信息安全技术第二次实验报告

**一、实验目的**

学习虚拟机安装，版本控制软件、抓包软件使用及初步具备协议分析能力

**二、实验工具**

服务器端Centos7，客户端windows10，GitHub，Tcpdump，Wireshark

**三、过程记录**

**实验A**

首先使用ifconfig指令确定服务器网卡号和IP地址，如图1所示：

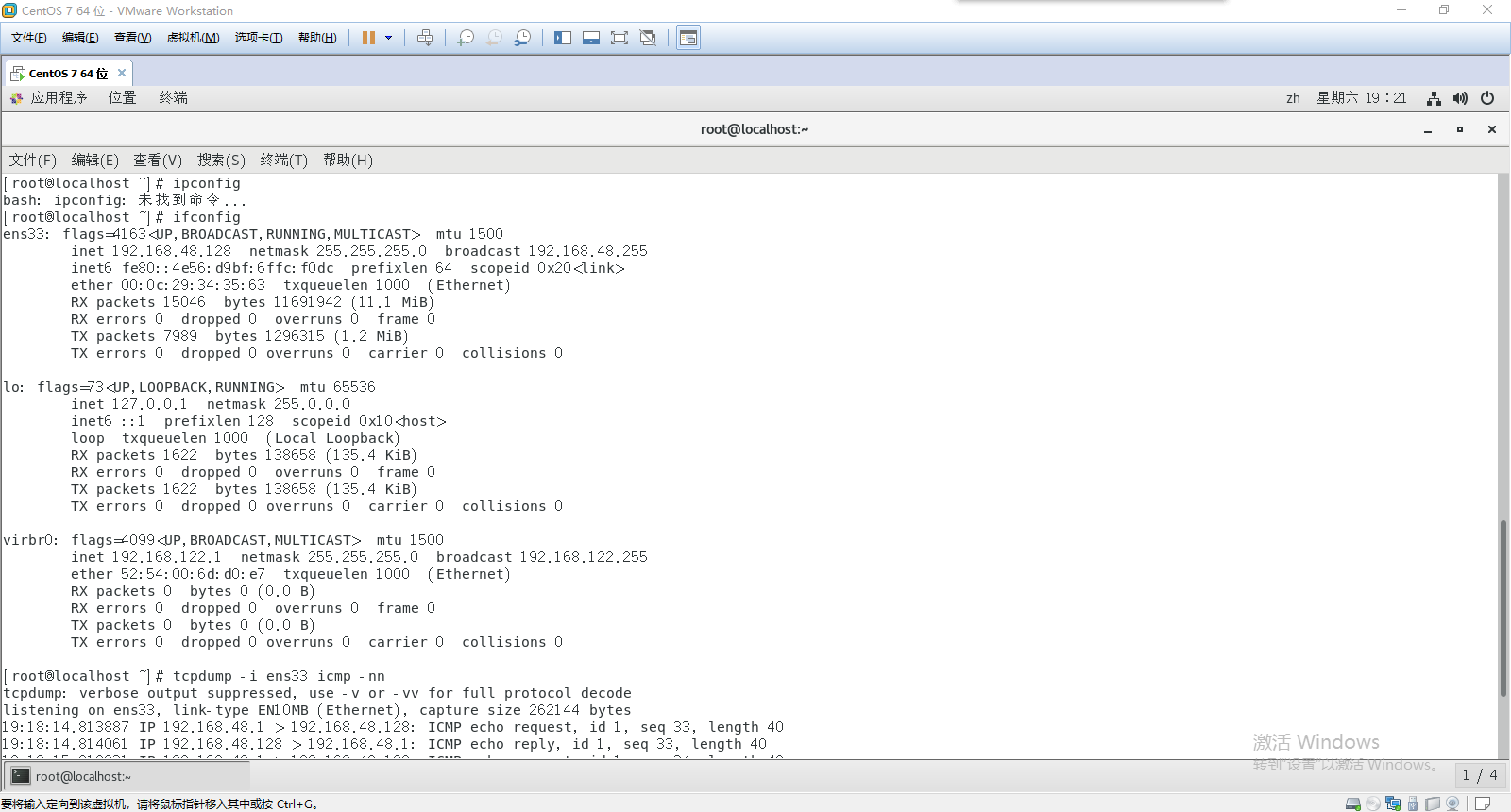


图1 查询网卡和IP

使用tcpdump –i ens33 icmp -nn指令开启服务器端的抓包功能，抓取通过ens33端口收到的ICMP协议报文，同时也可以用and host 限定主机IP，用-c限定抓包个数等，如图2所示：

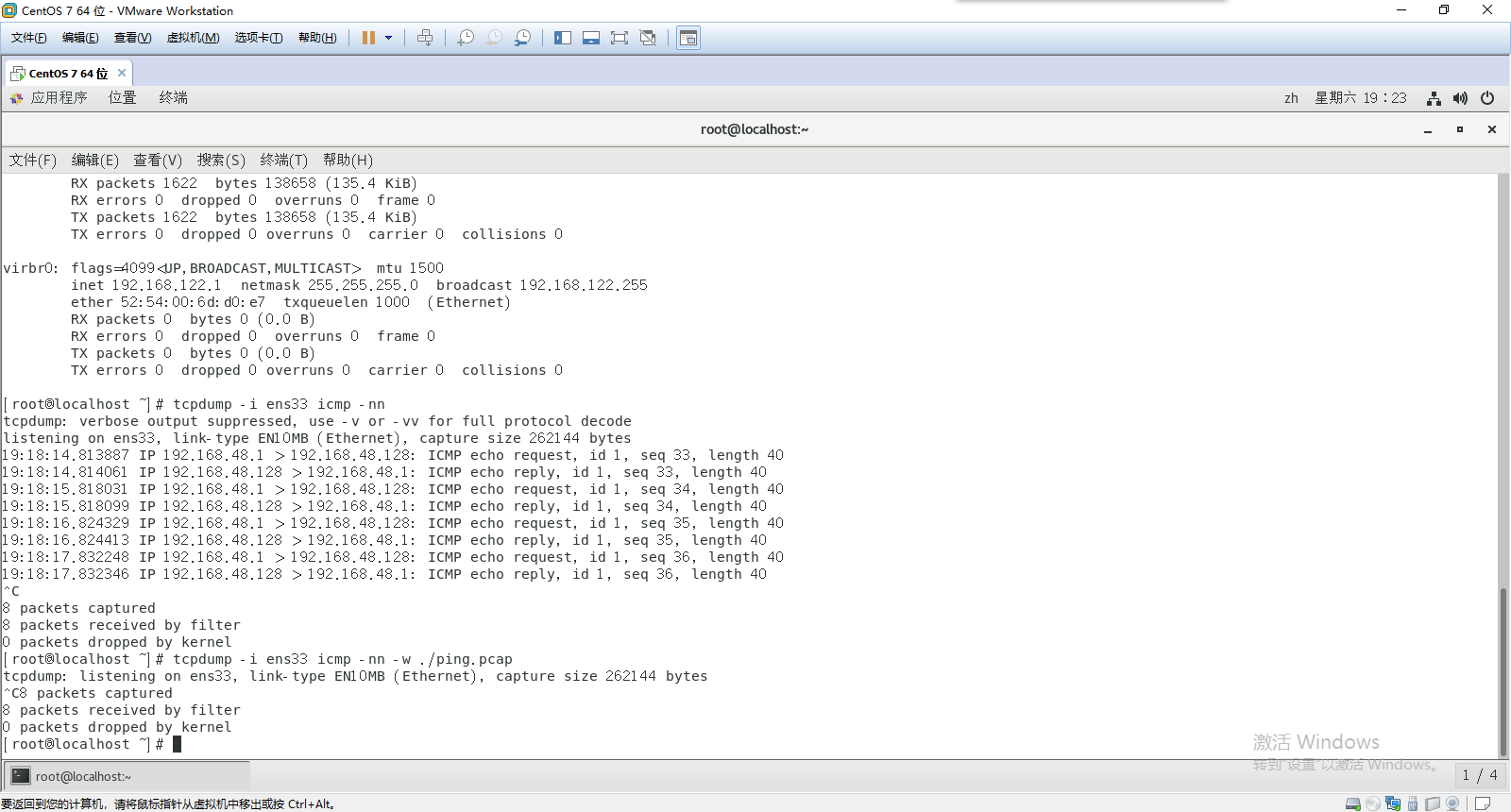


图2 tcpdump抓包

之后在客户端用ping 192.168.48.128指令通过ICMP协议访问服务器，在服务器端抓取到的ICMP包信息，如图3，图4所示：

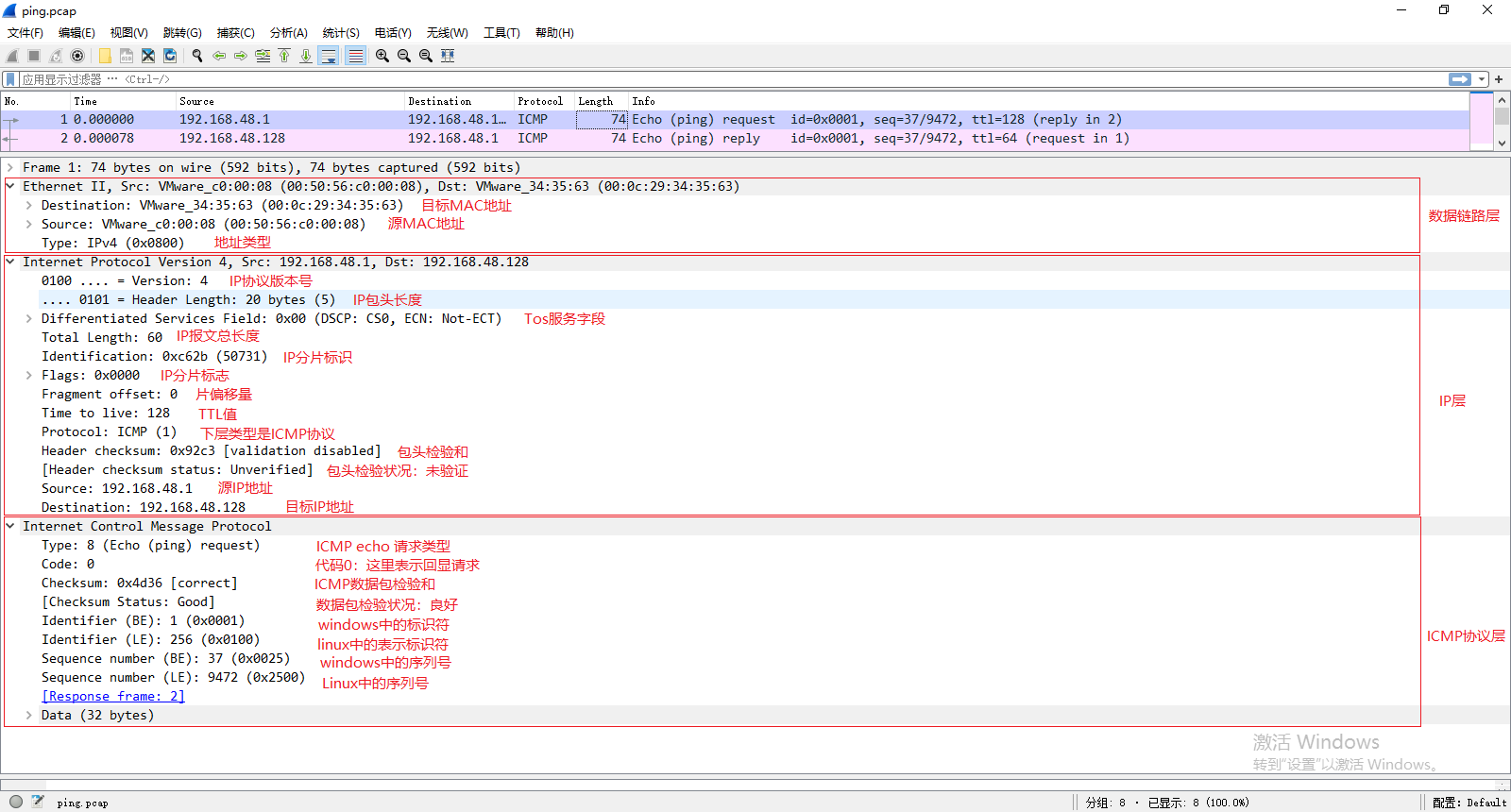
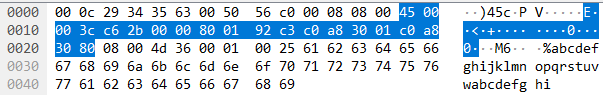


图3 请求报文解析



4500+003c+c62b+0000+8001+0000+c0a8+3001+c0a8+3080=36d39

0003+6d39=6d3c 取反可得 Checksum=92c3

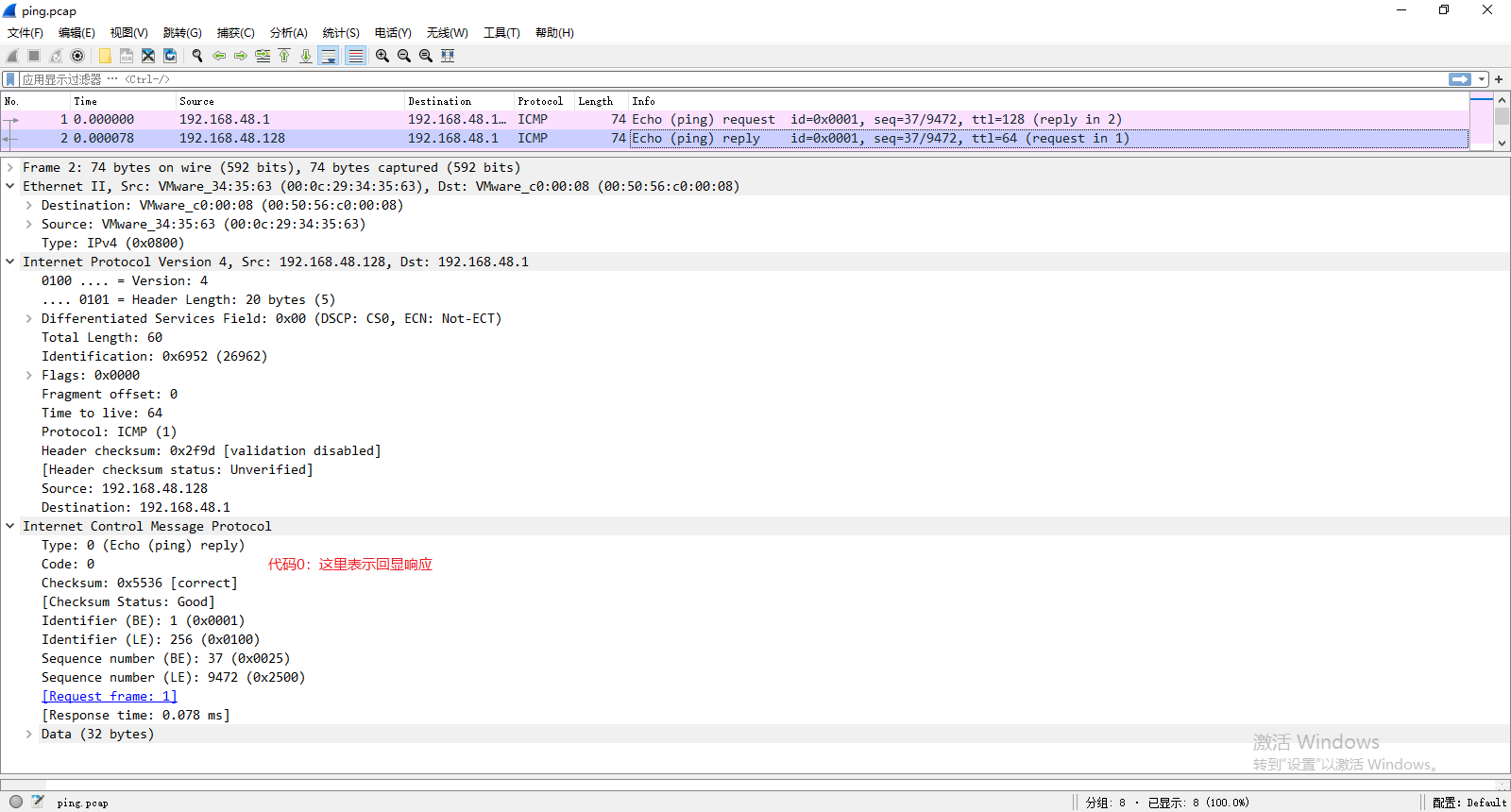
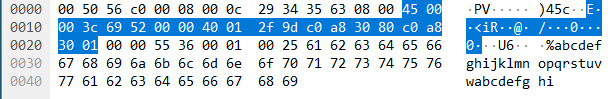


图4 应答报文解析



4500+003c+6952+0000+4001+0000+c0a8+3080+c0a8+3001=2d060

0002+ d060= d062 取反可得 Checksum=2f9d

**实验B**

首先利用firewall-cmd指令检测80端口和http协议是否打开，确认打开后在客户端利用curl指令对服务器进行访问，访问结果如图所示。

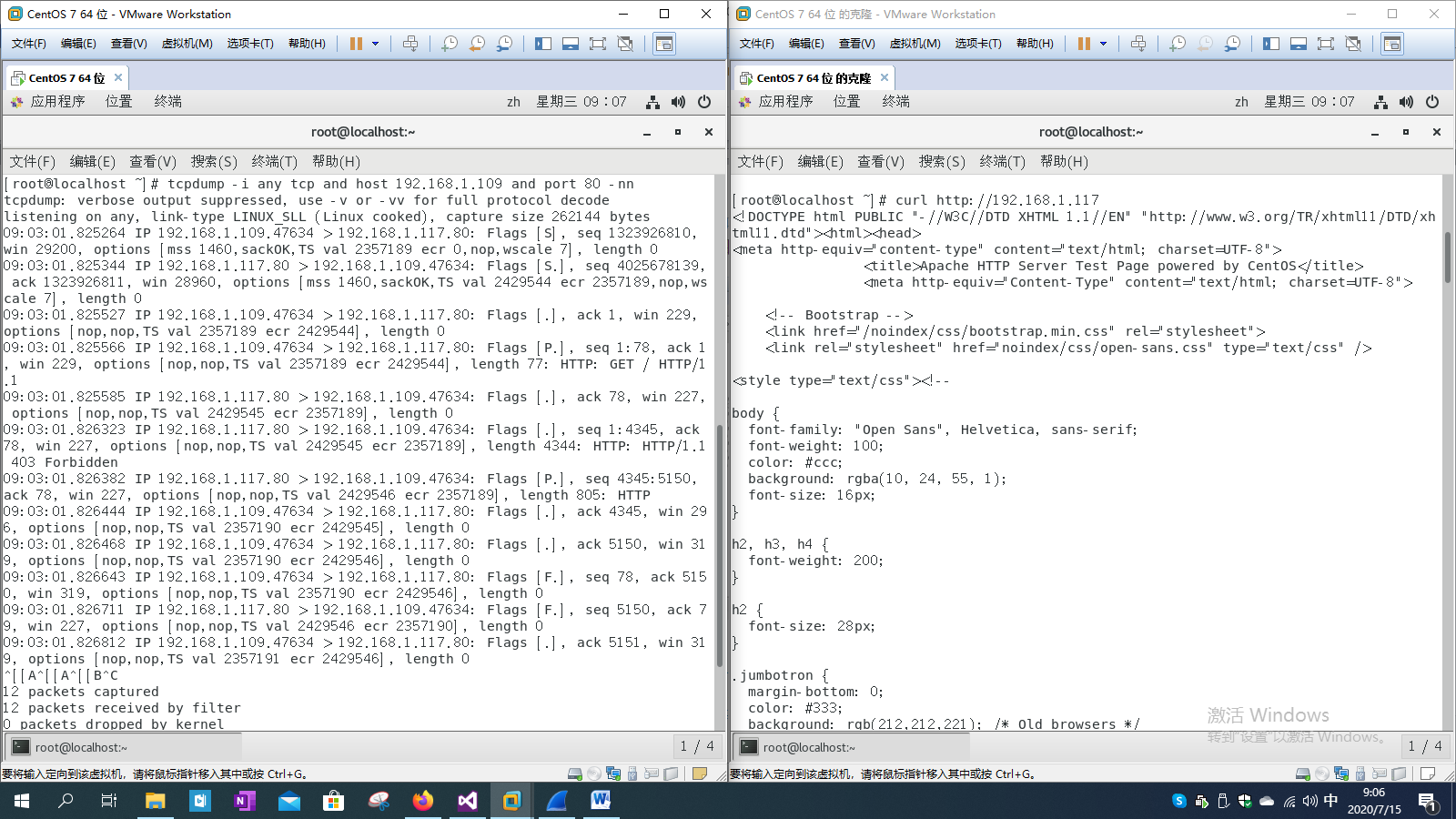


图5 http协议访问及抓包

三次握手实验结果如图6-8所示，四次挥手实验结果如图9-12所示，分析其中SYN，ACK，FIN等标志及 Sequence number，Acknowledgement number数值在握手连接中的变化情况。

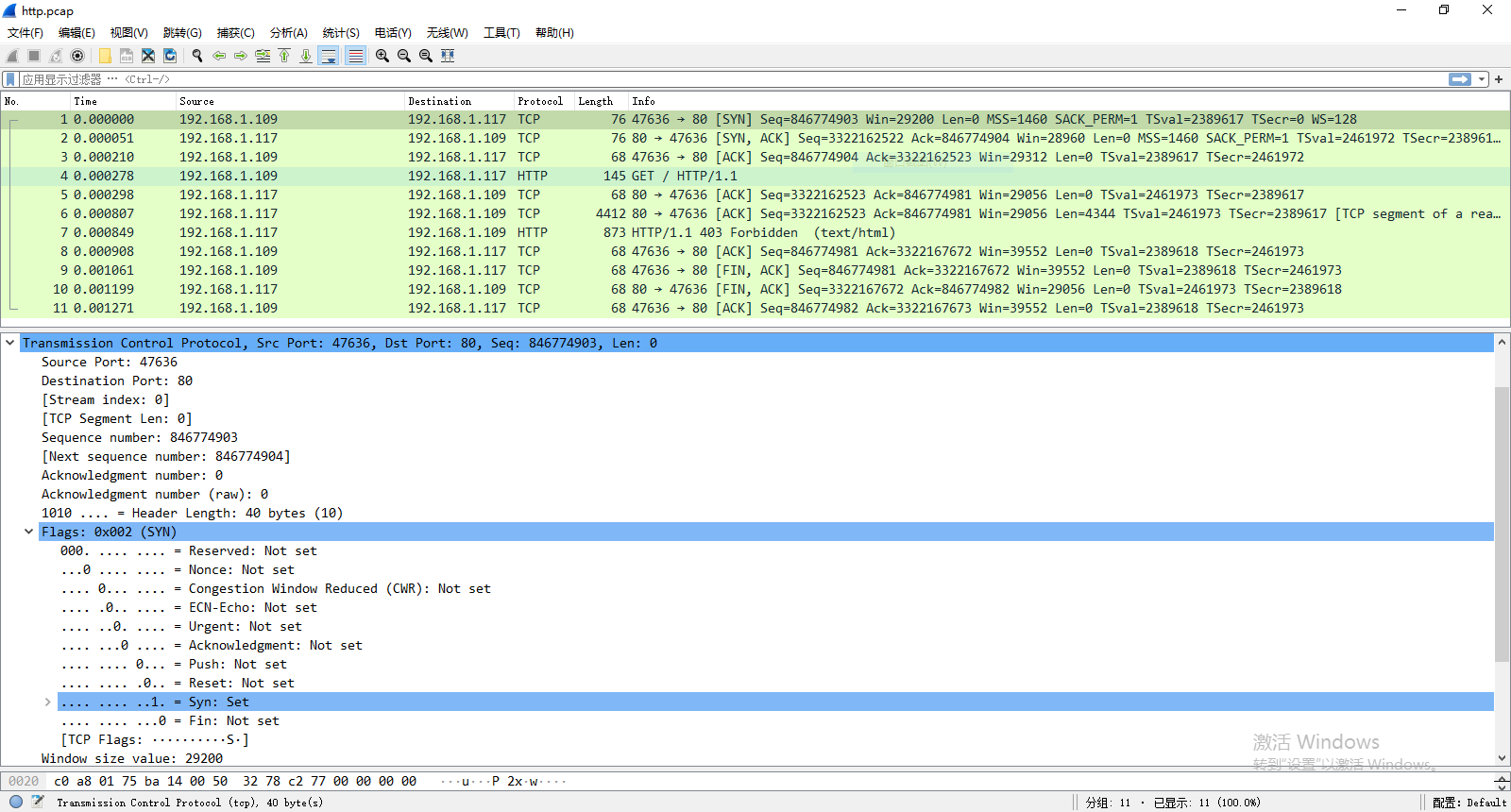


图6 TCP第一次握手

从图6中可以看到第一次握手时报文中SYN=1，ACK=0，FIN=0，Sequence number =846774903，Acknowledgement number=0；

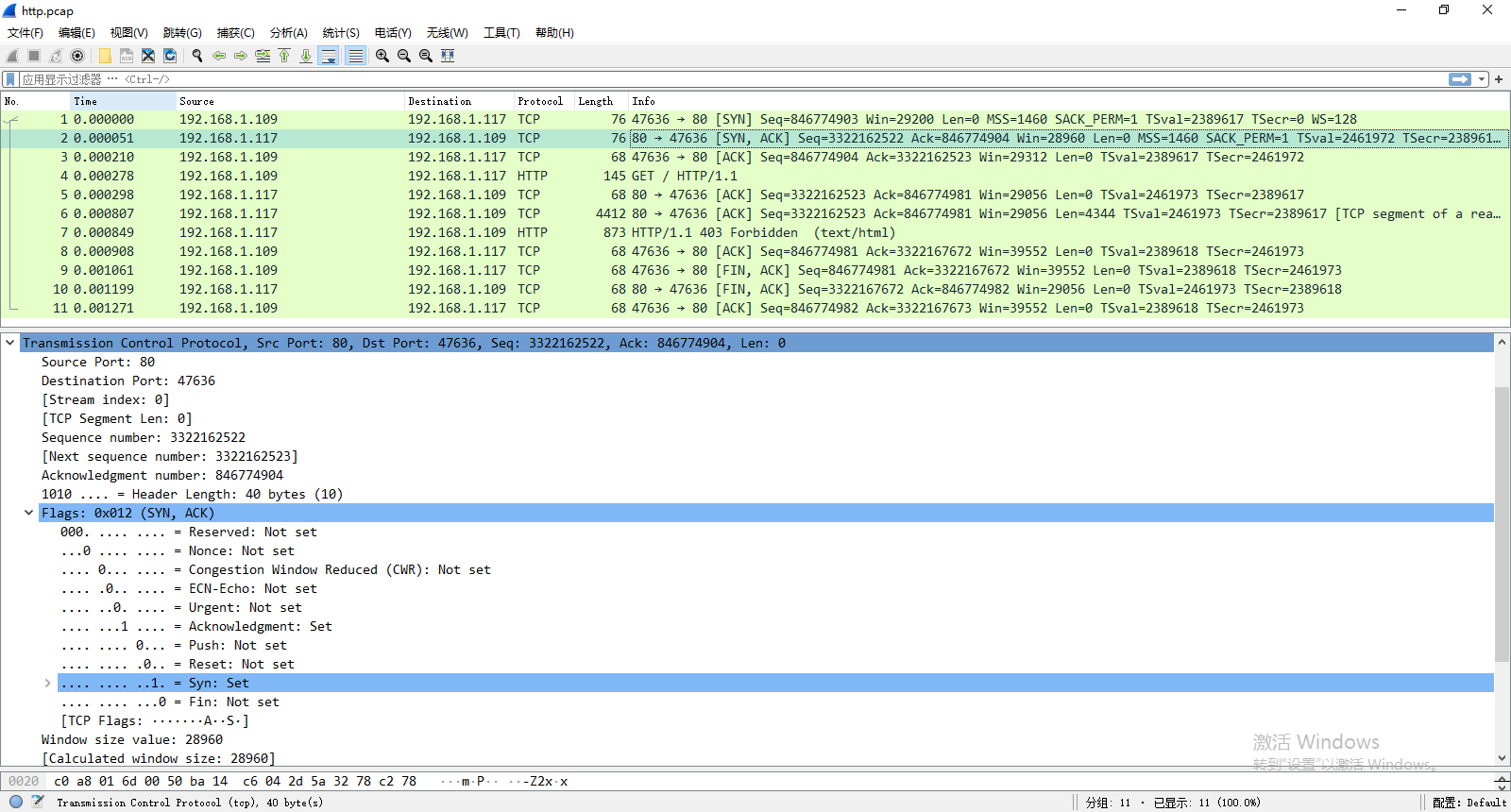


图7 TCP第二次握手

从图7中可以看到第一次握手时报文中SYN=1，ACK=1，FIN=0，Sequence number =3322162522，Acknowledgement number=846774904；

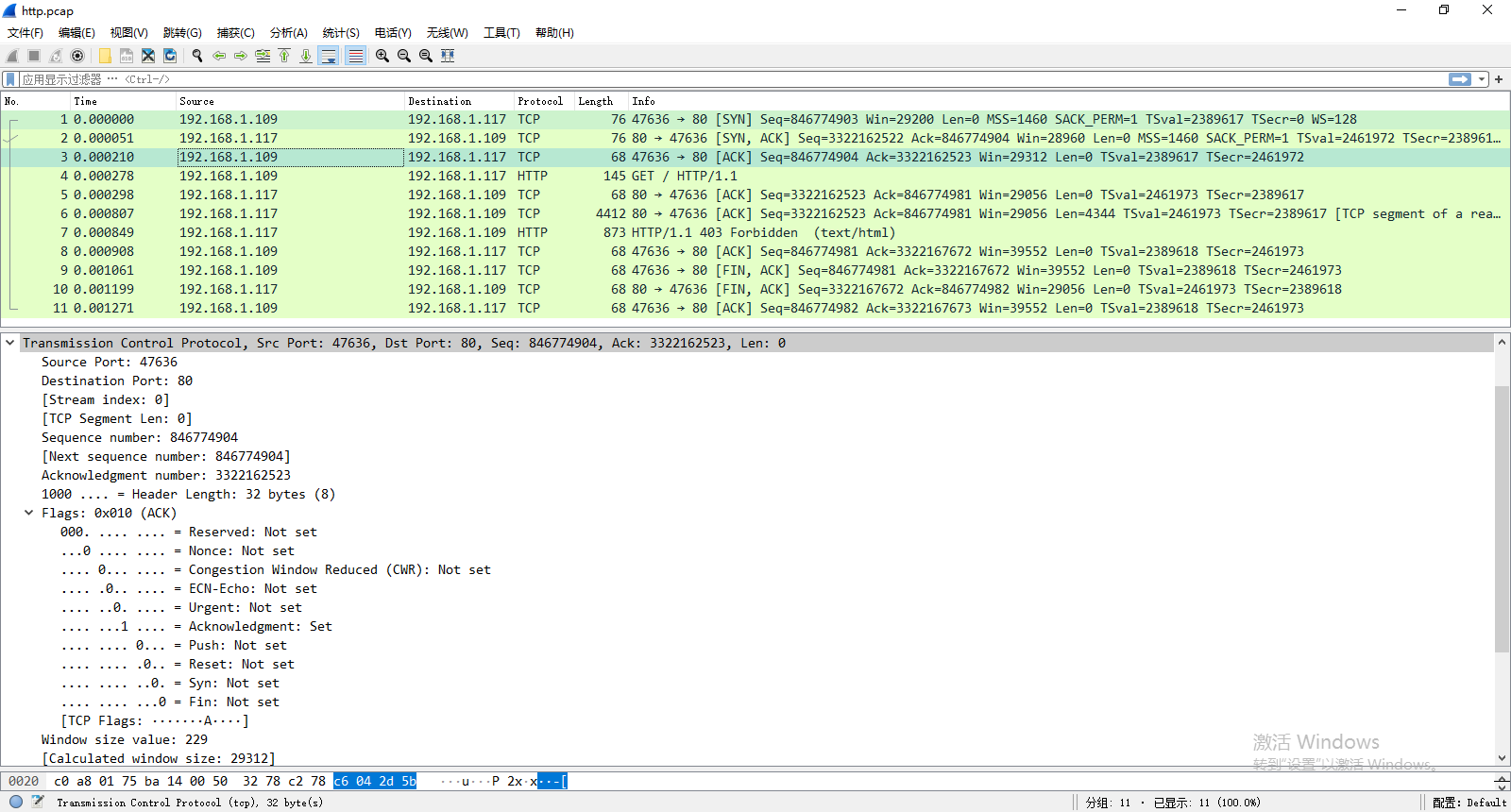


图8 TCP第三次握手

从图8中可以看到第一次握手时报文中SYN=0，ACK=1，FIN=0，Sequence number =846774904，Acknowledgement number=3322162523；

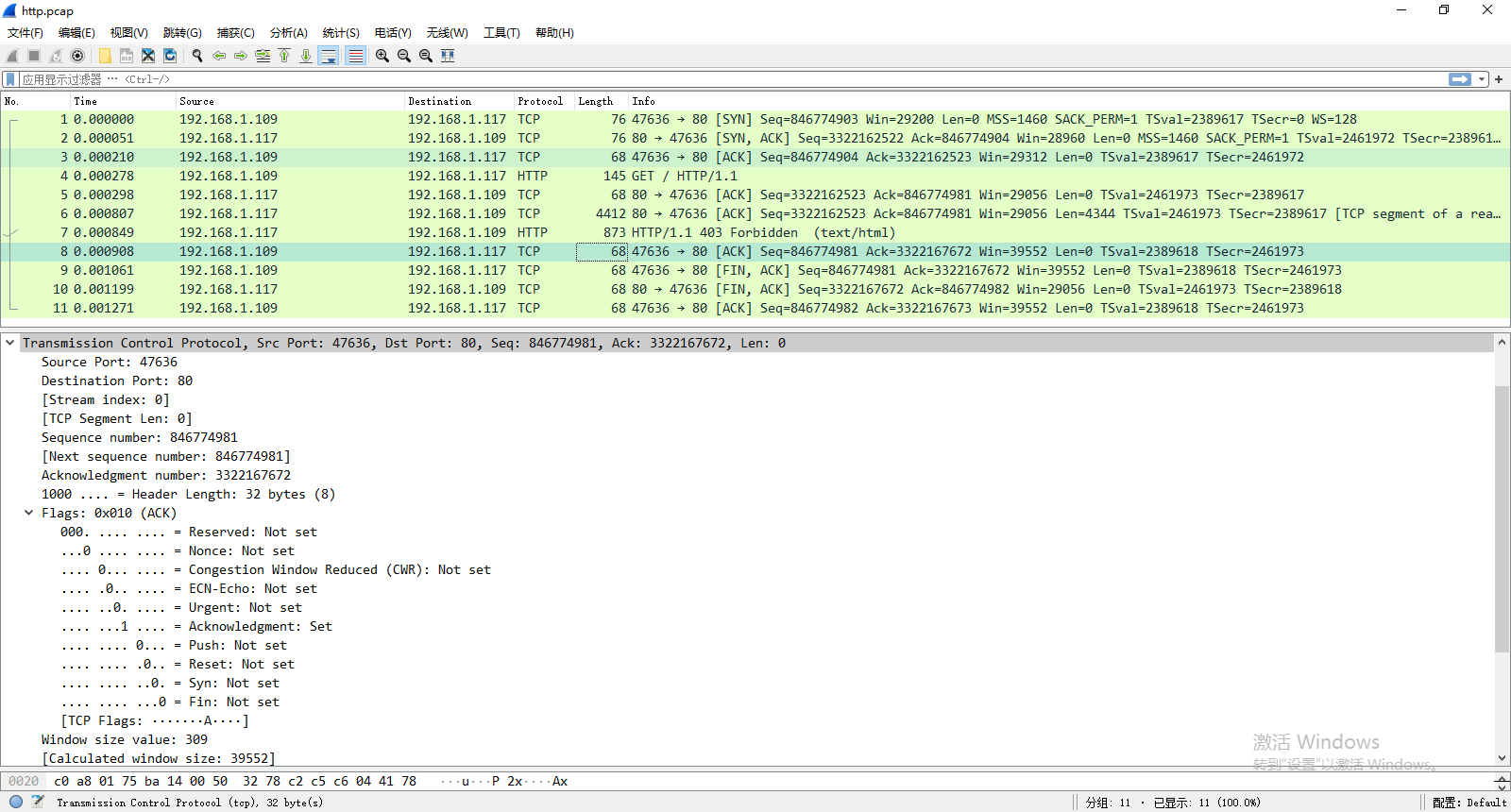


图9 TCP第一次挥手

从图9中可以看到第一次挥手时报文中SYN=0，ACK=1，FIN=0，Sequence number =846774981，Acknowledgement number=3322167672；

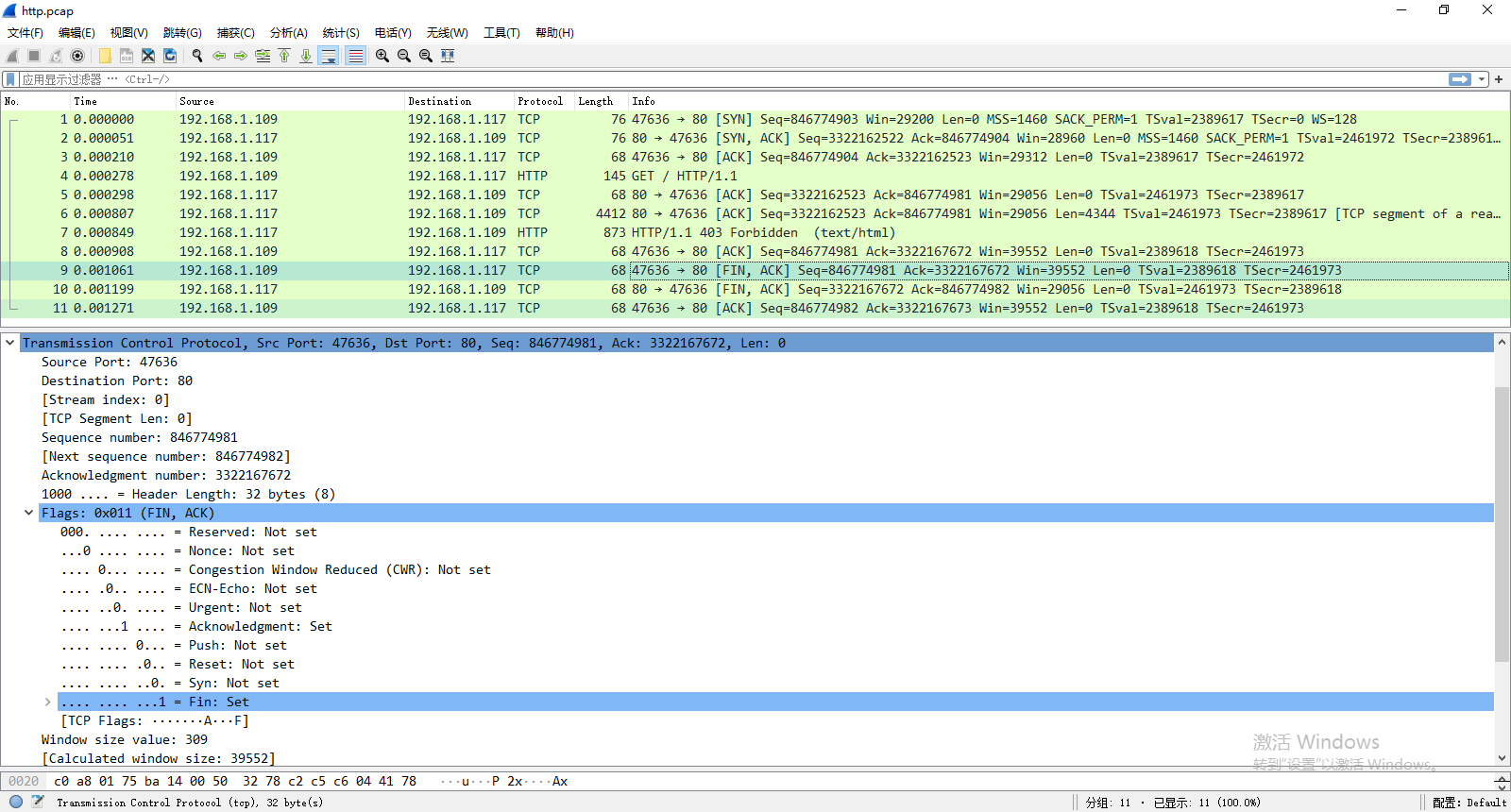


图10 TCP第二次挥手

从图10中可以看到第二次挥手时报文中SYN=0，ACK=1，FIN=1，Sequence number =846774981，Acknowledgement number=3322167672；

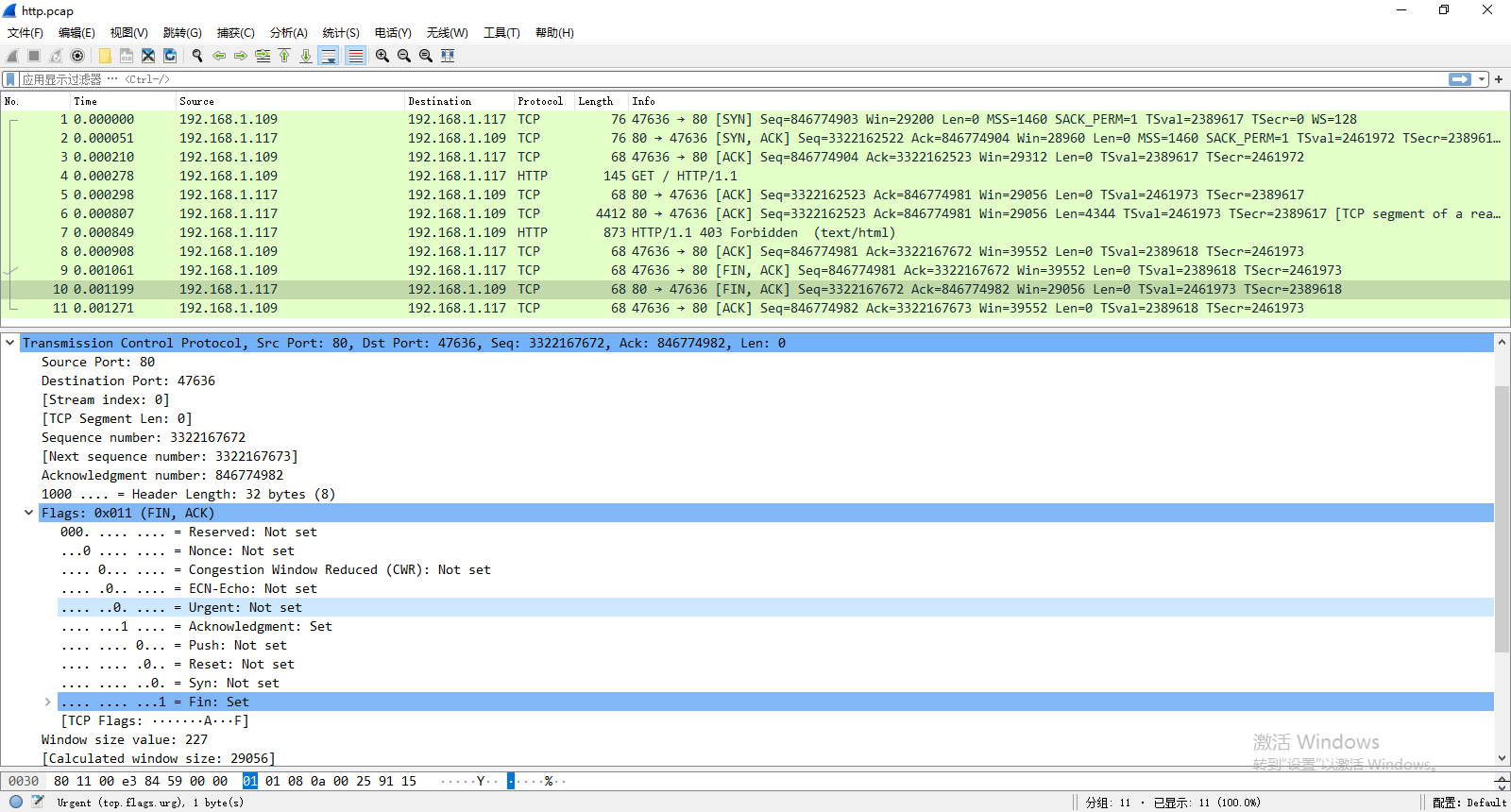


图11 TCP第三次挥手

从图11中可以看到第三次挥手时报文中SYN=0，ACK=1，FIN=1，Sequence number =3322167672，Acknowledgement number=846774982；

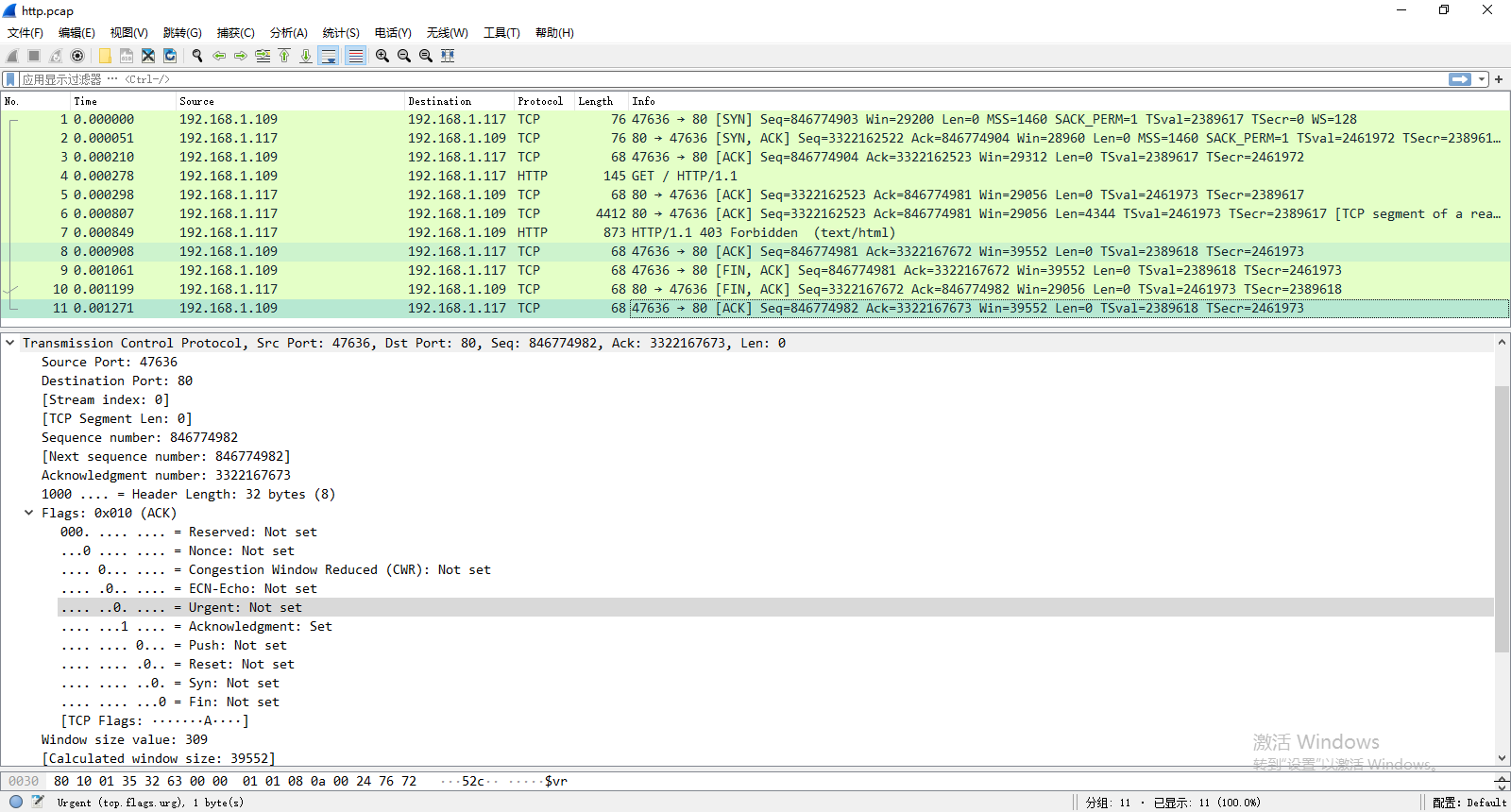


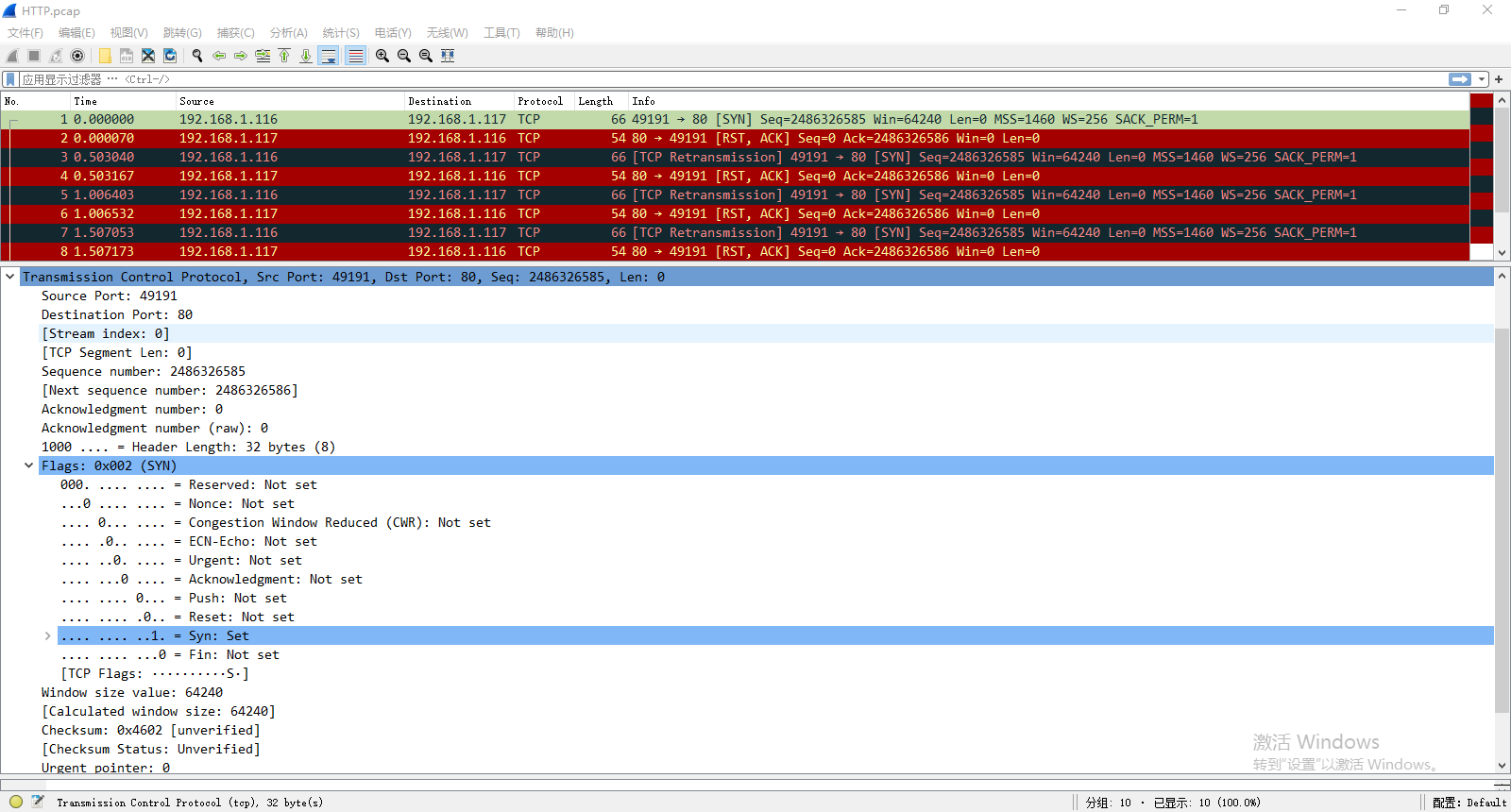
图12 TCP第四次挥手

从图12中可以看到第四次挥手时报文中SYN=0，ACK=1，FIN=0，Sequence number =846774982，Acknowledgement number=3322167673；

分析以上报文可以看出SYN的作用是表示SYN请求报文，ACK的作用表示ACK应答报文，FIN表示是否有数据需要发送，另外Sequence number是发送的数据包中第一个字节的序列号，Acknowledgement number是确认序列号。结合三次握手和四次挥手报文结果中Sequence number值和Acknowledgement number值的变化，可以发现Sequence number表示的是发送方packet的数据部分的第一位在整个data stream中所在的位置，因为对于没有数据的传输，如ACK，虽然它有一个Sequence number，但这次传输在整个data stream中不占位置，所以下一个实际有数据的传输，会依旧从上一次发送ACK的数据包的Sequence number开始，而Acknowledge number表示的接收方下一次Sequence number的值，这里SYN/FIN的传输虽然没有data，但是会让下一次packet传输的Sequence number增加1，但是，ACK的传输不会让下一次packet的传输加1。

**实验C**

问题一：TCP 第一次握手的 SYN 丢包了，会发生了什么？

图13 握手请求报文

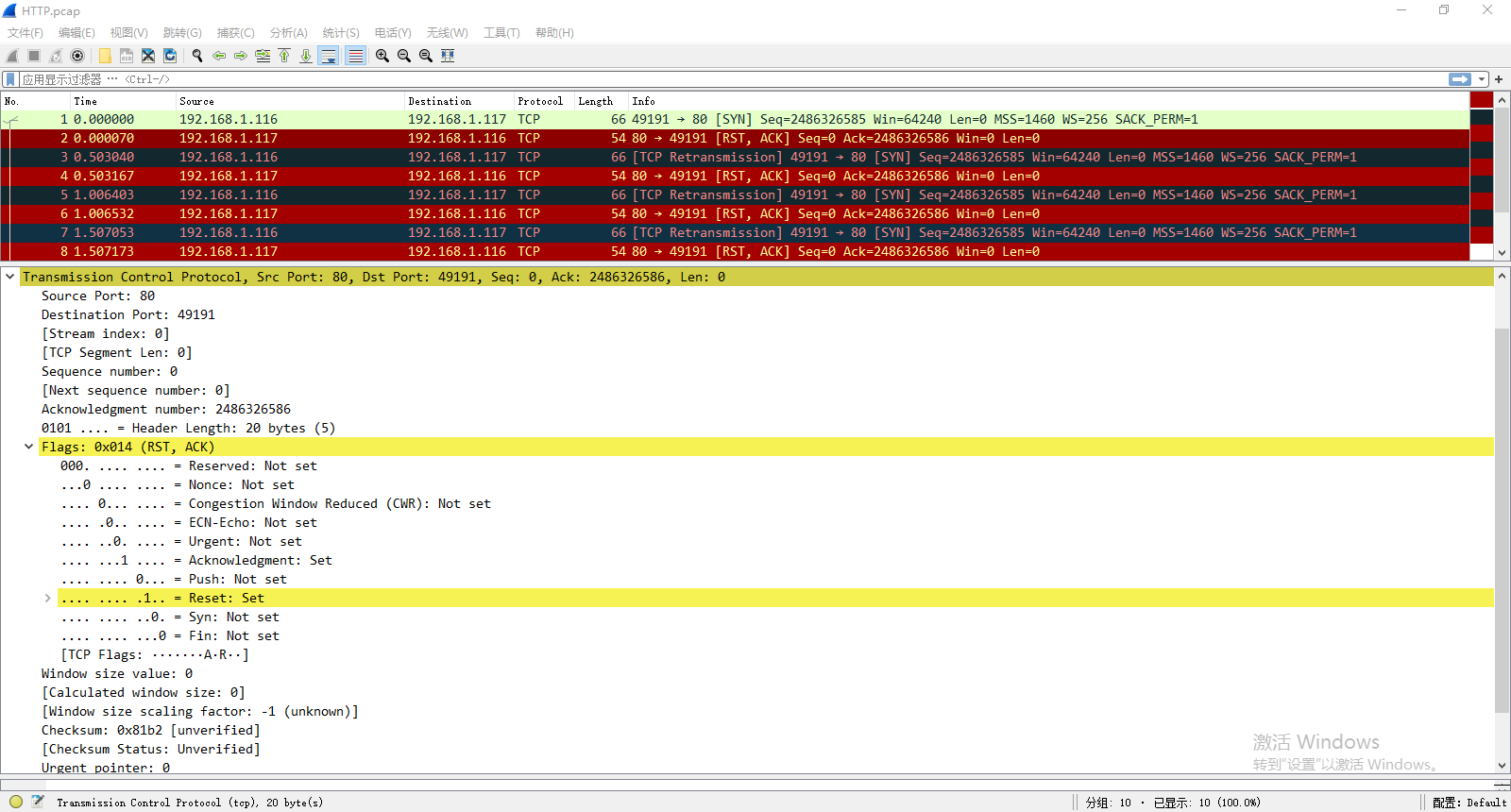


图14 握手响应报文

图13，14为第一次握手SYN丢包后的情况，与正常TCP通讯相比，当TCP 第一次握手的 SYN 丢包后，应答报文中reset值变为1，SYN值变为0，且重复传输多次，通讯无法建立，客户端在超时时间内没收到服务端的 ACK，就会重传数据包，每次超时重传的 RTO 是翻倍上涨的，直到 SYN 包的重传次数到tcp\_syn\_retries值后，客户端不再发送 SYN 包。

问题二：TCP 第二次握手的 SYN、ACK 丢包了，会发生什么？

当 TCP 第二次握手 SYN、ACK 包丢了后，客户端 SYN 包会发生超时重传，服务端 SYN、ACK 也会发生超时重传。客户端 SYN 包超时重传的最大次数，是由 tcp\_syn\_retries 决定的；服务端 SYN、ACK 包时重传的最大次数，是由 tcp\_synack\_retries 决定的。

问题三：TCP 第三次握手的 ACK 包丢了，会发生什么？

如果第三次握手的 ACK，服务端无法收到，则服务端就会短暂处于 SYN\_RECV 状态，而客户端会处于 ESTABLISHED 状态。由于服务端一直收不到 TCP 第三次握手的 ACK，则会一直重传 SYN、ACK 包，直到重传次数超过 tcp\_synack\_retries 值后，服务端就会断开 TCP 连接。而客户端有两种情况，如果客户端没发送数据包，一直处于 ESTABLISHED 状态，然后经过几个小时才可以发现一个「死亡」连接，于是客户端连接就会断开连接。如果客户端发送了数据包，一直没有收到服务端对该数据包的确认报文，则会一直重传该数据包，直到重传次数超过 tcp\_retries2 值后，客户端就会断开 TCP 连接。

**四、参考文献**

[1]ICMP协议：<https://baike.baidu.com/item/ICMP/572452?fr=aladdin>；

[2]搭建http协议服务：<https://www.cnblogs.com/snake553/p/8856729.html>；

[3]tcpdump抓包实现：<http://www.hansongda.cn/post/tcpdumpshakes>；

[4]TCP协议详解：<https://blog.csdn.net/qq_37884273/article/details/82188586>