



哈尔滨工业大学
Harbin Institute of Technology

计算机网络 课程实验报告

实验名称	IPv4 收发和转发实验					
姓名	石翔宇		院系	计算学部		
班级	1903103		学号	1190200523		
任课教师	刘亚维		指导教师	刘亚维		
实验地点	格物 207		实验时间	2021.11.14		
实验课表现	出勤、表现得分(10)		实验报告 得分(40)		实验总分	
	操作结果得分(50)					
教师评语						

实验目的：**IPv4 分组收发实验目的：**

IPv4 协议是互联网的核心协议，它保证了网络节点（包括网络设备和主机）在网络层能够按照标准协议互相通信。IPv4 地址唯一标识了网络节点和网络的连接关系。在我们日常使用的计算机的主机协议栈中，IPv4 协议必不可少，它能够接收网络中传送给本机的分组，同时也能根据上层协议的要求将报文封装为 IPv4 分组发送出去。

本实验通过设计实现主机协议栈中的 IPv4 协议，让学生深入了解网络层协议的基本原理，学习 IPv4 协议基本的分组接收和发送流程。

另外，通过本实验，学生可以初步接触互联网协议栈的结构和计算机网络实验系统，为后面进行更为深入复杂的实验奠定良好的基础。

IPv4 分组转发实验目的：

通过前面的实验，我们已经深入了解了 IPv4 协议的分组接收和发送处理流程。本实验需要将实验模块的角色定位从通信两端的主机转移到作为中间节点的路由器上，在 IPv4 分组收发处理的基础上，实现分组的路由转发功能。

网络层协议最为关注的是如何将 IPv4 分组从源主机通过网络送达目的主机，这个任务就是由路由器中的 IPv4 协议模块所承担。路由器根据自身所获得的路由信息，将收到的 IPv4 分组转发给正确的下一跳路由器。如此逐跳地对分组进行转发，直至该分组抵达目的主机。IPv4 分组转发是路由器最为重要的功能。

本实验设计模拟实现路由器中的 IPv4 协议，可以在原有 IPv4 分组收发实验的基础上，增加 IPv4 分组的转发功能。对网络的观察视角由主机转移到路由器中，了解路由器是如何为分组选择路由，并逐跳地将分组发送到目的主机。本实验中也会初步接触路由表这一重要的数据结构，认识路由器是如何根据路由表对分组进行转发的。

实验内容：**IPv4 分组收发实验内容：****1) 实现 IPv4 分组的基本接收处理功能**

对于接收到的IPv4分组，检查目的地址是否为本地地址，并检查IPv4分组头部中其它字段的合法性。提交正确的分组给上层协议继续处理，丢弃错误的分组并说明错误类型。

2) 实现 IPv4 分组的封装发送

根据上层协议所提供的参数，封装 IPv4 分组，调用系统提供的发送接口函数将分组发送出去。

IPv4 分组收发实验内容：**1) 设计路由表数据结构。**

设计路由表所采用的数据结构。要求能够根据目的 IPv4 地址来确定分组处理行为（转发情况下需获得下一跳的 IPv4 地址）。路由表的数据结构和查找算法会极大的影响路由器的转发性能，有兴趣的同学可以深入思考和探索。

2) IPv4 分组的接收和发送。

对前面实验（IP 实验）中所完成的代码进行修改，在路由器协议栈的 IPv4 模块中能够正确完成分组的接收和发送处理。具体要求不做改变，参见“IP 实验”。

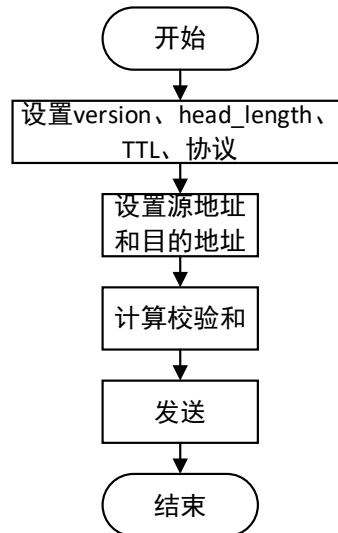
3) IPv4 分组的转发。

对于需要转发的分组进行处理，获得下一跳的 IP 地址，然后调用发送接口函数做进一步处理。

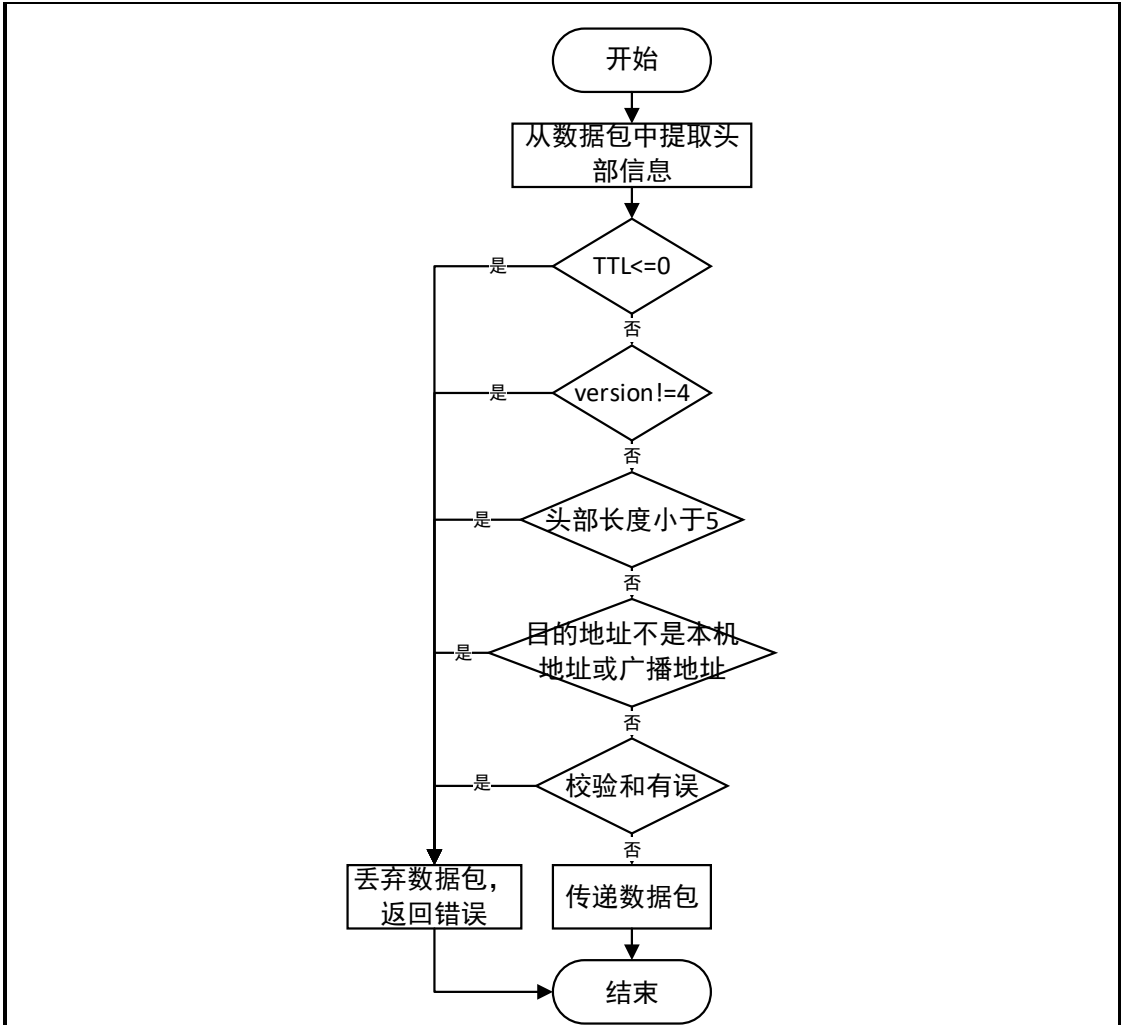
实验过程：

(一) IP分组收发实验

1. 发送函数流程图：



2. 接收函数流程图：



3. IPv4数据包头部信息:



4. 错误检测原理:

1) 版本号字段错误检验原理:

首先我们需要将第0字节的高4位提取出来, 再与4进行比较。实现代码如下:

```
int version = pBuffer[0] >> 4;
if (version != 4)
{
    ip_DiscardPkt(pBuffer, STUD_IP_TEST_VERSION_ERROR);
}
```

```
    return 1;
}
```

2) 头部长度字段检验原理:

我们首先需要提取出第0字节的低4位的信息。由于最少的IP数据包的头部信息为20字节，所以低于5的头部长度字段是错误的，与5进行比较即可。实现代码如下：

```
int head_length = pBuffer[0] & 0xf;
if (head_length < 5)
{
    ip_DiscardPkt(pBuffer, STUD_IP_TEST_HEADLEN_ERROR);
    return 1;
}
```

3) 生存时间字段检验原理:

我们首先需要提取出第8字节的信息，再与0进行比较。如果TTL小于0说明其已经过期。实现代码如下：

```
short ttl = (unsigned short)pBuffer[8];
if (ttl == 0)
{
    ip_DiscardPkt(pBuffer, STUD_IP_TEST_TTL_ERROR);
    return 1;
}
```

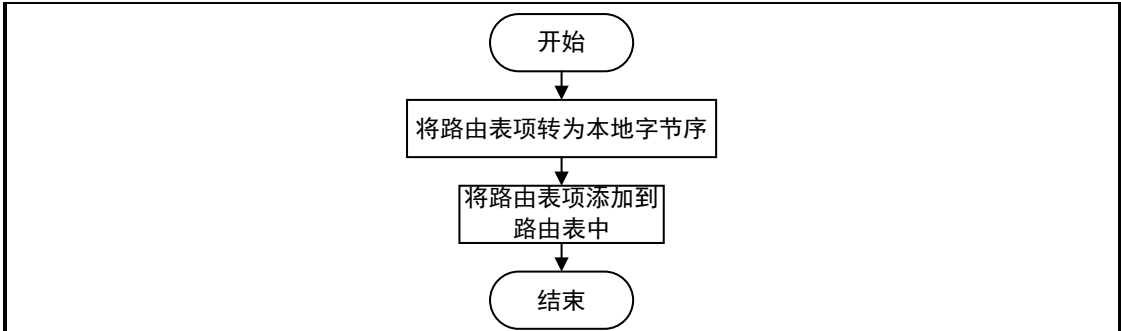
4) 头部校验和字段检验原理:

我们首先需要提取出第10个字节的信息，再根据之前进行计算时取反的性质，将所有的头部信息进行与checksum生成时相同的计算步骤，得到的结果应该为全1。实现代码如下：

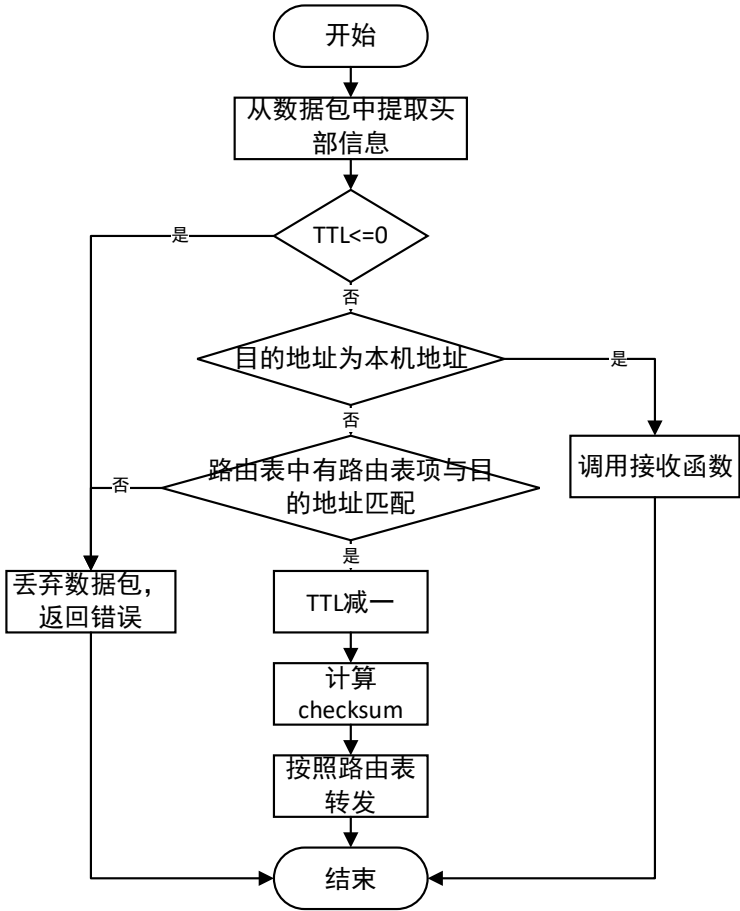
```
unsigned long sum = 0;
for (int i = 0; i < head_length * 2; i++)
{
    sum += (unsigned char)pBuffer[i * 2] << 8;
    sum += (unsigned char)pBuffer[i * 2 + 1];
}
unsigned short l_word = sum & 0xffff;
unsigned short h_word = sum >> 16;
if (l_word + h_word != 0xffff)
{
    ip_DiscardPkt(pBuffer, STUD_IP_TEST_CHECKSUM_ERROR);
    return 1;
}
```

(二) IP分组转发实验

1. 路由增加流程图:



2. 路由转发流程图:



3. 新建数据结构的说明:

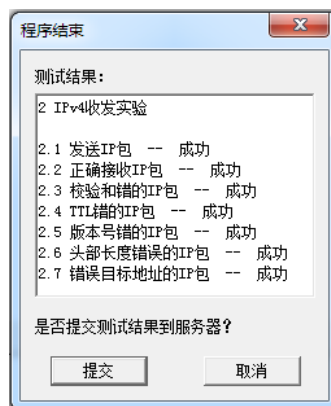
路由表使用的数据结构为STL中的vector。在初始化路由表时新建一个vector对象；增加路由表项时，在vector的尾部增加新的项即可；查找路由表项时，在vector进行查找时，对vector从头部依次进行搜索即可。

实验结果:

(一) IP分组收发实验

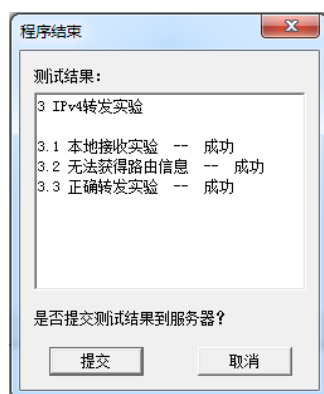
```

accept len = 6 packet
result = 0
send a message to main ui, len = 6 type = 1 subtype = 7
begin test!, testItem = 1 testcase = 1
accept len = 32 packet
accept len = 166 packet
accept len = 38 packet
send a message to main ui, len = 36 type = 2 subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6 type = 1 subtype = 7
begin test!, testItem = 1 testcase = 2
accept len = 32 packet
accept len = 166 packet
accept len = 38 packet
send a message to main ui, len = 36 type = 2 subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6 type = 1 subtype = 7
begin test!, testItem = 1 testcase = 3
accept len = 32 packet
accept len = 166 packet
accept len = 38 packet
send a message to main ui, len = 36 type = 2 subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6 type = 1 subtype = 7
begin test!, testItem = 1 testcase = 4
accept len = 32 packet
accept len = 166 packet
accept len = 38 packet
send a message to main ui, len = 36 type = 2 subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6 type = 1 subtype = 7
begin test!, testItem = 1 testcase = 5
accept len = 32 packet
accept len = 166 packet
accept len = 38 packet
send a message to main ui, len = 36 type = 2 subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6 type = 1 subtype = 7
begin test!, testItem = 1 testcase = 6
accept len = 32 packet
accept len = 166 packet
accept len = 38 packet
send a message to main ui, len = 36 type = 2 subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6 type = 1 subtype = 7
Test over!
    
```



(二) IP分组转发实验

```
begin test!, testItem = 2  testcase = 0
accept len = 32 packet
accept len = 244 packet
accept len = 41 packet
accept len = 55 packet
send a message to main ui, len = 53  type = 2  subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6  type = 1  subtype = 7
begin test!, testItem = 2  testcase = 1
accept len = 32 packet
accept len = 244 packet
accept len = 41 packet
accept len = 38 packet
send a message to main ui, len = 36  type = 2  subtype = 0
accept len = 6 packet
result = 0
send a message to main ui, len = 6  type = 1  subtype = 7
begin test!, testItem = 2  testcase = 2
accept len = 32 packet
accept len = 244 packet
accept len = 41 packet
accept len = 55 packet
send a message to main ui, len = 53  type = 2  subtype = 0
send a message to main ui, len = 53  type = 2  subtype = 1
accept len = 6 packet
result = 0
send a message to main ui, len = 6  type = 1  subtype = 7
Test over!
```



问题讨论:

IP分组转发实验中，如何提高转发效率？

改进路由表的数据结构，可以使用平衡树等非线性结构来存储路由表中的数据，使得在查询路由由表项和添加路由表项时都能够得到时间上的压缩。

心得体会:

通过本次实验，我对IP协议和分组转发有了更加深刻的理解，对IP数据报头部信息各项字段也更加清楚其含义与功能，对路由器的分组转发过程有了更加深刻的理解。体会到了网络协议的必要与其所发挥的极大的作用。