

v3.4-TKN-003 – ZK-Secured Capability Exchange & Sovereign Token Validation

Document Title	ZK-Secured Capability Exchange & Sovereign Token Validation
Version	v3.4
Document ID	v3.4-TKN-003
Date	2025-03-22
Author	Take Back Your Data – Identity & Capability Group
Document Type	Public / Certification / Internal

1. Purpose & Scope

This document defines the zero-knowledge-secured logic for capability exchange and token validation in MaxOneOpen. It enables forks to verify permissions and enforce sovereignty rules without revealing internal execution traces.

2. Capability Exchange Logic

- All tokens must allow ZK-sealed validation of capability without exposing contents
- Twin-level exchanges must occur in isolated, ephemeral message chains
- Each capability must carry proof of scope, origin, and audit-bound intent
- Forks must support exchange trace replay and compliance audit trails

3. Validation Architecture

Validation Element	Verification Scope	ZK Proof Anchor
Token Origin Match	Signature and issuance path	Ledger + twin log
Capability Trace Link	Schema-aligned execution	Trace hash replay
Role Scope Match	Permitted action boundary	Policy fingerprint
Abuse Detection	Excess use or bypass	Drift delta + execution log

4. Certification Hooks

- All capability validation must use ZK-proof based execution with sealed verification anchors
- Forks must be able to replay exchanges and expose policy-aligned decision trees
- Capability overreach or mismatch must trigger rejection or suspension

5. Certification Triggers

- Plaintext capability checks or runtime evaluation without ZK logic disqualify fork
- Failure to detect abuse patterns or scope violations invalidates certification

6. Certification Relevance

MaxOneOpen-certified forks must secure all token capability exchanges and role validations using zero-knowledge mechanisms. Only forks with traceable, sealed and policy-bound execution qualify for sovereign certification.