###### 前端面试指南 — http 篇面试题

###### 1.问题： HTTP 的请求报文由哪三部分构成？

请求行，请求头，请求体（请求数据）

###### 2.问题：HTTP 的状态码有哪些？

200("OK")

一切正常。实体主体中的文档（若存在的话）是某资源的表示。

400("Bad Request")

客户端方面的问题。实体主题中的文档（若存在的话）是一个错误消息。希望客户端能够理解此错误消息，并改正问题。

500("Internal Server Error")

服务期方面的问题。实体主体中的文档（如果存在的话）是一个错误消息。该错误消息通常无济于事，因为客户端无法修复服务器方面的问题。

301("Moved Permanently")

当客户端触发的动作引起了资源URI的变化时发送此响应代码。另外，当客户端向一个资源的旧URI发送请求时，也发送此响应代码。

404("Not Found") 和410("Gone")

当客户端所请求的URI不对应于任何资源时，发送此响应代码。404用于服务器端不知道客户端要请求哪个资源的情况；410用于服务器端知道客户端所请求的资源曾经存在，但现在已经不存在了的情况。

409("Conflict")

当客户端试图执行一个”会导致一个或多个资源处于不一致状态“的操作时，发送此响应代码。

其他查看：<https://www.cnblogs.com/xflonga/p/9368993.html>

###### 3.问题：HTTP 和HTTPS的区别？

HTTPS 协议需要到 CA （Certificate Authority，证书颁发机构）申请证书，一般免费证书较少，因而需要一定费用。(以前的网易官网是http，而网易邮箱是 https 。)

HTTP 是超文本传输协议，信息是明文传输，HTTPS 则是具有安全性的 SSL 加密传输协议。

HTTP 和 HTTPS 使用的是完全不同的连接方式，用的端口也不一样，前者是80，后者是443。

HTTP 的连接很简单，是无状态的。HTTPS 协议是由 SSL+HTTP 协议构建的可进行加密传输、身份认证的网络协议，比 HTTP 协议安全。(无状态的意思是其数据包的发送、传输和接收都是相互独立的。无连接的意思是指通信双方都不长久的维持对方的任何信息。)

详细地址：https://mp.weixin.qq.com/s/PnB9BMzs7QxtiJv7oXS2fA

###### 4.问题：浏览器地址栏输入 URL 敲下回车后发生了什么

**答案**

1. URL解析
2. DNS 查询
3. TCP连接
4. HTTP请求
5. 相应请求
6. 页面渲染

详细了解：<https://mp.weixin.qq.com/s/pb2Qk0dEsowFnPPUIXV4Qg>

###### 5.问题：get和post的区别是什么

GET在浏览器回退时是无害的，而POST会再次提交请求。

GET产生的URL地址可以被Bookmark，而POST不可以。

GET请求会被浏览器主动cache，而POST不会，除非手动设置。

GET请求只能进行url编码，而POST支持多种编码方式。

GET请求参数会被完整保留在浏览器历史记录里，而POST中的参数不会被保留。

GET请求在URL中传送的参数是有长度限制的，而POST么有。

对参数的数据类型，GET只接受ASCII字符，而POST没有限制。

GET比POST更不安全，因为参数直接暴露在URL上，所以不能用来传递敏感信息。

GET参数通过URL传递，POST放在Request body中。

###### 6.问题：Http与Https的区别

HTTP 的URL 以http:// 开头，而HTTPS 的URL 以https:// 开头

HTTP 是不安全的，而 HTTPS 是安全的

HTTP 标准端口是80 ，而 HTTPS 的标准端口是443

在OSI 网络模型中，HTTP工作于应用层，而HTTPS 的安全传输机制工作在传输层

HTTP 无法加密，而HTTPS 对传输的数据进行加密

HTTP无需证书，而HTTPS 需要CA机构wosign的颁发的SSL证书

###### 7.问题：Http与Https的区别

无状态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

也就是说，当客户端一次HTTP请求完成以后，客户端再发送一次HTTP请求，HTTP并不知道当前客户端是一个”老用户“。

可以使用Cookie来解决无状态的问题，Cookie就相当于一个通行证，第一次访问的时候给客户端发送一个Cookie，当客户端再次来的时候，拿着Cookie(通行证)，那么服务器就知道这个是”老用户“。

###### 8.问题：常用的HTTP方法有哪些？

GET： 用于请求访问已经被URI（统一资源标识符）识别的资源，可以通过URL传参给服务器

POST：用于传输信息给服务器，主要功能与GET方法类似，但一般推荐使用POST方式。

PUT： 传输文件，报文主体中包含文件内容，保存到对应URI位置。

HEAD： 获得报文首部，与GET方法类似，只是不返回报文主体，一般用于验证URI是否有效。

DELETE：删除文件，与PUT方法相反，删除对应URI位置的文件。

OPTIONS：查询相应URI支持的HTTP方法。

###### 9.问题：HTTP请求报文与响应报文格式？

请求报文包含四部分：

a、请求行：包含请求方法、URL、HTTP版本信息

b、请求首部字段

c、请求内容实体

d、空行

响应报文包含四部分：

a、状态行：包含HTTP版本、状态码、状态码的原因短语

b、响应首部字段

c、响应内容实体

d、空行

###### 10.问题HTTPS工作原理？

一、首先HTTP请求服务端生成证书，客户端对证书的有效期、合法性、域名是否与请求的域名一致、证书的公钥（RSA加密）等进行校验；

二、客户端如果校验通过后，就根据证书的公钥的有效， 生成随机数，随机数使用公钥进行加密（RSA加密）；

三、消息体产生的后，对它的摘要进行MD5（或者SHA1）算法加密，此时就得到了RSA签名；

四、发送给服务端，此时只有服务端（RSA私钥）能解密。

五、解密得到的随机数，再用AES加密，作为密钥（此时的密钥只有客户端和服务端知道）。

HTTPS其实是有两部分组成：HTTP + SSL / TLS，

也就是在HTTP上又加了一层处理加密信息的模块。服务端和客户端的信息传输都会通过TLS进行加密，所以传输的数据都是加密后的数据

1. 客户端发起HTTPS请求

这个没什么好说的，就是用户在浏览器里输入一个https网址，然后连接到server的443端口。

2. 服务端的配置

采用HTTPS协议的服务器必须要有一套数字证书，可以自己制作，也可以向组织申请。区别就是自己颁发的证书需要客户端验证通过，才可以继续访问，而使用受信任的公司申请的证书则不会弹出提示页面(startssl就是个不错的选择，有1年的免费服务)。这套证书其实就是一对公钥和私钥。如果对公钥和私钥不太理解，可以想象成一把钥匙和一个锁头，只是全世界只有你一个人有这把钥匙，你可以把锁头给别人，别人可以用这个锁把重要的东西锁起来，然后发给你，因为只有你一个人有这把钥匙，所以只有你才能看到被这把锁锁起来的东西。

3. 传送证书

这个证书其实就是公钥，只是包含了很多信息，如证书的颁发机构，过期时间等等。

4. 客户端解析证书

这部分工作是有客户端的TLS来完成的，首先会验证公钥是否有效，比如颁发机构，过期时间等等，如果发现异常，则会弹出一个警告框，提示证书存在问题。如果证书没有问题，那么就生成一个随即值。然后用证书对该随机值进行加密。就好像上面说的，把随机值用锁头锁起来，这样除非有钥匙，不然看不到被锁住的内容。

5. 传送加密信息

这部分传送的是用证书加密后的随机值，目的就是让服务端得到这个随机值，以后客户端和服务端的通信就可以通过这个随机值来进行加密解密了。

6. 服务段解密信息

服务端用私钥解密后，得到了客户端传过来的随机值(私钥)，然后把内容通过该值进行对称加密。所谓对称加密就是，将信息和私钥通过某种算法混合在一起，这样除非知道私钥，不然无法获取内容，而正好客户端和服务端都知道这个私钥，所以只要加密算法够彪悍，私钥够复杂，数据就够安全。

7. 传输加密后的信息

这部分信息是服务段用私钥加密后的信息，可以在客户端被还原

8. 客户端解密信息

客户端用之前生成的私钥解密服务段传过来的信息，于是获取了解密后的内容。整个过程第三方即使监听到了数据，也束手无策。

###### 11.问题HTTP优化方案

TCP复用：TCP连接复用是将多个客户端的HTTP请求复用到一个服务器端TCP连接上，而HTTP复用则是一个客户端的多个HTTP请求通过一个TCP连接进行处理。前者是负载均衡设备的独特功能；而后者是HTTP 1.1协议所支持的新功能，目前被大多数浏览器所支持。

内容缓存：将经常用到的内容进行缓存起来，那么客户端就可以直接在内存中获取相应的数据了。

压缩：将文本数据进行压缩，减少带宽

SSL加速（SSL Acceleration）：使用SSL协议对HTTP协议进行加密，在通道内加密并加速

TCP缓冲：通过采用TCP缓冲技术，可以提高服务器端响应时间和处理效率，减少由于通信链路问题给服务器造成的连接负担。

详情参考：

<http://blog.51cto.com/virtualadc/580832>

<http://www.cnblogs.com/cocowool/archive/2011/08/22/2149929.html>