## 实验6用户数据报协议（[TCP](https://so.csdn.net/so/search?q=UDP&spm=1001.2101.3001.7020)）

实验环境：拓扑结构一  
实验目的：

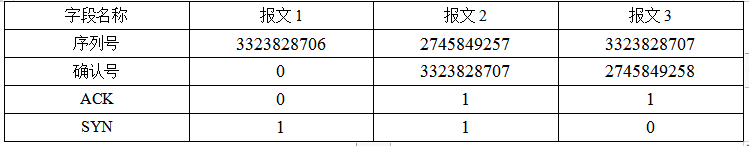
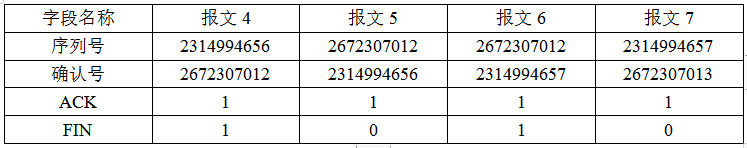
1. 掌握TCP协议的报文格式
2. 掌握TCP连接的建立和释放过程
3. 掌握TCP数据传输中编号与确认的过程
4. 掌握TCP协议校验和的计算方法
5. 理解TCP重传机制  
   实验原理：  
   一. TCP协议简介  
   二. TCP报文格式  
   三. TCP封装  
   四. TCP校验和  
   五. TCP连接建立与释放
   * 连接建立
   * 三次握手
   * 连接终止
   * 三次握手方式终止连接
   * 半关闭的四次握手方式终止连接  
     六. 流量控制
   * 滑动窗口协议  
     七. 差错控制
   * 校验和
   * 确认
   * 重传  
     实验步骤：  
     练习1（察看TCP连接的建立和释放）：  
     各主机打开协议分析器，进入相应的网络结构并验证网络拓扑的正确性，如果通过拓扑验证，关闭协议分析器继续进行实验，如果没有通过拓扑验证，请检查网络连接。  
     本练习将主机A、B、C、D、E、F作为一组进行实验。
6. 主机B、C、D启动协议分析器捕获数据，并设置过滤条件（提取TCP协议）。
7. 主机A启动TCP工具连接主机C。  
   （1）主机A启动“实验平台工具栏中的地址本工具”。点击[主机扫描]按钮获取组内主机信息，选中主机C点击[端口扫描]按钮获取主机C的TCP端口列表。  
   （2）主机A启动“实验平台工具栏中的TCP工具”。选中“客户端”单选框，在“地址”文本框中填入主机C的IP地址，在“端口”文本框中填入主机C的一个TCP端口，点击[连接]按钮进行连接。
8. 察看主机B、C、D捕获的数据，填写下表。  
   

表7-3 实验结果

● TCP连接建立时，前两个报文的首部都有一个“最大字段长度”字段，它的值是多少？作用是什么？结合IEEE802.3协议规定的以太网最大帧长度分析此数据是怎样得出的。  
**值为1460；  
它是由发送端指定，表示能在网络上传输的最大的段尺寸  
最大帧长度= MTU-IP首部长度20- TCP首部长度20**  
4. 主机A断开与主机C的TCP连接。  
5. 察看主机B、C、D捕获的数据，填写下表。

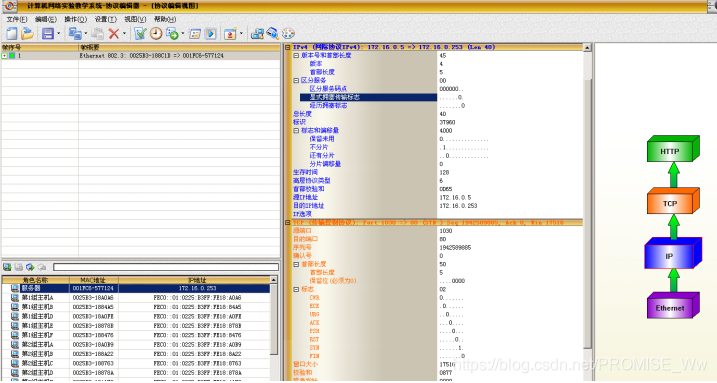
表7-4 实验结果



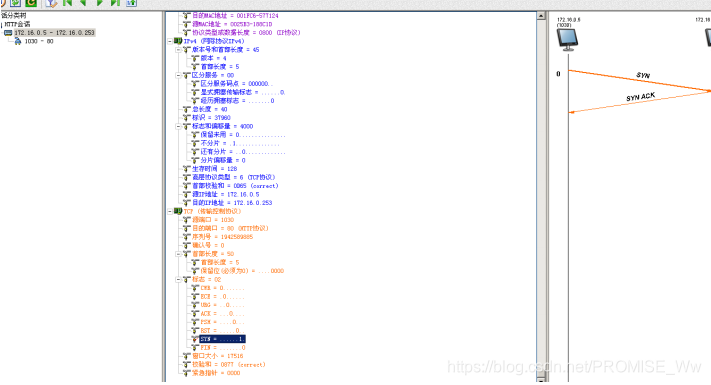
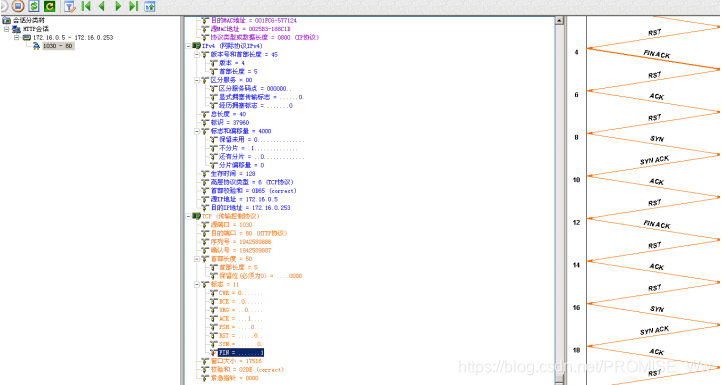
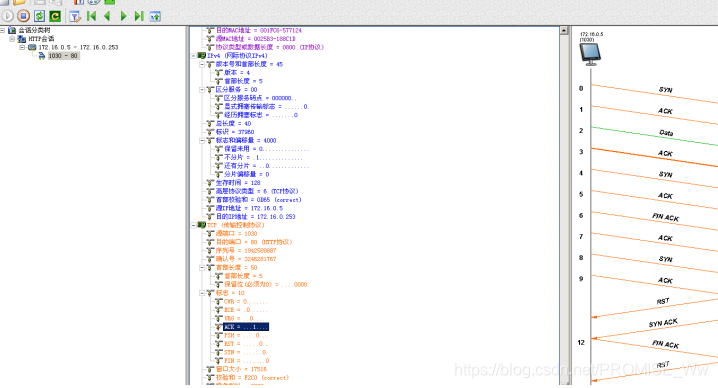
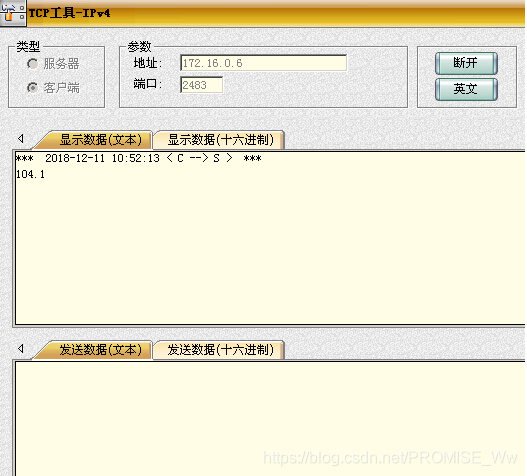
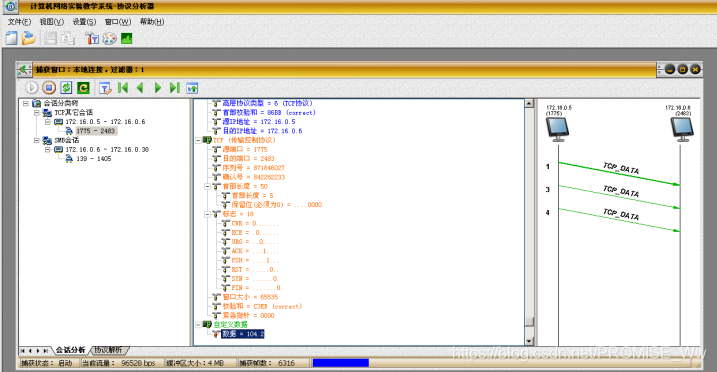
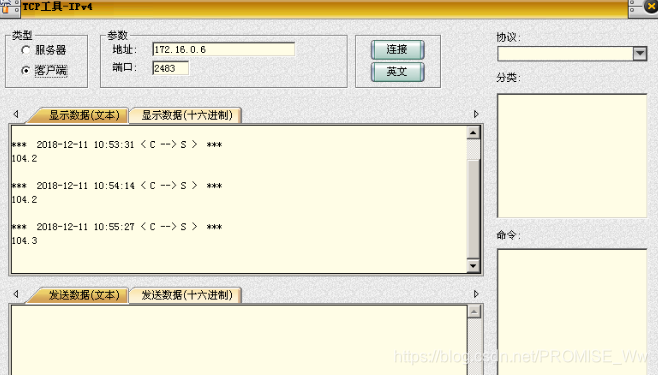
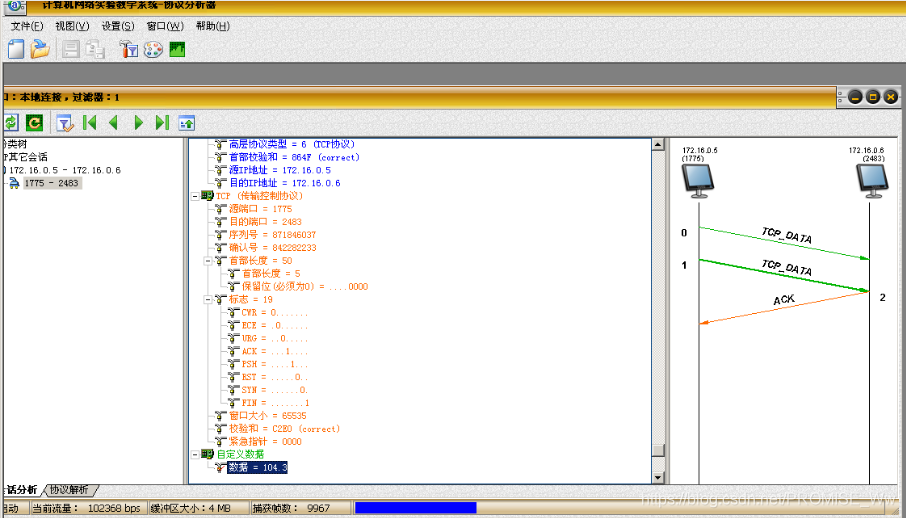
● 结合步骤3、5所填的表，理解TCP的三次握手建立连接和四次握手的释放连接过程，理解序号、确认号等字段在TCP可靠连接中所起的作用。

* 1

练习2（利用协议编辑器编辑并发送TCP数据包）：  
本练习将主机A和B作为一组，主机C和D作为一组，主机E和F作为一组。现仅以主机A、B所在组为例，其它组的操作参考主机A、B所在组的操作。  
在本实验中由于TCP连接有超时时间的限制，故协议编辑器和协议分析器的两位同学要默契配合，某些步骤（如计算TCP校验和）要求熟练、迅速。  
为了实现TCP[三次握手](https://so.csdn.net/so/search?q=%E4%B8%89%E6%AC%A1%E6%8F%A1%E6%89%8B&spm=1001.2101.3001.7020)过程的仿真，发送第一个连接请求帧之前，编辑端主机应该使用TCP屏蔽功能来防止系统干扰（否则计算机系统的网络会对该请求帧的应答帧发出拒绝响应）。  
通过手工编辑TCP数据包实验，要求理解实现TCP连接建立、数据传输以及断开连接的全过程。在编辑过程中注意体会TCP首部中的序列号和标志位的作用。  
首先选择服务器主机上的一个进程作服务器进程，并向该服务器进程发送一个建立连接请求报文，对应答的确认报文和断开连接的报文也编辑发送。其步骤如下：

1. 主机B启动协议分析器捕获数据，设置过滤条件（提取HTTP协议）。
2. 主机A上启动协议编辑器，在界面初始状态下，程序会自动新建一个单帧，可以利用协议编辑器打开时默认的以太网帧进行编辑。
3. 填写该帧的以太网协议首部，其中：  
   源MAC地址：主机A的MAC地址  
   目的MAC地址：服务器的MAC地址  
   协议类型或数据长度：0800（IP协议）
4. 填写IP协议头信息，其中：  
   高层协议类型：6（上层协议为TCP）  
   总长度：40（IP首部 + TCP首部）  
   源IP地址：主机A的IP地址  
   目的IP地址：服务器的IP地址（默认为172.16.0.253）  
   其它字段任意。  
   应用前面学到的知识计算IP首部校验和。
5. 填写TCP协议信息，其中：  
   源端口：任意大于1024的数，不要使用下拉列表中的端口  
   目的端口：80（HTTP协议）  
   序列号：选择一个序号ISN（假设1942589885），以后的数据都根据它来填写  
   确认号：0  
   首部长度：50（长度20字节）  
   标志位：02（标志SYN=1）  
   窗口大小：任意  
   紧急指针：0  
   使用协议编辑器的“手动计算”方法计算校验和；再使用协议编辑器的“自动计算”方法计算校验和。将两次计算结果相比较，若结果不一致，则重新计算。  
   

● TCP在计算校验和时包括哪些内容？  
**伪首部、TCP首部、应用层传来的数据。**

1. 将设置完成的数据帧复制3份。  
   修改第二帧的TCP 层的“标志”位为10（即标志位ACK=1），TCP层的“序列号”为1942589885+1。  
   修改第三帧的TCP层的“标志”位为11（即标志位ACK=1、FIN=1），TCP层的“序列号”为1942589885+1。  
   修改第四帧的TCP层的“标志”位为10（即标志位ACK=1），TCP层的“序列号”为1942589885+2。
2. 在发送该TCP连接请求之前，先ping一次目标服务器，让目标服务器知道自己的MAC地址。
3. 启动“实验平台工具栏中的启动屏蔽”，为TCP/IP协议栈过滤掉收到的TCP数据。
4. 点击菜单栏中的[发送]按钮，在弹出对话框中选择发送第一帧。  
   
5. 在主机B上捕获相应的应答报文，这里要求协议分析器一端的同学及时准确地捕获应答报文并迅速从中获得应答报文的接收字节序列号，并告知协议编辑器一端的同学。
6. 假设接收字节序号为：3246281765，修改第二帧和第三帧TCP层的“确认号”的值为：3246281766。
7. 计算第二帧的TCP校验和，将该帧发送。对服务器的应答报文进行确认。  
   
8. 计算第三帧的TCP校验和，将该帧发送。  
   
9. 在主机B上观察应答报文，要及时把最后一帧“序列号”告知协议编辑器一端的同学。
10. 修改第四帧的TCP层“确认号”为接收的序列号+1（即3246281767）。
11. 计算第四帧的TCP校验和，将该帧发送。断开连接，完成TCP连接的全过程。  
    
12. 协议分析器一端截获相应的请求及应答报文并分析，注意观察“会话分析”中的会话过程。
13. 编辑端主机启动“实验平台工具栏中的停止屏蔽”，恢复正常网络功能。  
    练习3（TCP的重传机制）：  
    本练习将主机A和B作为一组，主机C和D作为一组，主机E和F作为一组。现仅以主机A、B所在组为例，其它组的操作参考主机A、B所在组的操作。
14. 主机B上启动“实验平台工具栏中的TCP工具”，作为服务端，监听端口设置为2483。
15. 主机B启动协议分析器开始捕获数据并设置过滤条件（提取TCP协议）。
16. 主机A启动TCP工具连接主机B。  
    （1）主机A启动“实验平台工具栏中的TCP工具”。  
    （2）选中“客户端”单选框。  
    （3）在“地址”文本框中填入主机B的IP地址。  
    （4）在“端口”文本框中填入主机B的TCP监听端口（2483）。  
    （5）点击[连接]按钮进行连接。
17. 主机A向主机B发送一条信息。  
    
18. 主机B启动“实验平台工具栏中的启动TCP屏蔽”，过滤掉接收到的TCP数据。
19. 主机A向主机B再发送一条信息。
20. 主机B刷新捕获显示，当发现“会话分析视图”中有两条以上超时重传报文后，启动“实验平台工具栏中的停止TCP屏蔽”，恢复正常网络功能。  
      
    
21. 主机A向主机B再发送一条信息，之后断开连接。  
    
22. 主机B停止捕获数据。依据“会话分析视图”显示结果，绘制本练习的数据报交互图。  
    

思考问题：

1. 为什么在TCP连接过程中要使用三次握手？如不这样做可能会出现什么情况。  
   3次握手完成两个重要的功能，既要双方做好发送数据的准备工作(双方都知道彼此已准备好)，也要允许双方就初始序列号进行协商，这个序列号在握手过程中被发送和确认。现在把三次握手改成仅需要两次握手，死锁是可能发生的。
2. 解释TCP协议的释放过程？  
   单方主动关闭的TCP连接释放过程：  
   1）一开始，通信双方都处于ESTABLISHED（连接建立）状态。如果客户端认为数据全部发送完了，想结束本次传输连接，则由应用层的对应应用进程调用CLOSE服务原语，然后向服务器发出一个FIN字段置1的数据段（假设此数据段的序号为m），客户端进入FIN WAIT 1状态，等待服务器的确认。  
   2）服务器在收到客户端发来的FIN数据段后，确认客户端没有新的数据要发送了，向客户端发送一个ACK字段置1，确认号为m+1（假设此数据段序号为w，服务器与客户端的数据段序号可以不一样），表示前面的数据已全部收到了，然后进入到CLOSE WAIT（关闭等待）状态。与此同时服务器的TCP实体通知对应的应用层进程，释放从客户机到服务器方向的传输连接，进入半关闭状态。但此时服务器仍可以向客户端发送数据段；客户端也可接收来自服务器的数据。而且这可能要持续一段时间，直到服务器的数据也全部发送完。  
   3）当客户端收到服务器的ACK数据段后便进入到了FIN WAIT 2状态，进一步等待服务器发出连接释放的数据段。  
   4）当服务器发送完全部的数据后，其对应的应用进程也会通知TCP实体释放此方向的TCP传输连接，向客户机发送FIN字段置1，ACK字段置1，ack=m+1（假设此时的数据段序号已变为w）的确认数据段。这时服务器进入LAST ACK（最后确认）状态，等待客户端的确认。  
   5）客户端在收到服务器的FIN+ACK数据段后，向服务器发送一个ACK字段置1，ack=w+1，序列号为m+1的数据段，进入到TIME WAIT状态。但此时TCP连接还没有释放，必须等待2MSL时间（RFC 793建议设MSL为2分钟）后，客户端才进入到CLOSED状态，彻底释放了TCP连接。  
   6）服务器在收到客户端发来的ACK数据段后，也进入CLOSED状态，彻底释放连接。完成整个TCP传输连接释放过程。  
   双方主动关闭的TCP连接释放流程：  
   当两端对应的网络应用层进程同时调用CLOSE原语，发送FIN数据段执行关闭命令时，两端均从ESTABLISHED状态转变为FIN WAIT 1状态。任意一方收到对端发来的FIN数据段后，其状态均由FIN WAIT 1转变到CLOSING状态，并发送最后的ACK数据段。当收到最后的ACK数据段后，状态转变化TIME\_WAIT，在等待2MSL后进入到CLOSED状态，最终释放整个TCP传输连接。
3. 使用TCP协议对实时话音数据的传输有什么有缺点？使用UDP协议在传送数据文件时会有什么问题？  
   1）如果语音数据不是实时播放(边接受边播放)就可以使用TCP,因为TCP传输可靠。接收端用TCP讲话音数据接受完毕后，可以在以后的任何时间进行播放。但假定是实时传输，则必须使用UDP。  
   2） UDP不保证可靠交付，但UC P比TC P的开销要小很多。因此只要应用程序接受这样的服务质量就可以使用UDP。  
   4.TCP协议在进行流量控制时是以数据包丢失作为产生拥塞的标志。有没有不是因拥塞而引起的数据包丢失的情况？如有，请列举出三种情况。  
   当IP数据报在传输过程中需要分片，但其中的一个数据报未能及时到达终点，而终点组装IP数据报已超时，因而只能丢失该数据报; IP数据报已经到达终点，但终点的缓存没有足够的空间存放此数据报，数据报在转发过程中经过一个局域网的网桥，但网桥在转发该数据报的帧没有足够的差错空间而只好丢弃。