

v3.4-OPS-002 – Policy-Driven Execution, Operator Roles & Override Mechanics

Document Title	Policy-Driven Execution, Operator Roles & Override Mechanics
Version	v3.4
Document ID	v3.4-OPS-002
Date	2025-03-22
Author	Take Back Your Data – Autonomous Operations Unit
Document Type	Public / Certification / Internal

1. Purpose & Scope

This document defines the policy execution model, operator roles and override mechanics in MaxOneOpen. It ensures that any manual interventions are traceable, role-bound and policy-compliant, while maintaining operational autonomy.

2. Policy Execution Model

- Policies must be schema-bound, signed, and audit-traceable
- Execution logic must derive from sealed policy snapshots
- Forks must reject any execution without matching policy scope
- Emergency and exception logic must be predefined and limited

3. Operator Role Model & Override Logic

Operator Role	Function Scope	Override Mechanism
Policy Steward	Policy authoring + trace review	ZK-sealed schema path
Ops Auditor	Lifecycle replay, anomaly tracking	Snapshot seal + token proof
Twin Maintainer	Manual trigger during fallback	Emergency override token
SAC Delegate	Governance signal escalation	Policy override audit link

4. Certification Hooks

- All override logic must be traceable to schema-bound, cryptographic policies
- Forks must expose operator logs and show replayable execution sequences
- Emergency paths must terminate in certified fallback policy

5. Certification Triggers

- Untraceable overrides or non-signed policies disqualify fork
- Operator abuse or undocumented interventions invalidate certification

6. Certification Relevance

Certified forks must separate policy execution from runtime logic and expose operator roles via schema-defined override mechanics. All manual interventions must be cryptographically anchored and auditable.