

Даже люди, бесконечно далекие от темы криптовалют, скорее всего слышали про майнинг. Наверное и ты, дорогой читатель, задумывался о том, чтобы включить свой игровой Pentium 4 на ночь, а утром проснуться уже богатым.

Но, как это часто случается в мире блокчейна, тех кто слышал - много, а вот тех, кто реально понимает процесс от начала до конца, - единицы. Поэтому в последней главе я постарался максимально подробно охватить все тонкости, начиная от технической реализации PoW, заканчивая рентабельностью майнинга на видеокартах.



Table of content

1. Explain me like I'm five
2. Sky is the limit?
3. Reward
4. Hard challenge
5. Technical side
6. 2 Blocks 1 Chain
7. Hardware
8. Conclusion
9. Links

Explain me like I'm five

Майнинг, также **добыча** (от [англ. mining](#) — добыча полезных ископаемых) — деятельность по поддержанию распределенной платформы и созданию новых блоков с возможностью получить вознаграждение в форме эмитированной валюты и комиссионных сборов в различных [криптовалютах](#), в частности в [Биткойн](#). Производимые вычисления требуются для обеспечения защиты от повторного расходования одних и тех же единиц валюты, а связь майнинга с эмиссией стимулирует людей расходовать свои вычислительные мощности и поддерживать работу сетей - [Wikipedia](#)

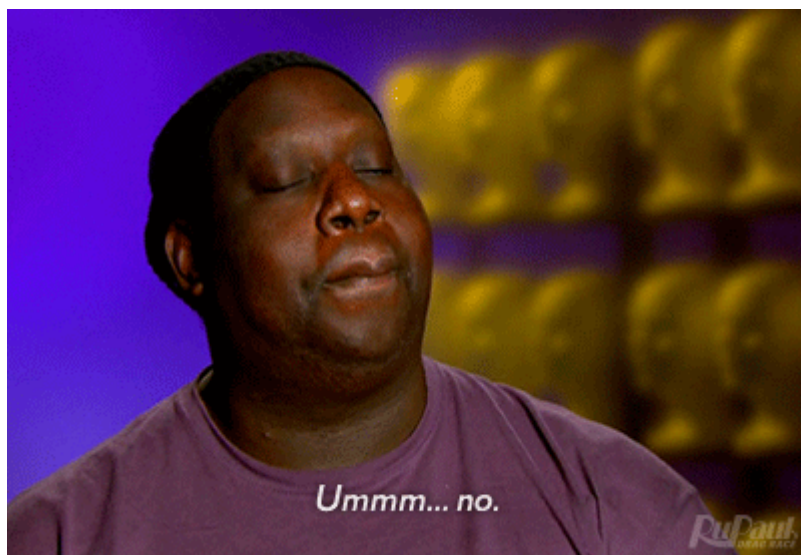
Если на пальцах, то майнинг - это критически важный для Bitcoin процесс, состоящий в создании новых блоков и преследующий сразу две цели. Первая - производство денежной массы. Каждый раз, когда майнер создает новый блок, ему за это полагается награда в N-ое число монет, которые он потом где-нибудь тратит, тем самым запуская в сеть новые средства.

Вторая, и куда более важная цель, - обеспечение работы всей сети. Наверняка, читая предыдущие статьи, вы уже задавали себе вопросы **"Кто тот человек, который проверяет скрипты транзакций?"** или **"Если в качестве входа я укажу уже использованный выход, в какой момент это заметят?"**.

Так вот, все эти действия выполняют в первую очередь майнеры. Ну, на самом деле каждый участник сети в той или иной степени обеспечивает ее безопасность. Синхронизировать Bitcoin так долго не потому что приходится качать 100 ГБ, а потому что надо проверить каждый байт, посчитать каждый хэш, запустить каждый скрипт и так далее.

Но если нарисовать весь процесс, начиная с нажатия кнопки *"Send"* в кошельке и заканчивая просмотром блока с вашей транзакцией где-нибудь на [blockchain.info](#), то именно майнеры будут решать, окажется ваша транзакция в блоке или нет.

Sky is the limit?



Для начала давайте еще раз пройдемся по первому пункту и обсудим понятие денежной массы.

Одна из фундаментальных фишек, которой часто бравируют сторонники криптовалют - [заложённая изначально дефляция](#). Это связано с тем, что еще на этапе проектировки системы, было указано суммарное ограничение в 21 миллион монет (примерно), и даже если очень сильно захотеть, поднять этот порог не получится. В отличие от рубля или доллара, которые по желанию казначейства могут

быть напечатаны в любом количестве, что иногда приводит к печальным последствиям, как в [Зимбабве](#).

BTW не все считают дефляцию таким уж [однозначным плюсом](#).

Reward

Следующий хороший вопрос - откуда взялась цифра в 21 миллион?

Я думаю вы понимаете, что **сумма выпущенных монет в конкретный момент времени равна сумме вознаграждений за блоки, созданные к этому моменту**. Довольно очевидный факт, учитывая что существует только один путь, по которому новые монеты попадают в сеть.

Но вознаграждение не фиксировано, и более того, каждые 210.000 блоков (примерно раз в 4 года) оно уменьшается в два раза.

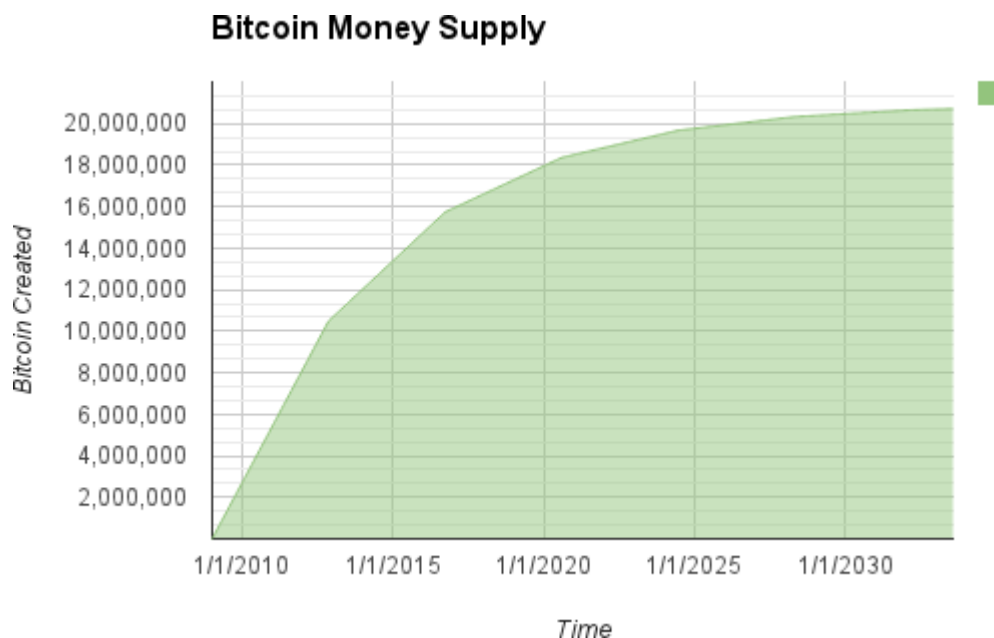
```
consensus.nSubsidyHalvingInterval = 210000;  
// https://github.com/bitcoin/bitcoin/blob/master/src/chainparams.cpp#L73
```

Так, например, когда все начиналось в январе 2009, награда за блок составляла 50 BTC. Спустя 210.000 блоков, в ноябре 2012 она упала до 25 BTC, и совсем недавно, 9 июля 2016, [снизилась до 12.5 BTC](#).

Несложно посчитать точное число Сатоши, которые будут произведены на свет, если предположить, что Bitcoin не заглохнет в ближайшие 200 лет:

```
start_block_reward = 50  
reward_interval = 210000  
  
def max_money():  
    # 50 BTC = 50 0000 0000 Satoshis  
    current_reward = 50 * 10**8  
    total = 0  
    while current_reward > 0:  
        total += reward_interval * current_reward  
        current_reward /= 2  
    return total  
  
print "Total BTC to ever be created:", max_money(), "Satoshis"  
# Total BTC to ever be created: 2099999997690000 Satoshis
```

На картинке ниже изображена кривая добычи, которая будет все более плавно подходить к отметке в 21 миллион BTC, достигнув пика примерно в 2140 году. В это время награда за блок станет 0 BTC.



Остается только гадать, что тогда произойдет с Bitcoin, но одно мы можем знать точно - совсем без денег майнеры не останутся. Как минимум у них еще есть *transaction fee*, другое дело, что эта самая комиссия может на порядок увеличиться.

Возьмем для наглядности какой-нибудь свежий блок, например [#447119](#). Сумма комиссий со всех транзакций в блоке составляет примерно 0.78 BTC, при том что вознаграждение за него - 12.5 BTC. То есть если завтра *reward* исчезнет, то в нашем случае комиссия должна вырасти более чем в 16 раз, чтобы нивелировать это неприятное событие. Понятно, что никакими микроплатежами тут уже и не пахнет.

Mining for dummies

Давайте постараемся еще раз представить процесс майнинга на нашем, пока что примитивном уровне.

Существует сеть с кучей участников. Некоторые из участников называют себя *майнерами* - они готовы собирать на своем ПК новые транзакции, проверять их на валидность, потом каким-то образом *майнить* из них новый блок, раскидывать этот блок по сети и получать за это денежку. Логичный вопрос - если все так просто, то почему этим не занимается каждый участник сети?

Понятно, что если все было бы так, как я сейчас описал, то блоки выходили бы по сто раз в секунду, валюты было бы столько, что за нее никто не дал бы и цента, и так далее.

Поэтому Сатоши был вынужден придумать алгоритм, со следующими свойствами:

- Создание нового блока - вычислительно сложная задача. Нельзя вот так просто включить мощный ПК и за минуту намайнить сто блоков.
- На вычисление нового блока у всей сети уходит 10 минут (в среднем). Если посмотреть на Litecoin, то там блоки выходят раз в 2-3 минуты, суть заключается именно в том, что среднее время заранее установлено.
- Более того, это время не зависит от числа участников сети. Даже если однажды майнеров станет в сто раз больше, то алгоритм должен так изменить свои параметры, чтобы блок стало находить сложнее, и *block time* опустился обратно в окрестность десяти минут.

- Помним, что сеть распределенная и одноранговая, а значит, она должна сама понимать, в какой момент и как нужно подкрутить эти параметры. Никаких управляющих нод, все полностью автономно.
- Если решение задачи по созданию нового блока - это сложная задача, требующая времени и ресурсов, то проверка блока на "корректность" должна быть простой и практически мгновенной.

Proof-of-Work (PoW)

Скорее всего вы сейчас прибываете в полной прострации и не очень понимаете, как такое вообще возможно. Но Сатоши не растерялся и смог придумать решение для всех этих проблем - алгоритм получил название *Proof-of-Work*, вот так он выглядит (советую сначала прочитать [Bitcoin in a nutshell - Blockchain](#)):

Пусть вы - майнер. У вас есть 10 транзакций, которые вы хотите замайнить в блок. Вы проверяете эти транзакции на валидность, формируете из них блок, в поле *nonce* указываете 0 и считаете хэш блока. Потом меняете *nonce* на 1, снова считаете хэш. И так до бесконечности.

Ваша задача - найти такой *nonce*, при котором хэш блока (256 битное число) меньше заранее заданного числа N. Поиск такого хэша возможен только тупым перебором *nonce*, никаких красивых алгоритмов здесь нет. Поэтому чем быстрее вы хотите найти *nonce*, тем больше мощностей вам понадобится.

Число N - именно тот параметр (его еще называют *target*), который сеть настраивает в зависимости от суммарной мощности майнеров. Если завтра блоки начнут выходить, условно говоря, раз в три минуты, то N будет как-то уменьшено, времени на поиск *nonce* потребуется больше и *block time* снова вырастет до 10 минут. И наоборот.

Technical side



Самое время перейти от слов к делу и продемонстрировать, как работает *Proof-of-Work* и майнинг в целом. А по моему скромному мнению, нет ничего лучше, чем показать вообще весь процесс прямо в боевых условиях. Для этого мы сейчас с ходу напишем свою майнинг ноду и даже попробуем сделать новый блок раньше всех, хотя шансы на успех невелики.

Receive transactions

По-хорошему здесь нужно снова погружаться в спецификацию протокола, устанавливать контакт с другими нодами и ждать, пока нам пришлют свежие транзакции. В этом случае у нас получится самый настоящий real-time майнер, ничем не хуже уже готовых решений (но это не точно).

Я предлагаю пойти упрощенным путем. Открываем blockchain.info и выбираем несколько транзакций из списка "*Последние транзакции*". Они только-только попали в сеть и пока что не входят ни в один из блоков. Далее открываем другой block explorer - chainquery.com. Он умеет выдавать транзакции в сыром формате и по хэсам получаем транзакции в уже знакомом нам виде. Я ограничился двумя ([раз](#), [два](#)):

```
txn_pool = []
txn_pool.append("0100000001440d795fa6267cbae00ae18e921a7b287eaa37d7f41b96ccbc61ef9a323a003d010000006a47304402204137ef9ca79bcd8a953c0def89578838bbe882fe7814d6a7144eaa25ed156f66022043a4ab91a7ee3bf58155d08e5f3f221a783f645daf9ac54fed519e18ca434aea012102965a03e05b2e2983c031b870c9f4afef1141bf30dc5bb993197ee4a52f1443e0fefffff0200a3e111000000001976a914f1cfa585d096ea3c759940d7bacd8c7259bbd4d488ac4e51320800000001976a9146701f2540186d4135eec14dad6cb25bf757fc43088accbd50600")
txn_pool.append("0100000001517063b3d932693635999b8daaed9ebf020c66c43abf504f3043850bca5a936d010000006a47304402207473cda71b68a414a53e01dc340615958d0d79dd67196c4193a0ebcf0d9f70530220387934e7317b60297f5c6e0ca4bf527faaad830aff45f1f5522e842595939e460121031d53a2c228aedcde79b6ccd2e8f5bcfb56e2046b4681c4ea2173e3c3d7ffc686fffff0220bcbe0000000001976a9148cc3704cbb6af566598fea13a3352b46f859581188acba2cfb09000000001976a914b59b9df3700adae0ea819738c89db3c2af4e47d188ac00000000")
```

Check

Следующим шагом нужно проверить полученные транзакции. Я этого делать не буду, просто перечислю основные пункты:

- Верно соблюдены структура и синтаксис транзакции
- Список входов / выходов не может быть пустым
- Транзакции на входе должны существовать либо в UTXO pool, либо в пуле неподтвержденных транзакций
- Сумма входов не меньше суммы выходов
- Полный список можете найти [здесь](#)

Некоторые майнеры отвергают транзакции с нулевой или слишком маленькой комиссией, но это каждый решает сам.

Sort

На всякий случай поясню, что [ничто не мешает вам](#) включать транзакции в блок в каком угодно порядке, главное, чтобы они прошли все проверки. В моем случае транзакций всего две, поэтому сортировать их тем более нет никакого смысла. Но не стоит забывать, что размер блока ограничен 1 МБ, поэтому если у вас в пуле 10.000 транзакций, то будет разумно отсортировать их по комиссии и записать в блок только самые "дорогие".

BTW Часто попадаются статьи / книги, в которых сказано, что перед майнингом нового блока, Bitcoin Core сортирует транзакции по специальному параметру *priority*, который считается как

```
Priority = Sum (Value of input * Input Age) / Transaction Size
```

Это было верно вплоть до версии 0.12.0, потом сортировку по *priority* [отключили](#).

Get reward

version	02000000
previous block hash (reversed)	17975b97c18ed1f7e255adf297599b55330edab87803c8170100000000000000
Merkle root (reversed)	8a97295a2747b4f1a0b3948df3990344c0e19fa6b2b92b3a19c8e6badc141787
timestamp	358b0553
bits	535f0119
nonce	48750833
transaction count	63
coinbase transaction	
transaction	
...	

Block hash

0000000000000000
e067a478024addfe
cdc93628978aa52d
91fabd4292982a50

Если вы посмотрите на структуру любого блока, то самой первой всегда идет так называемая *coinbase* транзакция - именно она отправляет вознаграждение на адрес майнера. В отличие от обычных транзакций, *coinbase transaction* не тратит в качестве входов выходы из *UTXO pool*. Вместо этого у нее указан только один вход, называемый *coinbase*, который "создает" монеты из ничего. Выход у такой транзакции тоже только один. Он отправляет на адрес майнера награду за блок плюс сумму комиссий со всех транзакций в блоке. В моем случае это $12.5 + 0.00018939 + 0.0001469 = 12.50033629$.

Давайте подробнее рассмотрим структуру *coinbase* транзакции, а если конкретнее - ее вход. На всякий случай напомним, как выглядит вход у "обычной" транзакции:

Size	Field	Description
32 bytes	Transaction Hash	Pointer to the transaction containing the UTXO to be spent
4 bytes	Output Index	The index number of the UTXO to be spent, first one is 0
1-9 bytes (VarInt)	Unlocking-Script Size	Unlocking-Script length in bytes, to follow
Variable	Unlocking-Script	A script that fulfills the conditions of the UTXO locking script.
4 bytes	Sequence Number	Currently disabled Tx-replacement feature, set to 0xFFFFFFFF

Вот три отличия входа *coinbase* транзакции:

- Вместо настоящего *transaction hash* указывается 32 нулевых байта
- Вместо *output index* указывается `0xFFFFFFFF`.
- В поле *unlocking script* можно указать что угодно размером от 2 до 100 байт, поэтому это поле еще называют *coinbase data*. Например в *genesis block* там спрятана фраза "The Times 03/Jan/2009 Chancellor on brink of second bailout for banks". Как правило, майнеры вставляют в *coinbase data* свое имя / имя майнинг пула / еще что-нибудь.

Часто в *coinbase data* вставляют так называемый *extra nonce*, подробнее [здесь](#). Суть в том, что может не найтись нужного *nonce*, при котором хэш блока меньше *target* (на самом деле это будет происходить в большинстве случаев). Тогда остается что-нибудь менять в транзакции, чтобы получились другие хэши, например - *UNIX timestamp*. Но если вы читали [Bitcoin in a nutshell - Blockchain](#), то знаете, что *timestamp* тоже сильно не изменишь, иначе другие ноды отвергнут ваш блок. Решение оказалось довольно простым: достаточно добавить какое-нибудь число в *coinbase data* и менять его, если для текущего *header* не нашлось нужного *nonce*.

Процесс создания новой транзакции подробно описан в главе [Bitcoin in a nutshell - Protocol](#), поэтому здесь я просто приведу уже полученную *coinbase transaction*, весь код, как обычно, доступен на [Github](#):

[illegible]

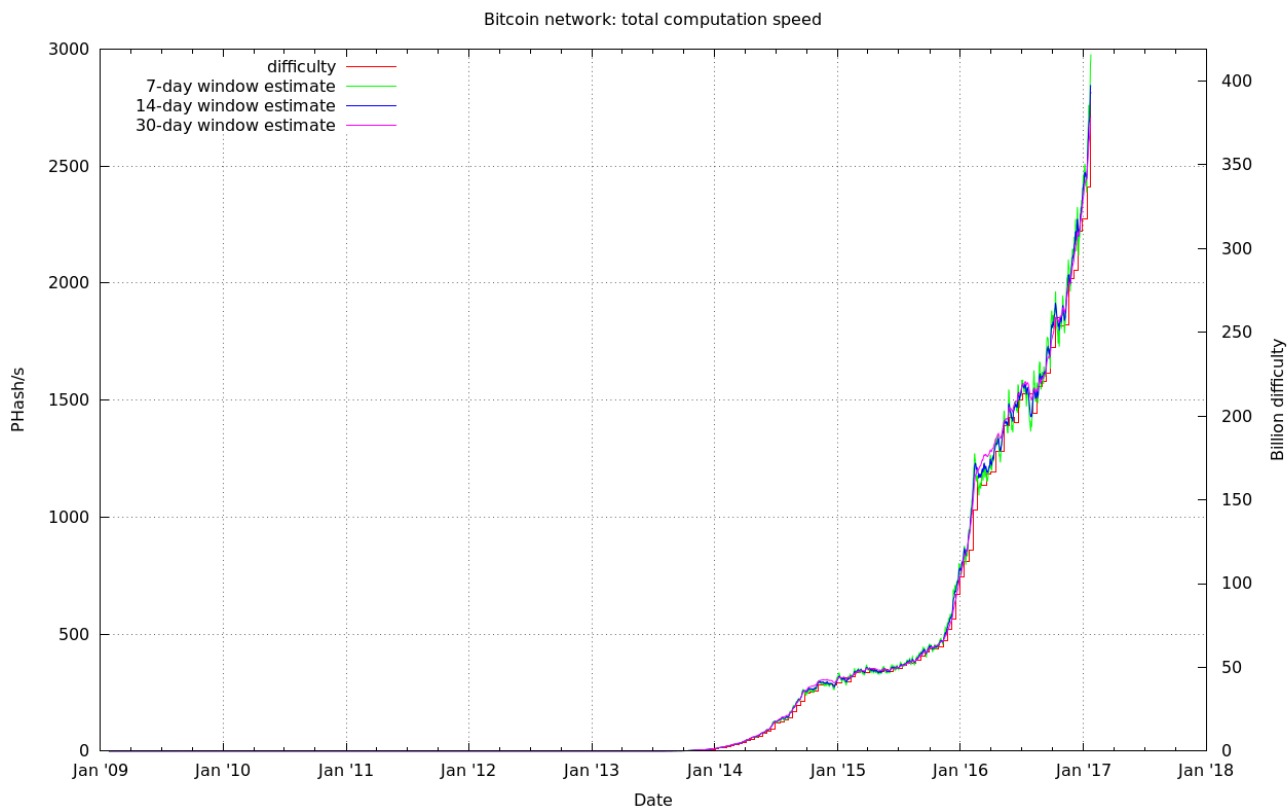
Осталось только посчитать для этих трех транзакций *merkle root*. Для этого воспользуемся фрагментом кода из [Bitcoin in a nutshell - Blockchain](#):

```
txn_pool.insert(0, coinbase_txn)
txn_hashes = map(getTxnHash, txn_pool)

print "Merkle root: ", merkle(txn_hashes)
# Merkle root: 4b9ff9ab901df82050f858accde99b9169067acafaeade25598ea5505fb53836
```

Target

Как я уже написал выше, весь майнинг сводится к тому, чтобы найти хэш блока меньше числа, называемого *target*. В структуре блока это число записывается в поле *bits*, например для блока #277,316, *target* равнялся 1903a30c .



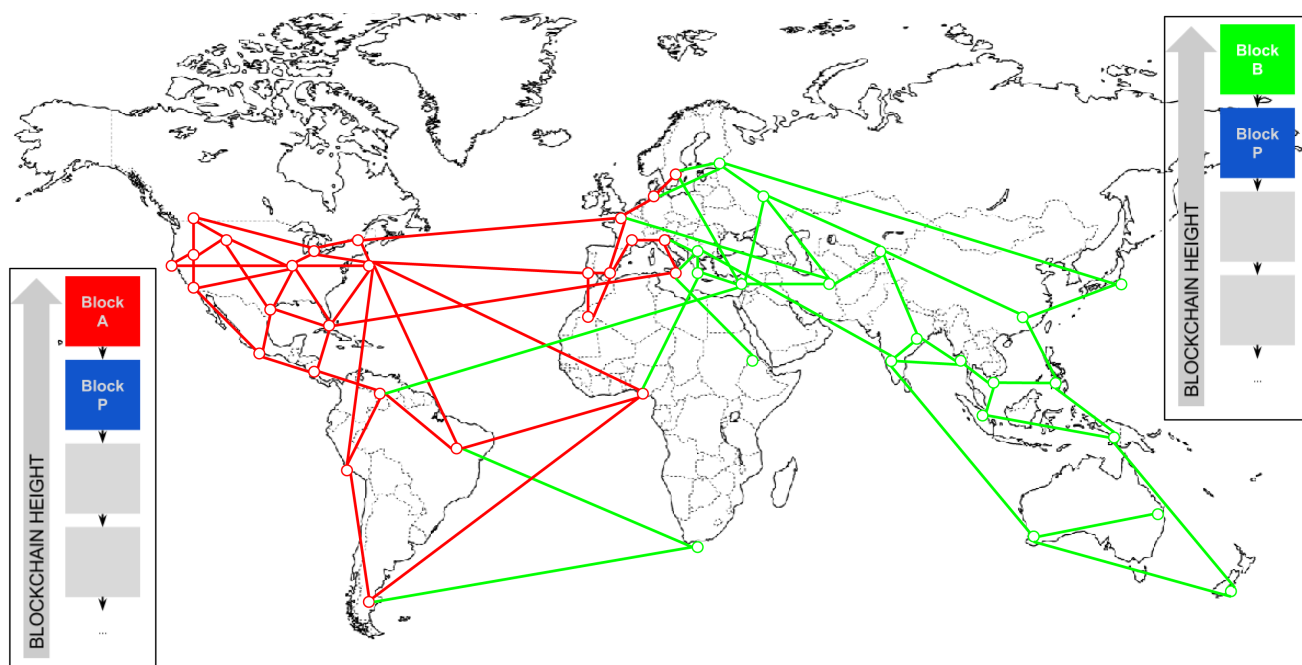
Будем считать, что хэшрейт составляет $2.000 \text{ PH/s} = 2.000.000 \text{ TH/s} = 2.000.000.000 \text{ GH/s} = 2.000.000.000.000 \text{ MH/s} = 2.000.000.000.000.000 \text{ KH/s}$. А наша программа даже 100 KH/s не может осилить, поэтому соревноваться со всей сетью нет никакого смысла.

2 Blocks 1 Chain

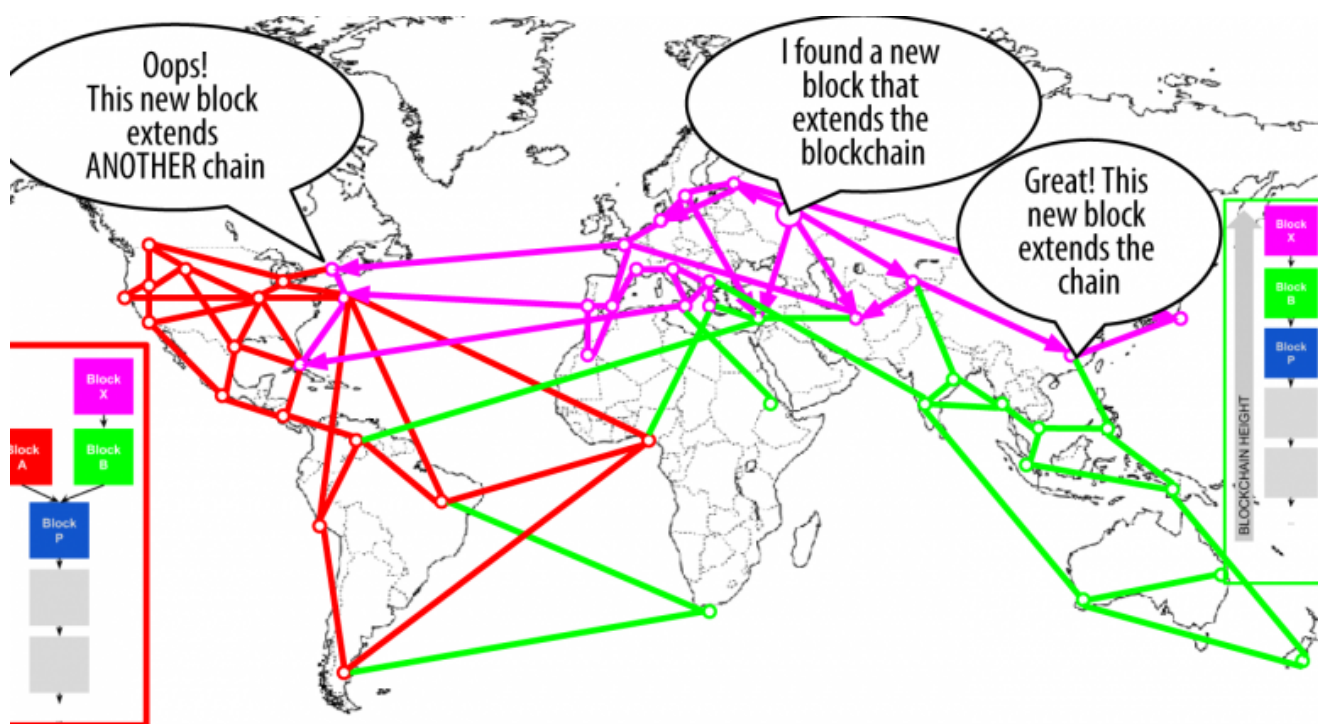
Fork

Давайте на минуту представим, что майнеры ищут блок #123456. И примерно в одно и то же время он был найден двумя независимыми майнерами, один из которых живет в Австралии, а другой в США. Каждый из них начинает раскидывать свою версию блока по сети, и в результате получается, что у одной половины мира один блокчейн, а у другой - другой.

Возможно ли такое и что произойдет в этом случае?



Да, возможно. Более того, такое происходит довольно часто. В этом случае каждая нода продолжает придерживаться своей версии блокчейна до тех пор, пока кто-нибудь не найдет следующий блок. Предположим, что новый блок продолжает "зеленую" ветку, как на картинке ниже.



В этом случае те ноды, которые придерживаются "красной" версии, автоматически синхронизирует зеленую, потому что в мире Bitcoin работает правило: **"истинна" самая длинная версия блокчейна**. "Красная" версия блокчейна будет попросту забыта, вместе с наградами для тех, кто ее нашел.

Вы можете спросить: а что если форк пойдет дальше? То есть одновременно с "фиолетовым" блоком найдут еще один, который будет продолжать "красную" версию блокчейна?

Скорее всего, эту книгу будут читать не только люди с хорошим математическим образованием, поэтому дам самый общий ответ - такое безусловно возможно. Но вероятность даже первого форка довольно мала, второго - еще меньше и так далее. Чтобы вы понимали, самый длинный форк за всю историю Bitcoin составил всего [4 блока](#). Так что в какой-то момент одна из веток все таки вырвется вперед, и вся сеть перейдет на нее.

Если вам интересна эта проблема именно с ракурса теории вероятностей, то можете прочесть ["What is the probability of forking in blockchain?"](#) Еще этот вопрос неплохо расписан в знаменитой ["Bitcoin: A Peer-to-Peer Electronic Cash System" by Satoshi Nakamoto](#).

51% attack

На том простом факте, что в блокчейне самая длинная цепочка - доминирующая, основана целая атака:

Представьте, что вы мошенник и покупаете товар на 1000 BTC в каком-нибудь магазине. Вы договариваетесь с продавцом и отправляете ему деньги. Продавец проверяет блокчейн, видит, что такая транзакция действительно была, прошла все проверки и даже попала в какой-нибудь блок, например #123. После этого продавец идет на почту и отправляет вам товар.

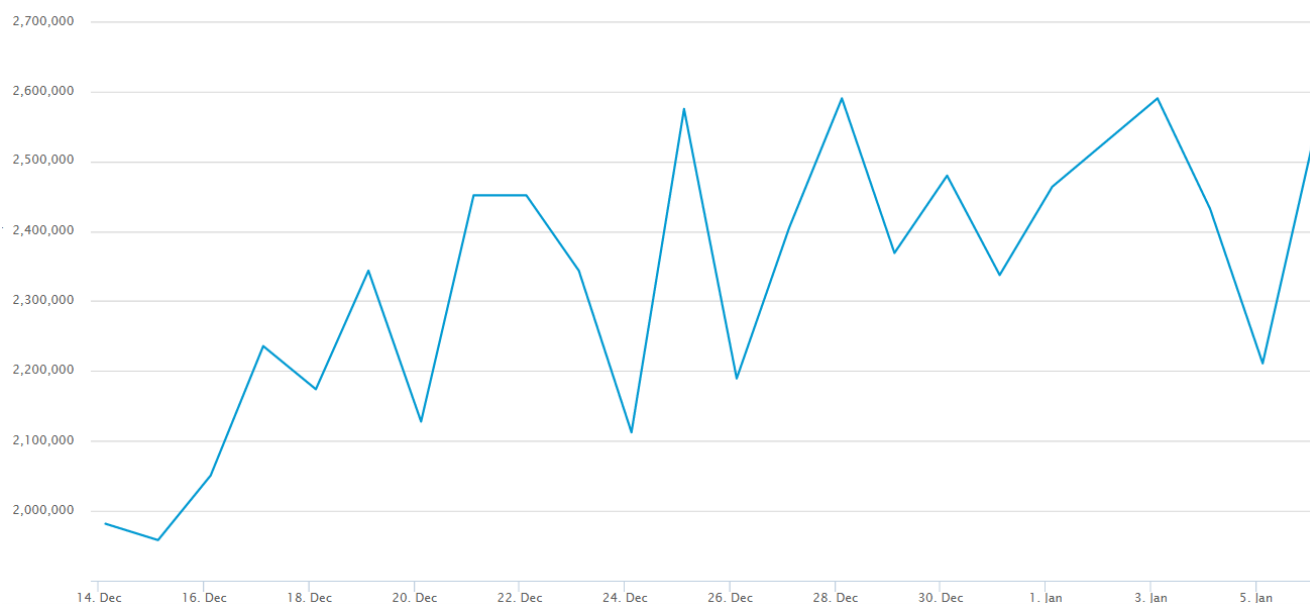
В это время вы включаете свою майнинг-ферму и начинаете майнить, **начиная с блока #122**. Если у вас достаточно мощностей, то вы можете обогнать всю остальную сеть и быстрее всех досчитать до блока #124, после чего весь мир перейдет на вашу версию блокчейна. При этом свою транзакцию на 1000 BTC, вы не будете включать ни в один из блоков, а значит она будет навсегда забыта, как будто ее никогда и не было. В результате продавец лишится товара и не получит своих денег.

Не буду вдаваться в теорию вероятностей, но осуществить такую атаку невозможно, если только у вас нет как минимум половины хэшрейта всей сети. Подробнее можете прочитать в [bitcoin.pdf](#).

Тем не менее некоторые майнинг пулы обладают очень значительными мощностями. Так например BTC Guild в 2013 году [почти преодолел](#) порог в 51% хэшрейта. В какой-то момент они замайнили сразу 6 блоков подряд, так что при желании смогли бы осуществить данную атаку. Поэтому рекомендуется считать транзакцию *подтвержденной* только после того, как было создано 6 блоков сверху.

Hardware

Можете сразу забыть про майнинг на CPU или GPU. Чтобы вы понимали, ниже изображен [хэшрейт](#) на начало 2017 года. Будем считать, что он в среднем составляет 2.300.000 TH/S, то есть 2.300.000.000.000 MH/s. Для сравнения, самые зверские видеокарты, такие как [ATI Radeon™ HD 5870 Eyefinity](#) или [AMD Radeon HD 7970 \(x3\)](#), выдают [в лучшем случае 2000 MH/S](#). Среди процессоров первое место занимает [Xeon Phi 5100](#) со смешными 140 MH/s.







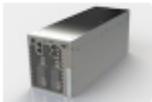
Так что даже исходя из курса в 1000 \$/BTC и имея на руках 10.000 MH/s, вы в среднем будете зарабатывать [20 центов в месяц](#).

Difficulty Factor	3.36899932796e+11	
Hash Rate	10000.0	MH/s ▼
Exchange Rate	1000.0	(\$/BTC) [user]
BTC / Block	12.50000000	
<button>Calculate</button>		

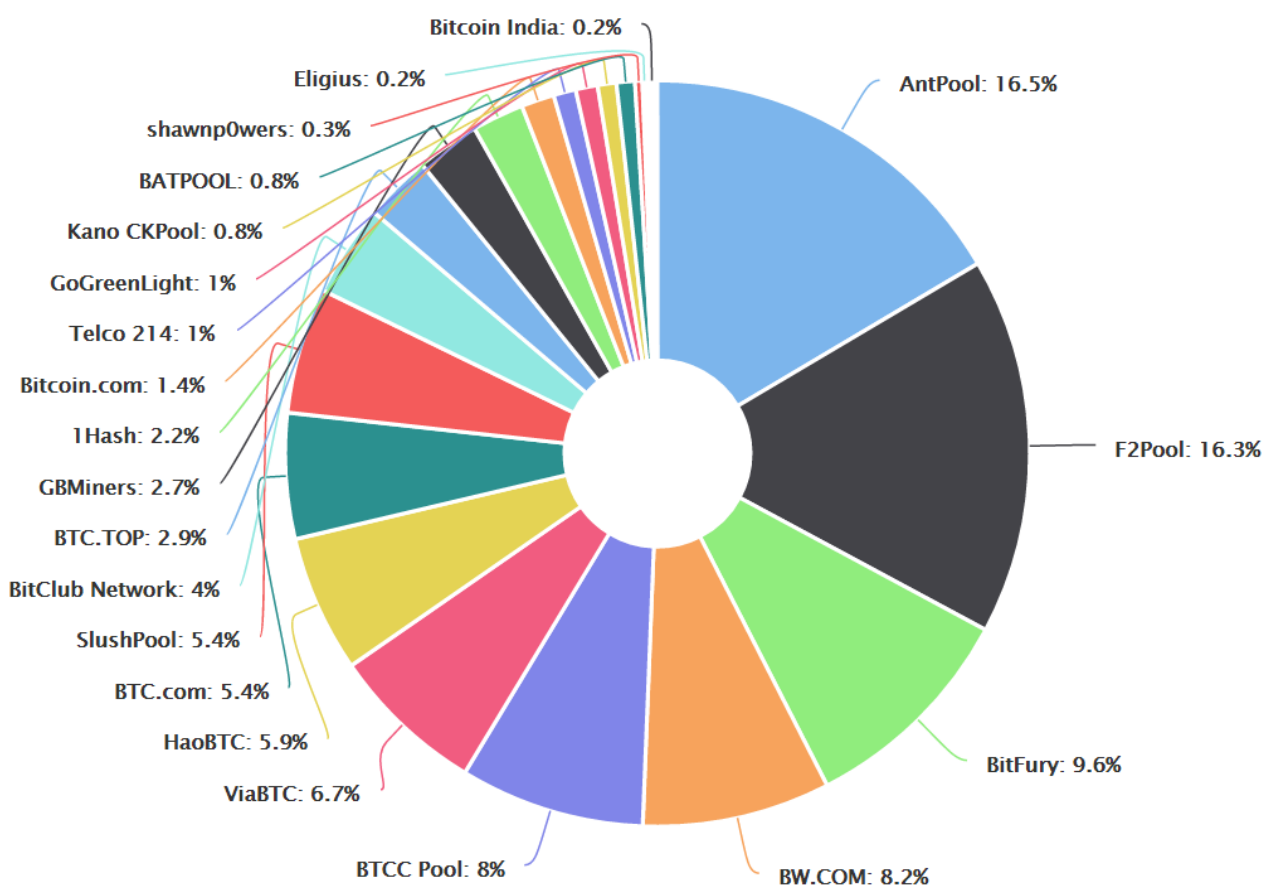
This Difficulty			Next Difficulty [estimated]		
	Coins	Dollars		Coins	Dollars
per Day	0.00000746 BTC	\$0.01	per Day	0.00000626 BTC	\$0.01
per Week	0.00005225 BTC	\$0.05	per Week	0.00004382 BTC	\$0.04
per Month	0.00022718 BTC	\$0.23	per Month	0.00019052 BTC	\$0.19
this diff (est)	0.00006920 BTC	\$0.07			

CPU майнинг перестал быть рентабельным еще в 2011 году, GPU держался примерно до 2013 года, но тоже прогорел, когда широкое распространение получили так называемые *application-specific integrated circuit* - [ASIC](#). Это специальные чипы, заточенные под майнинг на уровне железа. Самые простые стоят в районе 100\$, что сильно дешевле топовой видеокарты, но при этом способны выдавать от 1 TH/s.

То есть при прочих равных, имея два [Antminer S9](#) по 3.000\$ за штуку, вы будете зарабатывать почти [700 долларов в месяц](#) (без учета счетов за электричество)

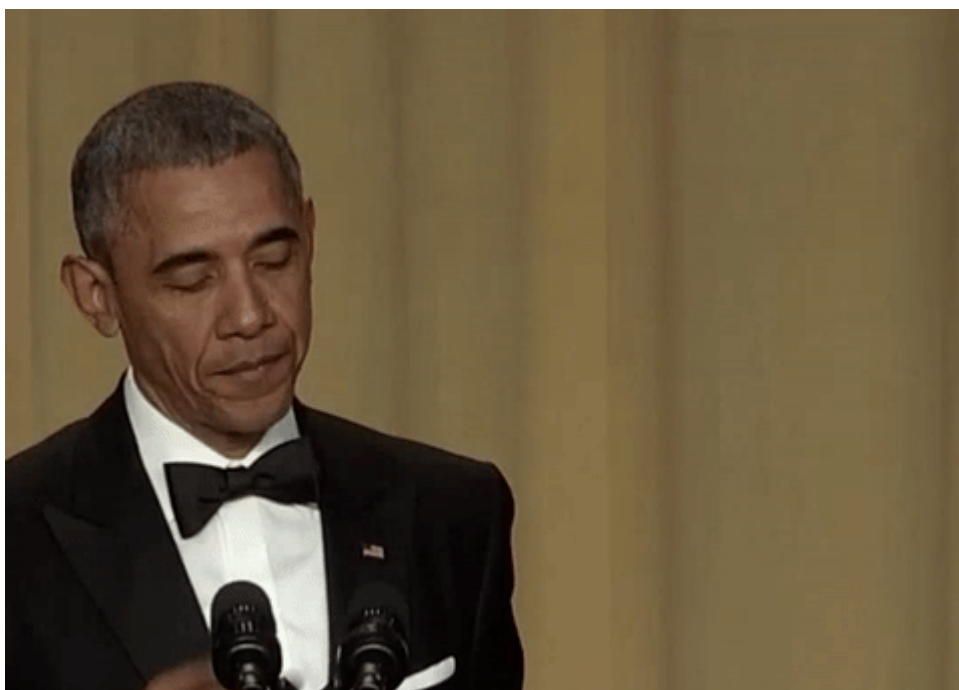
	Miner	Hash Power	Price
	Antminer S5	1.16 TH/s	\$139.99
	Antminer S7	4.73 TH/s	\$489.99
	Antminer S9	14.0 TH/s	\$3,000
	Avalon 6	3.50 TH/s	\$559.95
	SP20 Jackson	1.3-1.7 TH/s	\$90.00

Но и на этом еще не все. Вы можете объединиться с другими майнерами в *mining pool* и начать майнить вместе, а заработанные деньги делить пропорционально вложенным мощностям. Это, очевидно, намного выгодней, чем пытаться заработать хоть что-нибудь в одиночку, поэтому именно пулы на сегодняшний день составляют главную движущую силу в мире майнинга. На начало 2017 года [основными игроками](#) на рынке пулов являются [AntPool](#), [F2Pool](#) и [Bitfury](#), обеспечивающие более 40% хэшрейта всей сети.



Conclusion

На этой высокой ноте я заканчиваю свой рассказ про техническое устройство Bitcoin. Исходники текста плюс примеры кода [здесь](#), там же pdf версия. Pull requests welcome, задавайте свои вопросы в Issues или в комментариях.



Links

-
- [Bitcoin mining the hard way: the algorithms, protocols, and bytes](#)
 - [Mining Bitcoin with pencil and paper: 0.67 hashes per day](#)
 - [Mining hardware comparison](#)
 - [Bitcoin: история развития, ASIC](#)
 - [Biggest & Best Bitcoin Mining Pools and Companies](#)