Blockchains & Cryptocurrencies

Introduction

Instructors: Matthew Green & Abhishek Jain
Johns Hopkins University - Spring 2019

Our background

- Matt: mainly work in applied cryptography (TLS, messaging systems, privacy-preserving protocols)

  Co-founded a private cryptocurrency (Zcash) and boy was that weird

- Abhishek: works on theoretical cryptography, MPC, obfuscation

What is a blockchain?

- A distributed ledger or database ("DLT")
- Used for building decentralized cryptocurrencies such as Bitcoin
- Several other applications such as distributed Domain Name system (DNS), Public-Key Infrastructure (PKI), stock trade database, etc.

Course objectives

- Understanding the mechanics of blockchains
- Understanding why current implementations work
- Understanding the necessary cryptographic background
- Exploring applications of blockchains to cryptocurrencies and beyond
- Understanding limitations of current blockchains/DLT tech

Course objectives (contd)

- Introduction to recent exciting research
- **Main course goal:**
  Extend this research
  - Entrepreneurial or research projects by student teams

Disclaimer

This is not a finance course on cryptocurrencies.
You should not expect to be taught how to
invest in cryptocurrencies or how to become a
millionaire overnight.

Disclaimer

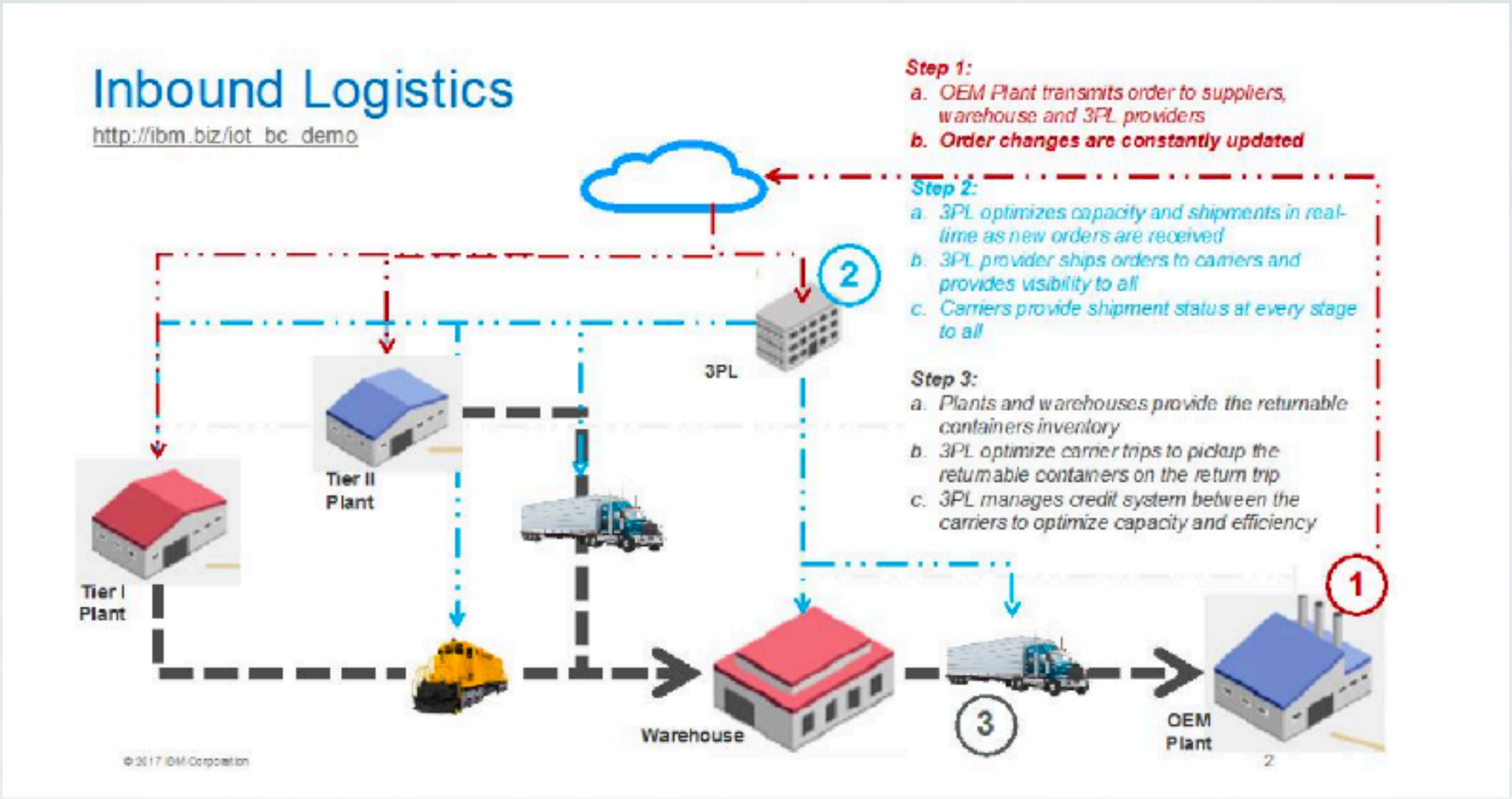
This is not a finance course on cryptocurrencies. You should not expect to be taught how to invest in cryptocurrencies or how to become a millionaire overnight.

(Unless you're already a billionaire — then we can certainly help you become a millionaire.)

Pre-requisites

- No background in cryptography is necessary. However, the following are expected:
  - Basic mathematical maturity
  - Comfort with basic probability
  - Basic familiarity with asymptotic (Big-O) notation
  - Programming capability (in Python/Java, etc.)

Boring course logistics



Source: https://www.ibm.com/blogs/internet-of-things/blockchain-automotive-supply-chain/

Resources

- **Course website & syllabus:**

  https://github.com/maxzinkus/BlockchainsAndCryptocurrencies-Spring2019/wiki

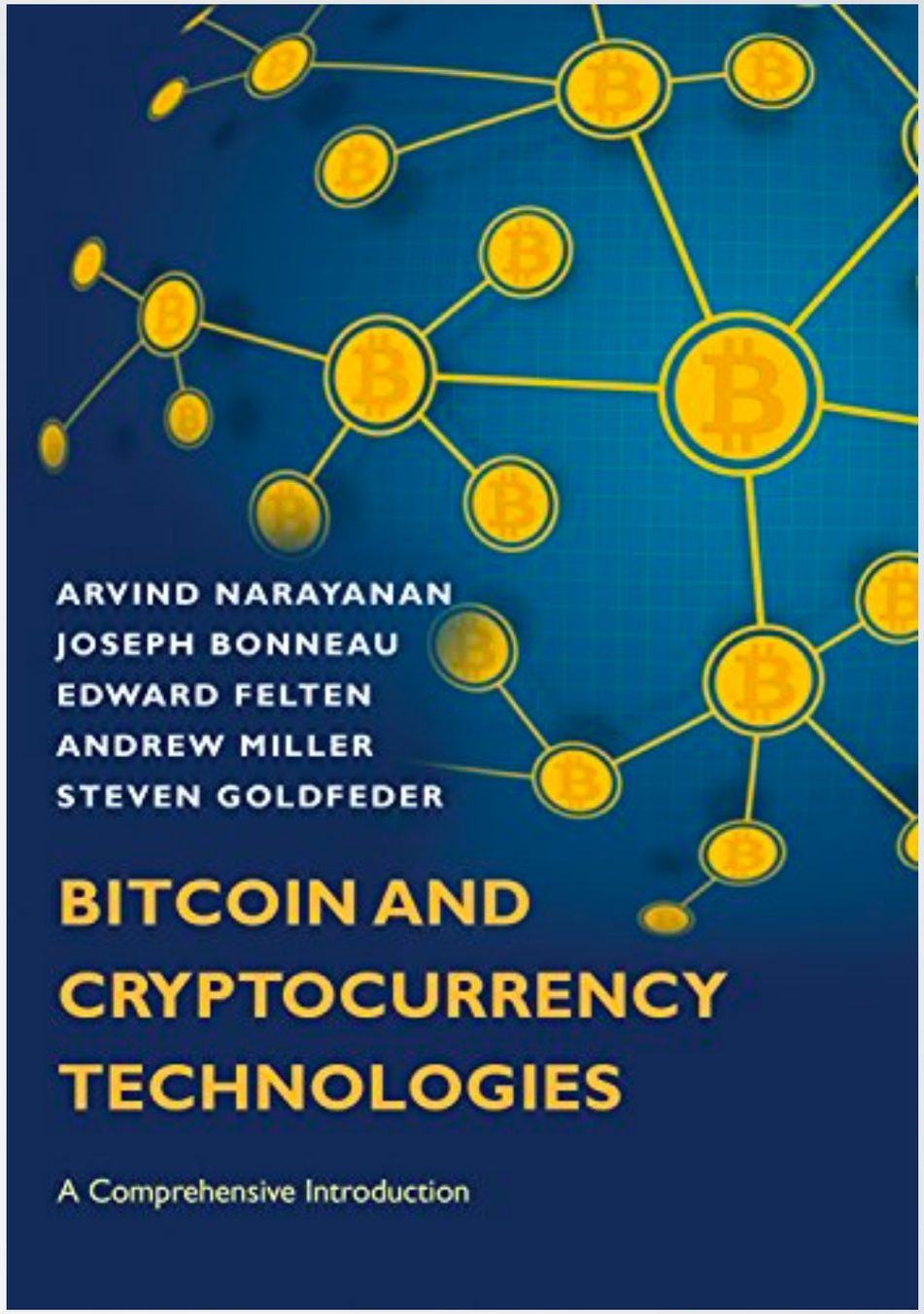
  (Or visit my/Abhishek's website, or Google)

- **Piazza:** https://piazza.com/class/jrcardqygz1ub

  (Or search for 601.441/641, course title)

Texts/Readings

- **Text:** NBFMG (right)
- However, many readings will be drawn from online papers and resources
- Readings on syllabus are <u>assigned</u> the day they are listed, will be <u>discussed</u> during the following lecture



Texts/Readings

Course Syllabus 2019

acrypto edited this page 2 days ago · 1 revision

Dates and topics are subject to change. Reading is assigned on the day specified, and will be discussed in the following class. All assignments are submitted via Gradescope.

1/28/19: Introduction (Matt)

- Reading: NBFMG Chapter 0

1/30/19: Crypto background (AJ)

- Reading: NBFMG Chapter 1
- Reading: Mihir's Notes
- **Assignment 1 is out**

TA & Office Hours

- **TA:** Max Zinkus (zinkus@cs.jhu.edu)

  TA Office hours:
  Tuesday 1-2:30pm (Location TBD)

  Instructor Office hours:
  Tuesday 2-3pm (Malone 313/315)

Grading & Exams

- **Grading:**

  60% assignments+exams,
  40% project

- **Exams:**

  Midterm exam: 3/13
  Final exam: as assigned by JHU

- **Assignments:**

  Programming & written, submit
  via Gradescope. Code: **MYRR43**

Course project

- This is a research-quality project, conducted by groups of 1-3 students. You must have instructor approval for your project topic.

- There is an (unfinished) list of project ideas on the course website. Other ideas are also fine with approval.

- Deliverables: new software, high-quality written report, detailed presentation (choose 2)

- **1-page proposal due 2/27**

Honor Code

- Except where explicitly marked, assignments and exams are individual work. You're expected to do your own work on these. Don't give or receive exam-specific assistance on these.
- See the JHU academic integrity code.
- Exceptions for general-purpose programming advice, etc.
- We hope never to discuss this again.

Honor Code ++, Cryptocurrency edition

- Many legal aspects are unsettled in the blockchain/cryptocurrency space
- E.g., what happens if you discover and exploit a vulnerability in an experimental blockchain project?
- In this course we practice responsible disclosure. If this comes up *e.g.*, in your course project, **see the TA or instructors.**

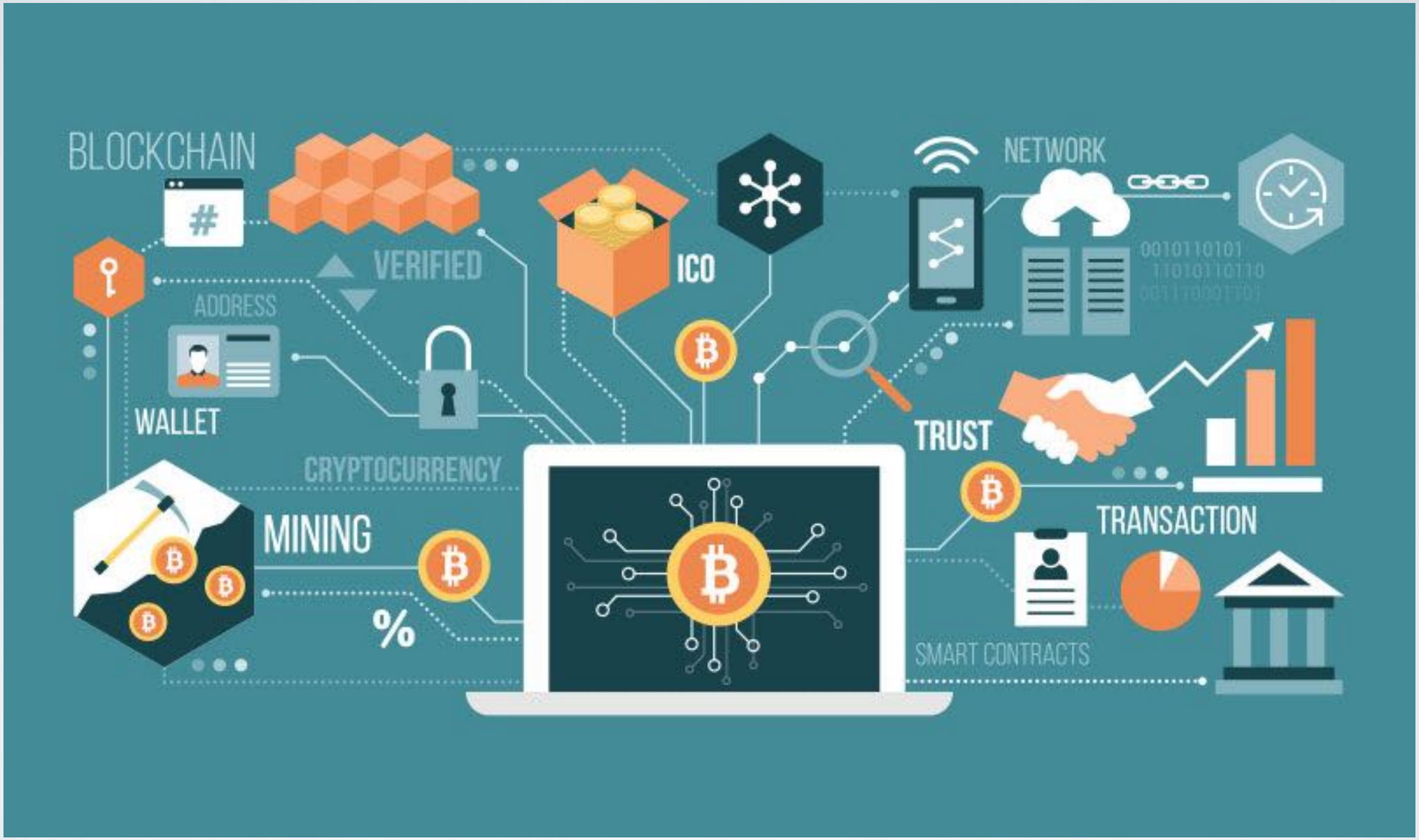
News

- RTFN!
- CoinDesk, CoinTelegraph, etc.
- Twitter: maybe @VitalikButerin, @pwuille, @IOHK_Charles, @iam_preethi, @officialmcafee (for entertainment), @ethereumJoseph, @starkness, @adam3us, etc.
- Lots of currency-specific forum sites, Discords, etc.

Any other questions?

- Come up and ask after class, or send an email to **abhishek@cs.jhu.edu** :)

Towards blockchains



Source: https://www.intheblack.com/articles/2018/03/22/blockchain-future-record-keeping

Cryptocurrencies Aren't 'Crypto'

As the price of Bitcoin and Ethereum skyrocket, and more and more people who are unfamiliar with technology join in the craze, words start to lose their original and correct meaning.


$20,000.00
$15,000.00
$10,000.00
$5,000.00
$0
Price (USD)
May '17
Sep '17
Jan '18
May '18
Sep '18
Jan '19

- **Whether you love it or hate it…**
  - Cryptocurrencies are exerting a massive influence on our field
  - Most people's first major exposure to cryptography
- That's both a good thing and a bad thing
  - The good: we get to deploy some really exciting new cryptography
  - The bad: if you stare into the abyss…

Before blockchains: 1980s-2007 (or: *how we got PayPal*)


Harrah's Council Bluffs

1 Harrah's Blvd. Council Bluffs, IA 51501

CASHOUT TICKET

VALIDATION 51-3601-7245-2982-3145

07/31/2010 18:13:16 TICKET # 7680

2580 DOLLARS AND SEVENTEEN CENTS

$0.17

Ticket Void after 90 days ASSET# 1724

1980s: Retail Payments

- **Goal: Digital payment system that**
  - Allows payments between customers and merchants (c2m)
  - Or between individual customers (c2c)
- **Strong cryptographic security**
- **Privacy**

1980s: Retail Payments

- **Some of the earliest ideas:**
  - Let's make digital cash!
  - Let's make digital checks!

“Digital cash”

- **Some of the earliest ideas:**
  - What’s the problem with the idea of (offline) digital cash?

"Digital cash"

- **Some of the earliest ideas:**
  - What's the problem with the idea of (offline) digital cash?
  - Idea: hardware tokens?


SAFECHIP BY BOURGOGNE ET GRASSET
US5895321 PROGRESSIVE FR2739587

- Son
  - W
  - Ide

Experts Videos

$1.5M Robbery of Bellagio Casino Foiled Thanks to RFID Chips

By **Aaron Saenz** - Feb 12, 2011 51,689

If you're thinking of robbing a Las Vegas casino, and you're not George Clooney, I have a word of advice: give up now. As Anthony Carleo recently found out, even if you leave the casino in one piece, the chips you stole are going to be worthless long before you make your get away. The 29 year old suspect is accused of robbing the Bellagio on December 14th of 2010, stealing chips whose face value totaled around $1.5 million dollars. Their real value, however, was zero. Thanks to RFID tags embedded inside them, the chips with denominations of $100 to $25,000 could be immediately deactivated rendering them unredeemable for cash value. Watch CCTV footage from the December 14th robbery in the video clip below, followed by the recent press conference from the Las Vegas Police concerning Carleo's arrest. Stealing worthless chips and then getting caught trying to sell them to undercover officers? Danny Ocean this guy is not.

“Digital cash”

- **Some of the earliest ideas:**
  - BankAmericard (?) ATMs: record debit balance on magstripe
    - Problem?
  - Use offline smartcards to store balances

Source: https://www.elprocus.com/working-of-smart-card/

"Digital checks"

- **Some of the earliest ideas:**
  - Ideas to use smart cards with digital signatures to write IOUs to merchants, who could later redeem them
  - Problem: **double spending**
    - Can spend my whole bank balance at fifty different merchants
    - When they each go to claim the funds, I'm long gone

Summary of problems

- **Double spending**
  - To capture double spending you need an online (networked) party that must be trusted
  - They can attack the system or simply fail
- **Privacy**
  - In many naive systems, the bank sees every transaction you make
- **Origin**
  - How is new currency created?

Centralized electronic $

- Use a centralized bank database ("ledger") to record account balances
  - Require merchants/ATMs to contact the bank for approval
  - Ledger can be "account-based" or "transaction-based"
- Typically it's both, and the two are reconciled

Private e-Cash

- Devised by Chaum, Chaum/Fiat/Naor, Brands, etc.
  - Move to a "cash" model, with added privacy
  - Individuals would carry redeemable tokens
  - Reduces the problem to detecting double spending and user privacy

Chaum (CRYPTO '83)

Payer

σ

(Blind) signature

sk

Bank

pk

Merchant

Redeem/
verify not previously
spent

CHL (Eurocrypt '05)

Payer

σK

(Blind) signature

sk

Bank

pk

Merchant

$SN = PRF(K, i)$

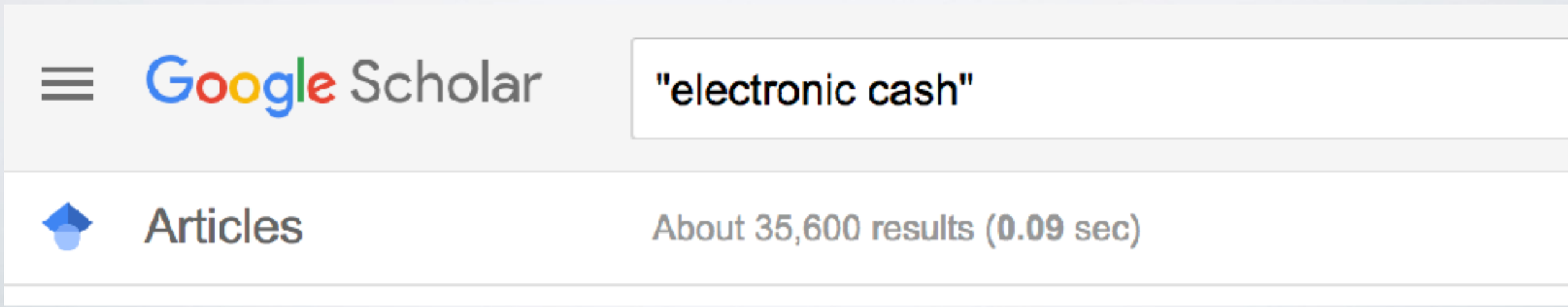
$NIZK\ \Pi$

For I = 1 to N

Redeem/
verify not previously
spent

e-Cash

- Huge number of academic works / practical improvements
  - Online schemes / offline schemes
    - (Offline required using tamper-resistant storage)
    - Main research problem continued to be privacy



Why did centralized e-Cash fail?

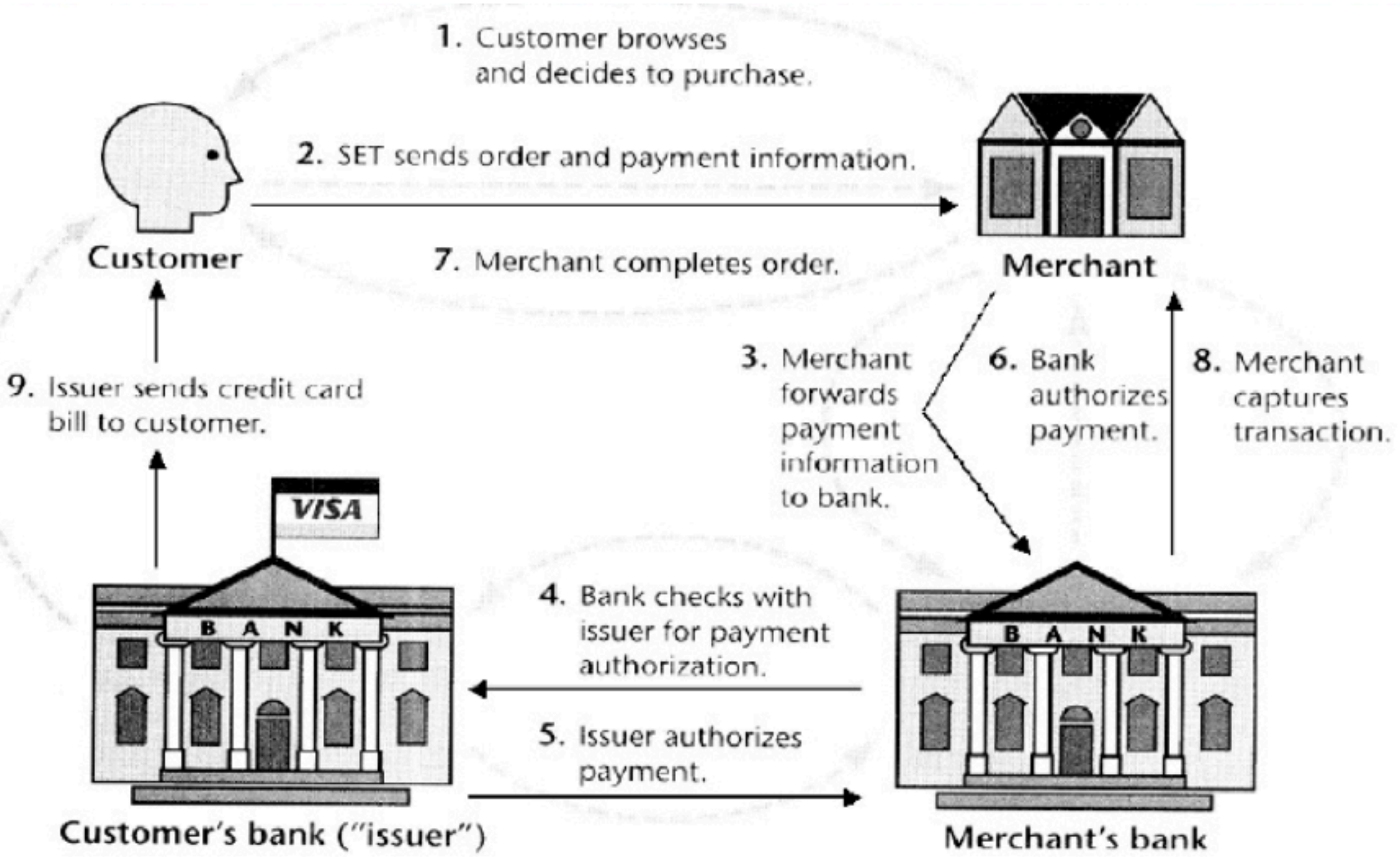
- Deploying e-Cash systems required a centralized bank
  - Required a trusted server with money issuing powers
  - In 1994, EU regulations made this more challenging
- 9/11 and beyond saw closures of *non-anonymous* currencies (e-Gold and Liberty Reserve)

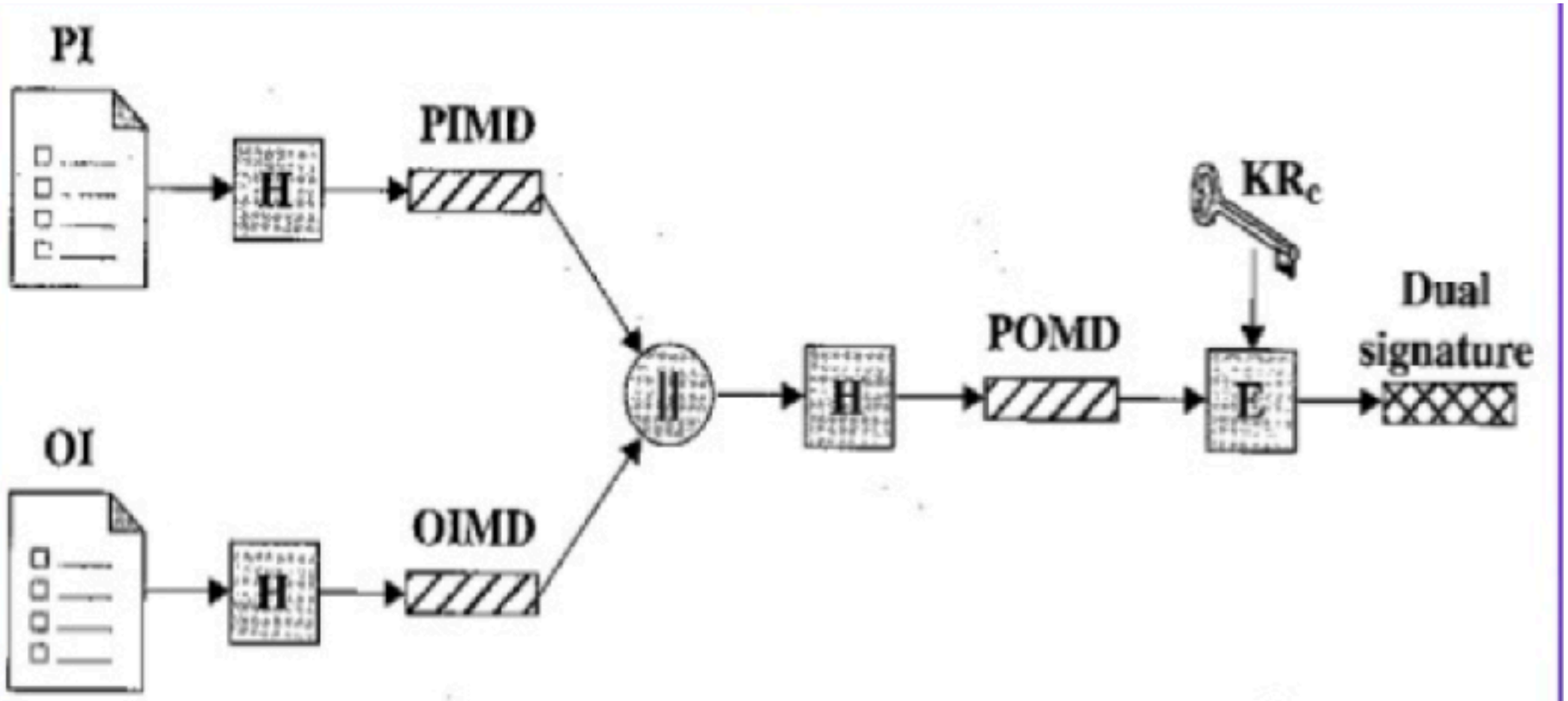
Why did e-Cash fail? (2)

- Were these technical or policy failures? Maybe both.
- The e-Cash model was centralized and relied on a vulnerable interface with the banking system
- Privacy was (eventually) off the table for regulators
- Any solution would have to work around those (manufactured) technical problems

1996: SET

- Developed by Visa and MasterCard
  - Cryptographic architecture based on certificates
  - Assurance, authenticity and confidentiality


1. Customer browses and decides to purchase.
2. SET sends order and payment information.
Customer
7. Merchant completes order.
Merchant
3. Merchant forwards payment information to bank.
6. Bank authorizes payment.
8. Merchant captures transaction.
9. Issuer sends credit card bill to customer.
VISA
BANK
4. Bank checks with issuer for payment authorization.
5. Issuer authorizes payment.
BANK
Customer's bank ("issuer")
Merchant's bank

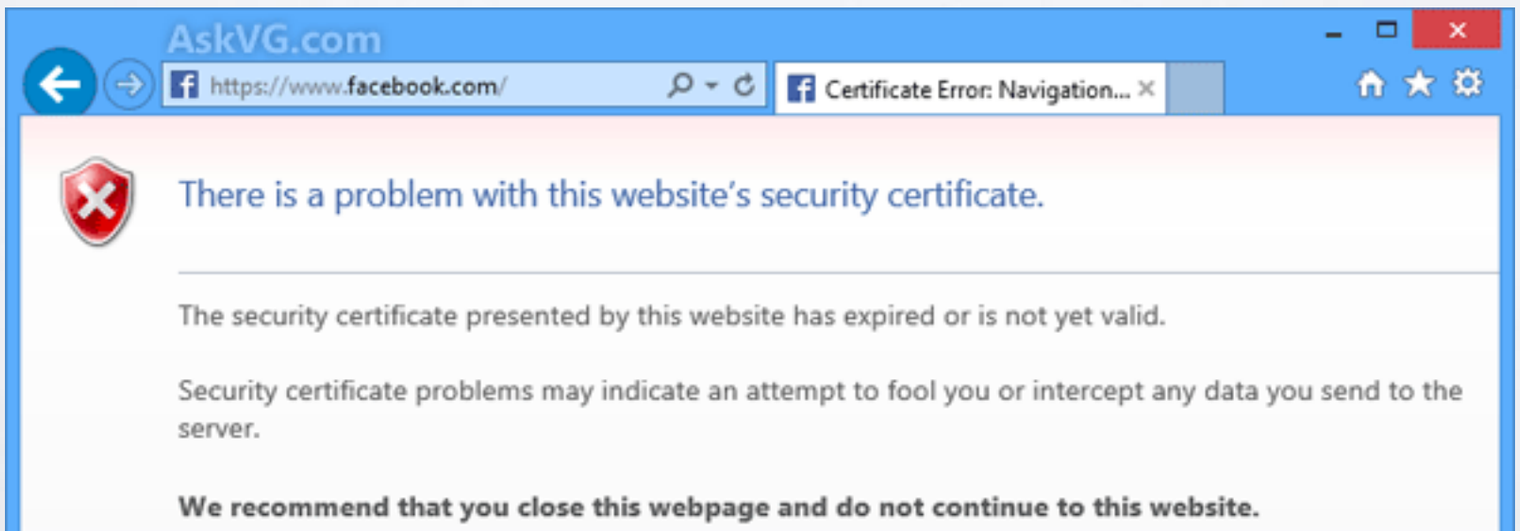


PI = Payment information
OI = Order information
H = Hash function (SHA-1)
|| = Concatenation

PIMD = PI message digest
OIMD = OI message digest
POMD = Payment order message digest
E = Encryption (RSA)
KR_c = Customer's private signature key

Why SET failed

- Required end-user certificates
- All the problems of key management PLUS all of the problems of identity verification
- Binding keys to user identities seems to trouble users



Conclusions (1980s-2007)

- Most cryptographic solutions too complex, or had "undesirable" features (privacy)
- Commercial solutions (existing credit cards, SET) failed to support the case of person->person transfers
- Web browsers didn't support fancy crypto anyway.
- **We got PayPal**

- M
  "u
- C
  su
- W
  an
- **W**

You can no longer use PayPal

At PayPal, we value a safer community in which our customers can do business. Some of your recent transactions violated our User Agreement and Acceptable Use Policy.

Any bank account or card linked to your PayPal account cannot be removed or used to create a new account. You can still log in and see your account information but you can't send or receive payments. Any money in your balance will be held for 180 days, at which point we'll email you instructions about withdrawing your money.

Reference # PP-005-921-770-133

Continue

Conclusions (1980s-2007)

- Mo ons were too complex, or had unc acy)
- Co sting credit cards, SET) failed to sup n->person transfers
- We port fancy crypto any
- **W**

Conclusions (1980s-2007)

- Mo on
  unc acy
- Co sti to
  sup n-
- We oo
  any
- **W**

The decentralized era
2008-2019

Nakamoto, 2008

- Replace the server with a **distributed** ledger (blockchain)
- Use a new consensus technique to construct the ledger

Nakamoto, 2008

- Replace the server with a distributed ledger (blockchain)
- Use a new consensus technique to construct the ledger
- Use puzzles to handle consensus & generate funds [Credit to Dai, (B-Cash) Back (HashCash) etc.]

Nakamoto, 2008

- Replace the server with a distributed ledger (blockchain)
- Use a new consensus technique to construct the ledger
- Use puzzles to handle consensus & generate funds
- Eliminate the need for explicit key/identity bindings

Nakamoto, 2008

- Replace the server with a distributed ledger (blockchain)
- Use a new consensus technique to construct the ledger
- Use puzzles to handle consensus & generate funds
- Eliminate the need for explicit key/identity bindings
- Everything else is straightforward crypto and excellent engineering

Lessons of Bitcoin

- Getting the consensus algorithm right makes all the difference

Lessons of Bitcoin

> [B]lockchain-style consensus indeed achieves certain robustness properties in the presence of sporadic participation and node churn that none of the classical style protocols can attain.

- Pass, Shi 2018 (also '16, '17, Daian, Pass, Shi '16)

Lessons of Bitcoin

- Using the right consensus algorithm really makes a difference
- **Eliminating the need for key/identity management significantly simplifies the currency problem**

Lessons of Bitcoin

- Using the right consensus algorithm really makes a difference
- Eliminating the need for key/identity management significantly simplifies the currency problem
- **Human beings are weird**

Lessons of Bitcoin

- U… …nce
- P…
  s…
- H…

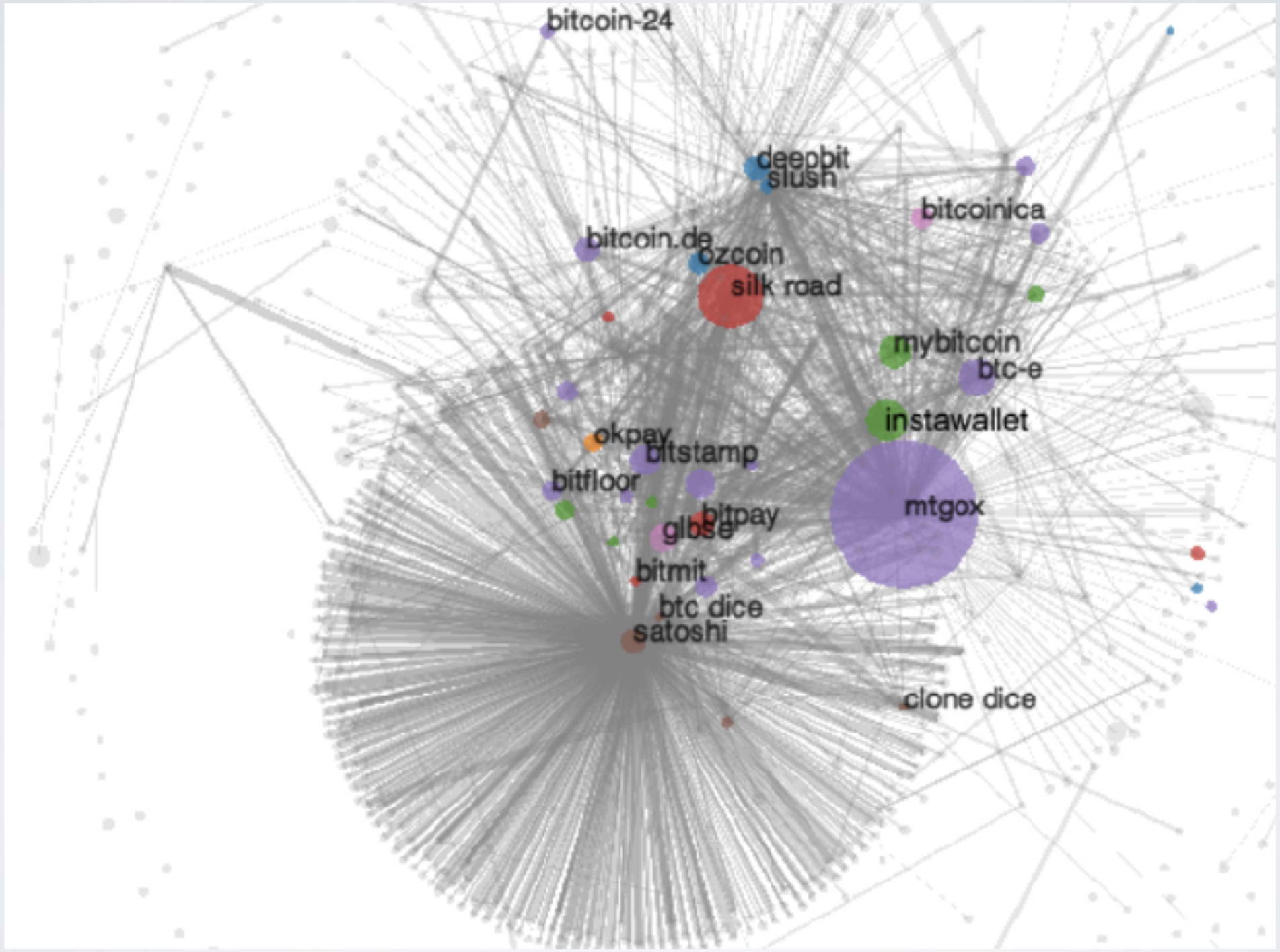
This is simultaneously trivial and the most unexpected lesson of the entire cryptocurrency experiment:

People will assign significant value to **meaningless electronic tokens** — *if* you convince them that the tokens are **secure** and have a **predictable supply.**

Limitations of Bitcoin

- Privacy limitations
- Functionality limitations
- Scalability & Sustainability limitations

Bitcoin & Privacy



Source: MPJLMVS13

(TS//NF) Met with SSG11 and S2F on the MR access. The following topics were discussed:

- Checking to see if the DTG/Port/IP Address could be assessed to validate if it hits against the BITCOIN Targets
- Checking to see if the partner does any user validation
- The relationship between BITCOIN targets and the MONKEYROCKET data
- Additional data that is not found in XKS-central, but can be made available to the customer
    - The following files were sent to the customer for analysis:
        - Mac_address.csv
        - Password_hash_history.csv
        - Provider user full.csv
        - User_sessions full.csv

As of right now, MONKEYROCKET is offering a sole source for SIGDEV for the BITCOIN Targets. We requested feedback and any mission highlights they have or will have. (SNM)

Zerocoin/Zcash

WARNING

THIS IS DEVELOPMENT SOFTWARE. WE DON'T CERTIFY IT FOR PRODUCTION USE. WE ARE RELEASING THIS DEV VERSION FOR THE COMMUNITY TO EXAMINE, TEST AND (PROBABLY) BREAK. IF YOU SEE SOMETHING, SAY SOMETHING! IN THE COMING WEEKS WE WILL LIKELY MAKE CHANGES TO THE WIRE PROTOCOL THAT COULD BREAK CLIENT COMPATIBILITY. SEE HOW TO CONTRIBUTE FOR A LIST OF WAYS YOU CAN HELP US.

WARNING WARNING

NO, SERIOUSLY. THE ABOVE WARNING IS NOT JUST BOILERPLATE. THIS REALLY IS DEVELOPMENT CODE AND WE'RE STILL ACTIVELY LOOKING FOR THE THINGS WE'VE INEVITABLY DONE WRONG. PLEASE DON'T BE SURPRISED IF YOU FIND OUT WE MISSED SOMETHING FUNDAMENTAL. WE WILL BE TESTING AND IMPROVING IT OVER THE COMING WEEKS.

WARNING WARNING WARNING

WE'RE NOT JOKING. DON'T MAKE US PULL AN ADAM LANGLEY AND TAKE AWAY THE MAKEFILE.

From payments to state

- Of course once you have a ledger…
  - Each Bitcoin transaction can be considered a function f() consuming some previous state and producing a state update
  - Obviously this generalizes nicely to more complex programs and stored data

From payments to state

```
1  contract MetaCoin {
2    mapping (address => uint) balances;
3
4    function MetaCoin() {
5      balances[tx.origin] = 10000;
6    }
7
8    function sendCoin(address receiver, uint amount) returns(bool sufficient) {
9      if (balances[msg.sender] < amount) return false;
10     balances[msg.sender] -= amount;
11     balances[receiver] += amount;
12     return true;
13   }
14
15   function getBalance(address addr) returns(uint) {
16     return balances[addr];
17   }
18 }
19
```

The future: 2018-

What we'll also talk about

- Scaling, including payment channels
- Replacing PoW
- Other advanced applications (not related to currency)

Scaling

- Current Bitcoin/Ethereum transaction rate is ~7TX/s
- Compare with Visa at 10,000-40,000+ TX.s globally
- This gets worse as transaction complexity increases
- Problems are storage/throughput/validation bandwidth

L2 (Channels)

1

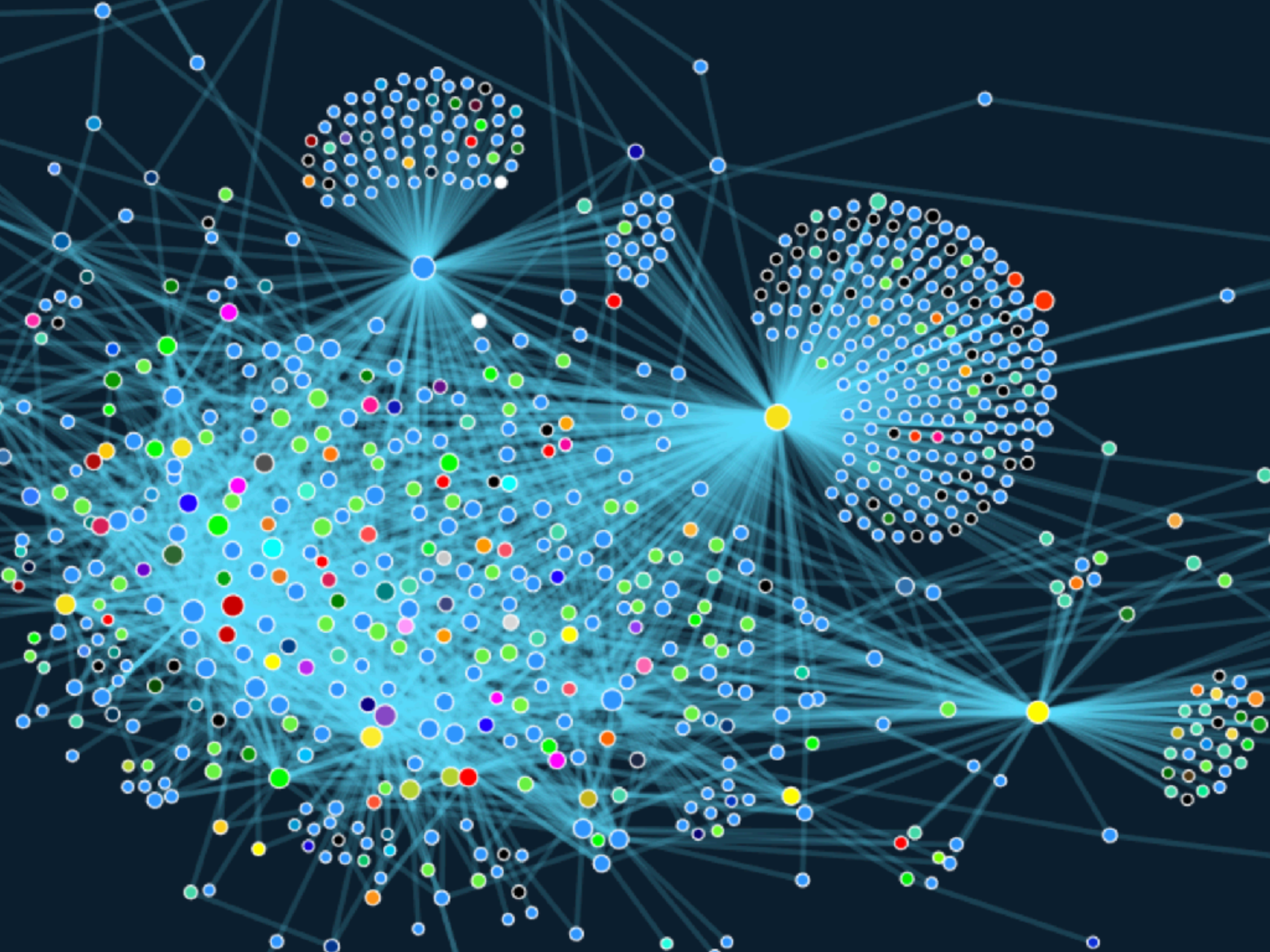
Open

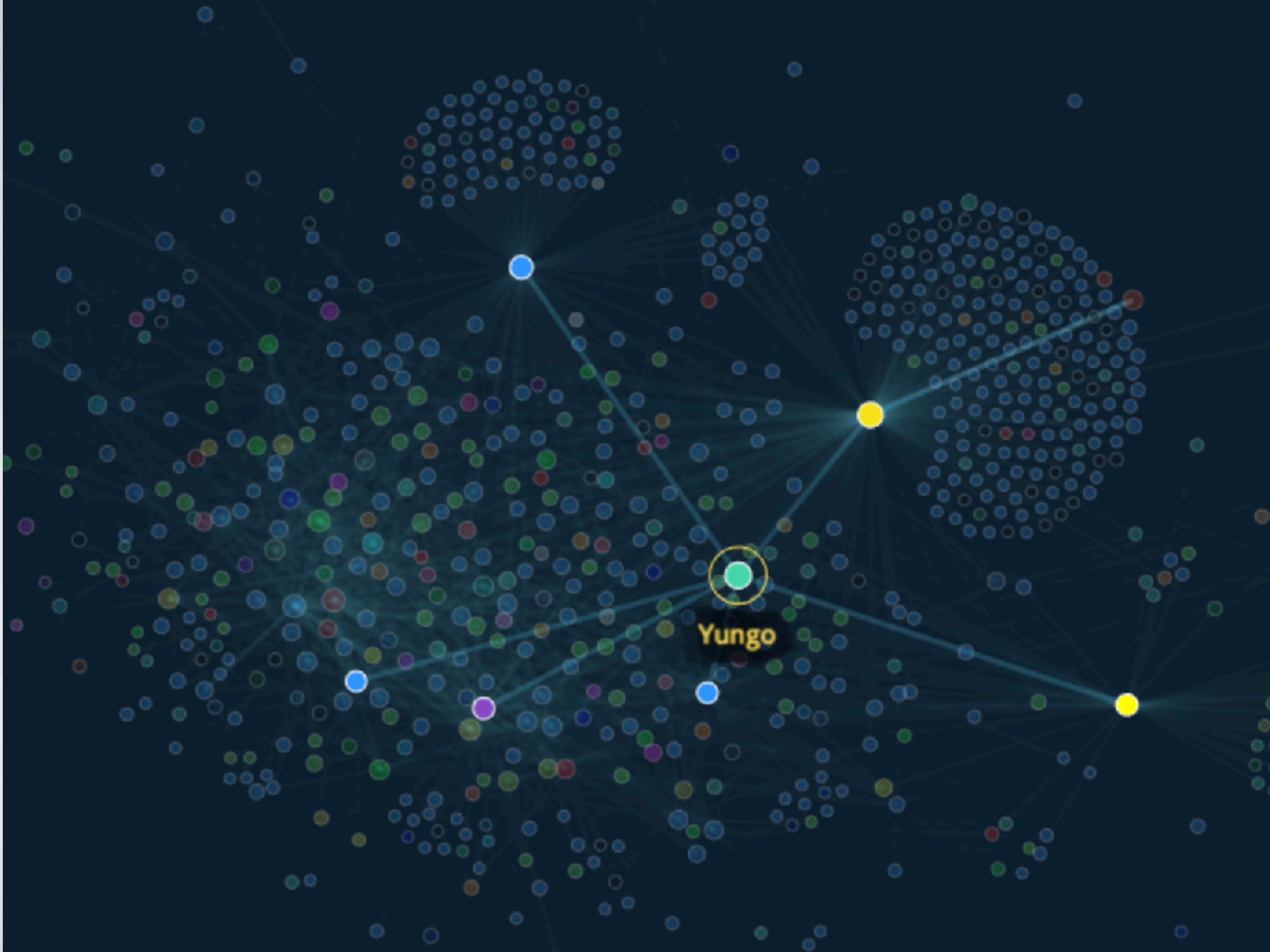
0

0.9 Update 0.1

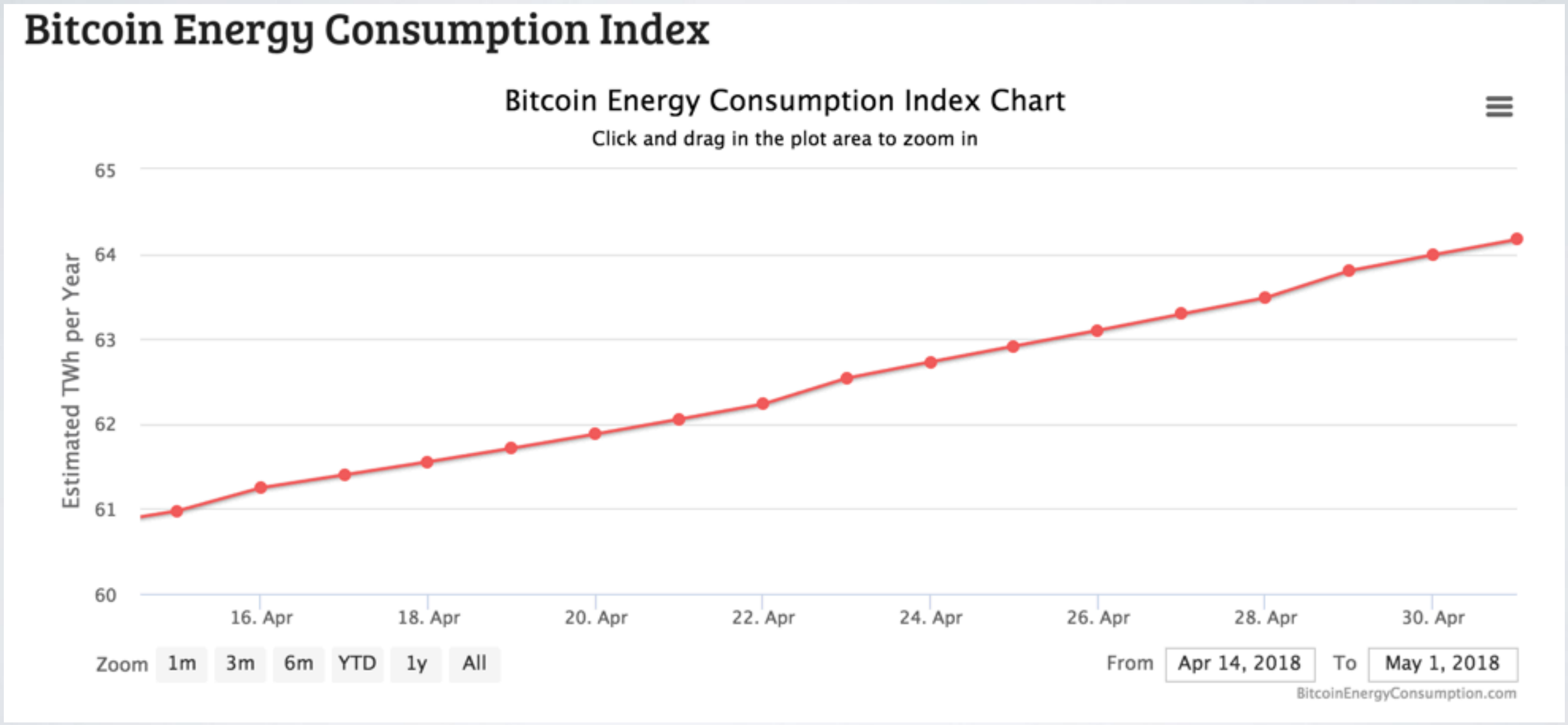
0.8 Update 0.2

… Close result on blockchain …




Yungo

Replacing PoW



Proof of Stake

- Current PoW design is obviously unsustainable
- Most common solution (in permissionless) chains is Proof of Stake”
- Rough summary: enumerate all stakeholders of the coin, scaled by their stake — and then sample one to construct the next block

Moving forward

- Weds: Abhishek gives crypto background
- Next week: consensus networks, Bitcoin
- A1 out Weds afternoon
- Do the reading!