

Zero Knowledge

application area & developer tools

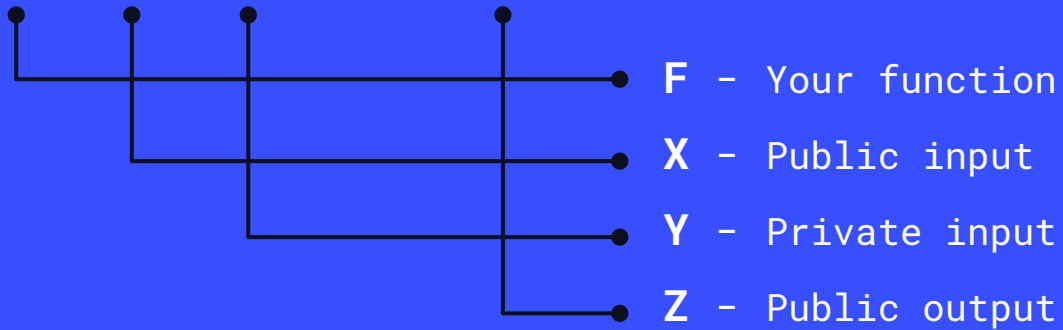
Zero Knowledge Proofs

What is Zero Knowledge Proofs (ZKP)?

Why & where do we need ZKP?

What technologies can be used
for programming?

$$F(X, Y) = Z$$



Proof

Here is **X** and **Z**, I know of an **Y**
such that $F(X, Y) = Z$

Range (Age) proof example

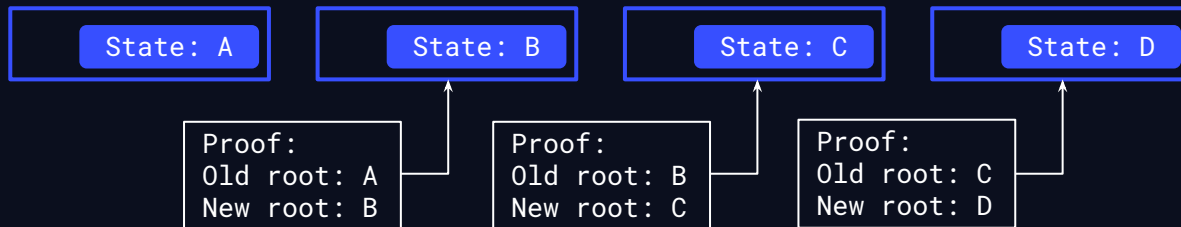
$$F(X, Y) = Z$$



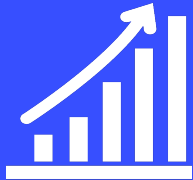
Proof

Here is **X** and **Z**, I know of an **Y**
such that $F(X, Y) = Z$

$$F(X, Y) = Z$$



Application area



Scaling



Privacy



Universal L2
solution for DeFi

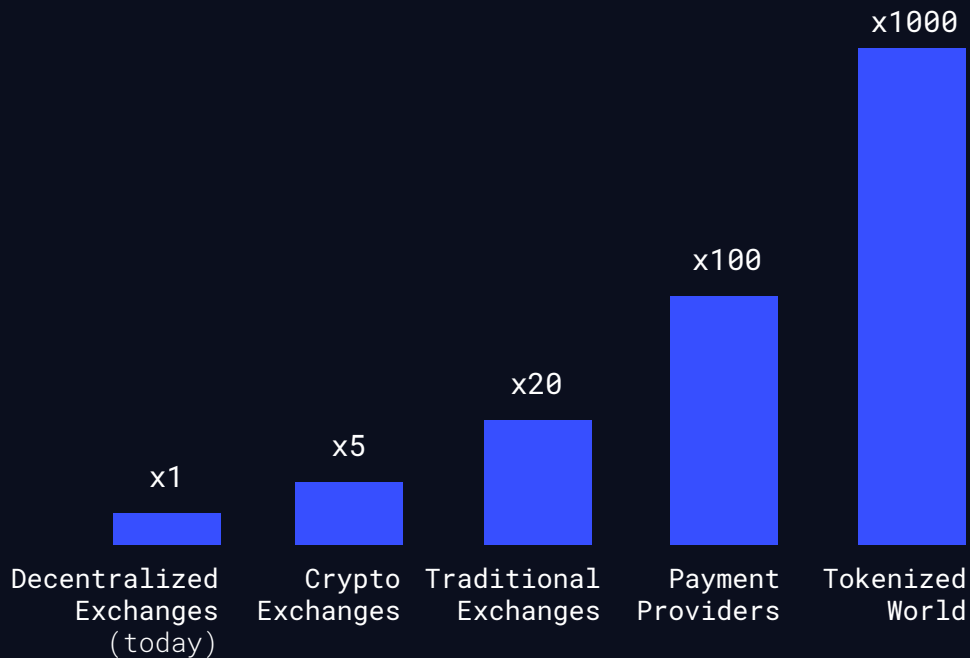
Scaling



Scaling

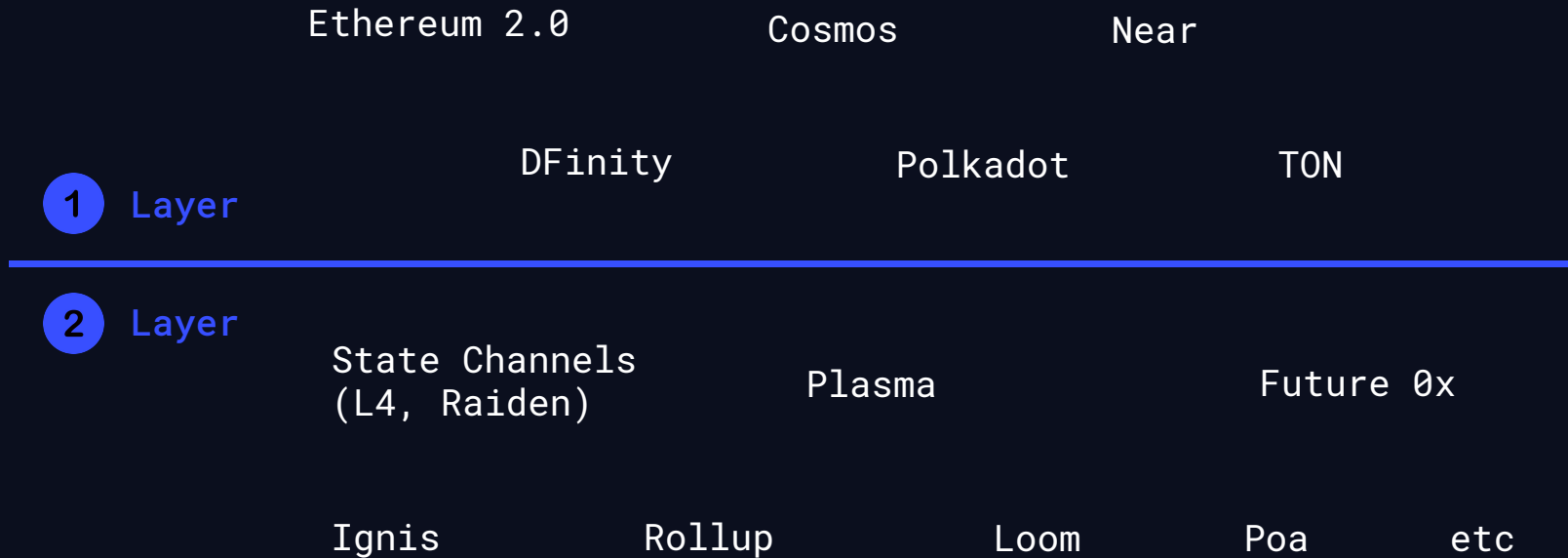


Why scaling?



- 1 Enable new use-cases (games, prediction, logistics, etc.)
- 2 Future need for many TXs

Scaling blockchains



Comparison

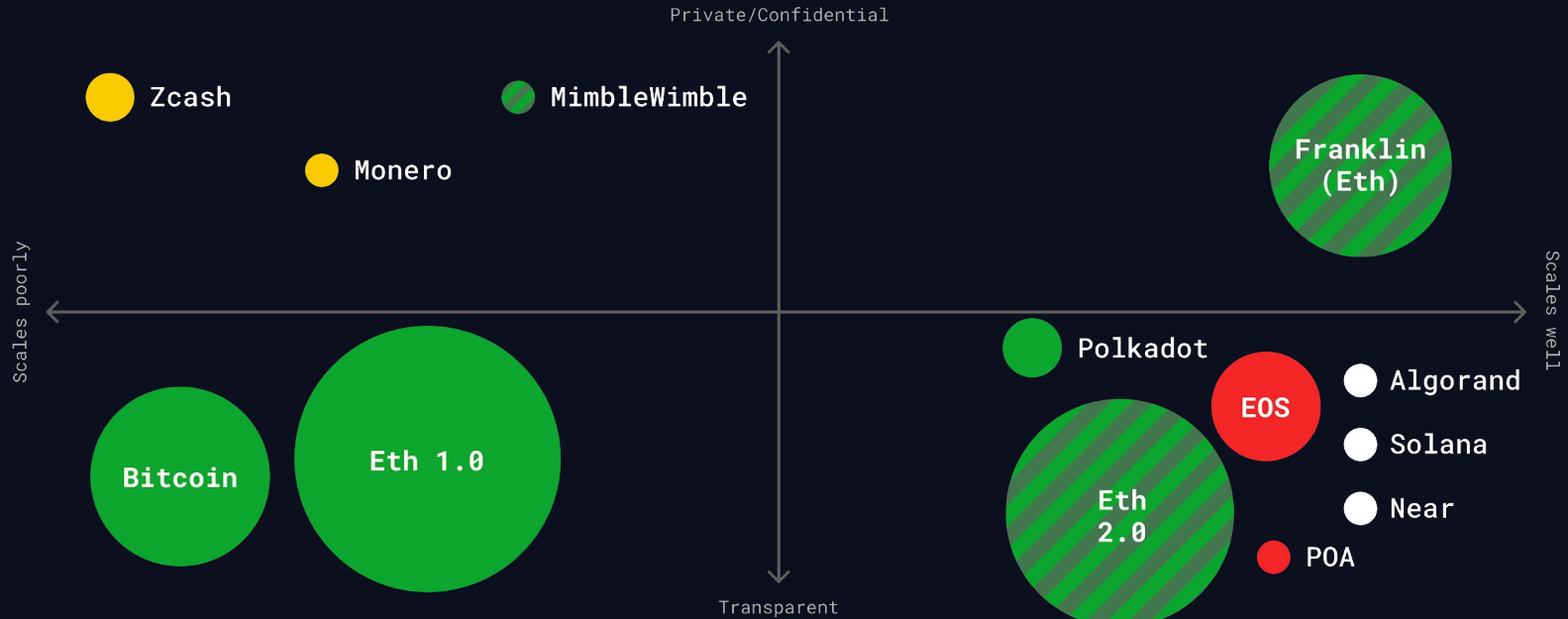
● Secure

● Less secure

● Insecure

○ Not available yet

○ Radius = size of existing ecosystem



Layer two scaling



Privacy

1

**Secure
Payments**

mixers

2

**Settlement layer
for DEXes**

prevent front-running
attack

3

**Private
smart contracts**

Tech approach

zkSNARKs

zkSTARKs

Bulletproofs

Based on range proof &
pedersen commitments
(Monero)

Aztec

Custom privacy protocol
With custom elliptic curve

Legend

- ZKP = Zero-Knowledge Proof
- zkSNARK = ZK Succinct Non-Interactive ARgument of Knowledge
- zkSTARK = ZK Scalable Transparent ARgument of Knowledge
- AZTEC = Anonymous Z(K) Transactions with Efficient Communication



VS



SNARKs

- Required trusted setup: Groth16 SONIC
- Based on elliptic curves: BN256 for Ethereum, bls12-381 for Zcash

STARKs

- Based on hashes in merkle trees
- Not proven by time
- Post quantum resistant

SNARKs vs STARKs

	SNARKs	STARKs
Algorithmic complexity: power	$O(N * \log(N))$	$O(N * \text{poly-log}(N))$
Algorithmic complexity: verifier	$\sim O(1)$	$O(\text{poly-log}(N))$
Communication complexity (proof size)	$\sim O(1)$	$O(\text{poly-log}(N))$
Size estimate for 1 TX	Tx: 200 bytes, Key: 50 MB	45 kb
Size estimate for 10.000 TX	Tx: 200 bytes, Key: 500 MB	135 kb
Ethereum/EVM verification gas cost	$\sim 600k$ (Groth16)	$\sim 2.5M$ (estimate, no impl.)
Trusted setup required?	YES 😞	NO 😄
Post-quantum secure	NO 😞	YES 😄
Crypto assumptions	Strong 😞	Collision resistant hashes 😄

What are SONICs?

SONIC is a proof system, that:

🌐 **Universal**

🔄 **Updatable**



<https://eprint.iacr.org/2019/099>

Libraries

#first_in_class

- **ZoKrates**

Python style, Rust based
Oldest one

#user_friendly

- **Iden3 – Circom**

JS based
Recommended as entry point

#faster

- **LibSnark / EthSnarks**

C++ based
Examples: Roll_up, miximus

#faster

- **Bellman**

Used by zCash, Rust based

Contact



GitHub / Telegram / Twitter

@skywinder



**Petr
Korolev**

Researcher |
Developer |
ETHusiast |

Proposal



- 1 Precompile for generic elliptic curves (BN256, Groth16)
- 2 Cost of transaction data vs Storage
- 3 Wallet support for DApp specific crypto
- 4 WebAssembly support for ADDC, MULQ, CMUL
- 5 WebCrypto support for custom crypto?

Ethereum Improvement Proposals

✨ Fellowship of Ethereum Magicians ✨

Reduce the cost of transaction data

■ EIPs istanbul, eip, scaling, gas

Extensible crypto for wallets

■ Working Groups ■ Wallet Ring security

ZKP study links

BASICS


Awesome ZKP list


<https://github.com/gluk64/awesome-zero-knowledge-proofs>

ZKP From Zero to Hero: R1CS + QAP (Quadratic Arithmetic Programs)

<https://medium.com/@VitalikButerin/quadratic-arithmetic-programs-from-zero-to-hero-f6d558cea649>

ELLIPTIC CURVES IMPLEMENTATIONS

 <https://github.com/dis2/bls12>

 <https://github.com/ethereum/go-ethereum/tree/master/crypto/bn256>

PAIRING

Explainer (by Vitalik)


<https://medium.com/@VitalikButerin/exploring-elliptic-curve-pairings-c73c1864e627>

About Pairings by zcash:


<https://z.cash/blog/snark-explain7/>

ZKP dev links


CIRCUM

 <https://github.com/iden3/circom/>


Examples:

 **Original**


<https://github.com/iden3/circom/blob/master/TUTORIAL.md>


 **Confidential transactions – EthDenver winner project**

<https://github.com/zdai-io/zDai-mixer>

 https://github.com/GuthL/roll_up_circom_tutorial


LIBSNARK / ETHSNARKs

 <https://github.com/HarryR/ethsnarks>


 <https://github.com/howardwu/libsnark-tutorial>

ZKP dev links

BELLMAN (RUST)


 <https://github.com/matter-labs/bellman>


Examples:

 **Edcon2019 material**
https://github.com/matter-labs/Edcon2019_material

 **Igor's example**
https://github.com/snjax/bellman_cube


ZKP in WebAssembly

 https://github.com/kobigurk/wasm_proof

 <https://blog.decentriq.ch/zk-snarks-primer-part-one/>

 <https://slideslive.com/38911801/snarks-for-mixing-si>

Zokrates

 **Devcon ZKPs tutorial**
<https://github.com/leanthebean/puzzle-hunt>