计算机网络实验报告

警示

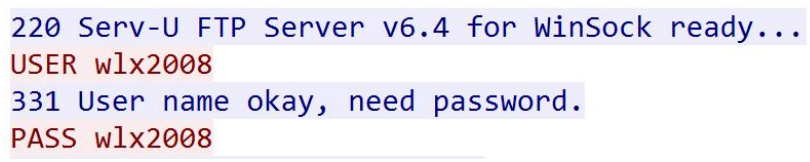
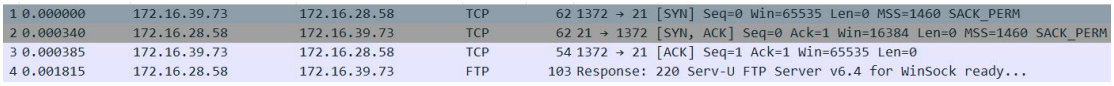
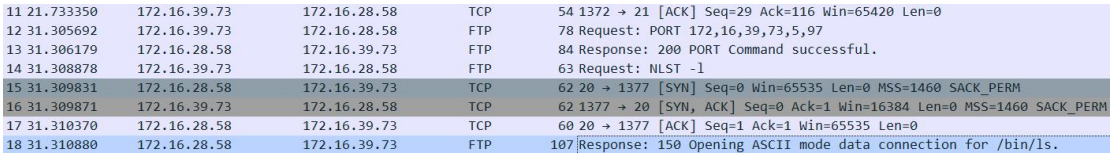
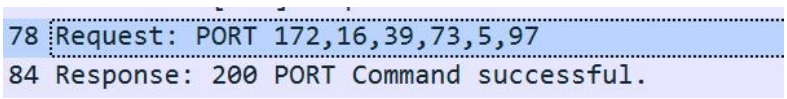
1. 实验报告如有雷同，雷同各方当次实验成绩均以 0 分计。
2. 当次小组成员成绩只计学号、姓名登录在下表中的。
3. 在规定时间内未上交实验报告的，不得以其他方式补交，当次成绩按 0 分计。
4. 实验报告文件以 PDF 格式提交。

院系	计算机学院	班 级	人工智能与大数据	组长	陈欣宇
学号	21307347	21307350	21307100		
学生	陈欣宇	高宇	陈华清		

Ftp 协议分析实验

一、打开“FTP 数据包”的“ftp 例 1.cap”文件，进行观察分析，回答以下问题(见附件)

题号	
1	FTP 客户端的 mac 地址是多少？
答案	00:14:2a:20:12:96
截图	› Frame 1: 62 bytes on wire (496 bits), 62 bytes captured ⌄ Ethernet II, Src: Elitegro_20:12:96 (00:14:2a:20:12:96), › Destination: DigitalC_02:b7:57 (00:03:0f:02:b7:57)
分析	第一个报文由客户端传向服务端，故客户端 mac 地址为 src 地址
2	第 1、2、3 号报文的作用是什么？
答案	三次握手，在 FTP 客户端和服务端之间建立连接
截图	Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) Ethernet II, Src: Elitegro_20:12:96 (00:14:2a:20:12:96), Dst: DigitalC_02:b7:5 Internet Protocol Version 4, Src: 172.16.39.73, Dst: 172.16.28.58 Transmission Control Protocol, Src Port: 1372, Dst Port: 21, Seq: 0, Len: 0 Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) Ethernet II, Src: DigitalC_02:b7:57 (00:03:0f:02:b7:57), Dst: Elitegro_20:12:9 Internet Protocol Version 4, Src: 172.16.28.58, Dst: 172.16.39.73 Transmission Control Protocol, Src Port: 21, Dst Port: 1372, Seq: 0, Ack: 1, L Source Port: 21 Destination Port: 1372 [Stream index: 0] [Conversation completeness: Complete, WITH_DATA (31)] [TCP Segment Len: 0] Sequence Number: 0 (relative sequence number) Sequence Number (raw): 2054701995 [Next Sequence Number: 1 (relative sequence number)] Acknowledgment Number: 1 (relative ack number) Frame 3: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) Ethernet II, Src: Elitegro_20:12:96 (00:14:2a:20:12:96), Dst: DigitalC_02:b7:57 Internet Protocol Version 4, Src: 172.16.39.73, Dst: 172.16.28.58 Transmission Control Protocol, Src Port: 1372, Dst Port: 21, Seq: 1, Ack: 1, Ler Source Port: 1372 Destination Port: 21 [Stream index: 0] [Conversation completeness: Complete, WITH_DATA (31)] [TCP Segment Len: 0] Sequence Number: 1 (relative sequence number) Sequence Number (raw): 1709874007 [Next Sequence Number: 1 (relative sequence number)] Acknowledgment Number: 1 (relative ack number)

分析	第 1 号报文发送 SYN=1 和随机生成的 ack number 的连接请求；第 2 号报文发送 Seq=0、Ack=1、ack=1 确认连接请求；第 3 号报文发送 ACK=1、ack=1 给服务端，检查后连接建立成功
3	该数据包中共有多少个 TCP 流？
答案	5
截图	
分析	通过追踪 TCP 流中可以查看到有 5 个流，主要表现为端口的不同
4	用什么用户和密码登录成功？
答案	用户：wlx2008 密码：wlx2008
截图	
分析	由图片 USER PASS 内容可见
5	该 FTP 的命令连接和数据连接分别是什么样的连接？
答案	三次握手后建立命令连接，数据连接为主动连接。
截图	命令连接：  数据连接： 
分析	一、命令连接：第 1、2、3 号报文是客户端使用 1372 号端口向服务端 21 号端口发起三次握手，第四号报文中服务端回应客户端，表示形成命令连接，服务端和客户端之间可以使用 ftp 命令。 二、数据连接：客户端请求并提供端口，FTP 服务器通过 20 端口返回同意，然后在第 15 至 17 号报文中，服务器开放端口 20 接受来自客户端的数据连接，客户端则将使用端口 1377 进行数据连接，数据连接建立，可以传输数据。
6	该 FTP 的连接模式是那种？为什么？
答案	主动连接
截图	
分析	使用 PORT 命令，向服务器发送客户端口，待服务器主动连接
7	最后四个报文的作用是什么？
答案	四次挥手，断开连接

截图	207 168.026381 172.16.39.73 172.16.28.58 TCP 54 1372 → 21 [FIN, ACK] Seq=248 Ack=1203 Win=64333 Len=0 208 168.026708 172.16.28.58 172.16.39.73 TCP 60 21 → 1372 [ACK] Seq=1203 Ack=249 Win=65288 Len=0 209 168.026762 172.16.28.58 172.16.39.73 TCP 60 21 → 1372 [FIN, ACK] Seq=1203 Ack=249 Win=65288 Len=0 210 168.026800 172.16.39.73 172.16.28.58 TCP 54 1372 → 21 [ACK] Seq=249 Ack=1204 Win=64333 Len=0
分析	一：客户端发送 FIN 报文，Seq=248 二：服务端发送 ACK 报文，Ack=248+1 表示确认收到，之后客户端到服务端的连接释放 三：服务器发送 FIN 报文，Seq=1023、Ack=248+1，等待用户确认 四：客户发送 ACK 报文，Ack=1023+1 应答，随后经过一定时间进入关闭状态
8	该数据包中有多少个 ftp 的命令及应答，其含义分别是什么？
答案	32 个，有命令 USER 后加用户名命令表示登入用户名，PASS 后加密码表示登录密码，PORT 命令向服务端发送连接端口号，NLST -l 命令为列举目录，RNFR 和 RNTO 为重命名操作，STOR 为存储文件到服务端，RETR 为从服务器获取文件，QUIT 命令为终止连接。
截图	220 Serv-U FTP Server v6.4 for WinSock ready... USER wlx2008 331 User name okay, need password. PASS wlx2008 230 User logged in, proceed. PORT 172,16,39,73,5,97 200 PORT Command successful. NLST -l 150 Opening ASCII mode data connection for /bin/ls. 226-Maximum disk quota limited to 307200 kBytes Used disk quota 0 kBytes, available 307200 kBytes 226 Transfer complete. XMKD jjj 257 "/jjj" directory created. RNFR jjj 350 File or directory exists, ready for destination name RNTO ppp 250 RNTO command successful. RETR 888.xls 150 Opening ASCII mode data connection for 888.xls (57856 Bytes). 226-Maximum disk quota limited to 307200 kBytes Used disk quota 56 kBytes, available 307143 kBytes 226 Transfer complete. QUIT 221 Goodbye!
分析	该图为用户成功登录后，发送连接端口，进行一些列命令操作

二、打开“FTP 数据包”的“ftp 例 2.cap”文件，进行观察分析，回答以下问题

题号	
1	FTP 服务器的 ip 是多少？FTP 客户端的 mac 地址是多少？
答案	服务器的 ip 地址为 172.16.3.240，客户端的 mac 地址为 00:14:2a:20:12:96
截图	> Frame 3: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) v Ethernet II, Src: Elitegro_20:12:96 (00:14:2a:20:12:96), Dst: DigitalC_02:b7:57 (00:03:0f:02:b7:57)
分析	该第三个报文为第一次握手，客户端发向服务器端，根据源地址和目的地址可知
2	该数据包中共有多少个 TCP 流？

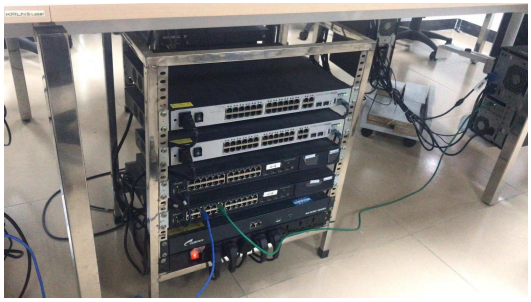
答案	9 个
截图	
分析	通过追踪 TCP 流中可以查看到有 9 个流，
3	最后用什么用户和密码登录成功？
答案	用户：kjdown 密码：kjdown
截图	USER kjdown 331 User name okay, need password. PASS kjdown 230 User logged in, proceed.
分析	由图片 USER PASS 内容最后 logged in 可见
4	该 FTP 的命令连接和数据连接分别是什么？
答案	三次握手报文建立命令连接，数据连接为被动模式
截图	一、命令连接： 171 346.347532 172.16.39.93 172.16.3.240 TCP 62 1454 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM 172 346.347757 172.16.3.240 172.16.39.93 TCP 62 21 → 1454 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM 173 346.347811 172.16.39.93 172.16.3.240 TCP 54 1454 → 21 [ACK] Seq=1 Ack=1 Win=65535 Len=0 174 349.026954 172.16.3.240 172.16.39.93 FTP 79 Response: 220-FTP Server ready... 二、数据连接： 223 398.635312 172.16.3.240 172.16.39.93 TCP 60 21 → 1454 [ACK] Seq=831 Ack=71 Win=65465 Len=0 224 400.851141 172.16.3.240 172.16.39.93 FTP 74 Response: 200 Type set to A. 225 400.933248 172.16.39.93 172.16.3.240 FTP 60 Request: PASV 226 401.048537 172.16.3.240 172.16.39.93 TCP 60 21 → 1454 [ACK] Seq=851 Ack=77 Win=65459 Len=0 227 403.308826 172.16.3.240 172.16.39.93 FTP 102 Response: 227 Entering Passive Mode (172,16,3,240,18,44) 228 403.311489 172.16.39.93 172.16.3.240 TCP 62 1654 → 4652 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM 229 403.312292 172.16.3.240 172.16.39.93 TCP 62 4652 → 1654 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM 230 403.312346 172.16.39.93 172.16.3.240 TCP 54 1654 → 4652 [ACK] Seq=1 Ack=1 Win=65535 Len=0
分析	一、命令连接：第 171、172、173 号报文是客户端使用 1454 号端口向服务端 21 号端口发起三次握手，第 174 号报文中服务端回应客户端，表示形成命令连接，服务端和客户端之间可以使用 ftp 命令。其他的一些命令连接的建立与上述类似。 二、数据连接：第 225 号报文中，客户端请求被动模式，FTP 服务器通过 21 端口返回同意，然后在第 228 至 230 号报文中，服务器开放端口 4652 接受来自客户端的数据连接，客户端则将使用端口 1654 进行数据连接，数据连接建立，可以传输数据。其他的一些数据连接的建立与上述类似。
5	哪几个报文是 FTP 数据连接的三次握手报文？
答案	三次握手报文如下图（第一列为报文序号）
截图	228 403.311489 172.16.39.93 172.16.3.240 TCP 62 1654 → 4652 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM 229 403.312292 172.16.3.240 172.16.39.93 TCP 62 4652 → 1654 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM 230 403.312346 172.16.39.93 172.16.3.240 TCP 54 1654 → 4652 [ACK] Seq=1 Ack=1 Win=65535 Len=0 256 439.360533 172.16.39.93 172.16.3.240 TCP 62 1791 → 1137 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM 257 439.360823 172.16.3.240 172.16.39.93 TCP 62 1137 → 1791 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM 258 439.360876 172.16.39.93 172.16.3.240 TCP 54 1791 → 1137 [ACK] Seq=1 Ack=1 Win=65535 Len=0 286 476.228404 172.16.39.93 172.16.3.240 TCP 62 1934 → 1587 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM 287 476.228638 172.16.3.240 172.16.39.93 TCP 62 1587 → 1934 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM 288 476.228669 172.16.39.93 172.16.3.240 TCP 54 1934 → 1587 [ACK] Seq=1 Ack=1 Win=65535 Len=0
分析	1. 客户端发送 SYN=1 和随机生成的 seq number 到服务器 2. 服务器由 SYN=1 得知客户端想要建立连接，返回 SYN=1、Seq=0、Ack=1、ack nu

	mber(=客户端的 seq+1）和随机生成的 seq number 3. 客户端收到后检查 ack number 是否正确，即第一次发送的 seq number+1，以及 ack 是否为 1，若正确，客户端会再发送 ack number=(服务器的 seq+1)，ack=1，服务器收到后确认 Seq 值与 Ack=1 则连接建立成功。
6	哪几个报文是 FTP 数据连接的挥手报文（结束报文）？
答案	挥手报文如下图（第一列为报文序号）
截图	270 447.419304 172.16.3.240 172.16.39.93 TCP 60 1137 → 1791 [FIN, ACK] Seq=2992 Ack=1 Win=65535 Len=0 271 447.419373 172.16.39.93 172.16.3.240 TCP 54 1791 → 1137 [ACK] Seq=1 Ack=2993 Win=65464 Len=0 272 447.419475 172.16.39.93 172.16.3.240 TCP 54 1791 → 1137 [FIN, ACK] Seq=1 Ack=2993 Win=65464 Len=0 273 447.419643 172.16.3.240 172.16.39.93 TCP 60 1137 → 1791 [ACK] Seq=2993 Ack=2 Win=65535 Len=0 293 476.501474 172.16.3.240 172.16.39.93 TCP 60 1587 → 1934 [FIN, ACK] Seq=1131 Ack=1 Win=65535 Len=0 294 476.501536 172.16.39.93 172.16.3.240 TCP 54 1934 → 1587 [ACK] Seq=1 Ack=1132 Win=64405 Len=0 295 476.541711 172.16.39.93 172.16.3.240 TCP 54 1454 → 21 [ACK] Seq=178 Ack=1362 Win=64174 Len=0 296 476.561030 172.16.39.93 172.16.3.240 TCP 54 1934 → 1587 [FIN, ACK] Seq=1 Ack=1132 Win=64405 Len=0 297 476.561201 172.16.3.240 172.16.39.93 TCP 60 1587 → 1934 [ACK] Seq=1132 Ack=2 Win=65535 Len=0 620 534.787848 172.16.3.240 172.16.39.93 TCP 60 2118 → 2097 [FIN, ACK] Seq=239105 Ack=1 Win=65535 Len=0 621 534.787917 172.16.39.93 172.16.3.240 TCP 54 2097 → 2118 [ACK] Seq=1 Ack=239106 Win=65535 Len=0 622 534.788371 172.16.39.93 172.16.3.240 TCP 54 2097 → 2118 [FIN, ACK] Seq=1 Ack=239106 Win=65535 Len=0 623 534.789817 172.16.3.240 172.16.39.93 TCP 60 2118 → 2097 [ACK] Seq=239106 Ack=2 Win=65535 Len=0
分析	一：客户端发送 FIN 报文和 Seq =k 二：服务端发送 ACK 报文，Ack=k+1 表示确认收到，之后客户端到服务端的连接释放 三：服务器发送 FIN 报文，Seq=w、Ack=k+1，等待用户确认 四：客户发送 ACK 报文，Ack=w+1 应答，随后经过一定时间进入关闭状态（其中服务器端口为 PASV 命令时提供）
7	该 FTP 的连接模式是那种？为什么？
答案	被动模式
截图	Request: PASV 21 → 1454 [ACK] Seq=85 Response: 227 Entering 1654 → 4652 [SYN] Seq= 4652 → 1654 [SYN, ACK]
分析	使用 PASV 命令，要求服务器新建端口发送给客户，待客户端连接，可见之后的服务器数据连接端口不再是 20

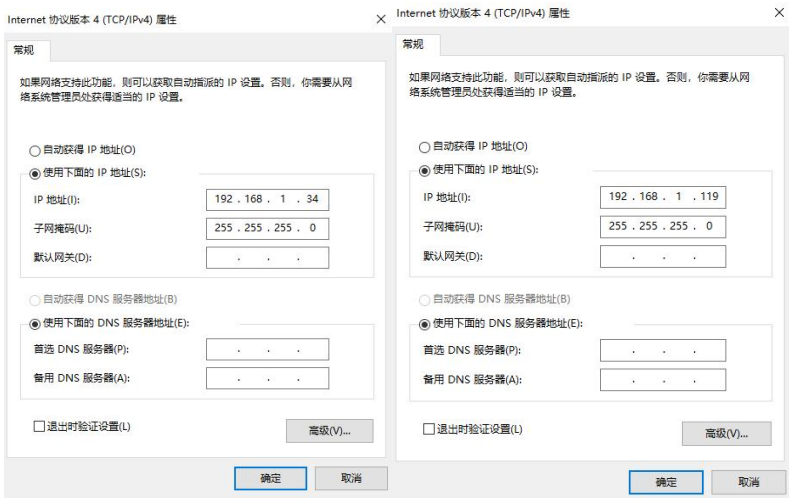
三、在线捕获数据包实验

1. 阅读教材 P64-69 内容，熟悉 FTP 协议。
2. 参考 p67 图 2-10 搭建 FTP 实验拓扑，建议采用 FileZilla 客户端和服务器端 (https://filezilla-project.org/)

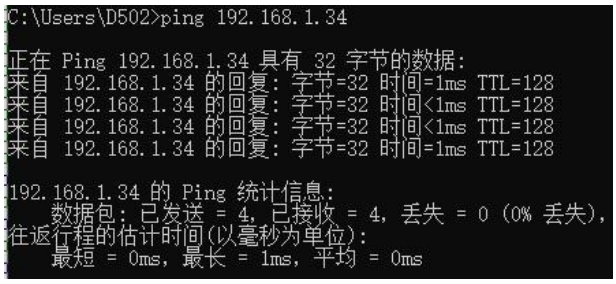
(1) 交换机与两主机之间连接网线



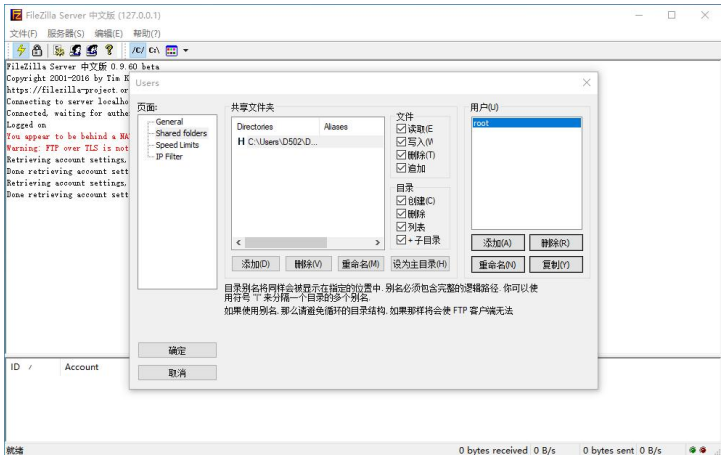
(2) 关闭防火墙，配置主机网卡



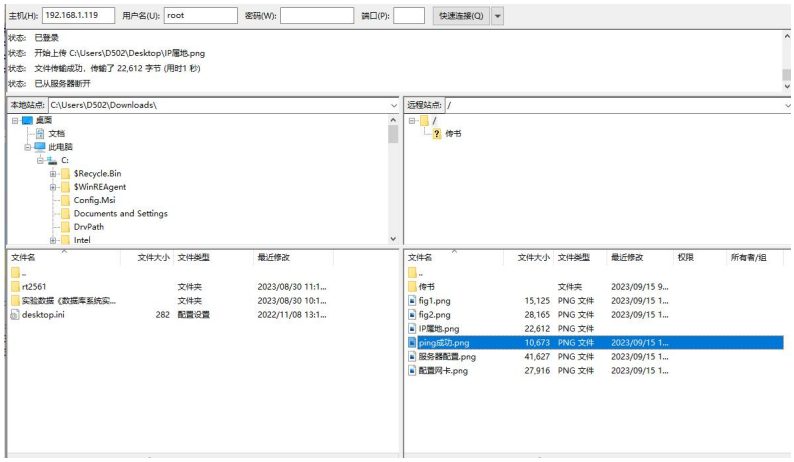
(3) 尝试互 ping，成功



(4) 服务端下载 FileZilla Server，配置用户信息及传输文件夹



(5) 客户端根据设置的用户进行连接



(6) wireshark 抓包分析（服务端）

抓包期间操作，由于客户端已连接上，先进行断开连接，重新连入服务器，后续进行图片的下载上传操作抓包主要分析重连之后操作，下图为该期间 TCP 流，客户端通过 USER PASS 登录（PASS 设置为空），CWD 跳到指定共享文件夹，TYPE I 转换文件传输类型，执行 PASV 被动连接数据端口，RETR 下载服务器文件，STOR 上传文件到服务器。

```
220-FileZilla Server ..... 0.9.60 beta
220-written by Tim Kosse (tim.kosse@filezilla-project.org)
220 Please visit https://filezilla-project.org/
AUTH TLS
502 Explicit TLS authentication not allowed
AUTH SSL
502 Explicit TLS authentication not allowed
USER root
331 Password required for root
PASS
230 Logged on
CWD /
250 CWD successful. "/" is current directory.
TYPE I
200 Type set to I
PASV
227 Entering Passive Mode (192,168,1,119,252,252)
RETR .............png
150 Opening data channel for file download from server of "/.............png"
226 Successfully transferred "/.............png"
PASV
227 Entering Passive Mode (192,168,1,119,243,78)
RETR ...............png
150 Opening data channel for file download from server of "/...............png"
226 Successfully transferred "/...............png"
PASV
227 Entering Passive Mode (192,168,1,119,221,61)
STOR ...............1.png
150 Opening data channel for file upload to server of "/...............1.png"
226 Successfully transferred "/...............1.png"
```

三次握手建立连接

```
12 6.730771   192.168.1.34   192.168.1.119  TCP   66 55109 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
13 6.730829   192.168.1.119  192.168.1.34   TCP   66 21 → 55109 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
14 6.731363   192.168.1.34   192.168.1.119  TCP   60 55109 → 21 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
```

命令连接

```
31 12.703238  192.168.1.34   192.168.1.119  TCP   66 55110 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
32 12.703390  192.168.1.119  192.168.1.34   TCP   66 21 → 55110 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
33 12.703862  192.168.1.34   192.168.1.119  TCP   60 55110 → 21 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
```

数据连接

```
47 12.713205  192.168.1.34   192.168.1.119  FTP   60 Request: PASV
48 12.713397  192.168.1.119  192.168.1.34   FTP   105 Response: 227 Entering Passive Mode (192,168,1,119,252,252)
49 12.713835  192.168.1.34   192.168.1.119  FTP   77 Request: RETR 配置网卡.png
50 12.713904  192.168.1.34   192.168.1.119  TCP   66 55111 → 64764 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=128 SACK_PERM
51 12.713963  192.168.1.119  192.168.1.34   TCP   66 64764 → 55111 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
52 12.714151  192.168.1.34   192.168.1.119  TCP   60 55111 → 64764 [ACK] Seq=1 Ack=1 Win=4194304 Len=0
```

四次挥手

```
159 68.081692  192.168.1.34   192.168.1.119  TCP   60 55109 → 21 [FIN, ACK] Seq=44 Ack=322 Win=2102016 Len=0
160 68.081782  192.168.1.119  192.168.1.34   TCP   54 21 → 55109 [ACK] Seq=322 Ack=45 Win=2102272 Len=0
161 68.081984  192.168.1.119  192.168.1.34   TCP   54 21 → 55109 [FIN, ACK] Seq=322 Ack=45 Win=2102272 Len=0
162 68.082333  192.168.1.34   192.168.1.119  TCP   60 55109 → 21 [ACK] Seq=45 Ack=323 Win=2102016 Len=0
```

3.完成 P51 的实例 2-1。

(1)捕获了 140 的数据量

```
132 25.629182  Hangzhou_80:da:75  Spanning-tree-(for-… STP    119 MST. Root = 32768/0/1c:20:db:5c:a3:10  Cost = 24020  Port = 0x800d
133 26.283419  172.16.26.2        239.255.255.250      SSDP   217 M-SEARCH * HTTP/1.1
134 26.539194  172.16.26.1        172.16.0.1           UDP    106 5598 → 5526 Len=64
135 26.579266  172.16.6.1         172.16.255.255       UDP   1482 54471 → 1689 Len=1440
136 26.742294  172.16.26.1        51.105.71.137        TCP     55 55058 → 443 [ACK] Seq=1 Ack=1 Win=512 Len=1 [TCP segment of a reassembled PDU]
137 27.036366  51.105.71.137      172.16.26.1          TCP     66 443 → 55058 [ACK] Seq=1 Ack=2 Win=2049 Len=0 SLE=1 SRE=2
138 27.086666  172.18.186.93      239.255.255.250      SSDP   215 M-SEARCH * HTTP/1.1
139 27.295266  172.16.26.2        239.255.255.250      SSDP   217 M-SEARCH * HTTP/1.1
140 27.629166  Hangzhou_80:da:75  Spanning-tree-(for-… STP    119 MST. Root = 32768/0/1c:20:db:5c:a3:10  Cost = 24020  Port = 0x800d
```

可见最后一行的数据标号为 140

(2)

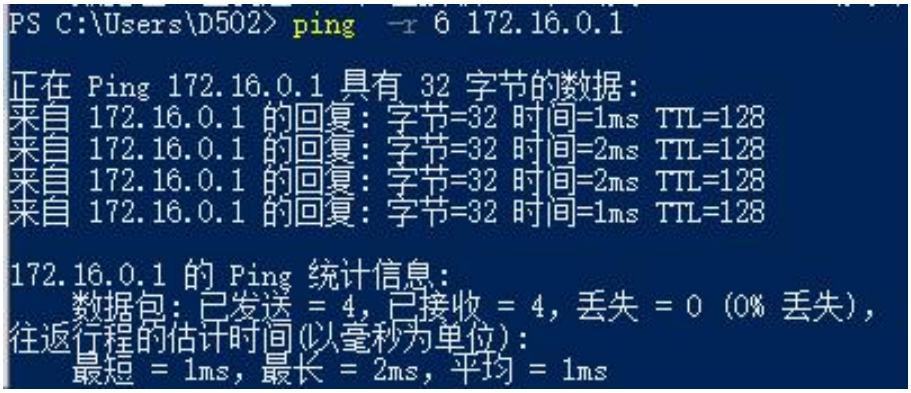
本机 ip 地址及归属地

本机发送出去的目的 ip

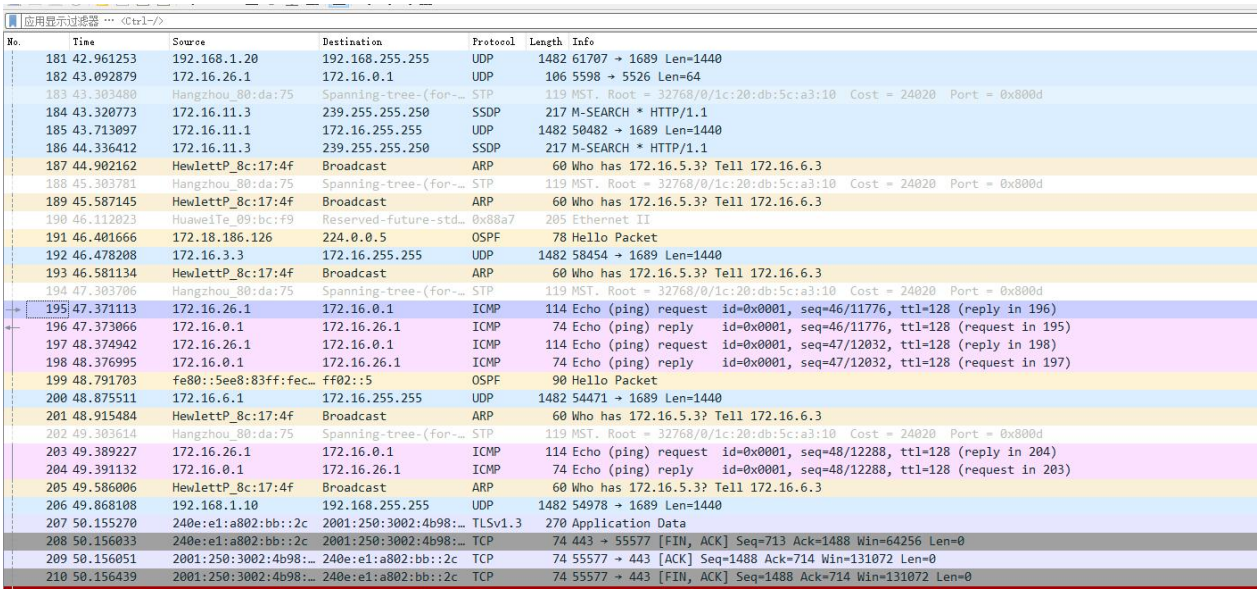
```
> Frame 2: 75 bytes on wire (600 bits), 75 bytes captured (600 bits) on interface \Device\NPF_{4F39B87B-6364-4739-BB4
> Ethernet II, Src: HewlettP_88:b0:8e (18:60:24:88:b0:8e), Dst: HuaweiTe_c4:ec:e4 (5c:e8:83:c4:ec:e4)
v Internet Protocol Version 6, Src: 2001:250:3002:4b98:ad27:e034:4274:5fd4, Dst: 2409:8c54:b000:710:8000::d
    0110 .... = Version: 6
  > .... 0000 0000 .... .... .... .... .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
    .... 1111 1101 1100 0111 1111 = Flow Label: 0xfdc7f
    Payload Length: 21
    Next Header: TCP (6)
    Hop Limit: 64
    Source Address: 2001:250:3002:4b98:ad27:e034:4274:5fd4
    Destination Address: 2409:8c54:b000:710:8000::d
```

查询其归属地为广东移动

（3）执行指令

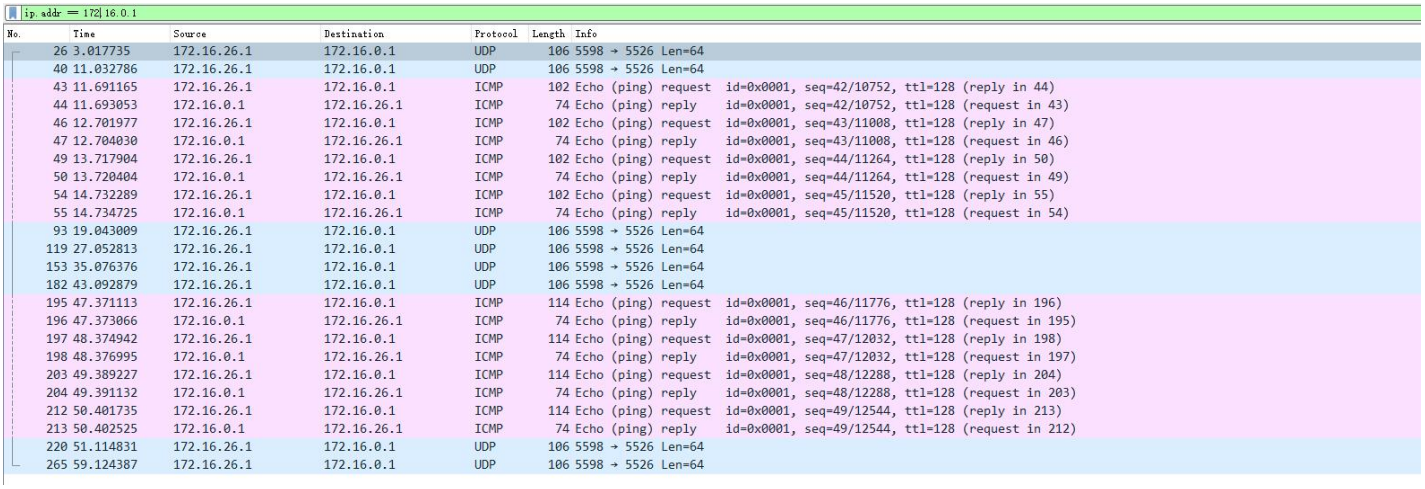


用 wireshark 捕获数据包



No.	Time	Source	Destination	Protocol	Length	Info
181	42.961253	192.168.1.20	192.168.255.255	UDP	1482	61707 → 1689 Len=1440
182	43.092879	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
183	43.303480	Hangzhou_80:da:75	Spanning-tree-(for-…	STP	119	MST. Root = 32768/0/1c:20:db:5c:a3:10 Cost = 24020 Port = 0x800d
184	43.320773	172.16.11.3	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
185	43.713097	172.16.11.1	172.16.255.255	UDP	1482	50482 → 1689 Len=1440
186	44.336412	172.16.11.3	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
187	44.902162	HewlettP_8c:17:4f	Broadcast	ARP	60	Who has 172.16.5.3? Tell 172.16.6.3
188	45.303781	Hangzhou_80:da:75	Spanning-tree-(for-…	STP	119	MST. Root = 32768/0/1c:20:db:5c:a3:10 Cost = 24020 Port = 0x800d
189	45.587145	HewlettP_8c:17:4f	Broadcast	ARP	60	Who has 172.16.5.3? Tell 172.16.6.3
190	46.112023	HuaweiTe_09:bc:f9	Reserved-future-std…	0x88a7	205	Ethernet II
191	46.401666	172.18.186.126	224.0.0.5	OSPF	78	Hello Packet
192	46.478208	172.16.3.3	172.16.255.255	UDP	1482	58454 → 1689 Len=1440
193	46.581134	HewlettP_8c:17:4f	Broadcast	ARP	60	Who has 172.16.5.3? Tell 172.16.6.3
194	47.303706	Hangzhou_80:da:75	Spanning-tree-(for-…	STP	119	MST. Root = 32768/0/1c:20:db:5c:a3:10 Cost = 24020 Port = 0x800d
195	47.371113	172.16.26.1	172.16.0.1	ICMP	114	Echo (ping) request id=0x0001, seq=46/11776, ttl=128 (reply in 196)
196	47.373066	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=46/11776, ttl=128 (request in 195)
197	48.374942	172.16.26.1	172.16.0.1	ICMP	114	Echo (ping) request id=0x0001, seq=47/12032, ttl=128 (reply in 198)
198	48.376995	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=47/12032, ttl=128 (request in 197)
199	48.791703	fe80::5ee8:83ff:fec…	ff02::5	OSPF	90	Hello Packet
200	48.875511	172.16.6.1	172.16.255.255	UDP	1482	54471 → 1689 Len=1440
201	48.915484	HewlettP_8c:17:4f	Broadcast	ARP	60	Who has 172.16.5.3? Tell 172.16.6.3
202	49.303614	Hangzhou_80:da:75	Spanning-tree-(for-…	STP	119	MST. Root = 32768/0/1c:20:db:5c:a3:10 Cost = 24020 Port = 0x800d
203	49.389227	172.16.26.1	172.16.0.1	ICMP	114	Echo (ping) request id=0x0001, seq=48/12288, ttl=128 (reply in 204)
204	49.391132	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=48/12288, ttl=128 (request in 203)
205	49.586006	HewlettP_8c:17:4f	Broadcast	ARP	60	Who has 172.16.5.3? Tell 172.16.6.3
206	49.868108	192.168.1.10	192.168.255.255	UDP	1482	54978 → 1689 Len=1440
207	50.155270	240e:e1:a802:bb::2c	2001:250:3002:4b98:…	TLSv1.3	270	Application Data
208	50.156033	240e:e1:a802:bb::2c	2001:250:3002:4b98:…	TCP	74	443 → 55577 [FIN, ACK] Seq=713 Ack=1488 Win=64256 Len=0
209	50.156051	2001:250:3002:4b98:…	240e:e1:a802:bb::2c	TCP	74	55577 → 443 [ACK] Seq=1488 Ack=714 Win=131072 Len=0
210	50.156439	2001:250:3002:4b98:…	240e:e1:a802:bb::2c	TCP	74	55577 → 443 [FIN, ACK] Seq=1488 Ack=714 Win=131072 Len=0

（4）执行 ip.addr==172.16.0.1



ip.addr == 172.16.0.1

No.	Time	Source	Destination	Protocol	Length	Info
26	3.017735	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
40	11.032786	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
43	11.691165	172.16.26.1	172.16.0.1	ICMP	102	Echo (ping) request id=0x0001, seq=42/10752, ttl=128 (reply in 44)
44	11.693053	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=42/10752, ttl=128 (request in 43)
46	12.701977	172.16.26.1	172.16.0.1	ICMP	102	Echo (ping) request id=0x0001, seq=43/11008, ttl=128 (reply in 47)
47	12.704030	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=43/11008, ttl=128 (request in 46)
49	13.717904	172.16.26.1	172.16.0.1	ICMP	102	Echo (ping) request id=0x0001, seq=44/11264, ttl=128 (reply in 50)
50	13.720404	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=44/11264, ttl=128 (request in 49)
54	14.732289	172.16.26.1	172.16.0.1	ICMP	102	Echo (ping) request id=0x0001, seq=45/11520, ttl=128 (reply in 55)
55	14.734725	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=45/11520, ttl=128 (request in 54)
93	19.043009	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
119	27.052813	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
153	35.076376	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
182	43.092879	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
195	47.371113	172.16.26.1	172.16.0.1	ICMP	114	Echo (ping) request id=0x0001, seq=46/11776, ttl=128 (reply in 196)
196	47.373066	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=46/11776, ttl=128 (request in 195)
197	48.374942	172.16.26.1	172.16.0.1	ICMP	114	Echo (ping) request id=0x0001, seq=47/12032, ttl=128 (reply in 198)
198	48.376995	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=47/12032, ttl=128 (request in 197)
203	49.389227	172.16.26.1	172.16.0.1	ICMP	114	Echo (ping) request id=0x0001, seq=48/12288, ttl=128 (reply in 204)
204	49.391132	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=48/12288, ttl=128 (request in 203)
212	50.401735	172.16.26.1	172.16.0.1	ICMP	114	Echo (ping) request id=0x0001, seq=49/12544, ttl=128 (reply in 213)
213	50.402525	172.16.0.1	172.16.26.1	ICMP	74	Echo (ping) reply id=0x0001, seq=49/12544, ttl=128 (request in 212)
220	51.114831	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64
265	59.124387	172.16.26.1	172.16.0.1	UDP	106	5598 → 5526 Len=64

(5) 协议有 UDP 和 ICMP

Echo 请求与响应

request 为请求，reply 为响应

```
 43 11.691165    172.16.26.1     172.16.0.1      ICMP     102 Echo (ping) request  id=0x0001, seq=42/10752, ttl=128 (reply in 44)
 44 11.693053    172.16.0.1      172.16.26.1     ICMP      74 Echo (ping) reply    id=0x0001, seq=42/10752, ttl=128 (request in 43)
```

Stamp 的请求与响应

```
195 47.371113    172.16.26.1     172.16.0.1      ICMP     114 Echo (ping) request  id=0x0001, seq=46/11776, ttl=128 (reply in 196)
196 47.373066    172.16.0.1      172.16.26.1     ICMP      74 Echo (ping) reply    id=0x0001, seq=46/11776, ttl=128 (request in 195)
```

[1]第一层 Frame 给出了数据帧的全局信息，显示了包括帧长，帧到达的时间，接口的编号和帧的类型。

```
Frame 43: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface \Device\NPF_{4F39B87B-6364-4739-I
  Section number: 1
> Interface id: 0 (\Device\NPF_{4F39B87B-6364-4739-BB47-0DC1D5EF56F2})
  Encapsulation type: Ethernet (1)
  Arrival Time: Sep 15, 2023 11:19:58.218479000 中国标准时间
  [Time shift for this packet: 0.000000000 seconds]
  Epoch Time: 1694747998.218479000 seconds
  [Time delta from previous captured frame: 0.155063000 seconds]
  [Time delta from previous displayed frame: 0.658379000 seconds]
  [Time since reference or first frame: 11.691165000 seconds]
  Frame Number: 43
  Frame Length: 102 bytes (816 bits)
  Capture Length: 102 bytes (816 bits)
  [Frame is marked: False]
  [Frame is ignored: False]
  [Protocols in frame: eth:ethertype:ip:icmp:data]
  [Coloring Rule Name: ICMP]
  [Coloring Rule String: icmp || icmpv6]
```

[2] 第二行显示了数据帧头部信息，可以看到以太帧头部包括的三个字段，目的 MAC 地址，源 MAC 地址、类型字段，类型字段取值为十六进制的 0800，说明数据帧中包含的是一个 IP 分组。

```
Ethernet II, Src: HewlettP_88:b0:8e (18:60:24:88:b0:8e), Dst: Dell_e2:5a:34 (a4:bb:6d:e2:5a:34)
> Destination: Dell_e2:5a:34 (a4:bb:6d:e2:5a:34)
> Source: HewlettP_88:b0:8e (18:60:24:88:b0:8e)
  Type: IPv4 (0x0800)
```

[3] 第三行显示了 IP 分组头部信息，包括版本号 4，头部长度 48 字节，服务类型，数据报总长度，用于分片的标志字 0，分片偏移字段 0。生存周期 128，表示最多允许经过 128 跳路由器的转发。协议字段 1，说明 IP 分组里面封装的是一个 ICMP 报文，头部校验、源 IP 地址、目的 IP 地址。

```
Internet Protocol Version 4, Src: 172.16.26.1, Dst: 172.16.0.1
  0100 .... = Version: 4
  .... 1100 = Header Length: 48 bytes (12)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 88
  Identification: 0x4e49 (20041)
> 000. .... = Flags: 0x0
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 128
  Protocol: ICMP (1)
  Header Checksum: 0x0000 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 172.16.26.1
  Destination Address: 172.16.0.1
> Options: (28 bytes), Record Route
```

[4] 第四行是 ICMP 的协议报文，具体内容是类型 8，Code：0 表明这是一个 Echo (ping) request 也就是一个回应请求报文，校验和字段，这三个字段是所有 ICMP 报文的通用首部,下面的标识字段（Identifier (BE)）和序号字段（Sequence Number (BE)）,标识代表的是当前运行的 ping 进程的标识、序号字段代表 ping 生成的 ICMP 报文的编号。Data 字段中，包含了 32 个字节的随机生成的数据。

计算机网络实验报告

```
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x4d31 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 42 (0x002a)
    Sequence Number (LE): 10752 (0x2a00)
    [Response frame: 44]
  > Data (32 bytes)
```

本次实验完成后，请根据组员在实验中的贡献，请实事求是，自评在实验中应得的分数。（按百分制）

学号	学生	自评分
21307347	陈欣宇	95
21307350	高宇	94
21307100	陈华清	95