以太坊交易详解
Ethereum Transactions

2018.10

让天下没有难学的技术

交易的本质

- 交易是由外部拥有的账户发起的签名消息，由以太坊网络传输，并被序列化后记录在以太坊区块链上。
- 交易是唯一可以触发状态更改或导致合约在EVM中执行的事物。
- 以太坊是一个全局单例状态机，交易是唯一可以改变其状态的东西。
- 合约不是自己运行的，以太坊也不会“在后台”运行。以太坊上的一切变化都始于交易。

让天下没有难学的技术

交易数据结构

交易是包含以下数据的序列化二进制消息：

- **nonce**：由发起人EOA发出的序列号，用于防止交易消息重播。
- **gas price**：交易发起人愿意支付的gas单价（wei）。
- **start gas**：交易发起人愿意支付的最大gas量。
- **to**：目的以太坊地址。
- **value**：要发送到目的地的以太数量。
- **data**：可变长度二进制数据负载（payload）。
- **v,r,s**：发起人EOA的ECDSA签名的三个组成部分。
- 交易消息的结构使用递归长度前缀（RLP）编码方案进行序列化，该方案专为在以太坊中准确和字节完美的数据序列化而创建。

让天下没有难学的技术

交易中的nonce

- 黄皮书定义： 一个标量值，等于从这个地址发送的交易数，或者对于关联code的帐户来说，是这个帐户创建合约的数量。
- nonce不会明确存储为区块链中帐户状态的一部分。相反，它是通过计算发送地址的已确认交易的数量来动态计算的。
- nonce值还用于防止错误计算账户余额。nonce强制来自任何地址的交易按顺序处理，没有间隔，无论节点接收它们的顺序如何。
- 使用nonce确保所有节点计算相同的余额和正确的序列交易，等同于用于防止比特币“双重支付”（“重放攻击”）的机制。但是，由于以太坊跟踪账户余额并且不单独跟踪 UTXO ，因此只有在错误地计算账户余额时才会发生“双重支付”。nonce机制可以防止这种情况发生。

让天下没有难学的技术

并发和nonce

- 以太坊是一个允许操作（节点，客户端，DApps）并发的系统，但强制执行单例状态。例如，出块的时候只有一个系统状态。
- 假如我们有多个独立的钱包应用或客户端，比如 MetaMask和 Geth，它们可以使用相同的地址生成交易。如果我们希望它们都够同时发送交易，该怎么设置交易的nonce呢?
- 用一台服务器为各个应用分配nonce，先来先服务——可能出现单点故障，并且失败的交易会将后续交易阻塞。
- 生成交易后不分配nonce，也不签名，而是把它放入一个队列等待。另起一个节点跟踪nonce并签名交易。同样会有单点故障的可能，而且跟踪nonce和签名的节点是无法实现真正并发的。

让天下没有难学的技术

交易中的gas

- 当由于交易或消息触发 EVM 运行时，每个指令都会在网络的每个节点上执行。这具有成本：对于每个执行的操作，都存在固定的成本，我们把这个成本用一定量的 gas 表示。
- gas 是交易发起人需要为 EVM 上的每项操作支付的成本名称。发起交易时，我们需要从执行代码的矿工那里用以太币购买 gas 。
- gas 与消耗的系统资源对应，这是具有自然成本的。因此在设计上 gas 和 ether 有意地解耦，消耗的 gas 数量代表了对资源的占用，而对应的交易费用则还跟 gas 对以太的单价有关。这两者是由自由市场调节的：gas 的价格实际上是由矿工决定的，他们可以拒绝处理 gas 价格低于最低限额的交易。我们不需要专门购买 gas ，只需将以太币添加到帐户即可，客户端在发送交易时会自动用以太币购买汽油。而以太币本身的价格通常由于市场力量而波动。

让天下没有难学的技术

gas的计算

- 发起交易时的 gas limit 并不是要支付的 gas 数量，而只是给定了一个消耗 gas 的上限，相当于“押金”
- 实际支付的 gas 数量是执行过程中消耗的 gas （gasUsed），gas limit 中剩余的部分会返回给发送人
- 最终支付的 gas 费用是 gasUsed 对应的以太币费用，单价由设定的 gasPrice 而定
- 最终支付费用 totalCost = gasPrice * gasUsed
- totalCost 会作为交易手续费（Tx fee）支付给矿工

让天下没有难学的技术

交易的接收者（to）

- 交易接收者在to字段中指定，是一个20字节的以太坊地址。地址可以是EOA或合约地址。
- 以太坊没有进一步的验证，任何20字节的值都被认为是有效的。如果20字节值对应于没有相应私钥的地址，或不存在的合约，则该交易仍然有效。以太坊无法知道地址是否是从公钥正确派生的。
- 如果将交易发送到无效地址，将销毁发送的以太，使其永远无法访问。
- 验证接收人地址是否有效的工作，应该在用户界面一层完成。

让天下没有难学的技术

交易的 value 和 data

- 交易的主要“有效负载”包含在两个字段中：value 和 data。交易可以同时有 value 和 data，仅有 value，仅有 data，或者既没有 value 也没有 data。所有四种组合都有效。
- 仅有 value 的交易就是一笔以太的付款
- 仅有 data 的交易一般是合约调用
- 进行合约调用的同时，我们除了传输 data， 还可以发送以太，从而交易中同时包含 data 和 value
- 没有 value 也没有 data 的交易，只是在浪费 gas，但它是有效的

让天下没有难学的技术

向 EOA 或合约传递 data

- 当交易包含数据有效负载时，它很可能是发送到合约地址的，但它同样可以发送给 EOA
- 如果发送 data 给 EOA，数据负载（data payload） 的解释取决于钱包
- 如果发送数据负载给合约地址，EVM 会解释为函数调用，从 payload 里解码出函数名称和参数，调用该函数并传入参数
- 发送给合约的数据有效负载是32字节的十六进制序列化编码：

  ——函数选择器：函数原型的 Keccak256 哈希的前4个字节。这允许 EVM 明确地识别将要调用的函数。

  ——函数参数：根据 EVM 定义的各种基本类型的规则进行编码。

让天下没有难学的技术

特殊交易：创建（部署）合约

- 有一中特殊的交易，具有数据负载且没有 value，那就是一个创建新合约的交易。
- 合约创建交易被发送到特殊目的地地址，即零地址0x0。该地址既不代表 EOA 也不代表合约。它永远不会花费以太或发起交易，它仅用作目的地，具有特殊含义“创建合约”。
- 虽然零地址仅用于合同注册，但它有时会收到来自各种地址的付款。这种情况要么是偶然误操作，导致失去以太；要么是故意销毁以太。
- 合约注册交易不应包含以太值，只包含合约的已编译字节码的数据有效负载。此交易的唯一效果是注册合约。

让天下没有难学的技术

Q&A

让天下没有难学的技术


尚硅谷