



id: SI_005

model: mistralai/mistral-large-2512

seeds:

04_risiko: sårbarhet

03_relasjon: makt

05_handling: utforsk

01_perspektiv: strategisk

02_tidhorisont: kortsiktig

Instans SI_005

Seeds: strategisk (maktbalanse) | kortsiktig (tidspress) | makt (agentens handlingsrom) | sårbarhet (systemets eksponering) | utforsk (kontrollert autonomi)

Analyse: Agentisk KI som maktprojeksjon i et sårbart landskap

Agentiske KI-agenter er ikke verktøy, men aktører – de utøver makt på vegne av organisasjonen, ofte utenfor designede kontrollstrukturer. Den kortsiktige tidslinjen (6–9 måneder) tvinger dere til å balansere utforskning med sårbarhetsminimering, men risikoen er asymmetrisk: En feilhandling kan eskalere uforutsigbart (f.eks. en agent som endrer data i et eldre fagsystem via en dårlig dokumentert API), mens gevinsten (automatiserte beslutninger) er diffus.

Strategiske maktspill i arkitekturen

- 1. Handlingsrom = Makt:** Agentenes handlingsrom må defineres negativt (hva de ikke kan gjøre), ikke positivt. Eksempel:
 - **Forbudt:** Endre masterdata uten menneskelig godkjenning.
 - **Tillatt:** Foreslå endringer som krever 2FA-godkjenning fra en rolle med “dataeier”-tilgang.
 - **Teknisk:** Bruk Open Policy Agent (OPA) til å håndheve dynamiske regler (f.eks. “ingen dataendringer mellom 02:00–04:00”).



2. **Sårbarhet som designprinsipp:** Eldre systemer er per definisjon sårbare for autonome agenter. Løsning:

- **Sandboxing:** Kjør agenter i isolerte miljøer (f.eks. Kubernetes-namespaces) med rate-limiting (maks 10 API-kall/minutt mot eldre systemer).
- **Honeytokens:** Plasser "lokke-data" i systemene (f.eks. en falsk brukerkonto). Hvis agenten interagerer med den, utløses alarm.

3. **Kortsiktig vs. langsiktig makt:** RBAC er utilstrekkelig. Bygg en temporær IAM-bro med:

- **Tidsbegrensede tokens** (JWT med 15-minutters levetid, automatisk rotasjon).
- **Kontekstbasert autorisasjon** (f.eks. "agenten kan kun aksessere API X hvis den har mottatt et godkjent saksnummer fra system Y").

Blindsoner og undervurderte risikoer

- **Integrasjoner som svart boks:** Agenter vil oppdage uoffisielle API-endepunkter eller datakilder som ikke er dokumentert (f.eks. en CSV-fil som brukes til midlertidig datautveksling). Løsning: Kjør en "API-discovery"-fase før pilot, og blokker ukjente endepunkter.
- **Feilhandlinger som smittekilde:** En agent som misforstår en regel (f.eks. "slett gamle loggfiler" → "slett alle loggfiler") kan skape kaskadefeil. Løsning: Implementer automatisert rollback basert på hendelseslogg (f.eks. AWS CloudTrail + Step Functions for å reversere handlinger).
- **Eksterne API-er som trojanske hester:** Agenter kan utilsiktet eksponere sensitiv data via tredjeparts-API-er (f.eks. en agent som sender personopplysninger til en ekstern oversettelsestjeneste). Løsning: Bruk data masking (f.eks. Microsoft Presidio) før data forlater organisasjonen.

Kontrollmodell: Autonomi innenfor synlige grenser

1. **Policy-sandkasser:** Definer handlingskorridorer per agent-type (f.eks. "Finansagenten kan kun overføre beløp < 10 000 NOK uten godkjenning").
2. **Revisjonsspor som maktredskap:** Alle agenthandlinger logges i et uforanderlig format (f.eks. blockchain-lignende lagring med immudb). Inkluder:
 - Agentens motivasjon (prompt/regel som utløste handlingen).
 - Konsekvenser (endringer i data, eksterne API-kall).
3. **Eskaleringsmekanismer:** Hvis en agent når en sårbarhetsgrense (f.eks. prøver å aksessere et blokkert endepunkt), fryses den og krever menneskelig intervensjon.



Neste steg (konkrete handlinger)

1. Kjør en “Chaos Engineering”-workshop (1 dag):

- Inviter utviklere og sikkerhetsfolk til å aktivt prøve å bryte en dummy-agent (f.eks. få den til å slette data eller kalle eksterne API-er med sensitiv info).
- Dokumenter alle funn og oppdater policy-sandkassene.

2. Implementer en “Agent Zero”-pilot (3 måneder):

- Velg én agent med begrenset handlingsrom (f.eks. en “dokumentklassifiseringsagent” som kun kan lese og kategorisere filer).
- Bruk OPA + Kubernetes for sandboxing, og immudb for logging.
- Mål: Ingen uautoriserte handlinger og 100% sporbarhet.

3. Bygg en “Maktkartlegging”-rapport:

- Lag et diagram som viser:
 - Hvilke systemer agentene kan påvirke (fargekode etter risikonivå: rød = eldre systemer, grønn = moderne API-er).
 - Hvilke dataflyter som er mest sårbare (f.eks. “Agent X → Ekstern oversettelsestjeneste”).
- Del rapporten med ledelsen for å sikre bevissthet om maktfordeling.