浙江大学

本科实验报告

课程名称：	计算机网络基础
姓　　名：	蒋仕彪
学　　院：	计算机学院
系：	求是科学班（计算机）
专　　业：	计算机科学与技术
学　　号：	3170102587
指导教师：	董玮

2020 年　3　月　7 日

浙江大学实验报告

课程名称：计算机网络基础 实验类型：操作实验

实验项目名称：WireShark 软件初探和常见网络命令的使用

学生姓名：蒋仕彪 专业：计算机科学与技术 学号：3170102587

同组学生姓名：蒋仕彪 指导老师：董玮

实验地点：计算机网络实验室 实验日期：2020 年 3 月 7 日

一、 实验目的和要求：

- 初步了解 WireShark 软件的界面和功能
- 熟悉各类常用网络命令的使用

二、 实验内容和原理

- Wireshark 是 PC 上使用最广泛的免费抓包工具，可以分析大多数常见的协议数据包。有 Windows 版本、Linux 版本和 Mac 版本，可以免费从网上下载
- 初步掌握网络协议分析软件 Wireshark 的使用，学会配置过滤器
- 根据要求配置 Wireshark，捕获某一类协议的数据包
- 在 PC 机上熟悉常用网络命令的功能和用法：Ping.exe，Netstat.exe, Telnet.exe, Tracert.exe, Arp.exe, Ipconfig.exe, Net.exe, Route.exe, Nslookup.exe
- 利用 WireShark 软件捕捉上述部分命令产生的数据包

三、 主要仪器设备

- 联网的 PC 机
- WireShark 协议分析软件

四、 操作方法与实验步骤

- 安装网络包捕获软件 Wireshark
- 配置网络包捕获软件，捕获所有机器的数据包
- 配置网络包捕获软件，只捕获特定类型的包
- 在 Windows 命令行方式下，执行适当的命令，完成以下功能(请以管理员身份打开命令行)：
    1. 测试到特定地址的联通性、数据包延迟时间
    2. 显示本机的网卡物理地址、IP 地址
    3. 显示本机的默认网关地址、DNS 服务器地址
    4. 显示本机记录的局域网内其它机器 IP 地址与其物理地址的对照表

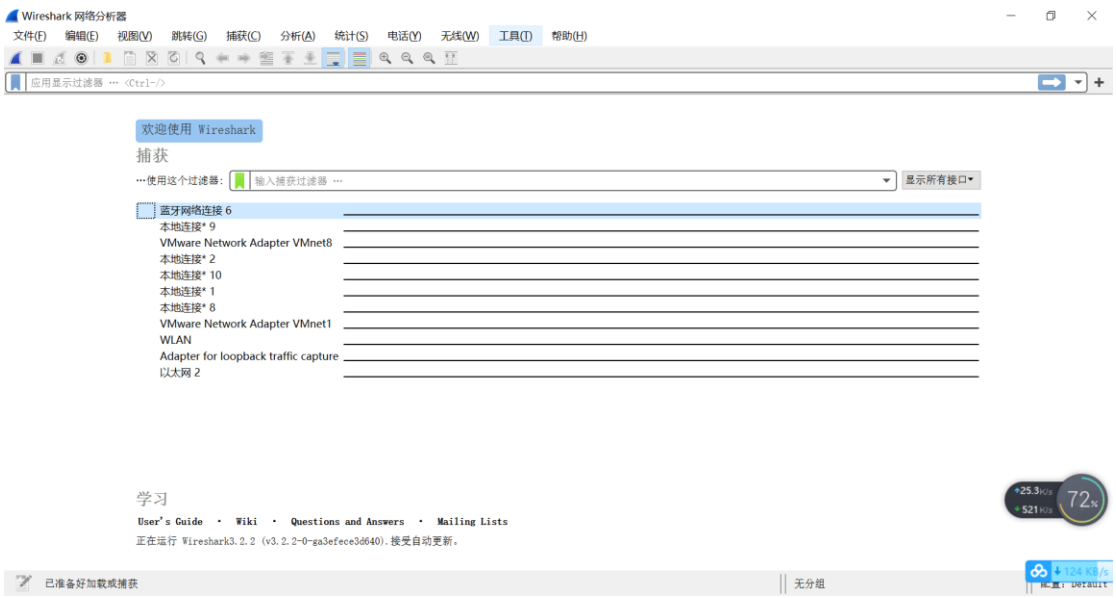
5. 显示从本机到达一个特定地址的路由
6. 显示某一个域名的 IP 地址
7. 显示已经与本机建立 TCP 连接的端口、IP 地址、连接状态等信息
8. 显示本机的路由表信息，并手工添加一个路由
9. 显示本机的网络映射连接
10. 显示局域网内某台机器的共享资源
11. 使用 telnet 连接 WEB 服务器的端口，输入（<cr>表示回车）获得该网站的主页内容：

```
GET / HTTP/1.1<cr>
Host: 任意字符串<cr>
<cr>
```

- 利用 WireShark 实时观察在执行上述命令时，哪些命令会额外产生数据包，并记录这些数据包的种类。

五、 实验数据记录和处理

- **运行 Wireshark 软件，界面是由哪几个部分构成？各有什么作用？**



界面大概由以下六部分构成：

菜单栏：包括一些文件、编辑、视图、跳转等的详细命令。

工具栏：包括一些启动按钮、停止按钮、重新启动按钮、网卡接口设置按钮等快捷按钮。

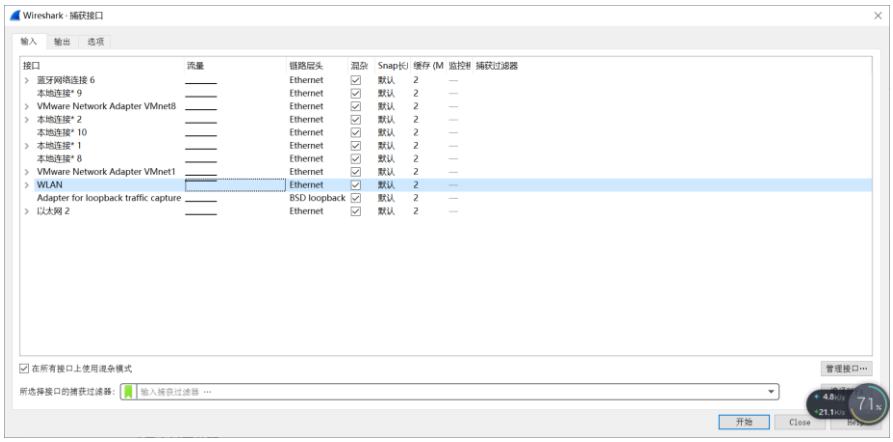
过滤栏：用来设置显示过滤器

主体部分：直接显示了本地电脑上的所有网卡列表，以及通过时序图显示了哪些网卡上面有流量传输，以及流量传输的分布情况。

学习指导信息。

状态栏：显示 Wireshark 状态信息。

- **开始捕获网络数据包，你看到了什么？有哪些协议？**

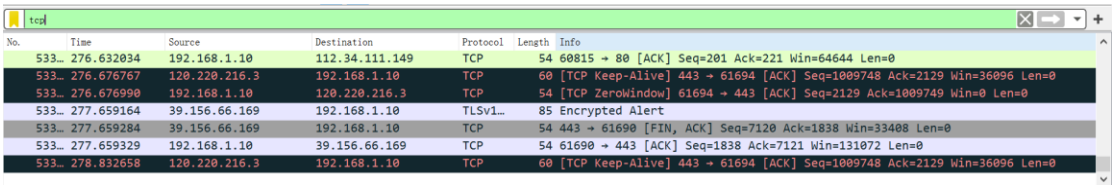


No.	Time	Source	Destination	Protocol	Length	Info
7290	49.783788	192.168.1.10	120.220.216.3	TCP	66	61594 → 443 [ACK] Seq=1 Ack=1383749 Win=16 Len=0 SLE=1385161 SRE=1386573
7291	49.783991	192.168.1.10	120.220.216.3	TCP	54	61594 → 443 [ACK] Seq=1 Ack=1386573 Win=5 Len=0
7292	49.784329	192.168.1.10	112.34.112.252	TCP	54	61610 → 443 [ACK] Seq=322 Ack=4557 Win=130816 Len=0
7293	49.785844	192.168.1.10	112.34.112.252	TLSv1…	1251	Application Data
7294	49.813536	112.34.112.252	192.168.1.10	TCP	60	443 → 61610 [ACK] Seq=4602 Ack=1519 Win=18176 Len=0
7295	49.991534	117.184.250.46	192.168.1.10	OICQ	129	OICQ Protocol
7296	50.034811	192.168.1.10	120.220.216.3	TCP	85	[TCP Retransmission] 61573 → 443 [FIN, PSH, ACK] Seq=1 Ack=700770 Win=1947 …

No.	Time	Source	Destination	Protocol	Length	Info
170…	98.200876	192.168.1.10	183.232.93.211	UDP	276	53967 → 8000 Len=234
170…	98.251360	183.232.93.211	192.168.1.10	UDP	308	8000 → 53967 Len=266
170…	98.389317	120.220.216.3	192.168.1.10	TCP	60	[TCP Keep-Alive] 443 → 61627 [ACK] Seq=613487 Ack=2191 Win=36096 Len=0
170…	98.389670	192.168.1.10	120.220.216.3	TCP	54	[TCP ZeroWindow] 61627 → 443 [ACK] Seq=2191 Ack=613488 Win=0 Len=0
170…	98.687397	117.184.250.46	192.168.1.10	OICQ	129	OICQ Protocol
170…	98.914176	192.168.1.10	120.220.216.3	TCP	54	61602 → 443 [RST, ACK] Seq=1861 Ack=2005698 Win=0 Len=0
170…	99.182997	fe80::85dc:6b8c:cd3…	fe80::1	ICMPv6	86	Neighbor Solicitation for fe80::1 from 9c:b6:d0:e2:c5:cb

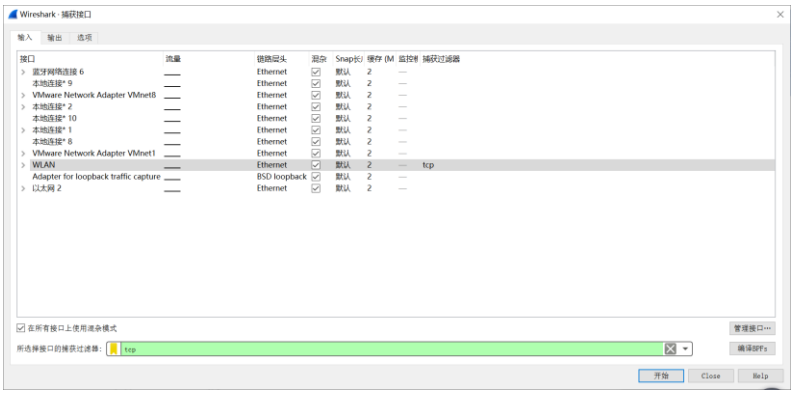
我看到了很多 records，每条包含一个协议。
我看到的协议分为 TCP/TLSv1/UDP/OICQ/ICMPv6……

- **配置应用显示过滤器，让界面只显示某一协议类型的数据包。**



在显示过滤器中过滤 tcp，则界面中只会显示 TCP 协议的数据包。

- **配置捕获过滤器，只捕获某类协议的数据包。**



在捕获过滤器中过滤 tcp，则界面中只会显示 TCP 协议的数据包。

- **利用 Ping.exe，Netstat.exe， Telnet.exe, Tracert.exe, Arp.exe, Ipconfig.exe, Net.exe, Route.exe 命令完成在实验步骤中列举的 11 个功能。**

1. 测试到特定地址的联通性、数据包延迟时间

```
C:\Users\jiang\Desktop>ping 14.215.177.39

正在 Ping 14.215.177.39 具有 32 字节的数据:
来自 14.215.177.39 的回复: 字节=32 时间=46ms TTL=49
来自 14.215.177.39 的回复: 字节=32 时间=50ms TTL=49
来自 14.215.177.39 的回复: 字节=32 时间=46ms TTL=49
来自 14.215.177.39 的回复: 字节=32 时间=46ms TTL=49

14.215.177.39 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 46ms，最长 = 50ms，平均 = 47ms
```

2. 显示本机的网卡物理地址、IP 地址（蓝色）
3. 显示本机的默认网关地址、DNS 服务器地址（红色）

```
C:\Windows\system32\cmd.exe
C:\Users\jiang\Desktop>ipconfig/all

Windows IP 配置

   主机名  . . . . . . . . . . . . . : DESKTOP-7HNC82T
   主 DNS 后缀 . . . . . . . . . . . :
   节点类型  . . . . . . . . . . . . : 混合
   IP 路由已启用 . . . . . . . . . . : 否
   WINS 代理已启用 . . . . . . . . . : 否

无线局域网适配器 本地连接* 2:

   媒体状态  . . . . . . . . . . . . : 媒体已断开连接
   连接特定的 DNS 后缀 . . . . . . . :
   描述. . . . . . . . . . . . . . . : Microsoft Wi-Fi Direct Virtual Adapter
   物理地址. . . . . . . . . . . . . : 9E-B6-D0-E2-C5-CB
   DHCP 已启用 . . . . . . . . . . . : 是
   自动配置已启用. . . . . . . . . . : 是

无线局域网适配器 本地连接* 1:

   媒体状态  . . . . . . . . . . . . : 媒体已断开连接
   连接特定的 DNS 后缀 . . . . . . . :
   描述. . . . . . . . . . . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #3
   物理地址. . . . . . . . . . . . . : AE-B6-D0-E2-C5-CB
   DHCP 已启用 . . . . . . . . . . . : 是
   自动配置已启用. . . . . . . . . . : 是

以太网适配器 VMware Network Adapter VMnet1:

   连接特定的 DNS 后缀 . . . . . . . :
   描述. . . . . . . . . . . . . . . : VMware Virtual Ethernet Adapter for VMnet1
   物理地址. . . . . . . . . . . . . : 00-50-56-C0-00-01
   DHCP 已启用 . . . . . . . . . . . : 是
   自动配置已启用. . . . . . . . . . : 是
   本地链接 IPv6 地址. . . . . . . . : fe80::d74:c5c7:7ad:6cb1%8(首选)
   IPv4 地址 . . . . . . . . . . . . : 192.168.238.1(首选)
   子网掩码  . . . . . . . . . . . . : 255.255.255.0
   获得租约的时间  . . . . . . . . . : 2020年3月7日 9:41:38
   租约过期的时间  . . . . . . . . . : 2020年3月7日 14:52:01
   默认网关. . . . . . . . . . . . . :
   DHCP 服务器 . . . . . . . . . . . : 192.168.238.254
   DHCPv6 IAID . . . . . . . . . . . : 939544662
   DHCPv6 客户端 DUID  . . . . . . . : 00-01-00-01-21-F8-B1-62-9C-B6-D0-E2-C5-CB
   DNS 服务器  . . . . . . . . . . . : fec0:0:0:ffff::1%1
                                       fec0:0:0:ffff::2%1
```

4. 显示本机记录的局域网内其它机器 IP 地址与其物理地址的对照表

用 arp 命令可以显示。

```
C:\Users\jiang\Desktop>arp -a

接口: 192.168.238.1 --- 0x8
  Internet 地址         物理地址              类型
  192.168.238.254       00-50-56-e7-d3-d1     动态
  192.168.238.255       ff-ff-ff-ff-ff-ff     静态
  224.0.0.22            01-00-5e-00-00-16     静态
  224.0.0.251           01-00-5e-00-00-fb     静态
  224.0.0.252           01-00-5e-00-00-fc     静态
  224.97.73.103         01-00-5e-61-49-67     静态
  227.64.122.157        01-00-5e-40-7a-9d     静态
  227.161.232.116       01-00-5e-21-e8-74     静态
  232.36.40.120         01-00-5e-24-28-78     静态
  236.10.30.172         01-00-5e-0a-1e-ac     静态
  239.255.255.250       01-00-5e-7f-ff-fa     静态
  255.255.255.255       ff-ff-ff-ff-ff-ff     静态
```

5. 显示从本机到达一个特定地址的路由

用 tracert 命令可以追踪。我选择跟踪百度的 IP。

```
C:\Users\jiang\Desktop>tracert www.baidu.com

通过最多 30 个跃点跟踪
到 www.a.shifen.com [36.152.44.96] 的路由:

  1     2 ms     2 ms     2 ms  192.168.1.1
  2    10 ms     8 ms     6 ms  10.22.48.1
  3     6 ms     7 ms     7 ms  117.149.140.177
  4    13 ms    15 ms    12 ms  183.247.227.57
  5    16 ms     *        *     221.183.47.169
  6    20 ms    22 ms    22 ms  221.183.42.129
  7    21 ms    22 ms    21 ms  221.183.59.54
  8    22 ms    20 ms    20 ms  170.23.207.183.static.js.chinamobile.com [183.207.23.170]
  9    16 ms    19 ms    16 ms  182.61.253.214
 10     *        *        *     请求超时。
 11    22 ms    22 ms    23 ms  36.152.44.96

跟踪完成。
```

6. 显示某一个域名的 IP 地址

可以通过 ping 操作的返回信息获得。

```
C:\Users\jiang\Desktop>ping www.baidu.com

正在 Ping www.a.shifen.com [36.152.44.96] 具有 32 字节的数据:
来自 36.152.44.96 的回复: 字节=32 时间=20ms TTL=55
来自 36.152.44.96 的回复: 字节=32 时间=22ms TTL=55
来自 36.152.44.96 的回复: 字节=32 时间=20ms TTL=55
来自 36.152.44.96 的回复: 字节=32 时间=22ms TTL=55

36.152.44.96 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 20ms, 最长 = 22ms, 平均 = 21ms
```

7. 显示已经与本机建立 TCP 连接的端口、IP 地址、连接状态等信息

```
C:\Users\jiang\Desktop>netstat

活动连接

  协议   本地地址          外部地址        状态
  TCP    127.0.0.1:1080         hub5btmain:65034       ESTABLISHED
  TCP    127.0.0.1:1080         hub5btmain:65054       ESTABLISHED
  TCP    127.0.0.1:9012         hub5btmain:55694       ESTABLISHED
  TCP    127.0.0.1:54530        hub5btmain:63644       ESTABLISHED
  TCP    127.0.0.1:55694        hub5btmain:9012        ESTABLISHED
  TCP    127.0.0.1:63644        hub5btmain:54530       ESTABLISHED
  TCP    127.0.0.1:63645        hub5btmain:63646       ESTABLISHED
  TCP    127.0.0.1:63646        hub5btmain:63645       ESTABLISHED
  TCP    127.0.0.1:65034        hub5btmain:1080        ESTABLISHED
  TCP    127.0.0.1:65054        hub5btmain:1080        ESTABLISHED
  TCP    192.168.1.10:49478     52.139.250.253:https   ESTABLISHED
  TCP    192.168.1.10:54671     221.228.75.64:9202     ESTABLISHED
  TCP    192.168.1.10:60815     112.34.111.149:http    ESTABLISHED
  TCP    192.168.1.10:60848     36.152.44.139:5287     ESTABLISHED
  TCP    192.168.1.10:60892     36.152.44.139:5287     ESTABLISHED
  TCP    192.168.1.10:60956     112.34.111.108:https   CLOSE_WAIT
```

8. 显示本机的路由表信息，并手工添加一个路由

Netstar -r 或者 route print 命令可以显示路由表。

```
IPv4 路由表
===========================================================================
活动路由:
网络目标        网络掩码          网关       接口   跃点数
          0.0.0.0          0.0.0.0    192.168.1.1    192.168.1.10     50
        127.0.0.0        255.0.0.0         在链路上         127.0.0.1    331
        127.0.0.1  255.255.255.255         在链路上         127.0.0.1    331
  127.255.255.255  255.255.255.255         在链路上         127.0.0.1    331
      192.168.1.0    255.255.255.0         在链路上       192.168.1.10    306
     192.168.1.10  255.255.255.255         在链路上       192.168.1.10    306
    192.168.1.255  255.255.255.255         在链路上       192.168.1.10    306
     192.168.64.0    255.255.255.0         在链路上       192.168.64.1    291
     192.168.64.1  255.255.255.255         在链路上       192.168.64.1    291
   192.168.64.255  255.255.255.255         在链路上       192.168.64.1    291
    192.168.238.0    255.255.255.0         在链路上      192.168.238.1    291
    192.168.238.1  255.255.255.255         在链路上      192.168.238.1    291
  192.168.238.255  255.255.255.255         在链路上      192.168.238.1    291
        224.0.0.0        240.0.0.0         在链路上         127.0.0.1    331
        224.0.0.0        240.0.0.0         在链路上       192.168.1.10    306
        224.0.0.0        240.0.0.0         在链路上       192.168.64.1    291
        224.0.0.0        240.0.0.0         在链路上      192.168.238.1    291
  255.255.255.255  255.255.255.255         在链路上         127.0.0.1    331
  255.255.255.255  255.255.255.255         在链路上       192.168.1.10    306
  255.255.255.255  255.255.255.255         在链路上       192.168.64.1    291
  255.255.255.255  255.255.255.255         在链路上      192.168.238.1    291
===========================================================================
永久路由:
  无
```

在命令行里输入 在命令提示符中，输入

route add 10.188.0.0 mask 255.255.0.0 192.168.1.253 -p

再次查看路由列表时候，可以看到新增了一个永久路由：

```
永久路由:
  网络地址          网络掩码  网关地址  跃点数
      10.188.0.0      255.255.0.0    192.168.1.253        1
===========================================================================
```

9. 显示本机的网络映射连接

用 net use 可以查看列表。

```
C:\WINDOWS\system32>net use
会记录新的网络连接。

列表是空的。
```

10. 显示局域网内某台机器的共享资源

用 net share 可以查看。

```
C:\WINDOWS\system32>net share

共享名       资源                            注解

-------------------------------------------------------------------------------
C$           C:\                             默认共享
D$           D:\                             默认共享
E$           E:\                             默认共享
F$           F:\                             默认共享
G$           G:\                             默认共享
IPC$                                         远程 IPC
print$       C:\WINDOWS\system32\spool\drivers
                                             打印机驱动程序
ADMIN$       C:\WINDOWS                      远程管理
命令成功完成。
```

11. 使用 telnet 连接 WEB 服务器的端口，输入（<cr>表示回车）获得该网站的主页内容：

GET / HTTP/1.1<cr>

Host: 任意字符串<cr>

<cr>

先用 telnet 连上百度。

```
C:\Users\jiang\Desktop>telnet www.baidu.com 80
```

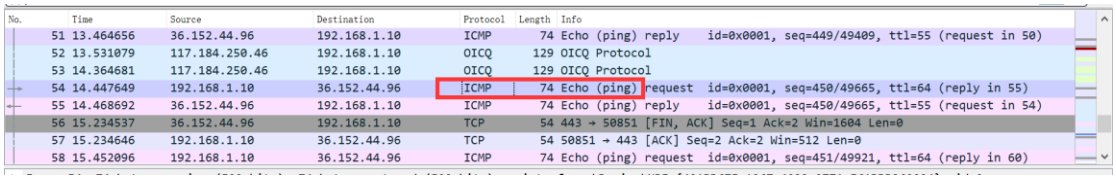
按住 CTRL+] 可以调节成显式模式。

```
Telnet www.baidu.com
欢迎使用 Microsoft Telnet Client

Escape 字符为 'CTRL+]'
```

输入命令后得到结果。

```
Telnet www.baidu.com
GET / HTTP/1.1
Host: www.baidu.com

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: no-cache
Connection: Keep-Alive
Content-Length: 14615
Content-Type: text/html
Date: Mon, 25 Mar 2019 09:05:14 GMT
Etag: "5c908b24-3917"
Last-Modified: Tue, 19 Mar 2019 06:24:36 GMT
P3p: CP=" OTI DSP COR IVA OUR IND COM "
Pragma: no-cache
Server: BWS/1.1
Set-Cookie: BAIDUID=E9BBA5952A3B82364F3B76238F880E50:FG=1; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path
=/; domain=.baidu.com
Set-Cookie: BIDUPSID=E9BBA5952A3B82364F3B76238F880E50; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/;
domain=.baidu.com
Set-Cookie: PSTM=1553504714; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/; domain=.baidu.com
```

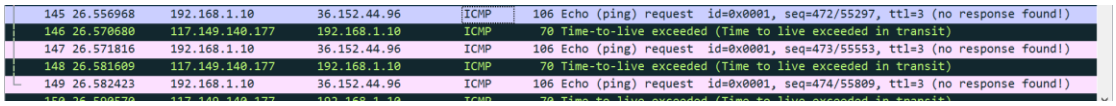
- **观察使用 Ping 命令时在 WireShark 中出现的数据包并捕获。这是什么协议？**



No.	Time	Source	Destination	Protocol	Length	Info
51	13.464656	36.152.44.96	192.168.1.10	ICMP	74	Echo (ping) reply id=0x0001, seq=449/49409, ttl=55 (request in 50)
52	13.531079	117.184.250.46	192.168.1.10	OICQ	129	OICQ Protocol
53	14.364681	117.184.250.46	192.168.1.10	OICQ	129	OICQ Protocol
54	14.447649	192.168.1.10	36.152.44.96	ICMP	74	Echo (ping) request id=0x0001, seq=450/49665, ttl=64 (reply in 55)
55	14.468692	36.152.44.96	192.168.1.10	ICMP	74	Echo (ping) reply id=0x0001, seq=450/49665, ttl=55 (request in 54)
56	15.234537	36.152.44.96	192.168.1.10	TCP	54	443 → 50851 [FIN, ACK] Seq=1 Ack=2 Win=1604 Len=0
57	15.234646	192.168.1.10	36.152.44.96	TCP	54	50851 → 443 [ACK] Seq=2 Ack=2 Win=512 Len=0
58	15.452096	192.168.1.10	36.152.44.96	ICMP	74	Echo (ping) request id=0x0001, seq=451/49921, ttl=64 (reply in 60)

显然这是 ICMP 协议。

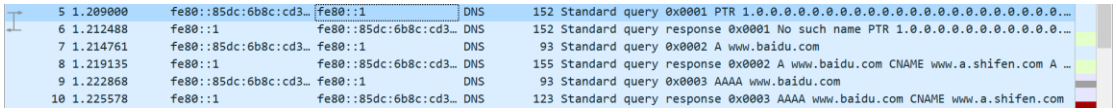
- **观察使用 Tracert 命令时在 WireShark 中出现的数据包并捕获。这是什么协议？**



145	26.556968	192.168.1.10	36.152.44.96	ICMP	106	Echo (ping) request id=0x0001, seq=472/55297, ttl=3 (no response found!)
146	26.570680	117.149.140.177	192.168.1.10	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
147	26.571816	192.168.1.10	36.152.44.96	ICMP	106	Echo (ping) request id=0x0001, seq=473/55553, ttl=3 (no response found!)
148	26.581609	117.149.140.177	192.168.1.10	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
149	26.582423	192.168.1.10	36.152.44.96	ICMP	106	Echo (ping) request id=0x0001, seq=474/55809, ttl=3 (no response found!)

也是 ICMP 协议。

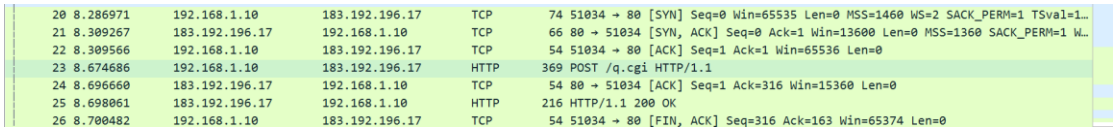
- **观察使用 Nslookup 命令时在 WireShark 中出现的数据包并捕获。这是什么协议？**



5	1.209000	fe80::85dc:6b8c:cd3…	fe80::1	DNS	152	Standard query 0x0001 PTR 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.…
6	1.212488	fe80::1	fe80::85dc:6b8c:cd3…	DNS	152	Standard query response 0x0001 No such name PTR 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.…
7	1.214761	fe80::85dc:6b8c:cd3…	fe80::1	DNS	93	Standard query 0x0002 A www.baidu.com
8	1.219135	fe80::1	fe80::85dc:6b8c:cd3…	DNS	155	Standard query response 0x0002 A www.baidu.com CNAME www.a.shifen.com A …
9	1.222868	fe80::85dc:6b8c:cd3…	fe80::1	DNS	93	Standard query 0x0003 AAAA www.baidu.com
10	1.225578	fe80::1	fe80::85dc:6b8c:cd3…	DNS	123	Standard query response 0x0003 AAAA www.baidu.com CNAME www.a.shifen.com

这是 DNS 协议。

- **观察使用 Telnet 命令时在 WireShark 中出现的数据包并捕获。这是什么协议？**



20	8.286971	192.168.1.10	183.192.196.17	TCP	74	51034 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=2 SACK_PERM=1 TSval=1…
21	8.309267	183.192.196.17	192.168.1.10	TCP	66	80 → 51034 [SYN, ACK] Seq=0 Ack=1 Win=13600 Len=0 MSS=1360 SACK_PERM=1 W…
22	8.309566	192.168.1.10	183.192.196.17	TCP	54	51034 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
23	8.674686	192.168.1.10	183.192.196.17	HTTP	369	POST /q.cgi HTTP/1.1
24	8.696660	183.192.196.17	192.168.1.10	TCP	54	80 → 51034 [ACK] Seq=1 Ack=316 Win=15360 Len=0
25	8.698061	183.192.196.17	192.168.1.10	HTTP	216	HTTP/1.1 200 OK
26	8.700482	192.168.1.10	183.192.196.17	TCP	54	51034 → 80 [FIN, ACK] Seq=316 Ack=163 Win=65374 Len=0

这是 TCP/IP 协议。

六、 实验结果与分析

- WireShark 的两种过滤器有什么不同？

  对于捕获过滤器，在捕获时就已经根据规则进行过滤。
  对于显示过滤器，其实捕获的时候一并捕获进来，显示的时候再过滤。

- 哪些网络命令会产生在 WireShark 中产生数据包，为什么？

  Ping.exe Telnet.exe, Tracert.exe, Nslookup.exe, arp.exe（有时） 会产生数据包。
  因为它们要根据相应的协议发送请求，自然会产生数据包。

- Ping 发送的是什么类型的协议数据包？什么时候会出现 ARP 消息？Ping 一个域名和 Ping 一个 IP 地址出现的数据包有什么不同？

  Ping 发送的是 ICMP 类型的协议数据包。
  当我们需要向某一个 IP 地址发送数据的时候，我们需要到 ARP 缓存表中查询 IP 对应的 MAC 地址；但是当缓存表中不存在这样的记录的时候，就会发送一个 ARP 请求询问该 IP 对应的 MAC 地址。
  Ping 一个域名的时候，可能还需要发送 DNS 请求数据，查询域名对应的 ip 地址。

七、 讨论、心得

我在抓包的时候只是按照实验的要求抓一些包，但是我们其实并不是十分了解包的含义。希望在之后仔细学习网络各层协议以后可以回顾一次这次实验，加深对网络协议的理解。

我也产生了一些问题，希望以后能够弄懂和解决，比如：在访问 www.zju.edu.cn 的时候，为什么访问了那么多不同的域名，这其中的行为具体是由什么所决定的？数据包具体是怎么形成的？不同层之间是怎么分工完成数据包的？

本次实验做下来给我感觉内容很丰富，通过使用 WireShark 捕获数据包的形式让我们动 手实践，体验了接收发送数据包报文的过程。内容很充实，让我收获很多。**但是感觉本次实验还是需要一定的理论，建议可以放到一定的理论课之后再布置**，或者 也可以提供一些学习资料、网站链接给同学去提前了解。为了大致理解这些概念，**我在每一步实验前，都需要自行在网上查阅资料，感觉有点费时**。**下次我自己也可以考虑先去系统性地学一下。**