



id: SI\_002

model: openai/gpt-5.4-mini

seeds:

04\_risiko: mulighet

03\_relasjon: makt

05\_handling: implementer

01\_perspektiv: bruker

02\_tidhorisont: kortsiktig

## Instans SI\_002

**Seeds:** bruker | kortsiktig | makt | mulighet | implementer

Agentisk KI bør i en offentlig virksomhet behandles som **en ny aktørklasse med begrenset makt**, ikke som “smart automasjon”. Det viktigste er ikke full frihet, men å **gjøre agentens makt kortsiktig, reverserbar og målbar**.

### Arkitekturprinsipper

- **Delt kontrollplan:** skill mellom planlegging (LLM) og utførelse (deterministiske wrappers).
- **Default deny + positiv tillatelse:** agenten får bare gjøre eksplisitt tillatte operasjoner per system, datasett og tidspunkt.
- **Runtime-isolasjon:** kjør agenten i sandkasse med nettverks- og filrestriksjoner; ingen direkte tilgang til prod uten mellomlag.
- **Execution graph per oppdrag:** modeller hvert oppdrag som en DAG med noder for beslutning, kall, validering, og kompensasjon. Hver node får unik trace-id, input, output, policy-resultat.

### Zero Trust, policy og least privilege

- Bruk **policy-as-code** før hvert kall: OPA/Kyverno/Gatekeeper-lignende beslutning, men start enkelt hvis plattformen er svak.
- Gi agenter **ephemeral credentials** med TTL 5–15 min, scopes per API og per ressursklasse.
- Bruk **delegert autorisasjon**: agenten handler på vegne av bruker/rolle, men aldri bredere enn nødvendig.



- Kombiner med **step-up approval** for høyrisiko: betaling, sletting, masseoppdatering, eksterne API-kall.

---

## Integrasjoner og API-risiko

Legacy-API-er uten idempotens er høyrisiko. Anta alltid:

- dobbeltkall
- delvise sideeffekter
- skjulte avhengigheter

Tiltak:

- idempotency keys der mulig
- rate limiting per agent og per system
- circuit breakers
- “dry-run/preview mode” for første fase
- kompensatoriske handlinger definert før produksjon

---

## Audit og hendelser

Bygg en **append-only audit-pipeline** som logger:

1. prompt / oppgave
2. policy-beslutning
3. kall og respons
4. sideeffekt
5. kompensasjon/rollback

Hvis dette ikke er sentralisert, er kontrollen i praksis svak. Distribuerte logger alene er en blindfleck.

---

## Blindsoner og worst case

Undervurdert risiko er **latent coupling**: agenten trigger flere “små” hendelser som sammen gir stor effekt. Worst case er ikke én feil, men en **runaway loop** som endrer data, starter jobber, og eskalerer rettigheter indirekte.

---

## Kortsiktig anbefaling

Ikke prøv full autonomi først. Implementer en **“high-friction autonomy”**: begrenset handlingsrom, sterk logging, tidsbegrensede tokens, og automatisk isolering ved avvik.



## Neste steg

1. Velg 1–2 use cases med lav integritetsrisiko og definer eksplisitt execution graph.
2. Etabler policy-gate og ephemeral credentials før noen agent får prod-tilgang.
3. Bygg audit-spor og kill-switch før pilot, ikke etter.