CSE 575: Advanced Cryptography
Fall 2024
Lecture 19

Paul Grubbs

paulgrub@umich.edu

Beyster 4709

Agenda for this lecture

- Announcements
- zk-SNARKs: overview and motivation
- Roadmap of rest of class
- interactive oracle proofs (IOPs)
- A first IOP: polynomial identity testing

Agenda for this lecture

- Announcements
- zk-SNARKs: overview and motivation
- Roadmap of rest of class
- interactive oracle proofs (IOPs)
- A first IOP: polynomial identity testing

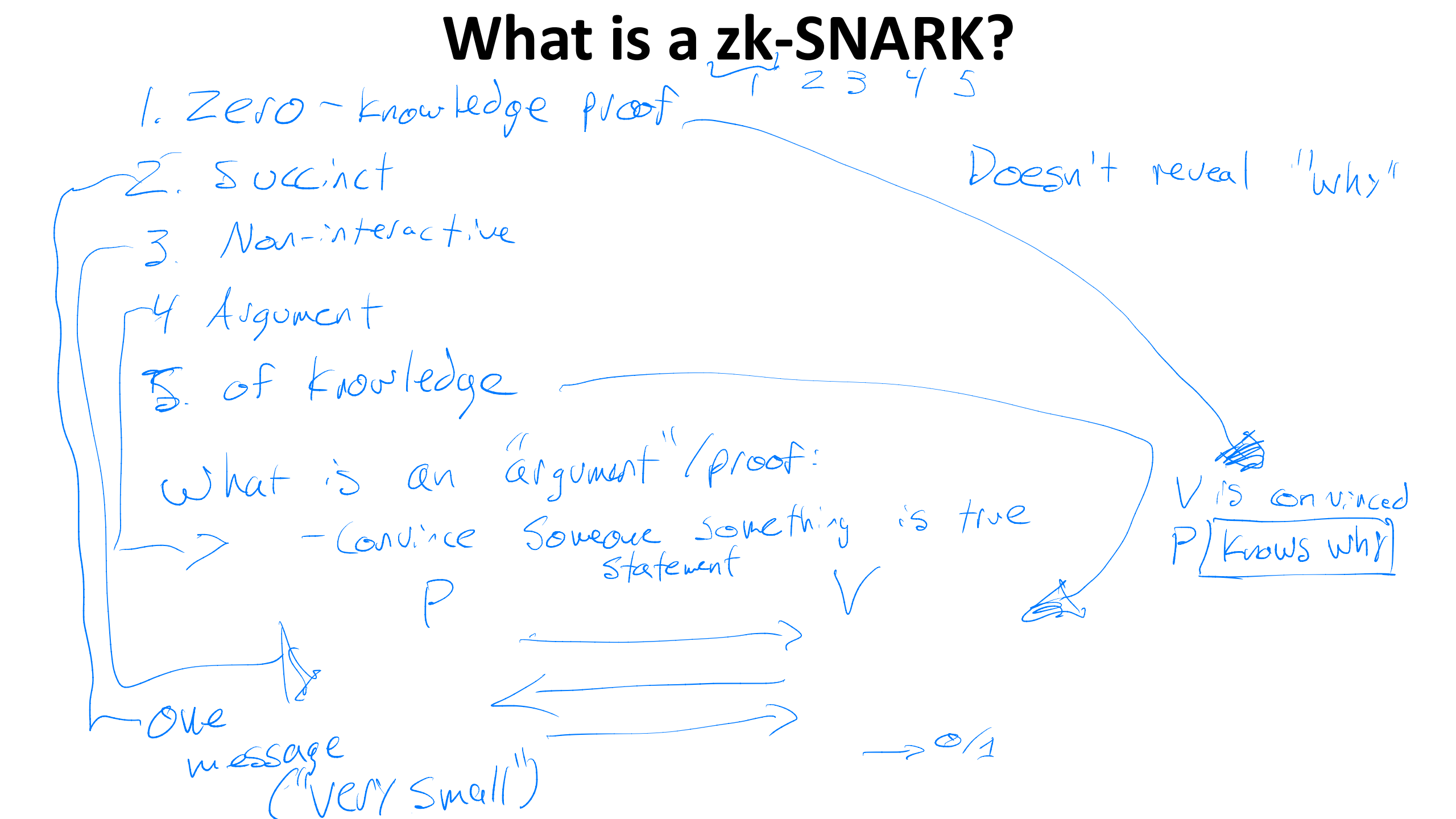
Announcements

- HW3 is out! Due 11/7
- Rest of class: no typeset lecture notes.
  - Can get extra credit for typesetting
- CANDY

Agenda for this lecture

- Announcements
- zk-SNARKs: overview and motivation
- Roadmap of rest of class
- interactive oracle proofs (IOPs)
- A first IOP: polynomial identity testing

What is a zk-SNARK?

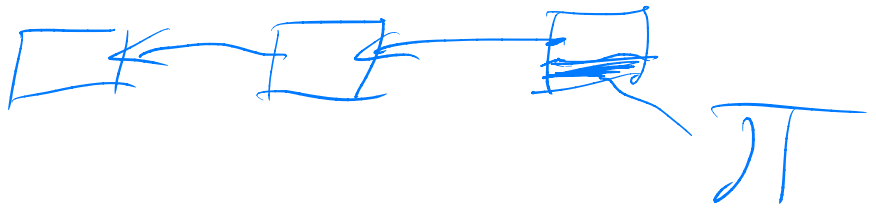


Why do we care about zk-SNARKs/ZKPs?

- Check something without seeing it

Privacy + verification

Zcash: private payments with bitcoin



LLM auditing

Zero-knowledge middleboxes

Photo editing proofs

etc.

Anonymous credentials

P "I am 18 years old" V

Prove something about credential

Agenda for this lecture

- Announcements
- zk-SNARKs: overview and motivation
- Roadmap of rest of class
- interactive oracle proofs (IOPs)
- A first IOP: polynomial identity testing

Roadmap

Plonk from scratch

1. Represent computation in Arithmetic circuit or other constraint systems
2. Transform statement into polynomials
   Polynomials property => computation is correct/ statement is true
3. Design protocols to prove polynomial properties in "ideal" model (IOP)
4. Use cryptography to "compile"
   a. Compile IOP to interactive argument (of knowledge)
   b. Compile interactive to non-interactive (F-S)
   c. Compile to ZK with random masking

Nearly all ZK-SNARKs are built this way

Agenda for this lecture

- Announcements
- zk-SNARKs: overview and motivation
- Roadmap of rest of class
- interactive oracle proofs (IOPs)
- A first IOP: polynomial identity testing

Interactive (oracle) proofs

Relations, languages

- A Language $L \subseteq \{0,1\}^*$. objects having some property.

L_{3COL} = all graphs with 3-coloring

L_{3SAT} = all bool. statements with sat. assignment

- A Relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$

Associate R to L:

$x \in L$ iff $\exists w$ s.t. $(x,w) \in R_L$

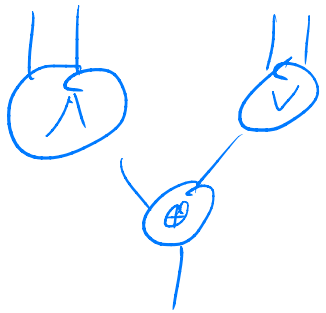
NP relation: has det. PT checker

$R(x,w) = 1$ iff $(x,w) \in R$

Interactive (oracle) proofs

Example relation: circuit-SAT

Language: boolean circuits



$x \in L$ if $\exists$ assignment (w) that satisfies

$R(x, w)$: eval circuit x on input w, return

Interactive (oracle) proofs

- Protocol w/ $\mathbb{P}$, $\mathbb{V}$. Interactive
- $\mathbb{P}$ has two msg. types: normal and "oracle"
  oracle message: label/handle to $\mathbb{P}$ value
- $\mathbb{V}$ msgs: normal or "query oracle i on x"

$\Pi_{IOP} = \langle \mathbb{P}(x, w), \mathbb{V}(x) \rangle$ for relation R

IOP security

- Completeness: $\forall (x, w) \in \mathcal{R}$,

$$\Pr[\text{out}_V \langle P(x,w), V(x) \rangle = 1] = 1$$

- ε-Soundness: $\forall x \notin L$, $P^*(x)$,

$$\Pr[\text{out}_V \langle P^*(x), V(x) \rangle = 1] \leq \varepsilon$$

Polynomial IOP: oracles are bounded-degree polys

Public-coin IOP: V's messages are random bits

Agenda for this lecture

- Announcements
- zk-SNARKs: overview and motivation
- Roadmap of rest of class
- interactive oracle proofs (IOPs)
- A first IOP: polynomial identity testing

IOP for polynomial identity testing