**保护HTTP的安全**

* http的安全版本应该具有：高效、可移植且易于管理，不但能够适应不断变化的情况而且还应该能满足社会的各项要求。总结如下：

1、服务器认证：（客户端知道它们是在与真正的而不是伪造的服务器通话）

2、客户端认证：（服务器知道它们是在与真正的而不是伪造的客户端通话）

3、完整性：（客户端与服务器的数据不会被修改）

4、加密：（客户端和服务器的对话是私密的，无需担心被窃听）

5、效率：（一个运行的足够快的算法，以便低端的客户端和服务器使用）

6、普适性：（基本上所有的客户端和服务器都支持这些协议）

7、管理的可扩展性：（在任何地方的任何人都可以立即进行安全通信）

8、适应性：（能够支持当前最知名的安全方法）

9、在社会上的可行性：（满足社会的政治文化需要）

* HTTPS：https是http的安全版本协议，所有现代浏览器和服务器都支持这个协议。用户可以在地址栏那里的看url的方法是https还是http来区分。所有https要求所有数据在进行网络传输前进行加密处理，https是在http和tcp/ip之间加了一个安全传输层SSL（或者TLS，跟SSL区别不大，不作特别声明的情况下，它们可以互指）。大部分困难和复杂的编码和解码算法都在SSL中完成的，所以客户端和服务器在使用https不用做过多的协议处理逻辑！

**数字加密**

1、密码：对文本进行编码，是偷窥者无法识别的算法（注意是“算法”）

2、秘钥：改变密码行文的数字化参数

3、对称秘钥加密系统：编/解码使用相同密钥的算法

4、公开秘钥加密系统：一种能够使数百万计算机便捷地发送机密报文的系统

5、数字签名：用来验证报文未被伪造或篡改的检验和

6、数字证书：由一个可信的组织验证和签发的识别信息

* 密码：一种（方法）特殊的报文编码方式和一种稍后使用的相应解码方式的结合体。被编码的报文称为密文，被解码的报文称为明文。在一次请求（响应）中过程中，报文的状态为：明文————密文————明文。如果只使用密码的话，其实有很大的安全缺陷的，因为对方只要知道的你的密码算法，就能够解码处明文。
* 秘钥：正因为密码算法有缺陷，所有才有了秘钥。说的直白一点，你只有密码的话，别人知道了算法，照样能解码，但是除了密码还有秘钥，别人知道密码就不一定能解码出来了，至少有些困难，那么秘钥是什么呢！它就是一个数字参数，有了它就可以实现，每次解码的算法不一样。就好像一个函数，每次传的参数不一样。秘钥就是那个“参数”！
* 对称秘钥加密技术：发送端和接受端使用同样的秘钥进行加密的密码算法技术。
* 枚举攻击：输完所有的可能值来达到解码的一种攻击方案。也就是说如果秘钥的可能方案越多，那么解码所花的代价越大。而秘钥的数量取决于秘钥的位数以及有效位数。所以可以从这些方面来增加密码的安全级别。
* 公开秘钥加密技术：由于对称秘钥加密技术会有秘钥数目的N\*N的问题，所有公开秘钥加密技术要求发送端使用公用的秘钥，在接收端使用私有秘钥的解码。
* 数字签名：签名就是一种证明身份的机制，是一种校验机制。
* 数字证书：一种由某官方机构颁发的证书

**HTTPS————细节介绍**

* 建立安全传输：https与http的不同就是，中间了增加了一次SSL握手的步骤。具体情况就是，https在建立tcp/ip连接之后还需要进行一次SSL层的传输连接工作连接，同时在关闭tcp/ip连接之前需要增加一次关闭SSL层的通知工作。
* SSL握手过程中确定了以下工作细节：

1、交换协议版本号；

2、选择一个两端都了解的密码；

3、对两端的身份进行认证；

4、生成临时的会话秘钥，以便加密信道。

* SSL握手步骤为：

1、客户端发送可供选择的密码并请求证书

2、服务器发送选中的密码和证书

3、客户端发送保密信息；客户端和服务端生成秘钥

4、客户端和服务器相互告知，开始加密过程