

```
moi@opscraft ~ $ whoami
```

Mohamed Ghenaim

Architecte Cloud AWS, DevOps & SRE freelance. Build, run, debug depuis 9 ans.

AWS

Terraform

Kubernetes

IA / LLM

GitOps

CI/CD

Docker

Linux

```
moi@opscraft ~ $ fastfetch --logo-type kitty
```



focus

email

linkedin

lieu

langues

uptime

Plateformes d'IA générative : LLM on-prem air-gapped et AWS EKS

mohamed.ghenaim@opscraft.fr

in/ghenaim-mohamed

Bois d'Arcy (78), Île-de-France ou à distance

Français, anglais courant (C2, TOEIC 915)

9 ans en production



CA-GIP, Crédit Agricole Group Infrastructure Platform

Architecte Cloud & DevOps, plateforme IA générative souveraine (Mistral AI)

Référent technique du build de la plateforme GenAI Mistral AI du groupe, on-premise sur OpenShift air-gapped (GPU NVIDIA B300) et sur AWS EKS. Moteur de la conception et de la mise en production, dans une équipe de quatre.

@@ [on-prem/openshift-airgapped](#) @@

- + Déploiement et montées de version de la stack Mistral AI en sandbox, DEV et PROD, chart Helm adapté à OpenShift.
- + Migration d'ArgoCD vers FluxCD à mon initiative : organisation GitOps par application, documentation, montée en compétence de l'équipe.
- + Déploiement et optimisation des modèles avec vLLM (Mistral Small 4, Medium, Large, OCR, Voxtral, Embed).
- + Benchmarks des endpoints LLM avec GuideLLM et identification des goulots d'étranglement.
- + Observabilité des nodes, des workloads, de l'inférence vLLM et des GPU (Tensor Cores, température, consommation) sous Grafana avec alerting, Dynatrace et Kibana.
- + Bases PostgreSQL CNPG sauvegardées vers S3 on-prem, restaurations testées et validées.
- + Stockage des modèles et des backups migré vers S3 on-prem, synchronisation automatisée des modèles en air-gapped.
- + Secrets avec HashiCorp Vault et External Secrets, chiffrement des PVC.
- + Redis in-cluster sur image construite en CI GitLab, en remplacement d'un service managé inadapté.
- + Mise en production des clusters applicatif et d'inférence : flux réseau, validation sécurité, socle, avec les équipes réseau, sécurité et l'éditeur.
- + Remise en service d'une application RAG interne après un incident de perte de données.

@@ [aws/eks](#) @@

- + Stack Mistral AI déployée sur EKS sur tous les environnements, de la DEV à la PROD.
- + Exposition via Network Load Balancers, étude d'EKS Auto Mode.
- + RDS Proxy pour le pooling des connexions aux bases RDS.
- + Création des comptes utilisateurs Keycloak automatisée.
- + Secrets dans AWS Secrets Manager, chiffrés par une clé KMS adossée à CloudHSM.

Stack: openshift, eks, helm, fluxcd, vllm, guidellm, grafana, dynatrace, cnpg, vault, keycloak, kms, cloudhsm

**epex-spot**

juil. 2025 - sept. 2025

EPEX SPOT

Architecte Cloud AWS, équipe Fondation

Architecture et déploiement des briques AWS critiques d'une plateforme en cours de migration.
Modules Terraform standardisés pour les équipes DevOps.

- + DNS hybride Route 53 : DNS Hub central en Terraform, relié au datacenter par Direct Connect et DX Gateway, partagé via Organizations et RAM. Routage asymétrique résolu avec Network Firewall.
- + PKI managée AWS PCA : hiérarchie Root et Subordinate CA en Terraform, partage cross-account, renouvellement automatique ACM.
- + Infrastructure GitLab Runners reconstruite en IaC après un incident majeur, sans interruption des déploiements.
- + Module runners GitLab sur EC2 : manager et workers autoscalés (plugin fleeting AWS), AMI Debian standardisée avec Packer.
- + Catalogue de modules Terraform : gitlab-runner, pki-certificate, dns-record, ecr-repository, codeartifact-repository.
- + Arrêt et démarrage des ressources par tags via Lambda, sur les heures ouvrées : **-60 %** de coût EC2.
- + Documentation d'architecture (ADR), guides opérationnels et montée en compétence AWS de l'équipe.

Stack: terraform, route53, direct-connect, network-firewall, aws-pca, acm, gitlab-ci, packer, lambda

**cLaranet**

déc. 2022 - janv. 2025

Claranet France

CloudOps AWS, BU Cloud Native

Build et run de production sur des environnements AWS multi-comptes pour **11** clients : fintech, immobilier (CAC 40), streaming, IoT, pharma, banque, e-commerce.

- + Support N3 et astreinte de nuit sur l'ensemble du portefeuille.
- + Stack ECS complète : images Docker PHP, IAM, ECR, task definitions, RDS, ALB, Route 53, ACM, secrets HashiCorp Vault.
- + Monitoring Splunk et OpenTelemetry sur EC2 et ECS.
- + Patch management : AMI migrées d'Ubuntu vers Debian, mise à jour des GitLab Runners et de Docker Machine.
- + Playbook Ansible de déploiement applicatif sur EC2.

Stack: ecs, ecr, rds, alb, route53, vault, splunk, opentelemetry, ansible



orange

oct. 2020 - déc. 2022

Orange France

Ingénieur production Linux / DevOps, Service Commun Portail

- + Build du service de redirection d'URL à fort trafic : migration Ubuntu 14.04 vers 18.04, Apache, adressage IP, playbook Ansible.
- + Support N3 de la production du Portail Orange.
- + Templates de pipelines GitLab CI mutualisés pour toutes les équipes du SI.
- + Migration de services vers Kubernetes et CloudFoundry, dashboards Kibana de suivi de production.
- + Application DAM (Nuxeo) migrée sur Docker Swarm : Dockerfile optimisé, pipelines CI/CD, certificats HTTPS.

Stack: apache, ansible, gitlab-ci, kubernetes, cloudfoundry, docker-swarm, kibana



malakoff-humanis

sept. 2017 - sept. 2020

Malakoff Humanis

Ingénieur production Linux Infrastructure, équipe CSI/Linux

- + Administration de l'infrastructure Linux : Bash, Nagios, LDAP, Kerberos.
- + Cluster ELK complet (Elasticsearch, Logstash, Kibana, Kafka, Zookeeper) sur Docker Swarm.
- + Automatisation CI/CD avec Rundeck, Ansible, Docker, GitLab et Jenkins. ITSM avec GLPI et FusionInventory.
- + RedHat Satellite (canaux, hôtes) et Identity Manager.

Stack: rhel, elk, kafka, docker-swarm, rundeck, ansible, jenkins, satellite

Stack

stack.tf

\$ ls ~/.logos



AWS

EKS, Lambda, KMS,
CloudHSM, multi-
comptes



Terraform

Modules réutilisables,
IaC de toute l'infra



Kubernetes

OpenShift, EKS, K3s,
GitOps



Docker

Images custom, Swarm,
build en CI



Linux

Debian, Ubuntu, RHEL,
depuis 2017



Mistral AI

vLLM, plateforme GenAI
souveraine

```

1  # Compétences, par domaine
2  resource "skills" "cloud_iac" {
3      aws    = ["EKS", "Secrets Manager", "KMS", "CloudHSM"]
4      tools  = ["Terraform", "Packer", "Ansible", "Hetzner (dédié)"]
5  }
6
7  resource "skills" "containers" {
8      runtime = ["Kubernetes", "OpenShift", "K3s", "Docker"]
9      gitops  = ["Helm", "FluxCD", "ArgoCD"]
10 }
11
12 resource "skills" "network_security" {
13     mesh    = ["Cilium", "Linkerd"]
14     secrets = ["Vault", "External Secrets", "SOPS", "Infisical", "Keycloak"]
15 }
16
17 resource "skills" "ai_llm" {
18     serving = ["vLLM", "GuideLLM", "LiteLLM", "Mistral AI", "Ollama"]
19     usage   = ["RAG", "pipelines d'embedding", "n8n AI Agents"]
20 }
21
22 resource "skills" "data" {
23     databases = ["PostgreSQL (CNPg, Barman)", "Redis", "MongoDB", "S3"]
24     vectors   = ["Qdrant", "pgvector"]
25 }
26
27 resource "skills" "observability" {
28     tools = ["Grafana", "Prometheus", "Datadog", "Dynatrace", "Splunk", "New Relic", "ELK"]
29 }
30
31 resource "skills" "ci_cd" {
32     tools = ["GitLab CI", "Artifactory", "Rundeck", "n8n"]
33 }
34
35 resource "skills" "ai_assisted_dev" {
36     claude_code = ["Rust", "Python / FastAPI", "TypeScript / React", "Tauri", "Alpine.js"]
37 }

```

Plan: 8 to add, 0 to change, 0 to destroy.

Projets personnels

projets/

README.md : conçus, architecturés, déployés et exploités par mes soins. Les applications sont développées avec Claude Code.

Plateforme Kubernetes GitOps auto-hébergée

Mon terrain d'essai permanent : un serveur dédié Hetzner sous K3s, construit et opéré à la main depuis 2022, et reconstruit à chaque fois que mes missions m'apprenaient une meilleure façon de faire. Tout est décrit dans Git, déployé par ArgoCD, avec des secrets chiffrés et un runbook à jour.

30

releases Helm déclarées dans Helmfile

26

applications synchronisées par ArgoCD

3

générations d'architecture

12 mois

de sauvegardes : quotidiennes, hebdo, mensuelles

v5

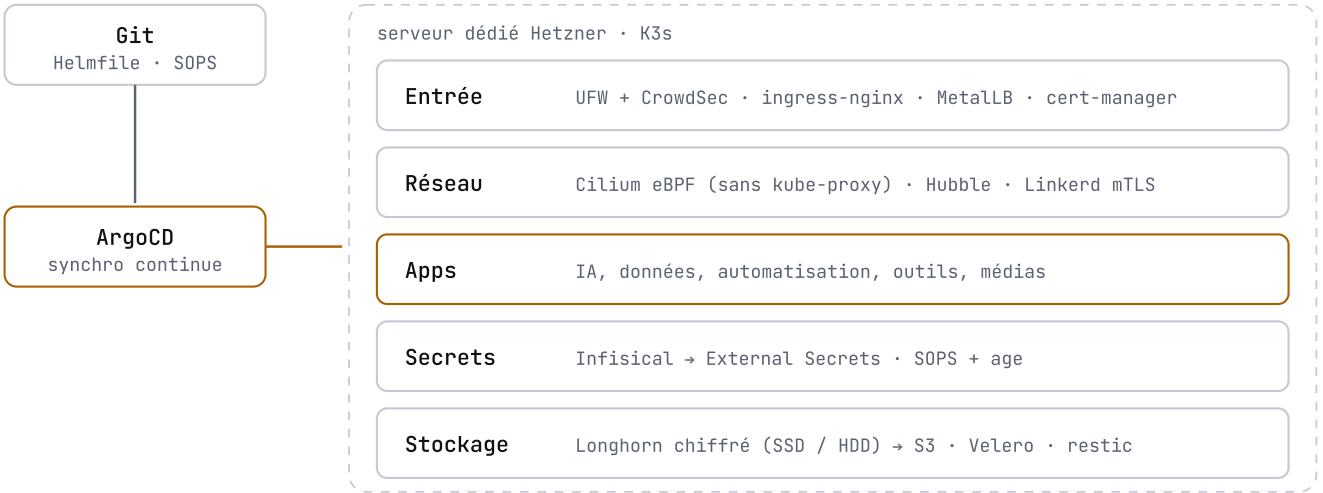
Manifests et Helm
K3s, Traefik, Prometheus et Grafana, premiers services auto-hébergés.

v6

GitOps
Helmfile, ArgoCD, secrets SOPS + age, hooks pre-commit, runbook Wiki.js.

v7

Essai tout-Terraform
Chaque app en OpenTofu. Leçon : Terraform pour l'infra, GitOps pour le cycle de vie des apps.



```
$ argocd app list
GROUPE      APPLICATIONS                                     SYNC
network     cilium, linkerd, ingress-nginx, metallb         Synced
security     infisical, external-secrets, cert-manager, crowdsec Synced
storage      longhorn, velero                                Synced
data         pgvector, qdrant, mongodb                       Synced
ai           open-webui, ollama, n8n                         Synced
tools        gitea, wiki.js, hedgedoc, affine, matomo         Synced
tuning       vpa, goldilocks                                 Synced
media        media-center                                    Synced

# extrait, 26 applications au total
```

Sauvegardes et reprise

Snapshots Longhorn et Velero vers Hetzner Object Storage, fichiers via restic. Rotation quotidienne, hebdomadaire et mensuelle, procédures de restauration documentées dans le runbook.

- k3s
- argocd
- helmfile
- sops
- cilium
- linkerd
- longhorn
- velero
- infisical
- crowdsec
- n8n

IA sur les logs

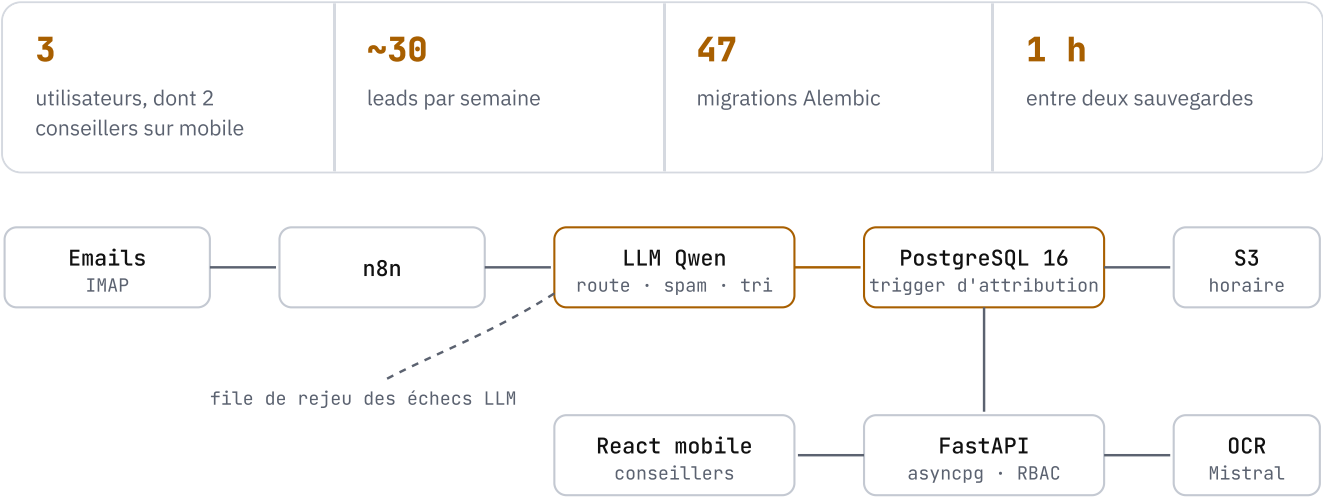
POC de RAG sur les logs système et d'ingress : les logs sont vectorisés (embeddings), stockés dans Qdrant et pgvector, puis interrogés via un LLM. En parallèle, un workflow n8n lit les logs d'ingress, les confronte aux rapports précédents et produit un rapport avec score d'urgence, commité dans Git et signalé sur Discord.

projets/crm-immo/

en prod depuis avr. 2026

CRM métier en production, PME de l'immobilier neuf

Un CRM web qui remplace un suivi Excel. Je l'ai conçu, développé et déployé, et je l'exploite seul : du schéma de base à l'exploitation, en passant par le mobile des conseillers sur le terrain.



Applicatif

FastAPI (asyncpg, Pydantic v2), PostgreSQL 16, React 18 et TypeScript (TanStack Query, Tailwind), pensé mobile d'abord. RBAC par rôle, exports Excel et PDF.

Déploiement

Prod en docker-compose sur Dedibox Scaleway, préprod sur K3s (CNPG, Longhorn). Images rootless Buildah, healthchecks, smoke tests, procédure de rollback.

- fastapi
- postgresql
- alembic
- react
- typescript
- n8n
- llm
- buildah
- pytest

IA et automatisation

Leads ingérés et classés par LLM, OCR des justificatifs, attribution pondérée aux conseillers directement en base.

Résilience et qualité

Backups horaires vers S3 avec alerte Discord en cas d'échec. Tests pytest sur une base PostgreSQL éphémère dans K3s, vittest côté front.

Manga Shelf, liseuse de manga serverless

Une liseuse en production, 100 % serverless sur le Free Tier AWS, infrastructure entièrement en Terraform. Pendant la lecture, les images passent par CloudFront avec des cookies signés : aucune Lambda n'est sollicitée.

34

routes API Gateway

7

tables DynamoDB

0

Lambda pendant la lecture

Free Tier

100 % serverless



Site et API

Lambda Rust arm64, Cognito PKCE avec groupes admin, clé privée CloudFront dans SSM. Front PWA en Alpine.js, tests UI Puppeteer et tests de charge k6.

Application d'administration

Application desktop Rust (WebView2) : publication vers S3 et invalidation CloudFront, client BitTorrent intégré, tunnel WireGuard userspace, titres de chapitres extraits par OCR via l'API Claude.

rust

lambda arm64

api-gateway

dynamodb

cognito

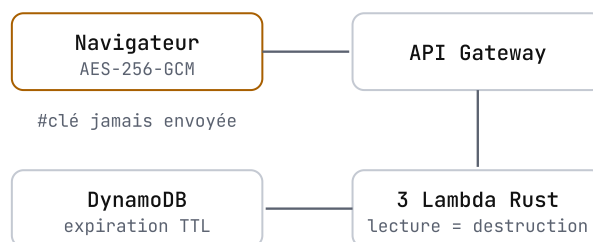
cloudfront

terraform

k6

Secret Vault, secrets à usage unique

Chiffrement zero-knowledge dans le navigateur (AES-256-GCM, PBKDF2 à 200 000 itérations). La clé reste dans le fragment d'URL et n'atteint jamais le serveur. Trois Lambda Rust, destruction atomique à la lecture, expiration par TTL DynamoDB, throttling API Gateway, tests E2E Playwright.



rust

webcrypto

dynamodb ttl

playwright

projets/pulse-guard/

en prod

Pulse Guard

Monitoring d'URLs chaque minute via EventBridge et une Lambda Rust : statut HTTP, latence, expiration TLS. Alertes Discord, page de statut publique, authorizer JWT Cognito natif d'API Gateway.

rust

eventbridge

cognito jwt

discord

projets/desktop-tools/

depuis mars 2026

Outils desktop Windows en Rust

Chuchotement : dictée vocale push-to-talk, reconnaissance 100 % locale (Parakeet).

Note de frais : Tauri 2, OCR des tickets par l'API Claude, rapports PDF et Excel.

Texto : correcteur en streaming, diff mot à mot, coût suivi par requête.

Recommandations

reviews.md

✓ approved

Florent Michel Head of SysOps / Product Owner, EPEX SPOT

« Mohamed est intervenu chez nous pour un support ponctuel sur AWS afin de nous aider à respecter des deadlines critiques. Malgré la durée limitée de sa mission, il a su démontrer un haut niveau de compétence et d'efficacité. Son expertise, sa réactivité et son implication ont été des atouts indéniables pour la réussite de nos projets. »

✓ approved

Grégory Daudin Product Thinking & Agile Delivery, EPEX SPOT

“He played a key role in helping us catch up on significant AWS technical debt and even saved us from a really complex situation that could have blocked critical progress. We would have loved to keep him longer in the team. It was a pleasure!”

Formation

formation.txt

2018 - 2020

Master of Science (M2) Cybersécurité ITESCIA, Cergy

2017 - 2018

Licence Informatique Générale CNAM

2015 - 2017

BTS Services Informatiques aux Organisations Plaisir

```
$ ./contact.sh
```

Une plateforme à construire, ou à remettre d'aplomb ?

Architecture AWS, Kubernetes, IA générative : décrivez le contexte par email.

servi par S3 et CloudFront, déployé avec terraform apply

Polices JetBrains Mono et IBM Plex Sans sous licence SIL OFL 1.1. Logos issus du dépôt dashboard-icons (licence Apache-2.0) ; les marques appartiennent à leurs propriétaires respectifs.