华南理工大学

《大学计算机基础》课程实验报告

实验题目： 网络报文抓取与分析

姓名： 江泽群 学号： 201530371299

班级： 15级电子科学与技术1班 组别：

合作者：

指导教师：

|  |
| --- |
| **实验概述** |
| 【实验目的及要求】  （1）、学习了解网络侦听  （2）、学习抓包工具wireshark的简单使用  （3）、对所侦听到的信息作初步分析  （4）、从侦听到的信息中分析TCP的握手过程，进行解释  （5）、分析了解TCP握手失败时的情况  【实验环境】  macOS Sierra 10.12.6 |
| **实验内容** |
| 【实验过程】  一、实验步骤及实验数据： 3.1 wireshark的安装 wireshark的二进制安装包可以在官网https://www.wireshark.org/download.html下载，或者可以在其他网站下载。  注意：下载后双击执行二进制安装包即可完成wireshark的安装。安装包里包含了WinPcap，并不需要单独安装WinPcap。 3.2 查看本机的网络适配器列表 **操作：**单击菜单Capture中的Options选项    记录下你看到的信息，并回答问题：  （1）、你机器上的网络适配器有几个？14个。  （2）、它们的编号分别是？  如图3-1所示，有5个接口的 mac 地址及 IP 地址能够被显示，其余的均无地址。   3.3 在指定网络适配器上进行监听 **操作：**在步骤3.2中弹出的选项中，选择指定的网络适配器WiFi: en0并单击开始按钮  记录并解释wireshark监听的包内容（解释1条记录即可）    报文第一行：641Bytes是报文大小  报文第二行：发送方与接收方各自的 mac 地址  报文第三行：发送方与接收方各自的 ip 地址  报文第四行：发送方的端口号（55680）接收方的端口号（80）请求序列号为1，回执序列号1。数据长度为587。 3.4 记录一个TCP三次握手过程 **操作：**在步骤3.3的基础上，单击start按钮后，打开命令行窗口并输入：telnet bbs.gznet.edu.cn，然后停止继续侦听网络信息。  在wireshark的Filter中输入表达式：  (ip.src==192.168.191.3 or ip.dst==192.168.191.3) and (tcp.dstport==23 or tcp.srcport==23)    其中192.168.191.3是我所在机器的IP。telnet服务的传输层采用了tcp协议，并且其默认端口是23。  在wireshark窗口中，记下所显示的内容（可事先通过重定向的方式记录）并回答问题。   1. 根据得到的信息解释所键入的filter定制中的参数的含义？   发送方IP地址是192.168.191.3或者接收方IP地址是192.168.191.3且发送方端口是23或接收方端口是23的TCP连接。  （2）请从得到的信息中找出一个TCP 的握手过程。并用截图形式记录下来。    （3）结合得到的信息解释TCP 握手的过程。  第一行：192.168.191.3请求建立连接(seq=0)  第二行：223.252.199.66收到报文(ack=1)，做出回应(seq=0)  第三行：192.168.191.3收到报文(seq=1)，做出回应(seq=1) 3.5 一个TCP握手不成功的例子 **操作：**在步骤3.3的基础上，单击start按钮后，打开命令行窗口并输入：telnet 192.168.1.101，然后停止继续侦听网络信息。  在wireshark的Filter中输入表达式：  ip.src==172.28.234.86 or ip.dst==172.28.234.86 and (tcp.dstport==23 or tcp.srcport==23)  其中172.28.234.86是我机器更换的IP，telnet服务是tcp协议并且其默认端口是23。  上面的IP 172.28.234.86 可改为任何没有打开telnet 服务的IP。比如：可以用身边同学的IP。（注：此IP 的机器上要求没有打开telnet 服务，但要求机器是开的，否则将无法主动拒绝一个TCP 请求）   1. 试从得到的信息中找出一个TCP 的握手不成功的过程，并用截图记录下来      1. 并结合所得到的信息解释这个握手不成功的例子。   本地的 IP发送第一次请求的报文给125.217.247.47，由于125.217.247.47没有打开 telnet 服务，故收到的是拒绝连接的报文，握手不成功。 3.6 侦听网络上的ARP包 关于 ARP 的说明：IP 数据包常通过以太网发送。但以太网设备并不识别32 位IP 地址:它们是以48 位以太网地址传输以太网数据包的。因此，必须把IP 目的地址对应到以太网的MAC 地址。在这两种地址之间存在着某种静态的或算法的映射，常常需要查看一张表。地址解析协议(Address Resolution Protocol,ARP)就是用来确定这些映象的协议。  ARP 工作时,送出一个含有所希望的IP 地址的以太网广播数据包。目的地主机或另一个代表该主机的系统以一个含有IP 和以太网地址对的数据包作为应答。发送者将这个地址对高速缓存起来，以后再需要向这个IP 发送消息时，通过直接查表就可以得要这台主机对就的MAC 地址了。  要看本机的 ARP 表（也即IP 与MAC 地址对应表）中的内容，只需在命令行方式下键入：arp –a命令即可。在下面的实验中，为了能够捕捉到ARP 消息，最好的做法是先将本机的ARP 表中的内容清空。这样当你使用Ping 命令时，它会首先使用arp 消息来查询IP 的MAC 地址。（当本地的ARP 表中有这个IP 对应的MAC 地址时，是不会再查询的）。要将本机的ARP 表中的内容清空，请使用命令：arp –d \*。关于ARP 更进一步的说明，请同学到网上查阅相关资料。  **操作：**在步骤3.3的基础上，单击start按钮后，打开命令行窗口并输入：  arp –d (清除ARP表)  ping 192.168.1.101 (Ping 任意一个和你的主机在同一个局域网的IP，说明:被Ping 的主机不能开防火墙）。  在wireshark的Filter中输入表达式：arp，然后就能会出现ARP 消息的记录。  4  请根据记录回答以下问题：   1. 记录下你所看到的信息，用截图形式。（找到ARP请求和ARP应答两个报文）      1. 请分析解释你的记录中的内容表示什么意思，从而说明ARP的工作原理。   ARP 通过广播的方式发出一个请求，寻找192.168.191.1，192.168.191.1收到这个包，并予以回应，回复其 mac 地址。 3.7 侦听网络上的ICMP包 关于 ICMP 的说明：ICMP 是“Internet Control Message Protocol”（Internet 控制消息协议）的缩写。它是TCP/IP 协议族的一个子协议，用于在IP 主机、路由器之间传递控制消息。控制消息是指网络通不通、主机是否可达、路由是否可用等网络本身的消息。这些控制消息虽然并不传输用户数据，但是对于用户数据的传递起着重要的作用。  我们在网络中经常会使用到 ICMP 协议，只不过我们觉察不到而已。比如我们经常使用的用于检查网络通不通的Ping 命令，这个“Ping”的过程实际上就是ICMP 协议工作的过程。还有其他的网络命令如跟踪路由的Tracert 命令也是基于ICMP 协议的。另外，ICMP 消息也常常被用于作为网络攻击的手段。关于ICMP 更进一步的说明，请同学到网上查阅相关资料。  **操作：**在步骤3.3的基础上，单击start按钮后，打开命令行窗口并输入：  ping 192.168.1.101 (Ping 任意一个和你的主机在同一个局域网的IP,说明:被Ping 的主机不能开防火墙）。  在wireshark的Filter中输入表达式：icmp，然后就能会出现ICMP 消息的记录。  5  请根据记录回答以下问题：   1. 记录下你所看到的信息？（找到回声请求和回声应答两个报文）      1. 请分析解释一下你的记录中的内容，从而说明ping应用的原理。（提示：因为ICMP 报文是放在IP 报文中发送的，故wireshark侦听到的报文中有部分内容是属于ICMP 报文的，另有部分内容是属于IP 报文的，请注意加以区分）   第一行中，本机向192.168.192.1发送一个Echo request 报文，目标地址接收到报文后，回应一个Echo reply 报文，本机收到报文后就可以确认目标地址是可达的，并确认存活。因此 ping 应用的原理就是通过发送Echo request报文到某一个特定的地址，根据是否收到Echo reply报文判断目标地址是否可达。 |
| **小结** |
| **本次实验内容较多，在这个过程我也学习到很多东西，包括 wireshark 的基本使用；观察到 TCP三次握手的过程以及了解了握手失败的情况；掌握了 ping 应用的使用以及原理，对 ARP 表和 IMCP 的工作方式有了初步的认识。实验过程也遇到不少问题，很多问题的原因是对理论知识了解不够透彻，通过把基础知识巩固之后，很多困难也就迎刃而解了。** |
| **指导教师评语及成绩** |
| 评语：  成绩：            指导教师签名：                                                 批阅日期： |