课程编号：B080203100

《网络安全》实验报告

姓　　名	薛　　旗	学　　号	2 0 1 5 5 3 6 2
班　　级	软信-1503	指　导　教　师	侯琳
实　验　名　称	数据包捕获与分析		
开　设　学　期	2017-2018 第一学期		
开　设　时　间	第 17 周——第 18 周		
报　告　日　期	2018 年 1 月 8 日		
评　定　成　绩		评　定　人	
		评　定　日　期	2018 年 1 月 12 日

东北大学软件学院

实验一　数据包捕获与分析

一、实验目的

1.利用数据链路访问接口来进行网络数据包监控。

2.加深对 TCP/IP 协议族中常见协议的理解。

3.掌握如何利用数据包监控程序来了解当前网络状况。

4.编写过滤条件，对网络数据包进行过滤，提取出所关心的网络数据。

二、实验内容

1.底层模块(数据包捕获)

（1）通过 Libpcap 提供的网络数据包捕获接口,捕获流经本网卡的所有原始数据包。

本实验设计的底层模块的初始化工作在函数 pcap_t* open_pcap_socket(char* device, const char* bpfstr)中,定义的回调函数为 void capture_loop(pcap_t* pd, int packets, pcap_handler func),捕获的数据包处理主函数为 void parse_packet(u_char *user, struct pcap_pkthdr *packethdr,u_char *packetptr)

（2）数据包捕获步骤及函数说明

a.网络设备查找

char *pcap_lookupdev(char *errbuf)

获取可被函数调用的网络设备名指针。

b.打开网络设备

pcap_t *pcap_open_live(char *device, int snaplen, int promisc, int to_ms, char *ebuf)

获得用于捕获网络数据包的数据包捕获描述字。

c.获取网络参数

int pcap_lookupnet(char *device, bpf_u_int32 *netp,bpf_u_int32 *maskp, char *errbuf)

获得指定网络设备的网络号和掩码。

d.编译过滤策略

int pcap_compile(pcap_t *p, struct bpf_program *fp,char *str, int optimize, bpf_u_int32 netmask)

将 str 参数指定的字符串编译到过滤程序中。

e.设置过滤器

int pcap_setfilter(pcap_t *p, struct bpf_program *fp)

指定一个过滤程序。

f.利用回调函数捕获数据包

int pcap_loop(pcap_t *p, int cnt,pcap_handler callback, u_char *user)

捕获并处理数据包。

g.关闭网络设备

void pcap_close(pcap_t *p)

关闭 p 参数相应的文件， 并释放资源。

2.中层模块(MAC 层处理模块,IP 层处理模块,TCP 处理模块,UDP 处理模块,ICMP 处理模块)

（1）模块结构

void print_ethernet(struct ether_header* eth) 显示以太网帧头部结构信息

void print_arp(struct ether_arp *arp)显示 arp 报头结构信息

void print_ip(struct ip *ip) 显示 ip 报头结构信息

void print_tcp(struct tcphdr *tcp) 显示 tcp 报头结构信息

void print_udp(struct udphdr *udp) 显示 udp 报头结构信息

void print_icmp(struct icmp *icmp) 显示 icmp 报头结构信息

void dump_packet(unsigned char * buff, int len)

将从 Ethernet 报头的初始地址到 FCS 之前的值使用十六进制整数和 ASCII 码来表示。

char *mac_ntoa(u_char *d) 将 MAC 地址变换为字符串

char *ip_ttoa(int flag) 将 IP 报头中的标志变换为 ASCII 码辅助函数

char *ip_ftoa(int flag) 将 IP 报头中的标志变换为 ASCII 码辅助函数

char *tcp_ftoa(int flag) 将 TCP 报头中的标志变换为 ASCII 码辅助函数

（2）实现方法

通过 Libpcap 提供的网络数据包捕获接口捕获流经本网卡的所有原始数据包。在回调函数中循环处理捕获的数据包.首先，开始处理 Ethernet 的报头。检查 Ethernet 类型之后，分析进行 ARP 协议、IP 协议、其他协议的处理。如果为 IP 协议，则进一步地进行 IP 报头的处理，然后，分别进行下面的 TCP 协议、UDP 协议、ICMP 协议、其他协议等的任一处理。如果判明了协议类型，则按照命令行可选域的指示，显示相关的报头。报头的显示在传输层一级上知道了包的种类之后进行。并且，按照 Ethernet 报头的顺序进行显示。

3.上层统计处理模块(数据包统计模块,数据包协议统计模块,网络元发现模块,数据包构造模块,数据包过滤模块)

（1）相关函数:

void parse_packet(u_char *user, struct pcap_pkthdr *packethdr,u_char *packetptr)

统计变量寄存位置

void bailout(int signo)

打印并输出统计信息

int main(int argc, char **argv)

开始时间统计变量寄存位置

（2）主要功能

a.网络元发现:发现网络上的主机。

b.数据包统计模块及数据包协议统计模块.本次实验中统计模块包含的统计信息有:

开始时间,结束时间,运行时间,捕获的所有数据包,丢弃的数据包,数据包捕获速度,网络超长帧,网络超短帧,数据帧大小(MAC Bytes),捕获数据帧速率(bits/s),ARP 数据包,IP 数据包,TCP 数据包,UDP 数据包,ICMP 数据包,RARP 数据包,其他数据包。

c.设计过滤规则.根据过滤条件对数据包进行过滤.本次实验中实现了对不同协议的数据包进行过滤。

d.构造任意数据包.本实验实现的功能是可构造任意 ICMP 畸形数据包。

4.系统字符命令接口

终端运行程序的语法格式

./test [-ahed] [-i ifrname] [-p protocol] [-n packets]

当表示全部包信息时才指定-a。在不指定-a 时，Ethernet 类型不显示除了 ARP 或 IP 以外的协议。在指定-a 时，则以收到的所有包为显示对象。

如果指定-e，则显示 Ethernet 报头。但是，在指定-a 时，Ethernet 类型不显示除了 ARP 或 IP 以外的协议。

-d 表示包的内容是以 16 进制整数和 ASCII 码来显示的。

-h 表示 help，用来简单地显示使用方法。

在[-i ifname]中，指定读取包的接口名称。在 Linux 操作系统中，如果输入[-i eth0]，则表示从 eth0 接口传输包。在不指定时，通过循环测试接口 Lo 表示通信包。

在[-p protocol]中，指定要显示的包的类型，可以指定的包的类型有：arp、ip、icmp、tcp 和 udp。一次可以指定多个协议，例如，在要显示 TCP 端和 IP 报头时，可以指定[-p tcp ip]。

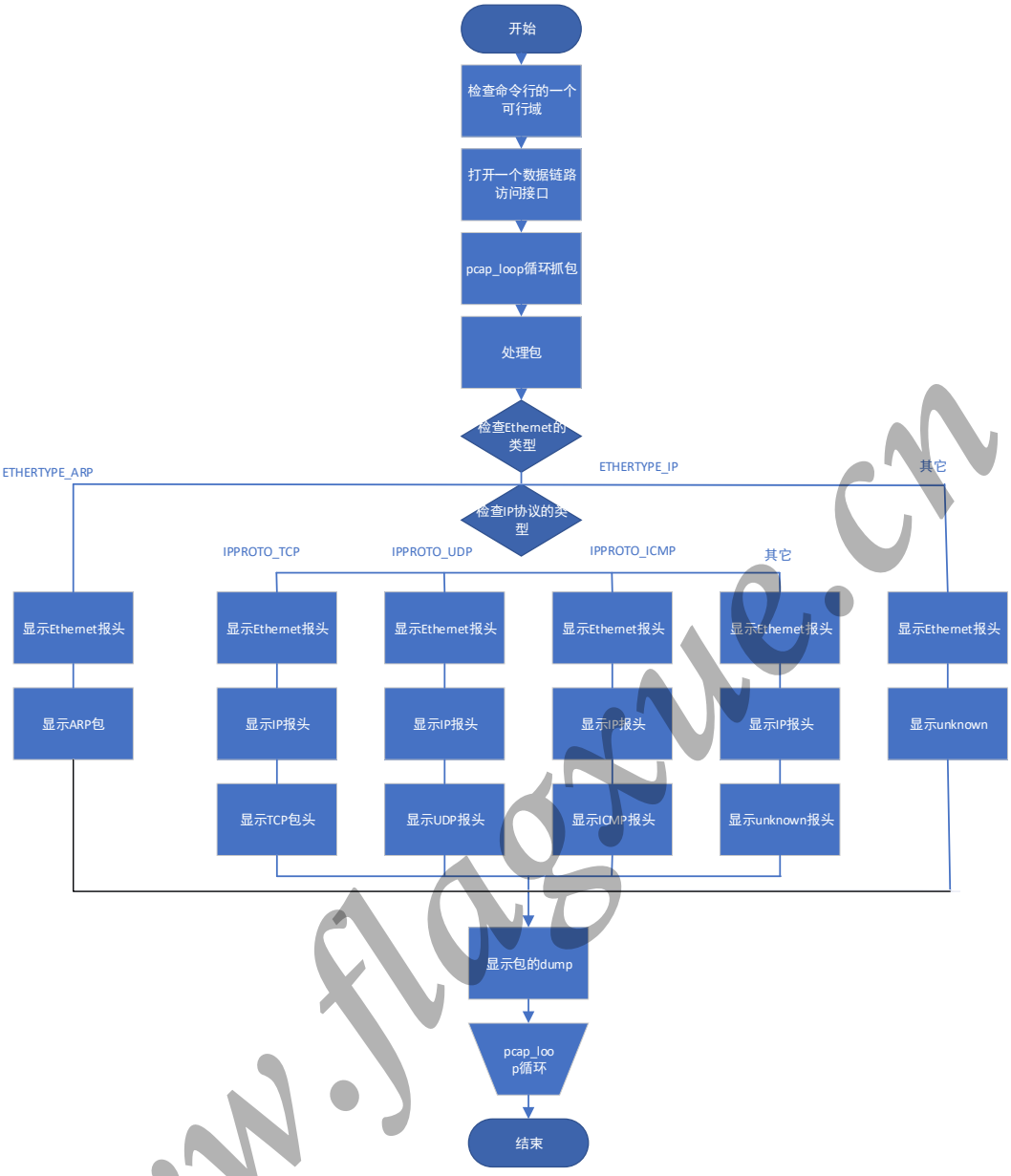
在[-n packets]中,指定要捕获的数据包个数,负数的 packets 表示 pcap_loop 永远循环抓包,直到出现错误或手动终止.

5.拓展

程序在终端运行通过后,通过 Qt Creator 自行设计了图形界面.实现的主要功能有:根据协议进行数据包过滤,数据包显示规则选项,差错处理(未选择过滤规则而运行程序提示警告信息),统计信息,清屏及重置,版权关于页。

三、流程图

（见下页）



四、实验运行结果

1.帮助信息

```
deepin@deepin-PC:~/Program/project/project1$ sudo ./test -h
[sudo] deepin 的密码：
Usage: ./test [-aedh] [-n cnt] [-i ifname] [-p protocols]
Protocols: arp ip icmp tcp udp
default: ./test -p arp ip icmp tcp udp
```

2.默认规则捕获的 IP 和 TCP 包

```
deepin@deepin-PC:~/Program/project/project1$ sudo ./test
Success! Device: wlp6s0

Protocol:IP
|-------------------------------------------------------------------------------|
| IV:4 | HL:  5 | TOS:00000000 | Total_len:         60 |
|-------------------------------------------------------------------------------|
| Identifier:14652 | Flag(R D M):0D0 | Off_set:16384 |
|-------------------------------------------------------------------------------|
| TTL: 64 | Pro:  6 | Header checksum: 62284 |
|-------------------------------------------------------------------------------|
| Soure IP Address:   192.168.0.100 |
| Destination IP Address: 172.217.160.77 |
|-------------------------------------------------------------------------------|

Protocol:TCP
|-------------------------------------------------------------------------------|
| Source Port: 37962 | Destination Port:   443 |
|-------------------------------------------------------------------------------|
| Seq_num:  828016766 |
|-------------------------------------------------------------------------------|
| Ack_numr:          0 |
|-------------------------------------------------------------------------------|
| Off_len: 10 | Reserved | F:000OS0 | Win_size: 29200 |
|-------------------------------------------------------------------------------|
| Checksum: 46993 | Urg_pointer:     0 |
|-------------------------------------------------------------------------------|
```

3.显示统计信息

```
----------------------------
Found New Mac Addresses:

e4:7d:eb:06:d8:5d
00:1d:e0:13:dd:3d


-----------------------------------------------------
 Start Time              |     2018-01-03 22:05:10
 End Time                |     2018-01-03 22:05:18
 Run Time                |     0d 0h 0m 8s ( 8s )
 All Packet(s)           |     15
 Dropped Packet(s)       |     0
 Packet(s) Speed         |     1 packet(s)/s
 MAC Long                |     0
 MAC Short               |     0
 MAC Byte(s)             |     1232
 MAC rate                |     1232 bit/s
 Arp Packet(s)           |     0
 Ip Packet(s)            |     15
   TCP Packet(s)         |     13
   UDP Packet(s)         |     2
   ICMP Packet(s)        |     0
 RARP Packet(s)          |     0
 Other Packet(s)         |     0
-----------------------------------------------------
```

4.过滤规则 UDP，显示包的内容

```
deepin@deepin-PC:~/Program/project/project1$ sudo ./test -d -p udp
Success! Device: wlp6s0

Protocol: UDP
|-------------------------------------------------------------------------------|
| Source Port: 57798 | Destination Port: 13568 |
|-------------------------------------------------------------------------------|
| Length:   39 | Checksum: 27788 |
|-------------------------------------------------------------------------------|

Frame Dump:
e47d eb06 d85d 001d e013 dd3d 0800 4500 .}...].....=..E.
003b ebfc 4000 4011 ccff c0a8 0064 c0a8 .;..@.@......d..
0001 e1c6 0035 0027 6c8c 7aac 0100 0001 .....5.'l.z.....
0000 0000 0000 0373 7030 0562 6169 6475 .......sp0.baidu
0363 6f6d 0000 0100 01                  ;.com.....
```

5.过滤规则为 UDP 的统计信息

```
----------------------------
Found New Mac Addresses:

e4:7d:eb:06:d8:5d
00:1d:e0:13:dd:3d

-----------------------------------------------------
 Start Time              |     2018-01-03 22:07:49
 End Time                |     2018-01-03 22:07:57
 Run Time                |     0d 0h 0m 8s ( 8s )
 All Packet(s)           |     30
 Dropped Packet(s)       |     0
 Packet(s) Speed         |     3 packet(s)/s
 MAC Long                |     0
 MAC Short               |     0
 MAC Byte(s)             |     13615
 MAC rate                |     13615 bit/s
 Arp Packet(s)           |     0
 Ip Packet(s)            |     30
   TCP Packet(s)         |     0
   UDP Packet(s)         |     30
   ICMP Packet(s)        |     0
 RARP Packet(s)          |     0
 Other Packet(s)         |     0
-----------------------------------------------------
```

6.发送自定义的 ICMP 包

```
deepin@deepin-PC:~/Program/project/project2$ sudo ./test
[sudo] deepin 的密码：
Interface: wlp6s0:
Mac Address: 00:1d:e0:13:dd:3d
```

7.过滤规则为 ICMP，显示包信息（此 ICMP 包为自己构造的数据包）

8.图形界面：当未选择过滤规则时，显示警告信息。

9.图形界面：默认过滤规则下显示捕获的 ARP 数据报头信息

10.图形界面：默认过滤规则下显示捕获的 TCP 数据报头部信息

11.图形界面：默认规则下捕获的数据包统计信息

12.图形界面：关于页面

五、实验总结

通过本次实验，我掌握了如何利用 Libpcap 提供的网络数据包捕获接口捕获流经本网卡的所有原始数据包，并在此基础上对捕获的数据包进行统计分析。同时，知道了如何通过编写过滤条件，对网络数据包进行过滤，提取出所关心的网络数据。在实验过程中，不仅加深了我对 TCP/IP 协议族中常见协议的的理解，还提高了自身的编程能力，获益匪浅。

评价表格

考核标准	得分
（1）正确理解和掌握实验所涉及的概念和原理（20%）；	
（2）按实验要求合理设计数据结构和程序结构（20%）；	
（3）运行结果正确（20%）；	
（4）认真记录实验数据，原理及实验结果分析准确（20%）；	
(5)实验过程中,具有严谨的学习态度和认真、踏实、一丝不苟的科学作风(10%)；	
（7）实验报告规范（10%）。	
合计	