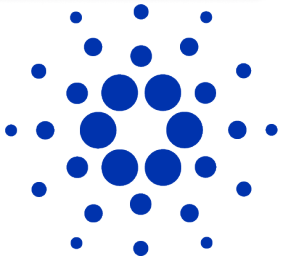
Get these slides ⇨

typst



Zero-knowledge-proofs - part 1

Engineering workshop - Dec 2025 (Ireland)

Pawel Jakubas

Plan of the tutorial

Plan of the tutorial

Let's get a little deeper than usual and understand what main building blocks of ZKP looks like

This tutorial will focus on

Plan of the tutorial

Let's get a little deeper than usual and understand what main building blocks of ZKP looks like

This tutorial will focus on

1. sketching the landscape of what we want to understand during 3-4 parts
2.

Plan of the tutorial

Let's get a little deeper than usual and understand what main building blocks of ZKP looks like

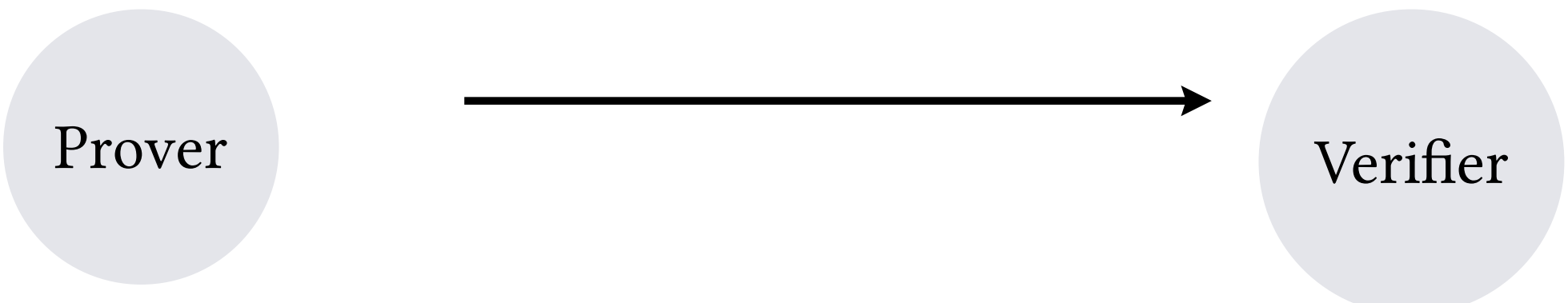
This tutorial will focus on

1. sketching the landscape of what we want to understand during 3-4 parts
2. cover the first part in some detail **elliptic curves**

Verifiable computing vs ZKP (1)

Verifiable computing vs ZKP (1)

There is **asymmetry** built into those systems. It is much easy to get public key from secret. But not the other way



Verifiable computing vs ZKP (1)

There is **asymmetry** built into those systems. It is much easy to get public key from secret. But not the other way

secret -> (easy) -> public

Verifiable computing vs ZKP (1)

There is **asymmetry** built into those systems. It is much easy to get public key from secret. But not the other way

proof + public data shared

Verifiable computing vs ZKP (1)

There is **asymmetry** built into those systems. It is much easy to get public key from secret. But not the other way

public -> (very hard) -> secret

Verifiable computing vs ZKP (1)

There is **asymmetry** built into those systems. It is much easy to get public key from secret. But not the other way

proof verified using public data

Verifiable computing vs ZKP (2)

There is **asymmetry** built into those systems. It is much quicker to verify than prove something.

Verifiable computing vs ZKP (2)

There is **asymmetry** built into those systems. It is much quicker to verify than prove something.

O(n) off-chain

Verifiable computing vs ZKP (2)

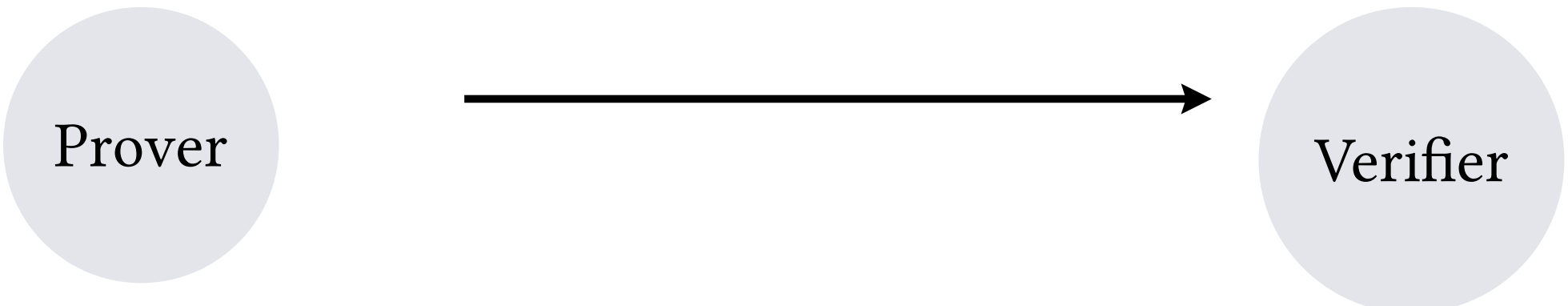
There is **asymmetry** built into those systems. It is much quicker to verify than prove something.

O(n) off-chain

O(log n) on-chain

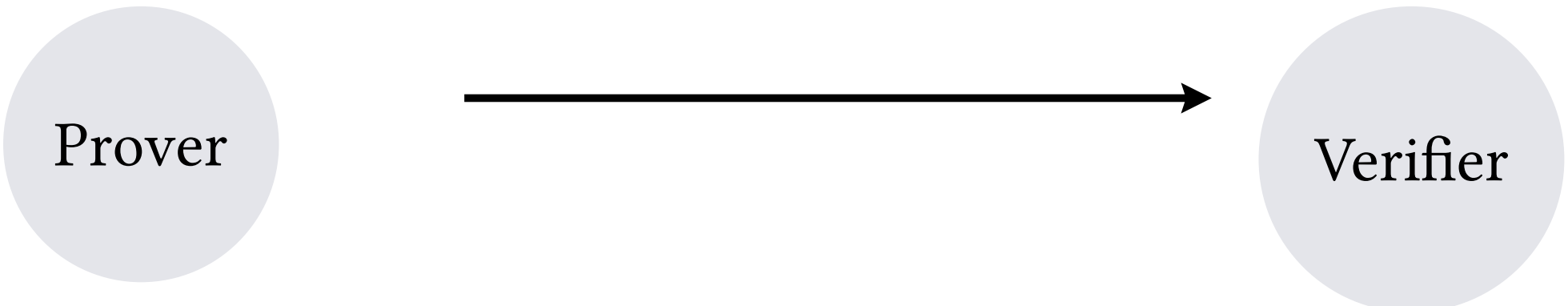
ZKP (3)

Data sent to verifier is compressed, and can be hidden



ZKP (3)

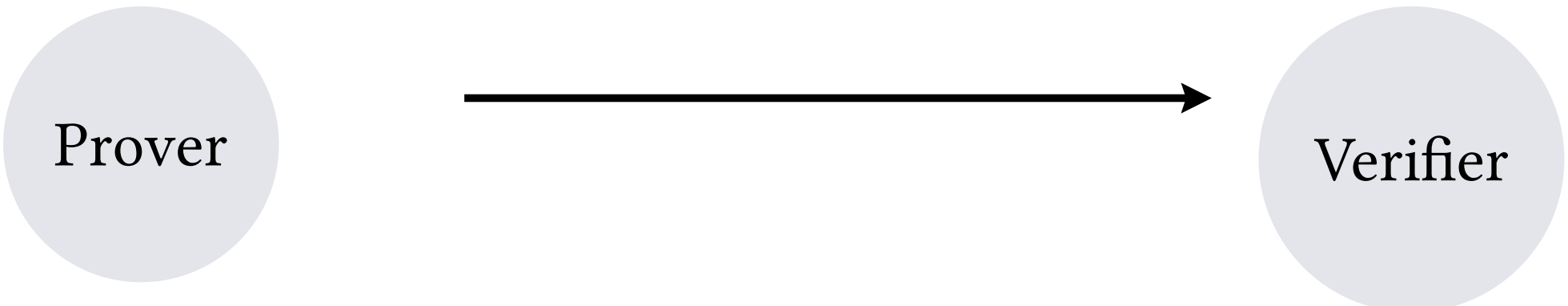
Data sent to verifier is compressed, and can be hidden



size: n

ZKP (3)

Data sent to verifier is compressed, and can be hidden

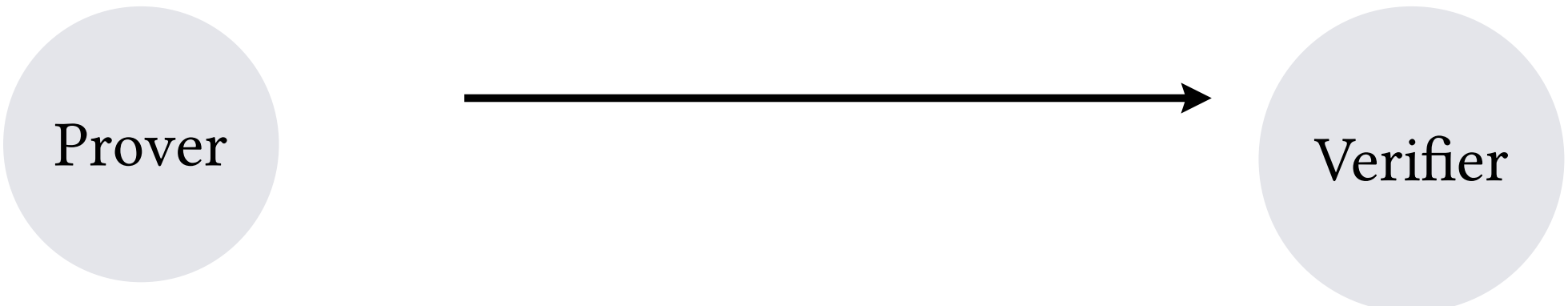


size: n

size: at least log n

ZKP (3)

Data sent to verifier is compressed, and can be hidden



size: n

size: at least log n

size: at least log n

That's it! More to come in the future

⇧My notes on GitHub

Get in touch 👋