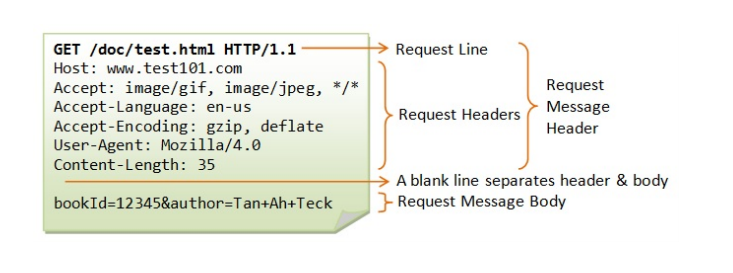
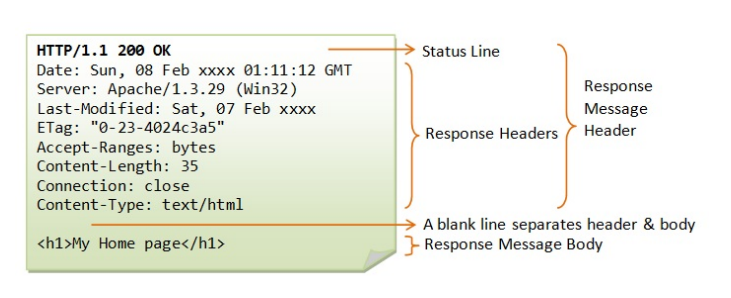
HTTP基础知识

1. 基础概念
2. 一些基本概念
3. HTTP(HyperText Transfer Protocol)超文本传输协议
4. WWW(World Wide Web)的三种技术:http,html,url
5. RFC(Request For Comments)征求修正意见书
6. URI (统一资源标识符), URL(统一资源定位符), URN(统一资源名称)
7. 请求报文



1. 响应报文



1. HTTP方法

在请求报文的第一行包含了方法字段。

常用的方法:GET,POST,DELETE,PUT,

1. GET
2. POST

Post的主要目的不是获取资源而是传输存储在内容实体中的数据。

一点区别:

1. GET的传参方式相比POST安全性要差,同时GET只支持ASCII,如果有中文字符会出现乱码。
2. 使用GET,浏览器会将Header和Data部分一起发出去，响应200并返回数据

使用POST,浏览器先发放Header,服务器响应100(continue)然后再发送data

1. HEAD

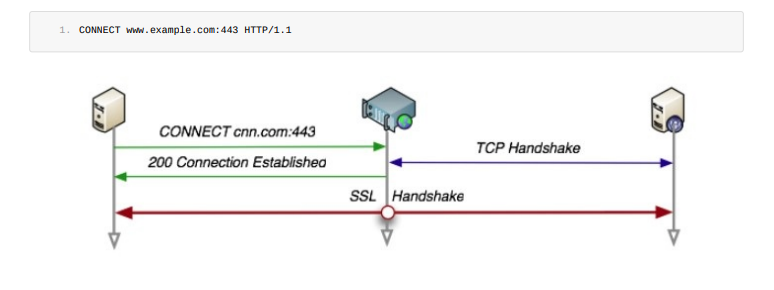
和GET方法一样,主要用于确认URL的有效性以及资源更新的日期时间

1. PUT,PATCH,DELETE
2. OPTIONS

查询指定URL所支持的方法,会返回Allow:GET,POST.....

1. CONNECT

要求在与代理服务器通信时建立隧道。使用SSL和TLS协议将通信内容加密



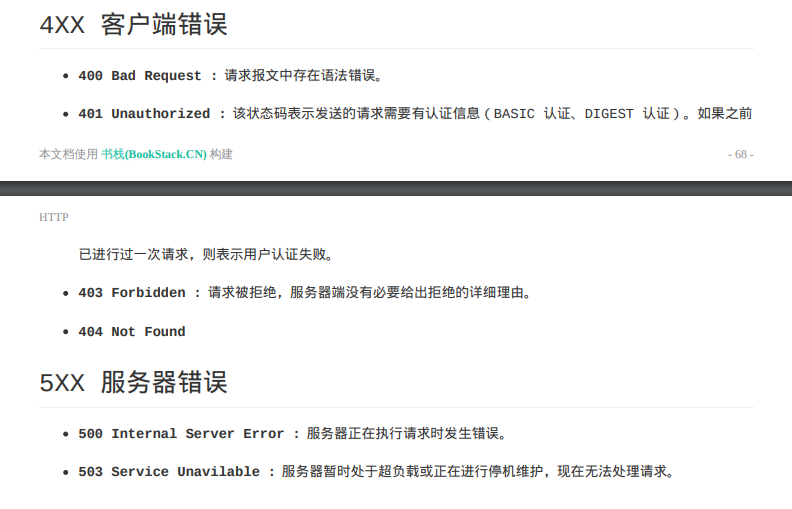
1. Trace

服务器将通信路径返回给客户端

1. HTTP状态码







1. HTTP首部

HTTP有4种类型的首部字段:通用首部,请求首部,响应首部,实体首部。









1. 具体应用
2. Cookie

HTTP/1.1引入Cookie来保存状态信息。

Cookie是服务器发送给客户端的信息,它会保存再浏览器中,客户端的下一次请求报文中会包含Cookie数据。通过Cookie可以让服务器知道两个请求是否来自于同一个客户端。从而实现保持登录状态的功能。

主要是Set-Cookie



1. Session和Cookie的区别

Session通过再服务端记录信息确定用户身份。Cookie在客户端记录信息确定用户身份。服务器通过Cookie中sessionID来找到具体session

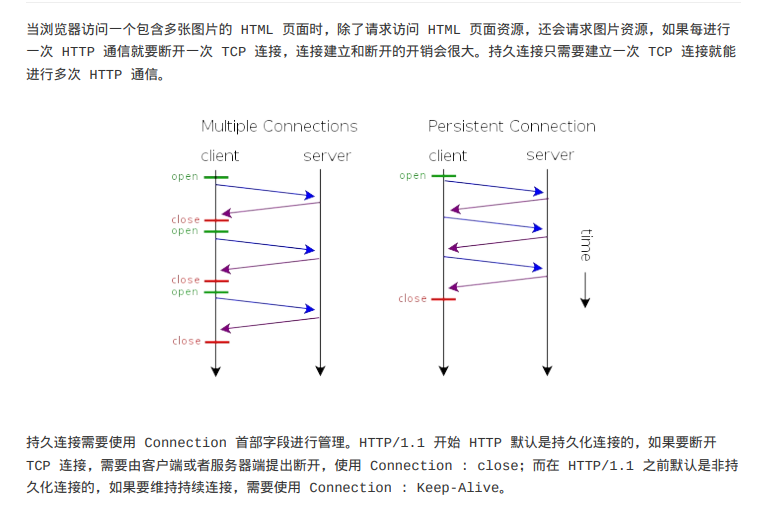
1. 缓存

让代理服务器进行缓存

让客户端浏览器进行缓存

通过通用首部中的Cache-Control来控制缓存

1. 持久连接



使用的是通用首部中Connection字段。

1. 编码

编码主要是为了对实体进行压缩,常用的编码有:gzip,compress,defalte,identify,

其中identify表示不执行压缩

1. 分块传输

把数据分割成多块，让浏览器逐步显示页面。

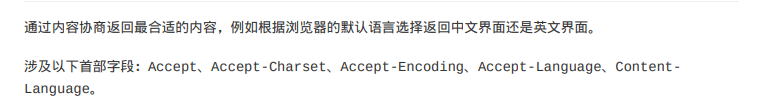
1. 多部分对象集合



1. 范围请求



1. 内容协商



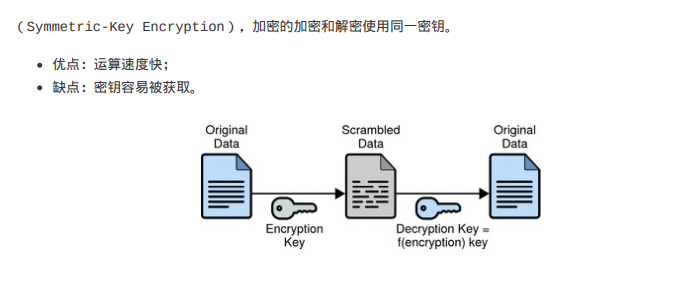
1. 虚拟主机



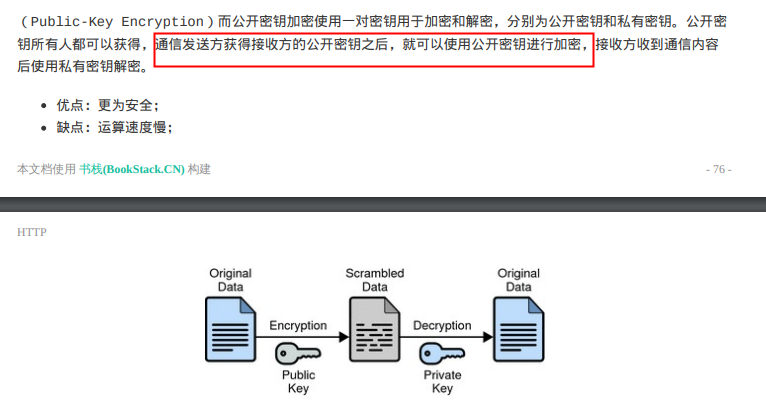
1. HTTPS
2. HTTP的安全问题
3. 使用明文进行通信，通信内容可能别窃听
4. 不验证通信方的身份，通信方的身份可能被伪装
5. 无法证明报文的完整性，可能被篡改

HTTP -> SSL -> TCP的过程

1. 加密知识
2. 对称密钥，使用同一密钥



1. 公开密钥，使用一对密钥



1. HTTPS使用的加密方式

HTTPS使用混合的加密机制，使用公开密钥加密用于传输对称密钥，之后使用对称密钥进行通信。

4）验证身份

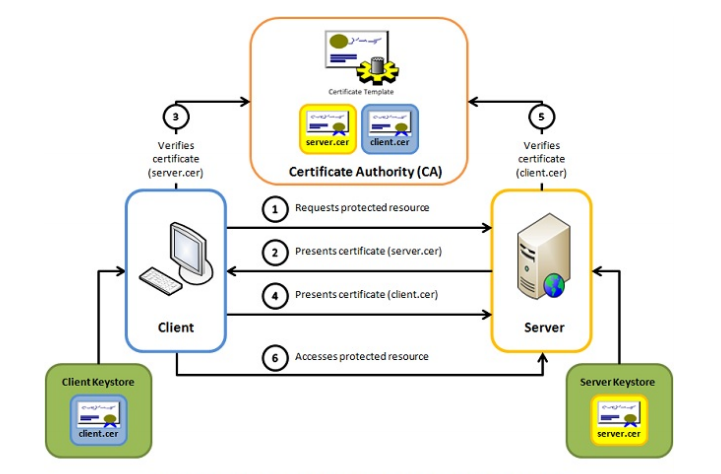
使用证书来对通信方进行验证。

数字证书CA是客户端与服务器双方都可以信赖的第三方机构。

大体流程总结下：

1. 服务器端向CA提出公开密钥的申请
2. CA在判断申请者身份后对公开密钥做数字签名,然后分配，并将公开密钥放入证书
3. 进行HTTPS通信时，服务器将证书发送给客户端,客户端获得其中的密钥后会先进行验证，验证通过后才开始通信

开源程序OpenSSL



1. WEB攻击技术
2. HTTP版本比较