二、主要算法及数据结构

2.1 引用库

sys：用于设置程序运行环境的参数，如使用utf-8进行编码

pcap：用于抓包

dpkt：用于解析包

datetime：用于对时间戳数据结构的转换

util（自己编写）：用于处理mac地址使其成为一个string对象（mac\_addr函数）、处理ip地址使其成为一个string对象（inet\_to\_str函数）、保存文件时把包信息表示为可写入的数据结构（deal\_save函数）

re：主要用于查找字符串子串（search函数）

json：包信息存储使用json格式的编码，易于阅读和机器解析

threading：用于多线程抓包

pyqt5：用于界面设计（配合qtdesigner）

Ui\_main：程序界面的代码版本

2.2 sniffer.py文件详解

该文件定义了一个类sniffer，其属性有：

btn\_stop：用于判断是否按下停止键（停止抓包）

filter：用于pcap数据包过滤器设置，过滤规则由输入框读取（QInputDialog.getText函数），并通过pcap库中内置的过滤表达式语法来设置。

package\_tcp：用于存放tcp包

package\_udp：用于存放udp包

package\_icmp：用于存放icmp包

file\_tcp：用于存放tcp包组成的文件

sniffer类的方法有：

\_\_init\_\_：在这个方法中，创建了程序的UI并且获取网卡列表

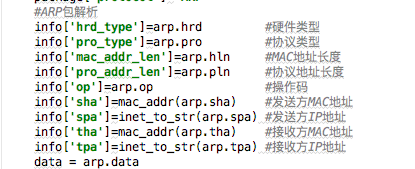
get\_pkg\_icmp、set\_pkg\_icmp、del\_pkg\_icmp：返回、设置、删除icmp包（udp、tcp包、file\_tcp同理）

deal\_package：

\*在这个函数中，包被抓取后通过dkpt库的Ethernet函数解析，并赋值给eth。

\*函数定义了一个名为package的词典，用于存放解析出来的各类信息，其中包括时间戳（键为timestamp）、包的长度（键为len）、通过十六进制转换后的整个包（键为buf）以及用以存储包特定信息的字典info

\*通过类名比较来确定包的类别，使用dkpt库来解析包的各类信息并赋值给info，把info存入package中并且在UI中显示（变量show）。详细包信息可见代码注释，如下图



\*如果是ip6包，则可通过dpkt.ip6.nxt属性来判断协议，icmp包的值为1，igmp包的值为2，tcp包的值为6，udp包的值为17

\*如果是ip4包，通过isinstance函数来检查包的类型，信息存储在package词典中；检查属性offset和mf来判断包是否分段，若有后续部分，则将包暂存到list列表中

package\_reader：在这个方法中，会根据过滤包规则来进行抓包

on\_btn\_begin\_clicked：定义了触发“开始抓包”按钮点击的事件，会调用package\_reader函数开始抓包；开启多个线程抓包

on\_btn\_clear\_history\_clicked：定义了触发“清屏”按钮点击的事件

on\_btn\_stop\_clicked：定义了触发“停止抓包”按钮点击的事件

on\_btn\_filter\_clicked：定义了触发“过滤包”按钮点击的事件，把弹窗中获取的信息返回给rule来设置规则

on\_btn\_search\_clicked：定义了触发“搜索”按钮点击的事件

on\_btn\_viewlog\_clicked：定义了触发“查看全部”按钮点击的事件

on\_btn\_recover\_clicked：

\*定义了触发“文件重组 ”按钮点击的事件

\*for循环遍历所有包的包信息，查找特征字段来筛选需要合并的包

\*把包存入list列表中，使用sort函数通过offset来排序，写入本机文件中

on\_btn\_save\_clicked：定义了触发“保存”按钮点击的事件

on\_btn\_exit\_clicked、on\_choose\_eth\_activated：pass

on\_choose\_eth\_currentIndexChanged：显示当前网卡（控制台输出）

on\_package\_info\_cellClicked：点击某个包之后，显示包信息