



id: SI\_005

model: qwen/qwen3.6-plus

seeds:

04\_risiko: sårbarhet

03\_relasjon: konkurranse

05\_handling: implementer

01\_perspektiv: strategisk

02\_tidhorisont: langsiktig

## Instans SI\_005

**Seeds:** strategisk | langsiktig | konkurranse | sårbarhet | implementer

Din utfordring er ikke mangel på verktøy, men arkitektonisk disiplin. **Strategisk** må du slutte å jakte nye plattformer og i stedet bygge en lukket, selvforsterkende kunnskapsflyt. **Langsiktig** vinner den som eier datastrukturen, ikke den som bare konsumerer strømmer. **Konkurranse** i sverm-orkestrering kommodifiseres raskt; din differensiering ligger i proprietær ruting, lav-latens beslutningsløkker og proprietær vekting av signaler, ikke i rå antall instanser.

**Sårbarheten** din er manuelle overganger og avhengighet til ustabile eksterne API-er (X, YouTube). Løsningen er en fire-rolles agent-sverm på din server:

1. **Scout** (Python + `yt-dlp` /RSS): Fanger opp nye kilder via GitHub, akademiske preprints og kuraterte X-liste-RSS-er. Unngå offisielle X-API-er.
2. **Filter** (Llama 3.1 70B via OpenRouter): Scorer relevans (0.0–1.0) mot din eksisterende KB. Dropper automatisk innhold under terskel.
3. **Synth** (Whisper-large-v3 + Mistral Large): Transkriberer, oppsummerer, ekstraherer metadata og skriver `.md`-filer direkte til en lokal SQLite + Qdrant cache.
4. **Sync** (Git-push): Commits og pusher til Obsidian-vaulten din via SSH. Ingen manuell import, ingen nettleser-API-avhengighet.

En lettvekt admin-side (Astro + SQLite) gir deg oversikt over agent-helse, token-kostnad og KB-vekst. Bruk webhook-triggere i stedet for cron-polling for å redusere rate-limit-**sårbarhet**.

**Implementer** dette som en pipeline, ikke et prosjekt. Kjør på Docker Compose med `langgraph` eller `crewai`. Bygg en lokal prompt-cache for hyppige ruter. Din KB må være kilde-kritisk: hver oppføring får en `confidence_score` basert på agent-samsvar og kildehistorikk.

**Neste steg:**

1. Setter opp 3-agent-pipeline (Scout→Filter→Sync) med OpenRouter-proxy. Test i 48 timer på kun 5 kilder.
2. Implementer git-basert Obsidian-sync fra server. Skriv et enkelt `post-receive` -hook som trigger vault-oppdatering.
3. Definer `relevance_threshold=0.72` . La systemet automatisk arkivere lav-score innhold. Evaluer nøyaktighet etter 7 dager og juster vektingen.