Privacy on the blockchain

怎么在区块链上保护隐私？

Blockchains get their security from the fact that there are many nodes verifying each transaction

区块链的安全性就是因为有很多节点验证每个交易

However, having many nodes verifying every transaction is very bad for privacy.

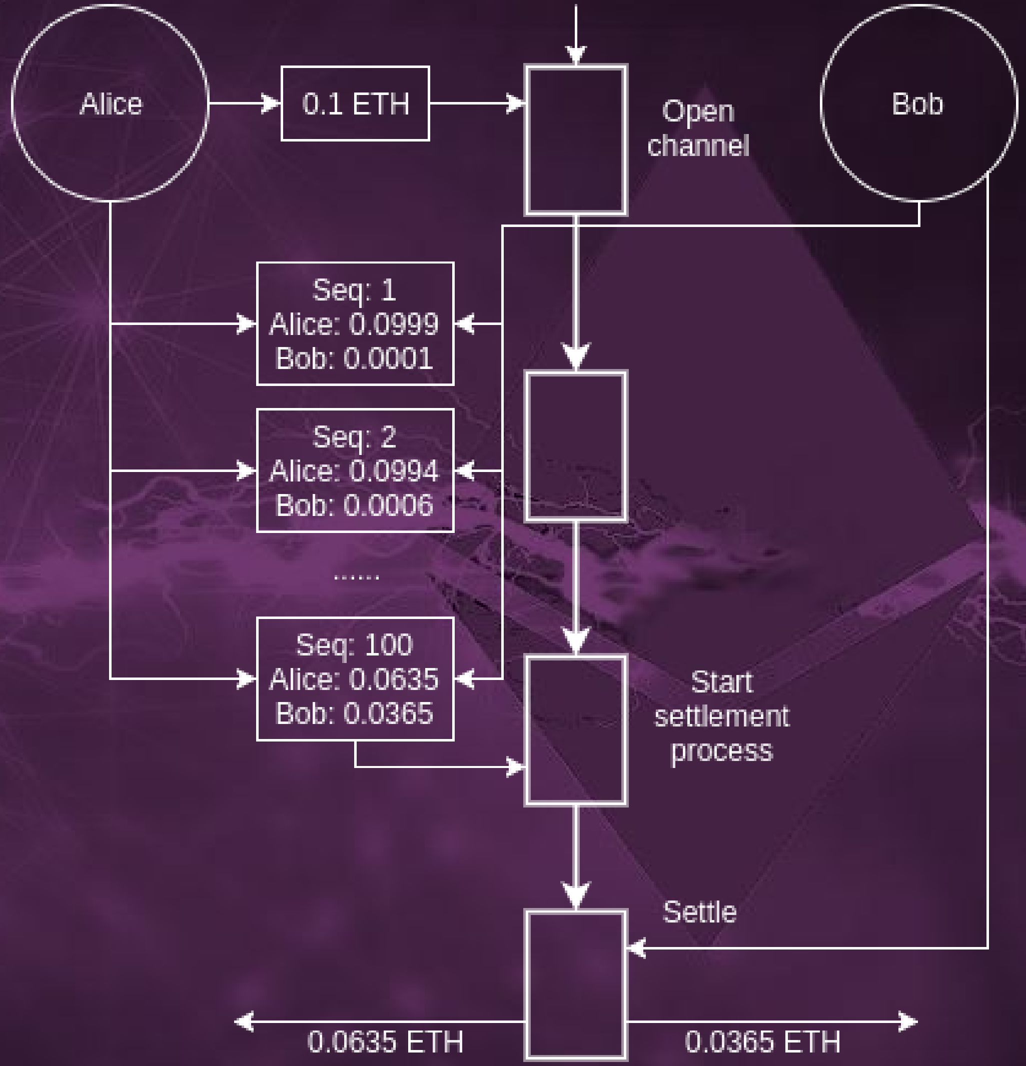
但是，有很多人验证每个交易特别不利于保护隐私

How do we solve this problem? How do we benefit from the security that comes from many nodes verifying the blockchain, without letting those nodes see the data?

怎么解决这个问题？如何有很多节点验证每个交易的安全性，但是不让所有节点看交易的信息？

Solution 1: channels

第一个解决方案：通道

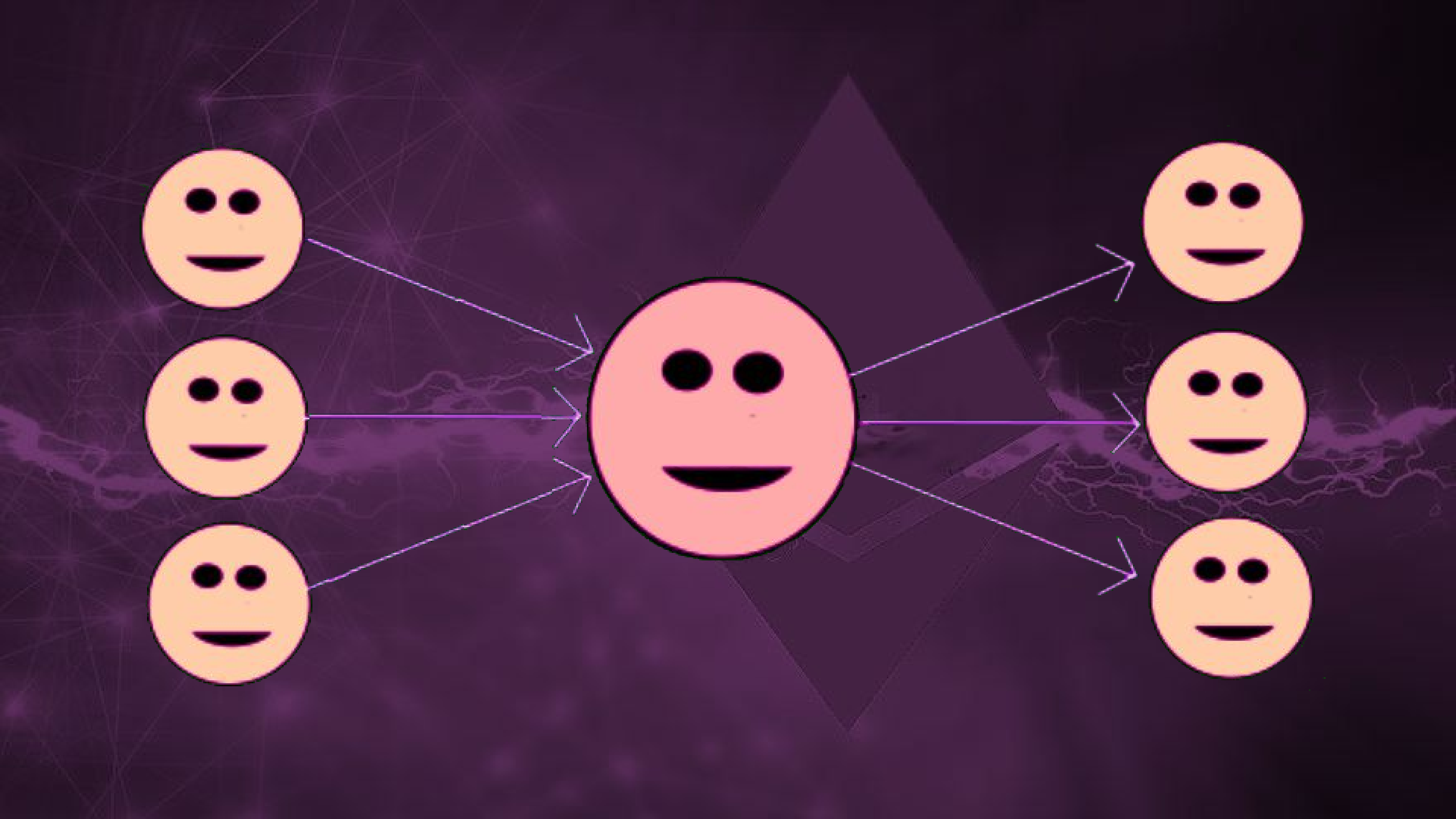

Alice
0.1 ETH
Open channel
Bob
Seq: 1
Alice: 0.0999
Bob: 0.0001
Seq: 2
Alice: 0.0994
Bob: 0.0006
......
Seq: 100
Alice: 0.0635
Bob: 0.0365
Start settlement process
Settle
0.0635 ETH
0.0365 ETH

Transactions inside the channel happen off-chain, and are only visible to the two participants. Only deposit, withdrawal and conflict resolution are on-chain.

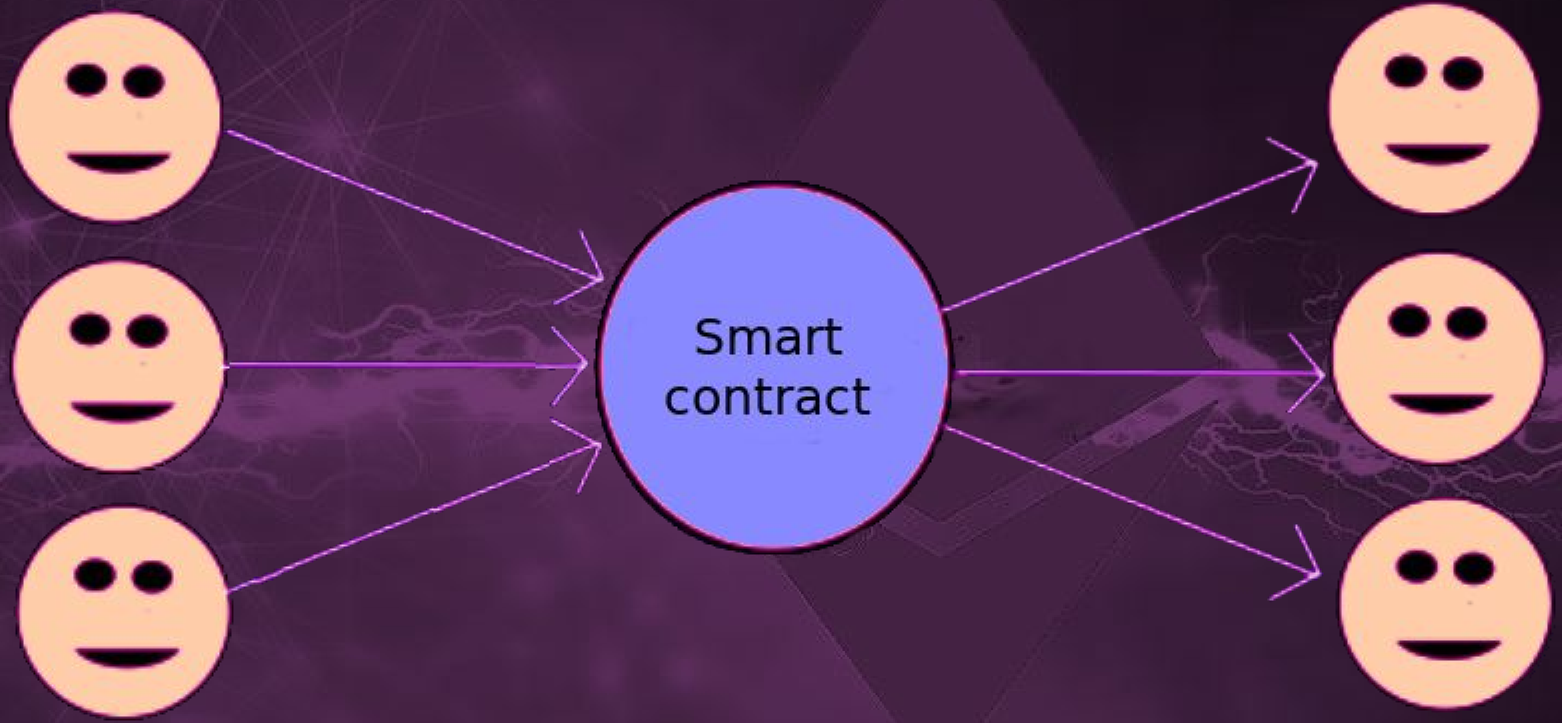
通道之内的交易都不是在链上，只有两个参与者能看到。只有存款、取款和处理相互冲突的交易会在链上

Solution 2: mixers

第二个解决方案:混合器



CoinJoin



Mixers have applications beyond just currency:

除了货币以外，混合器还有很多应用

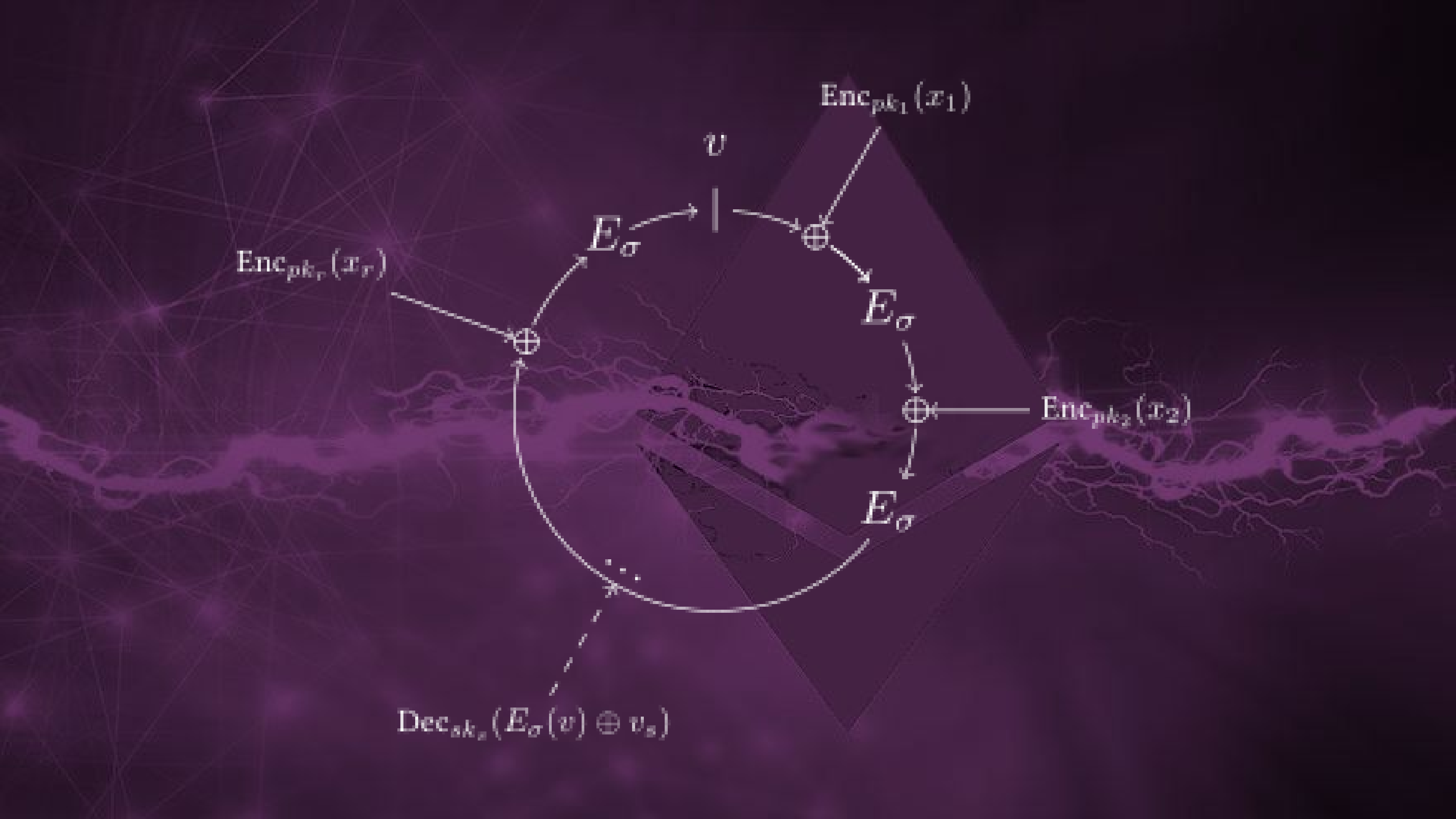
- Privacy-preserving Sybil resistance
  以保护隐私的方式抵抗 Sybil 攻击
- Privacy-preserving polls
  保护隐私的投票

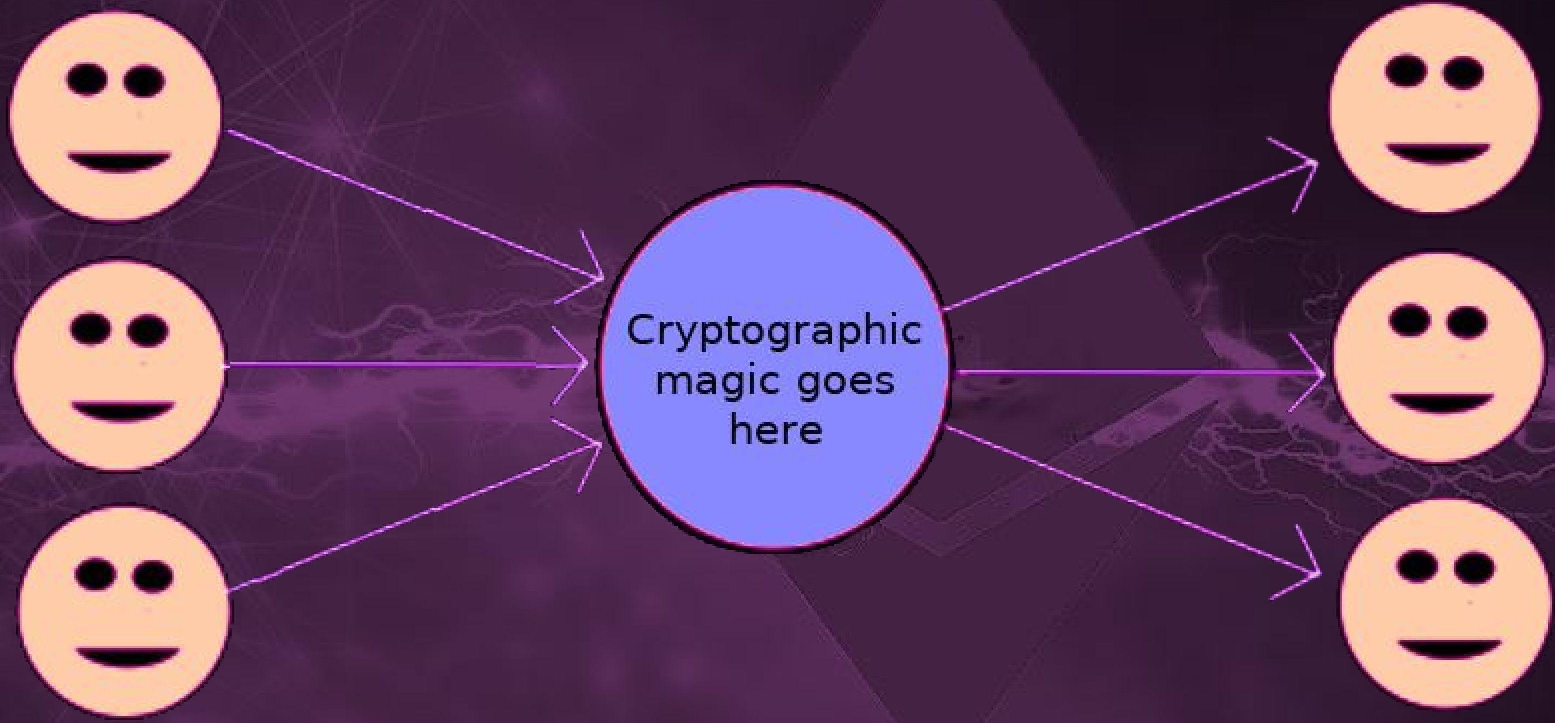
High tech solutions

高科技的解决方案

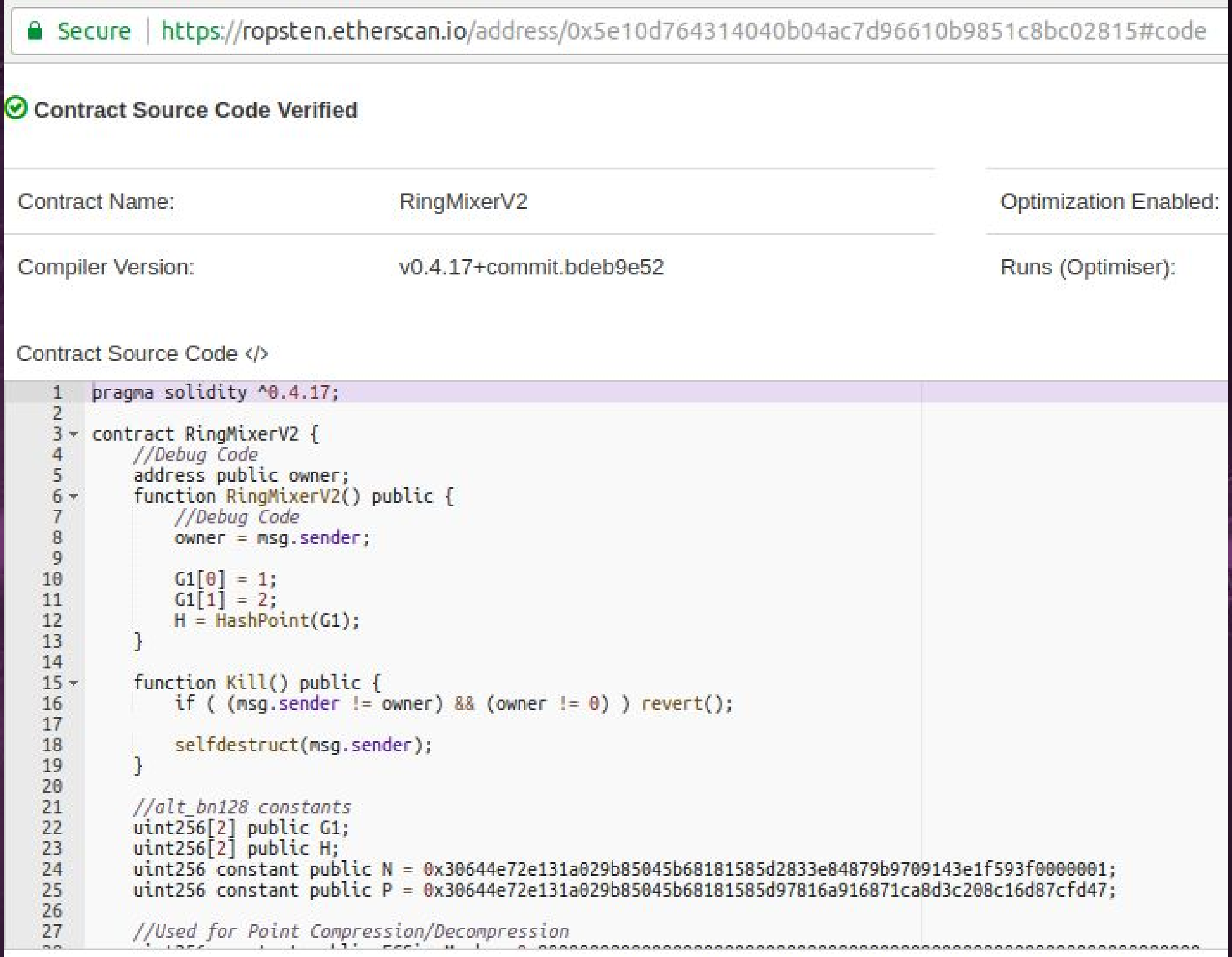
Solution 3: ring signatures

第三个解决方案：环签名


$\mathrm{Enc}_{pk_1}(x_1)$
v
E_σ
$\mathrm{Enc}_{pk_r}(x_r)$
E_σ
$\mathrm{Enc}_{pk_2}(x_2)$
E_σ
$\cdots$
$\mathrm{Dec}_{sk_s}(E_\sigma(v) \oplus v_s)$



Cryptographic magic goes here



Secure
https://ropsten.etherscan.io/address/0x5e10d764314040b04ac7d96610b9851c8bc02815#code

Contract Source Code Verified

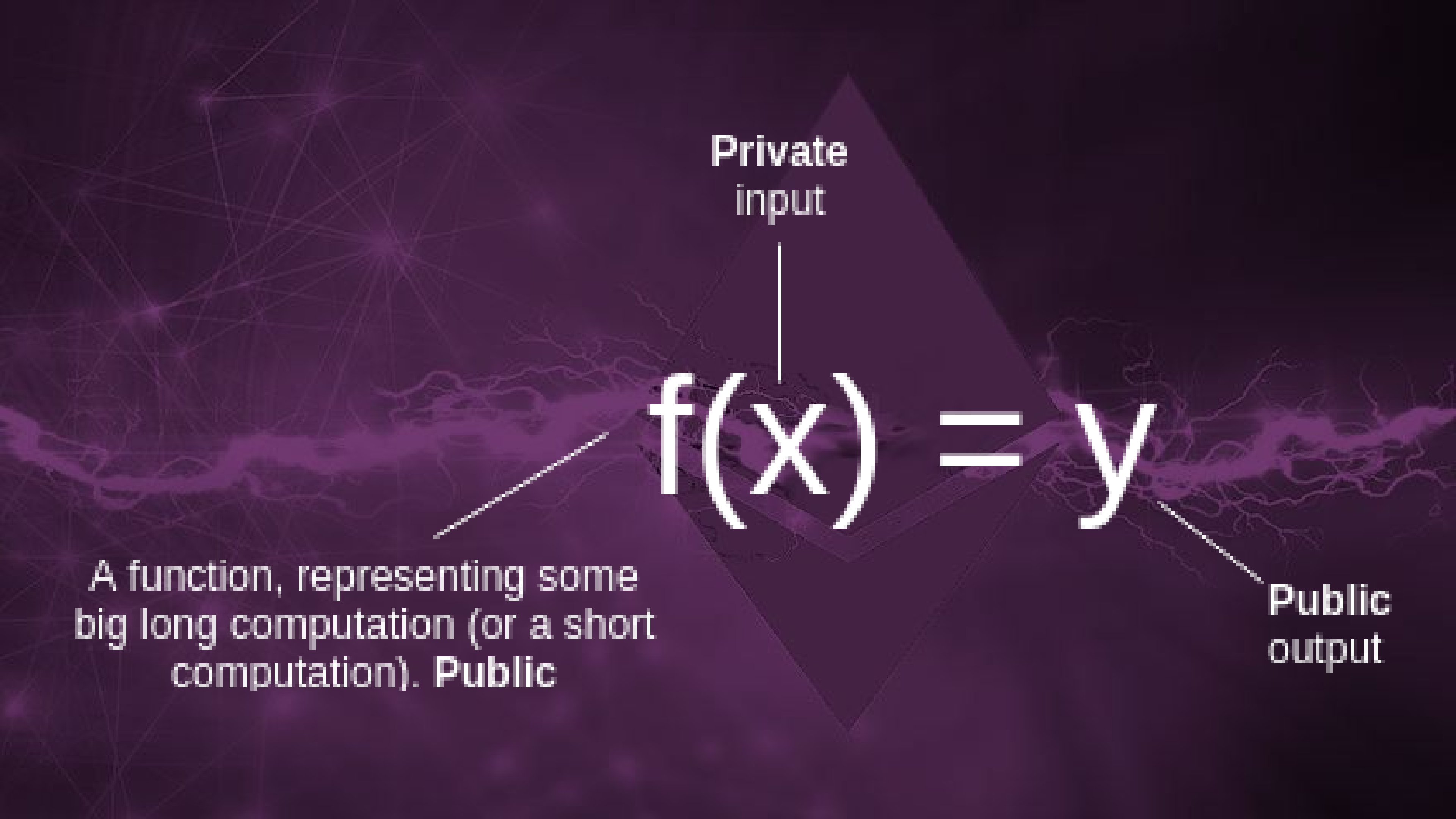
Contract Name:	RingMixerV2	Optimization Enabled:
Compiler Version:	v0.4.17+commit.bdeb9e52	Runs (Optimiser):

Contract Source Code </>

```
pragma solidity ^0.4.17;

contract RingMixerV2 {
    //Debug Code
    address public owner;
    function RingMixerV2() public {
        //Debug Code
        owner = msg.sender;

        G1[0] = 1;
        G1[1] = 2;
        H = HashPoint(G1);
    }

    function Kill() public {
        if ( (msg.sender != owner) && (owner != 0) ) revert();

        selfdestruct(msg.sender);
    }

    //alt_bn128 constants
    uint256[2] public G1;
    uint256[2] public H;
    uint256 constant public N = 0x30644e72e131a029b85045b68181585d2833e84879b9709143e1f593f0000001;
    uint256 constant public P = 0x30644e72e131a029b85045b68181585d97816a916871ca8d3c208c16d87cfd47;

    //Used for Point Compression/Decompression
```

Solution 4: zero knowledge proofs

第四个解决方案:零知识证明



Private input

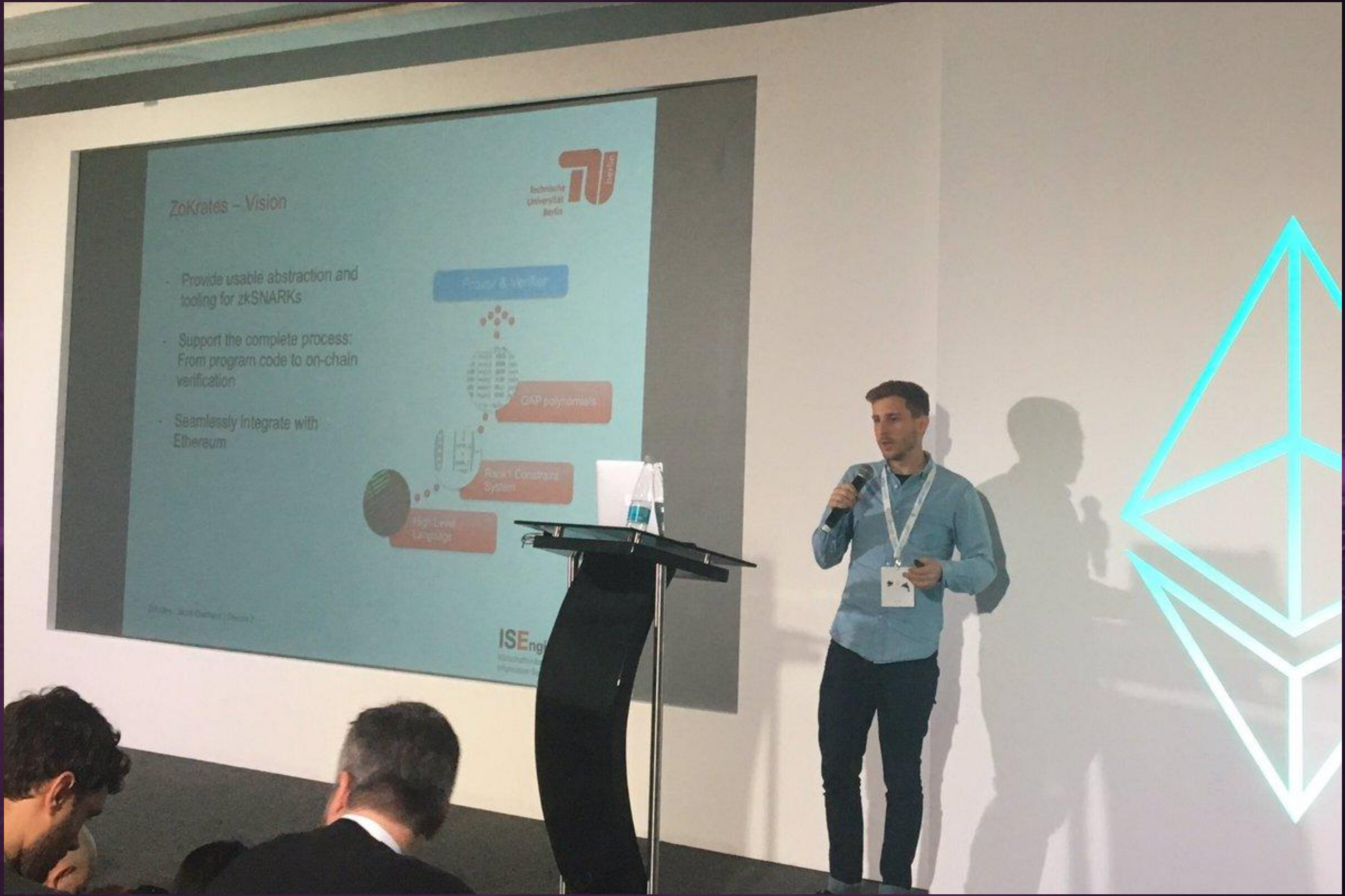
$$f(x) = y$$

A function, representing some big long computation (or a short computation). **Public**

Public output

You can trust me, this transaction is valid, but I won't tell you why!

12qwdiunb23y 87rtx294btrilwabgfli7wfw
rq3ru3brgwavrf1 2HQ38RGW3RZW43T
3WR2fuqi3bgfxv3w g73wuf214124indsfa
wgfiubwggw1y24892412841241uabsfwq



ZoKrates – Vision

- Provide usable abstraction and tooling for zkSNARKs
- Support the complete process: From program code to on-chain verification
- Seamlessly integrate with Ethereum

Prover & Verifier

QAP polynomials

Rank1 Constraint System

High Level Language

Benefits

好处

- General purpose (like ethereum!)
  通用（像以太坊一样！）
- Very strong privacy and security guarantee
  很健全地保护隐私和安全性

Drawbacks

缺点

- Trusted setup 需要信任的设置
- Proof generation inefficient 生成证明的效率低
- Relatively untested technology 相对未完全测试的技术

Resources

资料

- ZK-SNARKS: https://medium.com/@VitalikButerin/zk-snarks-under-the-hood-b33151a013f6
- 零知识证明，中文版：http://unitimes.media/knowledge/zk_snarks_under_the_hood_cn.html
- Ring signatures (code) 环签名（代码）:https://ropsten.etherscan.io/address/0x5e10d764314040b04ac7d96610b9851c8bc02815#code
- Privacy on the blockchain 区块链上的隐私 (article from 2016年的文章） https://blog.ethereum.org/2016/01/15/privacy-on-the-blockchain/