IS4302
Blockchain and Distributed Ledger Technology

Lecture 1

13 Jan, 2022

NUS
National University
of Singapore

Overview

- **Course introduction**
- **Introduction to the crypto ecosystem**
- **What is Blockchain?**
  - Blockchain in its original form (Bitcoin Network)

Module details

Instructor: **Xiaofan Li**

- *E-mail: li.x@nus.edu.sg*

Teaching Assistants:

Lab sessions: Siddharth Gohil (siddharth_gohil@u.nus.edu)

Grading: Xiaoyu Liu (lxiaoyu@u.nus.edu)

Class participation

- **Poll Everywhere**
  - pollev.com/is4302
  - Join with your NUS email and account!
  - Important for your class participation score!

- **Quizzes and surveys will be operated through this platform.**

Learning objectives

- **What is the definition of blockchain and distributed ledger technologies?**

- **How do they technically work?**

- **How to think about the business, economic, and financial performance of a blockchain application?**

- **How do such performances relate to their technical setup?**

Learning objectives

- **In the first half the module, we will learn how to build an application on Ethereum**

- **In the second half, we will try to analyse the features of blockchain applications.**

- **By the end of the module, you will be able to build simple blockchain applications and understand existing blockchain applications from deeper perspectives.**

Tentative weekly schedule

Week	Lecture	Lab	Submission
1	➢ Course intro ➢ Blockchain basics	Lab 1: Intro to javascript (self-learning)	
2	➢ Solidity language basics		
3	➢ Web3 and ERC standards	Lab 2: Remix IDE, solidity 1	
4	➢ Ethereum clients, security	Lab 3: Truffle, solidity 2	Lab 2 assignment
5	➢ Blockchain application ideation	Lab 4: Eth client, solidity 3	Lab 3 assignment
6	➢ No lecture ➢ Group project consultation	Group project consultation	Lab 4 assignment
Recess week			

Tentative weekly schedule

7	**Mid-term assignment**		
8	➢ The oracle problem		
9	➢ Commonly used patterns in smart contracts		Writing assignment 1
10	➢ Cryptocurrencies and Non-fungible tokens		Writing assignment 2
11	➢ DAO and off-chain governance of public blockchains		Writing assignment 3
12	➢ Group project presentation		Architecture and design document Github code
13	➢ Group project presentation		

Assessment

- **Class participation (5%)**
  - Polleverywhere quizzes

- **Assignments (30%)**
  - Lab assignment (6%*3=18%)
  - Writing assignment (4%*3=12%)

- **Midterm test (30%)**
  - Similar to lab assignments (but more complicated)

Assessment

- **Group project (35%)**
  - Architecture and design document (10%)
  - Final presentation and demo (15%)
  - Peer review (5%)
  - GitHub code (graded on implementation and code quality) (5%)

Group project

- **Topic: build a blockchain application**
  - More details will be discussed in week 5 lecture

- **Group of 5-6 students**

- **Formed by week 4 lecture.**

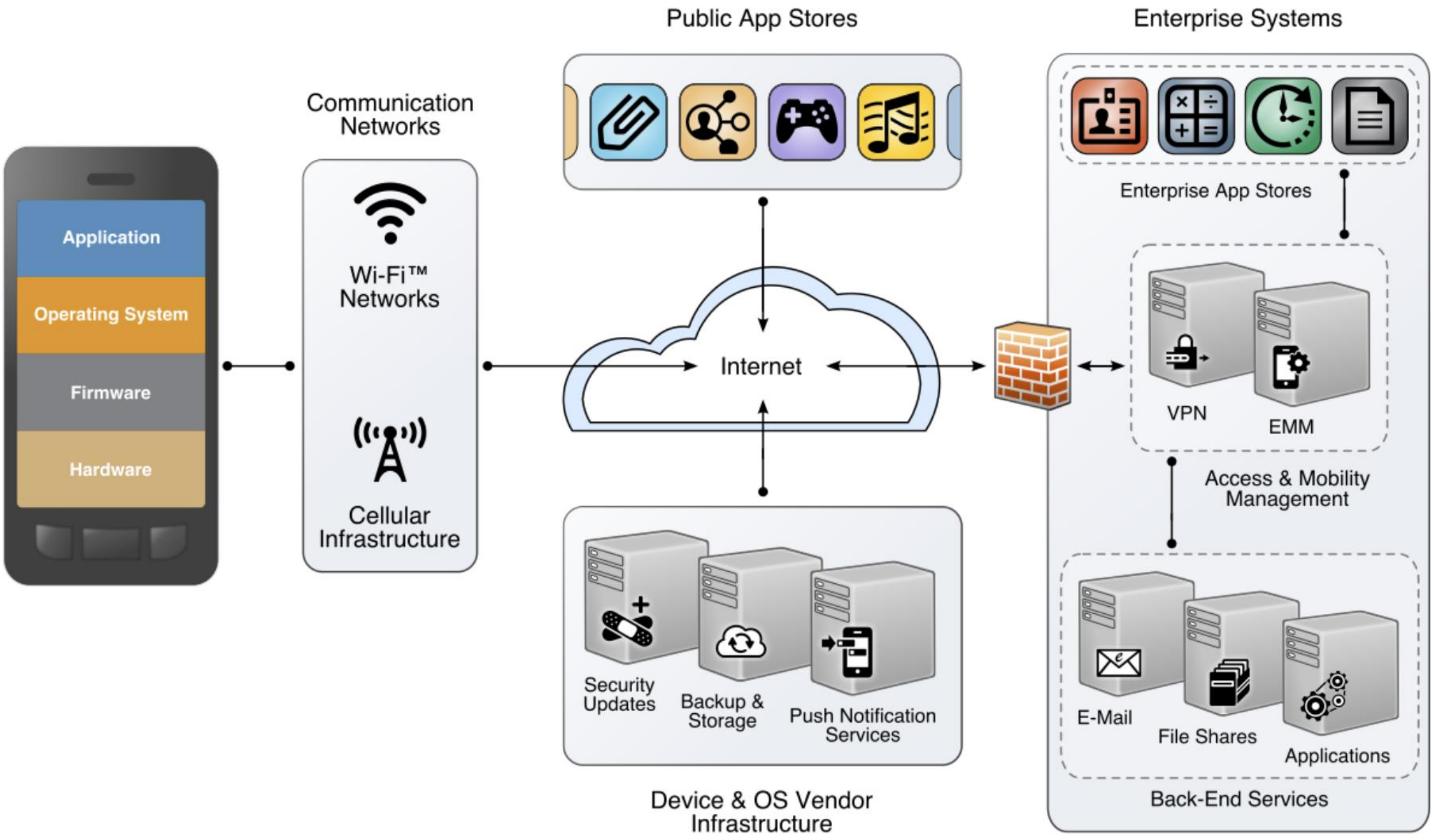
- **Will send google sheets to register.**

- **Telegram:** https://t.me/+L0_gQVehjbcxOWRl

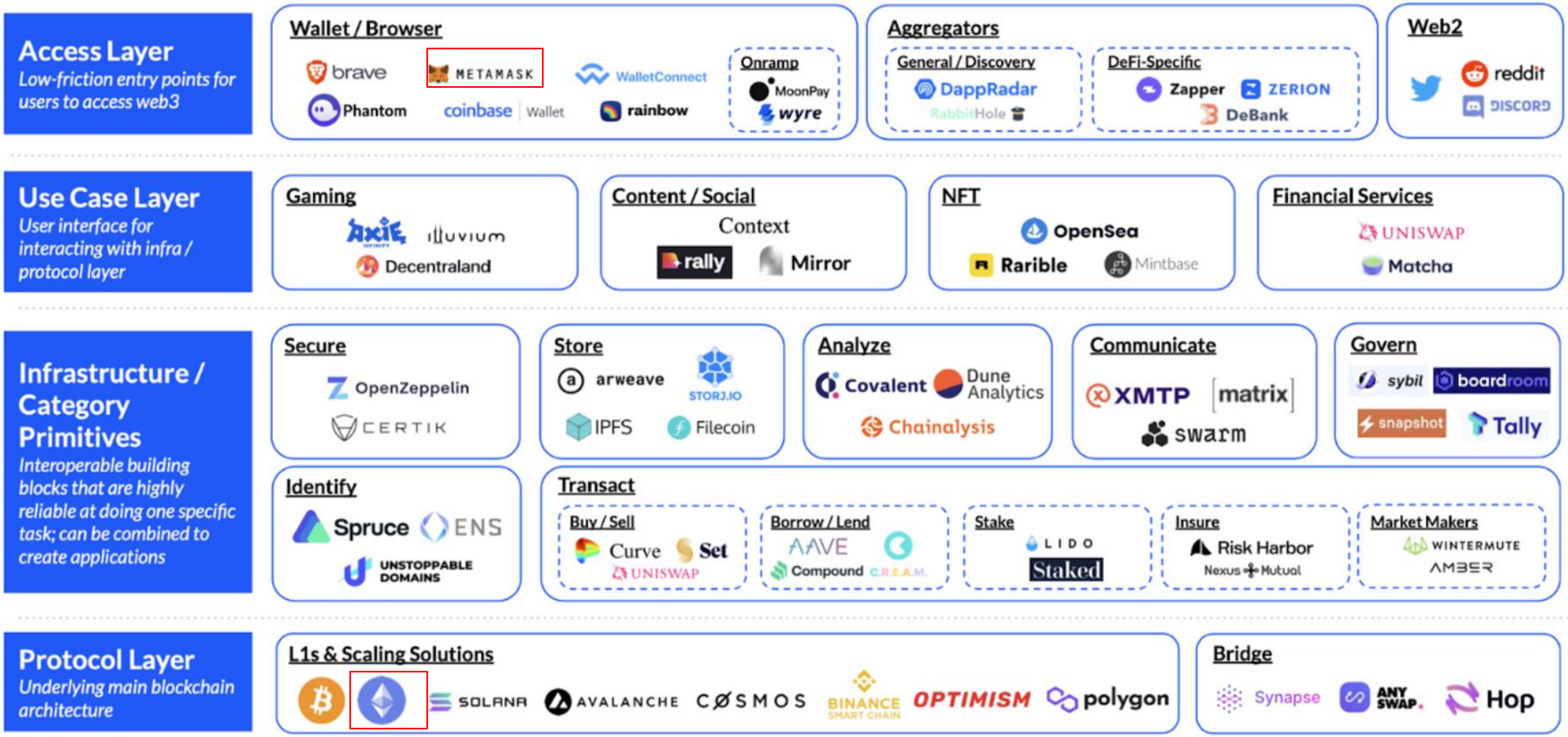
Overview

- **Course introduction**
- **Introduction to the crypto ecosystem**
- **What is Blockchain?**
  - Blockchain in its original form (Bitcoin Network)

Mobile ecosystem



Web3 ecosystem



Access Layer
Low-friction entry points for users to access web3

Wallet / Browser
- brave
- METAMASK
- WalletConnect
- Phantom
- coinbase | Wallet
- rainbow

Onramp
- MoonPay
- wyre

Aggregators

General / Discovery
- DappRadar
- RabbitHole

DeFi-Specific
- Zapper
- ZERION
- DeBank

Web2
- reddit
- DISCORD

Use Case Layer
User interface for interacting with infra / protocol layer

Gaming
- Axie Infinity
- illuvium
- Decentraland

Content / Social
- Context
- rally
- Mirror

NFT
- OpenSea
- Rarible
- Mintbase

Financial Services
- UNISWAP
- Matcha

Infrastructure / Category Primitives
Interoperable building blocks that are highly reliable at doing one specific task; can be combined to create applications

Secure
- OpenZeppelin
- CERTIK

Store
- arweave
- STORJ.IO
- IPFS
- Filecoin

Analyze
- Covalent
- Dune Analytics
- Chainalysis

Communicate
- XMTP
- [matrix]
- swarm

Govern
- sybil
- boardroom
- snapshot
- Tally

Identify
- Spruce
- ENS
- UNSTOPPABLE DOMAINS

Transact

Buy / Sell
- Curve
- Set
- UNISWAP

Borrow / Lend
- AAVE
- Compound
- C.R.E.A.M.

Stake
- LIDO
- Staked

Insure
- Risk Harbor
- Nexus Mutual

Market Makers
- WINTERMUTE
- AMBER

Protocol Layer
Underlying main blockchain architecture

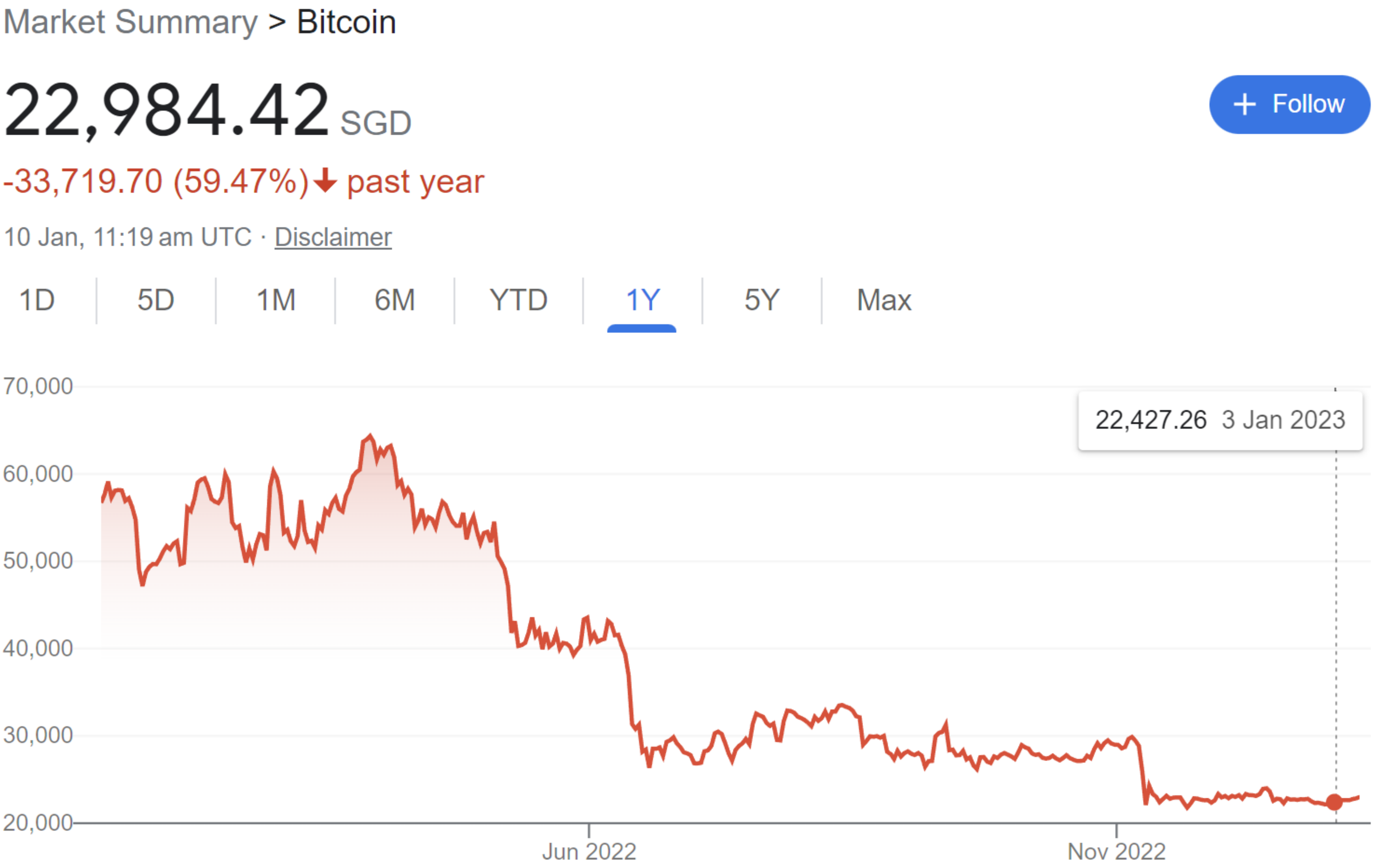
L1s & Scaling Solutions
- SOLANA
- AVALANCHE
- COSMOS
- BINANCE SMART CHAIN
- OPTIMISM
- polygon

Bridge
- Synapse
- ANY SWAP.
- Hop

Cryptocurrencies

- **Usage of applications on public blockchains requires its native cryptocurrencies**
  - Bitcoin network --- Bitcoin
  - Ethereum --- Ether
  - Solana --- Solana
  - …

- **Fundamental demand/value for these cryptocurrencies comes from the demand for the applications**



Market Summary > Bitcoin

22,984.42 SGD

-33,719.70 (59.47%)↓ past year

10 Jan, 11:19 am UTC · Disclaimer

The First Crash

- **Luna and TerraUSD (UST): two native tokens of the Terra network**

- **UST: supposed to be a stablecoin pegged to $1 USD**

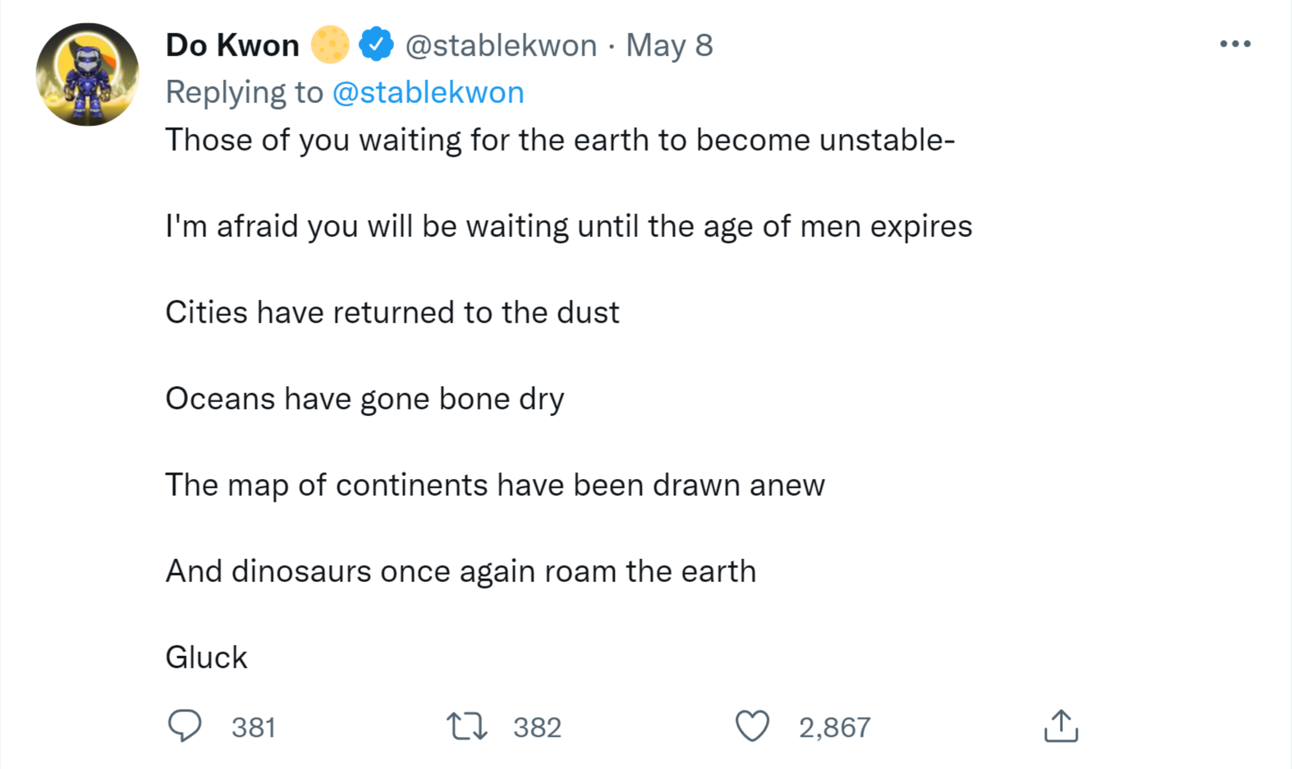
- **Instead of relying on a reserve of assets, UST is an algorithmically stabilized coin.**
  - The algorithm keeps the price of UST anchored to $1 by burning LUNA to mint new UST, or the other way.

The First Crash

- **Why does Luna have a value?**
- **An application on the Terra network provides a high interest rate (~12%) for Luna.**
- **How should this interest be paid?**
- **How is this interest paid?**

The First Crash

- **May 7: Signs of capital flight from UST**
  - an 85 million UST swap for 84.5 million USDC
- **The bank run started.**



Do Kwon @stablekwon · May 8

Replying to @stablekwon

Those of you waiting for the earth to become unstable-

I'm afraid you will be waiting until the age of men expires

Cities have returned to the dust

Oceans have gone bone dry

The map of continents have been drawn anew

And dinosaurs once again roam the earth

Gluck

381 382 2,867

Some thoughts

- **In a healthy financial system:**
  - your money + my money + technology/idea/… = productivity

- **In an unhealthy financial system:**
  - your money + my money + marketing/speculation/… = 0

- **A blockchain system's integrity depends on many factors, especially including many human and social factors.**

Overview

- **Course introduction**

- **Introduction to the crypto ecosystem**

- **What is Blockchain?**
  - Blockchain in its original form (Bitcoin Network)

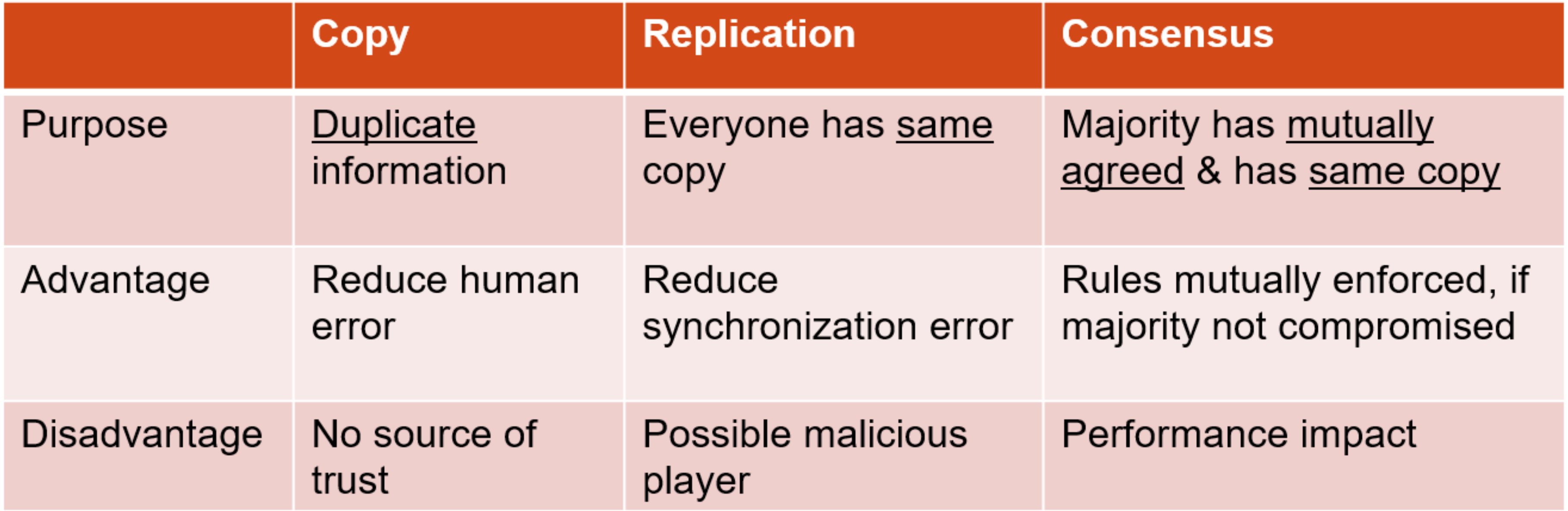
What is blockchain?

- **Blockchain is a ledger writing technology, that can record**
  - Transactions
  - Inventory
  - Personal data
  - …
  - Any information you want to record

- **Distributed ledger with consensus**

What is blockchain?

- **Copied vs Replicated vs Replicated via consensus**



	Copy	Replication	Consensus
Purpose	Duplicate information	Everyone has same copy	Majority has mutually agreed & has same copy
Advantage	Reduce human error	Reduce synchronization error	Rules mutually enforced, if majority not compromised
Disadvantage	No source of trust	Possible malicious player	Performance impact

What are the features of blockchains?

- **3 main properties of blockchain:**
  - Distributed ledger – everyone has the same info
  - Consensus – everyone(majority) agreed to the same info
  - Immutability – no one can erase a transaction previously agreed upon

- **Really? To what extent?**

Are blockchains immutable?

- **There was a successful coordinated 51% attack on Bitcoin.**
    - However, it is not for malicious reasons but instead to ensure the integrity of the Bitcoin Network.
- **When version 0.8 of the beta Bitcoin client software was released, most miners upgraded to the latest version, but most users did not.**
- **Due to a change in the code, the 0.8 software recognized a block of transactions that the 0.7 software did not. This discrepancy caused the two different versions of the software client to use different chains of transactions.**

Are blockchains immutable?

- **Vitalik Buterin (2014), a prominent member of the Bitcoin community, noted that to make sure that everyone used the same blockchain, the “mining pool operators came together on IRC chat” and decided that they had to intentionally cause a 51% attack in order to resolve the fork.**

- **Blockchains are not “that” immutable.**

Blockchain in its original form

- **2008 Nov: Paper by Satoshi Nakamoto**
  “Bitcoin: A Peer-to-Peer Electronic Cash System”
  - 2009 Bitcoin initial release
  - 2009-01-12 Genesis block
  - 2010-05-22 First real BTC payment (10k BTC for a pizza)

Blockchain in its original form

- **Bitcoin: A Peer-to-Peer Electronic Cash System by *Satoshi Nakamoto***
  - Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments.
  - Suffers from the inherent weaknesses of the trust based model.
  - The cost of mediation increases transaction costs.
  - These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

Satoshi Nakamoto’s proposal

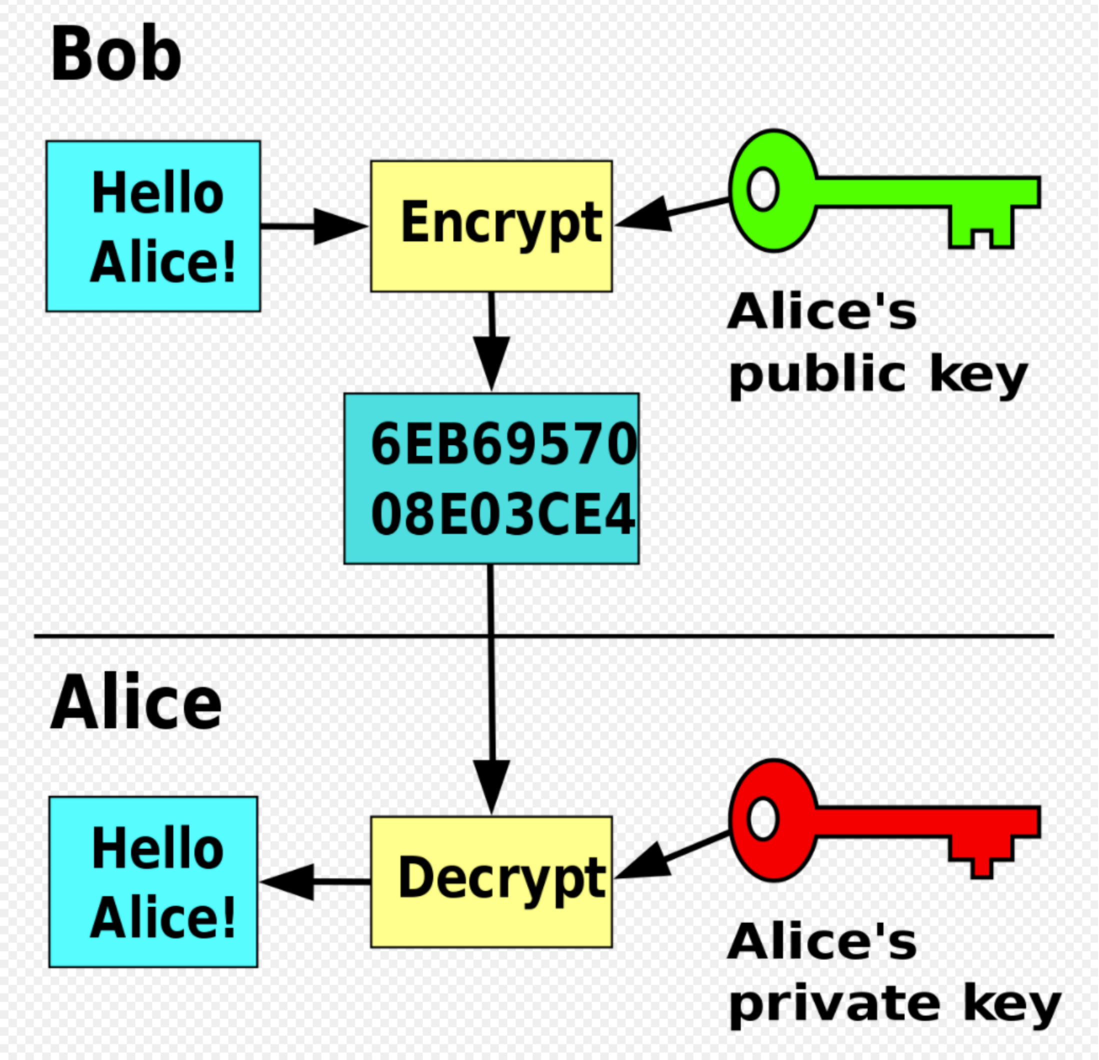
- **An electronic payment system based on cryptographic proof instead of trust.**
- **Two willing parties can transact directly with each other without the need for a trusted third party.**
- **Transactions that are computationally impractical to reverse would protect sellers from fraud.**
- **Routine escrow mechanisms could easily be implemented to protect buyers.**

Public-key cryptography

- **Asymmetric key algorithms**
- **The keys work in pairs**
  - A pair of keys, A and B
  - What encrypted by A can be decrypted by B but not A
  - What encrypted by B can be decrypted by A but not B
- **Public key and private key**
  - For each pair of keys, one is kept secret (private key) and the other is publicly known (public key).

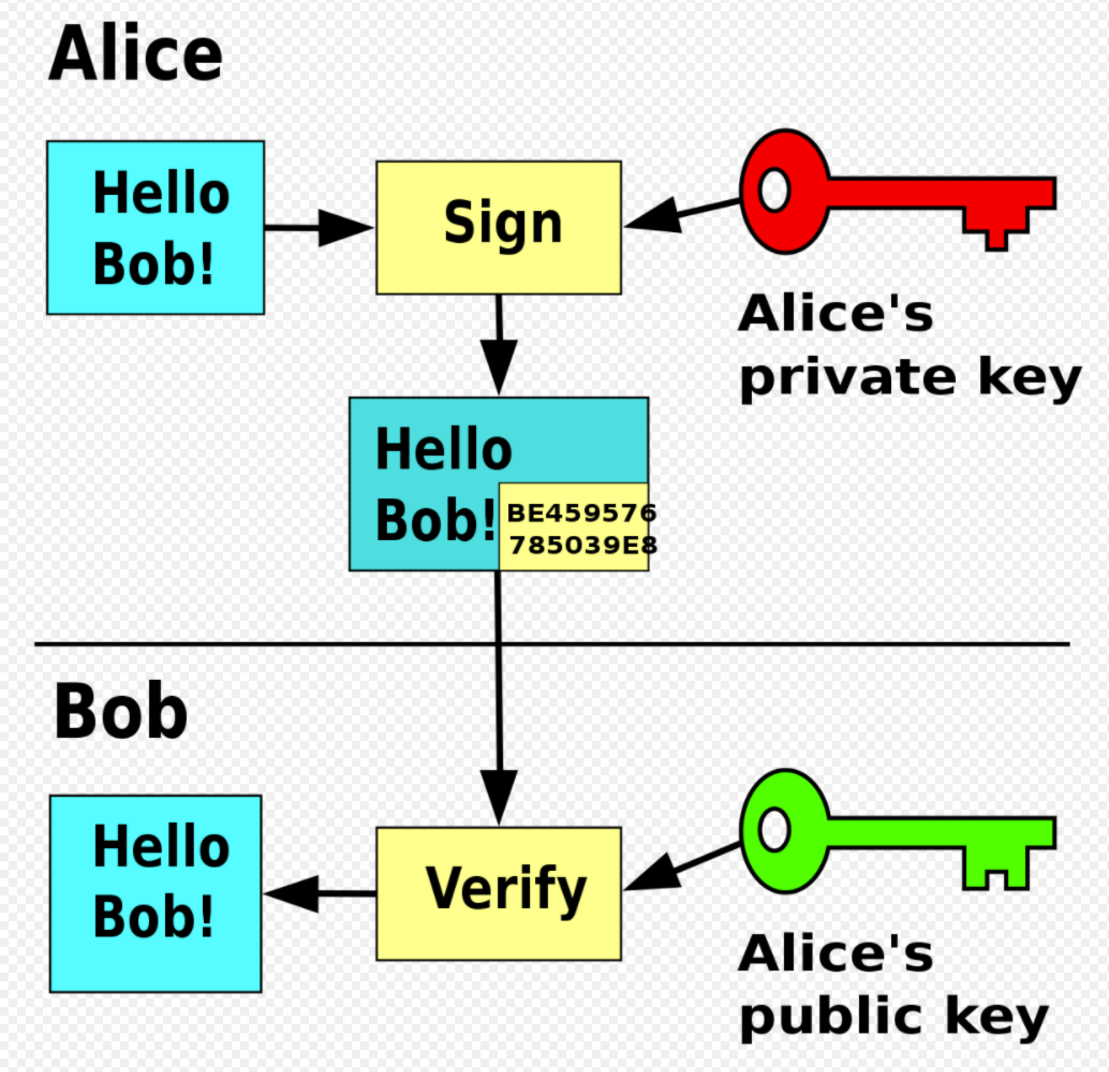
Public-key cryptography

- **Public key encryption**
  - A message is encrypted with a recipient's public key.
  - Anyone can encrypt messages using the public key, but only the holder of the paired private key can decrypt.



Public-key cryptography

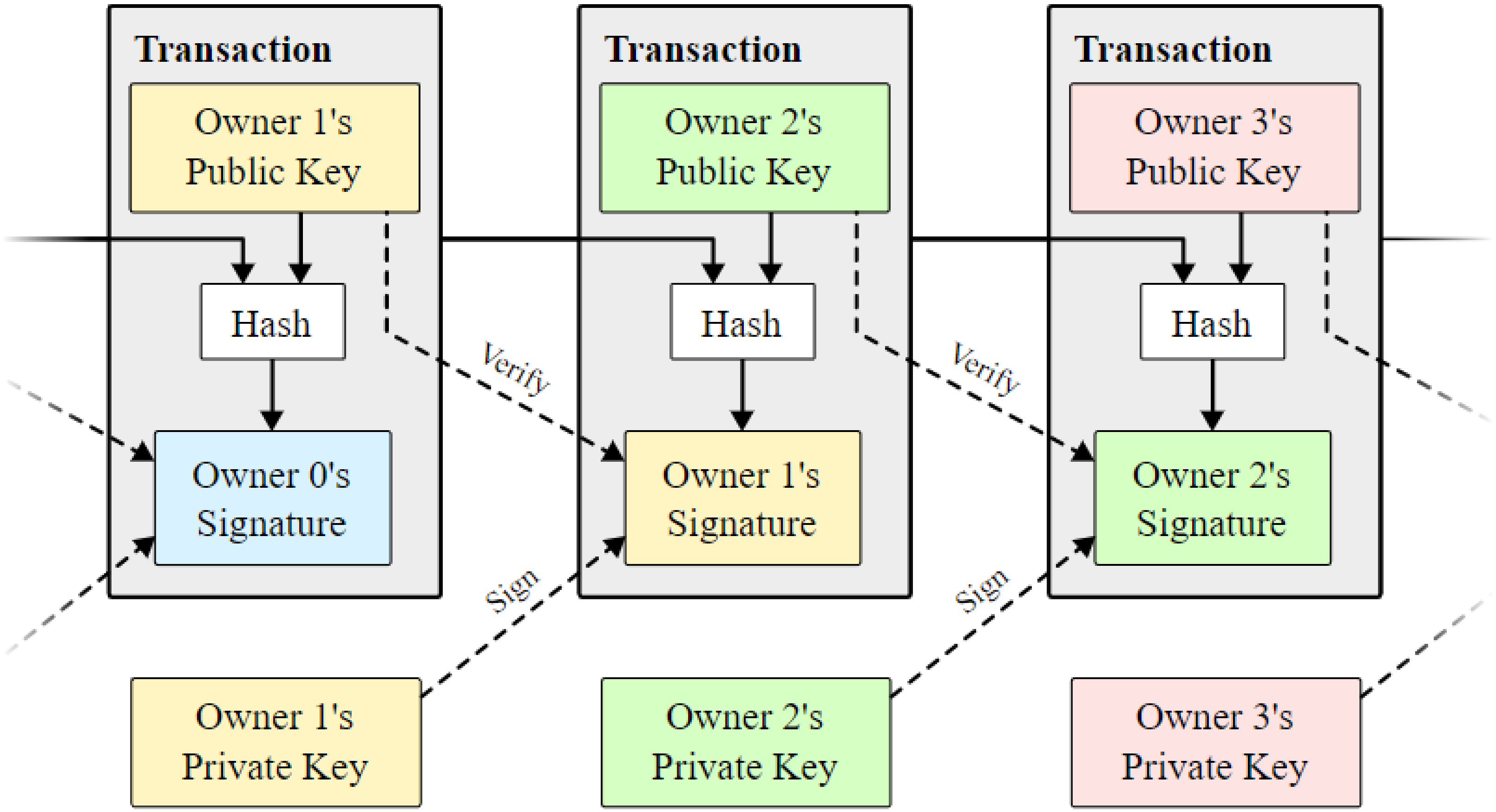
- **Digital signature**
  - A message is signed with the sender's private key and can be verified by anyone who has access to the sender's public key.
  - Ensures that the message has not been tampered with, as a signature is mathematically bound to the message it originally was made with.



Bitcoin's transaction

- **Bitcoin is defined as a chain of digital signatures, that records the list of owners.**
- **Each owner transfers the coin to the next by digitally signing (a hash of) the previous transaction and the public key of the next owner and adding these to the end of the coin.**
  - Hash: a function that converts one value to another for various reasons. Mainly for compression here.
- **A payee can verify the signatures to verify the chain of ownership.**

Bitcoin’s transaction



Double spending problem

- **The payee can’t verify that one of the owners did not double-spend the coin.**
- **A common solution is to introduce a trusted central authority that checks every transaction for double spending.**
  - Banks
- **We need a way for the payee to know that the previous owners did not sign any earlier transactions.**
  - For our purposes, the earliest transaction is the one that counts, so we don’t care about later attempts to double-spend.

Double spending problem

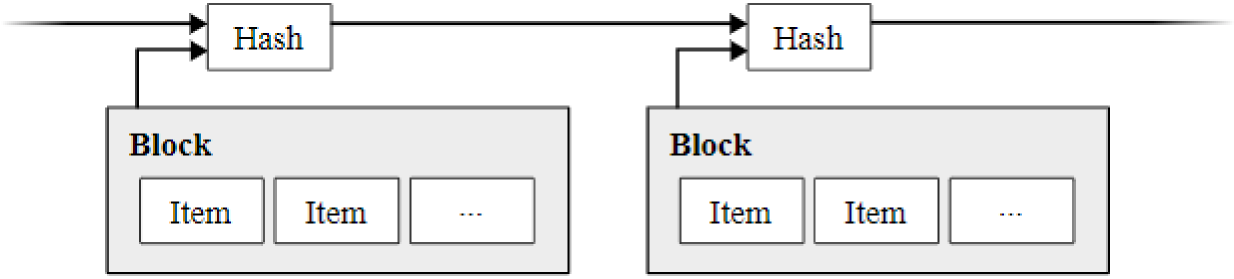
- **The only way to confirm the absence of a transaction is to be aware of all transactions.**

- **To accomplish this without a trusted party, transactions must be publicly announced.**

- **Need a system for participants to agree on a single history of the order in which they were received.**

Timestamp Server

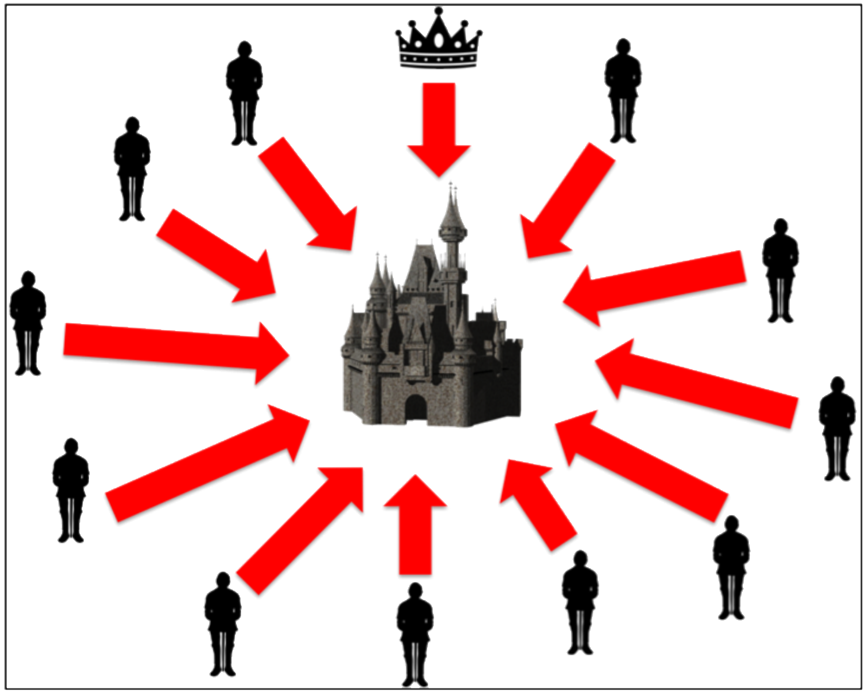
- **A timestamp server works by taking a hash of a block of items to be timestamped and widely publishing the hash.**
- **The timestamp proves that the data must have existed at the time, obviously, in order to get into the hash.**
- **Each timestamp includes the previous timestamp in its hash, forming a chain, with each additional timestamp reinforcing the ones before it.**



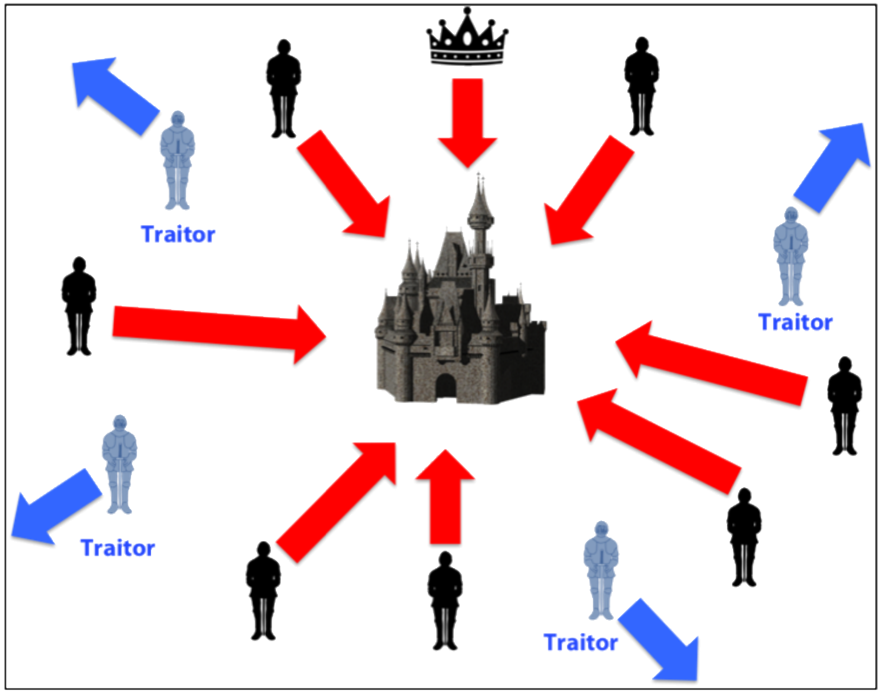
Consensus algorithm

- **Who runs the timestamp server?**
  - A distributed timestamp server on a peer-to-peer basis.
  - The people who run the timestamp server are known as miners or writers of the ledger.
- **What if there are disagreements among the writers?**
  - Validity of transactions.
  - Timing of transactions.
  - …
- **Consensus algorithm**
  - Determines how disagreements are resolved.

Byzantine General Problem

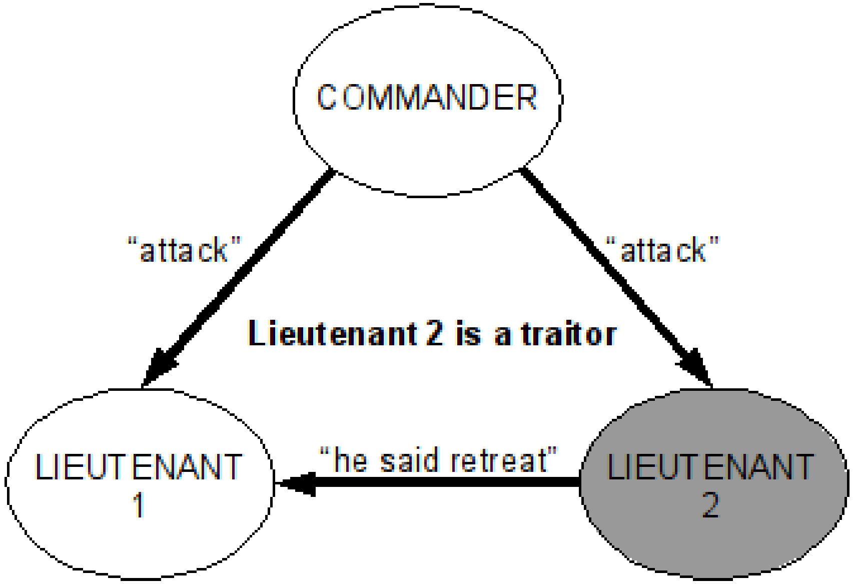


Coordinated Attack Leading to Victory

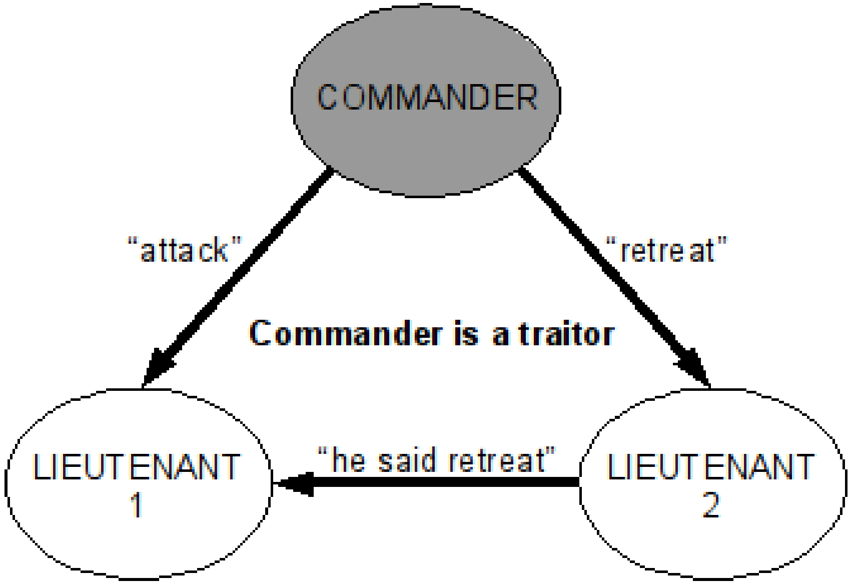


Uncoordinated Attack Leading to Defeat

Byzantine Fault Tolerance



Lieutenant 2 is a traitor



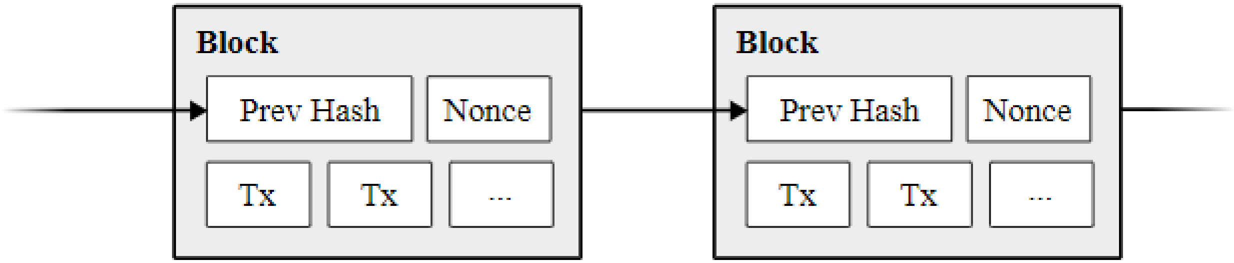
Commander is a traitor

Proof-of-work system

- **Analogous to a majority voting system.**
  - Vote not counted by heads but the amounts of computational power that supports various arguments in a disagreement.
- **To implement this, we need a puzzle that is moderately hard (yet feasible) to solve but easy to check.**
  - Calculating a hash value is easy.
  - Reverting a hash function can be computationally not feasible.
  - Find an input, the hash of which is within a range. This range can be changed to tune the difficulty of the puzzle.

Proof-of-work system

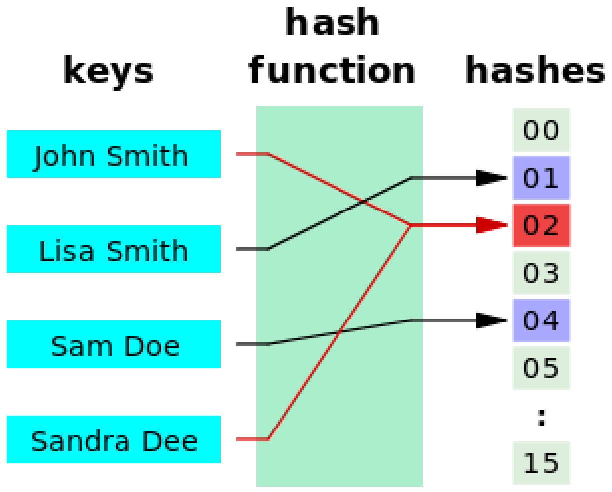
- **Increment a nonce in the block until a value is found that gives the block's hash within the required range.**



- **Finding the nonce takes computational effort.**
  - Whoever finds the nonce first gets the power to write the next block.
  - The likelihood of finding a nonce first is proportional to the computational power (hash rate).

Cryptographic hash

- **What is a cryptographic hash?**
  - Special class of hash function suitable for use in cryptography.
  - It is a designed to be a one-way function => a function which is infeasible to invert.

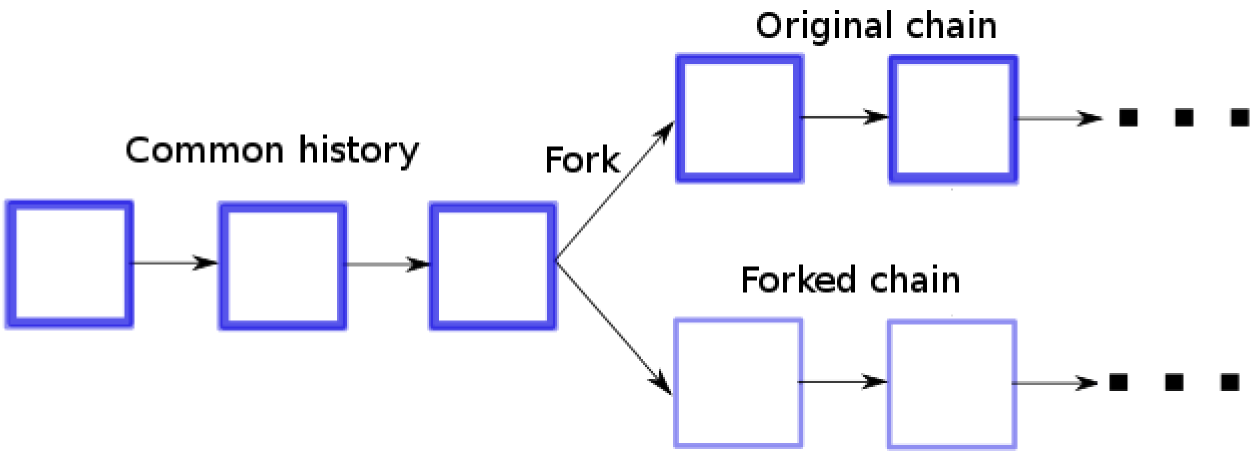


Network

- **New transactions are broadcast to all nodes.**
- **Each node collects new transactions into a block.**
- **Each node works on finding the nonce for its block.**
- **When a node finds a nonce, it broadcasts the block (nonce) to all nodes.**
- **Nodes accept the block only if all transactions in it are valid and not already spent.**
- **Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.**

Consensus algorithm

- **Forking:**



- **The longest chain is considered to be the correct one.**
- **(Honest) nodes will keep working on extending it.**

Incentive

- **The first transaction in a block is a special transaction that starts a new coin owned by the creator of the block.**
  - Only the creator of blocks in the correct fork can receive the coin rewards.
- **A way to initially distribute coins into circulation.**
  - No central authority to issue them.
- **The steady addition of a constant of amount of new coins is analogous to gold miners expending resources to add gold to circulation.**

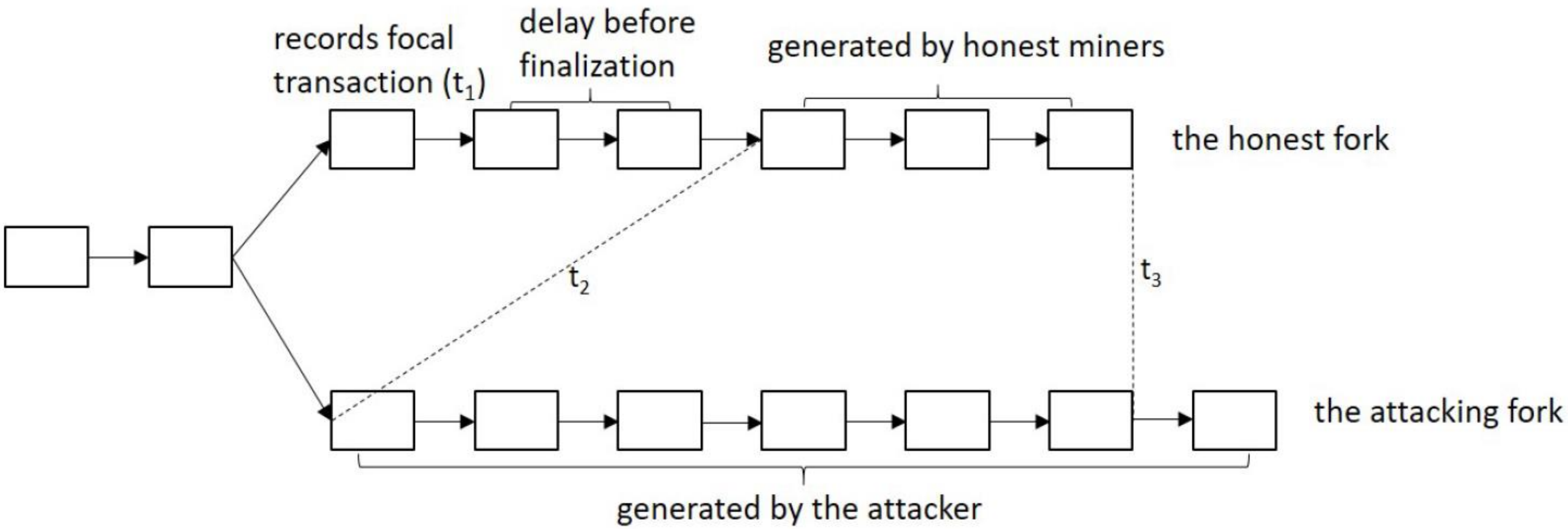
Summary of Bitcoin Network's consensus algorithm

- **Writers show disagreement through forking.**
- **Writers show support for each fork by working on extending it.**
- **The speed of extending a fork is (statistically) proportional to the computational power works on it.**
- **The longest chain is considered the valid one and the creators of blocks on it can earn coin rewards.**
- **Results of the consensus algorithm:**
  - If a writer cares about the voting results, it is a majority voting where the number of votes is proportional to computational power.
  - If a writer cares only about the coin rewards instead of voting results, this writer always works on the longest (winning) fork.

Double spending attack

- **An attacker trying to generate an alternate chain longer than the honest chain.**
- **Even if this is accomplished, it does not throw the system open to arbitrary changes.**
- **An attacker can only try to change one of his own transactions to take back money he recently spent.**
  - And then spend it again.

Double spending attack


records focal transaction (t_1)
delay before finalization
generated by honest miners
the honest fork
t_2
t_3
the attacking fork
generated by the attacker

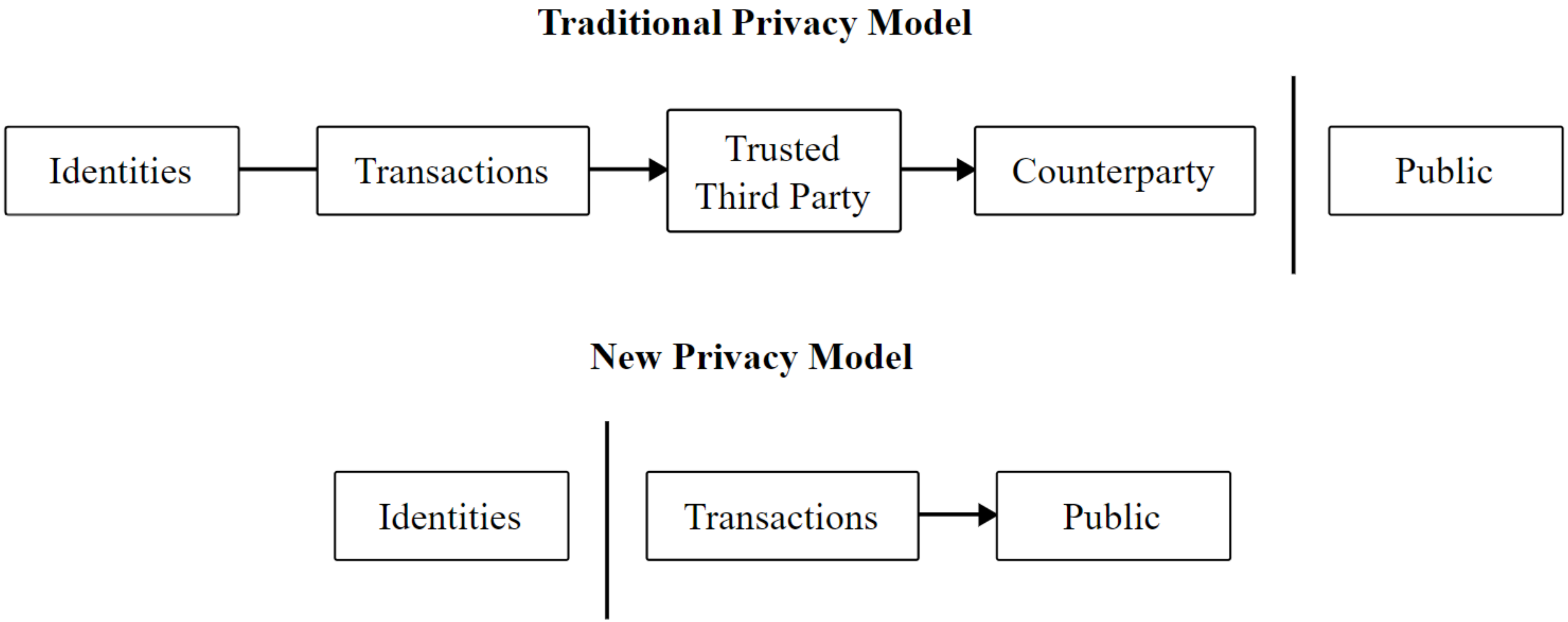
Security

- **Delay before finalization**
  - Usually recommend 6 blocks (10 min/block)
- **The more blocks the receiver of the transaction wait before the finalization of the deal, the more security the receiver has.**
  - If the attacker controls less than half computational power, exponentially less likely for the attack to be successful.
- **This is why the “chain” matters.**

Privacy

- **The traditional banking model achieves a level of privacy by limiting access to information to the parties involved and the trusted third party.**
- **The necessity to announce all transactions publicly precludes this method.**
- **Privacy can still be maintained by breaking the flow of information in another place: by keeping public keys anonymous.**
  - The public can see that someone is sending an amount to someone else.
  - But not knowing who they personally are.
  - Similar to the level of information released by stock exchange

Privacy


Traditional Privacy Model
Identities
Transactions
Trusted Third Party
Counterparty
Public
New Privacy Model
Identities
Transactions
Public

Privacy

- **A new key pair should be used for each transaction to keep them from being linked to a common owner.**
- **Some linking is still unavoidable with multi-input transactions, which necessarily reveal that their inputs were owned by the same owner.**
- **If the owner of a key is revealed, linking could reveal other transactions that belonged to the same owner.**

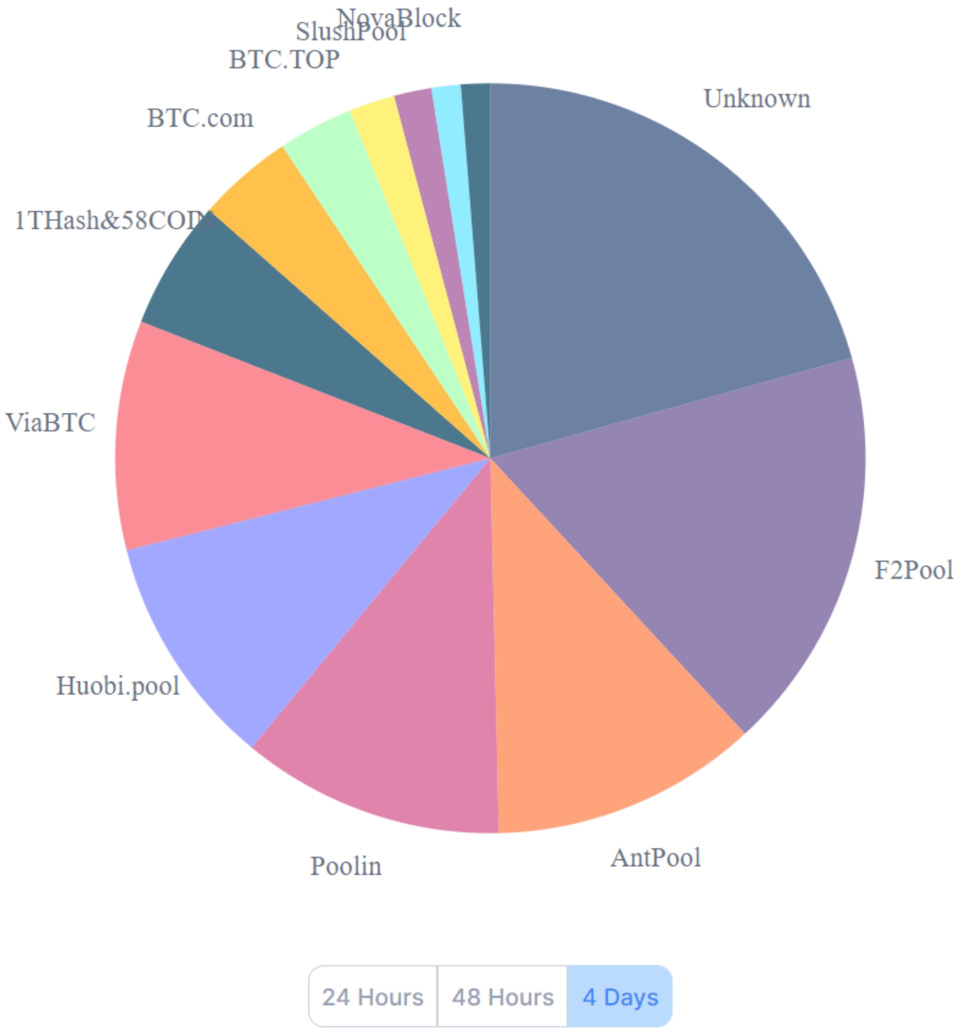
My summary of blockchain in its original form

- **Key ideas**
  - Distributed power of writing the ledger to avoid trust on centralized authority.
  - “chain” structure to ensure the security of the distributed ledger.

- **Key assumptions**
  - The majority of the nodes will work honestly on writing the ledger.
  - The nodes will publicize the nonce once they found it for a block.

Computational power distribution of Bitcoin Network

- **Do the assumptions hold?**
- **How many nodes are there?**
- **Are they independent?**



Mining pools/Significant miners

- **Miners tend to form pools or become significant miners because of:**
- **Risk aversion**
  - Miners form pools to share risk of mining.
- **Scale of economics**
  - Large mining factories are much more cost effective than a single CPU/GPU.

- **Do these mining pools/significant miners act independently?**
  - It is very hard to say!

Decentralized trust

- **Original idea: distributed power of writing the ledger to avoid trust on centralized authority.**
  - Many independent honest miners
  - Each miner does not have much power, and because they are independent, the majority of them can be trusted.
- **Result: a few (or even less) entities control most of the power**
  - Trust on these (semi-)centralized entities
  - They are anonymous, not bounded by law, auditing,…
- **Decentralization does not equal to independence!**

Decentralized trust

- **Can these entities with power be trusted?**
- **They have their own incentives**
  - Investments in Bitcoin and its mining hardware
  - Businesses dependent on Bitcoin (cyber attacks, Darknet retailing,…)
- **Are these incentives aligned with the security and integrity of Bitcoin Network?**
  - For now, yes.

Selfish mining

- **Assumption: the nodes will publicize the nonce once they found it for a block.**

- **The nodes may be better off not publicizing the nonce immediately.**

- **In some cases (computational power around 30%~40%), not publicizing the nonce immediately can earn disproportional coin rewards.**

- **Not observed in Bitcoin Network**

Thank you!