HTTPS 传输协议原理介绍

Hatter Jiang Apr 9th, 2008

我们常常在使用网上银行时看到的连接都是以“https”开始的，那么这个 https 是什么呢？这其实是表示目前连接使用了 SSL 进行加密，能保证客户端到服务器端的通信都在被保护起来，那么浏览器是如果实现的呢？下面我们介绍一下 SSL 的基本实现方法。

首先我们有两种基本的加解密算法类型：对称加密，非对称加密（公私钥加密），现在介绍一下这两种加密算法的特点：

对称加密：密钥只有一个，加密解密为同一个密码，且加解密速度快，典型的对称加密算法有 DES、AES 等，示意图如下：

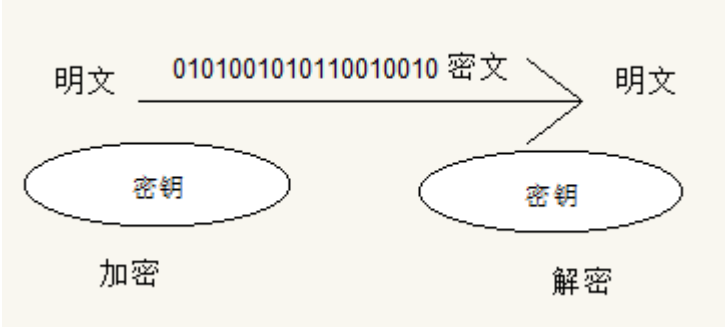

图 1 对称加密

非对称加密：密钥成对出现（且根据公钥无法推知私钥，根据私钥也无法推知公钥），加密解密使用不同密钥（公钥加密需要私钥解密，私钥加密需要公钥解密），相对对称加密速度较慢，典型的非对称加密算法有 RSA、DSA 等，示意图如下：

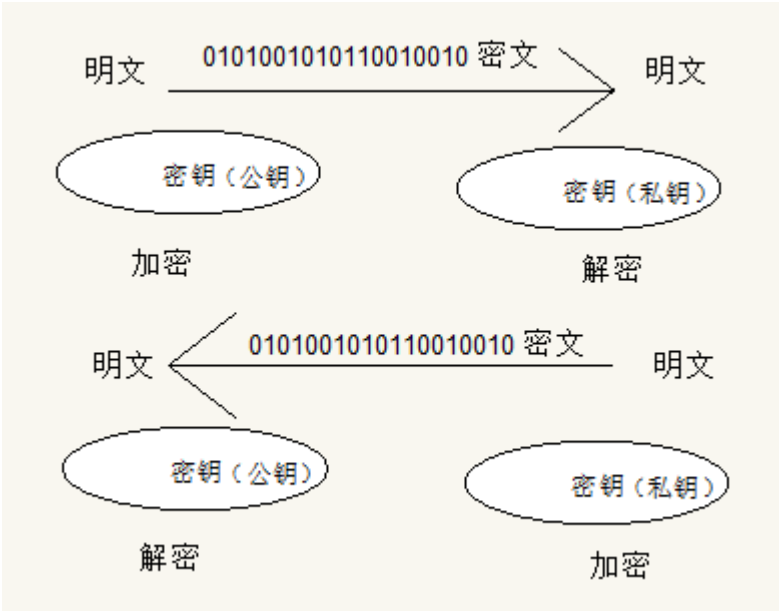

图 2 非对称加密

根据上面的两种加密方法，现在我们就可以设计一种无法让他人在互联网上知道你的通讯信息的加密方法：

1. 在服务器端存在一个公钥及私钥
2. 客户端从服务器取得这个公钥
3. 客户端产生一个随机的密钥
4. 客户端通过公钥对密钥加密（非对称加密）
5. 客户端发送到服务器端
6. 服务器端接受这个密钥并且以后的服务器端和客户端的数据全部通过这个密钥加密（对称加密）

HTTPS 通信过程的时序图如下：

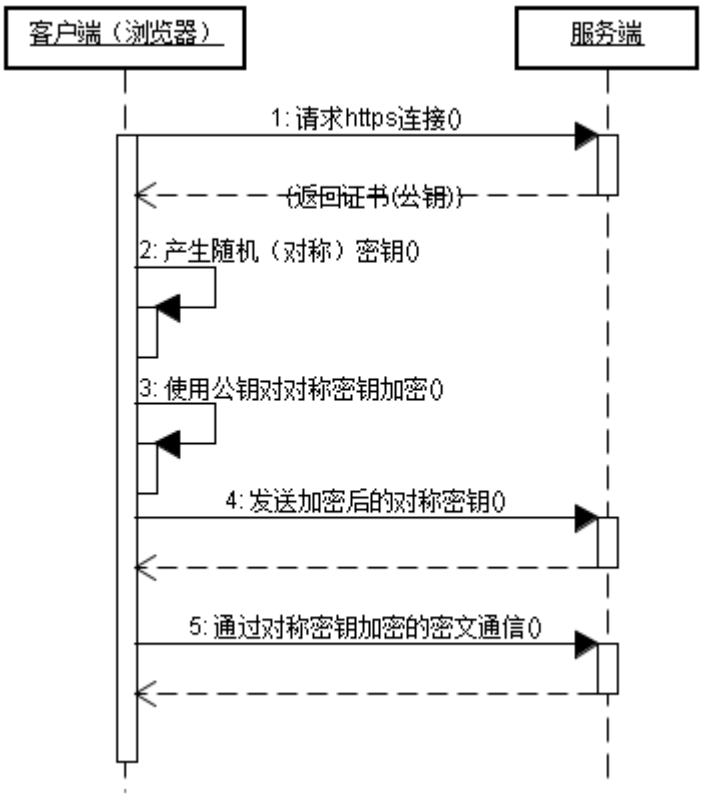

图 3 HTTPS 通信时序图

正如上图所示，我们能保证下面几点：

1. 客户端产生的密钥只有客户端和服务器端能得到
2. 加密的数据只有客户端和服务器端才能得到明文
3. 客户端到服务端的通信是安全的

当然实际的 SSL 实现算法复杂的多，并有数据校验、身份验证等功能，如果需要更多了角请参看 RFC2246 及 RFC4346 文档。

参考文献：

[1] RFC2246 T. Dierks, C. Allen The TLS Protocol Version 1.0
[2] RFC4346 T.Dierks, E. Rescorla The Transport Layer Security (TLS) Protocol Version 1.1