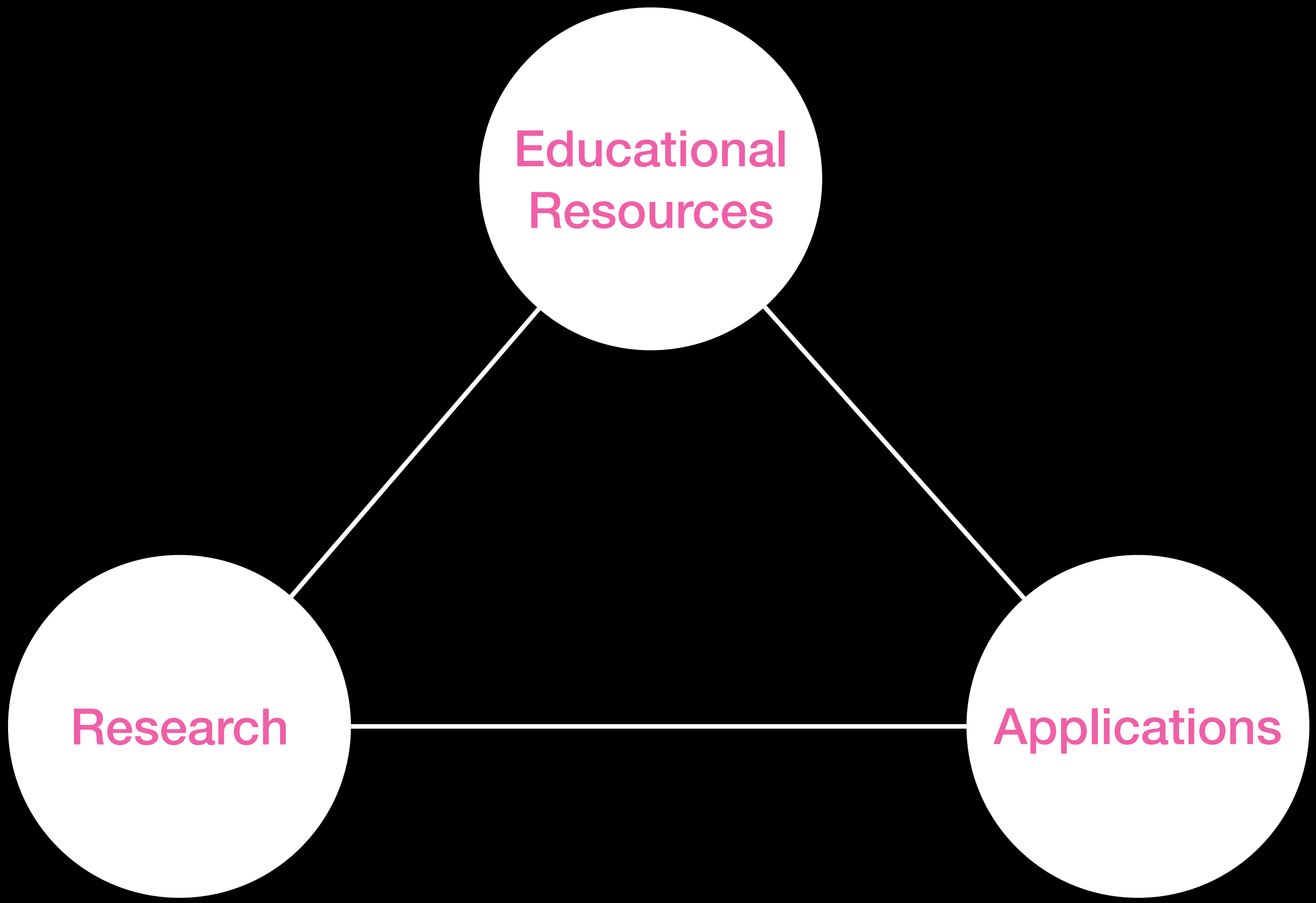
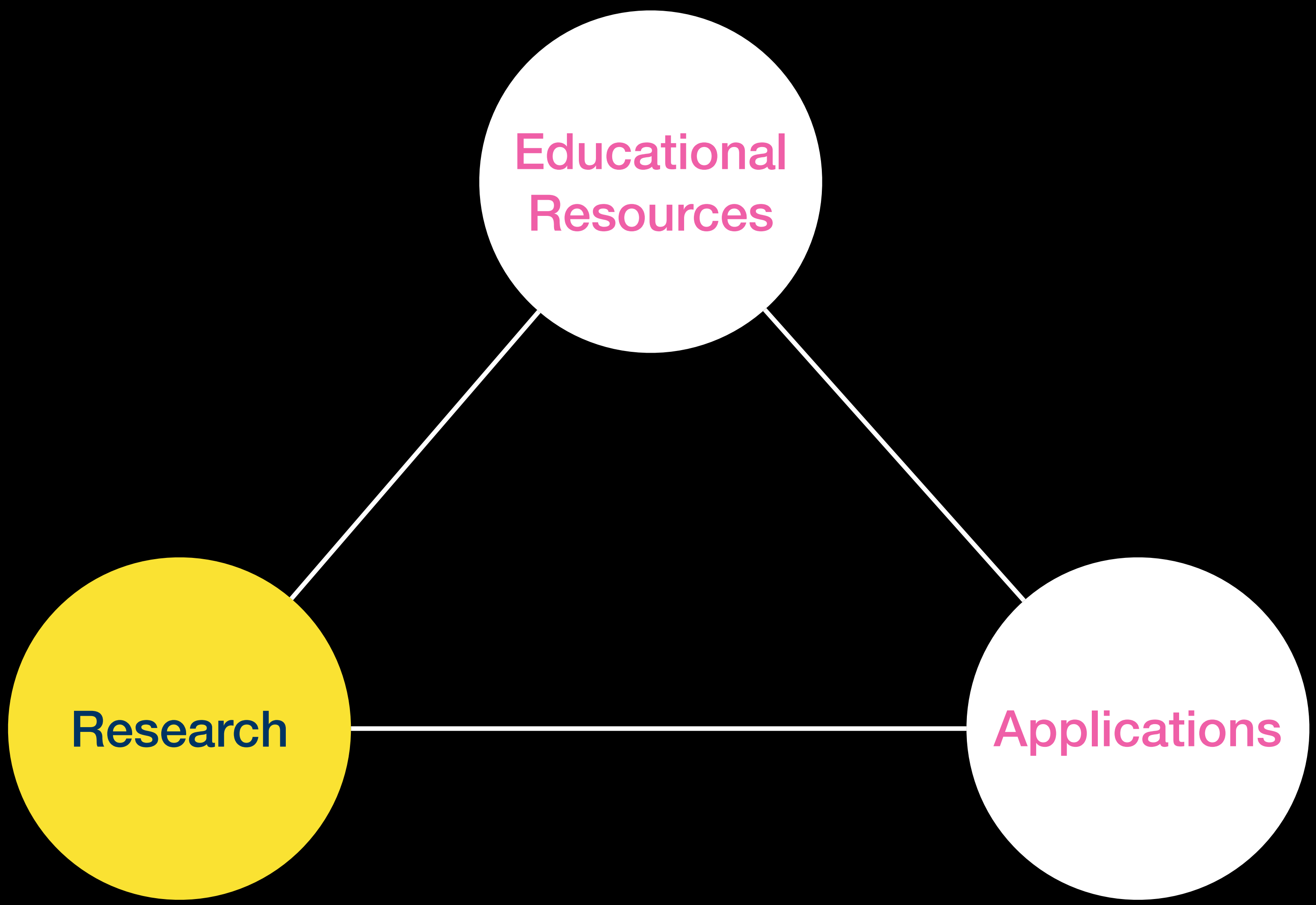
The Roaring Twenties: Recent Advances in Zero-Knowledge Proofs

Mary Maller
Ethereum Foundation


Educational Resources
Research
Applications


Educational Resources
Research
Applications

Research

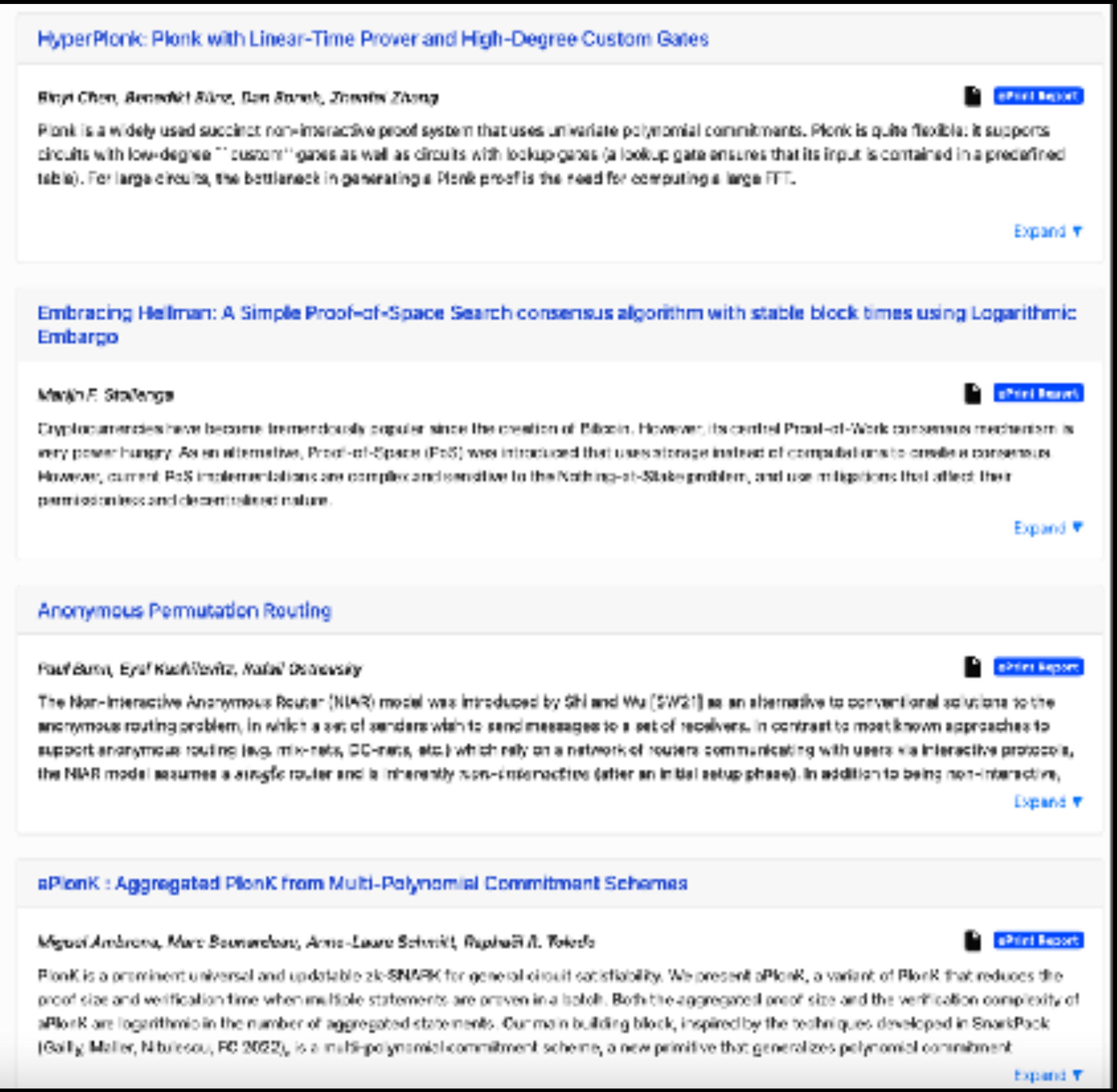
Where is research published?

International Association for Cryptologic Research

“The International Association for Cryptological Research (IACR) is a non-profit scientific organisation whose purpose is to further research in cryptology and related fields”

Research

Where is research published?



Hyperplonk: like Plonk, but without ffts

aPlonk: aggregate Plonk proofs with SNARKpack

Source of all research: *https://eprint.iacr.org/*
Be the first to know: *https://www.iacr.org/news*

Research

Where is research published?

Source of most bibtexs: *https://dblp.org/*

Research

Where is research published?

BibTeX record conf/stoc/Kilian92

> Home > conf/stoc/Kilian92

download as .bib file

```
@inproceedings{DBLP:conf/stoc/Kilian92,
  author    = {Joe Kilian},
  editor    = {S. Rao Kosaraju and
               Mike Fellows and
               Avi Wigderson and
               John A. Ellis},
  title     = {A Note on Efficient Zero-Knowledge Proofs and Arguments (Extended
               Abstract)},
  booktitle = {Proceedings of the 24th Annual {ACM} Symposium on Theory of Computing,
               May 4-6, 1992, Victoria, British Columbia, Canada},
  pages     = {723--732},
  publisher = {{ACM}},
  year      = {1992},
  url       = {https://doi.org/10.1145/129712.129782},
  doi       = {10.1145/129712.129782},
  timestamp = {Tue, 06 Nov 2018 11:07:06 +0100},
  biburl    = {https://dblp.org/rec/conf/stoc/Kilian92.bib},
  bibsource = {dblp computer science bibliography, https://dblp.org}
}
```

Source of most bibtexs: *https://dblp.org/*

Research

Where is research peer reviewed?

Theory

- Crypto
- Eurocrypt
- TCC
- Asiacrypt

Applied

- ACM CCS
- Real World Crypto
- S&P/ Oakland
- Usenix

Research

ZK research focus of the 20's

- In this presentation I say zk loosely.
- Sometimes I mean zero-knowledge.
- Sometimes I mean succinct proof

Research

ZK research focus of the 20's

- Lookup Arguments
- Linear Time Provers
- Vector Commitments
- Recursive Arguments
- PQ Signatures
- Forking Lemmas

Research

Recursive Arguments

- A proof of a proof of a proof… that a computation is correct.
- Very good for long, ongoing computations such as a *verifiable delay function* (more on this later).

Research

Recursive Arguments

Nova: Recursive Zero-Knowledge Arguments from Folding Schemes

Abhiram Kothapalli[†] Srinath Setty[*] Ioanna Tzialla[‡]

[†]Carnegie Mellon University [*]Microsoft Research [‡]New York University

- Recursive Argument: a proof of a proof of a proof… that a computation is correct.
- Recursion overhead = 2 group scalar multiplications
- Prover work = 2 multiexponentiations of size O(|F|)

Research

Recursive Arguments

If $(A\vec{x}) \cdot (B\vec{x}) = 0$ and $(A\vec{y}) \cdot (B\vec{y}) = 0$

then $(A(\vec{x} + \gamma\vec{y})) \cdot (B(\vec{x} + \gamma\vec{y})) = 0 + \gamma^2[\_]$

- Nova improves on BCLMS21, which improves on Halo.
- No FFTs, no PCPs, just a DLOG commitment scheme.

Proof-Carrying Data without Succinct Arguments

Benedikt Bünz
benedikt@cs.stanford.edu
Stanford University

Alessandro Chiesa
alexch@berkeley.edu
UC Berkeley

William Lin
will.lin@berkeley.edu
UC Berkeley

Pratyush Mishra
pratyush@berkeley.edu
UC Berkeley

Nicholas Spooner
nspooner@bu.edu
Boston University

December 1, 2021

Recursive Proof Composition without a Trusted Setup

Sean Bowe[1], Jack Grigg[1], and Daira Hopwood[1]

[1] Electric Coin Company
{sean,jack,daira}@electriccoin.co
https://electriccoin.co/

Research

Recursive Arguments

If $\boxed{C_A} \cdot \boxed{C_B} = 0$ and $\boxed{C'_A} \cdot \boxed{C'_B} = 0$

then $\boxed{C_A + \gamma C'_A} \cdot \boxed{C_B + \gamma C'_B} = 0 + \gamma^2[\_]$

- Nova improves on BCLMS21, which improves on Halo.
- No FFTs, no PCPs, just a DLOG commitment scheme.

Proof-Carrying Data without Succinct Arguments

Benedikt Bünz
benedikt@cs.stanford.edu
Stanford University

Alessandro Chiesa
alexch@berkeley.edu
UC Berkeley

William Lin
will.lin@berkeley.edu
UC Berkeley

Pratyush Mishra
pratyush@berkeley.edu
UC Berkeley

Nicholas Spooner
nspooner@bu.edu
Boston University

December 1, 2021

Recursive Proof Composition without a Trusted Setup

Sean Bowe[1], Jack Grigg[1], and Daira Hopwood[1]

[1] Electric Coin Company
{sean,jack,daira}@electriccoin.co
https://electriccoin.co/

Research

ZK research focus of the 20's

Lookup Arguments

Linear Time Provers

Vector Commitments

Recursive Arguments

PQ Signatures

Forking Lemmas

Research

Linear Time Provers

- General theme: proving time is the current bottleneck of many systems.
- Most SNARKs are *quasi*-linear time due to polynomial multiplication.
- The polynomial multiplication is hard to parallelise.
- Linear time provers are better.

Research

Linear Time Provers

HyperPlonk: Plonk with Linear-Time Prover and High-Degree Custom Gates

Binyi Chen, Espresso Systems
Benedikt Bünz, Stanford University, Espresso Systems
Dan Boneh, Stanford University
Zhenfei Zhang, Espresso Systems

IOP for Plonkish constraint systems

1) Supports custom gates
2) Operates over boolean hypercube

Improvement on Brakedown

Orion: Zero Knowledge Proof with Linear Prover Time

Tiancheng Xie[1], Yupeng Zhang[2], and Dawn Song[1]
[1] University of California, Berkeley
{tianc.x,dawnsong}@berkeley.edu
[2] Texas A&M University
zhangyp@tamu.edu

Error correcting codes (very efficient)

1) Uses code switching
2) log^2(N) proof size and verifier time.

Zero-Knowledge IOPs with Linear-Time Prover and Polylogarithmic-Time Verifier

Jonathan Bootle, jbt@zurich.ibm.com, IBM Research – Zurich
Alessandro Chiesa, alexch@berkeley.edu, UC Berkeley
Siqi Liu, sliu18@berkeley.edu, UC Berkeley

IOP for R1CS

Gets either:
1) Uses proof composition
2) log^2(N) proof size and verifier time.

Gemini: Elastic SNARKs for Diverse Environments

Jonathan Bootle, jbt@zurich.ibm.com, IBM Research
Alessandro Chiesa, alessandro.chiesa@epfl.ch, EPFL
Yuncong Hu, yuncong_hu@berkeley.edu, UC Berkeley
Michele Orrù, michele.orru@berkeley.edu, UC Berkeley

Memory costs also important.

Gets either:
1) quasilinear memory and linear prover
2) linear memory and quasilinear prover

Research

Linear Time Provers

HyperPlonk: Plonk with Linear-Time Prover and High-Degree Custom Gates

Binyi Chen, Espresso Systems
Benedikt Bünz, Stanford University, Espresso Systems
Dan Boneh, Stanford University
Zhenfei Zhang, Espresso Systems

IOP for Plonkish constraint systems

1) Supports custom gates
2) Operates over boolean hypercube

Improvement on Brakedown

Orion: Zero Knowledge Proof with Linear Prover Time

Tiancheng Xie[1], Yupeng Zhang[2], and Dawn Song[1]

[1] University of California, Berkeley
{tianc.x,dawnsong}@berkeley.edu
[2] Texas A&M University
zhangyp@tamu.edu

Error correcting codes (very efficient)

1) Uses code switching
2) log^2(N) proof size and verifier time.

Zero-Knowledge IOPs with Linear-Time Prover and Polylogarithmic-Time Verifier

Jonathan Bootle, jbt@zurich.ibm.com, IBM Research – Zurich
Alessandro Chiesa, alexch@berkeley.edu, UC Berkeley
Siqi Liu, sliu18@berkeley.edu, UC Berkeley

IOP for R1CS

Gets either:
1) Uses proof composition
2) log^2(N) proof size and verifier time.

Gemini: Elastic SNARKs for Diverse Environments

Jonathan Bootle, jbt@zurich.ibm.com, IBM Research
Alessandro Chiesa, alessandro.chiesa@epfl.ch, EPFL
Yuncong Hu, yuncong_hu@berkeley.edu, UC Berkeley
Michele Orrù, michele.orru@berkeley.edu, UC Berkeley

Memory costs also important.

Gets either:
1) quasilinear memory and linear prover
2) linear memory and quasilinear prover

Research

Linear Time Provers

Improvement on Brakedown

HyperPlonk: Plonk with Linear-Time Prover and High-Degree Custom Gates

Binyi Chen, Espresso Systems
Benedikt Bünz, Stanford University, Espresso Systems
Dan Boneh, Stanford University
Zhenfei Zhang, Espresso Systems

IOP for Plonkish constraint systems

1) Supports custom gates
2) Operates over boolean hypercube

Orion: Zero Knowledge Proof with Linear Prover Time

Tiancheng Xie[1], Yupeng Zhang[2], and Dawn Song[1]
[1] University of California, Berkeley
{tianc.x,dawnsong}@berkeley.edu
[2] Texas A&M University
zhangyp@tamu.edu

Error correcting codes (very efficient)

1) Uses code switching
2) log^2(N) proof size and verifier time.

Zero-Knowledge IOPs with Linear-Time Prover and Polylogarithmic-Time Verifier

Jonathan Bootle, jbt@zurich.ibm.com, IBM Research – Zurich
Alessandro Chiesa, alexch@berkeley.edu, UC Berkeley
Siqi Liu, sliu18@berkeley.edu, UC Berkeley

IOP for R1CS

Gets either:
1) Uses proof composition
2) log^2(N) proof size and verifier time.

Gemini: Elastic SNARKs for Diverse Environments

Jonathan Bootle, jbt@zurich.ibm.com, IBM Research
Alessandro Chiesa, alessandro.chiesa@epfl.ch, EPFL
Yuncong Hu, yuncong_hu@berkeley.edu, UC Berkeley
Michele Orrù, michele.orru@berkeley.edu, UC Berkeley

Memory costs also important.

Gets either:
1) quasilinear memory and linear prover
2) linear memory and quasilinear prover

Research

Linear Time Provers

HyperPlonk: Plonk with Linear-Time Prover and High-Degree Custom Gates

Binyi Chen
Espresso Systems

Benedikt Bünz
Stanford University, Espresso Systems

Dan Boneh
Stanford University

Zhenfei Zhang
Espresso Systems

IOP for Plonkish constraint systems

1) Supports custom gates
2) Operates over boolean hypercube

Improvement on Brakedown

Orion: Zero Knowledge Proof with Linear Prover Time

Tiancheng Xie[1], Yupeng Zhang[2], and Dawn Song[1]

[1] University of California, Berkeley
{tianc.x,dawnsong}@berkeley.edu
[2] Texas A&M University
zhangyp@tamu.edu

Error correcting codes (very efficient)

1) Uses code switching
2) log^2(N) proof size and verifier time.

Zero-Knowledge IOPs with Linear-Time Prover and Polylogarithmic-Time Verifier

Jonathan Bootle
jbt@zurich.ibm.com
IBM Research – Zurich

Alessandro Chiesa
alexch@berkeley.edu
UC Berkeley

Siqi Liu
sliu18@berkeley.edu
UC Berkeley

IOP for R1CS

Gets either:
1) Uses proof composition
2) log^2(N) proof size and verifier time.

Gemini: Elastic SNARKs for Diverse Environments

Jonathan Bootle
jbt@zurich.ibm.com
IBM Research

Alessandro Chiesa
alessandro.chiesa@epfl.ch
EPFL

Yuncong Hu
yuncong_hu@berkeley.edu
UC Berkeley

Michele Orrù
michele.orru@berkeley.edu
UC Berkeley

Memory costs also important.

Gets either:
1) quasilinear memory and linear prover
2) linear memory and quasilinear prover

Research

ZK research focus of the 20’s

Lookup Arguments

Linear Time Provers

Vector Commitments

Recursive Arguments

PQ Signatures

Forking Lemmas

Research

Forking Lemmas

- Theoretically: multi-round arguments and the Fiat-Shamir transform do not always mix well.

- E.g. the way in which we implement non-interactive Bulletproofs was not proven secure for a long time.

- Practically: implications were not understood.

Research

Forking Lemmas

- Largely good news.
- We've learnt many of the ways we use Fiat-Shamir are actually okay.

A Compressed Σ-Protocol Theory for Lattices

Thomas Attema[1,2,3,⋆], Ronald Cramer[1,2,⋆⋆], and Lisa Kohl[1,⋆⋆⋆]

[1] CWI, Cryptology Group, Amsterdam, The Netherlands
[2] Leiden University, Mathematical Institute, Leiden, The Netherlands
[3] TNO, Cyber Security and Robustness, The Hague, The Netherlands

Parallel Repetition of $(k_1, \ldots, k_\mu)$-Special-Sound Multi-Round Interactive Proofs

Thomas Attema[1,2,3,⋆] and Serge Fehr[1,2,⋆⋆]

[1] CWI, Cryptology Group, Amsterdam, The Netherlands
[2] Leiden University, Mathematical Institute, Leiden, The Netherlands
[3] TNO, Cyber Security and Robustness, The Hague, The Netherlands

Research

ZK research focus of the 20’s

- Lookup Arguments
- Linear Time Provers
- Vector Commitments
- Recursive Arguments
- PQ Signatures
- Forking Lemmas

Research

Lookup Arguments

- General theme: proving time is the current bottleneck of many systems.
- Arithmetising efficiently makes proving time considerably faster.
- Idea: check a result is in a table, rather than recompute.

Research

Lookup Arguments

$$C(X) = \lambda_1(X) + 2\lambda_2(X) + \ldots + n\lambda_n(X)$$

$$\phi(X) = a_1\lambda_1(X) + a_2\lambda_2(X) + \ldots + a_n\lambda_n(X)$$

- Prove that a_i is in 1, … , n.
- I.e. that for all w_i, phi(w_i) = C(w_j) for some w_j

Research

Lookup Arguments

- General theme: proving time is the current bottleneck of many systems.
- Arithmetising efficiently makes proving time considerably faster.
- Idea: check a result is in a table, rather than recompute.

The halo2 Book

PLONKish Arithmetization

The arithmetization used by Halo 2 comes from PLONK, or more precisely its extension UltraPLONK that supports custom gates and lookup arguments. We'll call it ***PLONKish***.

Halo2 book: *https://zcash.github.io/halo2/concepts/arithmetization.html*

Research

Lookup Arguments

plookup: A simplified polynomial protocol for lookup tables

Ariel Gabizon — Aztec
Zachary J. Williamson — Aztec

The halo2 Book

Lookup argument

Halo 2 uses the following lookup technique, which allows for lookups in arbitrary sets, and is arguably simpler than Plookup.

1) Good for small tables.
2) Work for Bulletproofs, KZG, and FRI instantiations.

Research

Lookup Arguments

Succinct Zero-Knowledge Batch Proofs for Set Accumulators*

Matteo Campanelli
Protocol Labs
San Francisco, USA
matteo@protocol.ai

Dario Fiore
IMDEA Software Institute
Madrid, Spain
dario.fiore@imdea.org

Semin Han
Hanyang University
Seoul, Korea
seminhan@hanyang.ac.kr

Jihye Kim
Kookmin University
Seoul, Korea
jihyek@kookmin.ac.kr

Dimitris Kolonelos
IMDEA Software Institute, Spain
Universidad Politecnica de Madrid
Madrid, Spain
dimitris.kolonelos@imdea.org

Hyunok Oh
Hanyang University
Seoul, Korea
hoh@hanyang.ac.kr

1) Good for large tables.
2) Work for Groups of Hidden Order.
3) No bound on the table size

Flookup: Fractional decomposition-based lookups in quasi-linear time independent of table size

Ariel Gabizon
Zeta Function Technologies

Dmitry Khovratovich
Ethereum Foundation

1) Good for large tables.
2) Works for KZG groups.
3) Restricted to set membership.

Research

Lookup Arguments

Succinct Zero-Knowledge Batch Proofs for Set Accumulators*

Matteo Campanelli
Protocol Labs
San Francisco, USA
matteo@protocol.ai

Dario Fiore
IMDEA Software Institute
Madrid, Spain
dario.fiore@imdea.org

Semin Han
Hanyang University
Seoul, Korea
seminhan@hanyang.ac.kr

Jihye Kim
Kookmin University
Seoul, Korea
jihyek@kookmin.ac.kr

Dimitris Kolonelos
IMDEA Software Institute, Spain
Universidad Politecnica de Madrid
Madrid, Spain
dimitris.kolonelos@imdea.org

Hyunok Oh
Hanyang University
Seoul, Korea
hoh@hanyang.ac.kr

1) Good for large tables.
2) Work for Groups of Hidden Order.
3) No bound on the table size

Flookup: Fractional decomposition-based lookups in quasi-linear time independent of table size

Ariel Gabizon
Zeta Function Technologies

Dmitry Khovratovich
Ethereum Foundation

1) Good for large tables.
2) Works for KZG groups.
3) Restricted to set membership.

Research

Lookup Arguments

Baloo: Nearly Optimal Lookup Arguments

Arantxa Zapico*, Ariel Gabizon[3], Dmitry Khovratovich[1], Mary Maller[1], and Carla Ràfols[2]

[1] Ethereum Foundation
[2] Universitat Pompeu Fabra
[3] Zeta Function Technologies
arantxa.zapico@upf.edu, ariel.gabizon@gmail.com, khovratovich@gmail.com, mary.maller@ethereum.org, carla.rafols@upf.edu

1) Good for large tables.
2) Work for KZG instantiations.

- We released Baloo last week.
- The prover time is quasilinear in the number of lookups, and independent of the table size.

Research

Lookup Arguments

$$a(X) = \sum_i a_i \lambda_i(X), \lambda_i(X) = \frac{\prod_{j \in I, j \neq i} (X - \omega_i)}{\prod_{j \in I, j \neq i} (\omega_j - \omega_i)}$$

- We can work with polynomials over unknown basis, defined by secret set that is not fft friendly.

Research

Lookup Arguments

- Baloo is follow up work to Caulk and Caulk+

Caulk: Lookup Arguments in Sublinear Time

Arantxa Zapico*[1], Vitalik Buterin[2], Dmitry Khovratovich[2], Mary Maller[2], Anca Nitulescu[3], and Mark Simkin[2]

[1] Universitat Pompeu Fabra†
[2] Ethereum Foundation‡
[3] Protocol Labs§

Caulk+: Table-independent lookup arguments

Jim Posen[1] and Assimakis A. Kattis[2]

[1] Ulvetanna jimpo AT ulvetanna.io
[2] New York University kattis AT cs.nyu.edu

Research

ZK research focus of the 20’s

Lookup Arguments

Linear Time Provers

Vector Commitments

Recursive Arguments

PQ Signatures

Forking Lemmas

Research

Post Quantum Signatures

New Call for Proposals: Digital Signature Algorithms with Short Signatures and Fast Verification

NIST also plans to issue a new Call for Proposals for public-key (quantum-resistant) digital signature algorithms by the end of summer 2022. NIST is primarily looking to diversify its signature portfolio, so signature schemes that are not based on structured lattices are of greatest interest. NIST would like submissions for signature schemes that have short signatures and fast verification.

NIST wants PQ digital signatures not based on lattices

Link: https://csrc.nist.gov/News/2022/pqc-candidates-to-be-standardized-and-round-4

Research

Post Quantum Signatures

New Call for Proposals: Digital Signature Algorithms with Short Signatures and Fast Verification

NIST also plans to issue a new Call for Proposals for public-key (quantum-resistant) digital signature algorithms by the end of summer 2022. NIST is primarily looking to diversify its signature portfolio, so signature schemes that are not based on structured lattices are of greatest interest. NIST would like submissions for signature schemes that have short signatures and fast verification.

NIST wants PQ digital signatures not based on lattices

Common approach is to prove knowledge of a pre-image of a hash using MPC-in-the-Head techniques.

Link: https://csrc.nist.gov/News/2022/pqc-candidates-to-be-standardized-and-round-4

Research

Post Quantum Signatures

New Call for Proposals: Digital Signature Algorithms with Short Signatures and Fast Verification

NIST also plans to issue a new Call for Proposals for public-key (quantum-resistant) digital signature algorithms by the end of summer 2022. NIST is primarily looking to diversify its signature portfolio, so signature schemes that are not based on structured lattices are of greatest interest. NIST would like submissions for signature schemes that have short signatures and fast verification.

NIST wants PQ digital signatures not based on lattices

Common approach is to prove knowledge of a pre-image of a hash using MPC-in-the-Head techniques.

Signatures are typically smaller than FRI-based approaches, due to small constants.

Research

Post Quantum Signatures

Syndrome Decoding in the Head: Shorter Signatures from Zero-Knowledge Proofs

Thibauld Feneuil[1,2], Antoine Joux[3], and Matthieu Rivain[1]

[1] CryptoExperts, Paris, France
[2] Sorbonne Université, CNRS, INRIA, Institut de Mathématiques de Jussieu-Paris Rive Gauche, Ouragan, Paris, France
[3] CISPA Helmholtz Center for Information Security, Saarbrücken, Germany

Hardness of syndrome decoding on linear codes.

Banquet: Short and Fast Signatures from AES*

Carsten Baum[1], Cyprien Delpech de Saint Guilhem[2], Daniel Kales[3], Emmanuela Orsini[2], Peter Scholl[1], and Greg Zaverucha[4]

[1] Dept. Computer Science, Aarhus University, Aarhus, Denmark.
[2] imec-COSIC, KU Leuven, Leuven, Belgium.
[3] Graz University of Technology, Graz, Austria.
[4] Microsoft Research

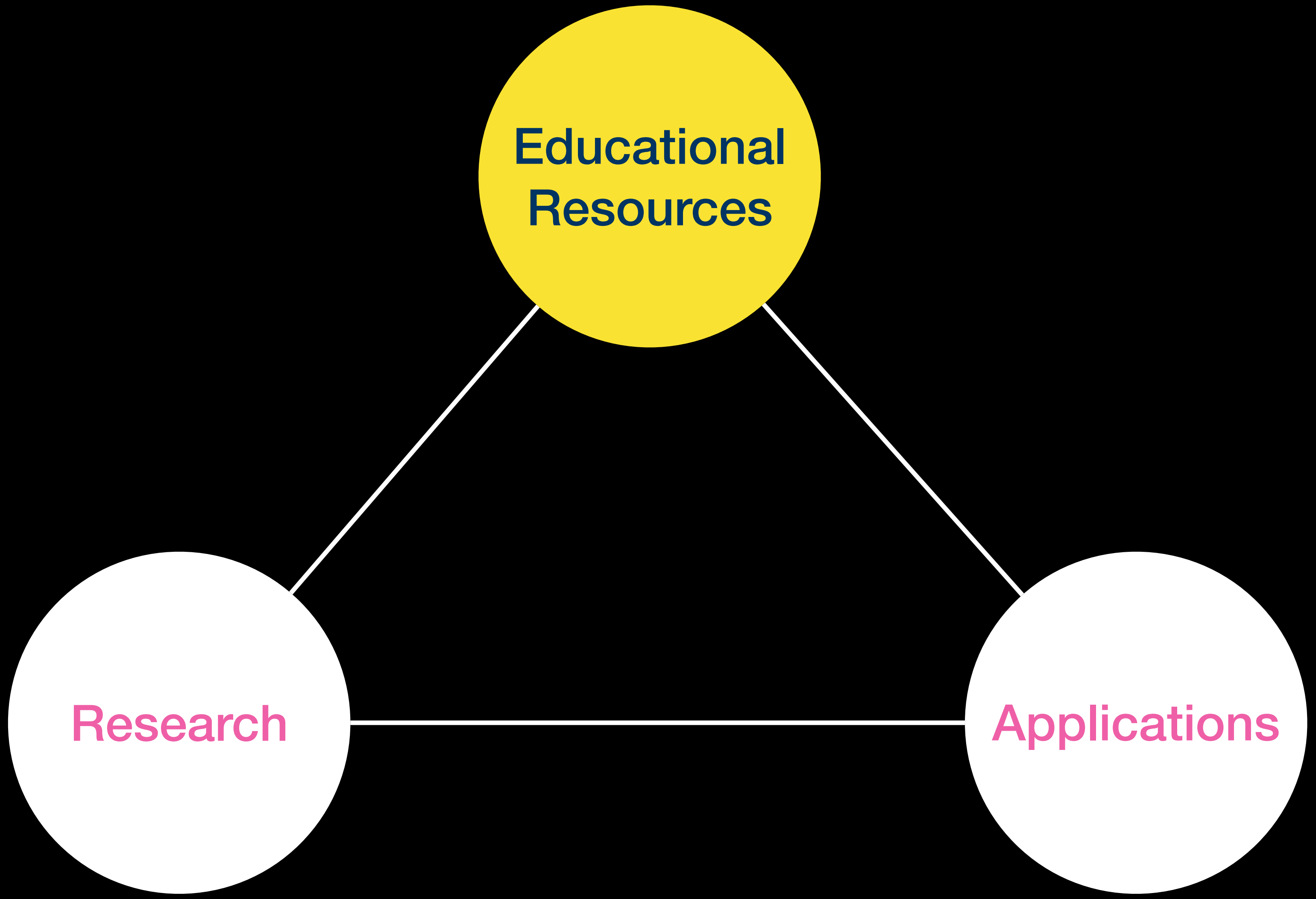
Hardness of inverting AES.

Efficient Lifting for Shorter Zero-Knowledge Proofs and Post-Quantum Signatures

Daniel Kales
Graz University of Technology
daniel.kales@iaik.tugraz.at

Greg Zaverucha
Microsoft Research
gregz@microsoft.com

Picnic4. Hardness of inverting LowMC.


Educational Resources
Research
Applications

Educational Resources

Section objectives

- In the recent past, resources for learning about zk-SNARKs were scarce.
- There are many great people and teams working on good teaching resources.
- Here I will advertise some of them.

Educational Resources

The Community Reference Document…

ZKProof Community Reference

Version 0.3

July 17, 2022

This document is a work in progress.

Feedback and contributions are welcome.

Find the latest version at https://zkproof.org.

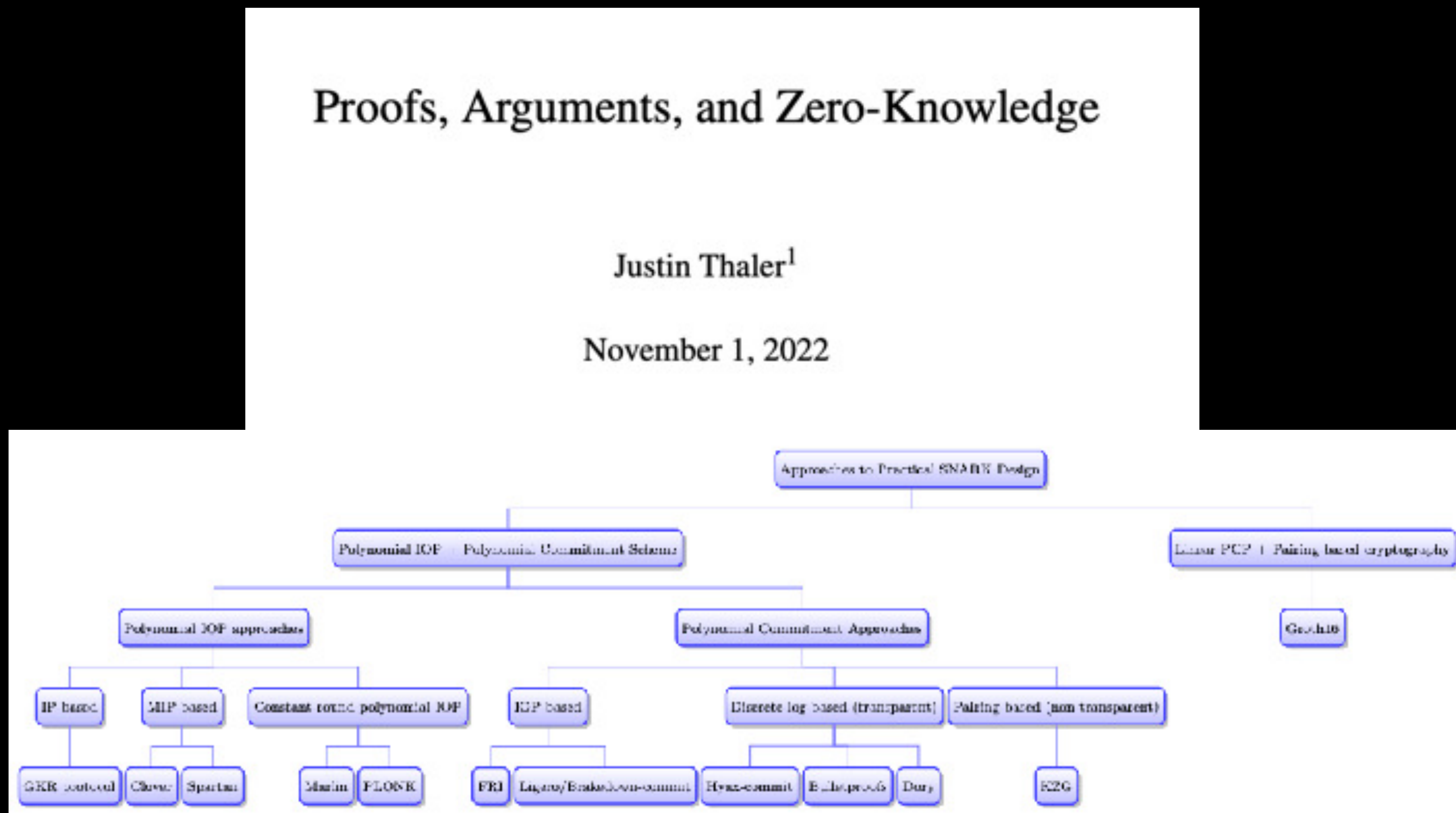
Send your comments to editors@zkproof.org.

ZKProof promotes the best practices for proper development and deployment of zero-knowledge proof (ZKP) systems, aligned with security and interoperability.

Very important link: *https://docs.zkproof.org/pages/reference/versions/ZkpComRef-0-3.pdf*

Educational Resources

Zero-Knowledge Textbooks



Book: *https://people.cs.georgetown.edu/jthaler/ProofsArgsAndZK.pdf*

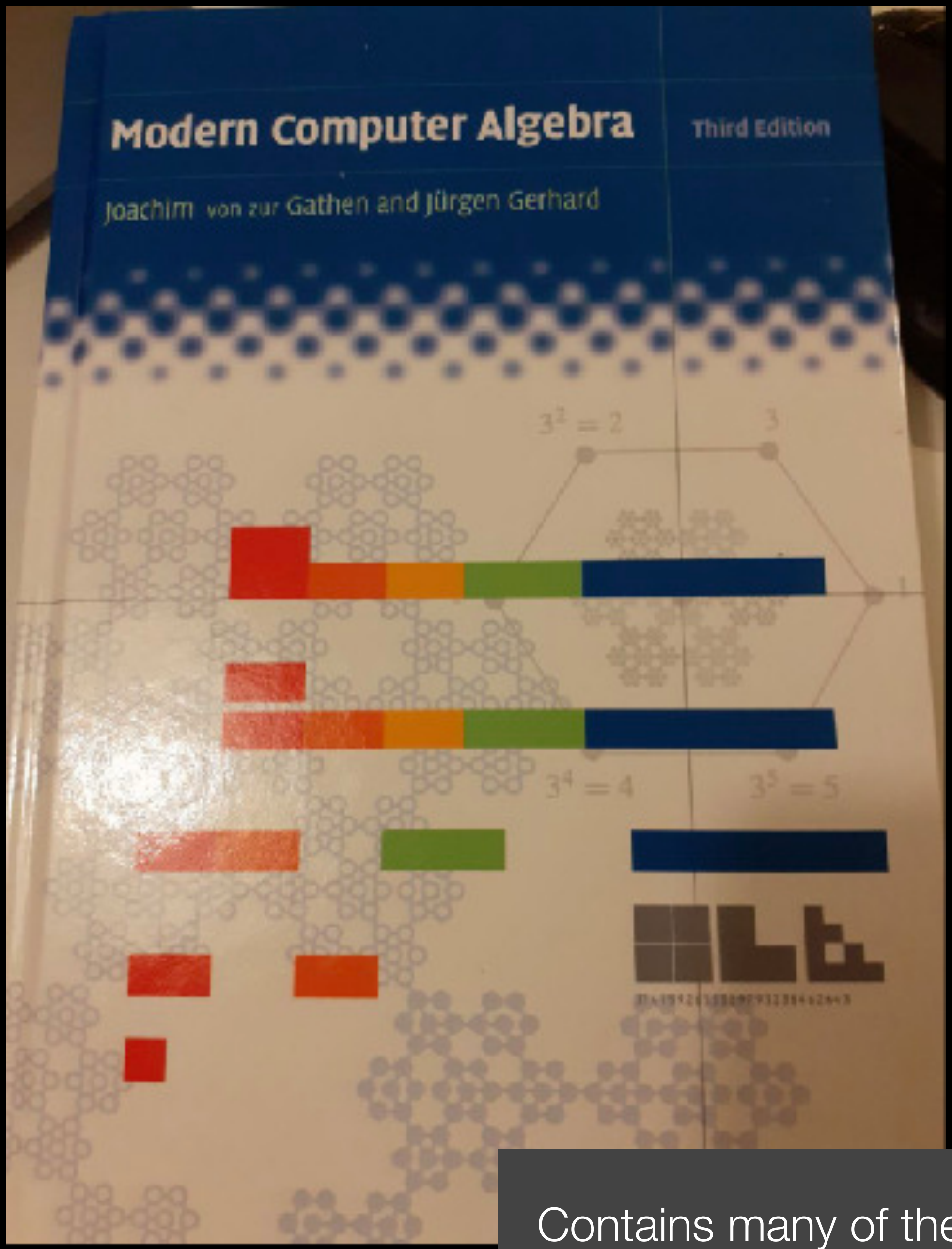
Educational Resources

Zero-Knowledge Textbooks

Book: *https://leastauthority.com/community-matters/moonmath-manual/*

Educational Resources

Zero-Knowledge Textbooks

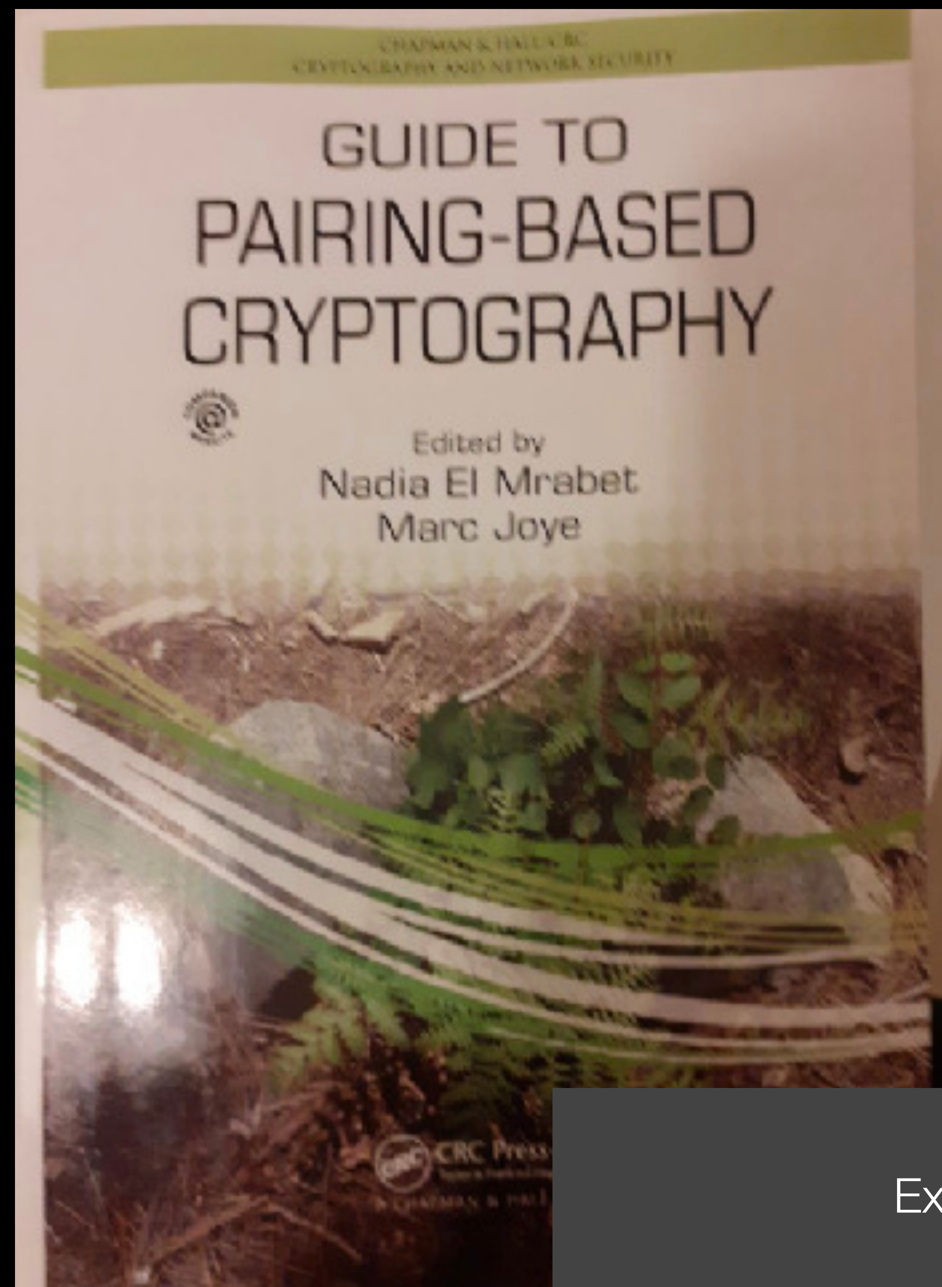


Contains many of the important algorithms for computing polynomial operations efficiently.

Book: Not free :(

Educational Resources

Zero-Knowledge Textbooks

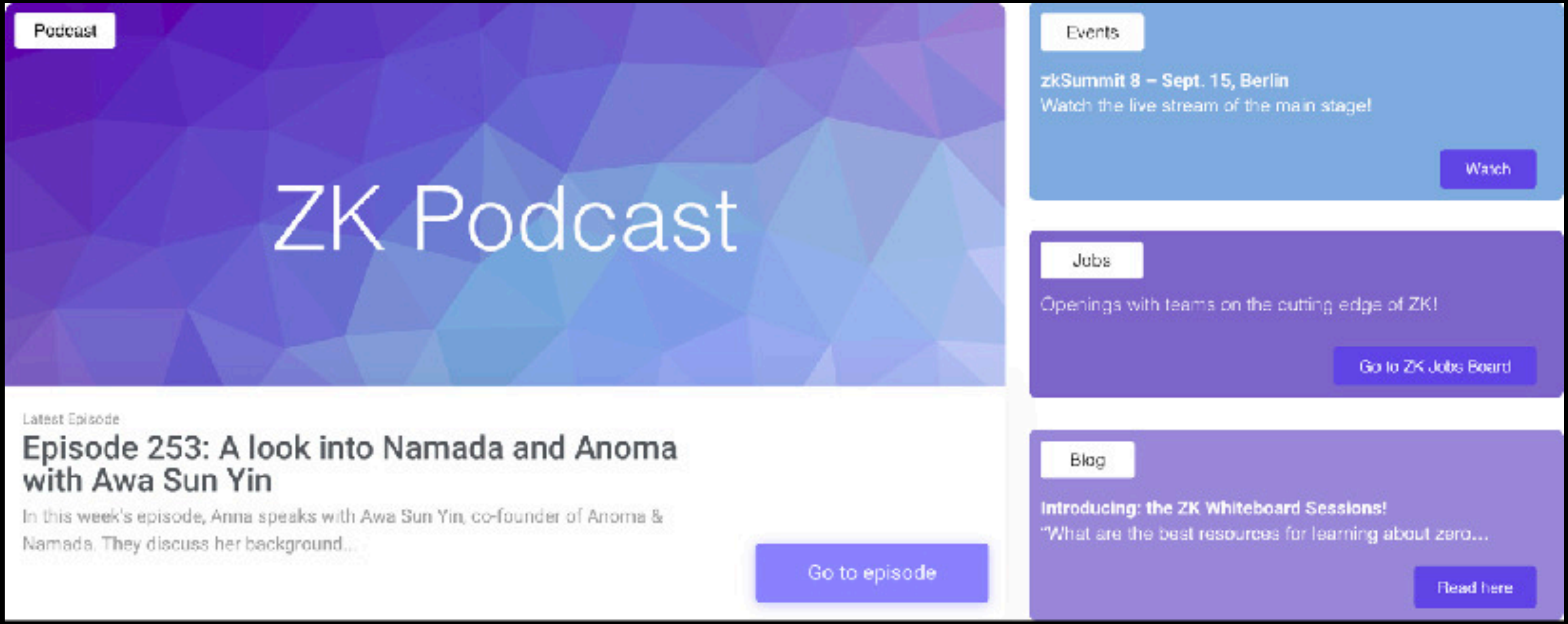


Explains pairings

Book: *https://www.math.u-bordeaux.fr/~damienrobert/csi/book/pairings.pdf*

Educational Resources

Podcasts and zkSummit

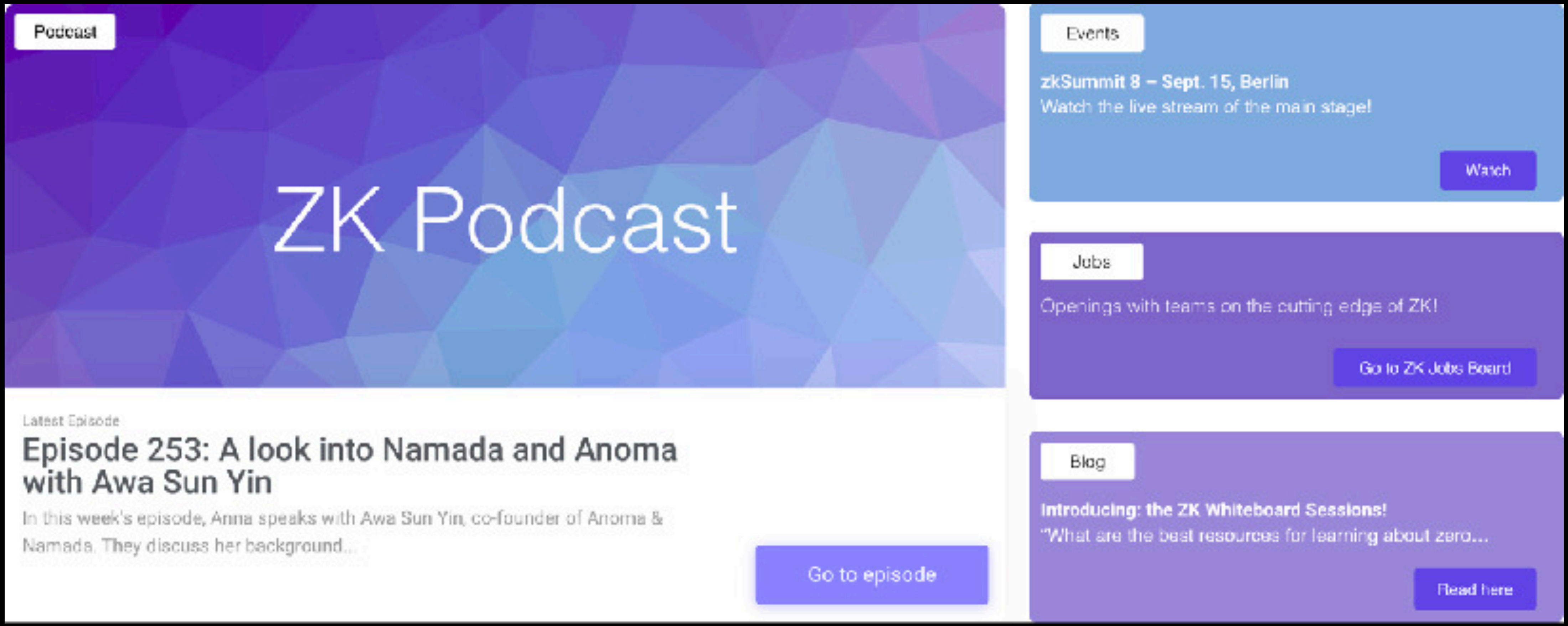


Anna Rose interviews zk people.

Podcast website: *https://zeroknowledge.fm/*

Educational Resources

Podcasts and zkSummit



Podcast website: *https://zeroknowledge.fm/*

Educational Resources

zkHack and whiteboard sessions

zk-puzzles for hands on experience

zkhack website: *https://zkhack.dev/*

Educational Resources

zkHack and whiteboard sessions

ZK Hack is a 4-week virtual event featuring weekly workshops and advanced puzzle solving competitions.

Each week, participants will be able to learn about key ZK concepts and tools and/or compete to find bugs in protocols in our puzzle competition and win prizes.

Starts November 22nd

zkhack website: *https://zkhack.dev/zkhackIII/*

Educational Resources

Blog Posts

Mon, October 24, 2022

HyperPlonk, a zk-proof system for ZKEVMs

By Binyi Chen and Benedikt Bünz | 8 min read

Paper: https://eprint.iacr.org/2022/1355

The core building block of both privacy and scalability for blockchains are zero-knowledge proof systems. These systems allow a prover to convince a verifier that some state transition was done correctly. This could, for example, be a CAPE transaction or a roll-up of many transactions. The two incredible features of zero-knowledge proofs (aka zk-SNARKs) are that (i) the proof reveals no information about the state transition, beyond that it is valid; and (ii) the proof is short and efficient to check, even if the state transition is complex and involves many transactions.

Understanding PLONK

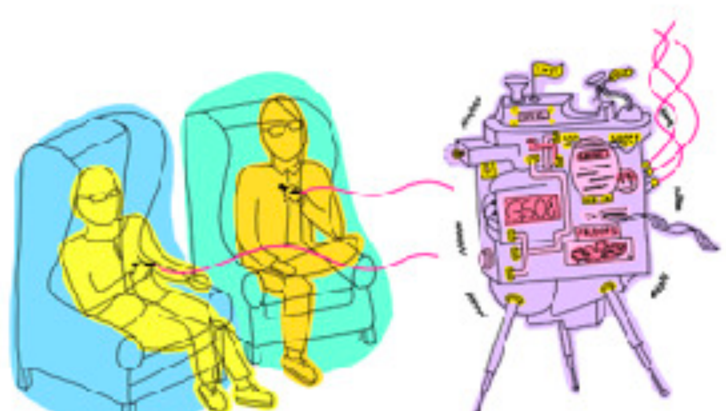
2019 Sep 22 See all posts

Special thanks to Justin Drake, Karl Floersch, Hsiao-wei Wang, Barry Whitehat, Dankrad Feist, Kobi Gurkan and Zac Williamson for review

Very recently, Ariel Gabizon, Zac Williamson and Oana Ciobotaru announced a new general-purpose zero-knowledge proof scheme called PLONK, standing for the unwieldy quasi-backronym "Permutations over Lagrange-bases for Oecumenical Noninteractive arguments of Knowledge". While improvements to general-purpose zero-knowledge proof protocols have been coming for years, what PLONK (and the earlier but more complex SONIC and the more recent Marlin) bring to the table is a series of enhancements that may greatly improve the usability and progress of these kinds of proofs in general.

Groth-Sahai Proofs Are Not That Scary

Mikhail Volkhov, Dimitris Kolonelos, Dmitry Khovratovich, Mary Maller · June 6, 2022



Dankrad Feist About

Inner Product Arguments

Jul 27, 2021

中文版本: 内积证明

Introduction

You might have heard of Bulletproofs: It's a type of zero knowledge proof that is used for example by Monero, and that does not require a trusted setup. The core of this proof system is the Inner Product Argument [1], a trick that allows a prover to convince a verifier of the correctness of an "inner product". An inner product is the component by component product of two vectors:

$$\vec{a} \cdot \vec{b} = a_0 b_0 + a_1 b_1 + a_2 b_2 + \cdots + a_{n-1} b_{n-1}$$

where $\vec{a} = (a_0, a_1, \dots, a_{n-1})$ and $\vec{b} = (b_0, b_1, \dots, b_{n-1})$.

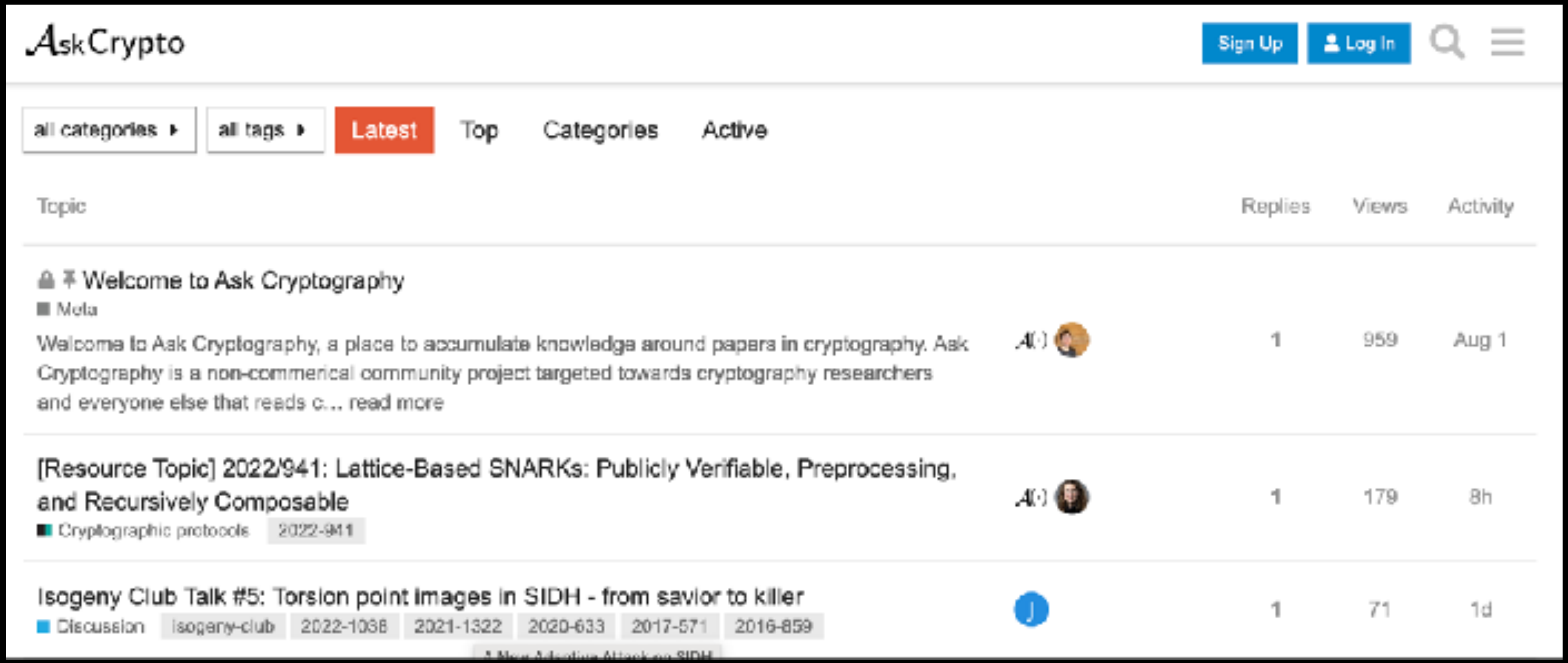
Zero-Knowledge Proofs: STARKs vs SNARKs

Zero-knowledge proof technologies bring privacy to Ethereum. Two of the most compelling zero-knowledge technologies in the market today are zk-STARKs and zk-SNARKs

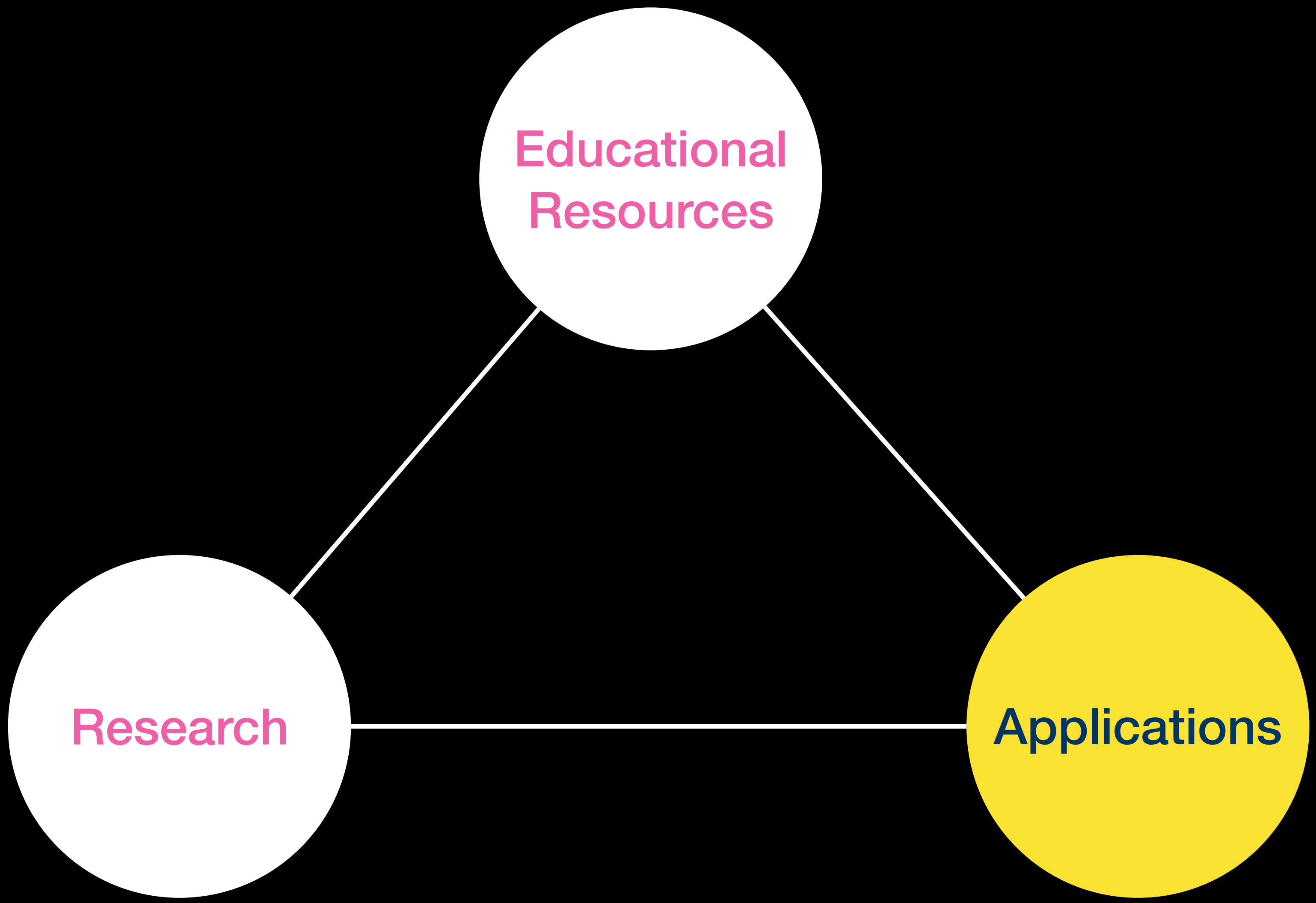
by **Mattison Asher, Coogan Brennan – May 18, 2021**

Educational Resources

Forums

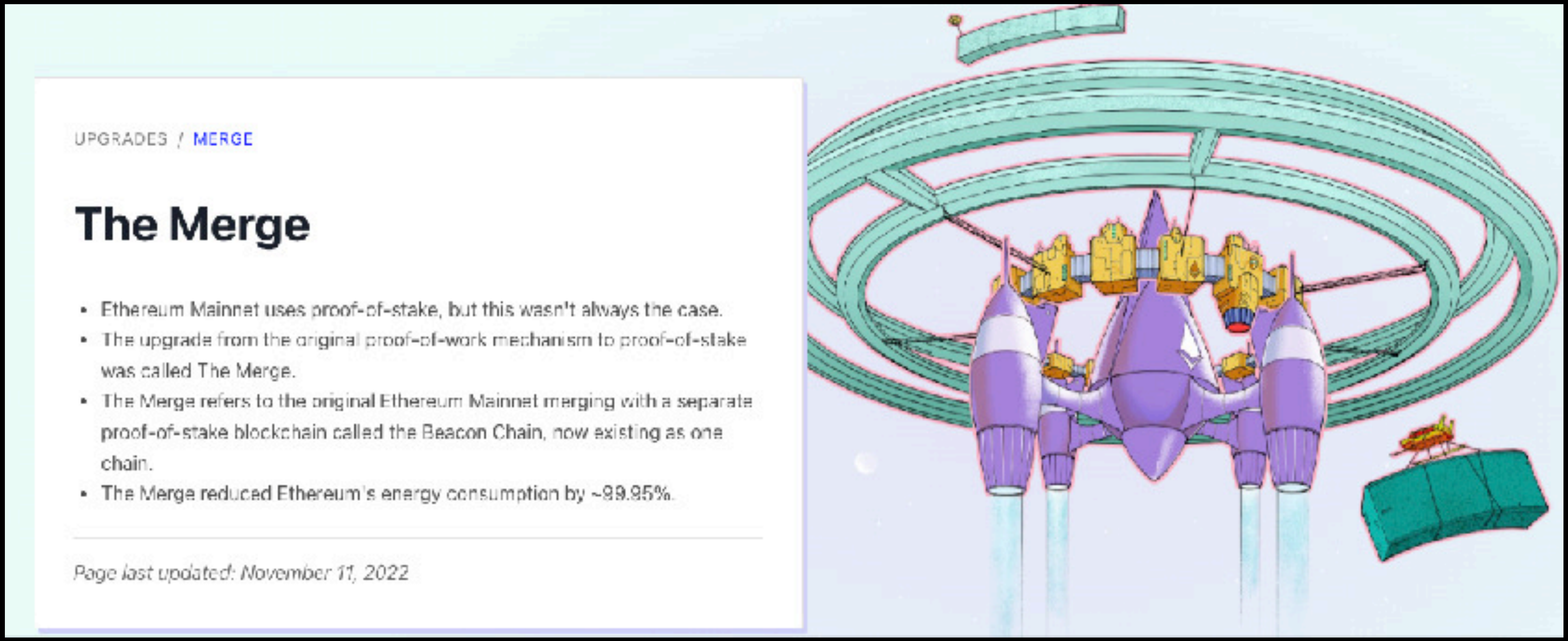


zkhack website: *https://askcryp.to*


Educational Resources
Research
Applications

Applications

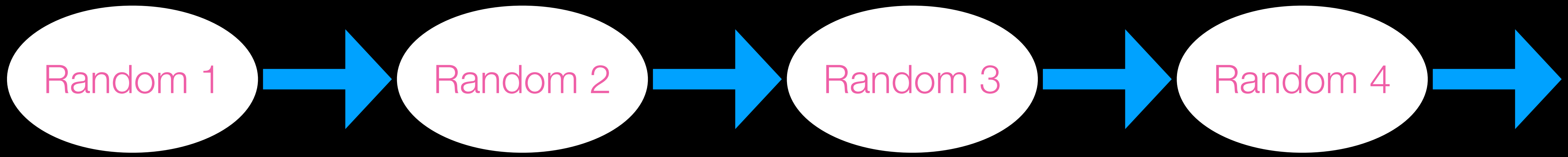
Single Secret Leader Election



Ethereum now uses a proof of stake consensus algorithm.

Applications

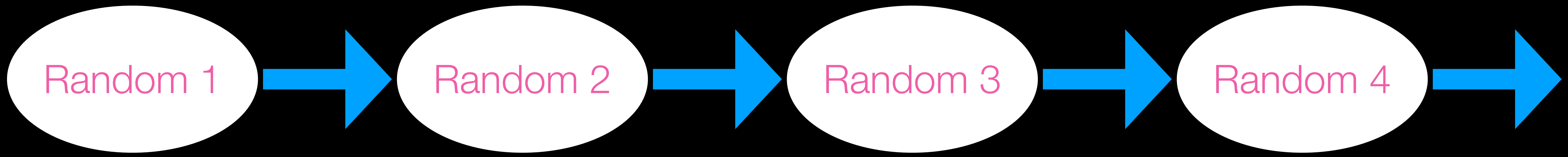
Single Secret Leader Election



- Beacon Chain: consensus protocol relies on a good source of randomness.
- At each time slot, a committee agrees on a new random value.
- The random value determines who the new leader is.

Applications

Single Secret Leader Election

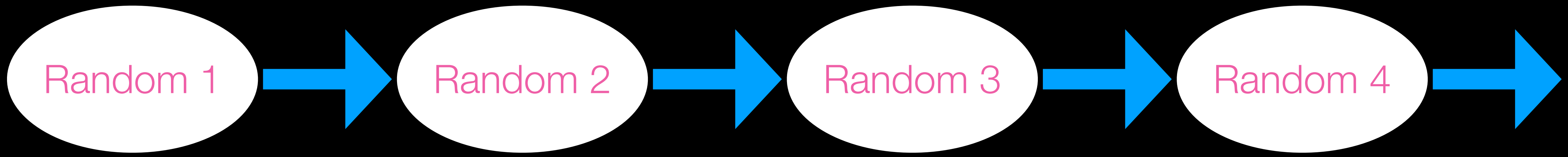


- Beacon Chain: consensus protocol relies on a good source of randomness.
- At each time slot, a committee agrees on a new random value.
- The random value determines who the new leader is.

Leader chooses the new block.

Applications

Single Secret Leader Election

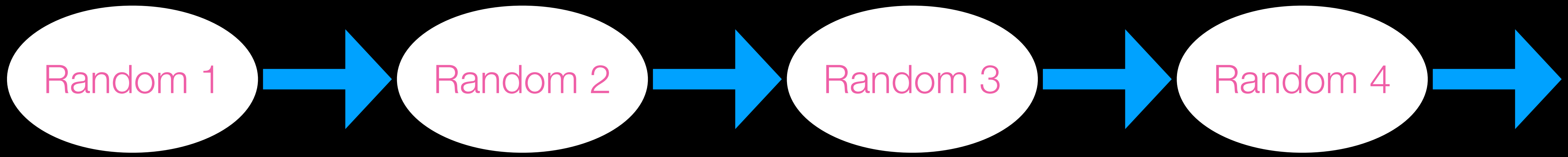


- The leader is a single point of failure.
- If they go offline then the protocol is not live for that time slot.
- Vulnerable to DDOS attacks.

Leader chooses the new block.

Applications

Single Secret Leader Election



- The leader is a single point of failure.
- If they go offline then the protocol is not live for that time slot.
- Vulnerable to DDOS attacks.

Unless nobody knows who the leader is….

Leader chooses the new block.

Applications

Single Secret Leader Election

- Single secret leader election chooses the leader in secret.
- The leader knows who they are, but nobody else does.
- Attacker doesn't know who to DDOS until it's too late.

Unless nobody knows who the leader is….

Applications

Single Secret Leader Election

- Single secret leader election chooses the leader in secret.
- The leader knows who they are, but nobody else does.
- Attacker doesn't know who to DDOS until it's too late.

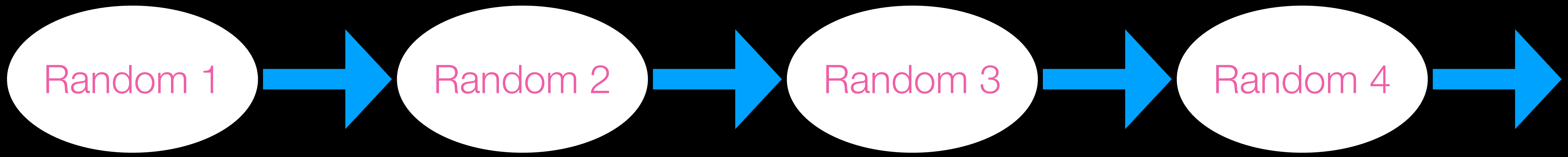
Zero-knowledge shuffle argument a core component of SSLE

Curdleproofs is a *zero-knowledge shuffle argument* inspired by BG12.

docs: https://docs.rs/curdleproofs/latest/curdleproofs/index.html

Applications

Verifiable delay function



- Beacon Chain: consensus protocol relies on a good source of randomness.
- At each time slot, a committee agrees on a new random value.
- Person who plays last can bias the random value by either going offline, or not.

Biasable randomness means biased leader selection.

Applications

Verifiable delay function

- Ongoing inter-company collaboration to build a *verifiable delay function.*
- VDF is a slow function that returns a random value, and is difficult to parallelise.
- Beacon chain returns random value which is then passed to a VDF.

ELECTRIC COIN CO.

ethereum foundation

Filecoin

Protocol Labs

Microsoft Research

Blog post: *https://zkproof.org/2021/11/24/practical-snark-based-vdf/*

Analysis: *https://github.com/khovratovich/MinRoot/blob/main/Sloth_Review.pdf*

Applications

zero-knowledge Virtual Machines

- a.k.a. verifiable computation.
- Virtual machine e.g. tinyRAM or the Etheruem VM specifies memory accesses and allowed operations.
- Check that a computation carried out by the machine is correct.

Applications

zero-knowledge Virtual Machines

- a.k.a. verifiable computation.
- Virtual machine e.g. tinyRAM or the Etheruem VM specifies memory accesses and allowed operations.
- Check that a computation carried out by the machine is correct.

Scroll

zkSync

privacy-scaling-explorations/**zkevm-chain** PSE

STARKNET

polygon Hermez

Applications

zero-knowledge Virtual Machines

- a.k.a. verifiable computation.
- Scaling blockchains: one very important use-case.
- Verifiable cloud computations.

Microsoft | Research   Our research   Programs & events   Blogs & podcasts   About   Sign up: Research Newsletter

Pinocchio: Nearly Practical Verifiable Computation

Bryan Parno, Jon Howell, Craig Gentry, Mariana Raykova
Proceedings of the IEEE Symposium on Security and Privacy | May 2013
Published by IEEE
Best Paper Award

We've been trying for this application for a long time.

Thank-you for listening