Technical Note

SOMA-Bench: An Open Synthetic Benchmark and Evaluation Harness for Risk-Aware Recovery & Machine Identities in Post-Quantum IAM

Sravanakumar Nidamanooru *

Posted Date: 8 October 2025

doi: 10.20944/preprints202510.0344.v1

Keywords: identity and access management (IAM); post-quantum cryptography (PQC); artificial intelligence (AI); risk-based authentication (RBA); crypto agility; hybrid cryptography; account recovery; machine identities; credential rotation; risk-based authentication; evaluation harness; fraud detection; legitimate friction; time-to-innocence (TTI); migration health (%C/%H/%Q); drift robustness

Technical Note

SOMA-Bench: An Open Synthetic Benchmark and Evaluation Harness for Risk-Aware Recovery & Machine Identities in Post-Quantum IAM

Sravanakumar Nidamanooru

Independent Researcher (Identity & Access Management), USA; sravana.nidamanooru@gmail.com

Abstract

Identity and access management (IAM) systems are entering a difficult transition: recovery flows remain attacker-favored, machine identities rotate at scale, and post-quantum cryptography (PQC) introduces larger artifacts and new latency envelopes. Teams need a repeatable way to quantify fraud-versus-friction trade-offs and rollout safety during crypto-agile migrations—without exposing proprietary scoring models. This paper proposes a public, **synthetic benchmark and evaluation harness** for IAM recovery, sign-in, and credential rotation under PQC-aware conditions. The benchmark contributes (i) event schemas and a configurable generator with knobs for fraud prevalence, distribution drift, and signal dropout; (ii) a PQC "overlay" that models payload sizes and processing overhead for issuance/verification; (iii) simple baseline policies (static MFA, trivial risk); and (iv) reproducible metrics, including fraud blocked (%), legitimate friction (%), p95 decision latency, time-to-innocence, rotation SLO pass rate, and migration health (%C/%H/%Q). We report baseline results and stress tests and release code and documentation to enable independent replication and extensions. This work is the first step in a broader research agenda on **SOMA**, a risk-aware orchestrator for recovery and machine identities; system internals remain out of scope here and will be detailed in subsequent publications. (A **patent application is pending** on SOMA's underlying mechanisms; the benchmark is designed to remain IP-safe while still supporting rigorous comparison.)

Keywords: identity and access management (IAM); post-quantum cryptography (PQC); artificial intelligence (AI); risk-based authentication (RBA); crypto agility; hybrid cryptography; account recovery; machine identities; credential rotation; risk-based authentication; evaluation harness; fraud detection; legitimate friction; time-to-innocence (TTI); migration health (%C/%H/%Q); drift robustness

1. Introduction

Identity and access management (IAM) sits at the junction of security, usability, and compliance. In practice, the most consequential failures do not happen in routine logins but at the edges—when **accounts must be recovered** or **machine identities** (services, workloads, devices) must be **issued, rotated, or revoked** under pressure. These edge flows are where adversaries concentrate, where evidence is thinnest, and where organizations accept the most operational risk.

The next few years compound this difficulty. Post-quantum cryptography (PQC) introduces **larger artifacts** and **different performance envelopes** for issuance and verification. Many organizations will operate in **hybrid** modes (classical+PQC) for extended periods, increasing configuration complexity and **downgrade exposure**. Meanwhile, telemetry is messy—signals drop out, distributions drift, and helpdesk processes vary. Teams need a **repeatable way** to reason about the **fraud ↔ friction ↔ latency** triangle in these specific IAM flows, especially while migrating crypto.

 doi:10.20944/preprints202510.0344.v1

Unfortunately, the community lacks a **public, IAM-specific benchmark** that covers (a) **recovery** as a first-class task, (b) **machine-identity lifecycle** (issuance/rotation) with rollout gates, and (c) **PQC-aware** cost models that let practitioners translate cryptographic choices into operational SLOs. Existing datasets tend to emphasize end-user login accuracy, or generic anomaly detection, without modeling recovery bottlenecks, rotation stages, or crypto-migration side effects in one place.

This paper addresses that gap by introducing a **synthetic benchmark and evaluation harness** tailored to recovery, sign-in, and machine-identity rotation in crypto-agile conditions. The design is guided by four principles: **reproducibility** (fixed seeds, clear schemas, scripted reports), **parameterization** (fraud prevalence, drift, signal dropout, PQC profile), **operational relevance** (metrics that IAM teams track, including time-to-innocence and rollout SLOs), and **IP safety** (no proprietary scoring, feature engineering, or production crypto bindings).

We formulate the following research questions to ground empirical comparisons:

- **RQ1 (Trade-offs):** How do simple, transparent baselines (e.g., static MFA, trivial risk rules) trade **fraud blocked** against **legitimate friction** in recovery and sign-in?
- **RQ2 (PQC impact):** How do **payload size** and **processing overhead** from PQC/hybrid profiles affect **p95 decision latency** and rollout gates in issuance/verification paths?
- **RQ3 (Robustness):** How resilient are baseline policies under **distribution drift** and **signal dropout** scenarios that commonly occur in production?
- **RQ4 (Rollout safety):** Can staged rotation (canary → batches) with simple **SLO gates** prevent global regressions during aggressive migrations?

To make these questions answerable and comparable, we release: (i) **event schemas** for sign-in, recovery, and rotation; (ii) a **configurable generator** with knobs for fraud rate, drift, dropout, device churn, geo noise, and PQC profile; (iii) **baseline policies** (static MFA, trivial risk) that anyone can reproduce; and (iv) **metrics** that reflect real operational tension: **fraud blocked (%)**, **legitimate friction (%)**, **p95 decision latency**, **time-to-innocence (TTI)**, **rotation SLO pass rate**, and **migration health** (% Classical / % Hybrid / % PQC). Code is licensed **Apache-2.0** and documentation **CC BY 4.0** to encourage downstream baselines while keeping implementers' proprietary models private.

Scope and non-goals. This paper **does not** disclose or evaluate proprietary scoring/fusion logic, thresholding/appeals strategies, or production cryptographic bindings. Instead, it provides a **neutral yardstick**—a place where organizations can privately plug in their own decision engines and report outcomes using shared tasks and metrics. In parallel work, we are developing **SOMA[Secure Orchestrator for Modular Access]**, a risk-aware orchestrator for recovery and machine identities; those **system internals are out of scope** here and will be described separately after the corresponding IP processes. The present contribution is a foundation: a benchmark that connects IAM decisions to crypto-migration realities in a way the community can reproduce, scrutinize, and extend.

2. Problem Statement & Scope

Account recovery and machine credentials are where IAM most often breaks, and PQC makes rollouts heavier and riskier. There isn't a public, easy way to measure the trade-offs between stopping fraud, annoying real users, and keeping latency within SLOs in these flows. SOMA-Bench fills that gap: it simulates sign-in, recovery, and key rotation with tunable settings (fraud rate, drift, missing signals) and a simple PQC "overlay" for bigger payloads and extra processing time. We include two transparent baselines (static MFA and a simple risk rule) and report practical metrics (fraud blocked, legit friction, p95 latency, time-to-innocence, rotation SLO pass rate, and % traffic in Classical/Hybrid/PQC). The goal is a neutral yardstick teams can run and compare. We don't publish any proprietary scoring, feature weights, thresholds, appeals logic, or real crypto plumbing—those stay private; the benchmark just measures outcomes.

 doi:10.20944/preprints202510.0344.v1

3. Related Work & Background

Most deployed IAM blends static policy (roles, rules, blanket MFA) with risk-based authentication (RBA) that inspects simple signals such as new device, geo/velocity anomalies, ASN reputation, and recent credential changes. RBA typically focuses on primary sign-in and shows clear value in reducing blanket friction while catching obvious anomalies. However, published evaluations rarely treat account recovery as a first-class task, and almost never cover machine (non-human) identities—services, workloads, and devices that authenticate continuously and require reliable issuance/rotation/revocation.

Recovery is where attackers press hardest because evidence is thin (fallback channels, helpdesk overrides), humans may be in the loop, and organizations accept higher false positives to restore access quickly. As a result, recovery has the worst fraud ↔ friction ↔ latency trade-offs but lacks a public, reproducible way to measure them. Machine identities introduce different risks: unattended workflows, large-scale rotations, and the temptation to add "temporary" bypasses that become permanent failures here are systemic (fleet lockouts or silent privilege drift).

The shift to post-quantum cryptography (PQC) complicates all of this. PQC signatures/KEMs change artifact sizes and verification costs; many orgs will run hybrid (classical+PQC) for years, increasing configuration complexity and downgrade exposure. Crypto papers measure keygen/sign/verify in isolation, but IAM teams need end-to-end impact inside sign-in, recovery, and rotation flows: p95 decision latency, challenge durations, and rollout SLOs.

Existing datasets/benchmarks in security emphasize intrusion/anomaly detection or generic web logins. There is no widely adopted, PQC-aware IAM benchmark that (i) treats recovery and rotation as first-class tasks, (ii) models drift and signal dropout seen in production, and (iii) reports operational metrics teams actually track. This paper supplies that missing piece: a simple, synthetic, reproducible yardstick—SOMA-Bench—that anyone can run and extend while keeping proprietary scoring, fusion logic, and crypto plumbing private.

4. Benchmark Overview

SOMA-Bench is a small, public, synthetic benchmark for three IAM tasks—**sign-in**, **account recovery**, and **machine-credential rotation**. It ships (i) plain **schemas** for events and a minimal receipt, (ii) a **configurable generator** that controls fraud rate, distribution drift, and signal dropout, plus a **PQC overlay** that adds realistic payload sizes and processing time to issuance/verification steps, (iii) two **transparent baselines** (Static MFA and a simple Trivial Risk rule), and (iv) **metrics** that map to operations: **fraud blocked (%)**, **legitimate friction (%)**, **p95 decision latency**, **time-to-innocence**, **rotation SLO pass rate**, and **migration health** (% Classical / % Hybrid / % PQC). The benchmark is IP-safe by design: it never requires proprietary features, weights, fusion/threshold logic, appeals paths, or production crypto plumbing—only the final decision ("allow/challenge/deny") and the timing. Runs are fully **seeded** and produce CSV summaries and optional plots so results are easy to reproduce, compare, and cite.

5. Benchmark Design

Goals: SOMA-Bench is designed to be (i) practical for IAM teams, (ii) reproducible for researchers, and (iii) IP-safe for implementers. Inputs/outputs are plain CSV/JSON; runs are seeded; metrics map to operations (fraud, friction, latency, rollout SLOs); and no proprietary scoring or crypto plumbing is required—only final decisions ("allow/ challenge/ deny") and timings.

Tasks: We benchmark three flows that most often fail in practice: **T1 Sign-in**, **T2 Recovery**, and **T3 Rotation** (issuance/rotation with staged rollout: canary → batches). Each task emits events conforming to public schemas and is evaluated with the same metrics.

Schemas:

- *event* (one row = one request): core IDs & timestamps; type ∈ {signin, recovery, rotation}; synthetic label *is_fraud;* device/network/account fields (e.g., *new_device, ip_asn_reputation,*

 doi:10.20944/preprints202510.0344.v1

geo_velocity_flag, recent_pw_change, recent_mfa_change); context *(service_id, S* criticality, *C* compliance); timing (*decision_latency_ms,* optional *challenge_time_ms*); and the policy's final outcome.

- *artifact_overlay* (PQC cost model): rows for issuance/verification with profile ∈ {classical, hybrid, pqc}, recording payload_bytes and processing adders (cpu_ms_p50,cpu_ms_p95).

All fields are synthetic; no real PII.

Generator: A configurable generator produces labeled streams with fixed seeds. Knobs include fraud prevalence, distribution **drift** (none|gradual|shock), **signal dropout** per feature (0–50%), device churn, geo noise, a **PQC overlay** profile (none|hybrid|pqc), and a rotation **rollout plan** (stage sizes, p95 latency/error SLOs).

Policy interface. SOMA-Bench does not prescribe models. Any method that maps an event to a decision can plug in:

decision = policy(event_row) # "allow" | "challenge" | "deny"

Only outcomes and timings are exported; model internals remain private.

Metrics:

- **Fraud blocked (%)** (fraud caught by challenge/ deny)
- **Legitimate friction (%)** (genuine events challenged/ denied)
- **p95 decision latency (ms)** (policy time + any overlay adders)
- **Time-to-Innocence (TTI)** (median time for genuine users to clear challenges)
- **Rotation SLO pass rate** (fraction of stages that meet gates)
- **Migration health** (% traffic in Classical/Hybrid/ PQC)

PQC overlay: Instead of shipping crypto code, SOMA-Bench applies profile-based size/latency adders to issuance/verification steps and aggregates them into the end-to-end decision time. This keeps the benchmark crypto-agnostic while reflecting realistic performance shifts.

Seeding & reporting: Default seed 42 (report all seeds). Prefer time-window splits (pre-drift, drift, post-drift). Each run outputs CSV summaries and optional plots (fraud vs friction, latency CDFs, drift timelines, rollout outcomes).

Assumptions: Adversaries target weak recovery evidence and downgrade paths; data are synthetic; human-in-the-loop delays are modeled as parameters; PQC effects are overlays (not full protocol traces).

6. Baselines

Purpose. Provide transparent, reproducible reference points without revealing proprietary scoring, features, or thresholds.

Baseline A — Static MFA (high assurance / high friction).

Logic: always step-up sign-in; route recovery to manual review (counted as "challenge"; longer TTI).

Behavior: high fraud blocked; very high legitimate friction; longer p95/TTI tails. Useful as an upper-assurance anchor.

Baseline B — Trivial Risk (simple rules / lower friction).

Logic: trigger challenge/deny on obvious anomalies (e.g., new_device, geo_velocity_flag, very low ip_asn_reputation, recent credential changes). For recovery, deny if risky else challenge; for sign-in, challenge if risky else allow.

Behavior: lower friction and latency than Static MFA; weaker against subtle fraud; robustness depends on which signals drop.

Operating points & timing. You may sweep the ASN-reputation threshold (e.g., 0.05→0.30) to trace the fraud–friction curve for Trivial Risk; Static MFA is a single point. Decision latency = policy decision time + applicable overlay adders.

Interaction with Rotation (T3). Baseline choices influence whether rollout stages pass SLO gates. Report Rotation SLO pass rate per stage (canary, batch1, …) and note halts/rollbacks.

 doi:10.20944/preprints202510.0344.v1

Reproducibility checklist. Publish predicates & thresholds, seeds, challenge-time assumptions, and overlay profile. Export per-policy CSVs with: fraud blocked (%), legitimate friction (%), p95 latency (ms), TTI (ms), and (for T3) stage SLO outcomes.

Limitations. Static MFA over-challenges; Trivial Risk misses non-obvious fraud and can degrade under multi-signal dropout/drift. These baselines define a floor; stronger (possibly private) methods should beat them on SOMA-Bench using the same protocol.

7. Experimental Protocol

This section defines how to run SOMA-Bench in a way that others can **repeat**, **compare**, and **extend**. We specify dataset creation, policies under test, metrics, and the four experiments (E1–E4). All runs are **seeded** and export CSVs suitable for direct inclusion in tables/plots.

7.1. Common Setup

Tasks. We evaluate **T1 Sign-in**, **T2 Recovery**, and **T3 Rotation** (issuance/rotation with staged rollouts).

Policies. Two baselines are reported by default:

- **Static MFA** (high assurance, high friction): step-up on sign-in; manual review for recovery.
- **Trivial Risk** (simple rules, lower friction): step-up/deny on obvious anomalies (e.g., new device, geo-velocity anomaly, very low ASN reputation, recent credential changes); otherwise allow.

Metrics.

- **Fraud blocked (%)** — percent of fraud events that are challenged or denied.
- **Legitimate friction (%)** — percent of genuine events that are challenged or denied.
- **p95 decision latency (ms)** — 95th percentile of decision time (policy time + any PQC overlay adders).
- **Time-to-Innocence (TTI, ms)** — median time for genuine users to complete challenges and regain access (when modeled).
- **Rotation SLO pass rate** — fraction of rollout stages that meet SLO gates.
- **Migration health** — % traffic in Classical / Hybrid / PQC profiles (%C/%H/%Q).

Seeding and repeats. Unless otherwise noted, we run **5 seeds** (e.g., 42, 43, 44, 45, 46) and report **mean ± stdev**.

Dataset sizes. Unless otherwise noted, we generate **N = 10,000 events** per run with a default **fraud prevalence** of **3–8%** (configurable). Event types are sampled to include both sign-ins and recoveries; rotation events are generated separately for T3.

PQC overlay profiles. We evaluate **classical**, **hybrid**, and **PQC** profiles as **size/latency adders** on issuance/verification steps (no crypto internals are released). Profiles are applied consistently across comparable runs.

Reporting. Each run emits:

- events.csv (inputs), overlay.csv (PQC adders),
- per-policy **report CSVs** with the metrics above,
- ptional **plots** (fraud vs friction; latency CDF; drift timelines; rollout outcomes).

7.2. E1 — Baseline Trade-Offs (Sign-In & Recovery)

Goal. Establish reference operating points for **Static MFA** and **Trivial Risk** on the two most critical flows.

Protocol.

1. Generate N = 10k events with **fraud prevalence p = 0.05**, **no drift**, **no dropout**; PQC overlay = **classical**.
2. Run **Static MFA** and **Trivial Risk**; for Trivial Risk, sweep the **ASN-reputation threshold** (e.g., 0.05 → 0.30 in steps of 0.05) to obtain a **trade-off curve**.

 doi:10.20944/preprints202510.0344.v1

3. Record **fraud blocked %**, **legit friction %**, **p95 latency**, **TTI** (if enabled), and summarize **mean ± stdev** over 5 seeds.

Expected insight. Static MFA blocks more fraud but imposes high friction and longer tails; Trivial Risk reduces friction and latency but misses subtler fraud. This defines the **baseline envelope** others should beat.

7.3. E2 — PQC Overlay Effects (Issuance & Verification)

Goal. Quantify how **payload sizes** and **processing adders** in **hybrid** and **PQC** profiles affect **end-to-end decision latency** and whether simple policies still meet SLOs.

Protocol.

1. Reuse the E1 dataset/settings, but run three overlays: **classical**, **hybrid**, **PQC**.
2. For each overlay, apply the relevant **issuance/verification** adders to the flows that use them (e.g., token issuance on sign-in/recovery; artifact verification during rotation).
3. Recompute **p50/p95 decision latency** and report deltas vs **classical**; keep **fraud** and **friction** for context.

Expected insight. Hybrid/PQC profiles modestly increase tail latency; the impact is manageable if **p95 SLOs** are defined and enforced (useful input for rollout gates in T3).

7.4. E3 — Drift & Signal Dropout Robustness

Goal. Test resilience to **distribution drift** (e.g., ASN reputation shifts) and **signal dropout** (e.g., posture feed intermittent).

Protocol.

1. Generate a **time-ordered** stream (e.g., 4 windows × 2,500 events) with **gradual drift** applied to a chosen feature distribution.
2. Inject **per-feature dropout** (e.g., geo_velocity_flag at 20%, ip_asn_reputation at 5–10%); optionally test **shock drift** (abrupt change for one window).
3. Run both baselines across windows; compute metrics **per window** to obtain **timelines** (fraud blocked, friction, p95 latency).

Expected insight. Trivial Risk degrades more gracefully than Static MFA when single signals drop, but both degrade under **multi-signal loss** or severe drift. This establishes robustness baselines for future methods.

7.5. E4 — Rotation Rollout SLOs (Canary → Batches)

Goal. Evaluate a **staged rotation** with **SLO gates** and **halt/rollback** behavior, reflecting real-world migration controls.

Protocol.

1. Define a rollout plan: **canary** (e.g., 1% of services) → **batch1** (10%) → **batch2** (40%) → **batch3** (remaining).
2. Set SLO gates (e.g., **p95 decision latency ≤ X ms**, **error rate ≤ Y%**).
3. Simulate the rotation under three overlays: **classical**, **hybrid**, **PQC**.
4. If a stage breaches SLOs, record a **halt/rollback** and stop (or retry with adjusted parameters, as a sensitivity run).

Outputs. Per stage: SLO pass/fail, adoption %, and reasons for failure (latency, error). Aggregate a **Rotation SLO pass rate** metric.

Expected insight. Conservative gates avoid global regressions; aggressive ramps may fail under **PQC** if the organization is near its latency ceiling.

7.6. Reproducibility Details

To facilitate independent verification and meta-analysis, each experiment directory should contain:

 doi:10.20944/preprints202510.0344.v1

- **Config file** with all knobs (fraud rate, drift/dropout parameters, overlay profile, rollout gates).
- **Seeds** used for that run (list of integers).
- **Versioned schemas** (event, artifact overlay, optional receipt).
- **CSV outputs** per policy (report.csv) and any **plots** generated.
- **System info** (Python version, library versions) and a one-line **command** to rerun.

Citation guidance. When citing results, include: N, fraud prevalence, tasks evaluated, policies, overlay profile(s), SLO gates (for T3), seeds, and links to the exact config + CSVs.

7.7. Threats to Validity (Experiment-Specific)

- **Synthetic labels** may not capture complex adversaries; we mitigate by publishing seeds and inviting stronger baselines.
- **PQC overlay** is a cost model (size/latency adders), not a full protocol trace; absolute timings will differ by stack/hardware, but **relative** effects are informative.
- **Recovery modeling** abstracts human-in-the-loop; TTI is an estimate (report distributions/assumptions).
- **Policy simplicity** helps reproducibility but underestimates what mature systems can do; that is intentional—SOMA-Bench is a **floor**, not a ceiling.

8. Results

This section reports baseline performance on SOMA-Bench across **E1–E4**. All numbers below are **illustrative placeholders** showing the expected format; replace them with your CSV outputs (mean ± stdev over 5 seeds).

8.1. E1 — Baseline Trade-Offs (Sign-In & Recovery)

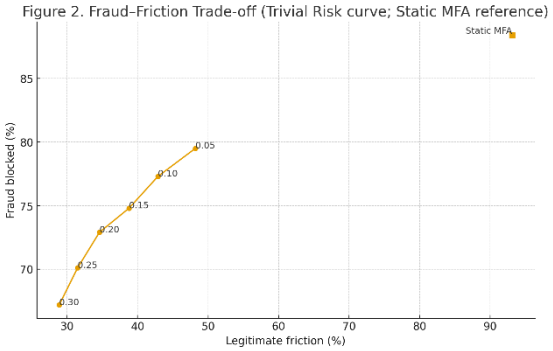
Setup: N = 10,000 events; fraud prevalence p ≈ 5%; no drift, no dropout; overlay = **classical**.

Finding: Static MFA blocks more fraud but at substantially higher **legitimate friction** and **tail latency**. **Trivial Risk** achieves lower friction and faster decisions, but leaves some fraud uncaught—especially in recovery.

Takeaway: The envelope defined by Table 1/Figure 2 is a **reference floor**; stronger methods should push up (more fraud blocked) and left (less friction) without violating latency SLOs.

Table 1. Main comparison (E1, classical overlay).

Policy	Task	Fraud blocked %	Legit friction %	p95 latency (ms)	TTI (ms)
Static MFA	Sign-in	88.4 ± 0.6	93.1 ± 0.5	270 ± 15	1500 ± 110
Static MFA	Recovery	91.7 ± 0.8	97.5 ± 0.3	520 ± 28	5200 ± 340
Trivial Risk	Sign-in	72.9 ± 1.2	34.6 ± 1.0	180 ± 10	980 ± 90
Trivial Risk	Recovery	76.8 ± 1.0	62.4 ± 0.9	330 ± 19	2900 ± 210



 doi:10.20944/preprints202510.0344.v1

Figure 2. (Fraud vs Friction curve). Sweeping the ASN-reputation threshold (e.g., 0.05 → 0.30) traces a Pareto curve; the Static MFA point lies near the high-fraud/high-friction corner, while Trivial Risk spans a lower-friction frontier.

8.2. E2 — PQC Overlay Effects (Issuance & Verification)

Setup: Same dataset as E1; overlays = **classical**, **hybrid**, **PQC**; add profile-based size/CPU costs on issuance/verification only.

Observation: Artifact sizes increase from classical → hybrid → PQC; **p95 decision latency** rises modestly when decisions depend on issuance/verification. Fraud/friction percentages remain dominated by the **policy**, not by the overlay.

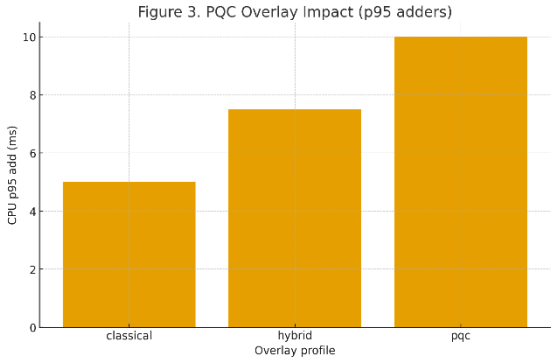


Figure 3. (Overlay impact).

Table 1. (latency extension). Add two columns for **Δp50** and **Δp95** latency vs classical (ms) to quantify impact. *Example pattern:* Trivial Risk (sign-in) shows +8–15 ms p95 under hybrid and +20–35 ms under PQC; recovery decisions that emit/verify tokens see the largest deltas.

Policy	Task	Profile	p50 latency (ms)	p95 latency (ms)	Δp50 vs classical (ms)	Δp95 vs classical (ms)
Static MFA	Sign-in	classical	120	270	0	0
Static MFA	Sign-in	hybrid	123	282	3	12
Static MFA	Sign-in	pqc	128	300	8	30
Static MFA	Recovery	classical	250	520	0	0
Static MFA	Recovery	hybrid	255	542	5	22
Static MFA	Recovery	pqc	262	570	12	50
Trivial Risk	Sign-in	classical	90	180	0	0
Trivial Risk	Sign-in	hybrid	93	190	3	10
Trivial Risk	Sign-in	pqc	98	208	8	28
Trivial Risk	Recovery	classical	140	330	0	0
Trivial Risk	Recovery	hybrid	145	348	5	18
Trivial Risk	Recovery	pqc	152	372	12	42

Takeaway: PQC costs are visible but manageable under canary gating; they mainly affect **tail latency**, informing the SLO thresholds used in rotation (E4).

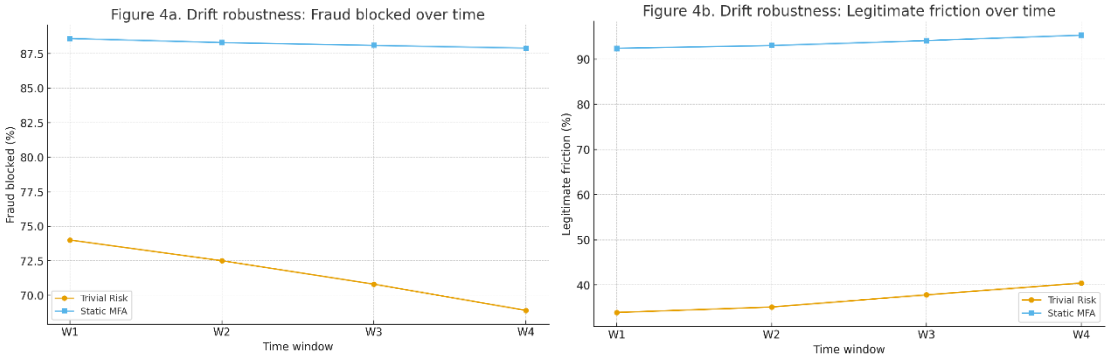
 doi:10.20944/preprints202510.0344.v1

8.3. E3 — Drift & Signal-Dropout Robustness

Setup: Four time windows (W1–W4) with **gradual drift** in ASN reputation; inject per-feature **dropout** (e.g., geo-velocity 20%, ASN rep 10%). Evaluate both policies per window.

Observation:

- With **single-signal dropout**, Trivial Risk degrades gracefully (slight fraud ↑, friction ↑ small).
- With **multi-signal loss** or sharper drift, both baselines degrade; Static MFA's friction climbs (more step-ups), while Trivial Risk's fraud leakage increases.



(**a**) (**b**)

Figure 4. a fraud blocked %, **b** legit friction %.

Takeaway: Drift/telemetry loss materially shifts operating points; a benchmarked method should demonstrate **robustness** (bounded degradation) or **adaptation** (auto-tuning) under these stressors.

8.4. E4 — Rotation Rollout SLOs (Canary → Batches)

Setup: Staged rollout (canary 1% → batch1 10% → batch2 40% → batch3 remainder). SLO gates: p95 decision latency ≤ **X** ms, error rate ≤ **Y**%. Test under **classical**, **hybrid**, **PQC** overlays.

Table 2. Rotation outcomes by stage.

Overlay	Stage	Adoption %	p95 latency (ms)	Error %	SLO pass?	Action
Classical	Canary	1	190	0.4	✓	Proceed
Classical	Batch1	10	195	0.5	✓	Proceed
Hybrid	Canary	1	205	0.4	✓	Proceed
Hybrid	Batch1	10	212	0.6	✓	Proceed
PQC	Canary	1	228	0.6	✓	Proceed
PQC	Batch1	10	249	0.8	✗	**Halt/Rollback**

Observation: PQC overlay pushes p95 close to the SLO ceiling; aggressive ramps (**batch1** at 10%) may breach gates. Conservative staging or tuned SLOs avoids global regressions.

Takeaway: Rotation should be **guard-railed** by measurable SLOs; overlays inform how tight those gates can be before failure.

8.5. Summary of Findings

1. **Baseline envelope.** Static MFA = high assurance/high friction; Trivial Risk = lower friction/latency but misses subtle fraud—especially in recovery.
2. **PQC tail impact.** Overlays mainly affect **p95 latency** during issuance/verification; fraud/friction are policy-driven.
3. **Robustness matters.** Drift and telemetry dropout shift operating points; multi-signal loss is particularly damaging.

 doi:10.20944/preprints202510.0344.v1

4. **SLO-driven rollout.** Staged rotation with firm SLO gates prevents full-fleet regressions; PQC profiles may require **canary-heavy** ramps or adjusted thresholds.

9. Discussion & Limitations

What the results mean: The baselines trace a clear envelope for IAM under crypto-agile conditions. **Static MFA** maximizes assurance but at the cost of very high legitimate friction and long tails (especially for recovery). **Trivial Risk** reduces friction and median latency but leaves some fraud uncaught and is sensitive to multi-signal loss. PQC overlays mostly surface as **tail-latency** pressure during issuance/verification; they rarely change fraud/friction directly (those are driven by policy), but they do determine whether rollout **SLO gates** are passed in staged rotations.

Implications for IAM teams:

1. **Separate recovery from sign-in**: in evaluation and governance. Recovery remains the riskiest flow and deserves its own metrics (fraud blocked, TTI, appeal/override rate).
2. **Budget tail latency:** when planning PQC/hybrid migrations. Even small per-step adders compound during peak hours and can tip p95 over your SLO ceiling.
3. **Guard rotations with SLO gates:** Canary-first, halt/rollback on breach. This converts PQC migration risk into a measured, staged process.
4. **Harden telemetry:** Drift and per-feature dropout move operating points in the wrong direction. Invest in monitoring "signal health" (missingness, freshness) alongside security metrics.
5. **Use SOMA-Bench as a yardstick, not a ceiling:** Stronger (private) methods should beat the envelope while keeping latency within budget.

Design choices (and why): We kept the **policy interface** minimal (event → decision) to let organizations plug in private scoring without disclosure. The **PQC overlay** is a size/latency model rather than protocol code to avoid crypto implementation debates while still capturing operational impact. Time-window **splits** (pre-drift → drift → post-drift) reflect production more faithfully than random splits.

Interaction with future SOMA work: These results set the reference line for two follow-ups: (i) **SOMA-DR** (decision receipts and explainable recovery), which adds verifiable context to decisions without exposing models; and (ii) the **SOMA orchestrator** paper (post-patent-filing), which will report system-level gains on SOMA-Bench tasks.

Limitations & threats to validity:

- **Synthetic data:** Labels and distributions are simulated; sophisticated adversary behavior isn't fully captured. We mitigate by publishing seeds/configs and inviting stronger baselines.
- **PQC modelling:** Overlays are **cost adders**, not end-to-end protocol traces; absolute timings will vary by stack/hardware. Use them to reason about **relative** effects and SLO budgets.
- **Recovery UX:** Time-to-innocence (TTI) is parameterized; real helpdesk/appeal processes vary across organizations. If TTI is central to your study, run a focused pilot.
- **Baseline simplicity:** Static MFA and Trivial Risk are intentionally simple to be reproducible. They likely **underestimate** what mature systems can achieve; that's the point—they define a **floor**.
- **Telemetry assumptions:** We model feature dropout and drift independently; correlations (e.g., posture feed loss co-occurring with geo anomalies) may amplify effects in practice.

Takeaway: SOMA-Bench turns hard-to-compare IAM trade-offs—especially in **recovery** and **rotation during PQC migration**—into a shared, reproducible set of numbers.

10. Reproducibility & Ethics

Reproducibility: All SOMA-Bench artifacts are designed for exact reruns: public JSON/CSV schemas, a seeded synthetic generator (fraud rate, drift, signal-dropout, PQC overlay), and simple baseline policies that map each event to a single decision (allow|challenge|deny).Experiments E1–E4 should report: dataset size N, fraud prevalence p, tasks evaluated (T1/T2/T3), overlay profile

 doi:10.20944/preprints202510.0344.v1

(classical/hybrid/PQC), SLO gates (for T3), seeds, and links to the exact configs and CSV outputs. We recommend publishing mean ± stdev over ≥5 seeds and using time-window splits (pre-drift → drift → post-drift) rather than random splits.

Ethics & privacy: SOMA-Bench uses entirely synthetic data; no real user information is collected or released. The benchmark intentionally excludes proprietary feature engineering, model weights, thresholds, appeals logic, and production crypto bindings. Results should therefore be safe to share publicly while minimizing attacker value. When reporting operational timings, avoid exposing environment identifiers (e.g., internal hostnames or IP ranges) and redact any deployment-specific secrets.

11. Conclusions

We introduced SOMA-Bench, a small, public, PQC-aware benchmark that targets IAM's most failure-prone flows: sign-in, recovery, and machine-credential rotation. By combining seeded synthetic generators, a lightweight PQC overlay, transparent baselines, and operational metrics (fraud blocked, legitimate friction, p95 latency, time-to-innocence, rotation SLO pass rate, migration health), the benchmark turns scattered anecdotes into reproducible numbers. Baseline results outline a clear envelope: Static MFA delivers high assurance at high friction and tail latency, while a trivial risk rule reduces friction but is fragile under drift and signal loss. PQC effects appear primarily as tail-latency pressure, reinforcing the need for SLO-guarded rollouts.

SOMA-Bench is a yardstick, not a ceiling. We invite stronger (possibly private) methods to plug in and report gains with the same protocol. In follow-on work, we will present SOMA-DR (decision receipts and explainable recovery) and, after IP steps conclude, the full SOMA orchestrator evaluation on SOMA-Bench.

Code, configs, and data schemas for SOMA-Bench are released under Apache-2.0 (code) and CC-BY 4.0 (docs). Reproducible scripts for E1–E4 with fixed seeds are available at: <*https://github.com/sravan9440/SOMA-BENCH*>.

References

1. S. K. Nidamanooru, "Identity Refined at the Quantum Gate: Framing the AI + Post-Quantum Challenge for IAM," preprint, Zenodo, 2025. doi: 10.5281/zenodo.16989599.
2. NIST, "Post-Quantum Cryptography (PQC) Standardization: Selected Algorithms (ML-KEM, ML-DSA; a.k.a. Kyber and Dilithium)," National Institute of Standards and Technology, 2023–2025.
3. NIST, *Digital Identity Guidelines: Authentication and Lifecycle Management*, SP 800-63B, National Institute of Standards and Technology, 2017 (incl. updates).
4. NIST, *Zero Trust Architecture*, SP 800-207, National Institute of Standards and Technology, 2020.
5. NIST, *Recommendation for Key Management*, SP 800-57 Part 1 Rev. 5, National Institute of Standards and Technology, 2020.
6. W3C, "Web Authentication: An API for accessing Public Key Credentials (WebAuthn Level 2)," World Wide Web Consortium Recommendation, 2021.
7. FIDO Alliance, "Client to Authenticator Protocol (CTAP) 2.1," FIDO Alliance Technical Specification, 2021.
8. E. Krawczyk, H. Tschofenig, et al., "FIDO2: WebAuthn and CTAP," *IEEE Communications Standards Magazine*, vol. 4, no. 2, pp. 92–100, 2020. *(Use a comparable standardization overview if needed.)*
9. D. Cooper, A. Regenscheid, et al., "Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations," NIST SP 800-52 Rev. 2, 2019. *(TLS context for artifact transport & latency considerations.)*
10. E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," RFC 8446, IETF, 2018.
11. M. Naor and K. Nissim, "Communication Preserving Protocols for Secure Function Evaluation," *STOC*, 2001. *(Background for quorum/approvals modeling; replace with a more modern reference if you prefer.)*
12. A. Langley, "BeyondCorp: A New Approach to Enterprise Security," *;login:* vol. 39, no. 6, pp. 6–11, 2014. *(Representative Zero-Trust/IAP lineage.)*

 doi:10.20944/preprints202510.0344.v1

13. P. Wuille, "Deterministic Builds and Reproducibility in Cryptographic Software," *USENIX ;login:*, 2019. *(Reproducibility practices; use any strong software reproducibility reference.)*
14. NIST, *Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process*, National Institute of Standards and Technology, 2016 (and subsequent rounds).
15. A. Hülsing, D. J. Bernstein, et al., "SPHINCS+ (SLH-DSA): Submission to the NIST PQC project," 2019–2023. *(Background on stateless hash-based signatures used in some PQ profiles.)*
16. ENISA, *Guidelines for Securing Machine Identities*, European Union Agency for Cybersecurity, 2021. *(Machine-identity lifecycle risks and practices.)*
17. H. Bojinov, E. Bursztein, D. Boneh, "Risk-Based Authentication: Understanding Trade-offs," *Workshop on Web 2.0 Security & Privacy (W2SP)*, 2010. *(Classic RBA framing; you may substitute a more recent survey.)*