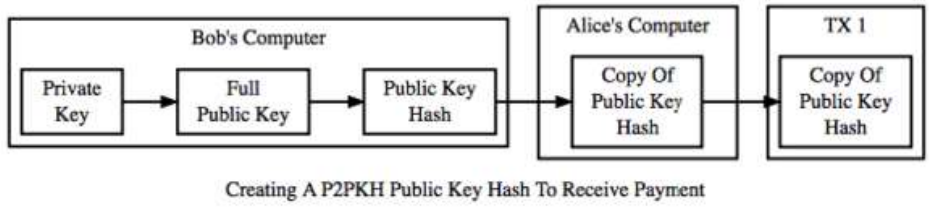


Creating A P2PKH Public Key Hash To Receive Payment

比特币脚本(Script)演进(I)——P2PK && P2PKH

亲王来啦
在草原中为羊写诗
+ 关注他

比特币交易简述

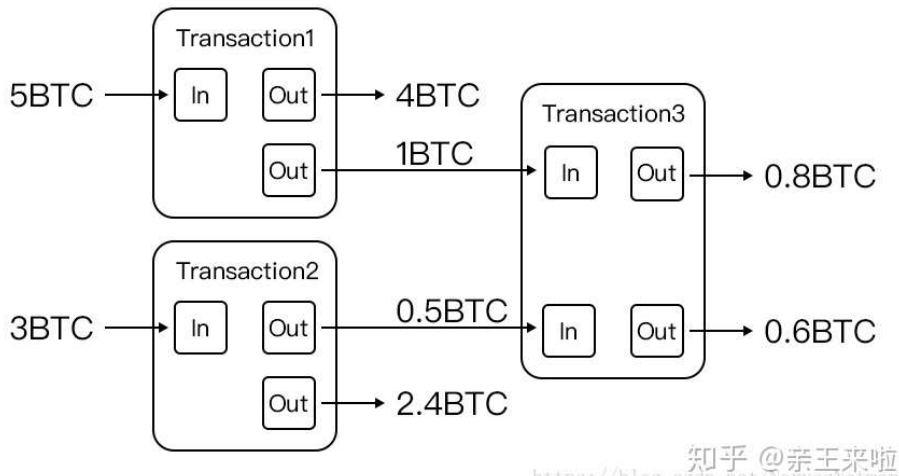
比特币采用的是UTXO模型。我们来看看交易1，这笔交易**销毁**了5 BTC的UTXO并**铸成**了4 BTC和1 BTC两个UTXO。一看下去感觉没什么，就是5=4+1嘛。 然而，经过深入思考，我们发现了一切价值传递的本质：价值并没有发生变化，但所有权变化了。

- 比特币中In称为**解锁脚本scriptSig**。如果你要使用UTXO，如交易1中5 BTC的UTXO，那么你需要证明自己有这5 BTC的所有权，而这个证明就放在In里面。一个简单的所有权证明例子：**5 BTC是由私钥/地址k所持有的，那么使用该UTXO交易中的In只需要包含k的数字签名即可**。整个过程就是让锁定的UTXO解锁，然后才能被使用，而解锁的输入逻辑放在In中，因此被称为解锁脚本。
- 比特币中Out称为**锁定脚本ScriptPubKey**。在上面，你已经证明了拥有5 BTC的UTXO的所有权，那么你现在做的是所有权的重新分配。这其实就是交易的内涵。继续上面的例子，小辉解锁了5 BTC后，打算给小青1 BTC买零食。那么，小辉铸成了两个新的UTXO：（1）能由小青解锁的1BTC的UTXO；（2）能由小辉解锁的4BTC的UTXO（找零）。注意，“能由小青解锁”和“能由小辉解锁”就是交易1中Out所放的东西。Out被称为锁定脚本，它指代的是满足什么条件才对这个UTXO有所有权。
- 交易被打包进主链后，交易生效。例如，交易1**销毁**了5 BTC的UTXO并**铸成**了4 BTC和1 BTC两个UTXO。

交易1的状态转移：

- 转移前状态：`Out0: (5BTC, <小辉能用>)`
- 权限证明：`In: (<我是小辉>)`，`Out0<In>` 返回 `true`
- 转移后状态：`Out1: (4BTC, <小辉能用>)` 以及 `Out2: (1BTC, <小青能用>)`（状态转移后 `Out0` 被销毁了，铸成了 `Out1` 和 `Out2`）

总的来说，`<解锁脚本> <锁定脚本>` 返回 `true`，那么**构造出解锁脚本的人**就会对**锁定脚本对应的UTXO**拥有所有权。这个判断过程是基于一个栈虚拟机以及非图灵完备的脚本语言Script。



Pay-to-PublicKey（P2PK）

P2PK要实现的就是最简单最常用的点对点转账，但是后面被P2PKH所取代了。

锁定脚本：`<Public Key> OP_CHECKSIG`

解锁脚本：`<Signature from Private Key>`

验证时的组合脚本：`<Signature from Private Key> <Public Key> OP_CHECKSIG`

验证过程:

Stack	Script	Description
`Empty.`	`<sig> <pubKey> OP_CHECKSIG`	scriptSig and scriptPubKey are combined.
`<sig> <pubKey>`	`OP_CHECKSIG`	Constants are added to the stack.
`true`	`Empty.`	Signature is checked for top two stack items.

Pay-to-PublicKey Hash（P2PKH）

用于取代P2PK，P2PKH实现的也是最简单最常用的点对点转账。

锁定脚本：`OP_DUP OP_HASH160 <Public KeyHash> OP_EQUAL OP_CHECKSIG`

解锁脚本：`<Signature> <Public Key>`

验证时的组合脚本：`<Signature> <Public Key> OP_DUP OP_HASH160 <Public KeyHash> OP_EQUAL OP_CHECKSIG`

验证过程：

Stack	Script	Description
`Empty.`	`<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG`	scriptSig and scriptPubKey are combined.
`<sig> <pubKey>`	`OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG`	Constants are added to the stack.
`<sig> <pubKey> <pubKey>`	`OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG`	Top stack item is duplicated.
`<sig> <pubKey> <pubHashA>`	`<pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG`	Top stack item is hashed.
`<sig> <pubKey> <pubHashA> <pubKeyHash>`	`OP_EQUALVERIFY OP_CHECKSIG`	Constant added.
`<sig> <pubKey>`	`OP_CHECKSIG`	Equality is checked between the top two stack items.
`true`	`Empty.`	Signature is checked for top two stack items.

中本聪决定用P2PKH取代P2PK的原因

- 椭圆曲线加密体系（ECC）容易受到**用于解决椭圆曲线上离散对数问题的改进Shor算法**的攻击。简单来说，未来量子计算机可能能够通过公钥计算出私钥。P2PKH的设计中，只有币被花费时公钥才被暴露（假设地址不被重用），那么这种攻击将变得无效。
- 用P2PK中的更小的哈希（20字节）取代P2PK中的公钥作为地址会更方便打印，也更容易嵌入到

发布于 2019-12-18

比特币 (Bitcoin)　区块链(Blockchain)　智能合约

文章被以下专栏收录

双花的区块链　关注专栏

推荐阅读

真正理解以太坊智能合约

前言：智能合约这个词你可能听过无数遍，但有多少人真正理解什么是智能合约？本文帮你深入理解以太坊智能合约。本文作者是 Gjermund Bjaanes，由“蓝狐笔记”社群的“Dyna”翻译。 你可...

蓝狐笔记

bitcoin 源码解析 - 交易 Transaction(四) - Script2

bitcoin 源码解析 - 交易 Transaction(四) - Script2现在发现写文章真是好没有什么动力··· 所以就写的简洁些吧·· 随心说一些最关键的点，细节就不强调了。 接上一文的《bitcoin 源码解...

金晓　发表于链块与分散...

以太坊创新高，能涨多少？ | 疑问解答

道说区块链　发表于区块链投资...

比特币和以太坊的记账方式——UTXO和账户余额

闪电

1 条评论

切换为时间排序

写下你的评论...

天生我万古长夜　2019-12-24

更清晰了，你是玩币的吗？

赞

▲ 赞同　▼　1 条评论　分享　喜欢　收藏　申请转载　···