

AI Trinity Experiment: Karel

Prostředí e-INFRA CZ

- **Kubernetes** multi-tenant cluster
 - ~3500 CPU jader
 - uživatelé mohou používat kontejnery
 - musí uvádět požadavky na zdroje
 - namespace má kvótu
 - **overprovisioning** – z valné většiny jsou požadavky nadsazené → plýtvání kapacitou
- na clusteru se stále něco děje
 - aktualizace systému
 - aktualizace uživatelských objektů – zapomenuté objekty – ne vždy neškodné (ip fw)

Prostředí e-INFRA CZ

- **Ceph** clustery (jiný než DU CESNET)
 - součástí Kubernetes clusteru
 - Rook Ceph (20.2.4)
 - CephFS/S3/NFS exporty
- Některé jeho komponenty potřebují „**babysitting**“
 - Chyby komponent
 - Akutní opravy (a opravy oprav)

AI v e-INFRA CZ a agenti

- On-prem inferenční služba pro členy
 - Kimi-K3, GLM-5.3, DeepSeek 4.1 Flash (**1M kontext**)
- Bezpečná inference z hlediska datové suverenity
- Přiměřeně schopné modely
- Podpora **agentických** nástrojů
 - Claude Code, OpenCode, Hermes

Karel monitorující

- Prostředí
 - **OpenCode** agent – „no ask“
 - Pod **sandbox**
 - Read-only **RBAC** vyjma **Secret** objektů
 - **MCP Server**
 - Prometheus
 - K8s na filtrované Secret objekty
 - Request Tracking – uživatelská podpora

Karel monitorující

- Kombinace skriptů a AI
- **Skripty**
 - Stažení prometheus metrik (trvá dlouho, agent nemá trpělivost)
 - Stažení informací o expirovaných účtech
 - Stažení informací o aktualizaci GPU ovladače
- **AI**
 - Provádí další kroky a skládá zprávu

Karel monitorující – AI

- Generuje Tarot (posledniping.cz)
- Sestavuje zprávu o využití zdrojů na základě dat z Prometheus
- Sestaví seznam a provede analýzu všech neběžících Podů

Bad pods

The count jumped from 51 to ~73, because of course it did. Bitnami deleted tags, PostgreSQL found seven new ways to suffer, and the whole-GPU pool remains a private club with a waiting list.

- **Scope:** ~73 bad pods; **22 identified namespaces** (21 named + prometheus-rancher-monitoring inferred), plus ~25 more pods in sections without namespace labels (mostly AlphaFold/OmegaFold GPU jobs, and qdrant/redmine/regstar/etc.).
- **Breakdown by root cause (by pod):** scheduling / GPU exhaustion 14 (all Pending, whole-GPU pool full) · CrashLoopBackOff 23 (config/permissions, bad images, cascades) · ImagePullBackOff 11 (8 missing/removed tags, 3 registry auth rejections) · Evicted on ephemeral storage 12 (all cf-prot-ns) · OOMKilled 4 (rationai xai-framework) · other 9 (missing secrets/config ×4, failed jobs ×3, collateral not-ready ×2).
- **Top 5 namespaces:** cf-prot-ns (12), gdi-ega-ns (6), rationai-jobs-ns (5), masar-ns (3), sadigzade-ns (3).
- **Patterns worth flagging:**
 - **Same image failure across 5 namespaces:** Bitnami/Broadcom deleted `docker.io/bitnami/*` tags (postgresql ×3, mongodb, kubectl) — one fix each: retag to `bitnamilegacy/` or mirror into Harbor.
 - **GPU capacity exhaustion** is the top Pending cause: every whole A100-80GB/H100 GPU is allocated; kub-b7/b8 carry A100 labels but expose 0 whole GPUs (MIG-only), shrinking the affinity pool.
 - **PostgreSQL is the most-affected component** (7 namespaces: tag removal, data-dir permissions, PG17→18 version drift, lost PVC data, expired CA); 8 pods are collateral cascades of 4 primary failures (RabbitMQ cookie perms, harbor-db, bookhub-db, student-db).
 - **Expired/stale credentials recur** across the cluster: GitLab/GHCR pull tokens, an e-INFRA API token, and a CNPG CA certificate all lapsed without alerting.

Karel monitorující – AI

- Sledujeme opuštěné Service, Ingress, ACME, Hello Kubernetes
- Neřešené uživatelské požadavky
- Zpráva o celém stavu každé ráno
- **Problémy**
 - Některé akce přes AI lze i naskriptovat (ale kdo by to dělal)
 - Nebezpečí prompt injection – nevyřešeno

Karel nasazující

- Jak dostat github **projekt** do **Kubernetes**?
 - Sestavit Docker Image
 - Napsat manifesty
 - Analyzovat chybu
 - Opakovat od prvního kroku
- Al má Kubernetes a Docker hodně dobře nastudovaný

Karel nasazující

- **Hermes** agent
- **Skills:** <https://github.com/CERIT-SC/infra-skills>
 - Jak sestavit správně Docker Image
 - Co musí splňovat K8s manifesty
 - Security Context
 - Ingress class
 - Storage class

Karel nasazující

- Stažení Github projektu
- Spuštění hermes agenta
- **Příkaz:** seznam se s projektem, nasad' ho do Kubernetes
- **Agent:**
 - Prozkoumá projekt, zda obsahuje Dockerfile nebo K8s manifesty
 - Pokud nejsou, vytvoří nové
 - Ve smyčce sestavuje a testuje, zda aplikace běží

Karel nasazující

- **Sandbox**

- Běží na serveru pod vlastním uživatelským účtem
- Má R/W práva na jeden vlastní K8s namespace
- Má push práva na vlastní projekt v docker registry
- Je schopen sestavit docker image
- Je schopen pustit docker (test), i když nebezpečné

- **Do budoucna**

- Pod sandbox
- Problém s Docker runnerem v Podu

Karel ukládající

- **Sada skriptů** monitorující CEPH
 - Několik tierů testů dle závažnosti
 - Stav OSD, replikace, zaplněnost, scruby, stav NFS
- **Znalostní báze**
 - Dokumentace
 - Issues z trackeru
- Kompilovaná zpráva nálezů a možnosti řešení
- Upozornění na problémy, které se nás mohou týkat

Karel ukládající

- **Ve vývoji**
- Cílový stav jsou PR do infrastrukturního gitu
- Co největší autonomie, ale stále human-in-the-loop
- AI testováno na reálných chybách provozu
 - Analýza problému a návrh řešení = 100% úspěšné
- **Lobotomie adminů?**
 - Google+Askslashdot → AskAI

Děkuji za pozornost!

k8s@cerit-sc.cz

<https://docs.cerit.io>

<https://blog.e-infra.cz>