GitHub链接：<https://github.com/Fattysand/simple_sniffer.git>

实验环境：Windos10专业版，QT5.9.2，QTcreator4.4.1，WpdPack（WinPcap）

实验中包含release分支，包含exe文件可以直接执行。

**1A网络嗅探器实现**

整体架构如下图所示：

图示

描述已自动生成

具体功能：列举主机的所有网卡，并选择一个进行监听，获取所有经过该网卡的数据包，利用WinPcap提供的函数来进行过滤规则的设置，对每一个捕获到的数据包，按照各种协议的格式进行解析并显示在主界面上。最后允许以文件的形式保存或读取之前获取到的数据包。

按流程依次实现该嗅探器的功能：

1. 获取主机网卡

使用for循环将主机网卡读取出来，并存入alldevs这个链表中

截图里有图片

描述已自动生成

然后将读取到的每个网卡信息项目添加到UI界面的复选框中

文本

描述已自动生成

1. 过滤规则设置

使用winpcap中的pcap\_setfilter函数进行数据包过滤

1. 创建线程捕获数据包

sniff\_startCap函数实现了如何创建线程并利用Winpcap提供的函数进行数据包捕获。

首先获取已选择的网卡和已设置的过滤规则

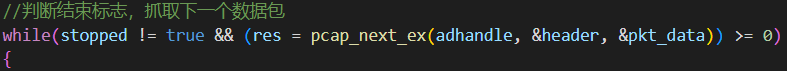
文本

描述已自动生成

然后做一些必要的检查：以太网，过滤器语法，临时文件能否生成（保存数据包）

然后执行capthread抓包线程：

使用while循环不断抓取数据包



1. 解析数据包

要解析数据包的话首先要定义好每种协议的结构体：

MAC头

文本

描述已自动生成

ARP头

文本

描述已自动生成

IP头

文本

描述已自动生成

TCP头

文本

描述已自动生成

UDP头

文本

描述已自动生成

ICMP头

文本

描述已自动生成

在获取到数据包之后，使用analyze\_frame函数对链路层数据包进行分析

图形用户界面, 文本

中度可信度描述已自动生成

分析其中是IP包还是ARP包

屏幕上有字

描述已自动生成

获取数据包时间

图片包含 文本

描述已自动生成

获取MAC地址

电脑屏幕截图

描述已自动生成

获取IP地址

图片包含 文本

描述已自动生成

同时将获取的数据包信息显示在主界面上

再对数据包的内容进行详细解析：

文本

描述已自动生成

在analyze\_frame函数中，利用数据包协议字段对数据包进行区分

电脑屏幕截图

描述已自动生成

在analyze\_ip函数中，对数据包进行进一步区分

电脑屏幕的截图

描述已自动生成

其余信息按照之前定义好的协议结构体进行赋值即可完成数据包解析，以TCP解析为例：

文本

描述已自动生成

简要总结：捕获数据包后，先分是IP数据包还是ARP数据包，如果是IP数据包，则继续利用数据包中协议字段进行区分，是TCP，UDP还是ICMP报文，然后进行分析即可。

1. 文件保存与读取

通过WinPcap官方文档得知，将数据包保存到脱机堆文件中需要使用到两个函数：pcap\_dump\_open(): 该调用将打开一个堆文件，并将它关联到特定的接口上；pcap\_dump()：数据包通过pcap\_dump()函数写入到堆文件中。

使用开始获取时间作为保存文件的名字：

电脑屏幕截图

描述已自动生成

数据包的读取则根据保存的文件来进行分析：

使用pcap\_createsrcstr函数，将文件看成数据流，对该数据流重新执行如上文所示的数据包捕获与分析，即可完成文件的读取，如下图所示：

文本

描述已自动生成

然后再创建一个capthread对该文件进行分析即可。

补充：

在使用QT生成release文件时，发生错误，提示缺少dll文件。

使用命令：windeployqt Simple\_Sniffer.exe 补充缺少的dll文件，使其可以直接执行。