



Debrief — mistralai/mistral-large-2512

- Instanser: 9
 - Tokens inn: 28,617
 - Tokens ut: 16,608
 - Kostnad: \$0.3350
 - Kjørt: 2026-04-15 17:30:09.442267+00:00
-

Sverm-debrief: Agentisk KI i heterogene systemlandskap

Konsensus

- Policy-as-Code som nødvendig kontrollmekanisme** Alle instanser understreker at **Open Policy Agent (OPA)** eller lignende runtime-policyer er uunngåelige for å håndheve autonomigrenser. Policyer må evalueres før handlinger utføres, med eksplisitte regler for tillatte systemer, datasett og API-kall. Eksempel: SI_001 og SI_008 foreslår OPA med økonomiske og juridiske begrensninger (f.eks. maks beløp per transaksjon).
 - Ephemeral credentials som maktbegrenser** Statistiske service accounts er en sikkerhetsrisiko. Konsensus er **tidsbegrensede tokens** (15–30 minutter) med scopede tilganger (f.eks. JWT med `exp`, `aud`, og handlingsspesifikke claims). SI_002 og SI_005 anbefaler **HashiCorp Vault + SPIFFE/SPIRE** for dynamisk identitetsutstedelse.
 - Execution Graph for sporbarhet og kontroll** Agentenes handlinger må logges som en **graf** med noder for prompts, beslutninger, API-kall og sideeffekter. SI_003 og SI_007 foreslår **OpenTelemetry + Neo4j** for å spore emergent behavior og kjedereaksjoner. Audit-loggen må være immuterbar (f.eks. AWS QLDB) for juridisk ansvarlighet.
 - Isolasjon og “kill switches” for resiliens** Agenter må kjøre i **ephemeral miljøer** (f.eks. Kubernetes + gVisor) med automatiske **circuit breakers** ved avvik (f.eks. 3 feilkall på rad). SI_004 og SI_009 understreker behovet for **manuelle “break glass”-mekanismer** for å stanse agenter umiddelbart.
 - Juridisk forankring av autonomi** Agentenes handlinger må knyttes til **lovhjemmel** eller eksplisitte fullmakter (f.eks. “Autonomi-Charter”). SI_004 og SI_009 advarer mot at manglende juridisk sporbarhet kan føre til ansvarsfraskrivelse ved feil (f.eks. GDPR-brudd).
-



Dissens

1. Kortsiktig vs. langsiktig tilnærming

- SI_002 (kortsiktig): Fokuserer på implementerbare grep innen 9 måneder (f.eks. OPA-pilot, ephemeral credentials).
- SI_008 (langsiktig): Argumenterer for strategisk venting og maktfordeling – agentisk KI krever en organisatorisk omstilling, ikke bare tekniske løsninger.

2. Tillit vs. kontroll

- SI_003: Ser på agentisk KI som et tillitsproblem – sårbarhet må eksponeres for å bygge tillit (f.eks. “tillitsdashboard” for brukere).
- SI_006: Ser på tillit som en økonomisk buffer – agentene må begrenses til det virksomheten har råd til å feile med (f.eks. kostnadsbudsjetter per agent).

3. Zero Trust vs. Zero Surprise

- SI_001 og SI_005: Zero Trust er nødvendig for å hindre uønskede handlinger.
- SI_008: Zero Trust er feil utgangspunkt – systemet må i stedet forvente avvik og designe for Zero Surprise (f.eks. runtime-policyer som blokkerer uventede handlinger).

Blindsoner avdekket

1. **Latent coupling i legacy-systemer** Flere instanser (SI_004, SI_007, SI_009) påpeker at agenter kan utløse skjulte avhengigheter mellom systemer som ikke er dokumentert. Eksempel: En endring i et HR-system kan trigge en batch-jobb i et økonomisystem. **Løsning:** Chaos Engineering (f.eks. Gremlin) for å teste uforutsette kjedereaksjoner.
2. **Emergent behavior og “reward hacking”** Agenter kan utvikle uforutsette strategier for å oppnå mål (f.eks. slette feilmeldinger for å “minimere feilrapporter”). SI_009 foreslår **reward shaping** i treningsdata for å straffe handlinger som reduserer sporbarhet.
3. **Dataens sårbarhet** Agenter som opererer på dårlig kvalitet eller inkonsistente data kan forsterke feil (f.eks. automatisk korrigere adressefelt basert på utdaterte registre). SI_003 anbefaler **data-kvalitetssensorer** som stopper agenter ved avvik.
4. **Menneskelig overstyring vs. autonomi** Hvor går grensen mellom autonomi og menneskelig kontroll? SI_006 foreslår en **tillitsmodell** der agenter får utvidet



autonomi basert på historisk adferd, mens SI_004 krever manuell godkjenning for alle handlinger i høyrisiko-systemer.

Anbefalinger

1. Pilot med “sikkerhetsbur” for én agent

- Velg en lavrisiko-prosess (f.eks. dokumentarkivering) og implementer:
 - **OPA** for runtime-policyer (f.eks. “ingen endringer utenfor arbeidstid”).
 - **Ephemeral credentials** (Vault + SPIFFE) med 15-minutters tokens.
 - **Execution Graph** (OpenTelemetry + Neo4j) for sporbarhet.
 - **Automatisk rollback** for feilhandling.
- Tidsramme: 3 måneder.

2. Kartlegg latent coupling med Chaos Engineering

- Bruk **Gremlin** eller lignende til å simulere agentfeil i legacy-systemer (f.eks. tilfeldige API-kall, manglende idempotens).
- Dokumenter alle uforutsette kjedereaksjoner og bygg compensating actions for å reversere dem.
- Tidsramme: 2 måneder.

3. Etabler en “Autonomi-Charter” med juridisk forankring

- Definer:
 - Maksimalt endringsnivå (f.eks. “ingen endringer i masterdata uten godkjenning”).
 - Tidsbegrensede mandater (f.eks. 30-dagers levetid for credentials).
 - Reverseringsplikt (f.eks. “alle handlinger må kunne tilbakestilles innen 4 timer”).
- Få charteret godkjent av juridisk avdeling før piloten starter.
- Tidsramme: 1 måned.

4. Bygg en “økonomisk audit”-pipeline

- Koble eksisterende logging (f.eks. ELK-stack) til et kostnadsdashboard (Grafana) som viser:
 - Agentens kumulative kostnad (f.eks. API-kall, databasetransaksjoner).
 - Risikoeksponering (f.eks. “høyrisiko-handlinger i rød sone”).



- Sett opp automatiske varsler for handlinger som overstiger budsjett (f.eks. "Agent X har brukt 80% av dagens API-budsjett").
- Tidsramme: 2 måneder.

5. Opprett en "maktkomité" for løpende risikovurdering

- Team med representanter fra juridisk, sikkerhet, fagsystemeiere og utvikling.
- Oppgaver:
 - Ukentlig gjennomgang av agentenes autonomi-grenser og risikoscorer.
 - Veto-rett over nye handlinger i høyrisiko-systemer.
 - Simulering av worst-case-scenarier (f.eks. "agent sletter alle fakturaer for 2023").
- Tidsramme: Påbegynnes umiddelbart.

Avslutning: Agentisk KI er en maktforskyvning som krever både tekniske og organisatoriske grep. Prioriter kontrollert autonomi – begrens agentene til det dere har råd til å feile med, og bygg systemer som forventer avvik. Start små, lær raskt, og skalér kun når tillit er etablert.