深入浅出HTTPS工作原理

前言

在HTTP协议中有可能存在信息窃听或身份伪装等安全问题。使用HTTPS通信机制可以有效地防止这些问题。本文我们就了解一下HTTPS。

一、什么是 HTTPS

HTTPS，是以安全为目标的HTTP通道，简单讲是HTTP的安全版。即HTTP下加入SSL层，HTTPS的安全基础是SSL，因此加密的详细内容就需要SSL。 现在它被广泛用于万维网上安全敏感的通讯，例如交易支付方面。

经常会在Web的登录页面和购物结算界面等使用HTTPS通信。使用HTTPS通信时，不再用`http://`，而是改用`https://`。另外，当浏览器访问HTTPS通信有效的Web网站时，浏览器的地址栏内会出现一个带锁的标记。对HTTPS的显示方式会因浏览器的不同而有所改变。

二、HTTP 与 HTTPS 的区别

- HTTP 是明文传输，HTTPS 通过 SSLTLS 进行了加密
- HTTP 的端口号是 80，HTTPS 是 443
- HTTPS 需要到 CA 申请证书，一般免费证书很少，需要交费
- HTTPS 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。

为什么说HTTPS比较安全了，接下我们介绍下HTTP存在哪些问题?

三、HTTP通信有什么问题?

1.通信使用明文（不加密），内容可能被窃听

由于HTTP本身不具备加密的功能，所以也无法做到对通信整体（使用HTTP协议通信的请求和响应的内容）进行加密。即，**HTTP报文使用明文（指未经过加密的报文）方式发送**。

此外互联网是由联通世界各个地方的网络设施组成,所有发送和接收经过某些设备的数据都可能被截获或窥视。例如大家都熟悉的抓包工具:Wireshark,它可以获取HTTP协议的请求和响应的内容，并对其进行解析。**即使经过加密处理，就有可能让人无法破解报文信息的含义，但加密处理后的报文信息本身还是会被看到的**。

2.不验证通信方的身份，因此有可能遭遇伪装

HTTP协议中的请求和响应不会对通信方进行确认。在HTTP协议通信时，由于不存在确认通信方的处理步骤，任何人都可以发起请求。另外，服务器只要接收到请求，不管对方是谁都会返回一个响应（但也仅限于发送端的IP地址和端口号没有被Web服务器设定限制访问的前提下）

HTTP协议的实现本身非常简单，不论是谁发送过来的请求都会返回响应，因此不确认通信方，会存在以下各种隐患。比如目标的Web服务器有可能是已伪装的Web服务器。

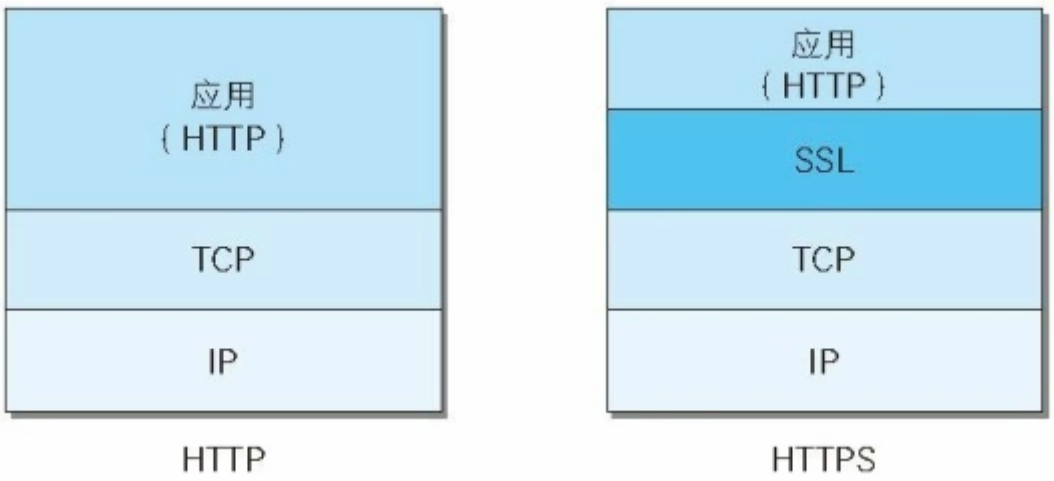
3.无法证明报文的完整性，所以可能遭篡改

所谓完整性是指信息的准确度。若无法证明其完整性，通常也就意味着无法判断信息是否准确。由于HTTP协议无法证明通信的报文完整性，因此，在请求或响应送出之后直到对方接收之前的这段时间内，即使请求或响应的内容遭到篡改，也没有办法获悉。
换句话说，**没有任何办法确认，发出的请求/响应和接收到的请求/响应是前后相同的**。

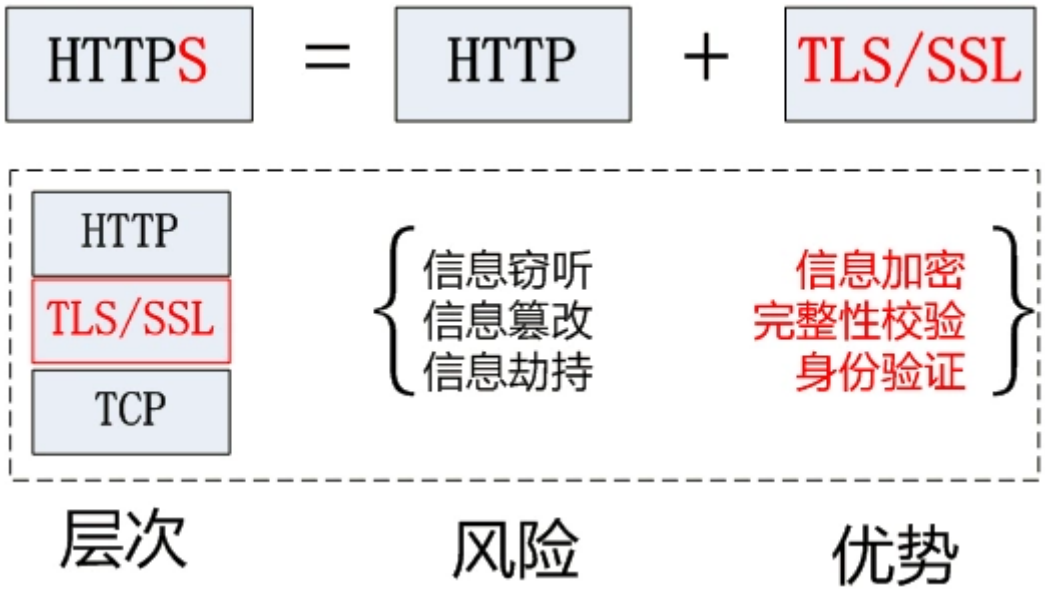
四、HTTPS如何解决上述三个问题?

HTTPS并非是应用层的一种新协议。只是HTTP通信接口部分用SSL（Secure Socket Layer）和TLS（Transport Layer Security）协议代替而已。

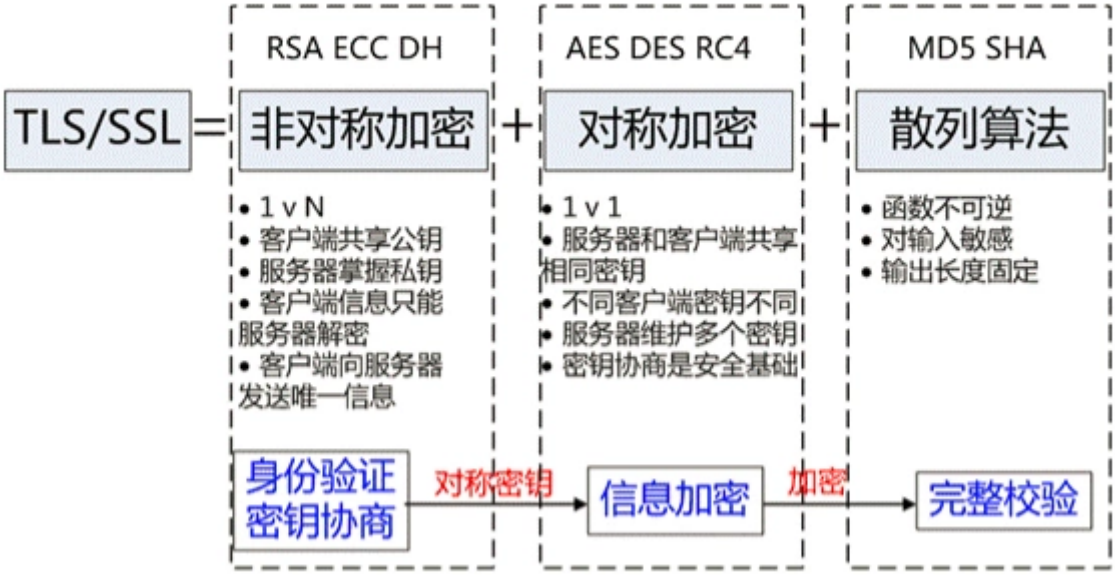
通常，HTTP直接和TCP通信。当使用SSL时，则演变成先和SSL通信，再由SSL和TCP通信了。简言之，**所谓HTTPS，其实就是身披SSL协议这层外壳的HTTP**。



在采用SSL后，HTTP就拥有了HTTPS的加密、证书和完整性保护这些功能。也就是说**HTTP加上加密处理和认证以及完整性保护后即是HTTPS**。



HTTPS 协议的主要功能基本都依赖于 TLS/SSL 协议，TLS/SSL 的功能实现主要依赖于三类基本算法：散列函数 、对称加密和非对称加密，**其利用非对称加密实现身份认证和密钥协商，对称加密算法采用协商的密钥对数据加密，基于散列函数验证信息的完整性**。



（一）解决内容可能被窃听的问题——加密

1.对称加密

这种方式加密和解密同用一个密钥。加密和解密都会用到密钥。**没有密钥就无法对密码解密，反过来说，任何人只要持有密钥就能解密了**。

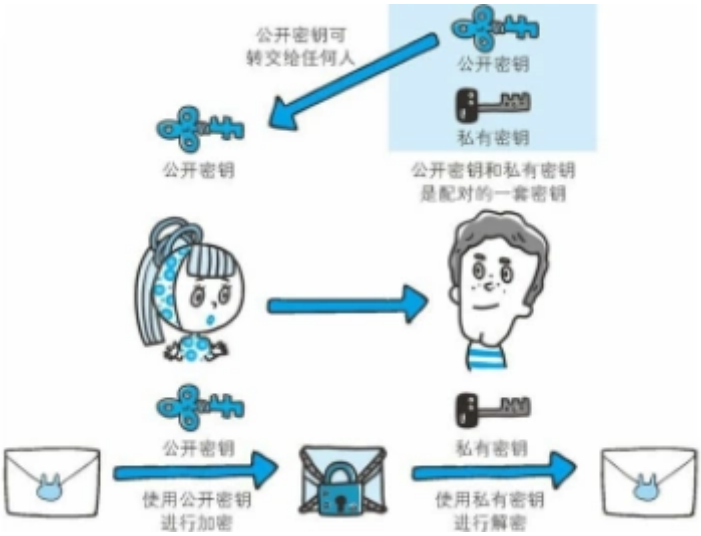
以对称加密方式加密时必须将密钥也发给对方。可究竟怎样才能安全地转交？在互联网上转发密钥时，如果通信被监听那么密钥就可会落人攻击者之手，同时也就失去了加密的意义。另外还得设法安全地保管接收到的密钥。

2.非对称加密

公开密钥加密使用一对非对称的密钥。一把叫做私有密钥，另一把叫做公开密钥。顾名思义，**私有密钥不能让其他任何人知道，而公开密钥则可以随意发布，任何人都可以获得**。

使用公开密钥加密方式，发送密文的一方使用**对方的公开密钥**进行加密处理，对方收到被加密的信息后，再使用自己的私有密钥进行解密。利用这种方式，不需要发送用来解密的私有密钥，也不必担心密钥被攻击者窃听而盗走。

非对称加密的特点是信息传输一对多，服务器只需要维持一个私钥就能够和多个客户端进行加密通信，但服务器发出的信息能够被所有的客户端解密，且该算法的计算复杂，加密速度慢。



3.对称加密+非对称加密

尽管非对称加密设计奇妙,但它加解密的效率比对称加密要慢多了。那我们就将对称加密与非对称加密结合起来,充分利用两者各自的优势，将多种方法组合起来用于通信。**在交换密钥环节使用非对称加密方式，之后的建立通信交换报文阶段则使用对称加密方式**。具体做法是：**发送密文的一方使用对方的公钥进行加密处理“对称的密钥”，然后对方用自己的私钥解密拿到“对称的密钥”，这样可以确保交换的密钥是安全的前提下，使用对称加密方式进行通信**。所以，HTTPS采用对称加密和非对称加密两者并用的混合加密机制。

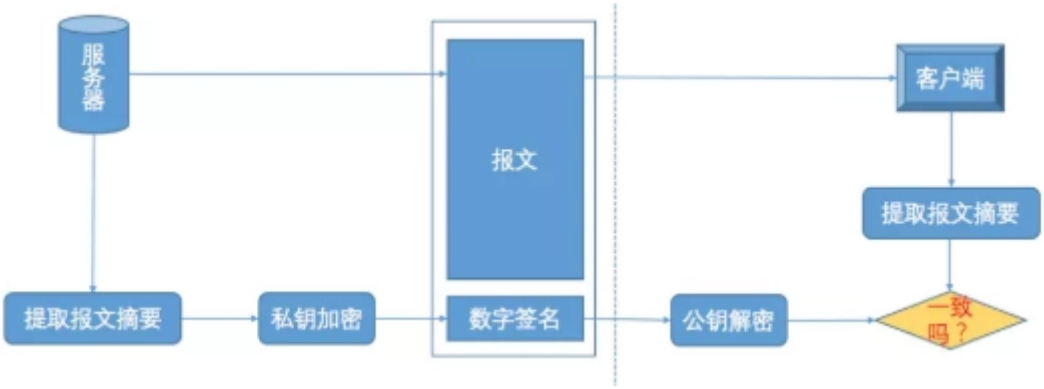
（二）解决报文可能遭篡改问题——数字签名

网络传输过程中需要经过很多中间节点，虽然数据无法被解密，但可能被篡改，那如何校验数据的完整性呢？----校验数字签名。

数字签名有两种功效：

- 能确定消息确实是由发送方签名并发出来的，因为别人假冒不了发送方的签名。
- 数字签名能确定消息的完整性,证明数据是否未被篡改过。

校验数字签名流程见下图：



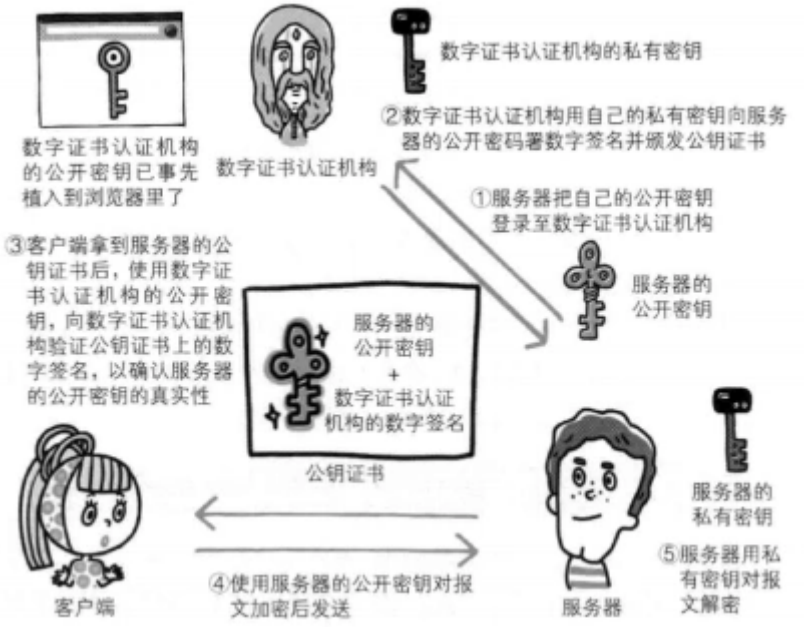
数字签名技术就是对“非对称密钥加解密”和“数字摘要“两项技术的应用，它将摘要信息用发送者的私钥加密，与原文一起传送给接收者。接收者只有用发送者的公钥才能解密被加密的摘要信息，然后用HASH函数对收到的原文产生一个摘要信息，与解密的摘要信息对比。如果相同，则说明收到的信息是完整的，在传输过程中没有被修改，否则说明信息被修改过，因此数字签名能够验证信息的完整性。

（三）解决通信方身份可能被伪装的问题——认证

非对称加密方式还是存在一些问题的。那就是无法证明公开密钥本身就是货真价实的公开密钥。比如，正准备和某台服务器建立公开密钥加密方式下的通信时，如何证明收到的公开密钥就是原本预想的那台服务器发行的公开密钥。
为了解决上述问题，可以使用由数字证书认证机构（CA，Certificate Authority）和其相关机关颁发的公开密钥证书。

数字证书认证机构处于客户端与服务器双方都可信赖的第三方机构的立场上。我们来介绍一下数字证书认证机构的业务流程。首先，服务器的运营人员向数字证书认证机构提出公开密钥的申请。数字证书认证机构在判明提出申请者的身份之后，会对已申请的公开密钥做数字签名，然后分配这个已签名的公开密钥，并将该公开密钥放入公钥证书后绑定在一起。
服务器会将这份由数字证书认证机构颁发的公钥证书发送给客户端，以进行非对称加密方式通信。公钥证书也可叫做数字证书或直接称为证书。

接到证书的客户端可使用数字证书认证机构的公开密钥，对那张证书上的数字签名进行验证，**一旦验证通过，客户端便可明确两件事：一，认证服务器的公开密钥的是真实有效的数字证书认证机构。二，服务器的公开密钥是值得信赖的**。



五、为什么不一直使用HTTPS?

既然HTTPS那么安全可靠，那为何所有的Web网站不一直使用HTTPS?

其中一个原因是，因为**与纯文本通信相比，加密通信会消耗更多的CPU及内存资源**。如果每次通信都加密，会消耗相当多的资源，平摊到一台计算机上时，能够处理的请求数量必定也会随之减少。
因此，如果是非敏感信息则使用HTTP通信，只有在包含个人信息等敏感数据时，才利用HTTPS加密通信。
特别是每当那些访问量较多的Web网站在进行加密处理时，它们所承担着的负载不容小觑。

除此之外，**想要节约购买证书的开销也是原因之一**。要进行HTTPS通信，证书是必不可少的。而使用的证书必须向认证机构（CA）购买。