securosys

# Tokenizing Real-World Assets

April 2019

Marcel Dasen

VP Engineering

Securosys SA

# Securosys' products protect the Swiss banking

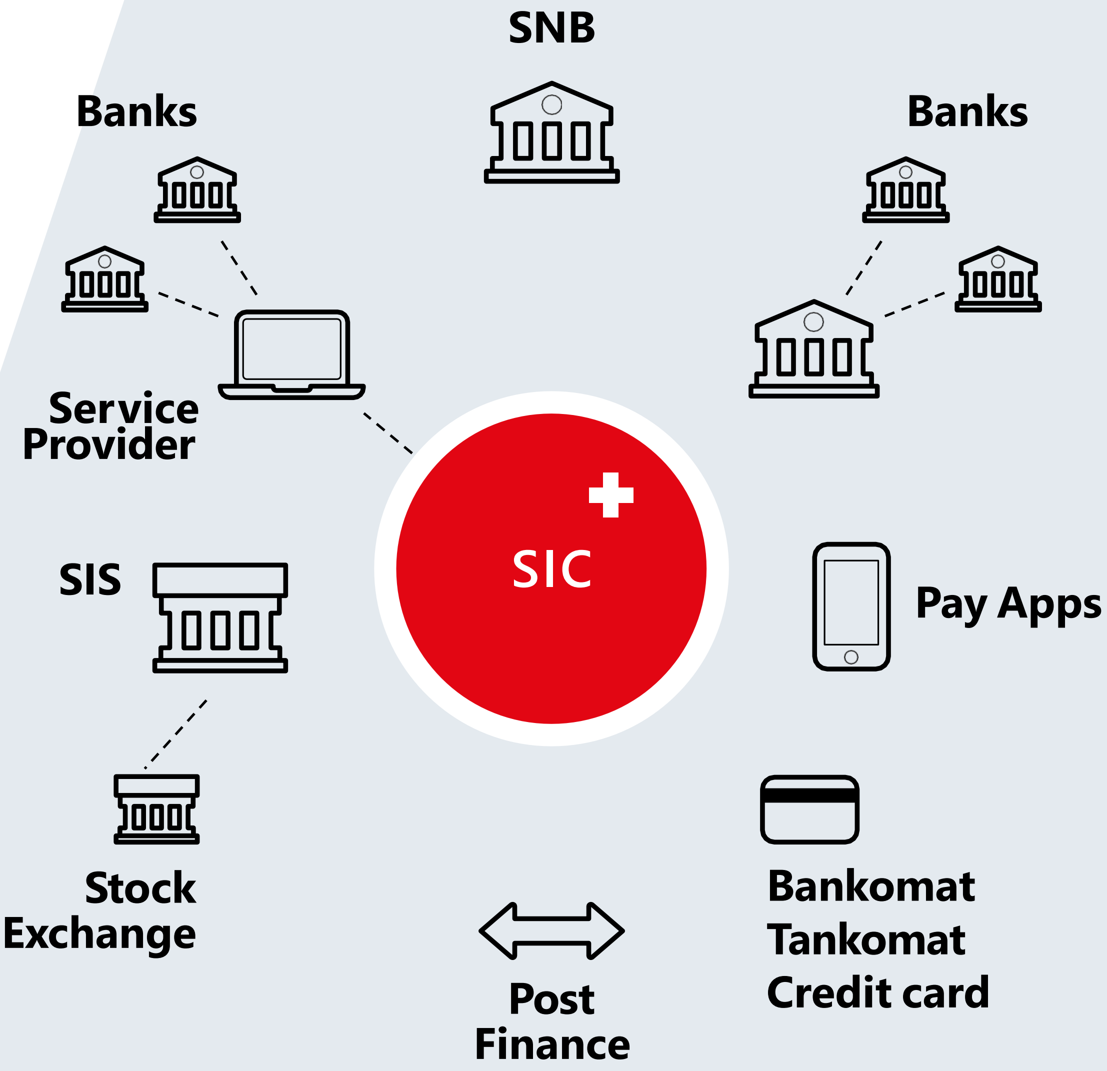
## Traditional HSM

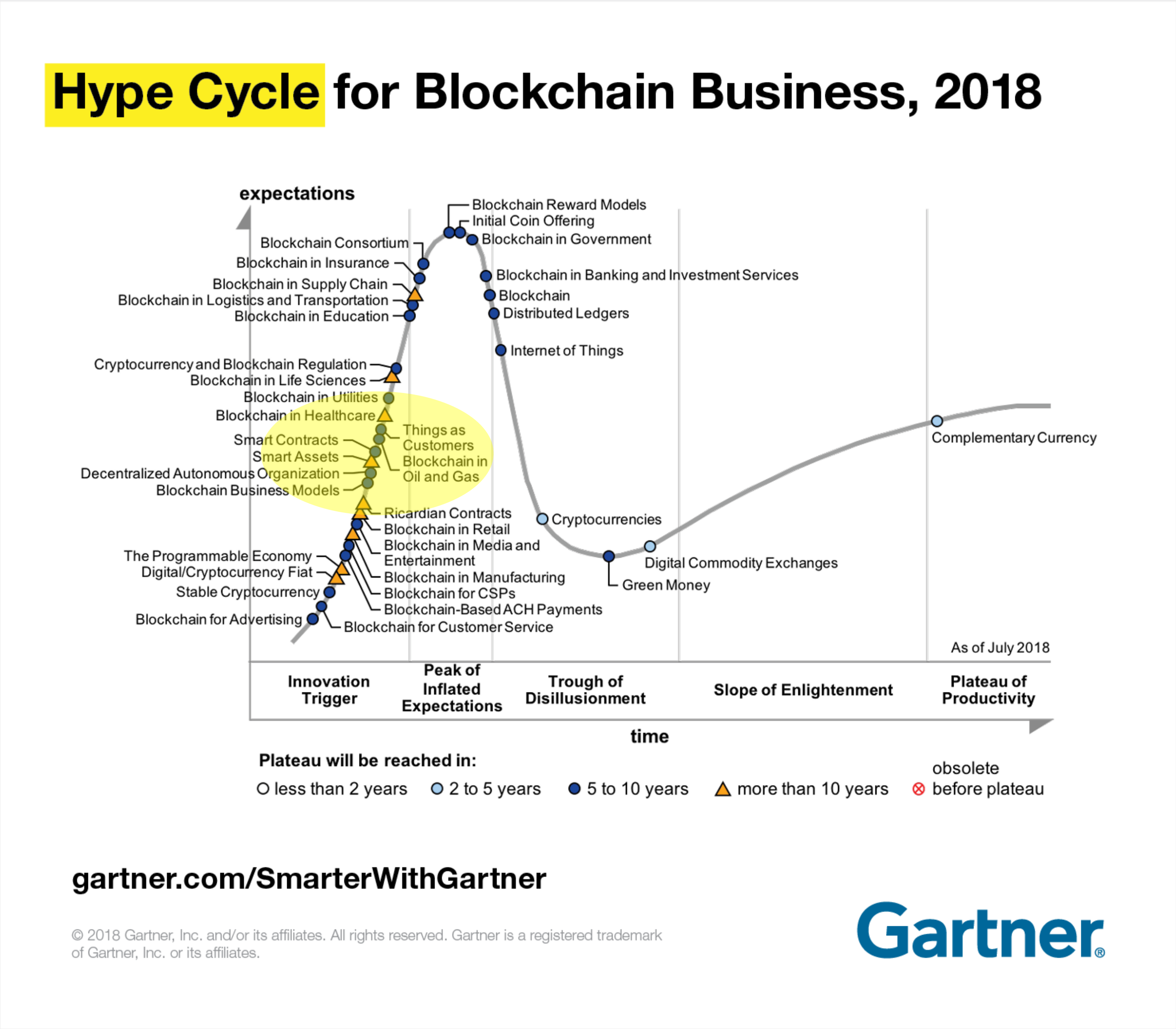
### SECUROSYS HSM PROTECT THE SWISS BANKING SYSTEM (SIC AND SECOM)

Over 100 Billion Swiss Francs per day

Up to 700 transactions per second

10 year maintenance & support agreement





## Hype Cycle for Blockchain Business, 2018

gartner.com/SmarterWithGartner

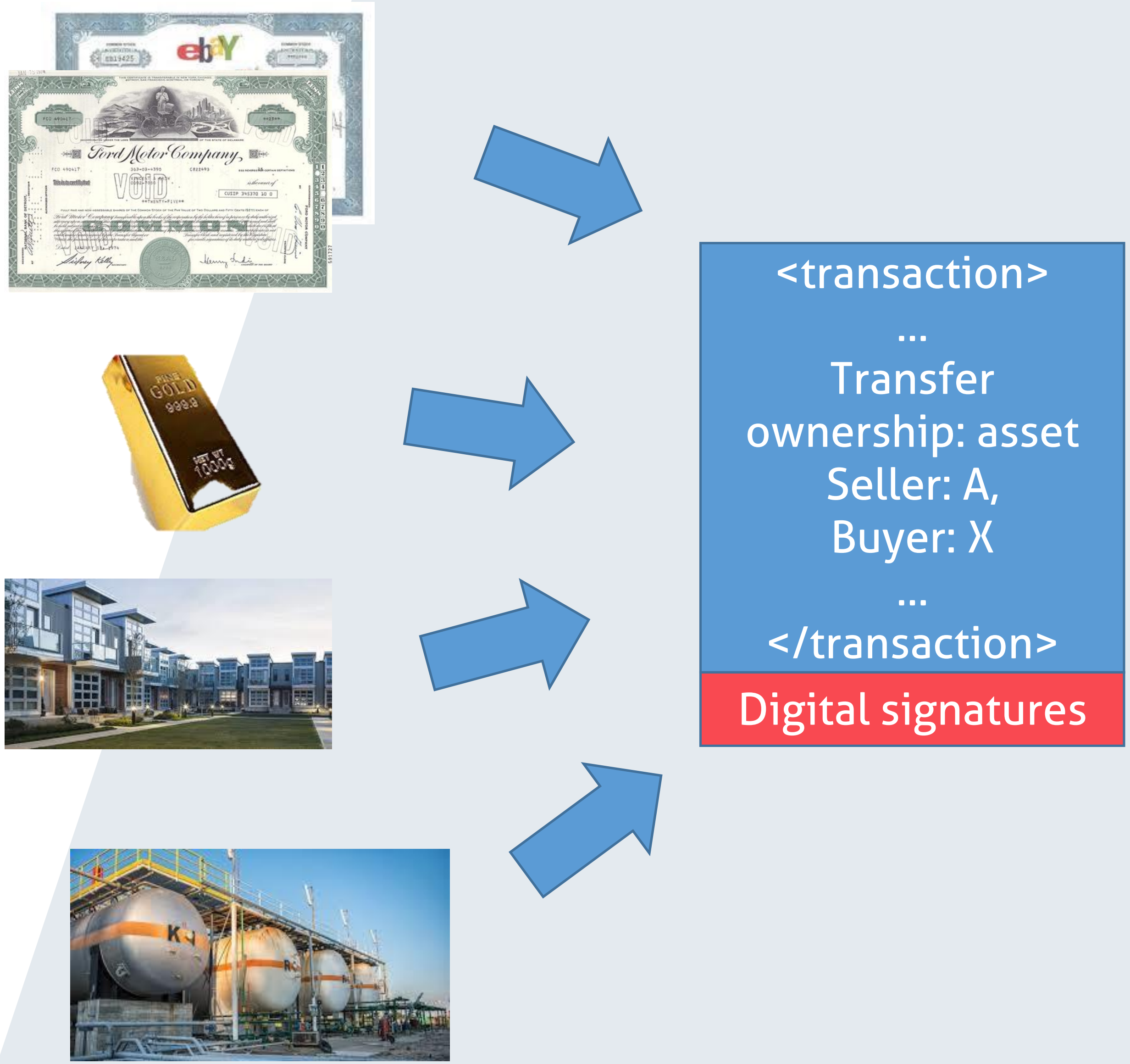
Gartner.

# Blockchain HSM

## SECUROSYS BLOCKCHAIN HSM PROTECT TOKENIZED ASSETS AND PERMISSIONED BLOCKCHAINS

# The tokenized asset

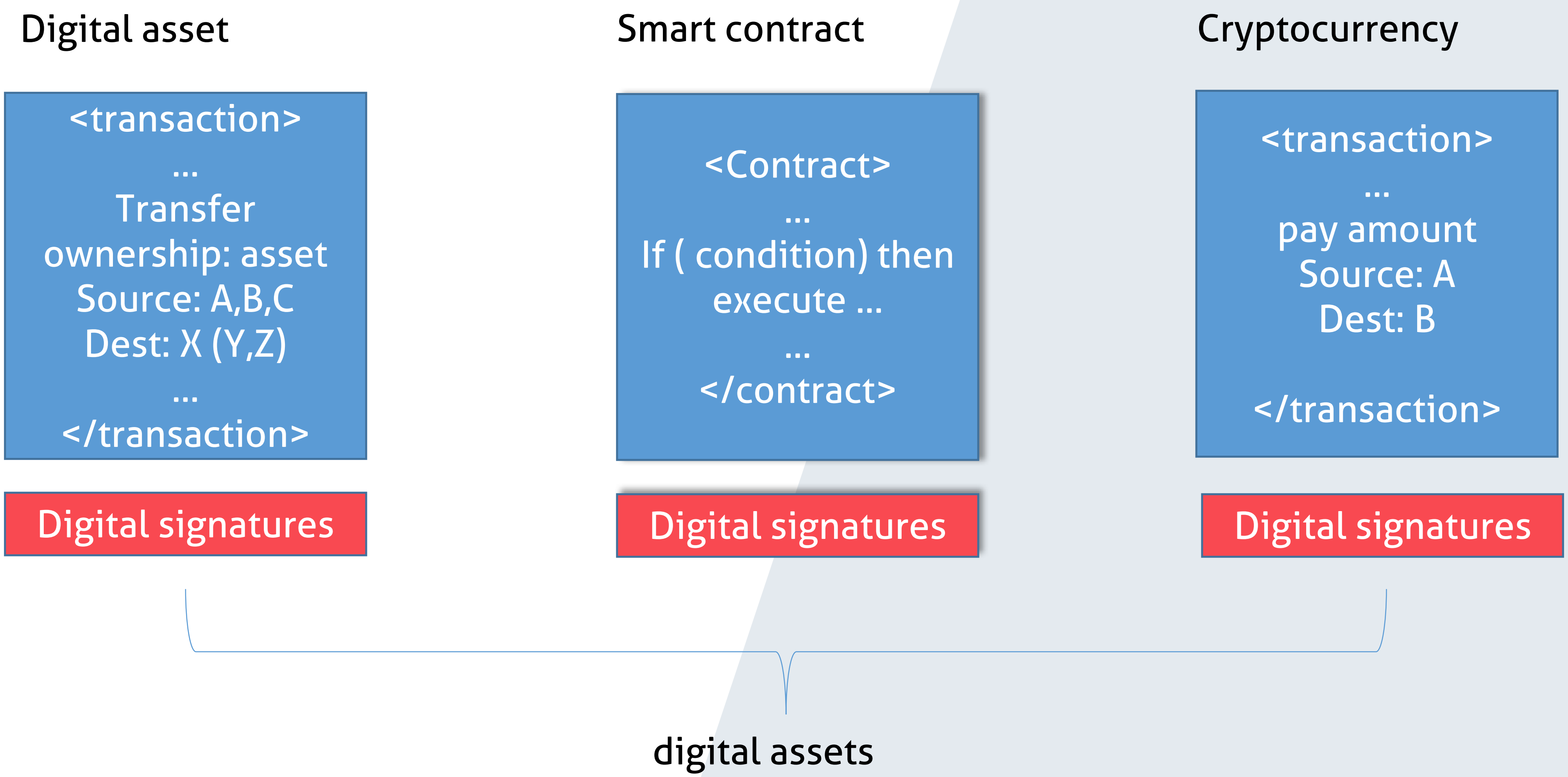
- A tokenized asset is a:
  - immutable
  - digital representation
- of a real asset



<transaction>
…
Transfer ownership: asset
Seller: A,
Buyer: X
…
</transaction>

Digital signatures

# Digital assets, smart contracts and crypto currencies



Digital asset

<transaction>
…
Transfer ownership: asset
Source: A,B,C
Dest: X (Y,Z)
…
</transaction>

Digital signatures

Smart contract

<Contract>
…
If ( condition) then execute …
…
</contract>

Digital signatures

Cryptocurrency

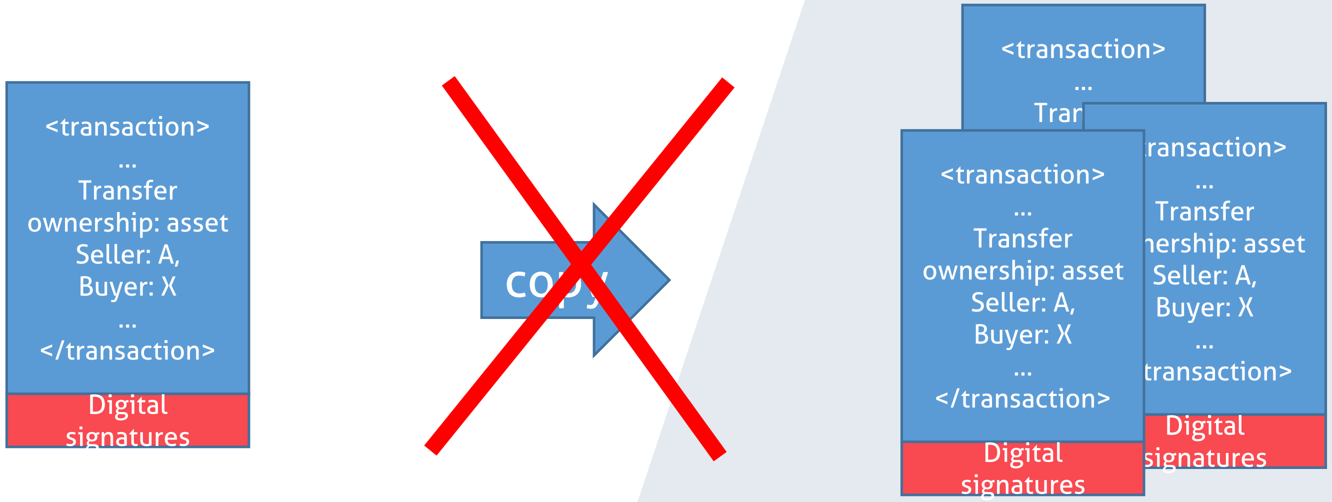
<transaction>
…
pay amount
Source: A
Dest: B

</transaction>

Digital signatures

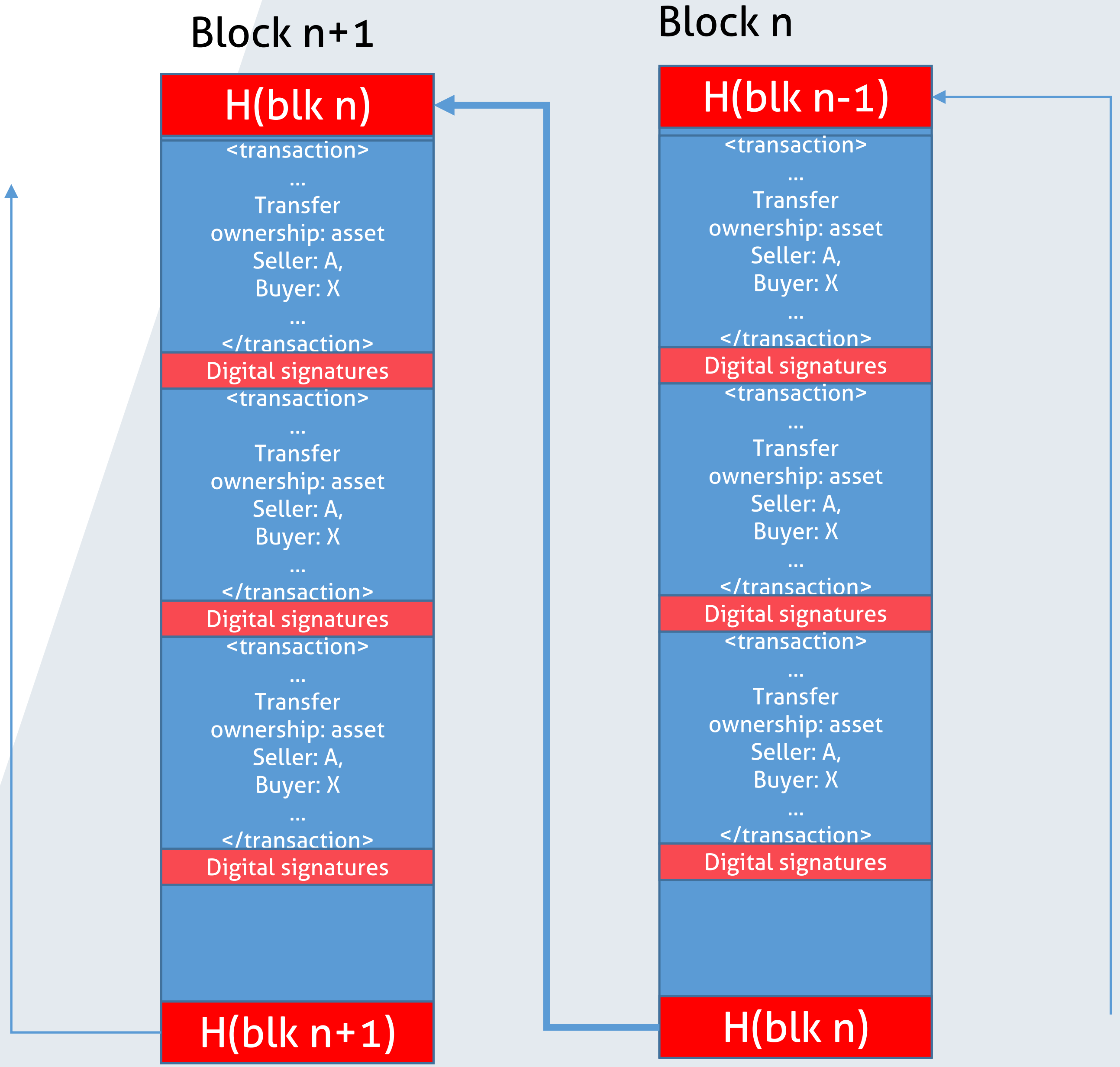
digital assets

# Problem: Copy protection

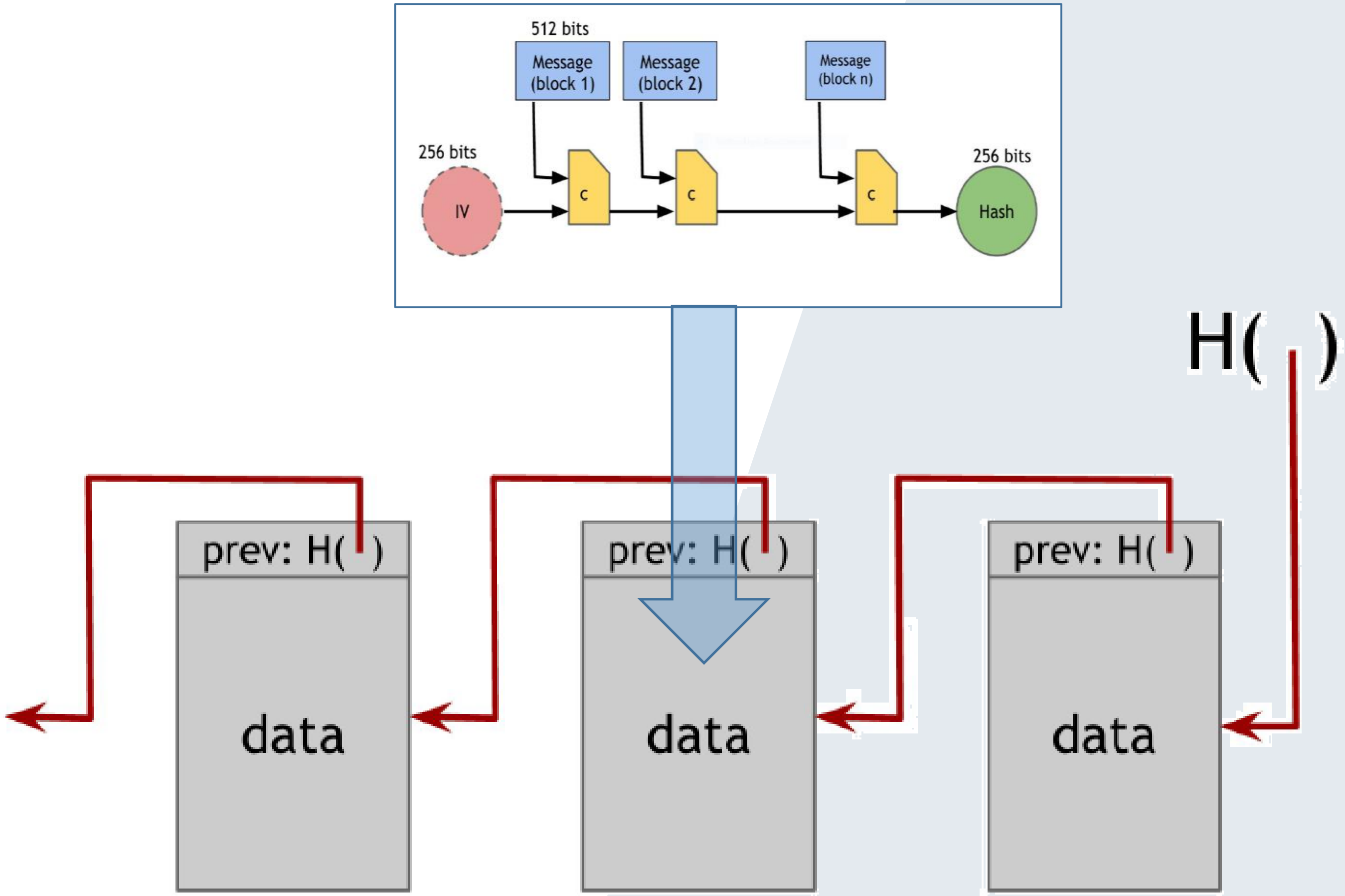


- Store on immutable data structure:
  => Blockchain

# Combine transaction in blocks:



# Storage of blocks in a chain



Through chaining of blocks with hash H() the blocks cannot be altered

# Problem: blockchains can be copied

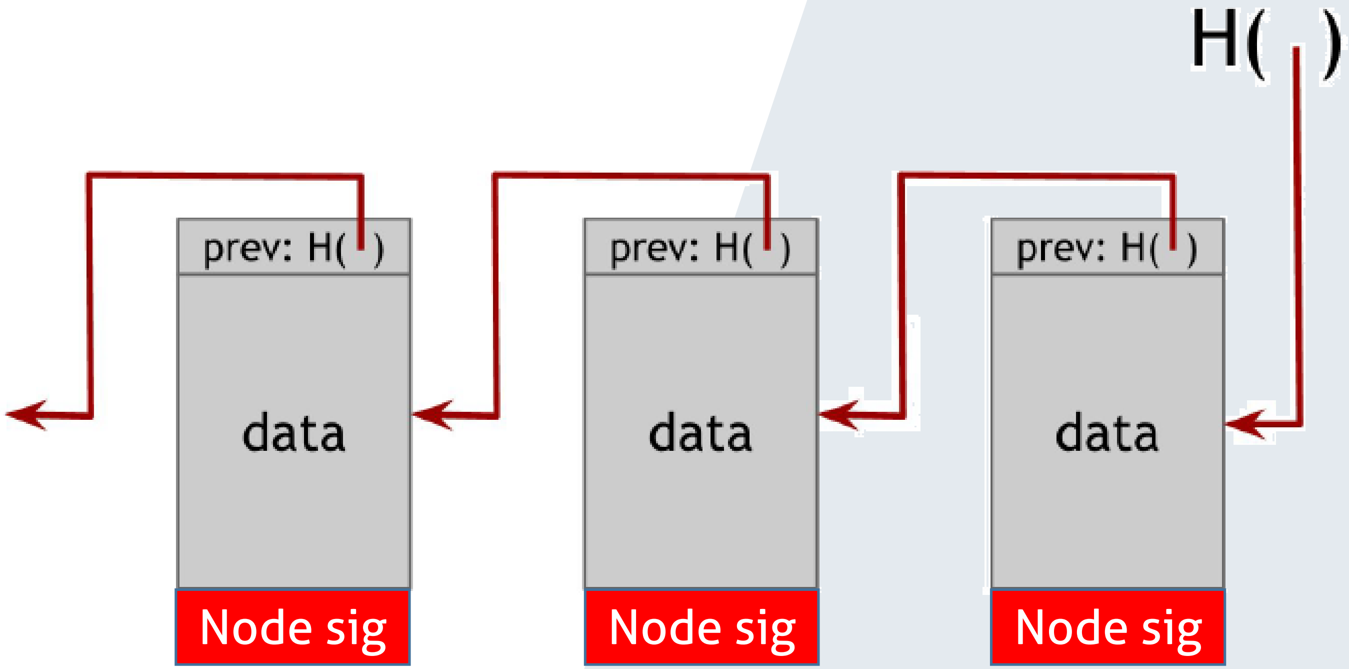
- **It’s a feature not a problem!**
  - Everybody can keep his own copy as proof
  - Multiple copies = redundancy = increased fault protection
  - Maintaining can be distributed (DLT)

**But there is a new problem: If copied chains are amended locally, which amendment is the “right” one?**

# Distributed ledger technology (DLT)

- Blocks are amended at multiple locations (nodes)
- A **consensus algorithm** guarantees that only one consensus state prevails
  - Permissioned (a distributed DB like algorithm, typically using a **digitally signed** state variable) – examples: Hyperledger, Corda, …
  - Proof of work The first to solve a puzzle – example: BTC
  - Proof of stake Proof that you are willing "to pay" – example
  - … various creative ideas …

# Example permissioned block chain



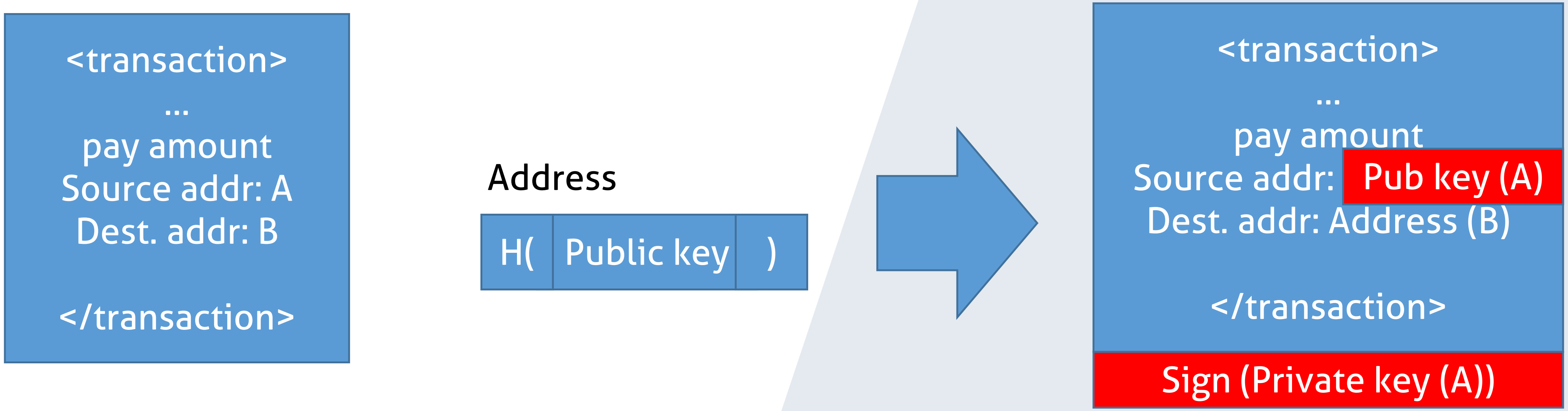
# Security of digital assets

- Storage “public” and unalterable on blockchain
- Blockchain can be copied; thus, system is reliable
- Consensus on transaction can be distributed
  - no central trusted authority needed (but possible)
- Transaction validation by digital signing
  - Need for reliable storage of private signature keys

# Why tokenizing real world assets

- Easier to process than physical goods
- Easier to transfer ownership
  - Clearing & Settlement
- Transparency: The history is on the blockchain
- No intermediaries or trusted 3rd parties needed for trades
  - but trust in algorithms
  - but trust in proper execution of algorithms (TEE or SEE)

# The transaction process



<transaction>
…
pay amount
Source addr: A
Dest. addr: B

</transaction>

Address

H( Public key )

<transaction>
…
pay amount
Source addr: Pub key (A)
Dest. addr: Address (B)

</transaction>

Sign (Private key (A))

# Transaction basics: Digital signatures

- Three methods required
  - Key generation method: (sk, pk) := generateKeys( keysize )
  - Sign method: sig := sign( sk , message )
  - Verify method: isValid := verify( pk , message , sig )

- Practical concerns
  - Keep sk secret
  - Use addresses (for PQC concern)

# Trust in algorithms: Trusted execution environment

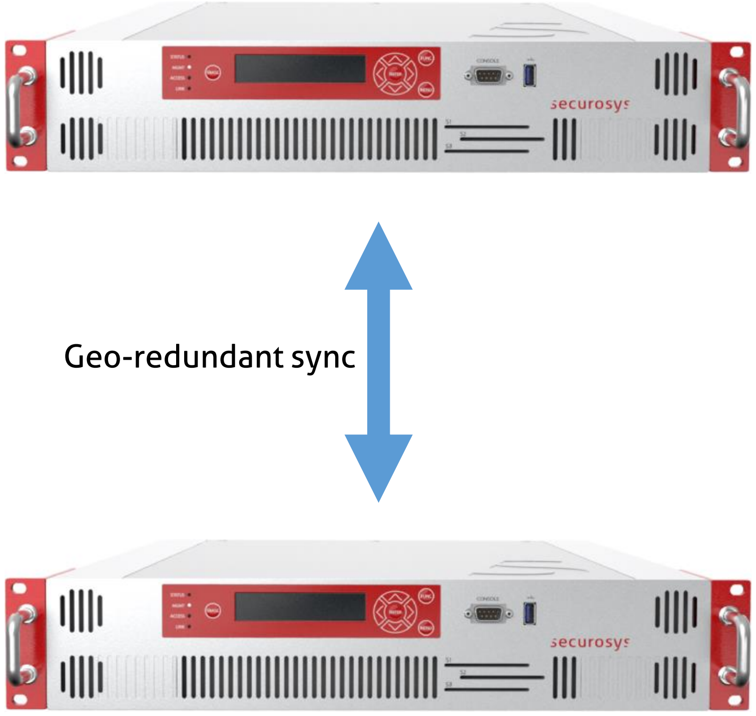
- Asserts the “validated code” is executed
  - Verifies the digital signature of the code
  - Asserts code integrity during execution
- Asserts the “validated transaction” is processed
- Returns a trustable result
  - Digitally signs the result

# Tokenization requires a wealth of digital keys

- Billions of asset keys
- Millions of user keys
- Ten thousands of TEE keys
- Thousands of node keys

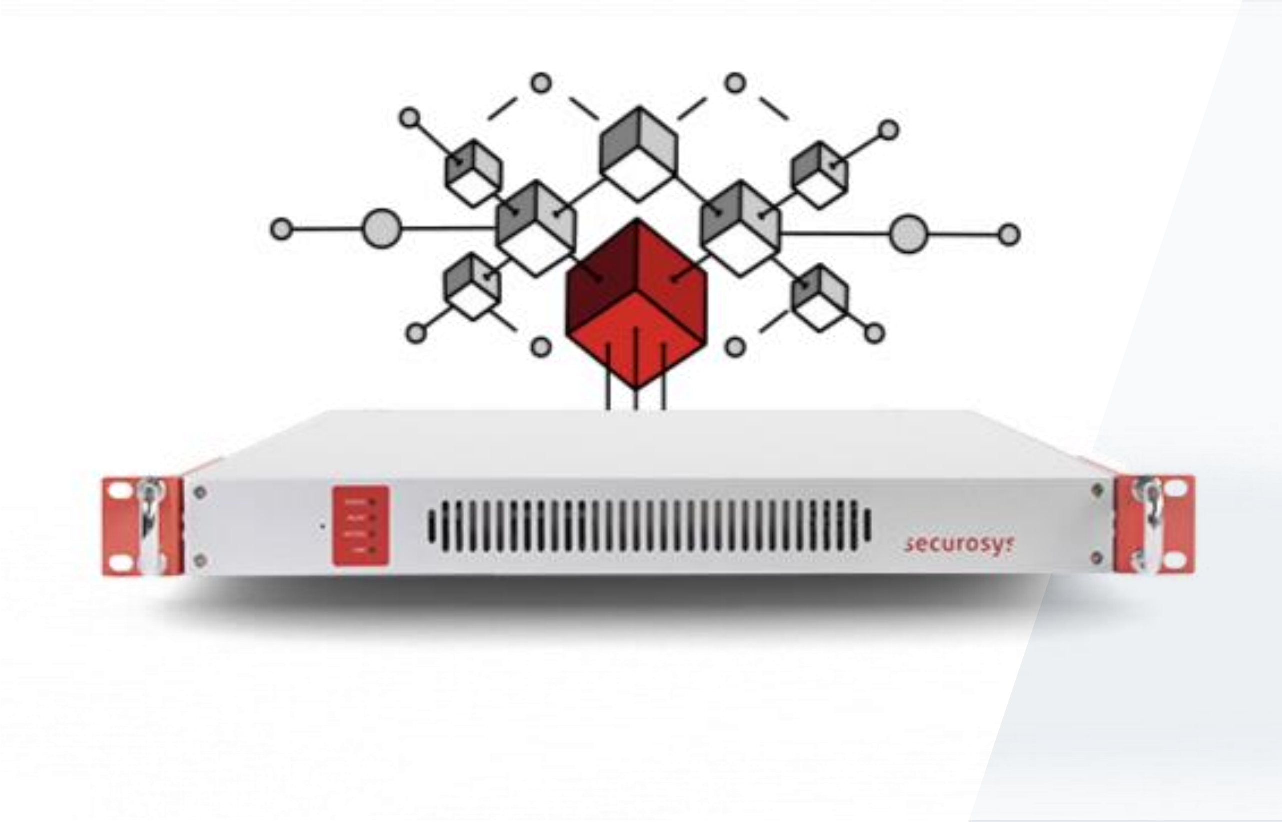
- Many of which have to be publicly trusted and thus must come from:
  - A trusted party!
  - Have to be stored (highly) secure
  - Have to be managed

# Primus HSM for storage of blockchain and asset keys



- PQC save addr generation Hash(pub key)
- SKA access control on keys
- n of m rule sets
- Redundancy & Reliability
- Algorithms (p256, ed25519, iota iss, ..)
- Tamper protection
- Secure key generation (TRNG)
- “Long”-term storage

# Trusted execution platform



- Planned

# Digital assets revolution

- Digital automation of transfer of ownership for any kind of asset
    - Fractional ownership
    - Tokenization of physical goods
    - Public registries
    - Proof of origin (certificate of origin)
- Security and management of private keys is key for block chain systems
    - Losing a key is losing your asset!
    - Control of access to your keys is control of your assets (vault)
- Putting trust in algorithms requires:
    - Trusted executors
    - Trusted input and output

# Your Contact

securosys

**Förrlibuckstrasse 70**
**8005 Zürich**
**Switzerland**

info@securosys.ch

www.securosys.ch
+41 44 552 31 00