

TCP/IP、Http、Socket的区别

2015-12-15 iOS开发

网络由下往上分为：

物理层、数据链路层、网络层、传输层、会话层、表示层和应用层。

OSI		TCP/IP协议集
应用层	应用层	Telnet, FTP, SMTP, DNS, HTTP 以及其他应用协议
表示层		
会话层		
传输层	传输层	TCP, UDP
网络层	网络层	IP, ARP, RARP, ICMP
数据链路层	网络接口	各种通信网络接口（以太网等） （物理网络）
物理层		

如上图，IP协议对应于网络层；TCP协议对应于传输层；而HTTP协议对应于应用层；

三者从本质上来说没有可比性，socket则是对TCP/IP协议的封装和应用(程序员层面上)。

也可以说：

- TCP/IP协议是传输层协议，主要解决数据如何在网络中传输；
- 而HTTP是应用层协议，主要解决如何包装数据。

关于TCP/IP和HTTP协议的关系，网络有一段比较容易理解的介绍：

我们在传输数据时，可以只使用(传输层)TCP/IP协议，但是那样的话，如果没有应用层，便无法识别数据内容。如果想要使传输的数据有意义，则必须使用到应用层协议。

应用层协议有很多，比如HTTP、FTP、TELNET等，也可以自己定义应用层协议。

WEB使用HTTP协议作应用层协议，以封装HTTP文本信息，然后使用TCP/IP做传输层协议将它发到网络上。

而我们平时说的最多的socket是什么呢，实际上socket是对TCP/IP协议的封装，Socket本身并不是协议，而是一个调用接口(API)。

通过Socket，我们才能使用TCP/IP协议。实际上，Socket跟TCP/IP协议没有必然的联系。Socket编程接口在设计的时候，就希望也能适应其他的网络协议。

所以说，Socket的出现只是使得程序员更方便地使用TCP/IP协议栈而已，是对TCP/IP协议的抽象，从而形成了我们知道的一些最基本的函数接口，比如create、listen、connect、accept、send、read和write等等。

网络有一段关于socket和TCP/IP协议关系的说法比较容易理解：

TCP/IP只是一个协议栈，就像操作系统的运行机制一样，必须要具体实现，同时还要提供对外的操作接口。

这个就像操作系统会提供标准的编程接口，比如win32编程接口一样，TCP/IP也要提供可供程序员做网络开发所用的接口，这就是Socket编程接口。

关于TCP/IP协议的相关知识，用“博大精深”来讲我想也不为过，单单查一下网上关于此类只是的资料和书籍文献的数量就知道。这个我打算会买一些经典的书籍(比如《TCP/IP详解：卷一、卷二、卷三》)进行学习。

今天就先总结一些基于基于TCP/IP协议的应用和编程接口的知识，也就是刚才说了很多的HTTP和Socket。

CSDN上有个比较形象的描述：HTTP是轿车，提供了封装或者显示数据的具体形式;Socket是发动机，提供了网络通信的能力。

实际上，传输层的TCP是基于网络层的IP协议的；而应用层的HTTP协议又是基于传输层的TCP协议的；而Socket本身不算是协议，就像上面所说，它只是提供了一个针对TCP或者UDP编程的接口。

下面是一些经常在笔试或者面试中碰到的重要的概念，特在此做摘抄和总结。

一、什么是TCP连接的三次握手

第一次握手：客户端发送syn包(syn=j)到服务器，并进入SYN_SEND状态，等待服务器确认；

第二次握手：服务器收到syn包，必须确认客户的SYN(ack=j+1)，同时自己也发送一个SYN包(syn=k)，即SYN+ACK包，此时服务器进入SYN_RECV状态；

第三次握手：客户端收到服务器的SYN+ACK包，向服务器发送确认包ACK(ack=k+1)，此包发送完毕，客户端和服务器进入ESTABLISHED状态，完成三次握手。

握手过程中传送的包里不包含数据，三次握手完毕后，客户端与服务器才正式开始传送数据。

理想状态下，TCP连接一旦建立，在通信双方中的任何一方主动关闭连接之前，TCP 连接都将被一直保持下去。

断开连接时服务器和客户端均可以主动发起断开TCP连接的请求，断开过程需要经过“四次握手”(过程就不细写了，就是服务器和客户端交互，最终确定断开)

二、利用Socket建立网络连接的步骤

建立Socket连接至少需要一对套接字，其中一个运行于客户端，称为ClientSocket，另一个运行于服务器端，称为ServerSocket。

套接字之间的连接过程分为三个步骤：服务器监听，客户端请求，连接确认。

1、服务器监听：服务器端套接字并不定位具体的客户端套接字，而是处于等待连接的状态，实时监控网络状态，等待客户端的连接请求。

2、客户端请求：指客户端的套接字提出连接请求，要连接的目标是服务器端的套接字。

为此，客户端的套接字必须首先描述它要连接的服务器的套接字，指出服务器端套接字的地址和端口号，然后就向服务器端套接字提出连接请求。

3、连接确认：当服务器端套接字监听到或者说接收到客户端套接字的连接请求时，就响应客户端套接字的请求，建立一个新的线程，把服务器端套接字的描述发给客户端，一旦客户端确认

了此描述，双方就正式建立连接。

而服务器端套接字继续处于监听状态，继续接收其他客户端套接字的连接请求。

三、HTTP链接的特点

HTTP协议即超文本传送协议(Hypertext Transfer Protocol)，是Web联网的基础，也是手机联网常用的协议之一，HTTP协议是建立在TCP协议之上的一种应用。

HTTP连接最显著的特点是客户端发送的每次请求都需要服务器回送响应，在请求结束后，会主动释放连接。从建立连接到关闭连接的过程称为“一次连接”。

四、TCP和UDP的区别

(考得最多。。快被考烂了我觉得- -\\)

- TCP是面向链接的，虽然说网络的不安全不稳定特性决定了多少次握手都不能保证连接的可靠性，但TCP的三次握手在最低限度上(实际上也很大程度上保证了)保证了连接的可靠性；
- 而UDP不是面向连接的，UDP传送数据前并不与对方建立连接，对接收到的数据也不发送确认信号，发送端不知道数据是否会正确接收，当然也不用重发，所以说UDP是无连接的、不可靠的一种数据传输协议。
- 也正由于1所说的特点，使得UDP的开销更小数据传输速率更高，因为不必进行收发数据的确认，所以UDP的实时性更好。

知道了TCP和UDP的区别，就不难理解为何采用TCP传输协议的MSN比采用UDP的QQ传输文件慢了。但并不能说QQ的通信是不安全的，因为程序员可以手动对UDP的数据收发进行验证，比如发送方对每个数据包进行编号然后由接收方进行验证啊什么的。

即使是这样，UDP因为在底层协议的封装上没有采用类似TCP的“三次握手”而实现了TCP所无法达到的传输效率。

文章地址：<http://jingyan.baidu.com/article/08b6a591e07ecc14a80922f1.html>

推荐几篇阅读量最高的文章

1. 推荐：打造高效率iOS开发实战交流VIP群
 2. 一个程序员这辈子需要挣多少钱才能维系一个家庭？
 3. VIP群福利：Git使用教程（视频+PPT）
-

[阅读原文](#)



微信扫一扫
关注该公众号