Web3 - Outliers Cohort

Awesome ZK !!

Manmit Singh

Duke University and DMA Labs

2022

Contents

- Conceptual overview
- College-level math lesson
- Hands-on practice exercises
- Links to hand-picked resources
- Curated list of online communities

Slides - https://github.com/**SupremeSingh/Awesome-ZK**

Zero Knowledge in Culture

vitalik.eth 
@VitalikButerin

Replying to @tarunchitra

I expect ZK-SNARKs to be a significant revolution as they permeate the mainstream world over the next 10-20 years.

8:40 PM · Sep 1, 2021 · Twitter

ZK-Rollups likely to be main Layer 2 solution for Ethereum, says Vitalik Buterin

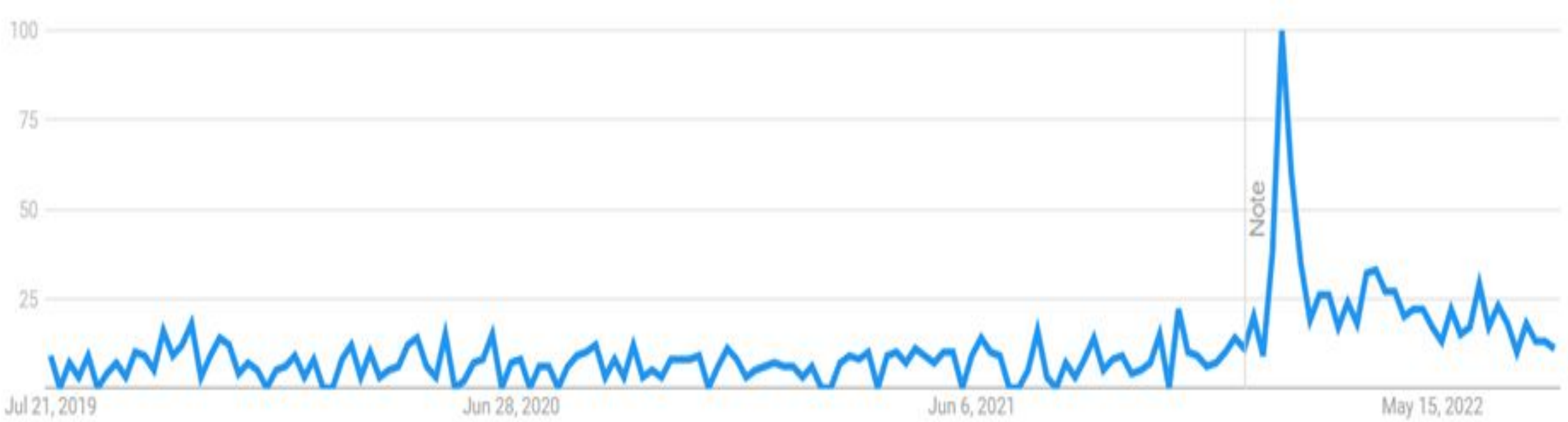
August 8, 2022, 10:45AM

Eli | STARK Maxi ✨🐺
@EliBenSasson

I *ALMOST* agree with this, with 2 reservations:
1) STARKs, not SNARKs, will dominate
2) 3-5 years to permeate mainstream

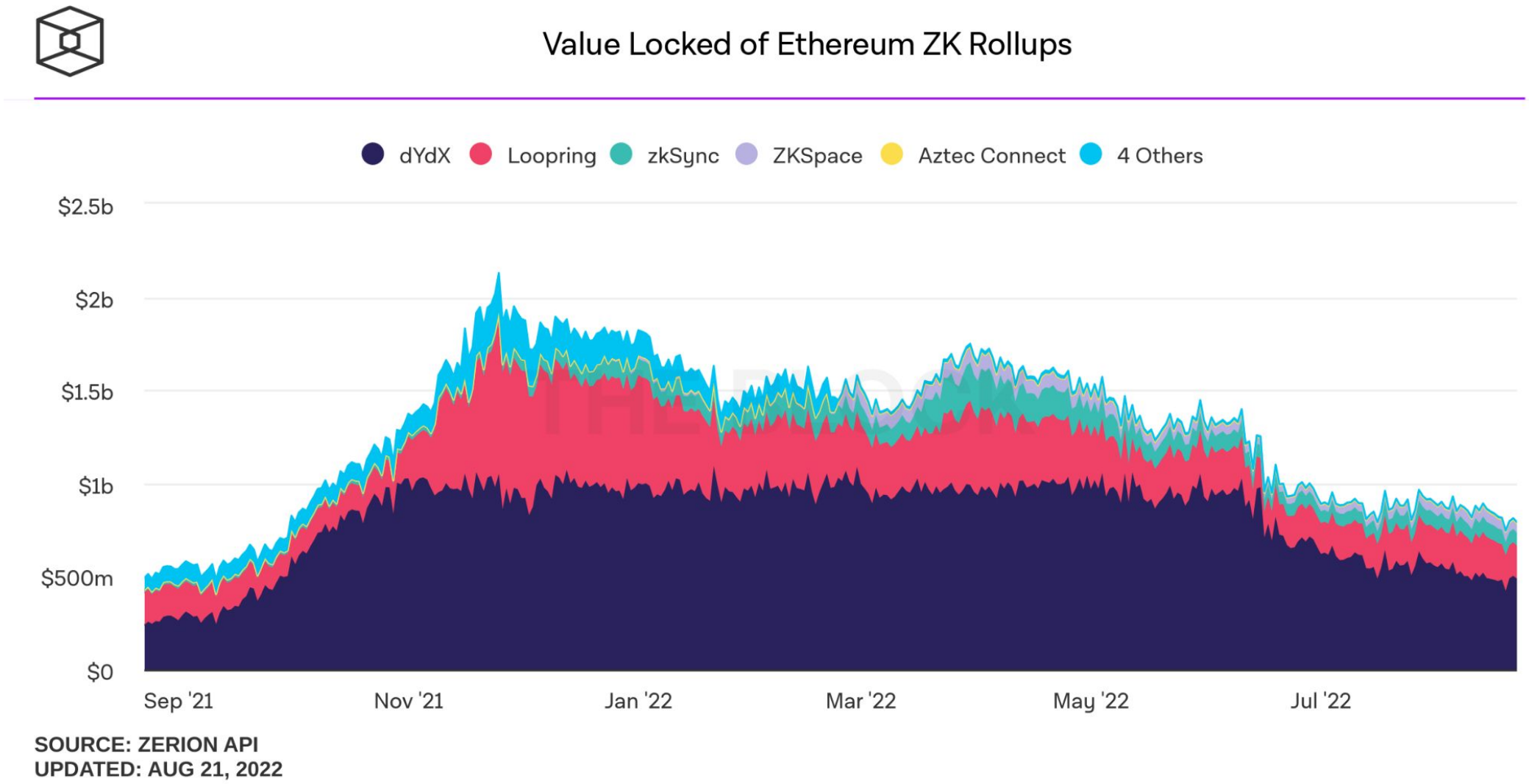
Put a reminder in my calendar to check this in 4 years.

ZK Proof Interest



https://trends.google.com/trends/explore?date=today%205-y&q=%2Fm%2F029z84

ZK Total Value Locked



https://www.theblock.co/data/scaling-solutions/scaling-overview/value-locked-of-ethereum-zk-rollups

Context

Why talk about this ?

Might be the next step in web3 evolution

- Any society needs trust to function
- The Web3 solution - If you can, don’t trust but **verify**
- From BTC -> ETH -> DeFi and NFTs, this seems to work well

But, it’s still not perfect …

Context

What do ZKs bring to the table ?

Web3's biggest bottlenecks are -

- All the data needs to be public for it to work
- Interdependent computations need to be run every time

So we lack scale, and privacy - which are **needed to get the next 1B + users.** And ZKPs are built to solve exactly these problems.

https://vitalik.ca/general/2017/11/09/starks_part_1.html

Definition

What is a ZKP ?

ZKP = Zero Knowledge Proof

i.e A **cryptographic** tool to prove an **honest** computation **without revealing** inputs to the verifier

- Complete - If you are honest, you can always convince someone
- Sound - If you are lying, you cannot convince anyone you are honest
- Non Revealing - Convince or don’t, you never have to reveal your secrets

Introduced in 1985 by Goldwasser et al., popularised after 2011

Already a major player in web3, **hundreds of projects, billions of USD in TVL**

Definition

Where are ZKPs today ?

Primary use case is **Layer 2 Optimisations**, identity, voting coming soon …

Proofs have split into various “families”, based on specific properties -

- SNARKs - Succinct, Non-Interactive Arguments of Knowledge
- STARKs - Scaleable, Transparent Arguments of Knowledge
- BulletProofs - A lightweight middle-ground between STARKs and SNARKs

SNARK adopters* - ZCash, Mina, Aztec, Scroll and many more
STARK adopters - Invented by Ben Sasson et al. and spun off into StarkEx

* The divide isn’t binary, many products use a mix of all of them

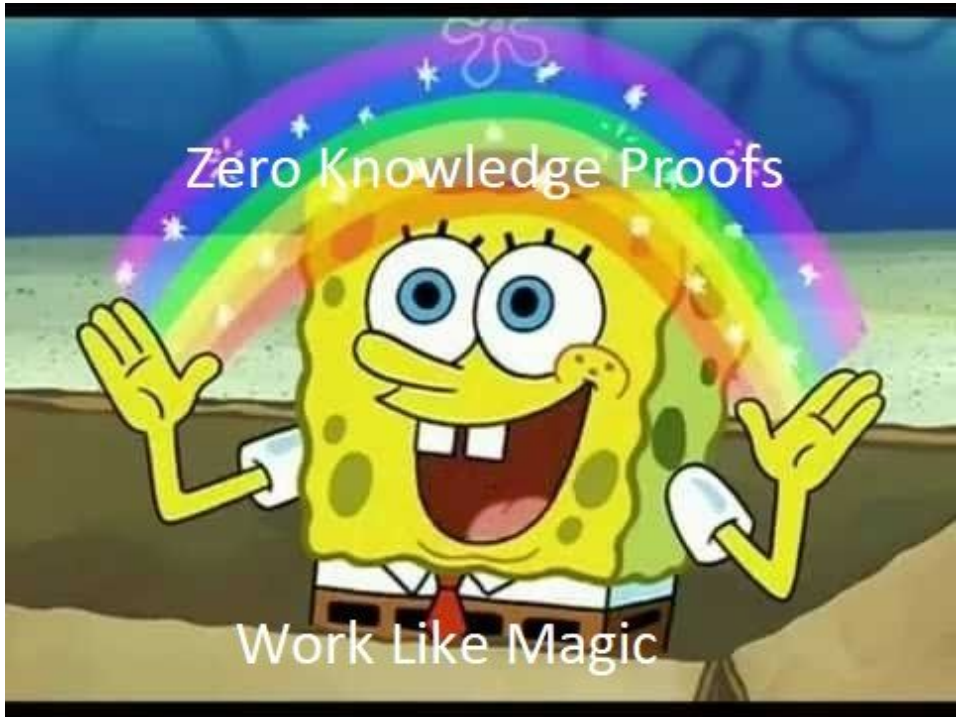
Recap …

ZK Proofs are scary, complicated and in early stages of development (Sure)

They require a deep understanding of math (100 %)

Mere mortals cannot understand this math (Kinda)

No point in trying / building with ZK just yet (No)



My Approach

An introductory ZK course modelled after an MBA class I TA'd at Duke.

Took professional training for ZK Proofs and Cairo programming.

EXTROPY.IO

Solved coding challenges, built mini-projects, aggregated industry information

Importantly, this is all **open-source** and completely open to **feedback** and improvement.

A *succinct* On-Ramp

A hands-on ZK on-ramp for noobs, built by a noob

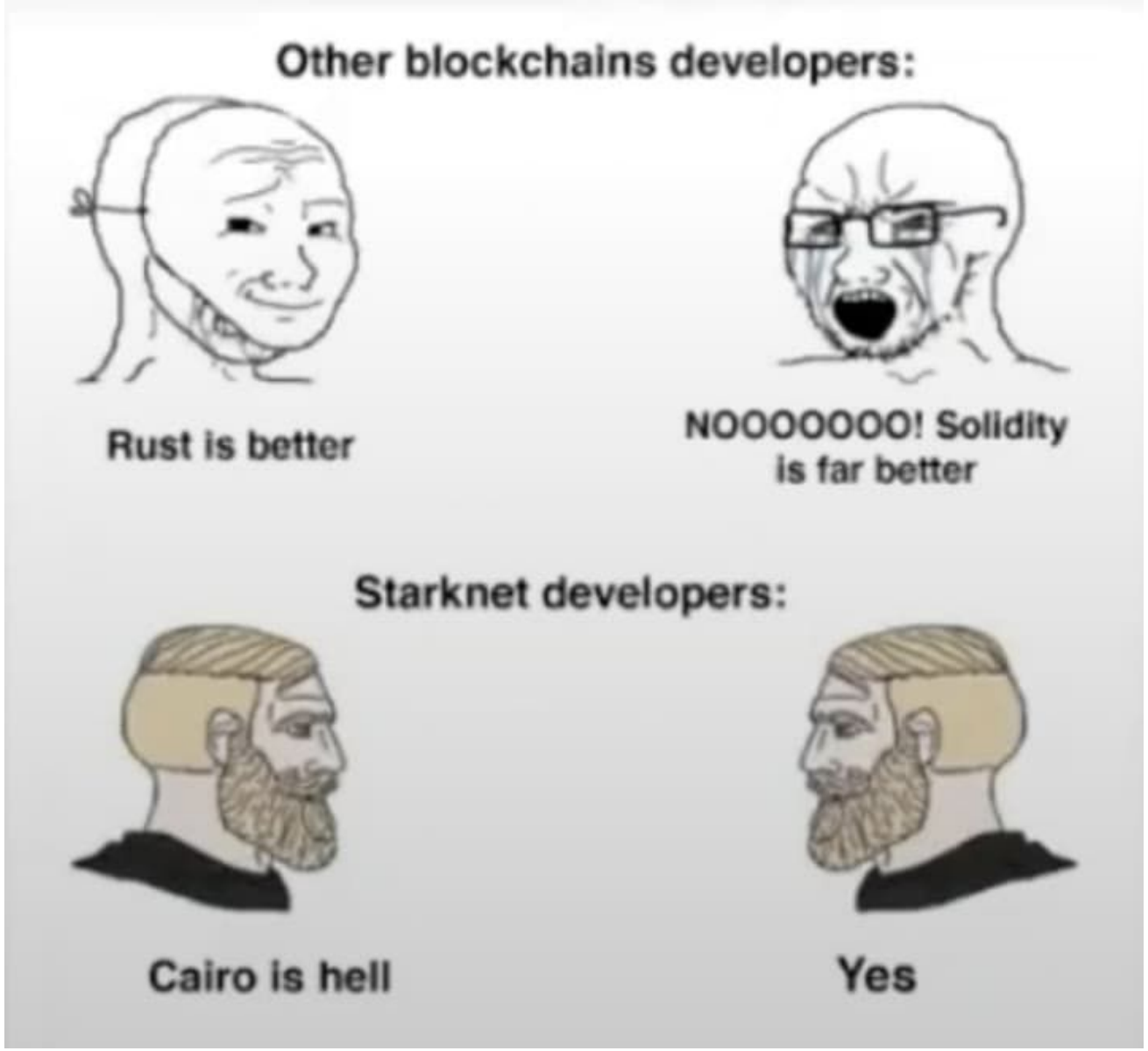
Ideal involvement - Self Paced, 4-6 weeks with about 10 hours a week.

Contains

- Short conceptual tutorials
- Case studies - ZCash and StarkNet
- An industry analysis and product layout as of now
- Solutions to Cairo Playground and Starklings Exercise
- Math - Polynomials, Fields, Encryption and Complexity Theory

But most importantly, this is all **open-source** and completely open to **feedback** and improvement.

But ...


Other blockchains developers:
Rust is better
NOOOOOOO! Solidity is far better
Starknet developers:
Cairo is hell
Yes

Notable Challenges

L2 programming today is like building an airplane while flying it

Focussed on the **Cairo**, **Protostar** and **Starknet.js** stack to streamline dependencies

Problems

- Cairo is changing quick, to become user friendly - really quick
- Documentation is fragmented, and beginner-friendly code hard to come by
- SNARK Tech is easier to build with, but often requires a change of paradigm, Eg. Mina

OpenZeppelin has done some fantastic work here, creating a **code wizard**, **standards** and conducting **security audits** of Cairo.

Future Roadmap

A few quick additions to make the content even more interactive

Based on feedback so far -

- A discord channel for group study
- A full-stack StarkNet Dapp tutorial
- More videos, explanations and interviews
- An active editorial community

Outlook - A New Web3 Primitive

Primitive: A digital “building-block” used to represent things

ZKP is **not** the name of a product. It is a branch of cryptography.

The Primitive =

ZKP-based standards and implementations that users can point to for every Web3 application

Examples -

- Shielded Identity and Reputation
- Shielded Transactions
- Shielded yet Secure Computations

https://www.aleo.org/post/zero-knowledge-primitives-by-aleo

Outlook - ZK Based Backends

Backend: Part of system responsible for storing and manipulating data

Moving from (sometimes) insecure **cloud** computing to **cloaked** computing

Cloud infrastructure has become the standard for enterprise-grade IT today - but it is still a largely centralized paradigm. Today …

- Users send private data to a centralized server for verification
- Companies store and process proprietary data and business logic on the cloud

Solutions -

- Secure and private attestation of identity on unsecured networks
- Cloaked computations over sensitive data on privately owned servers

https://twitter.com/ZK_Daily/status/1560990540157849604

Outlook - Multi Chain Future

Eg. Cheaper Smart Contracting

Let’s deploy and mint an NFT on an L2

Under the hood, we are

- Performing a computation
- Turning computation into a proof “object”
- Submitting the proof to a Layer 1, like Ethereum

Upon submission, the Layer 1 verifies and accepts the computation if it is legitimate

So, we get the security of Ethereum, and the speed + cost of an L2

https://vitalik.ca/general/2017/11/09/starks_part_1.html