此处给出一个简单的示例程序，通过原始套接字将发给本机的所有IP数据报打印出来：

// 必要的头文件

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <net/ethernet.h>

// IP结构体

struct ip\_packet

{

unsigned char dst\_mac[6];

unsigned char soc\_mac[6];

unsigned short ethertype;

unsigned char ip\_v:4; // 版本

unsigned char ip\_hl:4; // 首部长度

unsigned char ip\_tos; // 服务类型

unsigned short tot\_len; //总长度

unsigned short id; //标志

unsigned short off; // 分片偏移

unsigned char ttl; //生存时间

unsigned char pro;

unsigned short chk\_sum;

unsigned char soc\_ip[4];

unsigned char dst\_ip[4];

};

int main()

{

int fd,i;

struct ip\_packet ip;

fd = socket(PF\_PACKET,SOCK\_RAW,htons(ETH\_P\_IP));

if(fd == -1)

{

perror("创建套接字失败！");

}

while(1)

{

if(recv(fd,&ip,sizeof(ip),0) == -1)

continue;

printf("\n==========================================");

printf("\n\t%d bytes read",ntohs(ip.tot\_len));

// 打印接收到的IP数据报中MAC地址和IP地址的信息

printf("\n\tDestMACAddr:");

for(i=0;i<6;i++)

{

printf(i > 0 ? ":%.2x" : "%.2x", ip.dst\_mac[i]);

}

printf("\n\tSourceMACAddr:");

for(i=0;i<6;i++)

{

printf(i > 0 ? ":%.2x" : "%.2x", ip.soc\_mac[i]);

}

printf("\n\tSource host:");

for(i=0;i<4;i++)

{

printf(i > 0 ? ".%d" : "%d", ip.soc\_ip[i]);

}

printf("\n\tDest host:");

for(i=0;i<4;i++)

{

printf(i > 0 ? ".%d" : "%d", ip.dst\_ip[i]);

}

}

return 0;

}

该程序需要以root身份编译和运行。最终打印出IP数据报的目的和源MAC地址和IP地址信息。

