基于ECDSA算法的门限签名技术（以太坊非合约多签技术）

一、普通以太坊多签技术及其局限性：

通常的以太坊多签技术都是基于合约来实现的，较为著名的几个钱包 BitGo 钱包、Gnosis 钱包和 Parity 钱包就是这么做的。基于智能合约，更易于实现。然而，一旦发生错误，错误的成本也确实很高。对于私钥的管理者，他是基于对合约的信任来达到目的的，假设合约不可信或者存在漏洞，私钥管理者并没有更多的方法进行调整。

首先，基于智能合约的多重签名机制依赖于拥有单独而唯一的密钥的各方，他们用这些密钥来对一笔交易签名，并且，签名者一个接一个地提供其签名。每个签名者都有自己的密钥，他们用它来签署一笔交易。当验证者 想要验证交易时，他需要分别检查每个签名。

对于门限签名，情况有所不同，因为我们只有一把公钥和一个签名。每个签名者都有相同的公钥和私钥的一个唯一部分。签名者使用一种特殊的通信协议进行协作，我们姑且称之为“分布式签名运算”，该协议不会为了生成签名而暴露私钥，并且只会生成一个签名。

1. 相关概念

RLP：一种序列化的方式，其与网络传输中json的序列化/反序列化有一些不同，RLP不仅兼顾网络传输，其编码特性更确保了编码后的一致性，因为每笔交易过程中要进行Keccak256，如果不能保证编码后的一致性，会导致其Hash值不同，那么验证者就无法验证交易是否由同一个人发出。

Keccak256 ：以太坊的Hash算法，生成32个字节Hash值。

三、门限签名的优点

就验证而言，门限签名比多重签名交易便宜。同时避免链上多签漏洞，降低链上多签手续费。多重签名需要为每个签名者创建一个新的签名，并且成本随着签名者数量的增加而增加。对于门限签名，只有一个签名，因此在签名及验证方面成本是固定的。

再验证者只需检查一个签名即可确定交易是否正确。

四、缺点

第一个陷阱是，尽管门限签名并不是什么新鲜事物，但大多数人仍不熟悉它们。这意味着要花一些时间来接受这项技术。而且，工具还不存在。对于多重签名，我们有一堆工具，比如有良好 UI 支持的硬件钱包。不过，现在我们提供了一个算法，大家可以基于这个算法实现在手机端的云钱包（私钥分布存储）上达到与硬件离线钱包一样的安全等级，当然，你也可以把这个算法直接集成到硬件钱包中。

五、操作步骤

操作步骤如下：

1.用户A离线生成自己的ECDSA私钥，同时生成一个大随机数，通过随机数与椭圆曲线及私钥的运算得到 D1

2. 用户B离线生成自己的ECDSA私钥，同时生成一个大随机数，通过随机数与椭圆曲线及私钥的运算得到 D2

3.用户A将D1发送给用户B,用户B通过D1、D2合成公钥并得到以太坊地址，该公钥完全符合以太坊规范，基于ECDSA。

4.用户A及用户B对待签名信息进行签名运算，分布得到中间值S1,S2

5.用户A或用户B通过S1,S2及待签名信息生成签名结果r，s，得到ECDSA标准签名。

6.所有得到公钥的有用户都可以通过ECDSA验签方法验证用户A与用户B分布式运算得到的签名结果。

六、签名过程

1.构建原始交易对象

nonce: 记录发起交易的账户已执行交易总数。Nonce的值随着每个新交易的执行不断增加，这能让网络了解执行交易需要遵循的顺序，并且作为交易的重放保护。

gasPrice: 该交易每单位gas的价格，Gas价格目前以Gwei为单位（即10^9wei），其范围是大于0.1Gwei，可进行灵活设置。

gasLimit: 该交易支付的最高gas上限。该上限能确保在出现交易执行问题（比如陷入无限循环）之时，交易账户不会耗尽所有资金。一旦交易执行完毕，剩余所有gas会返还至交易账户。

to：该交易被送往的地址（调用的合约地址或转账对方的账户地址）。

value：交易发送的以太币总量。

data: 若该交易是以太币交易，则data为空；若是部署合约，则data为合约的bytecode；若是合约调用，则需要从合约ABI中获取函数签名，并取函数签名hash值前4字节与所有参数的编码方式值进行拼接而成，具体参见文章Ethereum的合约ABI拓展

chainId：防止跨链重放攻击。

1. 签署交易

