping、tcpdump、Wireshark）监控和分析网络数据包。   提升了在 Linux 系统中配置网络接口、路由表和启用 IP 转发的实践技能。学会了如何排查网络连接问题，如使用 ip route 和 arp 命令检查路由和 ARP 表的正确性。   * 在实验中遇到了一些问题，例如虚拟机网络配置不正确、IP 地址无法分配、ping 不通等。解决这些问题需要运用各种命令和工具进行排查和修复，这个过程加深了对网络配置细节的理解。 |