

Privacy and Zero-Knowledge Proofs

Thierry Sans

Zero-Knowledge Proofs

ZK proofs in a nutshell

A Zero-Knowledge Proof lets a **prover** prove to a verifier it knows a secret without revealing it

1. **Proof generation**

The prover generates a zero-knowledge proof with a secret input

2. **Proof verification**

The verifier verifies the proof without the secret input

➔ The verifier **does not know the secret** (privacy) but is convinced that the prover knows the secret since it can prove it using a ZK-proof

Two types of zero-knowledge proofs

Interactive proofs

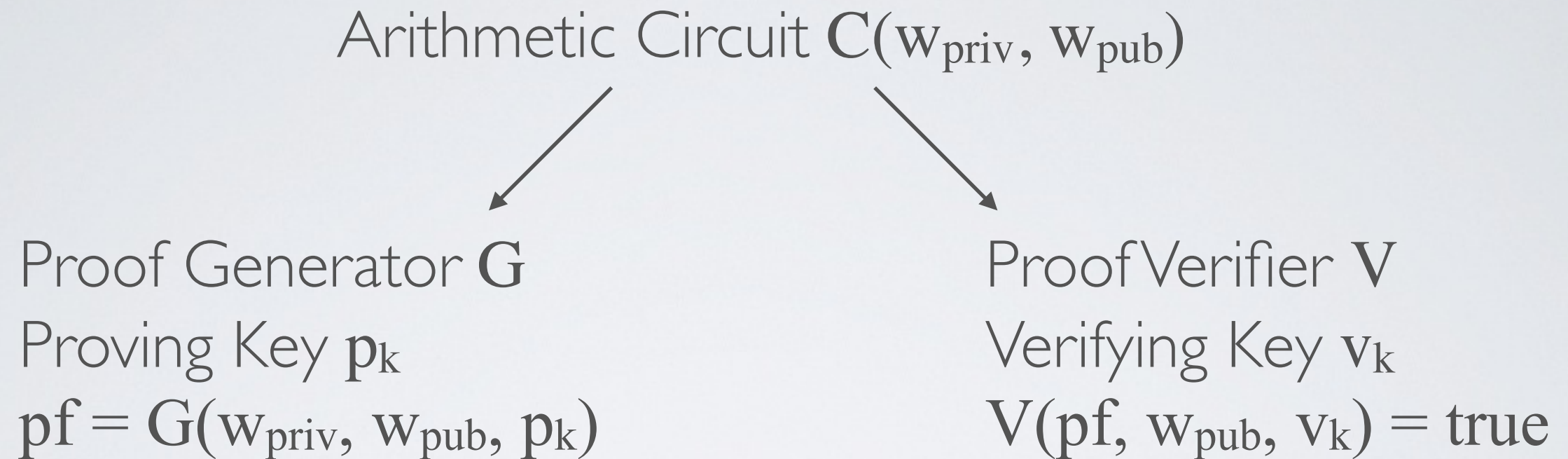
A back-and-forth conversation to prove something

Non-interactive proofs

A single message to prove something

- e.g. digital signature, **ZK-snarks**

Zero-Knowledge Proofs using zk-SNARK



✓ **Soundness**

can always generate a valid proof pf knowing $w_{\text{priv}}, w_{\text{pub}}$

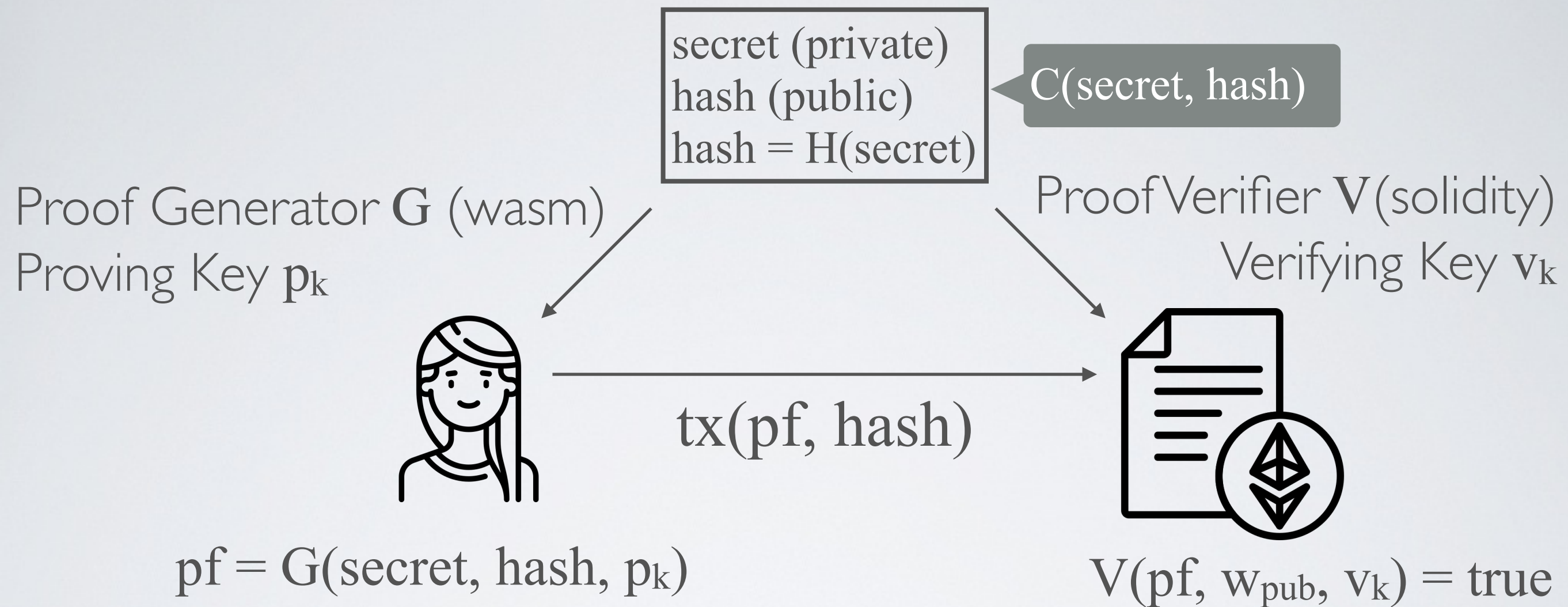
✓ **Completeness**

Cannot generate a valid proof pf knowing w_{pub} only

✓ **Zero-Knowledge**

Verifying pf using w_{pub} does not reveal anything about w_{priv}

Proof of Secret



The generator code is compiled as wasm module (Web Assembly)

The verifier code is compiled into a solidity smart contract

➡ Alice can prove that she knows the secret input without revealing it

✓ The proof and the hash in the transaction does not reveal anything about the secret