计算机网络实验报告

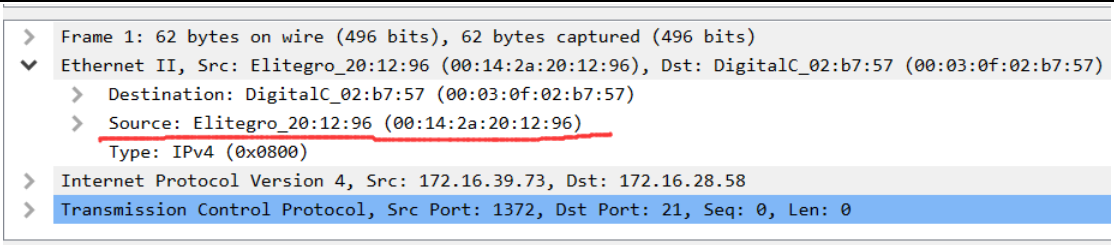
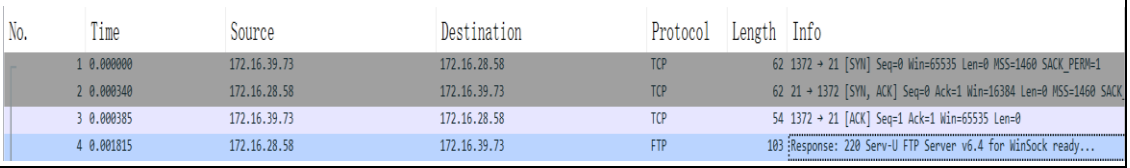
警示

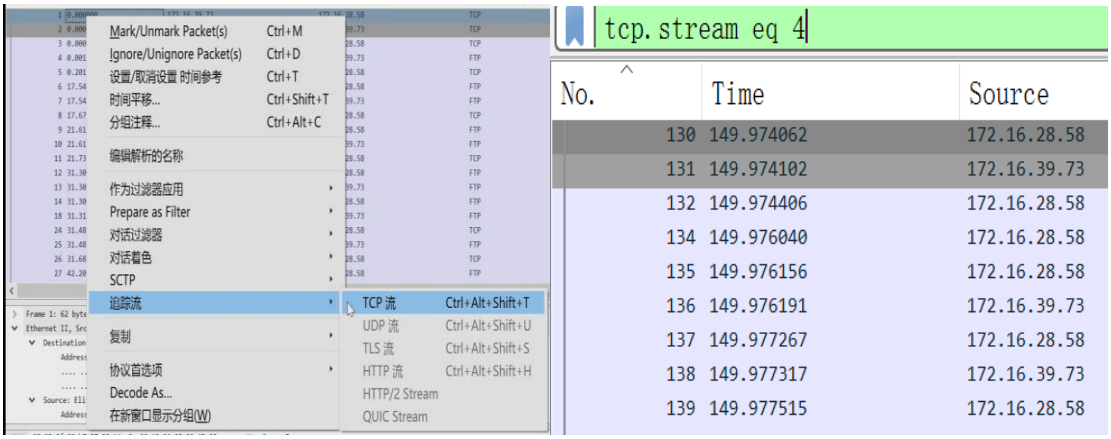
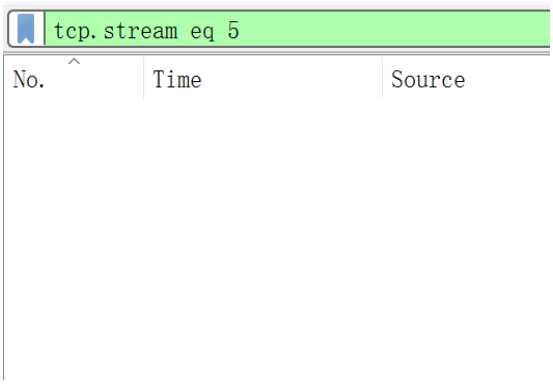
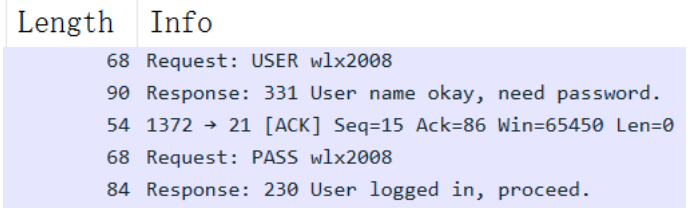
1. 实验报告如有雷同，雷同各方当次实验成绩均以 0 分计。
2. 当次小组成员成绩只计学号、姓名登录在下表中的。
3. 在规定时间内未上交实验报告的，不得以其他方式补交，当次成绩按 0 分计。
4. 实验报告文件以 PDF 格式提交。

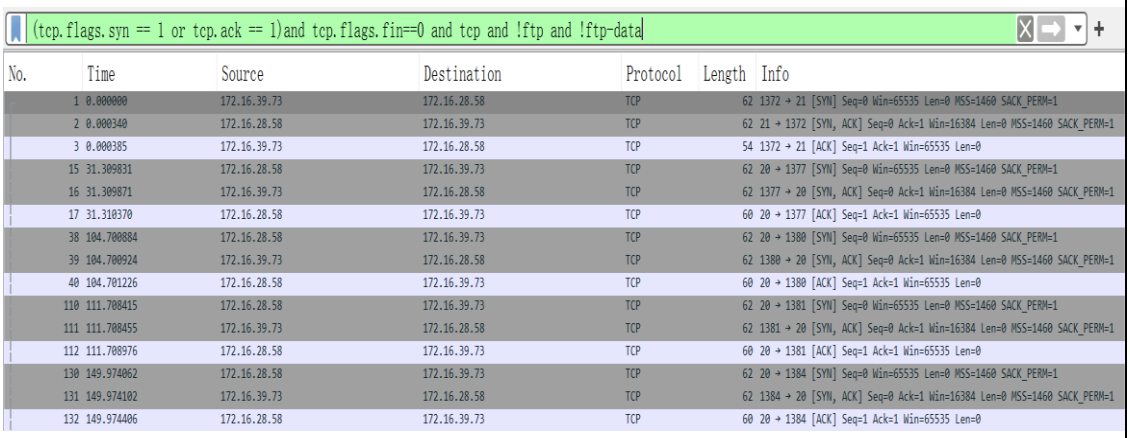
院系	计算机学院	班 级	计科 1 班	组长	郝裕玮
学号	18329015	18325071	19335153		
学生	郝裕玮	张闯	马淙升		

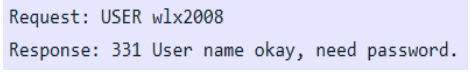
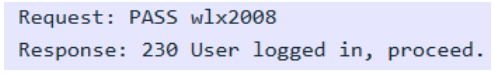
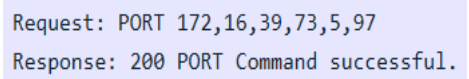
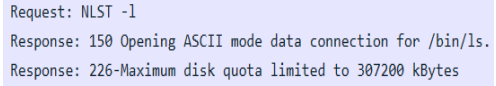
Ftp 协议分析实验

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	
分析	根据框中划线信息即可确定 FTP 客户端（即 source）的 mac 地址
2	第 1、2、3 号报文的作用是什么？
答案	1-3 号报文代表 3 次握手，使得客户端和服务端建立连接。 1-3 号报文的作用见分析。
截图	
分析	1 号报文：客户端发送请求（SYN），向服务端申请建立连接，客户端状态由 closed 变为 syn_send； 2 号报文：服务端确认第一次握手的报文段，返回 SYN+ACK，同意建立连接，服务器状态由 listen 变为 syn_received； 3 号报文：客户端发送确认报文段，返回 ACK，客户端状态变为：established(完成连接)； 补充：最后，也即 4 号报文：服务器收到确认报文段，服务器状态由 syn_received 变为 established(完成连接)

3	该数据包中共有多少个 TCP 流？
答案	共有 5 个 TCP 流
截图	 
分析	利用追踪流中的 TCP 流选项即可统计出共多少条 TCP 流，同时利用 tcp.stream eq 指令也可验证 TCP 流的序号为 0-4.没有 5，即共 5 条 TCP 流（同时该文件中也正好只有 5 次握手，与 TCP 流数量相等）
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	根据 Info 即可确认用户和密码
5	该 FTP 的命令连接和数据连接分别是什么样的连接？
答案	命令连接：传输用户名和密码的连接（服务器端口为 21 的连接）

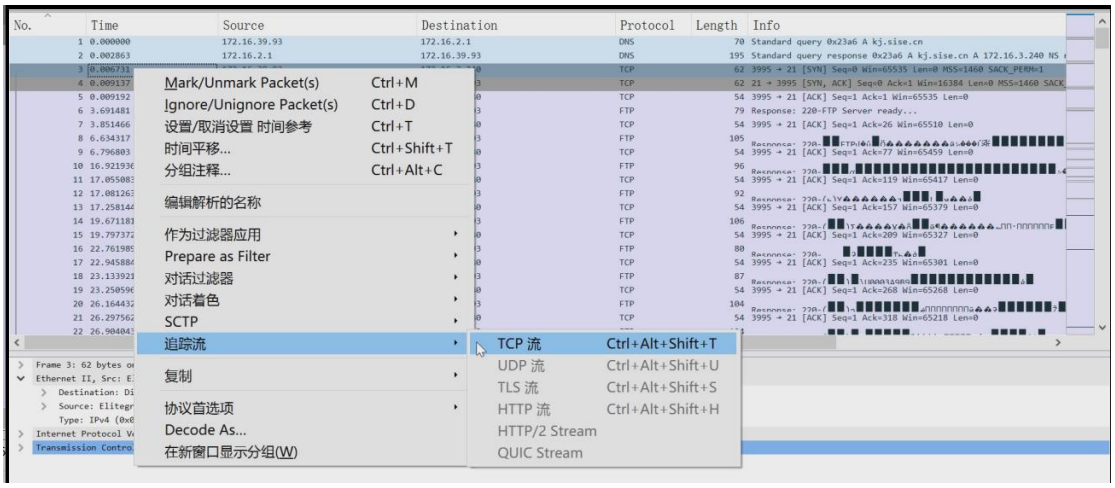
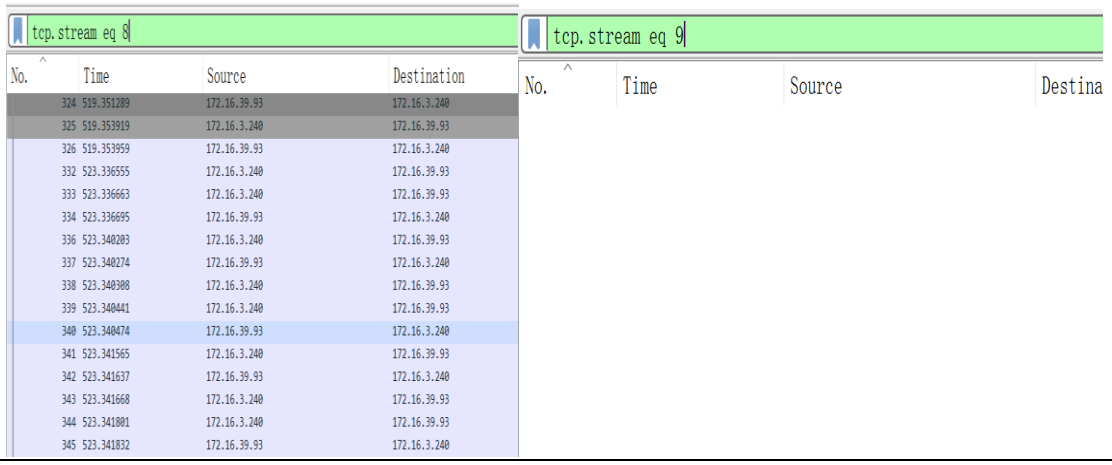
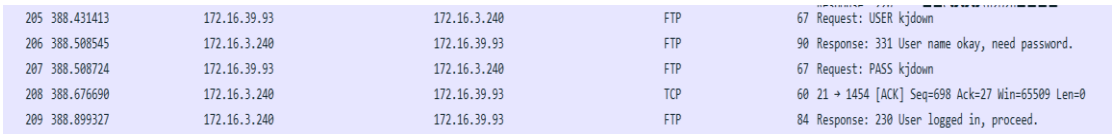
	数据连接：传输用户名和密码的连接之后（即用户 wlx2008 登陆成功之后）的连接都是数据连接（客户端端口为 20 的连接）
截图	 No. Time Source Destination Protocol Length Info 1 0.000000 172.16.39.73 172.16.28.58 TCP 62 1372 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 2 0.000340 172.16.28.58 172.16.39.73 TCP 62 21 → 1372 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 3 0.000385 172.16.39.73 172.16.28.58 TCP 54 1372 → 21 [ACK] Seq=1 Ack=1 Win=65535 Len=0 15 31.309831 172.16.28.58 172.16.39.73 TCP 62 20 → 1377 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 16 31.309871 172.16.39.73 172.16.28.58 TCP 62 1377 → 20 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 17 31.310370 172.16.28.58 172.16.39.73 TCP 60 20 → 1377 [ACK] Seq=1 Ack=1 Win=65535 Len=0 38 104.700884 172.16.28.58 172.16.39.73 TCP 62 20 → 1380 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 39 104.700924 172.16.39.73 172.16.28.58 TCP 62 1380 → 20 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 40 104.701226 172.16.28.58 172.16.39.73 TCP 60 20 → 1380 [ACK] Seq=1 Ack=1 Win=65535 Len=0 110 111.708415 172.16.28.58 172.16.39.73 TCP 62 20 → 1381 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 111 111.708455 172.16.39.73 172.16.28.58 TCP 62 1381 → 20 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 112 111.708976 172.16.28.58 172.16.39.73 TCP 60 20 → 1381 [ACK] Seq=1 Ack=1 Win=65535 Len=0 130 149.974062 172.16.28.58 172.16.39.73 TCP 62 20 → 1384 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 131 149.974102 172.16.39.73 172.16.28.58 TCP 62 1384 → 20 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 132 149.974406 172.16.28.58 172.16.39.73 TCP 60 20 → 1384 [ACK] Seq=1 Ack=1 Win=65535 Len=0
分析	通过过滤器可对数据包进行筛选，截图中的过滤条件可翻译为：选出 TCP 协议中 SYN==1 或者 Ack==1（Acknowledgement Number 确认编号，而非确认值 ACK）并且同时 FIN==0 的数据包。这样的数据包就是我们需要的每次连接中的三次握手的数据包。并根据用户 wlx2008 登录成功前后来区分命令连接和数据连接。 命令连接：No.1-3; 数据连接：No.15-17,No.38-40,No.110-112,No.130-132;
6	该 FTP 的连接模式是哪种？为什么？
答案	主动模式，因为使用了 PORT 命令和固定端口 21.
截图	11 21.733350 172.16.39.73 172.16.28.58 TCP 54 1372 → 21 [ACK] Seq=29 Ack=116 Win=65420 Len=0 12 31.305692 172.16.39.73 172.16.28.58 FTP 78 Request: PORT 172,16,39,73,5,97
分析	每次客户端需要接收数据时，都会向服务器端的 21 端口发送 PORT 命令。
7	最后四个报文的作用是什么？
答案	四次挥手，使客户端与服务端断开连接。 作用详见分析。
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	第一次挥手：客户端发出连接释放报文，并且停止发送数据，返回 FIN+ACK。 第二次挥手：服务端收到 FIN 之后，会发送 ACK 报文，此时服务端处于 CLOSE_WAIT 状态。 第三次挥手：若服务端没有要向客户端发出的数据，则服务端发出连接释放报文段(FIN+ACK)，服务端进入 LAST_ACK（最后确认）状态，等待客户端的确认。 第四次挥手：客户端收到服务端的连接释放报文段后，对此发出确认报文段（ACK=1），客户端进入 TIME_WAIT（时间等待）状态。此时 TCP 未释放掉，需要经过时间等待计时器设置的

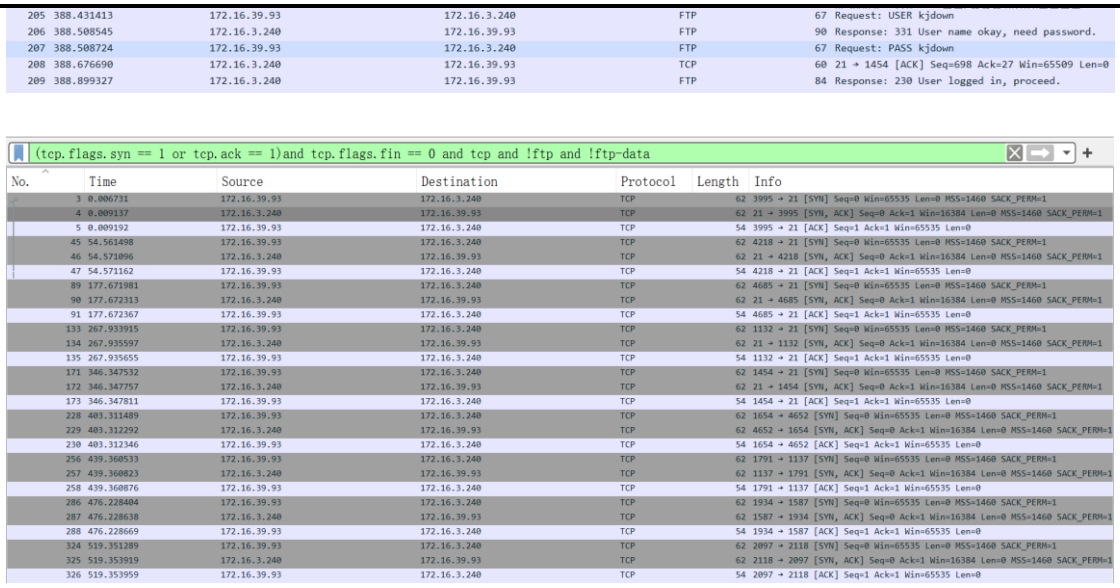
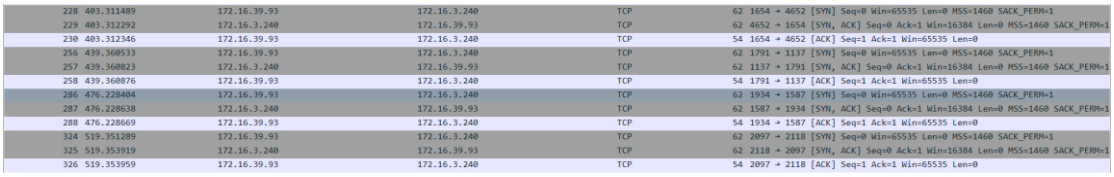
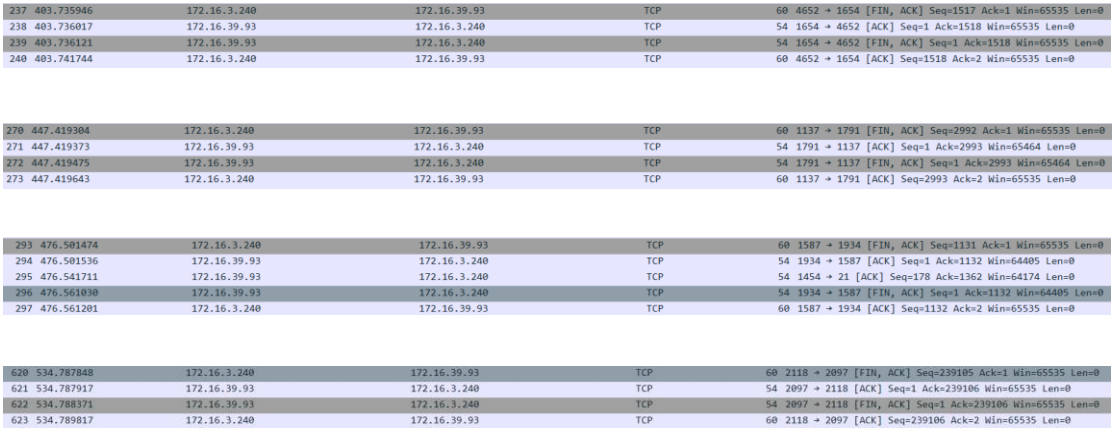
	时间 2MSL 后，客户端才进入 CLOSED 状态。
8	该数据包中有多少个 ftp 的命令及应答，其含义分别是什么？
答案	共 15 条命令及其应答： （1）NLST -l:返回指定路径下的目录列表，省略<路径>时，返回当前目录； （2）PASS wlx2008:向服务器发送密码 wlx2008； （3）PORT 172,16,39,73,5,100:主动模式，包含了客户端用什么端口接收数据； （4）PORT 172,16,39,73,5,101:与（3）相同； （5）PORT 172,16,39,73,5,104:与（3）相同； （6）PORT 172,16,39,73,5,97:与（3）相同； （7）QUIT:与服务端断开连接； （8）RETR 888.xls:下载文件（888.xls）； （9）RNFR jjj:准备对文件名为 jjj 的文件进行重命名（只是指定需要改名的文件，但尚未修改，修改名字需要 RNTO 命令来实现）； （10）RNFR xs2009-9.xls:同（9）； （11）RNTO 888.xls:和（10）相配对，将文件名改为 888.xls； （12）RNTO ppp:和（9）相配对，将文件名改为 ppp； （13）STOR xs2009-9.xls:上传文件 xs2009-9.xls； （14）USER wlx2008:向服务器发送用户名 wlx2008； （15）XMKD jjj:在当前目录下建立”/jjj”文件夹 补充：与每条命令所对应的应答内容会和命令一起放在截图里，通过应答内容也可以进一步确认每条命令的含义。
截图	   

	Request: XMKD jjj Response: 257 "/jjj" directory created. Request: RNFR jjj Response: 350 File or directory exists, ready for destination name Request: RNTO ppp Response: 250 RNTO command successful. Request: PORT 172,16,39,73,5,100 Response: 200 PORT Command successful. Request: STOR xs2009-9.xls Response: 150 Opening ASCII mode data connection for xs2009-9.xls. Response: 226-Maximum disk quota limited to 307200 kBytes Request: PORT 172,16,39,73,5,101 Response: 200 PORT Command successful. Request: NLST -l Response: 150 Opening ASCII mode data connection for /bin/ls. Response: 226-Maximum disk quota limited to 307200 kBytes Request: RNFR xs2009-9.xls Response: 350 File or directory exists, ready for destination name Request: RNTO 888.xls Response: 250 RNTO command successful. Request: PORT 172,16,39,73,5,104 Response: 200 PORT Command successful. Request: RETR 888.xls Response: 150 Opening ASCII mode data connection for 888.xls (57856 Bytes). Response: 226-Maximum disk quota limited to 307200 kBytes Request: QUIT Response: 221 Goodbye!
分析	根据每条命令本身内容和对应的回应内容即可分析出其含义

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 IP：172.16.3.240 客户端 mac 地址：00:14:2a:20:12:96;
截图	Frame 3: 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:b7:57 (00:03:0f:02:b7:57) Destination: DigitalC_02:b7:57 (00:03:0f:02:b7:57) Source: Elitegro_20:12:96 (00:14:2a:20:12:96) Type: IPv4 (0x0800) Internet Protocol Version 4, Src: 172.16.39.93, Dst: 172.16.3.240 Transmission Control Protocol, Src Port: 3995, Dst Port: 21, Seq: 0, Len: 0
分析	详见截图
2	该数据包中共有多少个 TCP 流？
答案	共有 9 个 TCP 流

截图	 
分析	利用追踪流中的 TCP 流选项即可统计出共多少条 TCP 流，同时利用 tcp.stream eq 指令也可验证 TCP 流的序号为 0-8.没有 9，即共 9 条 TCP 流（同时该文件中也正好只有 9 次握手挥手，与 TCP 流数量相等）
3	最后用什么用户和密码登录成功？
答案	用户名：kjdown 密码：kjdown
截图	
分析	详见截图
4	该 FTP 的命令连接和数据连接分别是什么？
答案	命令连接：用户 kjdown 登录成功之前（即 No.205 之前）的 5 次服务端端口为 21 的连接均是命令连接。这 5 次分别为 No.3-5,No.45-47,No.89-91,No.133-135,No.171-173; 数据连接：用户 kjdown 登录成功之后的 4 次连接都是数据连接。这 4 次分别为 No.228-230, No.256-258,No.286-288,No.324-326

截图	
分析	通过过滤器可对数据包进行筛选，截图中的过滤条件可翻译为：选出 TCP 协议中 SYN==1 或者 Ack==1（Acknowledgement Number 确认编号，而非确认值 ACK）并且同时 FIN==0 的数据包。这样的数据包就是我们需要的每次连接中的三次握手的数据包。并根据用户 kjdown 登录成功前后来区分命令连接和数据连接。
5	哪几个报文是 FTP 数据连接的三次握手报文？
答案	No.228-230,No.256-258,No.286-288,No.324-326;
截图	
分析	利用 4 的过滤条件和截图结果即可得到本题答案（即用户 kjdown 登录成功之后的 4 次连接都是数据连接。这 4 次分别为 No.228-230,No.256-258,No.286-288,No.324-326）
6	哪几个报文是 FTP 数据连接的挥手报文（结束报文）？
答案	No.237-240,No.270-273,No.293-297,No.620-623;
截图	
分析	详见截图即可。
7	该 FTP 的连接模式是哪种？为什么？

答案	被动模式，因为使用了 PASV 命令
截图	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)
分析	主动模式使用 PORT 命令和固定的 21 端口，被动模式使用 PASV 命令和随机端口。

三、在线捕获数据包实验

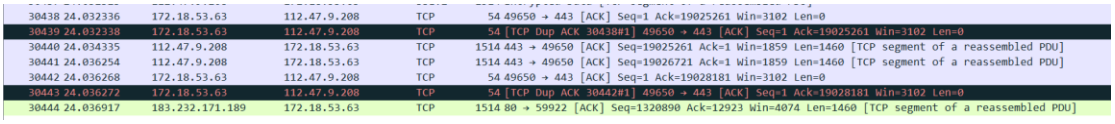
1. 阅读教材 P64-69 内容，熟悉 FTP 协议。

答：已阅读

2. 完成 P51 的实例 2-1。

（1）单击 Wireshark 工具栏左起第一个图标，在接口上开始侦听，片刻后停止侦听。这时捕获的数据量有多少？

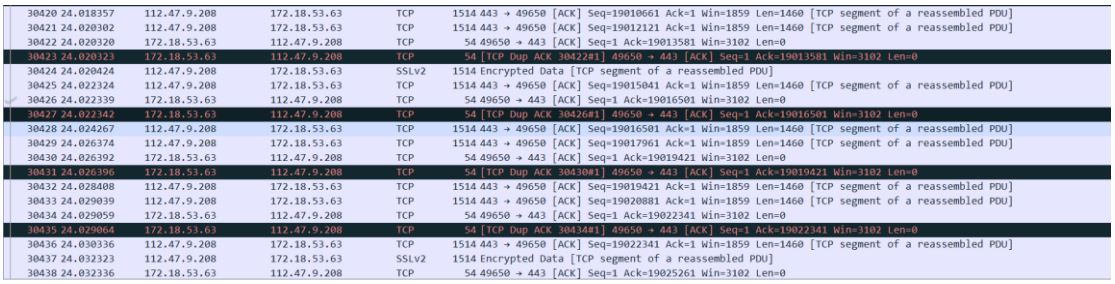
答：捕获的数据包的数量可以通过 Wireshark 捕获的数据包的行数看出，本次实验中捕获了数据包有 30444 个。



（2）观察捕获数据的源 IP 地址和目的 IP 地址，这些数据是发出的还是发过来的？选择几个 IP 地址，通过网站 www.ip138.com 查询这些 IP 地址的地理位置。

答：判断数据是发出还是发来方法：通过查找本机的 IP 地址，若在一个数据包中本机的 IP 地址为源 IP 地址，则该数据包为发出的数据包，若本机的 IP 地址为目的 IP 地址，则为发入本机的数据包。

如在以下截图中第 30423 个数据包中源 IP 地址为本机 IP 地址，故该数据包为发出数据包。第 30424 个数据包目的 IP 地址为本机 IP 地址，故该数据包为发入数据包。



查询 IP 地址的物理位置：登陆 www.ip138.com ，输入我们捕获数据包的一些 IP 地址，如第 30419 个数据包的目的 IP 地址 112.47.9.208，查询结果如下：

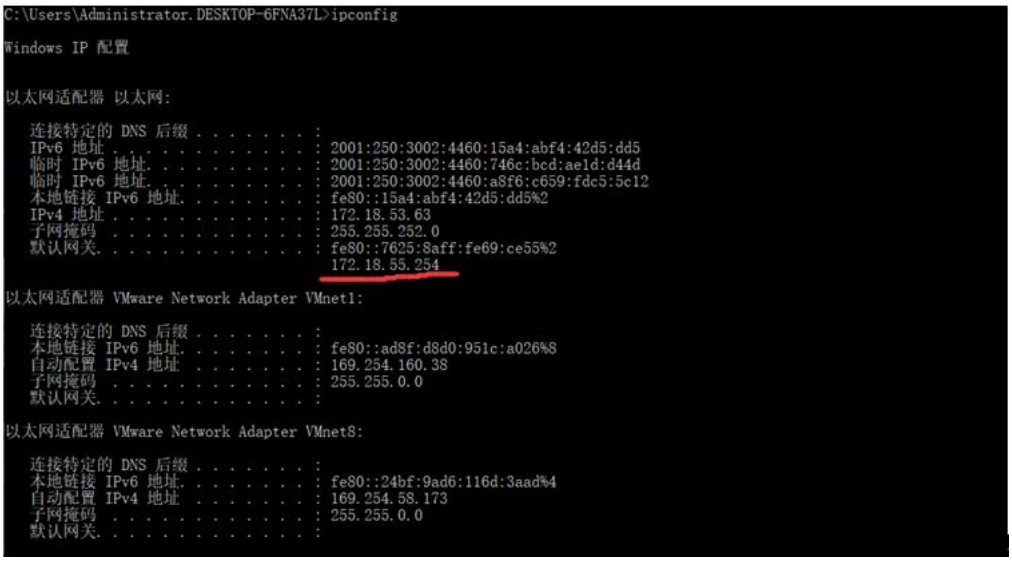
iP段起始	iP段结束	归属地	网络	Windows子网掩码	Linux子网掩码
112.47.0.0	112.47.35.255	中国 福建省 泉州市	移动	255.255.192.0	112.47.0.0/18

再如第 30374 个数据包的源 IP 地址 183.232.171.189 的查询结果如下：

iP段起始	iP段结束	归属地	网络	Windows子网掩码	Linux子网掩码
183.232.162.0	183.232.199.255	中国 广东省 佛山市	移动	255.255.128.0	183.232.128.0/17

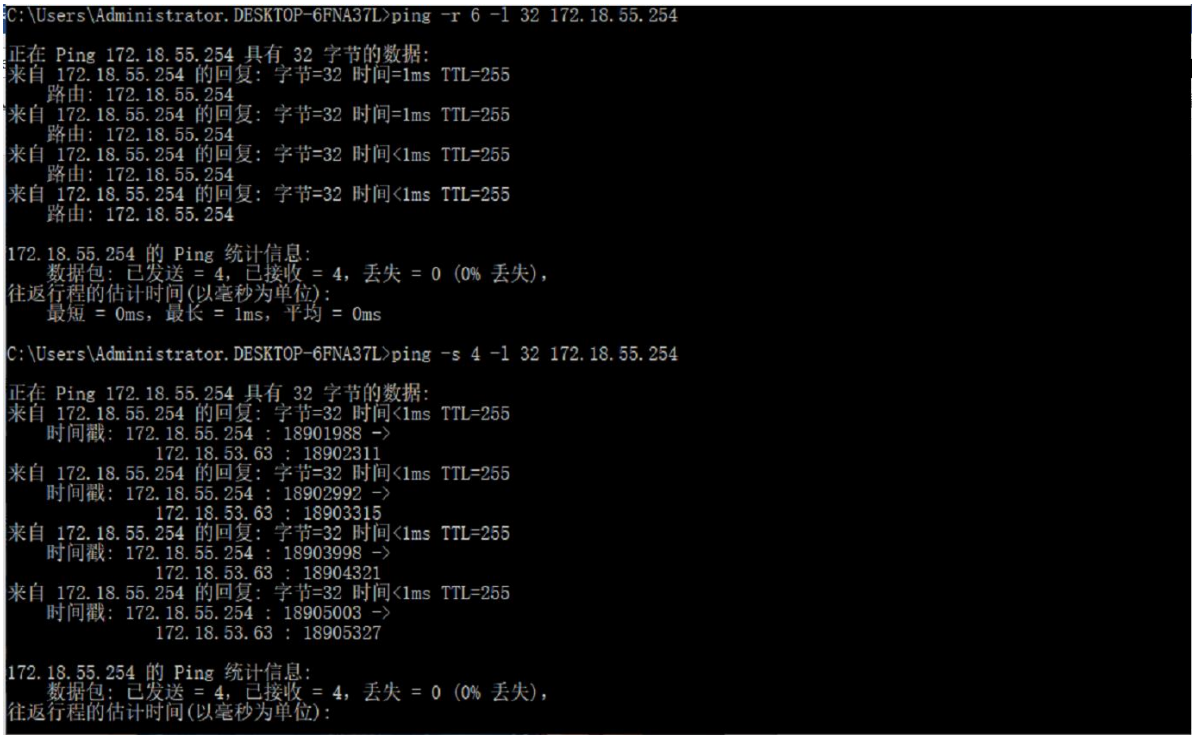
（3）查看所在网络的网关 IP 地址，假设查到的 IP 地址是 a.b.c.d，在命令窗口运行 ping -r 6 -l a.b.c.d 和 ping -s 4 -l a.b.c.d 命令并捕获数据包。

答：在终端中输入 ipconfig 显示如下：



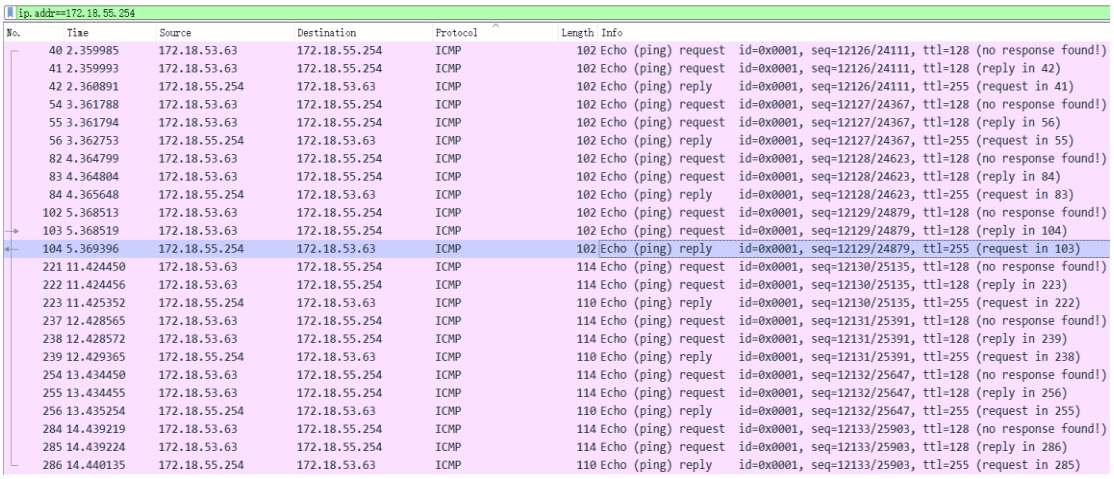
所以网关的 IP 地址为 172.18.55.254。（见默认网关那一行）

在终端中运行 ping -r 6 -l 32 172.18.55.254 和 ping -s 4 -l 32 172.18.55.254。



（4）执行 filter:ip.addr==a.b.c.d 命令查看，截屏运行结果。

答：如下图所示：



ip.addr==172.18.55.254

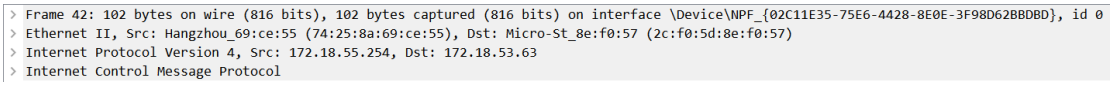
No.	Time	Source	Destination	Protocol	Length	Info
40	2.359985	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12126/24111, ttl=128 (no response found!)
41	2.359993	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12126/24111, ttl=128 (reply in 42)
42	2.360891	172.18.55.254	172.18.53.63	ICMP	102	Echo (ping) reply id=0x0001, seq=12126/24111, ttl=255 (request in 41)
54	3.361788	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12127/24367, ttl=128 (no response found!)
55	3.361794	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12127/24367, ttl=128 (reply in 56)
56	3.362753	172.18.55.254	172.18.53.63	ICMP	102	Echo (ping) reply id=0x0001, seq=12127/24367, ttl=255 (request in 55)
82	4.364799	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12128/24623, ttl=128 (no response found!)
83	4.364804	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12128/24623, ttl=128 (reply in 84)
84	4.365648	172.18.55.254	172.18.53.63	ICMP	102	Echo (ping) reply id=0x0001, seq=12128/24623, ttl=255 (request in 83)
102	5.368513	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12129/24879, ttl=128 (no response found!)
103	5.368519	172.18.53.63	172.18.55.254	ICMP	102	Echo (ping) request id=0x0001, seq=12129/24879, ttl=128 (reply in 104)
104	5.369396	172.18.55.254	172.18.53.63	ICMP	102	Echo (ping) reply id=0x0001, seq=12129/24879, ttl=255 (request in 103)
221	11.424450	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12130/25135, ttl=128 (no response found!)
222	11.424456	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12130/25135, ttl=128 (reply in 223)
223	11.425352	172.18.55.254	172.18.53.63	ICMP	110	Echo (ping) reply id=0x0001, seq=12130/25135, ttl=255 (request in 222)
237	12.428565	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12131/25391, ttl=128 (no response found!)
238	12.428572	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12131/25391, ttl=128 (reply in 239)
239	12.429365	172.18.55.254	172.18.53.63	ICMP	110	Echo (ping) reply id=0x0001, seq=12131/25391, ttl=255 (request in 238)
254	13.434450	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12132/25647, ttl=128 (no response found!)
255	13.434455	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12132/25647, ttl=128 (reply in 256)
256	13.435254	172.18.55.254	172.18.53.63	ICMP	110	Echo (ping) reply id=0x0001, seq=12132/25647, ttl=255 (request in 255)
284	14.439219	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12133/25903, ttl=128 (no response found!)
285	14.439224	172.18.53.63	172.18.55.254	ICMP	114	Echo (ping) request id=0x0001, seq=12133/25903, ttl=128 (reply in 286)
286	14.440135	172.18.55.254	172.18.53.63	ICMP	110	Echo (ping) reply id=0x0001, seq=12133/25903, ttl=255 (request in 285)

（5）捕获的数据中都有哪些协议？分别找出 Echo 和 Stamp 的请求和响应分组，分析这些数据主要字段的含义。

答：捕获的数据中有如下协议：0x9001，ARP，DHCPV6，EAP，ICMP，ICMPv6，IPv6，LLDP，LLMNR，MDNS，SSDP，TCP，UDP。

Echo 和 Stamp 请求和响应分组如（4）中图片所示（其中所选行上方为 Echo 请求和响应分组，所选行及其下方为 Stamp 请求和响应分组）：

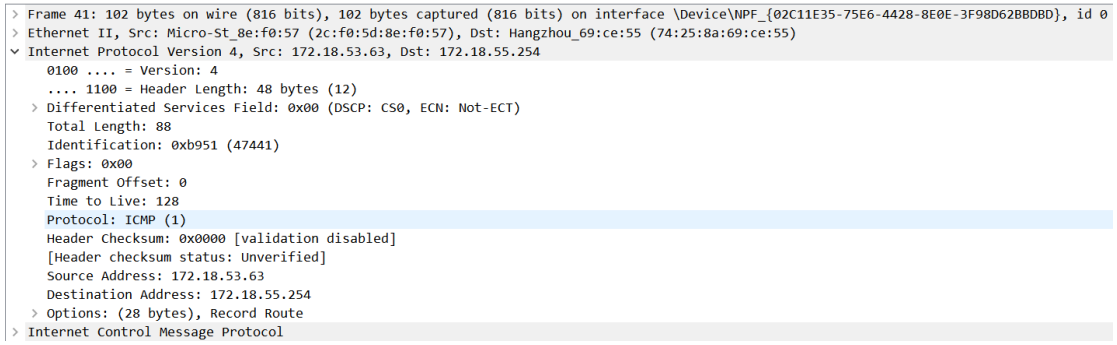
以下分析他们的主要字段：



```
> Frame 42: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface \Device\NPF_{02C11E35-75E6-4428-8E0E-3F98D62BBDBD}, id 0
> Ethernet II, Src: Hangzhou_69:ce:55 (74:25:8a:69:ce:55), Dst: Micro-St_8e:f0:57 (2c:f0:5d:8e:f0:57)
> Internet Protocol Version 4, Src: 172.18.55.254, Dst: 172.18.53.63
> Internet Control Message Protocol
```

上图第一行主要包括物理层的数据帧概况，第二行主要包括数据链路层以太网帧头部信息，第三行是互联网层 IP 包头部信息，第四行是网络层 ICMP 包头部信息。其中第一、二、四行内容在 Echo 和 Stamp 分组中差别不大，以下主要分析第三行内容：

Echo 分组：



```
> Frame 41: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface \Device\NPF_{02C11E35-75E6-4428-8E0E-3F98D62BBDBD}, id 0
> Ethernet II, Src: Micro-St_8e:f0:57 (2c:f0:5d:8e:f0:57), Dst: Hangzhou_69:ce:55 (74:25:8a:69:ce:55)
v Internet Protocol Version 4, Src: 172.18.53.63, Dst: 172.18.55.254
    0100 .... = Version: 4
    .... 1100 = Header Length: 48 bytes (12)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 88
    Identification: 0xb951 (47441)
  > Flags: 0x00
    Fragment Offset: 0
    Time to Live: 128
    Protocol: ICMP (1)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 172.18.53.63
    Destination Address: 172.18.55.254
  > Options: (28 bytes), Record Route
> Internet Control Message Protocol
```

从上图中可以看到数据帧的总长度 Total Length,首部校验 Identification,TTL,源地址 Source Address,目的主机地址 Destination Address

查看 Options 中内容可以看到路径中的路由信息：

```
Options: (28 bytes), Record Route
  IP Option - Record Route (27 bytes)
    Type: 7
    Length: 27
    Pointer: 4
    Empty Route: 0.0.0.0 <- (next)
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
  IP Option - End of Options List (EOL)
    Type: 0
```

```
Options: (28 bytes), Record Route
  IP Option - Record Route (27 bytes)
    Type: 7
    Length: 27
    Pointer: 8
    Recorded Route: 172.18.55.254
    Empty Route: 0.0.0.0 <- (next)
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
  IP Option - End of Options List (EOL)
    Type: 0
```

其中左图为请求分组的路由信息，右图为响应分组的路由信息

Stamp 分组：

```
Frame 223: 110 bytes on wire (880 bits), 110 bytes captured (880 bits) on interface \Device\NPF_{02C11E35-75E6-4428-8E0E-3F98D62BBDBD}, id 0
Ethernet II, Src: Hangzhou_69:ce:55 (74:25:8a:69:ce:55), Dst: Micro-St_8e:f0:57 (2c:f0:5d:8e:f0:57)
Internet Protocol Version 4, Src: 172.18.55.254, Dst: 172.18.53.63
  0100 .... = Version: 4
  .... 1110 = Header Length: 56 bytes (14)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 96
  Identification: 0xda5a (55898)
  Flags: 0x00
  Fragment Offset: 0
  Time to Live: 255
  Protocol: ICMP (1)
  Header Checksum: 0x70a5 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 172.18.55.254
  Destination Address: 172.18.53.63
  Options: (36 bytes), Time Stamp
    IP Option - Time Stamp (36 bytes)
Internet Control Message Protocol
```

Stamp 分组其余内容与 Echo 分组中相同，Options 中存储的是时间戳信息：

```
Options: (40 bytes), Time Stamp
  IP Option - Time Stamp (36 bytes)
    Type: 68
    Length: 36
    Pointer: 5
    0000 .... = Overflow: 0
    .... 0001 = Flag: Time stamp and address (0x1)
    Address: -
    Time stamp: 0
    Address: -
    Time stamp: 0
    Address: -
    Time stamp: 0
    Address: -
    Time stamp: 0
  IP Option - End of Options List (EOL)
    Type: 0
```

```
Options: (36 bytes), Time Stamp
  IP Option - Time Stamp (36 bytes)
    Type: 68
    Length: 36
    Pointer: 13
    0000 .... = Overflow: 0
    .... 0001 = Flag: Time stamp and address (0x1)
    Address: 172.18.55.254
    Time stamp: 18901988
    Address: -
    Time stamp: 0
    Address: -
    Time stamp: 0
    Address: -
    Time stamp: 0
```

左图为请求分组时间戳信息，右图为响应分组时间戳信息，且时间戳信息与命令行输出内容相对应。

【实验思考】

（1）捕获网络上的数据可谓轻而易举，网络嗅探可以说无处不在，如何发现网络中的嗅探行为？

答：①注意网速；

②搜索主机相关进程；

③使用相关软件进行监测；

（2）如何防范被嗅探？

答：①网络分段：一个网络段包括一组共享低层设备和线路的机器，如交换机，动态集线器和网桥等设

备，可以对数据流进行限制，从而达到防止嗅探的目的。

②加密：一方面可以对数据流中的部分重要信息进行加密，另一方面也可只对应用层加密，然而后者将使大部分与网络和操作系统有关的敏感信息失去保护。选择何种加密方式这就取决于信息的安全级别及网络的安全程度。

③一次性口令设置：口令并不在网络上传输而是在两端进行字符串匹配，客户端利用从服务器上得到的 Challenge 和自身的口令计算出一个新字符串并将之返回给服务器。在服务器上利用比较算法进行匹配，如果匹配，连接就允许建立，所有的 Challenge 和字符串都只使用一次。

④禁用杂错节点：安装不支持杂错的网卡，通常可以防止 IBM 兼容机进行嗅探。

学号	学生	自评分
18329015	郝裕玮	100
18325071	张闯	100
19335153	马淙升	100