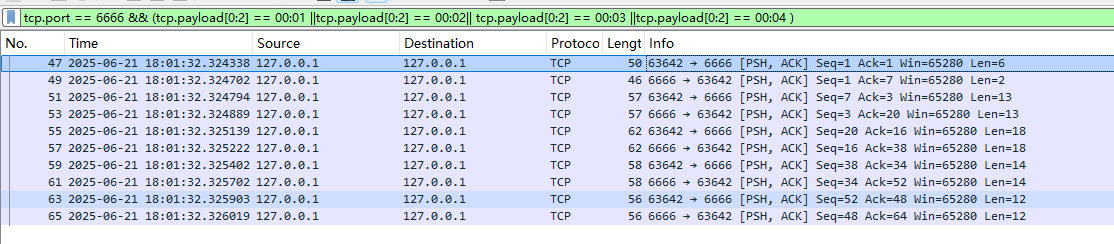
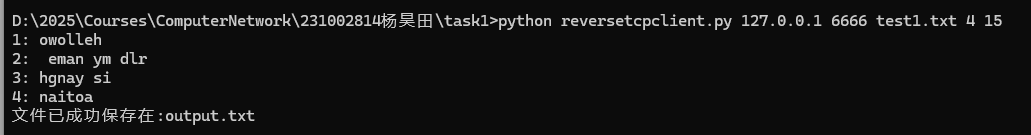
tcp\_packet\_capture

一.Wireshark数据包解释

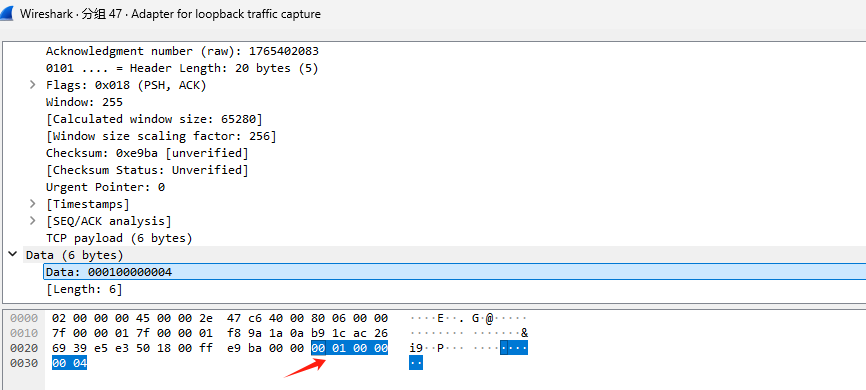


共有10条报文 分别为 1 2 3 4,3 4,3 4,3 4 号报文 共四块数据块

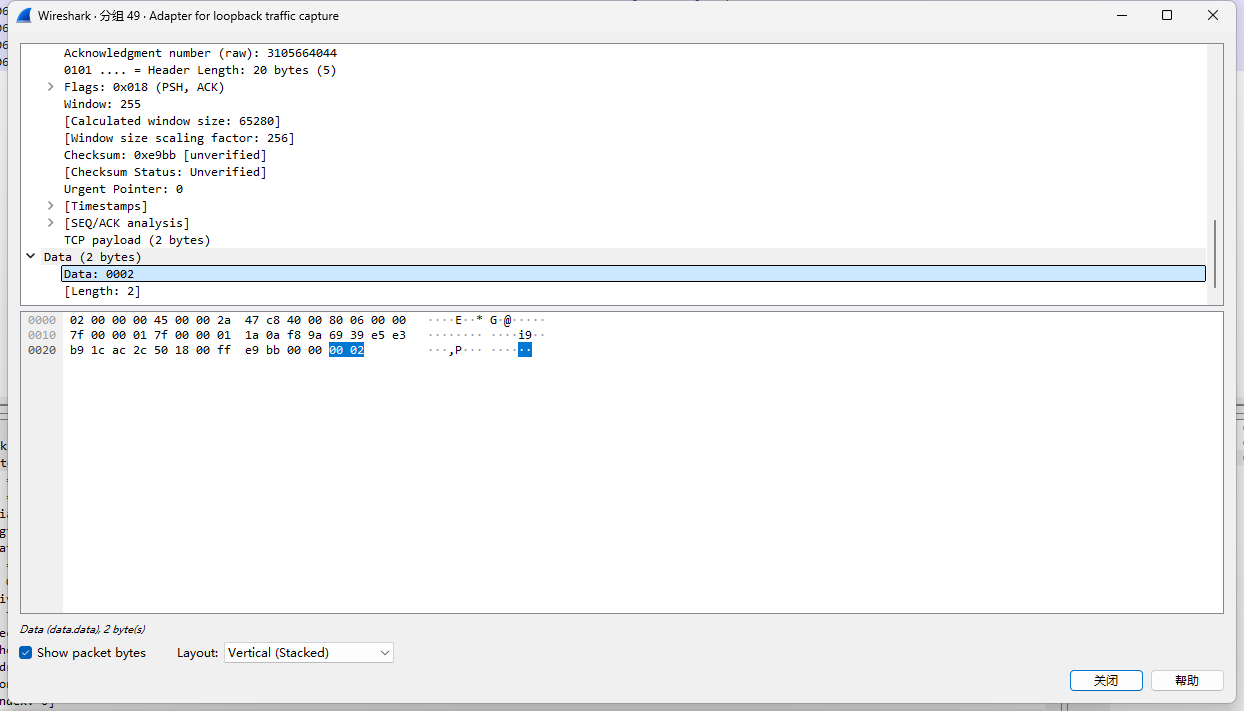
图为wireshark 捕获的报文



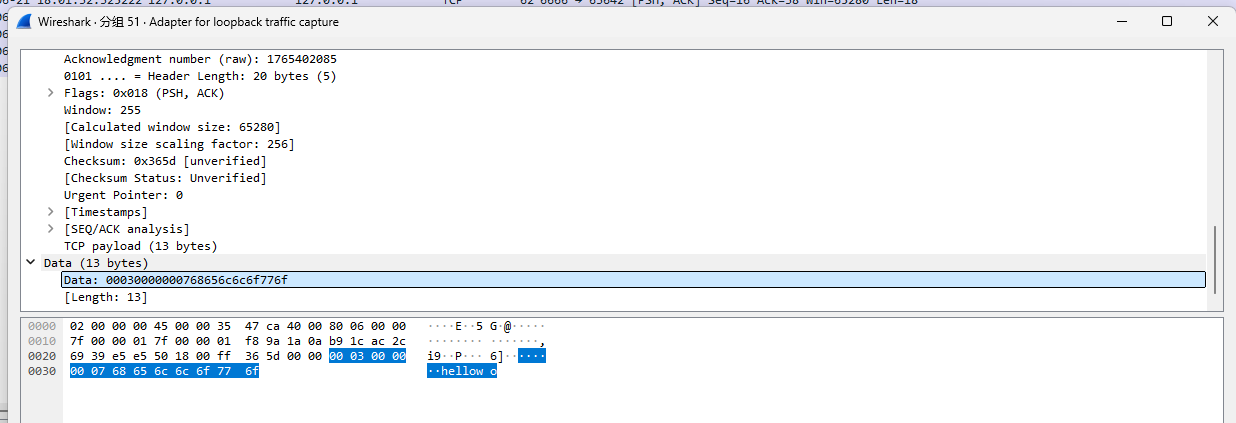
图为cmd中运行的结果



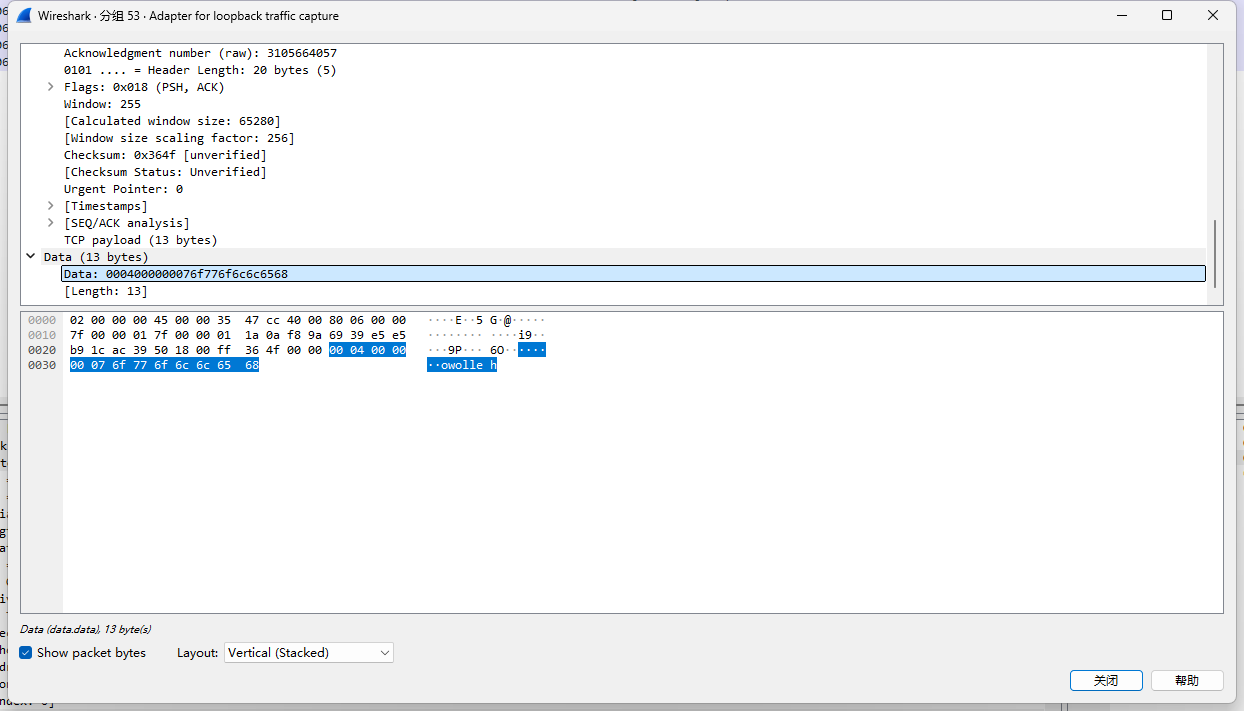
初始化报文 共 6个字节 前两个字节为 00 01,表示Type 后四个字节为n = 4 表示有四块需要反转,正好与cmd结果相对应



Agree报文 只有两个字节 00 02 表示Type = 2



如图为 reverserequest报文 13个字节 前两个字节 00 03 表示Type = 3, 之后4个字节 00 00 00 07 表示request的数据长度为7个字节 最后7个字节是数据hellowo



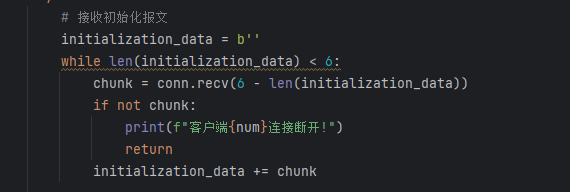
如图为reverseanswer报文共13个字节 前两个字节 00 04 表示 Type = 4,后4个字节 00 00 00 07 表示数据长度为7个字节最后7个字节是反转的数据 owokkeh

至此4个数据块的中的第一个就反转成功.之后同理

1. 关键点及代码实现:



1. 服务器采用多线程来连接多个客户端,



1. 接收报文时,要循环读入保证报文header字节够



1. 在发送数据的时候使用sendall来发送,保证一次性发送完成
2. 用到和掌握的知识点
3. 掌握客户端,服务器的python简单网络编程的流程
4. 能够使用struct来处理字节信息,pack,unpack 封装与解封装的方法
5. 通过wireshark捕获自己发送的报文,更能看出来header和data,以及这次通信的流程,像上面提到的各种数据都对的上,比如4块数据块,报文是10条. 每个报文的字节的具体分析.
6. 对于python 命令行参数的读入与解析
7. Git :