ECDSA在以太坊中的应用

**1.原理**

以太坊数字签名和比特币的关系：

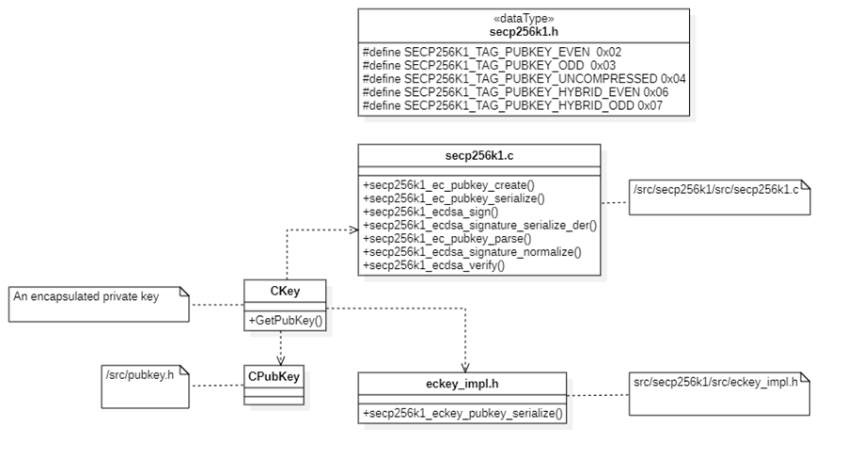
以太坊数字签名，几乎完全沿用了比特币的数字签名算法ECDSA-secp256k1。只有哈希的生成方式不一样，这个之后会说。ECDSA-secp256k1是一种非对称加密算法。

ECDSA

椭圆曲线数字签名算法（ECDSA）是使用椭圆曲线密码（ECC）对数字签名算法（DSA）的模拟。与普通的离散对数问题（DLP）和大数分解问题（IFP）不同，椭圆曲线离散对数问题没有亚指数时间的解决方法。因此椭圆曲线密码的单位比特强度要高于其他公钥体制。  
数字签名算法（DSA）在联邦信息处理标准FIPS中有详细论述，称为数字签名标准。它的安全性基于素域上的离散对数问题。可以看作是椭圆曲线对先前离散对数问题（DLP）的密码系统的模拟，只是群元素由素域中的元素数换为有限域上的椭圆曲线上的点。椭圆曲线离散对数问题远难于离散对数问题，单位比特强度要远高于传统的离散对数系统。因此在使用较短的密钥的情况下，ECC可以达到于DL系统相同的安全级别。这带来的好处就是计算参数更小，密钥更短，运算速度更快，签名也更加短小。  
ECDSA是ECC与DSA的结合，整个签名过程与DSA类似，所不一样的是签名中采取的算法为ECC，最后签名出来的值也是分为r,s。

secp256k1

椭圆曲线算法简单的说就是用X和Y坐标画一个曲线。这个曲线怎么画，需要很多个参数来确定。以太坊使用了一套叫secp256k1的参数确定了椭圆的形状。所以，以太坊的签名算法全称就是是ECDSA-secp256k1。



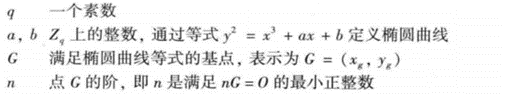
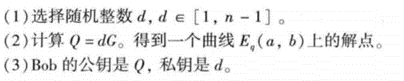
**2.ECDSA处理过程**

1.参与数字签名的所有通信方都使用相同的全局参数，用于定义椭圆曲线以及曲线上的基点

2.签名者首先生成一对公私钥。对于私钥，选择一个随机数或者伪随机数作为私钥，利用随机数和基点算出另一点，作为公钥

3.对消息计算Hash值，用私钥、全局参数和Hash值生成签名

4.验证者用签名者的公钥、全局参数等验证。

全局参数：  
  
密钥生成：  
每个签名者都要生成一对公私钥，假设是Bob  


**3.ECDSA 签名过程**

ECDSA 签名由两个数字（整数）组成：r 和 s。以太坊还引入了额外的变量 v（恢复标识符）。签名可以表示成 {r, s, v}。

在创建签名时，你要先准备好一条待签署的消息，和用来签署该消息的私钥（dₐ）。简化后的签名流程如下：

1. 对待签署消息进行哈希计算，得到哈希值（e）。
2. 生成一个安全的随机数 k。
3. 将 k 乘以椭圆曲线的常量 G，来计算椭圆曲线上的点（x₁, y₁）。
4. 计算 r = x₁ mod n。如果 r 等于 0，请返回步骤 2 。
5. 计算 s = k⁻¹(e + rdₐ) mod n。如果 s 等于 0，请返回步骤 2。

**4. ECDSA 验签过程**

验证过程如下：  
   1、接受方在收到消息(m)和签名值(r,s)后，进行以下运算；  
   2、计算：sG+H(m)P=(x1,y1), r1≡ x1 mod p；  
   3、验证等式：r1 ≡ r mod p；  
   4、如果等式成立，接受签名，否则签名无效。

**5.恢复标识符（“v”）**

v 是签名的最后一个字节，而且不是 27 (0x1b) 就是 28 (0x1c)。恢复标识符非常重要，因为我们使用的是椭圆曲线算法，仅凭r 和 s 可计算出曲线上的多个点，因此会恢复出两个不同的公钥（及其对应地址）。v 会告诉我们应该使用这些点中的哪一个。

在大多数实现中，v 在内部只是 0 或 1，而 27 是在签署比特币消息时加上的任意数。以太坊也接受了这一点。

从 EIP-155 开始，我们还使用链 ID 来计算 v 值。这可以防止跨链重放攻击：以太坊上签署的交易无法在以太坊经典上使用，反之亦然。目前，恢复标识符只用来签署交易而非消息。

**6.交易与签名**

为了产生有效的交易，交易发起者必须使用椭圆曲线数字签名算法对消息进行数字签名。签名应用于交易数据的哈希，而不是交易本身。

要在以太坊签署交易，发起人必须完成以下步骤：

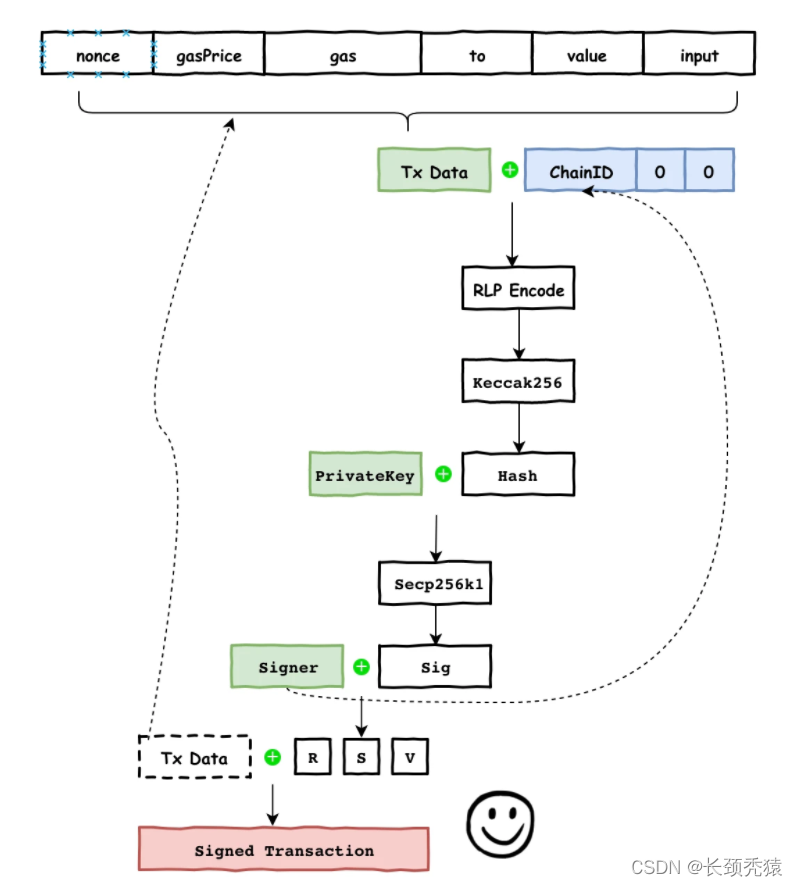
1）创建一个包含九个字段的交易数据结构：nonce，gasPrice，gasLimit，to，value，data，V，R，S

2）生成RLP编码的交易序列化消息

3）计算此序列化消息的Keccak256哈希值

4）计算ECDSA签名，使用发件人的私钥对哈希进行签名

1. 在交易中插入ECDSA签名计算出的R和S值



**7.总结**

椭圆曲线数字签名算法（Elliptic Curve Digital Signature Algorithm，缩写作 ECDSA）是一种基于椭圆曲线密码学的非对称式密码学的数字签名算法。它有两个密钥，一个是公开的密钥，另一个是私有密钥；在我们常用的RSA非对称加密算法中，公钥和私钥都可用于加密或解密，如果公钥用于加密，则私钥用于解密；如果私钥用于加密，则公钥用于解密。但在比特币中，所有的区块数据和交易数据都不需要加密和解密，因为这些数据都是公开透明的，任何人都可以通过区块链网络查询和使用。使用ECDSA主要用于数字签名，它的作用是：由于私钥只由用户自己持有，故可以在比特币的交易转账中，签发交易必定出自于拥有私钥的用户；其他节点可以验证用户签发的交易数据是否有效、中途有否曾被篡改，节点接收者可信赖这些数据确实来自于私钥拥有者。对用户来说确保私钥的安全绝对不能泄露，如果私钥一旦泄露，那么就失去了该私钥对应的钱包地址一切。