



id: SI_001

model: anthropic/claude-sonnet-4.6

seeds:

04_risiko: usikkerhet

03_relasjon: tillit

05_handling: utforsk

01_perspektiv: etisk

02_tidhorisont: historisk

Instans SI_001

Seeds: etisk | historisk | tillit | usikkerhet | utforsk

Perspektiv: Tillit som arkitektonisk fundament

Historisk har vi sett at teknologiske systemer som skalerer autonomi uten tillit-infrastruktur kollapse spektakulært — Knight Capital (2012) tapte 440 millioner dollar på 45 minutter grunnet ukontrollerte automatiserte handlinger. Dette er din fremtid uten riktige grenser.

Etisk kjerne: Autonomi krever ansvarskjede

Offentlig sektor bærer et særlig etisk ansvar. Agentisk KI som endrer data om innbyggere uten eksplisitt menneskelig godkjenning er ikke et teknisk spørsmål — det er et demokratisk legitimitetsspørsmål. **Prinsipp #1: Ingen agent skal kunne ta irreversible beslutninger om enkeltpersoner uten human-in-the-loop.**

Praktisk arkitektur under usikkerhet

Gitt begrenset DevSecOps-kapasitet anbefaler jeg ikke å bygge full Zero Trust fra dag én. Det er en historisk feil å overdesigne under usikkerhet. Gjør dette i stedet:

- Definér tre autonominivåer** — Read-only, Write-with-log, Execute-with-approval. Start piloten utelukkende på nivå 1.
- OPA som enkel gateway** — Implementér én sentralisert policy-sjekk på API-gateway-laget, ikke overalt. Én sjekk > ingen sjekk.
- Ephemeral credentials via Vault eller Azure Managed Identity** — TTL maks 15 minutter, aldri lagrede nøkler i agent-minne.



Blindsoner som undervurderes

Latent coupling er den farligste risikoen i legacy-landskap. Et fagsystem fra 2009 kan ha udokumenterte sideeffekter ved API-kall — agenten vet ikke, du vet ikke, ingen vet.

Utforsk dette aktivt: kjør read-only agenter først og logg alle responser, ikke bare suksess.

Emergent loops: Sett hard ceiling på agentens handlinger — maks 50 API-kall per kjøring, ellers automatisk isolering og varsling.

Audit som tillitsbygging

Audit-pipeline er ikke bare compliance — det er din tillit-kapital overfor ledelse og innbyggere. Bruk strukturert logging (JSON + correlation-ID) fra dag én. Uten dette kan du ikke forsvare beslutninger i etterkant.

Neste steg

1. **Uke 1-2:** Kartlegg alle API-er agenten vil nå — klassifiser etter idempotens og reversibilitet
2. **Uke 3-4:** Implementér OPA på API-gateway med deny-by-default og kun read-tillatelser
3. **Måned 2:** Gjennomfør et structured “red team”-scenario der agenten aktivt forsøker å bryte grensene — dokumentér og juster policy