[**笑笑小白**](http://www.cnblogs.com/rosesmall/)

归类自己需要的资料，慢慢爬行

[http与https的区别](http://www.cnblogs.com/rosesmall/archive/2012/04/09/2439710.html)

TTPS（Secure Hypertext Transfer Protocol）--安全超文本传输协议

它是由Netscape开发并内置于其浏览器中，用于对数据进行压缩和解压操作，并返回网络上传送回的结果。HTTPS实际上应用了Netscape的完全套接字层（SSL）作为HTTP应用层的子层。（HTTPS使用端口443，而不是象HTTP那样使用端口80来和TCP/IP进行通信。）SSL使用40 位关键字作为RC4流加密算法，这对于商业信息的加密是合适的。HTTPS和SSL支持使用X.509数字认证，如果需要的话用户可以确认发送者是谁。

WWW的核心——HTTP协议

众所周知，Internet的基本协议是TCP/IP协议，目前广泛采用的FTP、Archie Gopher等是建立在TCP/IP协议之上的应用层协议，不同的协议对应着不同的应用。<BR> WWW服务器使用的主要协议是HTTP协议，即超文体传输协议。由于HTTP协议支持的服务不限于WWW，还可以是其它服务，因而HTTP协议允许用户在统一的界面下，采用不同的协议访问不同的服务，如FTP、Archie、SMTP、NNTP等。另外，HTTP协议还可用于名字服务器和分布式对象管理。

2.1 HTTP协议简介   
HTTP是一个属于应用层的面向对象的协议，由于其简捷、快速的方式，适用于分布式超媒体信息系统。它于1990年提出，经过几年的使用与发展，得到不断地完善和扩展。目前在WWW中使用的是HTTP/1.0的第六版，HTTP/1.1的规范化工作正在进行之中，而且HTTP-NG(Next Generation of HTTP)的建议已经提出。   
HTTP协议的主要特点可概括如下：   
1.支持客户/服务器模式。   
2.简单快速：客户向服务器请求服务时，只需传送请求方法和路径。请求方法常用的有GET、HEAD、POST。每种方法规定了客户与服务器联系的类型不同。   
由于HTTP协议简单，使得HTTP服务器的程序规模小，因而通信速度很快。   
3.灵活：HTTP允许传输任意类型的数据对象。正在传输的类型由Content-Type加以标记。   
4.无连接：无连接的含义是限制每次连接只处理一个请求。服务器处理完客户的请求，并收到客户的应答后，即断开连接。采用这种方式可以节省传输时间。   
5.无状态：HTTP协议是无状态协议。无状态是指协议对于事务处理没有记忆能力。缺少状态意味着如果后续处理需要前面的信息，则它必须重传，这样可能导致每次连接传送的数据量增大。另一方面，在服务器不需要先前信息时它的应答就较快。

2.2 HTTP协议的几个重要概念   
1.连接(Connection)：一个传输层的实际环流，它是建立在两个相互通讯的应用程序之间。   
2.消息(Message)：HTTP通讯的基本单位，包括一个结构化的八元组序列并通过连接传输。   
3.请求(Request)：一个从客户端到服务器的请求信息包括应用于资源的方法、资源的标识符和协议的版本号  
4.响应(Response)：一个从服务器返回的信息包括HTTP协议的版本号、请求的状态(例如“成功”或“没找到”)和文档的MIME类型。   
5.资源(Resource)：由URI标识的网络数据对象或服务。   
6.实体(Entity)：数据资源或来自服务资源的回映的一种特殊表示方法，它可能被包围在一个请求或响应信息中。一个实体包括实体头信息和实体的本身内容。   
7.客户机(Client)：一个为发送请求目的而建立连接的应用程序。   
8.用户代理(User agent)：初始化一个请求的客户机。它们是浏览器、编辑器或其它用户工具。   
9.服务器(Server)：一个接受连接并对请求返回信息的应用程序。   
10.源服务器(Origin server)：是一个给定资源可以在其上驻留或被创建的服务器。   
11.代理(Proxy)：一个中间程序，它可以充当一个服务器，也可以充当一个客户机，为其它客户机建立请求。请求是通过可能的翻译在内部或经过传递到其它的服务器中。一个代理在发送请求信息之前，必须解释并且如果可能重写它。   
代理经常作为通过防火墙的客户机端的门户，代理还可以作为一个帮助应用来通过协议处理没有被用户代理完成的请求。   
12.网关(Gateway)：一个作为其它服务器中间媒介的服务器。与代理不同的是，网关接受请求就好象对被请求的资源来说它就是源服务器；发出请求的客户机并没有意识到它在同网关打交道。   
网关经常作为通过防火墙的服务器端的门户，网关还可以作为一个协议翻译器以便存取那些存储在非HTTP系统中的资源。   
13.通道(Tunnel)：是作为两个连接中继的中介程序。一旦激活，通道便被认为不属于HTTP通讯，尽管通道可能是被一个HTTP请求初始化的。当被中继的连接两端关闭时，通道便消失。当一个门户(Portal)必须存在或中介(Intermediary)不能解释中继的通讯时通道被经常使用。   
14.缓存(Cache)：反应信息的局域存储。

2.3 HTTP协议的运作方式   
HTTP协议是基于请求／响应范式的。一个客户机与服务器建立连接后，发送一个请求给服务器，请求方式的格式为，统一资源标识符、协议版本号，后边是MIME信息包括请求修饰符、客户机信息和可能的内容。服务器接到请求后，给予相应的响应信息，其格式为一个状态行包括信息的协议版本号、一个成功或错误的代码，后边是MIME信息包括服务器信息、实体信息和可能的内容。   
许多HTTP通讯是由一个用户代理初始化的并且包括一个申请在源服务器上资源的请求。最简单的情况可能是在用户代理(UA)和源服务器(O)之间通过一个单独的连接来完成(见图2-1)。   
图2-1   
当一个或多个中介出现在请求／响应链中时，情况就变得复杂一些。中介由三种：代理(Proxy)、网关(Gateway)和通道(Tunnel)。一个代理根据URI的绝对格式来接受请求，重写全部或部分消息，通过URI的标识把已格式化过的请求发送到服务器。网关是一个接收代理，作为一些其它服务器的上层，并且如果必须的话，可以把请求翻译给下层的服务器协议。一个通道作为不改变消息的两个连接之间的中继点。当通讯需要通过一个中介(例如：防火墙等)或者是中介不能识别消息的内容时，通道经常被使用。 图2-2   
上面的图2-2表明了在用户代理(UA)和源服务器(O)之间有三个中介(A,B和C)。一个通过整个链的请求或响应消息必须经过四个连接段。这个区别是重要的，因为一些HTTP通讯选择可能应用于最近的连接、没有通道的邻居，应用于链的终点或应用于沿链的所有连接。尽管图2-2是线性的，每个参与者都可能从事多重的、并发的通讯。例如，B可能从许多客户机接收请求而不通过A，并且／或者不通过C把请求送到A，在同时它还可能处理A的请求。   
任何针对不作为通道的汇聚可能为处理请求启用一个内部缓存。缓存的效果是请求／响应链被缩短，条件是沿链的参与者之一具有一个缓存的响应作用于那个请求。下图说明结果链，其条件是针对一个未被UA或A加缓存的请求，B有一个经过C来自O的一个前期响应的缓存拷贝。   
图2-3   
在Internet上，HTTP通讯通常发生在TCP/IP连接之上。缺省端口是TCP 80，但其它的端口也是可用的。但这并不预示着HTTP协议在Internet或其它网络的其它协议之上才能完成。HTTP只预示着一个可靠的传输。   
以上简要介绍了HTTP协议的宏观运作方式，下面介绍一下HTTP协议的内部操作过程。   
首先，简单介绍基于HTTP协议的客户/服务器模式的信息交换过程，如图2-4所示，它分四个过程，建立连接、发送请求信息、发送响应信息、关闭连接。   
图2-4   
在WWW中，“客户”与“服务器”是一个相对的概念，只存在于一个特定的连接期间，即在某个连接中的客户在另一个连接中可能作为服务器。WWW服务器运行时，一直在TCP80端口(WWW的缺省端口)监听，等待连接的出现。   
下面，讨论HTTP协议下客户/服务器模式中信息交换的实现。 1.建立连接 连接的建立是通过申请套接字(Socket)实现的。客户打开一个套接字并把它约束在一个端口上，如果成功，就相当于建立了一个虚拟文件。以后就可以在该虚拟文件上写数据并通过网络向外传送。   
2.发送请求   
打开一个连接后，客户机把请求消息送到服务器的停留端口上，完成提出请求动作。   
HTTP/1.0 请求消息的格式为：   
请求消息=请求行(通用信息|请求头|实体头) CRLF[实体内容]   
请求 行=方法 请求URL HTTP版本号 CRLF   
方 法=GET|HEAD|POST|扩展方法   
U R L=协议名称+宿主名+目录与文件名   
请求行中的方法描述指定资源中应该执行的动作，常用的方法有GET、HEAD和POST。不同的请求对象对应GET的结果是不同的，对应关系如下：   
对象 GET的结果   
文件 文件的内容   
程序 该程序的执行结果   
数据库查询 查询结果   
HEAD——要求服务器查找某对象的元信息，而不是对象本身。   
POST——从客户机向服务器传送数据，在要求服务器和CGI做进一步处理时会用到POST方法。POST主要用于发送HTML文本中FORM的内容，让CGI程序处理。   
一个请求的例子为：   
GET http://networking.zju.edu.cn/zju/index.htm HTTP/1.0   
头信息又称为元信息，即信息的信息，利用元信息可以实现有条件的请求或应答 。   
请求头——告诉服务器怎样解释本次请求，主要包括用户可以接受的数据类型、压缩方法和语言等。   
实体头——实体信息类型、长度、压缩方法、最后一次修改时间、数据有效期等。   
实体——请求或应答对象本身。   
3.发送响应   
服务器在处理完客户的请求之后，要向客户机发送响应消息。   
HTTP/1.0的响应消息格式如下：   
响应消息=状态行(通用信息头|响应头|实体头) CRLF 〔实体内容〕   
状 态 行=HTTP版本号 状态码 原因叙述   
状态码表示响应类型   
1×× 保留   
2×× 表示请求成功地接收   
3×× 为完成请求客户需进一步细化请求   
4×× 客户错误   
5×× 服务器错误   
响应头的信息包括：服务程序名，通知客户请求的URL需要认证，请求的资源何时能使用。   
4.关闭连接   
客户和服务器双方都可以通过关闭套接字来结束TCP/IP对话   
参考资料：匿名回答，助人为乐！  
回答者：ljpbxh027 - 高级魔法师 六级 12-13 15:04  
HTTP1.1（Hypertext Transfer Protocol Vertion 1.1）超文本传输协议-版本1.1   
它是用来在Internet上传送超文本的传送协议。它是运行在TCP/IP协议族之上的HTTP应用协议，它可以使浏览器更加高效，使网络传输减少。任何服务器除了包括HTML文件以外，还有一个HTTP驻留程序，用于响应用用户请求。您的浏览器是HTTP客户，向服务器发送请求，当浏览器中输入了一个开始文件或点击了一个超级链接时，浏览器就向服务器发送了HTTP请求，此请求被送往由IP地址指定的URL。驻留程序接收到请求，在进行必要的操作后回送所要求的文件。   
HTTPS（Secure Hypertext Transfer Protocol）安全超文本传输协议   
它是由Netscape开发并内置于其浏览器中，用于对数据进行压缩和解压操作，并返回网络上传送回的结果。HTTPS实际上应用了Netscape的完全套接字层（SSL）作为HTTP应用层的子层。（HTTPS使用端口443，而不是象HTTP那样使用端口80来和TCP/IP进行通信。）SSL使用40 位关键字作为RC4流加密算法，这对于商业信息的加密是合适的。HTTPS和SSL支持使用X.509数字认证，如果需要的话用户可以确认发送者是谁。  
注意：在远程管理网站默认设置是启用了SSL的时候也要用HTTPS啊!!!!!!!!!!!!!!