HTTPS加密那点事

用漫画的形式解释技术问题有眼前一亮的感觉。

背景

不知道从哪天开始，一禅也陷入了编程这条道路.....



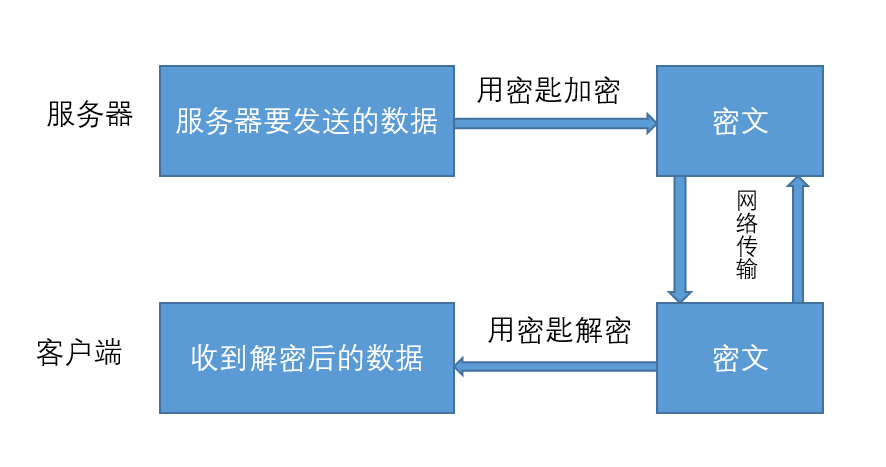




对称加密



在每次发送真实数据之前，服务器先生成一把密钥，然后先把密钥传输给客户端。之后服务器给客户端发送真实数据的时候，会用这把密钥对数据进行加密，客户端收到加密数据之后，用刚才收到的密钥进行解密。如图：



当然，如果客户端要给服务器发送数据，也是采用这把密钥来加密，这里为了方便，我采用单方向传输的形式









那万一密钥在传输的过程中被别人截取了怎么吧?

例如：

假如服务器用明文的方式传输密钥给客户端，然后密钥被中间人给捕获了，那么在之后服务器和客户端的加密传输过程中，中间人也可以用他捕获的密钥进行解密。这样的话，加密的数据在中间人看来和明文没啥两样。









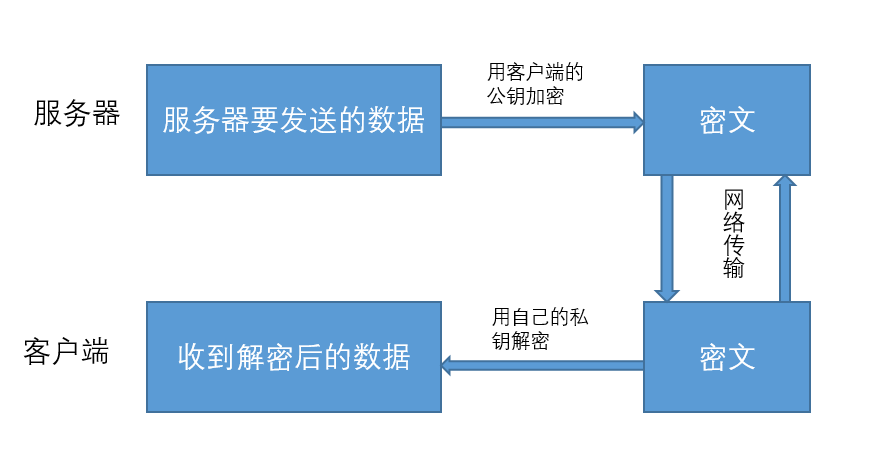


非对称加密



这种方法就是，让客户端和服务器都拥有两把钥匙，一把钥匙是公开的(全世界知道都没关系)，我们称之为公钥；另一把钥匙则是保密的(只有自己本人才知道)，我们称之为私钥。这且，用公钥加密的数据，只有对应的私钥才能解密；用私钥加密的数据，只有对应的公钥才能解密。

这样，服务器在给客户端传输数据的过程中，可以用客户端明文给他的公钥进行加密，然后客户端收到后，再用自己的私钥进行解密。客户端给服务器发送数据的时候也一样采取这样的方式。这样就能保持数据的安全传输了。画个图理解一下：

















处理方式就是结合 对称加密+非对称加密这两种方式，我们可以用非对称加密的方式来传输对称加密过程中的密钥，之后我们就可以采取对称加密的方式来传输数据了。具体是这样子的：

服务器用明文的方式给客户端发送自己的公钥，客户端收到公钥之后，会生成一把密钥(对称加密用的)，然后用服务器的公钥对这把密钥进行加密，之后再把密钥传输给服务器，服务器收到之后进行解密，最后服务器就可以安全着得到这把密钥了，而客户端也有同样一把密钥，他们就可以进行对称加密了。





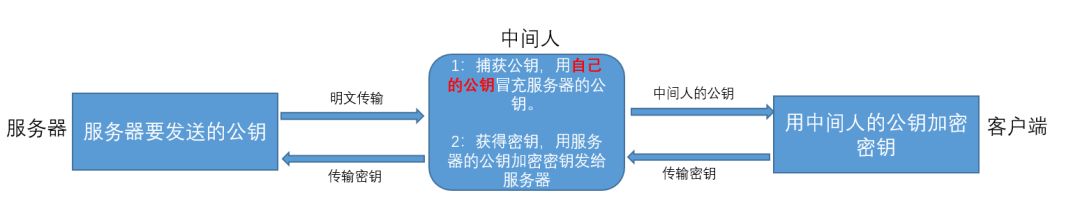


例如：

服务器以明文的方式给客户端传输公钥的时候，中间人截取了这把属于服务器的公钥，并且把中间人自己的公钥冒充服务器的公钥传输给了客户端。

之后客户端就会用中间人的公钥来加密自己生成的密钥。然后把被加密的密钥传输给服务器，这个时候中间人又把密钥给截取了，中间人用自己的私钥对这把被加密的密钥进行解密，解密后中间人就可以获得这把密钥了。

最后中间人再对这把密钥用刚才服务器的公钥进行加密，再发给服务器。如图：



毫无疑问，在这个过程中，中间人获取了对称加密中的密钥，在之后服务器和客户端的对称加密传输中，这些加密的数据对中间人来说，和明文没啥区别。













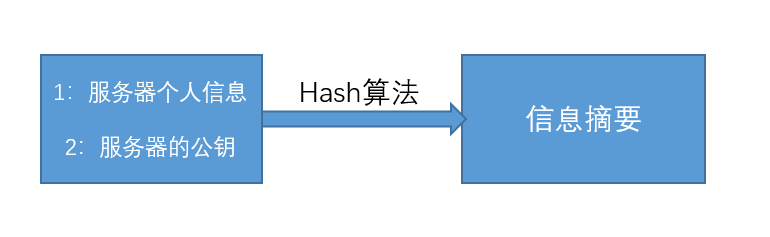
数字证书登场

在刚才的讲解中，我们知道，之所以非对称加密会不安全，是因为客户端不知道这把公钥是否是服务器的，因此，我们需要找到一种策略来证明这把公钥就是服务器的，而不是别人冒充的。

解决这个问题的方式就是使用数字证书，具体是这样的：

我们需要找到一个拥有公信力、大家都认可的认证中心(CA)。

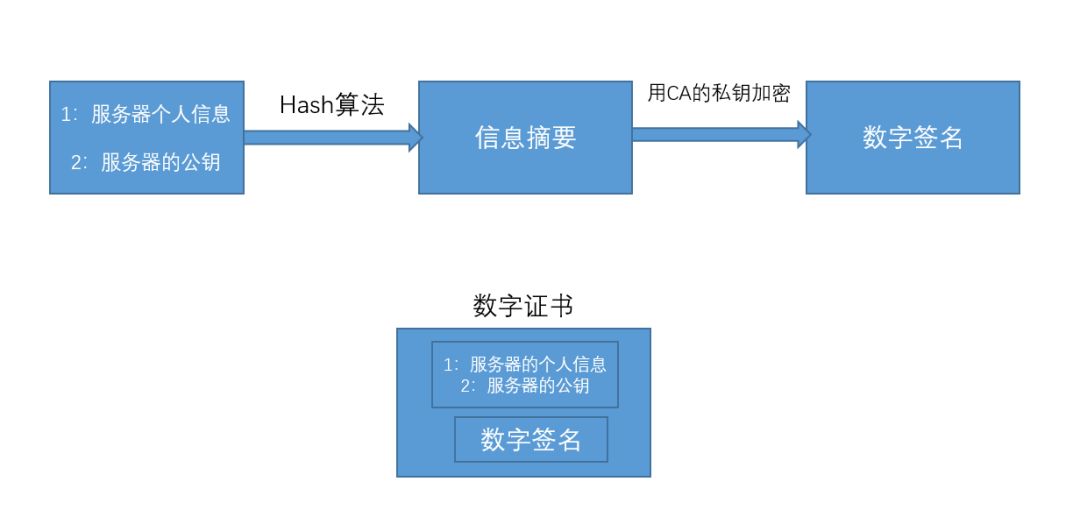
服务器在给客户端传输公钥的过程中，会把公钥以及服务器的个人信息通过Hash算法生成信息摘要。如图



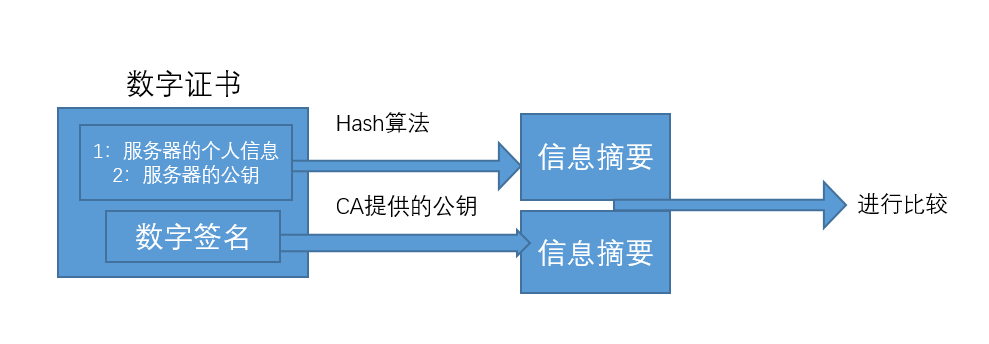
为了防止信息摘要被人调换，服务器还会用CA提供的私钥对信息摘要进行加密来形成数字签名。如图:



并且，最后还会把原来没Hash算法之前的个人信息以及公钥 和 数字签名合并在一起，形成数字证书。如图



当客户端拿到这份数字证书之后，就会用CA提供的公钥来对数字证书里面的数字签名进行解密来得到信息摘要，然后对数字证书里服务器的公钥以及个人信息进行Hash得到另外一份信息摘要。最后把两份信息摘要进行对比，如果一样，则证明这个人是服务器，否则就不是。如图：



这样，就可以保证服务器的公钥安全着交给客户端了。





其实，(有些)服务器一开始就向认证中心申请了这些证书了(有没有看过没有证书的网站在地址栏会被标出警告？)，而客户端是，也会内置这些证书。如图：



当客户端收到服务器传输过来的数据数字证书时，就会在内置的证书列表里，查看是否有解开该数字证书的公钥，如果有则...，如果没有则....







希望通过这种漫文的方式，能够让你更加轻松的读懂某些知识。

END

本文转载自微信公众号（苦逼的码农），原文地址：https://mp.weixin.qq.com/s/j-ss95ItMnWsZHLpUGBMkQ

公众号（苦逼的码农）专注于写【Java】、【计算机网络】、【数据结构与算法】，期待你加入交流。

公众号（苦逼的码农）专注于写【Java】、【计算机网络】、【数据结构与算法】，期待你加入交流。