Computer Security

10. Blockchain & Bitcoin

Paul Krzyzanowski

Rutgers University

Fall 2019

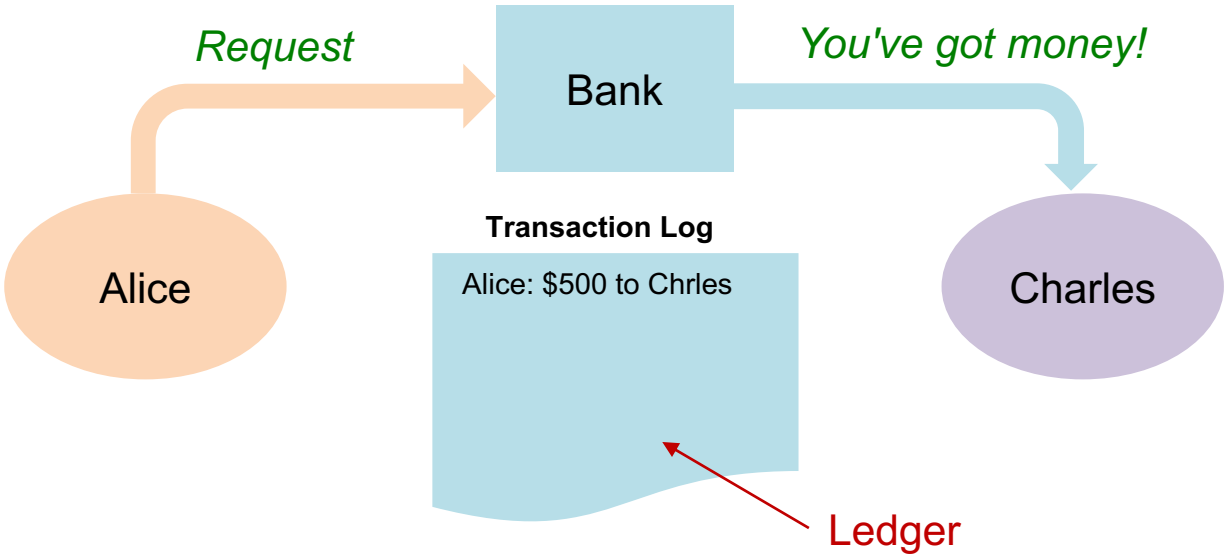
Bitcoin & Blockchain

Bitcoin cryptocurrency system

- Introduced in 2009 – anonymously by Satoshi Nakamoto
- First blockchain
- Designed to be public
  - Anyone can participate in the system & use it
  - Users are anonymous
- Currency that is totally separate from any sovereign government
  - Anyone can create money!

Traditional Payments

- Suppose Alice wants to pay Charles
  - Send a message to the bank: *transfer $500 from Alice to Charles*
- Bank is a trusted third party
  - Owns register of activity & account balances
  - Only the bank can manipulate the data
  - Also – banks control supply of money



Centralized systems

Transactions are simply modifications to the bank's database

- We can simply
  - Subtract $500 from Alice's account
  - Add $500 to Charles' account
- Having a log is just nice for auditing but not necessary

Problems?

This is a centralized system

We trust the bank – it is a trusted third party

- What if the bank disappears?
- What if the banker makes a mistake?
- What if the banker is corrupt?

Decentralized Solution – Bitcoin

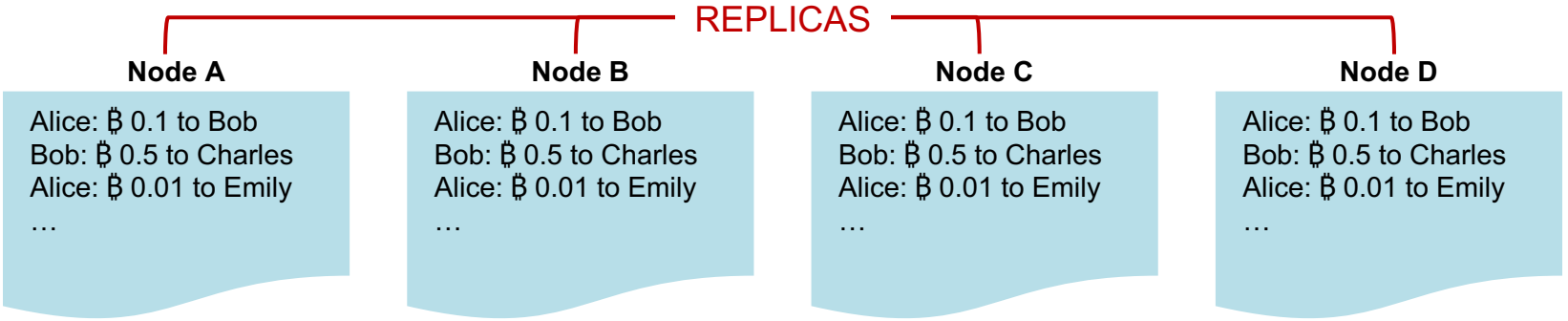
- **Blockchain** = ledger = complete list of ALL transactions
  - Since Bitcoin was started in January 2009
  - 247,433 MB as of November 4 2019

    See https://www.blockchain.com/en/charts/blocks-size for the current size of the ledgers

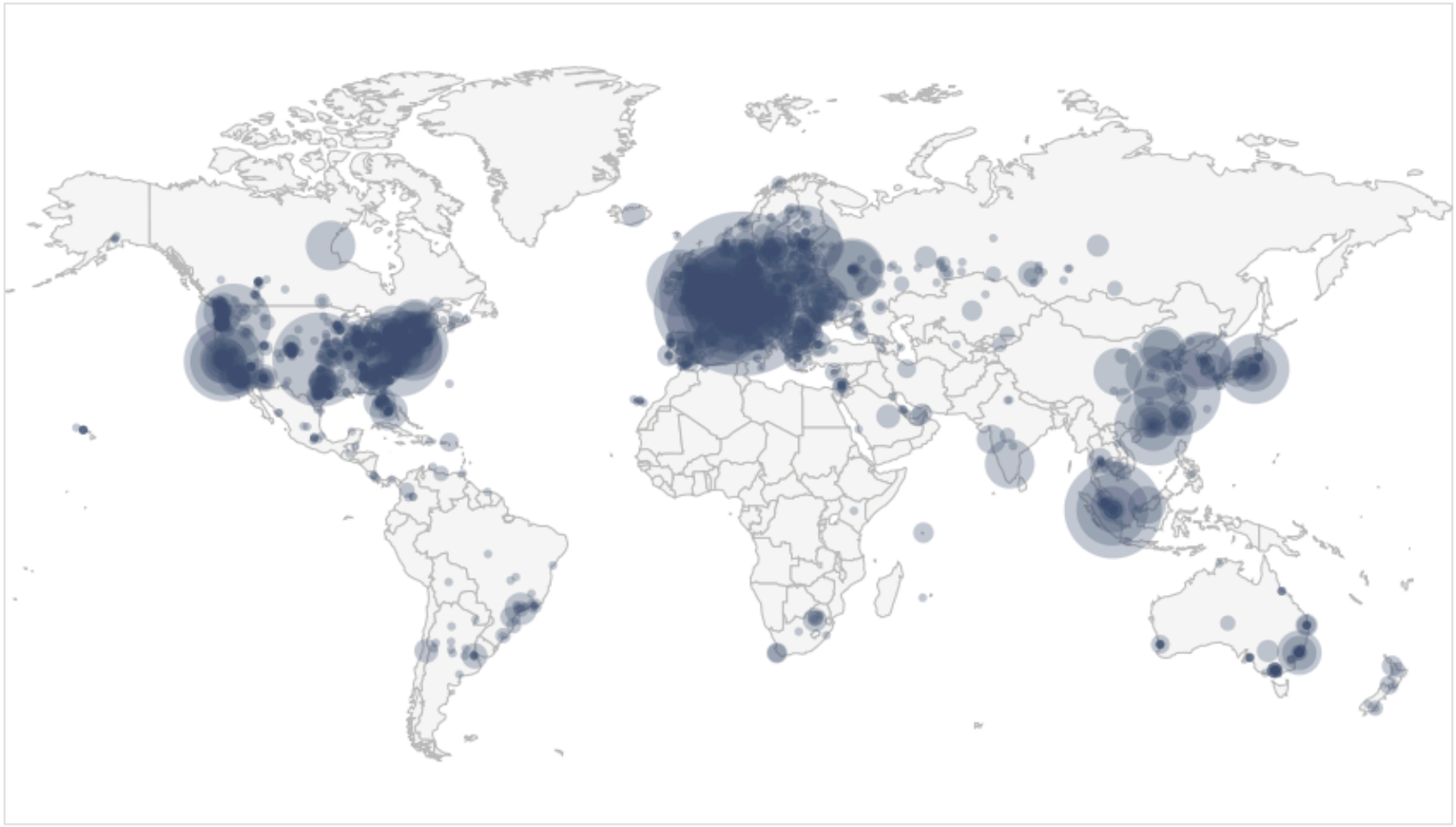
- Complete copies of the ledger are replicated around the world
  - 9,557 nodes (Nov 4, 2019) – all peers – running identical software
  - There is **no master node** or master version

    See https://bitnodes.earn.com

- New systems can do a DNS query for well known peers
  - Names hardcoded in source (DNS seeds)
  - Return list of IP addresses of bitcoin nodes
  - Then use peer discovery process to find others



Global Bitcoin Nodes



November 4 2019 • https://bitnodes.earn.com

Identities

- User creates a {public, private} key pair that defines her **wallet**
  - 256-bit Elliptic Curve Digital Signature Algorithm (ECDSA) used
  - **The wallet is just a place to store these keys**
    - Wallets *may* store a transaction list but that's just for user records – the bitcoin network doesn't care
- **Bitcoins are associated with keys, not users**
  - Users are anonymous
  - A user's ID is the public key – anonymous – no association to name
  - The user's identity is called their **address**
  - Users may have multiple keys & multiple addresses
- **Every transaction is signed with the creator's private key**
  - Transaction identifies the user by the public key and can be verified
  - We know only the person with the corresponding private key could have created the request
- ***Nobody to call if you lose your private key!***

Bitcoin address = hash(public_key)

Bitcoin uses ECDSA: Elliptic Curve Digital Signature Algorithm

A user creates one or more identities = { private, public } key pairs

You can create an identity (address) for each new transaction

How Bitcoin addresses are created*

1. Generate an ECDSA public, private key pair
2. Create a SHA-256 hash of the public key
3. Perform a RIPEMD-160 hash on that
4. Add a version byte in front of the result
5. Perform a SHA-256 hash on the result … and a SHA-256 hash on that
6. First 4 bytes of the result = address checksum
7. Add 4 bytes from [6] tp the end of the RIPEMD-160 hash from [4]
8. Convert the byte to a base-58 string using Base58Check encoding. This produces a 20-byte *address*

*You don't have to know this.

Addresses vs. keys

Spending: Bob wants to send Alice 5 bitcoins

- Bob creates a transaction with a **digital signature** using his private key
- Presents his public key along with the transaction
- Any receiving node can validate that the transaction was signed by someone with the corresponding private key
- The destination of the money is Alice's **address**

Addresses are ***not*** accounts

- They only receive funds
- You can use those funds if you prove you know the private key that corresponds to the address
- If Alice wants to use the coins she received
  - She creates a new transaction with her public key & a digital signature
  - Any node can validate that the address belongs to her
  - No node can figure out Alice's public key just by looking at the address

https://learnmeabitcoin.com/guide/public-key-hash160

Transactions: Inputs

If Alice wants to send some bitcoin to Bob

- She creates a **transaction** and sends it to one or more bitcoin nodes
- A node tells its peers about the transaction
- Within ~5 sec. every peer on the network has it
- The transaction is currently **unconfirmed**

A blockchain is NOT a database – it's a list of transactions

- There are no accounts to query
- Alice needs to provide links to previous transactions that will add up to at least the required amount – these are **inputs**

A node verifies inputs

- Make sure they have not been used by another transaction (this would be **double spending**)
- Make sure there is sufficient money in the inputs

Addresses vs. keys – Inputs and Outputs

If Alice wants to use the coins she received:

- She creates a new transaction with her public key & a digital signature
- Any node can validate that the address belongs to her
- No node can figure out Alice's public key just by looking at the address

Transaction 10732

Output: 1PMycacnJaSqwwJqjawXBErnLsZ7RkXUAs *Alice's address*
Amount: ฿ 0.1
…

Transaction 71991

Output: PWJ2sc9aV72kknbi3R9sjcXVcMXpkdh9Le5 Address to whom she's sending the money

Input: Source: 10732 *pointer to transaction where Alice got the money* *Alice's public key*
Public key: 0250863ad64a87ae8a2fe83c1af1a8403cb53f53e486d8511dad8a04887e5b2352
Signature: a3bb7c5f2201079c…. Alice's signature (using her private key)

The transaction can be validated by validating each input:

1. Validate the signature using Alice's public key (in the transaction). This proves that whoever created the signature has the private key corresponding to the public key.
2. Hash the public key in the transaction to create the address – see if it matches the address in the referenced transaction.

https://learnmeabitcoin.com/guide/public-key-hash160

Transactions: Inputs & Outputs

A transaction contains:

1. One or more **inputs**: transaction IDs & address where coin comes from
   - *Contains signature & public key*
   - **An input is a reference to the output from a previous transaction**
2. **Output**: who the money goes to – destination address & amount
3. **Change**: owner's address & amount
   - Every input must be completely spent
   - Any excess **change** can be generated as another output to the owner of the transaction
4. **Transaction fee** (anywhere from 10¢ to a few $ per transaction)
   - There's a limited amount of space (1 MB) in a block. A transaction is about 250 bytes. To get your transaction processed quickly, you need to outbid others.

The amount of bitcoin you own is the set of transactions in the system that are outputs to your address but have NOT been used as inputs in any transaction

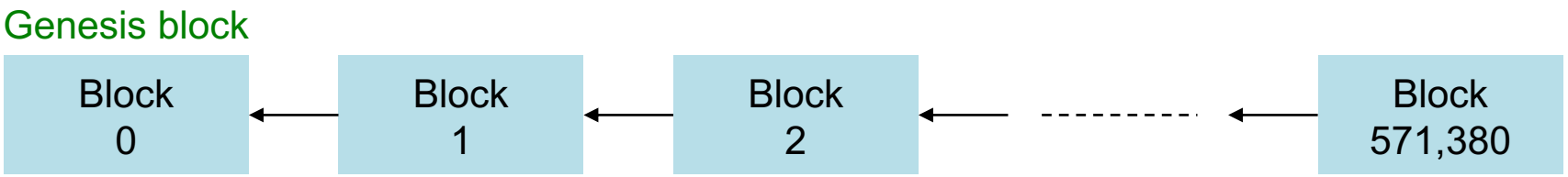
Blocks

Transactions are grouped into blocks

– Each block holds ~4,000 transactions @250 bytes and is ~1 MB in size

Bitcoin ledger = linked list of chronologically-ordered blocks

Approximately every 10 minutes, a new block of transactions is added to the blockchain



Each block has

– A link to the previous block

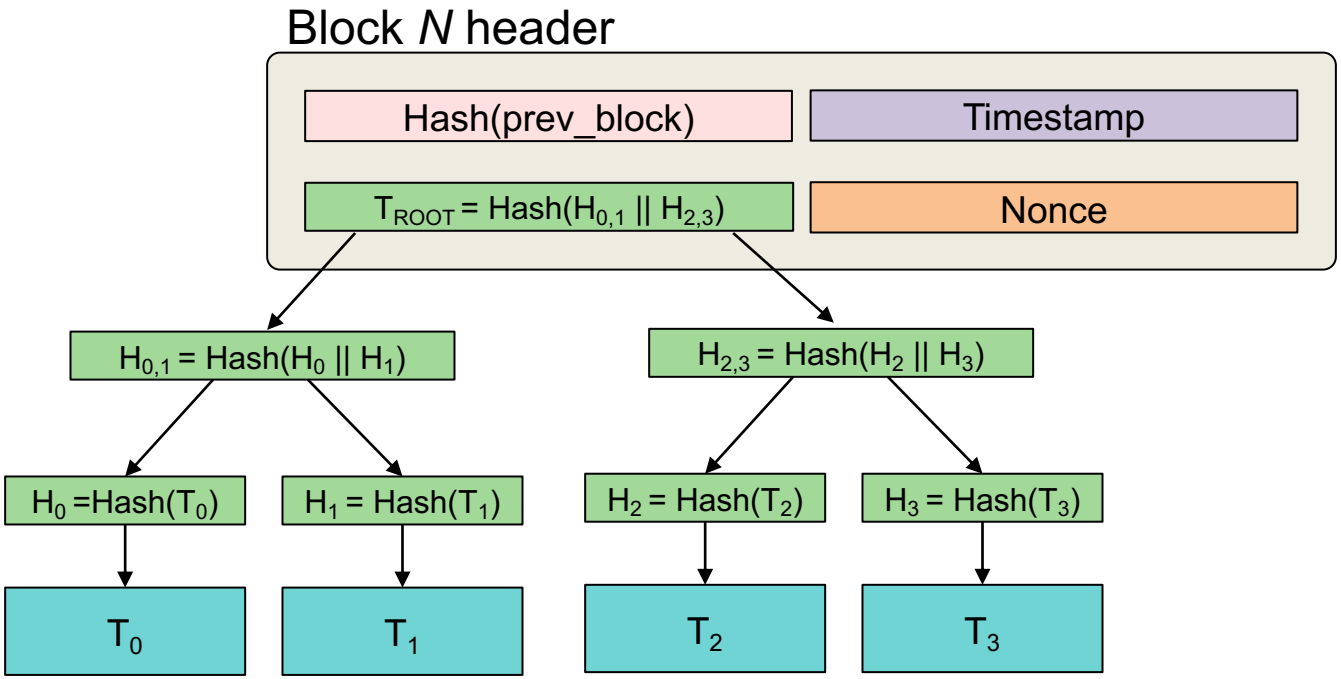
– SHA-256 hash of the previous block

This creates the **blockchain**:
hash pointers – a tamper-evident log

Transactions in blocks: Merkle trees

Transactions *within* a block are stored in a Merkle tree

- Binary tree of hash pointers
- Using a tree makes it easy to find one of thousands of transactions
- A Merkle tree makes it easy to check if the transaction is valid



Agreement & adding blocks

Each node groups transactions into a block & can propose it as the next block in the blockchain

- Transactions can reach nodes in a different order
- We want all nodes to agree on the sequence of blocks in the blockchain

A linked list of hash pointers (blockchain) is a tamper-proof structure

- If the contents of any block are modified, then the hash pointer that points to the block will not be valid (hash won't match)
- But can't anyone change the hash pointers?
  - We might want to use signed pointers but there's no central authority (no trusted party) so that won't work

Let's create a system where
(a) Everyone can agree on the sequence of blocks
(b) That sequence cannot be modified

To add a block to the chain, the hash of the block must meet a certain requirement

Let's make in challenging: create a puzzle

Suppose we want a hash output with a specific property:

– Example: the hash should start with with "0000"?

There is no algorithmic way to do this

Must try lots of variations of the input

But once found – it is easy for anyone to verify that the data hashes to the result

– Just hash the data and see if the hash starts with “0000”

Mining

Solving this "puzzle" is called **mining**

- A block has a 32-bit field in the block where we can try different numbers
- Try to get the block to hash to a desired output
- The resulting number is called the **Proof of Work**
  - *Difficult to generate but easy to verify*

We demonstrate that work has been put into figuring out what the value should be to create the desired hash

Everyone in the network can participate in this

- The first system that finds it **announces the block to everyone else** in the network
- Upon receiving an announcement
  - Each system validates the Proof of Work number against the block
  - A majority of systems must grant approval
  - If they do, the block (with the Proof of Work) is made part of the **blockchain**

What's the puzzle?

- Bitcoin uses a version of **hashcash** (created in 1997)
  - Hashcash searches for a SHA-256 *hash(message, random #, N)* where the leading *k* bits are 0
    - *Random #* – Starting value to make it unlikely that two systems start their search at the same point
    - *N* – the nonce: the number we vary until we get the hash we need
    - Choice of *k* sets the difficulty of the problem
- Ensure that one node doesn't take credit for another's work
  - 256-bit SHA-1 hash of
    - ***B***, transaction block, which includes hash pointer to previous block
    - ***A***, recipient's reward address (public key of who gets credit)
    - ***N*** – nonce: the number we vary until we get the hash we need
- Bitcoin uses a floating-point *k* to scale the work more precisely

  $hash(B, A, N) < 2^{n-k}$
- Currently, the first 74 of the 256-bit hash must be 0

Isn’t a 32-bit nonce too small?

A 32-bit nonce field might not be sufficient

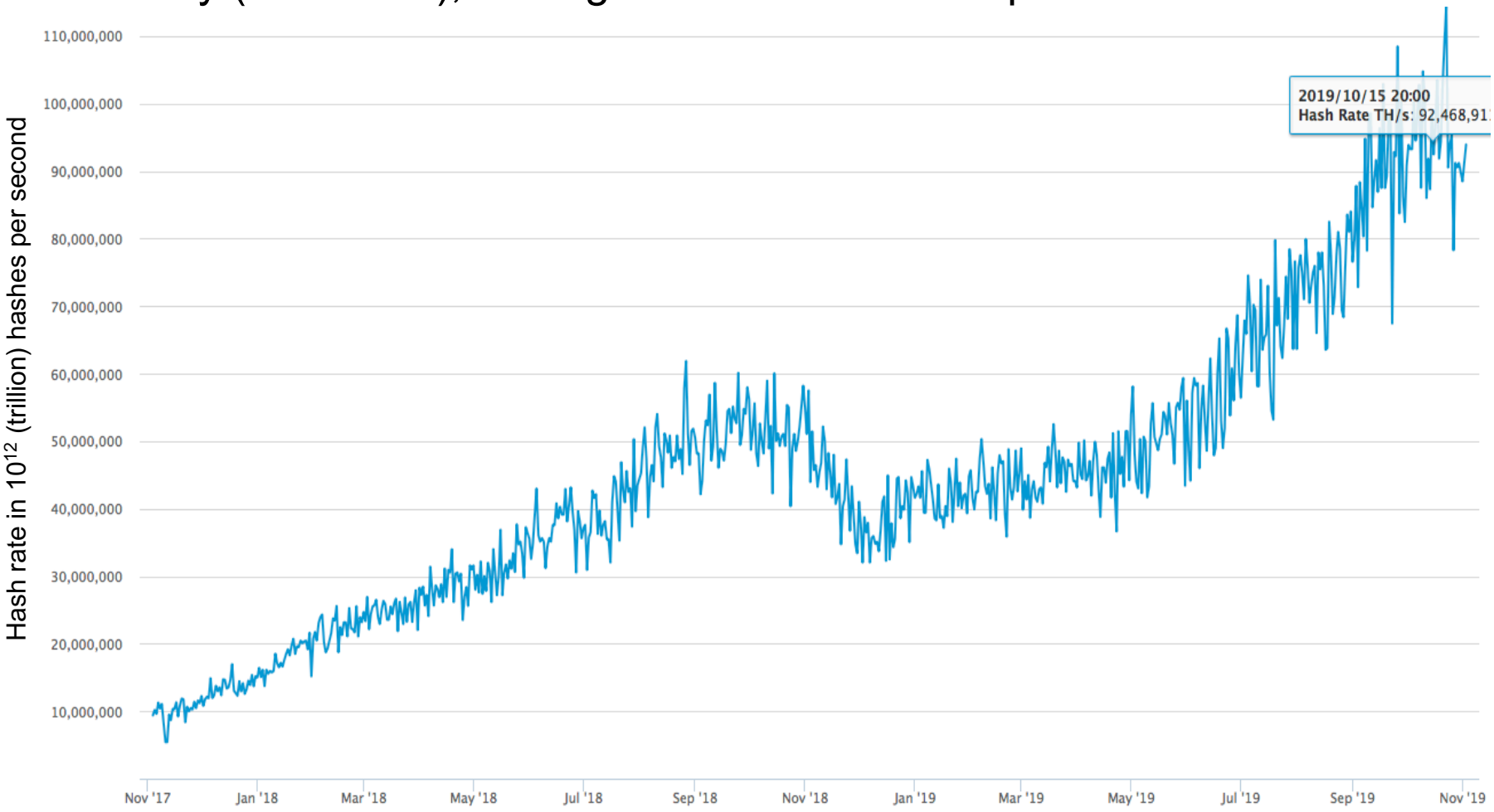
- There might be NO value of the nonce that will produce the right hash

The node then needs to modify other data in the block header & try again

- Make small changes to the timestamp
- New transactions may be added
- If nonce overflows, increment `extraNonce` & reset nonce
  - `extraNonce` is 2-100 bytes
  - Changing `extraNonce` alters the Merkle root hash
    - That needs to be recomputed

How much work is going on?

Currently (Nov 2019), average 93.9×10^{18} hashes per second



See https://www.blockchain.com/charts/hash-rate

Bitcoin mining

Computing the hash = mining

If you come up with the right answer, you win!

1. You send your block, with the nonce in it, to the whole network
2. Others validate it
3. When a computer validates your block, it adds it to its ledger
4. You get a reward for solving the puzzle! Currently 12.5 BTC
5. You also get paid transaction fees that were are to the transactions you put into the block
6. The block (not transactions!) is **confirmed** and you get paid
7. Individual transactions may require the confirmation of multiple subsequent blocks ... More on this later….

Bitcoin mining

The more hashes you can try, the better your chances of winning

- You're competing with every other miner
- People moved from CPU-based mining to GPU-based
  - GPU power approximately = 30 CPUs
  - Then FPGA mining: approximately 3-100x faster than GPUs
  - ASIC mining (application-specific integrated circuit):
    - Special hardware built for hash computation: faster & more power efficient
- Mining pools = group miners together & share rewards
  - There are over a dozen large pools for Bitcoin

Mining Hardware

CPU → GPU → FPGA → ASIC

Example:
EBIT E9.3
By Ebang Communication

$950

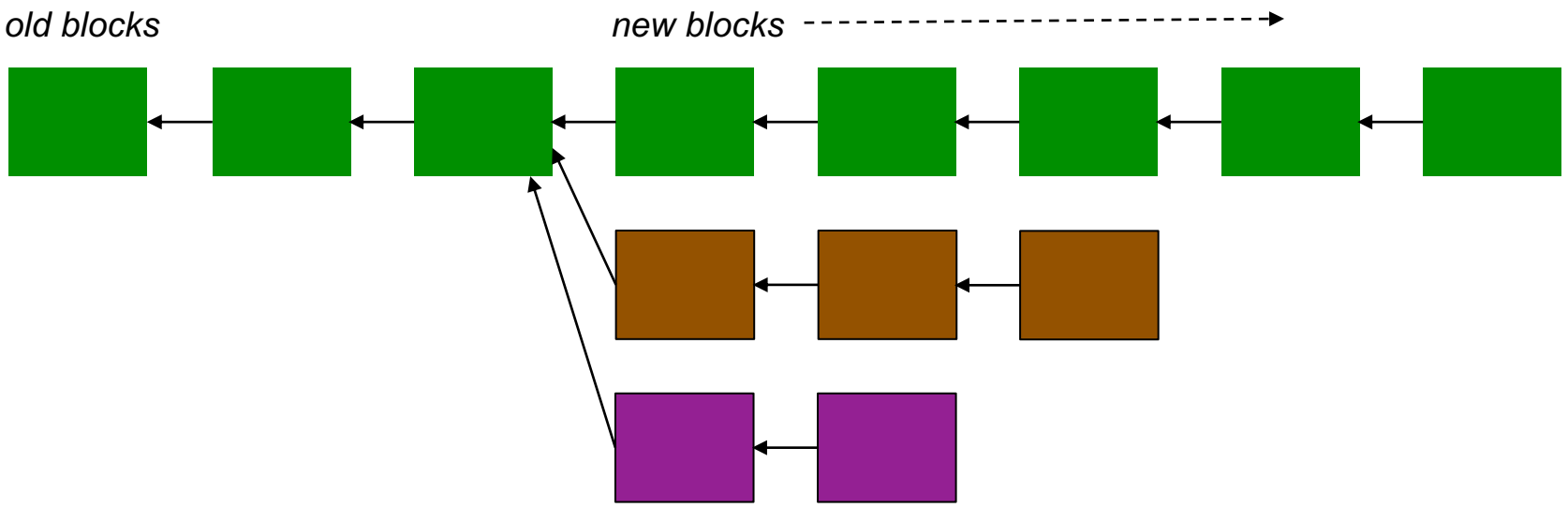
Computes SHA-256 hashes at 16 TH/s

Payback period ~ 393 days

Competing chains

What if a malicious participant wants to modify an old transaction?

- It will need to modify an old block
- And recompute the Proof of Work (which takes a lot of effort) for the block and each successive block (tons of work)
- **The participant will be creating another chain in the blockchain**



Competing chains

BUT:

- One malicious participant will not be able to catch up with the cumulative work of all the others
- It is expected that some nodes will occasionally have different versions
- **Length of chain** = **score**

If we observe two versions of the blockchain, we select the one that was the hardest to generate (= longest chain)

Blockchain rules state that

The longest chain in the network is the correct one

If a participant receives a higher-scoring version, it overwrites its blockchain with the better data & transmits updates to peers

Producing a longer ledger than the current one requires computing power that competes with the rest of the entire network

51% Attack

If the majority of participants decide to cheat, the protocol will fail

Blockchain works only because of the assumption that the *majority* of participants are honest

To double-spend a bitcoin:

- You would need to rewrite the blockchain (change past transactions)
- An attacker would need to control more than 50% of computing capacity
  - **This is a lot**: as of 12/17, The Economist estimates "*bitcoin miners now have 13,000 times more combined number- crunching power than the world's 500 biggest supercomputers*"
  - Even if someone tried to do this attack, they'd likely only modify transactions in the past few blocks
- Keeping history of all transactions among all participants allows anyone to check for double spending

Confirming transactions

A transaction is ***confirmed*** after *N* number of additional blocks are added to the blockchain

– Large values of *N* are recommended for high-value transactions

The more blocks are added after a transaction, the more difficult it is to modify it

Bitcoin Confirmation Recommendations

1: Small payments <$1,000
3: Deposits and payments of $1,000-$10,000
6: Large payments $10k-$1M
60: Payments >$1M

Higher values of *N* mean that an attacker will need to recompute *N+1* Proof of Work values to modify the blockchain

– Computationally not feasible

https://www.buybitcoinworldwide.com/confirmations/

Incentives

Computing the Proof of Work takes a lot of work – *why do it?*

To get paid with bitcoin:

- First participant to compute the Proof of Work gets rewarded with bitcoin
- BUT … only after another 99 blocks have been added to the ledger
- This gives miners an incentive to participate & validate transactions

Reward is decreasing (*assumption: bitcoins will be more valuable*)

- 50 bitcoins for the first 4 years since 2008
- 25 bitcoins from 2012-2015
- 12.5 bitcoins from block #420,000 July 9, 2016 – 2019
- 6.25 bitcoins at block #630,000 – around May 24, 2020

Eventually there will be a maximum of ~21 million bitcoins

There are also transaction fees even if the block reward = 0

Centralization

- Anyone can run a bitcoin node
  - Requires a good chunk of disk space but is accessible
  - Highly decentralized
- Mining
  - Anyone can mine but requires a lot of computing power
  - Not as decentralized as we'd like
- Software development/support
  - Open but there's a core set of trusted developers – not really decentralized
  - Bugs may be fixed … but transactions cannot be undone
- In theory
  - Teams of sneaky developers may be able to mount an attack
  - Mining pools may try to mount a 51% attack
  - Both scenarios highly unlikely today

51% attack: difficult, not impossible

MIT Technology Review

Once hailed as unhackable, blockchains are now getting hacked

More and more security holes are appearing in cryptocurrency and smart contract platforms, and some are fundamental to the way they were built.

By Mike Orcutt February 19, 2019

Early last month, the security team at Coinbase noticed something strange going on in Ethereum Classic, one of the cryptocurrencies people can buy and sell using Coinbase's popular exchange platform. Its blockchain, the history of all its transactions, was under attack.

An attacker had somehow gained control of more than half of the network's computing power and was using it to rewrite the transaction history. That made it possible to spend the same cryptocurrency more than once—known as "double spends." The attacker was spotted pulling this off to the tune of $1.1 million.

https://www.technologyreview.com/s/612974/once-hailed-as-unhackable-blockchains-are-now-getting-hacked/

Two Miners Purportedly Execute 51% Attack on Bitcoin Cash Blockchain

Max Boddy May 25, 2019

Two miners have reportedly executed a 51% attack on the bitcoin cash (BCH) blockchain, according to tweets by Cryptoconomy Podcast host Guy Swann on May 24.

A 51% attack occurs when someone controls the majority of mining power on a Proof-of-Work blockchain network. This means that the majority block verifier can prevent other users from mining and reverse transactions.

While many have assumed that a 51% attack would be carried out with malicious intent, the above case happened as the two mining pools attempted to prevent an unidentified party from taking some coins that — due to a code update — were essentially “up for grabs.”

https://cointelegraph.com/news/two-miners-purportedly-execute-51-attack-on-bitcoin-cash-blockchain

Achieving anonymity is difficult

CNN BUSINESS Markets Tech Media Success Perspectives Video

How Mueller used Bitcoin to catch Russia

By Donie O'Sullivan, CNN Business

Updated 6:17 PM ET, Fri April 19, 2019

The blockchain contains no personally identifiable information.

Once someone figures out a user is responsible for one transaction, they can track the entire Bitcoin history.

New York (CNN Business) – Russian operatives used cryptocurrency at almost every stage in their online efforts to interfere in the 2016 U.S. presidential election, according to Special Counsel Robert Mueller's final report on his investigation.

Systems used in the hacking of the Democratic Party were paid for using Bitcoin, as were online hosting services that supported websites which published hacked materials and were used in the targeting of disinformation at American voters. The hacking and disinformation campaigns accounted for the vast majority of Russia's online efforts to influence the 2016 election.

All Bitcoin transactions are posted to an immutable public ledger, known as a blockchain. While the blockchain doesn't contain obvious identifying information about the person behind a transaction, once someone figures out a user is responsible for one transaction it can be possible to track their entire Bitcoin history.

https://www.cnn.com/2019/04/19/tech/bitcoin-mueller-russia/index.html

A single anonymous market manipulator caused bitcoin to top $20,000 two years ago, study shows

Michael Sheetz
November 4, 2019

One of the SEC's top worries is that crypto is subject to manipulation

A forensic study found that tethers, a digital currency, being traded for bitcoins, revealed a pattern of manipulation during the 2017 cryprocurrency boom.

"Almost the entire price impact can be attributed to this one large player," finance professors John Griffin and Amin Shams wrote.

A forensic study on bitcoin's 2017 boom has found that nearly the entire rise of the digital currency at the time is attributable to "one large player," although the market manipulator remains unidentified.

Finance professors John Griffin and Amin Shams – instructors at University of Texas and the Ohio State University, respectively – analyzed over 200 gigabytes of data for the transaction history between bitcoin and tether, another digital currency. Tether is an asset known as a "stablecoin," which has its trading value connected to the dollar.

The professors' study found that tethers being traded for bitcoins revealed a pattern.

https://www.cnbc.com/2019/11/04/study-single-anonymous-market-manipulator-pushed-bitcoin-to-20000.html

Where are we heading?

- There are currently ~2300 cryptocurrencies
- Some are tied to real currency
  - E.g., Tether's stablecoin backed by $
  - Designed to park funds during times of high volatility
  - But they admitted that it only has 74% in of Tether backed by cash reserves

GIZMODO

French Students Will Now Have to Learn About Bitcoin

Jennings Brown • November 1, 2019

High school students in France may be among the first people in the world to actually understand how cryptocurrency works.

The Next Web reports that the French education ministry, Le Ministère de l'Éducation Nationale, will integrate cryptocurrency into its curriculum and teach students the influence that bitcoin has on the economy. An outline of the curriculum notes that under this new module, high school teachers will provide a basic overview of cryptocurrency so students can understand the framework of decentralized financial systems.

https://gizmodo.com/french-students-will-now-have-to-learn-about-bitcoin-1839547502

GREGORY BARBER BUSINESS 10.28.2019 07:00 AM

What's Blockchain Actually Good for, Anyway? For Now, Not Much

Not long ago, blockchain technology was touted as a way to track tuna, bypass banks, and preserve property records. Reality has proved a much tougher challenge.

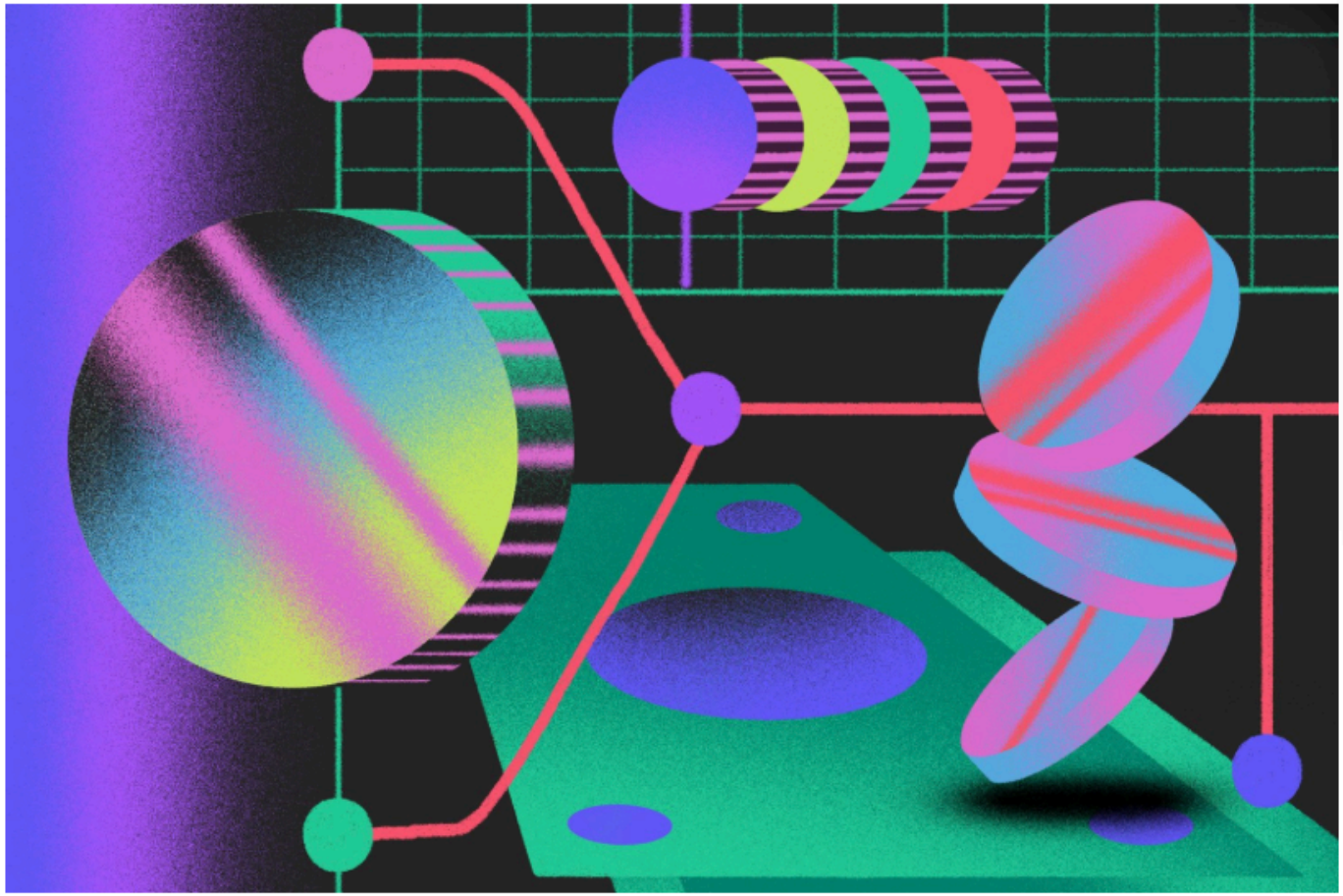


ILLUSTRATION: ARIEL DAVIS

The end