BITCOIN

Сергей Бартунов, ВЦ РАН

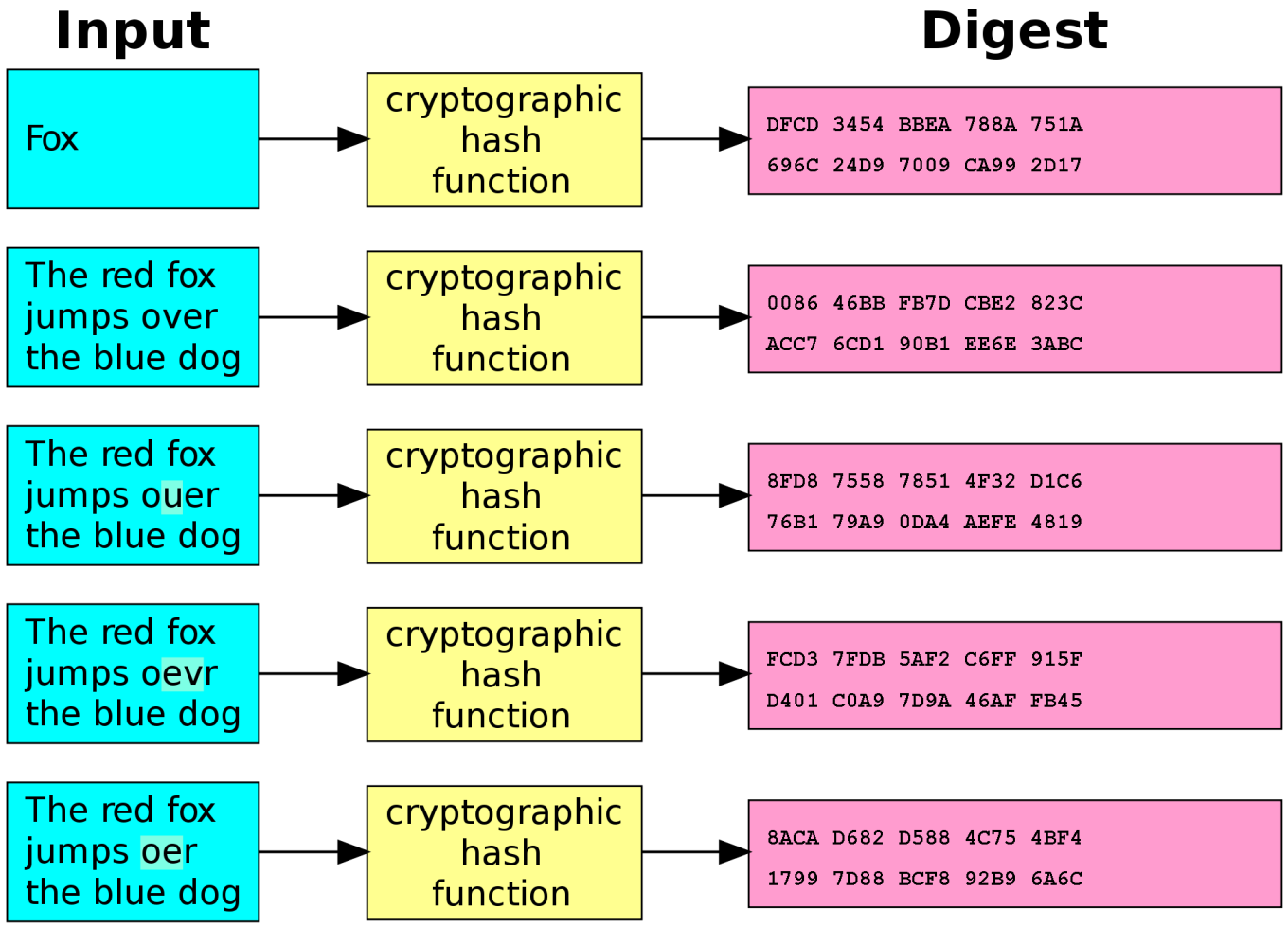
Недостатки традиционных денег

- Регулируются централизованно (есть орган управления)
- Трудно совершать электронные переводы
  - Комиссия
  - Время ожидания
- Требуют слишком много **доверия**

Что такое bitcoin?

Децентрализованная электронная платежная система, основанная на криптографических подтверждениях вместо доверия

Хеширование



Асимметричное шифрование

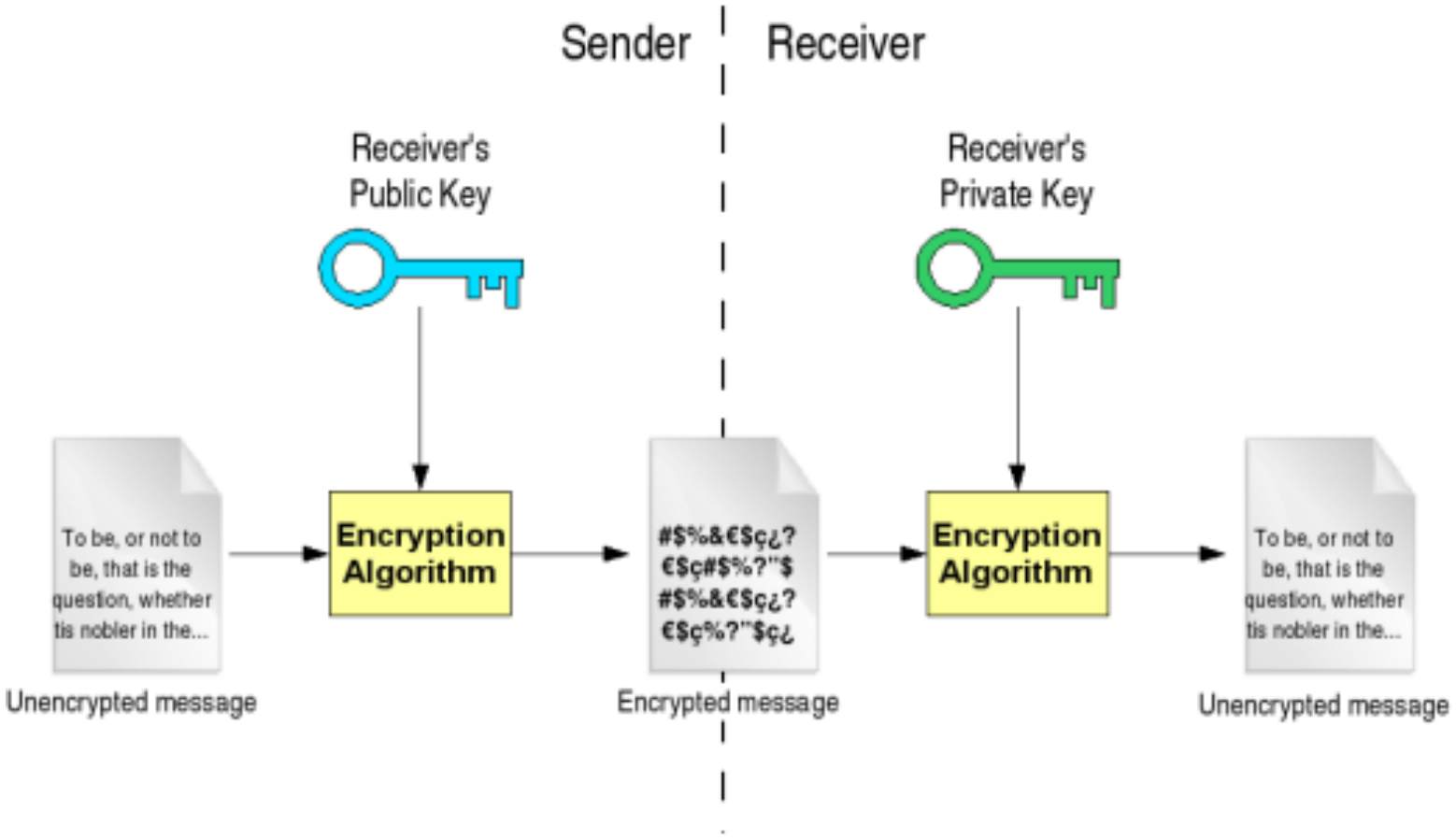
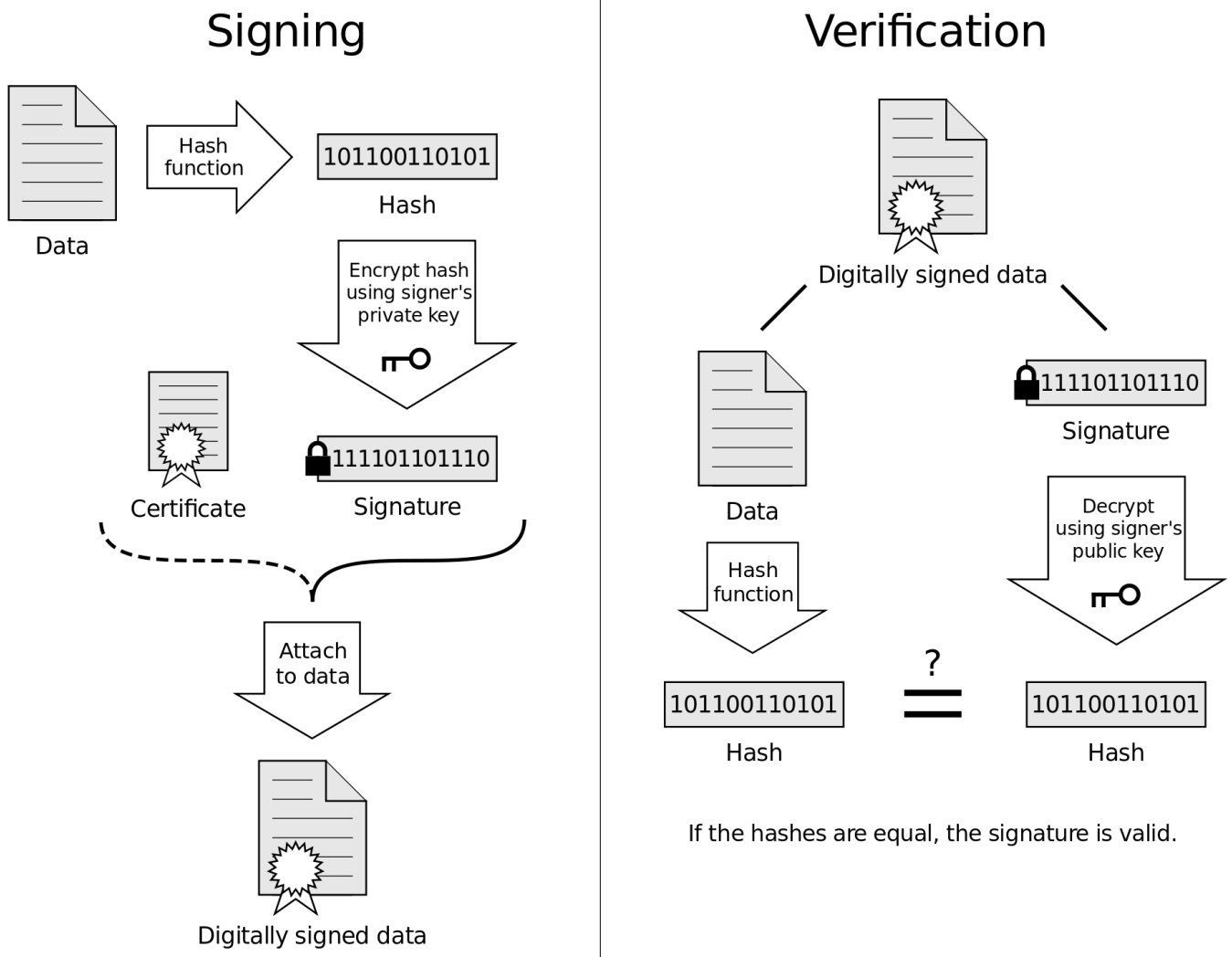


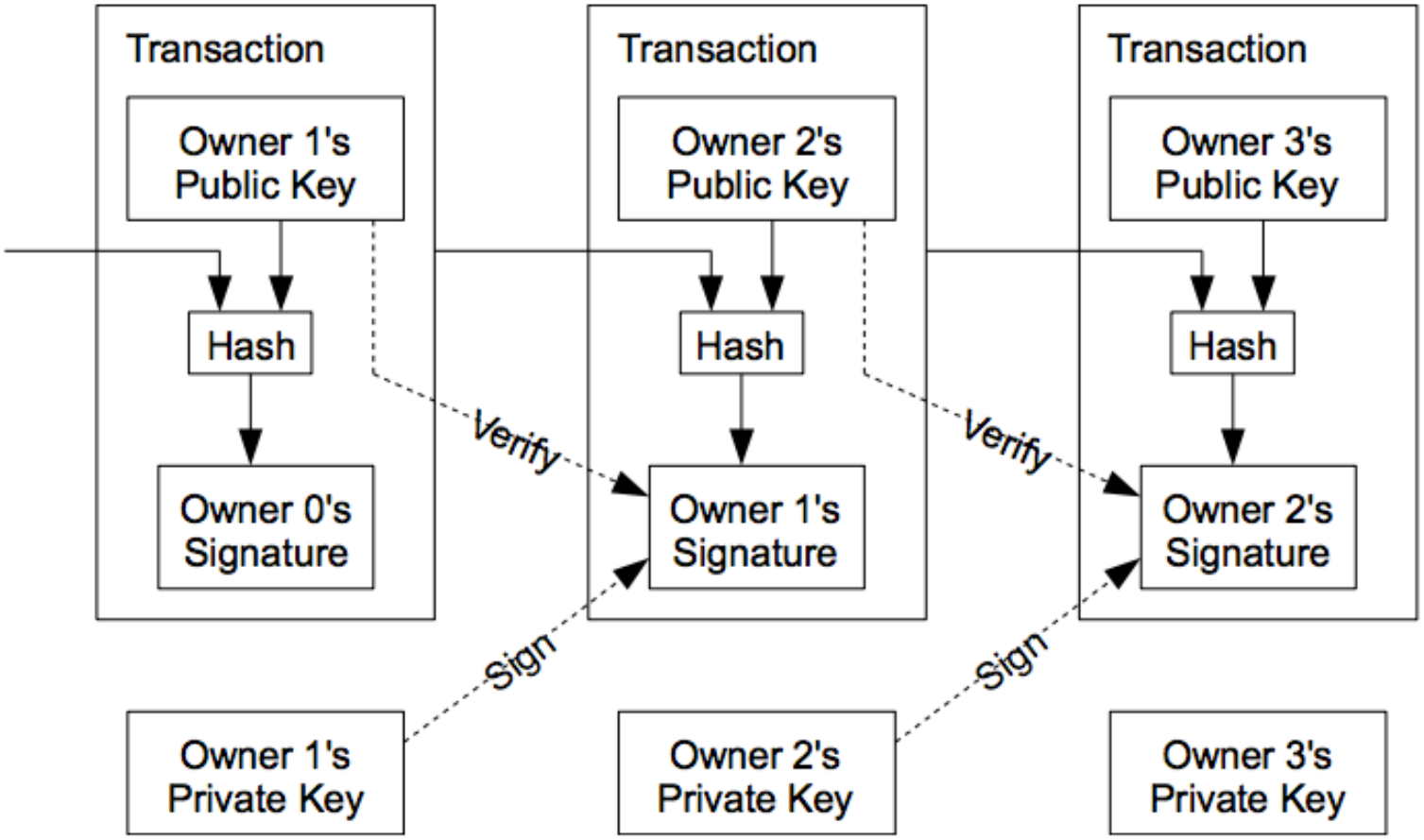
Схема RSA

- Выберем простые числа p и q, $n = pq$
- Выберем число e взаимно простое с $\phi(n) = (p-1)(q-1)$
- Найдем число d такое, что $ed = 1 \bmod \phi(n)$
- Пусть m – сообщение
- $P(x, k) = x^k \bmod n$ – функция преобразования сообщения
- $P(P(m, e), d) = m$ и $P(P(m, d), e) = m$

Цифровая подпись



Простая история транзакций

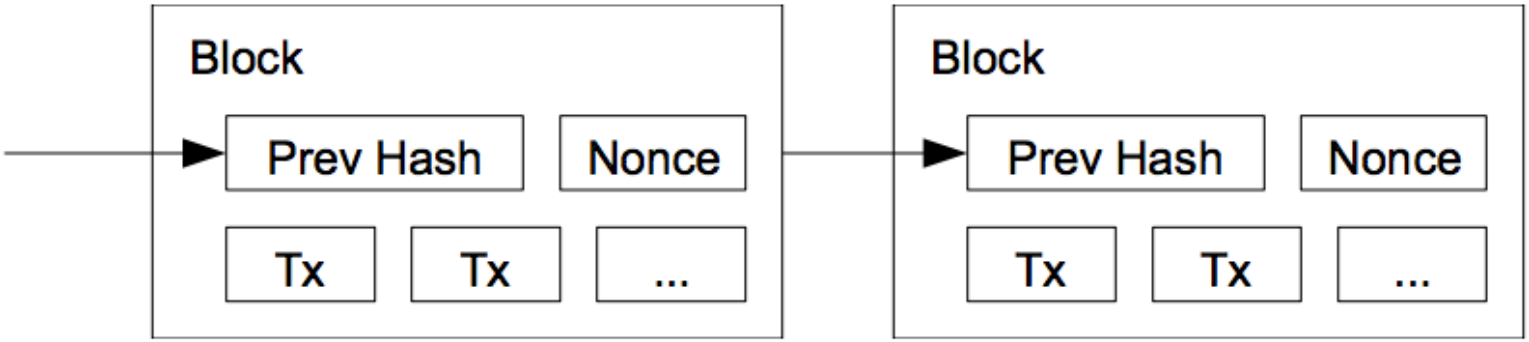


Простая история транзакций

- Легко доказать, что я перевел деньги
- Трудно доказать, что я не перевел их несколько раз (double spending)
- **Вывод:** необходимо сделать доступной историю транзакций
  - Согласованность распределенной сети
  - Защита от спама (proof of work)

Proof of work

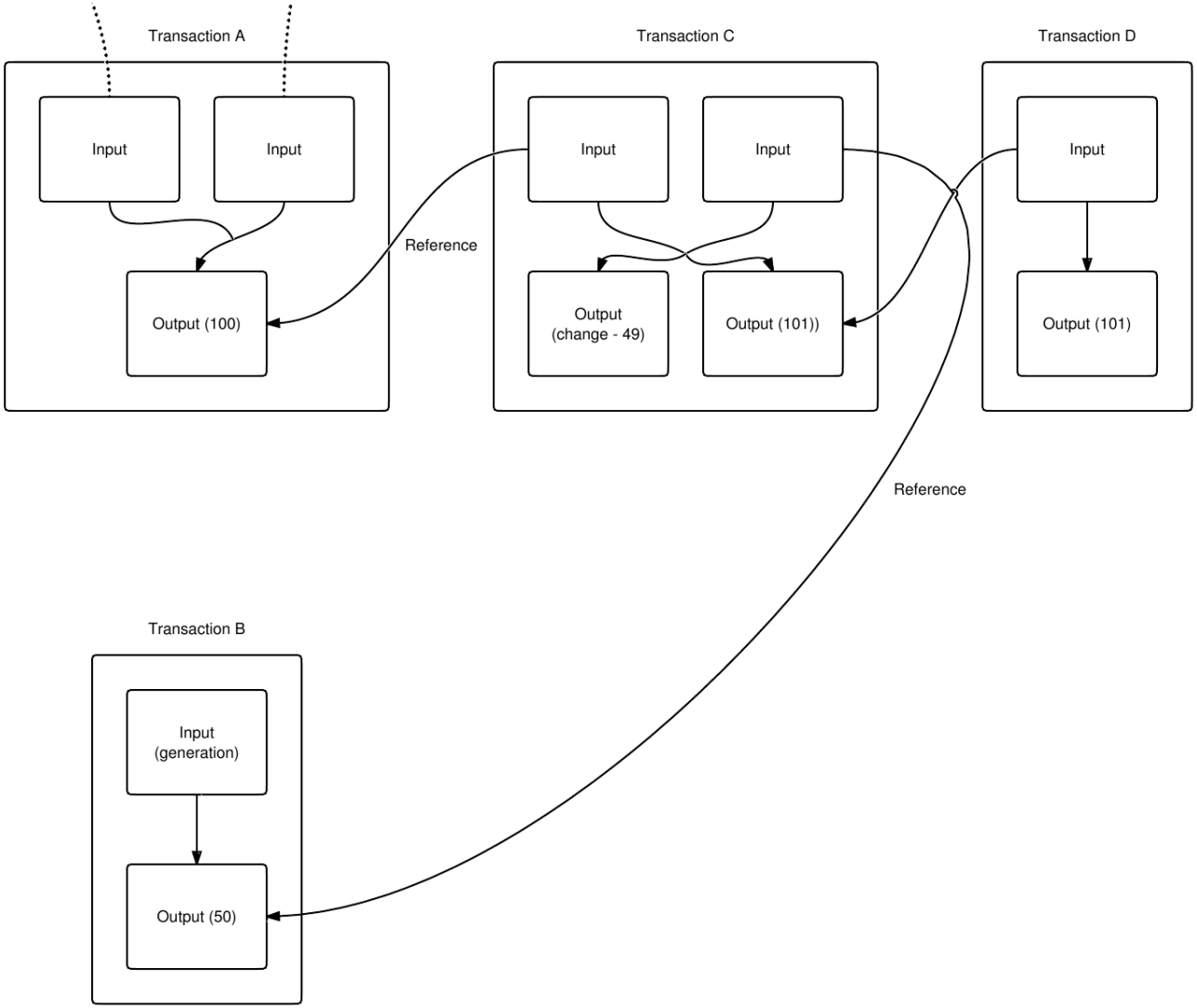
- Nonce увеличивается до тех пор, пока хеш блока не имеет заданное количество нулей в начале
- Сложность подбора зависит экспоненциально от количества нулей
- Сгенерировавший блок имеет право совершить первую транзакцию по переводу себе награды



Протокол действия сети

- Новые транзакции передаются всем участникам сети
- Каждый узел собирает новые транзакции в блок и пытается получить proof of work
- Новый блок передается другим узлам
- Блок принимается другим узлом, только все транзакции корректны
- Хеш нового блока используется в качестве хеша предыдущего блока принявшими его узлами
- Согласованность достигается продолжением самой длинной из доступных цепочек

Граф транзакций



Как устроены транзакции

- Вход(ы)
  - Адрес кошелька
  - Сумма
  - scriptSig
- Выход(ы)
  - Адрес кошелька
  - Сумма
  - scriptPubKey
- Проверка корректности
  - Выполняется scriptSig для входа
  - Результат подставляется в scriptPubKey

Простейшая транзакция

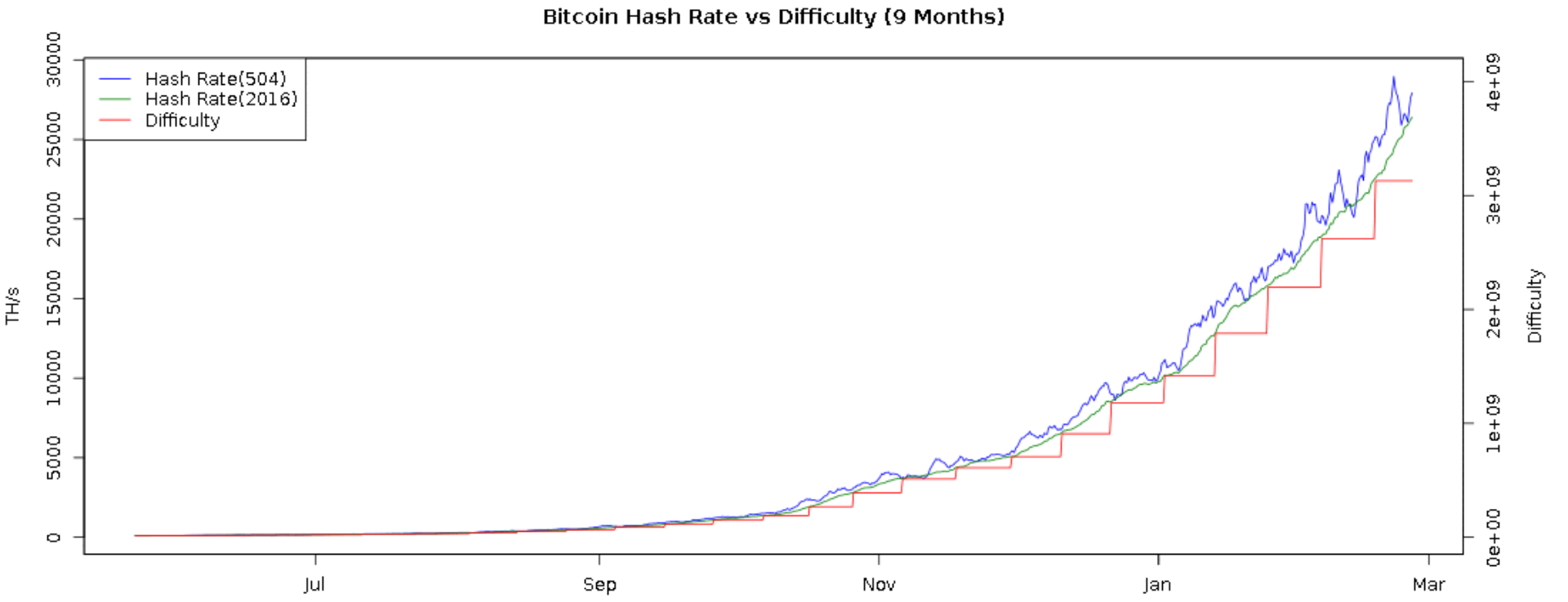
- Скрипты пишутся на некотором стековом языке программирования, похожим на Fort, не содержащего циклы
- Простейший пример (обычный перевод):

scriptPubKey: OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG
scriptSig: <sig> <pubKey>

Сложность

- Требуемое количество нулей в начале определяет сложность добычи
- Сложность автоматически корректируется каждые 2016 блоков так, чтобы в среднем скорость добычи была 1 блок / 10 мин
- Таким образом, контролируется выпуск btc. Вознаграждение снижается на 50% каждые 4 года

Историческая сложность добычи



Двойная трата в bitcoin

- Злоумышленник не сможет потратить чужие деньги и обойти цифровую подпись
- Он сможет потратить только свои деньги
- Возможный вектор атаки:
  - Злоумышленник совершает покупку (переводит btc)
  - Затем начинает строить альтернативную цепочку блоков, в которой переводит деньги на другой кошелек
  - Если его цепочка оказывается самой длинной, участники сети рассматривают ее как истинную

Двойная трата - анализ

- Пусть сложность добычи не меняется со временем
- Доля сообщества – p, доля сообщников злоумышленника – q
  - p + q = 1
- z – преимущество сообщества
  - если z < 0, то злоумышленник выиграл
  - $$z_i = \begin{cases} z_{i-1} + 1 & \text{с вероятностью } p \\ z_{i-1} - 1 & \text{с вероятностью } q \end{cases}$$

Двойная трата - анализ

- a_z – вероятность того, что злоумышленник сможет перегнать сообщество при отставании на z блоков
- $a_z = pa_{z+1} + qa_{z-1}$
- $$a_z = \min(q/p, 1)^{\max(z+1,0)} = \begin{cases} 1 & \text{если } z < 0 \text{ или } q > p \\ (q/p)^{z+1} & \text{если } z \geq 0 \text{ или } q \leq p \end{cases}$$
  - Успех атаки зависит от преимущества z
  - Если q > p, то атака завершится успехом
  - Если q < p, то вероятность успеха уменьшается экспоненциально в зависимости от z

Двойная трата с подтверждениями

- Транзакция признается, после подтверждения в n последовательных блоках
- Злоумышленник вынужден тоже дождаться n подтверждений своей первой транзакции
- За это время он успевает добыть m+1 блоков с вероятностью:

$$P(m) = \binom{m+n-1}{m} p^n q^m$$

- Начинается гонка с z = n – m - 1

Двойная трата с подтверждениями

- Вероятность успеха:

$$
\begin{aligned}
r &= \sum_{m=0}^{\infty} P(m) a_{n-m-1} \\
&= \sum_{m=0}^{n-1} \binom{m+n-1}{m} p^n q^m \left(\min(q/p, 1)\right)^{n-m} + \sum_{m=n}^{\infty} \binom{m+n-1}{m} p^n q^m \\
&= \begin{cases} 1 - \sum_{m=0}^{n} \binom{m+n-1}{m} \left(p^n q^m - p^m q^n\right) & \text{if } q < p \\ 1 & \text{if } q \geq p \end{cases}
\end{aligned}
$$

Двойная трата с подтверждениями