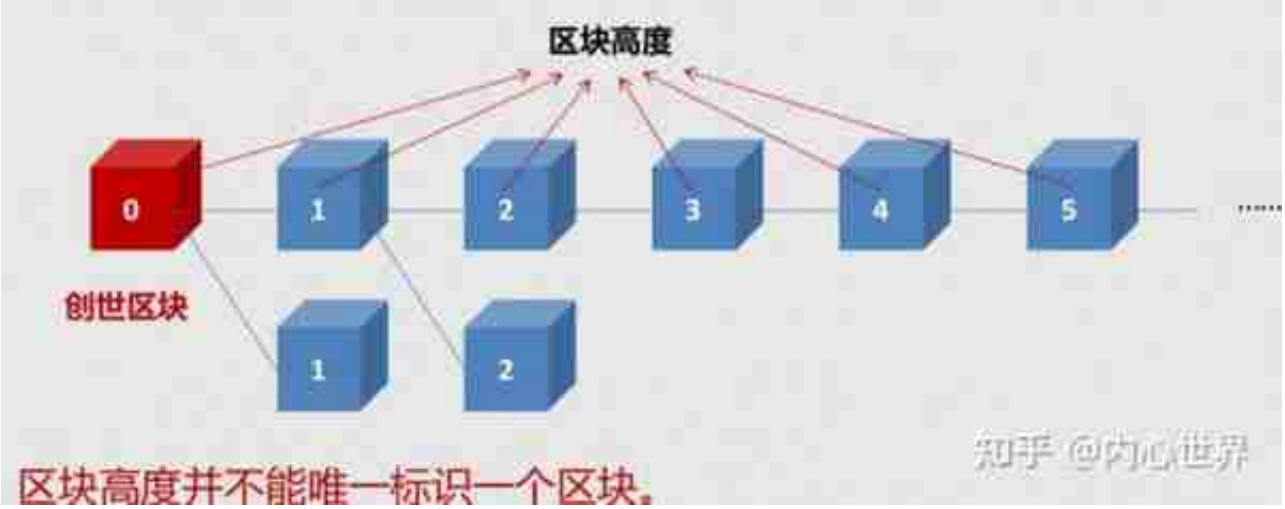
1.基本知识

1.1 区块

区块是一种容器数据结构，这种数据结构被包含在公开账簿里，聚合了交易信息；而这个公开账簿是区块链；区块是构成区块链的基本单元，由区块头和区块体构成。



为什么区块高度不能唯一标识一个区块？

因为在同一时间，有可能存在两个或两个以上的区块，他们相对于整条区块链来说，处于平等的位置，所以他们有着相同的区块高度。

2.货币的发展阶段

2.1 以物易物

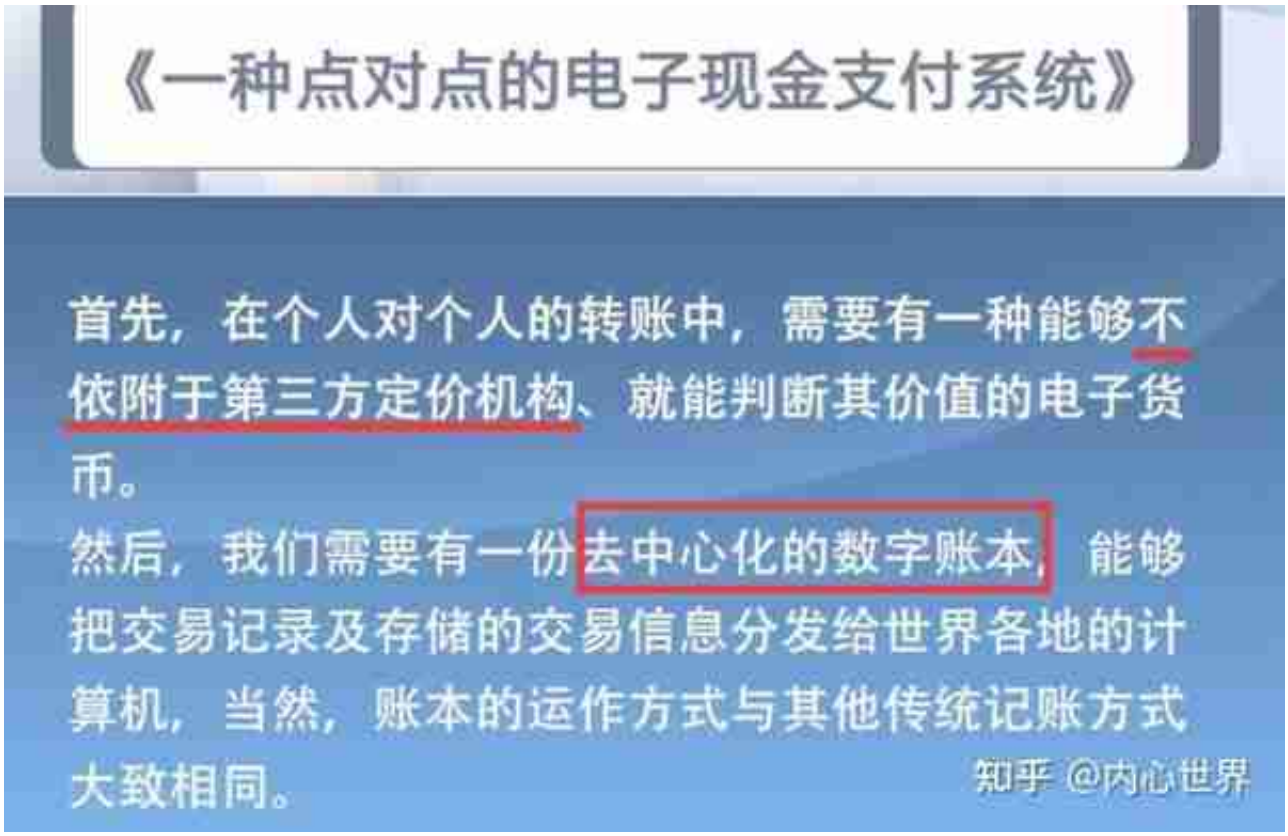
此时我们通过贝壳或者一些金银来作为等价物作为中介来交换，此时就不需要像之前的以物换物那样满足双重偶然性了；但是由于实物容易磨损、不易携带、数量有限等缺点进而发展出了纸币

2.3 纸币

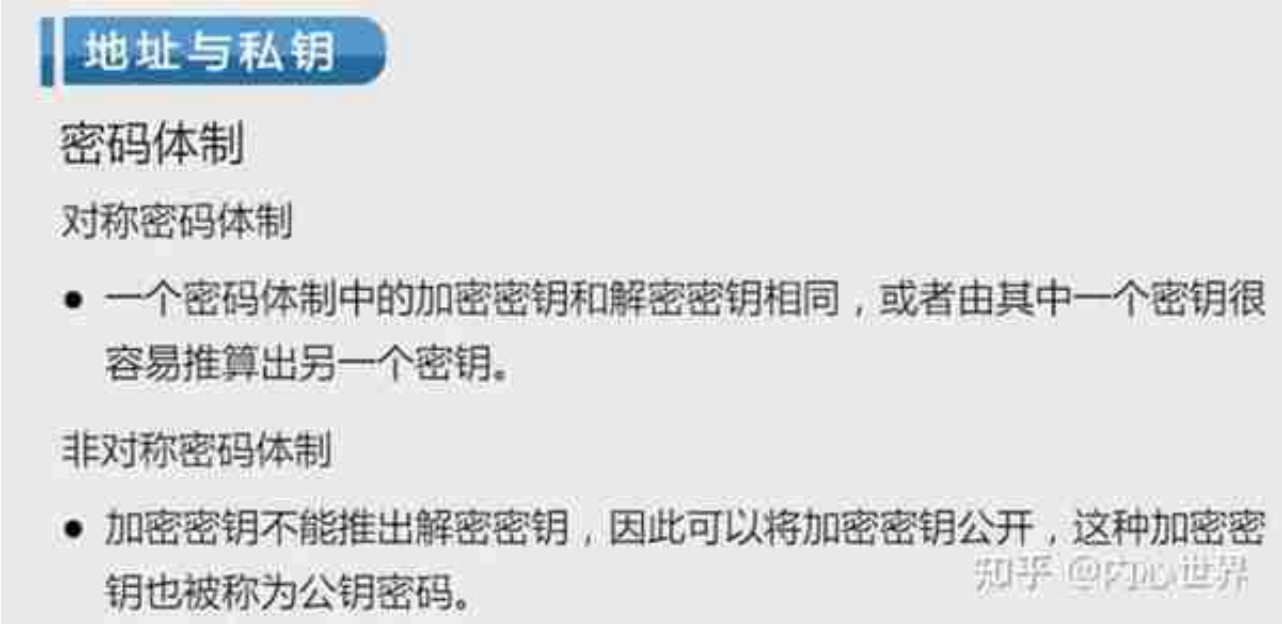
移动电子支付的实质为记账货币，是通过银行、第三方支付机构、央行负责记账的，而央行具有整个国家大账本的记账权，本质上就是一种中心化的记账方式。无论

是使用微信或者支付宝，我们都不得不完全信任第三方，因为我们的钱存储在他们这仅显示为数字，假如第三方数据遭受到篡改，我们的财产就会遭受到损失。

2.5 比特币诞生



为了满足比特币的要求，区块链技术应运而生了，或者说比特币是区块链兴起的源头，是区块链最早、最成功的应用。

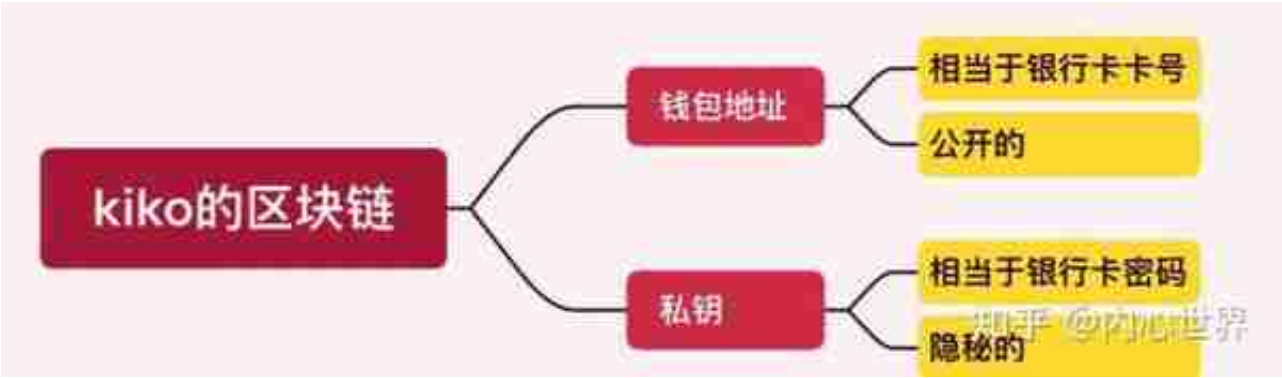


3.2 公钥与私钥

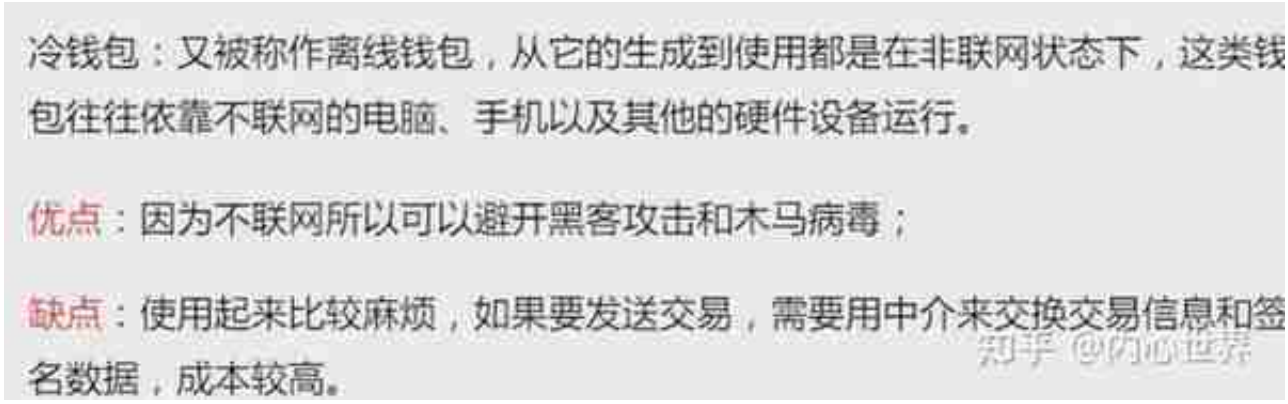
比特币采用的就是非对称密码体制 。

公钥加密发明于20世纪70年代，自从公钥加密被发明之后，一些不可逆的数学函数被提出，就是说只能向一个方向计算，但不可以向相反方向倒推。

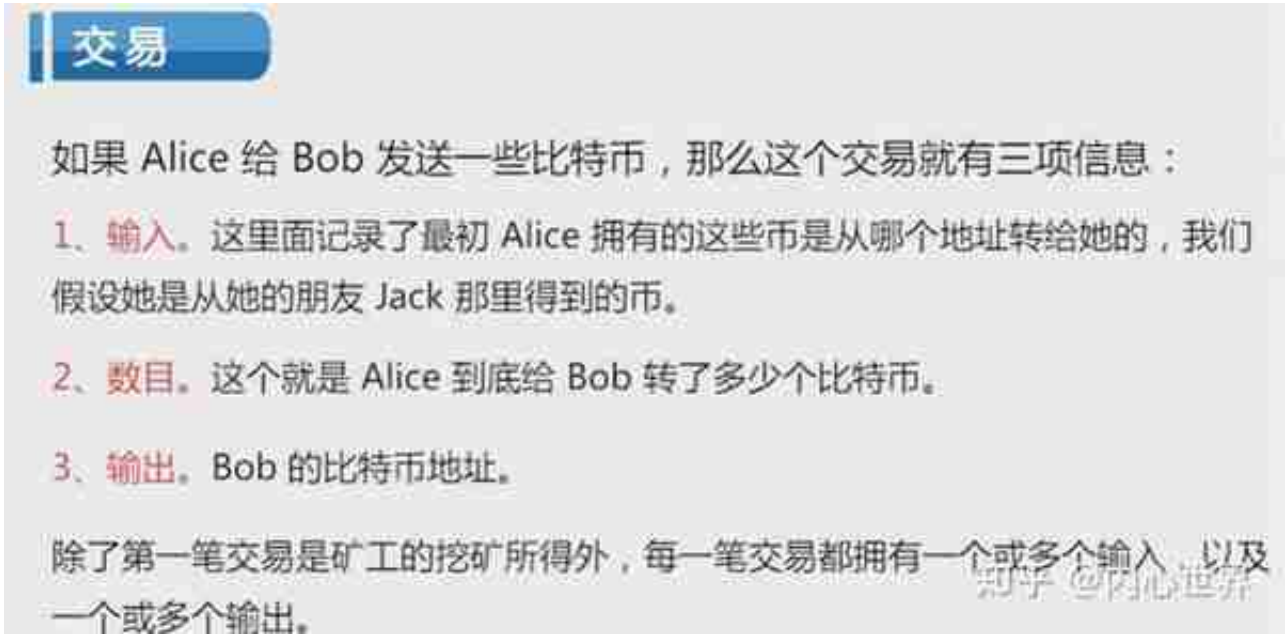
在比特币中，公钥用于接收比特币，私钥用于生成其对应地址上支付比特币所必需的签名，以唯一确定这些比特币的所有权。具体到比特币地址的生成，其实是先产生私钥，私钥通过椭圆曲线算法这种不可逆的函数来产生公钥，公钥经过一系列不可逆的运算再来产生地址。

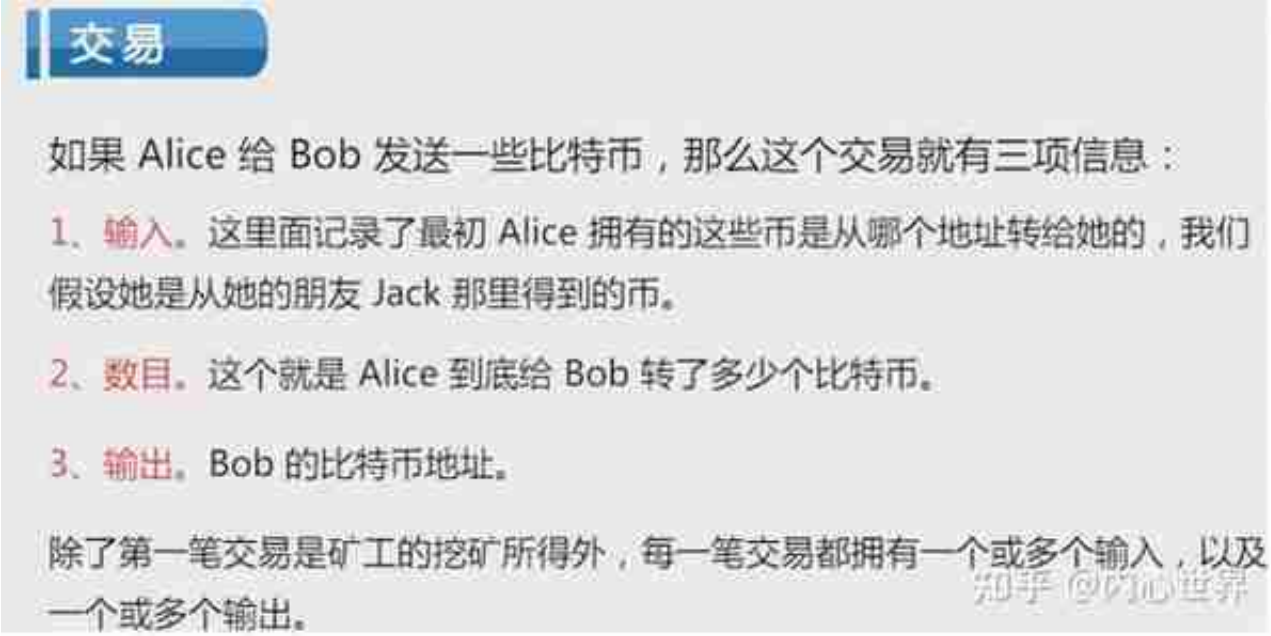


钱包定义


冷钱包：又被称作离线钱包，从它的生成到使用都是在非联网状态下，这类钱包往往依靠不联网的电脑、手机以及其他的硬件设备运行。
优点：因为不联网所以可以避开黑客攻击和木马病毒；
缺点：使用起来比较麻烦，如果要发送交易，需要用中介来交换交易信息和签名数据，成本较高。
知乎 @内心世界

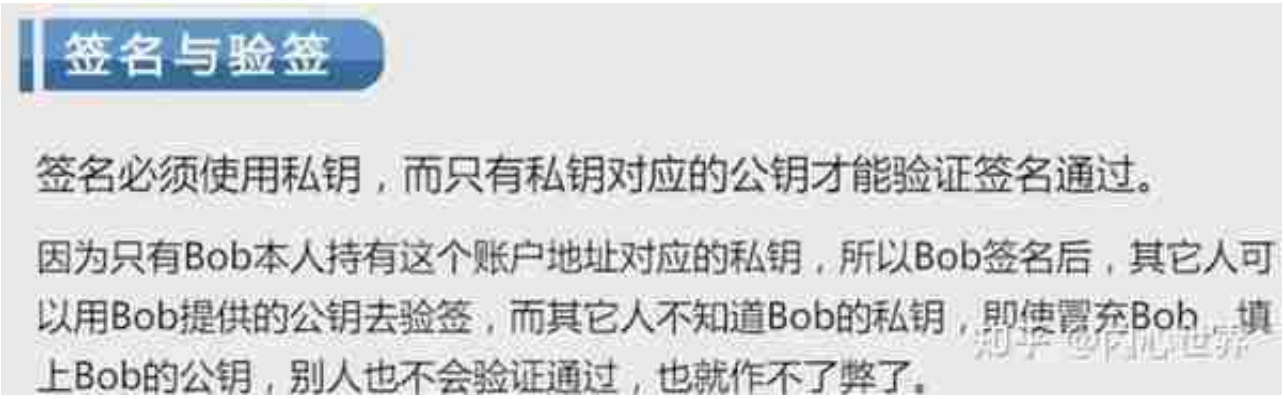
热钱包


交易
如果 Alice 给 Bob 发送一些比特币，那么这个交易就有三项信息：
1、输入。这里面记录了最初 Alice 拥有的这些币是从哪个地址转给她的，我们假设她是从她的朋友 Jack 那里得到的币。
2、数目。这个就是 Alice 到底给 Bob 转了多少个比特币。
3、输出。Bob 的比特币地址。
除了第一笔交易是矿工的挖矿所得外，每一笔交易都拥有一个或多个输入，以及一个或多个输出。
知乎 @内心世界



Q1：请写出Alice和Bob交易时产生的“输入”中的信息，及Bob的UTXO值

Q2：请写出Bob与Tom和Jimmy交易时产生的“输入”中的信息，及Bob的UTXO值



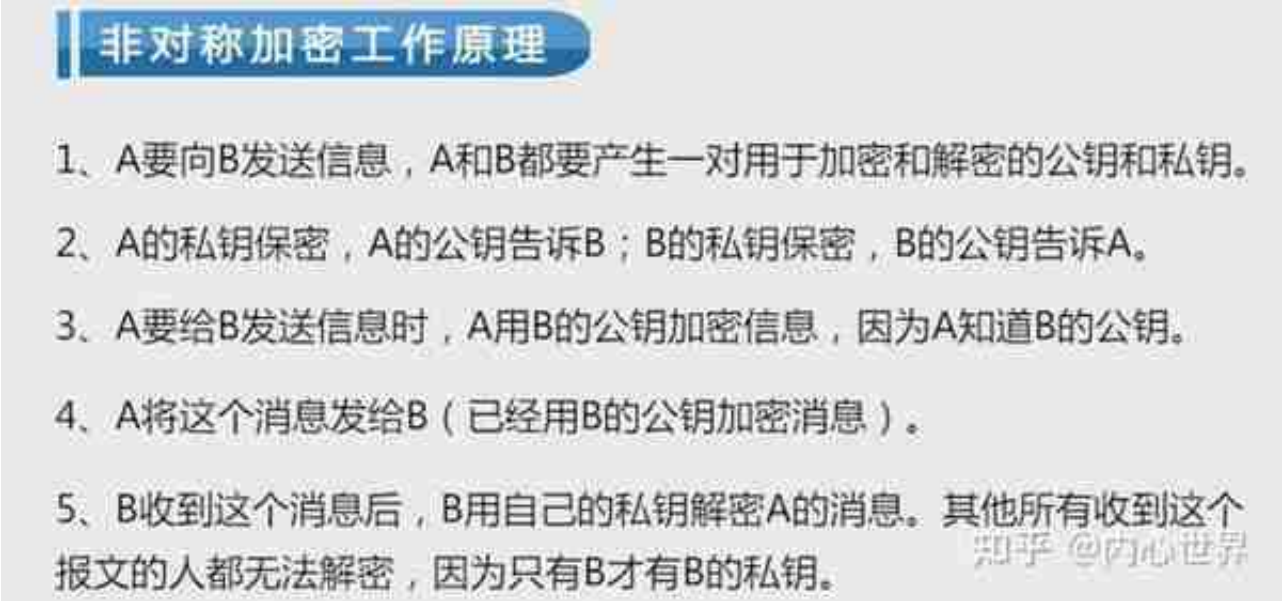
多方签名

比特币中还支持多方签名。如果Bob要使用Alice转给Bob的这两笔交易，那么不仅需要Bob签名，还需要Alice签名，这样万一其中一个人的私钥被盗，也不会丢失比特币的！

4.非对称加密算法

4.1 为什么要使用非对称加密？

4.2 什么是非对称加密？



4.4 非对称加密的应用

比特币的所有权

我们平时所说的某人对一个比特币拥有所有权，事实指的是他拥有这个比特币地址所对应的私钥

交易验证

当使用一个UTXO时，用户要提供这个UTXO中描述的地址对应的公钥、同时用这个公钥对应的私钥对这个交易进行签名，这样比特币的接收者才能去验证这笔交易是否有效。

5.如何避免记假账？

在比特币交易中有三个保障来避免记假账：

1.用私有密钥对交易信息签名，必须用配对的公共密钥验证签名，私用密钥的使用者必须是付款人（采用非对称加密算法）

2.被签名的交易信息在网络上进行广播，所有参与到比特币网络的人都可以接收到这笔交易信息，并且可以对交易信息进行验签，确保交易是合法的（采用非对称加密算法）

3.接收到交易信息后，大家会按照约定的规则生成区块，就是一个数据块，这个数据块中包括所有的交易明细信息，按照merkle树的方式组装起来（所有的交易数据是按照merkle树的方式组装起来，merkle树的数据结构可以很好的保证数据安全）

6.哈希运算与难以篡改

6.1 哈希算法

相同的数据输入将得到相同的结果。输入数据只要稍有变化（比如数据中一个1变

成了0）则将得到一个于差万别的结果，且结果无法事先预知。具体来说就是哈希算法将数据打乱、混合、压缩成一个摘要，使得数据量变小，重新创建一个叫做哈希值的指纹。

2.可以根据任意长度的消息计算出固定长度的散列

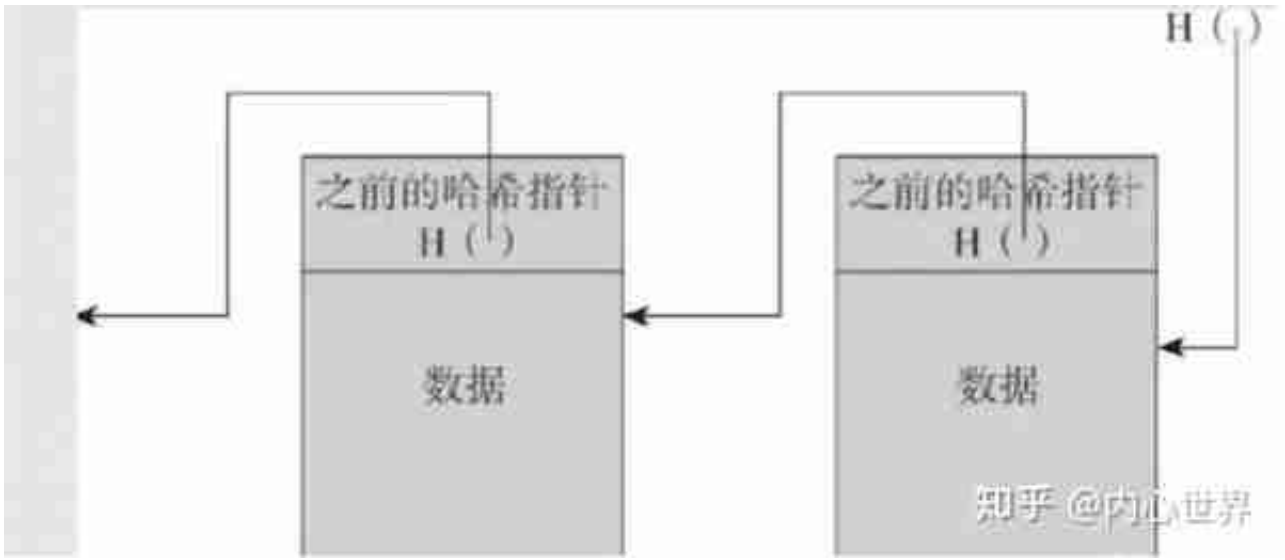
哈希算法生成的哈希值的长度必须是固定的，而且是针对任意长度的输入数据。这一性质保证了哈希算法的易用性。

因为哈希算法的一个使用目的就是用最后的哈希值来代表输入数据，那么最后的哈希值很长、不固定或跟原始数据一样长，那么通过哈希值的对比来确定原始数据就跟直接校验原始数据一样费力没有区别了。最后的长度要控制在一个固定的、合适的值，才能体现出哈希值作为原数据指纹或者说摘要的特征。

3.不同的输入就有不同的输出

哈希算法被发明的目的之一就是根据此需求来的：可以实现对数据完整性和一致性的判定，只有一模一样的数据经过同一个哈希运算得到的哈希值才是一样的。

无论输入数据的长度大小，都会产生定长的哈希值，这样比对结果就比较快速方便，对于数据的防篡改、防缺失都是很好的校验方式。



哈希指针与区块链的关系

可见哈希运算能够帮助识别区块链是否被篡改，区块链的哈希指针能够唯一而精准地识别一个区块，区块链中任意结点通过简单的哈希运算，都可以获得这个区块的哈希指针，计算出的哈希值没有变化就意味着区块链中的信息没有被篡改。另外，哈希运算还能够帮助把各个区块串连成区块链。每个区块都包含上一个区块的哈希值和下一个区块的值，就相当于通过上一个区块的哈希值挂钩到这一个区块，通过这一个区块的哈希值挂钩到下一个区块，然后自然而然形成一个链式结构的区块链

6.6 Merkle树

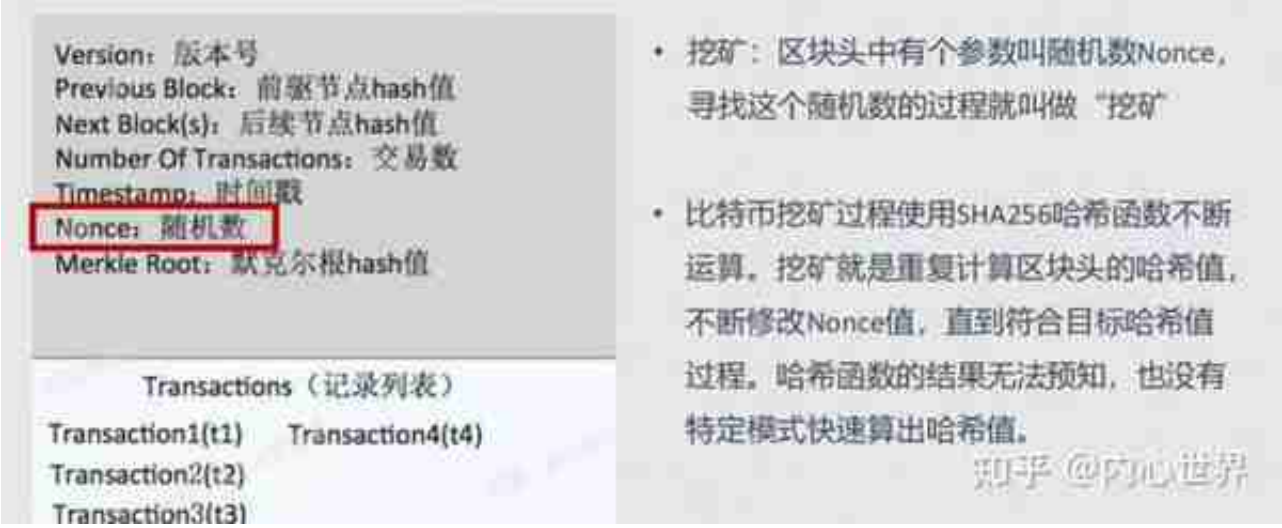
什么是Merkle树呢？

它使用单向哈希：哈希树的顶部为顶部哈希（根哈希/主哈希），它是通过并联两个子哈希来往树上爬直到找到根哈希。

Merkle树的作用

1.可以快速定位每笔交易，由于交易在存储上是线性存储，定位到某笔交易会需要遍历，效率低时间慢，通过这样的二叉树可以快速定位到想要找的交易。

2.在不需要存储整个数据的情况下，就可以简明地核实交易是否被篡改，从交易到每个二叉树的哈希值，有任何一个数字的变化都会导致根哈希的变化。同时，如果有错误，可以快速定位到错误的地方。



7.双花问题与UTXO

7.1 双花问题

7.3 UTXO机制

UTXO的概念

UTXO是Unspent Transaction Outputs的缩写，全称叫做“尚未使用的交易输出”。

U代表Unspent，表示未支付的或尚未使用的，“尚未支付”指的是这个交易输出还没有出现在其它交易的输入端。

TX是transaction的缩写，交易是从一个比特币钱包向另一个钱包转账，是唯一可以改变比特币所有权的方式，包含了输入、数目和输出这些基本内容。

O是Output，是输出的意思，TXO连起来就是指交易输出。

比特币与UTXO的关系

比特币里并没有用户帐户的概念。 我们说自己有多少比特币实际上是指的我们拥有所有权的那些UTXO中所指明的比特币的数量。Alice有10个比特币，本质上来说其实是当前区块链账本中，有若干笔交易的UTXO项收款人写的是Alice的地址，而这些UTXO项的数额总和是10。

比特币交易的规则

1.除了 coinbase交易之外，所有的资金来源都必须来自前面某一个或者几个交易的UTXO。

2.任何一笔交易的交易输入总量必须等于交易输出总量，等式两边必须配平；即每一次交易的输入值都必须全部花掉，不能只花掉部分。

比如，我要转出比特币给你的钱包地址中只有 8
个比特币，那么很简单，我发起一个交易，把这 8
个比特币转到你的钱包地址中，我签名确认这个交易。但假如我的钱包地址中有 25 个比特币，那我发起的交易就不是转给你 8
个比特币，然后自己的钱包地址中还剩下 17
个比特币。这时，我发起的交易是：从我的钱包地址中转 8 个比特币给你，同时转 17 个比特币给我的同一地址。

我们这里通过一个比特币交易的例子加深对于UTXO的理解：

1. 假设Alice之前通过挖矿获得了
12个比特币，在她的地址中，这些比特币是某个币基交易的 UTXO

普通交易
交易号：#2001

交易输入	交易输出（UTXO）		
资金来源	第几项	数额	收款人地址
1001（1）	（1）	2	Bob的地址
	（2）	10	Alice的地址

8.共识机制

8.1 什么是共识？

共识，从语文的角度进行理解，即许多不同的人对同一件事情达成一样的或者至少说方向一致的看法。这个解释同样适用于比特币网络。

达成共识的主体

当前这个区块链中的一些节点，到底哪些节点需要达成一致，这是一个需要考虑的问题。

对什么达成一致

共识机制涉及了区块该如何生成以及生成之后如何选择的问题。 ——区块和交易

在区块链当中，由于每个节点都是平等的，没有一个中心机构的存在，因此这时候就需要通过共识机制来达成节点间的一致。

8.2 什么是共识算法？

共识算法是为了达成共识所依据的一种规则，是筛选出具有代表性的节点的方法。为此，区块链设计了一定的底层算法，通过这个特定的算法来选出那个可以生成新区块的节点，同时对于每一笔在这条区块链上进行的交易是否准许完成进行了约束和规定，也就是共识算法。共识算法规定了，下一个新区块由哪个矿工生成，同时，在这条区块链上一笔交易要达成，需要被共识算法选出的部分节点达成一致的观点,也就是说对于一笔交易，如果利益不相干的若干个结点能够达成共识，就可以认为全网对此也能够达成共识。

所以，在工作量证明中，你工作的时间越长，工作时采用的设备越先进，你的工作量就越高，你收获的也会越多。虽然短期看可能有运气因素，但是宏观长期来看是公平的，谁工作付出的多，谁得到的就多。

9. (51%攻击)

9.1 算力

算力，也叫哈希率，是用来衡量进行哈希运算的能力的指标，或者说进行一次哈希计算所需要使用的时间。如果说网络达到了10T
hash/s（10T哈希每秒）的哈希率时，就意味着它可以每秒进行10万亿次计算。

哈希碰撞：解出随机哈希值不断尝试的过程。

一个挖矿机每秒钟能做这种碰撞的次数，代表其算力。

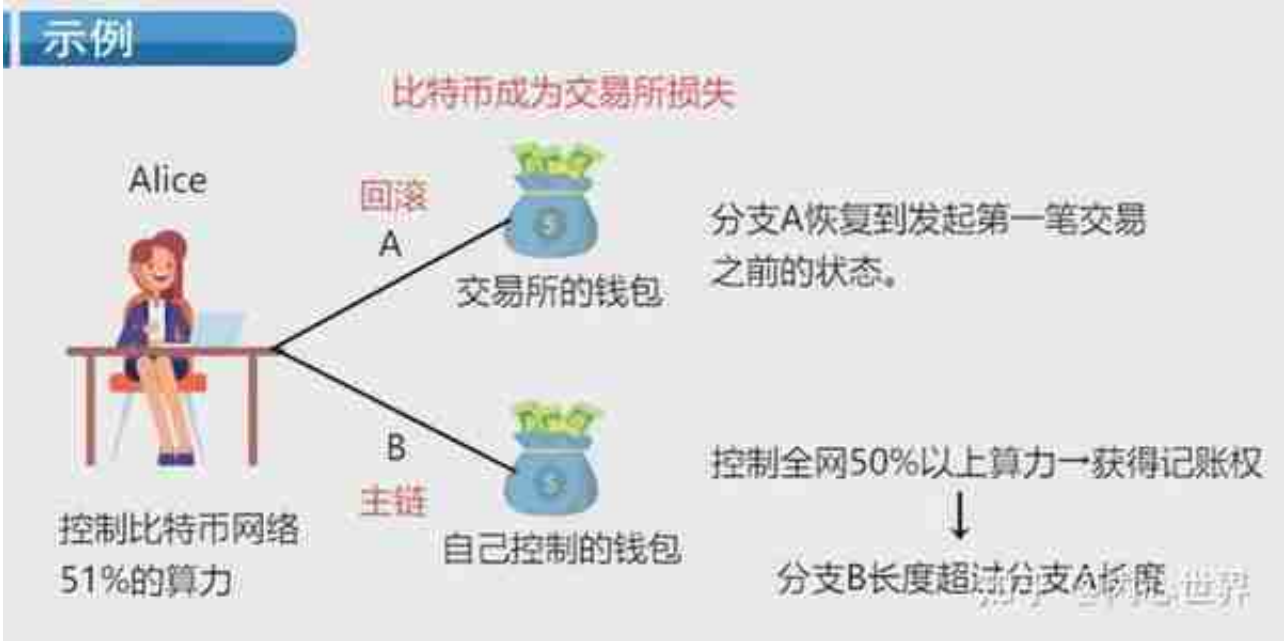
矿工进行挖矿所使用的机器越先进，算力就会越高。

9.2 区块链转账的基本原理

整个区块链网络之所以能运行，靠的就是整个网络的“矿工“，因为他们通过算力解答加密难题，从而挖出新的区块。挖出区块后，他们就有权力将转账信息放入区块中，然后完成这笔转账，这就是区块链转账的基本原理。

9.3 51%攻击

51%攻击（Majority Attack），就是说在整个网络中有人的算力超过了全网的50%。那么他就可以尝试对区块链的状态进行修改，进行反向交易，实现双花。



这时候，由于交易回滚，分支A恢复到Alice发起第一笔交易之前的状态，所以她之前换成现金的那些比特币又回到了自己手里。于是这些比特币就成为了交易所的损失。最后，Alice把这些比特币发到自己的另一个钱包。就这样，她凭借51%以上的算力控制，实现了同一笔token的“双花”。

9.4 什么时候才会发生51%攻击呢？

某个矿池的算力过大

ps：矿池（Mining Pool），为了将少量算力合并联合运作所建立的网站。

由于现在生产新区块越来越难，出现了矿工间合作组成矿池，汇聚数以千计参与者们的算力，一起参与挖矿并分享奖励的行为。在这种情况下，一旦某个矿池汇集了过多的矿工，其算力超过了全网的50%，这时就出现了51%攻击的风险

有无限的资本

拥有无限的资本，购买无限多的设备，就可以发起51%的攻击

9.5 51%攻击悖论

虽然看起来51%攻击非常恐怖，不过这通常仅存在于理论情况下。因为如果你想做到，首先需要足够的钱去掌控全网51%的算力，这将是非常大的一笔投资。其次在攻击后，币种价格会受到影响，你要卖出非常多的币才能达到收支平衡。这就是51%攻击悖论。