第3章 传输层

内容简介

本章内容主要介绍传输层 TCP 协议和 UDP 协议。实验利用基于 HTTP 协议的应用建立 TCP 连接。通过对数据的分析，使读者了解 TCP 协议建立连接、断开连接、数据传输等方面的内容。通过 DNS 应用构造 UDP 协议的通信，使读者通过分析 UDP 协议报文，理解 UDP 协议的相关知识。通过本章实践，熟悉传输层协议报文格式，理解可靠传输的工作原理。

3.1 传输层协议

对于 TCP/IP 网络，为应用层提供了两种截然不同的传输层协议。其中一种是 UDP 协议，为调用它的应用程序提供了一种不可靠的、无连接的服务。另一种是 TCP 协议，为调用它的应用程序提供了一种可靠的、面向连接的服务。当设计一个网络应用程序时，该应用程序的开发人员必须指定使用这两种传输层协议中的哪一种。表 3-1 是常用的因特网应用及其采用的传输层协议及其端口号。

表 3-1

序号	应　用	应用层协议	下面的协议	默认端口号
1	电子邮件	SMTP	TCP	25
2	电子邮件	POP3	TCP	110
3	远程终端访问	TELNET	TCP	23
4	Web	HTTP	TCP	80
5	文件传输	FTP	TCP	20,21
6	名字转换	NDS	TCP/UDP	53
7	动态主机配置	DHCP	UDP	67,68
8	选路协议	RIP	UDP	—
9	流式多媒体	通常专用	通常 UDP	—
10	因特网电话	通常专用	通常 UDP	—
11	网络管理	SNMP	UDP	—

如表 3-1 所示，电子邮件、远程终端访问、Web 及文件传输都是运行在 TCP 之上，因为这些应用都需要 TCP 可靠的数据传输服务。而 UDP 常应用在多媒体领域，这些应用都能容忍少量的分组丢失，但对实时性要求相对较高。

3.1.1 TCP 协议

1. TCP 协议概述

传输控制协议(Transmission Control Protocol,TCP),是面向连接的传输层协议,为上层提供可靠的数据传输,同时提供流量和拥塞控制。基于 TCP 的数据传输包括 3 个阶段:第一阶段是双方建立连接;第二阶段是数据传输阶段,该阶段双方传输数据;在第三阶段,当双方已经结束了数据传输时,就关闭连接。

2. MSS 的原理

MSS(Maximum Segment Size)就是 TCP 报文段每次能够传输的最大数据量。为了达到最佳的传输效能 TCP 协议在建立连接的时候通常要协商双方的 MSS 值,这个值受限于链路层帧长度,也就是最大传输单元(Maximum Transmission Unit,MTU)。以太网中 MTU=1500,所以 TCP 协议在实现的时候,往往用 MTU 值减去 IP 数据包的首部 20 字节和 TCP 数据段的首部 20 字节,也就是 1500−40=1460 字节作为 MSS。通信双方会根据双方提供的 MSS 值的最小值确定为这次连接的最大 MSS 值。

MSS 限制了报文段数据字段的最大长度,当 TCP 发送一个大文件时,通常是将文件分成长度为 MSS 的若干块进行传输。

3. TCP 报文格式

TCP 报文分为 TCP 首部和应用数据两部分,如图 3-1 所示。首部的前 20 个字节是固定的,后面的选项是可选的,但长度必须是 4 字节的整数倍。因此 TCP 报文首部的最小长度是 20 字节。

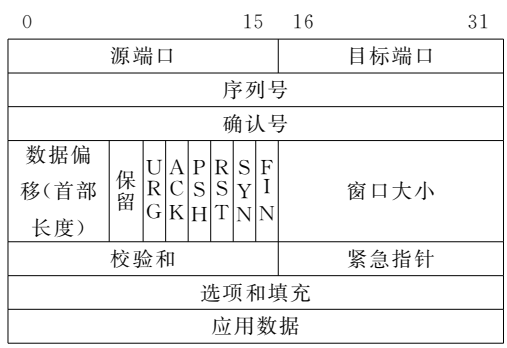

图 3-1

以下为 TCP 首部各字段的简要说明。

(1) 源端口:16 位的源端口号。

(2) 目标端口:16 位的目标端口号。TCP 报文中源端口字段、目标端口字段加上 IP 报文中的源 IP 地址、目标 IP 地址字段构成一个四元组(源端口,源 IP 地址,目标端口,目标 IP 地址),该四元组可以唯一标识一个 TCP 连接。

(3) 序列号:32 位,标识从发送端向接收端发送的数据字节流,表示数据段中第一个数

据字节的序号。如果 SYN 控制位是被置位的，则序列号是初始序号(n)，而第一个数据字节为 $n+1$。例如：一个报文段中序列号为 201，报文段共有 200 字节，则下一个报文段的序列号就是 401。

(4) 确认号：32 位，如果 ACK 控制位是被置位的，则这个字段表示接收方要接收的下一个数据包的第一个数据字节序列号，同时也表明在确认号之前的所有数据接收方都已经收到。例如：如果接收方收到一个报文段的序列号字段值是 501，数据长度是 200 字节，则其确认报文中确认号字段的值应该是 701，表明 501 到 700 之间的数据都已经收到，下一次期望收到从 701 开始的数据。

(5) 数据偏移：4 位，指明数据从哪里开始，也就是首部长度，以 32 位比特为单位。

(6) 保留字段：6 位。

(7) 标志位：6 位，控制字段，用于 TCP 的流量控制、连接的建立、终止以及数据的传送方式。

URG——指明报文段中存在着被发送方上层的实体标记为“紧急”的数据。紧急数据的最后一个字节由紧急指针字段指出。也就是为 1 时，表明紧急指针有效，否则无效。

ACK——确认，为 1 表示对已被成功接收报文段的确认。

PSH——该位为 1，指示接收方应立即将数据交给上层处理，而不是在缓存中排队。

RST——重建连接标志。用来复位那些产生错误的连接。

SYN——同步标志，为 1 时用来发起一个连接。

FIN——结束标志，为 1 时表示发送端完成发送任务。

(8) 窗口大小：16 位，用来配合 TCP 中的流量控制算法实现流量控制。表示希望收到的每个 TCP 数据段的大小。

(9) 校验和：16 位，用来校验整个 TCP 报文段的所有数据的正确性，包括 TCP 首部和数据。

(10) 紧急指针：16 位，只有在 URG=1 时有效。

(11) 选项：其长度可变，但必须是 4 字节的整数倍。每一选项的第一个字节为代码，其后一个字节为该选项长度，接着为选项内容。共有 7 种可能选项。最常见的选项是在 TCP 握手阶段用于发送方与接收方协商最大报文段长度。选项代码及其含义参见表 3-2。

表 3-2

代码	长度	含 义
0	—	选项列表末尾
1	—	无操作
2	4	最大报文段长度(MSS)
3	3	窗口比例
4	2	允许选择性确认
5	X	选择性确认
8	10	时间戳

(12) 填充：为了使 TCP 首部的总长度达到 32 位的倍数，使用全 0 的字节填充 TCP 首部。

通过上述 TCP 首部字段的分析，标志字段中，RST、SYN、FIN 用于建立连接和拆除连

接，URG、PSH、紧急指针用于对紧急数据的处理，在实际中不常用。源和目标端口用于多路复用/多路分解来自或送至上层应用数据。校验和、序列号、确认号字段用来实现可靠数据传输服务。窗口字段用于流量控制。

4. TCP 连接（3 次握手）

TCP 连接的建立是从客户机向服务器发送一个主动打开请求而启动的。如果服务器已经执行了被动打开操作，那么双方就可以交换报文以建立 TCP 连接。只有在 TCP 连接之后，双方才开始收发数据。如图 3-2 所示，TCP 连接建立过程使用了 3 次握手，主要是连接双方需要商定一些参数。

首先客户机发送一个连接建立请求报文给服务器，声明它将使用的初始序列号（SYN，seq＝x）。服务器用一个连接建立响应报文确认客户机的序号（ACK，ack＝seq＋1），同时声明自己使用的初始序列号（SYN，seq＝y）。也就是在第二个报文的标志字段中 SYN，ACK 标志位都置 1。最后客户机用第三个报文来响应并且确认服务器的初始序列号（ACK，ack＝y＋1）。

客户机、服务器的初始序列号一般都是随机产生的。

5. TCP 终止（断开连接）

TCP 连接建立之后，双方就可以传输数据。当其中一方发送完数据后，就会关闭它这一方的连接，同时向对方发送撤销 TCP 连接的报文。同样，另外一方发送数据结束，也会关闭自己一方的连接，同时向对方发送撤销 TCP 连接的报文。

如图 3-3 所示，客户机发出一个关闭连接报文（FIN，seq＝u）给服务器，服务器收到该报文就向客户机返回一个确认报文（ACK，ack＝u＋1）。然后，服务器向客户机发出关闭连接报文（FIN，seq＝v），同样客户机收到该关闭报文后，向服务器返回一个确认报文（ACK，ack＝v＋1）。

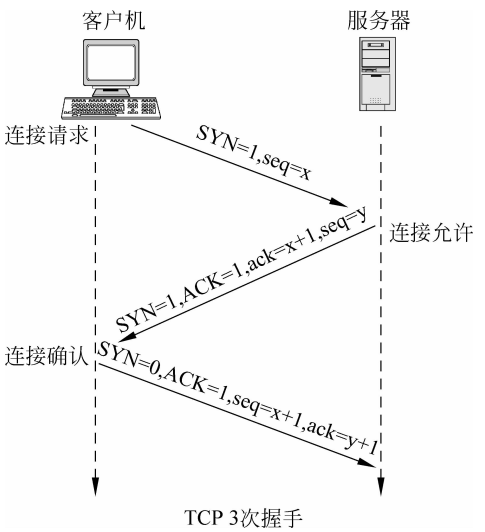

图 3-2

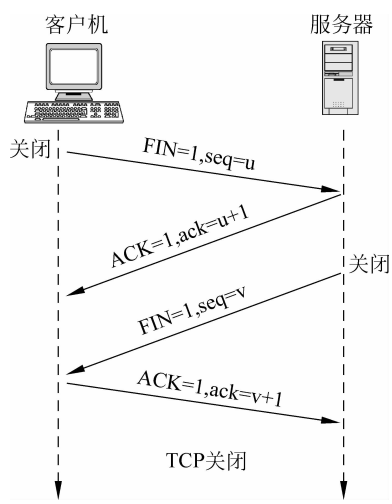

图 3-3

3.1.2 UDP 协议

1. UDP 协议概述

用户数据报协议(User Datagrame Protocol,UDP),是无连接的传输层协议。与 TCP 协议相比,它虽然不能直接为应用层提供可靠的数据传输。但通过在应用程序自身中建立可靠性机制(例如,可通过增加确认与重传机制),也可以实现可靠的数据传输。

2. UDP 报文格式

在 UDP 协议中,标识不同进程的方法是在 UDP 报文的首部包含了发送进程和接收进程使用的 UDP 端口号。

UDP 报文包括 UDP 首部和应用层数据两部分。其中 UDP 首部长度为 8 字节,包括 4 个字段,分别是源端口、目标端口、长度、校验和。

UDP 报文格式如下:

源端口	目标端口
长度	校验和
应用数据	

以下为 UDP 首部各字段的简要说明。

- 源端口:16 位的源端口号。
- 目标端口:16 位的目标端口号。
- 长度:16 位,包括首部在内的 UDP 报文段长度(以字节为单位)。
- 校验和:16 位。

3.2 实验九:TCP 协议分析

3.2.1 实验目的

1. 通过实验,熟悉 TCP 协议的报文格式。
2. 了解和掌握 TCP 协议建立连接和断开连接的过程和步骤。
3. 了解基于 TCP 协议进行数据传输的过程。

3.2.2 实验内容

1. 进行基于 TCP 协议的应用,捕获基于 TCP 协议的数据。
2. 分析和理解 TCP 协议创建连接(握手)和断开连接(分手)的过程。
3. 分析和理解 TCP 协议的数据传输机制和过程。

3.2.3 实验步骤

通过发送邮件过程捕获 TCP 协议数据包(创建连接、断开连接、数据传输)。图 3-4 是通过邮件代理 Outlook Express 发送邮件,利用 Ethereal 捕获的数据。

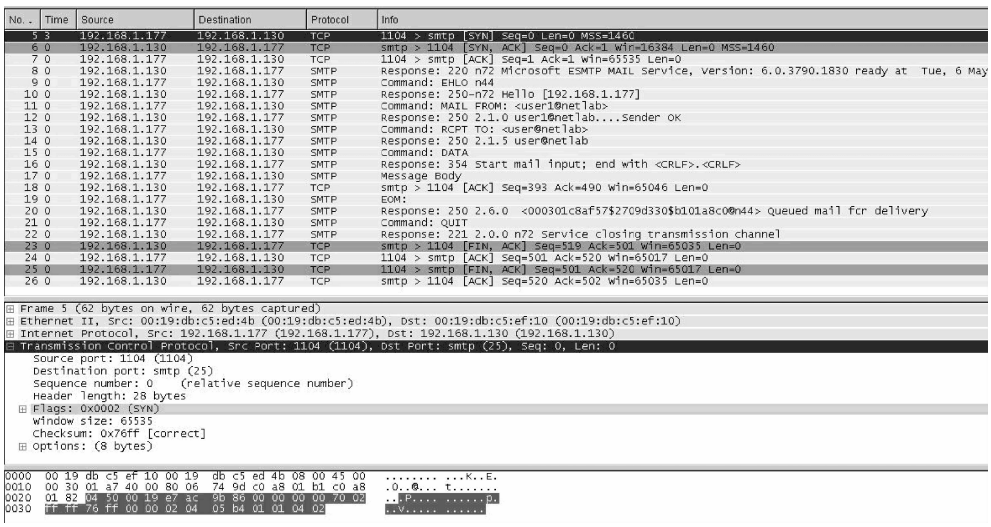

图 3-4

从图 3-4 可以看到,帧 5～7 是建立 TCP 的连接(3 次握手)过程,帧 23～26 是 TCP 连接断开过程。中间帧 8～22 是客户机与邮件服务器之间传输数据的过程,下面主要分析 TCP 的握手和断开连接过程。

1. 建立 TCP 连接(3 次握手)和第一次数据传输

3 次握手的目的是使数据段的发送和接收同步。同时也向其他主机表明其一次可接收的数据量(窗口大小),并建立逻辑连接。

(1) TCP 连接的第一个数据包分析(参见图 3-5)。

图 3-5 为 TCP 连接的第一个数据包的解码,下面来详细说明。

- 源端口:04-50(1104)。
- 目标端口:00-19(25)。邮件服务器端口号。
- 序列号:E7-AC-9B-86(3886848902,为客户机起始序列号)。
- 确认号:00-00-00-00(0)。
- 首部长度:70(28 个字节,20 字节的固定首部,8 字节的选项)。
- 标识符:02(00000010,其中 SYN 置 1,表示为握手报文)。
- 窗口大小:FF-FF(65535)。
- 校验和:76-FF。
- 紧急指针:00-00。
- 选项:02-04-05-B4(表示报文最大长度 05-B4,也就是 1460)。
- 选项中的两个 NOP:01-01。表示无操作。

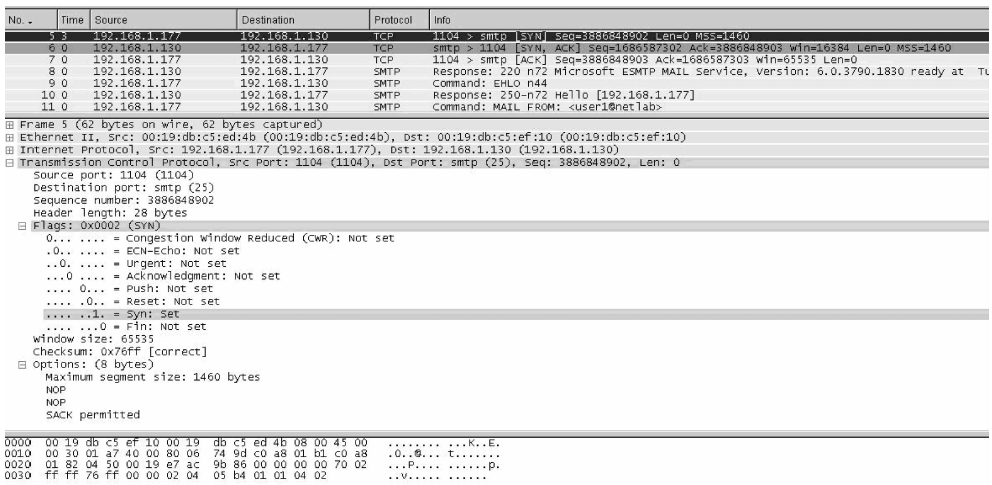

图 3-5

• 选项中的 SACK permitted：04-02。表示允许选择性确认。

(2) TCP 连接的第二个数据包分析(参见图 3-6)。

图 3-6 为 TCP 连接的第二个数据包的解码,下面来详细说明。

• 源端口：00-19(25)。
• 目标端口：04-50(1104)。
• 序列号：64-87-47-A6(1686587302,邮件服务器起始序列号)。
• 确认号：E7-AC-96-87(3886848903,也就是客户机起始序列号+1)。
• 头部长度：70(28 个字节)。
• 标识符：12(00010010,其中 ACK 置 1,SYN 置 1)。
• 窗口大小：40-00(16384)。
• 校验和：8A-C0。

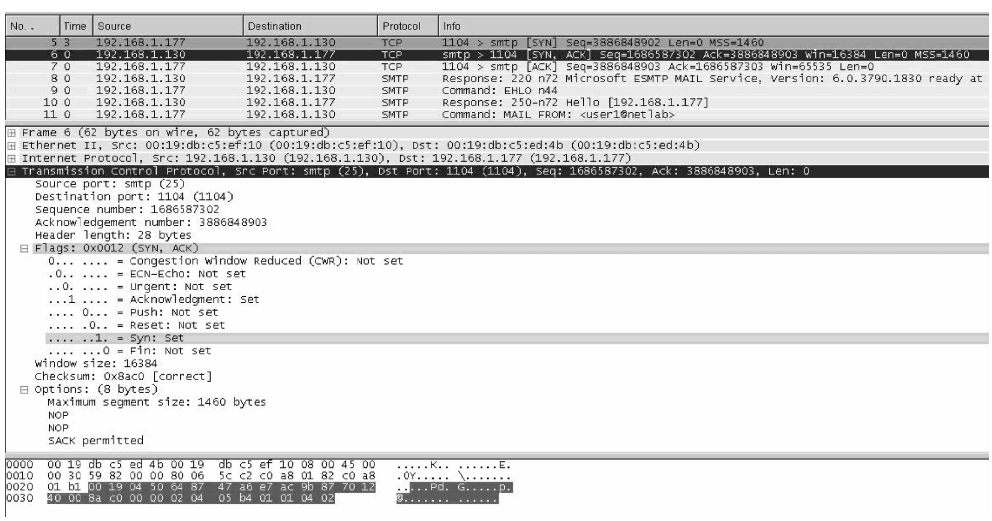

图 3-6

- 紧急指针：00-00。
- 选项：02-04-05-B4(表示报文最大长度 05-B4，也就是 1460)。
- 选项中的两个 NOP：01-01。表示无操作。
- 选项中的 SACK permitted：04-02。表示允许选择性确认。

(3) TCP 连接的第三个数据包分析(参见图 3-7)。

图 3-7 为 TCP 连接的第三个数据包的解码，下面来详细说明。

- 源端口：04-50(1104)。
- 目标端口：00-19(25)。
- 序列号：E7-AC-96-87(3886848903，客户机起始序列号＋1)。
- 确认号：64-87-47-A7(1686587303，也就是服务器起始序列号＋1)。
- 头部长度：50(20 个字节)。
- 标识符：10(00010000，其中 ACK 置 1)。
- 窗口大小：FF-FF(65535)。
- 校验和：F7-84。
- 紧急指针：00-00。

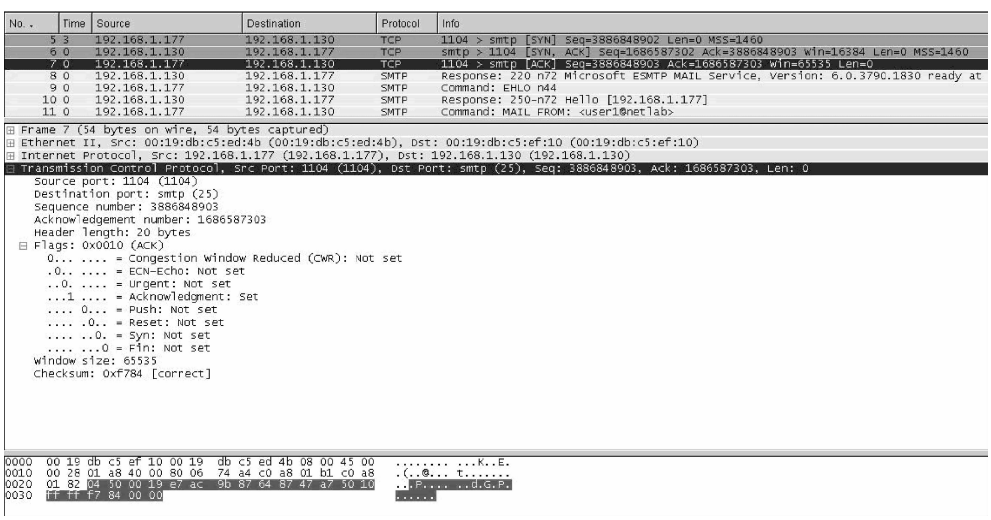

图 3-7

(4) TCP 握手后，服务器发给客户机的第一个数据包分析(参见图 3-8)。

图 3-8 为 TCP 3 次握手后服务器给客户机发送的第一个数据包，下面来详细说明。

- 源端口：00-19(25)。
- 目标端口：04-50(1104)。
- 序列号：64-87-47-A7(1686587303，服务器起始序列号＋1)。
- 确认号：E7-AC-96-87(3886848903，客户机起始序列号＋1)。
- 头部长度：50(20 个字节)。
- 标识符：18(00011000，其中 ACK 置 1，PSH 置 1)。
- 窗口大小：FF-FF(65535)。

- 校验和：CC-6A。
- 紧急指针：00-00。

应用层 SMTP 的数据：104 个数据。32-32-30-20……30-20-0D-0A。

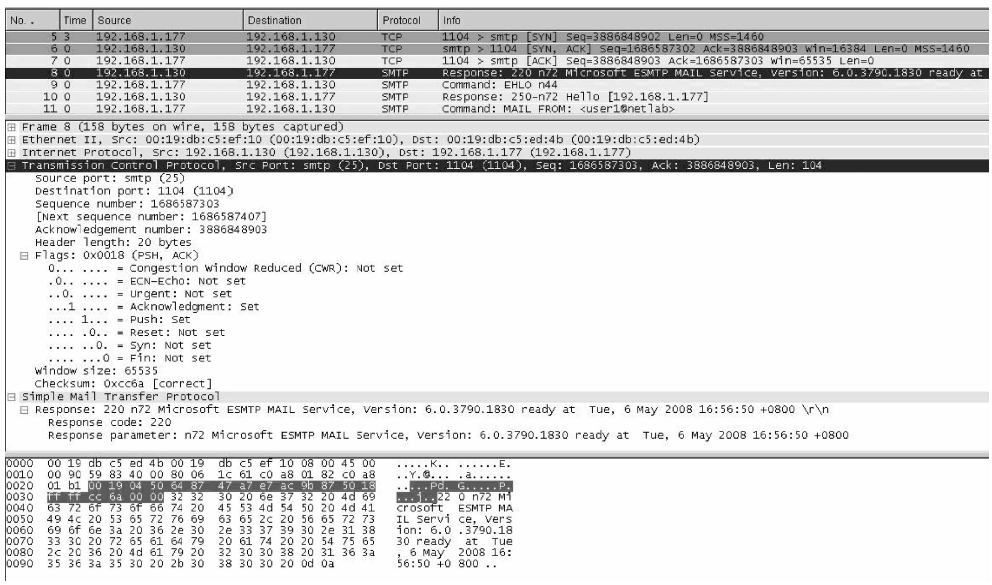

图 3-8

(5) TCP 握手后，客户机给服务器发送的第一个数据包分析(参见图 3-9)。

图 3-9 为 TCP 3 次握手后客户机给服务器发送的第一个数据包，下面来详细说明。

- 源端口：04-50(1104)。
- 目标端口：00-19(25)。
- 序列号：E7-AC-9B-87(3886848903，客户机起始序列号＋1)。

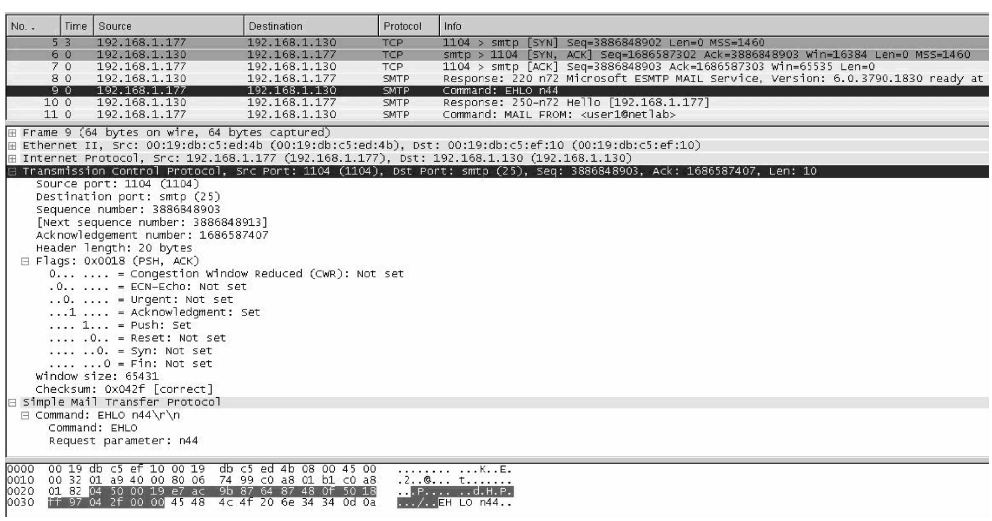

图 3-9

- 确认号：64-87-48-0F(1686587407,为上述步骤(3)中确认号 1686587303＋步骤(4)中发送的 104 个数据)。
- 头部长度：50(20 个字节)。
- 标识符：18(00011000,其中 ACK 置 1,PSH 置 1)。
- 窗口大小：FF-97(65431)。
- 校验和：04-2F。
- 紧急指针：00-00。
- 应用层 SMTP 的数据：10 个数据。45-48-4C…34-0D-0A。

从以上捕获到的数据包分析,TCP 的 3 次握手过程如下：

① 客户机发送一个带 SYN 标志的 TCP 报文到服务器,客户机起始序列号 3886848902,确认号 0。

② 服务器回应客户机一个带 ACK 标志和 SYN 标志的报文,服务器起始序列号 1686587302,确认号为客户机起始序列号 3886848902 加 1,即 3886848903。

③ 客户机必须再次回应服务器一个 ACK 报文,序列号为客户机起始序列号 3886848902 加 1,确认号为服务器起始序列号 1686587302 加 1,即 1686587303。

2. TCP 连接断开过程

图 3-10 为 TCP 断开前,客户机发给服务器的最后一个报文,序列号 3886849397,确认号 1686587769,应用层 SMTP 数据 6 个。

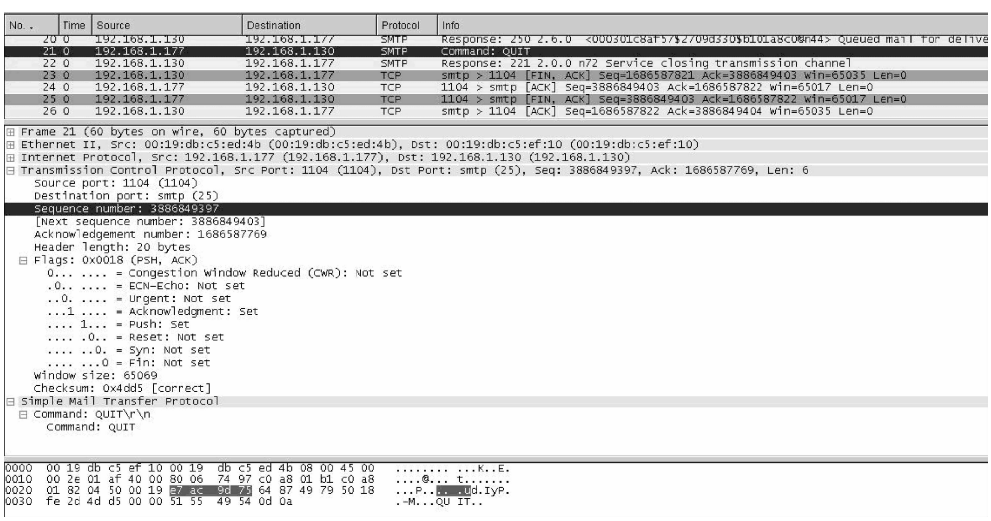

图 3-10

图 3-11 为 TCP 断开前,服务器发给客户机的最后一个报文,序列号 1686587769,确认号 3886849403,应用层 SMTP 数据 52 个。

(1) 服务器发出的断开 TCP 连接报文(参见图 3-12)

图 3-12 为服务器发出的断开 TCP 连接的数据包的解码,下面来详细说明。

- 源端口：00-19(25)。