University of BRISTOL

Systems Security

COMSM1500

bristol.ac.uk

University of BRISTOL

Blockchain

bristol.ac.uk

Plan

- Proof of work
- Transactions
- Chain and consensus protocol

Bitcoin

- Network protocol
- A set of cryptographic functions
- Game theory equilibrium / economics

Bitcoin

- Not something entirely new
- Take 5 or 6 technologies from the 70s, 80s and 90s
  - Every pieces existed years before bitcoin came out
- Bundle them together in an innovative fashion

Bitcoin

- Not something entirely new
- Take 5 or 6 technologies from the 70s, 80s and 90s
  - Every pieces existed years before bitcoin came out
- Bundle them together in an innovative fashion
- Innovation is the novel architecture

Proof of work

- SHA256
  - One way hash function that produce 256 bits

Proof of work

- SHA256
  - One way hash function that produce 256 bits
  - One way hash function date back to 1975

Proof of work

Homework/exam question:
Define one way hash function.

- SHA256
  - One way hash function that produce 256 bits
  - One way hash function date back to 1975

Proof of work

- SHA256
  - One way hash function that produce 256 bits
- 1997 Adam Back Hashcache
  - Antispam system
  - Before someone can post something need to perform 1000 SHA256 of the message
  - Can be verified
  - Cost time ~0.5sec per message
  - Genuine users fine! Attacker spend lost of resource
  - Proof of work!

Proof of work

- Difficulty target
- Generate SHA256 (i.e. iterate until)
    - Start by X zero
    - OR smaller than value Y
    - This is equivalent
- Chance of starting with one 0?

Proof of work

- Difficulty target
- Generate SHA256 (i.e. iterate until)
  - Start by X zero
  - OR smaller than value Y
  - This is equivalent
  - Value + Nonce
- Chance of starting with one 0?
  - 50%
  - Two 0 25%
  - Etc…
  - Exponentially difficult
  - 2^N
- Only solution is brute force (e.g. proof of work)

Proof of work

Homework/exam question: Explain proof of work and difficulty target

- Difficulty target
- Generate SHA256 (i.e. iterate until)
  - Start by X zero
  - OR smaller than value Y
  - This is equivalent
  - Value + Nonce
- Chance of starting with one 0?
  - 50%
  - Two 0 25%
  - Etc…
  - Exponentially difficult
  - 2^N
- Only solution is brute force (e.g. proof of work)

Proof of work

- Difficulty change every 2016 blocks
  - 2016 should take two weeks to compute
- Proof of work convert into electricity consumption
  - i.e. financial cost
  - Get reward if play fair
- Check X most recent transactions (do they meet the rules)
  - This generates some value
- This value is hashed (proof of worked)
  - The block must fit within the consensus
- Miner pay itself at the top of the block

Transactions

- e-currency with distributed generation and distribution of money
- Transactions
  - Irreversible
  - Inexpensive
  - Over anonymous peer-to-peer network
  - Broadcast in seconds, verified within 10 to 60 minutes (included in the chain)
  - Pay using private key (digital signature); verify with public key
    - “money” associated with the public key
  - Double spending prevention via distributed ledger

Transactions

- e-currency with distributed generation and distribution of money
- Transactions
  - Irreversible
  - Inexpensive
  - Over anonymous peer-to-peer network
  - Broadcast in seconds, verified within 10 to 60 minutes (included in the chain)
  - Pay using private key (digital signature); verify with public key
    - “money” associated with the public key
  - Double spending prevention via distributed ledger
  - Digital signature 1975

Transactions

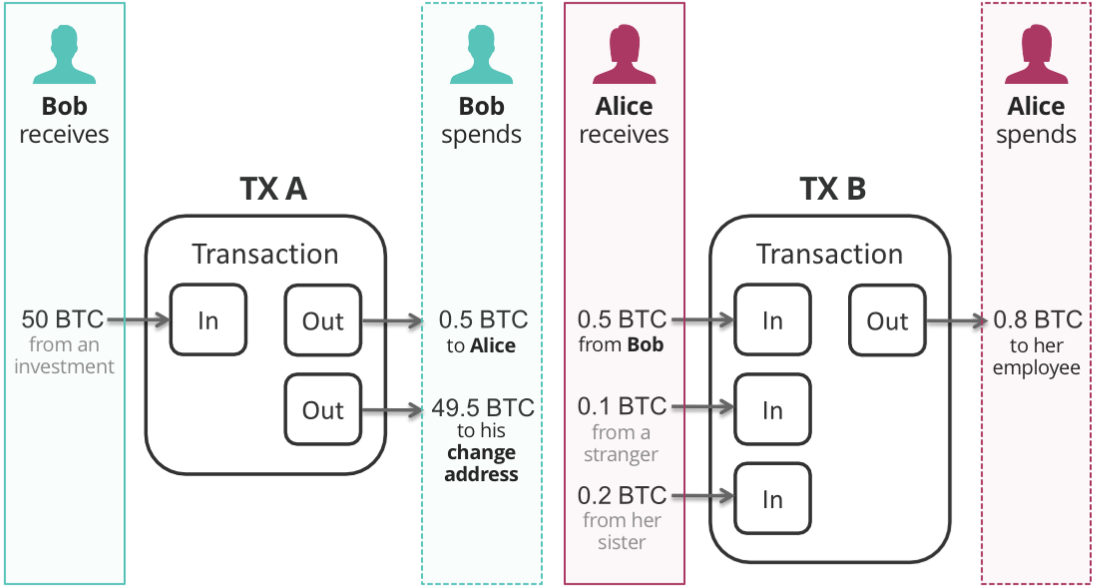
- e-currency with distributed generation and distribution of money
- Transactions
  - Irreversible
  - Inexpensive
  - Over anonymous peer-to-peer network
  - Broadcast in seconds, verified within 10 to 60 minutes (included in the chain)
  - Pay using private key (digital signature); verify with public key
    - “money” associated with the public key
  - Double spending prevention via distributed ledger
- Pseudonymous
  - Pay to public key
    - Can generated arbitrary pair and move money around
  - In many cases identification is possible
    - e.g. when going to an exchange (bitcoin -> £ or $ or € etc…)
    - e.g. IP addresses
    - non-trivial

Transactions

Homework/exam question: Bitcoin is pseudonymous not anonymous. Explain why.

- e-currency with distributed generation and distribution of money
- Transactions
  - Irreversible
  - Inexpensive
  - Over anonymous peer-to-peer network
  - Broadcast in seconds, verified within 10 to 60 minutes (included in the chain)
  - Pay using private key (digital signature); verify with public key
    - “money” associated with the public key
  - Double spending prevention via distributed ledger
- Pseudonymous
  - Pay to public key
    - Can generated arbitrary pair and move money around
  - In many cases identification is possible
    - e.g. when going to an exchange (bitcoin -> £ or $ or € etc…)
    - e.g. IP addresses
    - non-trivial

Transactions

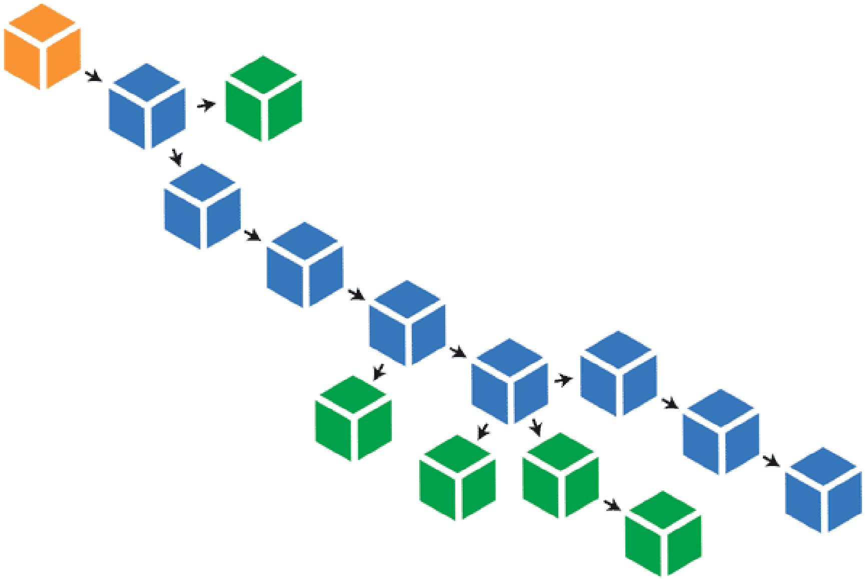


Chain

- Public decentralized ledger (block chain)
- Of transactions that transfer value from
  - One senders
  - To one recipients
  - Protected by signature
- Integrity of the ledger verified by miners
  - Audit transactions
  - Use proof of work for consensus
  - Miner receive reward (i.e. mint new “money”)

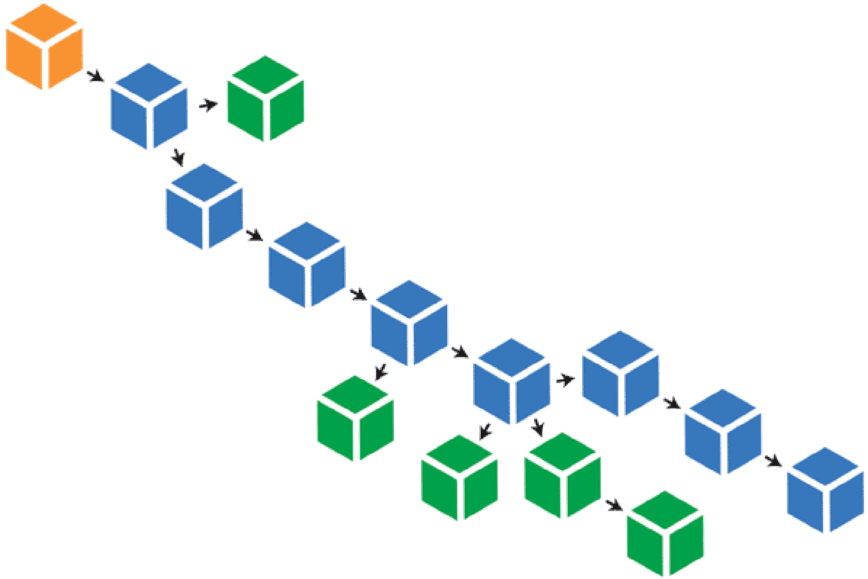
Chain

- You collect X transactions and you start building a block
- Once the block is ready you send it around and other miners verify and start computing the next block
- What happened if two blocks are computer in parallel?



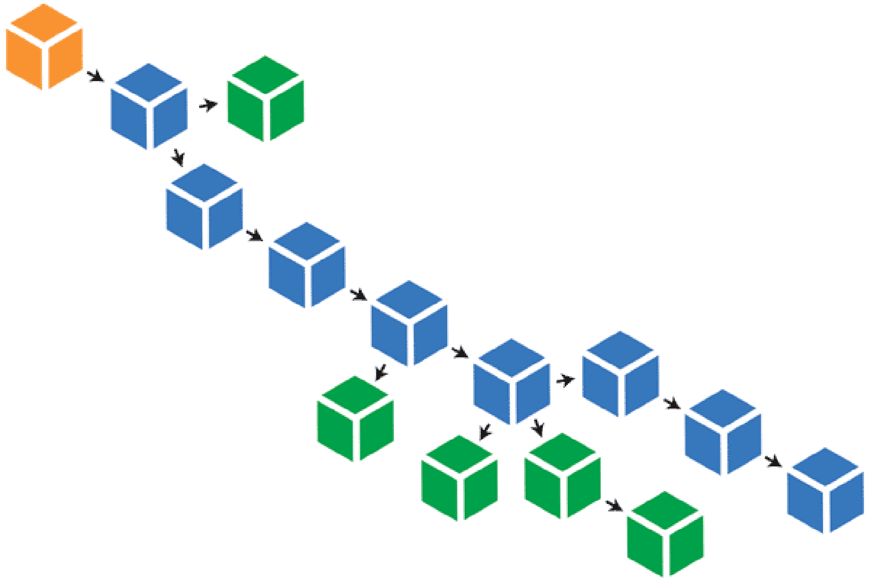
Chain

- You collect X transactions and you start building a block
- Once the block is ready you send it around and other miners verify and start computing the next block
- Parallel branches (fork)
  - Same branch validate it and go compute the next block (blue chain)
  - Different branch sees the next block is not on their branch of the chain discard their branch and start computing the right one (green chain)
  - This does not loose transactions (they are broadcasted, they will have ended in the blue block)



Chain

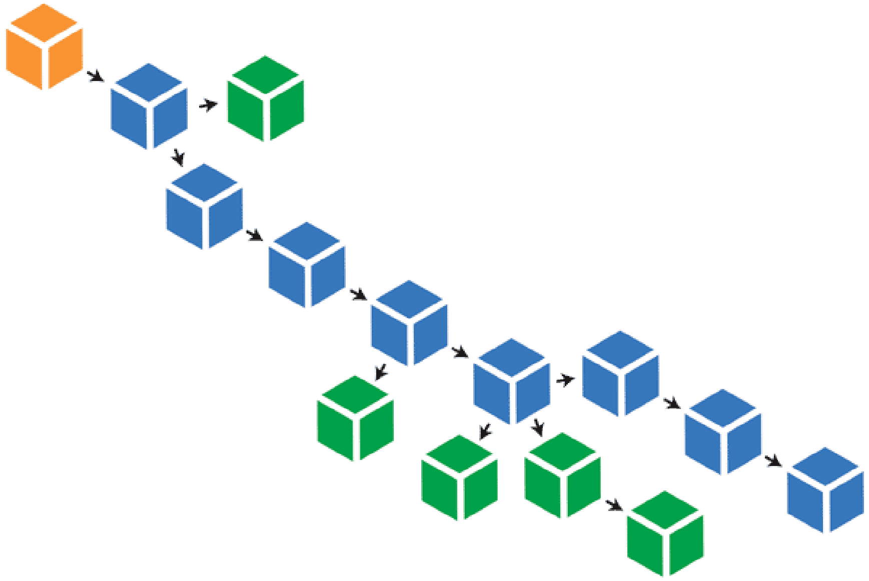
- You collect X transactions and you start building a block
- Once the block is ready you send it around and other miners verify and start computing the next block
- Parallel branches (fork)
  - Same branch validate it and go compute the next block (blue chain)
  - Different branch sees the next block is not on their branch of the chain discard their branch and start computing the right one (green chain)
  - This does not loose transactions (they are broadcasted, they will have ended in the blue block)
- Eventual convergence (that's bitcoin, there is other consensus protocol)
- Nakamoto concensus



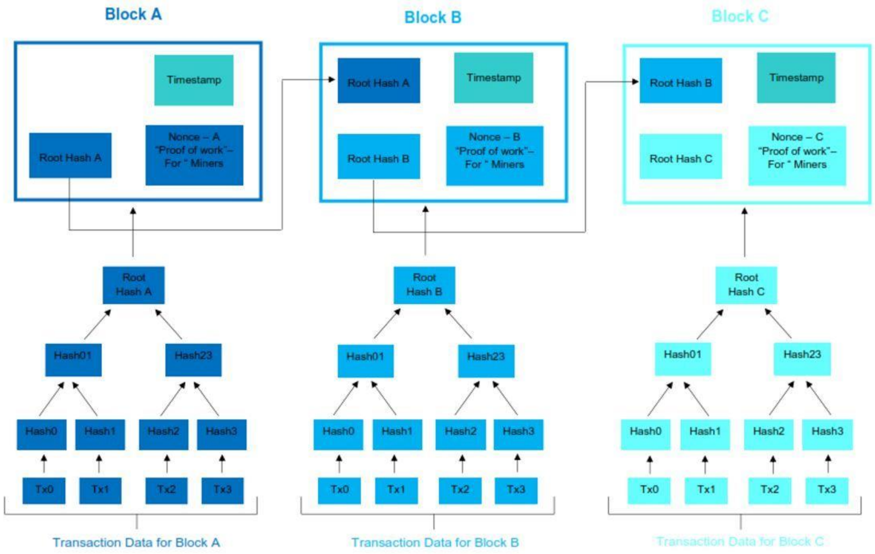
Chain

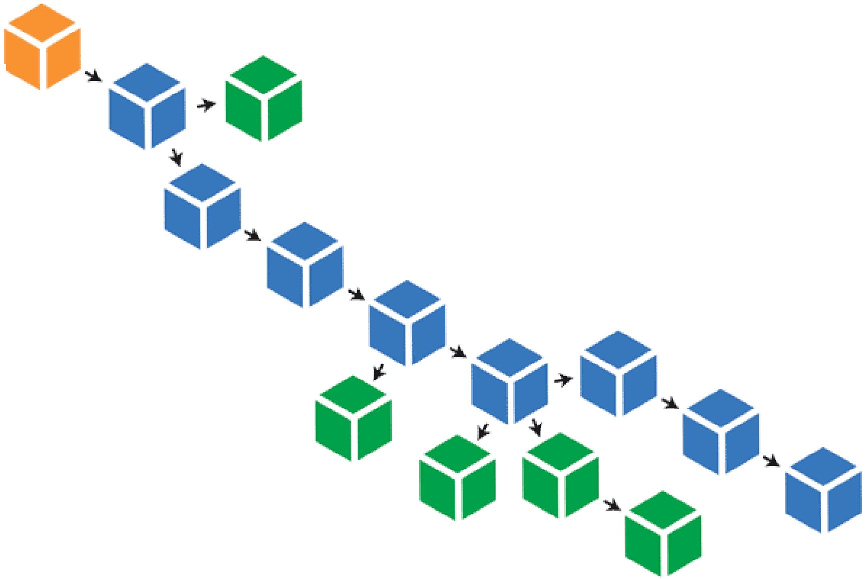
Homework/exam question:
Explain Nakamoto consensus.

- You collect X transactions and you start building a block
- Once the block is ready you send it around and other miners verify and start computing the next block
- Parallel branches (fork)
  - Same branch validate it and go compute the next block (blue chain)
  - Different branch sees the next block is not on their branch of the chain discard their branch and start computing the right one (green chain)
  - This does not loose transactions (they are broadcasted, they will have ended in the blue block)
- Eventual convergence (that's bitcoin, there is other consensus protocol)
- Nakamoto concensus

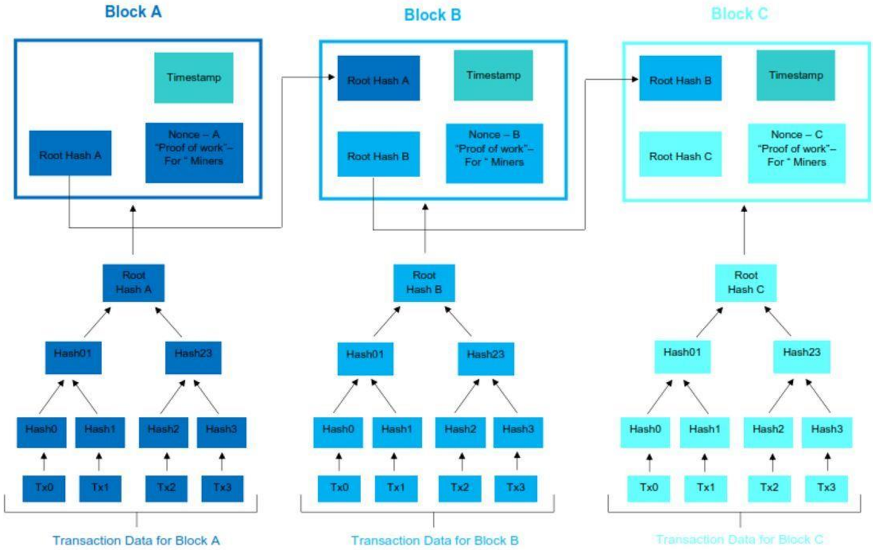


Chain


Block A
Timestamp
Root Hash A
Nonce – A "Proof of work"– For " Miners
Root Hash A
Hash01
Hash23
Hash0
Hash1
Hash2
Hash3
Tx0
Tx1
Tx2
Tx3
Transaction Data for Block A
Block B
Root Hash A
Timestamp
Root Hash B
Nonce – B "Proof of work"– For " Miners
Root Hash B
Hash01
Hash23
Hash0
Hash1
Hash2
Hash3
Tx0
Tx1
Tx2
Tx3
Transaction Data for Block B
Block C
Root Hash B
Timestamp
Root Hash C
Nonce – C "Proof of work"– For " Miners
Root Hash C
Hash01
Hash23
Hash0
Hash1
Hash2
Hash3
Tx0
Tx1
Tx2
Tx3
Transaction Data for Block C



Chain


Block A
Timestamp
Root Hash A
Nonce – A "Proof of work"– For " Miners
Block B
Root Hash A
Timestamp
Root Hash B
Nonce – B "Proof of work"– For " Miners
Block C
Root Hash B
Timestamp
Root Hash C
Nonce – C "Proof of work"– For " Miners
Root Hash A
Hash01
Hash23
Hash0
Hash1
Hash2
Hash3
Tx0
Tx1
Tx2
Tx3
Transaction Data for Block A
Root Hash B
Transaction Data for Block B
Root Hash C
Transaction Data for Block C

- Merkel Tree (1979)
  - To prove a value (transaction)
  - Verify only a path
- Timestamping (1990)
  - Collect documents
  - Build a Merkel Tree
  - Build a log of what happened
  - Can prove something happened

Latest block

- https://www.blockchain.com/explorer

Fork in practice?

Practice

- 1 block branch
  - ~ once a day on average
- 2 block branch
  - ~once a week/ once a month
- 3 block
  - starting to become very unlikely
  - some network failure leading to some partition
  - … or something else

Practice

- 1 block branch
  - ~ once a day on average
- 2 block branch
  - ~once a week/ once a month
- 3 block
  - starting to become very unlikely
  - some network failure leading to some partition
  - … or something else
- In April 2013 bitcoin experience long block fork
  - Emergency raised at 7 blocks
  - What happened?

Practice

- 1 block branch
  - ~ once a day on average
- 2 block branch
  - ~once a week/ once a month
- 3 block
  - starting to become very unlikely
  - some network failure leading to some partition
  - … or something else
- In April 2013 bitcoin experience long block fork
  - Emergency raised at 7 blocks
  - New bitcoin version release, using a new DB
  - Should not make a difference…

Practice

- In April 2013 bitcoin experience long block fork
  - Emergency raised at 7 blocks
  - New bitcoin version release, using a new DB
  - Should not make a difference…
  - … but bugs modify exhibited consensus
  - Old DB had a bug cannot validate more than 1024 transactions and crash
  - Updated version move forward
  - Old version can only make progress on the old version
  - 27 blocks before the problem was solved (but, not transaction lost!)

Could you replace the chain?

Can you replace the chain?

- In theory yes
  - Compute a competing chain of size N+1 (new longest chain)
- In practice no
  - You would need to computer the chain of size N+1
  - … in less time than computing a single block on the “real” chain

What do you need to be a performant miner?

What do you need to be a performant miner?

- High computational resource
  - Can execute proof of work fast
- Cheap electricity
  - Cost of proof of computation is function of electricity price
- Good network access
  - You need to share your block to as many node as possible as fast as possible
  - You want to receive newly mined block quickly

What do you need to be a performant miner?

- High computational resource
  - Can execute proof of work fast
- Cheap electricity
  - Cost of proof of computation is function of electricity price
- Good network access
  - You need to share your block to as many node as possible as fast as possible
  - You want to receive newly mined block quickly
- Do not do it on commodity hardware
  - CPU -> GPU -> FPGA -> ASIC (Dedicated Hardware)
  - Electricity cost > returns
  - Need dedicated hardware
  - … and a country with cheap electricity (e.g. China)

Distributed Ledger

- Less fancy name for blockchain
- Distributed database – only needed if
  - Multiple mutually distrustful writers
  - No intermediate party trusted by all players
  - Interactions or dependencies between the transactions
- Blockchain is a buzz word, a lot of useless solution built around it
- … don’t trust the hype

Bitcoin

- Not something entirely new
- Take 5 or 6 technologies from the 70s, 80s and 90s
  - Every pieces existed years before bitcoin came out
- Bundle them together in an innovative fashion
  - One hash function (1975)
  - Digital signature (1975)
  - Merkel Tree (1979)
  - Timestamping (1990)
  - Proof of work (1997)

Bitcoin

Homework/exam question: Discuss how blockchain derived from older technologies.

- Not something entirely new
- Take 5 or 6 technologies from the 70s, 80s and 90s
  - Every pieces existed years before bitcoin came out
- Bundle them together in an innovative fashion
  - One hash function (1975)
  - Digital signature (1975)
  - Merkel Tree (1979)
  - Timestamping (1990)
  - Proof of work (1997)

Plan

- Proof of work
- Transactions
- Chain and consensus protocol

University of BRISTOL

Thank you, questions?

Office MVB 3.26

bristol.ac.uk