TECHNOLOGY FOR SECURITIES

ZK-proof applications for the Confidential Security Contract (XSC) standard

Dusk Network

First permissionless blockchain for digital securities

Focus on privacy

Programmable smart contracts

Fork resistant

Direct settlement

DUSK

Zero-knowledge proofs

A zero-knowledge proof is a protocol in which one party can prove to another party the validity of a statement, without conveying any information apart from that fact.

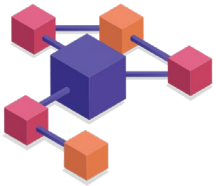
A ZK proof must satisfy the following three properties: **Completeness**, **Zero-knowledge(ness)** and **Soundness**.

What is the XSC standard

- The **Confidential Security Contract** is the standard for security tokenization.
- A replacement and optimisation to the current process for the **issuance**, **management** and **trading** of securities.
- A framework for removing middlemen from security issuance markets without losing regulatory compliance.
- The contract integrates the necessary steps for security trading into a unified protocol.

Issuance

- Enables the parties to issue securities with much less financial overhead than the traditional security markets.
- Allows for fractionalisation of securities.
- A distributed network of nodes enables to execute the contract logic consistently without hinging on a single point of failure.



Issuance

- The contract logic is straightforward, abstracted from the underlying protocol and enables the issuer to launch contracts without a need to understand the in-depth details of the technology underneath.

- The standard enables flexibility and provides the issuers with a featureset to fine-tune the contract to their needs.

Trading

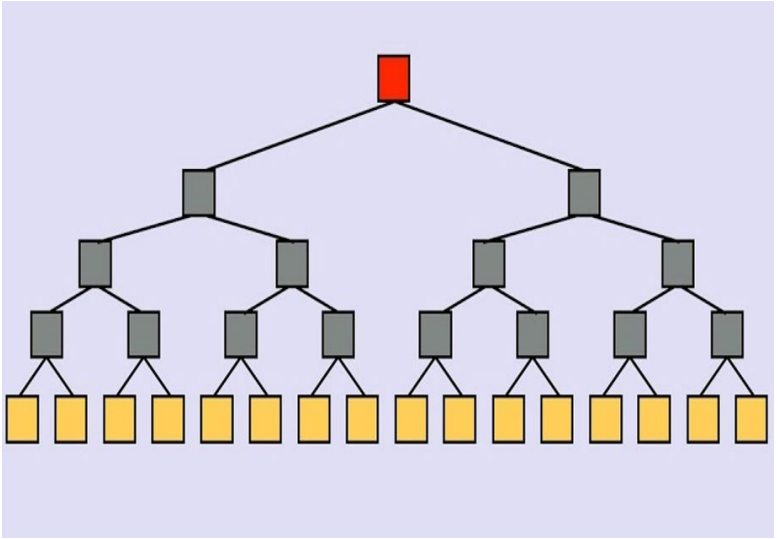
- The XSC enables near-instant anonymous security transfers. P2P.
- OTC can be flexibly integrated into the protocol through inter-contract communication.
- Tokenization of shares allows for the fractionalisation of securities.
- Trading can be performed on multiple regulated security exchanges.
- Smart contracts allow for programmable trading rules.

Management

- Smart contracts can been overseen directly by the Contract Owner or a third-party contracted to carry the aforementioned duties.
- Multiple contract user definitions enable smooth integration of regulatory bodies or auditors into the standard.
- Dividend payments can be made automated from smart contract payments, defined right from the start.

The Use of Zero-Knowledge Proofs I

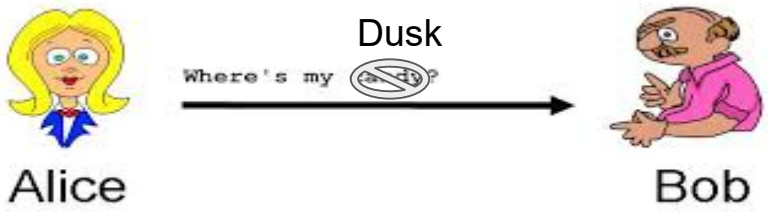
- Merkle trees are used to prove identity inclusion the whitelist via Poseidon hashing functions.
- Used as an accumulator for all possible whitelist members.
- Root verification functions are built into the blockchain.



The Use of Zero-Knowledge Proofs II

- With Peer to Peer, having varying amounts can result in a lack of balance in token ownership - as the contract owner sees fit.
- An arbitrary range proof allows for the prevention of surpassing company caps

Cap Example:

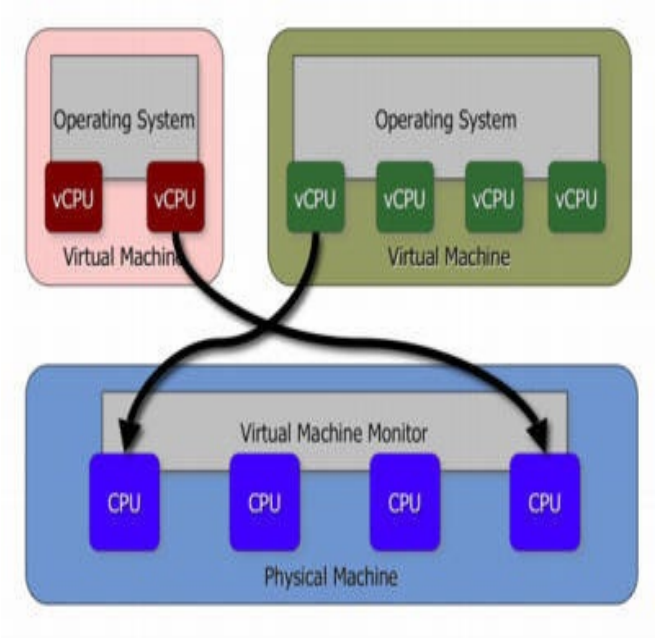


If Bob requests tokens from alice.
Bob will have to prove that his current holdings lie in the range:

$$0 \rightarrow (\text{Cap} - \text{Dusk} + 1)$$

Virtual Machine

- Turing-complete VM with zero-knowledge proof verification capability.
- Based on WebAssembly (WASM).
- With no computation done on the obfuscated values, the computational burden is shifted on the user, keeping the distributed state machine “lightweight”.



Distinct advantages

Feature		Owner	User	Description
1	Global financial ledger	✓	✓	• SSOT • Transparent transactions
2	Free market exposure	✓	✓	• Wider user base, including P2P's • Reduction in mispricing
3	Decentralisation	✓	✓	• Increased speed and security • Lower transaction fees
4	Smart contract application	✓	✓	• Administrative relief via automation • Removal of administration fees
5	Zero-knowledge proofs	✓	✓	• Regulatory compliance w/o forgoing privacy • GDPR

Questions?