**《计算机网络》实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验名称** | 协议数据的捕获和解析 | | **学 院** | 计算机学院 | **指导教师** | **高占春** |
| **班 级** | **班内序号** | **学 号** | | **学生姓名** | **成绩** | |
| **2023211303** | **31** | **2023212872** | | **计子毅** |  | |
|  |  |  | |  |  | |
|  |  |  | |  |  | |
| **实**  **验**  **内**  **容** | 在本实验中，学生通过使用 Wireshark 软件来捕获网络上实际传输的数据，可以加深对于上述协议的要点的理解，例如 IP 分段、TCP 的连接管理、ICMP 的功能、ARP 的功能、DHCP 的功能等；同时对于教材中没有包含或阐述不够详细的内容，例如 IP 包头校验和的计算、TCP 的 MSS 概念、ICMP 的 消息格式、ARP 的消息格式、DHCP 的消息格式及操作过程等，可以通过分析数据格式和协议流程进行自学和了解。  在本实验中，熟悉 Wireshark 软件并进行消息捕获的工作比较简单，实验的重点和难点在于协议的分 析工作，例如 DHCP 消息的含义和操作过程，学生需要对所捕获到的 DHCP 消息的格式和序列进行分析， 绘制出 DHCP 消息序列图，从而达到理解协议工作原理的目的。 | | | | | |
| **学生**  **实验**  **报告**  （附页） | 实验报告将在表格之后展示 | | | | | |
| **实**  **验**  **成**  **绩**  **评**  **定** | **评语**:  **成绩**:  助教签名：  年 月 日 | | | | | |

注：评语要体现每个学生的工作情况，可以加页。

实验二：协议数据的捕获和解析

1. 实验内容和实验目的描述
   1. 实验任务

* 使用Wireshark软件捕获在使用ping命令时产生的ICMP消息
* 分析网络层IP包头格式，理解各字段的作用，对于分段和校验和进行验证
* 使用Wireshark软件捕获ARP消息，分析其消息序列，理解DHCP的功能和操作原理
* 使用Wireshark捕获DHCP消息，分析其消息序列，理解DHCP的功能和操作原理
* 使用Wireshark捕获TCP消息，分析TCP报文段头格式，理解连接建立和释放的原理，差错控制原理、序号和窗口管理的原理
  1. 实验目的

以深入理解分层体系结构，理解和掌握 TCP/IP 协议栈的代表协议——IP、TCP、 UDP、ICMP、ARP 和 DHCP 协议的要点。

* 1. 实验环境

Windows操作系统

Wireshark软件

* 1. 实验步骤
     1. 准备工作

1. 下载Wireshark软件并了解其功能和使用方法
2. 确保计算机已经连接到网络
3. 启动Wireshark，设置捕获接口为本机网卡(本实验中使用WLAN接口进行实验)，选中混杂模式捕获选项，设置合适的捕获过滤器:

· 对于ping命令，设置捕获过滤器为 icmp

· 对于 DHCP 消息，设置捕获过滤器为 udp port 67

· 对于 ARP 消息，设置捕获过滤器为 arp

· 对于通过网页浏览应用来捕获 TCP 消息，设置捕获过滤器为 tcp port 80 4.开始捕获。

* + 1. 数据捕获
       1. 捕获ICMP协议数据

1. 运行 ping 命令（例如：c> ping 192.168.0.1），远程主机地址可以是本机地址、网关路由器地址，也可以是域名（如 www.bupt.edu.cn）。将捕获到的数据保存为文件。
2. 使用 Windows 中 ping 命令的-l 选项（例如：c>ping -l 8000 192.168.0.1），生成大于 8000 字 节的 IP 包并发送，捕获后分析其分段传输的包结构。
   * + 1. 捕获DHCP协议数据
3. 使用 ipconfig 命令释放计算机的 IP 地址（c>ipconfig -release）
4. 使用 ipconfig 命令重新申请 IP 地址（c>ipconfig -renew）
   * + 1. 捕获ARP协议数据

采用与 7.2.2 相同的方法释放 IP 地址并重新申请，在 wireshark 窗口中可以捕获到 ARP 请求 和响应消息，保存为文件。

* + - 1. 捕获TCP协议数据

打开浏览器，输入一个页面内容较简单的网页的 URL，如 www.baidu.com；网页全部显示后关闭浏览器。

* + 1. 协议分析

1. IP 包头分析：对于采用 ping 命令-l 选项捕获的 ICMP 消息，对承载ICMP 消息的 IP 包进行分析， 记录包头各字段的值，对照讲义和教材分析各字段的功能，并对于分段进行验证；
2. ICMP 消息分析：记录并分析 ICMP 消息中分析各字段的功能；
3. DHCP 消息分析：针对一次地址分配过程（Transaction ID 相同的 4 个消息），分析其通信过程，画出地址分配的消息序列图，并记录采用 DHCP 协议配置的各个参数。
4. ARP 消息分析：对照讲义理解 ARP 的操作过程，记录并分析消息中各字段的功能。
5. TCP 报头及消息分析：针对 TCP 连接建立、连接释放、数据和应答报文段，对照讲义和教材分 析各字段的功能；针对一次完整的 TCP 通信过程，画出消息序列图，应包含连接建立、数据传 送和连接释放阶段。
6. IP 协议分析

2.1 捕获IP数据包

第一步：运行Wireshark软件，在首页选择进行连接的网络，选择WLAN

第二步：在应用显示过滤器中输入ICMP，过滤出ICMP报文

第三步：打开命令提示符窗口，输入ping -l 8000 10.3.9.6，回车执行

屏幕上写着字

AI 生成的内容可能不正确。

捕获到的报文如下:

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。

图形用户界面, 应用程序

AI 生成的内容可能不正确。

2.2 IP 包头分析

2.2.1 包头各字段

表格

AI 生成的内容可能不正确。

|  |  |  |
| --- | --- | --- |
| 字段 | 报文(16进制) | 内容 |
| 包头长度 | 45 | 包头长20字节 |
| 服务类型 | 00 | 正常时延，正常吞吐量，正常可靠性 |
| 总长度 | 05dc | 数据分组长1500字节 |
| 标识 | 99ec | 标识为39404 |
| 标志 | 01 | MF=1，DF=0 允许分片，此片后还有一片 |
| 片偏移 | 0 | 偏移量为0 |
| 生存周期 | 3b | 每跳生存周期为59秒 |
| 协议 | 01 | 携带的数据来自ICMP协议 |
| 头部校验和 | De4c | IP头部校验和为de4c |
| 源地址 | 0a1dc4c2 | 源地址为10.29.196.194 |
| 目的地址 | 0a030906 | 目的地址为10.3.9.6 |

2.2.2 包头校验和

计算方法:

1. ip包头(共20个字节)按照每16个bit作为一个值依次进行相加

2. 将计算结果的进位加到低16位上

3. 将结果取反

验证:

包头内容:

45 00 05 dc 99 ec 20 00

3b 01 00 00 0a 03 09 06

0a 1d c4 c2

(2)

2.2.3 IP包分段

表格

AI 生成的内容可能不正确。

如图所示，整个过程由4对应答组成，1-6号帧为本主机向目标发送的请求帧，我们可以看到每个包的大小如下：

文本

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。

由上图可知：去掉IP头（20 bytes）后，前五个分片每个包含1480字节，最后一片包含608字节，一共8008字节，即为8000字节的数据 + 8字节的ICMP头。同时可知链路的MTU为1500。

第一片（1号）：

包长度为1500字节；

DF=0，表示允许分片；

段标识=0xe5ba=58810；

MF=1，表示当前不是最后一片；

偏移量=0，表示当前为第一片

第二片（2号）：

包长度为1500字节；

DF=0，表示允许分片；

段标识=0xe5ba=58810；

MF=1，表示当前不是最后一片；

偏移量=1480

后几片以此类推，其中同一个包的所有分片都有相同的段标识字段，从而可以确认哪些分片属于同一个包。

1. ICMP协议分析
   1. ICMP数据包的捕获

ICMP数据包的捕获过程在IP包分析部分已经给出:

第一步：运行Wireshark软件，在首页选择进行连接的网络，选择WLAN

第二步：在应用显示过滤器中输入ICMP，过滤出ICMP报文

第三步：打开命令提示符窗口，输入ping -l 8000 10.3.9.6，回车执行

* 1. ICPM功能

ICMP（Internet Control Message Protocol，因特网控制报文协议）是 TCP/IP 协议族中的一个重要子协议，通常被 IP 层或更高层协议（TCP 或 UDP）使用，属于网络层协议。它主要用于在 IP 主机和路由器之间传递控制消息，用于报告主机是否可达、路由是否可用等。

ICMP 报文被封装在 IP 数据包内部，作为 IP 数据包的数据部分通过互联网传递。当数据包处理过程出现差错时，ICMP 向数据包的源端设备报告这个差错，它既不会纠正这个差错，也不会通知中间的网络设备。

ICMP主要用来检测网络通信故障和实现链路追踪,最典型的应用是ping和tracerooute。Ping通过发送回送请求报文和回送回答报文来检测源主机到目的主机的来南路是否有问题，目的地是否可达，以及通信的延迟情况。

* 1. ICMP的包格式

图示, 工程绘图

AI 生成的内容可能不正确。

用来传送ICMP 报文的IP 数据包上实际上有不少字段。但是实际上与ICMP 协议相关的只有7 个子段:

(1)协议 (2)源IP地址 (3)目的IP地址 (4)生存时间 (5)类型 (6)代码

(7)选项数据

这里面，(1)协议字段值是1,表示ICMP协议。(2)和(3)是用来交流ICMP 报文的地址信息，没有特殊意义。对于理解ICMP 本身，重要的是(5)，(6)，(7)三个字段。这里面的可以称为核心的重要字段是(5)类型，(6)代码这两个字段。所有ICMP 用来交流错误通知和信息询问的报文，都是由类型和代码的组合来表示的。RFC 定义了15种类型。“报文不可到达”这样的错误通知和“回送请求”这样的信息查询是由类型字段来区分的。ICMP报文由类型来表达它的大概意义，需要传递细小的信息时由代码来分类。进一步，需要向对方传送数据的时候，用(7)选项数据字段来放置。

可能的类型代码组合:

表格

AI 生成的内容可能不正确。

本次实验捕获的ICMP报文，都属于信息报文；共捕获四对request包和reply包。

以第一组为例：

第一组request包报文内容如下：

文本, 信件

AI 生成的内容可能不正确。

此报文Type=8，Code=0，表示ping请求。

第一组reply包报文内容如下：

文本

AI 生成的内容可能不正确。

此报文Type=0，Code=0，表示ping应答。

第一组的Sequence number（BE）为1，第二组为2，以此类推。

1. DHCP协议分析

4.1 DHCP协议捕获

第一步：运行Wireshark软件，在首页选择进行连接的网络，选择WLAN

第二步：在应用显示过滤器中输入udp port 67，过滤出DHCP报文

第三步：打开命令提示符窗口，输入ipconfig -release，回车执行，释放计算机的IP地址

第四步：在软件中点击开始抓包，然后在命令提示符窗口输入ipconfig -renew，回车执行，重新分配IP地址

文本

AI 生成的内容可能不正确。

图片包含 文本

AI 生成的内容可能不正确。

捕获到的数据包如下:

图形用户界面, 文本

AI 生成的内容可能不正确。

4.2 DHCP的功能

DHCP也叫动态主机配置协议，负责管理和分配IP。安装了DHCP协议的主机被称为DHCP服务器，服务器控制一段IP地址范围，其他主机可以从DHCP服务器自动获取IP地址、默认网关、DNS服务器地址等信息。

4.3 DHCP的报文分析

抓包到的ACK消息如下:

文本

AI 生成的内容可能不正确。

图形用户界面, 文本

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。

如果DHCP服务器无法满足客户端的请求，例如，如果请求的IP地址已经被占用，它会发送一个NAK包，表示请求失败。

DHCP封包格式:

表格

AI 生成的内容可能不正确。

每个字段含义如下：

op：报文的操作类型。分为请求报文和响应报文。客户端发送给服务器的包为请求报文，值为 1；服务器发送给客户端的包为响应报文，值为 2。

htype：DHCP 客户端的 MAC 地址类型。htype 值为 1 时表示为最常见的以太网 MAC 地址类型。

hlen：硬件地址长度。以太网 MAC 地址长度为 6 个字节，即 hlen 值为 6。

hops：跳数，DHCP 报文经过的中继数量。每经过一个路由器，该字段就会增加 1。如果没有经过路由器，则值为 0（同一网内）。

xid：事务 ID。客户端发起一次请求时选择的随机数，用来标识一次地址请求过程。在一次请求中所有报文的 xid 都是一样的。

secs：DHCP 客户端从获取到 IP 地址或者续约过程开始到现在所过去的时间，以秒为单位。在没有获得 IP 地址前，该字段始终为 0。

flags：BOOTP 标志位。只使用第 0 比特位，是广播应答标识位，用来标识 DHCP 服务器应答报文是采用单播还是广播发送。其中，0 表示采用单播发送方式，1 表示采用广播发送方式。其余位尚未使用。

ciaddr：DHCP 客户端的 IP 地址。仅在 DHCP 服务器发送的 ACK 报文中显示，在其他报文中均显示为 0。这是因为在得到 DHCP 服务器确认前，DHCP 客户端还没有分配到 IP 地址。

yiaddr：DHCP 服务器分配给客户端的 IP 地址。仅在 DHCP 服务器发送的 Offer 和 ACK 报文中显示，其他报文中显示为 0。

siaddr：为 DHCP 客户端分配 IP 地址等信息的其他 DHCP 服务器 IP 地址。仅在 DHCP Offer、DHCP ACK 报文中显示，其他报文中显示为 0。

giaddr：转发代理（网关）IP 地址，DHCP 客户端发出请求报文后经过的第一个 DHCP 中继的 IP 地址。如果没有经过 DHCP 中继，则显示为 0。

chaddr：DHCP 客户端的 MAC 地址。在每个报文中都会显示对应 DHCP 客户端的 MAC 地址。

sname：为客户端分配 IP 地址的服务器名称（DNS 域名格式）。只在 DHCP Offer 和 DHCP ACK 报文中显示发送报文的 DHCP 服务器名称，其他报文显示为 0。

file：DHCP 服务器为 DHCP 客户端指定的启动配置文件名称及路径信息。仅在 DHCP Offer 报文中显示，其他报文中显示为空。

options：可选选项，格式为“代码+长度+数据”。

| **参数​​** | **​​DHCP选项号​​** | **​​功能说明​​** |
| --- | --- | --- |
| ​**​IP地址​**​ | -（直接分配） | 客户端从DHCP服务器获取的IPv4地址（如10.29.196.194）。 |
| ​**​子网掩码​**​ | 1 | 定义客户端所属子网的范围（如255.255.0.0表示/16网络）。 |
| ​**​默认网关（路由器）​**​ | 3 | 客户端的出口路由设备IP（如10.29.0.1）。 |
| ​**​DNS服务器​**​ | 6 | 提供域名解析服务的服务器列表（如8.8.8.8, 8.8.4.4）。 |
| ​**​IP租期​**​ | 51 | IP地址的有效时间（如86400秒=24小时），到期后需续租或重新获取。 |
| ​​域名（Domain Name）​​ | 15 | 客户端所属的DNS域名（如example.com）。 |
| ​​NTP服务器​​ | 42 | 网络时间协议（NTP）服务器地址，用于时间同步。 |
| ​​TFTP服务器​​ | 66 | 用于网络启动（如PXE）时下载引导文件的服务器。 |
| ​​启动文件名​​ | 67 | 网络启动时加载的引导文件路径（如pxelinux.0）。 |
| ​​WINS服务器（NetBIOS）​​ | 44 | Windows Internet Name Service服务器，用于旧版Windows名称解析。 |
| ​​NetBIOS节点类型​​ | 46 | 定义NetBIOS名称解析方式（如广播或DNS）。 |
| ​​MTU（最大传输单元）​​ | 26 | 指定客户端的MTU值（如1500字节）。 |
| ​​静态路由​​ | 33 | 推送静态路由表（如目标网络192.168.1.0/24，网关10.29.0.2）。 |
| ​​VLAN ID​​ | 132 | 为客户端分配VLAN标识（需交换机支持）。 |
| ​​代理自动配置（PAC）URL​​ | 252 | 指定浏览器的代理自动配置脚本地址（如http://proxy.example.com/proxy.pac）。 |
| ​​SIP服务器​​ | 120, 121 | VoIP会话初始协议（SIP）服务器地址。 |
| ​​ARP缓存超时​​ | 35 | 定义ARP条目的存活时间（如300秒）。 |
| ​​IP转发启用/禁用​​ | 19 | 控制客户端是否允许IP转发（如路由器需启用）。 |
| ​​源路由控制​​ | 20 | 禁用或启用源路由（安全防护）。 |
| ​​厂商特定信息（Option 43）​​ | 43 | 为特定设备（如IP电话、AP）传递私有配置（需厂商支持）。 |
| ​​时区偏移​​ | 24 | 客户端时区设置（如UTC+8）。 |
| ​​SMTP服务器​​ | 69 | 邮件发送服务器地址。 |
| ​​POP3服务器​​ | 70 | 邮件接收服务器地址。 |
| ​​HTTP代理​​ | 72 | 指定HTTP代理服务器（旧版选项，现代多用Option 252）。 |

4.4 DHCP工作原理

4.4.1 工作过程

DHCP分配IP地址时的主要工作过程如下:

1. DHCP Client以广播的方式发出DHCP Discover报文。

2. 所有的DHCP Server都能够接收到DHCP Client发送的DHCP Discover报文，所有的DHCP Server都会给出响应，向DHCP Client发送一个DHCP Offer报文。 DHCP Offer报文中“Your(Client) IP Address”字段就是DHCP Server能够提供给DHCP Client使用的IP地址，且DHCP Server会将自己的IP地址放在“option”字段中以便DHCP Client区分不同的DHCP Server。DHCP Server在发出此报文后会存在一个已分配IP地址的纪录。

3. DHCP Client只能处理其中的一个DHCP Offer报文（假如存在多个DHCP Server情况下），一般的原则是DHCP Client处理最先收到的DHCP Offer报文。

4. DHCP Client会发出一个广播的DHCP Request报文，在选项字段中会加入选中的DHCP Server的IP地址和需要的IP地址。

5. DHCP Server收到DHCP Request报文后，判断选项字段中的IP地址是否与自己的地址相同。如果不相同，DHCP Server不做任何处理只清除相应IP地址分配记录；如果相同，DHCP Server就会向DHCP Client响应一个DHCP ACK报文，并在选项字段中增加IP地址的使用租期信息。

图示

AI 生成的内容可能不正确。

4.4.2 工作模式

在捕捉到的报文信息中，在​DHCP Server Identifier信息树中我们可以知道服务器的IP地址是10.3.9.2和头部信息中的源地址相同

而在​​Relay Agent IP Address信息树中，得知中继代理地址为10.29.0.1，不是0.0.0.0，说明客户端与服务端不在同一子网，需要转发。

在Router信息中我们可以知道路由器的地址为10.29.0.1，所以Server端并不是由路由器充当，而是由独立主机10.3.9.2提供的，存在DHCP Relay，且其中路由器起到中继转发的作用。

1. ARP协议分析

5.1 ARP协议捕获

第一步：运行Wireshark软件，在首页选择进行连接的网络，选择WLAN

第二步：在应用显示过滤器中输入arp，过滤出ARP报文

第三步：打开命令提示符窗口，输入ipconfig -release，回车执行，释放计算机的IP地址

第四步：在软件中点击开始抓包，然后在命令提示符窗口输入ipconfig -renew，回车执行，重新分配IP地址

文本

AI 生成的内容可能不正确。

图片包含 文本

AI 生成的内容可能不正确。

捕获到的数据包如下:

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。

5.2 ARP的功能

在以太网环境中，数据的传输所依懒的是MAC地址而非IP地址，而将已知IP地址转换为MAC地址的工作是由ARP协议来完成的。

在局域网中，网络中实际传输的是“帧”，帧里面是有目标主机的MAC地址的。在以太网中，一个主机和另一个主机进行直接通信，必须要知道目标主机的MAC地址。但这个目标MAC地址是如何获得的呢？它就是通过地址解析协议获得的。

所谓“地址解析”就是主机在发送帧前将目标IP地址转换成目标MAC地址的过程。ARP协议的基本功能就是通过目标设备的IP地址，查询目标设备的MAC地址，以保证通信的顺利进行。

5.3 工作原理

由抓包到的一对请求包和应答包，分析如下:

文本

AI 生成的内容可能不正确。

文本

AI 生成的内容可能不正确。

请求报文的目的地址是广播，因为此时为获取某个IP地址的mac地址需要先进行广播，并且此时目的端的mac地址填充为全0，因为此时还不知道目的mac地址。

总结流程如下:

假设主机A和B在同一个网段，主机A要向主机B发送信息，具体的地址解析过程如下：

(1) 主机A首先查看自己的ARP表，确定其中是否包含有主机B对应的ARP表项。如果找到了对应的MAC地址，则主机A直接利用ARP表中的MAC地址，对IP数据包进行帧封装，并将数据包发送给主机B。

(2) 如果主机A在ARP表中找不到对应的MAC地址，则将缓存该数据报文，然后以广播方式发送一个ARP请求报文。

ARP请求报文中的发送端IP地址和发送端MAC地址为主机A的IP地址和MAC地址，目标IP地址和目标MAC地址为主机B的IP地址和全0的MAC地址。

由于ARP请求报文以广播方式发送，该网段上的所有主机都可以接收到该请求，但只有被请求的主机（即主机B）会对该请求进行处理。

(3) 主机B比较自己IP地址和ARP请求报文中的 目标IP 地址，当两者相同时进行如下处理：将ARP请求报文中的发送端（即主机A）的IP地址和MAC地址存入自己的ARP表中。

之后以单播方式发送ARP响应报文给主机A，其中包含了自己的MAC地址。

5.4 ARP包格式

图片包含 日程表

AI 生成的内容可能不正确。

硬件类型：硬件地址的类型，1 表示以太网地址。

协议类型：要映射的协议地址类型，ARP 协议的上层协议为IP 协议，因此该协议类型为IP 协议，其值为0x0800。

硬件地址长度：硬件地址（MAC 地址）的长度，以字节为单位。对于以太网上IP 地址的ARP 请求或者应答来说，该值为6。

协议地址长度：IP 地址的长度，以字节为单位。对于以太网上IP 地址的ARP 请求或者应答来说，该值为4。

OP：操作码，用于表示该数据包为ARP 请求或者ARP 应答。1 表示ARP 请求，2 表示ARP应答。

源MAC 地址：发送端的硬件地址。

源IP 地址：发送端的协议（IP）地址，如192.168.1.102。

目的MAC 地址：接收端的硬件地址，在ARP 请求时由于不知道接收端MAC 地址，因此该字段为广播地址，即48’hff\_ff\_ff\_ff\_ff\_ff。

目的IP 地址：接收端的协议（IP）地址，如192.168.1.10。

1. TCP协议分析

6.1 捕获TCP报文

第一步：运行Wireshark软件，在首页选择进行连接的网络，选择WLAN

第二步：在应用显示过滤器中输入tcp port 80，过滤出ARP报文

第三步：开始捕捉，并打开浏览器输入[www.baidu.com](http://www.baidu.com)进入百度网页

第四步：待网页加载完毕后，关闭浏览器

捕捉到的报文如下:

表格

AI 生成的内容可能不正确。

6.2 TCP报文首部

表格

AI 生成的内容可能不正确。

|  |  |  |
| --- | --- | --- |
| 名字 | 长度(bit) | 功能 |
| 源端口号 | 16 | 发送方主机的应用程序的端口号 |
| 目的端口号 | 16 | 目的主机的应用程序的端口号 |
| 序列号 | 32 | 表示本报文所发送数据的第一个字节的编号 |
| 确认号 | 32 | 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 |
| 首部长度 | 4 | 确定TCP报文的报头部分长度，告诉接收端应用程序，数据(有效载荷)从何处开始 |
| 保留字段 | 6 | 为TCP将来的发展预留空间，目前必须全部为0 |
| 标志位 | 6 | 共有6个标志位，每个标志位占1个bit |
| 窗口大小 | 16 | 表示发送该TCP报文的接受窗口还可以接受多少字节的数据量。该字段用于TCP的流量控制 |
| 校验和 | 16 | 用于确认传输的数据有无损坏 。 |
| 紧急指针 | 16 | 仅当标志位字段的URG标志位为1时才有意义。指出有效载荷中为紧急数据的字节数。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。 |
| 选项字段 | 长度不定，必须是32的整数倍 | 提供了一些额外的功能或对TCP协议进行扩展 |

6.3 连接建立消息和连接释放消息

6.3.1 标志位和序号的作用

标志位涉及TCP的连接管理，三次握手:

|  |  |
| --- | --- |
| 名称 | 功能 |
| URG | 表示本报文中发送的数据(有效载荷)是否包含紧急数据：URG=1时表示有紧急数据；当URG=1时，后续的16位紧急指针字段才有效 |
| ACK | 表示本报文前面的确认号字段是否有效：只有当ACK=1时，前面的确认号字段才有效；TCP规定，建立连接后，ACK必须为1 |
| PSH | 告诉对方收到该报文段后，上层应用程序立即把数据从TCP接收缓冲区读取，保证TCP接收缓冲区有能力接收新数据或清空TCP接收缓冲区 |
| RST | 表示是否重置连接：若RST=1，说明TCP连接出现严重错误(如主机崩溃)，必须释放连接，重新建立连接。携带RST标识的报文称为复位报文段 |
| SYN | 在建立连接时使用，用来同步序号；当SYN=1，ACK=0时，表示该报文为请求建立连接的报文；当SYN=1，ACK=1时，表示同意建立连接；只有在建立连接的前两次请求中SYN才为1。该报文称为同步报文段 |
| FIN | 标记数据是否发送完毕：若FIN=1，表示数据已经发送完毕，可以释放连接。该报文称为结束报文段 |

序号: 在网络通信中，应用层调用的write，sendto等接口，只是将我们自己定义的缓冲区数据拷贝到TCP的发送缓冲区。而TCP是面向字节流的。在TCP缓冲区中，以字节为单位，将缓冲区划分成了类似字符数组的形式，数组自带下标，所以TCP缓冲区的每个字节就有编号，序列号和确认号就是TCP缓冲区的编号(下标)。

6.3.2 TCP连接分析

序号2~4：客户端向服务器发送 SYN 请求建立连接，服务器返回 SYN+ACK 应答，客户端返回 ACK 完成连接建立。即“三次握手”。

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。

TCP连接建立阶段的第一个数据报内容如下：

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。

其中SYN = 1：用于建立连接；

ACK = 0：表示ACK字段无效；

FIN = 0：表示FIN字段无效；

Sequence number = 0：表示该报文的起始字节序号为 0；

Acknowledgement number = 0：表示接下来期望收到序号为 0 的字节。

第二个数据报内容如下：

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。

其中SYN = 1：用于建立连接；

ACK = 1：表示ACK字段有效；

FIN = 0：表示FIN字段无效；

Sequence number = 0：表示该报文的起始字节序号为 0；

Acknowledgement number = 1：表示接下来期望收到序号为 1 的字节。

第三个数据报内容如下：

文本

AI 生成的内容可能不正确。

其中SYN = 0：表示SYN字段无效；

ACK = 1：表示ACK字段有效；

FIN = 0：表示FIN字段无效；

Sequence number = 1：表示该报文的起始字节序号为 1；

Acknowledgement number = 1：表示接下来期望收到序号为 1 的字节。

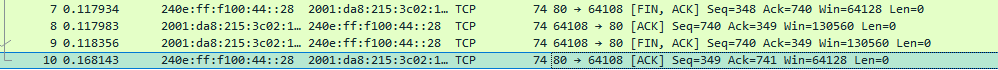
上述TCP连接的建立阶段可以总结为“本地发送SYN、服务端返回SYN+ACK、本地发送 ACK”，如下图所示：

图示

AI 生成的内容可能不正确。

6.3.3 TCP释放连接分析

序号7~10的数据报展示了客户端和服务器分别发送FIN请求，对方返回应答，最终返回ACK完成连接释放的过程。即“四次挥手”：



“第一次挥手”：

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。

其中SYN = 0：表示SYN字段无效；

ACK = 1：表示ACK字段有效；

FIN = 1：表示FIN字段有效；

Sequence number = 348：表示该报文的起始字节序号为 348；

Acknowledgement number = 740：表示接下来期望收到序号为 740 的字节。

“第二次挥手”：

文本

AI 生成的内容可能不正确。

其中SYN = 0：表示SYN字段无效；

ACK = 1：表示ACK字段有效；

FIN = 0：表示FIN字段无效；

Sequence number = 740：表示该报文的起始字节序号为 740；

Acknowledgement number = 349：表示接下来期望收到序号为 349 的字节。

“第三次挥手”：

图形用户界面, 文本, 应用程序

AI 生成的内容可能不正确。

其中SYN = 0：表示SYN字段无效；

ACK = 1：表示ACK字段有效；

FIN = 1：表示FIN字段有效；

Sequence number = 740：表示该报文的起始字节序号为 740；

Acknowledgement number = 349：表示接下来期望收到序号为 349 的字节。

“第四次挥手”：

图形用户界面, 文本

AI 生成的内容可能不正确。

其中SYN = 0：表示SYN字段无效；

ACK = 1：表示ACK字段有效；

FIN = 0：表示FIN字段无效；

Sequence number = 349：表示该报文的起始字节序号为 349；

Acknowledgement number = 741：表示接下来期望收到序号为 741的字节。

上述 TCP 连接的断开阶段可以总结为“一端发送 FIN、另一端返回 ACK、另一端发送 FIN、一端返回 ACK”，即“四次挥手”。如下图所示：

图表

AI 生成的内容可能不正确。

6.4 数据传输过程

发送序号（Sequence Number）​​

作用：标识数据字节流的起始位置（如本节图3中相对序列号1对应原始值3701725488）

示例：SYN包消耗一个序列号（即使无数据），如本节图1中SYN包的Next Seq=1

​​应答序号（Acknowledgment Number）​​

作用：期望收到的下一个字节序号（如本节图2中ACK=3701725488表示已收到SYN）

规则：ACK号=收到的Seq号+数据长度（本节图7中ACK=741是对FIN的确认）

​​标志位（Flags）​​

SYN：建立连接（本节图1标志0x002）

ACK：确认有效（本节图3标志0x010）

FIN：终止连接（本节图4标志0x011含FIN+ACK）

RST：强制中断

​​窗口大小（Window）​​

作用：接收方可用缓冲区大小（如本节图4窗口501→计算值64128字节）

动态调整：本节图5窗口510→130560显示流量控制过程

​​数据长度（Segment Len）​​

本节图1-3显示Len=0（握手阶段），本节图4-7实际数据传输未显示具体数据长度

​​MSS（Maximum Segment Size）​​

MSS指的是TCP协议中数据段的最大大小，它是在TCP连接建立时协商确定的。MSS是指在TCP数据包中的数据部分的最大允许长度，不包括TCP头部和IP头部的大小。TCP协议中的MSS通常由TCP连接的双方协商确定，取双方中较小的那个值。

流程图如下:

表格, 日程表

AI 生成的内容可能不正确。

1. 实验结论和心得

通过本次《计算机网络》实验中使用 Wireshark 软件捕获和解析协议数据，我对 TCP/IP 协议栈的核心协议有了更深入的理解与掌握。在 IP 协议分析中，通过 ping 命令生成大数据包，观察到 IP 分段传输现象，验证了包头校验和的计算过程，清晰理解了各字段在数据传输中的作用；ICMP 协议分析里，借助 ping 命令捕获的请求与应答报文，明确了其用于网络故障检测和连通性测试的功能及消息格式；DHCP 协议分析中，通过释放与重新申请 IP 地址，完整追踪了从 Discover 到 ACK 的四次消息交互过程，掌握了动态 IP 分配机制及参数配置；ARP 协议分析时，观察到广播请求与单播应答的流程，深刻理解了 IP 地址到 MAC 地址的解析原理；TCP 协议分析则通过访问网页的过程，详细剖析了三次握手建立连接、四次挥手释放连接的机制，以及序号、窗口管理在流量控制和差错控制中的作用。实验过程中，从软件的过滤器设置到协议字段的逐项解读，不仅提升了网络协议的理论认知，更强化了实际抓包分析与问题排查的实践能力，体会到网络协议在数据通信中层层封装、协同工作的精妙设计。

本次实验总的来说难度不大，抓包过程仅花费1小时，主要遇到的困难还是对捕获的数据包结合理论知识进行分析，有很多不懂得地方需要查找和学习。