加密&身份认证

对称加密 & 非对称加密

支付获取信息传递过程中，需要保证信息的机密性，加密是最常用的手段，加密过程中，涉及如下概念

> 明文：加密前的信息

> 密文：加密后的信息

> 密钥：将明文转换为密文过程中输入的参数

> 解密：通过密钥将密文还原为明文的过程

> 加密算法：加密 -> 解密过程中的操作

加密方式主要分为两大类

* 对称加密
* 非对称加密

对称加密

对称加密过程的示意图如下

k3u1fbpfcp/2e096cc339e14b6aaa942cc8f9b9f47b~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1239&h=315&s=18450&e=png&b=ffffff)

> 不论是加密操作还是解密操作，均使同一把`密钥A`来完成
>
>
> 对称加密较为流行的加密算法为[AES-高级加密标准 - 维基百科，自由的百科全书 (wikipedia.org)](http://cxyroad.com/”https://zh.wikipedia.org/wiki/%E9%AB%98%E7%BA%A7%E5%8A%A0%E5%AF%86%E6%A0%87%E5%87%86”)

非对称加密

非对称加密过程的示意图如下

> 加密和解密分别使用不同的密钥完成，即：使用公钥加密，则必须使用私钥才能解密，反之亦然
>
>
> 非对称加密中较为流行的加密算法为[RSA加密演算法 - 維基百科，自由的百科全書 (wikipedia.org)](http://cxyroad.com/”https://zh.wikipedia.org/zh-hk/RSA%E5%8A%A0%E5%AF%86%E6%BC%94%E7%AE%97%E6%B3%95”)

二者对比

对称加密

> 优点：运算速度快

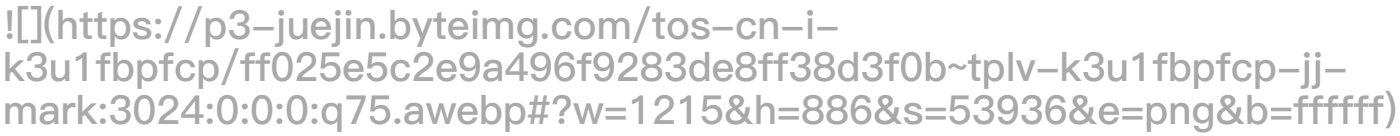
> 缺点：使用同一个密钥，一旦被窃取，消息就会被破解

非对称加密

> 优点：私钥严格保密，将公钥进行分发，第三方通过公钥无法破解消息

> 缺点：加密的运算效率较低

既想保证加密的安全性，又想提高加密效率，可以将二者结合使用，即：使用【非对称加密】加密【对称加密】的密钥，具体消息的加密，依然采用对称加密进行，如下图所示



> 先通过非对称加密传输密钥

> 通过交换的密钥使用对称加密进行消息的传递

身份认证

假设张三有一对公钥和私钥，他将公钥分发给了李四，王五和赵六，现有如下场景

> 场景一：李四写信给张三并使用张三的公钥进行加密，张三收到信后，使用自己的私钥解密，那么只要张三自己的私钥不泄露，整个信息交换的过程是安全的，即任何拥有张三公钥的人都可以写信给张三，整个过程是安全的

> 场景二：张三回信给李四，那么李四也必须拥有自己的公钥和私钥，因为如

果张三直接使用自己的私钥或公钥回信，会产生如下现象
>
>
> * 使用私钥回信：其他人可以使用张三的公钥解密信件
> * 使用公钥回信：李四没有张三的私钥，任何人都无法解密信件，信息传递没有产生实际意义
>
>
> 所以，李四必须拥有自己的公钥和私钥，张三通过李四的公钥加密回信，至此整个相互通信的过程是安全的

一般的情况都是使用公钥加密，私钥解密，倘若将二者进行转换，使用私钥加密，公钥解密，则可以产生【身份认证】的效果，例如

* 张三使用自己的私钥加密信件
* 李四使用张三的公钥解密信息

实际上只要拥有张三的公钥，就能够对张三的信进行解密，那么可以认为【私钥加密，公钥解密】的目的不是为了加密，而是为了身份认证，因为只有通过张三的私钥加密的内容才能被张三的公钥解密，那么可以认为这封信就是张三发的，即：张三的身份被确认

数字签名

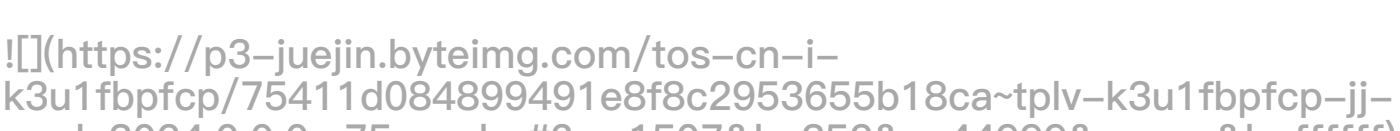
在消息传递的过程中，需要保证消息没有被串改过，也就是信息完整性的保证，实现完整性的主要手段是使用摘要算法（散列函数，哈希函数）

摘要算法有如下特点

* 不可逆：摘要算法是一个单向的过程，加密过后只有密文，没有密钥，再也不能通过加密后的密文解密出原始数据
* 难题友好性：想要通过密文破解出摘要算法计算前的原文，只能通过暴力枚举的方式（不断枚举原文，与摘要算法计算后的密文对比，一致，则代表破解成功，但原始可能是任意数据，随机性太强，破解难度极大）
* 发散性：对原文的任意改动，都会引起摘要算法加密结果的强烈变化
* 抗碰撞性：原文不同，计算后的摘要结果也不同

常见的摘要算法有：`MD5` `SHA1` `SHA2`

根据摘要算法以上的特性，其很适合用于校验信息是否被串改，在传输信息之前，通过摘要算法计算出信息的密文，信息的接收方接受到信息后，再次通过摘要算法计算信息的密文。相同，则代表信息没有被串改，如下图所示



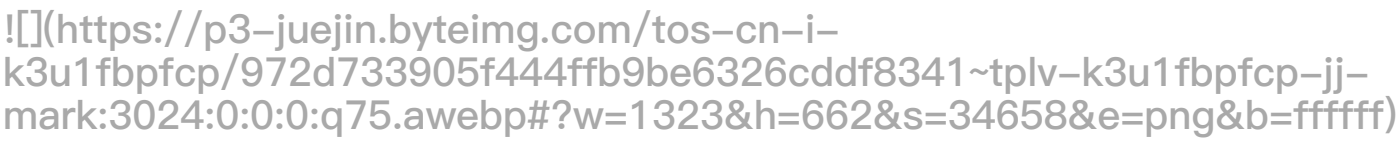
看起来好像保证了数据的完整性，但如果信息在传递的过程中，被第三方劫持，同时修改了原文和摘要，那么接收方就不能确定信息的完整性了

因此，在此基础上，还需要引入加密算法，通过非对称加密算法，将计算出的摘要使用私钥进行加密（身份确认），生成【数字签名】，这样即使数据被第三方劫持，由于第三方没有私钥，无法保证串改后的数据与数字签名相对应，那么接收方就能够校验数据的完整性（验签）

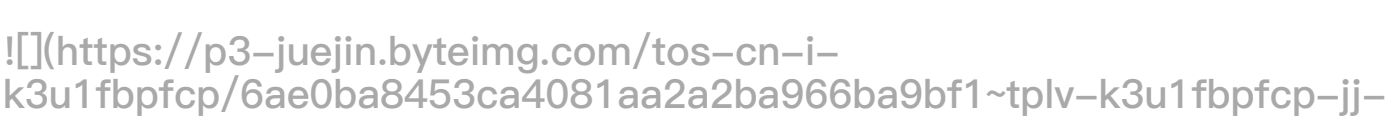
数字证书

A 伪装成 B，将自己的公钥传递给 C，一旦欺骗成功，后续 A 便可以使用自己的私钥进行数字签名，而 C 使用 A 给的公钥验签成功，则一直误认为自己在和 B 通信，核心的问题在于，任何人都可以发布公钥，无法确定公钥具体的所有者

为了解决这个问题，引入了数字证书，数字证书由第三方机构颁发，首先证书的申请者将自己的信息（必须提供真实身份），包括公钥，使用者（一般为域名），颁发机构等，如下图



将上述信息提交给第三方机构，第三方机构使用自己的私钥对上图中的数据进行加密得到密文（签名），最后将密文和原始明文数据再次发生给申请者（即数字证书），如下图



mark:3024:0:0:0:q75.awebp#?w=1271&h=903&s=60603&e=png&b=ffffff)

而证书的颁发机构也被称之为`CA-Certificate Authority`机构

接下来，假设 A 和 B 之间相互通信，需要携带自身的数字证书，当某一方接收到信息时，先根据 CA 机构公开的公钥解密证书中的签名，再与证书中的明文进行对比，如果一致，则可以证明对方的身份

> 这一机制的可靠性保证受第三方 CA 机构证书颁发的正确性影响，例如证书的颁发必须保证唯一性和绝对的公平透明
>
>
> 如果是浏览器中的数字证书，还会校验证书中的域名和浏览器正在访问的域名对比（即 https 协议）

总结
--

> 对称加密维持了信息加密的性能，非对称加密保障了密钥的安全，二者结合使用，能够提供高效安全的加密形式

> 通过数字签名与非对称加密的结合，保证了数据的完整性

> 引入数字证书的概念，保证了对公钥来源以及通信对象身份的确认

原文链接: https://juejin.cn/post/7389064690125176884