Scan me

Devoleum && Privacy

Using zk-SNARKs and federated learning

Lorenzo Zaccagnini

Devoleum

Devoleum tracks and stores every single step of the extra virgin olive oil supply chain using Blockchain and AI (data correctness and predictions), allowing transparency and privacy at the same time thanks to:

- Zero Proof Knowledge (zk-SNARKs)
- Federated Learning (distributed AI training)

And many other things….

The Team

LORENZO ZACCAGNINI

Devoleum co-founder

Blockchain Mentor @Udacity

Psychology Msc @Sapienza University

Blockchain & AI scholarships - Facebook, Google, Udacity

The Team

ELISA ROMONDIA

Devoleum co-founder

Data Analyst mentor @Udacity

Blockchain & AI scholarships - Facebook, Google, Bertelsmann, Udacity

Forbes 60 Women-Led Startups That Are Shaking Up Tech Across The Globe

InspiringFifty 2018

Top10 innovative female entrepreneurs in UE by StartHer @Station F

Privacy issues

Training centralized AI models requires data from users

Ethereum is open and immutable

We need to handle different scenarios:

- Be GDPR compliant (UE data privacy regulation)
- Protect customers privacy
- Companies has and produce confidential data that can’t be public (financial, employees etc...)

Privacy solutions

- Train AI models with federated learning (distributed), this does not require data from our users. The training occurs on the users devices and only the models updates are uploaded, not the data.

- Storing data outside Ethereum using IPFS or other traditional solution, keeping in our smart contracts only an anonymous hash reference.

- Using Zero Proof Knowledge zk-SNARKs, to prove a secret on Ethereum, without revealing confidential data

Zero Proof Knowledge

A zero-knowledge proof or zero-knowledge protocol is a method by which one party (the prover) can prove to another party (the verifier) that they know a value x, without conveying any information apart from the fact that they know the value x. **(Wikipedia)**

zk-SNARKs or “Zero-Knowledge Succinct Non-Interactive Argument of Knowledge” is the best ZK match for Ethereum because:

- Doesn’t require interaction between users (asynchronous)
- Succinct means light, improves scalability and fast verification

Zero Proof Knowledge

The verification is probabilistic, close to 100% but it’s never 100%

The example Zcash

Zcash is a privacy-protecting, digital currency built on strong science. Transact efficiently and safely with low fees while ensuring digital transactions remain private.

A Z-to-Z transaction appears on the public blockchain, so it is known to have occured and that the fees were paid. But the addresses, transaction amount and the memo field are all encrypted and not publicly visible. Using encryption on a blockchain is only possible through the use of zero-knowledge proofs (zk-SNARKs)

https://z.cash/technology/

Libsnark

The team that developed Zcash produced Libsnark, a C++ library to create algebraic circuits.

Resources:

https://github.com/scipr-lab/libsnark

https://z.cash/technology/zksnarks/

It’s not easy

zk-SNARKs cannot be applied to any computational problem directly; rather, you have to convert the problem into the right “form” for the problem to operate on. The form is called a “quadratic arithmetic program” (QAP), and transforming the code of a function into one of these is itself highly nontrivial.

Vitalik Buterin

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

Zokrates

ZoKrates is a toolbox for zkSNARKs on Ethereum. It helps you use verifiable computation in your DApp, from the specification of your program in a high level language to generating proofs of computation to verifying those proofs in Solidity.

https://zokrates.github.io/

It’s free and open source!

Zokrates pipeline

CODE

↓

R1CS

↓

QAP

↓

ZKSNARK

Zokrates steps

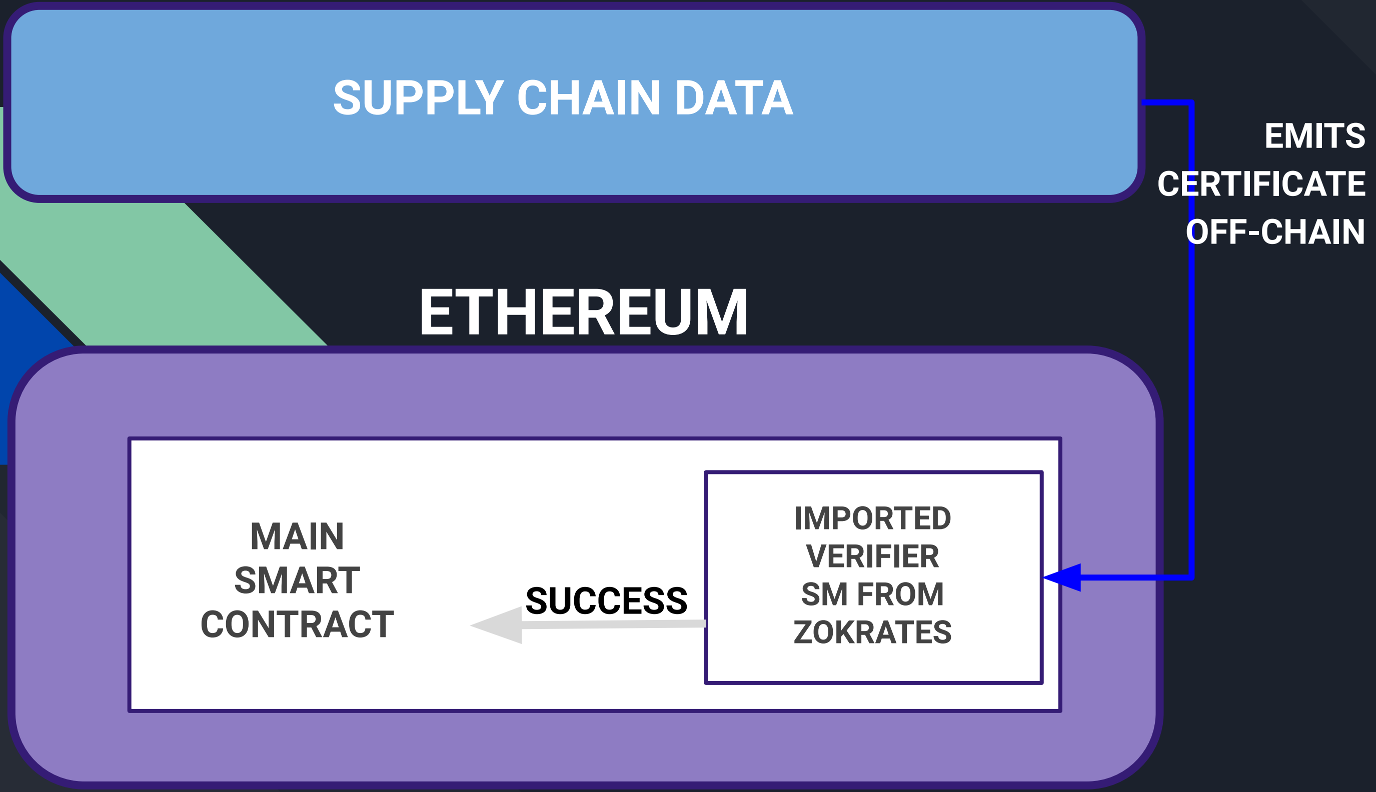
compile: generates the circuit

setup: one time process, generates a proving key and a verification key for this circuit, this is a pseudo-random process that will create different values each time.

compute-witness: a zk-SNARK proof asserts that the prover knows some private information that satisfies the original function.

generate-proof: produces a proof, using both the proving key and the witness encoding of the previous step.

export-verifier: creates a Solidity contract able to verify the proof.


SUPPLY CHAIN DATA
EMITS CERTIFICATE OFF-CHAIN
ETHEREUM
MAIN SMART CONTRACT
SUCCESS
IMPORTED VERIFIER SM FROM ZOKRATES

1. User emits certificate for an asset
2. SC verifies certificate without storing confidential data
3. If the verification is successful allows the operation (ex: token emission)

Devoleum OpenSea Marketplace example

ZKSNARK verified asset

https://rinkeby.opensea.io/assets/0x88a051d837fed5570329ecfb3ee42cff79e2a111/9#related

Devoleum zk-SNARK privacy benefits:

- Use financial open API from banks, allowing our project to operate in the public ethereum network without using cryptos.

- Easier adoption by enterprises, supply chains relations can be managed using smart contracts without requiring confidential data, especially in the digital twins creation.

- Protect data privacy in high risk scenarios, organized crime in food supply chains can use sensitive data against other competitors. Devoleum can expose workers exploitation, money laundering, frauds.

Devoleum zk-SNARK economic benefits:

- Verification is cheaper thanks to delegated computation
- Scalability, we can verify a large number of assets only by verification and requiring big quantity of data transfer (risking to exceeds the gas limits)
- Public ethereum becomes cheaper than a private consortium setup
- Proof always cost the same, we can predict costs

AI and zk-SNARKs integration

We’re planning to automate zk-SNARKs generation using our distributed AI models, all off-chain in oracles

1. AI verify correctness of the data (harvesting quantity, weather conditions etc...)
2. AI creates the zk-SNARK using company confidential data **client side**
3. Proof is stored on the blockchain and the ownership of the digital twin is stored on the blockchain

Note that our AI models are trained on the users devices using federated learning and we do not require any sensitive data of our users

Thank you

Devoleum: https://www.devoleum.com/

LinkedIn: https://www.linkedin.com/in/lorenzo-zaccagnini/
https://www.linkedin.com/in/elisa-romondia/

Twitter: @LorenzoZcg @elisaromondia