Hyperledger

FOUNDATION

Decentralization and Privacy On Blockchain

The Basics

Hart Montgomery
Hyperledger Foundation

Hyperledger FOUNDATION

Hart Montgomery

- Hyperledger Foundation CTO (and some other stuff at the Linux Foundation)
- Previously worked in blockchain and cryptography research at Fujitsu Research, where I helped lead Fujitsu's efforts in Hyperledger and also served on the Hyperledger TSC since 2016
- Ph.D. in cryptography at Stanford under Dan Boneh, where I was a Stanford Graduate Fellow.

Hyperledger FOUNDATION

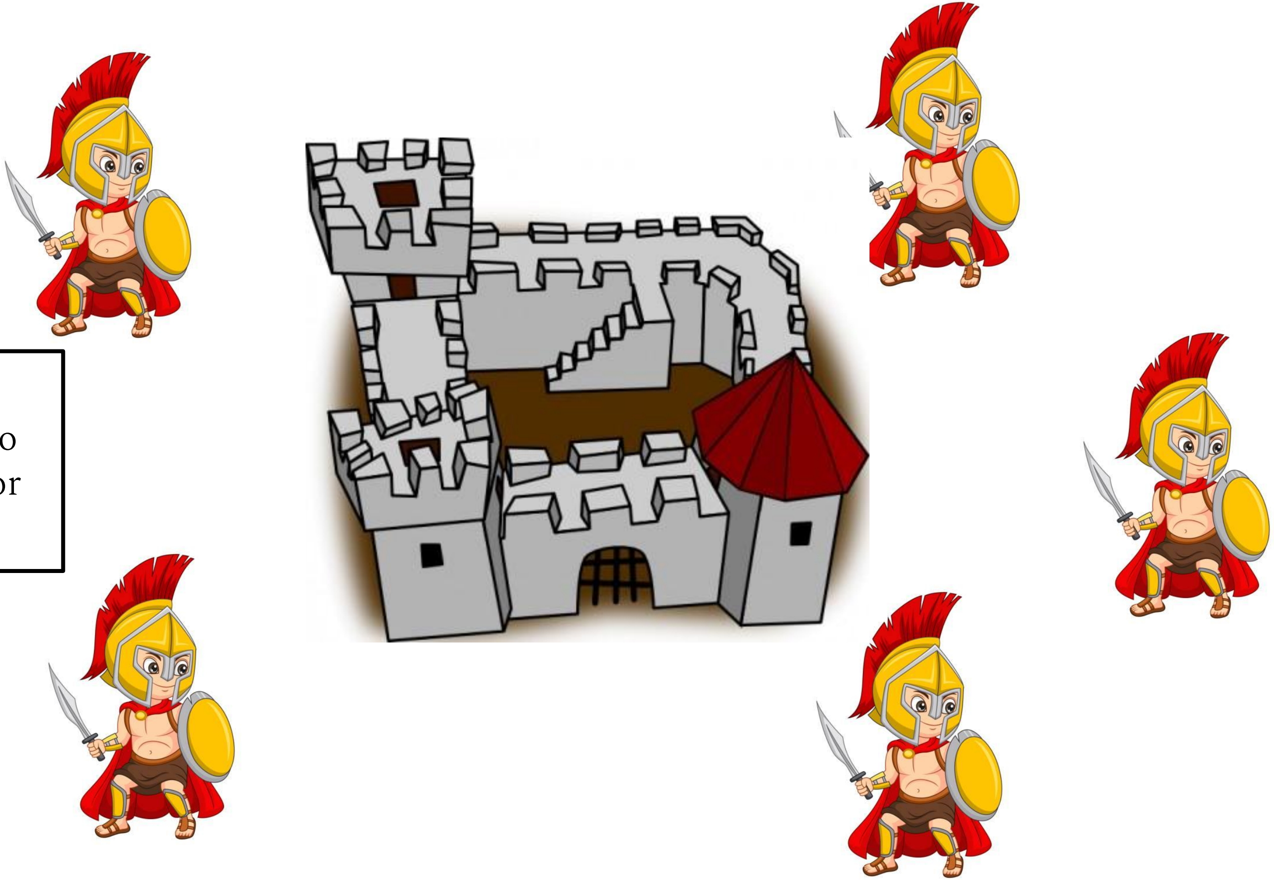
Talk Outline

- Background: the Byzantine generals’ problem
- Decentralization: the core of Web3 and blockchain
- Drawbacks of blockchain and decentralization
    - Privacy, and what we can do about it

Background:

The Byzantine Generals’ Problem

Byzantine Generals' Problem

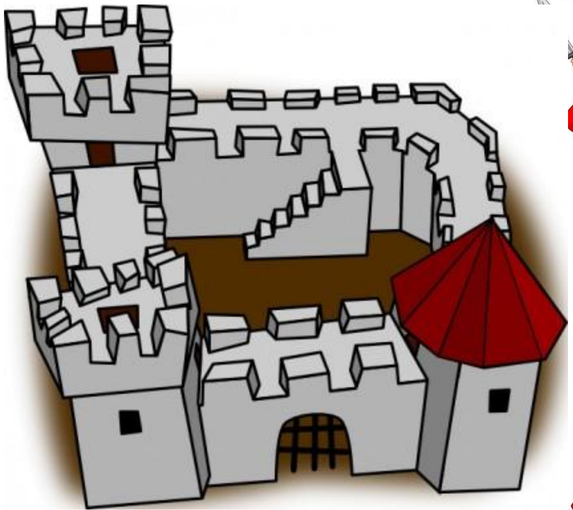


N = 5 Generals
Deciding whether to attack the fortress or retreat.

Byzantine Generals' Problem

If all generals attack:

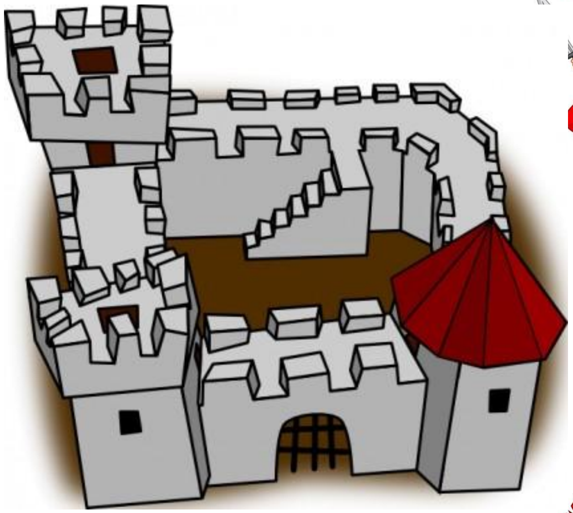
Victory is likely.



N = 5 Generals

Deciding whether to attack the fortress or retreat.

Byzantine Generals’ Problem



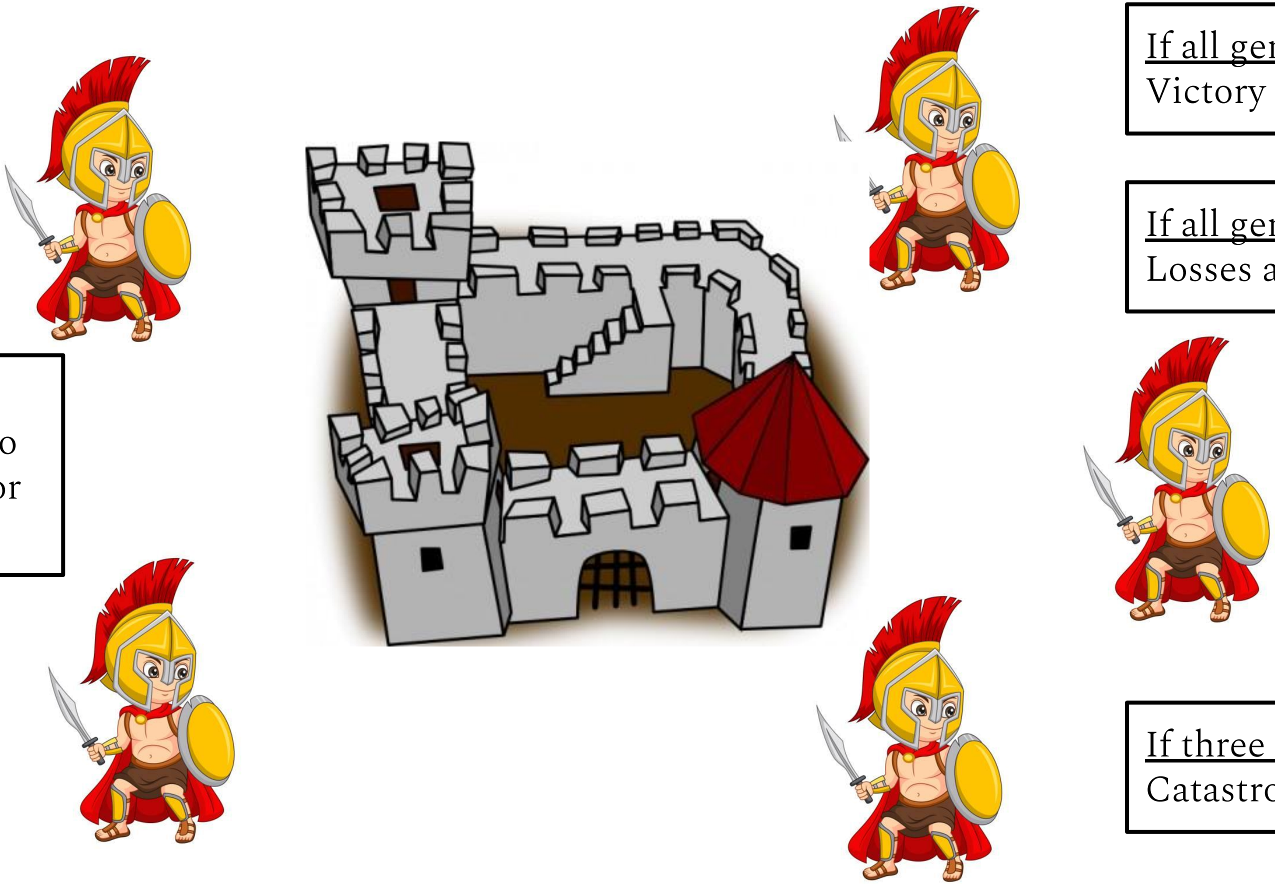
If all generals attack:
Victory is likely.

If all generals retreat:
Losses are minimal.

N = 5 Generals
Deciding whether to attack the fortress or retreat.

Byzantine Generals’ Problem

N = 5 Generals
Deciding whether to attack the fortress or retreat.

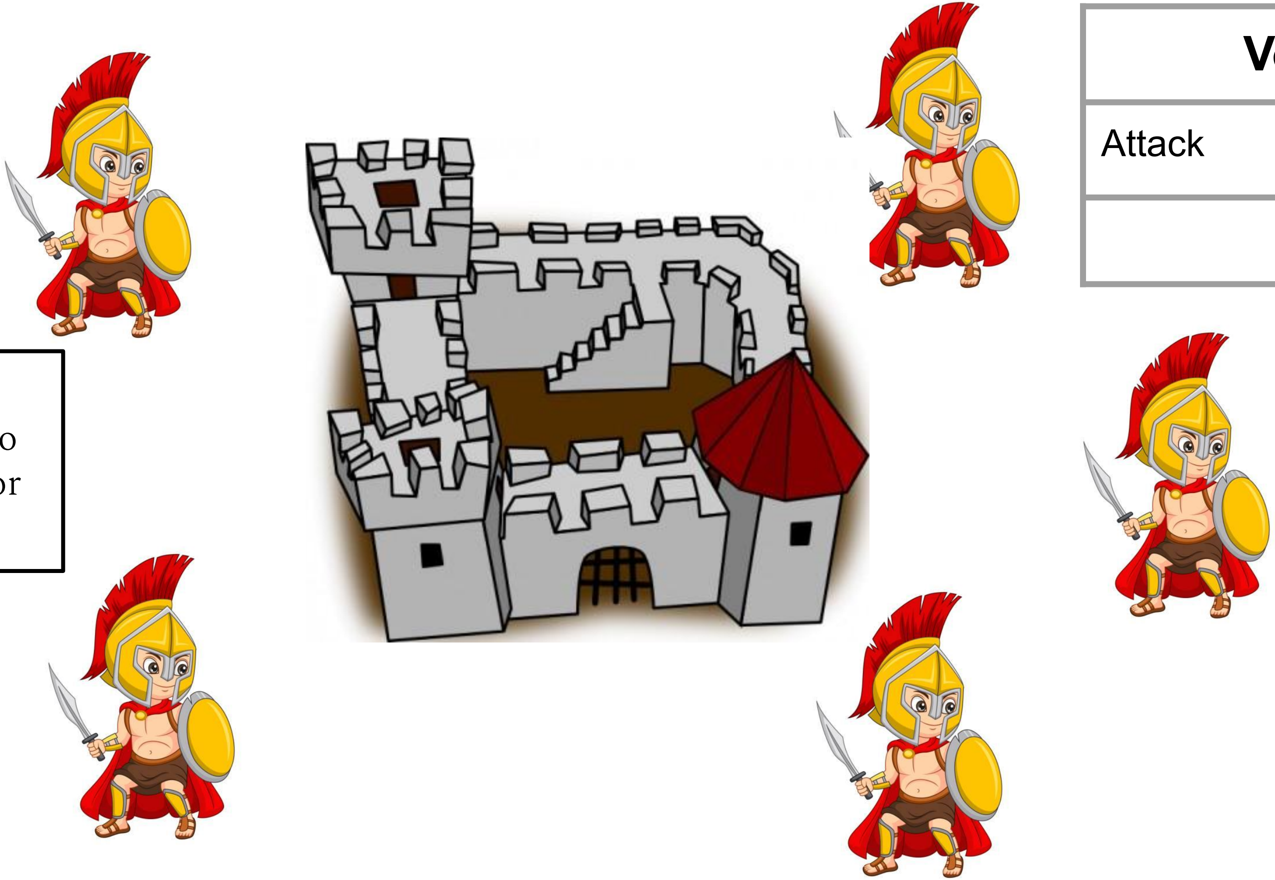


If all generals attack:
Victory is likely.

If all generals retreat:
Losses are minimal.

If three or less attack:
Catastrophe!

Byzantine Generals’ Problem

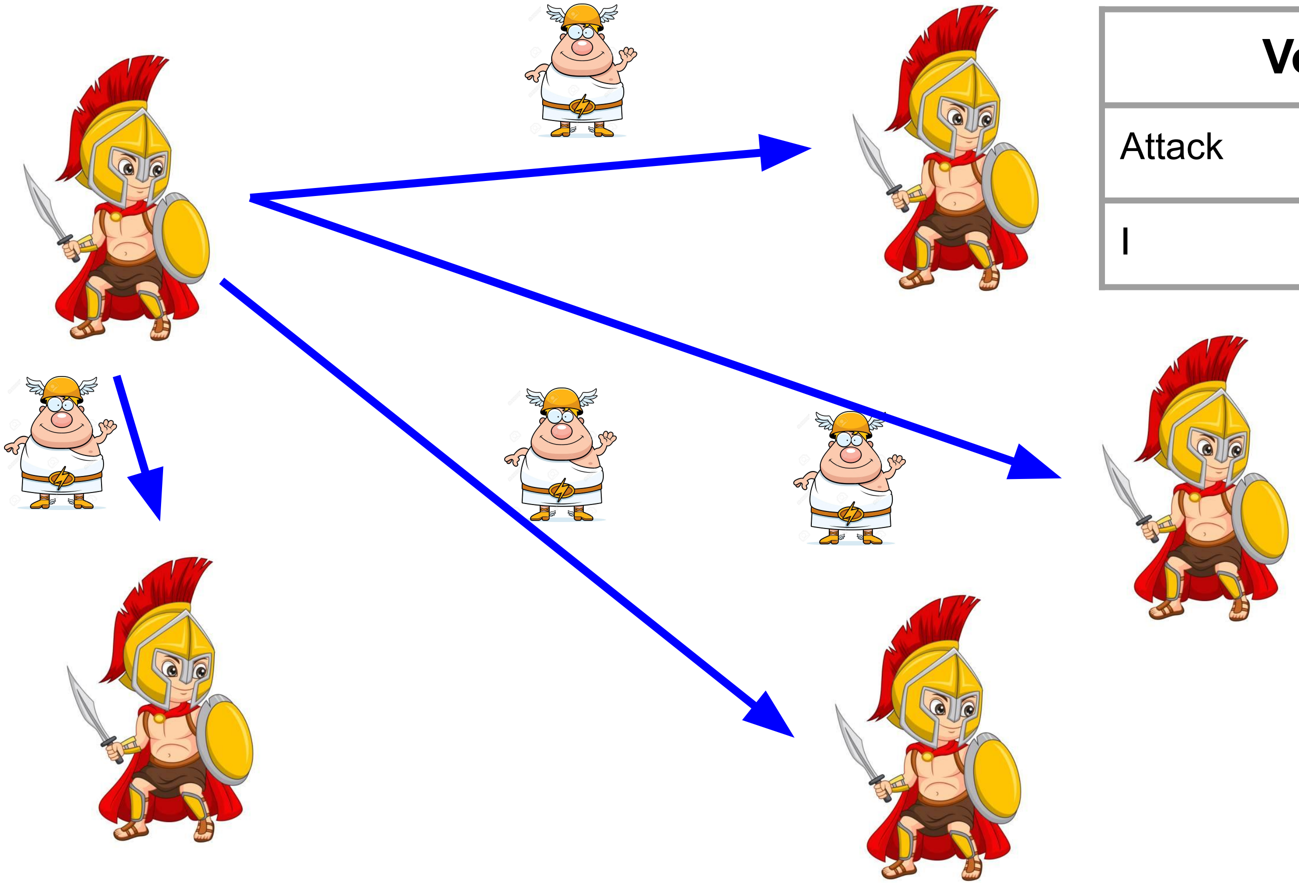


Vote	
Attack	Retreat

<u>N = 5 Generals</u>
Deciding whether to attack the fortress or retreat.

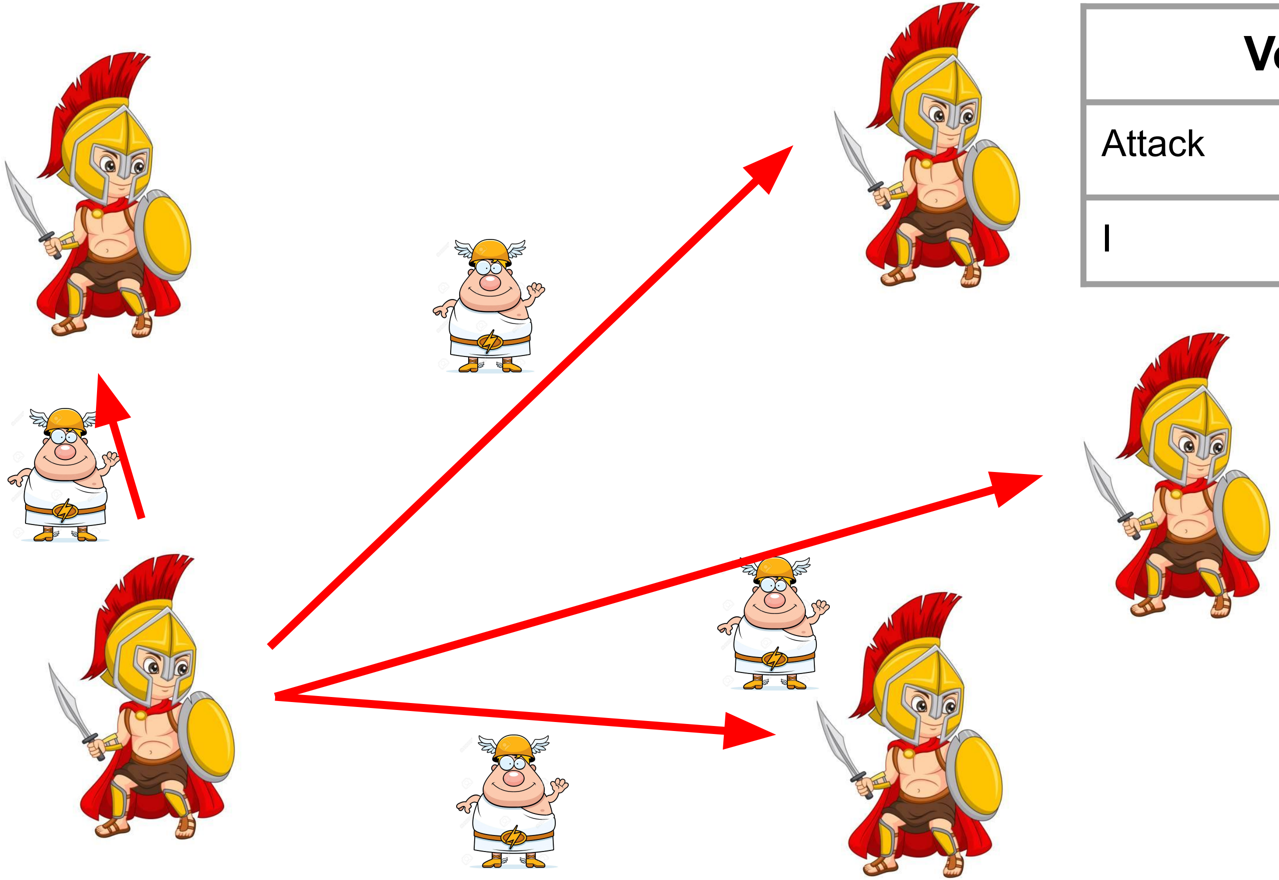
Byzantine Generals’ Problem

Attack!



Vote	
Attack	Retreat
I	

Byzantine Generals’ Problem

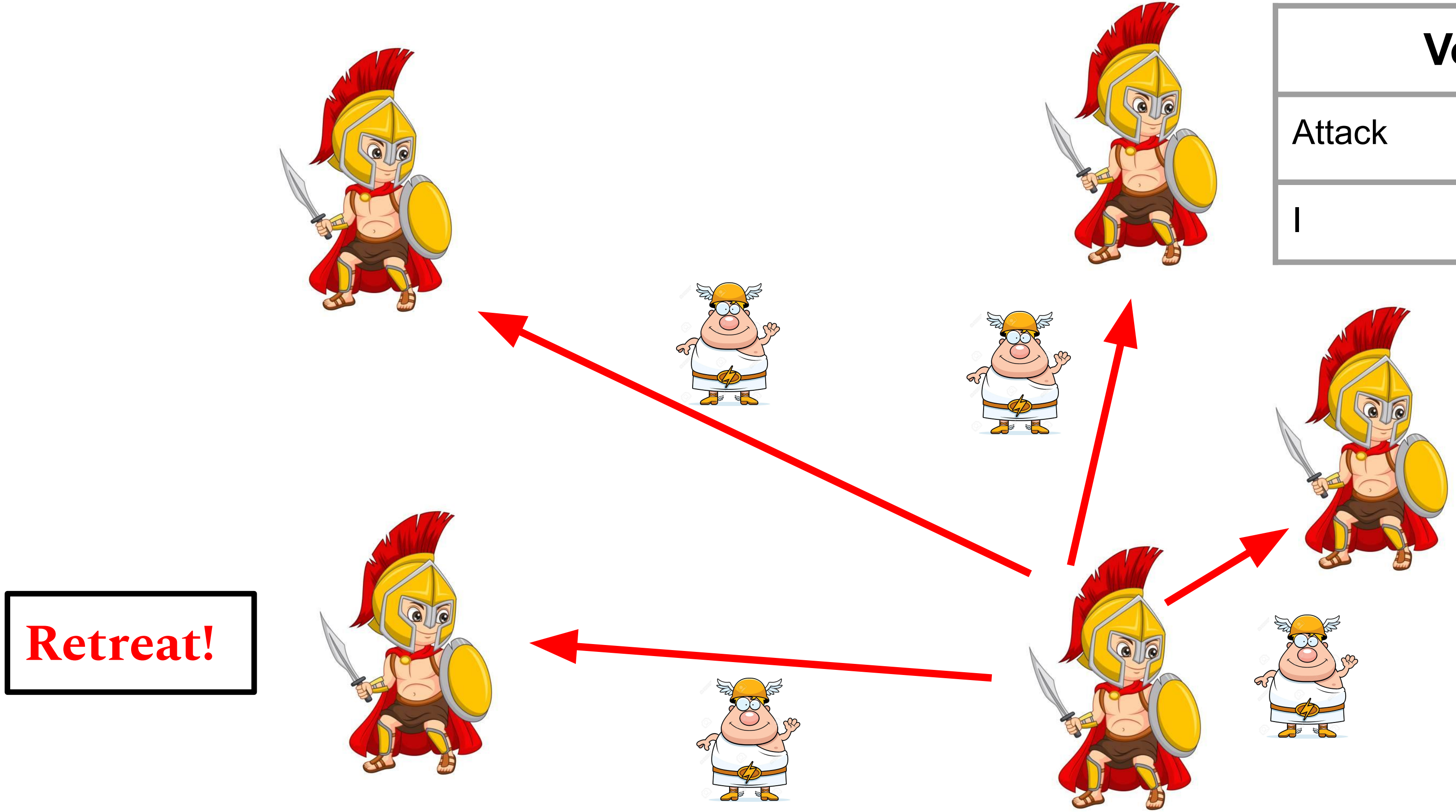


Retreat!

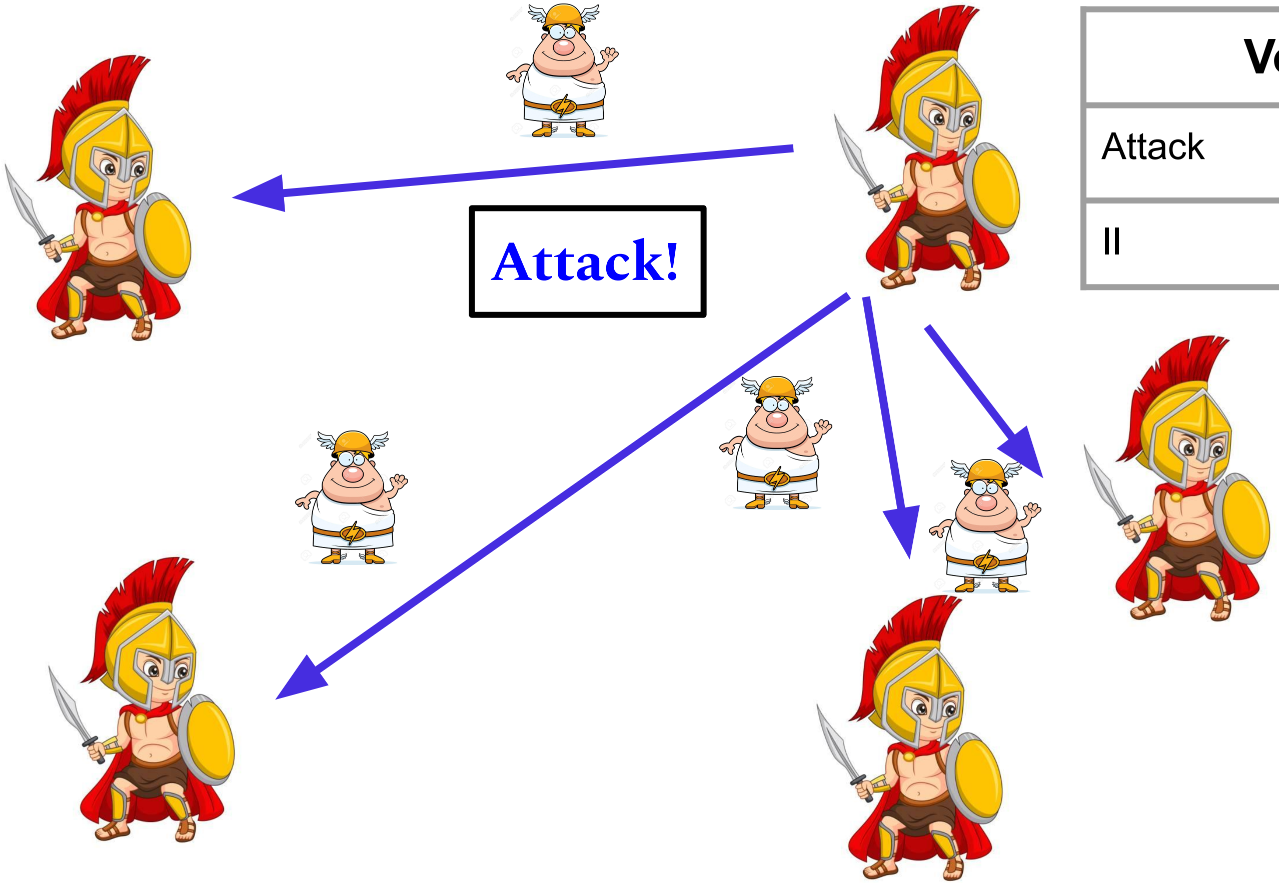
Vote	
Attack	Retreat
I	I

Byzantine Generals’ Problem

Vote	
Attack	Retreat
I	II

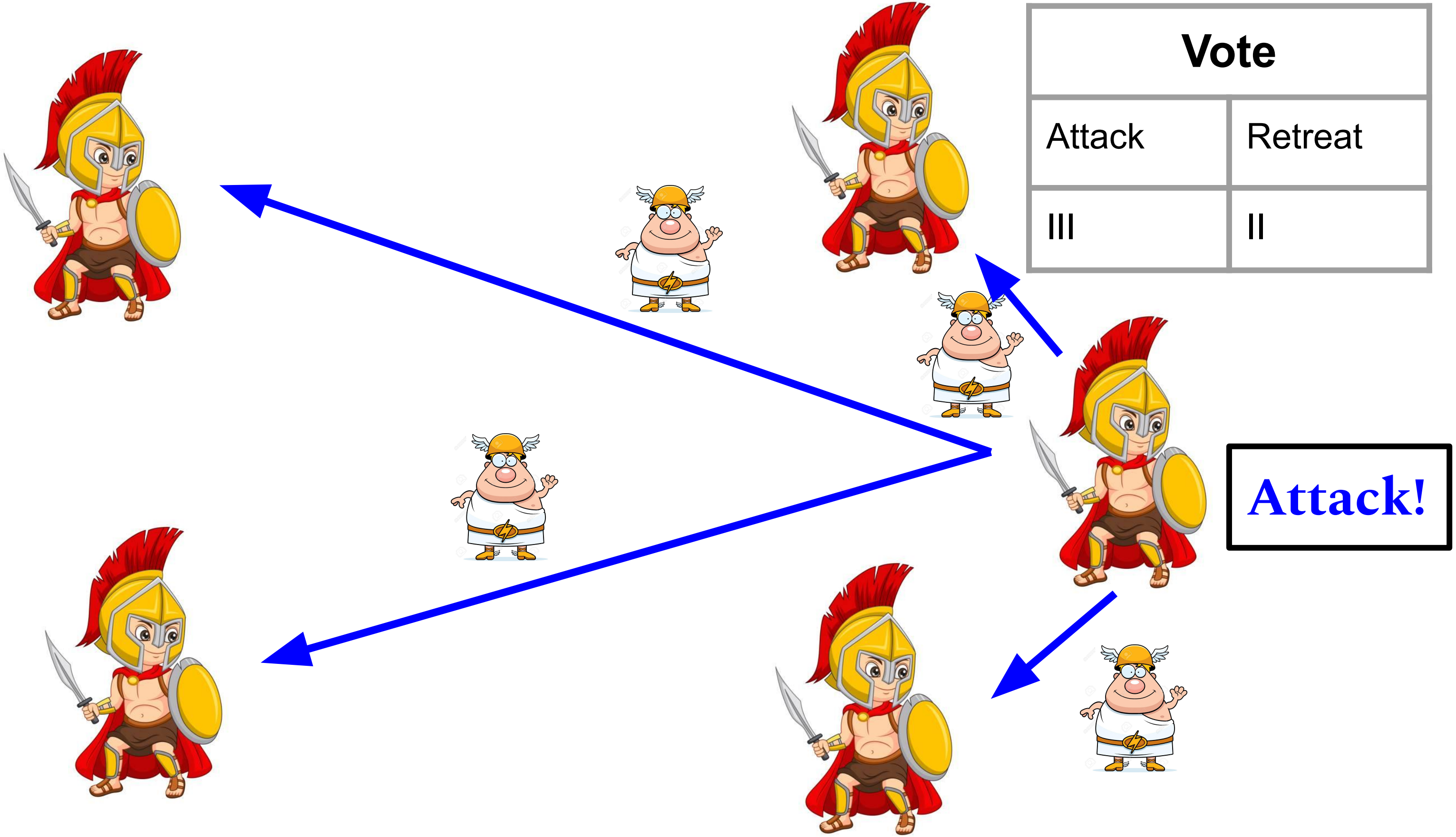


Byzantine Generals' Problem



Vote	
Attack	Retreat
II	II

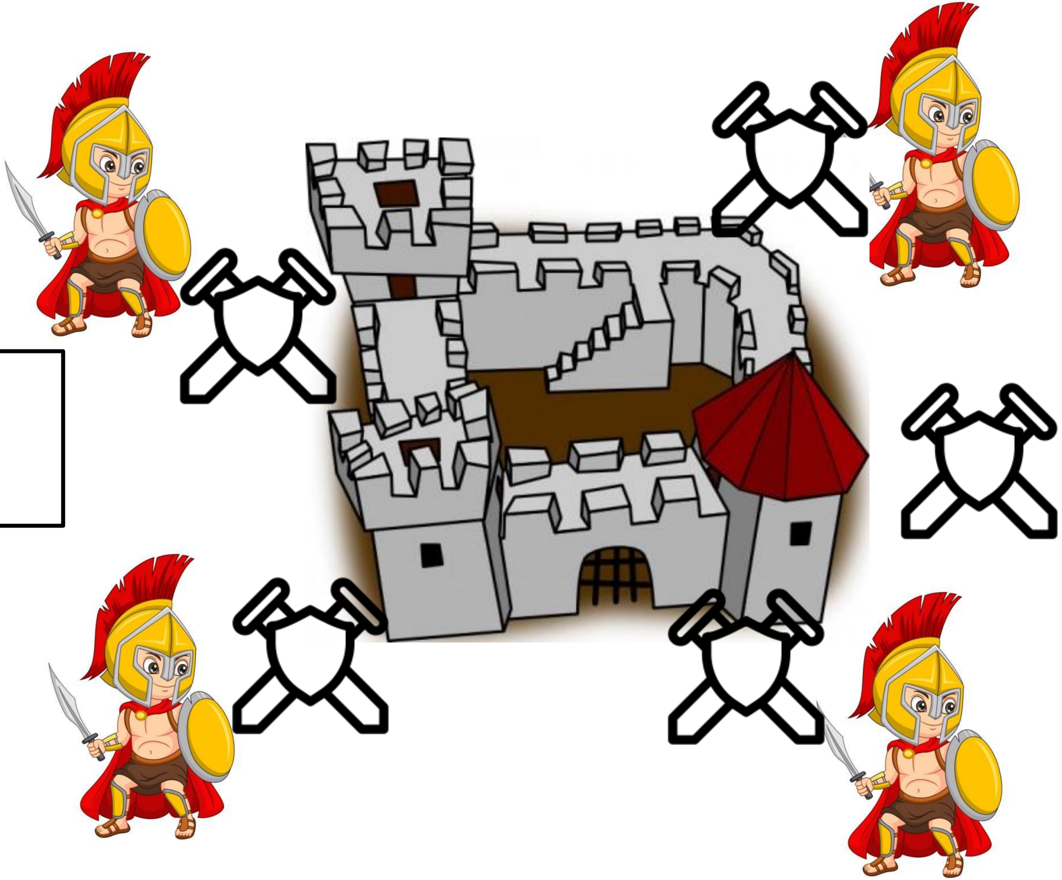
Byzantine Generals' Problem



Vote	
Attack	Retreat
III	II

Attack!

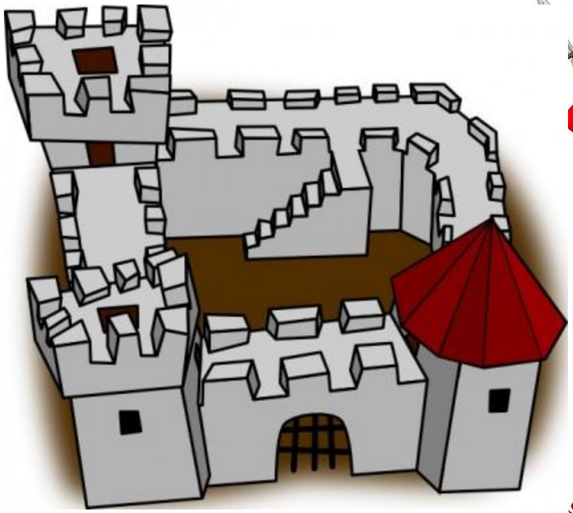
Byzantine Generals’ Problem



Vote	
Attack	Retreat
III	II

All attack:
Likely a good outcome

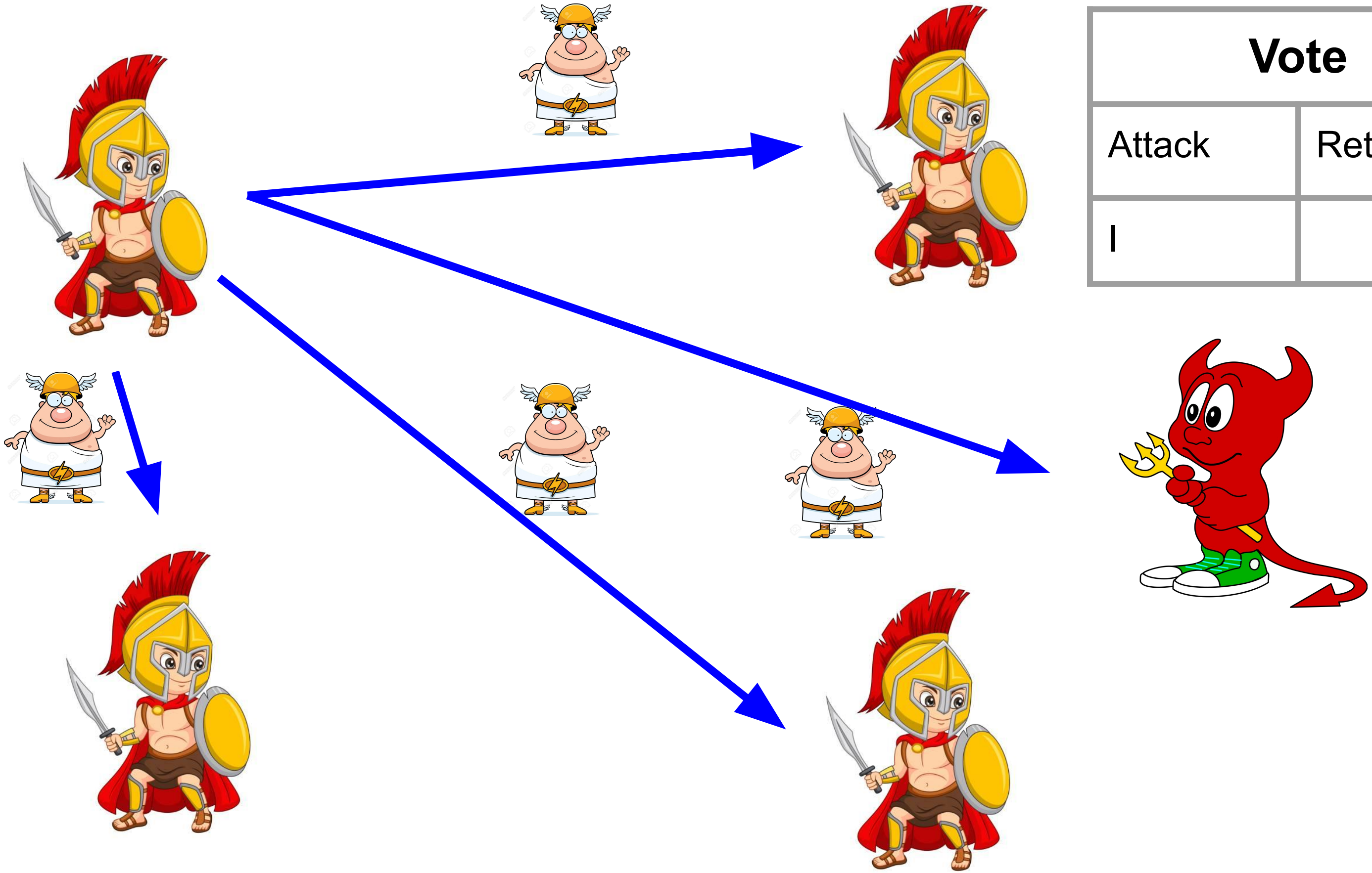
Byzantine Generals' Problem



What if a General is a Traitor?

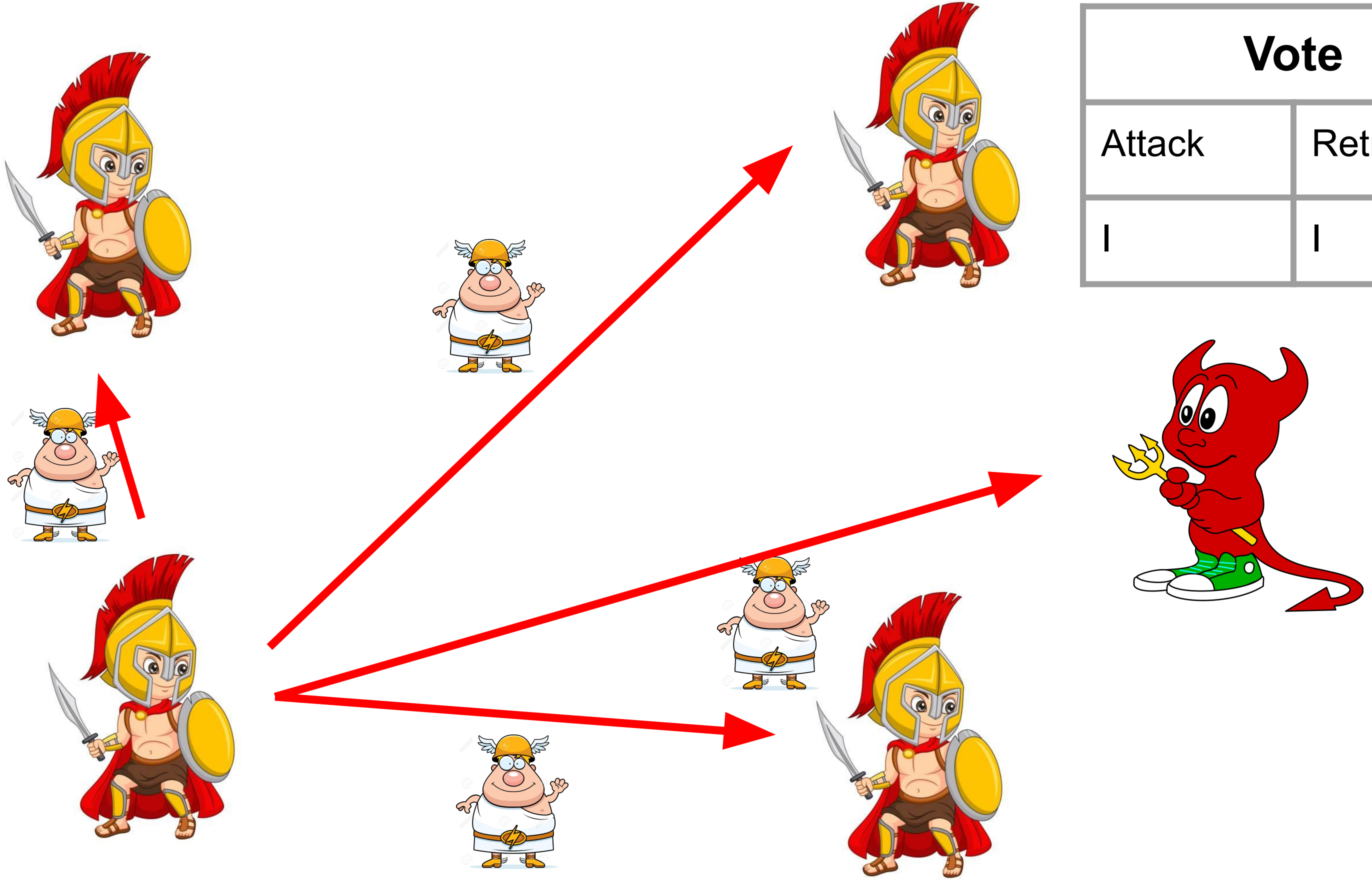
Byzantine Generals’ Problem

Attack!



Vote	
Attack	Retreat
I	

Byzantine Generals’ Problem

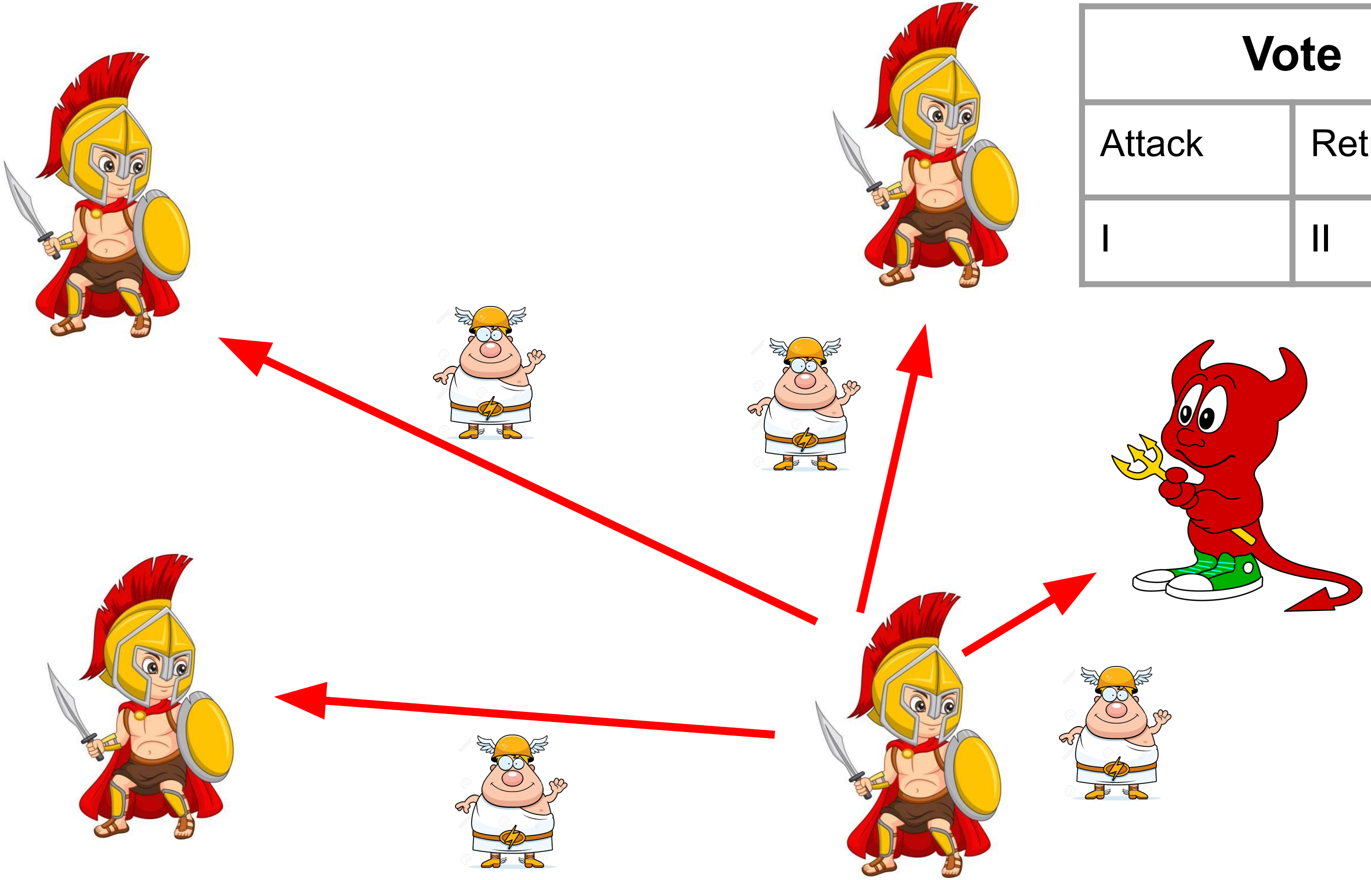


Retreat!

Vote	
Attack	Retreat
I	I

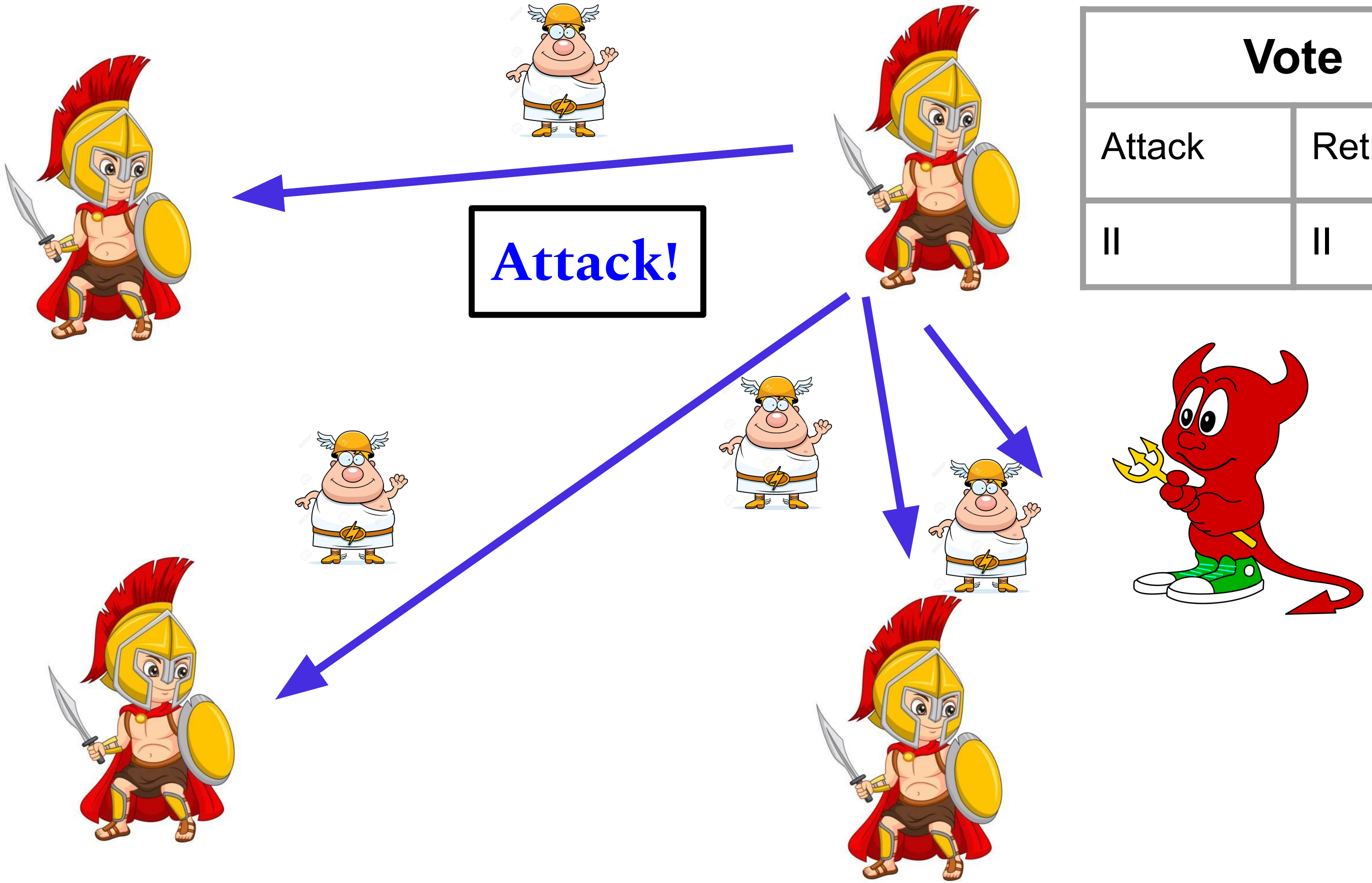
Byzantine Generals’ Problem

Vote	
Attack	Retreat
I	II



Retreat!

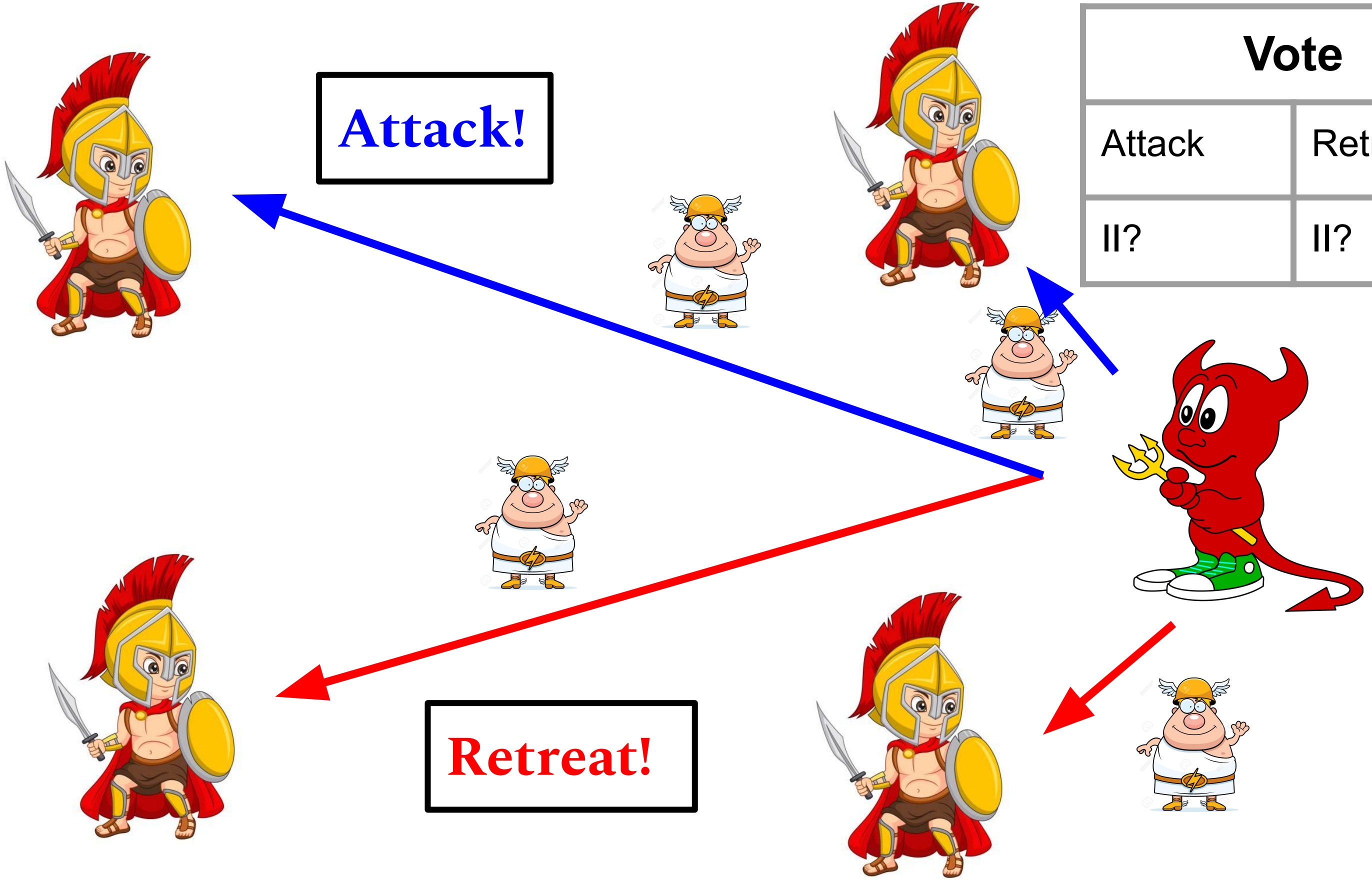
Byzantine Generals' Problem



Attack!

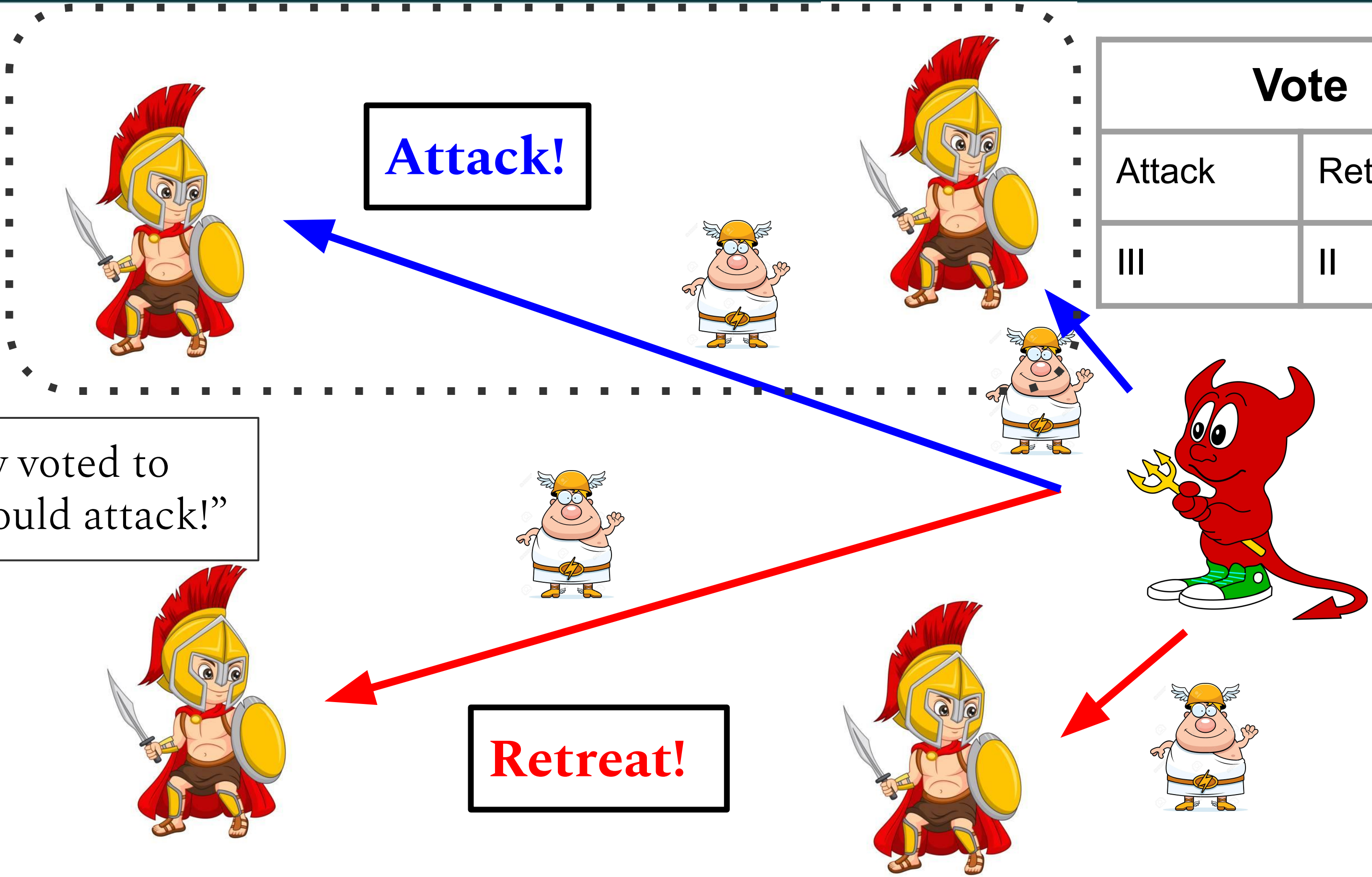
Vote	
Attack	Retreat
II	II

Byzantine Generals' Problem



Vote	
Attack	Retreat
II?	II?

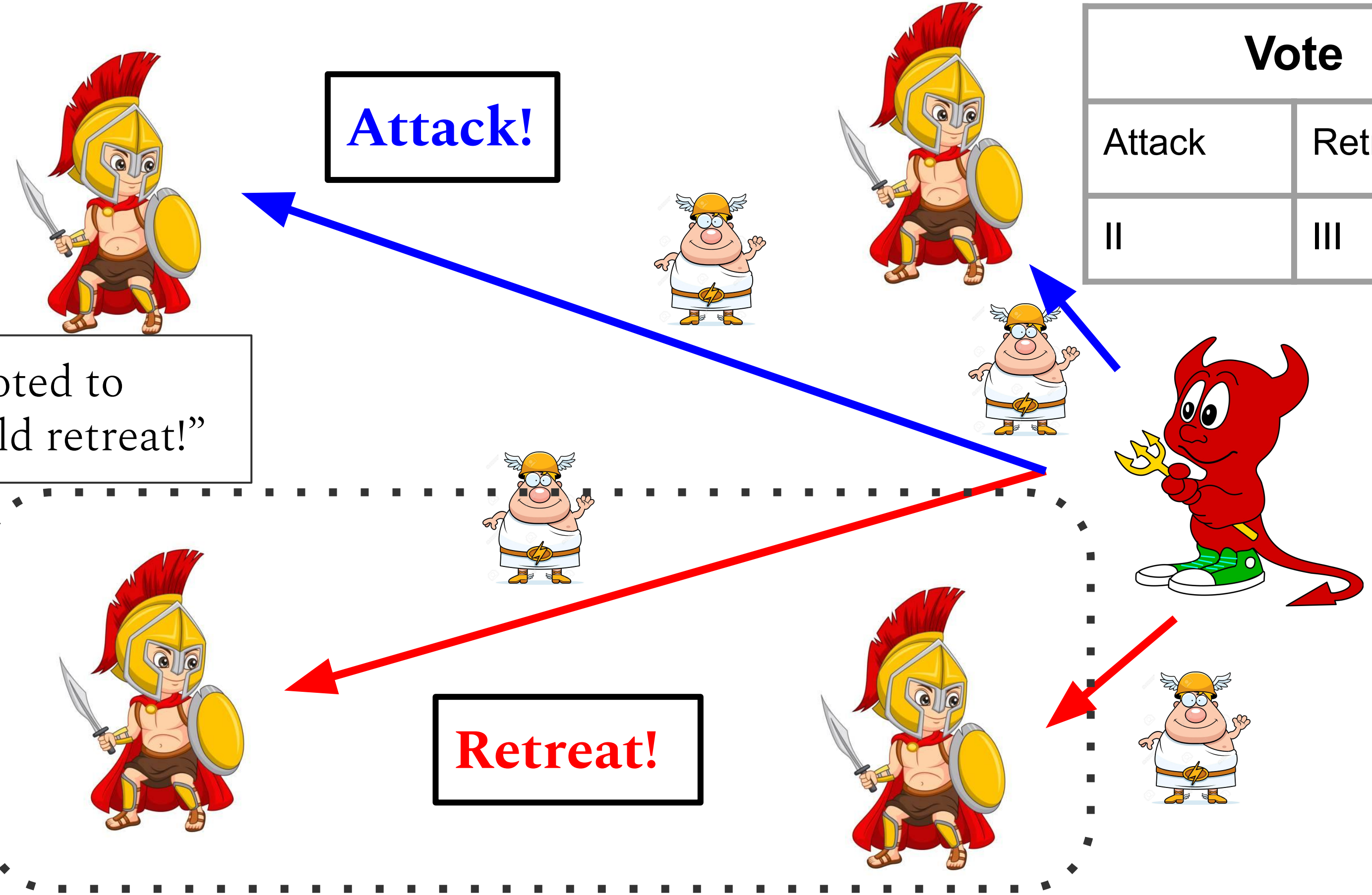
Byzantine Generals' Problem



Vote	
Attack	Retreat
III	II

"The majority voted to attack; we should attack!"

Byzantine Generals’ Problem

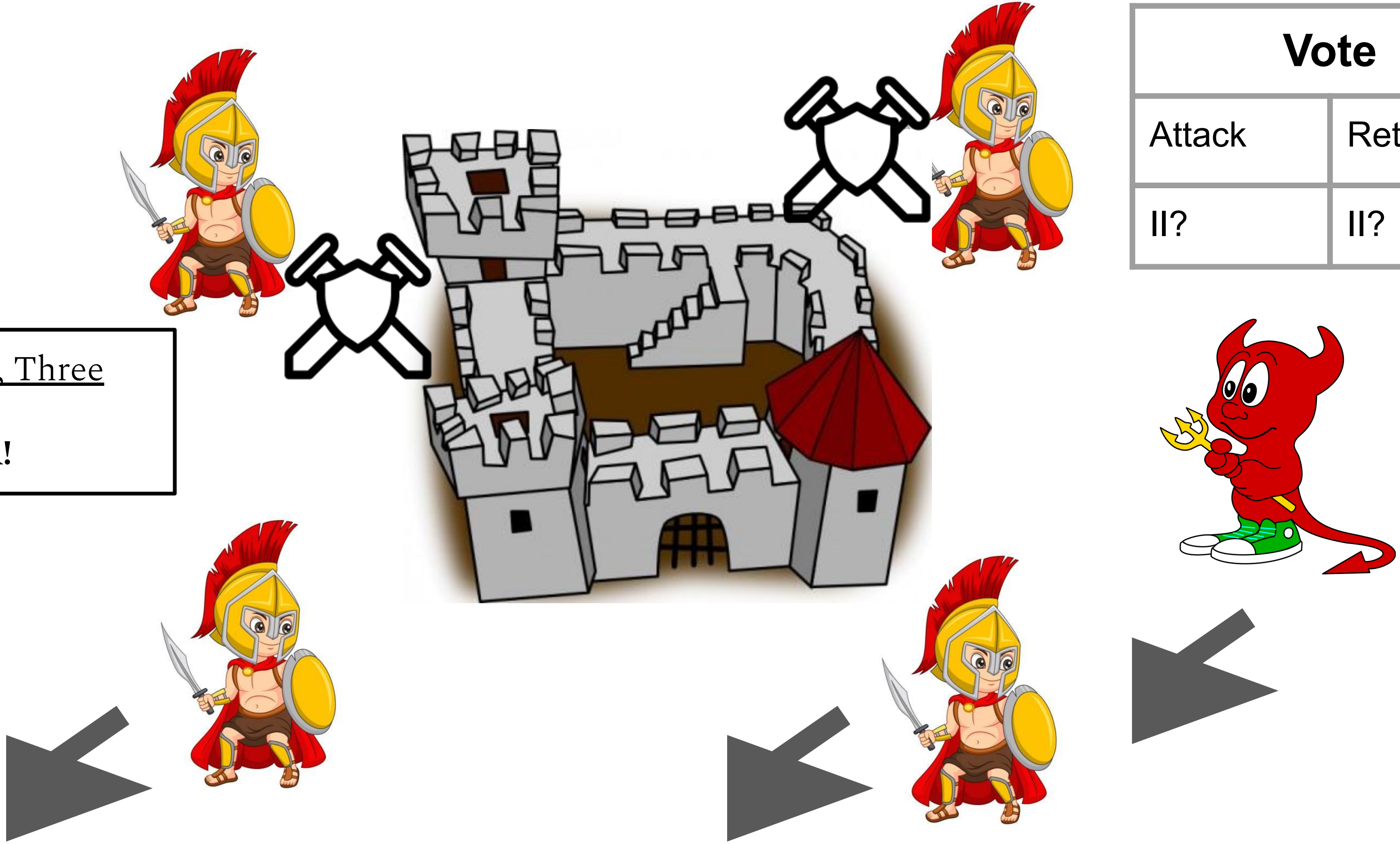


Vote	
Attack	Retreat
II	III

“The majority voted to retreat; we should retreat!”

Byzantine Generals' Problem

Vote	
Attack	Retreat
II?	II?



Two Attack, Three Retreat:
DISASTER!

Byzantine Generals' Problem

What Can Be Done?

Solution: **Distributed Consensus Protocols**

Byzantine Generals' Problem

What Can Be Done?
Solution: **Distributed Consensus Protocols**

Guarantee (with a good protocol): if more than ⅔ of the generals are "honest", all of the "honest" generals will take the same action (attack or retreat).

This is called **Byzantine fault-tolerant consensus.**

Byzantine Generals’ Problem

What Can Be Done?

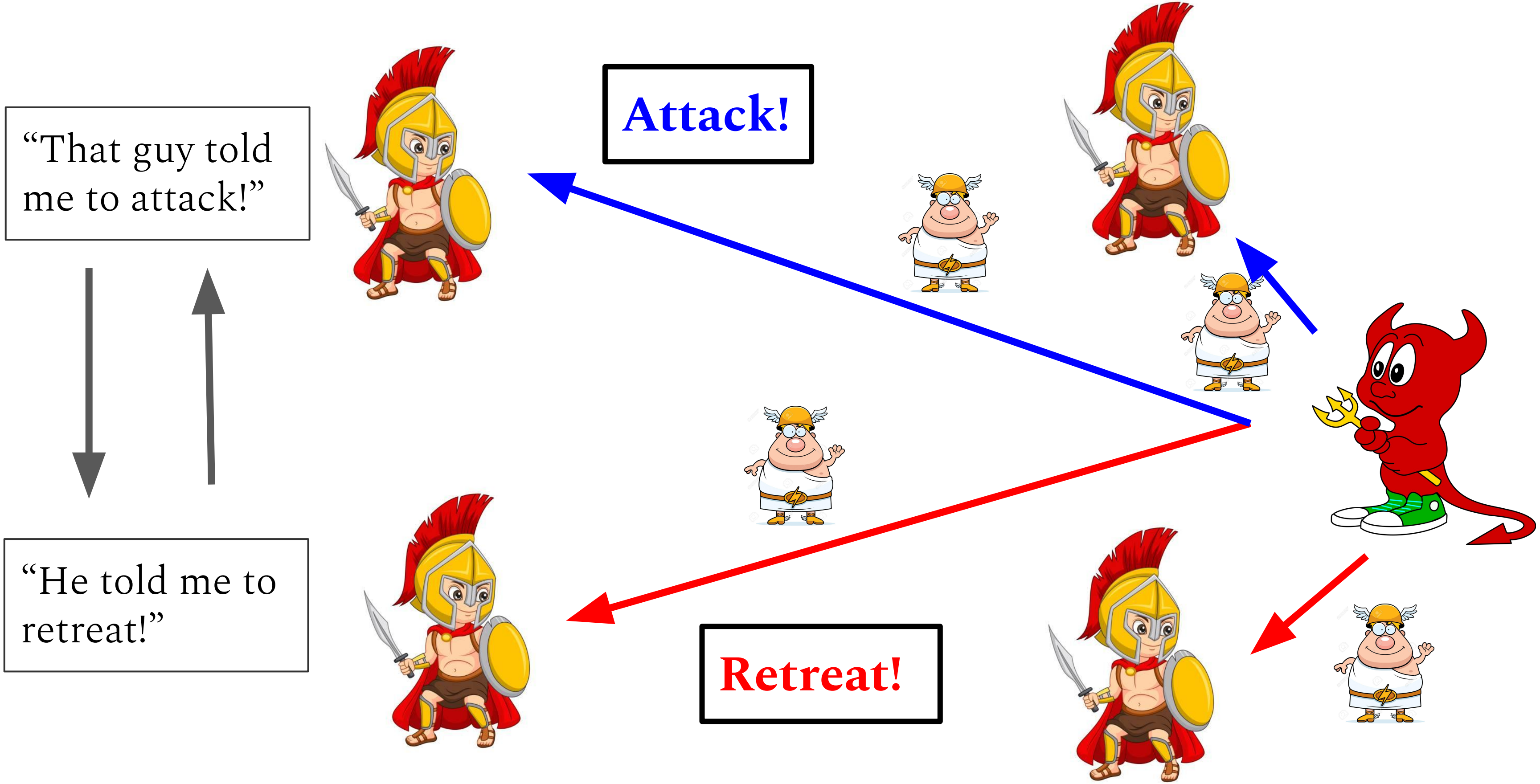
Solution: **Distributed Consensus Protocols**

Guarantee (with a good protocol): if more than ⅔ of the generals are “honest”, all of the “honest” generals will take the same action (attack or retreat).

This is called **Byzantine fault-tolerant consensus.**

Distributed consensus is the backbone of blockchain and Web3.

Byzantine Generals’ Problem



“That guy told me to attack!”

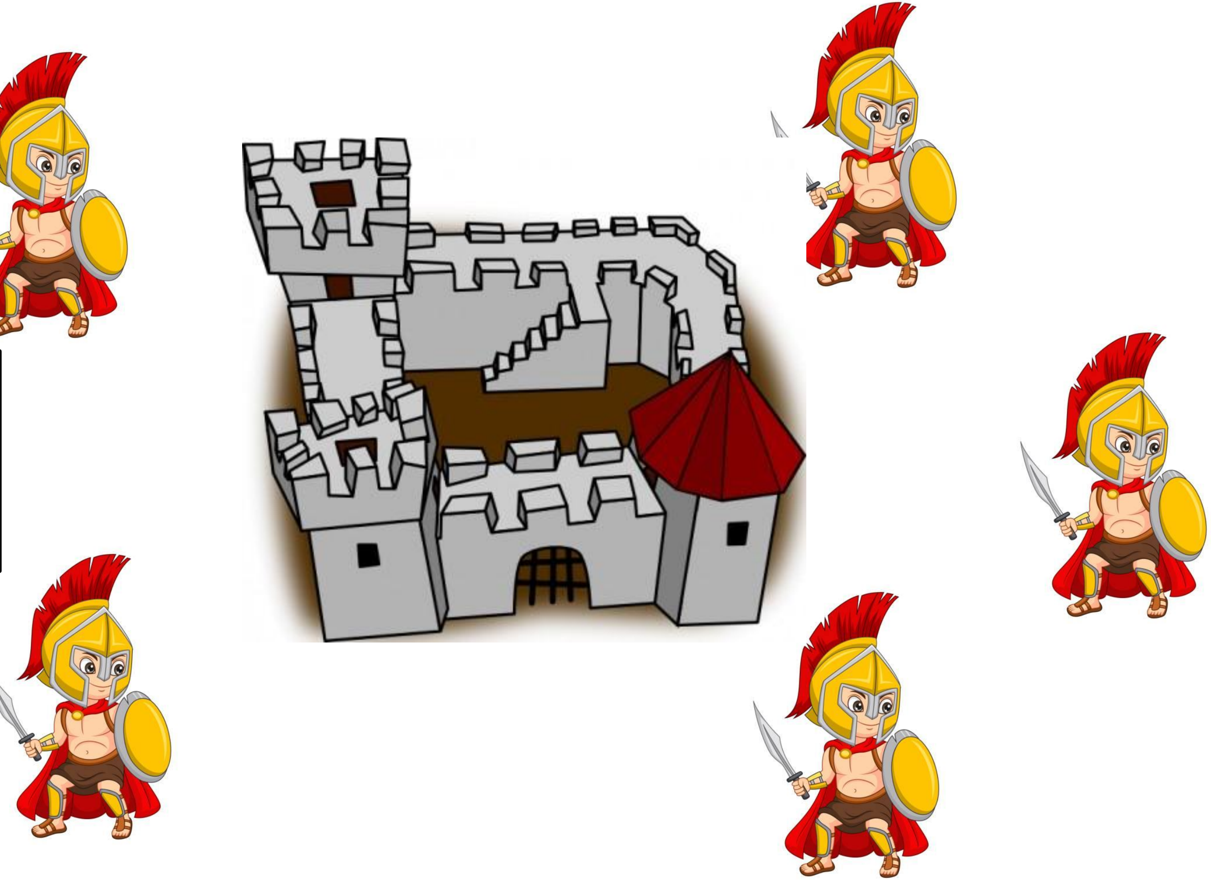
“He told me to retreat!”

Attack!

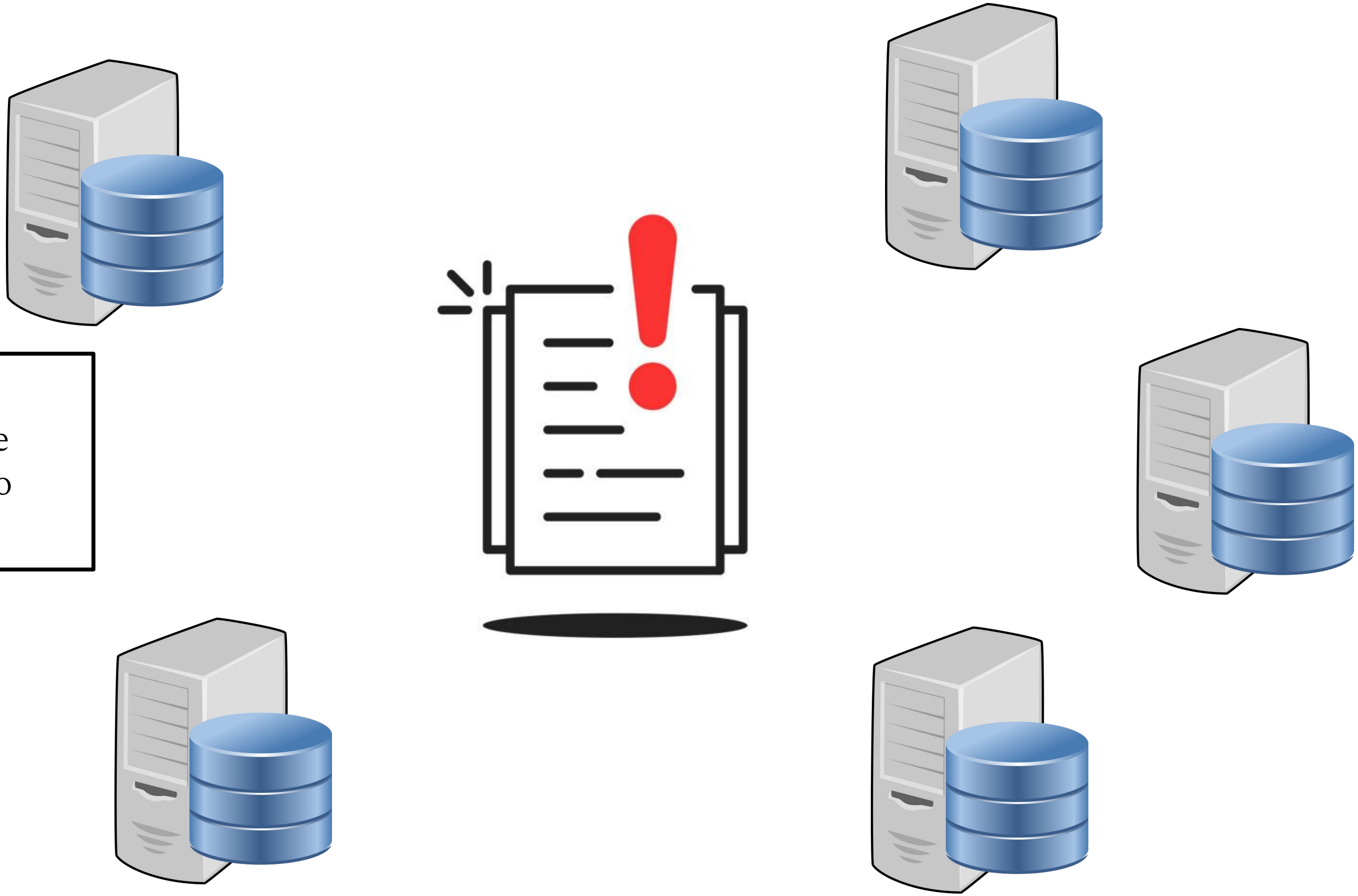
Retreat!

Byzantine Generals' Problem

N = 5 Generals: Deciding whether to attack the fortress or retreat.

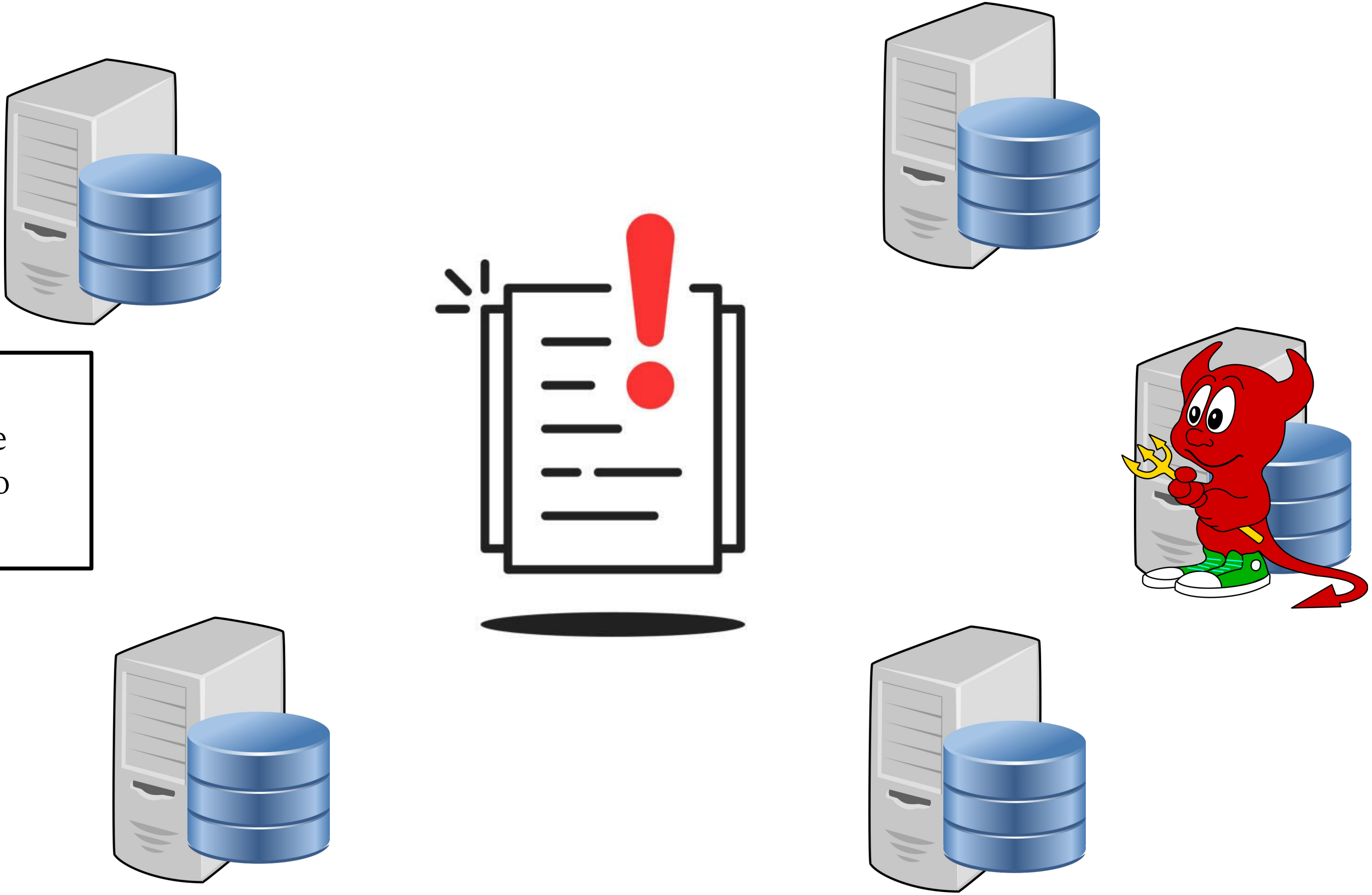


Byzantine Generals' Problem



N = 5 Servers:
Deciding on some state that needs to have agreement.

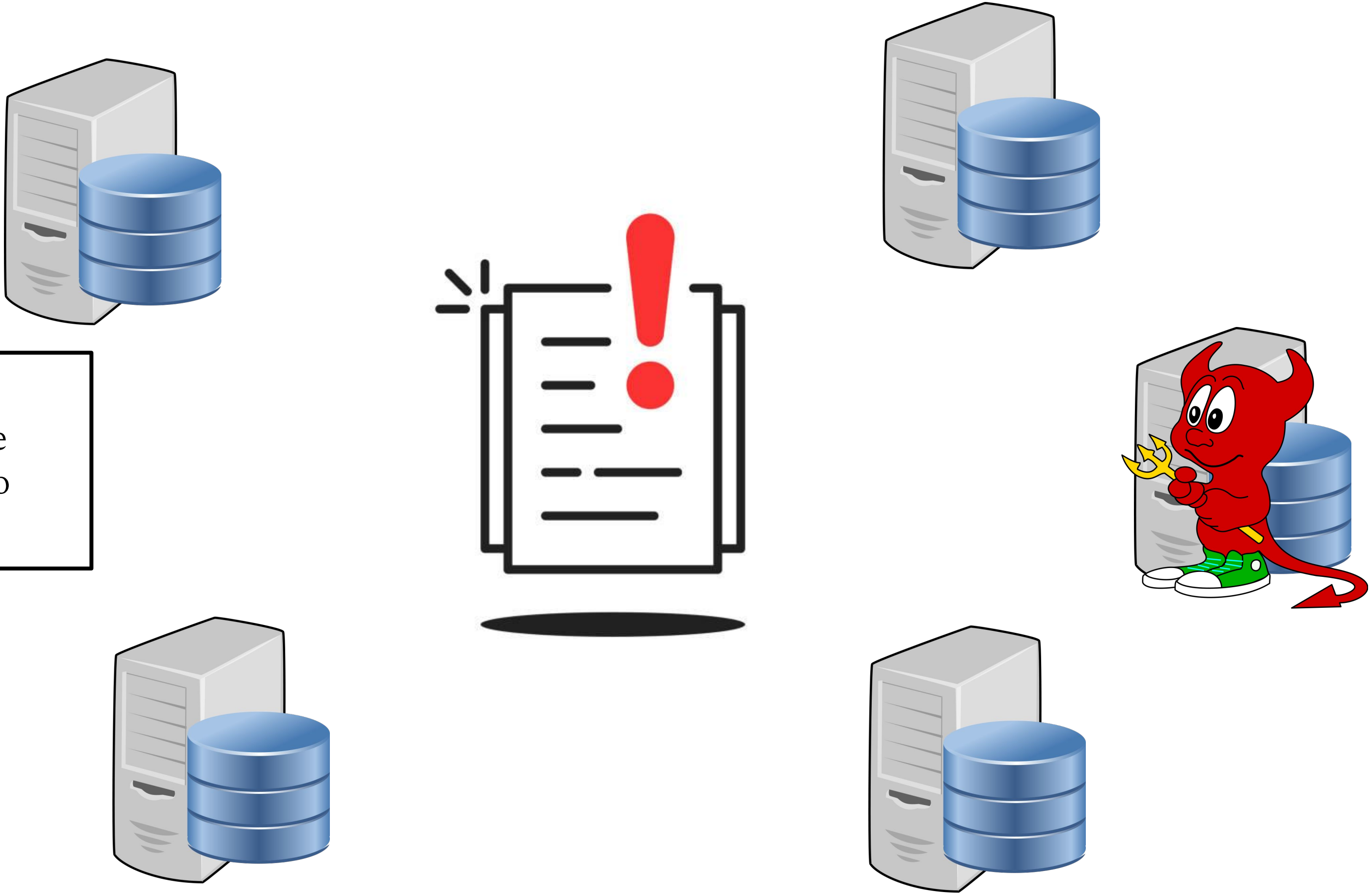
Byzantine Generals' Problem



N = 5 Servers:
Deciding on some state that needs to have agreement.

We Have Blockchain!

N = 5 Servers:
Deciding on some state that needs to have agreement.

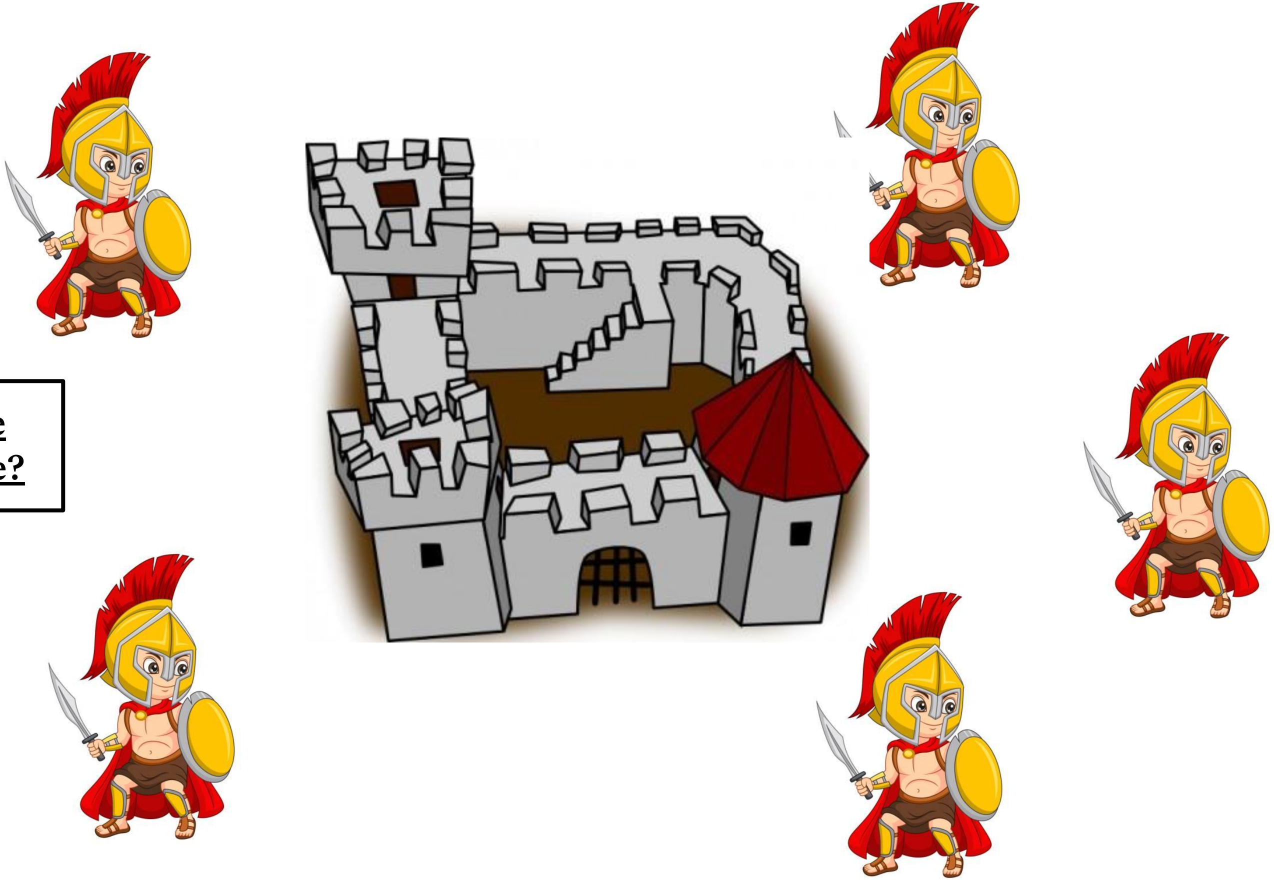


Decentralization:

The Core of Web3

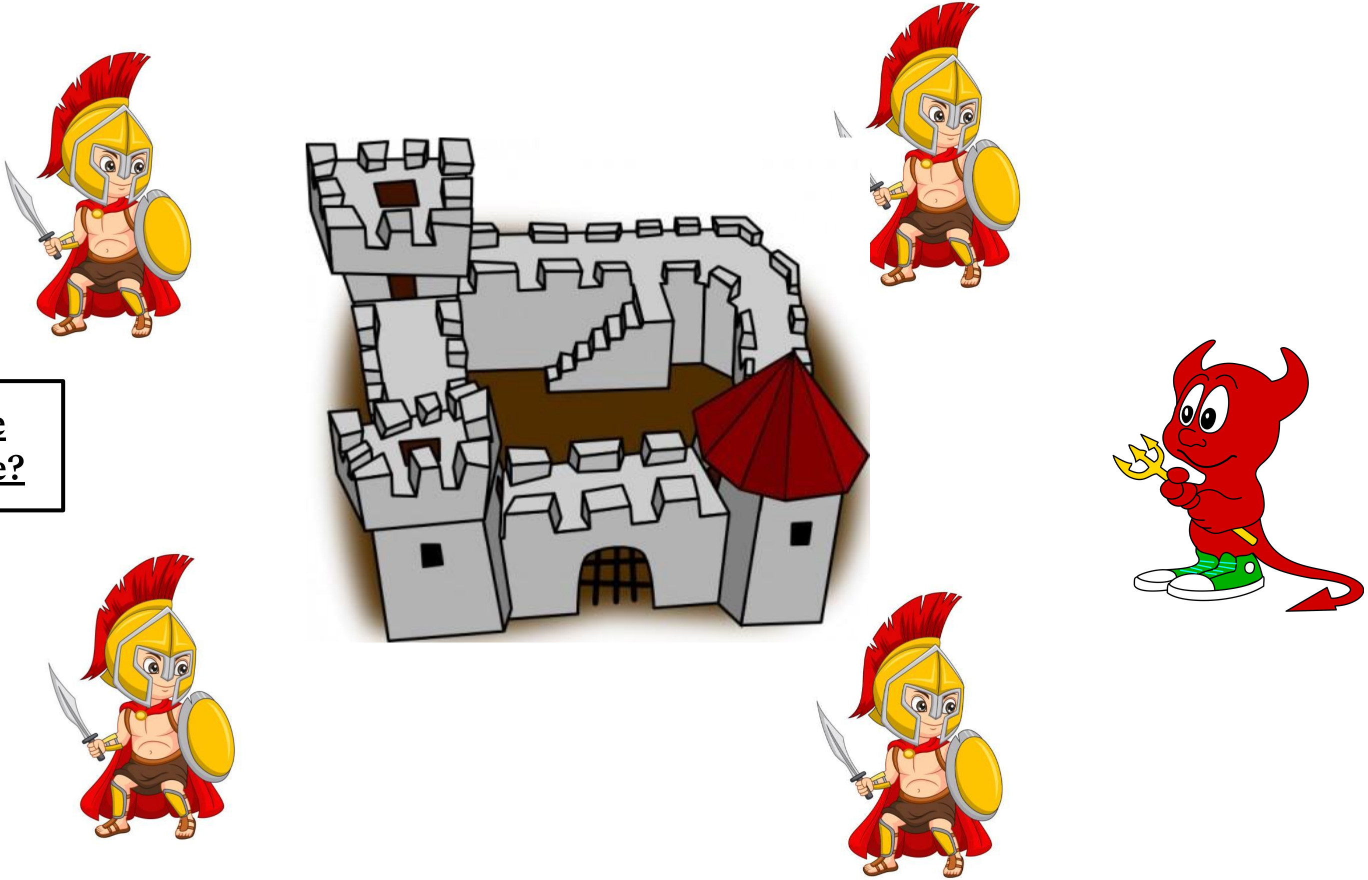
Hyperledger
FOUNDATION

Byzantine Generals' Problem



<u>Why shouldn't one general just decide?</u>

Byzantine Generals’ Problem



Why shouldn’t one general just decide?

What Is Decentralization?

We define **decentralization** as the degree to which a **single entity** or **group of entities** controls something.

What Is Decentralization?

We define **decentralization** as the degree to which a **single entity** or **group of entities** controls something.

This can be measured in a variety of different ways:

- The largest “proportion of control” by any single entity.
- The number of entities it takes to “completely control” a system.
- ...or other more complicated metrics!

What Is Decentralization?

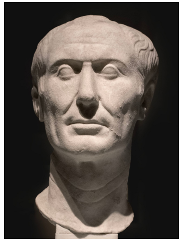
We define **decentralization** as the degree to which a **single entity** or **group of entities** controls something.

This can be measured in a variety of different ways:

- The largest “proportion of control” by any single entity.
- The number of entities it takes to “completely control” a system.
- …or other more complicated metrics!

Absolute Dictatorship

Totally Centralized



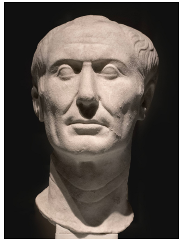
What Is Decentralization?

We define **decentralization** as the degree to which a **single entity** or **group of entities** controls something.

This can be measured in a variety of different ways:

- The largest “proportion of control” by any single entity.
- The number of entities it takes to “completely control” a system.
- …or other more complicated metrics!

Absolute Dictatorship
Totally Centralized



Representative Democracy
Moderately Decentralized

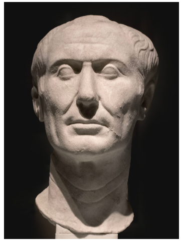
What Is Decentralization?

We define **decentralization** as the degree to which a **single entity** or **group of entities** controls something.

This can be measured in a variety of different ways:

- The largest "proportion of control" by any single entity.
- The number of entities it takes to "completely control" a system.
- ...or other more complicated metrics!

Absolute Dictatorship
Totally Centralized



Representative Democracy
Moderately Decentralized

Direct Democracy
Totally Decentralized*

What Is a Distributed Ledger?

A **blockchain** is an append-only system of record or transaction log.

What Is a Distributed Ledger?

A **blockchain** is an append-only system of record or transaction log.

A **distributed ledger** is a distributed database with decentralized trust.

What Is a Distributed Ledger?

A **blockchain** is an append-only system of record or transaction log.

A **distributed ledger** is a distributed database with decentralized trust.

Most popular blockchains are also distributed ledgers, and most popular distributed ledgers are also blockchains.

What Is a Distributed Ledger?

A **blockchain** is an append-only system of record or transaction log.

A **distributed ledger** is a distributed database with decentralized trust.

Most popular blockchains are also distributed ledgers, and most popular distributed ledgers are also blockchains.

What are popular blockchain systems, abstractly?

What Is a Distributed Ledger?

A **blockchain** is an append-only system of record or transaction log.

A **distributed ledger** is a distributed database with decentralized trust.

Most popular blockchains are also distributed ledgers, and most popular distributed ledgers are also blockchains.

What are popular blockchain systems, abstractly?

bitcoin	A distributed database for “money” with “fully” decentralized trust
	A distributed database for “programs” with “fully” decentralized trust
HYPERLEDGER FABRIC	A distributed database for “programs” with “partially” decentralized trust

Today: Distributed Ledgers

Today we will focus on distributed ledgers rather than blockchains, although our discussion will certainly apply to blockchains that are also distributed ledgers.

Today: Distributed Ledgers

Today we will focus on distributed ledgers rather than blockchains, although our discussion will certainly apply to blockchains that are also distributed ledgers.

The **immutability** (“nothing can be erased”) guarantees of blockchain can be great for public blockchain applications…

…but not always for enterprise applications.

Today: Distributed Ledgers

Today we will focus on distributed ledgers rather than blockchains, although our discussion will certainly apply to blockchains that are also distributed ledgers.

The **immutability** (“nothing can be erased”) guarantees of blockchain can be great for public blockchain applications…

…but not always for enterprise applications.

The core property we will use is decentralized trust.

The “Base Layer” of Web3

Databases are the backbone of much of how technology runs today. Everything from **cloud servers** to **internet hosting** to **businesses** runs on databases.

The “Base Layer” of Web3

Databases are the backbone of much of how technology runs today. Everything from **cloud servers** to **internet hosting** to **businesses** runs on databases.

Distributed ledgers are just **decentralized databases** at their core. Just as databases are core to doday’s technology, we expect distributed ledgers to be **ubiquitous in Web3**.

Adobe Stock

The “Base Layer” of Web3

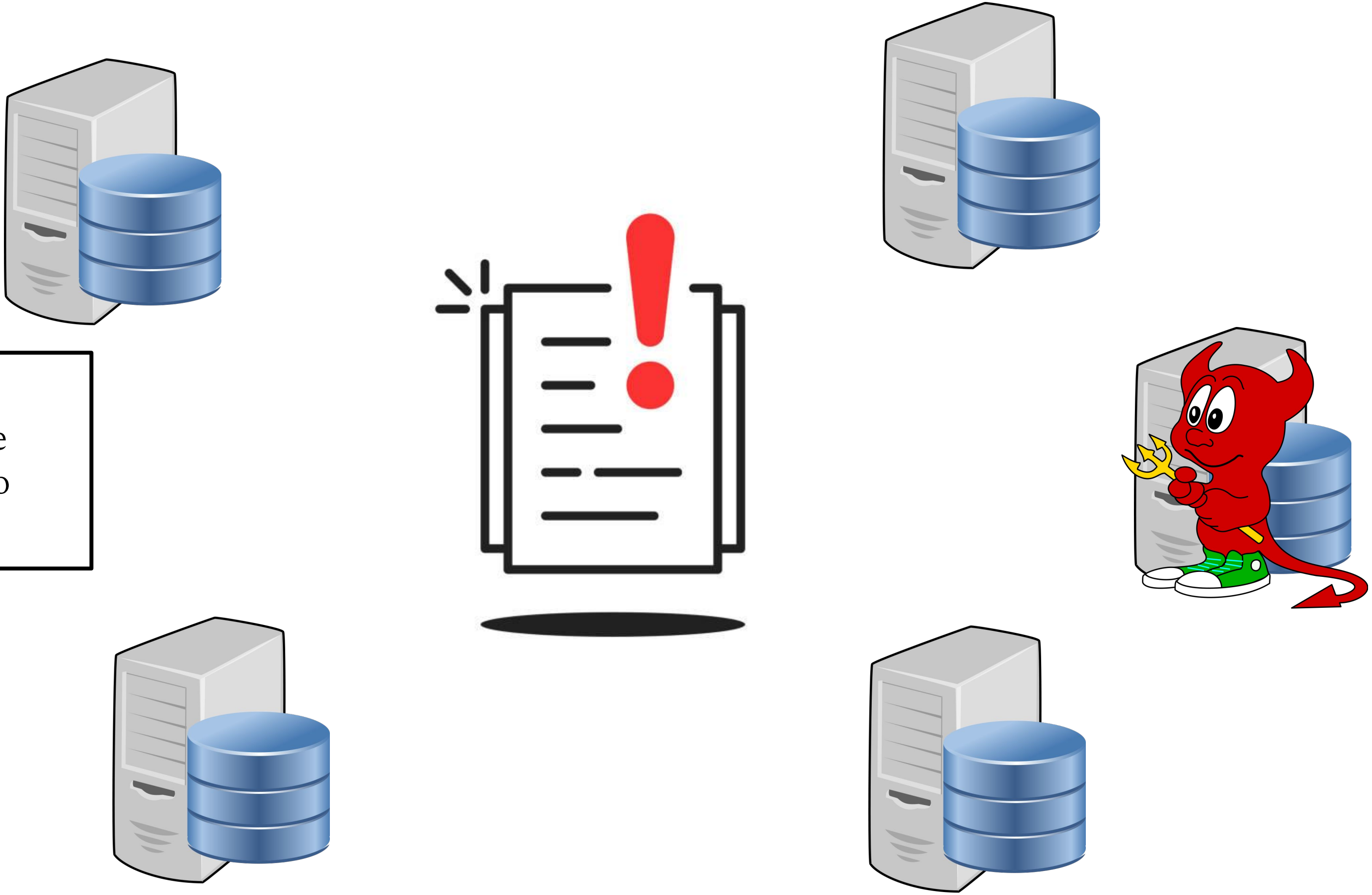
Databases are the backbone of much of how technology runs today. Everything from **cloud servers** to **internet hosting** to **businesses** runs on databases.

Distributed ledgers are just **decentralized databases** at their core. Just as databases are core to doday’s technology, we expect distributed ledgers to be **ubiquitous in Web3**.

Adobe Stock

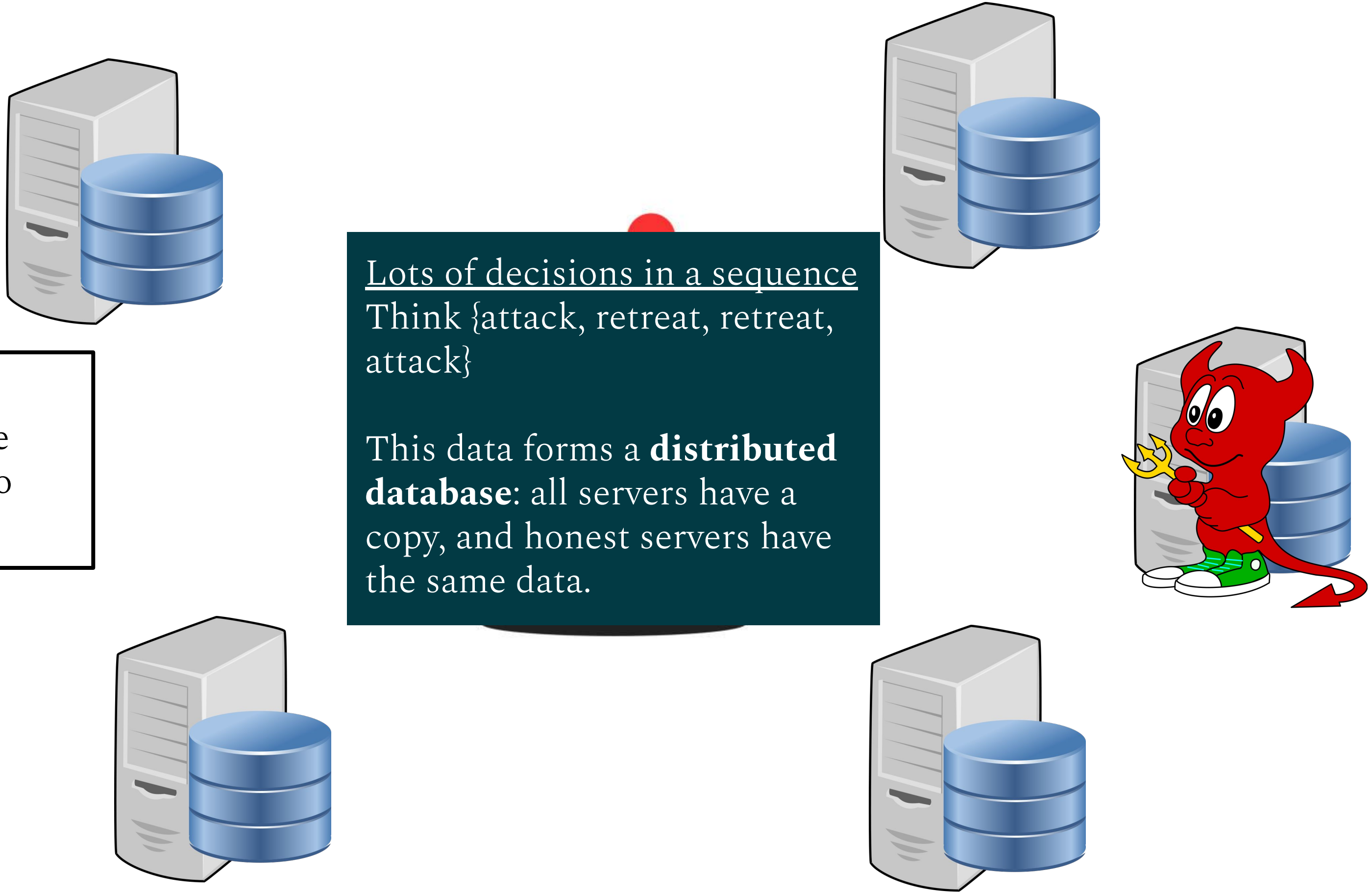
We will initially focus on distributed ledgers, and then extrapolate to general Web3.

We Have Blockchain!



N = 5 Servers:
Deciding on some state that needs to have agreement.

We Have Blockchain!



Lots of decisions in a sequence
Think {attack, retreat, retreat, attack}

This data forms a **distributed database**: all servers have a copy, and honest servers have the same data.

N = 5 Servers:
Deciding on some state that needs to have agreement.

What Is Decentralized Trust?

A database (or blockchain) can be thought as a **store of records**.

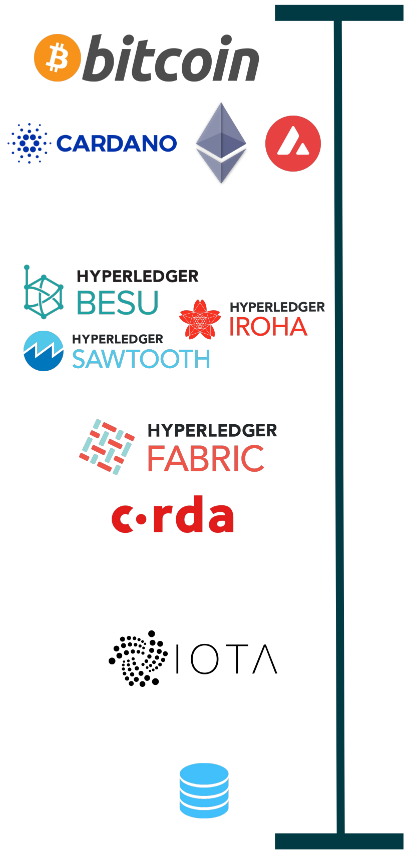
Who gets to decide what records belong in the database?

One person/entity decides → centralized
Many different entities decide → decentralized

Decentralized trust is a **continuum,**
not a “yes or no”

Technically: the consensus algorithm (or lack thereof) of the distributed ledger is the most impactful design choice on decentralization.

Fully Decentralized



Public cryptocurrencies with PoW/PoS consensus, or “public-style” BFT protocols

Distributed ledgers with BFT consensus

Distributed ledgers with crash fault-tolerant consensus

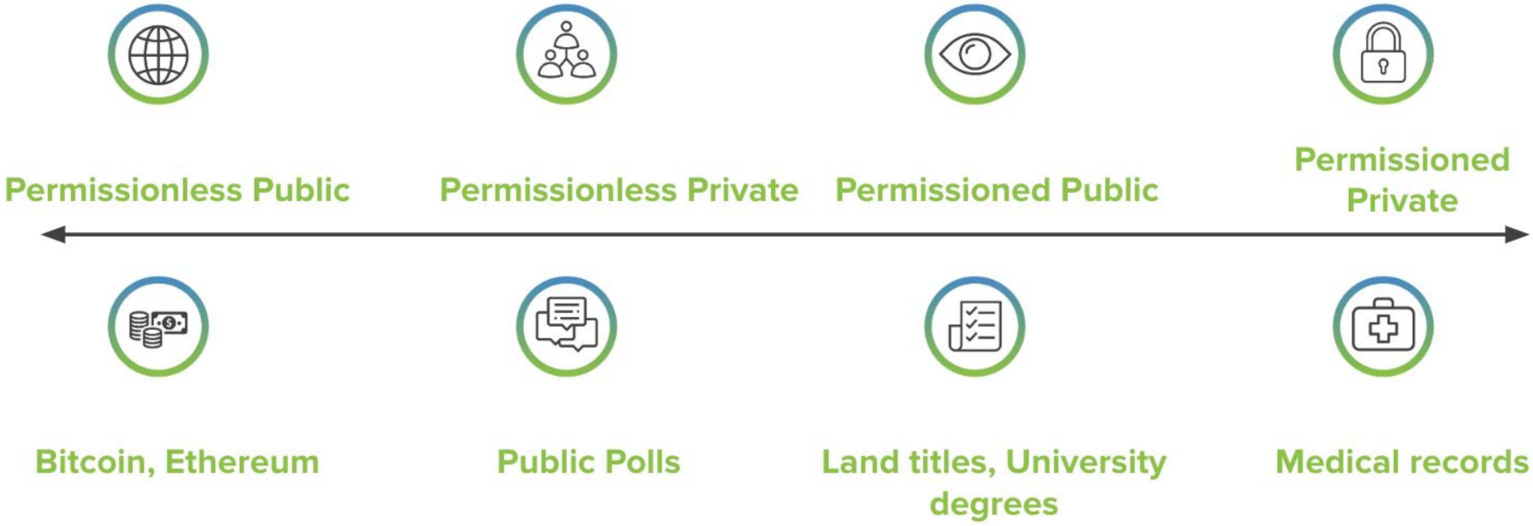
A single entity runs the consensus protocol

Traditional databases

Fully Centralized

Spectrum of Distributed Ledgers

Permissioned vs. Permissionless: Who can write to a Blockchain (i.e., accessibility)
Public vs. Private: Who can read from a Blockchain (i.e., visibility)



Permissionless Public

Permissionless Private

Permissioned Public

Permissioned Private

Bitcoin, Ethereum

Public Polls

Land titles, University degrees

Medical records

Distributed Ledgers on the Spectrum

	PERMISSIONED	PERMISSIONLESS
PUBLIC	Hedera ripple Stellar	Algorand HYPERLEDGER BESU ethereum AVALANCHE
PRIVATE	HYPERLEDGER FABRIC HYPERLEDGER BESU Quorum r3.corda	n/a

Why Decentralized Trust?

Several entities need to agree on some data, but no entity trusts any single other entity to be the “source of truth.”

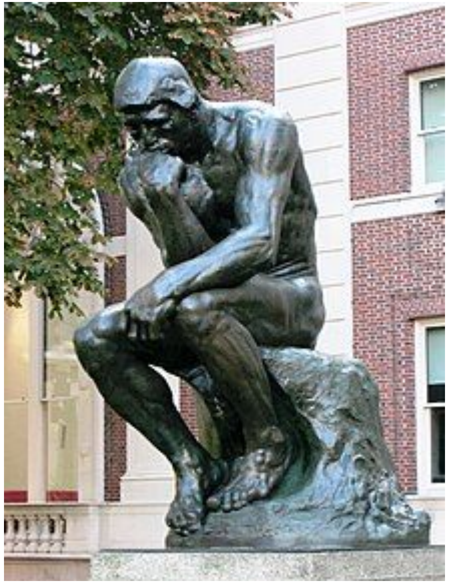
A store of information needs to be made redundant in the case of compromise or attack by a hacker.

The entity that would be the best official “source of truth” for some data doesn’t want to or cannot be responsible for the upkeep of the data.

People responsible for maintaining a data set are dynamic and change quickly.

“Do I need a distributed ledger?” == “Do I need a database with decentralized trust?”

“Do I need a distributed ledger?” == “Do I need a database with decentralized trust?”



If there is one point to take away from my talk today, this is it!

Whenever you think about blockchains or whether you want to use a blockchain, you want to consider:

- What is the information being stored in the “database” (even if it is programmatic)?
- Why is having one centralized entity maintain this information a bad idea, or generally infeasible?

This will make it easy in the future to distinguish cases where distributed ledger use is just “hype” rather than necessary.

Blockchain Drawbacks

Why not always blockchain?

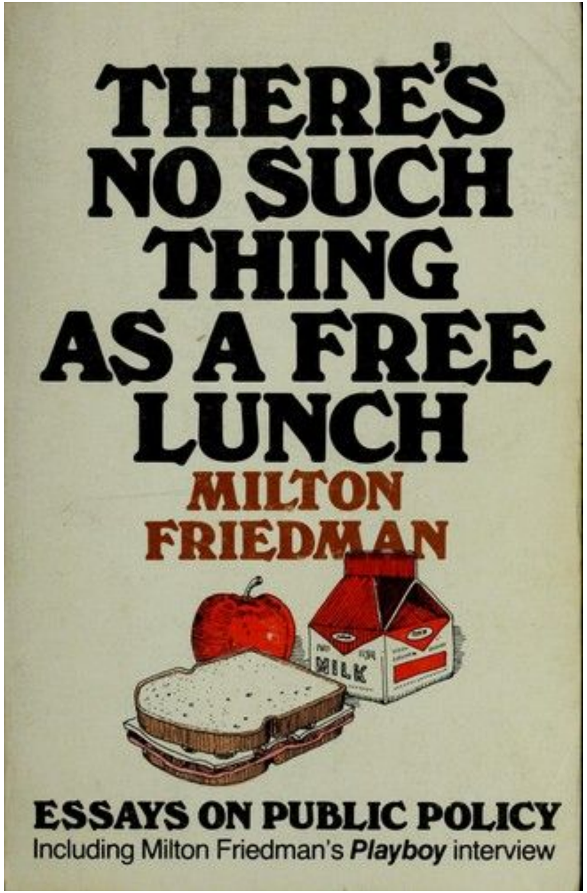
Why not ALWAYS distributed ledgers?

Decentralization is a fantastic tool. But there are always drawbacks to using powerful tools.

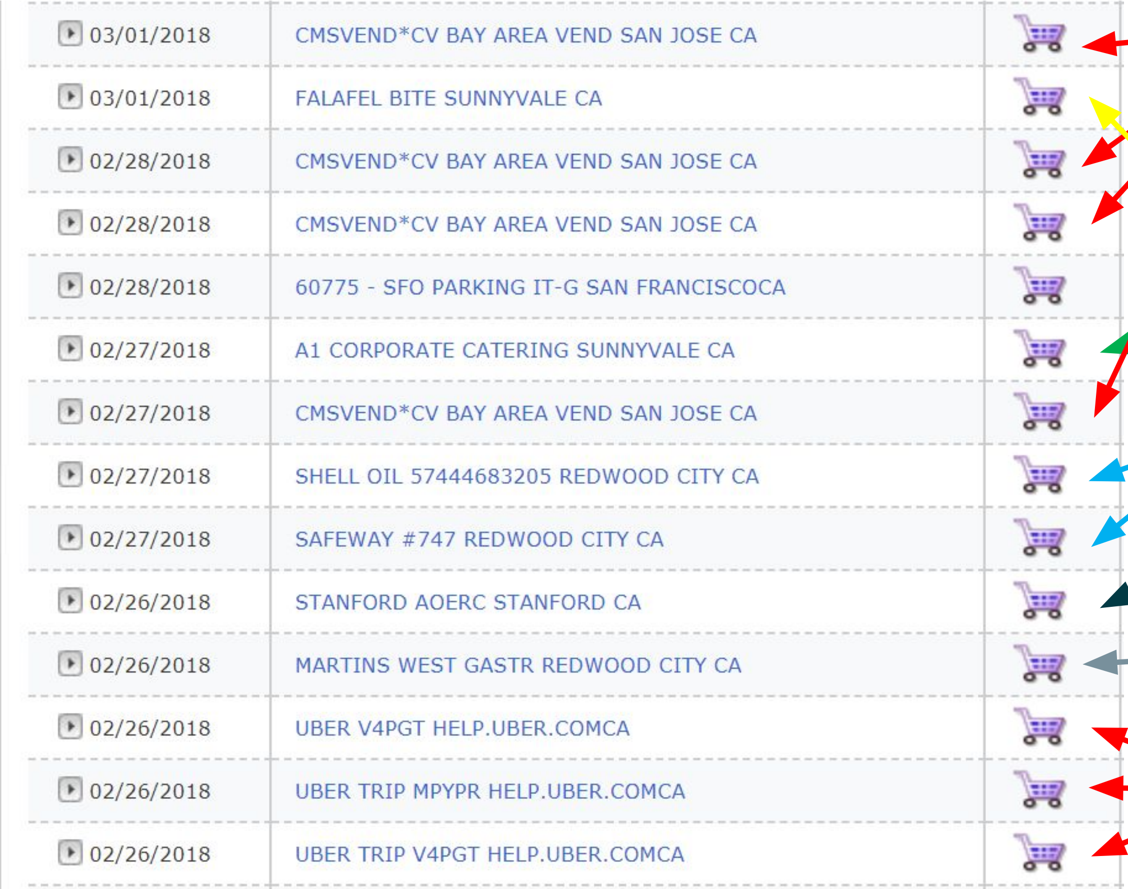
If we use distributed ledgers, there are issues that need to be addressed. Two of the more common that we will cover today:

- Privacy/Confidentiality
- Performance

These can be challenging but we address them in Hyperledger!


THERE'S NO SUCH THING AS A FREE LUNCH
MILTON FRIEDMAN
MILK
ESSAYS ON PUBLIC POLICY
Including Milton Friedman's *Playboy* interview

“We anonymize all users”



03/01/2018	CMSVEND*CV BAY AREA VEND SAN JOSE CA	🛒
03/01/2018	FALAFEL BITE SUNNYVALE CA	🛒
02/28/2018	CMSVEND*CV BAY AREA VEND SAN JOSE CA	🛒
02/28/2018	CMSVEND*CV BAY AREA VEND SAN JOSE CA	🛒
02/28/2018	60775 - SFO PARKING IT-G SAN FRANCISCOCA	🛒
02/27/2018	A1 CORPORATE CATERING SUNNYVALE CA	🛒
02/27/2018	CMSVEND*CV BAY AREA VEND SAN JOSE CA	🛒
02/27/2018	SHELL OIL 57444683205 REDWOOD CITY CA	🛒
02/27/2018	SAFEWAY #747 REDWOOD CITY CA	🛒
02/26/2018	STANFORD AOERC STANFORD CA	🛒
02/26/2018	MARTINS WEST GASTR REDWOOD CITY CA	🛒
02/26/2018	UBER V4PGT HELP.UBER.COMCA	🛒
02/26/2018	UBER TRIP MPYPR HELP.UBER.COMCA	🛒
02/26/2018	UBER TRIP V4PGT HELP.UBER.COMCA	🛒

Snack Machine
I work in the South Bay

Mediterranean Restaurant
I (probably) work in or near Sunnyvale

Fujitsu Cafeteria
I definitely work in Sunnyvale

Grocery/Gas
I (probably) live in Redwood City

Stanford Gym
I (probably) am a Stanford alum

Redwood City Gastropub
I (probably) live in Redwood City

Uber After Gastropub
I (probably) enjoy drinking

Just from my transactions over a few days, you learned that, in 2018:

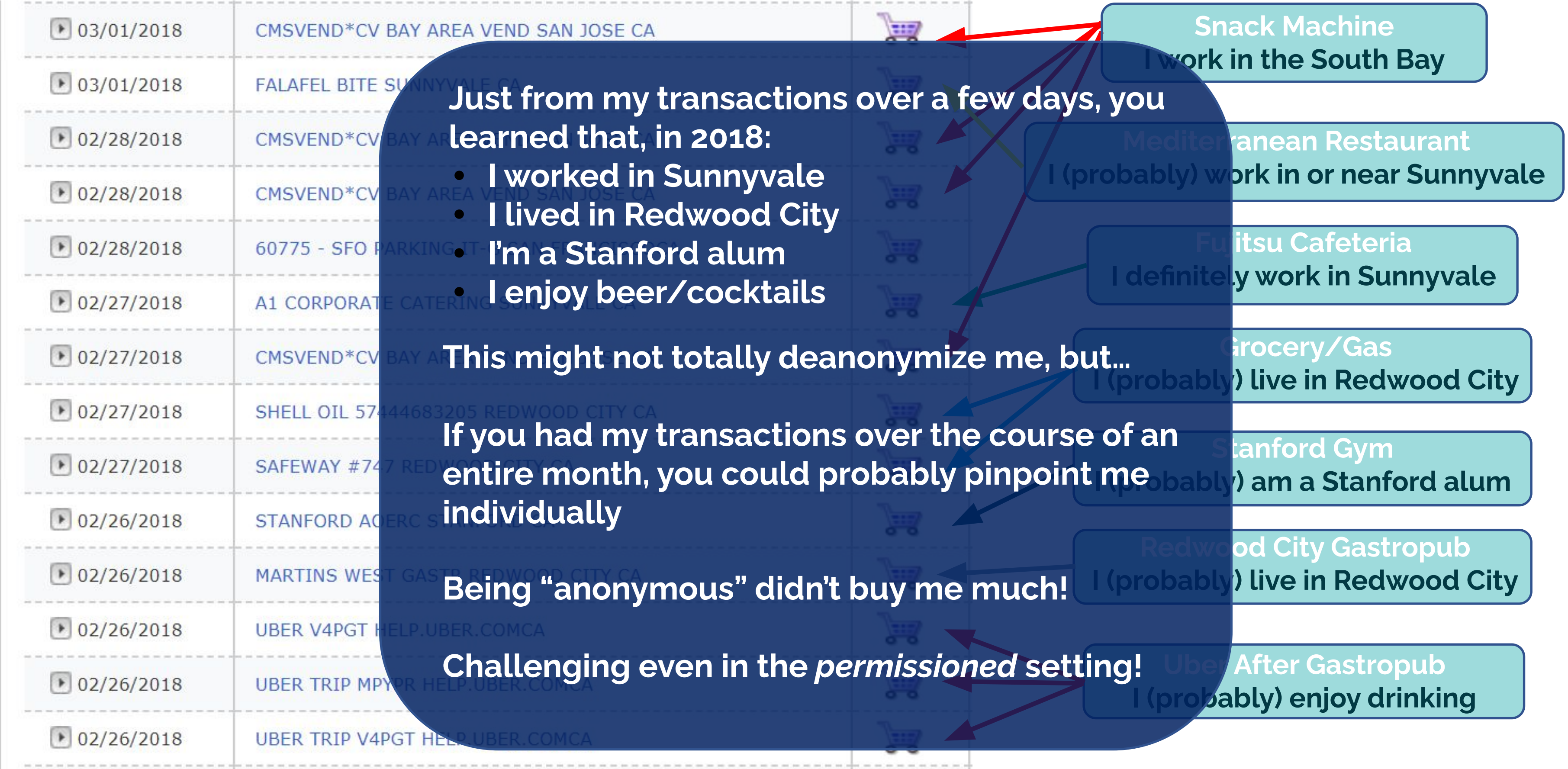
- I worked in Sunnyvale
- I lived in Redwood City
- I'm a Stanford alum
- I enjoy beer/cocktails

This might not totally deanonymize me, but…

If you had my transactions over the course of an entire month, you could probably pinpoint me individually

Being "anonymous" didn't buy me much!

Challenging even in the *permissioned* setting!



03/01/2018	CMSVEND*CV BAY AREA VEND SAN JOSE CA
03/01/2018	FALAFEL BITE SUNNYVALE CA
02/28/2018	CMSVEND*CV BAY AR
02/28/2018	CMSVEND*CV BAY AREA VEND SAN JOSE CA
02/28/2018	60775 - SFO PARKING IT-
02/27/2018	A1 CORPORATE CATERING SUNNYVALE CA
02/27/2018	CMSVEND*CV BAY AR
02/27/2018	SHELL OIL 57444683205 REDWOOD CITY CA
02/27/2018	SAFEWAY #747 REDWOOD CITY CA
02/26/2018	STANFORD AOERC ST
02/26/2018	MARTINS WEST GASTR REDWOOD CITY CA
02/26/2018	UBER V4PGT HELP.UBER.COMCA
02/26/2018	UBER TRIP MPYPR HELP.UBER.COMCA
02/26/2018	UBER TRIP V4PGT HELP.UBER.COMCA

Snack Machine
I work in the South Bay

Mediterranean Restaurant
I (probably) work in or near Sunnyvale

Fujitsu Cafeteria
I definitely work in Sunnyvale

Grocery/Gas
I (probably) live in Redwood City

Stanford Gym
I (probably) am a Stanford alum

Redwood City Gastropub
I (probably) live in Redwood City

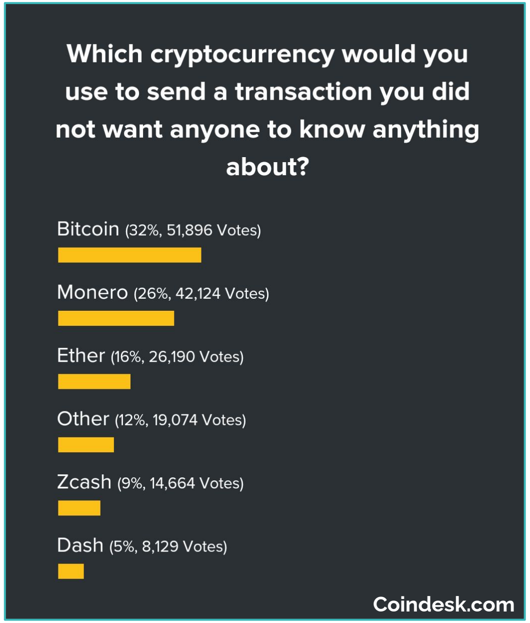
Uber After Gastropub
I (probably) enjoy drinking

Problems Even in the Permissionless Setting

Many cryptocurrencies incorporate privacy / anonymity techniques

Exact privacy and confidentiality guarantees are not always explicit!

Users don't agree on the best way to handle privacy and confidentiality

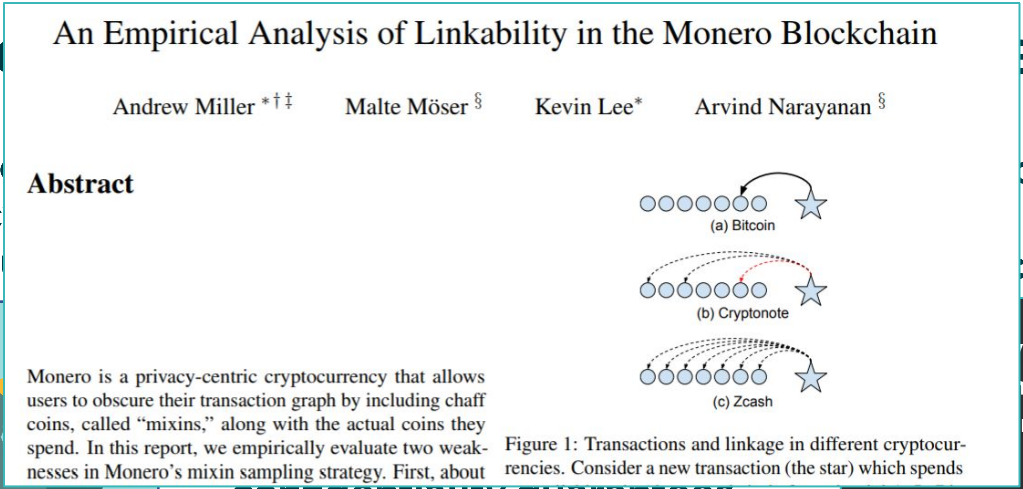

Which cryptocurrency would you use to send a transaction you did not want anyone to know anything about?

Coindesk.com

Probl etting

Many cryptocurren
incorporate privac
anonymity techniq

on't agree on the
to handle privacy
confidentiality



An Empirical Analysis of Linkability in the Monero Blockchain

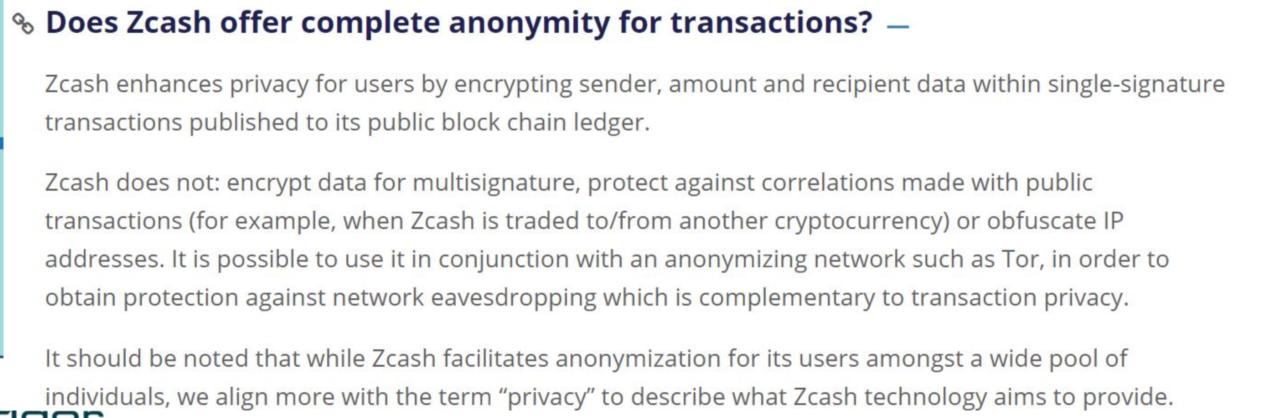
Andrew Miller [*†‡] Malte Möser [§] Kevin Lee[*] Arvind Narayanan [§]

Abstract

Monero is a privacy-centric cryptocurrency that allows users to obscure their transaction graph by including chaff coins, called "mixins," along with the actual coins they spend. In this report, we empirically evaluate two weaknesses in Monero's mixin sampling strategy. First, about

Figure 1: Transactions and linkage in different cryptocurrencies. Consider a new transaction (the star) which spends

confidentiality guarantees
are not always explicit!

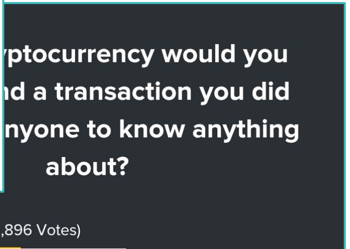


Does Zcash offer complete anonymity for transactions?

Zcash enhances privacy for users by encrypting sender, amount and recipient data within single-signature transactions published to its public block chain ledger.

Zcash does not: encrypt data for multisignature, protect against correlations made with public transactions (for example, when Zcash is traded to/from another cryptocurrency) or obfuscate IP addresses. It is possible to use it in conjunction with an anonymizing network such as Tor, in order to obtain protection against network eavesdropping which is complementary to transaction privacy.

It should be noted that while Zcash facilitates anonymization for its users amongst a wide pool of individuals, we align more with the term "privacy" to describe what Zcash technology aims to provide.



yptocurrency would you
nd a transaction you did
nyone to know anything
about?

Bitcoin (32%, 51,896 Votes)

Coindesk.com

“Everything is enrypted or hashed—
No data is given in the clear”

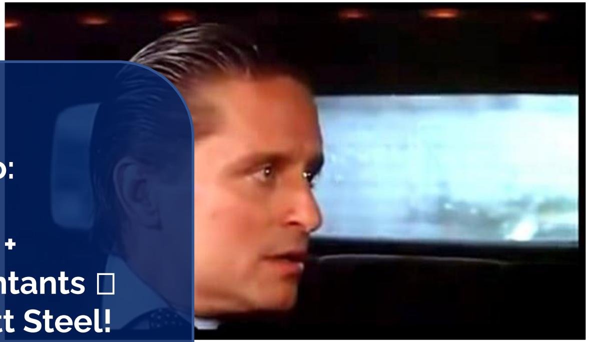
Wall Street: the movie

© 20th Century Fox

DAY Bud watches, wondering what to do as the plane taxies down the runway. He spots the flight mechanic and the answer comes to him. He starts running towards the mechanic.
EXT. APRON - DAY Bud races up to the mechanic.
BUD Oh shit, don't tell me Mr. Wildman was on board that plane? (the mechanic nods) My boss is gonna kill me. I was supposed to give him this. (holding his notebook) You know where that plane is going?
MECHANIC (walking off) Erie, Pennsylvania...
INT. PHONE BOOTH - AIRLINES TERMINAL - DAY BUD (into phone, proudly) ...after spending the morning at Kahn, Seidelman -- on the 14th floor, the junk bond department -- where Shane Mora works -- he had lunch at La Cirque with a group of well-dressed heavyset bean- counters... (Gekko voice back: "the adjectives are redundant, sport") ...he later stopped off at Morgan. I'd say from all the palm-pressing and sweet smiling going on that Larry got some nice fat financing...

INT. GEKKO LIMOUSINE - HEADING DOWN PARK AVENUE - DAY Alex and Susan are with him. Gekko playing the computer, eyes lighting up on the phone.
GEKKO ...bright but not bright enough, Sherlock, roll the dice and play a little monopoly... what box would Sir Lawrence land on in Erie, Pennsylvania?
INT. PHONE BOOTH - DAY Bud slapping his face, realizing.
BUD Jesus Christ, he's buying Anacott Steel!
INT. GEKKO LIMO - DAY Gordon already has the closing figures punched up on his quotron. Calls his shot.

Wall Street: the movie



© 20th Century Fox

Bud Fox and Gordon Gecko:

Lawrence flying to Erie, PA +
Lawrence talked to accountants →
Lawrence is buying Anacott Steel!

Side channel information like this is everywhere on blockchains!

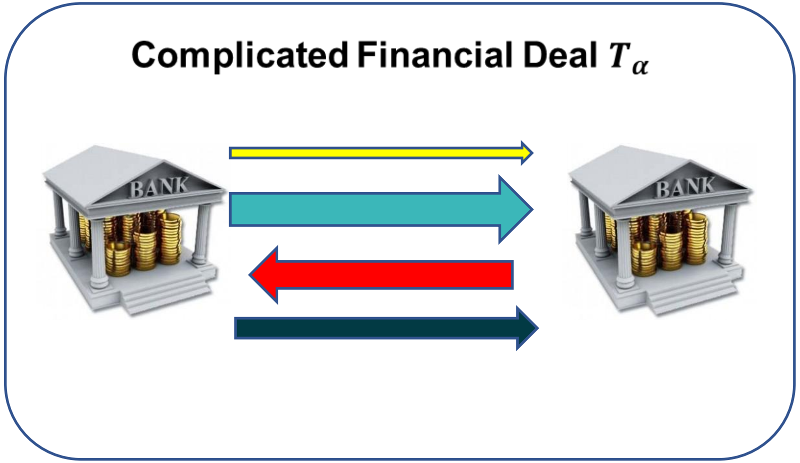
DAY Bud watches, wondering what to do as the plane taxies down the runway. He spots the flight mechanic and the answer comes to him. He starts running towards the mechanic.
EXT. APRON - DAY Bud races up to the mechanic.
BUD Oh shit, don't tell me Mr. Wildman was on board that plane? (the mechanic nods) My boss is gonna kill me. I was supposed to give him this. (holding his notebook) You know where that plane is going?
MECHANIC (walking off) Erie, Pennsylvania...
INT. PHONE BOOTH - AIRLINES TERMINAL - DAY BUD (into phone, proudly) ...after spending the morning at Kahn, Seidelman -- on the 14th floor, the junk bond department -- where Shane Mora works -- he had lunch at La Cirque with a group of well-dressed heavyset bean- counters... (Gekko voice back: "the adjectives are redundant, sport") ...he later stopped off at Morgan. I'd say from all the palm-pressing and sweet smiling going on that Larry got some nice fat financing...

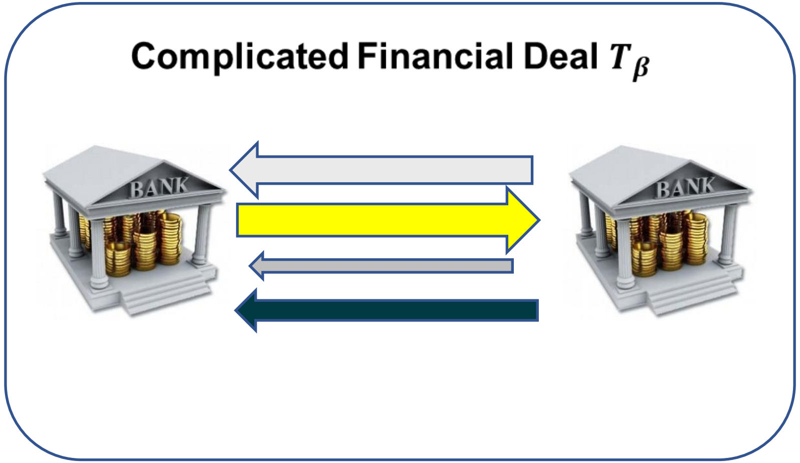
INT. GEKKO LIMOUSINE - HEADING DOWN PARK AVENUE - DAY Alex and Susan are with him. Gekko playing the computer, eyes lighting up on the phone.
GEKKO ...bright but not bright enough, Sherlock, roll the dice and play a little monopoly... what box would Sir Lawrence land on in Erie, Pennsylvania?
INT. PHONE BOOTH - DAY Bud slapping his face, realizing.
BUD Jesus Christ, he's buying Anacott Steel!
INT. GEKKO LIMO - DAY Gordon already has the closing figures punched up on his quotron. Calls his shot.

Hyperledger FOUNDATION

Transaction Patterns

Even if we have fully zero-knowledge transactions, the mere fact that transactions exist in certain patterns could break privacy or confidentiality!


Complicated Financial Deal T_α


Complicated Financial Deal T_β

We can tell whether T_α or T_β happened based on transaction flow!

What cryptography you use ≠
what security you get!

More Formal Guarantees:

Privacy Guarantees

~~“My users are anonymized”~~

→

It is cryptographically hard to distinguish the participants in any transaction from random

Data Security

~~“Everything is enrypted or hashed – Nothing is given in the clear”~~

→

It is cryptographically hard to learn any information about any transaction on the blockchain

Hyperledger FOUNDATION

More Formal Guarantees:

Privacy Guarantees

“My users are anonymized” ✗ →

It is cryptographically hard to distinguish the participants in any transaction from random

Data Security

“Everything is enrypted or hashed – Nothing is given in the clear” ✗ →

It is cryptographically hard to learn any information about any transaction on the blockchain

To Design Secure Systems:

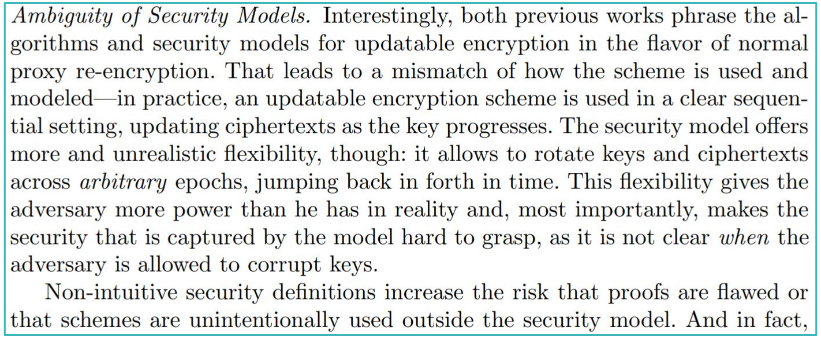
Start by writing down the security properties you need.

THEN pick the tools you need to achieve the desired security properties.

Hyperledger FOUNDATION

But Defining Security Is Hard!

- Yes, it can be—even for people who have spent their entire working lives studying cryptography.
- When in doubt, ask a cryptographer!
- Weaker guarantees are OK too!
  - Weak guarantee with proof > strong claim without!



Ambiguity of Security Models. Interestingly, both previous works phrase the algorithms and security models for updatable encryption in the flavor of normal proxy re-encryption. That leads to a mismatch of how the scheme is used and modeled—in practice, an updatable encryption scheme is used in a clear sequential setting, updating ciphertexts as the key progresses. The security model offers more and unrealistic flexibility, though: it allows to rotate keys and ciphertexts across *arbitrary* epochs, jumping back in forth in time. This flexibility gives the adversary more power than he has in reality and, most importantly, makes the security that is captured by the model hard to grasp, as it is not clear *when* the adversary is allowed to corrupt keys.

Non-intuitive security definitions increase the risk that proofs are flawed or that schemes are unintentionally used outside the security model. And in fact,

From “Updatable Encryption with Post-Compromise Security,” Anja Lehmann and Bjorn Tackmann, CRYPTO 2018

OK, We've Defined What Security Means (For Us)

Next step: build a system that meets the definition(s) of security.

Really two steps:

1. Build a system.
2. Prove it meets the required definition(s) of security.

Hyperledger FOUNDATION

Proofs Are Hard Too!

Yes, they are! But they are important to get right.

When in doubt—ask for help!

- Lots of resources in Hyperledger (more on this later)
- “Don’t get cryptography, get a cryptographer”!

If you are building a system that needs strong privacy and confidentiality guarantees, you should probably have a cryptographer on your team!

INTERNATIONAL ASSOCIATION FOR CRYPTOLOGIC RESEARCH

iacr

Privacy and Confidentiality Are Tricky!

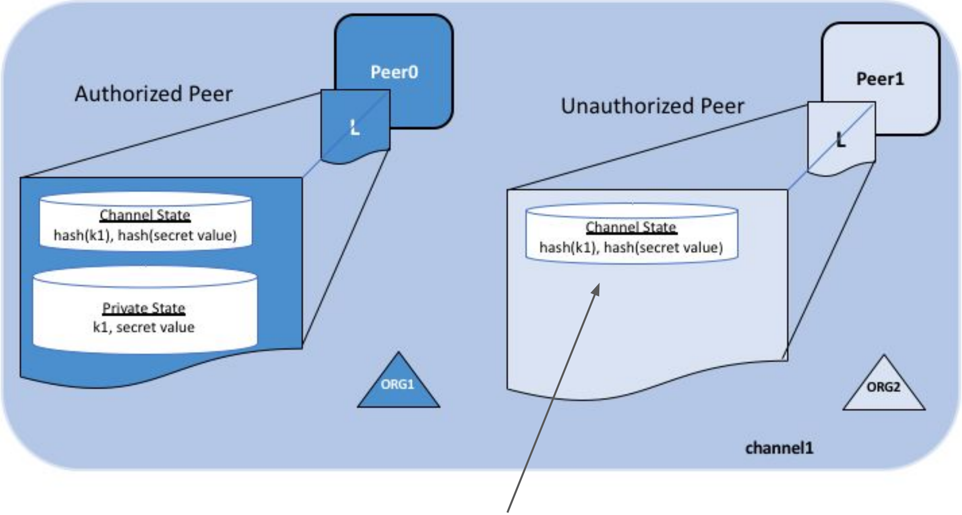
By default, everyone can see all transactions on a ledger. This makes privacy hard! As an example, we have several solutions in Hyperledger aimed at preserving privacy and confidentiality properties–but they always lower performance!

Example:

HYPERLEDGER FABRIC

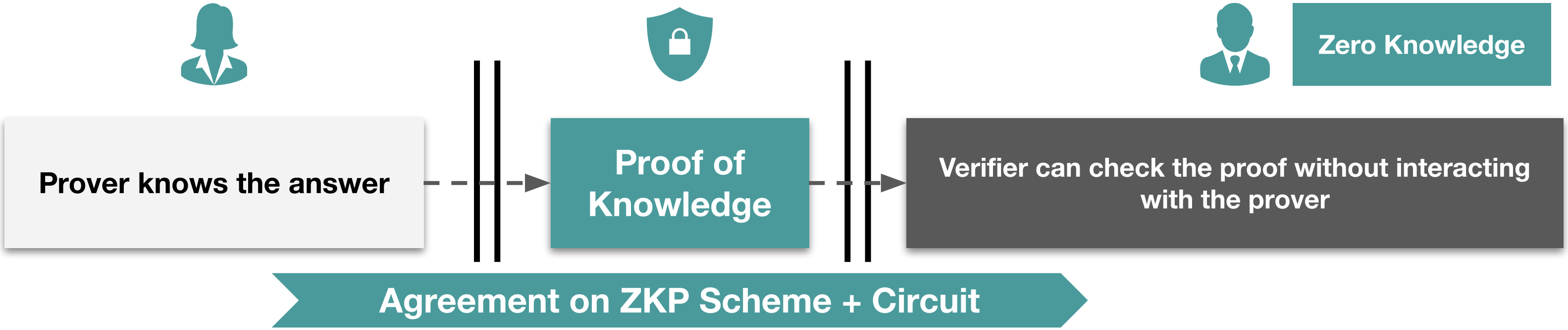
Private Data Collections/ Private Transactions

Main idea: limit the data others on the blockchain see by only posting hashes of sensitive data rather than the data itself.



The unauthorized peer sees no data in the clear.

Noninteractive Zero Knowledge Proofs



Zero Knowledge

Prover knows the answer

Proof of Knowledge

Verifier can check the proof without interacting with the prover

Agreement on ZKP Scheme + Circuit

ZKP Scheme

Different schemes with different properties
Ex: Groth16, Plonk, Stark
gnark library

Circuit

Public Inputs
Private Inputs
Statements

Private transaction vs zk-SNARKs Privacy

Private transactions

Concept

Send and execute transactions only to/by a subset of participants. There is 1 public state and N private state for each privacy group/set of participants.

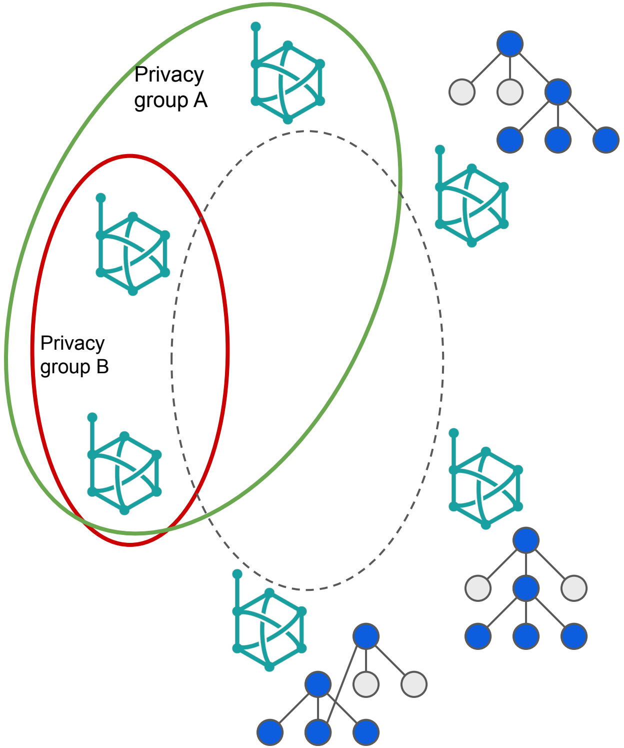
Pro

- EVM compatible
- Fully private

Cons

- Vulnerable to DDoS
- Siloed private state/no unified state → many use-cases (incl. assets) are impossible

Use-cases



zk-SNARKs privacy

Concept

Account state is split across actors, transactions and state are hashed in a merkle tree, zk-SNARKs are generated to ensure correctness of the protocol and prevent double spend.

Pros

- Unified state - can perform token transfers at scale, fully private
- Higher throughput

- zk-EVMs make code compatible

Cons

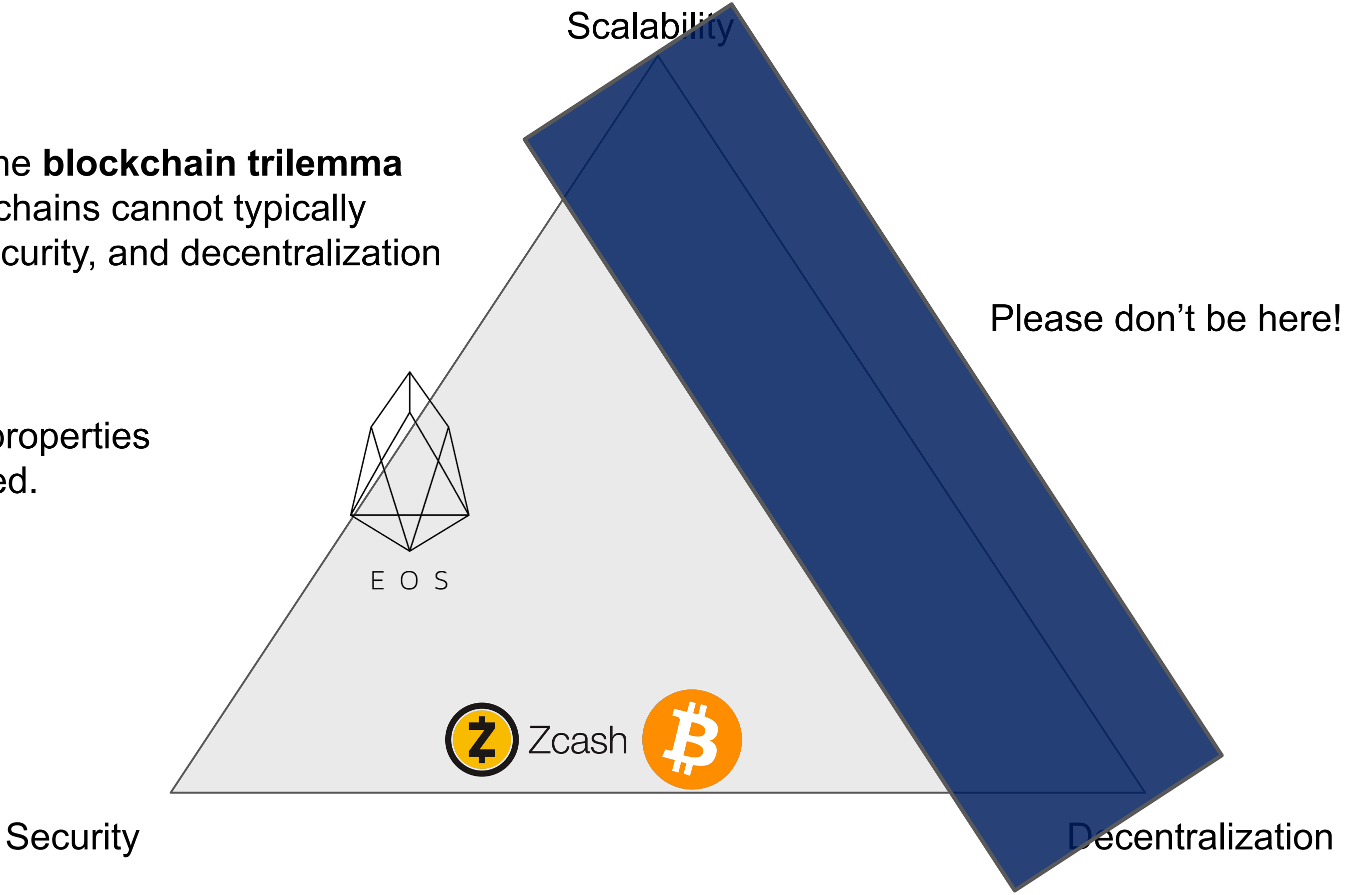
- zk-EVMs are nascent

- Require heavy machines to operate (but the technology is progressing at a high speed)

Blockchain Trilemma

Coined by Vitalik Buterin, the **blockchain trilemma** refers to the fact that blockchains cannot typically achieve all of scalability, security, and decentralization at the same time.

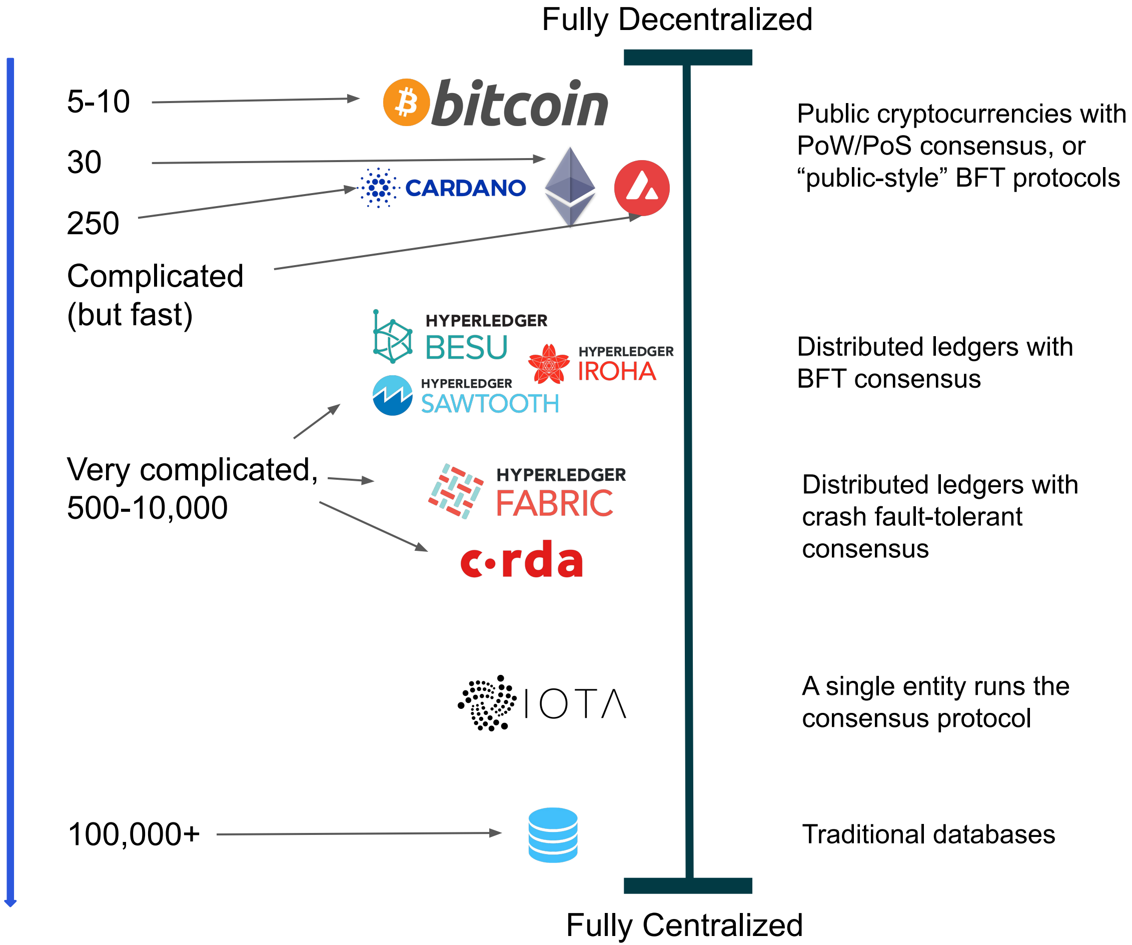
Tradeoffs between these properties must be carefully considered.



Distributed Ledger Performance

<u>In general</u>:
more decentralization →
slower performance

As in life, there are tradeoffs in distributed ledgers. If you have decided that you need a DL, the challenge is to pick where on the decentralization continuum balances your application's needs.



Fully Decentralized

5-10

30

250

Complicated (but fast)

Very complicated, 500-10,000

100,000+

Fully Centralized

Public cryptocurrencies with PoW/PoS consensus, or "public-style" BFT protocols

Distributed ledgers with BFT consensus

Distributed ledgers with crash fault-tolerant consensus

A single entity runs the consensus protocol

Traditional databases

Even More Distributed Ledger Tradeoffs!

There are lots of tradeoffs in distributed ledgers. We can optimize for each of the following, but generally at the expense of the others on the list

- Performance
- Privacy and confidentiality
- Decentralization
- Generality/expressivity of contract languages (e.g. UTXO vs account model)
- Number of users participating in consensus
- …

We have to carefully choose how we build a blockchain to ensure the properties we need while still getting good performance.

But main and most universal tradeoff: **decentralization vs. performance**.

Layers of Decentralized Trust

There are many different components of decentralized trust:

Code Layer: Who implements and maintains the project?

Specification/Architectural Layer: Who decides the specs? Who sets the roadmap for the project?

On-Chain Consensus: How is consensus on the DLT managed? What is the protocol itself, and is it decentralized?

Off-Chain Consensus (Governance): How are the rules above changed? What is the project governance and legal framework around the DLT?

Application Layer: Are the main applications of the blockchain inherently decentralized?

You Are Only as Decentralized as Your Weakest Link!

Code Layer:

Specification/Architectural Layer:

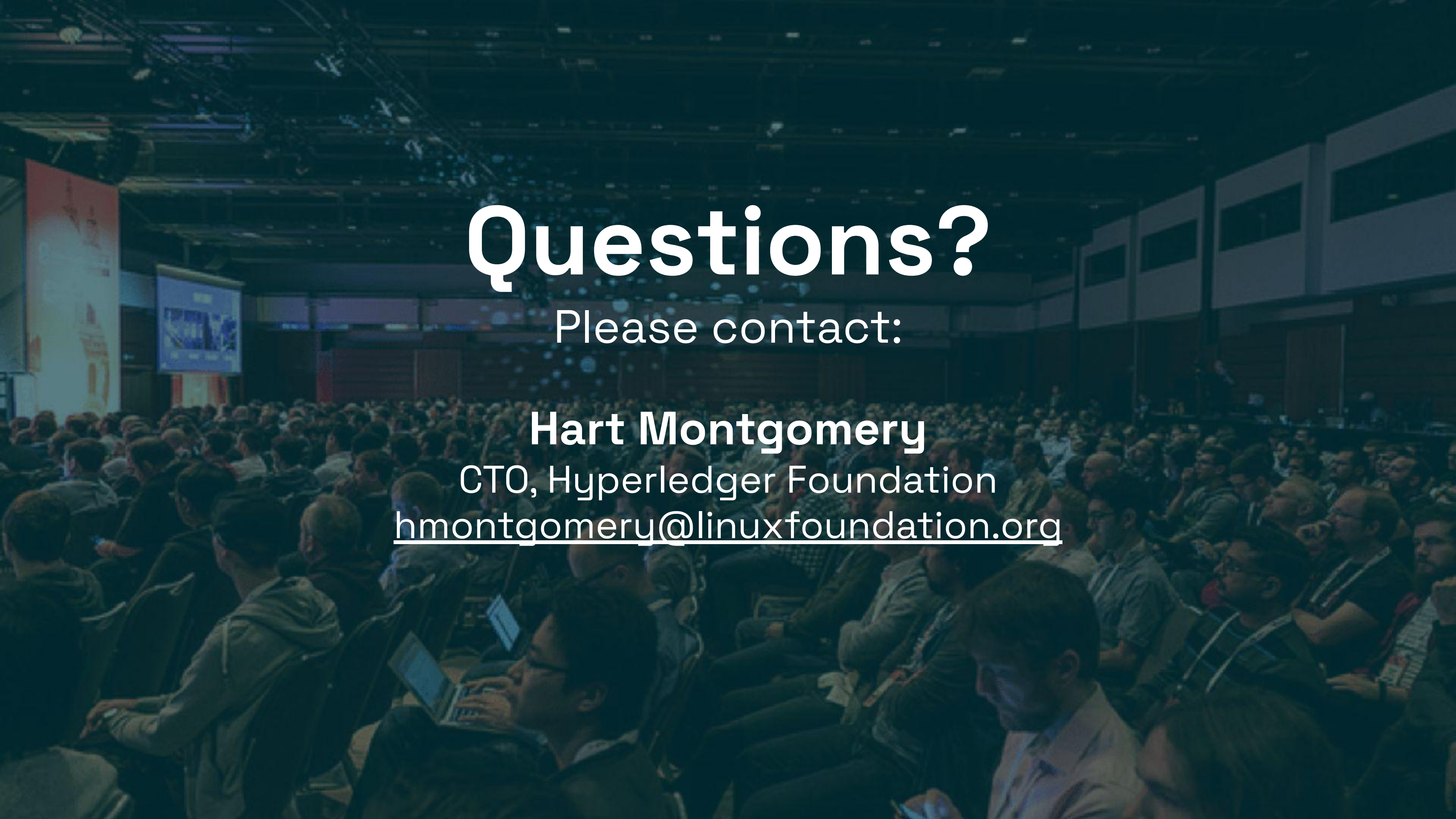
On-Chain Consensus:

Off-Chain Consensus (Governance):

Application Layer:

If any one of these layers is centralized–and thus, controlled by one party–then the entire blockchain can be effectively controlled by one party–and thus, is centralized.

Why not just have this single party run the system in a centralized way?



Questions?

Please contact:

Hart Montgomery
CTO, Hyperledger Foundation
hmontgomery@linuxfoundation.org

Antitrust Policy Notice

Linux Foundation meetings involve participation by industry competitors, and it is the intention of the Linux Foundation to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

Examples of types of actions that are prohibited at Linux Foundation meetings and in connection with Linux Foundation activities are described in the Linux Foundation Antitrust Policy available at http://www.linuxfoundation.org/antitrust-policy.

If you have questions about these matters, please contact your company counsel, or if you are a member of the Linux Foundation, feel free to contact Andrew Updegrove of the firm of Gesmer Updegrove LLP, which provides legal counsel to the Linux Foundation.

Hyperledger FOUNDATION