RSAConference2022

San Francisco & Digital | June 6 – 9

TRANSFORM

SESSION ID: **CRYP-R03**

Proofs Without Evidence: Assurance on the Blockchain and Other Applications

MODERATOR:

Dan Boneh
Stanford University
@danboneh

PANELISTS:

Dave Archer
Galois Inc

Riad Wahby
Carnegie Mellon University

Disclaimer

Presentations are intended for educational purposes only and do not replace independent professional judgment. Statements of fact and opinions expressed are those of the presenters individually and, unless expressly stated to the contrary, are not the opinion or position of RSA Conference LLC or any other co-sponsors. RSA Conference does not endorse or approve, and assumes no responsibility for, the content, accuracy or completeness of the information presented.

Attendees should note that sessions may be audio- or video-recorded and may be published in various media, including print, audio and video formats without further notice. The presentation template and any media capture are subject to copyright protection.

Panel

The goal of this panel is to discuss

- Where and in what situations can ZKPs be deployed today?
- What applications can one envisage in the coming years?
- What does one need to keep in mind when deploying such technologies?
- What are the limitations of the technology, both now and inherently?
- How is standardization going?
- Are there different recommended ZKPoK technologies for different applications?
- …..

To fix some terminology I will first give a quick overview

Verifiable Computation/Zero-Knowledge Proofs

VC and ZKPs allow us to solve the following problem…

How can we ask someone to compute something on our behalf and be sure it has been done correctly?

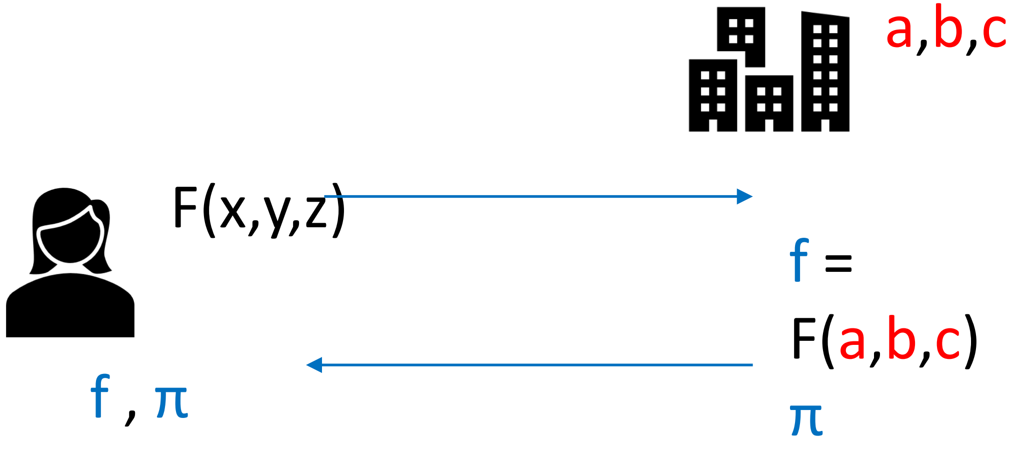
VC is used when the computing party has no secret information

ZKPs are used when the computing party has some secret information

In both situations VCs and ZKPs provide integrity….

> ***"Integrity is doing the right thing, even when no one is watching."***
> A quote often misattributed to C.S. Lewis

Verifiable Computation/Zero-Knowledge Proofs



The computing party has some data a,b,c that it has “committed” to in some way

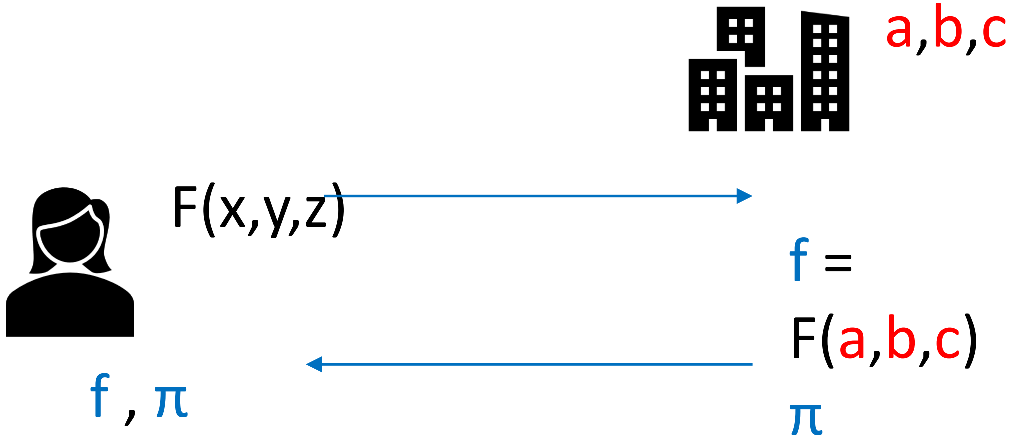
Someone asks for the function F to be computed on this data

This is done and a proof π is also sent back

π is “zero knowledge” in the sense it reveals nothing about a, b or c. Except they were used as inputs to the computation.

If there is no secret information then the proof is just a proof that f has been computed correctly (this is the VC setting)

Verifiable Computation/Zero-Knowledge Proofs



Efficiency Questions:

How complex a function F can one deal with?

What is the size of the proof π?

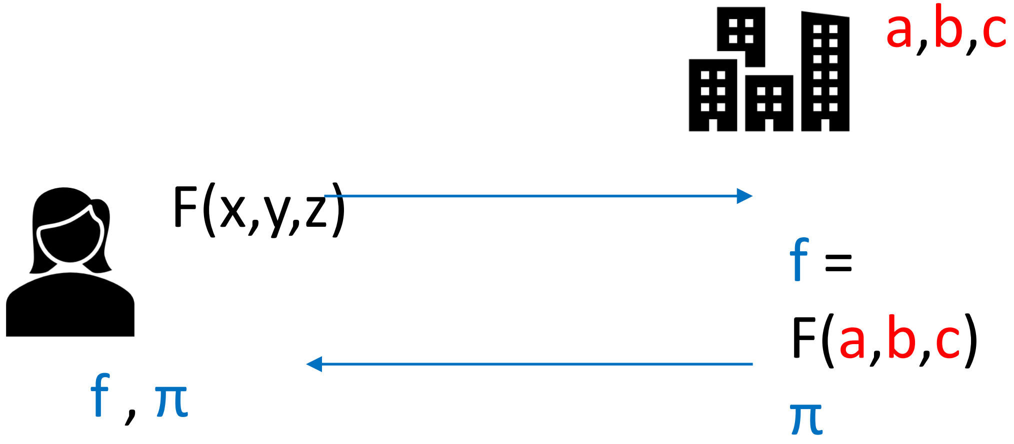
How fast is the prover in terms of size of F?

How fast is the verifier in terms of size of F or size of π?

Is a costly set-up procedure needed?

Is the set-up procedure independent of F?

Verifiable Computation/Zero-Knowledge Proofs



Security Questions:

Is the procedure post-quantum secure?

Does the set-up procedure need to be trusted?

What is the probability that a cheating prover can succeed? [Soundness]

What security is offered for the prover's secrets? [Zero-Knowledge]

SNARKS vs STARKS vs Bullet-Proofs vs MPC-in-the-Head

There are (at-least) five general purpose Zero-Knowledge technologies which are currently practical

- ZK-SNARKs: **Succinct Non-Interactive Argument of Knowledge**
- ZK-STARKs: **Scalable Transparent ARguments of Knowledge**
- Bullet-Proofs:
- MPC-in-the-Head:
- Designated Verifier Proofs: DV-Proofs such as Mac-n-Cheese

Each technology has a different trade off in terms of performance and security

SNARKS vs STARKS vs Bullet-Proofs vs MPC-in-the-Head

Proof Size *:
SNARKs (~128 Bytes)
Bullet-Proofs (~1.3 KBytes)
STARKs (45-200 KBytes)

Prover Time:
STARKs < SNARKs << BulletProofs

Verifier Time *:
SNARKs (~5 msec)
STARKs (~16 msec)
Bullet-Proofs (~1.1 sec)

Trusted Set-Up Required:
SNARK

Post-Quantum Secure:
STARKs, MPC-in-the Head, Modern DV-Proofs

DV-Proofs are more efficient than all others (in every respect), but one has a designated verifier.

MPC-in-the-Head vs STARKs:

MPC-in-the-Head and STARKs are both forms of IOP proofs, so comparable.

MPC-in-the-Head is more efficient for smaller statements compared to STARKs.

STARKs are better for big statements.

* Specific size and timings here, are for common functions seen in practice.