



id: SI\_003

model: google/gemma-4-26b-a4b-it

seeds:

04\_risiko: mulighet

03\_relasjon: tillit

05\_handling: implementer

01\_perspektiv: operasjonell

02\_tidhorisont: historisk

## Instans SI\_003

**Seeds:** operasjonell | historisk | tillit | mulighet | implementer

Analysen av agentisk KI i deres kontekst avdekker et fundamentalt gap mellom deterministisk arv og autonom fremtid.

**Arkitektonisk fundament og sikkerhetsmekanismer** Vi kan ikke bygge på eksisterende RBAC alene; det er en **historisk** blindvei for agenter. Dere må skifte fra identitetsbasert til kontekstbasert kontroll.

- **Implementer en “Policy-as-Code” motor (f.eks. Open Policy Agent - OPA):** Agenten må ikke bare ha tilgang, men må sende en “intent-token” som valideres mot sanntidsregler.
- **Operasjonell sandkassing:** Innfør en Sidecar-arkitektur for alle agenter. Agenten snakker ikke direkte med API-et, men med en kontrollproxy som utfører semantic rate-limiting (stopper ikke bare antall kall, men ulogiske sekvenser av handlinger).
- **Isolasjon:** Bruk kortlevde, rettighetsbegrensede “Ephemeral Identities”. Nøkler må roteres per oppgave, ikke per sesjon.

**Tekniske blindsoner og undervurderte risikoer** Den største **historiske** feilen er å tro at logging er det samme som etterprøvnbarhet. I fragmenterte systemer vil en agent skape “log-gap” – handlinger som ser legitime ut isolert, men er destruktive i korrelasjon.

- **Risiko:** Cascading failure i eldre fagsystemer som ikke håndterer raske, ikke-menneskelige transaksjonsmønstre.
- **Blindsoner:** “Semantic Data Leakage”. En agent kan tro den følger regler, men aggregere data fra tre API-er til en sensitiv profil som bryter personvernet.



**Mulighet: Kontrollmodellen “Human-in-the-loop-by-exception”** For å unngå flaskehalser må kontrollen være asynkron. Design en arkitektur basert på Confidence Scores:

1. **Score > 0.9:** Autonom utførelse + asynkron logging.
2. **Score 0.7-0.9:** Foreslå handling, vent på menneskelig “one-click” godkjenning.
3. **Score < 0.7:** Blokker og eskaler.

Dette skaper **tillit** ved at sikkerheten er innebygd i beslutningslogikken, ikke som et eksternt hinder.

**Neste steg:**

1. **POC på Policy-Proxy:** Test OPA mot ett eldre API for å se om dere kan interceptere og blokkere “ufornuftige” kall.
2. **Kartlegg “Blast Radius”:** Definer hvilke datafelt som er irreversible (f.eks. sletting/endring av utbetalinger) og krev 100% menneskelig verifisering her.
3. **Etabler “Observability-first” krav:** Ingen agent får tilgang til et system før en sentralisert logg-aggregator kan korrelere agentens `intent_id` med systemets `transaction_id`.