《网络与通信》课程实验报告

实验三：数据包结构分析

姓名	汪雨卿	院系	计算机学院	学号	19120191
任课教师	张瑞		指导教师	张瑞	
实验地点	计 706		实验时间	周三 5-6	
实验课表现	出勤、表现得分(10)		实验报告得分(40)		实验总分	
	操作结果得分(50)					

实验目的：

1. 了解 Sniffer 的工作原理，掌握 Sniffer 抓包、记录和分析数据包的方法；
2. 在这个实验中，你将使用抓包软件捕获数据包，并通过数据包分析每一层协议。

实验内容：

使用抓包软件捕获数据包，并通过数据包分析每一层协议。

实验要求：（学生对预习要求的回答）（10 分）　　得分：

- 常用的抓包工具

1. Flidder

Fiddler是位于客户端和服务器端的HTTP代理，也是目前最常用的http抓包工具之一 。 它能够记录客户端和服务器之间的所有HTTP请求，可以针对特定的HTTP请求，分析请求数据、设置断点、调试web应用、修改请求的数据，甚至可以修改服务器返回的数据，功能非常强大，是web调试的利器。

2. Httpwatch

3. Hping

Hping是最受欢迎和免费的抓包工具之一。它允许你修改和发送自定义的ICMP，UDP，TCP和原始IP数据包。此工具由网络管理员用于防火墙和网络的安全审计和测试。

4. Ostinato

Ostinato是一个开源和跨平台网络包生成器和分析工具。它带有GUI界面，使其易于使用和理解。它支持Windows，Linux，BSD和Mac OS X平台。您也可以尝试在其他平台上使用它。

5. Scapy

Scapy是另一种不错的交互式数据包处理工具。这个工具是用Python编写的。它可以解码或伪造大量协议的数据包。Scapy是一个值得尝试的工具。您可以执行各种任务，包括扫描，跟踪，探测，单元测试，网络发现。

6. Libcrafter

Libcrafter非常类似于Scapy。这个工具是用C ++编写的，使得更容易创建和解码网络数据包。它可以创建和解码大多数一般协议的数据包，捕获数据包和匹配请求或回复。这个工具可以多线程执行各种任务。

7. Yersinia

Yersinia是一个强大的网络渗透测试工具，能够对各种网络协议进行渗透测试。如果你正在寻找抓包工具，你可以试试这个工具。

8. packetETH

packETH是另一个数据包处理工具。它是一个Linux GUI的以太网工具。它允许你快速创建和

发送数据包序列。与此列表中的其他工具一样，它支持各种协议来创建和发送数据包。你还可以设置数据包数量和数据包之间的延迟，还可以在此工具中修改各种数据包内容。

9. wireshark

实验过程中遇到的问题如何解决的？（10 分）	得分：

问题 1：最开始发现限制协议后捕获的数据包中层数会缺少 TCP，HTTP 等等协议？

发现问题后，查阅相关资料了解到每一次互联网数据传输中所涉及到的数据包的层数是会不同的。同时利用 Ping 命令实现的测试，只是基于 icmp 协议，和真实的数据传输还是会有很大的区别。因此，后来改为直接访问网站获取数据包进行分析。

问题 2：利用 wireshark 所捕捉到的数据包分析以太网头部信息时，对于 MAC 源地址的对应方式产生困惑。不明白其中的对应方式。

首先，在 wireshark 中可以很轻松的锁定 Destination 后面对应的 16 进制数字的位数和位数。通过对于几个不同的数据包中，该部分的信息比对发现相同厂家对应的 16 进制数码是相同的。进而产生了猜测，该部分数码是和开发厂家有关。因此，再根据该猜测去搜集资料，最后证实了自己的猜想。

问题 3：在分析 HTTP 报文结构的过程对于 Statuscode 对应的不同数码的含义产生疑惑。

首先，联想到自己在网页浏览的过程中经常看到 404 not found，304 not found 的反馈信息。进而判断得出，该状态码和访问网页状态的情况有关。从而，搜集资料了解到不同状态码代表着不同的状态。例如 200 表示成功访问。

本次实验的体会（结论）（10 分）	得分：

本次对于数据包分析的实验，通过抓取一个个真实传输的数据包，清晰的将计算机网络课程中所涉及到的理论知识真实的转换成肉眼可见，可以研究的数据包。最开始面对成行成列的 16 进制数据，以及一层层的英文字段，我并不明白他们的含义以及对应的关系。但随着我慢慢开始学习报文的结构，研究每个字段的含义，它传输的数据，以及它如何实现二进制表达的过程，我逐渐了解了网络数据传输背后的秘密。也知道了每一次数据传输的背后，都是一个个协议，一层层工序通过增加标志，增加校验码，再实现同样还原，才能够实现数据的低差错传输。

此外，通过学习 wireshark 抓包软件的使用，我使用过滤和筛选操作找到了自己想要抓取的数据包类型。并且对于这些数据包进行学习，也对于计算机网络传输的 7 个层次有了更深入的了解。

思考题：（10 分）

思考题 1：（4 分）	得分：

写出捕获的数据包格式。

1. Frame 1 （物理层）

2. Ethernet II（数据链路层）: 目的 MAC 地址（6 bytes）+ 源 MAC（6 bytes）+ 类型（2 bytes）+数据字段（46-1500 bytes）+ 校验（4 bytes）

3. Internet Protocol Version 4（网络层，共 20 bytes）：IP 协议头+载荷

4. *ARP 报文结构

ARP报文结构如下：

物理网络类型		协议类型
物理地址长度	协议地址长度	操作
发送方物理地址		
发送方IP地址		
目标物理地址		
目标IP地址		

物理网络类型：也称为硬件类型，指明硬件类型，以太网为1。

协议类型：指明发送者映射到数据链路标识的网络协议的类型，IP对应0x0806.

物理地址长度：MAC地址长度为6，单位字节。

协议地址长度：网络层地址的长度，即IP地址长度为4，单位字节。

操作：操作指明ARP的操作类型，ARP请求为1，ARP响应为2，RARP请求为3，RARP响应为4。

在以太网环境下的ARP报文，硬件地址为48位。

5.1 Transmission Control Protocol（传输层）

5.2 UDP 报文结构 （传输层）

16	32bit
Source port	Destination port
Length	Checksum
Data	

思考题2：（6分）	得分：

写出实验过程并分析实验结果。

1. 设置网卡：选择菜单栏上Capture -> Option，勾选WLAN网卡。这里用ipconfig查看了IP地址点击Start。启动抓包。
2. 获取一个数据包，点开详情面板。
3. 各层数据分析

- 数据链路层：以太网帧头部信息

Ethernet II, Src: IntelCor_03:85:31 (98:2c:bc:03:85:31), Dst: RuijieNe_7d:49:25 (14:14:4b:7d:49:25)

Destination: RuijieNe_7d:49:25 (14:14:4b:7d:49:25)　　# 目的MAC

Source: IntelCor_03:85:31 (98:2c:bc:03:85:31)　　# 源MAC

对应的数据位置：

其中农对于MAC地址，前三个字节是由IEEE的注册管理机构RA负责给不同厂家分配的代码(高位24位)，也称为“编制上唯一的标识符”(Organizationally Unique Identifier)，后三个字节(低位24位)由各厂家自行指派给生产的适配器接口，称为扩展标识符（唯一性）。一个地址块可以生成2的24次方不同的地址。

```
0000  14 14 4b 7d 49 25 98 2c  bc 03 85 31 08 00 45 00   ··K}I%·, ···1··E·
0010  00 ff f4 95 40 00 80 06  00 00 0a 58 14 99 b6 3d   ····@··· ···X···=
0020  c8 07 ef c4 00 50 3f 71  ff 9d 89 b5 f0 74 50 18   ·····P?q ·····tP·
0030  01 fe 9e 27 00 00 48 45  41 44 20 2f 72 6f 62 6f   ···'··HE AD /robo
0040  74 73 2e 74 78 74 20 48  54 54 50 2f 31 2e 31 0d   ts.txt H TTP/1.1·
0050  0a 48 6f 73 74 3a 20 77  77 77 2e 62 61 69 64 75   ·Host: w ww.baidu
0060  2e 63 6f 6d 0d 0a 43 6f  6f 6b 69 65 3a 20 42 41   .com··Co okie: BA
0070  49 44 55 49 44 3d 37 38  33 30 31 43 45 34 31 45   IDUID=78 301CE41E
0080  33 34 35 35 35 46 37 31  41 30 33 38 36 34 32 46   34555F71 A038642F
0090  41 30 39 41 30 32 3a 46  47 3d 31 0d 0a 43 6f 6e   A09A02:F G=1··Con
```

```
0000  14 14 4b 7d 49 25 98 2c  bc 03 85 31 08 00 45 00   ··K}I%·, ···1··E·
0010  00 ff f4 95 40 00 80 06  00 00 0a 58 14 99 b6 3d   ····@··· ···X···=
0020  c8 07 ef c4 00 50 3f 71  ff 9d 89 b5 f0 74 50 18   ·····P?q ·····tP·
0030  01 fe 9e 27 00 00 48 45  41 44 20 2f 72 6f 62 6f   ···'··HE AD /robo
0040  74 73 2e 74 78 74 20 48  54 54 50 2f 31 2e 31 0d   ts.txt H TTP/1.1·
0050  0a 48 6f 73 74 3a 20 77  77 77 2e 62 61 69 64 75   ·Host: w ww.baidu
0060  2e 63 6f 6d 0d 0a 43 6f  6f 6b 69 65 3a 20 42 41   .com··Co okie: BA
0070  49 44 55 49 44 3d 37 38  33 30 31 43 45 34 31 45   IDUID=78 301CE41E
0080  33 34 35 35 35 46 37 31  41 30 33 38 36 34 32 46   34555F71 A038642F
0090  41 30 39 41 30 32 3a 46  47 3d 31 0d 0a 43 6f 6e   A09A02:F G=1··Con
```

```
0000  14 14 4b 7d 49 25 98 2c  bc 03 85 31 08 00 45 00   ··K}I%·, ···1··E·
0010  00 ff f4 95 40 00 80 06  00 00 0a 58 14 99 b6 3d   ····@··· ···X···=
0020  c8 07 ef c4 00 50 3f 71  ff 9d 89 b5 f0 74 50 18   ·····P?q ·····tP·
0030  01 fe 9e 27 00 00 48 45  41 44 20 2f 72 6f 62 6f   ···'··HE AD /robo
0040  74 73 2e 74 78 74 20 48  54 54 50 2f 31 2e 31 0d   ts.txt H TTP/1.1·
0050  0a 48 6f 73 74 3a 20 77  77 77 2e 62 61 69 64 75   ·Host: w ww.baidu
0060  2e 63 6f 6d 0d 0a 43 6f  6f 6b 69 65 3a 20 42 41   .com··Co okie: BA
0070  49 44 55 49 44 3d 37 38  33 30 31 43 45 34 31 45   IDUID=78 301CE41E
0080  33 34 35 35 35 46 37 31  41 30 33 38 36 34 32 46   34555F71 A038642F
0090  41 30 39 41 30 32 3a 46  47 3d 31 0d 0a 43 6f 6e   A09A02:F G=1··Con
```

- 网络层：IP报文结构分析

```
Internet Protocol Version 4, Src: 112.80.248.75, Dst: 10.88.20.153
    0100 .... = Version: 4                       // 版本
    .... 0101 = Header Length: 20 bytes (5)      // 首部长度
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 40                             // 总长度
    Identification: 0x8533 (34099)               // 标识字段
  > Flags: 0x40, Don't fragment                  // 标志
    Fragment Offset: 0                           // 段偏移
    Time to Live: 47                             // 寿命
    Protocol: TCP (6)                            // 协议：携带何种协议报文
    Header Checksum: 0x3f10 [validation disabled]  // 头部检验和：对IP首部的校验和
    [Header checksum status: Unverified]
    Source Address: 112.80.248.75                // 源IP地址
    Destination Address: 10.88.20.153            // 报文目的地址
```

1. Version + Header Length : 45 (1 bytes)
2. Differentiated Services Field: 00 (1bytes)
3. Total Length: 00 28 (2 bytes)　　// 40的十六进制表示
4. Identification: 85 33 (2 bytes)　　// 34099的十六进制表示
5. Flags: 0x40 (1 bytes)
6. Fragment Offset: 0x00 (1 bytes)
7. Time to Live: 0x2f (1 bytes)
8. Protocol: 0x06 (1 bytes)
9. Header Checksum: 0x3f10 (2 bytes)

10. Source Address: 70 50 f8 4b = 112.80.248.75 (4 bytes)

11. Destination Address: 0a 58 14 99 = 10.88.20.153 (4 bytes)

```
98 2c bc 03 85 31 14 14  4b 7d 49 25 08 00 45 00
00 28 85 33 40 00 2f 06  3f 10 70 50 f8 4b 0a 58
14 99 00 50 ed b7 27 d1  d8 00 47 b9 c3 cf 50 10
03 d0 2b 15 00 00 00 00
```

- ARP报文结构分析

```
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: Tp-LinkT_92:c2:eb (cc:08:fb:92:c2:eb)
    Sender IP address: 192.168.0.1
    Target MAC address: IntelCor_03:85:31 (98:2c:bc:03:85:31)
    Target IP address: 192.168.0.21
```

1. Hardware type: Ethernet (1) (0x0001)

// 硬件类型，2 bytes，定义运行ARP网络的类型。每个局域网基于其类型被指派给一个整数。

2. Protocol type: IPv4 (0x0800)

// 协议类型，2 bytes，定义协议的类型。ARP可用于任何高层协议。比如这里的IPV4对应0800。

3. Hardware size: 6 (0x06)

// 硬件长度，1 byte，定义以字节位单位的物理地址长度。

4. Protocol size: 4 (0x04)

// 协议长度，1 byte，定义以字节位单位的逻辑地址长度。

5. Opcode: reply (2) (0x0002)

// 操作，2 bytes，定义分组的类型。ARP请求（1），ARP回答（2）。

6. Sender MAC address: Tp-LinkT_92:c2:eb (cc:08:fb:92:c2:eb)

// 发送站硬件地址，可变长度字段，定义发送站的物理地址。例，对于以太网为6bytes。

7. Sender IP address: 192.168.0.1

// 发送站协议地址，可变长度字段，定义发送站的逻辑地址。例，对于IP地址为4bytes。

8. Target MAC address: IntelCor_03:85:31 (98:2c:bc:03:85:31)

// 目标硬件地址，可变长度字段，定义目标的物理地址。例，对于以太网为6bytes。

9. Target IP address: 192.168.0.21

// 目标协议地址，可变长度字段，定义目标的逻辑地址。例，对于IP地址为4bytes。

```
0000  98 2c bc 03 85 31 cc 08  fb 92 c2 eb 08 06 00 01   ·,···1·· ·····..
0010  08 00 06 04 00 02 cc 08  fb 92 c2 eb c0 a8 00 01   ........ ........
0020  98 2c bc 03 85 31 c0 a8  00 15 00 00 00 00 00 00   ·,···1·· ........
0030  00 00 00 00 00 00 00 00  00 00 00 00              ........ ....
```

- 传输层：TCP报文结构分析

```
Transmission Control Protocol, Src Port: 80, Dst Port: 60855, Seq: 1, Ack: 1, Len: 0
    Source Port: 80
    Destination Port: 60855
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 668063744
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 1203356623
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window: 976
    [Calculated window size: 976]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0x2b15 [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
  > [Timestamps]
```

1. Source Port: 80 (0x0050)

// 源端口名称（端口号）,2 bytes

2. Destination port:60855 (0xedb7)

// 目的端口号,2 bytes

3. Sequence Number: 1 (relative sequence number) (0x27d1d800)

// 序列号（相对序列号）, 4 bytes

4. 0101 .... = Header Length: 20 bytes (5) (0x50)

// 头部长度, 1 byte

5. Flags: 0x010 (ACK) (0x5010)

// TCP标记字段, 2 bytes

6. Window: 976 (0x03d0)

// 流量控制的窗口大小, 2 bytes

7. Checksum: 0x2b15 [unverified] (0x2b15)

// TCP数据段的校验和

8. Urgent Pointer: 0

```
98 2c bc 03 85 31 14 14  4b 7d 49 25 08 00 45 00    ·,···1·· K}I%··E·
00 28 85 33 40 00 2f 06  3f 10 70 50 f8 4b 0a 58    ·(·3@·/· ?·pP·K·X
14 99 00 50 ed b7 27 d1  d8 00 47 b9 c3 cf 50 10    ···P··'· ··G···P·
03 d0 2b 15 00 00 00 00                             ··+·····
```

- **UDP报文结构分析**

```
User Datagram Protocol, Src Port: 8000, Dst Port: 4021
    Source Port: 8000
    Destination Port: 4021
    Length: 95
    Checksum: 0xb837 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 0]
  > [Timestamps]
    UDP payload (87 bytes)
```

1. Source Port: 8000 (0x1f40)

// 源端口, 2 bytes

2. Destination Port: 4021 (0x0fb5)

// 目的端口，2 bytes

3. Length: 95 (0x005f)

// 长度，2 bytes

4. Checksum: 0xb837 [unverified] (0xb837)

// 检验和，2 bytes

5. UDP payload (87 bytes) (0x02 - 0x03)

- **应用层：DNS报文结构分析**

```
Domain Name System (query)
    Transaction ID: 0x8b3e
  > Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  > Queries
```

1. Transaction ID: 0x8b3e (0x8b3e)

// 事务ID，2 bytes

2. Flags: 0x0100 Standard query (0x0100)

// 报文中的标志字段

```
▾ Flags: 0x0100 Standard query
    0... .... .... .... = Response: Message is a query
    .000 0... .... .... = Opcode: Standard query (0)
    .... ..0. .... .... = Truncated: Message is not truncated
    .... ...1 .... .... = Recursion desired: Do query recursively
    .... .... .0.. .... = Z: reserved (0)
    .... .... ...0 .... = Non-authenticated data: Unacceptable
  Questions: 1
```

3. Questions: 1 (0x0001)

// 问题计数，这里有1个问题，2 bytes

4. Answer RRs: 0 (0x0000)

// 回答资源计数， 2 bytes

5. Authority RRs: 0 (0x0000)

// 权威名称服务器计数， 2 bytes

6. Additional RRs: 0 (0x0000)

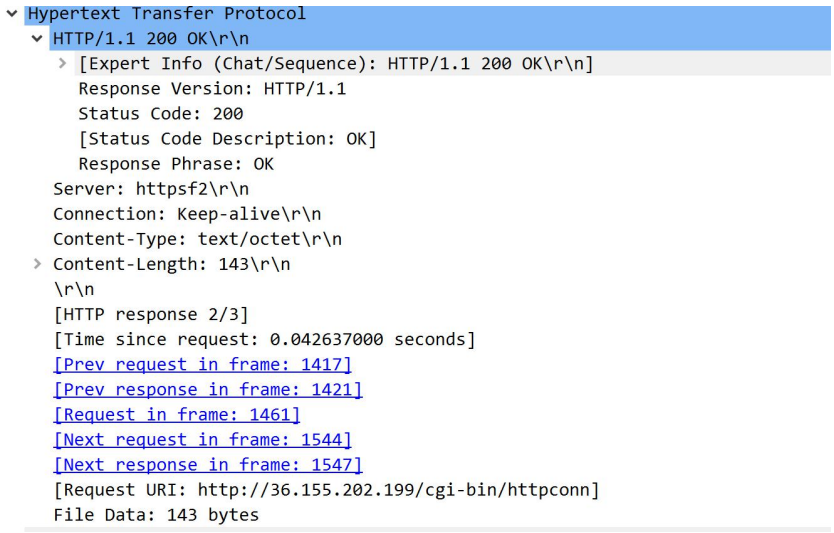
// 附加资源记录数， 2 bytes

7. Queries：www.baidu.com: type AAAA, class IN

// 查询请求

```
0000  14 14 4b 7d 49 25 98 2c  bc 03 85 31 86 dd 60 02   ··K}I%·, ···1··`·
0010  ac dd 00 27 11 40 20 01  0d a8 80 06 09 49 1d a5   ···'·@ · ·····I··
0020  b8 99 9b eb 6c 4d 20 01  0d a8 80 06 00 01 00 00   ····lM · ········
0030  00 00 00 00 66 66 dd 3c  00 35 00 27 a9 bf 8b 3e   ····ff·< ·5·'···>
0040  01 00 00 01 00 00 00 00  00 00 03 77 77 77 05 62   ········ ···www·b
0050  61 69 64 75 03 63 6f 6d  00 00 1c 00 01           aidu·com ·····
```

- **应用层：HTTP报文结构分析**



1. HTTP/1.1 200 OK\r\n

// 响应行

2. Response Version: HTTP/1.1

// 响应版本

3. Status Code: 200

// 返回的状态码

4. Response Phrase: OK

// 返回的信息

5. Accept-Ranges: bytes\r\n

// 接受的数据范围

6. Content-Encoding: gzip\r\n

// 内容编码

7. Date: Sun, 26 Sep 2021 06:15:46 GMT\r\n

// 日期

8. Last-Modified: Fri, 03 Jan 2020 08:33:49 GMT\r\n

// 最后修改时间

9. Server: Apache\r\n

// 服务器

指导教师评语：
日期：