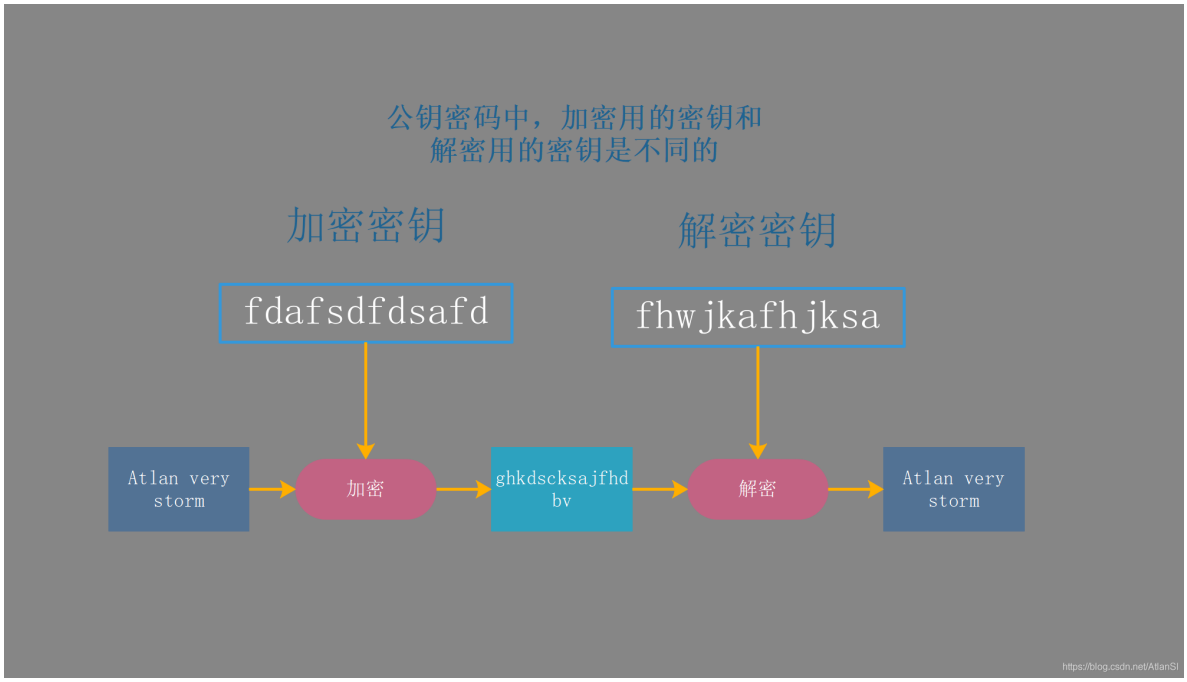
非对称加密

什么是非对称加密

“非对称加密也叫公钥密码：使用公钥加密，使用私钥解密



非对称加密中，密钥分为加密密钥和解密密钥两种。发送者用加密密钥对消息进行加密，接收者用解密密钥对密文进行解密。需理解公钥密码，清楚地分加密密钥和解密密钥是非常重要的。加密密钥是发送者加密时使用的，而解密密钥则是接收者解密时使用的。

区别

发送者只需要加密密钥

接收者只需要解密密钥

解密密钥不可以被窃听者获取

加密密钥被窃听者获取也没关系

就是说：解密密钥从一开始就是由接收者自己保管的，因此只要将加密密钥发给发送者就可以解决密钥配送问题了，而根本不需要配送解密密钥。

公钥是可以被公开和窃取的（发给别人），但是私钥就绝对不行，只可以自己保管和使用。

非对称加密通讯流程

假设A要给B发一条信息，A是发送者，B是接收者，窃听者C可以窃听他们之间的通讯内容。

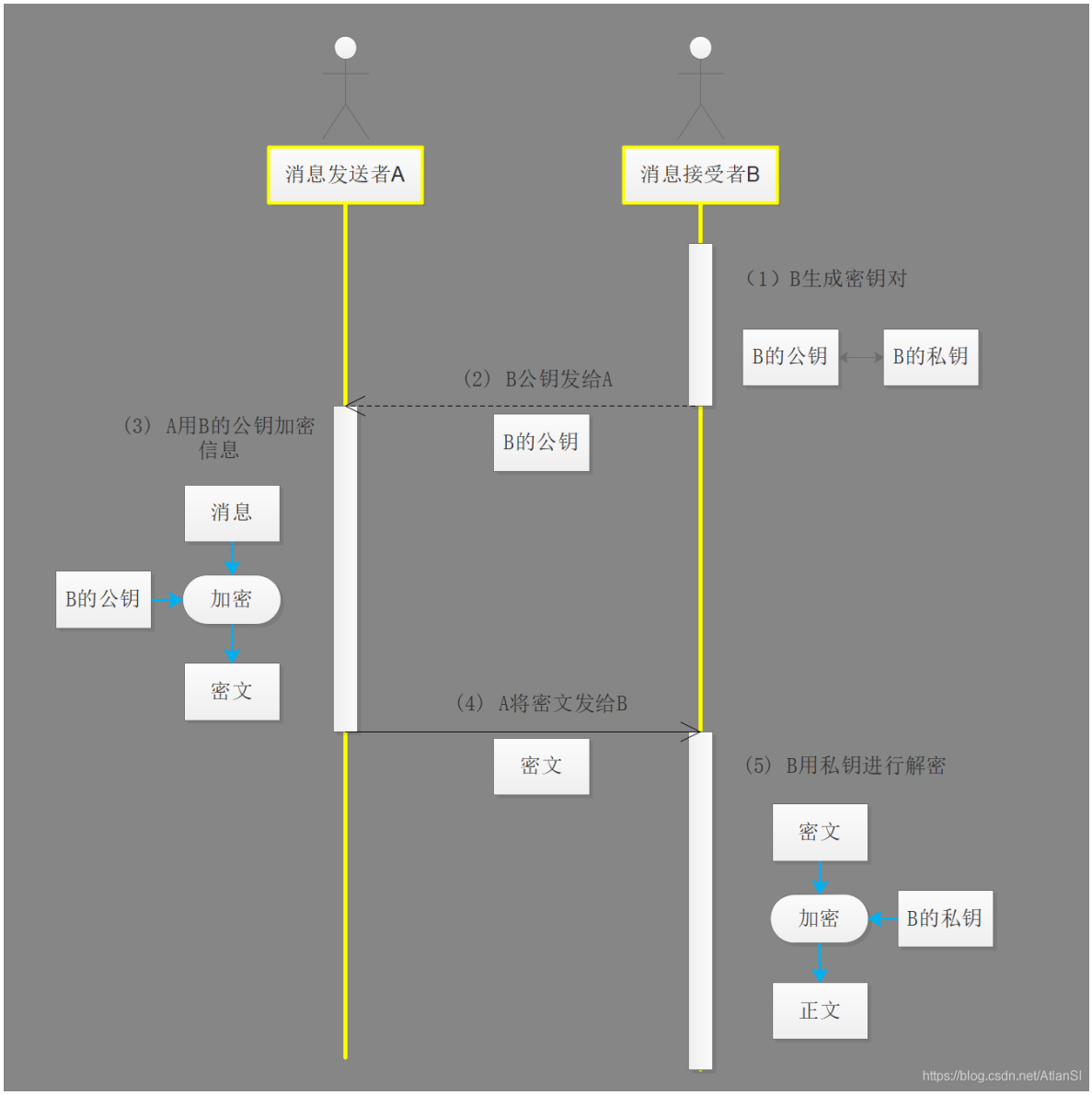
B生成一个包含公钥和私钥的密钥对
私钥由B自行妥善保管

B将自己的公钥发送给A
B的公钥被C截获也没关系。
将公钥发给A，表示B请A用这个公钥对消息进行加密并发送给他。

A用B的公钥对消息进行加密
加密后的消息只有B的西药才能够解密。
虽然A拥有B的公钥，但用B的公钥是无法对密文进行解密的。

A将密文发送给B
密文被C截获也没关系，C可能拥有B的公钥，但是B的公钥是无法进行解密的。

B用自己的私钥对密文进行解密。
参考下图



- **主要算法**

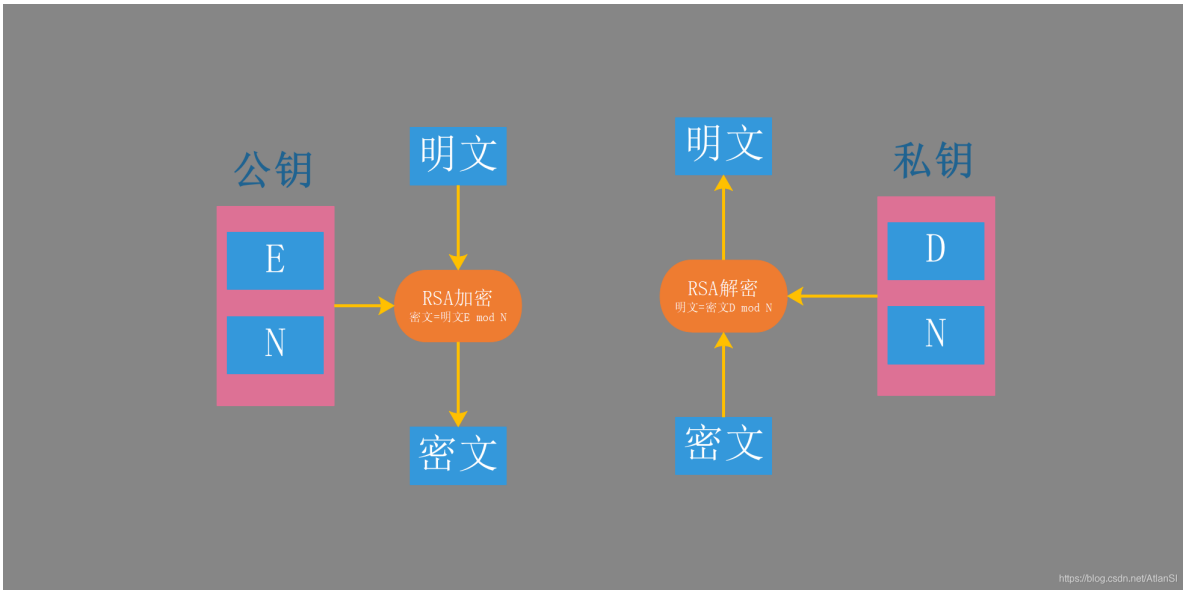
RSA、Elgamal、背包算法、Rabin、D-H、ECC (椭圆曲线加密算法)。使用最广泛的是 RSA 算法，Elgamal 是另一种常用的非对称加密算法

RSA

RSA的加密是求明文的E次方modN，因此只要知道E和N这两个数，任何人都可以完成加密的运算。所以说E和N是RSA加密的密钥。也就是说**E和N的组合就是公钥**

数D和数N组合起来就是RSA的解密密钥，因此D和N的组合就是私钥。只有知道D和N两个数的人才能够完成解密的运算。

密钥对	公钥	数E和数N
	私钥	数D和数N
加密		密文 = 明文 E mod N　　(明文的E次方除以N的余数)
解密		明文=明文 D mod N　　(密文的D次方除以N的余数)



ECC

椭圆曲线密码学，一种建立公开密钥加密的算法，基于椭圆曲线数学。在某些情况下它比其他的方法使用更小的密钥，比如RSA。

CC 164位的密钥产生的一个安全级相当于RSA 1024位密钥提供的保密强度，而且计算量较小，处理速度更快，存储空间和传输带宽占用较少。目前我国 居民二代身份证 正在使用 256 位的椭圆曲线密码，虚拟货币 比特币 也选择ECC作为加密算法。

比特币钱包公钥的生成使用了椭圆曲线算法，通过椭圆曲线乘法可以从私钥计算得到公钥， 这是不可逆转的过程。

DSA算法

简介

DSA (Digital Signature Algorithm) 是 Schnorr 和 ElGamal 签名算法的变种，被美国 NIST 作为 DSS (DigitalSignature Standard)。 DSA 是基于整数有限域离散对数难题的。

简单的说，这是一种更高级的验证方式，用作数字签名。不单单只有公钥、私钥，还有数字签名。私钥加密生成数字签名，公钥验证数据及签名，如果数据和签名不匹配则认为验证失败。数字签名的作用就是校验数据在传输过程中不被修改，数字签名，是单向加密的升级。

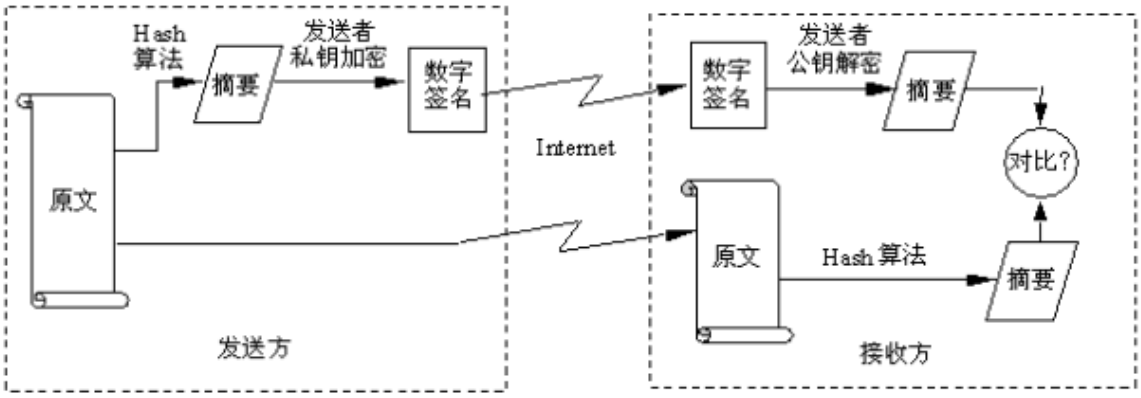


图1. 数字签名的原理图

处理过程

(1) 使用消息摘要算法将发送数据加密生成数字摘要。
(2) 发送方用自己的私钥对摘要再加密，形成数字签名。
(3) 将原文和加密的摘要同时传给对方。
(4) 接受方用发送方的公钥对摘要解密，同时对收到的数据用消息摘要算法产生同一摘要。
(5) 将解密后的摘要和收到的数据在接收方重新加密产生的摘要相互对比，如果两者一致，则说明在传送过程中信息没有破坏和篡改。否则，则说明信息已经失去安全性和保密性。

DH算法

DH，全称为"Diffie-Hellman"，它是一种确保共享 KEY 安全穿越不安全网络的方法，也就是常说的密钥一致协议。由公开密钥密码体制的奠基人 Diffie 和 Hellman 所提出的一种思想。简单的说就是允许两名用户在公开媒体上交换信息以生成"一致"的、可以共享的密钥。也就是由甲方产出一对密钥 (公钥、私钥)，乙方依照甲方公钥产生乙方密钥对 (公钥、私钥)。

以此为基线，作为数据传输保密基础，同时双方使用同一种对称加密算法构建本地密钥 (SecretKey) 对数据加密。这样，在互通了本地密钥 (SecretKey) 算法后，甲乙双方公开自己的公钥，使用对方的公钥和刚才产生的私钥加密数据，同时可以使用对方的公钥和自己的私钥对数据解密。不单单是甲乙双方两方，可以扩展为多方共享数据通讯，这样就完成了网络交互数据的安全通讯。

PS：这个加密一定要和数字签名区别开来，加密是公钥加密，私钥解密：但是对于数字签名来说就是私钥生成，公钥验证。

应用场景
(1) 信息加密

收信者是唯一能够解开加密信息的人，因此收信者手里的必须是私钥。发信者手里的是公钥，其它人知道公钥没有关系，因为其它人发来的信息对收信者没有意义。

(2) 登录认证

客户端需要将认证标识传送给服务器，此认证标识 (可能是一个随机数) 其它客户端可以知道，因此需要用私钥加密，客户端保存的是私钥。服务器端保存的是公钥，其它服务器知道公钥没有关系，因为客户端不需要登录其它服务器。

(3) 数字签名

数字签名是为了表明信息没有受到伪造，确实是信息拥有者发出来的，附在信息原文的后面。就像手写的签名一样，具有不可抵赖性和简洁性。

简洁性：对信息原文做哈希运算，得到消息摘要，信息越短加密的耗时越少。

不可抵赖性：信息拥有者要保证签名的唯一性，必须是唯一能够加密消息摘要的人，因此必须用私钥加密 (就像字迹他人无法学会一样)，得到签名。如果用公钥，那每个人都可以伪造签名了。

(4) 数字证书

问题起源：对1和3，发信者怎么知道从网上获取的公钥就是真的？没有遭受中间人攻击？

这样就需要第三方机构来保证公钥的合法性，这个第三方机构就是 CA (Certificate Authority)，证书中心。

CA 用自己的私钥对信息原文所有者发布的公钥和相关信息进行加密，得出的内容就是数字证书。

信息原文的所有者以后发布信息时，除了带上自己的签名，还带上数字证书，就可以保证信息不被篡改了。信息的接收者先用 CA给的公钥解出信息所有者的公钥，这样可以保证信息所有者的公钥是真正的公钥，然后就能通过该公钥证明数字签名是否真实了。