同济大学

TONGJI UNIVERSITY

《计算机网络》

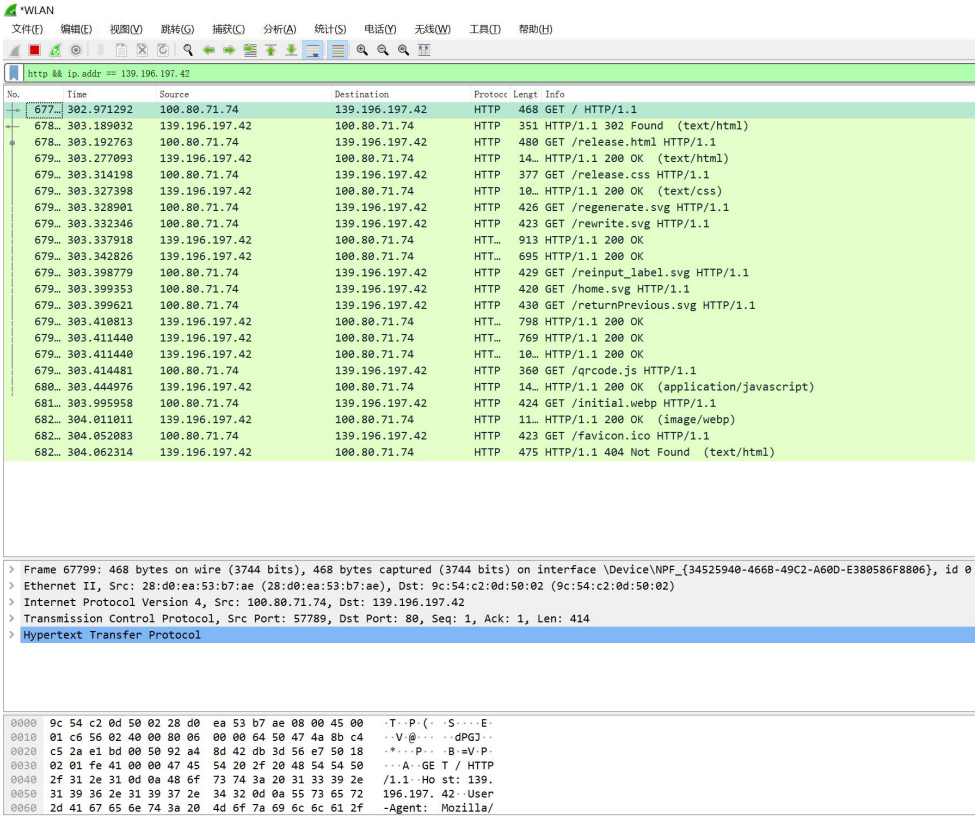
第三次实验报告

实验名称	WireShark 抓包实验
小组成员	
学院（系）	电子与信息工程学院
专　　业	计算机科学与技术
任课教师	陆有军
完成日期	2023 年 6 月 14 日

目录

一 实验要求

本次实验使用 WireShark 软件进行抓包分析，包括以太网 MAC 帧、无线局域网 MAC 帧、IP 协议、TCP 协议和 HTTP 协议。本实验抓取 ip 地址为 139.196.197.42。



二 实验内容

2.1 以太网 MAC 帧

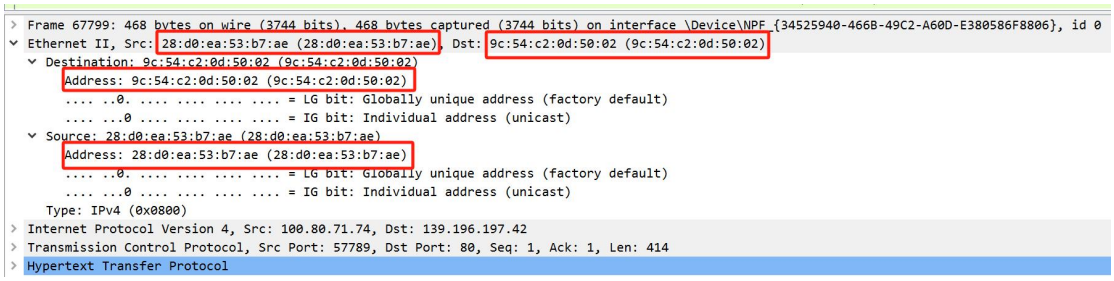
以太帧格式是一种以太网标准帧格式，也称为 DIX 帧格式。它由 Xerox、Intel 和 DEC 在 1982 年共同制定，后来被定义在 RFC894 中。以太帧格式的 Data 字段长度在 46-1500B 之间，因此以太网数据帧的长度在 64（6+6+2+46+4）到 1518（6+6+2+1500+4）字节之间。

2.1.1 以太网 MAC 帧格式分析

以太网 mac 帧格式包括以下字段

- 目标 MAC 地址：6 个字节
- 源 MAC 地址：6 个字节
- 类型：2 个字节，指示上层协议类型
- 数据：46-1500 个字节，包含上层协议的数据
- 帧校验序列（FCS）：4 个字节，用于检测帧是否损坏

2.1.2 以太网 MAC 帧抓包结果分析



捕获到的以太网 MAC 帧内容为：

9c	54	c2	0d	50	02	28	d0	ea	53	b7	ae

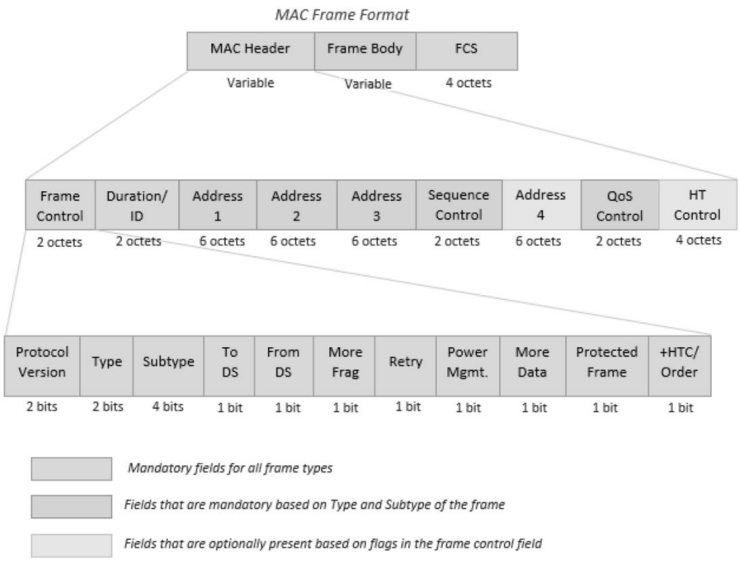
报文格式分析如下：

- 目标 MAC 地址：9c 54 c2 0d 50 02，占 6bit
- 源 MAC 地址：28 d0 ea 53 b7 ae，占 6bit
- 数据类型：08 00，即 IP 协议。占 2bit

2.2 无线局域网 MAC 帧

无线局域网（WLAN）是一种基于无线技术的局域网，它使用无线信号来传输数据。WLAN 使用一种称为“无线局域网协议”（Wireless Local Area Network Protocol，简称 WLAN Protocol）的协议来管理数据传输，其中包括无线局域网 MAC 帧。无线局域网 MAC 帧是 WLAN 协议中的一个关键部分，它包含了数据包的头部和数据部分，用于在无线信道上传输数据。802.11 协议中的 MAC 帧包括三种类型：管理帧、控制帧和数据帧。

2.2.1 无线局域网 mac 帧格式分析



IEEE 802.11 MAC 帧格式包括以下字段：

（一）帧控制域（Frame Control）

1. Protocol Version：长度 2bit，代表协议版本，默认为 0，其他值保留给未来使用。
2. Type：长度 2bit，表示帧类型：
   (1) 00：管理帧，负责监督网络，处理终端设备的加入/退出，设备的关联
   (2) 01：控制帧，负责区域的清空，信道的获取，载波的监听维护，数据的确认
   (3) 10：数据帧，负责在设备间传输数据
   (4) 11：保留不使用
3. Subtype：长度 2bit，表示帧的详细类型：
   (1) 1011:Request to send (RTS)
   (2) 1100:Clear to send (CTS)
   (3) 1101:Acknowledgment (ACK)
4. To DS / From DS：表示帧的源地址和目的地址
5. More Fragment：表示 MAC 帧是否分段；如果进行分段，除了最后一个片段，其他片段均会设置为 1 通过分段，MAC 层可以对数据进行分块传输，避免冲突。
6. Retry：如果该位设置为 1，表示为重传的帧
7. Power Management：表示完成当前的帧交换后是否进入省电模式，1 表示将进入省电模式。接入点需要管理连接，绝不会设置省电位。
8. More data：如果该位设置为 1，表示更多数据位用于缓冲分布式系统中收到的帧。接入点使用该位以方便处于省电模式的站点。它表示至少有一个帧是可用的，并针对所有连接的站。
9. Protected Frame：表示是否收到链路层安全协议的保护，加密标志，若为 1 表示数据内容加密，否则为 0
10. Order：表示帧或者帧片段是否按顺序传输，一般用于 PCF 模式下

To DS	From DS	Address1	Address2	Address3	Address4
0	0	Destination station	Source station	BSS ID	N/A
1	0	Receiving AP	Source station	Destination station	N/A
1	1	Receiving AP	Sending AP	Destination station	Source station
0	1	Destination station	Sending AP	Source station	N/A

（二） Duration/ID：长度 16bit。时长字段的值小于 32768 时，该值被解读为以 us 为单位的时长，被用于更新网络分配向量 NAV。如果在一个 PS-POLL 帧中两个高位比特被设置了，则低 14 位被解读为关联标识符 AID。

（三） 地址字段：共四个地址，每个地址字段为 48bit，详见图 2.2.2。

（四） 序列控制：长度 8bit。

1. Sequence Control 由 4 位的片段编号 (fragment number) 和 12 位的顺序编号 (sequence number) 字段组成
2. 顺序编号用于记录已传送的帧，如果是重传帧，顺便变好不会变化
3. 如果数据帧被分片，则所有分片的帧的顺序编号是一样，只是片段编号不同

（五）服务质量控制：存在于 QoS 数据帧中，长度为 16bit，可选。
（六） Framebody：MAC 帧的有效载荷，不同类型的 MAC 帧 (控制帧，数据帧，管理帧)， Frameboday 是不一样的
（七）帧检查序列（FCS）：标准 802.11 帧中的最后四个字节，使用 CRC 对帧进行校验。

2.2.2 无线局域网 mac 帧抓包结果分析

```
IEEE 802.11 Beacon frame, Flags: ........C
    Type/Subtype: Beacon frame (0x0008)
  Frame Control Field: 0x8000
      .... ..00 = Version: 0
      .... 00.. = Type: Management frame (0)
      1000 .... = Subtype: 8
    Flags: 0x00
        .... ..00 = DS status: Not leaving DS or network is operating in AD-HOC mode (To DS: 0 From DS: 0) (0x0)
        .... .0.. = More Fragments: This is the last fragment
        .... 0... = Retry: Frame is not being retransmitted
        ...0 .... = PWR MGT: STA will stay up
        ..0. .... = More Data: No data buffered
        .0.. .... = Protected flag: Data is not protected
        0... .... = +HTC/Order flag: Not strictly ordered
    .000 0000 0000 0000 = Duration: 0 microseconds
    Receiver address: Broadcast (ff:ff:ff:ff:ff:ff)
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
    Transmitter address: Cisco-Li_f7:1d:51 (00:16:b6:f7:1d:51)
    Source address: Cisco-Li_f7:1d:51 (00:16:b6:f7:1d:51)
    BSS Id: Cisco-Li_f7:1d:51 (00:16:b6:f7:1d:51)
    .... .... .... 0000 = Fragment number: 0
    1011 0010 0111 .... = Sequence number: 2855
    Frame check sequence: 0x39700f3d [unverified]
    [FCS Status: Unverified]
```

捕获到的无线局域网 MAC 帧内容为：

80 00 00 00 ff ff ff ff ff ff 00 16 b6 f7 1d 51 00 16 b6 f7 1d 51 70 b2

FCS:

3d 0f 70 39

报文格式分析如下：
1. 帧控制域 (Frame Control)：
   (1) Protocol Version: 0，占 2bit。Type: 0，占 2bit
   (2) Subtype: 8，占 4bit。To DS / From DS: 00，共占 2bit
   (3) More Fragment: 0，占 1bit。Retry: 0，占 1bit
   (4) Power Management: 0，占 1bit
   (5) More data: 0，占 1bit
   (6) Protected Frame: 0，占 1bit
   (7) Order: 0，占 1bit
2. Duration/ID：0ms，占 2 字节
3. 地址 1：ff ff ff ff ff ff，占 6 字节

4. 地址 2：00 16 b6 f7 1d 51，占 6 字节

5. 序列控制：

(1) 分片号：0，占 4bit

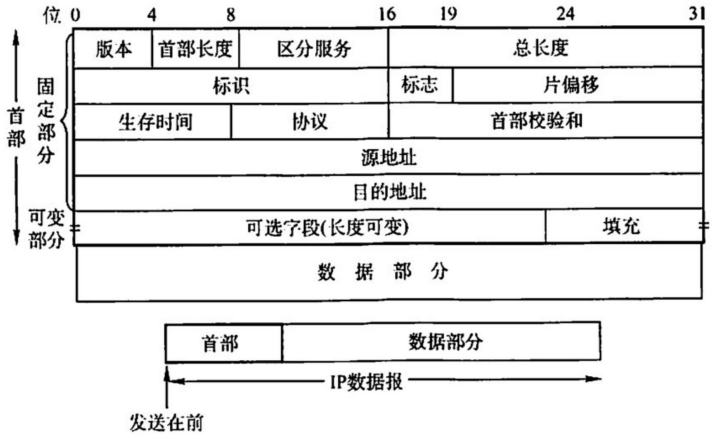
(2) 序列号：B27(十进制为 2855)，占 12bit

(3) FCS：0x39700f3d，占 4 字节

2.3 IP 协议

2.3.1 IPv4 报文格式分析

IP（Internet Protocol）是指因特网协议，是因特网的核心协议之一，用于在网络中传输数据包。IPv4 和 IPv6 是两个不同版本的 IP 协议。



IPv4 报文头格式及各字段功能：

1. 版本号（Version）：长度 4 bit 。标识目前采用的 IP 协议的版本号。一般的值 为 0100（IPv4），0110（IPv6）

版本号	版本
0	保留
1-3	未分配
4	Internet 协议版本 4（IPv4）
5	ST 数据报（Datagram）
6	IPv6
7	TP / IX
8	P Internet 协议（PIP）
9	使用更大地址的 TCP 和 UDP（TUBA）
10-14	未分配
15	保留

2. IP 报头长度（Header Length）：长度 4 bit 。这个字段的作用是为了描述 IP 报头的长度，因为在 IP 报头中有变长的可选部分。该部分占 4 个 bit，长度单 位

为 4 个字节，即本区域值 =IP 头部长度（单位为字节）/ 长度单位（4 个字 节）。当报头中无选项字段时，报头的总长度为 5，也就是 $5 \times 4 = 20$ 字节（此 时，报头长度的值为 0101）。这就是说 IP 数据报头部固定部分长度为 20 字节。当 IP 报头长度（最长）为 1111 时，头部的固定长度为 $15 \times 4 = 60$ 字节。但 报头长度必须是 32 位（四字节）的整数倍，如果不是，需要在选项字段的填充 （PAD）字段中补 0 凑齐。

3. 服务类型（Type of Service）：长度 8 bit 。8 位按位被如下定义：PPP DTRC0

(1) PPP：前 3 位，定义包的优先级，取值越大数据越重要

① 000 Routine。001 Priority

② 010 Immediate。011 Flash。100 Flash Override

③ 101 CRI / TIC / ECP。110 网间控制（Internetwork Control）

④ 111 网络控制（Network Control）

(2) DTRC0：后 5 位。D 时延：0：普通，1：延迟尽量小

① T 吞吐量：0：普通，1：流量尽量大

② R 可靠性：0：普通，1：可靠性尽量大

③ C 传输成本：0：普通，1：成本尽量小

④ 0 最后一位被保留，恒定为 0

1998 年 IETF 在 RFC2474 中把 IP 数据报中 ToS 字段改名为服务字段，同样为 8 位，前 6 位构成 DSCP（Different Services Code Point，区分服务码点），是 IP 优先级和服务类型字段的组合，定义了 0 63 共 64 个优先级。最后两位用来 携带显式拥塞通知信息。无论是哪种版本，该字段只有在使用区分服务时才起作 用，如果没有区分服务，则该字段值为 0。

4. IP 包总长度（Total Length）：长度 16 bit 。以字节为单位计算的 IP 包的长 度（包括头部和数据），所以 IP 包最大长度 65 535 字节。所以，数据包有效载 荷的大小 = IP 包总长度（Total Length）- IP 报头长度（Header Length）。

(1) 说明：在网络层下面的每一种数据链路层都有自己的格式，其中包括表示数据字段的最大长度，这称为最大传送单元（Maximum Transfer Unit，MTU）。当一个数据报封装成链路层的帧时，此数据报的总长度（包括报头和数据部分）一定不能超过下 面的数据链路层的 MTU 值

5. 标识符（Identifier）：长度 16 bit 。该字段和 Flags 和 Fragment Offest 字段联合使用，对较大的上层数据包进行分段（fragment）操作。路由器将一个包拆 分后，所有拆分开的小包被标记相同的值，以便目的端设备能够区分哪个包属于 被拆分开的包的一部分。

(1) IP 软件在存储器中维持一个计数器，每产生一个数据报，计数器就加 1，并将此值 赋给整个标识字段。但整个标识并不是序号，因为 IP 是无连接服务，数据报不存 在按序接受的问题。当数据报由于长度超过下面数据链路层的 MTU（最大传输单元） 值而必须分段的时候，这个标识符的值就被复制到所有的数据报分段的标识字段中。 相同的标识字段的值分段后的各数据报分段最后能正确地组装成原来的数据报。

6. 标记（Flags）：长度 3 bit 。

(1) 该字段第一位（最高位）不使用

(2) 第二位是 DF（Don’t Fragment）位，DF 位设为 1 时表明路由器不能对该上层数据 包分段。如果一个上层数据包无法在不分段的情况下进行转发，则路由器会丢弃该 上层数据包并返回一个错误信息

(3) 第三位是 MF（More Fragments）位，当路由器对一个上层数据包分段，则路由器会 在除了最后一个分段的 IP 包的报头中将 MF 位设为 1 7. 片偏移（Fragment Offset）：长度 13 bit，以 8 个八位组为单位。表示该 IP 包在该组分片包中位置，接收端靠此来组装还原 IP 包。

(4) 段偏移以 8 字节为偏移单位，即每个分段的长度一定是 8 字节（64 位）的整数倍。 第一个分段偏移值就是 0 0000 0000 0000，如果第一个分段一共是 64 字节，则 0 0000 0000 1001，相当于 10 进制数的 9，因为从第 9 个“8 字节”数据块开始的。 如果没有分段，则该字段值为 0。

7. 片偏移（Fragment Offset）：长度 13 bit，以 8 个八位组为单位。表示该 IP 包 在该组分片包中位置，接收端靠此来组装还原 IP 包。

(1) 段偏移以 8 字节为偏移单位，即每个分段的长度一定是 8 字节（64 位）的整数倍。 第一个分段偏移值就是 0 0000 0000 0000，如果第一个分段一共是 64 字节，则 0 0000 0000 1001，相当于 10 进制数的 9，因为从第 9 个“8 字节”数据块开始的。 如果没有分段，则该字段值为 0。

8. 生存时间（TTL）：长度 8 bit，设计之初是以秒（s）为单位的，但实际以跳数为 单位，建议的缺省值为 64 。当 IP 包进行传送时，先会对该字段赋予某个特定的 值。当 IP 包经过每一个沿途的路由器的时候，每个沿途的路由器会将 IP 包的

TTL 值减少 1 。如果 TTL 减少为 0，则该 IP 包会被丢弃。这个字段可以防止 由于路由环路而导致 IP 包在网络中不停被转发。

9. 协议（Protocol）：长度 8 bit。标识了上层所使用的协议。以下是比较常用的协议号：

协议号	协议名	缩写
1	互联网控制消息协议	ICMP
2	互联网组管理协议	IGMP
6	传输控制协议	TCP
17	用户数据报协议	UDP
41	IPv6 封装	ENCAP
88	内部网关路由协议	IGRP
89	开放式最短路径优先	OSPF
132	流控制传输协议	SCTP

10. 头部校验（Header Checksum）：长度 16 bit。用来做 IP 头部的正确性检测，但不包含数据部分。因为每个路由器要改变 TTL 的值，所以路由器会为每个通过的数据包重新计算这个值（RFC1141 讨论了一些简化计算的策略）。

11. 起源和目标地址（Source and Destination Addresses）：这两个地址都是 32 bit。标识了这个 IP 包的起源和目标地址。要注意除非使用 NAT，否则整个传输的过程中，这两个地址不会改变。

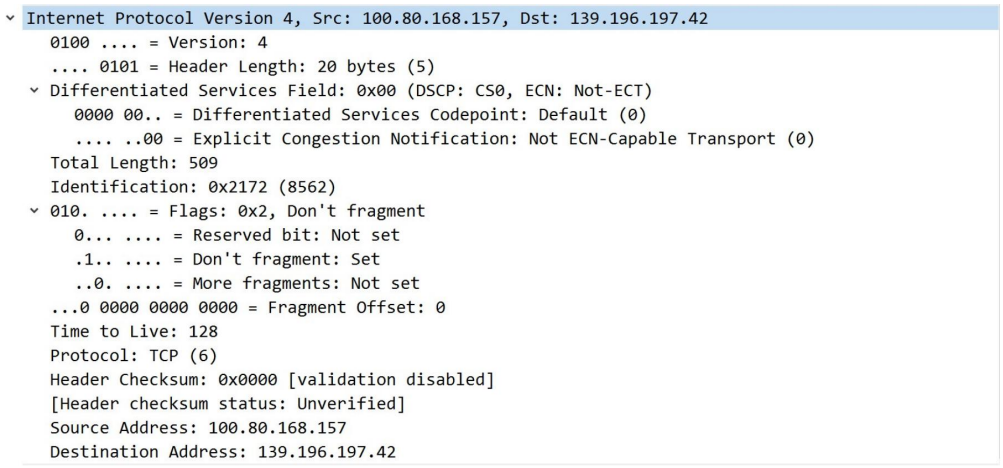
12. 可选项（Options）：这是一个可变长的字段。该字段属于可选项，主要用于测试，由起源设备根据需要改写。可选项目包含以下内容：

（1）松散源路由（Loose source routing）：给出一连串路由器接口的 IP 地址。IP 包必须沿着这些 IP 地址传送，但是允许在相继的两个 IP 地址之间跳过多个路由器。

（2）严格源路由（Strict source routing）：给出一连串路由器接口的 IP 地址。IP 包必须沿着这些 IP 地址传送，如果下一跳不在 IP 地址表中则表示发生错误。路由记录（Record route）：当 IP 包离开每个路由器的时候记录路由器的出站接口的 IP 地址。

（3）时间戳(Timestamps)：当 IP 包离开每个路由器的时候记录时间。填充(Padding)：因为 IP 报头长度(Header Length)部分的单位为 32 bit，所以 IP 报头的长度必须为 32 bit 的整数倍。因此，在可选项后面，IP 协议会填充若干个 0，以达到 32 bit 的整数倍。

2.3.2 IPv4 抓包结果分析



捕获到的 IPv4 报文为：

```
45 00 01 fd 21 72 40 00 80 06 00 00 64 50 a8 9d 8b c4 c5 2a
```

报文信息分析如下：

- 版本为 4，占 4bit
- 首部长度为 4x5= 20 字节，占 4bit
- 差分服务代码点为 CS0，占 6bit
- 显式拥塞通告为 Not-ECT，占 2bit
- 总长度为 01 fd (十进制为 509)字节，占 16bit
- 标识符为 21 72，占 16bit
- 标志为 Don’t Fragment，不允许分段，占 3bit
- 分片偏移为 0，占 13bit
- 存活时间为 128 跳，占 8bit
- 协议号为 6，协议为 TCP，占 8bit
- 首部检验和为 00 00，占 16bit
- 源地址为 64 50 a8 9d，即 100.80.168.157，占 32bit
- 目的地址为 8b c4 c5 2a，即 139.196.197.42，占 32bit

2.4 TCP 协议

TCP（Transmission Control Protocol）是一种面向连接的传输层协议，它提供可靠的、有序的、面向字节流的数据传输服务。TCP 协议通过三次握手建立连接，使用滑动窗口和拥塞控制等机制来保证数据传输的可靠性和效率。

2.4.1 TCP 报文格式分析

TCP 报文头格式及各字段功能：

1. 源端口和目的端口字段：

- TCP 源端口（Source Port）：源计算机上的应用程序的端口号，占 16 位
- TCP 目的端口（Destination Port）：目标计算机的应用程序端口号，占 16 位

2. TCP 序列号（Sequence Number）：占 32 位。它表示本报文段所发送数据的第一个字节的编号。在 TCP 连接中，所传送的字节流的每一个字节都会按顺序编号。当 SYN 标记不为 1 时，这是当前数据分段第一个字母的序列号；如果 SYN 的值是 1 时，这个字段的值就是初始序列值（ISN），用于对序列号进行同步。这时，第一个字节的序列号比这个字段的值大 1，也就是 ISN 加 1

3. TCP 确认号（Acknowledgment Number，ACK Number）：占 32 位。它表示接收方期望收到发送方下一个报文段的第一个字节数据的编号。其值是接收计算机即将接收到的下一个序列号，也就是下一个接收到的字节的序列号加 1

4. TCP 首部长度（Header Length）：数据偏移字段，数据偏移是指数据段中的“数据”部分起始处距离 TCP 数据段起始处的字节偏移量，占 4 位

5. 保留字段：占 4 位。为 TCP 将来的发展预留空间，目前必须全部为 0

6. 标志位字段：

- CWR（Congestion Window Reduce）：拥塞窗口减少标志，用来表明它接收到了设置 ECE 标志的 TCP 包。并且，发送方收到消息之后，通过减小发送窗口的大小来降低发送速率
- ECE（ECN Echo）：用来在 TCP 三次握手时表明一个 TCP 端是具备 ECN 功能的。在数据传输过程中，它也用来表明接收到的 TCP 包的 IP 头部的 ECN 被设置为 11，即网络线路拥堵

- 如果 SYN 标志被设置（1），说明 TCP 对等体有 ECN 能力

- 如果 SYN 标志清零（0），说明在正常传输过程中收到了 IP 头中设置有拥塞经

验标志（ECN=11）的数据包

• URG（Urgent）：表示本报文段中发送的数据是否包含紧急数据。URG=1 时表示有紧急数据。当 URG=1 时，后面的紧急指针字段才有效

• ACK：表示前面的确认号字段是否有效。ACK=1 时表示有效。只有当 ACK=1 时，前面的确认号字段才有效。TCP 规定，连接建立后，ACK 必须为 1

• PSH（Push）：告诉对方收到该报文段后是否立即把数据推送给上层。如果值为 1，表示应当立即把数据提交给上层，而不是缓存起来

• RST：表示是否重置连接。如果 RST=1，说明 TCP 连接出现了严重错误（如主机崩溃），必须释放连接，然后再重新建立连接

• SYN：在建立连接时使用，用来同步序号。当 SYN=1，ACK=0 时，表示这是一个请求建立连接的报文段；当 SYN=1，ACK=1 时，表示对方同意建立连接。SYN=1 时，说明
这是一个请求建立连接或同意建立连接的报文。只有在前两次握手中 SYN 才为 1

• FIN：标记数据是否发送完毕。如果 FIN=1，表示数据已经发送完成，可以释放连接

7. 窗口大小字段：窗口大小（Window Size）：占 16 位。它表示从 Ack Number 开始还可以接收多少字节的数据量，也表示当前接收端的接收窗口还有多少剩余空间。该字段可以用于 TCP 的流量控制

8. TCP 校验和字段：校验位（TCP Checksum）：占 16 位。它用于确认传输的数据是否有损坏。发送端基于数据内容校验生成一个数值，接收端根据接收的数据校验生成一个值。两个值必须相同，才能证明数据是有效的。如果两个值不同，则丢掉这个数据包。Checksum 是根据伪头 + TCP 头 + TCP 数据三部分进行计算的

9. 紧急指针字段：紧急指针（Urgent Pointer）：仅当前面的 URG 控制位为 1 时才有意义。它指出本数据段中为紧急数据的字节数，占 16 位。当所有紧急数据处理完后，TCP 就会告诉应用程序恢复到正常操作。即使当前窗口大小为 0，也是可以发送紧急数据的，因为紧急数据无须缓存

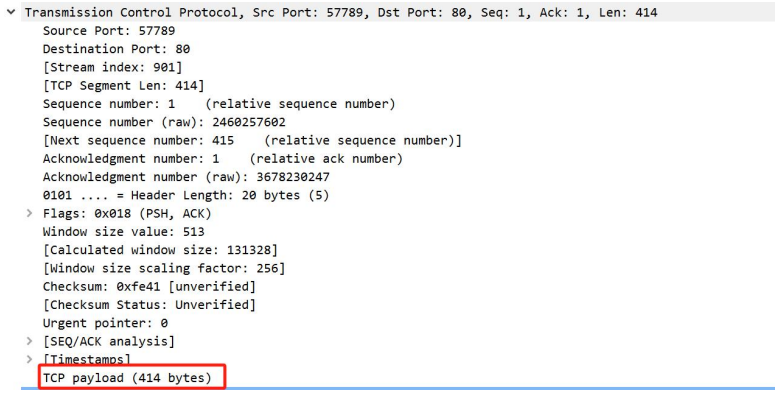
10. 可选项字段：（Option）：长度必须是 32bits 的整数倍，最大长度为 40 字节。TCP Options 字段格式为

Kind(1 字节)	Length(1 字节)	Info(n 字节)

常用的 TCP Options 字段为：

Kind(Type)	Length	Name	描述 & 用途
0	1	EOL	选项列表结束
1	1	NOP	无操作（用于补位填充）
2	4	MSS	最大 Segment 长度
3	3	WSOPT	窗口扩大系数（Window Scaling Factor）
4	2	SACK-Premitted	表明支持 SACK
5	可变	SACK	SACK Block（收到乱序数据）
8	10	TSPOT	Timestamps
19	18	TCP-MD5	MD5 认证
28	4	UTO	User Timeout（超过一定闲置时间后拆除连接）
29	可变	TCP-AO	认证（可选用各种算法）
253/254	可变	Experimental	保留，用于科研实验

2.4.2 TCP 抓包结果分析



捕获到的 TCP 报文为：

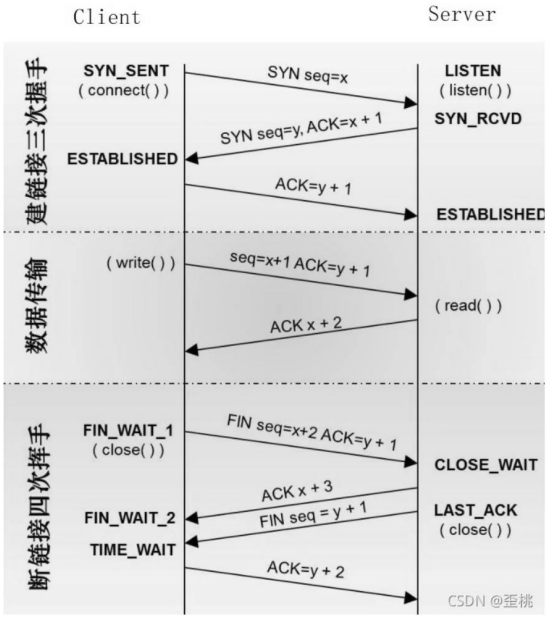
```
c7 71 00 50 fb d6 05 1f a4 1b c0 d3 50 18 02 01 5f cc 00 00
```

- 源端口为 c7 71（十进制 51057），占 16bit
- 目的端口为 00 50（十进制 80），占 16bit
- 序列号为 fb d6 05 1f（十进制 4225107231），占 32bit•确认号为 a4 1b c0 d3（十进制 2753282259），占 32bit•数据偏移 20 字节，占 4bit
- 保留占 3bit
- 标志占 9bit
    - NS = 0，占 1bit
    - CWR = 0，占 1bit
    - ECE = 0，占 1bit
    - URG = 0，占 1bit
    - ACK = 1，占 1bit

- PSH = 1，占 1bit
- RST = 0，占 1bit
- SYN = 0，占 1bit
- FIN = 0，占 1bit

• 窗口大小为 02 01（十进制 513），占 16bit
• 校验和为 0x5fcc，占 16bit
• 紧急指针为 0x0000，占 16bit
• 可选字段占 0bit

2.4.3 TCP 三次握手与四次挥手



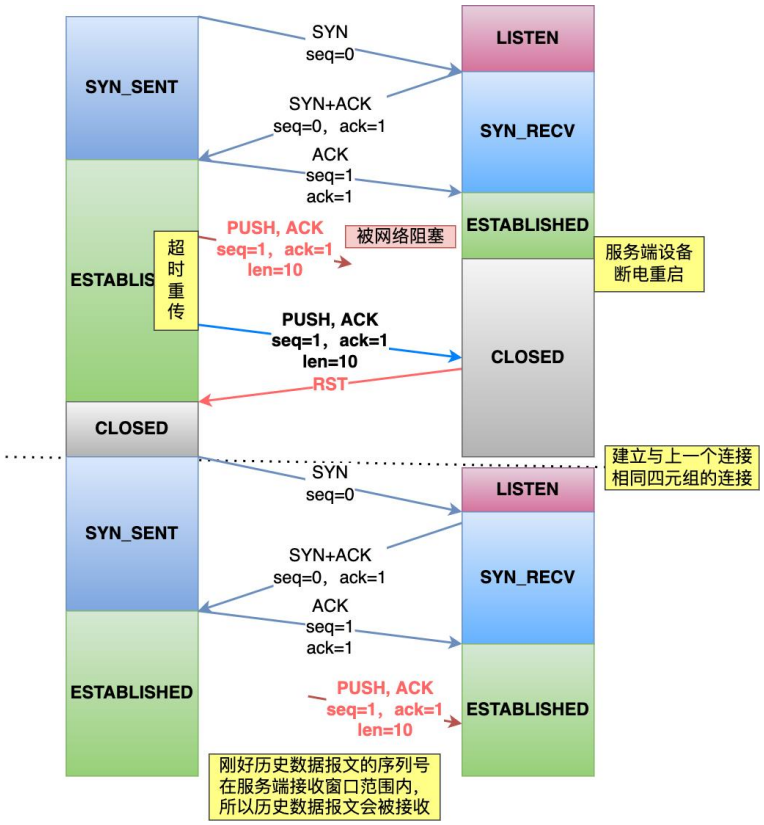
三次握手流程

1. 首先客户端向服务器端发送一段 TCP 报文。
   • 标记位为 SYN，表示“请求建立新连接”
   • 序号为 Seq=x(x 一般为 0)
   • 随后客户端进入 SYN-SENT 阶段
2. 服务器端接收到来自客户端的 TCP 报文之后，结束 LISTEN 阶段。并返回一段 TCP 报文
   • 标志位为 SYN 和 ACK，表示“确认客户端的报文 Seq 序号有效，服务器能正常接收客户端发送的数据，并同意创建新连接”
   • 序号为 Seq=y(y 一般为 0)
   • 随后客户端进入 SYN-SENT 阶段
   • 确认号为 Ack=x+1，表示收到客户端的序号 Seq 并将其值加 1 作为自己确认号 Ack 的值；随后服务器端进入 SYN-RCVD 阶段

3. 客户端接收到来自服务器端的确认收到数据的 TCP 报文之后，明确了从客户端到服务器的数据传输是正常的，结束 SYN-SENT 阶段。并返回最后一段 TCP 报文。

- 标志位为 ACK，表示“确认收到服务器端同意连接的信号”
- 序号为 Seq=x+1，表示收到服务器端的确认号 Ack，并将其值作为自己的序号值
- 确认号为 Ack=y+1，表示收到服务器端序号 Seq，并将其值加 1 作为自己的确认号 Ack 的值
- 随后客户端进入 ESTABLISHED 阶段。服务器收到来自客户端的“确认收到服务器数据”的 TCP 报文之后，明确了从服务器到客户端的数据传输是正常的。结束 SYN-SENT 阶段，进入 ESTABLISHED 阶段

在客户端与服务器端传输的 TCP 报文中，双方的确认号 Ack 和序号 Seq 的值，都是在彼此 Ack 和 Seq 值的基础上进行计算的，这样做保证了 TCP 报文传输的连贯性。一旦出现某一方发出的 TCP 报文丢失，便无法继续”握手”，以此确保了”三次握手”的顺利完成。



四次挥手流程

1. 首先客户端想要释放连接，向服务器端发送一段 TCP 报文。

- 标记位为 FIN，ACK，表示“请求释放连接”
- 序号为 Seq=U
- 随后客户端进入 FIN-WAIT-1 阶段，即半关闭阶段。并且停止在客户端到服务器端方向上发送数据，但是客户端仍然能接收从服务器端传输过来的数据

2. 服务器端接收到从客户端发出的 TCP 报文之后，确认了客户端想要释放连接，随后服务器端结束 ESTABLISHED 阶段，进入 CLOSE-WAIT 阶段（半关闭状态）并返回一段 TCP 报文。

- 标记位为 ACK，表示“接收到客户端发送的释放连接的请求”
- 序号为 Seq=V
- 确认号为 Ack=U+1，表示是在收到客户端报文的基础上，将其序号 Seq 值加 1 作为本段报文确认号 Ack 的值
- 随后服务器端开始准备释放服务器端到客户端方向上的连接。客户端收到从服务器端发出的 TCP 报文之后，确认了服务器收到了客户端发出的释放连接请求，随后客户端结束 FIN-WAIT-1 阶段，进入 FIN-WAIT-2 阶段

前两次挥手既让服务器端知道了客户端想要释放连接，也让客户端知道了服务器端了解了自己想要释放连接的请求。

3. 服务器端自从发出 ACK 确认报文之后，经过 CLOSED-WAIT 阶段，做好了释放服务器端到客户端方向上的连接准备，再次向客户端发出一段 TCP 报文

- 标记位为 FIN，ACK，表示“已经准备好释放连接了”。
- 序号为 Seq=W
- 确认号为 Ack=U+1；表示是在收到客户端报文的基础上，将其序号 Seq 值加 1 作为本段报文确认号 Ack 的值
- 随后服务器端结束 CLOSE-WAIT 阶段，进入 LAST-ACK 阶段。并且停止在服务器端到客户端的方向上发送数据，但是服务器端仍然能够接收从客户端传输过来的数据

4. 客户端收到从服务器端发出的 TCP 报文，确认了服务器端已做好释放连接的准备，结束 FIN-WAIT-2 阶段，进入 TIME-WAIT 阶段，并向服务器端发送一段报文。

- 标记位为 ACK，表示“接收到服务器准备好释放连接的信号”
- 序号为 Seq=U+1；表示是在收到了服务器端报文的基础上，将其确认号 Ack 值作为本段报文序号的值
- 确认号为 Ack=W+1；表示是在收到了服务器端报文的基础上，将其序号 Seq 值作为本段报文确认号的值

2.4.3 WireShark 抓包查看

TCP 协议的三次握手

No.	Time	Source	Destination	Protocc	Lengt	Info
1051	3.608531	100.80.71.74	139.196.197.42	TCP	66	50211 → 80 [SYN] Seq=0 Win=…
1056	3.620943	139.196.197.42	100.80.71.74	TCP	66	80 → 50211 [SYN, ACK] Seq=0…
1057	3.621030	100.80.71.74	139.196.197.42	TCP	54	50211 → 80 [ACK] Seq=1 Ack=…

TCP 协议的四次挥手

5227	69.348623	139.196.197.42	100.80.71.74	TCP	56	80 → 50241 [FIN, ACK] Seq=4…
5228	69.348696	100.80.71.74	139.196.197.42	TCP	54	50241 → 80 [ACK] Seq=394 Ac…
5229	69.348839	100.80.71.74	139.196.197.42	TCP	54	50241 → 80 [FIN, ACK] Seq=3…
5230	69.372008	139.196.197.42	100.80.71.74	TCP	56	80 → 50241 [ACK] Seq=423 Ac…

2.5 HTTP 协议

HTTP(HyperText Transfer Protocol，超文本传输协议）是一种用于分布式、协作 式和超媒体信息系统的应用层协议。HTTP 是万维网的数据通信的基础，是客户端和服 务端之间请求和应答的标准。通过使用网页浏览器或者其它的工具，客户端可以发起一 个 HTTP 请求到服务器上指定端口，默认端口为 80。

2.5.1 HTTP 请求报文

请求行

请求行由请求方式、请求 url 和请求 HTTP 协议版本字段 3 个字段组成，它们用空格分隔。例如：GET /login.html HTTP/1.1

请求方式：

HTTP/1.1 协议的请求方式主要有以下九种，其中 GET 和 POST 是最 常用的请求方式。

URL：

统一资源定位符，是一种资源位置的抽象唯一识别方法。统一资源定位符的 完整格式如下：

[协议类型]://[服务器地址]:[端口号]/[资源层级 UNIX 文件路径][文件名]?[查询]#[片段 ID]

HTTP **协议版本**:

协议版本的格式为：HTTP/主版本号. 次版本号，常用的有 HTTP/1.0 和 HTTP/1.1。

请求头:

请求头为请求报文添加了一些附加信息，格式为“名: 值”，名和值之间使用冒号分隔。请求头部通知服务器关于客户端请求的信息。请求头信息有：

信息名称	内容
Host	访问的 ip 地址或域名
Referer	当前文档的 URL，告诉服务器请求来源
User-Agent	客户端相关信息，如操作系统、浏览器等信息，可以用来解决浏览器兼容问题
Accept	客户端接收信息类型
Accept-Charset	客户端接受的字符集
Accept-Encoding	客户端可接受的内容编码
Accept-Lanquage	客户端可接受的语言
Connection	连接状态
Cookie	携带的 cookie 信息
Content-Type	请求体格式，如 application/x-www-form-urlencoded
Content-Length	请求体长度
Cache-Control	控制缓存

请求空行:

请求头部的最后会有一个空行，表示请求头部结束，用来分隔请求头和请求体。

请求体:

封装 POST、PUT、DELETE 等请求消息的请求参数，GRT、HEAD 等请求方法没有请求体。

2.5.2 HTTP 响应报文

服务器端发送给客户端的数据，它由响应行，响应头，响应空行，响应体组成。

响应行 :

状态行由 3 部分组成，分别为：协议版本、响应状态码、状态码描述。 响应行的组成格式为：协议/版本响应状态码状态码描述 (通常是 OK，还可能会看 到 Not Modified，Found 等)

响应状态码:

服务器告诉客户端浏览器本次请求和响应的一个状态，常用的状态码 分为以下几类：

格式	作用
1XX	服务器接收客户端消息，但没有接受完成
2XX	请求处理成功
3XX	重定向
4XX	客户端错误
5XX	服务器端错误

响应头:

常见的响应头信息有以下内容：

信息名称	内容
Server	HTTP 服务器的软件信息
Date	响应报文的时间
Set-Cookie	设置 Cookie
Content-Type	响应的类型和字符集
Content-Length	响应体长度
Content-Encoding	响应体长度内容编码
Content-Disposition	客户端打开响应体数据的方式
Expires	指定缓存过期时间
Last-Modified	资源最后修改时间
Location	指明重定向的位置，会在状态码为 302 时出现
Connection	连接状态
Cache-Control	控制缓存

响应空行:

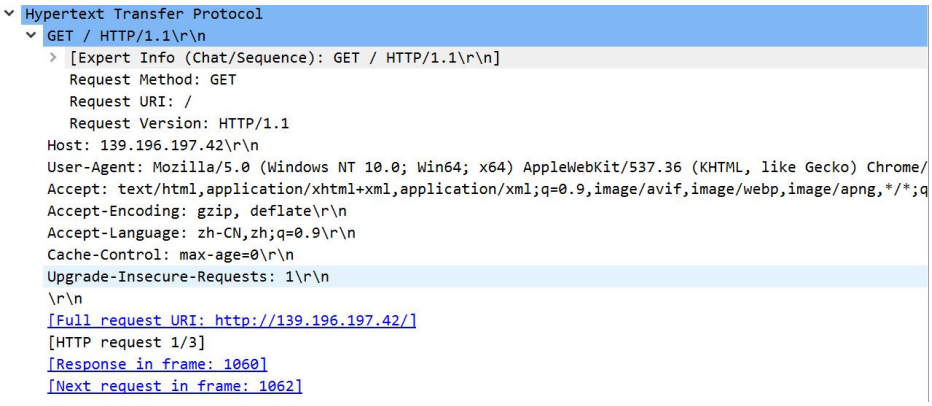
响应头部的最后会有一个空行，表示响应头部结束，用来分隔响应头和响应体。

响应体:

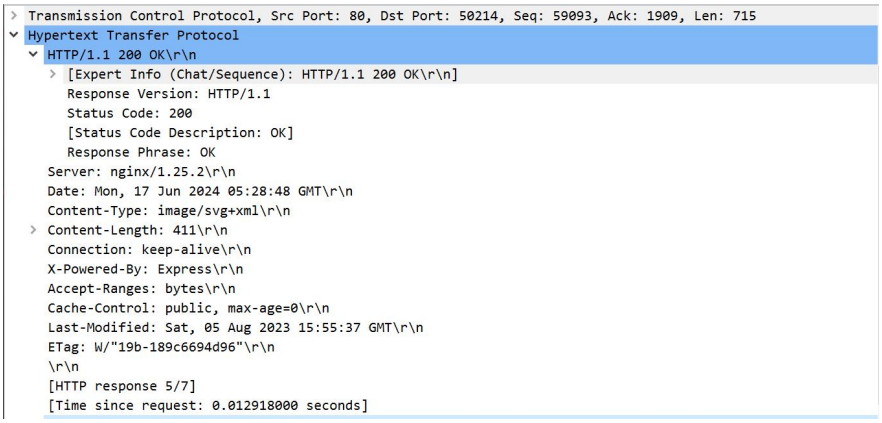
用于存放需要返回给客户端的数据信息，通常为 HTML/CSS/JS/JSON 等，有些类型的响应没有响应体。

2.5.3 HTTP 抓包结果分析

HTTP 请求报文



HTTP 响应报文



请求报文:

客户端请求的请求行、请求头分别对应下方代码的第 1 行和第 2 - 8 行。其中， 请求行由请求方法（GET）、请求路径（/）和 HTTP 版本（HTTP/1.1）组成。对于 GET 方法没有请求体。

```
1 GET / HTTP/1.1

2 Host: 139.196.197.42

3 Connection: keep-alive

4 Upgrade-Insecure-Requests: 1

5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like ,→ Gecko)

Chrome/113.0.0.0 Safari/537.36 Edg/113.0.1774.57
```

```
6  Accept:  text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;  」  ,→
q=0.8,application/signed-exchange;v=b3;q=0.7
7 Accept-Encoding: gzip, deflate
8 Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

响应报文:

服务器响应的响应行、响应头、响应空行、响应体分别在下方代码的第 1 行、第 2 - 10 行、第 11 行、第 12-27 行。其中，响应行由协议版本（HTTP/1.1）、状态码 （200）和状态码描述（OK）组成。

```
1 HTTP/1.1 200 OK
2 Date: Sat, 03 Jun 2023 15:43:47 GMT
3 Server: Apache/2.4.37 (Alibaba Cloud Linux)
4 Last-Modified: Tue, 07 Mar 2023 15:09:15 GMT
5 ETag: "2a7-5f650cb816208"
6 Accept-Ranges: bytes
 7 Content-Length: 679
8 Keep-Alive: timeout=5, max=100
9 Connection: Keep-Alive
10 Content-Type: text/html; charset=UTF-8
11
12
13 <html xmlns="http://www.w3.org/1999/xhtml">
14 <head>
15     <meta content="text/html;charset=utf-8" http-equiv="Content-Type"/>
16     <title> 登录</title>
17     <link rel="stylesheet" type="text/css" href="./index.css"> 1
8 </head>
19 <body style="overflow: hidden; margin: 0; padding: 0;">
20     <div class="login">
21         <input class="input" placeholder="学号">
22         <input class="input" placeholder="密码">
23        <div class="button" onclick="window.location.href='http://www.maiji.com/'"> ,→ 登录</div>
24        <div class="button" onclick="window.location.href='http://www.maiji.com/'"> ,→ 注册</div>
25     </div>
```

```
26 </body>
27 </html>
```

三 心得体会

通过抓包实验，我深入理解了各种网络协议和工具的工作原理和应用，提升了网络管理和调试的能力。我们掌握了使用抓包工具和 Wireshark 进行网络分析的方法，增强了诊断和解决网络问题的能力。这些知识和技能不仅帮助我们更好地管理局域网和无线网络，还加深了对 Web 应用和互联网架构的理解，主要有以下几个方面：

1. 以太网 MAC 帧分析

以太网 MAC 帧是网络通信的基本单元，包含了数据包的头部和数据部分。通过解析以太网 MAC 帧的结构和字段意义，我们能够更好地理解以太网协议，并进行局域网的管理。

2. 无线局域网 MAC 帧特点

无线局域网 MAC 帧与有线局域网 MAC 帧在帧控制字段、地址字段、序列控制字段等方面存在一些差异。分析这些不同点，有助于我们更好地理解和管理无线网络。

3. IP 协议分析

IP 协议是互联网通信的核心协议，负责将数据包从源节点传输到目的节点。通过研究 IP 协议的运行机制和特点，我们可以深入了解互联网的架构和功能。

4. TCP 协议研究

TCP 协议是一种可靠的传输协议，能够在数据传输过程中检测和纠正错误。通过分析 TCP 协议的拥塞控制、流量控制和错误恢复机制，我们可以更深入地理解其工作原理和应用场景。

5. HTTP 协议解析

HTTP 协议是一种基于请求-响应模型的应用层协议，广泛用于 Web 数据传输。通过分析 HTTP 协议的请求和响应消息，我们可以深入理解 Web 应用的工作原理和组成。