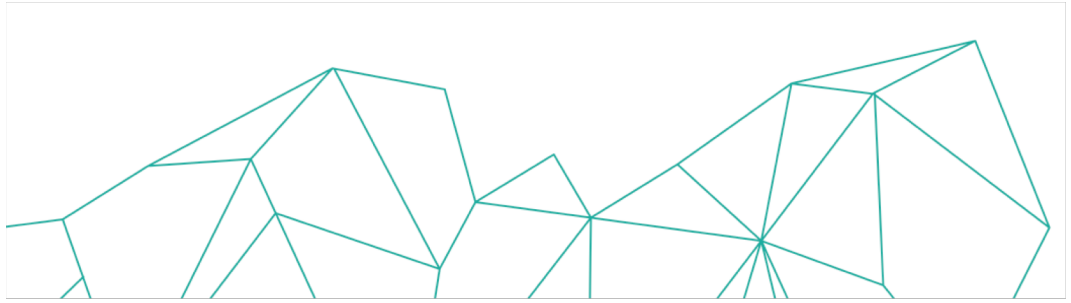
DECENTRALIZED
TRAINING SERIES

Setting the Scene

Day 1 - Session 1



UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS | iti Information Technologies Institute

Agenda

1. Introduction
2. Bitcoin's Approach To The Byzantine Generals' Problem
3. Bitcoin and Cryptography
4. Transactions and the Blockchain
5. Mining

Introduction

Some Useful Definitions

- Ledger: A **complete** record for a business's economic activities, usually used to keep track of transfer of money and transfer of asset ownership.
- Blockchain: A *tamper-proof*,* shared digital ledger that records transactions in a decentralized peer-to-peer network. The permanent recording of transactions in the Blockchain stores permanently the history of asset exchanges that take place between the peers(participants) in the network.
- SHA-256: A cryptographic hash acts like a 'signature' for a text or a data file. Can be used to confirm file integrity and authenticity. The **SHA-256** cryptographic hash generates an almost-unique 256-bit (32-byte) signature for a text or data file

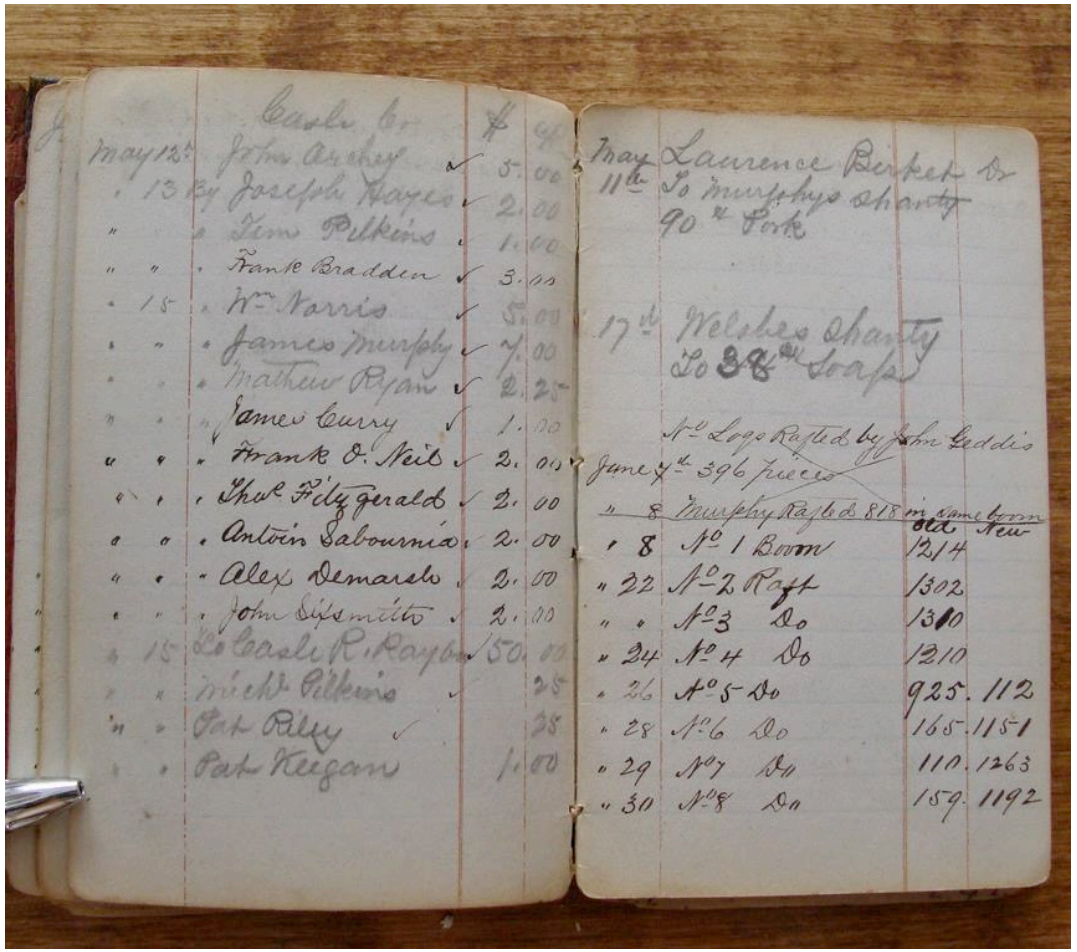
*Cannot be changed or interfered

The Role of Ledgers

- Ledgers are used to record economic activities and prove the ownership and the transfer of the value of assets among various stakeholders such as consumers, suppliers, producers and market makers

- The assets recorded in a ledger can be:
  - Tangible i.e. motor vehicles, houses

  or,
  - Intangible i.e. money, stock certificates, digital rights.



Centralized Ledgers

- We take centralized ledgers (with trusted record-keepers) for granted because we have never before had a practical alternative
- If we let any untrusted party enter transactions in an important traditional ledger, chaos is likely to ensue (would you, for example, let strangers keep track of your checking account balance?)
- Given this, a trusted party is in charge of all ledgers of importance in modern society, whether it is the bank which “stores” your funds, or your local land registry office for the title deeds for your house
- Centralized ledgers, however, are not perfect because record-keepers are not always trustworthy, act as gatekeepers and represent a Single Point of Failure
  - Record-keepers might not be trustworthy in practice. They may, for example, take a bribe to transfer a piece of land illegally
  - Record-keepers might exclude parties that they disapprove of (e.g. banks which do not allow transactions from/to cryptocurrency exchanges)
  - Record-keepers might lose important transaction records, even if they are well-intentioned, due to carelessness, natural disaster and so on

We are surrounded by centralized ledgers

Your bank account transactions

Your credit card transactions

The General Ledger underlying your company's financial statements

The ownership records of corporate securities

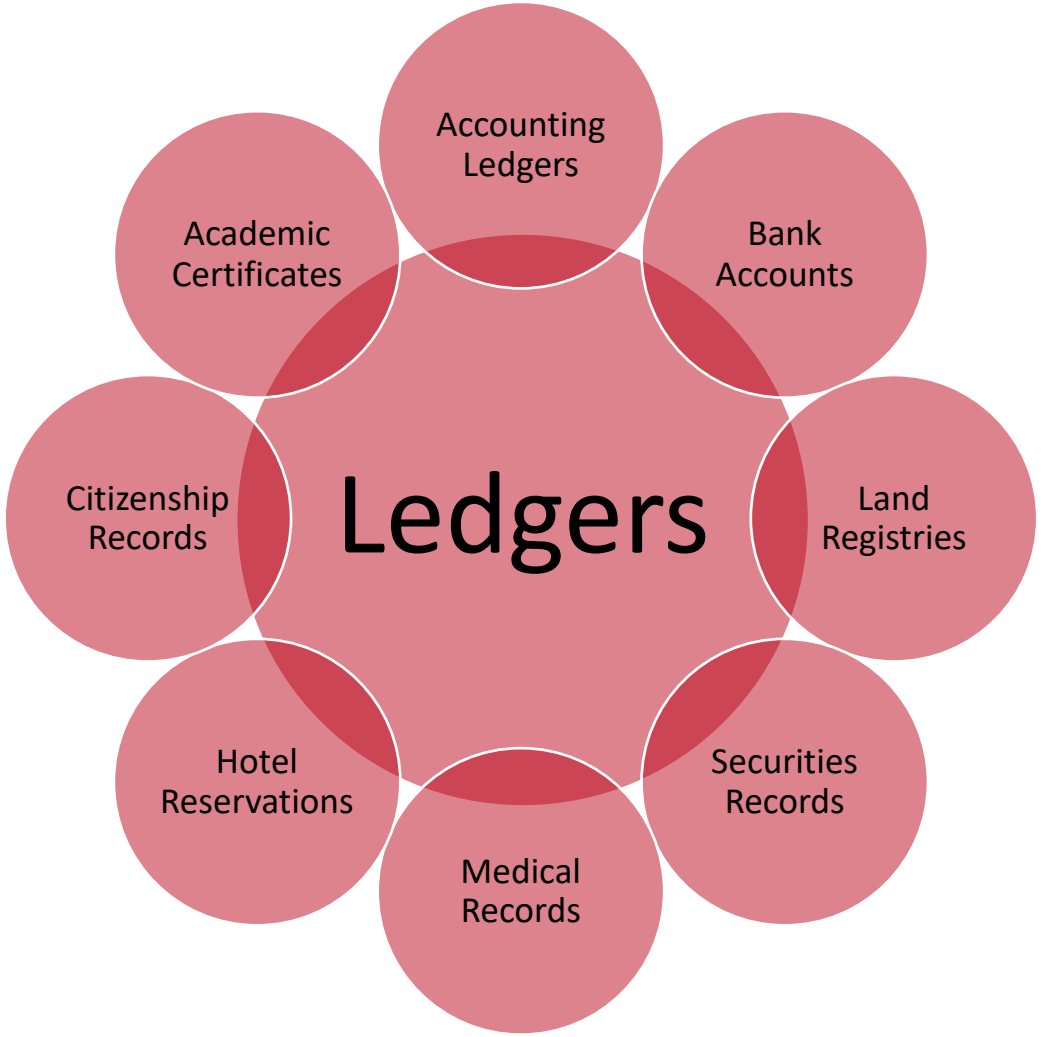
The list of title deed holders at your land registry office

The guest reservations at a hotel

The names of lessees of cars leased by a car company

The records relating to your citizenship, such as your national ID number

Examples of Traditional Business Ledgers



Are centralized record-keepers indeed trustworthy?

- Cyprus 2013 “haircut” on bank deposits
- Fake academic certificates presented to employers
- Loss of records of land registries in Less Developed Countries
- No transparency of patients’ medical records among different medical institutions

And many more…

Decentralized Ledgers

- A successful decentralized ledger that allowed parties that did not know or trust each other to transact together would have a wide range of advantages. In fact, it practically sounds like a fairy tale in traditional terms:
  - Invulnerable to censorship and exclusion
  - Invulnerable to malfeasance by record-keepers
  - Invulnerable to loss of records

One of the reasons for the excitement of technologists about Blockchain is that they believe Bitcoin is the first practical use case of this technology that could allow for the decentralization of all business ledgers

The Byzantine Generals' Problem (BGP)

- The problem of building a **distributed** and **trusted** system is not a new one in computer science. It is a common challenge in distributed systems with no central control to enforce trust. Imagine, for example, a computer system with distributed components that need to communicate information to each other, but that information might fail to communicate accurately due to technical failures

- The Byzantine Generals' Problem, first proposed by Marshall Pease, Robert Shostak and Leslie Lamport in 1982, provides a stylized description of this problem

- Past attempts at solving the currency side of the problem include the following research :
  - Chaum, D., 1984. Blind Signature System, in: Chaum, D. (Ed.), Advances in Cryptology. Springer US, pp. 153–153.
  - Chaum, D., Fiat, A., Naor, M., 1990. Untraceable Electronic Cash, in: Proceedings of the 8th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '88. Springer-Verlag, London, UK, UK, pp. 319–327.
  - Okamoto, T., Ohta, K., 1992. Universal Electronic Cash, in: Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '91. Springer-Verlag, London, UK, UK, pp. 324–337.
  - Wei Dai's B-Money - Wei Dai, 1998, http://www.weidai.com/bmoney.txt

- **Bitcoin**, however, a system proposed in a white paper released in November 2008, under the pseudonym Satoshi Nakamoto, is the best solution to this problem that has been proposed to date and has had, by far, the broadest adoption.

What is the BGP?

"We imagine that several divisions of the Byzantine army are camped outside an enemy city, each division commanded by its own general. The generals can communicate with one another only by messenger. After observing the enemy, they must decide upon a common plan of action. However, some of the generals may be traitors, trying to prevent the loyal generals from reaching agreement. The generals must have an algorithm to guarantee that (i) All loyal generals decide upon the same plan of action and (ii) A small number of traitors cannot cause the loyal generals to adopt a bad plan"

- The Byzantine Generals' Problem, 1982

Should we stay or should we go (to battle)?

Image Source: Wikimedia Commons. Text: The Byzantine Generals' Problem, Lamport, Shostak, Pease, 1982

The BPG: Problem Formulation

***Give attention to this slide. It is important to understand the problem Bitcoin and other decentralized systems aim to solve**

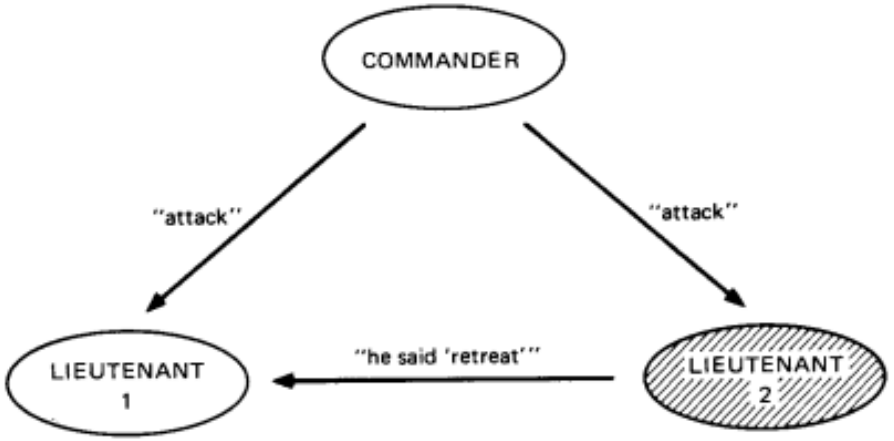


Fig. 1. Lieutenant 2 a traitor.

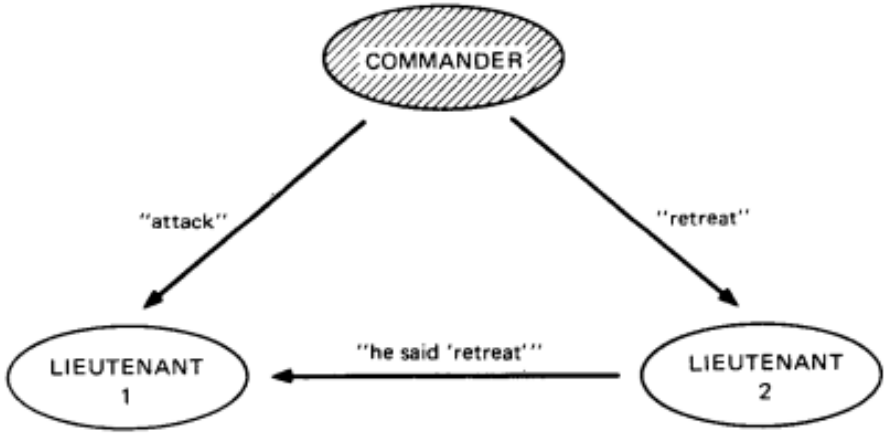


Fig. 2. The commander a traitor.

In this case, 1 traitor (either Lieutenant 2 or the Commander in the cases above) could cause the attack to fail

Lieutenant 1 would maybe retreat when he should attack instead.

A traitor prevents the group from ***reaching consensus.*** Now think of a traitor, as a malicious party within a ledger that aims to facilitate fraudulent transactions.

Image Source: The Byzantine Generals' Problem, Lamport, Shostak, Pease, 1982
Image Source: Wikimedia Commons

The Byzantine Generals

As the number of the parties in the system increase, the number of channels for communication (and opportunities for mistrust) increase exponentially.

Imagine the complexity of ***building consensus*** in a truly decentralized system with thousands or millions of parties involved.

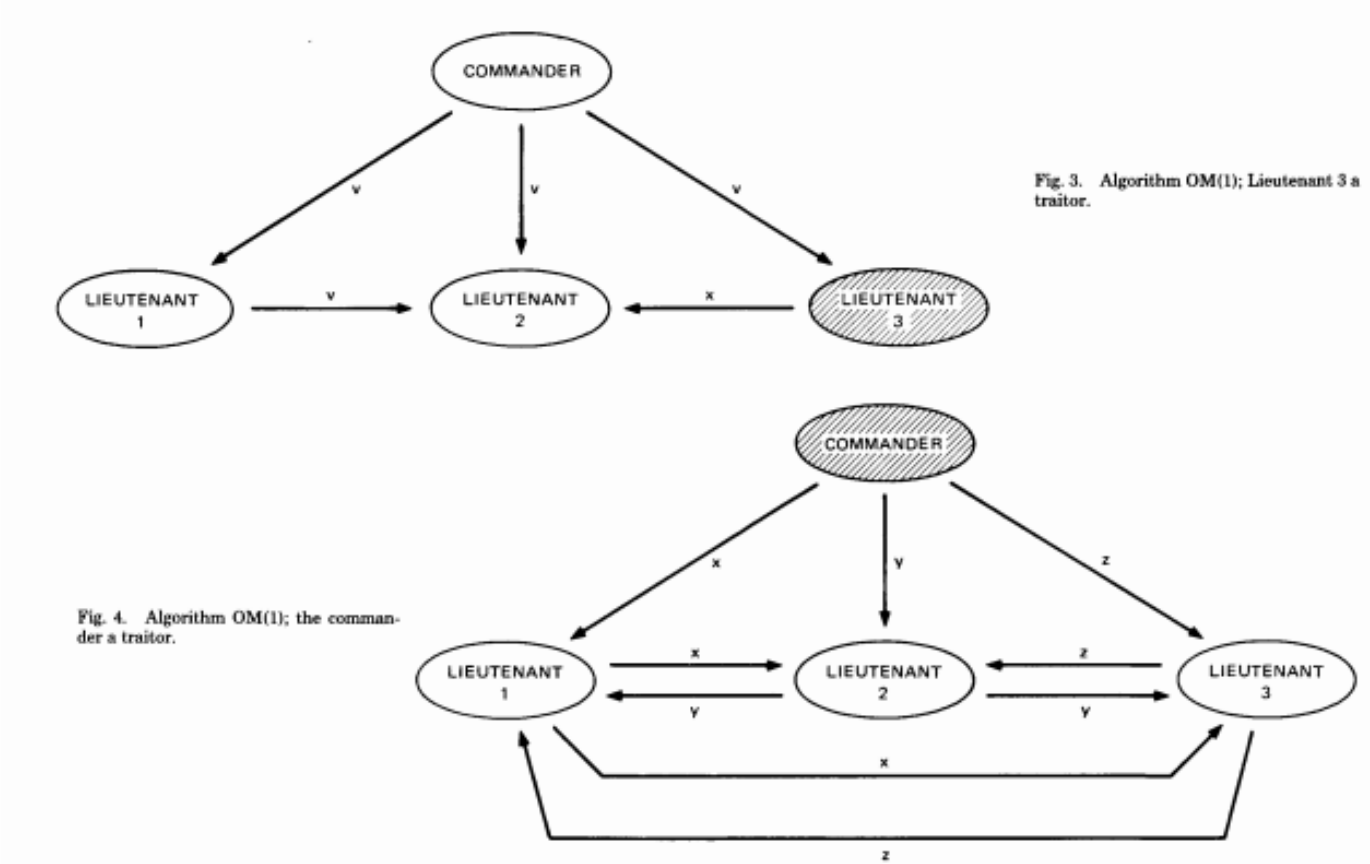


Image Source: The Byzantine Generals' Problem, Lamport, Shostak, Pease, 1982

Bitcoin’s Approach To The Byzantine Generals’ Problem

Some Useful Definitions

- **bitcoin**: without capitalization, is used to describe bitcoins as a unit of account
- **Bitcoin**: with capitalization, is used to describe the concept of Bitcoin, or the entire network itself
- **Bitcoin address**: a location that bitcoins have been sent to and reside at. It is a participant's unique identifier on the Bitcoin network and it is public
- **Transaction**: A record informing the network of a transfer of bitcoins from one bitcoin address to another. Think of it as a single line in a text book
- **Blockchain***: The complete transaction ledger of the Bitcoin network, showing how bitcoins have been transferred from one address to another over time. The blockchain is a public record of all bitcoin transactions in chronological order.

*All Bitcoin transactions are stored in blocks (think of them as pages of the text book), which are linked (or "chained") together in sequence to form the blockchain (think of it as the whole text book)

The Bitcoin Ledger: The Blockchain

- The starting point:
  - A Bitcoin user downloads a piece of software (the Bitcoin "client" software)
  - This client software will initially download the blockchain, the ledger of all transactions in the history of Bitcoin
  - Each Bitcoin client stores the complete record of all bitcoin transactions of all time. There is no central record-keeper, just a set of copies distributed among all the clients

- Once the blockchain is on a client computer, the issue of synchronization emerges:
  - How are the blockchains (ledgers) that are on each client kept in sync with each other?
  - Or, in other words, how do the blockchains reach "distributed consensus" without a central party holding the definitive transaction ledger?
  - Or, in other words, when a client receives conflicting messages about a transaction, which one should it accept and which one should it ignore? Which one is truthful, which one is a traitor?

- By now, you should realize that keeping the blockchain copies in sync is a manifestation of the Byzantine Generals' Problem

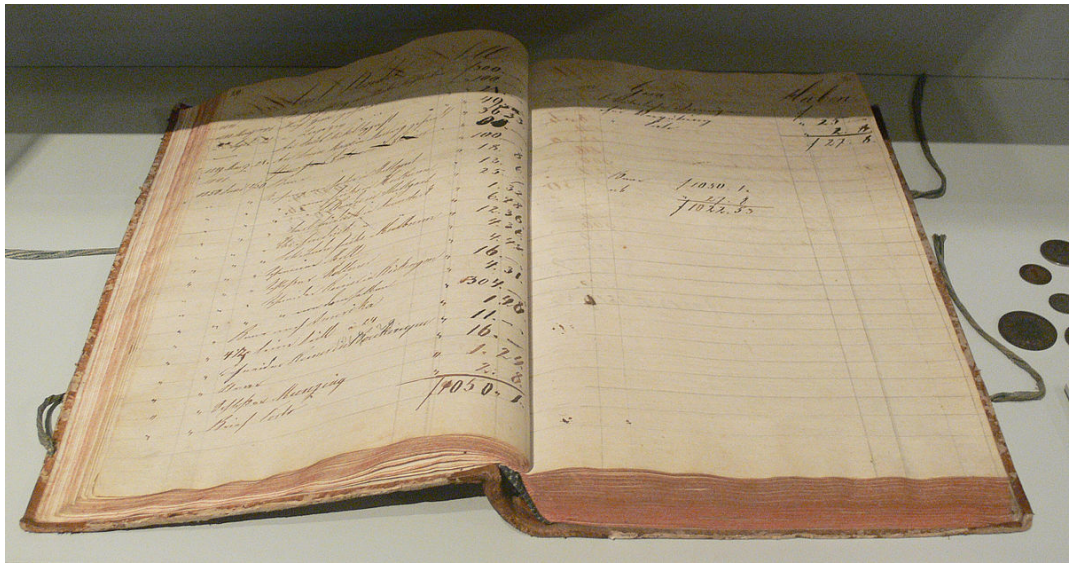
Syncing the Blockchain: Mechanics

1. When a Bitcoin client executes a transaction (sends bitcoins from one address to another), it broadcasts the transaction to all the users in the system. Within a few seconds, most of the clients in the world receive the transaction

2. At this point, however, the transaction is considered "unconfirmed" because it suffers from the Byzantine Generals' Problem. E.g., what if a dishonest Bitcoin client sent out two transactions moving the same bitcoin to two different addresses? Which one should the clients accept?

3. The mechanism that Bitcoin uses to confirm transactions and resolve the Byzantine Generals' Problem is a process called **"mining"**

Syncing the Blockchain: Mining?



This

Not This

Mining is a largely misleading analogy for what 'miners' do.
Think of the miners as bookkeepers and they will make much more sense

Image Source: Wikimedia Commons. Ledger and Coal Strip Mine

Mining & Proof-of-Work

https://www.coindesk.com/information/how-to-set-up-a-miner/

- **Mining:**
  - Creation of new bitcoins in each block, almost like a central bank printing new money. Remember though, that the amount of bitcoins to be created is fixed (21 million)
  - Creation of trust by ensuring that transactions are confirmed only when enough computational power was devoted to the block that contains them.

1st Step	2nd Step	3rd Step
Miners collect and broadcast their transactions in a block and verify they are valid	Apply a cryptographic hash function of the header of the previous block to the new block	Try to solve the **"Proof-of-Work"** Problem

Proof-of-Work Problem:

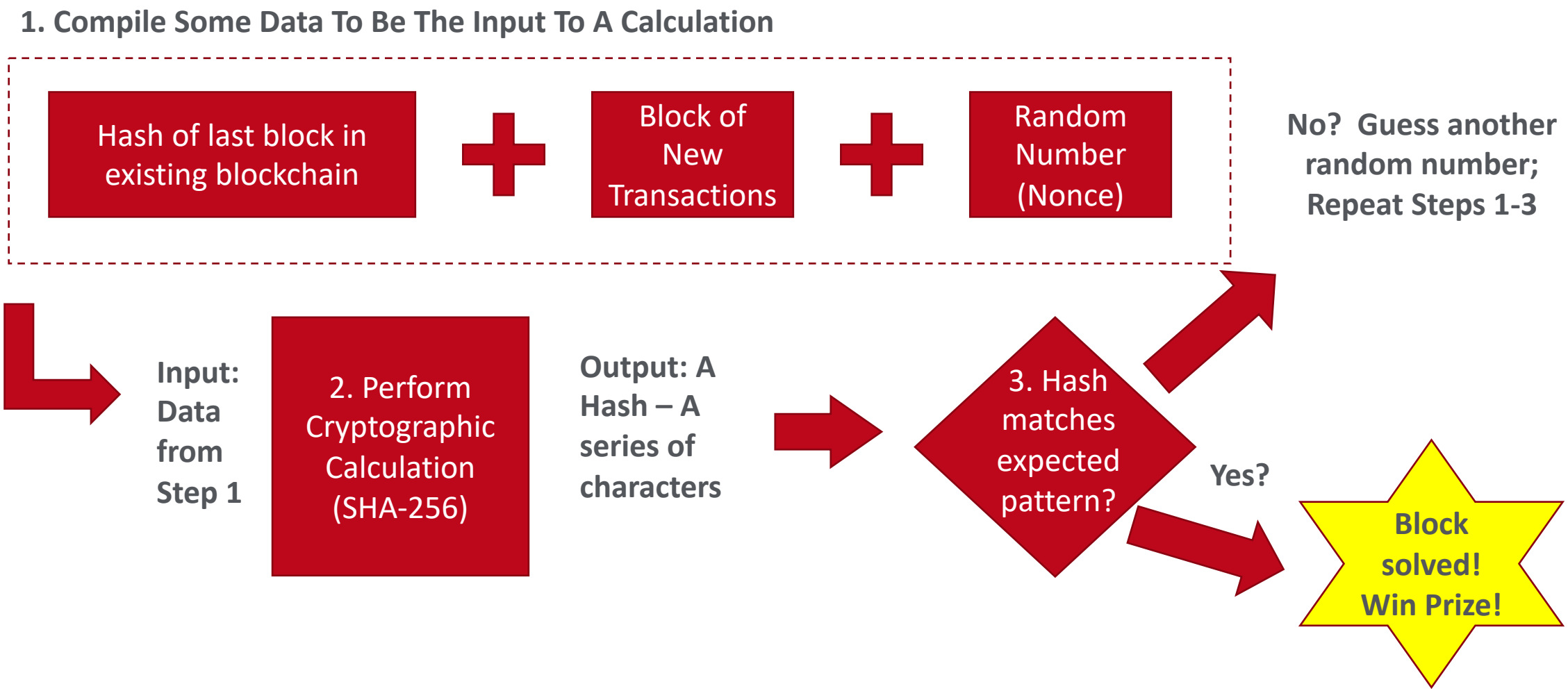
Step 1: Hash of Last Block Header + Block of New Transactions + Random Number (Nonce=32-bit number)

Step 2: Apply Cryptographic Function (SHA-256) to the above data

Step 3: Hash reviewed against a predetermined value(a desired pattern). Less than this value=Prize. Not less=Guess Nonce Again & Repeat

- *Winning Block Verified by Nodes as a Block and broadcasted to the network*

Mining in 3 steps (again)



Mining: Winning a Prize?

- Once a miner has a winning block, it broadcasts it to the other clients as a “block” of confirmed transactions.
  - The client nodes verify that the hash matches the pattern needed and accept the new block, adding it to the existing blockchain that they all store. Note: Blockchain = a chain of blocks (!)
  - After that, all miners start working on finding the next block, incorporating the new, larger blockchain as their starting point in Step 1

- The miner is allowed to collect as part of having a winning block:
  - An allocation of new bitcoins (currently, 12.5 bitcoins per block) that is an increase in the total number of bitcoins
  - The transaction fees from all the transactions that were included in the block

- Block reward started from 50 bitcoins per block and is halving every 210,000 blocks, approximately every 4 years

- In September 2018, the block reward is 12.5 bitcoins. This amount far outweighs the reward from transaction fees. The next halving will take place in 2020. Block rewards will stop once the network reaches Block 6.930.000 (sometime around the year 2140). The total number of bitcoins issued by then will be about 21 million - https://en.bitcoin.it/wiki/Controlled_supply

Mining: Auto-Adjusting Puzzles

- This winning of prizes sounds very pleasant. Why hasn't someone with a big computer mined all the bitcoins?

- Fortunately, the prizes (new blocks) auto-adjust their difficulty (leading zeros) to account for how much computing power is in the Bitcoin network.
  - The difficulty of guessing the correct random number (nonce) that produces the desired number of leading 0s in the block hash, is adjusted every 2016 blocks (approximately two weeks) so that the network produces one successful block every 10 minutes or so.
  - While any given block might take less time or more time to create due to luck, if blocks start being produced too often or not often enough, the prize puzzle gets more difficult or less difficult so that blocks are formed every ten minutes or so again

- **This means that, whether the Bitcoin network just has 20 old laptops doing mining or millions of super-computers mining, blocks will still be created every 10 minutes and anyone's expected reward from mining is their % of the total network's computing (mining) power**. And since anyone can setup a mining node, if it is very profitable to mine because, say, the price of a bitcoin has gone up, more miners will come into the system (and vice-versa, if it is unprofitable, miners will drop out)

Back to the Byzantine Generals' Problem

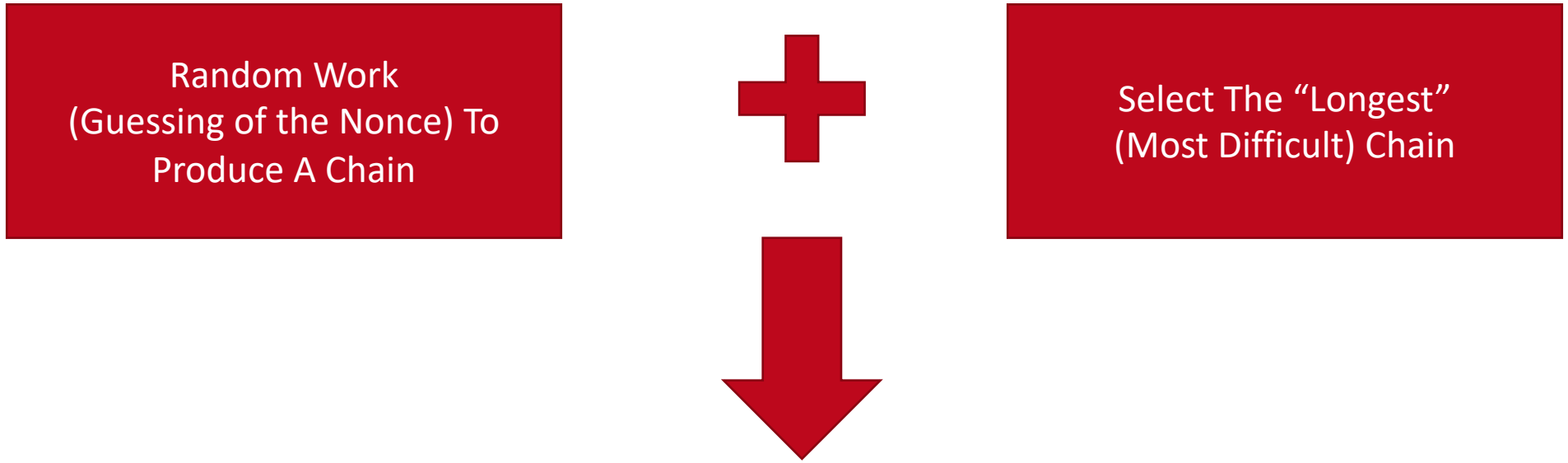
- An astute reader might say: "But, you have not yet solved the BGP, just moved it to the miners. What if two miners send out blocks with different information(i.e. different transactions within the block)? How do the clients choose which one to include?"

- The answer is that when a client is trying to decide which blockchain version to accept, it must choose the one that is "longest" (In Bitcoin terms), aka the one that has the "greatest combined difficulty" (of the hashes used to create it). In other words, the chain that took the most computation power to create will be chosen

- The shorter blockchain is discarded (as an "orphan block" and those transactions have to be re-processed if they were not in the longer blockchain)

- Given this, a traitor cannot keep entering bad signals into the blockchain (aka, spend money in one block, then erase the transaction in the next block) unless he or she can keep producing the longest blockchain (the one with the most computation associated with it) which, statistically, the traitor can't do, given the essentially random nature of block creation (unless he has enough hashing power and can sustain mining the majority of new blocks). More on this on the next slides.

Back to the Byzantine Generals' Problem



Solution to the Byzantine Generals' Problem

Bitcoin and Cryptography

Mining: Proof of Work

- The random number creation (“proof of work”) is the subject of great confusion by laymen
- They often consider it (a) wasted effort or (b) an indication of poor system design (“why do they need to do so much work to enter a transaction when my database can just do it instantly?”)
- In fact, it is the *key* aspect of providing ledger security, in that it prevents any one party from hijacking the ledger

One useful mental model is to think of it as a lottery relating to who gets to enter the next transactions in the system to prevent one person from taking control of the ledger

Bitcoin and Cryptography

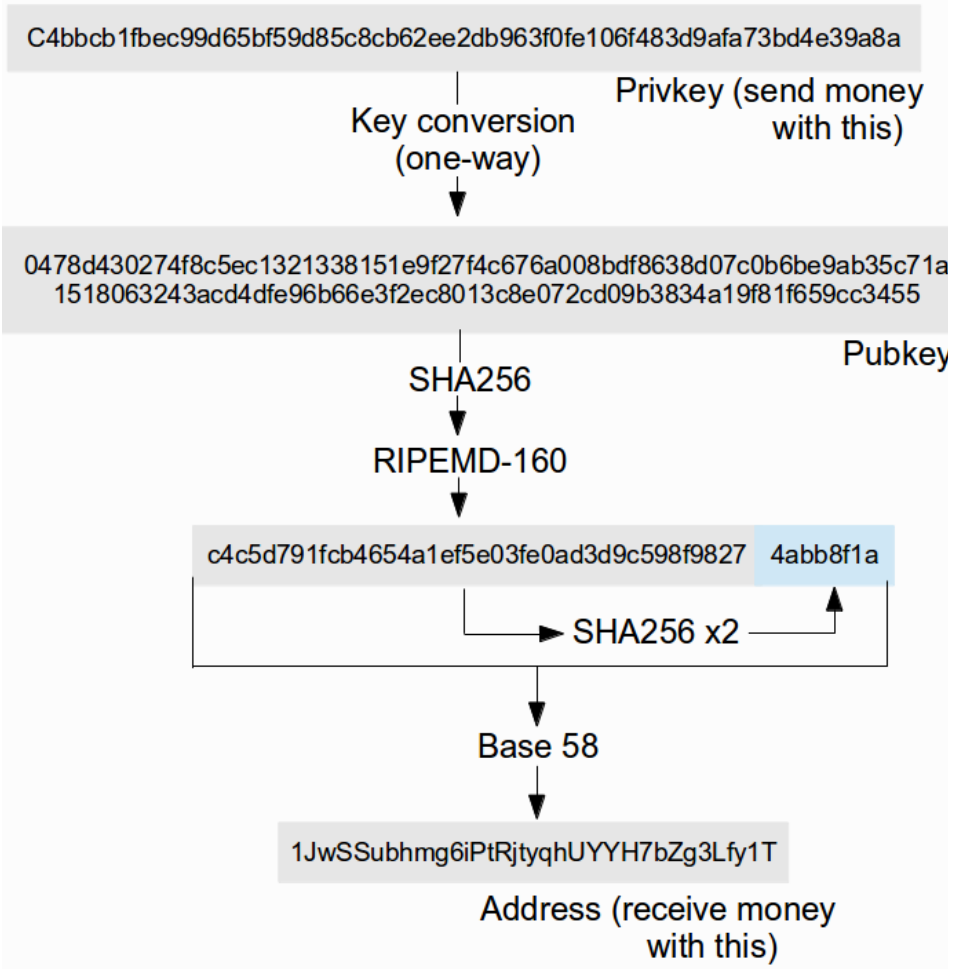
- **Bitcoin is a collection of concepts and technologies that form the basis of a digital money ecosystem, including:**
  - A decentralized peer-to-peer network (enabled by **the Bitcoin protocol**)
  - A public transaction ledger (**the blockchain**)
  - A decentralized mathematical and deterministic currency issuance mechanism (distributed **mining** and the **"Proof-of-Work"** concept)
  - A decentralized transaction verification system (transaction **script**)

(From "Mastering Bitcoin")

- **The Bitcoin system is based on decentralized trust, thus it heavily relies on cryptographic technologies, such as:**
  - Cryptographic hash functions (i.e. SHA-256 and RIPEMD-160)
  - Public Key Cryptography (i.e. ECDSA – the Elliptic Curve Digital Signature Algorithm)

Bitcoin and Cryptography

- In Bitcoin, a **transaction** is a record informing the network of a transfer of bitcoins from one owner to another owner.
  - *You may think of a transaction as the equivalent of a single line in a notebook page*
  - *You may think of a block as the equivalent of a page on that notebook*
  - *You may think of blockchain as the equivalent of the whole notebook*
  - *All the users are able to read, write and get updated on that notebook*
- Ownership of bitcoins is established through digital keys, Bitcoin addresses, and digital signatures.
- **Digital keys** are created and stored offline and consist of a mathematically-related Private-Public key-pair, created using the **E**lliptic **C**urve **D**igital **S**ignature **A**lgorithm (**ECDSA**).

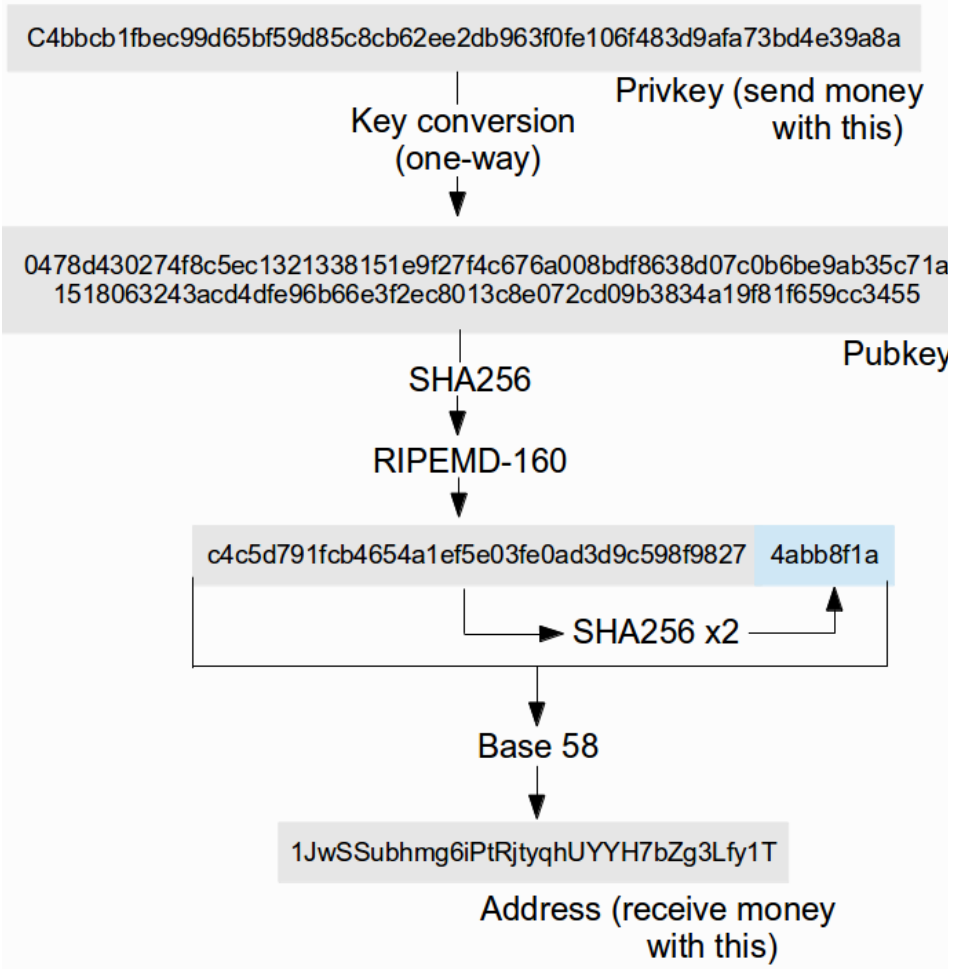


source

Bitcoin and Cryptography

- The **Private key (Privkey)** is initially generated at random, and is kept secret at all times. It is used by the current owner of bitcoins to digitally sign a Bitcoin transaction, when he authorizes the transfer to the new owner. A transaction's **digital signature** confirms ownership, and can be used to verify that the transaction is authentic.

- The **Public key (Pubkey)** is generated from the Private Key using a one-way cryptographic hash function. It is used by the new owner to validate a transaction's digital signature.



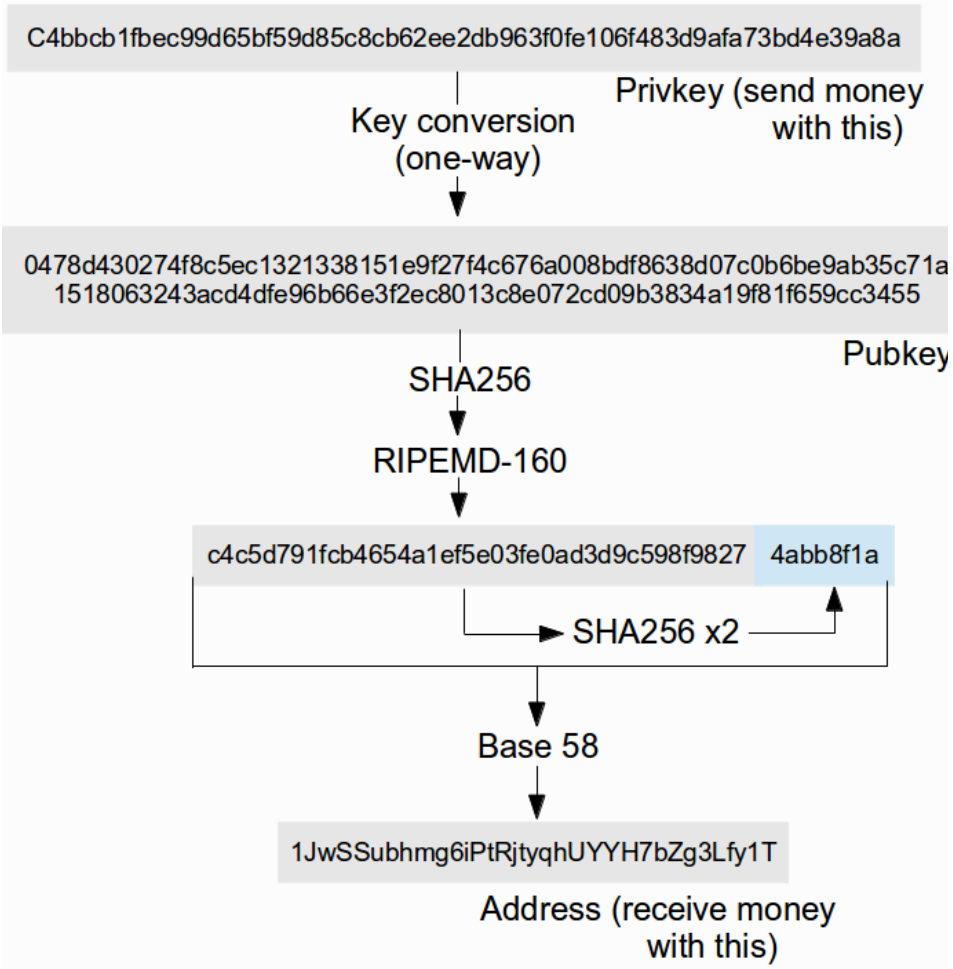
source

Bitcoin and Cryptography

- A **Bitcoin address**, which is a participant's unique identifier on the Bitcoin network, is usually generated by applying the SHA-256 and RIPEMD-160 cryptographic hash functions (discussed later), in series, on the Public key.

- Finally, Bitcoin addresses are encoded using Base58 encoding, which represents an address in a human-readable form of 58 alphanumeric characters.

Fun fact: While there are 62 characters if we take all small and capital letters and numbers, Satoshi wanted to avoid confusion in Bitcoin addresses over commonly mistaken characters, so he removed 4 of them: O0lI – which is which ?!?



source

Bitcoin and Cryptography

- When transactions are broadcasted over the network, the SHA-256 hash function is used to verify data integrity (i.e. to establish that data was not corrupted or modified during transmission).

- All Bitcoin transactions are stored in blocks, which are linked (or “chained”) together in sequence to form the blockchain. Cryptographic hash functions are generally used to:
  - verify block integrity, and
  - establish the chronological order of the blockchain

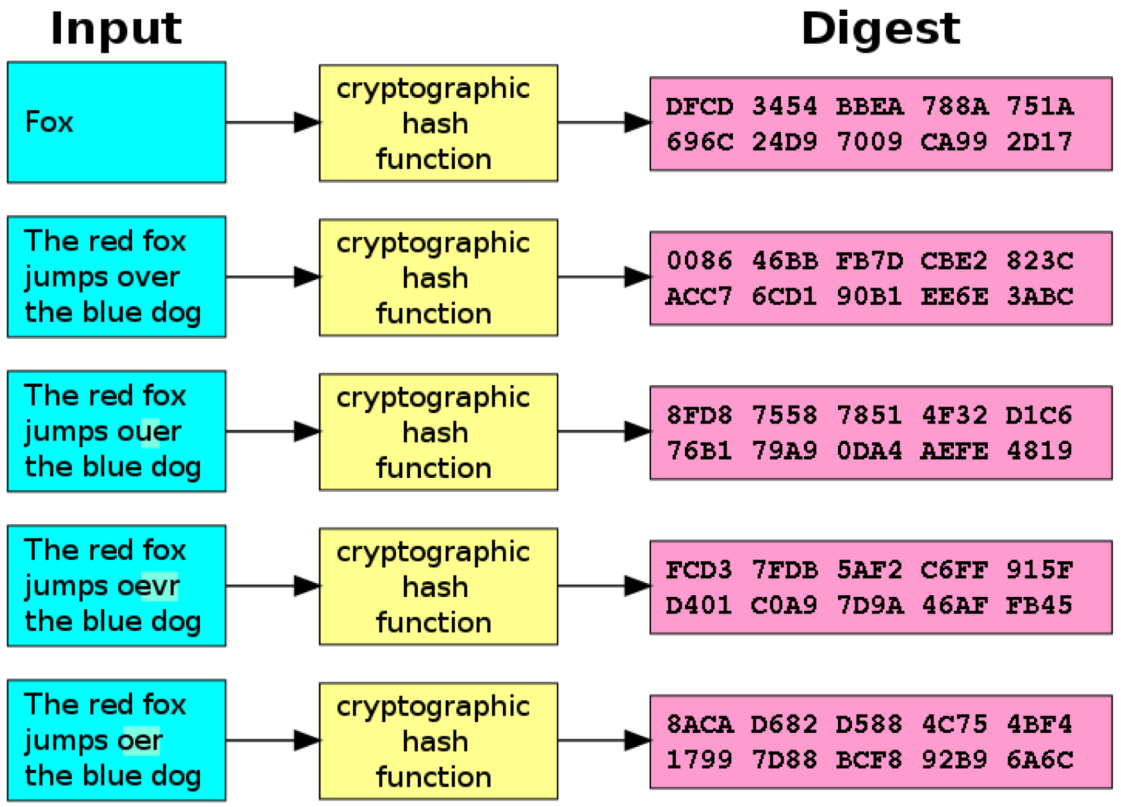
- Furthermore, hash functions are used as part of the ***Proof-of-Work (PoW)*** *algorithm*, which is a prominent part of the Bitcoin mining algorithm (discussed later in this session).

The following slides will explain the *basics* of cryptographic hash functions and public key cryptography, as used by Bitcoin.

Hash functions

- A cryptographic **hash function** is a mathematical function commonly used to verify the integrity of data, by transforming identical data to a unique, representative, fixed-size code. See the diagram for better understanding.

- Any accidental or intentional modification to the data (such as rearrange of letters) will completely change the hash code as displayed under *Digest*



Source

Hash functions

- Bitcoin uses the SHA-256 hash function, where the hash code is 256 bits (or 32 bytes) long

- The SHA-256 hash is usually presented as a string of 64 hexadecimal characters (i.e. each one of the 32 bytes is represented by 2 hexadecimal characters)

For example, the word "*Bitcoin*" produces the SHA-256 hash shown in the screenshot below (generated using the *sha256sum* Linux command).

```
# sha256sum
Bitcoin
b4056df6691f8dc72e56302ddad345d65fead3ead9299609a826e2344eb63aa4
```

Public Key Cryptography

- In Bitcoin, all transaction information is publicly visible to everyone in the network, and transactions are not encrypted. However, Bitcoin heavily relies on digital signatures (one of the main uses of Public Key Cryptography) to verify transactions in the network.

- In Public Key Cryptography, two keys are used:

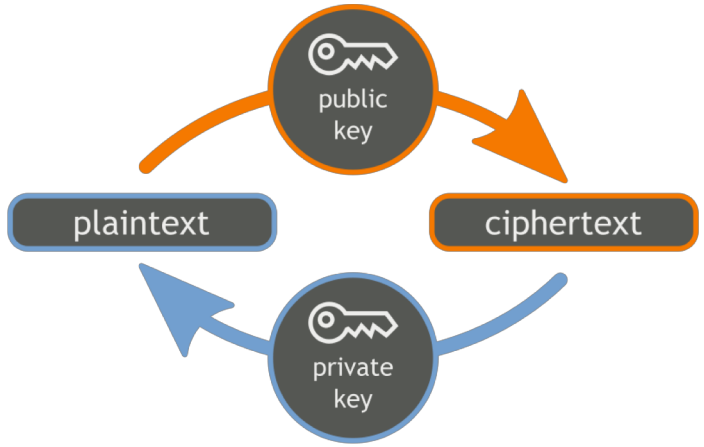
- K_{priv} the **Private key** which must be always kept secret by the owner, and
- K_{pub} the **Public key** which is visible by everyone

Public Key Cryptography

- The sender encrypts the message **M** using the recipient's public key: C = encrypt(M, K_{pub})
- The recipient decrypts the encrypted message **C** using his own private key: M = decrypt(C, K_{priv})

Where: **C** is the result of encryption (also known as "ciphertext"), and **M** is the unencrypted/ decrypted message (also known as "plaintext").

- There is an asymmetric mathematical relationship between the public and private keys:
  - The public key can be easily derived from the private key
  - The private key is nearly impossible (or computationally infeasible) to derive from the public key



Source: Wikimedia Commons

Transactions and the Blockchain

From Digital Signatures to Bitcoin

- We saw how Digital Signatures, a crucial part of many systems involving digital transactions, can be used to convey private information using Public Key Cryptography

- When we need to transfer a digital title of ownership without a central authority, we need a ledger that records these changes in ownership, so that these changes cannot be refuted or altered by malicious activity

- Next, **we will see how Digital Signatures and hash functions are used to form the blockchain** of the Bitcoin protocol, i.e. its distributed ledger of transactions.

P2P Network and Ownership

- Bitcoin is run over a peer-to-peer (P2P) network of computers, called **nodes**
- Nodes are responsible for processing transactions and maintaining all records of ownership
- Anyone can download the free open-source Bitcoin software and become a node
- All nodes are treated equally, and **no single node is trusted**. However, **the system is based on the assumption that the majority of computing power (i.e. at least 51%) will come from honest nodes**
- Ownership records are replicated on every node
- Bitcoin users possess digital keys that allow control over bitcoins recorded in a public ledger *(the blockchain)*
- The public ledger records transactions transferring ownership of a quantity of bitcoins from one owner to the another, like a double-entry bookkeeping ledger

Addresses

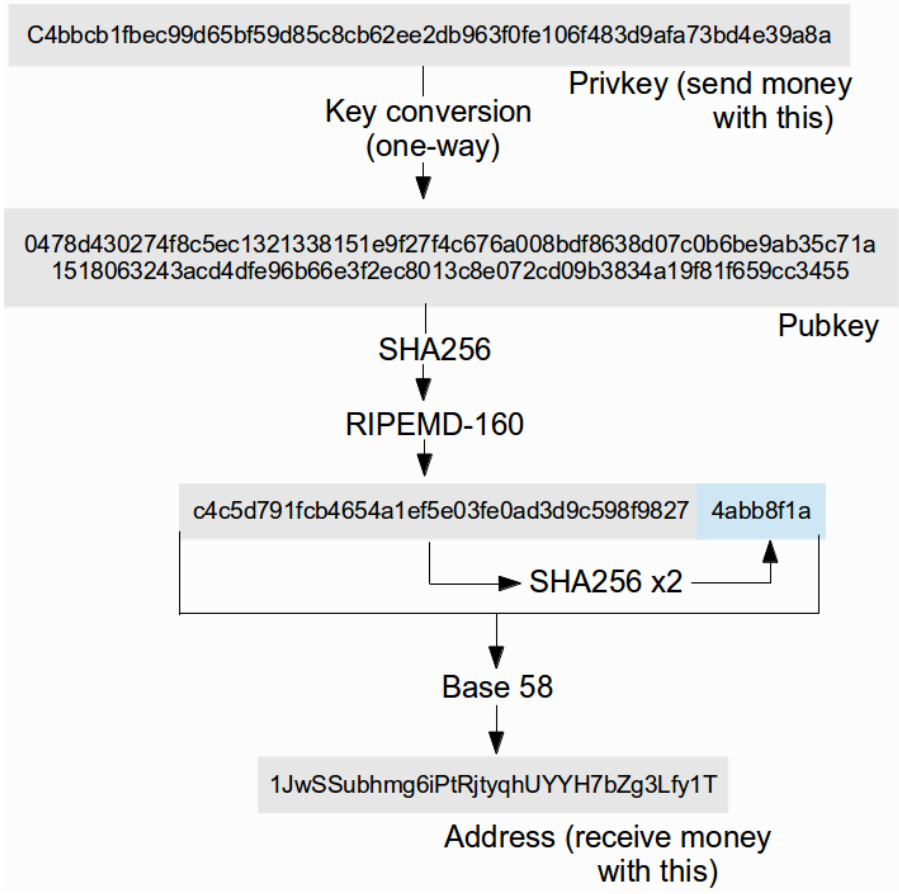
- Transactions in the blockchain do not record the public keys or recipients, but instead use an abstraction called a “Bitcoin address” to record the beneficiary of each amount, allowing for greater flexibility
- To create a Bitcoin address, the Bitcoin client software first generates an ECDSA Public-Private key-pair from a random number
- The Bitcoin address is then generated by applying the following algorithm, in order:

```
version  = (1 byte version number)
keyHash  = RIPEMD-160(SHA-256(publicKey))
data     = version + keyHash
dataHash = SHA-256(SHA-256(data))
checksum = (first 4 bytes of dataHash)
address  = Base58Encode(data + checksum)
```

Addresses

As noted earlier, a Bitcoin address is a computation based on the user's Public key:

- The ***keyHash*** is produced by applying the SHA-256 and RIPEMD-160 hash functions, in series, on the *Pubkey*
- ***Data*** is a concatenation of **keyHash** and an *address* ***version*** *number*
- The ***dataHash*** is produced by applying the SHA-256 algorithm twice on ***Data***
- However, only the first 4 bytes of the ***dataHash*** are used as a **checksum**
- The bitcoin ***address*** is a concatenation of ***Data*** and ***checksum*** encoded in Base58 encoding.
- ***Base58Encode*** is a function that encodes binary as text using the Base58 encoding.



source

Transactions

- A Bitcoin **transaction** tells the network that the owner of a number of bitcoins has authorized the transfer of some of these bitcoins to another owner
- The new owner can now spend these bitcoins by creating another transaction that authorizes transfer to another owner, and so on, in a chain of ownership
- Transactions are like lines in a double-entry bookkeeping ledger. Each transaction contains one or more **inputs**, which are debits against a Bitcoin account
- On the other side of the transaction, there are one or more **outputs**, which are credits added to a Bitcoin account
- The inputs and outputs (debits and credits) do not necessarily add up to the same amount. Instead, outputs add up to slightly less than inputs and the difference represents an implied **transaction fee**, a small payment collected by the miner who includes the transaction in the ledger
- The transaction contains proof of ownership for an amount of bitcoins (inputs) whose value is transferred, in the form of a digital signature from the owner, that can be independently validated by anyone in the Bitcoin network

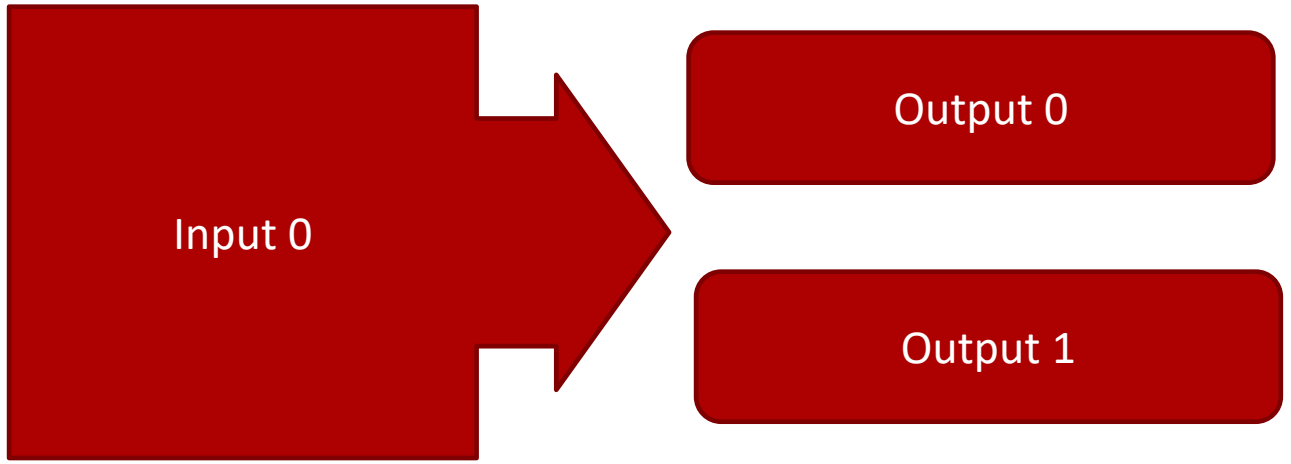
Transactions

We can see that outputs of one transaction are inputs for the next transaction

(From "Mastering Bitcoin")

Transactions

- The most common form of transaction is a simple payment from one Bitcoin address to another, which often includes some “change” to be returned to the original owner. This type of transaction has one input and two outputs and is shown below:



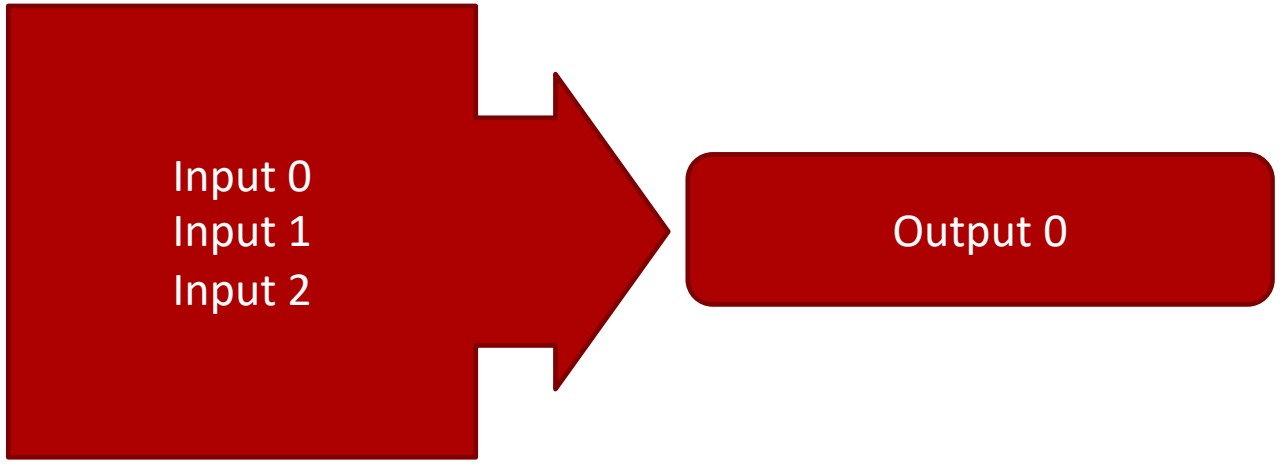
Transactions

- Real example of this type of transaction:

- You're welcome to use any "block explorer" to search through the whole blockchain, since all transactions are publicly visible. Here's list you can choose from : http://bit.ly/blockexplorers
- Pick one and follow any transaction in there, and see what you can find about the inputs, outpus, change and how it's connected with previous ones

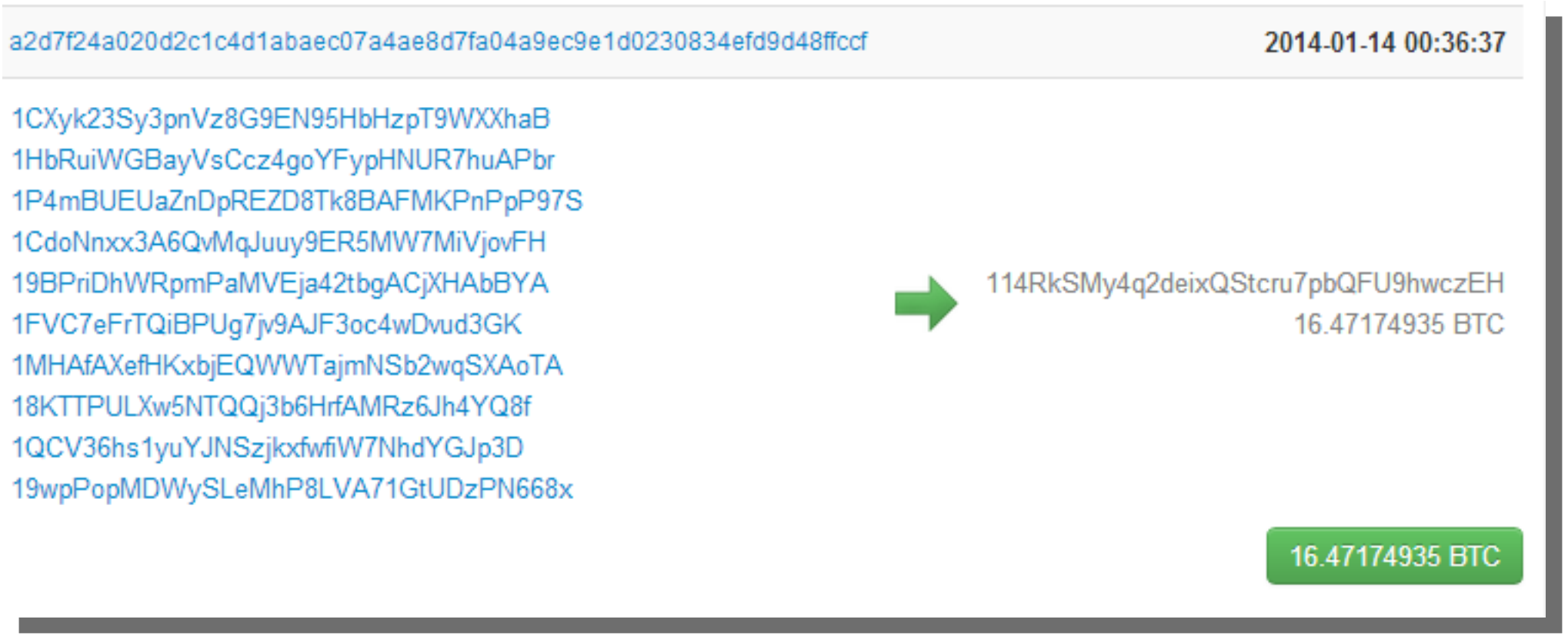
Transactions

- Another common form of transaction is a transaction that aggregates several inputs into a single output. This represents the real-world equivalent of exchanging a pile of coins and currency notes for a single larger note. Transactions like these are sometimes generated by wallet applications to cleanup lots of smaller amounts that were received as change for payments.



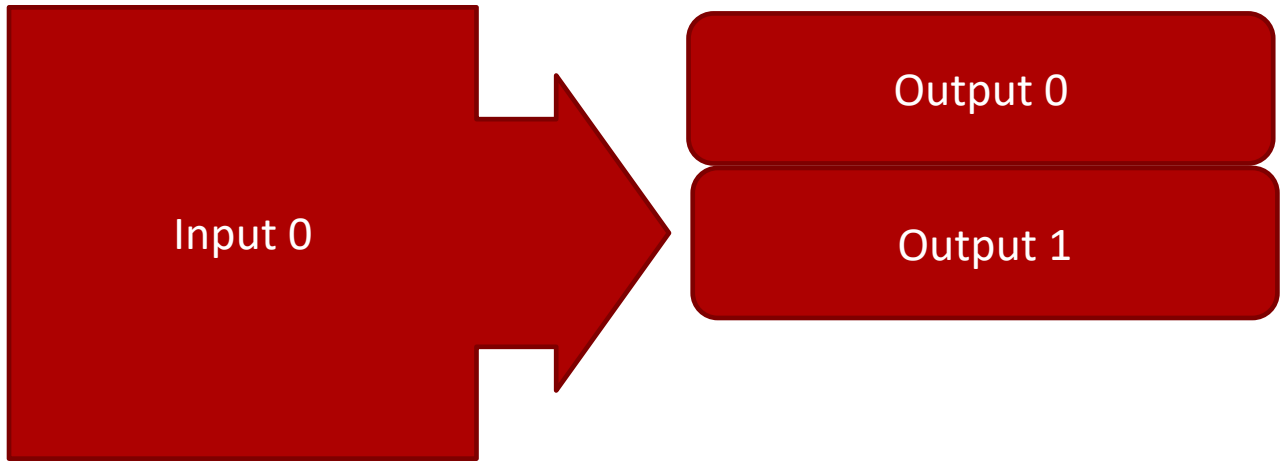
Transactions

- Real example of this type of transaction:



Transactions

- Finally, another transaction form that is often observed in the Bitcoin ledger is a transaction that distributes one input to multiple outputs representing multiple recipients. This type of transaction is sometimes used by commercial entities to distribute funds, such as when processing payroll payments to multiple employees.

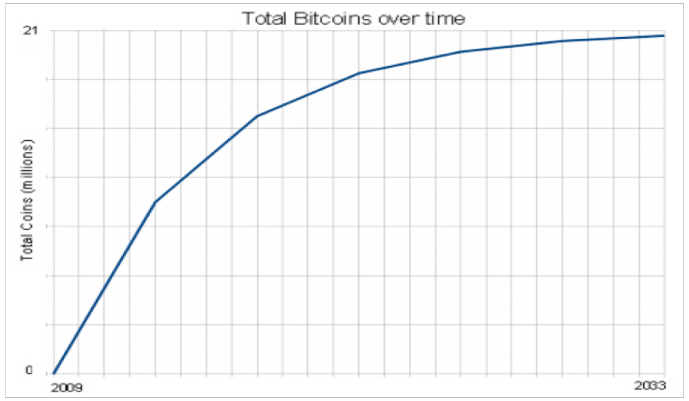


Mining

Mining

- The Bitcoin system of trust is based on computation. **Transactions** are bundled into **blocks**, which require an enormous amount of computation to “prove” (or “confirm”), but only a small amount of computation to verify as “proven”, in a process called **mining**

- **Mining creates new bitcoins in each block**, almost like a central bank printing new money. The amount of bitcoins to be created is fixed and diminishes with time
- **Mining creates trust** by ensuring that transactions are confirmed only when enough computational power was devoted to the block that contains them. More blocks mean more computation, which means more trust.



Bitcoin production over time

Mining algorithm

- Mining consists of the following steps, which are performed in a continuous loop:
  - **Bundling transactions** that were broadcast on the peer-to-peer network into a **block**. Each miner can arbitrarily decide which transactions to include in their block
  - **Verifying** that all **transactions** in the block are **valid**
  - **Selecting the most recent block** on the longest path in the blockchain and **inserting a hash of its header into the new block**
  - **Trying to solve the Proof of Work (PoW) problem for the new block** and simultaneously watching for new blocks coming from other nodes

- If a solution is found to the Proof-of-Work problem, the new block is added to the local blockchain and broadcast to the peer-to-peer network

Proof of Work

- Miners search for acceptable blocks using the following procedure, performed in a loop:
  - Increment (add 1 to) an arbitrary number in the block header called a **nonce**
  - Take the hash of the resulting block header
  - Check if the hash of the block header, when expressed as a number, is less than a predetermined target value
- If the hash of the block header is not less than the target value, the block will be rejected by the network. Finding a block that has a sufficiently small hash value is the PoW problem.
- **Mining performance, therefore, is measured in hashes/sec**. Currently the performance of miners is measured in GH/s (billions of hashes per second) or TH/s (trillions of hashes per second)

H/ s = Hashes per second
KH/ s = Kilo Hashes per second
MH/ s = Mega Hashes per second
GH/ s = Giga Hashes per second
TH/ s = Tera Hashes per second
PH/ s = Peta Hashes per second

1,000 H/ s = 1 KH/ s
1,000 KH/ s = 1 MH/ s
1,000 MH/ s = 1 GH/ s
1,000 GH/ s = 1 TH/ s
1,000 TH/ s = 1 PH/ s

Mining Difficulty

- Bitcoin nodes that mine, actively regulate the rate of creation of new blocks
- As more miners join, the rate of block creation will go up. As the rate of block creation goes up, the **mining difficulty** rises to compensate, which pushes the rate of block creation back down
- **The creation of new blocks must take an average of 10 minutes**
  (Ten minutes was specifically chosen by Satoshi Nakamoto as a tradeoff between fast confirmation time and the amount of work wasted due to chain splits and orphan blocks.)
- The regulation is done by periodically adjusting the hash target value for blocks
- Every 2,016 blocks (which ideally spans every 2 weeks, with each block taking 10 minutes to confirm) Bitcoin nodes calculate a new difficulty accordingly, based on the time it took to mine the last 2,016 blocks

Mining Reward

- Solving the Proof of Work problem requires a lot of computing power and that power costs money. To encourage participants to invest their resources in mining, Bitcoin provides a reward in each successfully mined block (**plus the transaction fees** of the transactions contained in the new block)
- When a block is discovered, the discoverer will award themselves a certain number of bitcoins, which is agreed-upon by everyone in the network
  - Currently this bounty is 12.5 bitcoins
  - Based on Bitcoin's algorithm, this bounty halves every 210,000 blocks (i.e. approximately every 4 years)
  - Eventually, the reward will be removed entirely when the limit of 21 million bitcoins is reached asymptotically, by the year 2140
  - After that, transaction processing will be rewarded solely by transaction fees
- As mentioned, the miner is awarded in addition the fees paid by Bitcoin users sending transactions

Other Consensus Mechanisms

- Several consensus mechanisms have evolved with the aim to approach distributed consensus on various ways
- Most notable examples are:
  - Proof-of-Stake
  - Delegated Proof-of-Stake
  - Proof-of-Burn
  - And more…
- Cryptocurrencies with a significant market share such as Peercoin, NEM, NXT rely on some of these consensus protocols. Ethereum is also expected to switch to Proof-of-Stake algorithm in 2018 to address scaling issues

DECENTRALIZED
TRAINING SERIES

Thank you

UNIC | Institute For the Future

CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS | iti Information Technologies Institute