



Deep into Blockchain Series

# Zero Knowledge Proof

Presenter(s): Dharmen Dhulla

Event Organizers

 **Centrum** Community  
Connect | Collaborate | Create

Venue Sponsor

**nagarro**

# Is Privacy a concern?



## Database Leak

- UK Medical Records
- Turkish Citizenship Database

## PII Leaks

- No Clear definition of “Relevant measures” and “security”
- Mostly Organizational measures, no encryption



# Blockchain and Privacy?

Blockchain: immutability, censorship-resistance, and open and permissionless

Bitcoin (Blockchain 1.0): pseudo-anonymity, public ledger, ... transactions

Ethereum (Blockchain 2.0): smart contracts, DApp ... data

Fabric (Blockchain 3.0): Privacy and channels, ... trust?

Do we need to continue transacting in the open to enjoy the benefits?

Mixers, ring signatures, partially homomorphic...

# What is Zero Knowledge Proof?



**Zero Knowledge Proof** is a method by which one party (the prover) can prove to another party (the verifier) that a given statement is true, without conveying any information apart from the fact that the statement is indeed true.

## Properties:

**Completeness:** If the statement is true the honest verifier (i.e the one following the protocol properly, will be convinced of that fact by an honest prover.

**Soundness:** If the statement is false, no cheating prover can convince the honest verifier that it is true, except with some small probability.

**Zero-Knowledge:** If the statement is true, no cheating verifier (or others) learns anything other than this fact. This is formalized by showing that every cheating verifier has some simulator that, given only the statement to be proved (and no access to the prover), can produce a transcript that “looks like” an interaction between the honest prover and the cheating verifier.



Let's Begin ...

Bob is color blind!





Let's Begin ...



Color 1





Let's Begin ...



Color 2







# Let's Begin ...

If Alice is cheating, then probability of fooling Bob :

$$\left(\frac{1}{2}\right)^n$$

For  $n = 15$  :

0.0000305





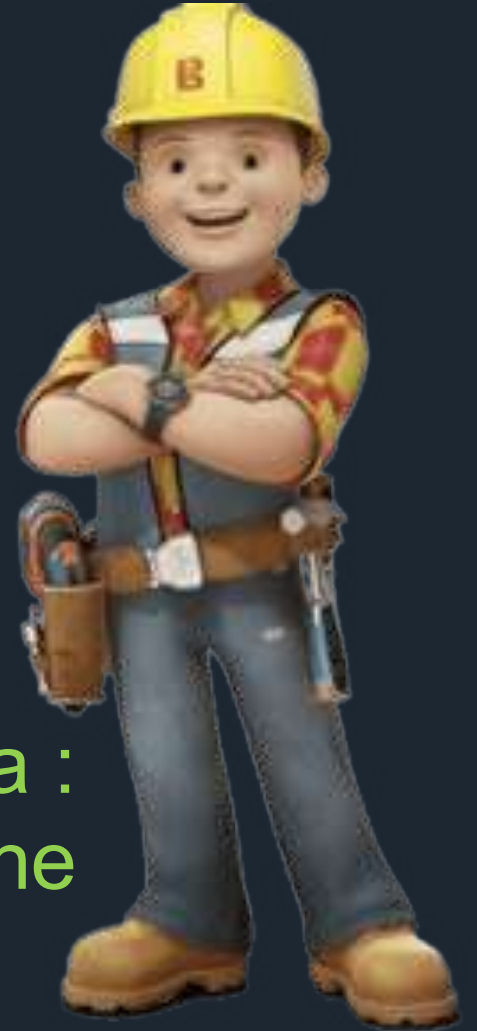


In the end ...

Alice Convinces Bob, that with very high probability :  
“She has **different** coloured hats.”



However, Bob has no idea :  
“What are the **colors** of the hats?”





# Zero Knowledge Proofs

- " A method by which one party: “**Prover**”, can convince another party: “**Verifier**”; about the “**truthfulness**” of a statement.
- " The protocol conveys nothing apart from the veracity of the statement.  

(Zero Knowledge)
- " A zero knowledge proof must satisfy :
  - " **Completeness** : Honest Prover can convince an honest Verifier.
  - " **Soundness** : Cheating Prover can convince an honest Verifier with negligible probability.
  - " **Zero Knowledge** : Verifier learns nothing apart from the veracity of the statement.



# Graph 3-Coloring (G3C)


No adjacent nodes  
have the same color.





## G3C : Problem ...

Given a Graph  $G$  with  $n$  vertices and  $m$  edges



$G$  is 3-colorable!  
I know the colouring



Really?  
**Prove it?**



## G3C : Protocol ...

**Alice:** Choose random assignment of 3 colors and color the graph, locks them in a box and sends to Bob

**Bob:** Chooses an edge (e) randomly from the graph

Request keys for the 2 boxes corresponding to e



**Bob:** If the colors in the boxes are different, colouring of the edge is correct.





## G3C : Protocol ...

**Alice:** Choose random assignment of 3 colors and color the graph

**Bob:** Chooses an edge (e) randomly from the graph

If these two colors are different:  
“The edge chosen was correctly  
coloured”

Repeat the same experiment again!





G3C: Repeat! How many times ?

If Alice is cheating, then probability of fooling Bob :

$$\left( \frac{m-1}{m} \right)^{m^2}$$

For  $m = 7$  :

0.000524





## G3C: Is it a Zero Knowledge Proof?

It can be shown that the protocol satisfies:

- " **Completeness:** Honest Prover can convince an honest Verifier.
- " **Soundness:** Cheating Prover can convince an honest Verifier with negligible probability.
- " **Zero Knowledge:** Verifier learns nothing apart from the veracity of the statement.



# Application of ZKP in Blockchains?

- " The proofs seen so far are **interactive**
- " Not suitable for blockchain applications. **Why?**
- " Desirable properties from a ZKP variant to be usable in blockchain:
  - " Non-interactiveness
  - " Small in size
  - " Fast verification



# Non Interactive Zero Knowledge Proofs

- " No interaction required between the Prover and the Verifier
- " Impossible in standard cryptographic model  
[Goldreich and Oren; 1993]
- " Possible in **common reference string** and **random oracle model**.
- " Common reference string can yield computational zero knowledge.



## zk-SNARKs

- " No interaction with Prover required
- " A proof once generated can be verified by anyone
- " To establish this, we need to establish some shared "reference string"
- " This is done during Setup Phase...generates toxic waste
- " For every function (circuit) we need to run the Setup Phase



# Zcash

- " Privacy preserving crypto currency based on zk-SNARKs
- " A transaction contains: sender(input txns), amount, reciever(output txn)
- " Publish proof that a private transaction follows the rules of the Zcash network, concealing the sender, recipient, and amount (shielded transactions)
- " Downside? Trusted Setup (Zcash Trusted Setup Ceremony)



## Zcash: Destroying toxic waste...



The burnt remains  
of one machine  
involved in Zcash's  
trusted setup

Pic Credit : Peter Todd



# Confidentiality Mechanism in Hyperledger Fabric

- Channels
- Private Transactions
- Zero-Knowledge Proof-based Technologies
  1. Anonymous Client Authentication with Identity Mixer
  2. ZK-AT (Zero Knowledge Asset Transfer)





# Identity Mixer for HLF: Executive Summary

- Identity Mixer is a Strong Privacy-Preserving Authentication solution
  - Better privacy than standard X.509 or OpenID
  - Protocols are verified by the scientific community
- Perfect fit for the Blockchain scenario & requirements
  - Better scalability, simplicity, privacy, security, auditability
  - Use cases: privacy-preserving asset transfer, banking, trading shares, KYC.
  - GDPR compliance
- A differentiator for HL Fabric (advanced privacy features)
  - Basis for Privacy-preserving asset transfer
  - Privacy-preserving & efficient hierarchical issuance of certificates is also possible (paper at CCS'17)

# Identity Mixer

- Attribute-based credentials
- Strong authentication (signatures)
- Privacy-preserving Access Control
  - Selective disclosure of attributes, predicates over attributes, full unlinkability
- Auditability
- Revocation



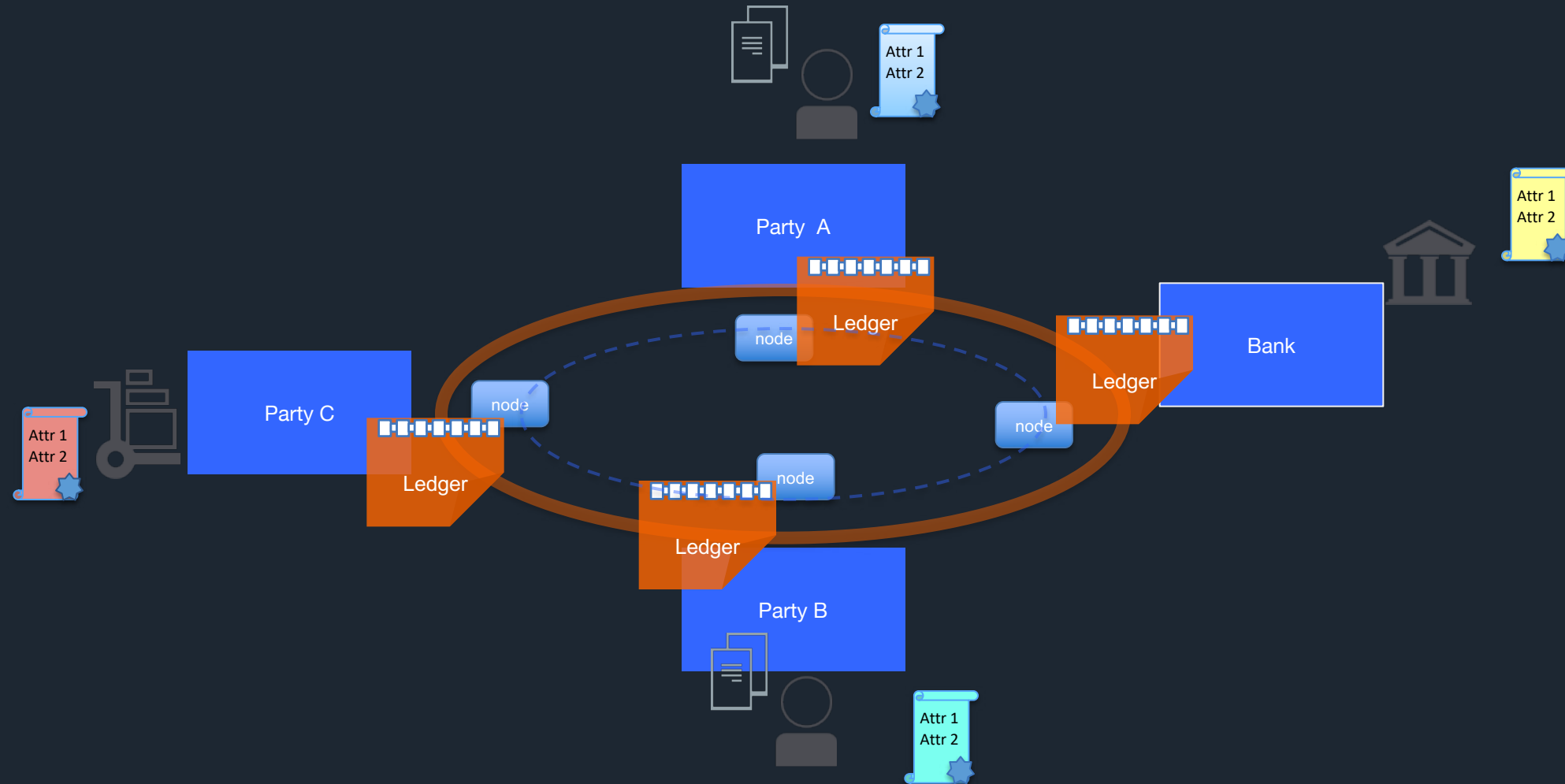
- Preserving privacy and unlinkability



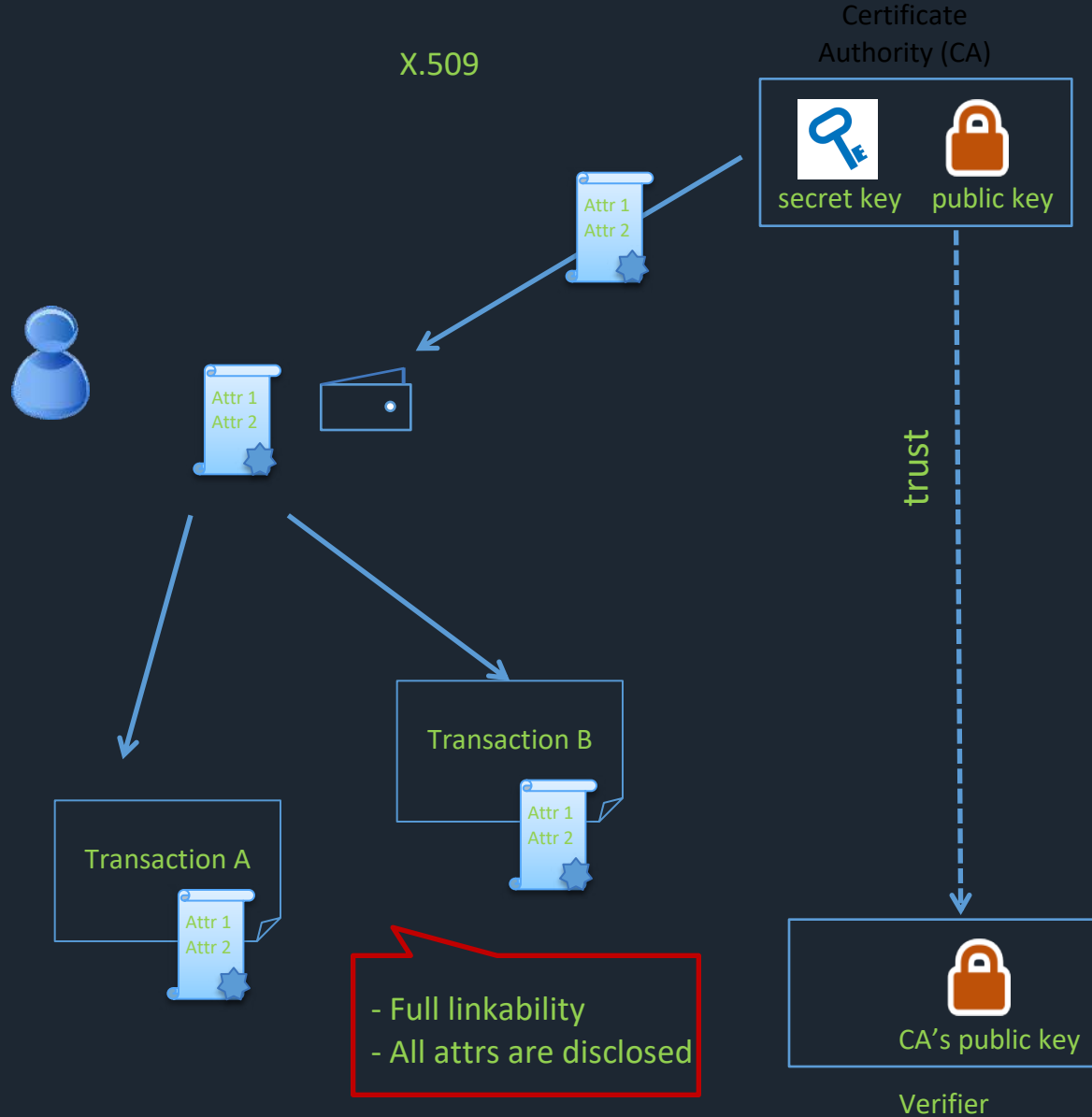
- Verification is done with the public key of the issuer only



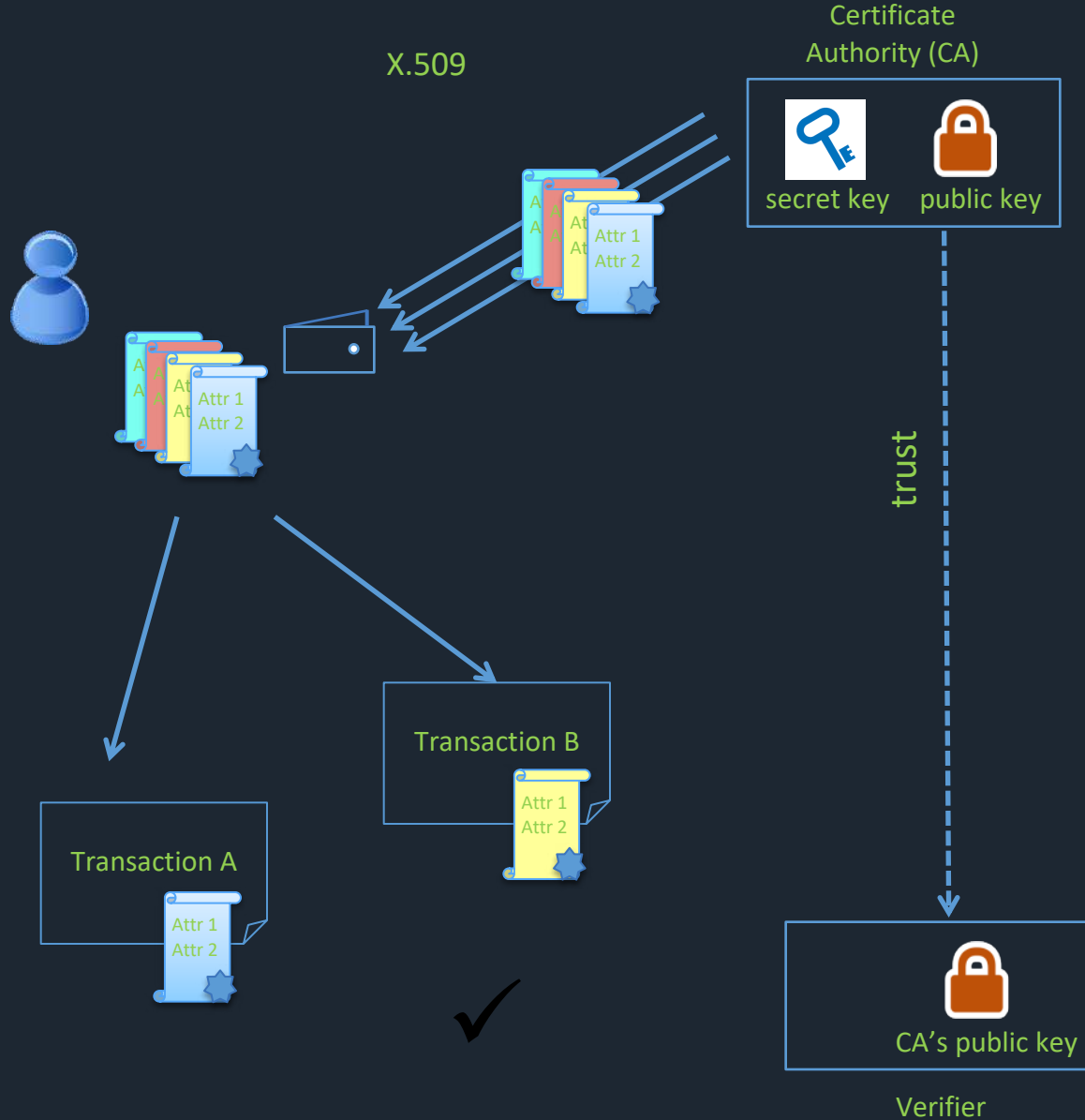
# Permissioned Blockchain



# Signing transactions with a single X.509 TCert



# Multiple X.509 Certs

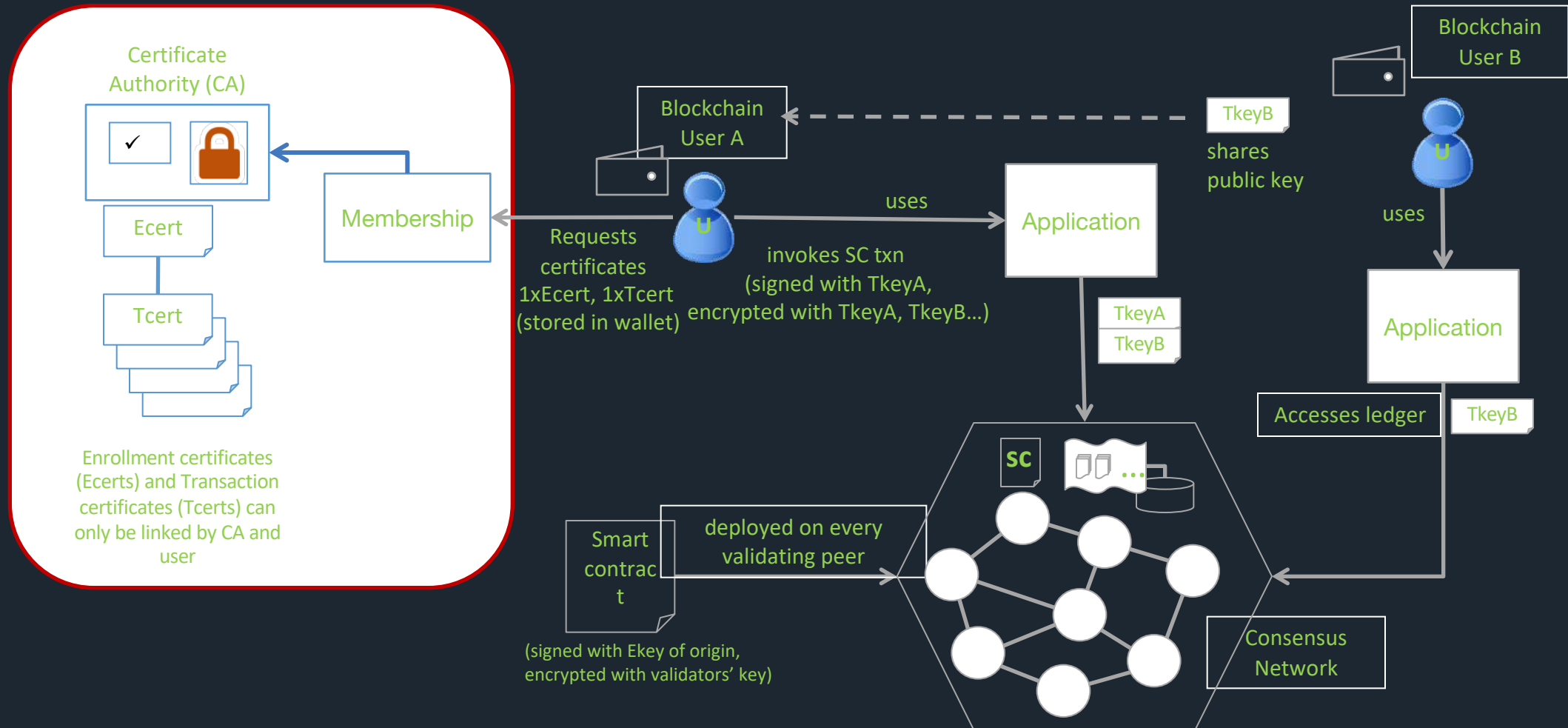


# Membership management

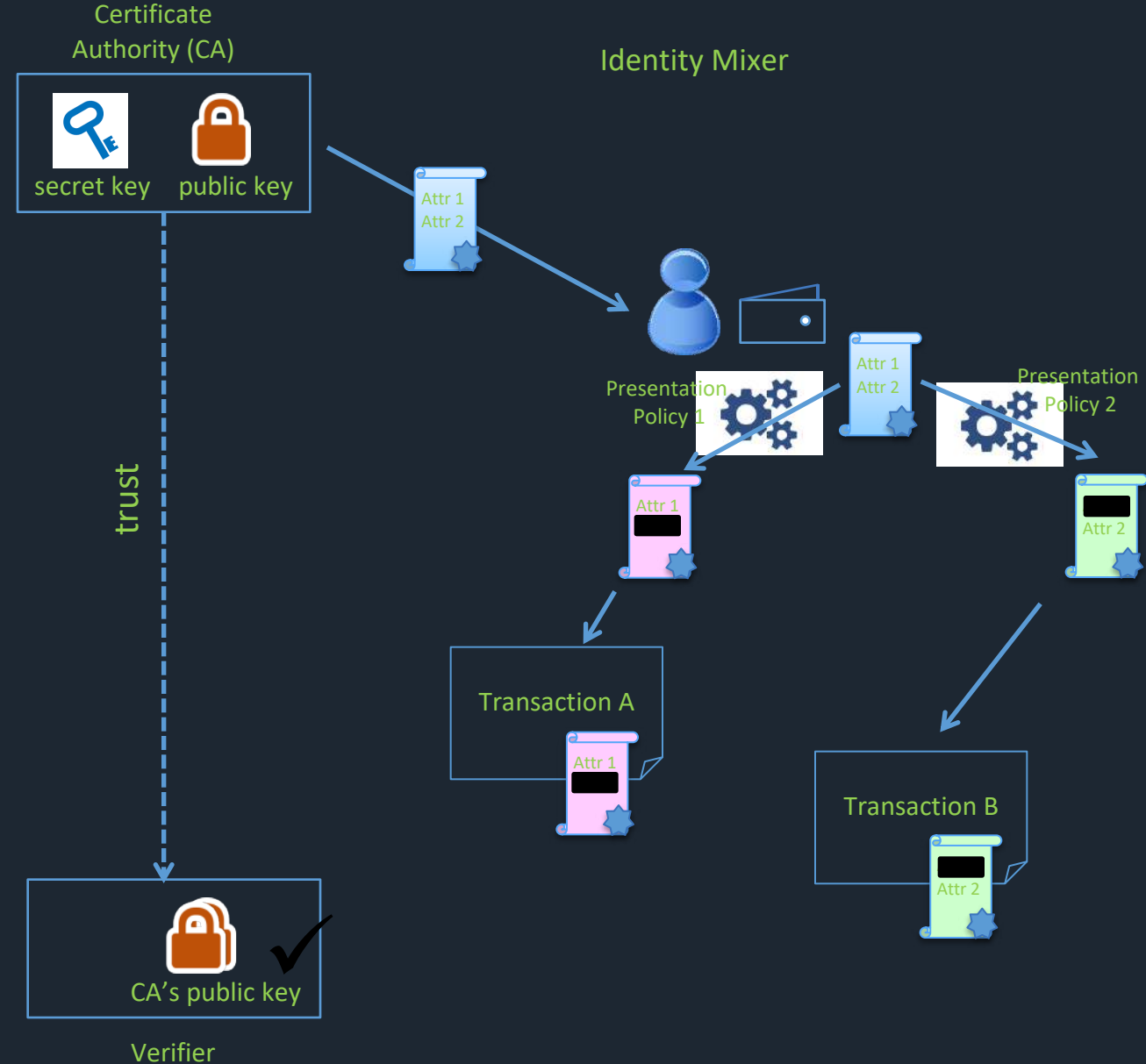


ECerts: (relatively) static enrollment certificates acquired via registration with an enrollment certificate authority (CA).

TCerts: transaction certificates that faithfully but pseudonymously represent enrolled users, acquired via a transaction CA.

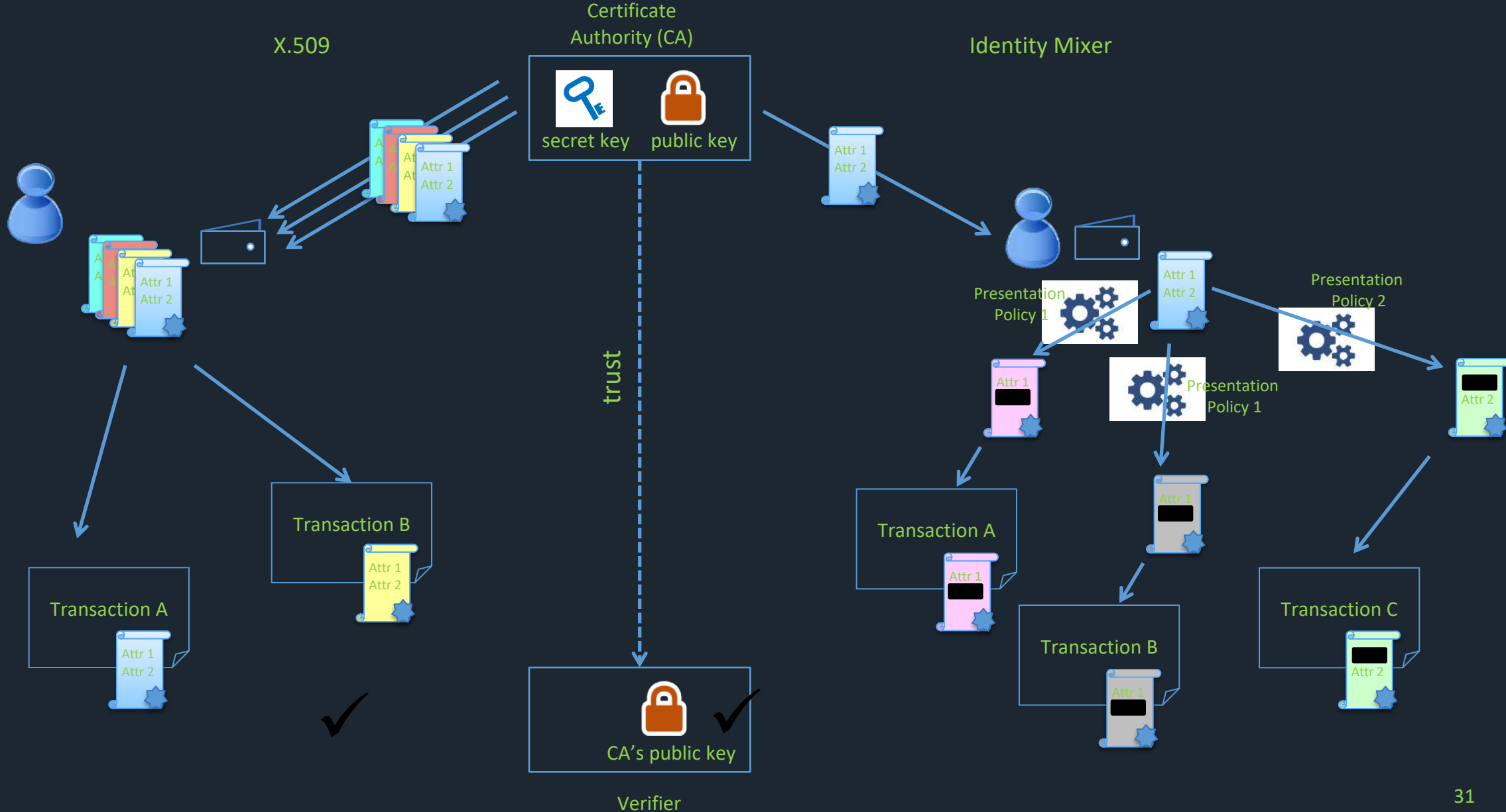


# How Identity Mixer works





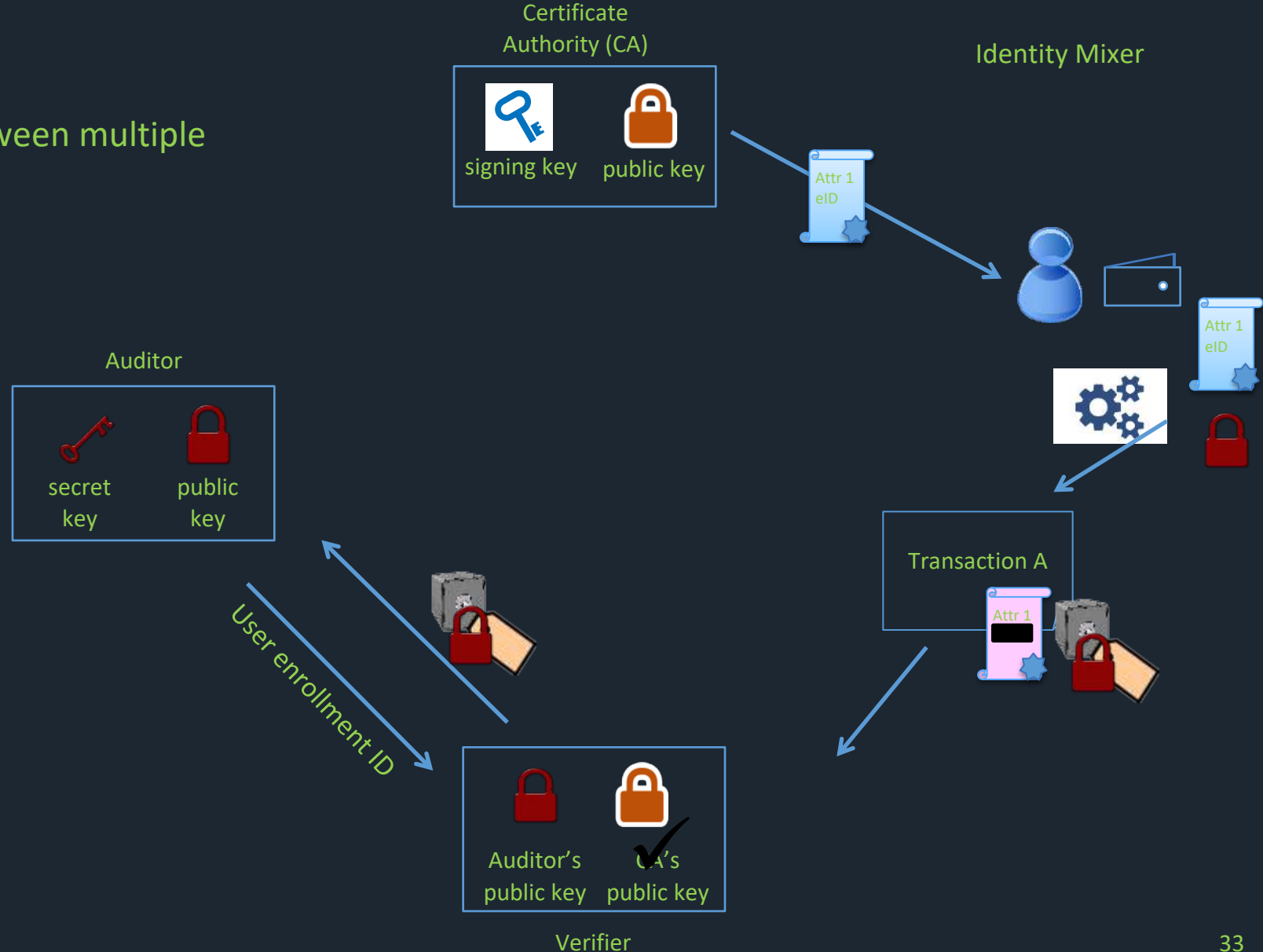
# Identity Mixer vs. multiple X.509 TCerts



Security & Privacy features	Hyperledger v1.0	Hyperledger + TCerts	Hyperledger + Idemix
User Anonymity	-	++	++
Transaction Security	++	++	++
Transaction Confidentiality	-	++	++
Accountability	++	++	++
Access Control	+	+ (only attribute disclosure)	++ (selective disclosure, predicates)
Auditability	++ (but without privacy)	+ (TCA have to participate)	++ (TCA is not involved in the audit)
Unlinkability	-	+ (TCA can link all transactions)	++ (TCA cannot link transactions, only auditors)
Simple Key Management	++ (but without privacy)	- (key derivation is required)	++ (single secret key on the user side)
TCA, Multiple TCAs	N/A	- (TCA is a bottleneck to request fresh Tcerts, multiple TCAs is a problem)	++ (only one ECert, TCA cannot link transactions, multiple TCAs is not a problem)
Solution Simplicity	++ (but without privacy)	-	++
Storage Efficiency	++ (but without privacy)	- (TCerts and keys need to be stored)	++ (only one ECert)
HSM & CSP support	++ (but without privacy)	- (interface changes required to implement key derivation)	+ (only custom implementation of the signing algorithms, no interface / flow changes)
Revocation	+	+ (only ECert? Privacy-preserving revocation of TCerts?)	++ (privacy-preserving revocation of Ecerts)

# Auditability (Inspection)

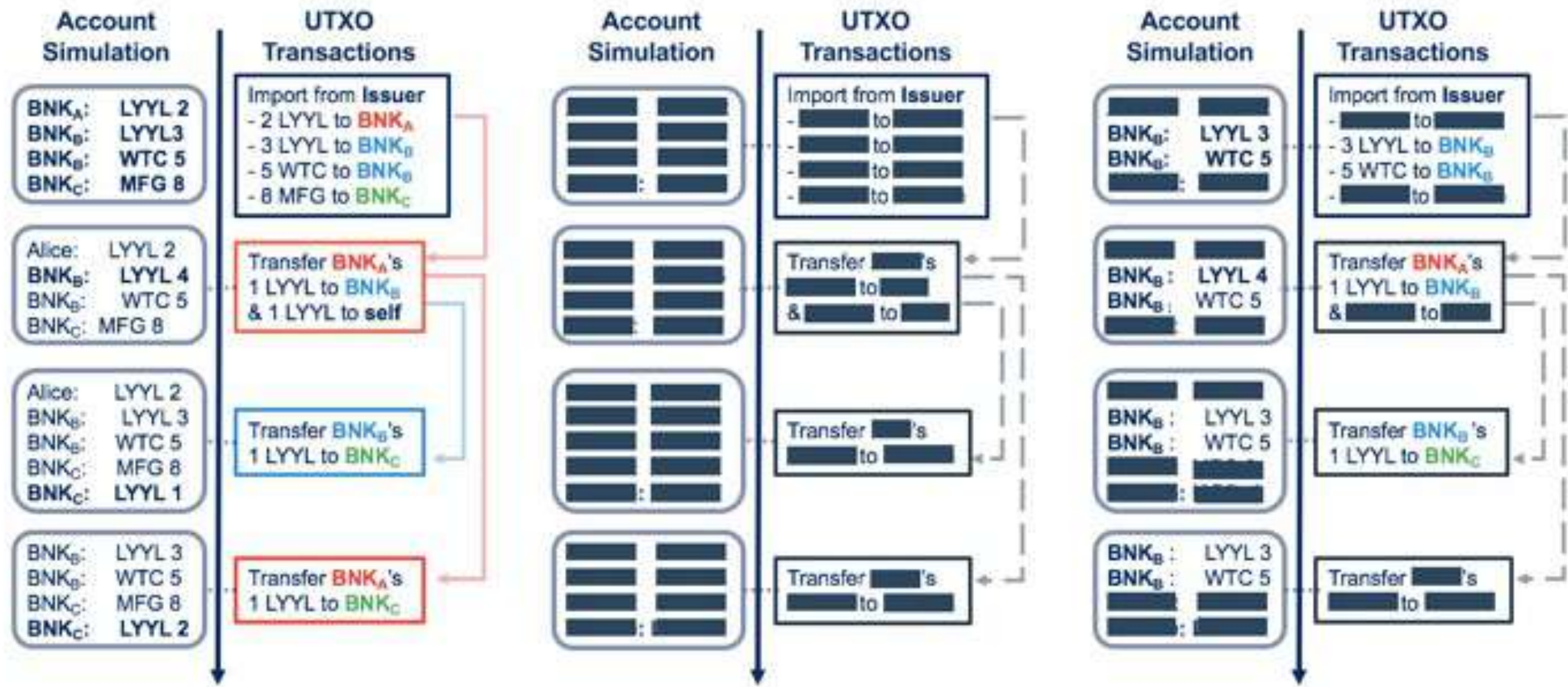
- Only Auditor can track the transactions
- Auditor's secret key can be shared between multiple parties to distribute the trust





## Hyperledger Fabric (ZKAT)

- " Integrate ZKP into a wider range of applications targeting asset management
- " Use of ZKP for privacy-preserving asset management with audit support
- " ZKAT is built on top of anonymous authentication mechanisms offered by Identity Mixer.
- " Asset Management: issue asset, request transfer of their assets in zero knowledge
- " Support Audibility: By assigning auditors to organizations, giving them unrestricted access.







Supported by

**nagarro**



**bradsol**  
SIMPLIFYING BUSINESS

