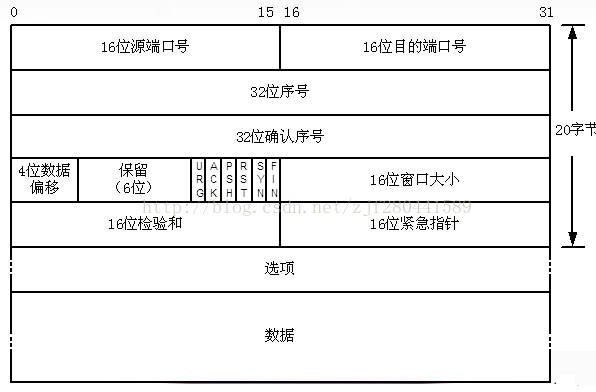
TCP首部格式

**tcp数据是被封装在IP数据包中的，和udp类似，在IP数据包的数据部分。tcp数据包的格式如下：**　　 

**源端口号和目的端口号（寻址）**与udp中类似，用于寻找发端和收端应用进程。**这两个值加上IP首部中的源端IP地址和目的端IP地址唯一确定一个，**在网络编程中**，一般一个IP地址和一个端口号组合称为一个套接字（socket）。**   
　　**序号（seq）：**用来**标识从TCP发端向TCP收端发送**的数据字节流，它**表示在这个报文段中的第一个数据字节。**在tcp中tcp用序号对每个字节进行计数（这个值与发送的帧数没有关系，而是与发送的数据字节数有关系，后面会有说明）。   
　　**确认序号（seq+1）：**包含**发送确认的一端所期望收到的下一个序号**。因此，确认序号应当是上次已成功收到**数据字节序号加 1（不是单纯的序号加1，还包括数据字节数）。**

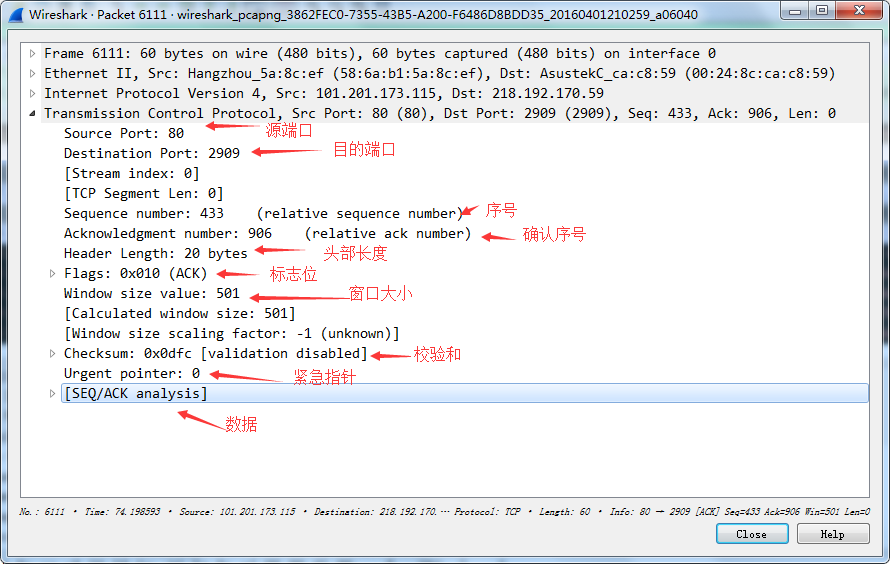
**首部长度：用于记录tcp数据报首部的长度，一般为20字节，实际值为首部长度除以4。**

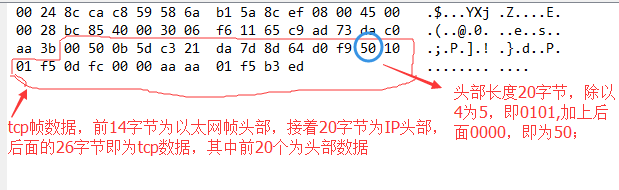
下面的是标志位对应的功能：  
　　URG： 紧急指针（ urgent pointer）有效。   
　　ACK： 确认序号有效。   
　　PSH： 接收方应该尽快将这个报文段交给应用层。   
　　RST： 重建连接。   
　　SYN： 同步序号用来发起一个连接。   
　　FIN： 发端完成发送任务。   
　　**窗口大小**：用于流量控制。   
　**检验和**：检验和覆盖了整个的 TCP报文段： TCP首部和TCP数据，与udp相似需要计算伪首部。

（1）每个TCP段都包括源端和目的端的端口号**，用于寻找发送端和接收端的应用进程**。这两个值加上IP首部的源端IP地址和目的端IP地址唯一确定一个TCP连接。   
（2）序号用来标识从TCP发送端向接收端发送的数据字节流，它表示在这个报文段中的第一个数据字节。如果将字节流看作在两个应用程序间的单向流动，则TCP用序号对每个字节进行计数**(seq)**。   
（3）当建立一个新连接时，SYN标志变1。序号字段包含由这个主机选择的该连接的初始序号ISN，该主机要发送数据的第一个字节的序号为这个ISN加1，因为SYN标志使用了一个序号。   
（4）既然每个被传输的字节都被计数，确认序号包含发送确认的一端所期望收到的下一个序号。因此，确认序号应当时上次已成功收到数据字节序号加1。只有ACK标志为1时确认序号字段才有效。**(ack)**   
（5）发送ACK无需任何代价，因为32位的确认序号字段和ACK标志一样，总是TCP首部的一部分。因此一旦一个连接建立起来，这个字段总是被设置，ACK标志也总是被设置为1。   
（6）**TCP为应用层提供全双工的服务**。因此，连接的每一端必须保持每个方向上的传输数据序号。   
（7）TCP可以表述为一个没有选择确认或否认的**滑动窗口协议**。因此TCP首部中的确认序号表示发送方已成功收到字节，但还不包含确认序号所指的字节。当前还无法对数据流中选定的部分进行确认。   
（8）首部长度需要设置，因为任选字段的长度是可变的。TCP首部最多60个字节。   
（9）6个标志位中的**多个可同时设置为1**   
◆ URG－紧急指针有效   
◆ ACK－确认序号有效   
◆ PSH－接收方应尽快将这个报文段交给应用层   
◆ RST－重建连接   
◆ SYN－同步序号用来发起一个连接   
◆ FIN－发送端完成发送任务   
（10）TCP的[**流量控制**](https://www.baidu.com/s?wd=%E6%B5%81%E9%87%8F%E6%8E%A7%E5%88%B6&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWrAN-m1w9Pycvmh79uWnd0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPjDdrjbsPWTk)**由连接的每一端通过声明的窗口大小**来提供。窗口大小为字节数，起始于确认序号字段指明的值，这个值是接收端期望接收的字节数。窗口大小是一个16为的字段，因而窗口大小最大为65535字节。   
（11）**检验和覆盖整个TCP报文端：TCP首部和TCP数据。**这是一个强制性的字段，一定是由发送端计算和存储，并由接收端进行验证。TCP检验和的计算和UDP首部检验和的计算一样，也使用伪首部。   
（12）**紧急指针是一个正的偏移量**，序号字段中的值相加表示紧急数据最后一个字节的序号。TCP的紧急方式是发送端向另一端**发送紧急数据的一种方式。**   
（13）最常见的可选字段是**最长报文大小MMS，**每个连接方通常都在通信的第一个报文段中指明这个选项。它指明本端所能接收的最大长度的报文段。

Wireshark抓包分析TCP结构

　　利用wireshark抓取一个tcp数据包，查看其具体数据结构和实际的数据：





TCP：传输控制协议

　　TCP是一种面向连接的、可靠的、基于字节流的传输层通信协议。   
　　面向连接： 面向连接意味着使用tcp的应用程序在传输数据前必须先建立连接，就如打电话一样，要先进行拨号，等待对方响应才能开始说话。   
　　可靠性：tcp协议通过下列方式来提高可靠性： 

* 1.应用**数据被分割成TCP认为最适合发送的数据块**。这和UDP完全不同，应用程序产生的数据报长度将保持不变。由TCP传递给IP的信息单位称为报文段或段。**（定长）**
* 2.当TCP发出一个段后，它启动一个定时器，等待目的端确认收到这个报文段。如果不能及时收到一个确认，将重发这个报文段。**（定时重发）**
* 3.当TCP收到发自TCP连接另一端的数据，它将发送一个确认。这个确认不是立即发送，通常将推迟几分之一秒。**（确认机制）**
* 4.TCP将保持它首部和数据的检验和。这是一个端到端的检验和，目的是检测数据在传输过程中的任何变化。如果收到段的检验和有差错，TCP将丢弃这个报文段和不确认收到此报文段（希望发端超时并重发）。**（检验和较检）**
* 5.既然TCP报文段作为IP数据报来传输，而IP数据报的到达可能会失序，因此TCP报文段的到达也可能会失序。如果必要，TCP将对收到的数据进行重新排序，将收到的数据以正确的顺序交给应用层。**（必要时重新排序）**
* 6.既然I P数据报会发生重复，TCP的接收端必须丢弃重复的数据。**（去重）**
* 7.TCP还能提供流量控制。TCP连接的每一方都有固定大小的缓冲空间。TCP的接收端只允许另一端发送接收端缓冲区所能接纳的数据。这将防止较快主机致使较慢主机的缓冲区溢出。**（流量控制）**

字节流：两个应用程序通过TCP连接交换8 bit字节构成的字节流。

　　另外，TCP对字节流的内容不作任何解释。TCP不知道传输的数据字节流是二进制数据，还是ASCII字符或者其他类型数据。对字节流的解释由TCP连接双方的应用层解释。