

区块链基础及应用

Chapter 4 如何存储和使用比特币

苏 明



概览

- 4.1 简单的本地存储
- 4.2 热存储与冷存储
- 4.3 密钥分存和密钥共享
- 4.4 在线钱包和交易所
- 4.5 支付服务
- 4.6 交易费
- 4.7 货币兑换市场



4.1 简单的本地存储

存储比特币其实就是如何保存和管理比特币私钥

1. 可获取性
2. 安全性
3. 便利性

4.1 简单的本地存储

比特币钱包软件

- 管理比特币和私钥信息并让你方便使用的一个应用软件

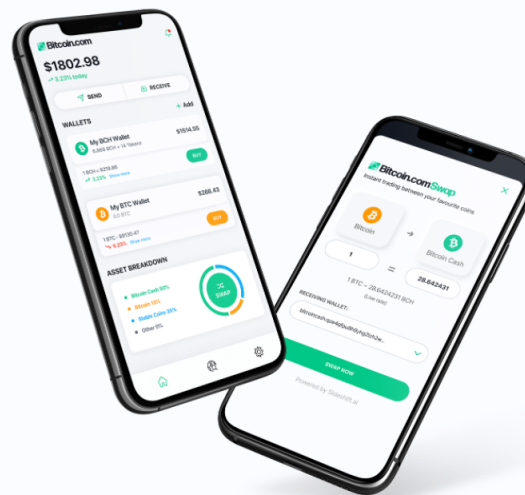
New to Bitcoin? Download the wallet to get started.

[Why do I need a wallet?](#)

Your journey into cryptocurrency starts with downloading a wallet. The Bitcoin.com Wallet allows you to safely store and spend your Bitcoin and Bitcoin Cash, along with other crypto assets.



[Text me the app](#)





4.1 简单的本地存储

编码解码

- **Base58编码**
- **Base64**是一种基于**64**个可打印字符来表示二进制数据的表示方法
- 包含**58**个字符的字符集：去掉几个比较容易混淆的字母
- 相比**Base64**，**Base58**不使用数字"**0**"，字母大写"**O**"，字母大写"**I**"，和字母小写"**l**"，以及"**+**"和"**/**"符号



4.1 简单的本地存储

设计Base58主要的目的：

1. 避免混淆。在某些字体下，数字0和字母大写O，以及字母大写I和字母小写l会非常相似。
2. 不使用"+"和"/"的原因是非字母或数字的字符串作为帐号较难被接受。
3. 没有标点符号，通常不会被从中间分。
4. 大部分的软件支持双击选择整个字符串。



4.1 简单的本地存储

■ Satoshi Nakamoto: base58.h

```
// Why base-58 instead of standard base-64 encoding?  
// - Don't want 0011 characters that look the same in some fonts and  
//     could be used to create visually identical looking account numbers.  
// - A string with non-alphanumeric characters is not as easily accepted as an account number.  
// - E-mail usually won't line-break if there's no punctuation to break at.  
// - Doubleclicking selects the whole number as one word if it's all alphanumeric.
```

4.1 简单的本地存储

QR码：帐户交易处理更方便

1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa

The address that received the very first Bitcoin block reward in the genesis block, base58 encoded.



a QR code representing an actual Bitcoin address



4.1 简单的本地存储

虚荣地址 (Vanity Address)

the gambling website Satoshi Bones:

- **1bones**EeTcABPjLzAb1VkFgySY6Zqu3sX

(all regular addresses begin with the character 1, indicating pay-to-pubkey-hash.)



4.1 简单的本地存储

虚荣地址 (Vanity Address)

How to Generate this type of Address?

How to **Speedup**?

$$g^{x+1} = g^x * g$$

椭圆曲线点乘优化



4.2 热存储和冷存储

- 热存储

存放在个人电脑里（像把钱放在钱包里）方便但不安全

- 冷存储

不联入互联网，封存起来（像保险箱），安全性高但不方便



4.2 热存储和冷存储

分层确定性钱包（hierarchical deterministic wallet）

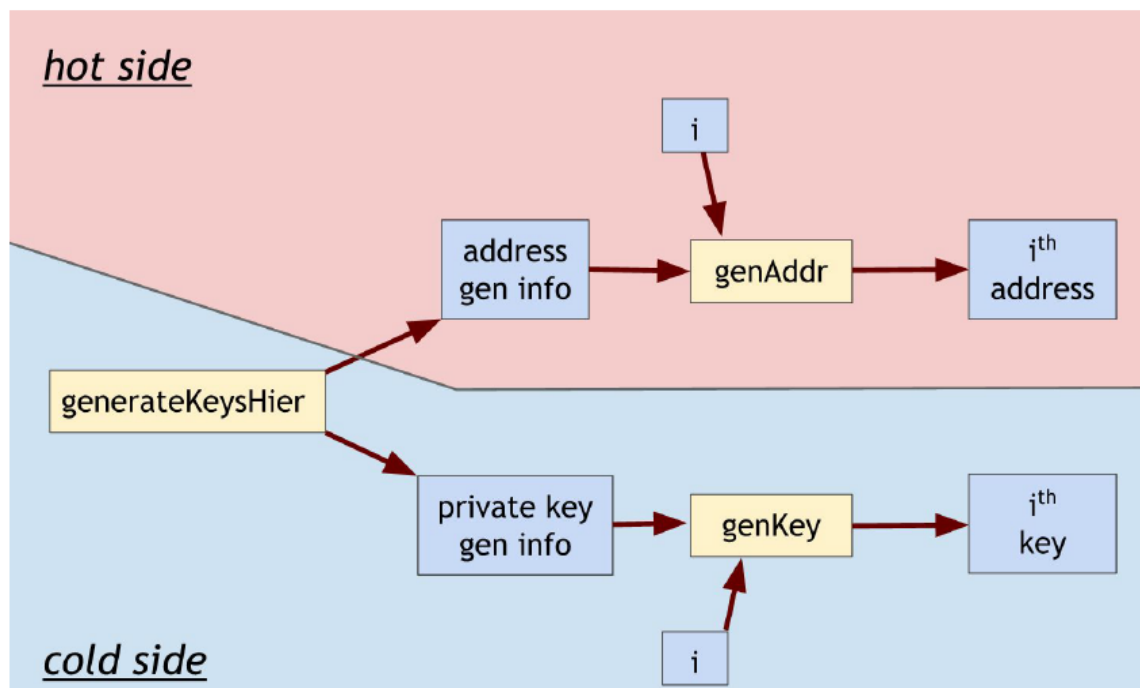
- 让冷存储端制造很多的地址数量，通过一个短暂的一次性的**交换**，热存储端就可知晓所有地址

Crypto technique（同步）

- 对每个*i*而言，第*i*个地址和第*i*个私钥相匹配 → 一长串配对的公私钥

4.2 热存储和冷存储

分层确定性钱包



Schema of a hierarchical wallet



4.2 热存储和冷存储

分层确定性钱包：技术方案

For hierarchical key generation, we'll need two other random values k and y

Private key generation info:	k, x, y
i^{th} private key:	$x_i = y + H(k \parallel i)$
Address generation info:	k, g^y
i^{th} public key:	$g^{x_i} = g^{H(k \parallel i)} \cdot g^y$
i^{th} address:	$H(g^{x_i})$



4.2 热存储和冷存储

分层确定性钱包：安全性分析

g^y can't deduce y , and also x_i (i -th private key)

Because of **DLP** (Discrete Logarithm Problem)



4.2 热存储和冷存储

大脑钱包（**Brain Wallet**）

Sidebar: generating memorable passphrases. One passphrase-generation procedure that gives about 80 bits of entropy is to pick a random sequence of 6 words from among the 10,000 most common English words ($6 \times \log_2(10000)$ is roughly 80). Many people find these easier to memorize than a random string of characters. Here are a couple of passphrases generated this way.

worn till alloy focusing okay reducing
earth dutch fake tired dot occasions

助记词(**mnemonic**)



热存储和冷存储

让攻击者尝试密钥破解的速度变慢

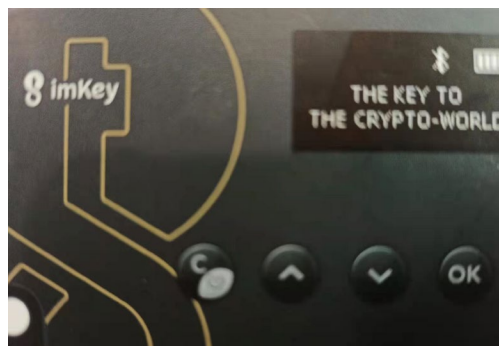
- Key Stretching
- 重复计算SHA256 (2^{20} Trials)

$$SHA256^{\{2^{20}\}}(\text{Password}) == \text{OpenValue}$$

4.2 热存储和冷存储

防损硬件(tamper-resistant device)

- 用来保存密钥或者用来生成密钥;
- 此类设备不会泄漏密钥或者输出密钥;
- 一旦设备丢失或者被盗, 马上能察觉。





4.3 密钥分存和密钥共享

- 把密钥保存在一个地方：一损俱损
- 分散风险： 密钥分存
- 密钥被分成 N 个片段，如果获得其中的 K 个片段就可以把原密钥重新还原。如果片段数目少于 K ，不能知道密钥的任何信息。



4.3 密钥分存和密钥共享

不能直接分割密钥！

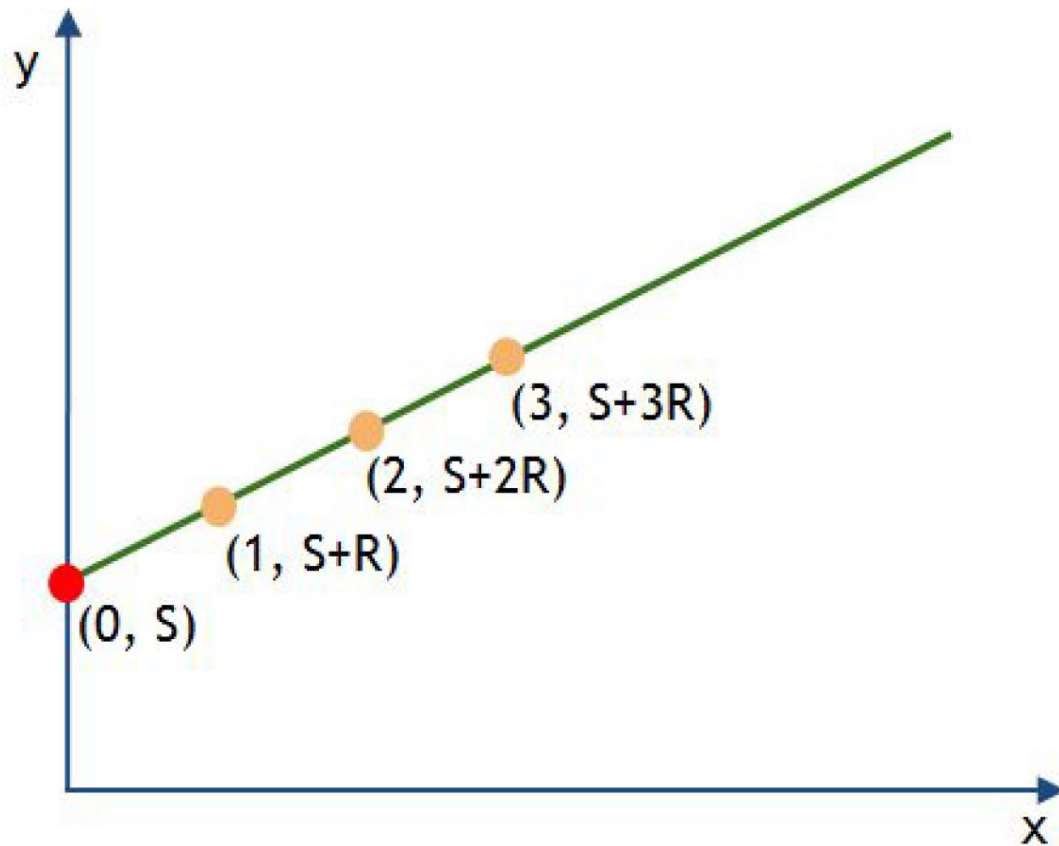
- 否则每一个片段会透露密钥的部分信息，降低了搜索复杂度；从而降低了安全性



4.3 密钥分存和密钥共享

- $N=2, K=2$
- Solutions?
- R and $R \oplus S$
- Extension to $N=K$ case

4.3 密钥分存和密钥共享



Geometric illustration of 2-out-of-N secret sharing



4.3 密钥分存和密钥共享

Equation	Degree	Shape	Random parameters	Number of points (K) needed to recover S
$(S + RX) \bmod P$	1	Line	R	2
$(S + R_1X + R_2X^2) \bmod P$	2	Parabola	R_1, R_2	3
$(S + R_1X + R_2X^2 + R_3X^3) \bmod P$	3	Cubic	R_1, R_2, R_3	4

Lagrange interpolation that allows you to reconstruct a polynomial of degree $K-1$ from any K points on its curve.

Shamir, Adi (1979), "How to share a secret", Communications of the ACM, 22 (11): 612–613.

Shares; Threshold



4.3 密钥分存和密钥共享

门限密码(Threshold Cryptography)

- 两个子密钥分别保存在个人电脑和手机上;
- 电脑生成一个签名片段, 发送到你手机上;
- 手机利用它的子密钥完成整个签名。



4.3 密钥分存和密钥共享

多重签名

- 比如：一个交易需要5个人至少3个人签名才能完成
- 多重签名可以妥善的管理在冷存储端的数字资产；需要多人参与才能实现



4.4 在线钱包和交易所

在线钱包

- 在线钱包可以在各种场合应用，但真正的钱包信息存储在云端
- Blockchain.info
- 网站存储着你的密钥，至少能够接触到你的密钥
- 安全前提：网站服务提供商可以信任



4.4 在线钱包和交易所

数字货币（比特币）交易所

- 银行的功能：面向个人存取款
- 银行会把钱用于投资； 储备金
- 数字货币交易所在交易前后，数字货币并没有真正在区块链中移动，只是你和银行的合约变化了



4.4 在线钱包和交易所

数字货币（比特币）交易所

- 优点：把数字货币（比特币）和法币经济结合，实现自由转换
- 风险：
 1. 挤兑
 2. 庞式骗局
 3. 黑客入侵



4.4 在线钱包和交易所

银行监管

- 政府要求银行有一个**最低准备金**要求：需要**3%-10%**的现金应对突发提款需求
- 政府对银行进行监管，必要时为银行或者储蓄者提供保护



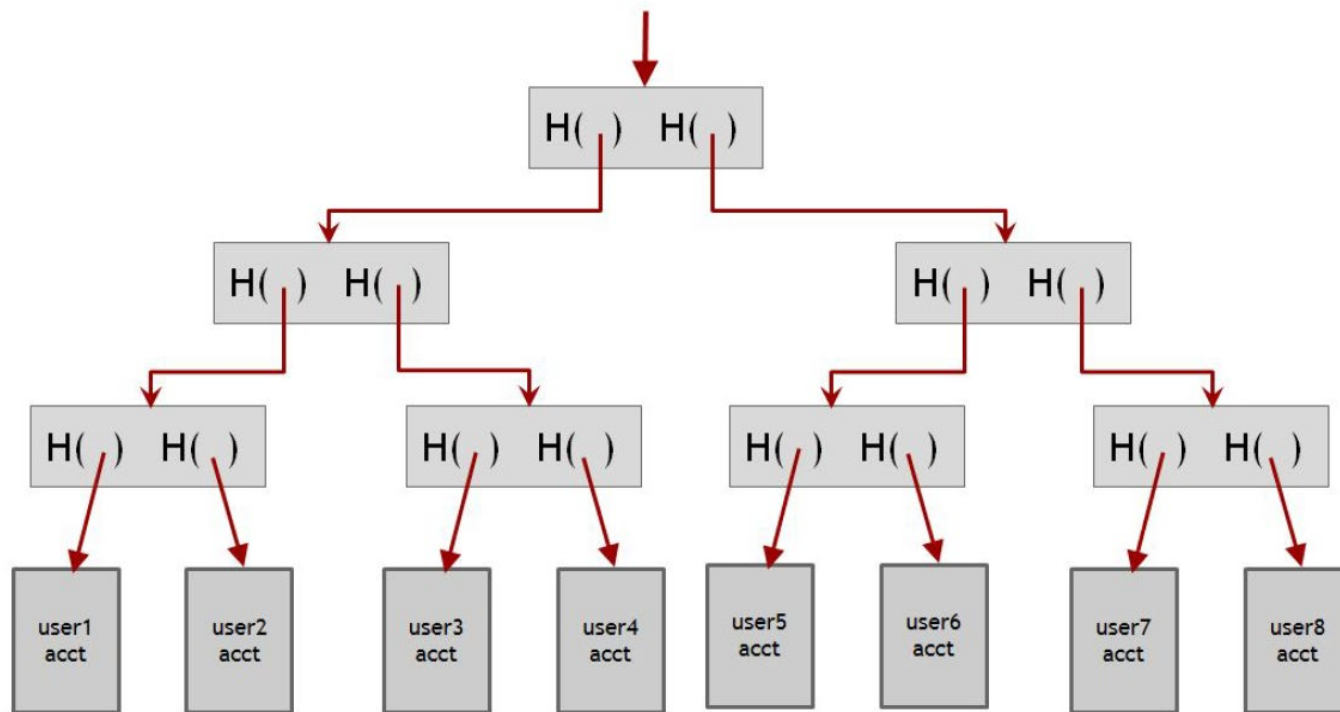
4.4 在线钱包和交易所

准备金证明

- 向储户证明留存了一部分储备金，消除投资人的担心
- 如何证明你(交易所)有 10^6 的数字准备金？
- 发起一笔转账交易，收款方为本人；利用私钥签名

4.4 在线钱包和交易所

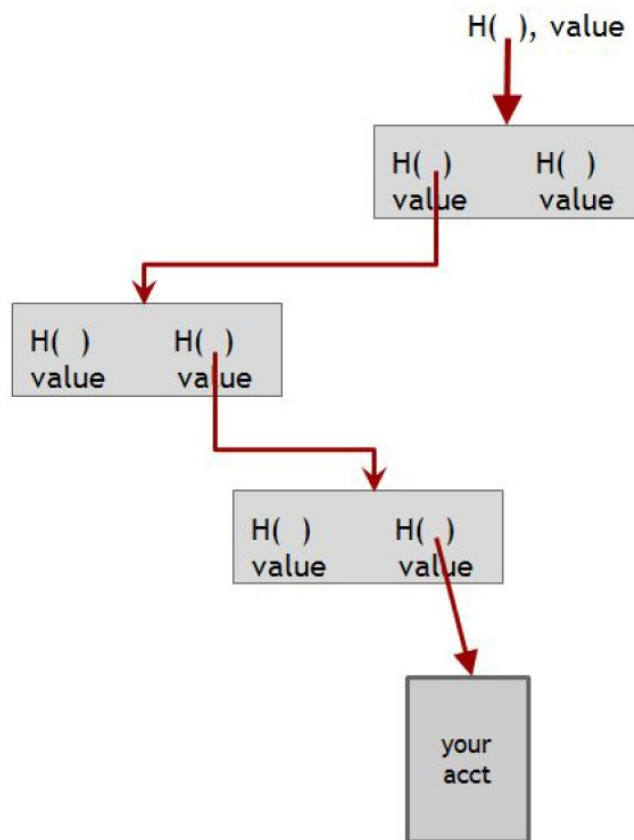
证明存款规模



The exchange publishes the root of a Merkle tree that contains all users at the leaves, including **deposit amounts**. Any user can request a proof of inclusion in the tree, and verify that the **deposit sums are propagated correctly to the root of the tree**.

4.4 在线钱包和交易所

- 每个客户可以向交易所索要存款证明





4.4 在线钱包和交易所

The customer then verifies that:

- 1. The **root** hash pointer and root value are the same as what the exchange signed and published.
- 2. The **hash** pointers are consistent all the way down, that is, *each hash value is indeed the cryptographic hash of the node it points to.*
- 3. The **leaf** contains the correct user account info (say, username/user ID, and deposit amount).
- 4. Each value is the **sum** of the values of the two values beneath it.
- 5. Neither of the values is a negative number.



4.4 在线钱包和交易所

- 交易所证明了其至少留存了X数字货币
- 之后证明其吸收的存款规模至多是Y数字货币
- 可让所有人能独立审计验证：准备金下限： X/Y



4.5 支付服务

如何接受数字货币付款

- 客户可以用数字货币(比如比特币)购物
- 商户如期收到法定货币(比如美元)
- 支付服务商获得手续费


4.5 支付服务


Choose A Way To Accept Bitcoin or [see examples](#) of each payment method.


Type ☒ Button ☐ Hosted Page ☐ iFrame ☐ Email invoice


Payment ☒ Buy now ☐ Donation ☐ Subscription

Button Style

☒ 

☐ 

☐ 

☐ 

Item Name Amount

Item Description

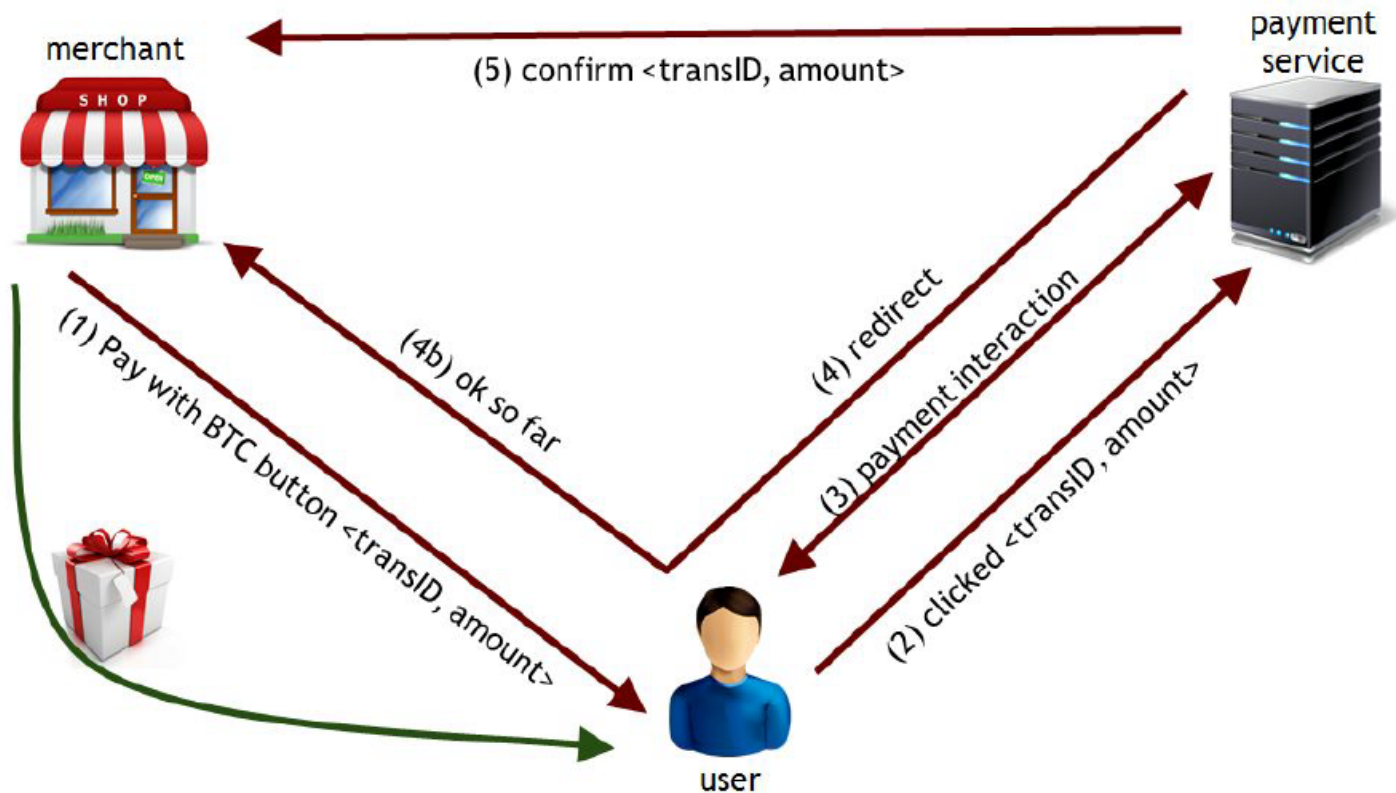
Send Funds To

[Show Advanced Options](#)

[Generate Button Code](#)

Payment service interface for generating a pay-with-Bitcoin button

4.5 支付服务



Payment process involving a user, merchant, and **payment service**



4.5 支付服务

支付服务商承担了所有**风险**

- **安全风险：** 好的安全措施来管理数字货币
- **汇率风险：**（比如 数字货币 与法定货币 之间 兑换的汇率波动）

另外一方面：如果支付商解决了这些问题，可从每笔交易中收取可观的**利润**



4.6 交易费

交易费=交易的输入金额-交易的输出金额

- 比特币网络中，传播你的交易信息需要成本
 - ✓ 比如区块中打进一笔你的交易，*区块就会变大，也会花费更多的时间传输到其他节点*
 - ✓ 确认你的交易需要*花费代价*：交易费用来补偿矿工处理交易付出的代价



4.6 交易费

- 通常而言，如果支付了更多的交易费，那么交易会被更快，更可靠的传播和记录
- 默认的交易费政策
- **No fee is charged** if a transaction meets all of these three conditions:
 - 1. the transaction is less than 1000 bytes in size,
 - 2. all outputs are 0.01 BTC or larger
 - 3. priority is large enough

$(\text{sum of input age} * \text{input value}) / (\text{transaction size})$



4.6 交易费

- The approximate size of a transaction:
148 bytes for each input,
34 bytes for each output
and 10 bytes for other information.

A transaction with *two inputs and two outputs* would be about **400 bytes**.

通常需要支付一笔标准的交易费用；
大多数矿工强制要求必须包含交易费用



4.7 货币兑换市场

- 比特币市场中的公允价格是由供给和需求决定的
- 比特币的需求包括将比特币作为支付中介，以及作为投资需求



4.7 货币兑换市场

- 根据供需平衡来决定比特币价格：假设 P 是比特币对美元的价格
- 供应侧：D秒内，市场内有S个比特币可以进行交易
- 需求侧：总共的支付交易规模是T美元

$$\frac{S}{D} = \frac{T}{P}$$

意味着 $P \propto T$