MONASH University

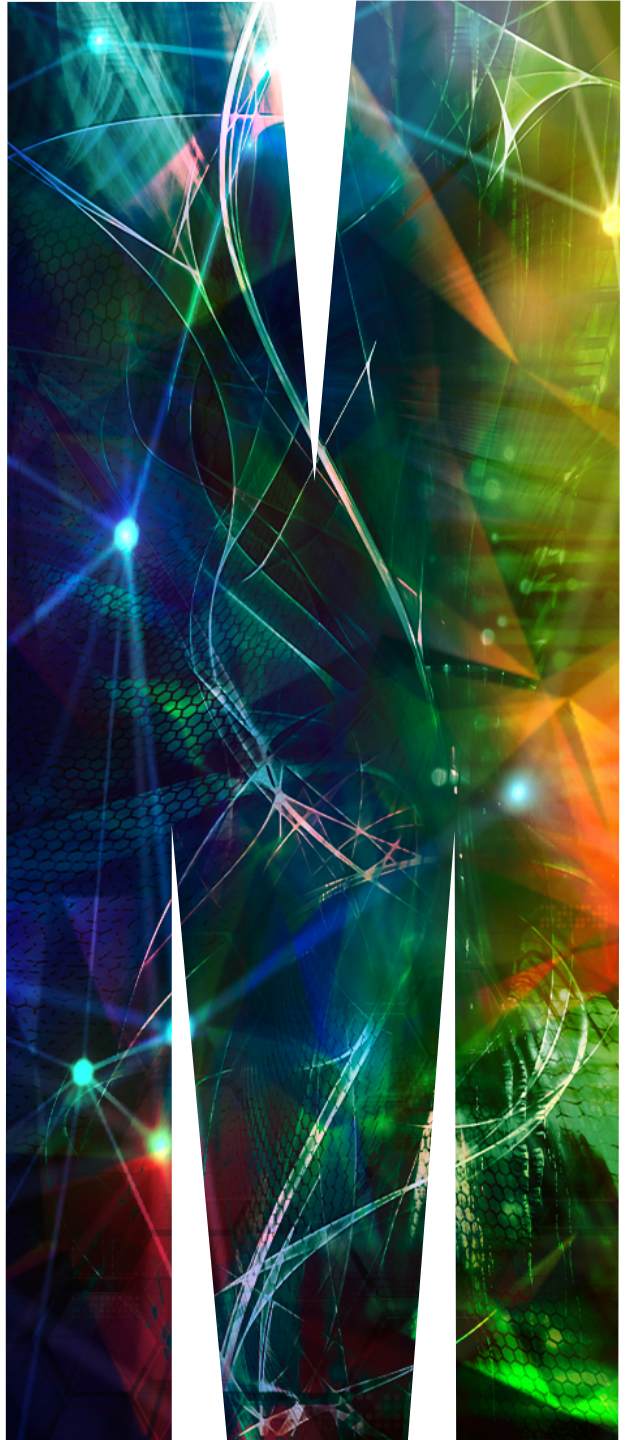
FIT5214: Blockchain

Lecture 4: Proof-of-Work

Lecturer: Rafael Dowsley

rafael.dowsley@monash.edu

https://dowsley.net



Unit Structure

- **Lecture 1: Introduction to Blockchain**
- **Lecture 2: Bitcoin**
- **Lecture 3: Ethereum and Smart Contracts**
- Lecture 4: Proof-of-Work (PoW)
- Lecture 5: Attacks on Blockchains
- Lecture 6: Class Test/Alternatives to PoW
- Lecture 7: Proof-of-Stake (PoS)
- Lecture 8: Privacy
- Lecture 9: Byzantine Agreement
- Lecture 10: Blockchain Network
- Lecture 11: Payment Channels
- Lecture 12: Guest Lecture

Unit Structure

- **Lecture 1: Introduction to Blockchain**
- **Lecture 2: Bitcoin**
- **Lecture 3: Ethereum and Smart Contracts**
- **Lecture 4: Proof-of-Work (PoW)**
- Lecture 5: Attacks on Blockchains
- Lecture 6: Class Test/Alternatives to PoW
- Lecture 7: Proof-of-Stake (PoS)
- Lecture 8: Privacy
- Lecture 9: Byzantine Agreement
- Lecture 10: Blockchain Network
- Lecture 11: Payment Channels
- Lecture 12: Guest Lecture

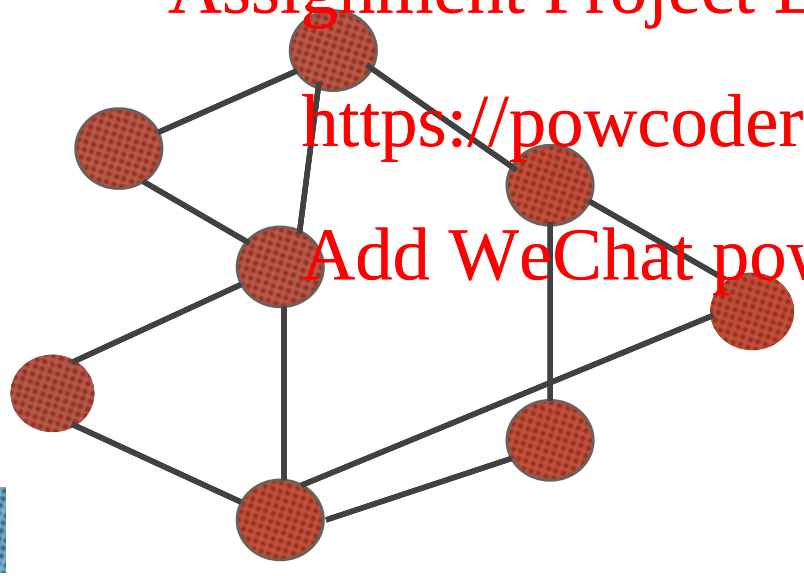
- ➡ Introduction to Bitcoin consensus
  - ➡ How PoW consensus works;
  - ➡ Its impact on Bitcoin;
- ➡ Understand existing design flaws and attacks
- ➡ Formal properties of PoW consensus
- ➡ How to address/avoid some of them (some of them are open challenges)

How Bitcoin works? Recap!

1. How to create money?
2. How to spend money?
3. How to verify transactions?
4. Why do we need consensus?

Motivation

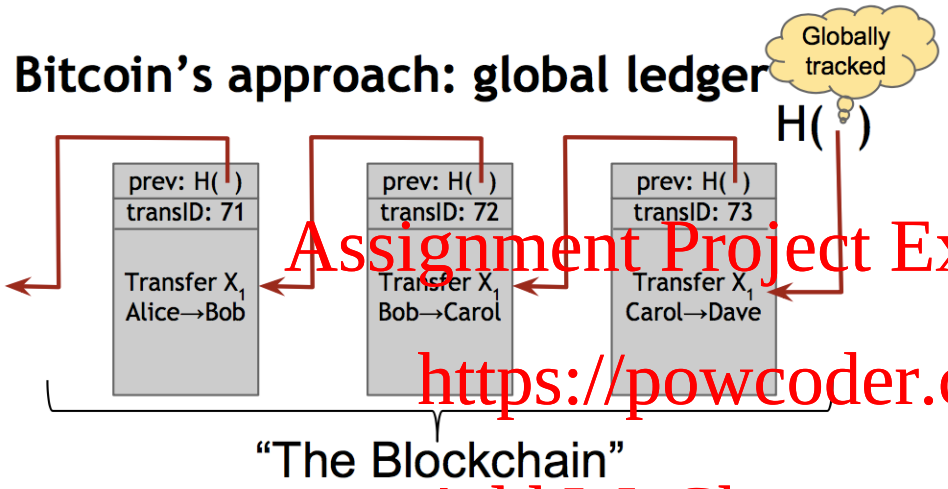
Let's vote



Give my coin **c_1** to Bob

Give my coin **c_1** to Chris

Basic Construction - Recap



1. Each block is referenced by the next block.
   e.g. H(block1) is included in the block2

2. Each block contains a set of transactions, organised as a Merkle tree

Ack: This picture is taken from Joseph Bonneau's slides at Bitcoin summer school, Greece, 2016.

Bitcoin Puzzle - Recap

Find a Nonce such that

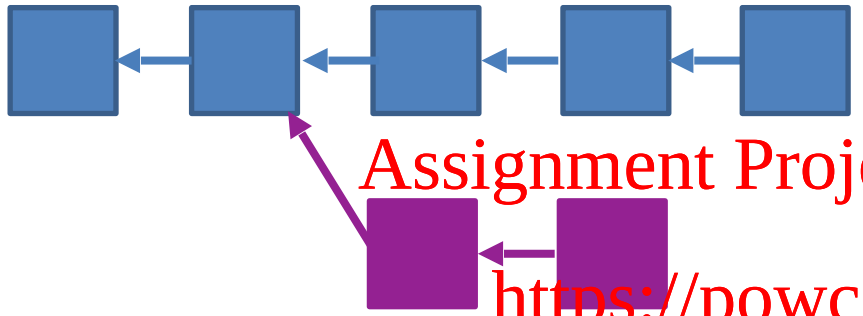
$$h(\text{prev\_hash}, \text{Nonce}, TX_1, TX_2, \ldots, TX_n) < D$$

where D is a difficulty parameter.

- $|\text{Nonce}| = 8$ bytes
- $|TX| \geq 250$ bytes
- h is SHA-256
- $|block| \leq 1$ MB
- n is about 4,000
- D is recalculated every 2016 blocks (currently about 2 weeks)

The process of solving this puzzle is called "Mining"

Nakamoto Consensus



The longest chain wins!

(Bitcoin assumes honest majority!)

Consensus Throughput

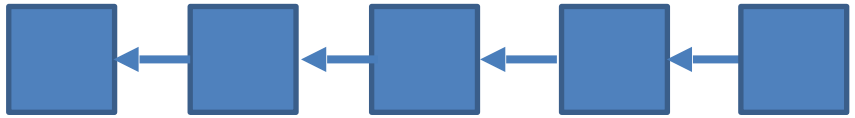
The number of transactions per second (TPS) Bitcoin can process.

7 TPS

Average: 115-200 TPS

Average: 1,700 TPS
Peak time: 56,000 TPS

Consensus Throughput



Roughly speaking:

- Each block is **1MB**
- Each transaction is about **256 bytes**
- Each block contains about **4000 transactions**
- Each block is created every **10 minutes**

Group discussion:

1. If you are going to design a **MonashCoin (as Bitcoin+)**, what are the potential ways to increase the throughput of Bitcoin transactions? Why?
2. Why is Bitcoin not already using your solution? What are the concerns?

Consensus Throughput

To increase throughput:

- Increase block size
- Decrease block creation time
- Graph of blocks to enable transaction inclusion in parallel
- Only record a “checksum” of transactions on the blockchain, and perform actual transactions in a payment channel (Week 11)

Why 10 minutes and 1MB?

It may not be the best choice:

- A new block should be created after most nodes learnt the previous block
- Disseminating blocks in the P2P network takes time
- Larger blocks take more time
- A shorter time creates more forks
- A shorter time reduces the difficulty of launching a double spending attack, unless the confirmation time is increased (details later)

Bitcoin ABC (Adjustable Blocksize Cap): 32MB per block
Bitcoin SV (Satoshi's Vision): Potentially 2GB per block

Enforce Block Creation Time

How to guarantee 10 minutes per block?

Difficulty parameter

Two weeks = 20160 minutes

14 days * 24 hours/day * 60 minutes/hour = 20160 minutes

10 minutes per block

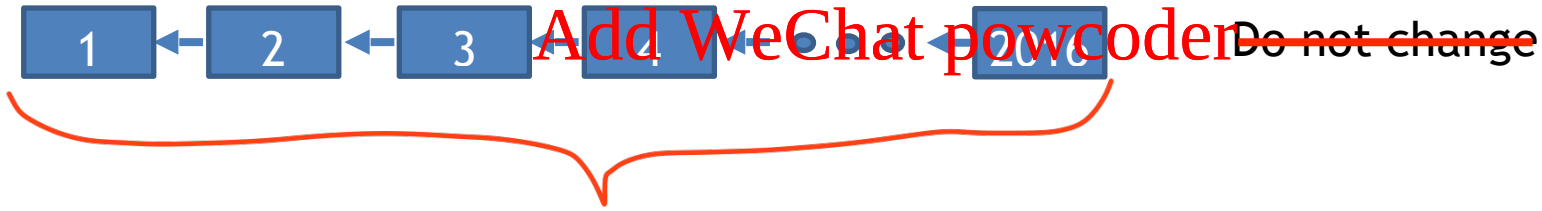
2016 blocks in two weeks

Enforce Block Creation Time

How to guarantee 10 minutes per block?

2016 blocks in two weeks

Difficulty parameter!

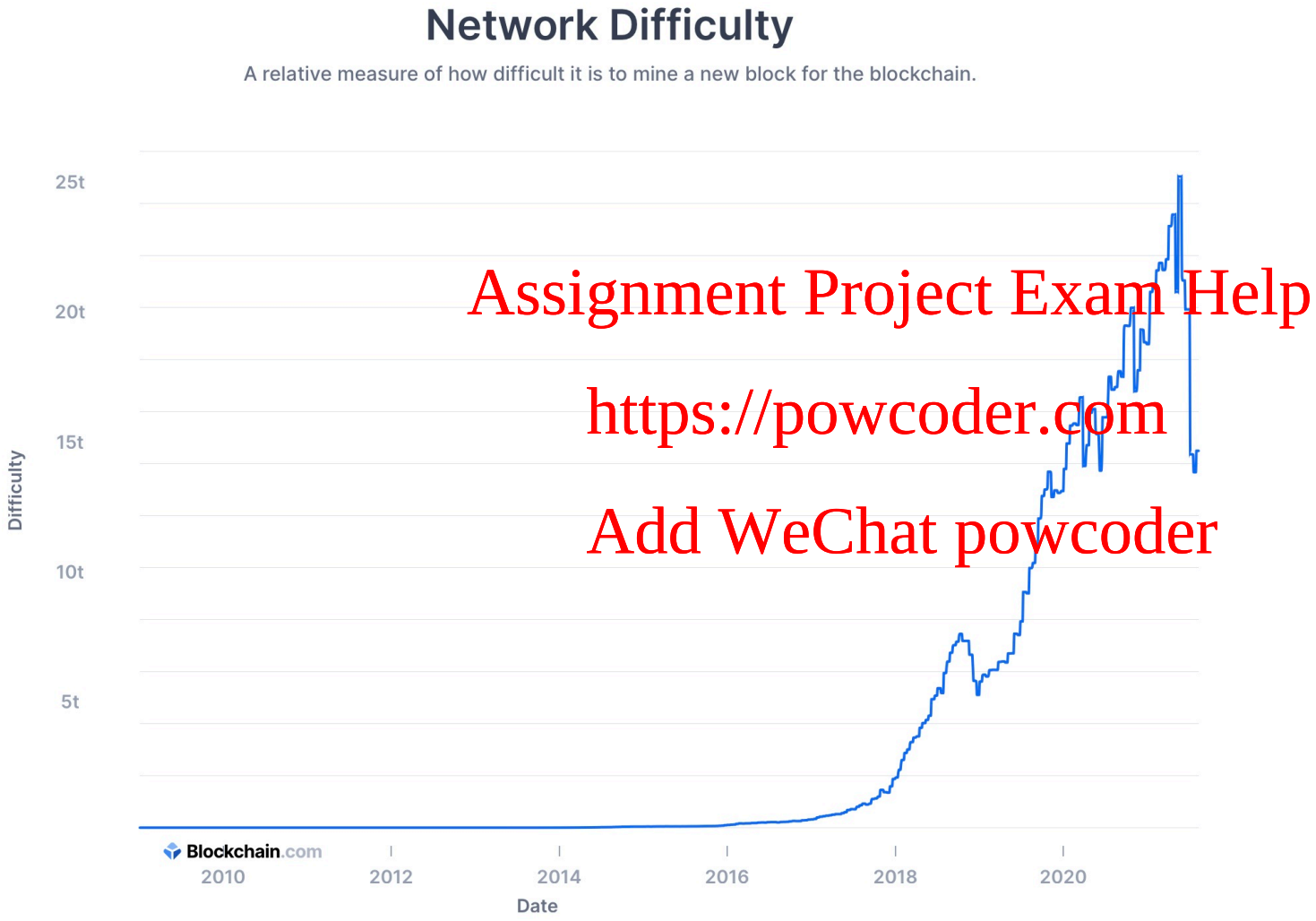


~~Two weeks~~

~~Four weeks~~ | ~~Difficulty * 2~~

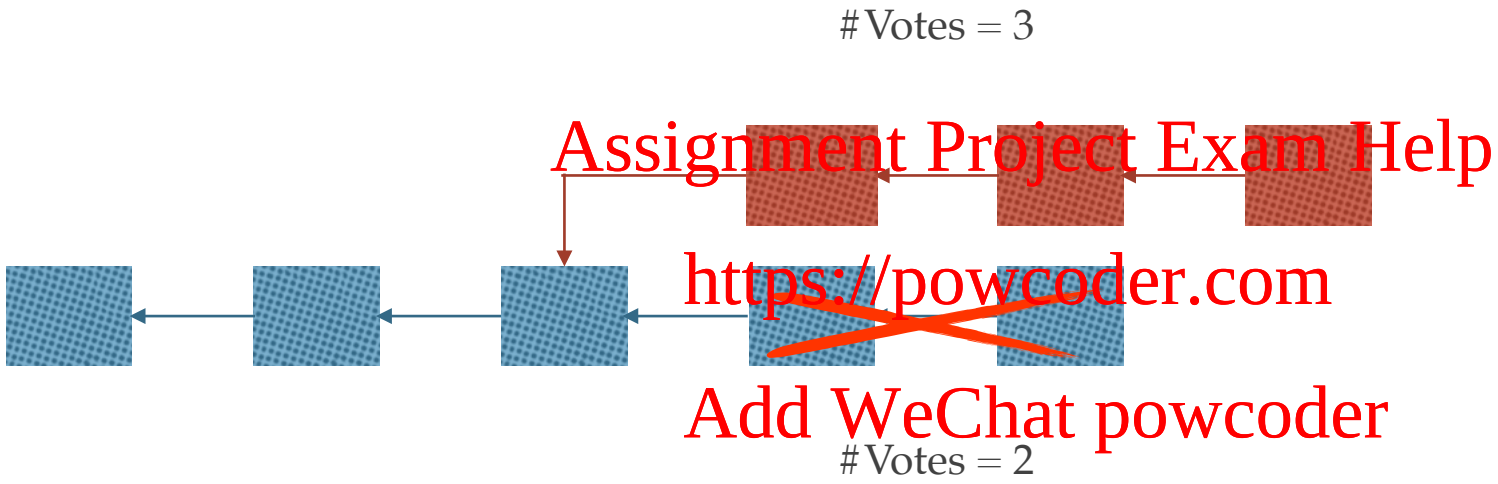
One week | Difficulty * 1/2

Evolving Difficulty



https://www.blockchain.com/charts/difficulty?timespan=all

Bitcoin Consensus: Longest Chain Rule



Confirmation

The number of blocks “confirmed” the validity of a transaction

Transactions in block 1 has 2016 confirmations
Transactions in block 2 has 2015 confirmations
…
Transactions in block 2015 has 2 confirmations
Transactions in block 2016 has 1 confirmation

For security reasons, it is recommended to wait many confirmations (Are 6 confirmations good enough?)

Read more: https://en.bitcoin.it/wiki/Confirmation

Confirmation

Each block is created every 10 minutes

- 1 confirmation - 10 minutes
- 2 confirmation - 20 minutes

…

- 5 confirmation 50 minutes
- **6 confirmation 1 hour**

Go and get coffee!

The number of blocks “confirmed” (the validity of a transaction)

Go and get coffee!

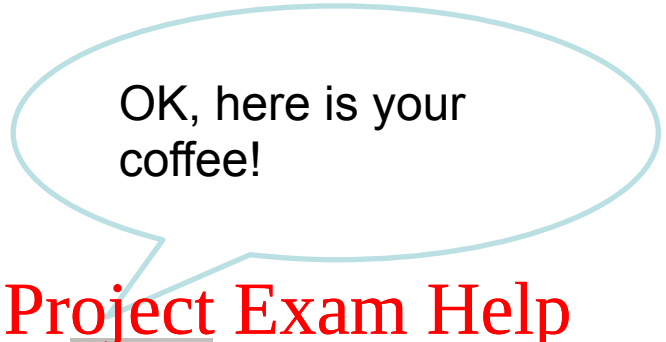
The number of blocks “confirmed” (the validity of a transaction)

What if no confirmation at all?

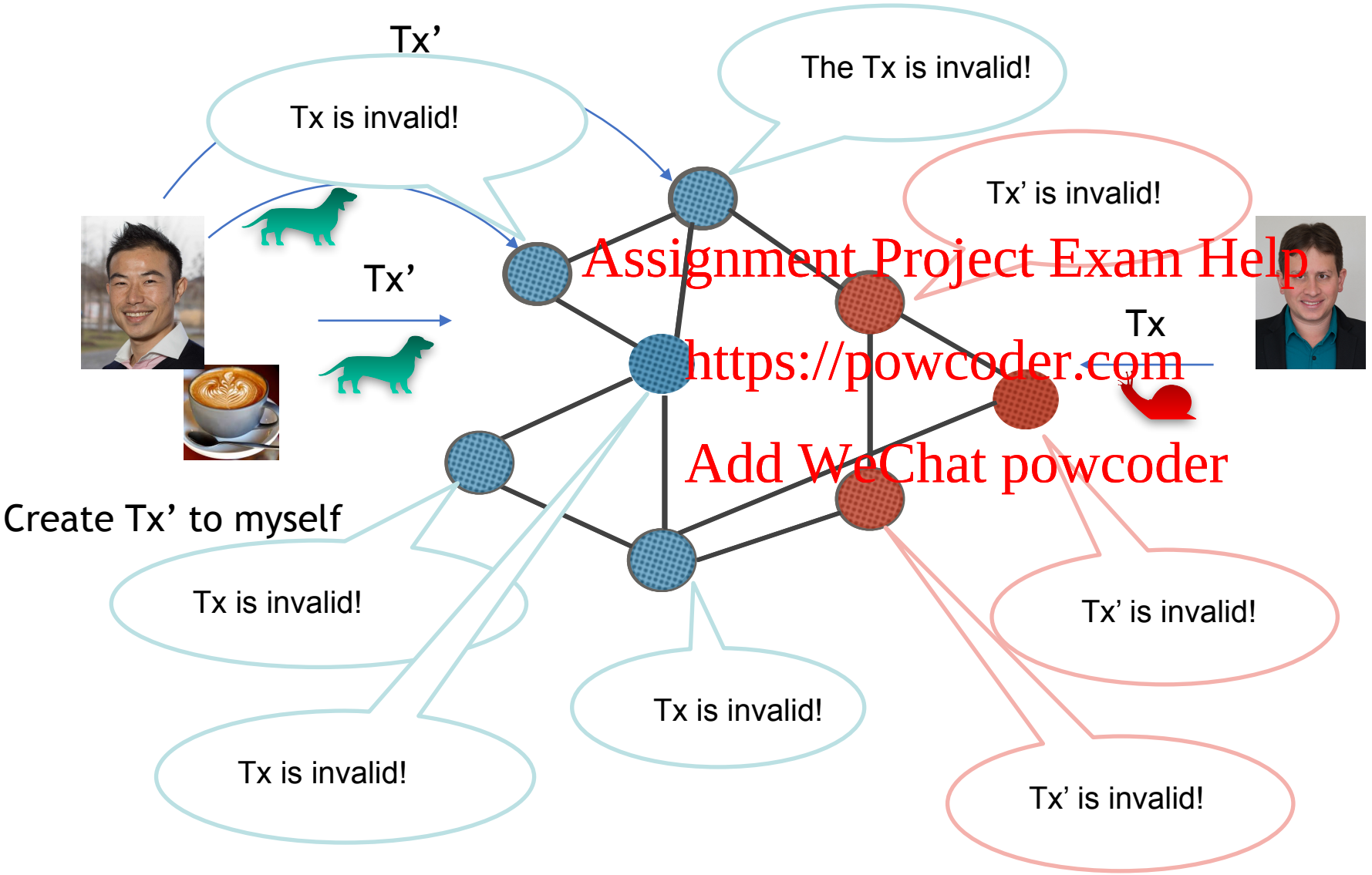
Group discussion:

1. What is the security problem if no confirmation is required?
2. Any solution you can propose?

Race Attack (0-confirmation)

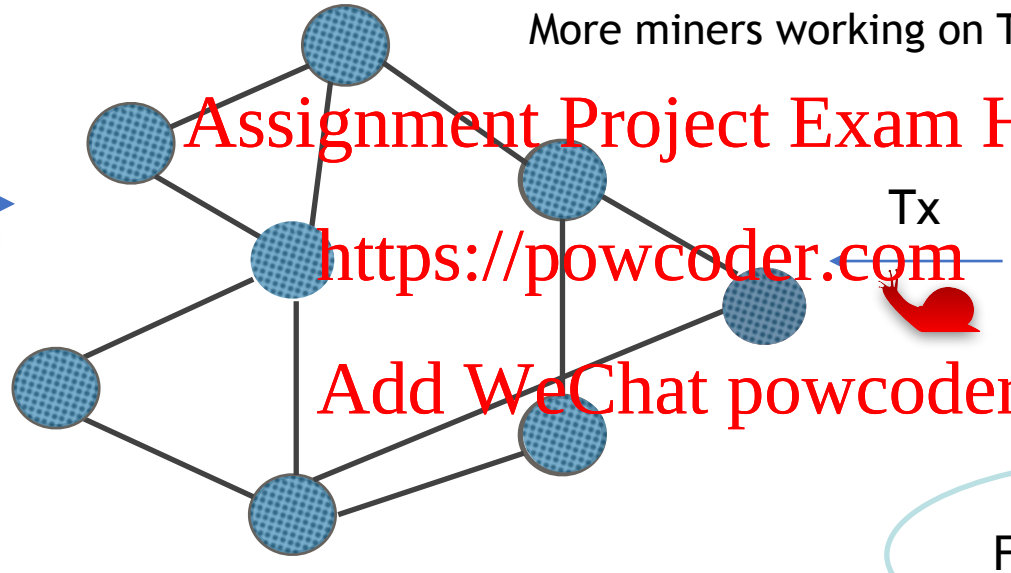


Race Attack (0-confirmation)



Race Attack (0-confirmation)

Tx'



More miners working on Tx' rather than Tx

Tx

What is the assumption in this example?

Free coffee!

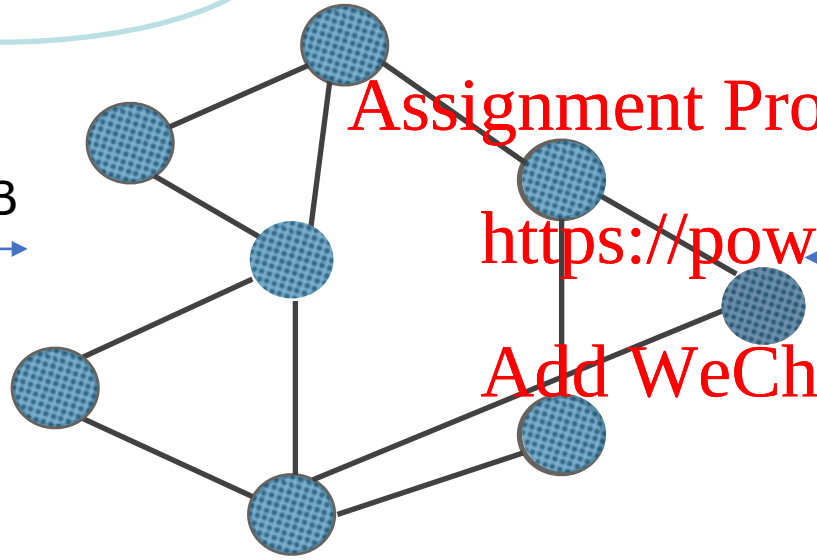
Requirement:
Faster network

MONASH University

Finney Attack (0-confirmation)

OK, here is my Tx

block B



Tx

OK, here is your coffee!

Create Tx and Tx'
Create a block B containing Tx'

The entire network only accepts Tx'

Free coffee!

Requirement:
Some mining power

MONASH University

Finney Attack (0-confirmation)

Alice launches a Finney attack as follows:

- Step 1. Create a valid transaction TX_1 to transfer a coin to PK_Alice;

Alice does NOT broadcast TX_1!

- Step 2. Create a valid block B containing TX_1 through block mining;

Alice does NOT broadcast the block, for now!

(So, to the network, the coin is unspent.)

- Step 3. Create a valid transaction spending the coin to PK_service;
- Step 4. Once the transaction is accepted by a 0-confirmation service, then broadcast B.

How about 1-confirmation?

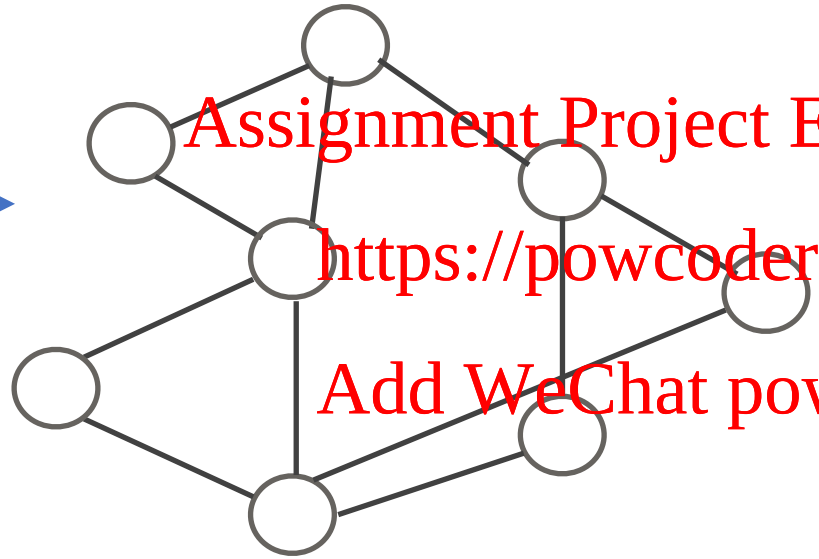
Vector76 attack (Finney attack+Race attack)

1. **Create two transactions Tx and Tx′**
2. **Create a block B containing Tx′**
3. **Send Tx to Payee**
4. **Wait for a block B′ containing Tx**
   (keep mining on B)
5. **Send B to the network faster than B′**

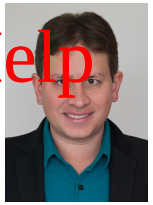
Finney attack: Step 1-4
Race attack: Step 5

Group Discussion

Tx’



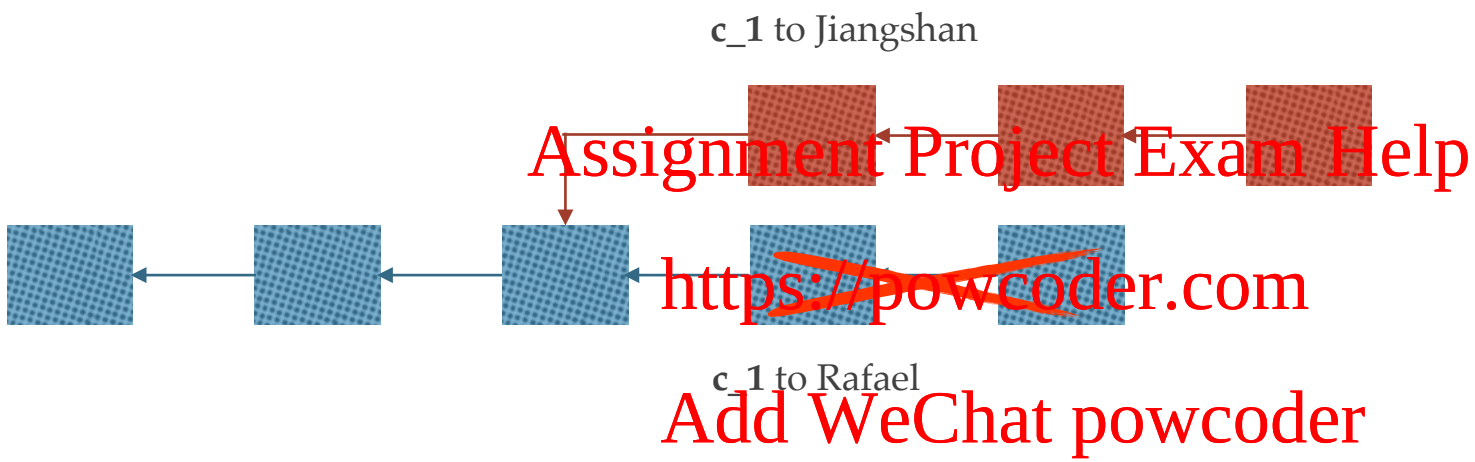
Tx



1. What are the possible ways to gain extra advantages on including your own transaction?
2. Why?
3. How to fix?

More Confirmations?

51% Attack



If an attacker has >50% CPU power, it can spend a coin more than once.

51% Attack

What's wrong?

The attack is on the Bitcoin consensus!

Terminology: Node

A **node,** or **replica**, is a single actor in the system.

- In a computer network, the computers are the nodes;
- In Blockchain, a node can be of different types:
  - A **client node** is a node which only stores a part of the blockchain.
  - A **full node** is a node which stores all the data of the blockchain, e.g. blockchain explorer/backup server
    NOTE: miners are full nodes!

- The **total number of nodes** in the system **is "n"**. (unless otherwise specified)

- Each node may have different powers in voting, we call the **total voting power** in the system **is "P"**.

What is “consensus”?

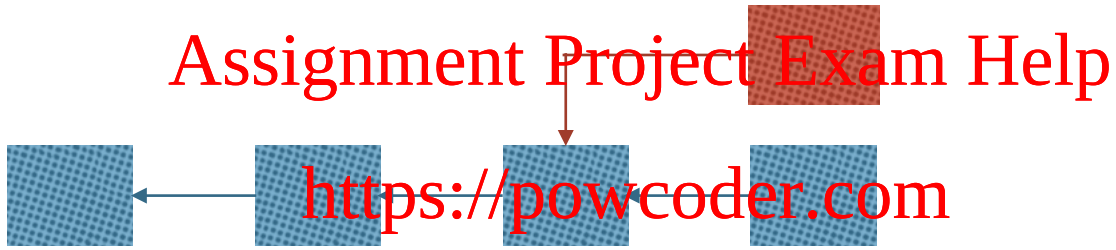
Definition 1. (Consensus.) There are n nodes, of which at most f can be malicious, i.e., at least $n - f$ nodes are correct. For all $i \in [1, n]$, node i starts with an input value v_i. The nodes must decide for one of those values, satisfying the following properties:

- **Agreement:** All correct nodes decide for the same value.

- **Termination:** All correct nodes terminate in finite time.

- **Validity:** The decision value must be the input value of a node.

Agreement cannot be reached

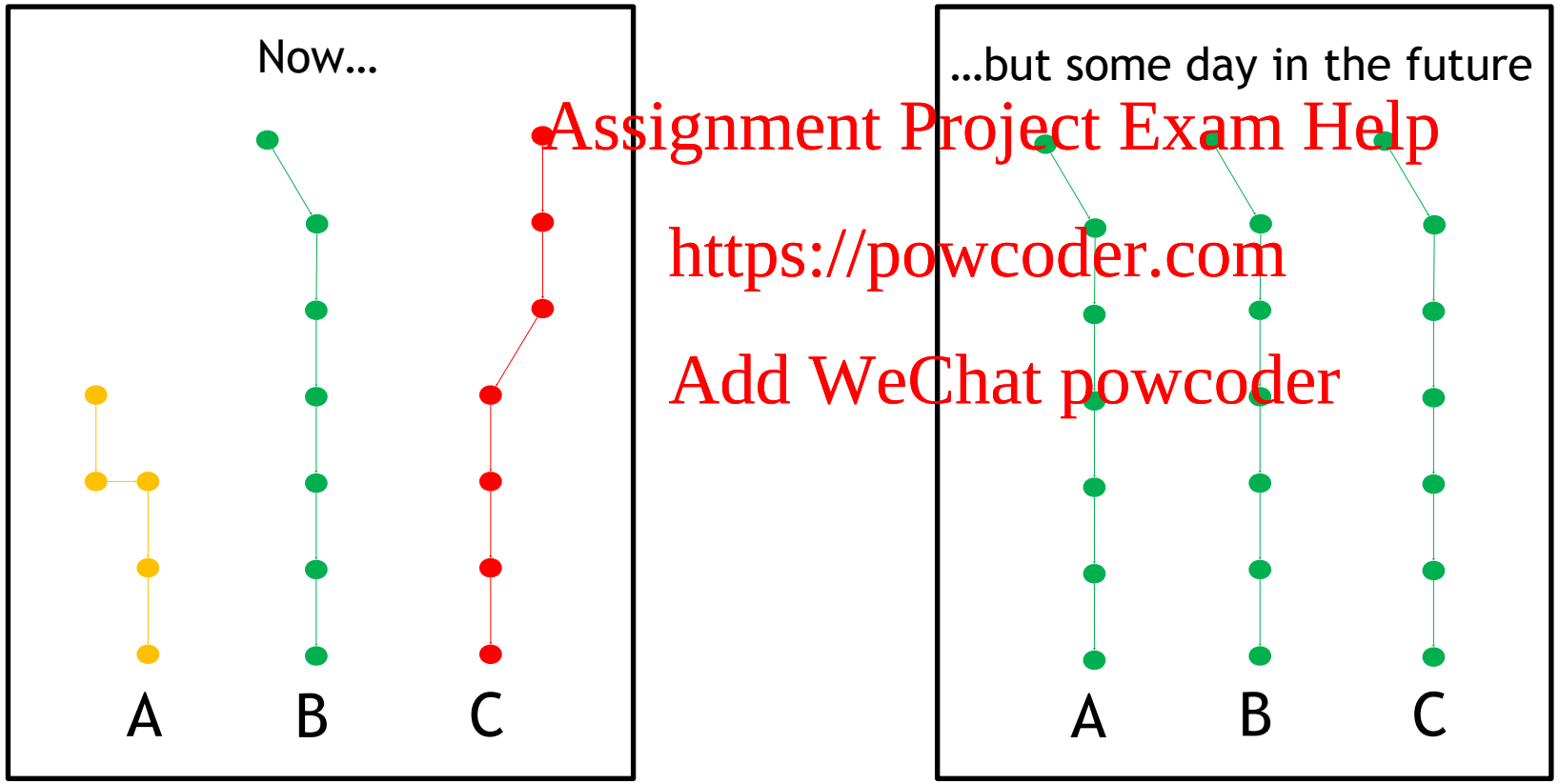


PoW consensus

Assuming a synchronous network, PoW guarantees **eventual consistency**, or more actually, nodes in PoW agrees on a ***Common Prefix*** of the blockchain.

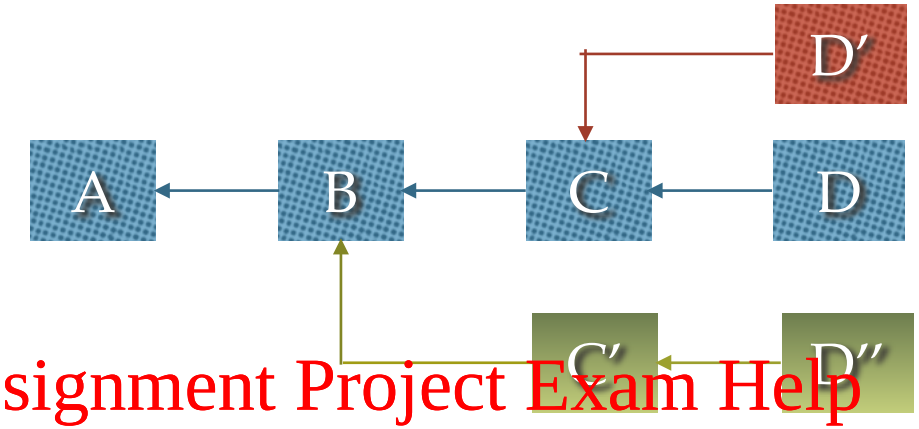
Eventual Consistency

Eventually, all the nodes will agree on the same blockchain



Common Prefix

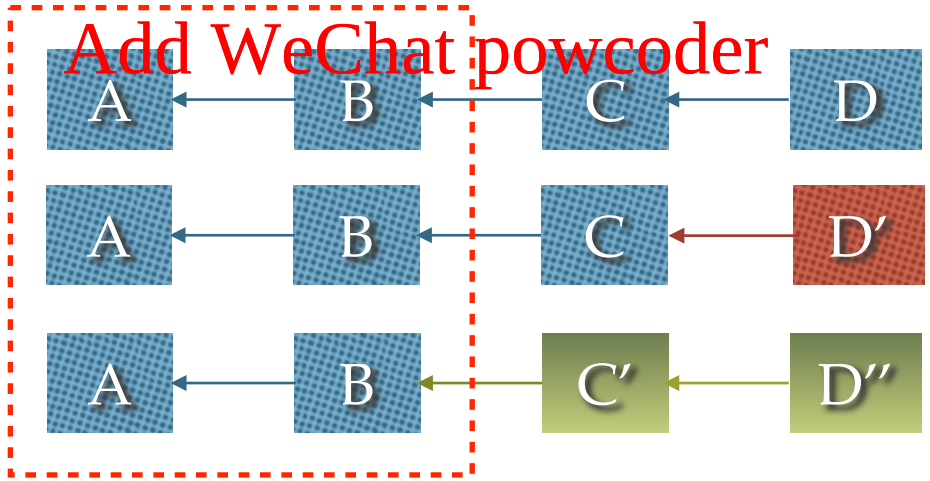
Global view:



Node 1:

Node 2:

Node 3:

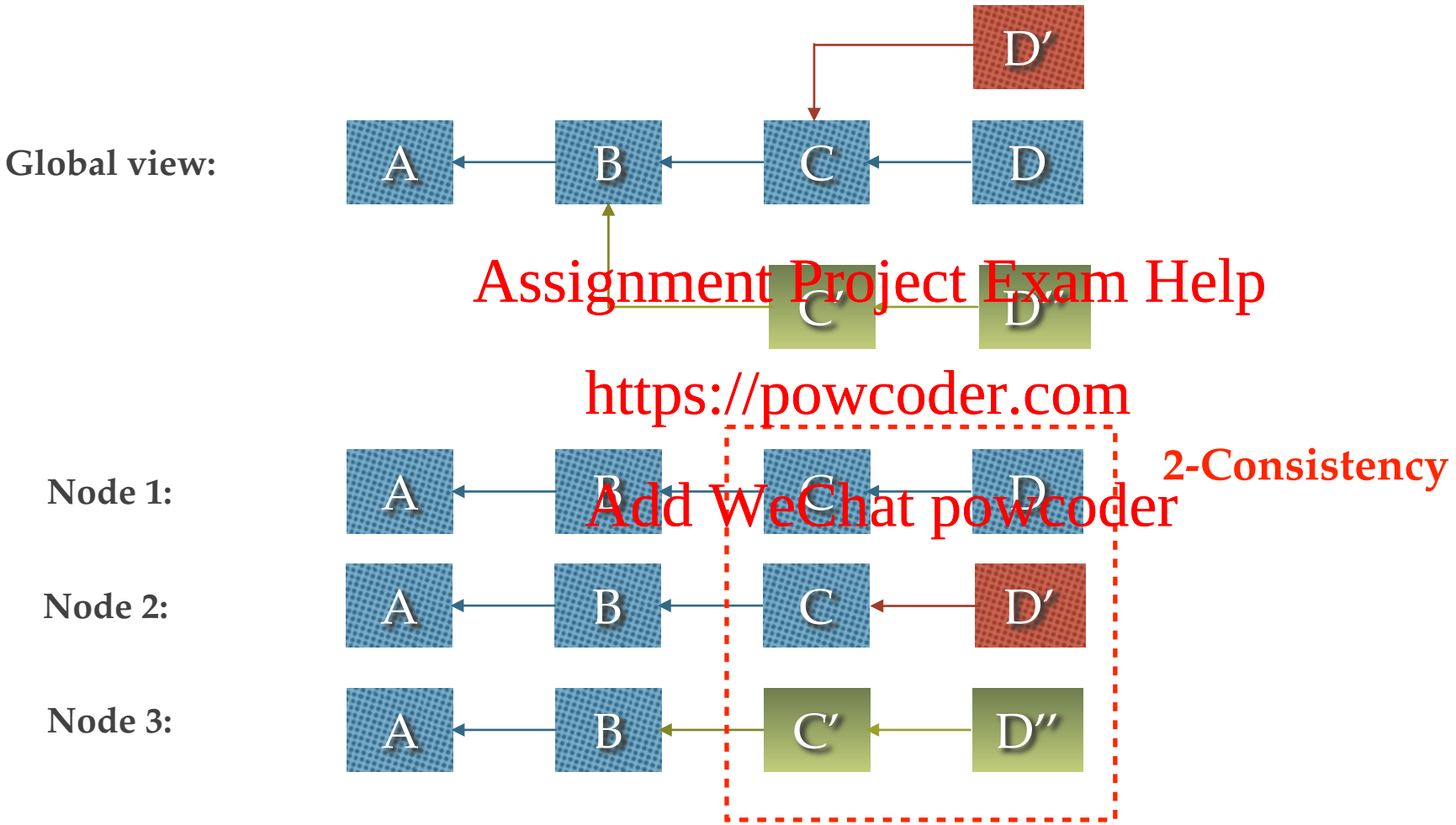


PoW Consensus

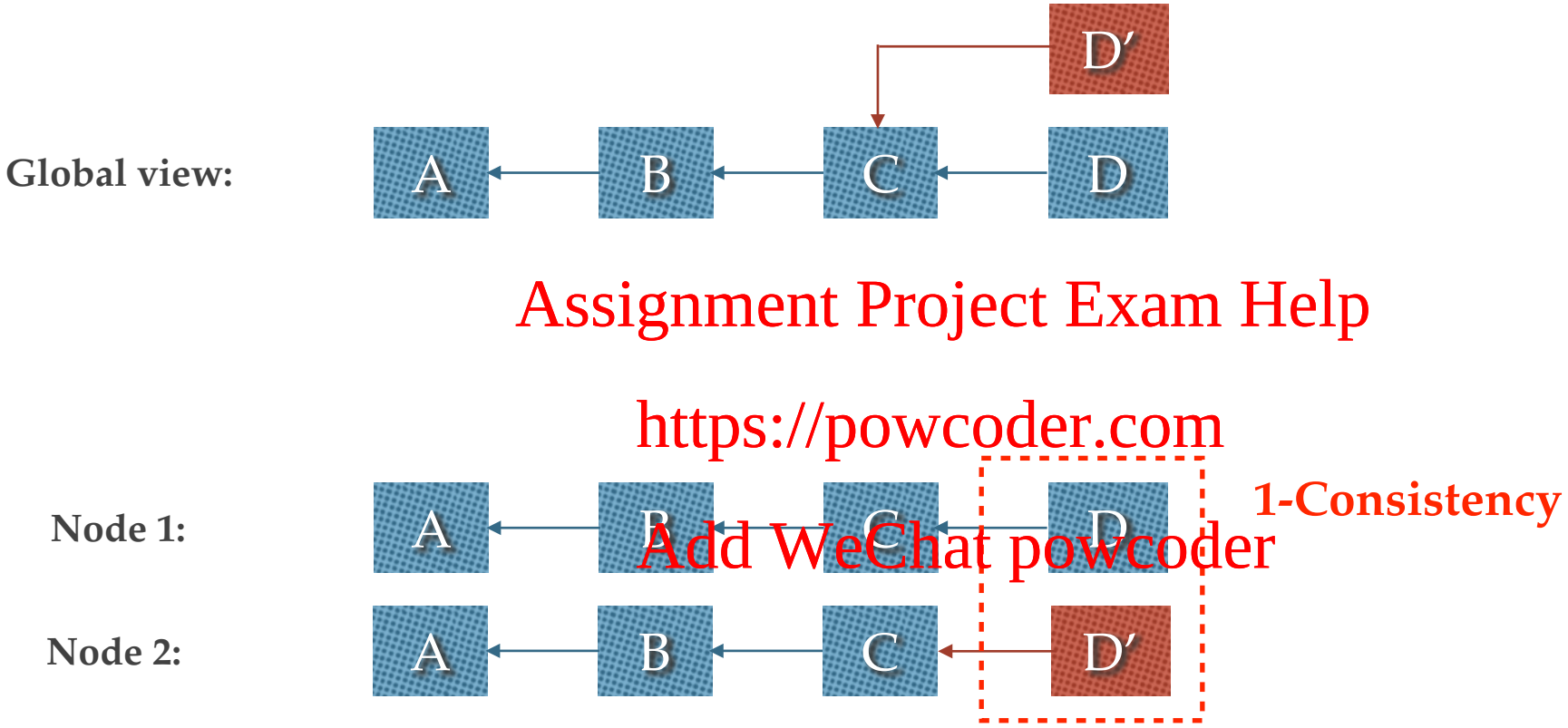
Assuming a synchronous network, PoW guarantees eventual consistency, or more actually, nodes in PoW agrees on a *Common Prefix* of the blockchain.

T-consistency **is used to quantify the quality of the blockchain consensus — after cutting down the last T blocks, all nodes should agree on the rest of the blockchain.**

T-Consistency

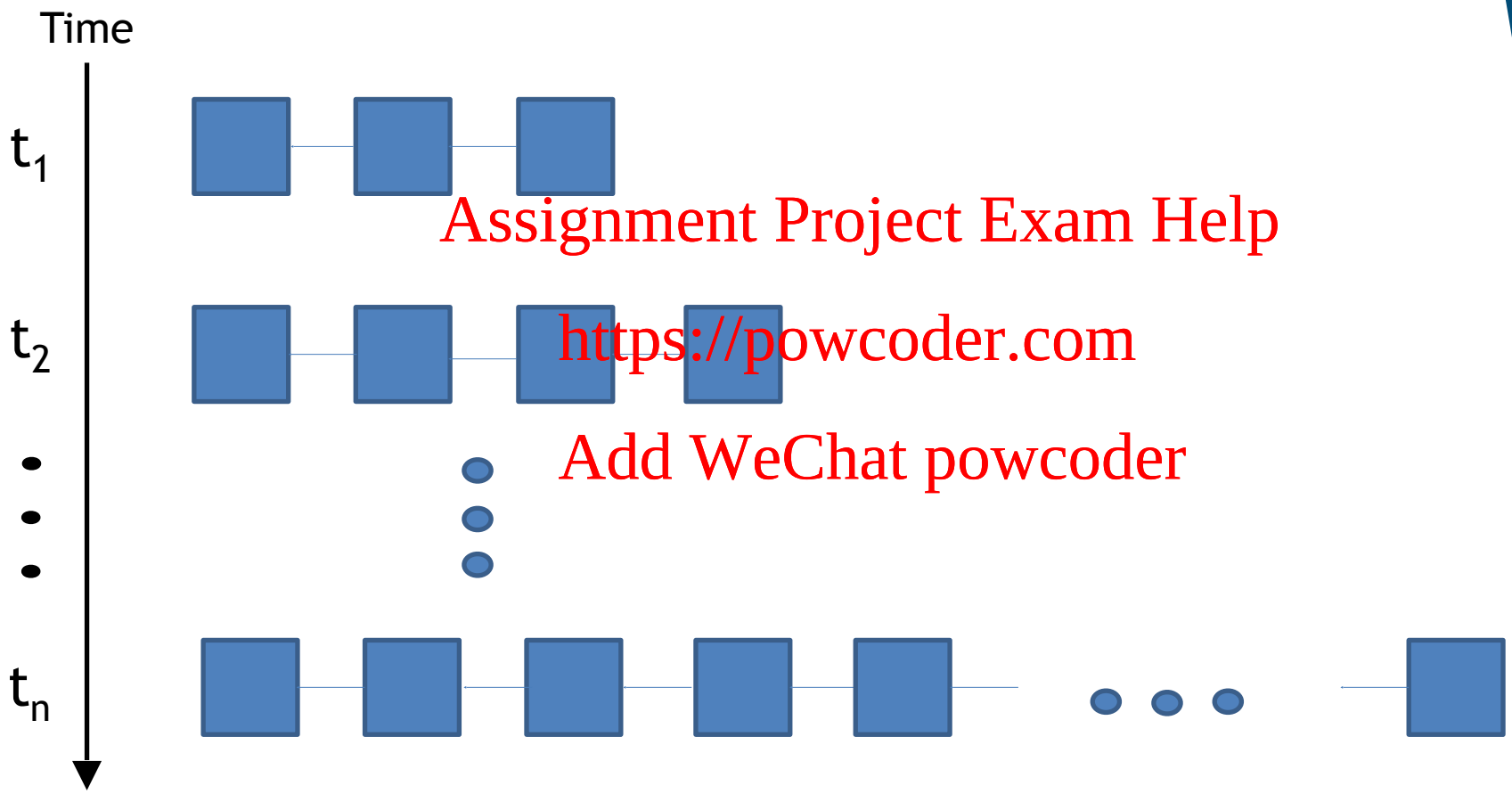


T-Consistency

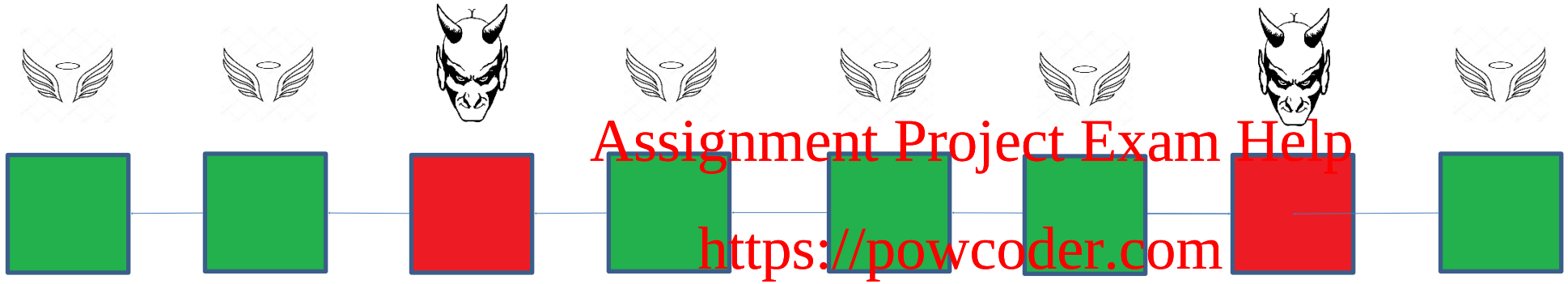


The best consistency is 0-consistency, which can be provided by traditional consensus protocols (Week 9)

Chain Growth

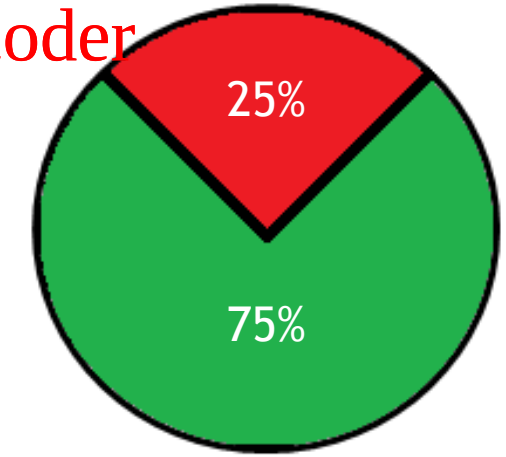


Chain Quality



Adversarial contribution = 2/8 = 25%

Ideal Chain Quality



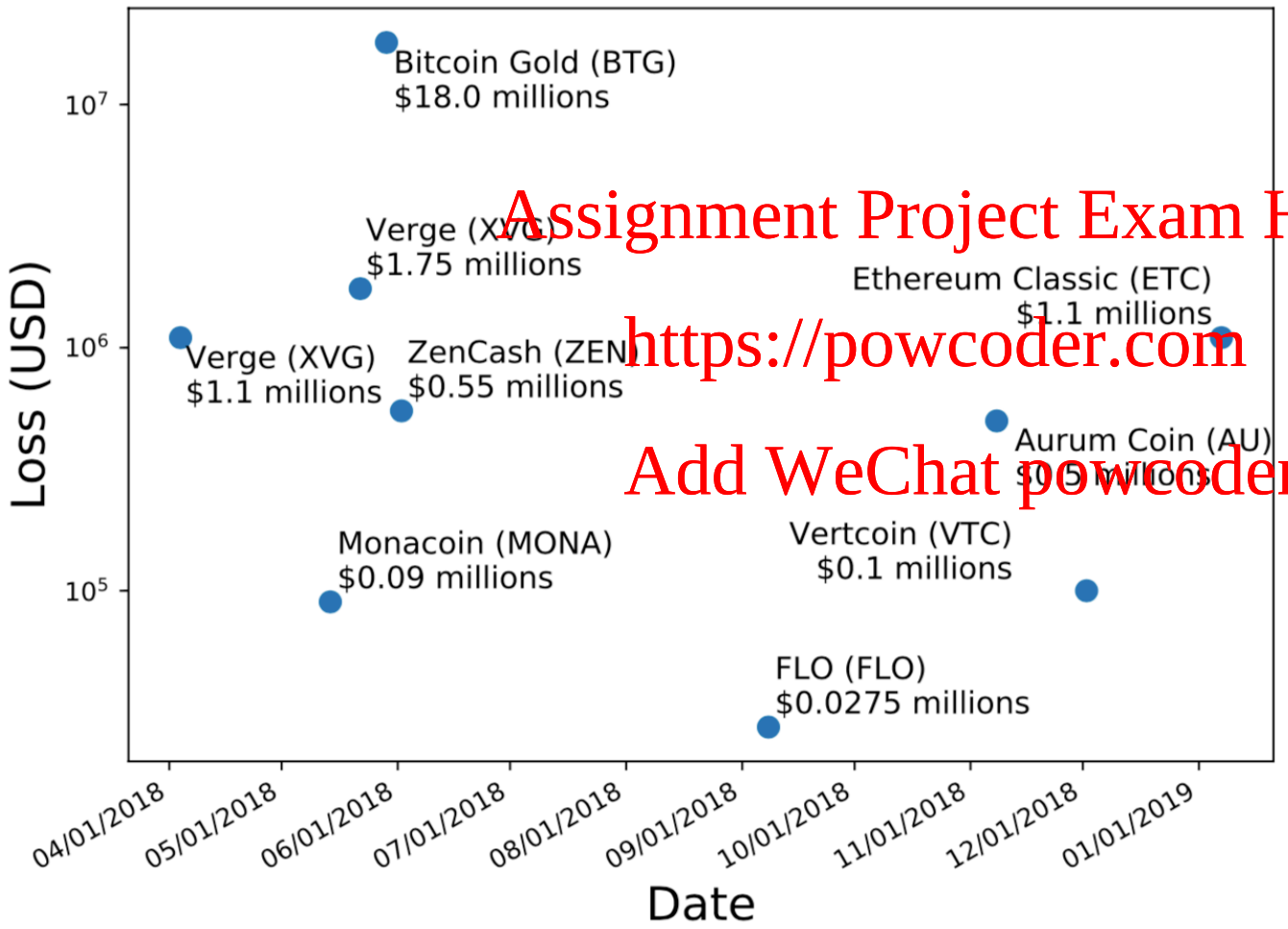
MONASH University

Is 51% attack feasible?

You need to control >51% hashing power in the world … of Bitcoin!!

Hard forks and competing coins reduce the mining power of a single chain!

51% Attacks

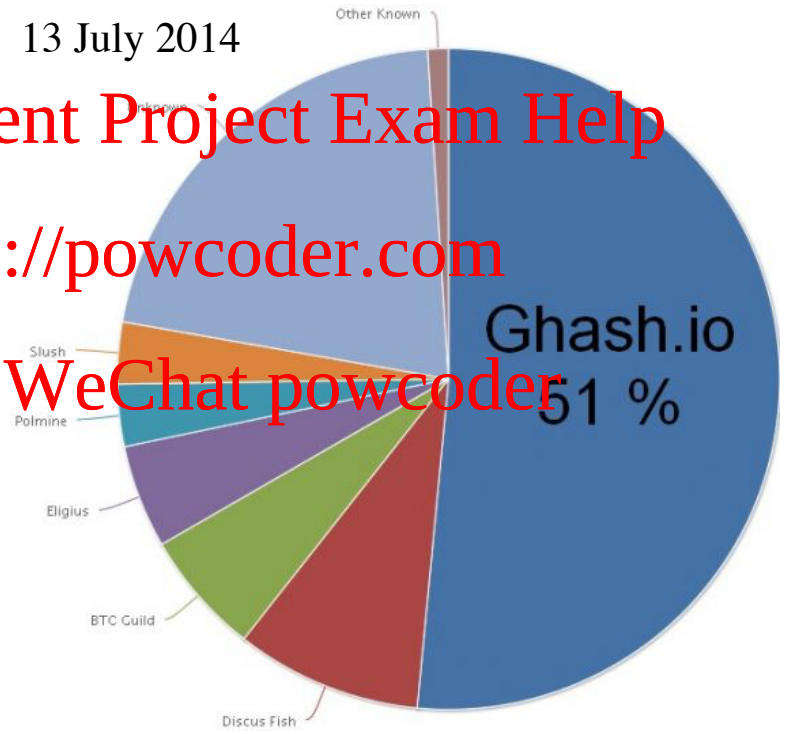


Assumption v.s. Reality

Assumption

Reality

13 July 2014



https://99bitcoins.com/wp-content/uploads/2014/06/PPVr0Wv.png

MONASH University

Mining Pool

Bitcoin mining process

Mining locally

latest block

Proof-of-Work

Mining reward

Mining Pool

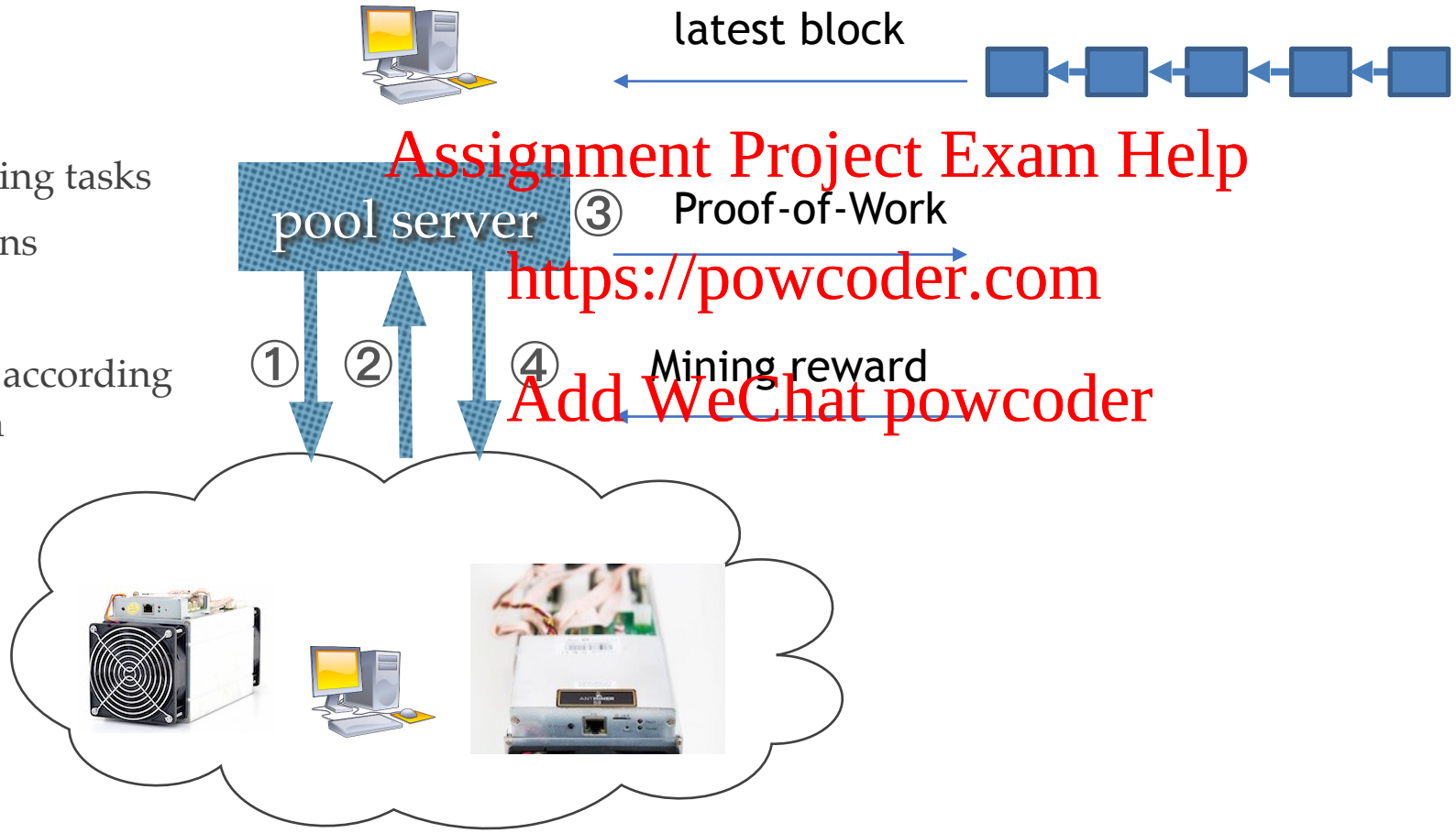
Bitcoin mining process

A single miner is likely to lose money

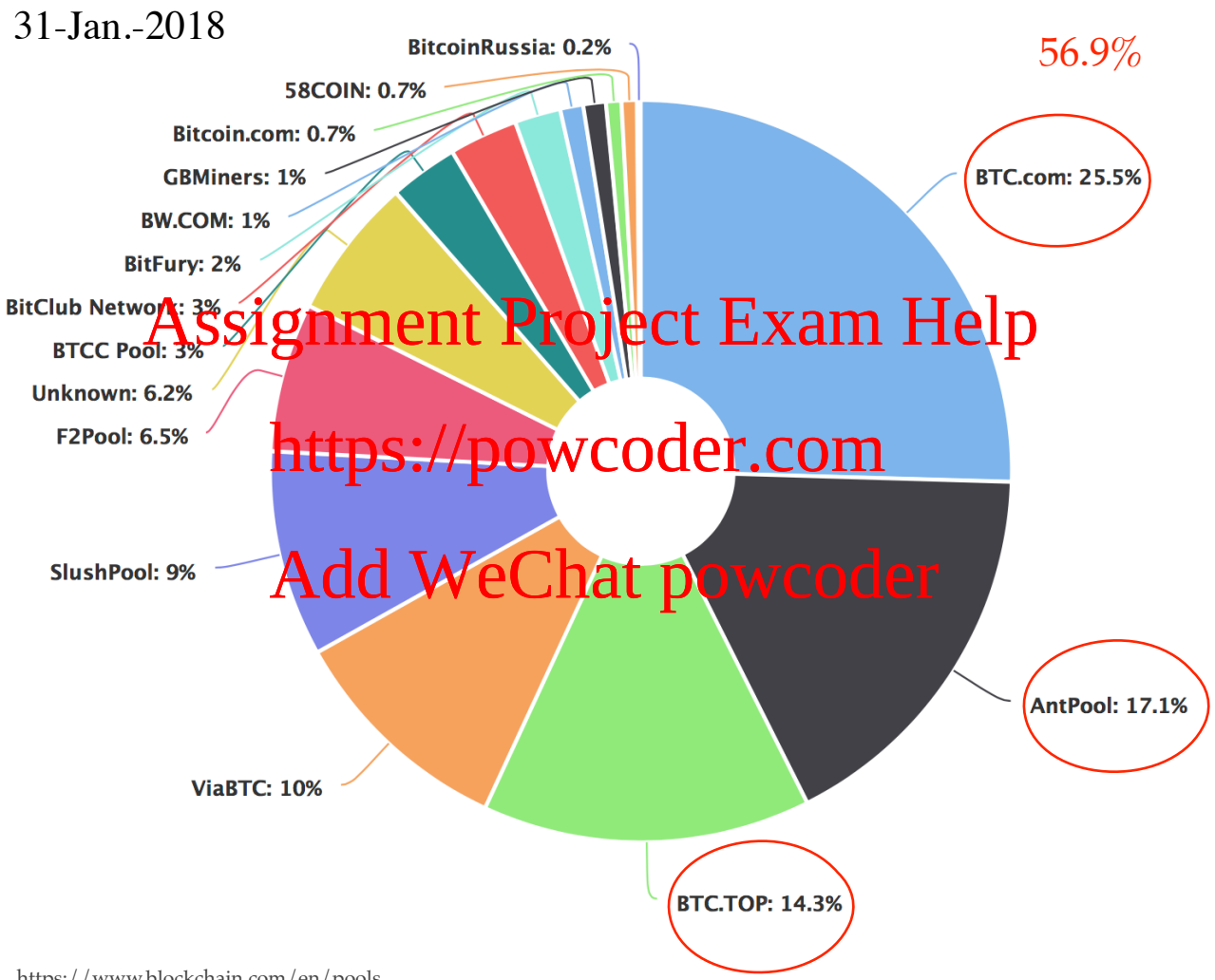
Electricity cost + machine cost > revenue

Mining Pool

① Distribute mining tasks
② Submit solutions
③ Claim rewards
④ Shares reward according to contribution



Jan 2018



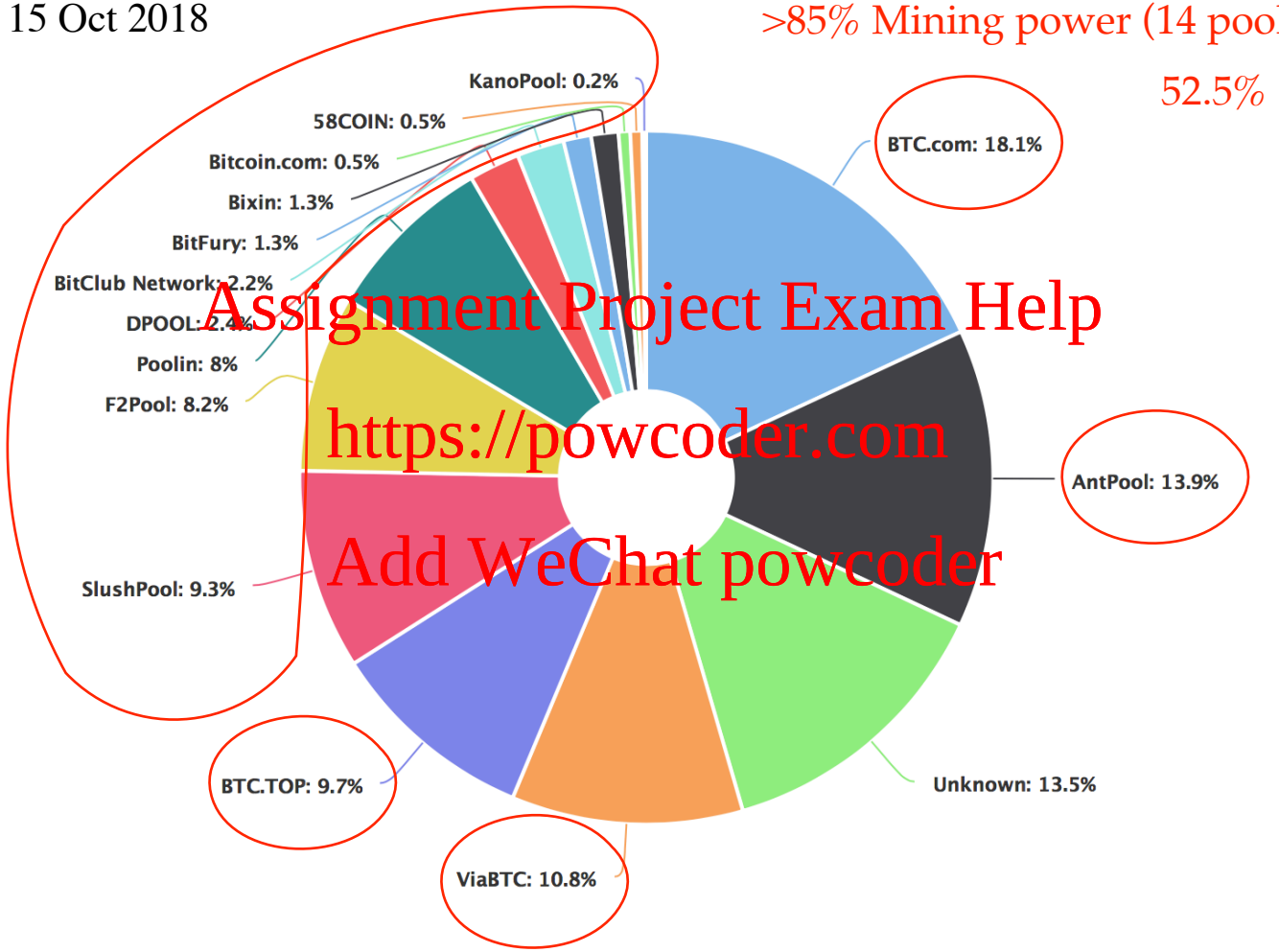
https://www.blockchain.com/en/pools

Oct 2018

15 Oct 2018

>85% Mining power (14 pools)

52.5%



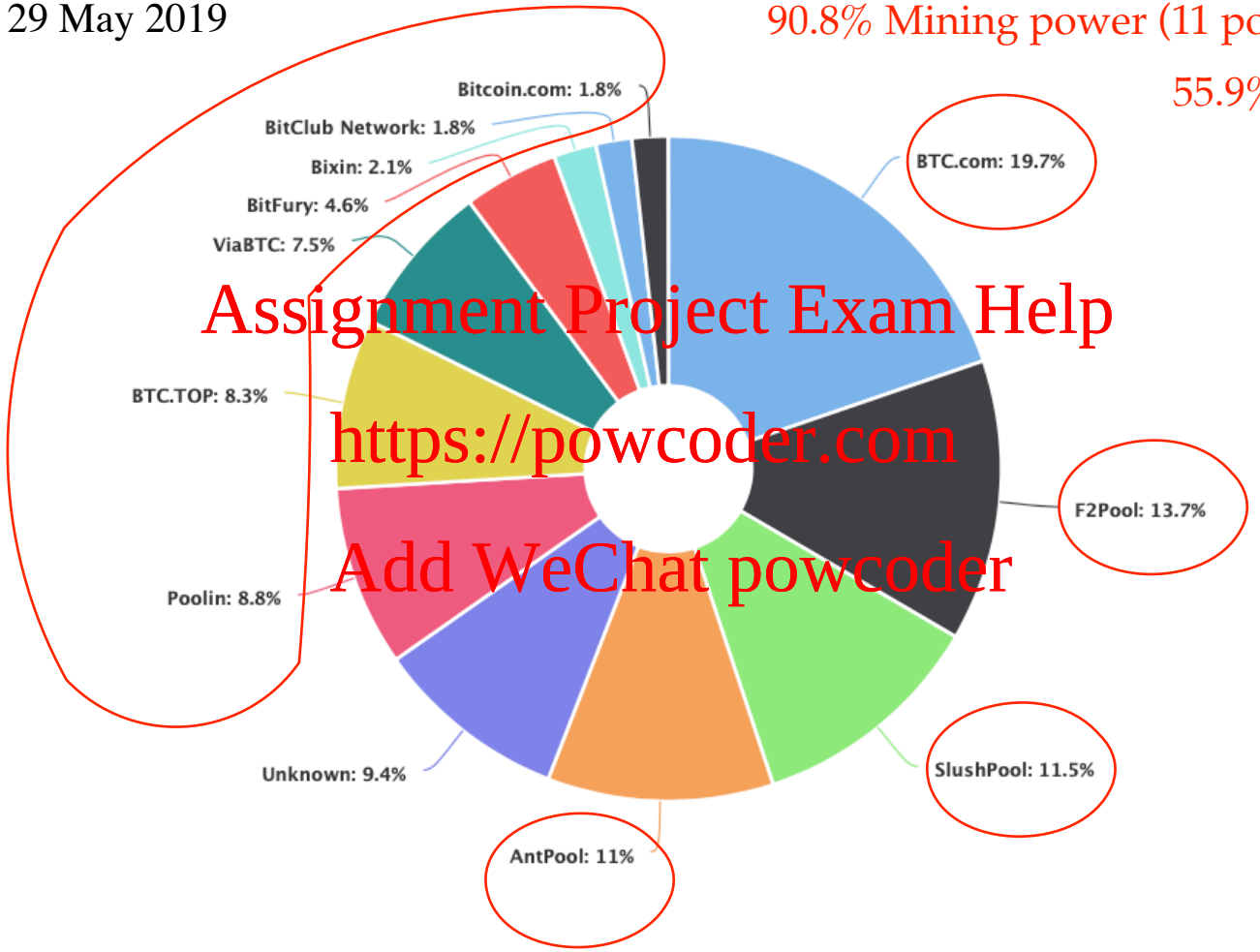
https://www.blockchain.com/en/pools

May 2019

29 May 2019

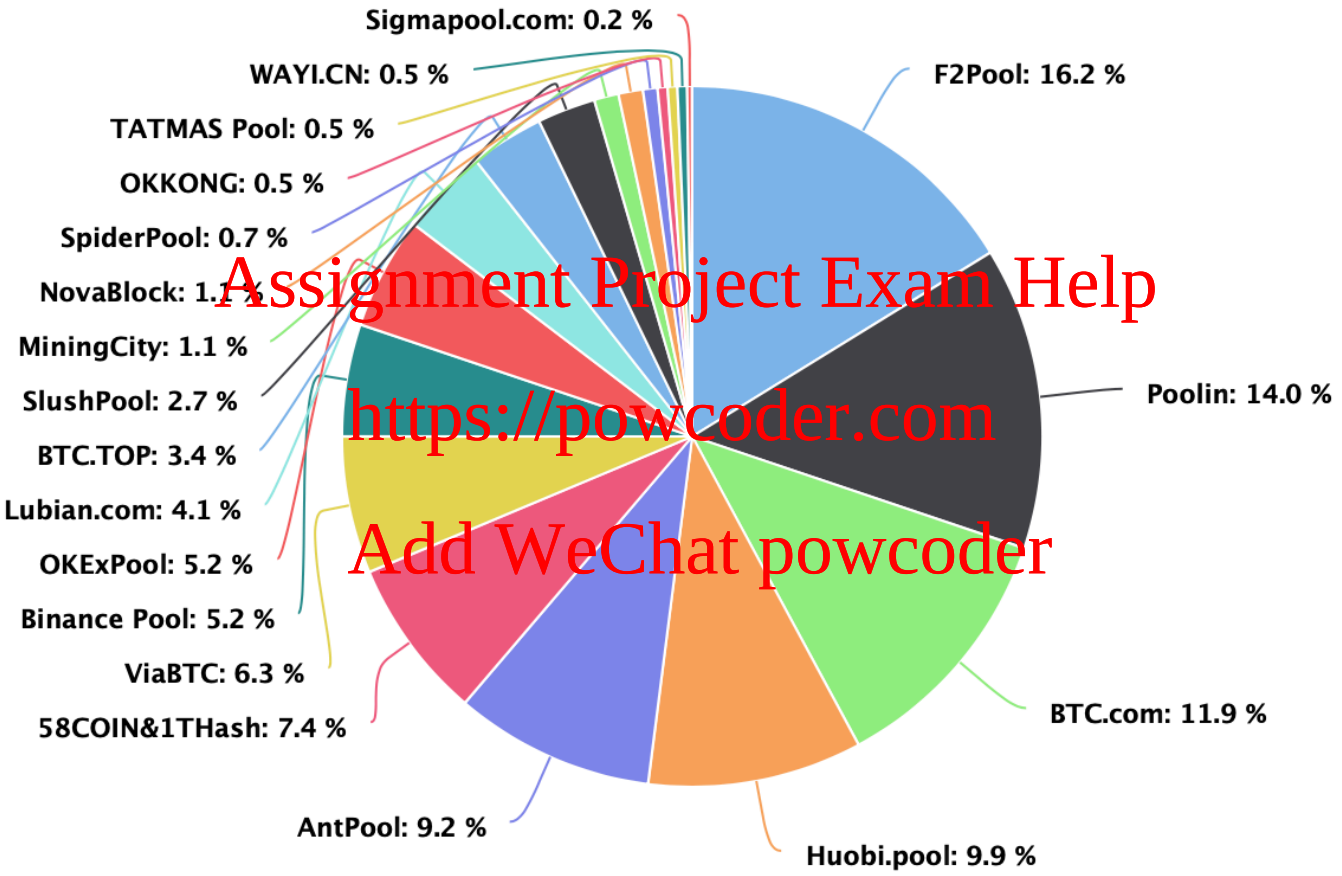
90.8% Mining power (11 pools)

55.9%

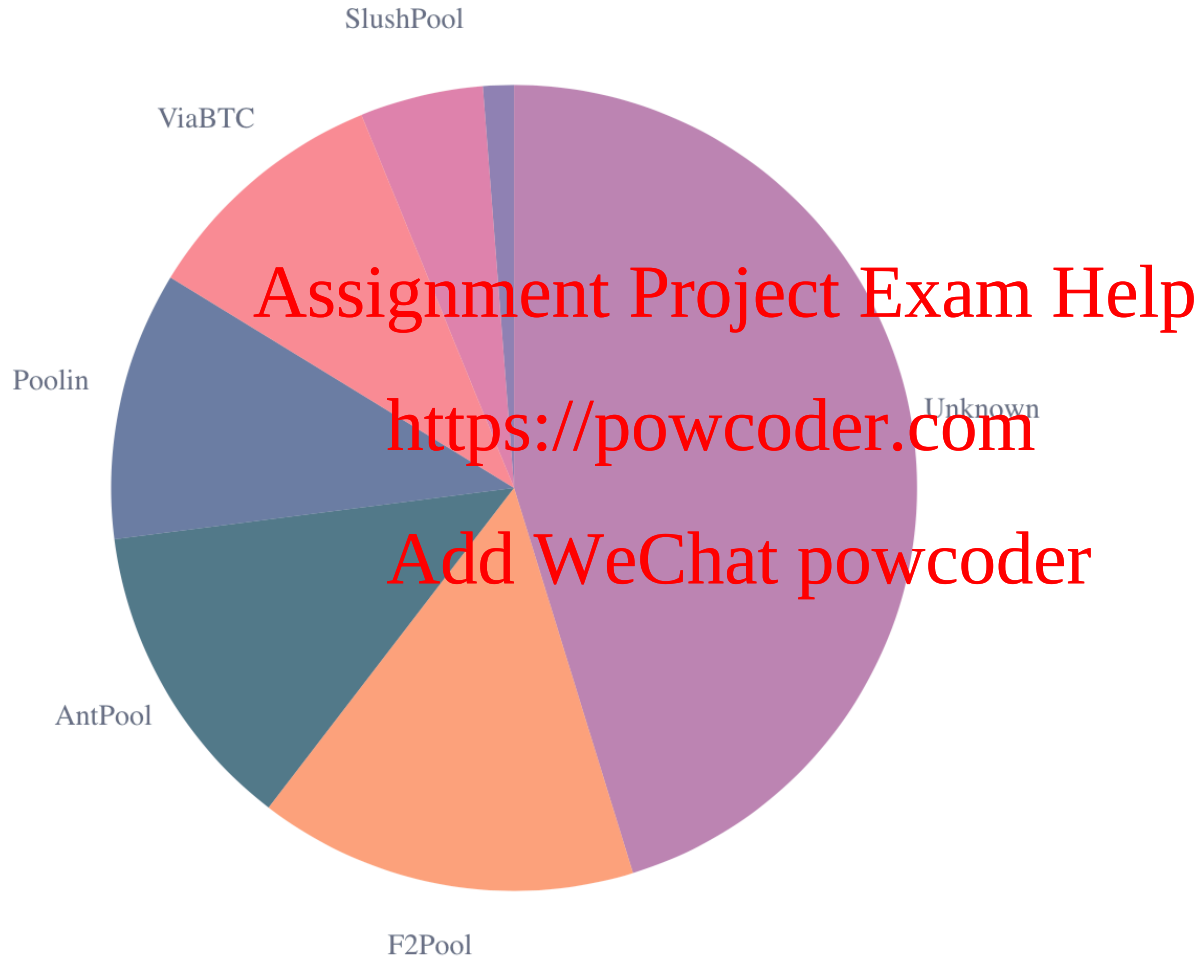


MONASH University

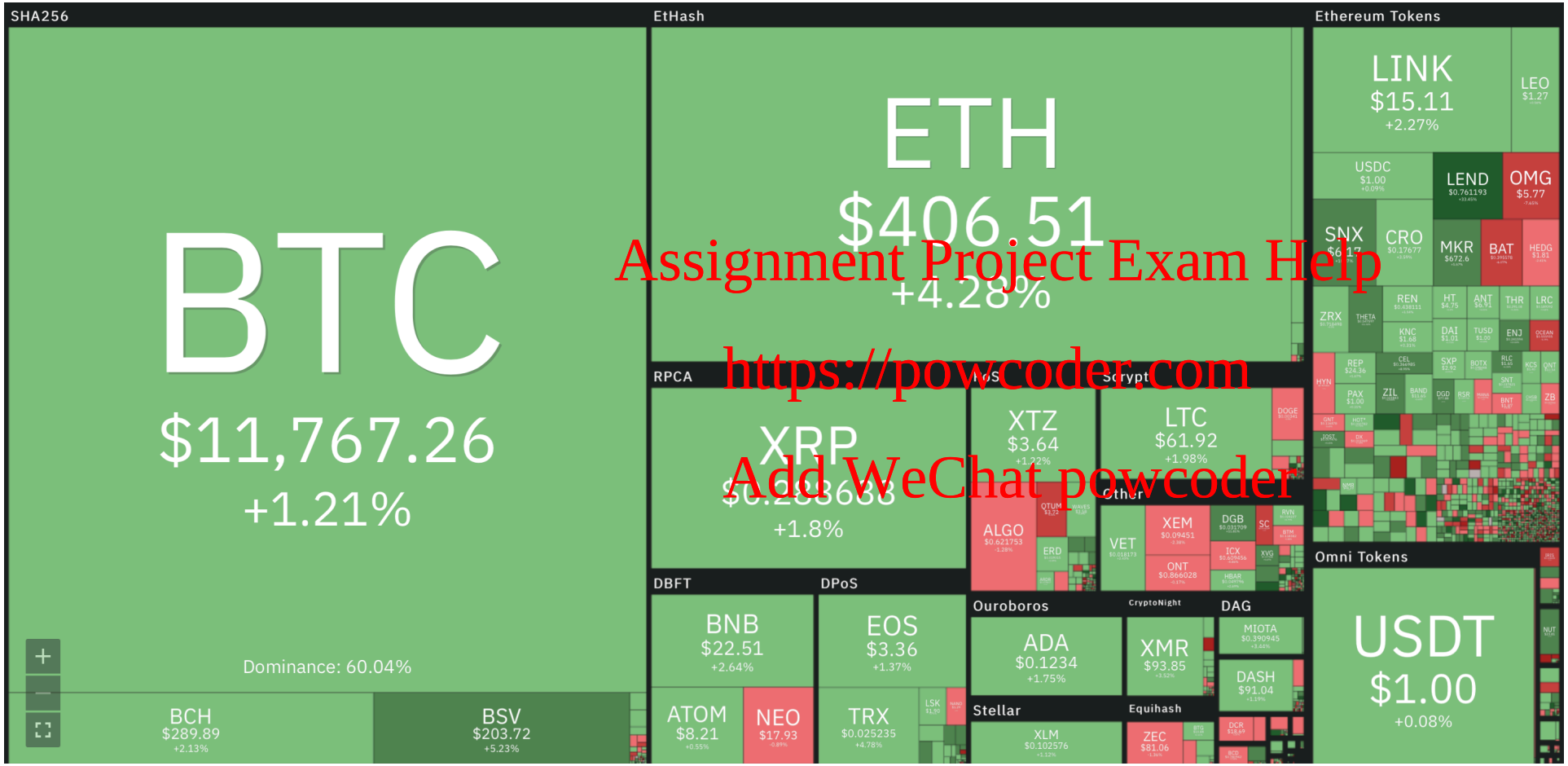
Aug 2020



Aug 2022



Is 51% attack feasible?



https://coin360.com

PoW 51% Attack Cost

PoW 51% Attack Cost

This is a collection of coins and the theoretical cost of a 51% attack on each network.

Learn More ⚡ Tip

Name	Symbol	Market Cap	Algorithm	Hash Rate	1h Attack Cost	NiceHash-able
Bitcoin	BTC	$458.80 B	SHA-256	294,753 PH/s	*$344,946*	0%
Ethereum	ETH	$229.60 B	Ethash	876 TH/s	*$994,009*	5%
Litecoin	LTC	$4.27 B	Scrypt	430 TH/s	*$58,238*	9%
BitcoinCash	BCH	$2.60 B	SHA-256	1,124 PH/s	*$5,188*	19%
BitcoinSV	BSV	$1.17 B	SHA-256	514 PH/s	*$2,374*	41%
Zcash	ZEC	$1.10 B	Equihash	8 GH/s	*$8,474*	4%
Dash	DASH	$572.78 M	X11	2 PH/s	*$1,279*	4%
BitcoinGold	BTG	$517.80 M	Zhash	2 MH/s	*$1,025*	57%

https://www.crypto51.app

MONASH University

Reading

Deconstructing Blockchains: A Comprehensive Survey on Consensus, Membership and Structure
https://arxiv.org/pdf/1908.08316.pdf

Take home message:

1. Blockchain is not as secure as you may think
2. There are many possible attacks on blockchains, including Bitcoin
3. Design details matter a lot!
4. **DO NOT LAUNCH ATTACKS!**
   (You can try attacks in a testnet, but don't do it on the real network!)
   (Ethics matters!)

Homework:

Try to find other possible attacks on Bitcoin-like system, and we will discuss next week!

Advanced attacks on blockchain next week!