



id: SI_004

model: mistralai/mistral-large-2512

seeds:

04_risiko: sårbarhet

03_relasjon: makt

05_handling: implementer

01_perspektiv: juridisk

02_tidhorisont: kortsiktig

Instans SI_004

Seeds: juridisk (ansvar), kortsiktig (pilotvindu), makt (autonomi), sårbarhet (systemlandskap), implementer (konkrete grep)

Kortsiktig makt - juridisk sårbarhet i et kaotisk landskap

Agentisk KI er makt uten historisk sidestykke: autonome aktører som handler på tvers av systemer, uten menneskelig mellomledd. Men makt uten juridisk forankring er en tikkende bombe. I et 6–9 måneders pilotvindu må dere implementere kontrollmekanismer som juridisk tåler både personvernforordningen (GDPR), offentlighetsloven og sektorregelverk (f.eks. helse- eller finanslovgivning). **Problem:** Dagens IAM og logging er ikke designet for sårbarheten som oppstår når en agent kan endre data uten deterministiske kontrakter.

Tre konkrete grep for å lukke sårbarheter

1. Juridisk sandkasse med policy-as-code (PaC)

- Implementer Open Policy Agent (OPA) som en kortsiktig løsning for runtime-policyer. Eksempel: En agent som skal oppdatere et pasientjournal-API må først sjekke mot en regel som sier “kun endringer med <5% risiko for feilklassifisering, og kun mellom 08–16”. Bruk Rego-språket til å definere juridisk gyldige grenser (f.eks. “ingen endringer i sensitive personopplysninger uten logget prompt”).
- Makt-begrensning: Krev at alle agenthandlinger må gå via en proxy-tjeneste som logger hver beslutning (prompt + kontekst + policy-evaluering) i et uforanderlig format (f.eks. AWS CloudTrail + immutabel S3-bøtte).



2. Ephemeral credentials + “break glass”-mekanismer

- Sårbarheten i legacy-systemer (f.eks. monolitter uten moderne IAM) krever kortsiktige løsninger:
 - Bruk HashiCorp Vault til å utstede tidsbegrensede tokens (levetid: 1–5 minutter) for hver agenthandling. Juridisk fordel: Tokens kan spores til en spesifikk agentinstans og prompt.
 - Makt-kontroll: Implementer en “break glass”-API som kan deaktivere alle agent-tokens innen 30 sekunder ved mistanke om eskalering. Eksempel: Hvis en agent gjør 3 mislykkede API-kall på rad, trigger dette en automatisk isolering.

3. Observability for emergent behavior

- Sårbarheten i uforutsigbare API-er (f.eks. manglende idempotens) krever kortsiktig overvåking:
 - Bruk OpenTelemetry til å spore agentens execution graph i sanntid. Eksempel: Hvis en agent gjør et API-kall som endrer tilstand i et legacy-system, må alle påfølgende handlinger logges med korrelasjons-ID for å kunne rulle tilbake.
 - Juridisk nødvendig: Lag en “audit trail” som fanger hver prompt, policy-evaluering og API-respons. Dette er kritisk for å bevise ansvar ved feil (f.eks. GDPR Artikkel 5 om “ansvarlighet”).

Dissens: Mainstream undervurderer latent coupling

De fleste fokuserer på direkte risikoer (f.eks. agenten gjør feil API-kall). Juridisk farligere er indirekte risikoer: En agent som endrer et felt i et legacy-system kan trigge en kjedereaksjon i et annet system (f.eks. en faktura blir sendt fordi et felt i et eldre økonomisystem endres). Dette kalles latent coupling – og det er her sårbarheten er størst. **Løsning:** Modellér alle systemer som en graf (f.eks. med Neo4j) og identifiser “high-risk paths” der en agenthandling kan eskalere.

Neste steg (innen 30 dager)

1. **Implementer OPA + proxy-tjeneste** for å tvinge alle agenthandling gjennom en juridisk sjekkpunkt. Start med 3–5 kritiske policyer (f.eks. “ingen endringer i personopplysninger uten logget prompt”).
2. **Kjør en “chaos engineering”-test** der dere simulerer en agent som gjør feil API-kall i et legacy-system. Mål tid til deteksjon og isolering.



3. **Opprett en “break glass”-protokoll** med ephemeral tokens og en nødstop-API. Test denne med en ekte agent i et isolert miljø.
-

Avslutning: Agentisk KI er ikke et teknisk problem – det er et juridisk og makt-problem i forkledning. I et pilotvindu på 6–9 måneder må dere prioritere sårbarhetsreduksjon over perfeksjon. Start med de tre grepene over, og aksepter at dere vil oppdage nye blindsoner underveis.