\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Report: HW0\_7

Author: F74071140 謝育萱 <[shirleysensible@gmail.com](mailto:shirleysensible@gmail.com)>

Class: 甲班

Description:

這次練習了struct的用法，以及使用fread的技巧。struct可以一次宣告不同資料型態的變數，因此在做資料歸類統整時非常方便；而fread是讀取資料，讀完一筆後會自動讀取下一筆，其用法為：fread（要存放進的變數名稱，一次讀幾個byte，讀幾次，從哪個資料讀）。這次功課有一個值得注意的地方就是feof的用法，feof放在不同地方會影響程式跑出不一樣的結果：用while(!feof(資料))呈現的結果會多一次，但用while(1){ if(!feof(資料)) break;}就可以避免掉多算一次。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Code:

#include<stdio.h>

#include<stdlib.h>

#include<stdint.h>

//IPV4 的STRUCT

struct IPV4{

int length;

unsigned char pro[1];

unsigned char SRC\_IP[4];

unsigned char DST\_IP[4];

}I;

//TCP的STRUCT

struct TCP{

int SRC\_Port;

int DST\_Port;

}T;

//UDP的STRUCT

struct UDP{

int SRC\_Port;

int DST\_Port;

}U;

//MAC的STRUCT

struct MAC{

unsigned char DST\_MAC[6];

unsigned char SRC\_MAC[6];

}M;

int main(int argc,char\* argv[]){

FILE \*fp;

fp=fopen(argv[1],"rb");

if(!fp){

printf("Can't open the file\n");

return -1;

}

unsigned char SRC\_Port[2]="";

unsigned char DST\_Port[2]="";

unsigned char length[2]="";

unsigned char dont\_need[65535]=""; //資料最長的長度(不包括Ethernet)

int Pcount=0,Tcount=0,Ucount=0,Icount=0,i=0,num=1;

while(1){

num++;

//計算DST MAC

fread(M.DST\_MAC,sizeof(char),6,fp);

if(feof(fp)) break;

printf("#%d\n",num);

printf("DST MAC:");

for(i=0;i<5;i++)

printf("%02x:",M.DST\_MAC[i]);

printf("%02x\n",M.DST\_MAC[5]);

//計算SRC MAC

fread(M.SRC\_MAC,sizeof(char),6,fp);

printf("SRC MAC:");

for(i=0;i<5;i++)

printf("%02x:",M.SRC\_MAC[i]);

printf("%02x\n",M.SRC\_MAC[5]);

//把不要的去掉

fread(dont\_need,sizeof(char),4,fp);

//計算長度

fread(length,sizeof(char),2,fp);

I.length=256\*length[0]+length[1]+14;

//把不要的去掉

fread(dont\_need,sizeof(char),5,fp);

//計算總共有幾個Packet

fread(I.pro,sizeof(char),1,fp);

Pcount++;

printf("Protocal:");

//為TCP時,計算該SRC IP,DST IP,SRC Port,DST Port

if(I.pro[0]==6){

printf("TCP\n");

Tcount++;

fread(dont\_need,sizeof(char),2,fp);

fread(I.SRC\_IP,sizeof(char),4,fp);

printf("SRC IP:");

for(i=0;i<3;i++)

printf("%d.",I.SRC\_IP[i]);

printf("%d\n",I.SRC\_IP[3]);

fread(I.DST\_IP,sizeof(char),4,fp);

printf("DST IP:");

for(i=0;i<3;i++)

printf("%d.",I.DST\_IP[i]);

printf("%d\n",I.DST\_IP[3]);

fread(SRC\_Port,sizeof(char),2,fp);

T.SRC\_Port=SRC\_Port[0]\*256+SRC\_Port[1];

fread(DST\_Port,sizeof(char),2,fp);

T.DST\_Port=DST\_Port[0]\*256+DST\_Port[1];

printf("SRC\_Port:%d\n",T.SRC\_Port);

printf("DST\_Port:%d\n",T.DST\_Port);

fread(dont\_need, sizeof(char), length[0]\*256+length[1]-24,fp);

printf("Length:%d\n",I.length);}

//為UDP時,計算該SRC IP,DST IP,SRC Port,DST Port

if(I.pro[0]==17){

printf("UDP\n");

Ucount++;

fread(dont\_need,sizeof(char),2,fp);

fread(I.SRC\_IP,sizeof(char),4,fp);

printf("SRC IP:");

for(i=0;i<3;i++)

printf("%d.",I.SRC\_IP[i]);

printf("%d\n",I.SRC\_IP[3]);

fread(I.DST\_IP,sizeof(char),4,fp);

printf("DST IP:");

for(i=0;i<3;i++)

printf("%d.",I.DST\_IP[i]);

printf("%d\n",I.DST\_IP[3]);

fread(SRC\_Port,sizeof(char),2,fp);

U.SRC\_Port=SRC\_Port[0]\*256+SRC\_Port[1];

fread(DST\_Port,sizeof(char),2,fp);

U.DST\_Port=DST\_Port[0]\*256+DST\_Port[1];

printf("SRC\_Port:%d\n",U.SRC\_Port);

printf("DST\_Port:%d\n",U.DST\_Port);

fread(dont\_need, sizeof(char), length[0]\*256+length[1]-24,fp);

printf("Length:%d\n",I.length);}

//為ICMP時,計算該SRC IP,DST IP

if(I.pro[0]==1){

printf("ICMP\n");

Icount++;

fread(dont\_need,sizeof(char),2,fp);

fread(I.SRC\_IP,sizeof(char),4,fp);

printf("SRC IP:");

for(i=0;i<3;i++)

printf("%d.",I.SRC\_IP[i]);

printf("%d\n",I.SRC\_IP[3]);

fread(I.DST\_IP,sizeof(char),4,fp);

printf("DST IP:");

for(i=0;i<3;i++)

printf("%d.",I.DST\_IP[i]);

printf("%d\n",I.DST\_IP[3]);

fread(dont\_need, sizeof(char), length[0]\*256+length[1]-20,fp);

printf("Length:%d\n",I.length);}

}

num--;

printf("Packet:%d\n",Pcount);

printf("TCP:%d\n",Tcount);

printf("UDP:%d\n",Ucount);

printf("ICMP:%d\n",Icount);

fclose(fp);

return 0;

}