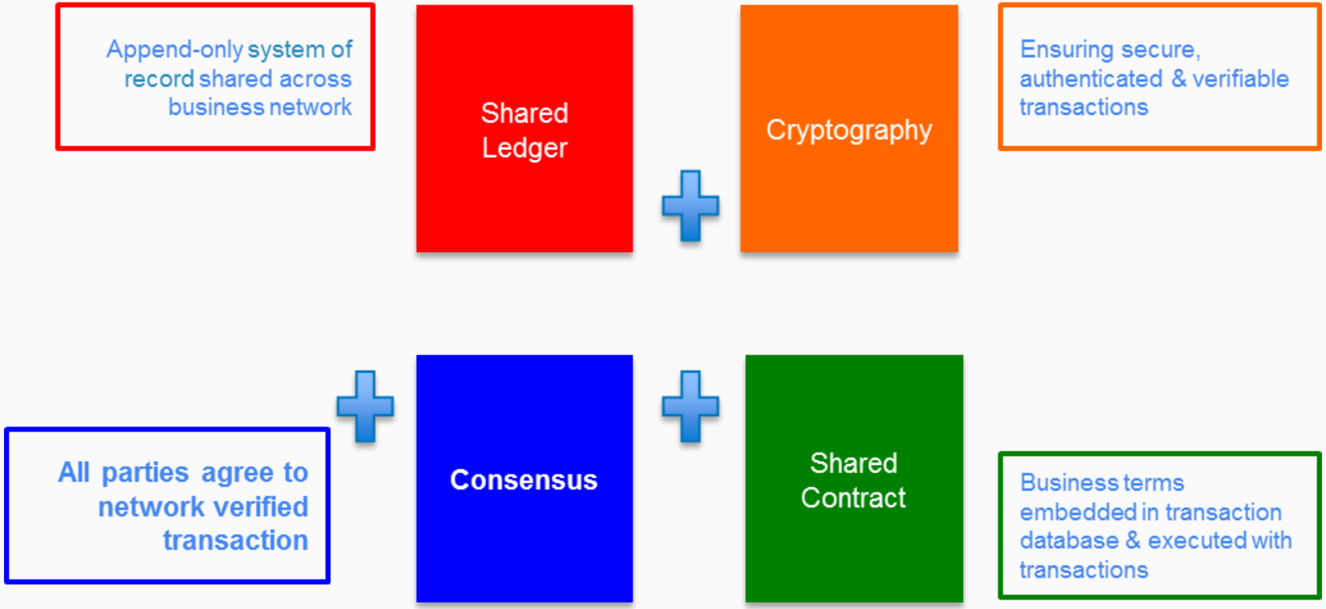
区块链：
技术基础、及未来方向

赵运磊
复旦大学
ylzhao@fudan.edu.cn

Blockchain in a nutshell



Append-only system of record shared across business network

Shared Ledger

Cryptography

Ensuring secure, authenticated & verifiable transactions

All parties agree to network verified transaction

Consensus

Shared Contract

Business terms embedded in transaction database & executed with transactions

Cryptocurrencies

Decentralized, compared to classic digital currencies

- Avoided a single point of attack

First realized by Bitcoin(BTC), in 2008

Presented a public transaction ledger, called *blockchain*

- Decentralized autonomous organizations
- Smart contracts
- Smart properties

Bitcoin Usage

Alice is to send 1BTC to Bob

Bob: my BTC address is Y.

Alice: I own address X, send 1BTC to Y, with an extra 0.2mBTC as the fee.

<Alice signs with private key>

<A Miner writes transaction to blockchain>

Bob receives BTC; miner receives the fee.

Alice is to send one gold bar to Bob

Bob: my vault is Y.

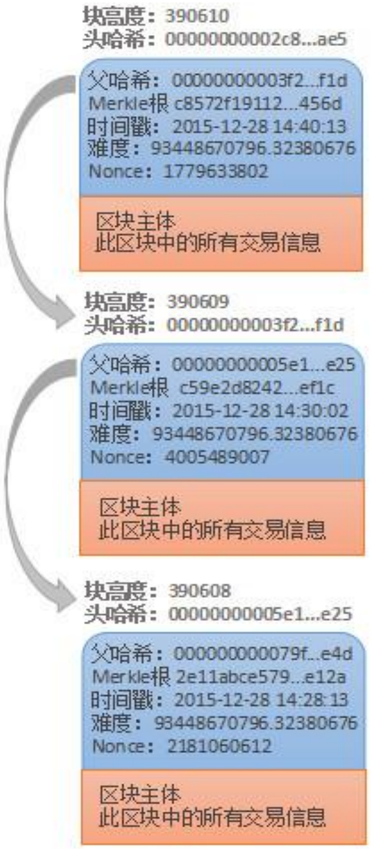
Alice: I own vault X, send 1 bar to Y, with an extra 1gram as the fee.

<Alice shows vault deed and signs>

<A worker moves gold>

Bob receives the bar; the worker receives the fee.

区块链基本结构结构



块高度：390610
头哈希：00000000002c8...ae5

父哈希：00000000003f2...f1d
Merkle根 c8572f19112...456d
时间戳：2015-12-28 14:40:13
难度：93448670796.32380676
Nonce：1779633802

区块主体
此区块中的所有交易信息

块高度：390609
头哈希：00000000003f2...f1d

父哈希：00000000005e1...e25
Merkle根 c59e2d8242...ef1c
时间戳：2015-12-28 14:30:02
难度：93448670796.32380676
Nonce：4005489007

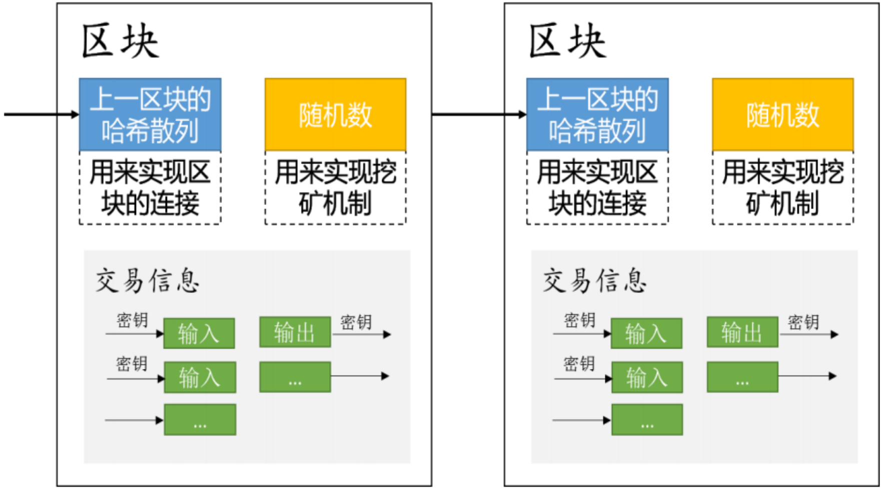
区块主体
此区块中的所有交易信息

块高度：390608
头哈希：00000000005e1...e25

父哈希：00000000079f...e4d
Merkle根 2e11abce579...e12a
时间戳：2015-12-28 14:28:13
难度：93448670796.32380676
Nonce：2181060612

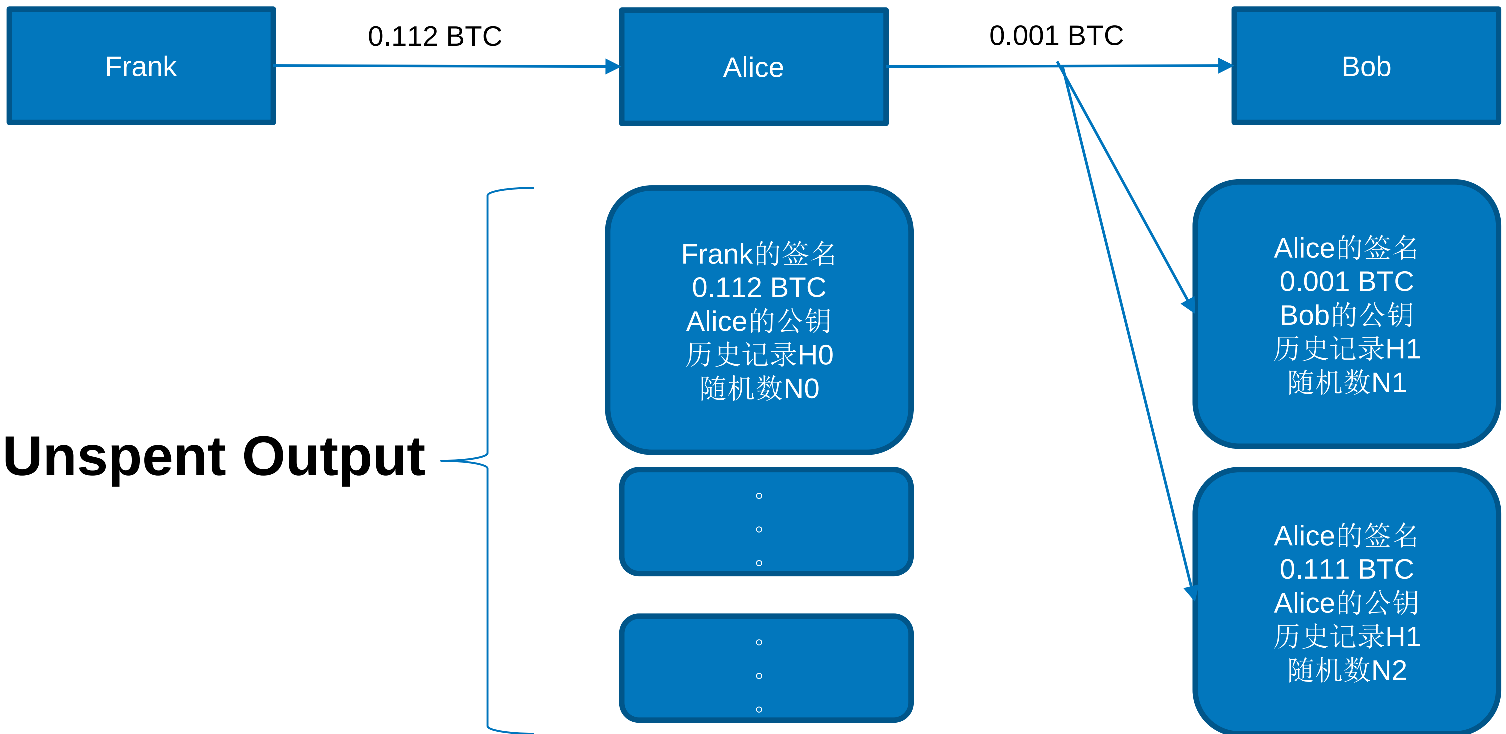
区块主体
此区块中的所有交易信息

区块头由三组区块元数据组成。首先是一组引用父区块哈希值的数据，这组元数据用于将该区块与区块链中前一区块相连接。第二组元数据，即难度、时间戳和nonce，与挖矿竞争相关 。第三组元数据是merkle树根



区块是一种记录交易的数据结构。每个区块由区块头和区块主体组成，区块主体只负责记录前一段时间内的所有交易信息，区块链的大部分功能都由区块头实现。

Transaction Structure in BlockChain



Bitcoin Transaction is like a cheque

Bitcoin Bank Date: 14-03-2016

From: Mining reward

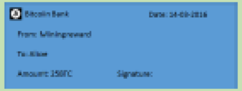
To: Alice

Amount: 25BTC Signature:

Alice: 25 BTC

Bitcoin Bank Date: 15-03-2016

From:



Bitcoin Bank Date: 14-03-2016

From: Mining reward

To: Alice

Amount: 25BTC Signature:

To: Bob

Amount: 25BTC Signature: Alice

Alice: 0 BTC
Bob: 25 BTC

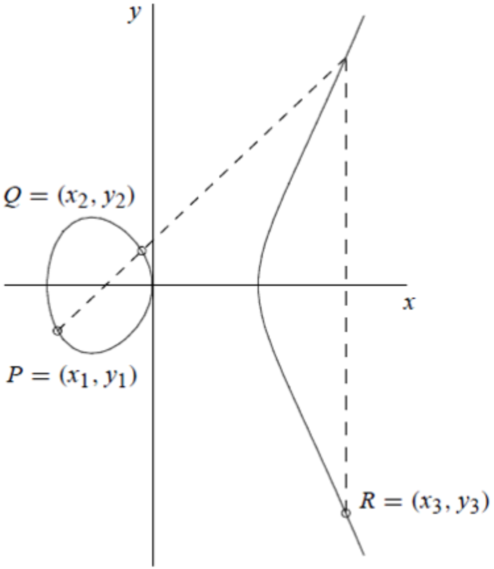
Pseudo Anonymous

Using public key cryptography

Transactions are sent to public key “addresses”

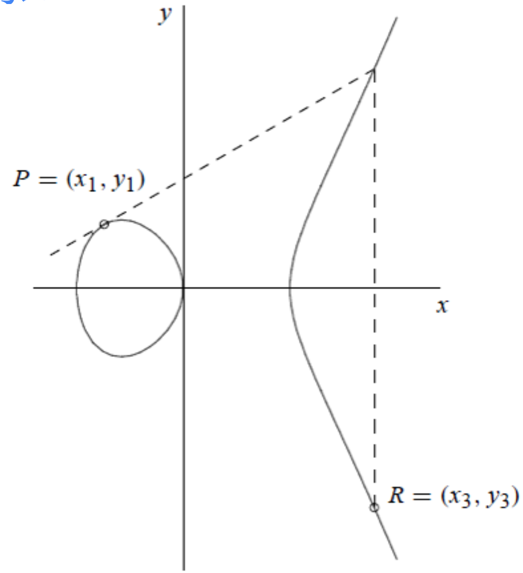
公钥及钱包地址（I）

这个加法的几何定义如下面两个图，两个点的加法结果是指这两点的连线和曲线的交点关于x轴的镜像。



(a) Addition: $P+Q=R$.

图1：两个不同的点相加



(b) Doubling: $P+P=R$.

图二：两个相同的点相加

公钥及钱包地址（II）

比特币使用了secp256k1标准所定义的一条特殊的椭圆曲线和一系列数学常数。该标准由美国国家标准与技术研究院（NIST）设立。secp256k1曲线由下述函数定义，该函数可产生一条椭圆曲线：

$y^2 = (x^3 + 7)\}$ over (Fp)

公钥及钱包地址（III）

Hash函数：HASH函数又称散列函数或杂凑函数，它是一种单向加密函数算法。

1、H(m)算法公开，不需要密钥；

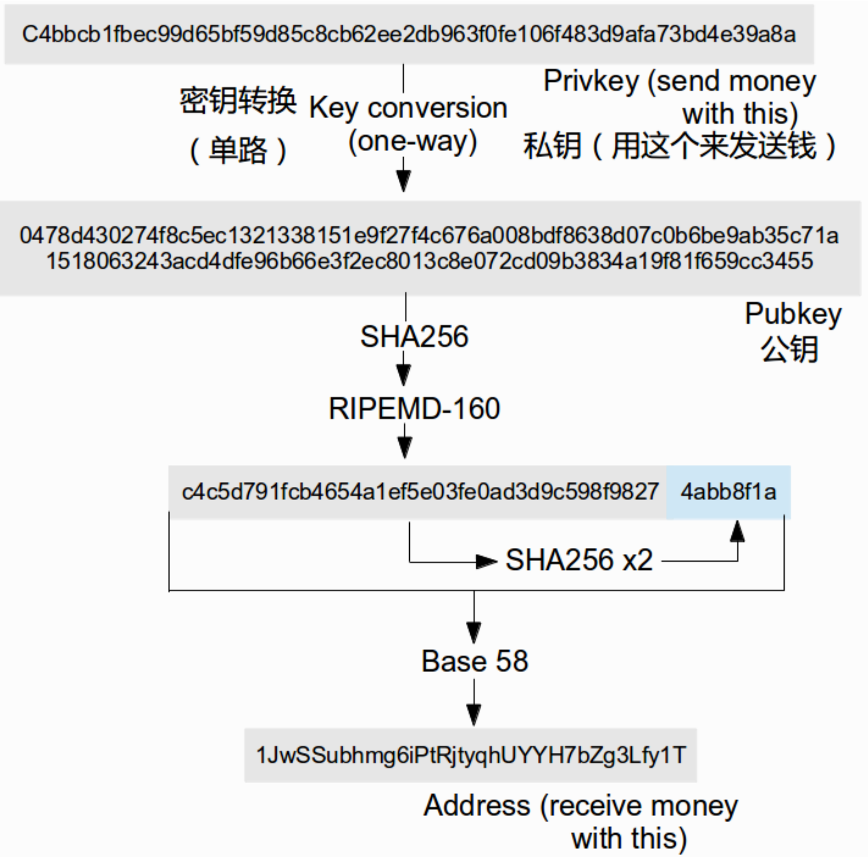
2、将任意长度输入数据转换成一个固定长度输出；

3、对任意m，H(m)易于计算，这也是使用普遍的原因之一。

4、Hash函数具有抗碰撞性

公钥及钱包地址（IV）

然而，比特币的地址不是公钥，而是公钥的哈希值，私钥和地址的关系如图



C4bbcb1fbec99d65bf59d85c8cb62ee2db963f0fe106f483d9afa73bd4e39a8a

Privkey (send money with this)
私钥（用这个来发送钱）

密钥转换（单路） Key conversion (one-way)

0478d430274f8c5ec1321338151e9f27f4c676a008bdf8638d07c0b6be9ab35c71a
1518063243acd4dfe96b66e3f2ec8013c8e072cd09b3834a19f81f659cc3455

Pubkey
公钥

SHA256

RIPEMD-160

c4c5d791fcb4654a1ef5e03fe0ad3d9c598f9827 4abb8f1a

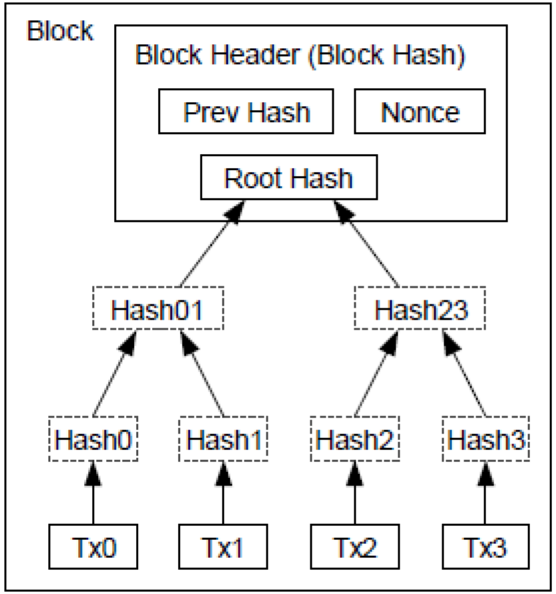
SHA256 x2

Base 58

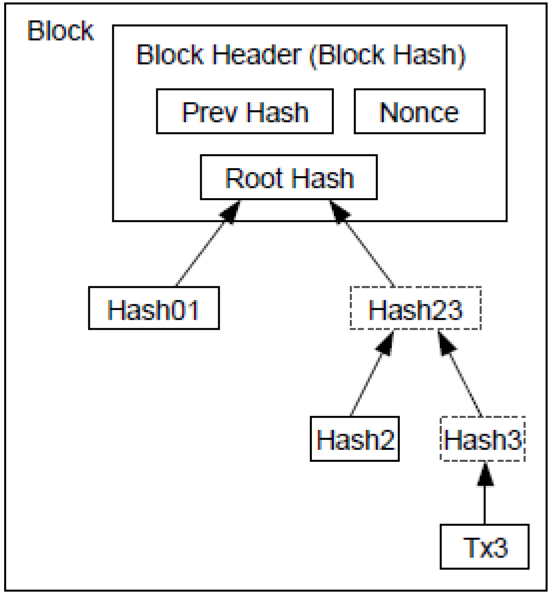
1JwSSubhmg6iPtRjtyqhUYYH7bZg3Lfy1T

Address (receive money with this)

Merkle Tree (I)

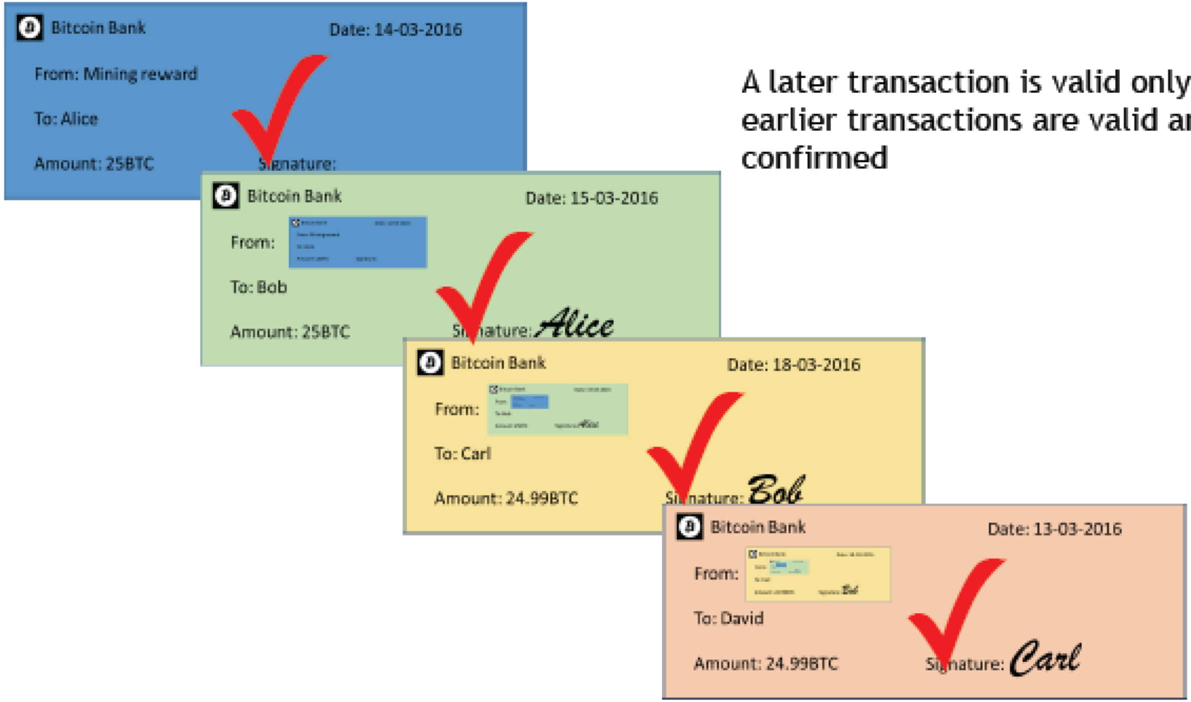


Transactions Hashed in a Merkle Tree



After Pruning Tx0-2 from the Block

A chain of transactions



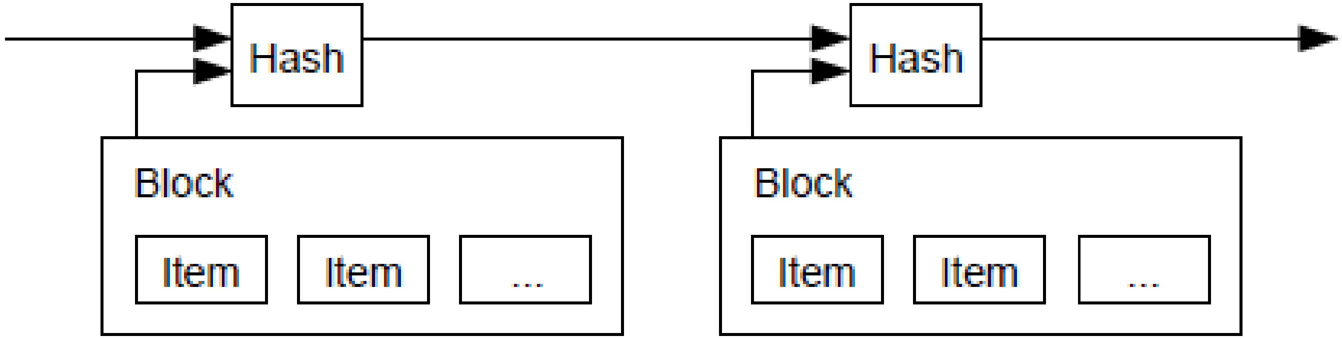
Bitcoin Bank
Date: 14-03-2016
From: Mining reward
To: Alice
Amount: 25BTC
Signature:

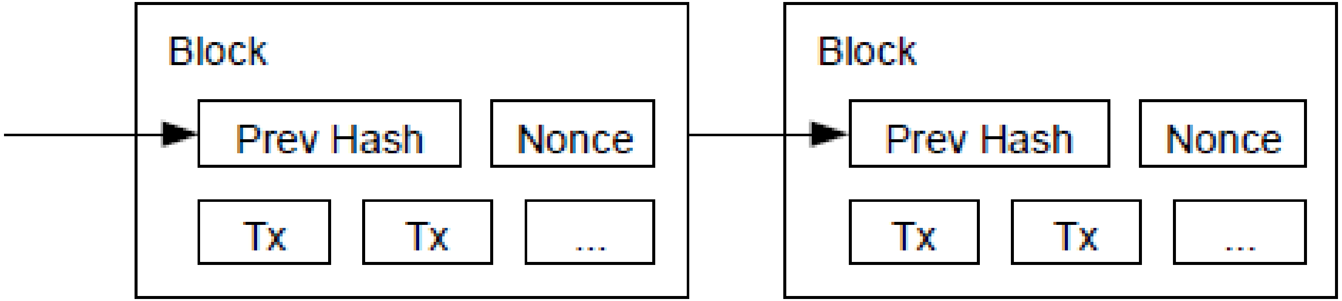
Bitcoin Bank
Date: 15-03-2016
From:
To: Bob
Amount: 25BTC
Signature: Alice

Bitcoin Bank
Date: 18-03-2016
From:
To: Carl
Amount: 24.99BTC
Signature: Bob

Bitcoin Bank
Date: 13-03-2016
From:
To: David
Amount: 24.99BTC
Signature: Carl

A later transaction is valid only if the earlier transactions are valid and confirmed


Hash
Hash
Block
Item
Item
...
Block
Item
Item
...


Block
Prev Hash
Nonce
Tx
Tx
...
Block
Prev Hash
Nonce
Tx
Tx
...

Bitcoin Consensus：POW Mining

Assume an optimal hash function *H(x)*

Miner search for nonce *n*, so that *H(data, n)* passes some difficulty check

- Data contains information about the previous block, and (very likely) some transactions.
- A typical difficulty check: *H(data, n)* < *threshold*

Miner broadcast this solution, everyone verifies independently

If valid and is tail of the longest chain, treats this block as latest and continue working

A block (h_{-1}, η, m, h)

- h_{-1}: a pointer to the previous record
- η: a nonce
- m: a message (a transaction)
- h: a pointer to the current block

Hardness parameter p

- $D_p = p \cdot 2^\kappa$
- $\forall (h, b)$, $\Pr[H(h, \eta, b) < D_p] = p$

Validness of block $b = (h_{-1}, \eta, m, h)$ w.r.t. block $b_{-1} = (h'_{-1}, \eta', m', h')$

- $h_{-1} = h'$
- $\mathrm{H.ver}\big((h_{-1}, \eta, m), h\big) = 1$
- $h < D_p = p \cdot 2^\kappa$

Validness of blockchain $(b_0, \ldots, b_l)$

- $b_0 = \big(0, 0, \perp, H(0, 0, \perp)\big)$ is the genesis block
- $\forall 1 \leq i \leq l$, b_i is valid w.r.t. b_{i-1}

Bitcoin Blockchain Control

21,000,000 BTC in total

Every 2016 blocks, adjust threshold

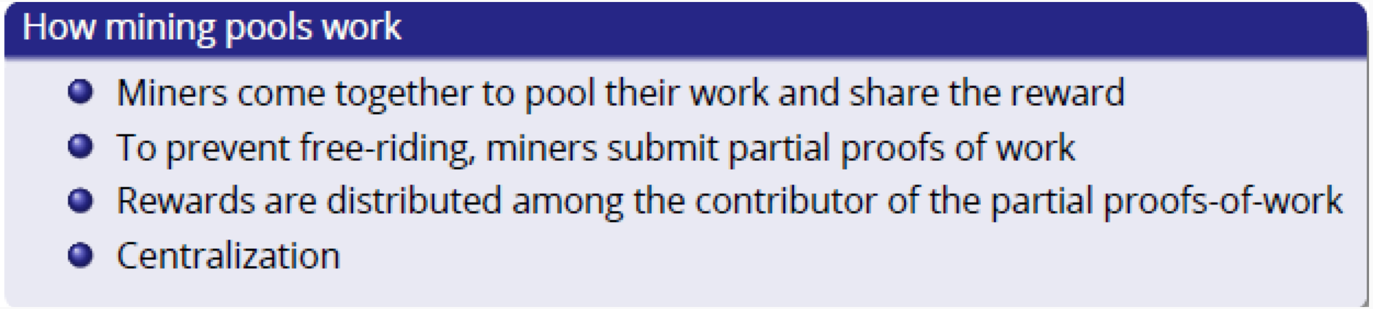
- Took less than two weeks, increase difficulty by decreasing threshold
- Took more than two weeks, reduce difficulty by increasing threshold
- So that a new block is generated every 10 minutes on average

Reward block creators with newly minted coins

- starting at 50BTC/block and halves every 210000 blocks (4 years). Will eventually stop

Mining Pool

It takes a very long time for an individual miner to mine a block. The payments received by miners has a very high variance.



How mining pools work

- Miners come together to pool their work and share the reward
- To prevent free-riding, miners submit partial proofs of work
- Rewards are distributed among the contributor of the partial proofs-of-work
- Centralization

Protocol: STRATUM (no pool authentication!)

Double spending attack

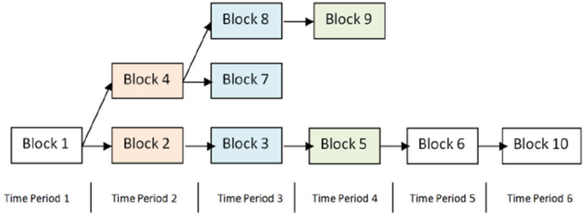
1. Attacker secretly mines a fork, with transaction T1 included.
   a. Mine it longer than the main chain
2. Attacker broadcast transaction T2, conflicting with T1.
   a. Miners include T2 in the chain
   b. Payee confirms payment and then delivers goods
3. Attacker broadcast the secret fork, confirmed by miners
   a. The previous main chain, including T2, gets discarded

Prevent double spending

Wait for a few blocks, for the chance of successful attack to lower

e.g., Below 0.1%

Regarded as confirmations

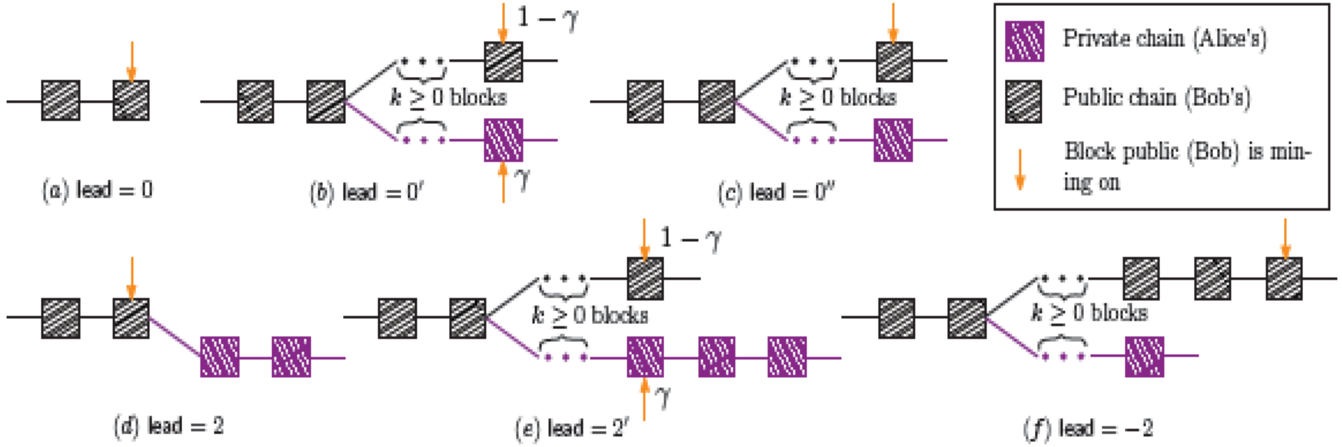


6 confirms => 10% compute power to successfully attack.

24 confirms => 30%

If attacker gets more than 50% compute power, this solution fails

Selfish Mining



Earn extra 63% profits, compared to honest mining!

Meaning 33% portion of computation power can obtain >50% profits!

共识机制

Motivation

POW缺点：

1.浪费电力和计算资源；

2.交易延迟、吞吐量瓶颈；

3.自私挖矿...

技术途径：

1.POS；

2.科学计算；

3.POW与拜占庭共识结合

POS:
consensus without expensive computation

In proof of stake algorithms, inequality is modified to depend on the user's ownership of the particular PoS protocol cryptocurrency.

Consider a user with address *A* and balance bal(*A*). A commonly used proof of stake algorithm uses a condition

$$\text{hash}(\text{hash}(B_{prev}), A, t) \leqslant \text{bal}(A)M/D;$$

B_{prev} denotes the block the user is building on

t is the current UTC timestamp.

Peercoin Implement

Definition of Coin age

Coin age of an unspent transaction output is its value multiplied by the time period after it was created.

A transaction spending a previously unspent output consumes, or destroys.

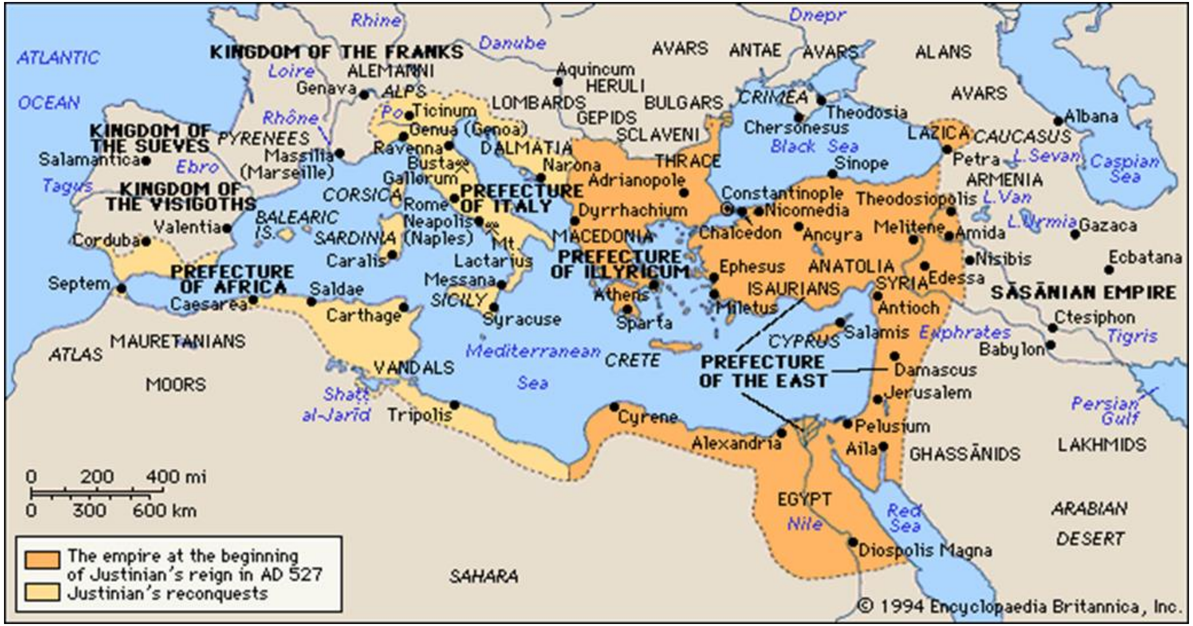
The possibility to create a block

$$\text{hash}(\text{hash}(B_{prev}),U,t) \leqslant \text{bal}(U)\ \text{age}(U)M/D$$

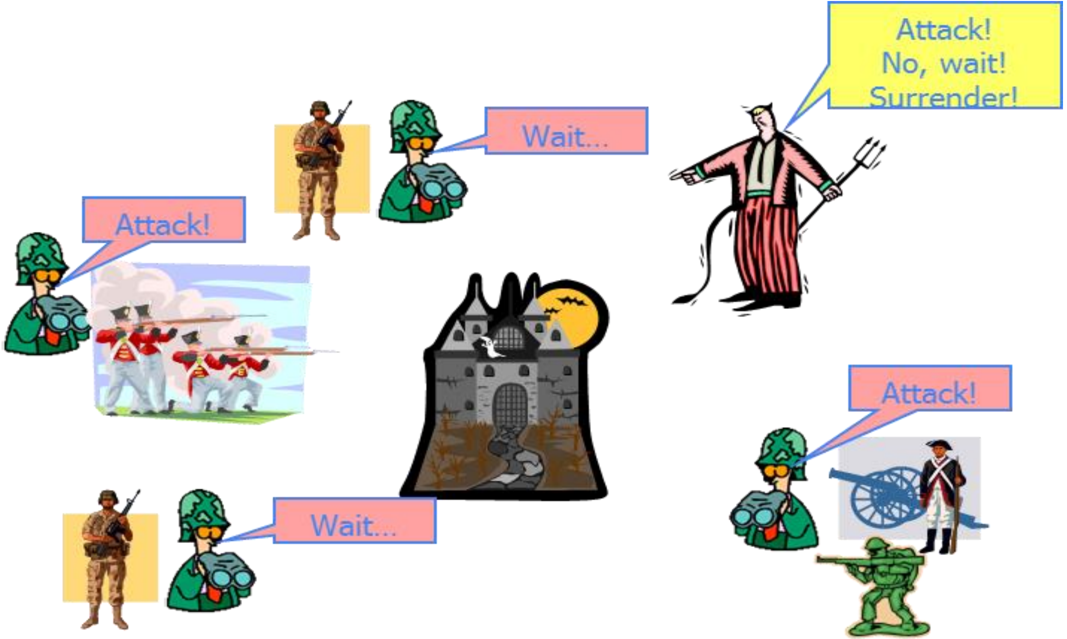
U is an unspent output

Both of coin number and coin age influence the possibility

Consensus basis (Byzantine Protocol)



© 1994 Encyclopaedia Britannica, Inc.



Termination:

Agreement

Integrity

Four Dimensions of Failure Models

Reliable vs. unreliable network

Synchronous vs. asynchronous communication

Byzantine vs. fail-stop

Fail-stop: faulty nodes stop and do not send.

Byzantine: faulty nodes may send arbitrary messages.

Authenticated vs. unauthenticated

Classic Results of Consensus

Easy: synchronous + reliable; synchronous + fail-stop;

Lamport’s 1982:

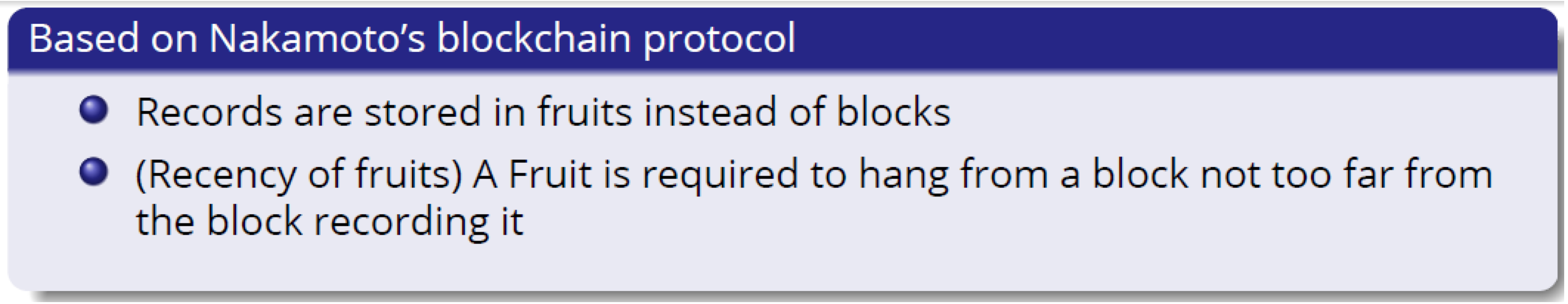
consensus is impossible, with byzantine failures + N $\leqslant$3F + synchronous;

consensus can be reached if N > 3F;

Fischer-Lynch-Patterson 1985:

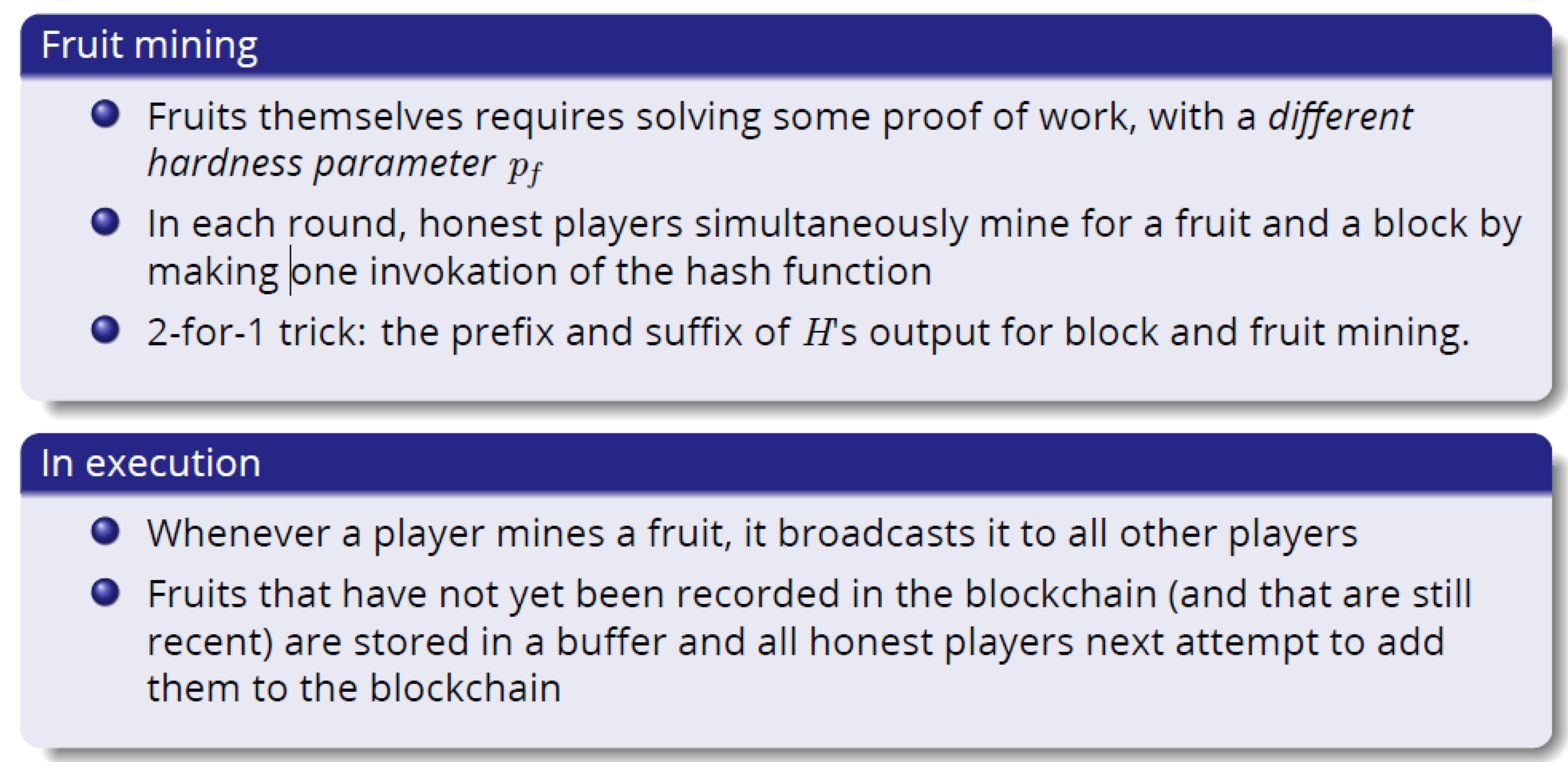
No consensus can be guaranteed in an asynchronous communication system in the presence of any failures.

Fruit Chain: Basic Idea



Based on Nakamoto’s blockchain protocol

- Records are stored in fruits instead of blocks
- (Recency of fruits) A Fruit is required to hang from a block not too far from the block recording it

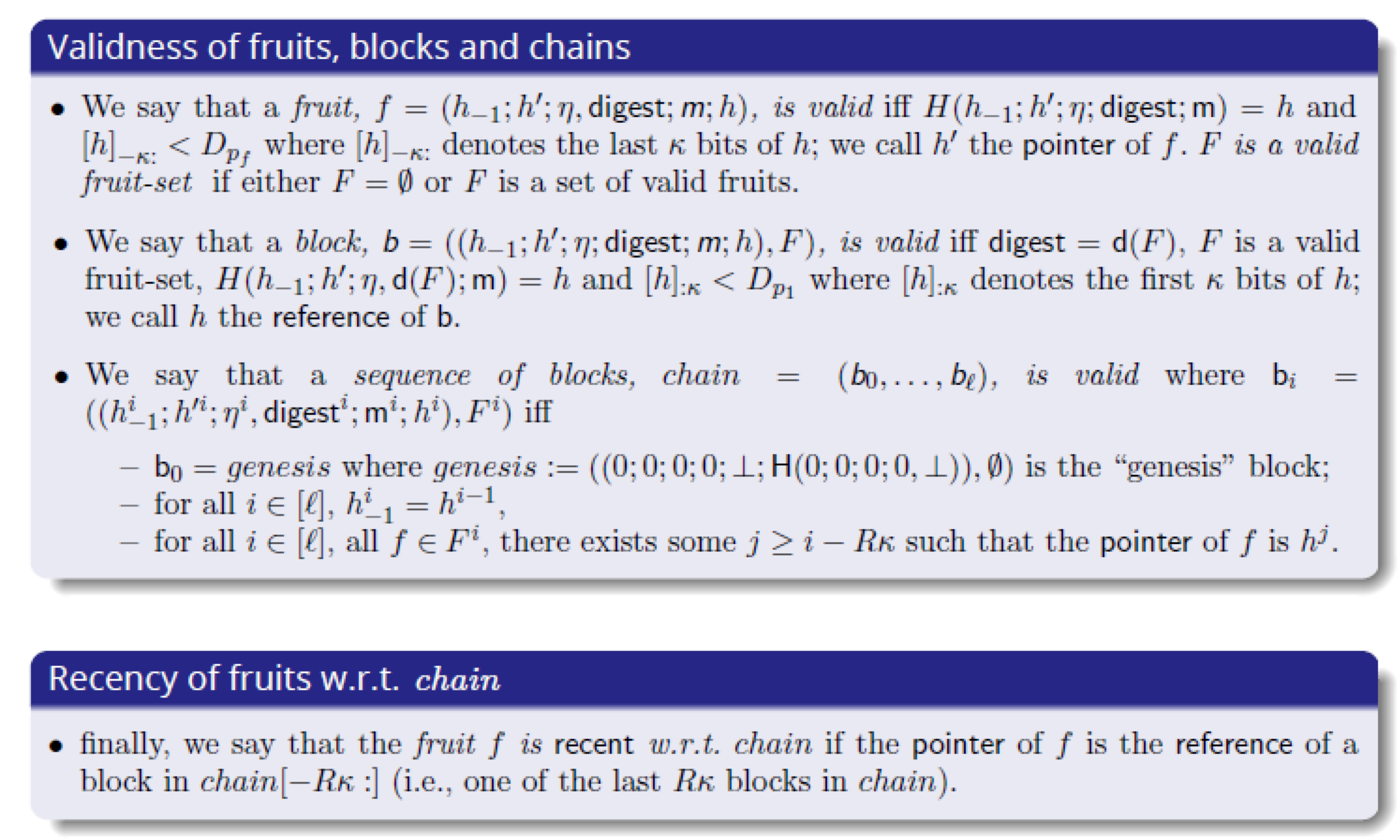


Fruit mining

- Fruits themselves requires solving some proof of work, with a *different hardness parameter* p_f
- In each round, honest players simultaneously mine for a fruit and a block by making one invokation of the hash function
- 2-for-1 trick: the prefix and suffix of H's output for block and fruit mining.

In execution

- Whenever a player mines a fruit, it broadcasts it to all other players
- Fruits that have not yet been recorded in the blockchain (and that are still recent) are stored in a buffer and all honest players next attempt to add them to the blockchain



Validness of fruits, blocks and chains

- We say that a *fruit*, $f = (h_{-1}; h'; \eta, \mathsf{digest}; m; h)$, *is valid* iff $H(h_{-1}; h'; \eta; \mathsf{digest}; \mathsf{m}) = h$ and $[h]_{-\kappa:} < D_{p_f}$ where $[h]_{-\kappa:}$ denotes the last κ bits of h; we call h' the pointer of f. F *is a valid fruit-set* if either $F = \emptyset$ or F is a set of valid fruits.

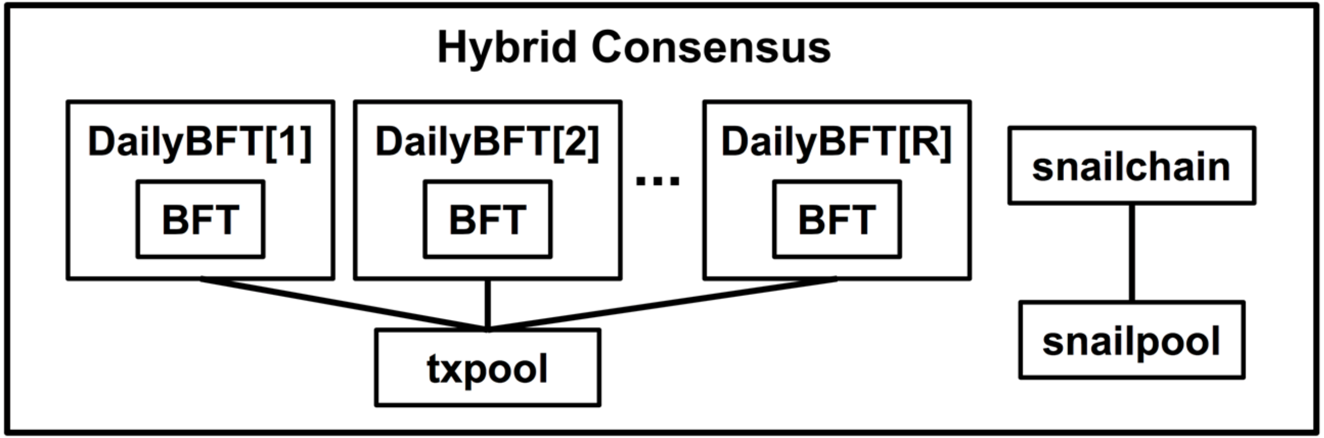
- We say that a *block*, $\mathsf{b} = ((h_{-1}; h'; \eta; \mathsf{digest}; m; h), F)$, *is valid* iff $\mathsf{digest} = \mathsf{d}(F)$, F is a valid fruit-set, $H(h_{-1}; h'; \eta, \mathsf{d}(F); \mathsf{m}) = h$ and $[h]_{:\kappa} < D_{p_1}$ where $[h]_{:\kappa}$ denotes the first κ bits of h; we call h the reference of b.

- We say that a *sequence of blocks, chain* $= (\mathsf{b}_0, \ldots, \mathsf{b}_\ell)$, *is valid* where $\mathsf{b}_i = ((h^i_{-1}; h'^i; \eta^i, \mathsf{digest}^i; \mathsf{m}^i; h^i), F^i)$ iff
  - $\mathsf{b}_0 = genesis$ where $genesis := ((0; 0; 0; 0; \bot; \mathsf{H}(0; 0; 0; 0, \bot)), \emptyset)$ is the "genesis" block;
  - for all $i \in [\ell]$, $h^i_{-1} = h^{i-1}$,
  - for all $i \in [\ell]$, all $f \in F^i$, there exists some $j \geq i - R\kappa$ such that the pointer of f is h^j.

Recency of fruits w.r.t. *chain*

- finally, we say that the *fruit f is* recent *w.r.t. chain* if the pointer of f is the reference of a block in $chain[-R\kappa :]$ (i.e., one of the last $R\kappa$ blocks in $chain$).

Hybrid Consensus



Use a slow blockchain protocol to bootstrap fast permissioned byzantine consensus

Hybrid Consensus: Basic Idea

Leverage snailchain to elect a static committee

- Honest nodes run the blockchain for $csize + \lambda$ blocks:

  $csize = \Theta(\lambda)$ denotes the targeted committee size

- Committee members sign any transaction committed as well as its sequence number

- Any node that was not elected as a committee member can simply count $\left\lceil \frac{1}{3} |csize| \right\rceil$number of signatures from committee members for deciding its own output log

Hybrid Consensus: Dynamic Committee

static committee fails to be secure against an adaptive adversary

The adversary can simply corrupt the committee once it is elected

Rotate committees

When an honest node’s chain reaches $R \cdot csize + \lambda$ in length, the Rth committee is elected by first removing the trailing λ number of blocks, and then from this pruned chain, we elect the last $csize$ blocks’ miners as the committee

Solidus: Basic Idea（I）

- Assume:无altruistic的结点，都是rational的
- 在一个动态变化的participants集上运行Byzantine协议

1. the current set is called a committee.
2. a new member is elected onto the committee and the oldest committee member leaves.
3. We denote the i-th member in the chronological order as Mi, and the committee size is n.
4. Then the i-th committee Ci = (Mi;Mi+1;Mi+2; ……;Mi+n-1)

Solidus: Basic Idea (II)

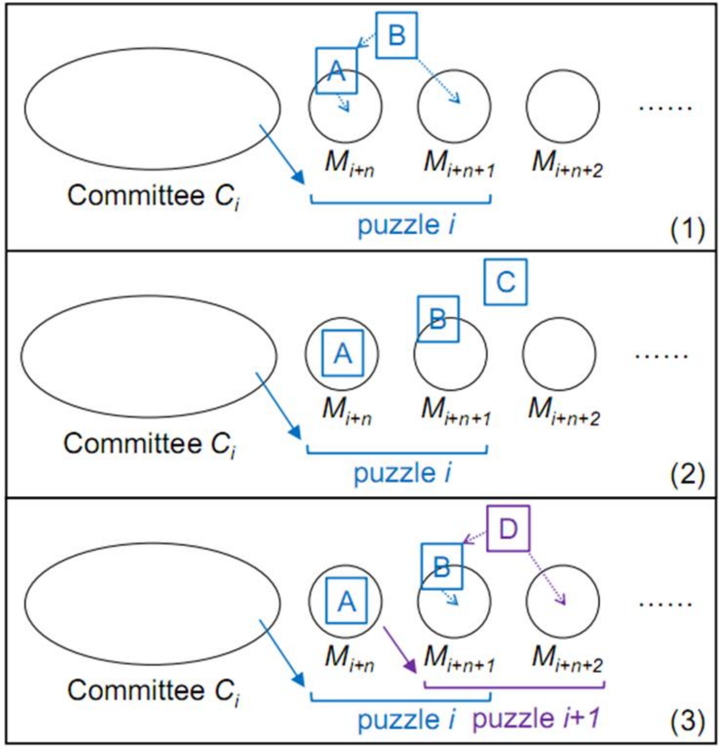
- M向committee提供了Pow（IV-B）

  M itself be elected as the next committee member , and a set of transactions be committed.

- One single Leader is elected, but 3 blocks are added into the chain

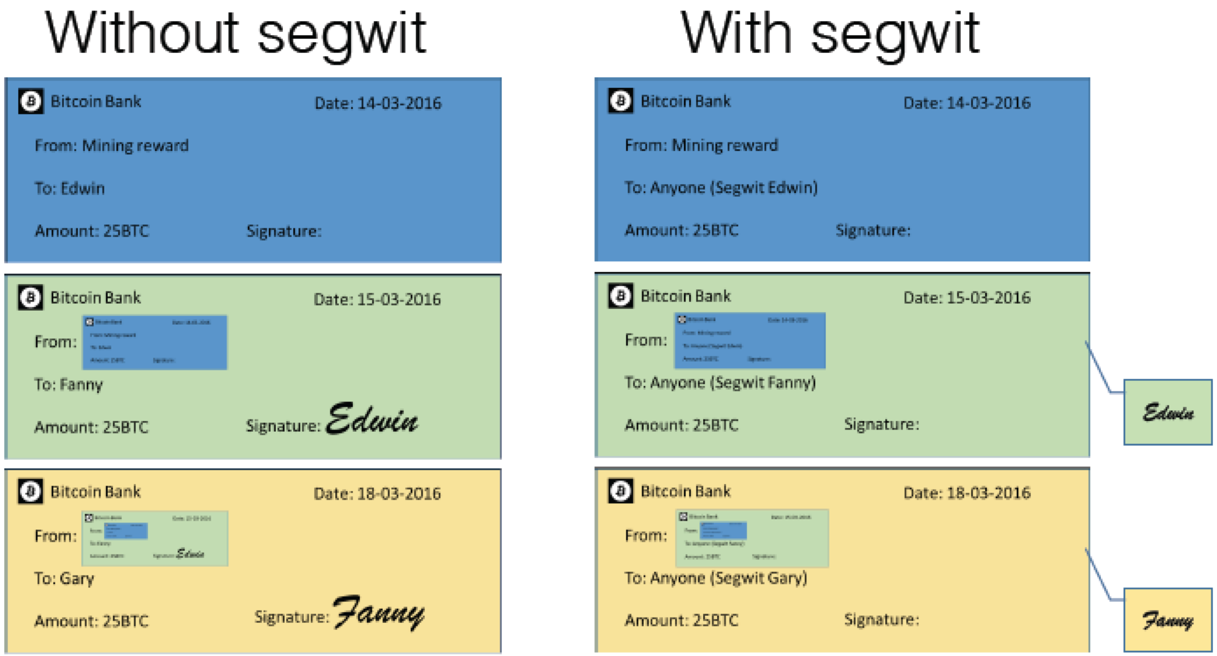
  - Bitcoin仅记录冠军，而Solidus记录前3甲

- M的decision会变成一个puzzle，激励传播就是解决puzzle（IV）



比特币区块链扩容

Segregated Witness (隔离认证）



Without segwit

Bitcoin Bank Date: 14-03-2016
From: Mining reward
To: Edwin
Amount: 25BTC Signature:

Bitcoin Bank Date: 15-03-2016
From:
To: Fanny
Amount: 25BTC Signature: Edwin

Bitcoin Bank Date: 18-03-2016
From:
To: Gary
Amount: 25BTC Signature: Fanny

With segwit

Bitcoin Bank Date: 14-03-2016
From: Mining reward
To: Anyone (Segwit Edwin)
Amount: 25BTC Signature:

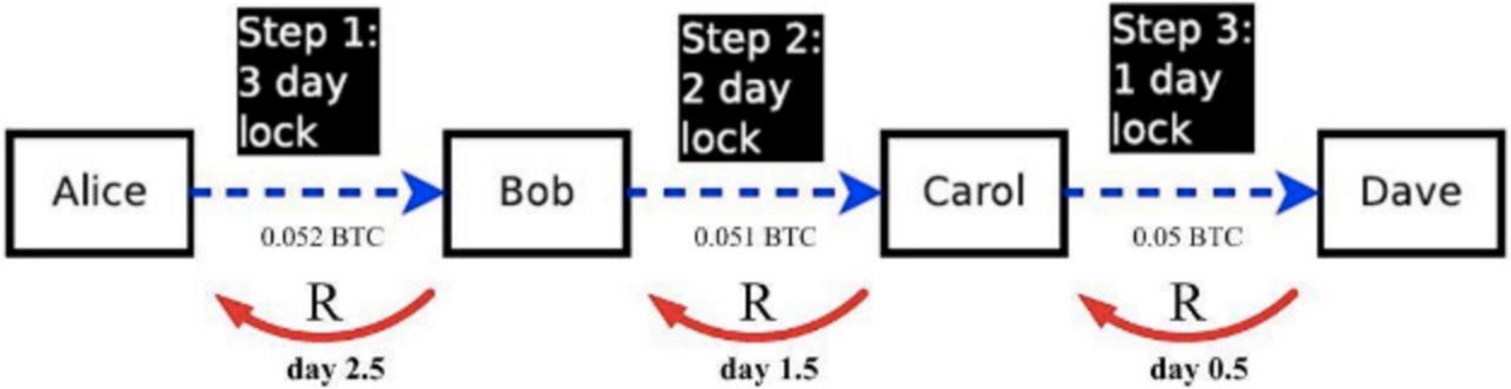
Bitcoin Bank Date: 15-03-2016
From:
To: Anyone (Segwit Fanny)
Amount: 25BTC Signature:

Edwin

Bitcoin Bank Date: 18-03-2016
From:
To: Anyone (Segwit Gary)
Amount: 25BTC Signature:

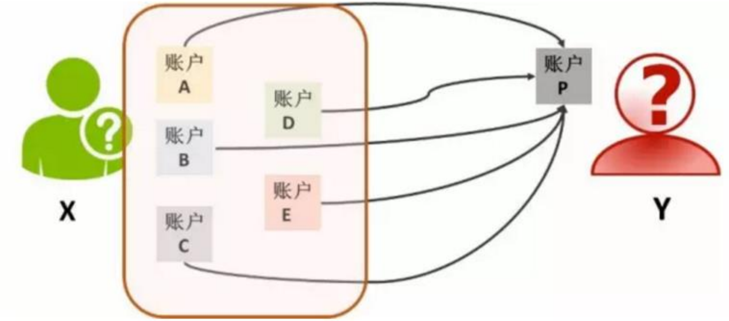
Fanny

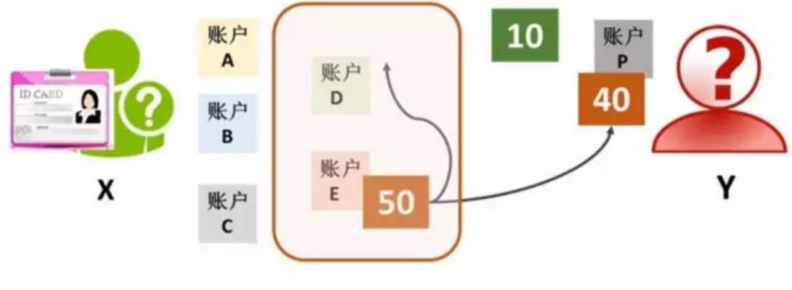
Lightning Network (闪电网络）

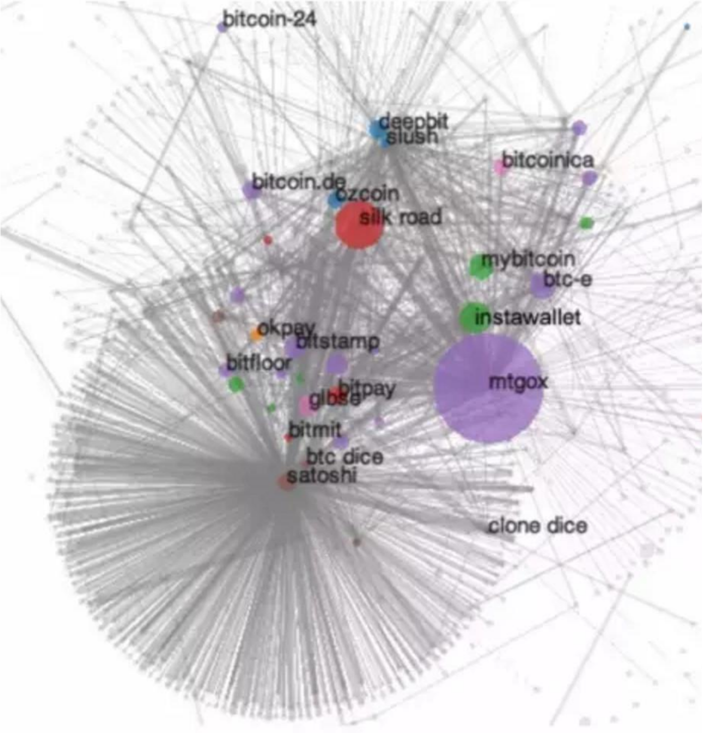


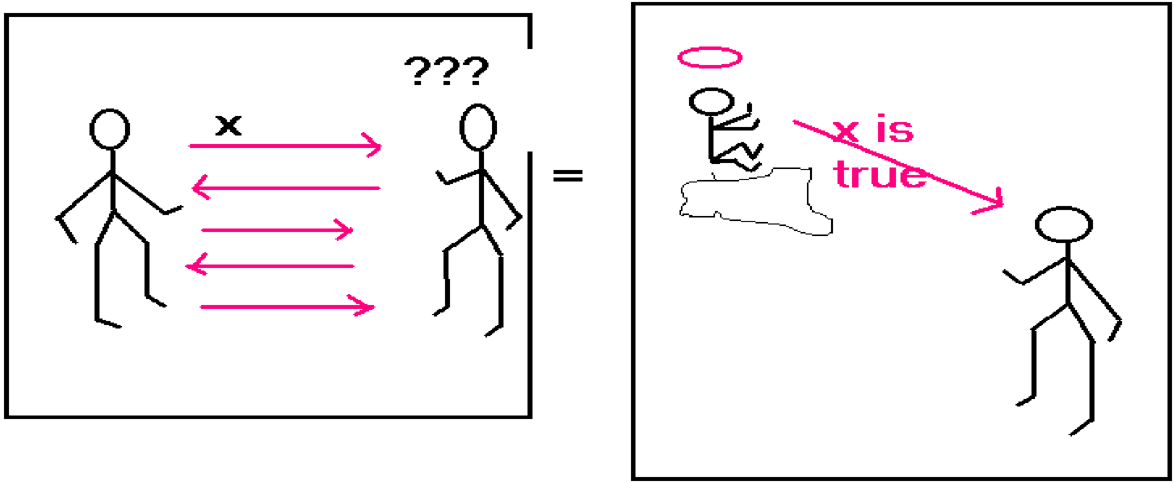
加密货币

比特币区块链虽然公钥与身份没绑定，由于区块链所有交易都公开可验证，大量隐私泄露！



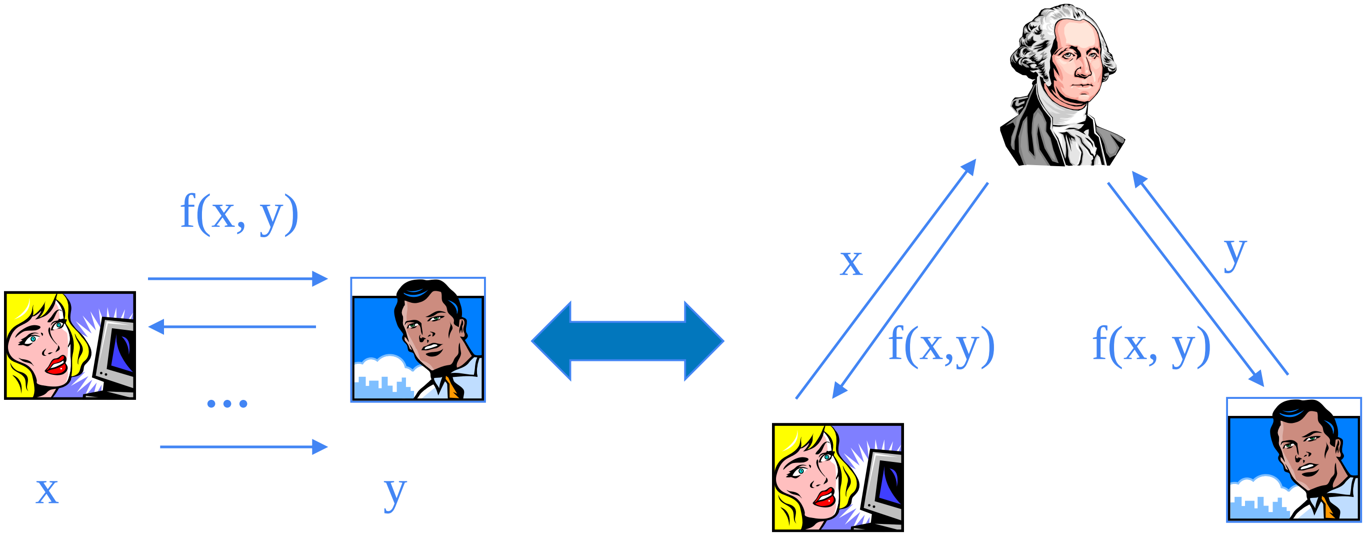






Zero-knowledge: an illustration

Secure Multi-Party Computation: An Illustration



动态简洁知识证明（SNARKs）

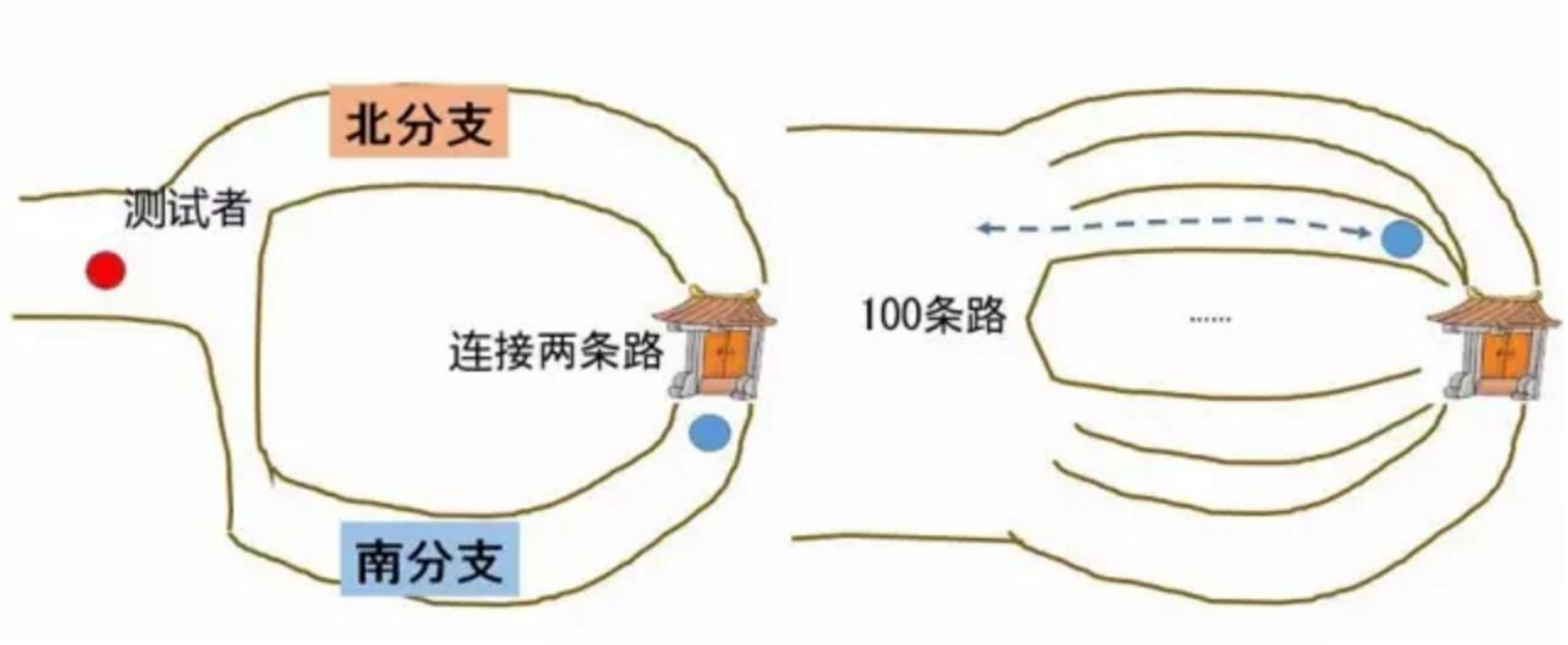
- 用户将数据x存在云端，需要计算$y=f(x)$。但$f(x)$计算量可能太大

- 用户让云服务器计算$y=f(x)$,并将y返回给用户。但用户担心云服务器是否正确计算了$y=f(x)$

- 云服务器不仅返回y，而且伴随一个动态简洁知识证明Π以证明的确$y=f(x)$

- 假设计算$y=f(x)$所需要的时间是$O(n)$,则验证动态简洁知识证明Π正确性所花费的时间仅仅是$O(\log n)$

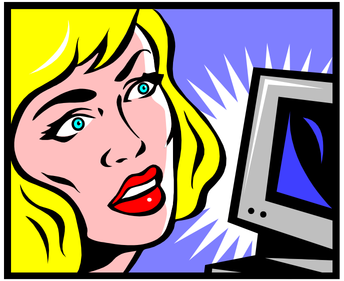
阿里巴巴零知识证明



ZK for DLP

$(p,\ q,\ g,\ h=g^w)$

Prover



$a=g^r \bmod p$ →

Verifier

← $e \in_R Z_q$

$z=r+ew \bmod q$ →

Verify: $g^z=ah^e$

Schnorr's Signature from ZK

Key generation

$w \leftarrow_R Z_q$ is the private key

$(H, p, q, g, h = g^w \bmod p)$ is the public key

Sign

Given private key and $m \in \{0, 1\}^*$

$r \leftarrow_R Z_q^*$

$a = g^r \bmod p$, $e = H(m,a)$, $z=r+ew \bmod q$

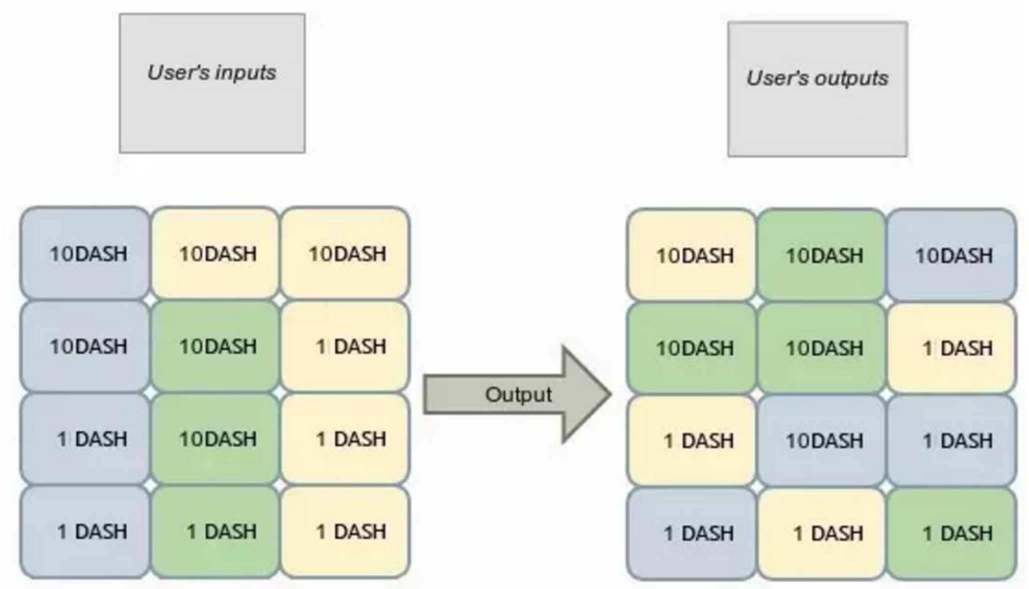
The signature is (a,e, z)

Verify

Given m, (a,e,z) and public key h

Output 1 iff $e=H(m,a)$ and $gz=ahe$ mod q

Dash (I)



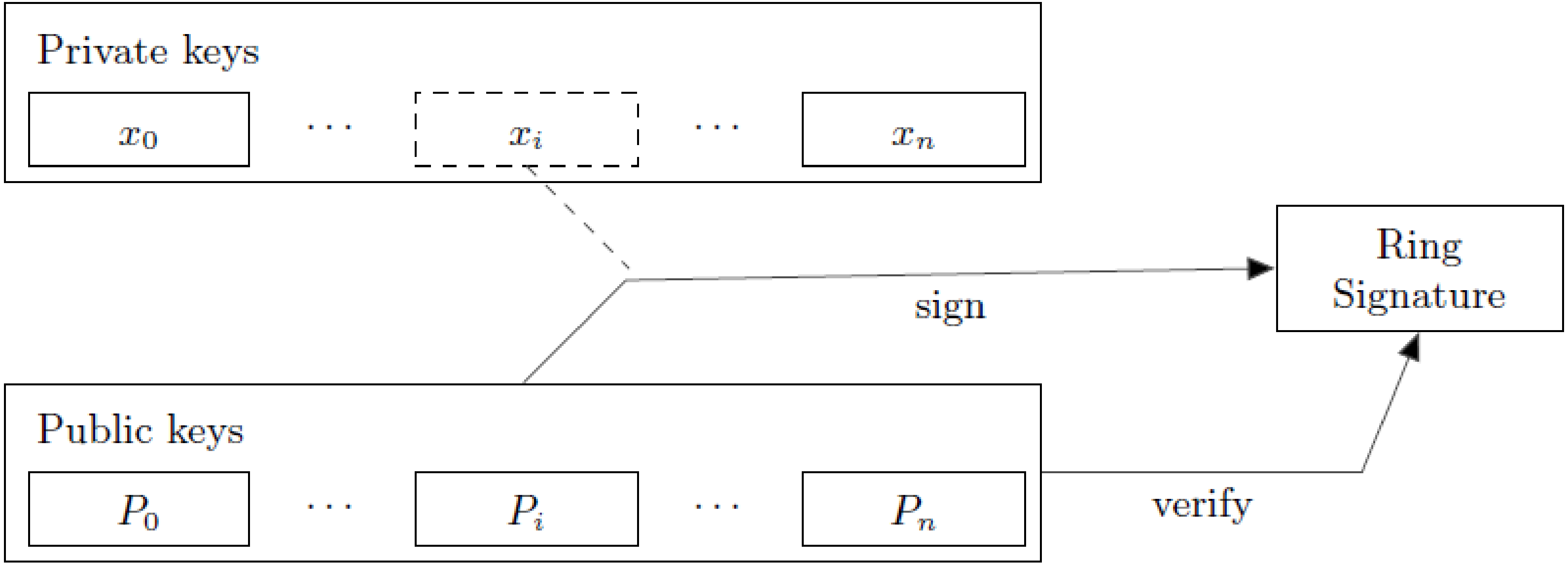
User's inputs

10DASH	10DASH	10DASH
10DASH	10DASH	1 DASH
1 DASH	10DASH	1 DASH
1 DASH	1 DASH	1 DASH

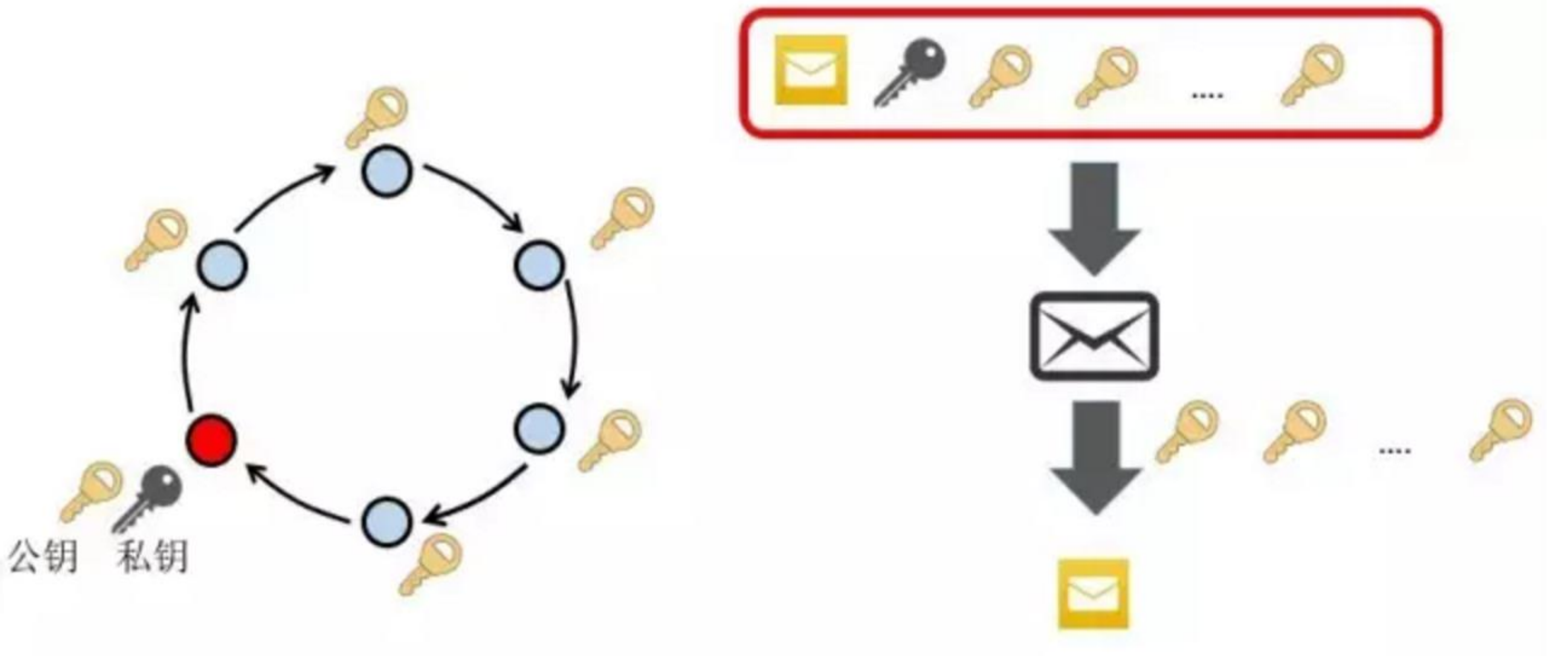
User's outputs

10DASH	10DASH	10DASH
10DASH	10DASH	1 DASH
1 DASH	10DASH	1 DASH
1 DASH	1 DASH	1 DASH

One-time ring signatures

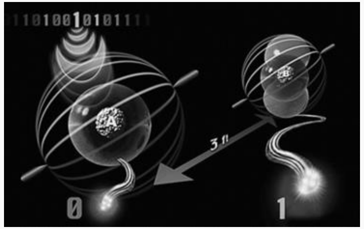


Monero



Post-Quantum Cryptography（后量子密码）

量子计算对现有公钥密码带来的影响是颠覆性的。



目前，大规模量子计算机的研制成功被广泛认为仅为一个工程挑战，并且IBM和Microsoft的工程师预计大规模量子计算机有望在未来15-20年的时间出现。

Yes, you can have one.

No, you're not dreaming. D-Wave offer the first commercial quantum computing system on the market. We believe in building great things that are as inspiring as they are powerful.

If you're passionate and curious about the future of computation, and you'd like to take a different approach to solving problems, then take a look at our products.

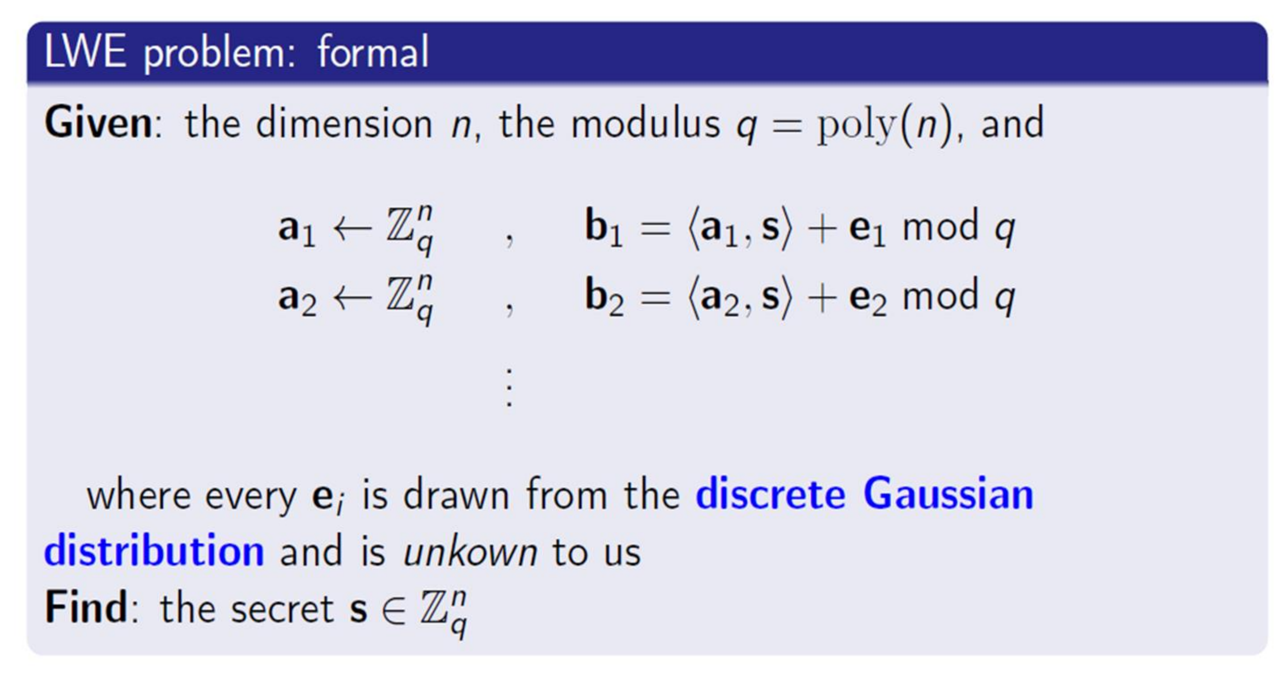
D-Wave One™ information

考虑到当代公钥密码体质从发明到大规模部署用了大约20年，需要从现在起就要研发后量子密码。

对于军事、外交和卫星保密通讯，若希望信息在15年之后仍保密，现在就应该尽快部署和应用抗量子分析的密钥协商技术。

NSA, NIST, CESG, the Tor Project, Google, ...

Lattice-based Crypto: Basic



LWE problem: formal

Given: the dimension n, the modulus $q = \mathrm{poly}(n)$, and

$$\begin{array}{lcl} \mathbf{a}_1 \leftarrow \mathbb{Z}_q^n & , & \mathbf{b}_1 = \langle \mathbf{a}_1, \mathbf{s} \rangle + \mathbf{e}_1 \bmod q \\ \mathbf{a}_2 \leftarrow \mathbb{Z}_q^n & , & \mathbf{b}_2 = \langle \mathbf{a}_2, \mathbf{s} \rangle + \mathbf{e}_2 \bmod q \\ & \vdots & \end{array}$$

where every $\mathbf{e}_i$ is drawn from the **discrete Gaussian distribution** and is *unkown* to us

Find: the secret $\mathbf{s} \in \mathbb{Z}_q^n$

Security and Privacy are the Bottle-Neck to the Future of Block-Chain Based System!

Smart Contracts
Smart Properties

The transaction and verification of Bitcoin process is a contract between transactor and miner nodes

If we express other contracts onto blockchain, we create useful applications

- Namecoin for name service；Ethereum for computing

If we can control some asset with asymmetric key pair, we get smart properties

- Trademarks, copyrights, patents
- Proof of existence service

Crime Contracts?

Smart Contract can also bring convenience of crime!

- Gambling
- Ransom
- Murder

Smart Contract can also bring convenience of crime!

- Gambling
- Ransom
- Murder

Thanks