

北京邮电大学软件学院

2019-2020 学年第 1 学期实验报告

课程名称: 计算机网络

实验名称: 实验一: 数据链路层实验

实验完成人:

姓名: 刘子豪 学号: 2017211971 成绩:

指导教师: 王文东

日 期: 2019 年 10 月 23 日

一、 实验目的

通过本实验使学生理解协议数据单元（PDU）概念、掌握以太网帧结构字段定义和功能。

二、 实验任务

搭建实验环境，使用网络抓包软件（如 Wireshark 软件等）抓取访问互联网所产生的数据包，分析其中的以太网帧结构字段组成，掌握以太网帧结构字段的功能。

三、 实验内容

- 1) 在可以访问互联网的主机上下载并安装网络抓包软件 Wireshark。
- 2) 运行 Wireshark 软件，启动 Wireshark 软件的抓包功能抓取本主机访问互联网中某网站过程中发送和接收的数据包。
- 3) 对所抓取的数据包进行分析，分析所发送和接收的数据包的以太网帧结构中的源 MAC 地址、目的 MAC 地址和类型（type）字段的使用方法；理解各字段的含义和功能。
- 4) 选做部分：分析所抓取的数据包中的 DNS（Domain Name System）消息、TCP 报文、IP 分组、HTTP 协议消息的字段组成及作用。

四、 实验环境

- 1) Windows 系统主机或 Linux 系统主机；
- 2) Wireshark 软件

五、 实验过程与结果

1. 查看本机 IP 地址以及 MAC 地址

在 Windows 系统下，输入命令 `ipconfig /all`，结果如下：

```
无线局域网适配器 WLAN:

    连接特定的 DNS 后缀 . . . . . : 
    描述. . . . . : Intel(R) Wireless-AC 9560 160MHz
    物理地址. . . . . : 40-74-E0-85-4C-83
    DHCP 已启用 . . . . . : 是
    自动配置已启用. . . . . : 是
    IPv6 地址 . . . . . : 2001:da8:215:3c01::4:c3b(首选)
    获得租约的时间 . . . . . : 2019年10月22日 22:51:06
    租约过期的时间 . . . . . : 2019年11月22日 16:02:10
    IPv6 地址 . . . . . : 2001:da8:215:3c01:8983:7348:7710:e95e(首选)
    临时 IPv6 地址. . . . . : 2001:da8:215:3c01:b5d8:8094:fc57:8c6a(首选)
    本地链接 IPv6 地址. . . . . : fe80::8983:7348:7710:e95e%9(首选)
    IPv4 地址 . . . . . : 10.128.222.211(首选)
    子网掩码 . . . . . : 255.255.192.0
    获得租约的时间 . . . . . : 2019年10月23日 16:02:22
    租约过期的时间 . . . . . : 2019年10月23日 17:02:20
    默认网关. . . . . : fe80::7685:c4ff:fe11:2001%9
    . . . . . : 10.128.192.1
    DHCP 服务器 . . . . . : 10.3.9.2
    DHCPv6 IAID . . . . . : 71333088
    DHCPv6 客户端 DUID . . . . . : 00-01-00-01-25-1A-4A-5D-98-FA-9B-97-B1-E5
    DNS 服务器 . . . . . : 10.3.9.4
    . . . . . : 10.3.9.5
    TCP/IP 上的 NetBIOS . . . . . : 已启用
```

因此，可以得到：

- ✧ 本机的 IPv4 地址：10.128.222.211
- ✧ 本机的 MAC 地址：40-74-E0-85-4C-83
- ✧ DNS 服务器：10.3.9.4 和 10.3.9.5

2. 访问百度网站，并抓取访问过程中的数据包

在浏览器中输入网址 www.baidu.com，对百度进行访问。当浏览器加载出网页后，抓取到了数据包，如下所示：

文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)						
应用显示过滤器 ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
124	3.081508	39.156.66.179	10.128.222.211	TLSv1.2	60	Change Cipher Spec
125	3.081508	39.156.66.179	10.128.222.211	TLSv1.2	99	Encrypted Handshake Message
126	3.081547	10.128.222.211	39.156.66.179	TCP	54	53604 → 443 [ACK] Seq=518 Ack=148 Win=132096 Len=0
127	3.082091	10.128.222.211	39.156.66.179	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
128	3.082257	10.128.222.211	39.156.66.179	TLSv1.2	1252	Application Data
129	3.111158	10.128.222.211	10.3.9.4	DNS	73	Standard query 0x8b26 A sp1.baidu.com
130	3.111319	10.128.222.211	10.3.9.4	DNS	73	Standard query 0x3f85 AAAA sp1.baidu.com
131	3.113514	10.3.9.4	10.128.222.211	DNS	132	Standard query response 0x8b26 A sp1.baidu.com CN
132	3.113514	10.3.9.4	10.128.222.211	DNS	157	Standard query response 0x3f85 AAAA sp1.baidu.com
133	3.114477	10.128.222.211	39.156.66.14	TCP	66	53605 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460
134	3.123940	2001:da8:215:3c01:b...	2404:6800:4008:802:...	TCP	86	53606 → 443 [SYN] Seq=0 Win=64800 Len=0 MSS=1440
135	3.126839	39.156.66.14	10.128.222.211	TCP	66	443 → 53605 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
136	3.126906	10.128.222.211	39.156.66.14	TCP	54	53605 → 443 [ACK] Seq=1 Ack=1 Win=132352 Len=0

从图中可以看到，编号为 129，130 的帧的发送方为本机，接收方为 DNS 服务器（IP 地址见 ipconfig 的查询结果），根据数据帧的 Info 字段，推断该帧的功能为：本机向 DNS 服务器询问 www.baidu.com 的 IP 地址。

编号为 131, 132 的帧的发送方为 DNS 服务器, 接收方为本机, 根据数据帧的 Info 字段, 推断该帧的功能为: DNS 服务器将 IP 地址返回到本机。

3. 对一个数据包的数据帧进行分析

下面对之前抓取的 DNS 数据包进行分析 (以编号为 129 的数据包为例):

文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(V) 无线(W) 工具(T) 帮助(H)

应用显示过滤器 ... <Ctrl+>

No.	Time	Source	Destination	Protocol	Length	Info
127	3.082091	10.128.222.211	39.156.66.179	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Mess
128	3.082257	10.128.222.211	39.156.66.179	TLSv1.2	1252	Application Data
129	3.111158	10.128.222.211	10.3.9.4	DNS	73	Standard query 0x8b26 A sp1.baidu.com
130	3.111319	10.128.222.211	10.3.9.4	DNS	73	Standard query 0x3f85 AAAA sp1.baidu.com
131	3.113514	10.3.9.4	10.128.222.211	DNS	132	Standard query response 0x8b26 A sp1.baidu.c
132	3.113514	10.3.9.4	10.128.222.211	DNS	157	Standard query response 0x3f85 AAAA sp1.baic
133	3.114477	10.128.222.211	39.156.66.14	TCP	66	53605 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=
134	3.123940	2001:da8:215:3c01:b...	2404:6800:4008:802:...	TCP	86	53606 → 443 [SYN] Seq=0 Win=64800 Len=0 MSS=
135	3.126839	39.156.66.14	10.128.222.211	TCP	66	443 → 53605 [SYN, ACK] Seq=0 Ack=1 Win=8192
136	3.126906	10.128.222.211	39.156.66.14	TCP	54	53605 → 443 [ACK] Seq=1 Ack=1 Win=132352 Len=
137	3.127166	10.128.222.211	39.156.66.14	TLSv1.2	571	Client Hello
138	3.127918	203.208.41.90	10.128.222.211	UDP	153	443 → 65206 Len=111
139	3.130777	39.156.66.179	10.128.222.211	TCP	60	443 → 53604 [ACK] Seq=148 Ack=569 Win=30336

> Frame 129: 73 bytes on wire (584 bits), 73 bytes captured (584 bits) on interface 0
> Ethernet II, Src: IntelCor_85:4c:83 (40:74:e0:85:4c:83), Dst: NewH3CTe_11:20:01 (74:85:c4:11:20:01)
> Internet Protocol Version 4, Src: 10.128.222.211, Dst: 10.3.9.4
> User Datagram Protocol, Src Port: 59578, Dst Port: 53
> Domain Name System (query)

0000 74 85 c4 11 20 01 40 74 e0 85 4c 83 08 00 45 00 t...@t ..L...E.
0010 00 3b 0b 5e 00 00 80 11 00 00 0a 80 de d3 0a 03 ;.^.....
0020 09 04 e8 ba 00 35 00 27 fc 92 8b 26 01 00 00 015...'...&...
0030 00 00 00 00 00 00 03 73 70 31 05 62 61 69 64 75s p1.baidu
0040 03 63 6f 6d 00 00 01 00 01com.....

在数据包解析窗口中, 第二行 (Ethernet II...) 对应数据包的数据链路层帧头部分, 因此对这一部分进行详细分析:

> Frame 129: 73 bytes on wire (584 bits), 73 bytes captured (584 bits) on interface 0

> Ethernet II, Src: IntelCor_85:4c:83 (40:74:e0:85:4c:83), Dst: NewH3CTe_11:20:01 (74:85:c4:11:20:01)

> Destination: NewH3CTe_11:20:01 (74:85:c4:11:20:01)

Address: NewH3CTe_11:20:01 (74:85:c4:11:20:01)
.... ..0. = LG bit: Globally unique address (factory default)
.... ..0 = IG bit: Individual address (unicast)

> Source: IntelCor_85:4c:83 (40:74:e0:85:4c:83)

Address: IntelCor_85:4c:83 (40:74:e0:85:4c:83)
.... ..0. = LG bit: Globally unique address (factory default)
.... ..0 = IG bit: Individual address (unicast)

Type: IPv4 (0x0800)

> Internet Protocol Version 4, Src: 10.128.222.211, Dst: 10.3.9.4

> User Datagram Protocol, Src Port: 59578, Dst Port: 53

> Domain Name System (query)

在该部分中, 包括 Destination, Source, Type 三个字段:

✧ Destination: 74:85:c4:11:20:01

接收方的 MAC 地址，在该数据包中代表的是 DNS 服务器的 MAC 地址。

0000	74	85	c4	11	20	01	40	74	e0	85	4c	83	08	00	45	00
0010	00	3b	0b	5e	00	00	80	11	00	00	0a	80	de	d3	0a	03
0020	09	04	e8	ba	00	35	00	27	fc	92	8b	26	01	00	00	01
0030	00	00	00	00	00	00	03	73	70	31	05	62	61	69	64	75
0040	03	63	6f	6d	00	00	01	00	01							

✧ Source: 40:74:e0:85:4c:83

发送方的 MAC 地址，在该数据包中代表的是本机的 MAC 地址，这与之前通过 ipconfig 命令查看的 MAC 地址相符合。

0000	74	85	c4	11	20	01	40	74	e0	85	4c	83	08	00	45	00
0010	00	3b	0b	5e	00	00	80	11	00	00	0a	80	de	d3	0a	03
0020	09	04	e8	ba	00	35	00	27	fc	92	8b	26	01	00	00	01
0030	00	00	00	00	00	00	03	73	70	31	05	62	61	69	64	75
0040	03	63	6f	6d	00	00	01	00	01							

✧ Type: IPv4(0x0800)

使用的 IP 协议，可以知道该数据包中网络层使用的是 IPv4 协议。

0000	74	85	c4	11	20	01	40	74	e0	85	4c	83	08	00	45	00
0010	00	3b	0b	5e	00	00	80	11	00	00	0a	80	de	d3	0a	03
0020	09	04	e8	ba	00	35	00	27	fc	92	8b	26	01	00	00	01
0030	00	00	00	00	00	00	03	73	70	31	05	62	61	69	64	75
0040	03	63	6f	6d	00	00	01	00	01							

4. （选做部分）分析实验中抓取的 DNS（Domain Name System）消息、HTTP 消息、TCP 报文和 IP 分组，分析 TCP 报文字段组成及含义，分析 IP 分组的字段组成及含义。

1) DNS 协议过程分析

以 ping 命令访问 www.baidu.com，通过该事例进行说明：

```
C:\Users\45989>ping www.baidu.com

正在 Ping www.a.shifen.com [39.156.66.14] 具有 32 字节的数据:
来自 39.156.66.14 的回复: 字节=32 时间=9ms TTL=48
来自 39.156.66.14 的回复: 字节=32 时间=10ms TTL=48
来自 39.156.66.14 的回复: 字节=32 时间=7ms TTL=48
来自 39.156.66.14 的回复: 字节=32 时间=7ms TTL=48

39.156.66.14 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 = 7ms, 最长 = 10ms, 平均 = 8ms
```

数据包抓取情况如下：

*WLAN						
文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(U) 无线(W) 工具(I) 帮助(H)						
[(dns 66 (udp.port == 63172)) icmp]						
No.	Time	Source	Destination	Protocol	Length	Info
3459	28.452698	10.128.222.211	10.3.9.5	DNS	73	Standard query 0x4c3a A www.baidu.com
3461	28.456020	10.3.9.5	10.128.222.211	DNS	132	Standard query response 0x4c3a A www.baidu.com CNAME www.a.shifen.com
3463	28.464523	10.128.222.211	39.156.66.14	ICMP	74	Echo (ping) request id=0x0001, seq=5/1280, ttl=64 (reply in 3464)
3464	28.473665	39.156.66.14	10.128.222.211	ICMP	74	Echo (ping) reply id=0x0001, seq=5/1280, ttl=48 (request in 3463)
3465	29.467707	10.128.222.211	39.156.66.14	ICMP	74	Echo (ping) request id=0x0001, seq=6/1536, ttl=64 (reply in 3466)
3466	29.478114	39.156.66.14	10.128.222.211	ICMP	74	Echo (ping) reply id=0x0001, seq=6/1536, ttl=48 (request in 3465)
3468	30.471967	10.128.222.211	39.156.66.14	ICMP	74	Echo (ping) request id=0x0001, seq=7/1792, ttl=64 (reply in 3469)
3469	30.479880	39.156.66.14	10.128.222.211	ICMP	74	Echo (ping) reply id=0x0001, seq=7/1792, ttl=48 (request in 3468)
3472	31.475853	10.128.222.211	39.156.66.14	ICMP	74	Echo (ping) request id=0x0001, seq=8/2048, ttl=64 (reply in 3473)
3473	31.483083	39.156.66.14	10.128.222.211	ICMP	74	Echo (ping) reply id=0x0001, seq=8/2048, ttl=48 (request in 3472)

在该过程中，产生了 2 个 DNS 数据包以及 8 个 ICMP 数据包。

其中，DNS 数据包的作用是向域名服务器请求某域名（www.baidu.com）对应的 IP 地址，ICMP 包的作用是执行 ping 命令的发送、应答过程（因为一共有 4 次 ping，因此总共有 8 个 ICMP 数据包）。

DNS 域名解析过程如下：

- ✧ 客户端向 DNS 服务器发送查询信息，查询 www.baidu.com 的 IP 地址（第 3459 行）。
- ✧ DNS 服务器收到查询请求，并发送应答消息，包中包含该域名对应的 IP 地址（第 3461 行）。

2) DNS 字段组成分析

第二个 DNS 包的详细结构如下：

```
> Internet Protocol Version 4, Src: 10.3.9.5, Dst: 10.128.222.211
> User Datagram Protocol, Src Port: 53, Dst Port: 63172
  > Domain Name System (response)
    Transaction ID: 0x4c3a
    > Flags: 0x8180 Standard query response, No error
    Questions: 1
    Answer RRs: 3
    Authority RRs: 0
    Additional RRs: 0
    > Queries
      > www.baidu.com: type A, class IN
    > Answers
      > www.baidu.com: type CNAME, class IN, cname www.a.shifen.com
      > www.a.shifen.com: type A, class IN, addr 39.156.66.14
      > www.a.shifen.com: type A, class IN, addr 39.156.66.18
      [Request In: 3459]
      [Time: 0.003322000 seconds]
```

下面对该数据包的 DNS 部分进行详细分析：

- ✧ Transaction ID 字段记录了本次事务的 ID，两个 DNS 数据包的 Transaction ID 字段的值是相同的。
- ✧ Flags 字段的详细信息如下所示：

```

Flags: 0x8180 Standard query response, No error
1... .. = Response: Message is a response
.000 0... .. = Opcode: Standard query (0)
....0... .. = Authoritative: Server is not an authority for domain
....0... .. = Truncated: Message is not truncated
....1... .. = Recursion desired: Do query recursively
....1... .. = Recursion available: Server can do recursive queries
....0... .. = Z: reserved (0)
....0... .. = Answer authenticated: Answer/authority portion was not authenticated by the server
....0... .. = Non-authenticated data: Unacceptable
....0000 = Reply code: No error (0)

```

该字段记录了该数据包的很多信息，比如该消息的类型是 query 还是 response，包中是否产生错误等等。

✧ Questions 字段表示问题计数，由于本次查询的问题只有一个（查询 www.baidu.com 的 IP 地址），因此该字段的值为 1。

✧ Answer RR 字段表示应答消息计数，可以知道该数据包中包含 3 个应答消息。

✧ Queries 字段记录了详细的问题，而查询的是 www.baidu.com 的 IP 地址。

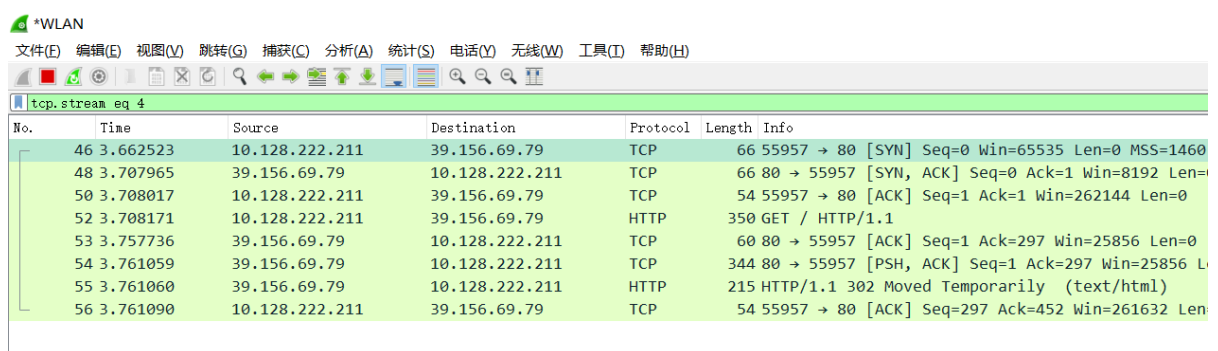
✧ Queries 字段记录了详细的应答消息

■ 回答 1 类型为 cname，代表别名，别名为 www.a.shifen.com

■ 回答 2 和回答 3 均为主机地址，IP 地址为 39.156.66.14 和 39.156.66.18

3) TCP 连接建立过程——三次握手

通过浏览器访问 www.baidu.com，并在 wireshark 中进行数据包抓取，过滤出 TCP 和 ipv4 的数据包之后，对 TCP 数据流进行追踪，得到的结果如下：



The screenshot shows a Wireshark packet capture on the *WLAN interface. The filter is set to 'tcp.stream eq 4'. The packet list shows the following packets:

No.	Time	Source	Destination	Protocol	Length	Info
46	3.662523	10.128.222.211	39.156.69.79	TCP	66	55957 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460
48	3.707965	39.156.69.79	10.128.222.211	TCP	66	80 → 55957 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
50	3.708017	10.128.222.211	39.156.69.79	TCP	54	55957 → 80 [ACK] Seq=1 Ack=1 Win=262144 Len=0
52	3.708171	10.128.222.211	39.156.69.79	HTTP	350	GET / HTTP/1.1
53	3.757736	39.156.69.79	10.128.222.211	TCP	60	80 → 55957 [ACK] Seq=1 Ack=297 Win=25856 Len=0
54	3.761059	39.156.69.79	10.128.222.211	TCP	344	80 → 55957 [PSH, ACK] Seq=1 Ack=297 Win=25856 Len=0
55	3.761060	39.156.69.79	10.128.222.211	HTTP	215	HTTP/1.1 302 Moved Temporarily (text/html)
56	3.761090	10.128.222.211	39.156.69.79	TCP	54	55957 → 80 [ACK] Seq=297 Ack=452 Win=261632 Len=0

前三个 TCP 数据包即为 TCP 建立连接所用的三个数据包，使用了三次握手的方法，下面对这三个数据包的功能进行分析，从而对三次握手的整个过程进行描述：

➤ [SYN]：客户端首先向服务端发送 syn=1 的包（第 46 行），该包用于与服务端建立同步，发送之后，客户端等待服务器的响应。

➤ [SYN, ACK]：服务器的响应（第 48 行）。一旦服务器接收到客户端的 SYN 报

文，就读取报文的序列号并且使用此编号作为响应，也就是说它告知客户机，服务器接收到了 SYN 报文，通过对原 SYN 报文序列号加一并且作为响应编号来实现，之后客户端就知道服务器能够接收通信。

- [ACK]: 客户端对服务器发送的确认报文（第 50 行），告诉服务器客户端接收到了 SYN/ACK 报文，并且与前一步一样客户端也将序列号加一，此包发送完毕，客户端和服务端进入 ESTABLISHED 状态，完成三次握手。

4) TCP 报文字段组成

下图为 TCP 部分的全部内容：

```
> Frame 46: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
> Ethernet II, Src: IntelCor_85:4c:83 (40:74:e0:85:4c:83), Dst: NewH3CTe_11:20:01 (74:85:c4:11:20:01)
> Internet Protocol Version 4, Src: 10.128.222.211, Dst: 39.156.69.79
v Transmission Control Protocol, Src Port: 55957, Dst Port: 80, Seq: 0, Len: 0
  Source Port: 55957
  Destination Port: 80
  [Stream index: 4]
  [TCP Segment Len: 0]
  Sequence number: 0 (relative sequence number)
  [Next sequence number: 0 (relative sequence number)]
  Acknowledgment number: 0
  1000 .... = Header Length: 32 bytes (8)
> Flags: 0x002 (SYN)
  Window size value: 65535
  [Calculated window size: 65535]
  Checksum: 0x5665 [unverified]
  [Checksum Status: Unverified]
  Urgent pointer: 0
> Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), SACK permitted
> [Timestamps]
```

一些字段的具体作用：

- ✧ Source Port: 发送方的端口号
- ✧ Destination Port: 接收方的端口号
- ✧ Sequence number: 报文的序列号（相对序列号）
- ✧ Acknowledgment number: 确认序列号，用来表示报文是否包含 ACK 信息。
- ✧ Header length: 报文头部长度的。
- ✧ Flags: 标记字段，记录了报文的性质，比如 ACK, SYN, RST, FIN 等等，下图为该字段的详细内容。


```

  ▾ Flags: 0x010 (ACK)
    000. .... = Reserved: Not set
    ...0 .... = Nonce: Not set
    .... 0... = Congestion Window Reduced (CWR): Not set
    .... .0.. = ECN-Echo: Not set
    .... ..0. = Urgent: Not set
    .... ...1 = Acknowledgment: Set
    .... .... 0... = Push: Not set
    .... .... .0.. = Reset: Not set
    .... .... ..0. = Syn: Not set
    .... .... ...0 = Fin: Not set

```

✧ Window size value: 流量控制的窗口大小，通过控制该字段，防止发送方发送消息速度过快，从而淹没接收方。

✧ Checknum: 数据段的校验和。

5) HTTP 协议过程

仍然以使用浏览器访问 www.baidu.com 为例：

*WLAN

文件(E) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(I) 帮助(H)

tcp.stream eq 4

No.	Time	Source	Destination	Protocol	Length	Info
46	3.662523	10.128.222.211	39.156.69.79	TCP	66	55957 → 80 [SYN] Seq=0 Win=65535 Len=0
48	3.707965	39.156.69.79	10.128.222.211	TCP	66	80 → 55957 [SYN, ACK] Seq=0 Ack=1 Win=0 Len=0
50	3.708017	10.128.222.211	39.156.69.79	TCP	54	55957 → 80 [ACK] Seq=1 Ack=1 Win=262 Len=0
52	3.708171	10.128.222.211	39.156.69.79	HTTP	350	GET / HTTP/1.1
53	3.757736	39.156.69.79	10.128.222.211	TCP	60	80 → 55957 [ACK] Seq=1 Ack=297 Win=2 Len=0
54	3.761059	39.156.69.79	10.128.222.211	TCP	344	80 → 55957 [PSH, ACK] Seq=1 Ack=297 Win=0 Len=344
55	3.761060	39.156.69.79	10.128.222.211	HTTP	215	HTTP/1.1 302 Moved Temporarily (text/html)
56	3.761090	10.128.222.211	39.156.69.79	TCP	54	55957 → 80 [ACK] Seq=297 Ack=452 Win=0 Len=0

HTTP 协议过程如下：

- ✧ 当客户端和服务端通过三次握手建立 TCP 连接（第 46，48，50 行，详细过程见 2）中的分析）。
- ✧ 客户端发送 HTTP 请求，请求获得服务端中的网页信息（第 52 行）。在图中，客户端发送了 GET 请求，协议版本为 HTTP/1.1。
- ✧ 服务器收到 GET 请求之后，发送 ACK 消息（第 53 行）。
- ✧ 服务器发送 HTTP 消息，里面包含具体的网页信息（第 55 行）。查看该数据包的数据字段，可以发现是一个完整的 HTTP 网页代码：

0000	40 74 e0 85 4c 83 01 20 11 c4 85 74 08 00 45 48	@t...L... ..t...EH
0010	00 c9 ae 69 40 00 22 06 53 3f 27 9c 45 4f 0a 80	...i@...". S?'EO...
0020	de d3 00 50 da 95 0a 64 f6 f4 52 b3 fe bc 50 18	...P...d...R...P...
0030	03 28 ed dc 00 00 3c 68 74 6d 6c 3e 0d 0a 3c 68	..(....<h tml>...<h
0040	65 61 64 3e 3c 74 69 74 6c 65 3e 33 30 32 20 46	ead><tit le>302 F
0050	6f 75 6e 64 3c 2f 74 69 74 6c 65 3e 3c 2f 68 65	ound</ti tle></he
0060	61 64 3e 0d 0a 3c 62 6f 64 79 20 62 67 63 6f 6c	ad>...<bo dy bgcol
0070	6f 72 3d 22 77 68 69 74 65 22 3e 0d 0a 3c 63 65	or="whit e">...<ce
0080	6e 74 65 72 3e 3c 68 31 3e 33 30 32 20 46 6f 75	nter><h1>302 Fou
0090	6e 64 3c 2f 68 31 3e 3c 2f 63 65 6e 74 65 72 3e	nd</h1>< /center>
00a0	0d 0a 3c 68 72 3e 3c 63 65 6e 74 65 72 3e 62 66	...<hr><c enter>bf
00b0	65 2f 31 2e 30 2e 38 2e 31 38 3c 2f 63 65 6e 74	e/1.0.8. 18</cent
00c0	65 72 3e 0d 0a 3c 2f 62 6f 64 79 3e 0d 0a 3c 2f	er>...</b ody>...</
00d0	68 74 6d 6c 3e 0d 0a	html>...

✧ 客户端收到 HTTP 消息，返回一个 ACK 消息（第 56 行）。

6) HTTP 协议消息格式

下图为 HTTP 请求消息的详细结构：

```

v Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n
    Accept: text/html, application/xhtml+xml, image/jxr, */*\r\n
    Accept-Language: zh-Hans-CN,zh-Hans;q=0.8,en-US;q=0.5,en;q=0.3\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko\r\n
    Accept-Encoding: gzip, deflate\r\n
    Host: baidu.com\r\n
    Connection: Keep-Alive\r\n
    \r\n
    [Full request URI: http://baidu.com/]
    [HTTP request 1/1]
    [Response in frame: 55]

```

查看数据包的具体数据内容，可以发现 HTTP 字段是以 ASCII 码来传输的：

0000	74 85 c4 11 20 01 40 74 e0 85 4c 83 08 00 45 00	t... ..@t...L...E..
0010	01 50 f4 9f 40 00 80 06 00 00 0a 80 de d3 27 9c	..P...@...
0020	45 4f da 95 00 50 52 b3 fd 94 0a 64 f5 d2 50 18	EO...PR... ..d...P...
0030	04 00 57 81 00 00 47 45 54 20 2f 20 48 54 54 50	..W...GE T / HTTP
0040	2f 31 2e 31 0d 0a 41 63 63 65 70 74 3a 20 74 65	/1.1..Ac cept: te
0050	78 74 2f 68 74 6d 6c 2c 20 61 70 70 6c 69 63 61	xt/html, applica
0060	74 69 6f 6e 2f 78 68 74 6d 6c 2b 78 6d 6c 2c 20	tion/xht ml+xml,
0070	69 6d 61 67 65 2f 6a 78 72 2c 20 2a 2f 2a 0d 0a	image/jx r, */*..
0080	41 63 63 65 70 74 2d 4c 61 6e 67 75 61 67 65 3a	Accept-L anguage:
0090	20 7a 68 2d 48 61 6e 73 2d 43 4e 2c 7a 68 2d 48	zh-Hans -CN,zh-H
00a0	61 6e 73 3b 71 3d 30 2e 38 2c 65 6e 2d 55 53 3b	ans;q=0. 8,en-US;
00b0	71 3d 30 2e 35 2c 65 6e 3b 71 3d 30 2e 33 0d 0a	q=0.5,en ;q=0.3..
00c0	55 73 65 72 2d 41 67 65 6e 74 3a 20 4d 6f 7a 69	User-Age nt: Mozi
00d0	6c 6c 61 2f 35 2e 30 20 28 57 69 6e 64 6f 77 73	lla/5.0 (Windows
00e0	20 4e 54 20 31 30 2e 30 3b 20 57 4f 57 36 34 3b	NT 10.0 ; WOW64;
00f0	20 54 72 69 64 65 6e 74 2f 37 2e 30 3b 20 72 76	Trident /7.0; rv
0100	3a 31 31 2e 30 29 20 6c 69 6b 65 20 47 65 63 6b	;11.0) l ike Geck
0110	6f 0d 0a 41 63 63 65 70 74 2d 45 6e 63 6f 64 69	o...Accep t-Encodi
0120	6e 67 3a 20 67 7a 69 70 2c 20 64 65 66 6c 61 74	ng: gzip , deflat
0130	65 0d 0a 48 6f 73 74 3a 20 62 61 69 64 75 2e 63	e...Host: baidu.c
0140	6f 6d 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20	om...Conn ection:
0150	4b 65 65 70 2d 41 6c 69 76 65 0d 0a 0d 0a	Keep-Ali ve....

一些重要的字段：

✧ GET / HTTP/1.1

- GET: 客户端的请求方法
- /: 请求的 URI

■ HTTP/1.1: 协议的版本

- ✧ 请求头: 包含许多有关的客户端环境和请求正文的有用信息。例如, 请求头可以声明浏览器所用的语言, 请求正文的长度等。

```
Accept: text/html, application/xhtml+xml, image/jxr, */*\r\n
Accept-Language: zh-Hans-CN,zh-Hans;q=0.8,en-US;q=0.5,en;q=0.3\r\n
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko\r\n
Accept-Encoding: gzip, deflate\r\n
Host: baidu.com\r\n
Connection: Keep-Alive\r\n
```

7) IP 字段组成

下图为 HTTP 请求消息的 IP 字段:

```
Internet Protocol Version 4, Src: 10.128.222.211, Dst: 39.156.69.79
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 336
    Identification: 0xf49f (62623)
  > Flags: 0x4000, Don't fragment
    Time to live: 128
    Protocol: TCP (6)
    Header checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source: 10.128.222.211
    Destination: 39.156.69.79
```

其中的一些重要字段:

- ✧ Version: 版本, 该包使用的是 IPv4 协议。
- ✧ Header Length: 首部长度, 该包中 IP 字段的总长度为 20 字节。
- ✧ Differentiated Services Field: 服务类型。
- ✧ Total Length: 数据包长度。
- ✧ Flags: 标记字段。
- ✧ Protocol: 上层协议, 该包中 IP 协议的上层协议为 TCP 协议。
- ✧ Header checksum: 首部校验和, 检查 IP 报头在传输过程中是否损坏。
- ✧ Source: 10.128.213.85 发送方的 IP 地址, 该包中为本机。
- ✧ Destination: 14.215.177.38 接收方的 IP 地址, 该包中为 HTTP 服务器。

六、 心得体会

通过使用 Wireshark 软件对网络数据包进行抓取, 对日常生活中的一些网络使用有了更清晰的了解, 对各种协议字段组成及含义有了更深的了解, 对数据链路层的协议数据单元及功能有了更深的认识。