

华中科技大学  
网络空间安全学院

《计算机通信与网络》实验报告

姓 名 袁 也  
班 级 网安 1902 班  
学 号 U201911808  
联系方式 13810682605  
分 数

## 实验报告及代码和设计评分细则

评分项目		满分	得分	备注
文档格式		10		
感想（含思政）		5		
意见和建议		10		
Socket 编程	代码可读性	5		
	注释	5		
	软件体系结构	5		
	问题描述及解决方案	5		
局域网组网实验	设备选型合理性	5		
	遇到的问题描述及解决方案	5		
	对理论的理解	5		
	对工程的理解	5		
广域网组网实验 （路由器实验）	遇到的问题描述及解决方案	5		
	对理论的理解	5		
	对工程的理解	5		
综合组网实验	拓扑结构	5		
	设备选型	5		
	经济、环境、管理的考虑	5		
	对工程的理解	5		
总分				

## 综合得分

操作得分		实验报告得分	
综合得分			
教师签名		日期	

# 目 录

目 录.....	II
1 Socket 编程实验 .....	1
1.1 环境 .....	1
1.2 实验内容 .....	1
1.3 实验要求 .....	1
1.4 实验过程 .....	2
1.4.1 系统结构设计 .....	2
1.4.2 系统数据处理流程 .....	7
1.4.3 系统详细设计 .....	9
1.4.4 系统测试方案 .....	12
1.5 系统测试及结果说明 .....	12
1.6 遇到的问题及解决方法 .....	15
2 局域网组网实验 .....	17
2.1 环境 .....	17
2.2 实验要求 .....	17
2.3 遇到的困难及解决办法 .....	17
3 广域网组网实验（路由器配置） .....	19
3.1 环境 .....	19
3.2 实验要求 .....	19
3.3 遇到的困难及解决办法 .....	19
4 综合组网实验 .....	21
4.1 环境 .....	21
4.2 实验要求 .....	21
4.3 遇到的困难及解决办法 .....	21
5 感想与建议 .....	22
5.1 感想 .....	22
5.2 意见和建议 .....	23

# 1 Socket 编程实验

## 1.1 环境

操作系统：Windows/Linux

编程语言：C

## 1.2 实验内容

完成一个 TFTP 协议客户端程序，实现一下要求：

- 1) 严格按照 TFTP 协议与标准 TFTP 服务器通信；
  - 2) 能够将文件上传到 TFTP 服务器；
  - 3) 能够从 TFTP 服务器下载指定文件；
  - 4) 能够向用户展现文件操作的结果：文件传输成功/传输失败；
  - 5) 针对传输失败的文件，能够提示失败的具体原因；
  - 6) 能够显示文件上传与下载的吞吐量；
  - 7) 能够记录日志，对于用户操作、传输成功，传输失败，超时重传等行为记录日志；
  - 8) 人机交互友好（图形界面/命令行界面均可）；
  - 9) 能够实现两种不同的传输模式 netascii 和 octet；
- （8）和（9）为额外功能，将视具体情况予以一定加分。

## 1.3 实验要求

- 1) 必须基于 Socket 编程，不能直接借用任何现成的组件、封装的库等；
- 2) 提交实验设计报告和源代码；实验设计报告必须包括程序流程图，源代码必须加详细注释。
- 3) 实验设计报告需提交纸质档和电子档，源代码、编译说明需提交电子档。
- 4) 基于自己的实验设计报告，通过实验课的上机试验，将源代码编译成功，运行演示给实验指导教师检查。

1.4 实验过程

1.4.1 系统结构设计

1. 模块框图

基于 TFTP 协议的可靠文件传输系统整体模块框图如图 1-1 所示：

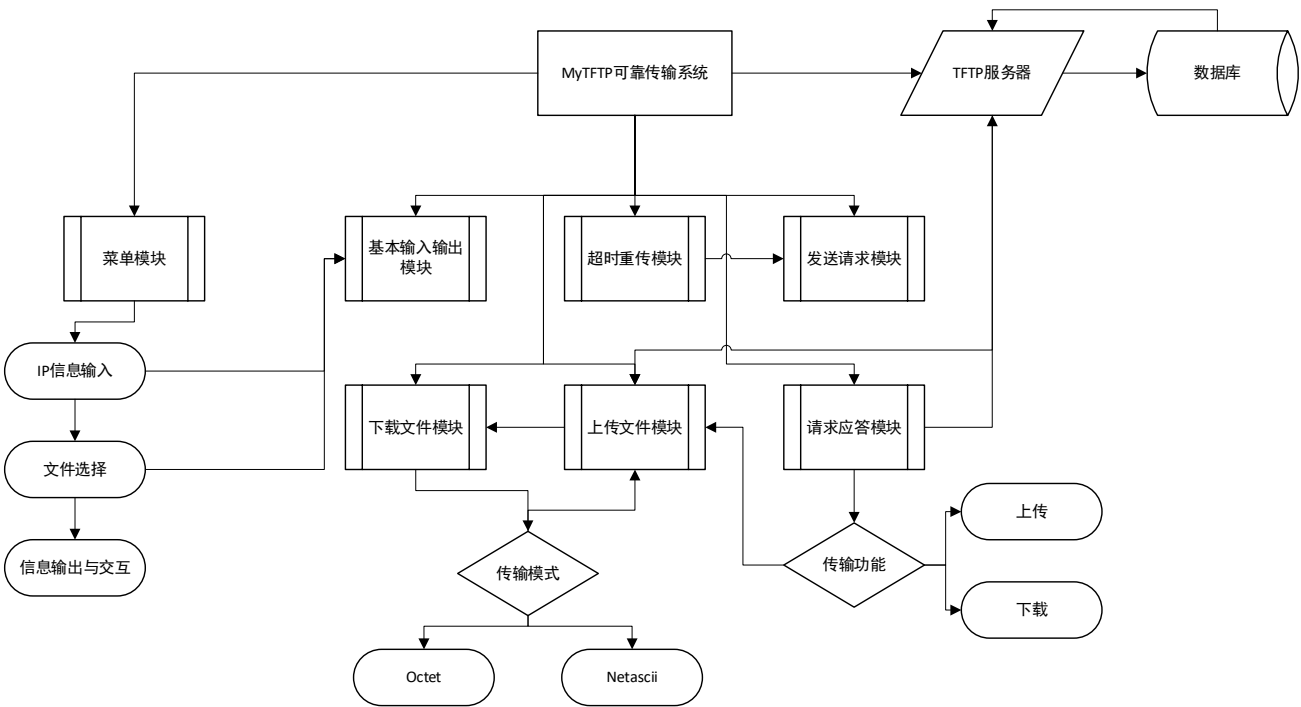


图 1-1 系统整体模块框图

如图 1-1，TFTP 客户端程序主要由菜单模块、基本输入输出模块、超时重传模块、发送请求模块、下载文件模块、上传文件模块以及请求应答模块这七大模块构成，以及错误处理模块及日志记录模块提供更加可靠的系统维护界面。

下面，将依次介绍这些模块的功能。

2. 菜单模块

菜单模块主要实现与用户的交互界面，并引导用户输入目的 IP 和端口等通信必要信息。

如图 1-2 所示，菜单界面可以显示的内容大致如下：

```
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

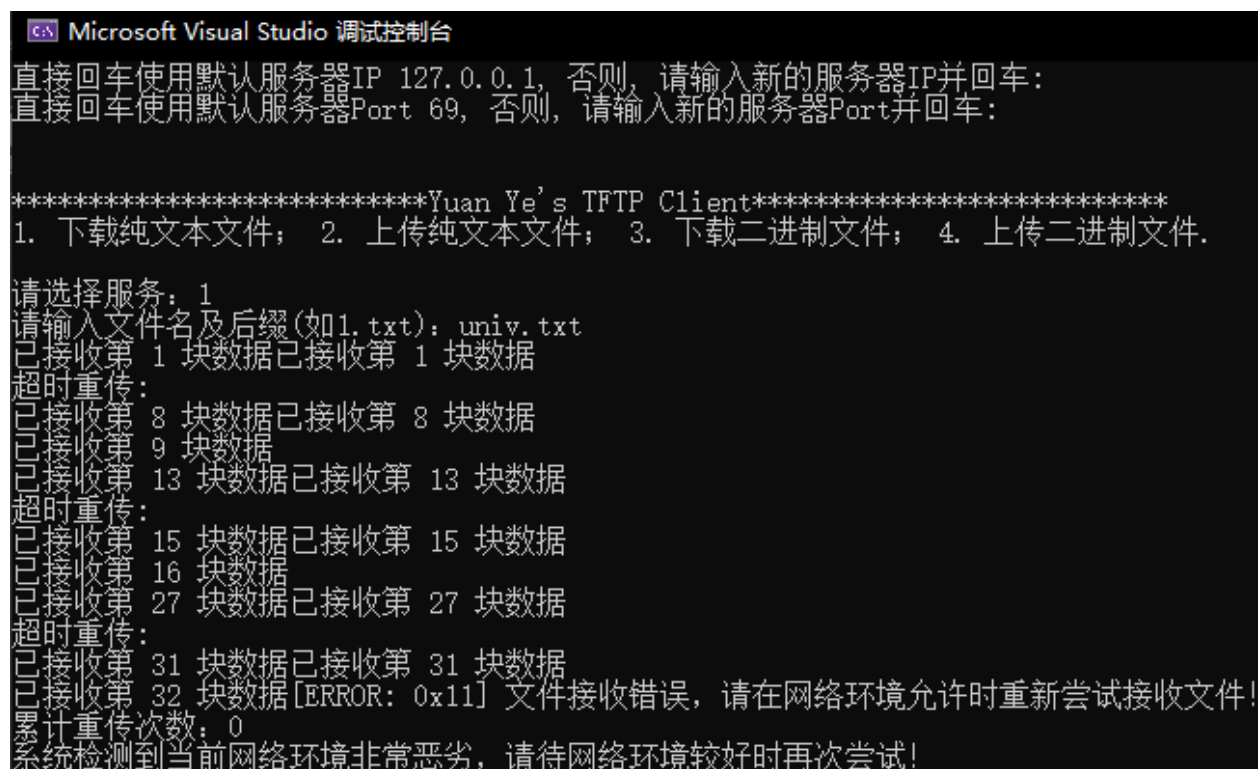
请选择服务: 1
请输入文件名及后缀(如1.txt): 1.txt
[WARNING: 0x90] 连接超时, 正在尝试重新连接服务器
```

图 1-2 菜单界面

用户可以通过菜单界面输入命令、选项、服务器 IP 地址、服务器端口号、文件名称等内容，同时菜单界面还可以根据错误检测模块发来的信号，及时将错误输出在交互界面中。菜单界面在 main 函数中实现，其中菜单的打印由单独函数实现。菜单界面将循环执行，直到用户输入 0 才会结束程序。

### 3. 基本输入输出模块

基本输入输出模块输入用户的请求模式、文件类型以及功能选择等内容，向用户输出当前传输状态、传输速率（实时和整体）、整体丢包率、整体网络情况等内容。此部分整合在菜单模块当中实现，基本输入输出的内容如图 1-3 所示。



```
Microsoft Visual Studio 调试控制台
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 1
请输入文件名及后缀(如1.txt): univ.txt
已接收第 1 块数据已接收第 1 块数据
超时重传:
已接收第 8 块数据已接收第 8 块数据
已接收第 9 块数据
已接收第 13 块数据已接收第 13 块数据
超时重传:
已接收第 15 块数据已接收第 15 块数据
已接收第 16 块数据
已接收第 27 块数据已接收第 27 块数据
超时重传:
已接收第 31 块数据已接收第 31 块数据
已接收第 32 块数据[ERROR: 0x11] 文件接收错误, 请在网络环境允许时重新尝试接收文件!
累计重传次数: 0
系统检测到当前网络环境非常恶劣, 请待网络环境较好时再次尝试!
```

图 1-3 基本输入输出模块显示内容

### 4. 超时重传模块

在最大重发次数之内，程序会一直尝试重新发送包，直到发送成功为止，若成功则返回发送成功的字节数，若失败则返回-1。

在程序的头部，可根据信息传输的需求自己定义超时时间以及超时最大重发次数。默认值为：超时最大重发次数=6，超时时间=2(秒)。

### 5. 发送请求模块

客户端通过发送请求告知服务器接下来的操作：根据 op 字段的值确定是上传文件还是下载文件，通过 filename 字段告知文件名称，通过 type 判断文件的类型是文本文件(netascii)

还是二进制(octet)文件。构建的请求文件数据报文格式如图 1-4 所示。

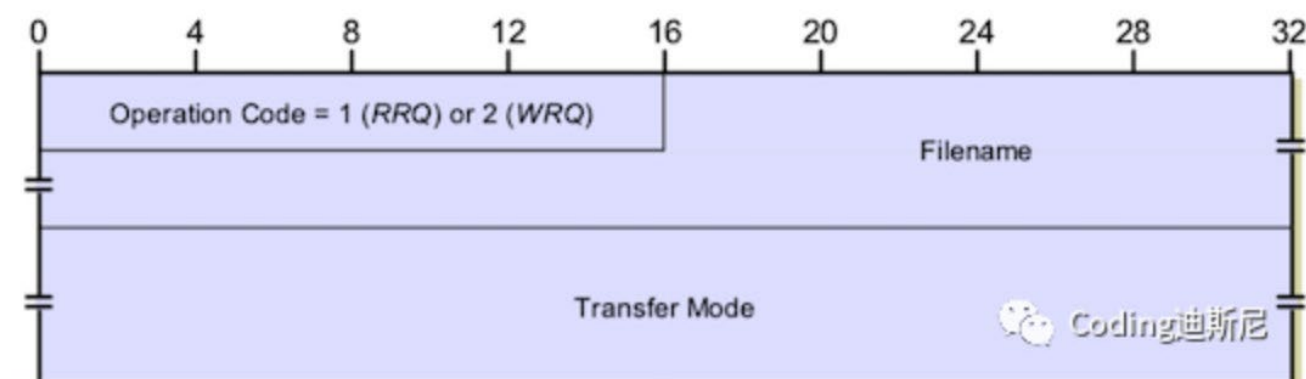


图 1-4 请求数据报文格式

文件请求包的前两个字节为操作码 op，文件名为变长字段，以\0 结尾，接着就是文件类型：使用字符串文件，MODE 字段使用“netascii”，传输二进制文件，MODE 字段使用“octet”，均以‘\0’结尾。

#### 6. 下载文件模块

服务器通过请求文件包确定下载请求后，向客户端发送传输数据块的 DATA 数据包，客户端开辟一个数组接收数据，并记录数据块的序号。根据数据块的编号判断数据传输过程是否有误，若编号失序则向服务器发送上一个未失序的数据块编号的应答请求，否则，向服务器发送接受的数据包的数据块编号的应答请求。传输数据块的 DATA 数据报文如图 1-5 所示。

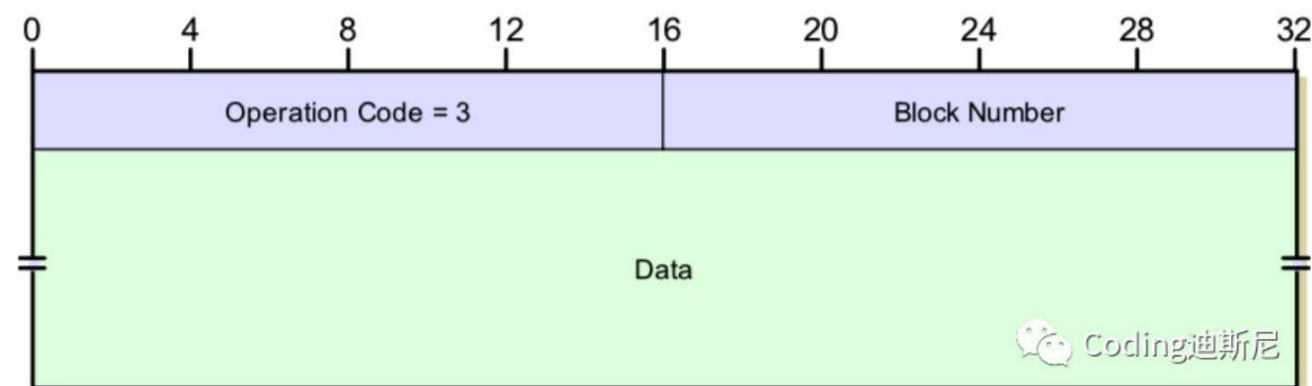


图 1-5 TFTP 协议 DATA 数据包格式

如图 1-5 所示的格式那样，操作码 op 固定为 3，占两个字节，数据块编号占两个字节，最多有 65535 个数据块（当然还可以从 0 重新开始编号，这样就不会有最大传输文件大小的限制了），Data 段最多 512 字节。

#### 7. 上传文件模块

文件上传模块提供上传任意类型的本地文件到 TFTP64 服务器的功能。服务器通过请求文

件包确定下载请求后，客户端向服务器发送传输数据块的 DATA 数据包，客户端开辟一个数组 `recvbuf` 接收应答数据包，以及另外的一个数组 `sendbuf` 传入文件的数据，并记录数据块的序号。向服务器传送数据之后接受应答数据包，根据 `recvbuf` 接受的数据判断是否接收成功，数据块是否失序等。如果接收失败，则调用错误处理模块进行处理，编号是否失序的处理方式同下载文件模块。传输数据块的 DATA 数据包同下载文件模块的实现。

#### 8. 请求应答模块

接收到数据包之后，客户端向服务器发送应答，说明这次接收文件的数据块编号，以便服务器能够正确判断数据包的传输情况。应答报文格式如图 1-6 所示：

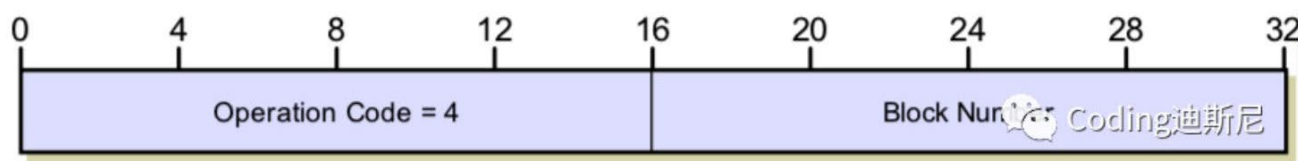


图 1-6 TFTP 协议 ACK 数据包格式

如图 1-6 显示的那样，应答报文长度固定为 4 个字节，其中前两个字节代表操作码，固定为 4，后两个字节为数据块编号，编号范围为 0~65535。

#### 9. 错误处理模块

错误处理模块提供了客户端与服务器之间的错误通信机制，其报文格式如图 1-7 所示。

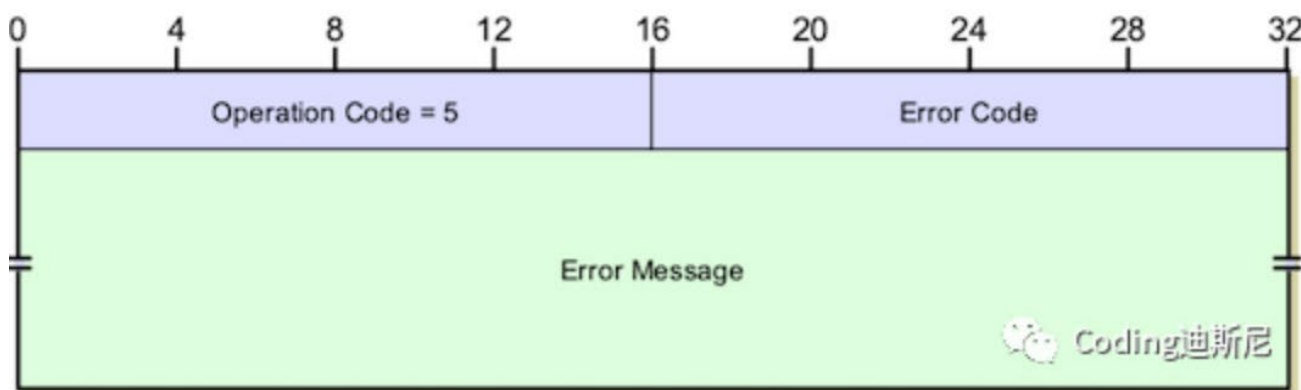


图 1-7 TFTP 协议 ERROR 数据包格式

就像图 1-7 所示的报文格式那样，错误报文长度不固定，主要是因为携带了具体的错误信息字符串。前两个字节为 `op`，固定为 5，第三个以及第四个字节为错误编号，具体的定义详见下方的表 1-1。后面则为长度不固定的具体错误信息字符串。

表 1-1 TFTP 协议错误码信息



错误码	表示的意思
0	未定义 Not defined, see error message
1	文件未找到 File not found
2	访问被拒绝 Access violation
3	磁盘满或超出可分配空间 Disk full or allocation exceeded
4	非法的 TFTP 操作 Illegal TFTP operation
5	未知的传输 ID Unknown transfer ID
6	文件已经存在 File already exists
7	没有该用户 No such user

#### 10. 本地日志记录模块

本地日志记录模块记录的内容如图 1-8 所示。

```

1  [2021/11/23 17:30:54]请求上传数据:univ.txt
2  [2021/11/23 17:31:28][ SUCCESS! ] 文件传输成功!
3  [2021/11/23 17:32:57]请求上传数据:p.pdf
4  [2021/11/23 17:33:08][ SUCCESS! ] 文件传输成功!
5  [2021/11/23 17:33:37]请求下载数据:univ.txt
6  [2021/11/23 17:33:37][2021/11/23 17:33:37][2021/11/
7  [2021/11/23 17:34:39]请求下载数据:univ.txt
8  [2021/11/23 17:34:39][2021/11/23 17:34:39][2021/11/
9  [2021/11/23 17:36:57]请求上传数据:univ.txt
10 [2021/11/23 17:37:02][ SUCCESS! ] 文件传输成功!
11 [2021/11/23 17:37:31]请求下载数据:univ.txt
12 [2021/11/23 17:37:31][2021/11/23 17:37:31][2021/11/
13 [2021/11/23 17:38:07]请求下载数据:pic.png
14 [2021/11/23 17:38:10][ SUCCESS! ] 文件传输成功!
15 [2021/11/23 17:38:52]请求上传数据:p.pdf
16 [2021/11/23 17:39:01][ SUCCESS! ] 文件传输成功!
17 [2021/11/23 18:34:09]请求上传数据:univ.txt
18 [2021/11/23 18:34:12][ SUCCESS! ] 文件传输成功!
19 [2021/11/23 18:34:28]请求上传数据:univ.txt
20 [2021/11/23 18:34:36][ SUCCESS! ] 文件传输成功!
21 [2022/01/04 13:12:18]请求下载数据:1.txt
22 [2022/01/04 13:12:18]请求下载数据:1.txt
23 [2022/01/04 13:12:18]请求下载数据:1.txt
24 [2022/01/04 13:12:18]请求下载数据:1.txt
25 [2022/01/04 13:12:18]请求下载数据:1.txt
26 [2022/01/04 13:12:18]请求下载数据:1.txt
27 [2022/01/04 13:12:18][ERROR: 0x10] 无法连接到服务器,

```

图 1-8 日志记录

1.4.2 系统数据处理流程

系统中的全部数据在系统运行全程的流向图如图 1-9 所示。

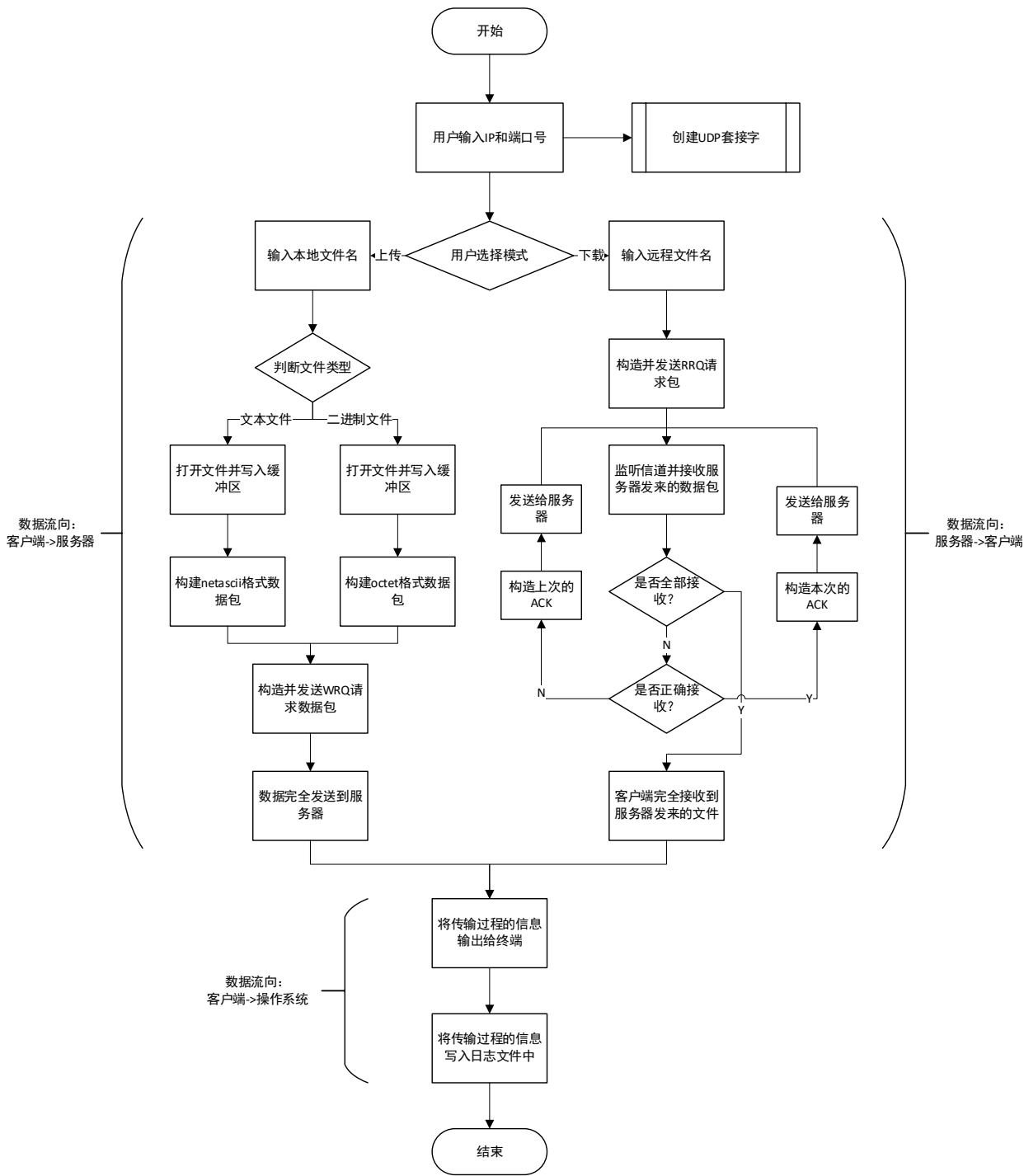


图 1-9 数据流向图

1. 上传文件数据流向

首先，终端接收用户输入的服务器 IP 以及端口等信息，创建 UDP 套接字并初始化

socket 服务，设置网络为非阻塞模式，bind 绑定。然后客户端在本地路径下查找相应的文件，并将文件内容读取到缓冲区中。然后构造并发送 WRQ 数据包，等待服务器发来的 ACK 数据包，如果在超时时间内未收到，则重发数据包。重发如果超过三次，则报错并终止本次上传操作。如果收到了 ACK 数据包，则进入下一步。接下来，将缓冲区中的内容分块构造 DATA 数据包，每个数据包同样有三次发送的机会，如果发送失败则报错并终止。直到将缓冲区中缓存的全部文件都发送出去，停止发送。最终将发送过程中的信息输出到终端，并写入日志文件中。具体如图 1-10 所示。

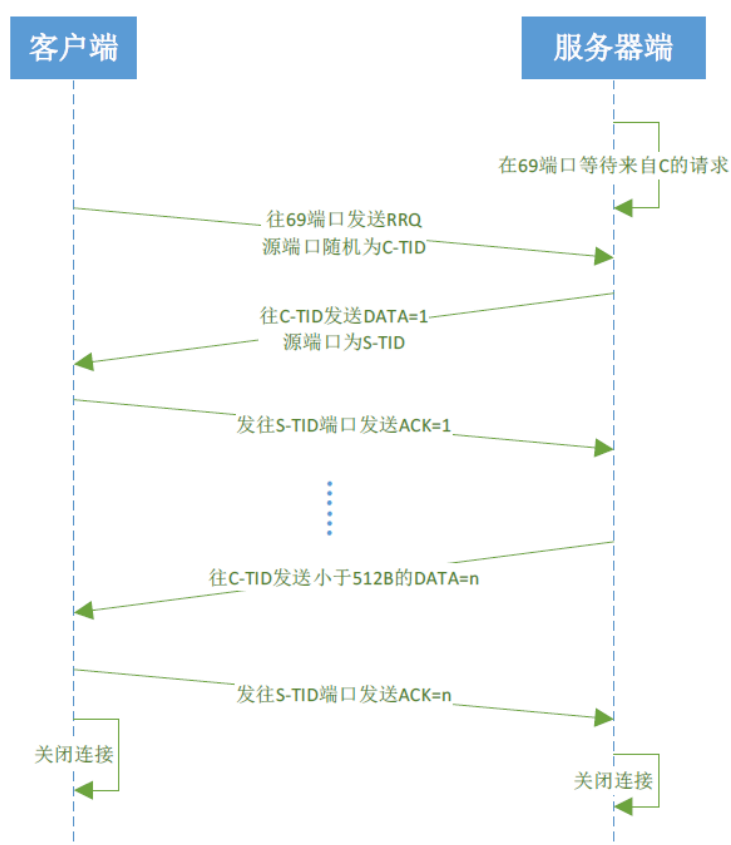


图 1-10 典型的 RRQ 工作流程

## 2. 下载文件数据流向

首先，终端接收用户输入的服务器 IP 以及端口等信息，创建 UDP 套接字并初始化 socket 服务，设置网络为非阻塞模式，bind 绑定。然后构造并发送 RRQ 数据包，等待服务器发来的 ACK 数据包，如果在超时时间内未收到，则重发数据包。重发如果超过三次，则报错并终止本次下载操作。如果收到了 ACK 数据包，则进入下一步。接下来，将缓冲区中的内容分块构造 DATA 数据包，每个数据包同样有三次发送的机会，如果发送失败则报错并终止。直到将缓冲区中缓存的全部文件都发送出去，停止发送。最终

将发送过程中的信息输出到终端，并写入日志文件中。具体如图 1-11 所示。

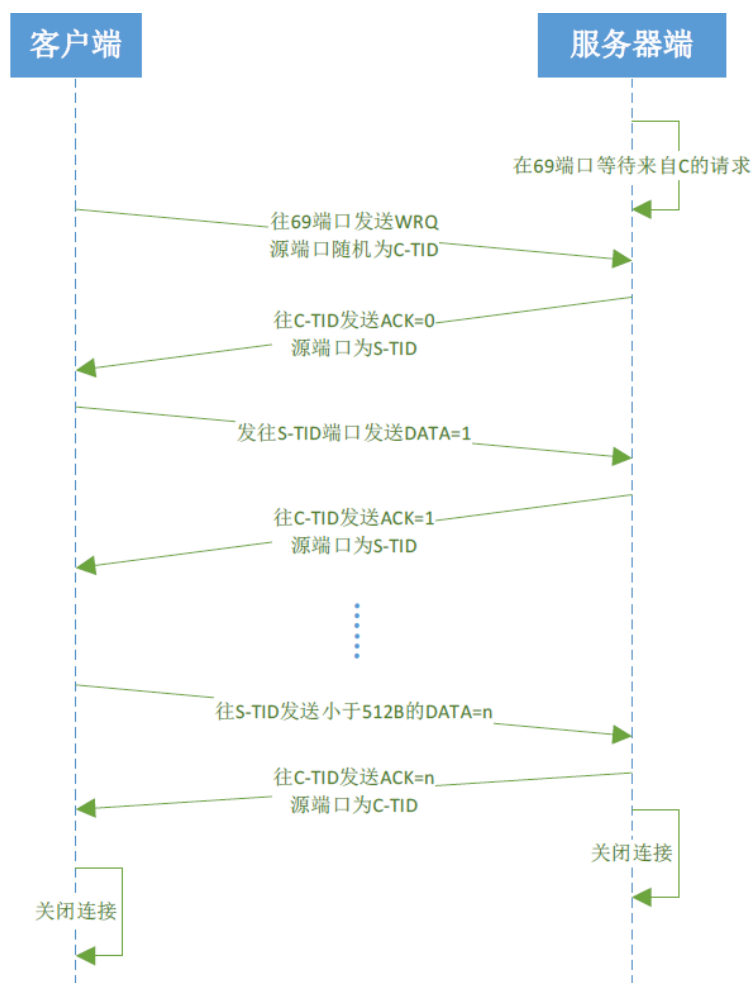


图 1-11 典型的 WRQ 工作流程

### 1.4.3 系统详细设计

#### 1. 上传文件功能的详细设计

上传文件功能的详细设计流程图如图 1-12 所示。

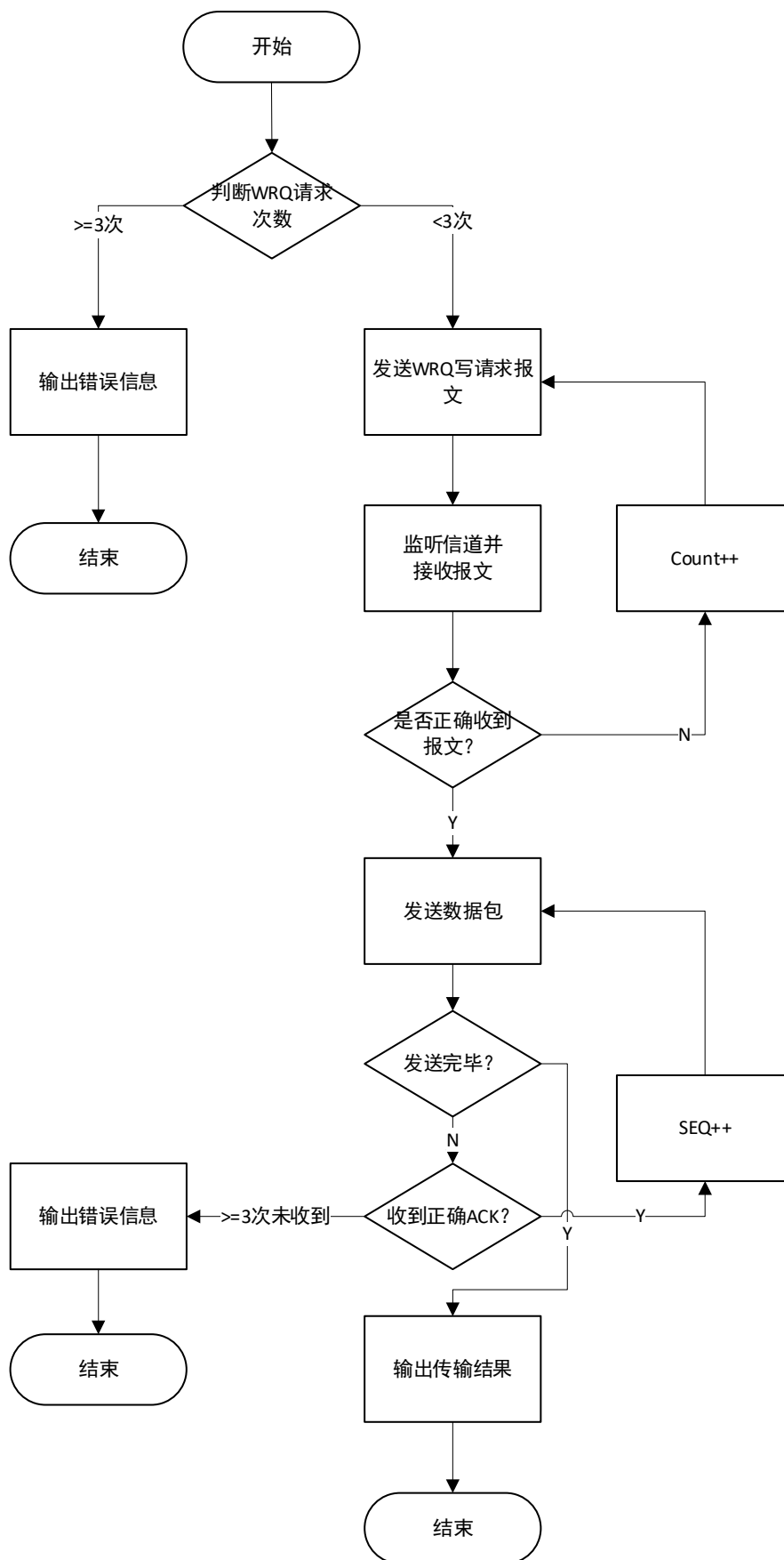


图 1-12 上传模块框图

## 2. 下载文件功能的详细设计

下载文件功能的详细设计流程图如图 1-13 所示。

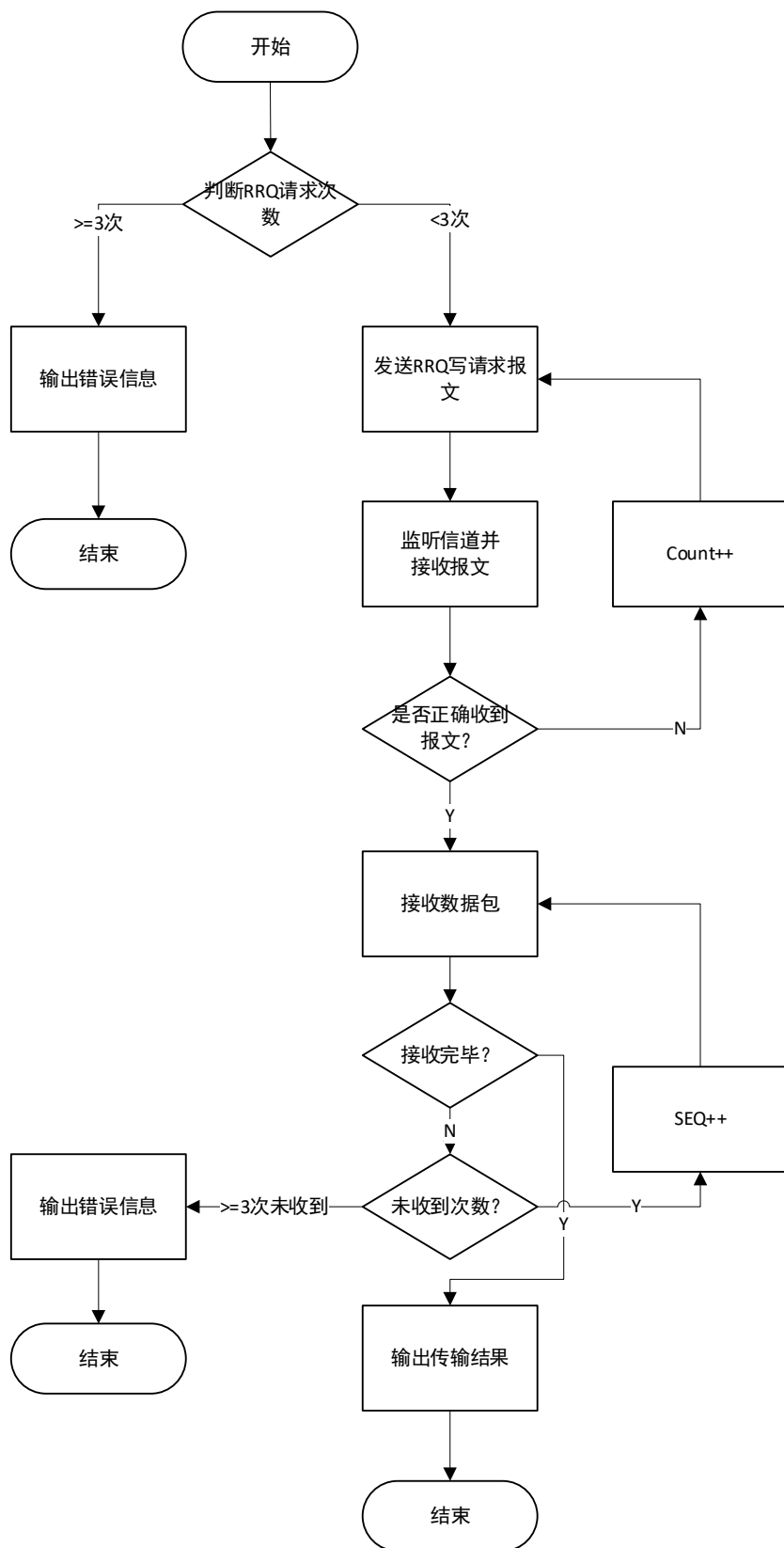


图 1-13 下载模块框图

#### 1.4.4 系统测试方案

##### 1. 硬件环境

Windows 10 操作系统(19042.1415)

CPU: Intel(R) Core(TM) i7-10875H CPU 2.30GHZ

##### 2. 软件环境:

Socket 版本: 0x0101

TFTP 服务器: TFTP64 v4.64

真实网络环境模拟: clumsy v0.3

客户机和服务器运行在同一台物理机上

服务器 IP 地址: 127.0.0.1

服务器端口号: 69

##### 3. 测试用例设置

1) 测试用例一: 使用错误输入, 包括操作码, 文件类型, 文件名等错误。

预期结果: 退出程序。

2) 测试用例二: 下载文本文件和 exe 文件, 分别以文本形式和二进制形式下载, 文件大小为 514KB 和 1MB。

预期结果: 成功下载文件并可以正常打开。

3) 测试用例三: 上传文本文件和 exe 文件, 分别以文本形式和二进制形式上传, 文件大小为 302KB 和 2130KB。

预期结果: 成功上传文件并可以正常打开。

4) 测试用例四: 在不安全的信道上传文件, 丢包率为 10%。

预期结果: 丢失的数据包重传。

#### 1.5 系统测试及结果说明

1. 输入错误的 IP 和端口号, 测试报错, 结果如图 1-14 所示。

```

直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:127.114.5.14
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:19

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 1
请输入文件名及后缀(如1.txt): 1.a
[WARNING: 0x90] 连接超时, 正在尝试重新连接服务器
[WARNING: 0x90] 连接超时, 正在尝试重新连接服务器
[WARNING: 0x90] 连接超时, 正在尝试重新连接服务器
[WARNING: 0x90] 连接超时, 正在尝试重新连接服务器
[WARNING: 0x90] 连接超时, 正在尝试重新连接服务器
[ERROR: 0x10] 无法连接到服务器, 请检查IP及端口号是否正确设置。这也有可能是因为你错误地设置了防火墙策略。
累计重传次数: 0
系统检测到当前网络环境非常恶劣, 请待网络环境较好时再次尝试!

```

图 1-14 测试报错

2. 输入服务器不存在的文件名, 结果如图 1-15 所示。

```

直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 1
请输入文件名及后缀(如1.txt): hhh.txt
传输失败: [ERROR: 1] 文件未找到: File not found

```

图 1-15 未找到相应文件

3. 下载 514KB 文本文件, 结果如图 1-16 所示:

```

Microsoft Visual Studio 调试控制台
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 1
请输入文件名及后缀(如1.txt): univ.txt
已接收第 44 块数据[ SUCCESS! ] 文件传输成功!
平均传输速率: 2.85053MB/s
累计重传次数: 0

```

图 1-16 下载 514KB 文本文件

4. 下载 1MB 文本文件, 结果如图 1-17 所示:



```
Microsoft Visual Studio 调试控制台
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 1
请输入文件名及后缀(如1.txt): 1MB.txt
已接收第 2049 块数据[ SUCCESS! ] 文件传输成功!
平均传输速率: 7.21807MB/s
累计重传次数: 0
```

图 1-17 下载 1MB 大小文本文件

5. 下载二进制文件, 结果如图 1-18 所示:

```
Microsoft Visual Studio 调试控制台
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 3
请输入文件名及后缀(如1.txt): pic.png
已接收第 40 块数据[ SUCCESS! ] 文件传输成功!
平均传输速率: 5.20833MB/s
累计重传次数: 0
```

图 1-18 下载二进制文件

6. 上传文本文件, 结果如图 1-19 所示:

```
Microsoft Visual Studio 调试控制台
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 2
请输入文件名及后缀(如1.txt): 5MB.txt
[ SUCCESS! ] 文件传输成功! 传输... 传输进度 [100.00 %] (5.00 MB / 5.00 MB).....
平均传输速率: 8.07754MB/s
累计重传次数: 0
信道嘈杂度: 0.0000
```

图 1-19 上传文本文件

7. 上传二进制文件, 结果如图 1-20 所示:

```
Microsoft Visual Studio 调试控制台
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 4
请输入文件名及后缀(如1.txt): p.pdf
[ SUCCESS! ] 文件传输成功! 传输进度 [101.47 %] (0.03 MB / 0.03 MB).....
平均传输速率: 0.297477MB/s
累计重传次数: 0
信道嘈杂度: 0.0000
```

图 1-20 上传二进制文件

8. 在不可靠信道上进行嘈杂环境鲁棒性测试, 设置丢包率为 10%, 延迟为 1ms, 测试下载情况, 如图 1-21 所示:

```
Microsoft Visual Studio 调试控制台
直接回车使用默认服务器IP 127.0.0.1, 否则, 请输入新的服务器IP并回车:
直接回车使用默认服务器Port 69, 否则, 请输入新的服务器Port并回车:

*****Yuan Ye's TFTP Client*****
1. 下载纯文本文件; 2. 上传纯文本文件; 3. 下载二进制文件; 4. 上传二进制文件.

请选择服务: 1
请输入文件名及后缀(如1.txt): univ.txt
已接收第 20 块数据已接收第 20 块数据
超时重传:
已接收第 21 块数据已接收第 21 块数据
超时重传:
已接收第 25 块数据已接收第 25 块数据
超时重传:已接收第 25 块数据
超时重传:已接收第 25 块数据
超时重传:已接收第 25 块数据
已接收第 26 块数据
已接收第 44 块数据[ SUCCESS! ] 文件传输成功!
平均传输速率: 0.000930114MB/s
```

图 1-21 不可靠信道上的可靠传输测试

## 1.6 遇到的问题及解决方法

问题 1: sendto 和 recvfrom 的参数设置错误

解决方法: 注意参数的填写顺序, 区分客户端和服务端谁是发送方谁是接收方, 根据状态机检查代码错误, 根据数据交换流程图判断是否缺少了步骤。

问题 2: socket 头文件的许多 API 都没有用过

解决方法：查阅文档和其他网上的相关资料。

## 2 局域网组网实验

### 2.1 环境

设备名称	设备型号	默认用户名	密码
无线 AP	华为 AP4050DN	Admin	admin@huawei.com
路由器	华为 AR1220C	Admin	Admin@huawei
三层交换机	-华为 S5720-28X-LI-AC	admin	admin@huawei.com
二层交换机	-华为 S5720-28P-LI-AC	admin	admin@huawei.com
	24 端口网络配线架		

### 2.2 实验要求

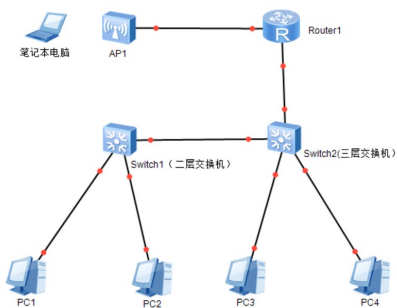


图 2.1

利用华为真实网络设备，完成图 2.1 所示的网络配置，实现如下组网要求：

- 1) PC1 与 PC3 在同一个 VLAN（VLAN2）中，能互相访问；
- 2) PC2 与 PC4 在同一个 VLAN（VLAN3）中，能互相访问；
- 3) 通过配置交换机和路由器，使得 PC1、PC2、PC3、PC4 可以相互访问；
- 4) 笔记本电脑通过 AP 无线上网，跟 PC1、PC2、PC3、PC4 之间互通。

### 2.3 遇到的困难及解决办法

问题 1：在制作双绞线的过程中，导线难以穿入水晶头的定位孔当中；

解决办法 1：将导线端头处用剪刀剪成一个斜面，使得每根导线都能依次穿入水晶头当中。

问题 2：eNSP 软件在安装后无法正常使用；

解决办法 2：卸载原来的 wireshark，重新安装配套版本的 wireshark。猜测是由于 eNSP 只能支持较古老版本的 wireshark，因此无法正常调用电脑中新版 wireshark 的 API，导致软件无法正常运行。

问题 3：eNSP 软件中模拟的 AP 为 fit 模式 AP，无法对其直接进行管理配置；

解决办法 3：在连接 AP 的接入交换机上再连接一个 AC，由此 AC 对 AP 进行管理和控制。在配置时，也只需要对 AC 进行配置即可。值得注意的是，在实物实验中，机柜中安装的 AP 为 fat 模式下的 AP，因此区别于虚拟仿真实验，在实物实验中我们无需额外添加 AC，可以直接对其进行管理与配置。

问题 4：实物实验过程中，由于 AP 的管理密码被更改，无法登录 AP 管理界面；

解决办法 4：用圆珠笔尖等较细的物品按住 AP 上的重置按钮超过 10 秒，可以在串口输出界面看到重置的提示信息，之后可以再 SecureCRT 中输入初始用户名和密码登录。

问题 5：实物实验中，用串口转 RJ-45 线连接电脑与设备上的 console 接口之后，在 secureCRT 软件中按照流程添加串口信息，但无法连接到设备；

解决办法 5：在更换连接接口及更换线缆重新测试后，发现问题是因为我们组一开始的线是坏的，更换新的线缆后就可以正常连接了。

## 3 广域网组网实验（路由器配置）

### 3.1 环境

华为虚拟仿真平台 eNSP。

### 3.2 实验要求

路由器实验

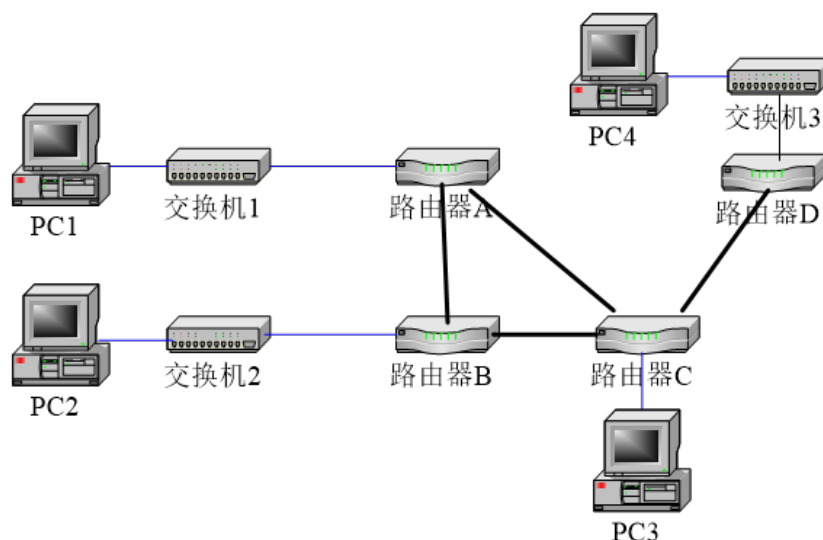


图 3.1

根据图 3.1 的拓扑结构进行组网，要求如下：

- 1) PC1: 192.168.1.0/24 网段; PC2: 192.168.2.0/24 网段; PC3: 192.168.3.0/24 网段;  
PC4: 192.168.4.0/24 网段
- 2) 路由器上配置 RIP 协议，使各 PC 机能互相访问
- 3) 路由器上配置 OSPF 协议，使各 PC 机能互相访问
- 4) 路由器上配置 RIP/OSPF 协议，使各 PC 机能互相访问之后，再对对路由器 1 进行访问控制配置，使得 PC1 无法访问其它 PC，也不能被其它 PC 机访问
- 5) 路由器上配置 RIP/OSPF 协议，使各 PC 机能互相访问之后，再对对路由器 1 进行访问、

### 3.3 遇到的困难及解决办法

问题 1：路由器配置界面上不断出现#####

解决方法：表示路由器正在启动过程中，等待一段时间后即可以正常配置。等待的时间可

能比较久，需要耐心等待。

问题 2：各设备的配置在重新打开 ensp 拓扑后消失不见。

解决方法：保存拓扑图并不能保存设备配置信息，需要单独导出各设备的配置文件。

## 4 综合组网实验

### 4.1 环境

华为虚拟仿真平台 eNSP。

### 4.2 实验要求

某学校申请了一个前缀为 211.69.4.0/22 的地址块，准备将整个学校连入网络。该学校有 4 个学院，1 个图书馆，3 个学生宿舍。每个学院有 20 台主机，图书馆有 100 台主机，每个学生宿舍拥有 200 台主机。对这些主机进行组网，要求：

- 1) 图书馆能够无线上网（选做，AP+AC 配合使用）；
- 2) 学院之间可以相互访问；
- 3) 学生宿舍之间可以相互访问；
- 4) 学院和学生宿舍之间不能相互访问；
- 5) 学院和学生宿舍皆可访问图书馆。

过程要求：

- 1) 网络拓扑结构的设计并在仿真软件上进行绘制；
- 2) 要求具有足够但最少的设备，不需要考虑设备冗余备份的问题；
- 3) 对全网的 IP 地址进行合理的分配；
- 4) 在绘制的网络拓扑结构图上对各类设备进行配置；
- 5) 测试是否满足组网需求，如有无法满足之处，请结合理论给出解释和说明。

### 4.3 遇到的困难及解决办法

问题 1：指导手册中设备列表中给出的设备型号在 ensp 中找不到对应的设备。

解决方法：在经过询问老师后，确认可以选择功能类似、端口数相同的设备进行代替，无需严格按照设备列表中的设备型号绘制拓扑图。

问题 2：实验指导手册中的网络拓扑结构描述并不明确，有些地方需要根据自己的想法进行设计。

解决方法：根据自己的想法，参考实际情况给出设计，并且要尽可能追求性价比和网络性能。



## 5 感想与建议

### 5.1 感想

计网实验是我上大学以来做过的最有趣的一门实验课，在这门课中，不仅有代码层面的理论实践，还有能动手亲自操作的实物实验。这在所有实验课中是绝无仅有的。

第一个实验中，我们利用 C++ 的 `socket` 库基于不可靠的运输层协议 UDP 实现了一个在应用层可靠的 TFTP 文件传输客户端。在实验过程中，主要学习了 `socket` 库的 API，练习了非阻塞模式下的网络编程，对于整个 TFTP 协议以及多线程的编程方式有了更加深入的理解。

在实验过程中，有一些小插曲。其中一个就是由于硬盘损坏，我第一遍写的代码全部丢失了，不得不重新写一遍代码。不过塞翁失马，焉知非福。在第二编写代码的时候，我重新梳理了自己的逻辑，不但发现了之前代码逻辑中的一些漏洞，还将代码中的冗余部分进行了合并，精简了整个系统，使得系统能够更高效的运行。此外，我还因为此次事件养成了及时将自己的代码同步到 `github` 上的好习惯，一方面能够通过开源来督促自己的开发进程，另一方面也是一个很好的版本控制系统。利用 `GitHub` 不但能帮助我及时备份重要的代码和文件，还能记录整个项目的版本变化过程，清晰开发思路。

本次编程的难点在于，需要对 `socket` 库中的函数有清晰的认识，以及对 TFTP 传输协议要了解清楚。一方面，需要熟练掌握 `socket` 库中的函数和 API，在遇到某个需求是需要很快地反映出来可以用哪个函数 `huoASPI` 来解决这个问题，这个其实也可以借助开发手册以及互联网资料。另一方面，也需要我们认真阅读实验指导手册，仔细理解清楚 TFTP 协议上传和下载文件的每一步都进行了什么操作、是谁想谁发报文、发的是什么报文、具体格式是什么样子的、遇到非预期情况怎么处置、怎么做特例判断等，这些细节都需要考虑清楚。我也认识到了，如果只是把 TFTP 协议看一遍，过一遍任务指导手册是远远不够的，就像我们平时课内学习，虽然把书本上的知识搞明白并不是什么难事，但是真正想把某个概念在代码层面上复现出来，还需要考虑很多细节。因此，在阅读实验指导手册时，在平时学习理论知识时，不但要看明白书上的内容，还需要多问自己问题。

第二个实验主要是动手操作实验，整体来说还是相当有趣的。首先，我自己制作了一根双绞线，并通过了连通性测试，可以说是非常有成就感的。在这个过程中，我掌握了 IEEE T-568B 线序——白橙—橙—白绿—蓝—白蓝—绿—白棕—棕，练习了剪线及处理水晶头的方法，收获颇丰。

之后练习使用 eNSP 进行虚拟环境下的仿真实验，实践了在同一交换机下静态配置不同 VLAN、在不同交换机下配置相同 VLAN 等操作，对理论课上学到的内容进行了实践。但由于我们是静态配置，因此需要对每一台设备进行很多操作，这点在模拟环境下，可以简单地来模拟小网络中的通讯配置，但是在实际当中会带来很大不便，因此实际运用中通常采用动态配置的方法。

最后，我们在实物上对刚刚在虚拟仿真实验环境中配置的 VLAN 网络进行了复现，尽管我们之前已经能够熟练地在 eNSP 中练习配置，但是当操纵实际设备时，还是遇到了不少的问题。比如在用线缆连接电脑与设备的 console 口，对设备进行实际配置时，我们经过多次尝试，发现电脑上的 SecureCRT 进程依然无法正常与设备之间进行通信，后来我们才发现是由于串口转 Rj-45 端口线缆出现故障，导致设备与电脑之间无连接建立。在更换线缆后，我们成功地与设备建立了联络，最终完成了对设备的配置。这一点也告诉我们一个道理：就算在虚拟仿真环境中验证了方案理论的正确性，当实际运用中真正进行配置时，依然有可能出现各种各样的问题，其中核心的问题在于，实际环境中的设备都不是理想状态下的，有可能线缆出现故障，无法正常连接两个设备，有可能接口处由于氧化，导致与线缆无法可靠连接，也有可能设备本身出现故障。在这些情况下，尽管方案本身并无问题，但是整个系统仍然无法达到预期的效果。当遇到这样的情况时，我们不应该直觉上第一时间怀疑方案的正确性，而是应该通过“Ablation Study”等方法，逐步缩小故障点所处的范围，找到故障的原因。

第三四次实验，在经过第二次实验对 eNSP 软件的初步学习和练习后，利用 eNSP 实现更加复杂的网络拓扑结构，并逐步考研工程设计能力。在实际中，往往更需要这种能力，即给定用户的需求，基于现有条件利用更低的成本实现用户需要的功能。

综上所述，计网实验不同于以往的实验——之前的实验往往只是在理论上对我们的课程学习起到一个加强巩固的作用，但计网实验更注重的是工程能力的培养。就算理论掌握的滚瓜烂熟，但如果工程能力不达标，不会制作双绞线、不会逐步缩小故障范围进行排错，那也没办法在实际当中真正解决工程难题。因此，在今后的学习与实践过程中，一定要做到理论功底和工程能力并重。两手抓，两手都要硬！

## 5.2 意见和建议

希望 socket 编程实验可以多分配几节课堂时间。这部分实验的任务量较大，但是只有一节课堂时间，同学们不得不利用大量的课余时间来完成这个实验，如果适当调整课时，为

socket 编程多提供几节课时，甚至将本实验拆分成多个步骤分级完成（可以参考 19 级编译原理实验在头歌平台上的检查），这样会给同学们的压力更小。