

《计算机网络》实验指导书

实验二 基于 TCP 的文件传输与 Socket 程序设计

北京邮电大学网络安全学院

2025 年 5 月

目 录

1. 实验类别	1
2. 实验内容和实验目的	1
3. 实验学时	1
4. 实验组人数	1
5. 实验设备环境	1
6. 教学要点与学习难点	1
7. 实验步骤	1
7.1 准备工作	1
7.2 开发设计	2
7.2.1 总体架构	2
7.2.2 客户端	3
7.2.3 服务器端	4
7.2.4 SSL 安全连接通信	4
7.3 撰写实验报告	4
8. 实验报告要求	4
9. 程序验收过程及要求	5

实验二：基于 TCP 的文件传输与 Socket 程序设计

1. 实验类别

网络应用开发型

2. 实验内容和实验目的

本次实验主要包含下列内容：

- 1) 使用 Socket API 通信实现文件传输客户端和服务端 2 个程序，客户端发送文件传输请求，服务器端将文件数据发送给客户端，两个程序均在命令行方式下运行，要求至少能传输 1 个文本文件和 1 个图片文件；
- 2) 客户端在命令行指定服务器的 IP 地址和文件名。为防止重名，客户端将收到的文件改名后保存在当前目录下。客户端应输出：新文件名、传输总字节数；或者差错报告；
- 3) 服务器端应输出：客户端的 IP 地址和端口号；发送的文件数据总字节数；必要的差错报告（如文件不存在）。
- 4) 可选功能：支持 SSL 安全连接。

通过本实验，学生可以深入理解传输层的 TCP 协议原理以及 Socket 套接字网络通信的流程。

3. 实验学时

6 学时。

4. 实验组人数

每组 2 人，分工协作完成客户端开发、服务器端开发、联合调试、实验测试并撰写实验报告。

5. 实验设备环境

一台装有 MS Windows 系列操作系统、Linux 或 Mac 操作系统的计算机，能够连接到因特网，并能基于 C/C++/Python/Java 进行开发。

6. 教学要点与学习难点

在课堂教学和教材中，涵盖了套接字通信、端口号、TCP 协议等传输层技术的原理，但是学生对于通信的细节及其作用过程缺乏感性认识，理解不足。在本实验中，学生通过基于 Socket API 进行文件传输客户端与服务端开发，可以加深对于上述技术要点的理解；并且对于教材中没有包含或阐述不够详细的内容，例如 Socket 类型、TCP 的协议状态转换等，均可以通过应用开发和程序运行流程进行自学和了解。

在本实验中，文件读写功能开发的工作比较简单，实验的重点和难点在于服务器端与客户端的 Socket API 开发与联调工作。

7. 实验步骤

7.1 准备工作

1. 下载 C/C++/Python/Java 的编译环境及开发 IDE 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。

7.2 开发设计

7.2.1 总体架构

Socket（套接字）提供了用户访问网络的通信接口（API），能够允许开发者通过自行开发的程序运行调用 OS 控制的 TCP/IP 协议栈软件进行网络通信。它起源于 80 年代初期，是 UNIX BSD 的一部分，是所有因特网应用的基础。

对于 OS 内核而言，Socket 是网络通信的一个端点（Endpoint）。一个 Socket 地址是由 IP 地址和端口号组成。因此，通信关系由一对 Socket 地址标识出来。

对于网络应用程序而言，Socket 是进程间（远程）通信的一种机制。Socket 是基于客户-服务器（C/S）通信模式的，它是系统 I/O 的扩展，客户/服务器通过 Socket 描述符（类似文件描述符）能够对网络进行读写（发送/接收）操作，同时还增强了一些特殊的系统调用。在 Unix/Linux 的文件模型中，所有 I/O 都被看作文件，由文件描述符标识。常见的进程文件描述符有：0-stdin, 1-stdout, 2-stderr, 3-file, 4-device, 5-socket 等。常见的文件操作则有：打开(Open)、关闭(Close)、读取(Read)、写入(Write)。

Socket 自身的类型又可以分为 5 种：1) 流式 Socket(Stream Socket)，基于 TCP，提供可靠的字节流传输，C 语言的实现方式为 `int s = socket (PF_INET, SOCK_STREAM, 2)`；数据报 Socket(Datagram Socket)，基于 UDP，提供不可靠的报文传输，C 语言的实现方式为 `int s = socket (PF_INET, SOCK_DGRAM, 3)`；Raw Socket，基于 IP，允许用户直接对 IP 操作，C 语言的实现方式为 `int s = socket (PF_INET, SOCK_RAW, protocol)`；4) 可靠交付报文 Socket，提供有确认的数据报功能，C 语言的实现方式为 `int s = socket (PF_NS, SOCK_RDM, 0)`；5) 有序包流 Socket，提供固定数据包长度的字节流功能，C 语言的实现方式为 `int s = socket (PF_NS, SOCK_SEQPACKET, 0)`；。

Socket 网络通信的过程可以与打电话类比。一次电话呼叫过程中，通话双方都有一个电话机；每个电话机有一个电话号码；被叫用户需要打开铃声，以便接听；主叫用户摘机、拨号；被叫用户的电话振铃，摘机接听；双方通话、交换数据；通话结束，双方挂机。而在网络应用通信中，通信双方各有一个通信端点 endpoint（电话机）；每个端点有一个地址，作为通信的唯一标识（电话号码）；一个端点（主叫用户）发起向另一个端点的通信（建立连接）；另一个端点（被叫用户）等待建立连接；连接建立后，双方开始交换数据（通话）；数据交换结束后，端点关闭（挂机）。

因此，可以得出 Socket 网络通信的系统调用有：Socket() 建立连接的端点，获得 Socket 描述符；Bind() 分配电话号码，也即 Server 绑定 Socket 地址（IP 地址+端口号）；Listen() 等待有电话呼入，也即 Server 等待 Client 连接；Connect() 拨号，也即 Client 连接到 Server；Accept() 接收电话呼叫，也即 Server 获得连接请求的 Client 的 Socket 地址；Send(), Recv() 通话、交换数据，也即在已建立的连接上发送/接收数据（TCP 方式）；sendto (), recvfrom () 无需连接，直接发送 接收数据（UDP 方式）；close()/shutdown() 挂机，关闭 Socket（双向/单向）；readn (), writen (), readline (), 读/写操作。

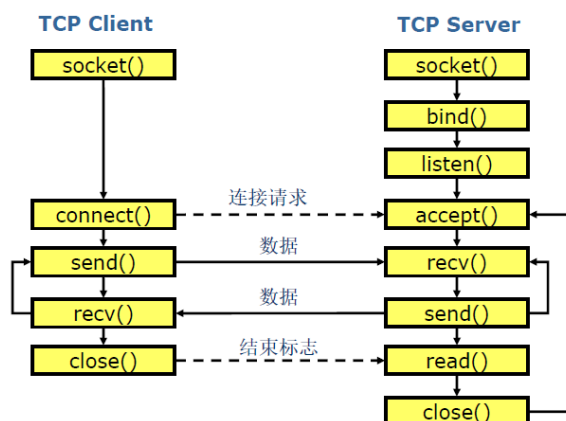


图 1 基于 TCP 的 Socket 网络通信系统调用流程

基于以上的系统调用，我们可以给出基于 TCP 的 Socket 网络通信系统调用流程，如上图所示。服务器端和客户端各自建立连接的端点后，服务器端绑定一个 Socket 地址后进入等待连接状态。当客户端发起连接请求，服务器端响应后建立起 TCP 连接。客户端发送请求，服务器端收到请求后把请求数据发送给客户端。所有通信内容结束后，双方关闭连接，四次挥手后结束通信。

Socket 网络通信中因特网特定地址(Internet specific socket address)与通用地址(Generic socket address)是两种不同的数据结构，如下图所示。许多 Socket 系统调用要求使用通用地址，例如： bind(), connect(), accept()等，则要进行强制转换，将 Internet specific socket address(sockaddr_in **) 转换成 generic socket address(sockaddr **).

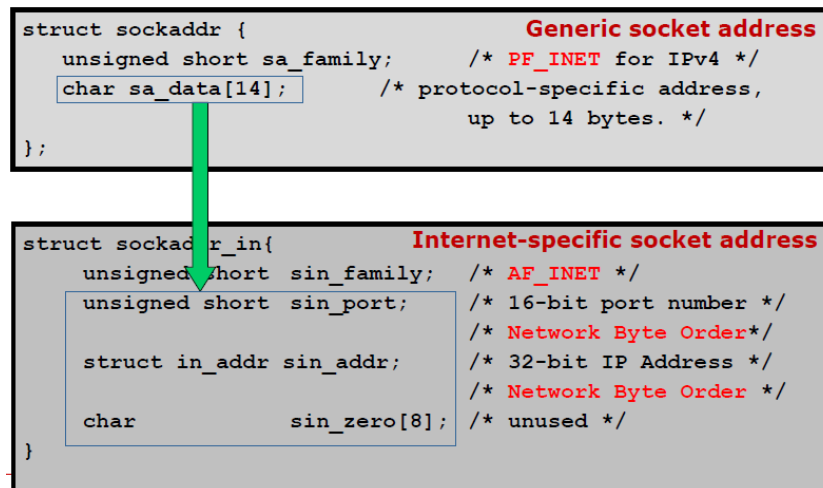


图 2 Socket 网络通信中因特网特定地址与通用地址的差异

Socket 网络通信还需要注意字节顺序问题。计算机对于多字节数据（如 IP 地址、端口号）的存储和发送有不同的顺序。在网络通信中，计算机采用网络字节顺序（NBO），是大端顺序（Big endian），首先传输低阶字节，高阶字节最后传输。在计算机存储中，计算机采用主机字节顺序（HBO），是主机特定的顺序。它既可以是小端顺序（Little endian），也即首先存储高阶字节；也可以是大端顺序（Big endian），首先存储低阶字节。由于存储和网络通信字节顺序可能不一致的问题，因此在 Socket 网络通信中，主机发送数据之前、接收数据之后，必须进行 HBO 与 NBO 之间的转换。相关的系统调用有：htonl()：将一个长整形数据从 HBO 转换给 NBO；htons()：将一个短整形数据从 HBO 转换给 NBO；ntohl()：将一个长整形数据从 NBO 转换给 HBO；ntohs()：将一个短整形数据从 NBO 转换给 HBO。

Socket 还可能要进行地址格式转换，实现 IP 地址的 ASCII 字符串格式与 NBO 格式之间的转换。相关系统调用有：inet_aton()：将 IP 地址从点分十进制数（ASCII 字符串）格式转换成 NBO 的无符号长整型格式；inet_addr()：将 IP 地址从点分十进制数（ASCII 字符串）格式转换成 NBO 的无符号长整型格式；inet_ntoa()：将 IP 地址从 NBO 的无符号长整型格式转换成点分十进制数（ASCII 字符串）格式。

其他常见的 Socket 系统调用还有：getsockopt()：查询 Socket 的相关信息；setsockopt()：设置 Socket 选项；gethostbyname()：使用域名从 DNS 数据库查询主机的信息（别名、IP 地址）；gethostbyaddr()：使用 IP 地址从 DNS 数据库查询主机的信息（域名）；gethostname()：查询主机的规范名（Canonical Name）；getservbyname()：查询指定服务的端口号；getservbyport()：查询指定端口号对应的服务名。

7.2.2 客户端

打开命令行，切换到要传输的文件目录下，输入待传输文件名、传输总字节数（传输成功时）；或者差错报告（传输错误时）。如下图所示，采用 C 语言开发的客户端在 Linux 系统环境下执行

命令后，从目标 IP 地址的服务器获取了 test.txt 文件，传输 32 字节后保存为 test.txt.bak，传输成功后退出。

```
student@BUPTIA:~/NP3/TCPFile$ ./tcpclt 127.0.0.1 test.txt
Connect to server: 127.0.0.1
file received
32 bytes received, and stored in test.txt.bak.
student@BUPTIA:~/NP3/TCPFile$
```

图 3 本实验中客户端程序运行效果示例

7.2.3 服务器端

打开命令行，切换到服务器程序目录下，运行服务器端程序，进入监听状态。接收到客户端建立连接请求后，建立连接，收到客户端要求传输的文件名后，将服务器端符合要求的文件发送给客户端。在此过程中，服务器端输出客户端的 IP 地址和端口号；发送的文件数据总字节数（发送成功时）；必要的差错报告（发送错误时）。如下图所示，采用 C 语言开发的客户端在 Linux 系统环境下执行命令后，接到了客户端发起的连接请求，建立连接后向客户端发送了 test.txt 文件，发送了 32 字节后传输完成，传输成功后关闭连接，继续监听。

```
student@BUPTIA:~/NP3/TCPFile$ ./tcpsvr
*****
Accept client 127.0.0.1 on TCP Port 34000
This client request for file name: test.txt
Entering file transfer...
End of the file
32 BYTES data have been sent
```

图 4 本实验中服务器端程序运行效果示例

传输完成后，可以把服务器端的 test.txt 与客户端的 test.txt.bak 进行比对，确认文件是否完全一致，验证 TCP 通信的可靠性。

7.2.4 SSL 安全连接通信

在 SSL 安全连接中，客户端和服务端会采用与非安全连接不同的端口进行通信。其过程如下：

客户端首先请求服务器标识自己的身份，服务器发送其 SSL 证书的副本（包括服务器的公钥），客户端检查证书合法后，使用服务器端的公钥创建会话密钥，然后加密并将其发回给服务器端。服务器端使用私钥解密获得会话密钥后，传回使用会话密钥加密的确认以开启加密会话。然后双方可以使用会话密钥加密所有被传输的数据。

7.3 撰写实验报告

按第 8 节的要求撰写实验报告。

8. 实验报告要求

本节描述了应提交实验报告的内容提纲和每项具体要求。实验完成后，应以电子版方式提交实验报告。提交的文件应包含报告（WORD 或 PDF）文档，命名为班级-学号 1-学号 2.DOC，附上源程序清单，打包压缩为班级-学号 1-学号 2.rar。

报告中应包含下列内容：

1. 实验封面，填写基本信息和实验内容，组员姓名、学号及分工
2. 实验内容和实验环境描述
描述本次实验的任务、内容和实验环境。
3. 软件设计
给出程序的数据结构、模块之间的调用关系和功能程序流程。
 - (1) 数据结构：数据结构是整个程序的要点之一，程序维护者充分了解数据结构就可以对主要算法和处理流程有个基本的理解。描述程序中自定义结构体中各成员的用途，定义的全局变量和主函数中的变量的变量名和变量所起的作用。
 - (2) 模块结构：给出程序中所设计的子程序所完成的功能，子程序每个参数的意义。给出子程序之间的程序调用关系图。
 - (3) 算法流程：画出流程图，描述算法的主要流程。
 - (4) 主要功能模块的实现要点
- 4 实验结果演示及分析
 - (1) 对于你所实现的功能，逐个进行测试，并将输出截图。
 - (2) 你的程序是否提供了差错处理功能？可以对哪些差错进行处理？
- 5 实验总结和心得体会
描述你在调试过程中都遇到了哪些问题和解决的过程。
 - (1) 完成本次实验的实际上机调试时间是多少？
 - (2) 描述你在调试过程中都遇到了哪些问题和解决的过程。
 - (3) 总结本次实验，你在 SOCKET 机制方面、协议软件设计方面、理论学习方面、软件工程方面等哪些方面上有所收获和提高？
 - (4) 本实验题目的设计和安排有哪些不足？
6. 源程序清单
按照附录 “源程序清单-样板.doc” 的格式整理你的 C 语言源程序，填写好姓名，学号等相关内容。

9. 程序验收过程及要求

约在 6 月 7 日（第十六周周六）左右安排现场验收，必须带学生证。每组带至少一台笔记本电脑，分别打开两个命令行窗口，一个运行服务器端程序，一个运行客户端程序。演示程序执行情况后，进行验收问答：由助教从一组人中随机选一个同学代表本组进行回答。