

PUBLIC PROJECT REPORT

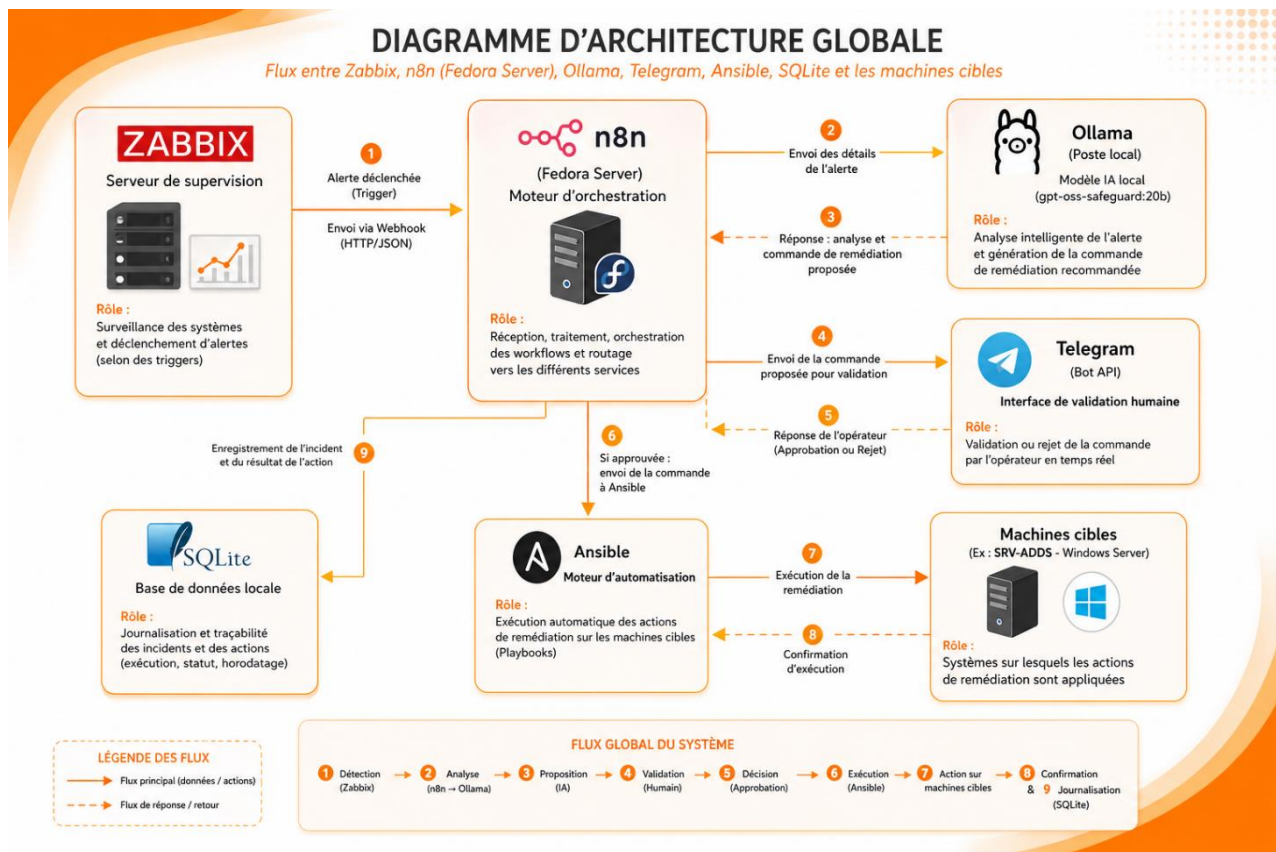
AIOps Human-in-the-Loop

Supervision proactive et auto-remediation controlee

Zabbix → n8n → Ollama → Telegram → Ansible → SQLite

Projet de fin d'etudes realise dans le cadre du stage chez 3GCOM

Achraf El Mouchtari | OFPPT | Infrastructure Digitale — Systemes et Reseaux | 2025-2026



PUBLIC SCOPE

Cette edition exclut les identifiants, credentials, adresses internes et endpoints prives.

Resume executif

Une solution AIOps locale pour reduire le temps de resolution sans supprimer le controle humain.

Le projet repond a une limite operationnelle classique: les outils de monitoring detectent rapidement un incident, mais le diagnostic, la validation et la remediation restent souvent manuels. Cette rupture augmente le MTTR et limite la tracabilite des interventions.

La solution construite orchestre une chaine event-driven complete. Zabbix detecte l'incident et emet un webhook vers n8n. Un modele Ollama local analyse l'alerte et propose une action. L'operateur valide ou rejette la proposition via Telegram. En cas d'approbation, Ansible execute la remediation, verifie l'etat cible et SQLite conserve l'historique.

2–5 min

MTTR apres

32 s

Webhook reçu

0 changed

Test idempotence

RESULTAT

Reduction du MTTR estimee a 85–90% pour les incidents de type service arrete dans le laboratoire.

- Architecture open-source et modulaire.
- Diagnostic IA execute localement, sans API cloud tierce.
- Validation humaine obligatoire avant l'action corrective.
- Journalisation technique exploitable pour l'audit et l'amelioration continue.

Probleme et objectifs

Transformer une alerte brute en action controlee, verifiee et historisee.

Le contexte de stage concerne un environnement IT virtualise associant Windows Server, Fedora, Zabbix et des outils d'automatisation. La chaine de traitement initiale reposait encore sur plusieurs etapes manuelles entre la detection et la resolution.

15–40 min MTTR initial	Manuel Diagnostic	Partielle Tracabilite
----------------------------------	-----------------------------	---------------------------------

Objectifs fonctionnels

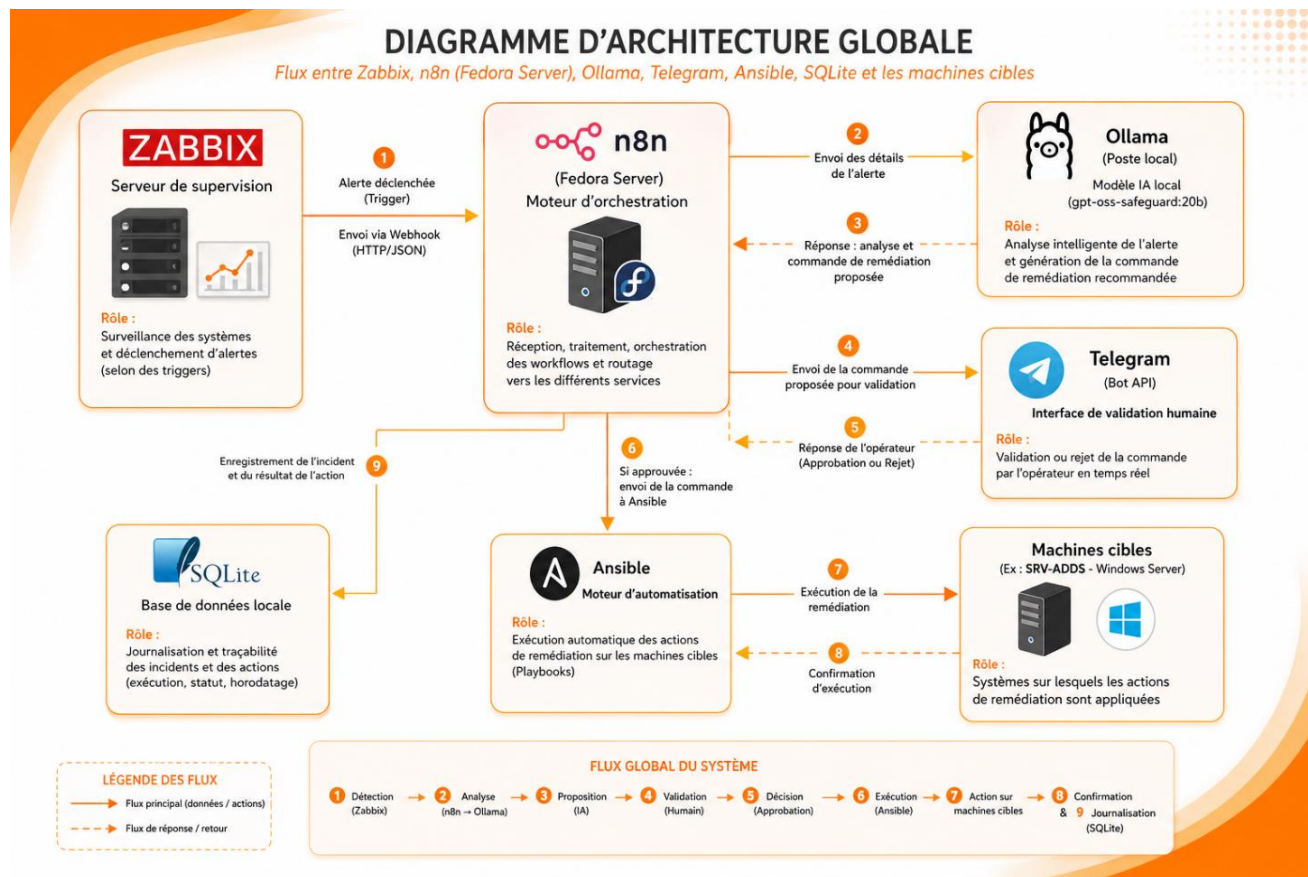
- Detecter les incidents affectant des services critiques avec Zabbix.
- Normaliser les alertes et orchestrer les decisions dans n8n.
- Generer un diagnostic assiste par un LLM local Ollama.
- Soumettre toute action critique a la validation de l'operateur via Telegram.
- Executer des playbooks Ansible idempotents et verifier le resultat.
- Journaliser l'incident, l'action et le statut final dans SQLite.

Contraintes non fonctionnelles

- Souverainete des donnees
- Disponibilite de la chaine
- Securite des acces
- Extensibilite vers de nouveaux incidents

Architecture globale

Une architecture hybride, orientée événements et centrée sur l'orchestration n8n.

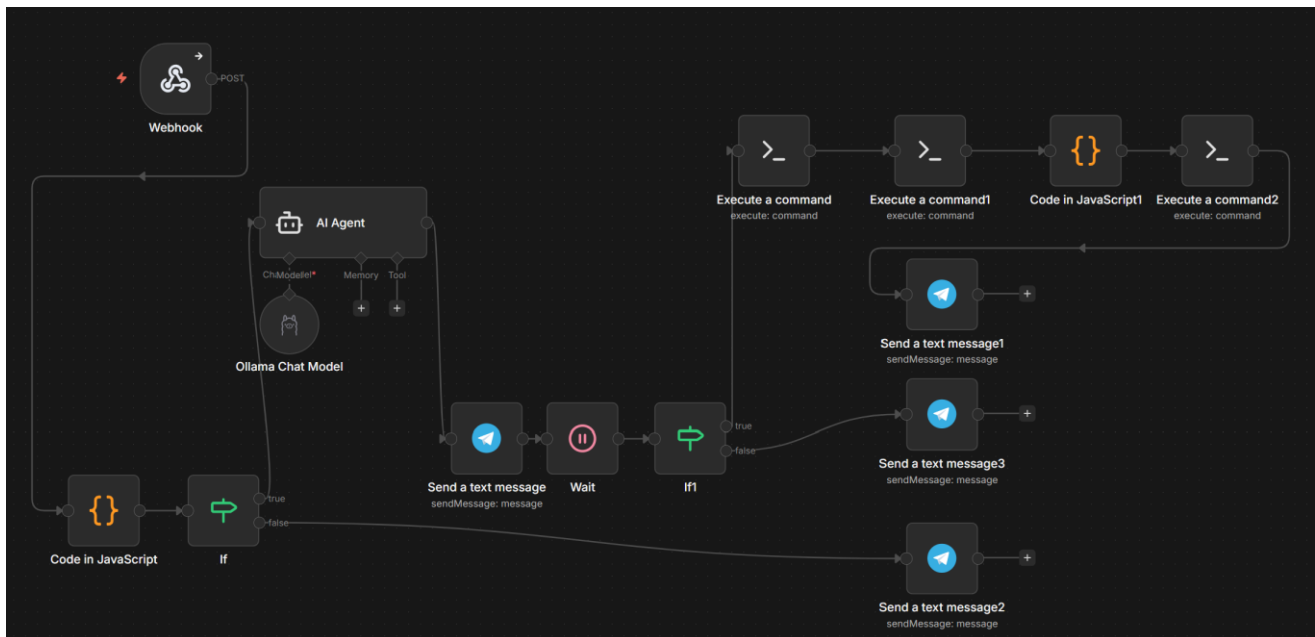


PRINCIPE

Chaque composant a un rôle unique; n8n coordonne les transitions, les branches et les erreurs.

Cycle de traitement de l'incident

Detection, diagnostic, validation, execution, verification et journalisation.



1 Detection

Zabbix declenche un webhook HTTP/JSON.

4 Decision

L'opérateur approuve ou rejette via Telegram.

2 Normalisation

n8n extrait l'hôte, le service, la sévérité et le message.

5 Execution

Ansible applique et vérifie la remédiation.

3 Diagnostic

Ollama formule une proposition structurée.

6 Audit

SQLite archive l'état final et le délai de résolution.

Stack technique et responsabilites

Des composants specialises relies par un workflow explicite.

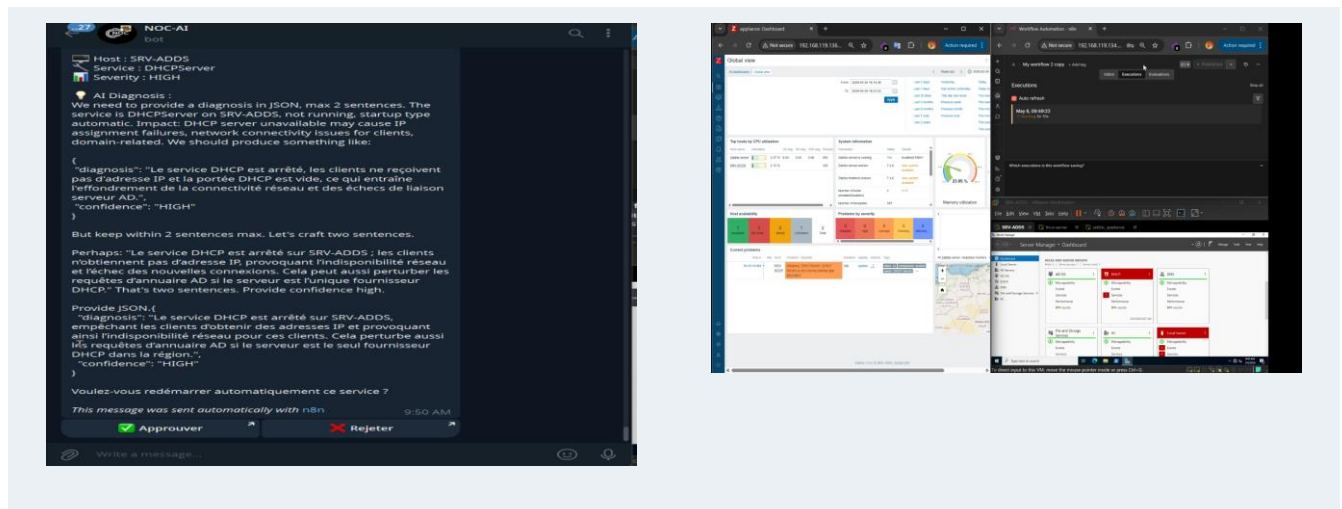
COMPOSANT	RESPONSABILITE	IMPLEMENTATION
Zabbix	Monitoring	Items, triggers et emission du webhook.
n8n	Orchestration	Parsing JavaScript, routage, attente et reprise.
Ollama	IA locale	Diagnostic assiste et sortie structuree.
Telegram	Decision humaine	Approbation ou rejet en temps reel.
Ansible	Remediation	Execution corrective idempotente et verification.
SQLite	Traceabilite	Historique des incidents et statut final.

Choix d'architecture

- Ollama reste local afin de conserver les donnees techniques dans l'environnement maitrise.
- n8n apporte une orchestration visuelle, des branches conditionnelles et des etats suspendus.
- Ansible est prefere a l'execution distante directe pour l'idempotence, les erreurs et l'auditabilite.
- SQLite suffit au volume du laboratoire sans imposer un serveur de base de donnees dedie.

Diagnostic local et validation humaine

L'IA assiste la decision; elle ne remplace pas l'administrateur.



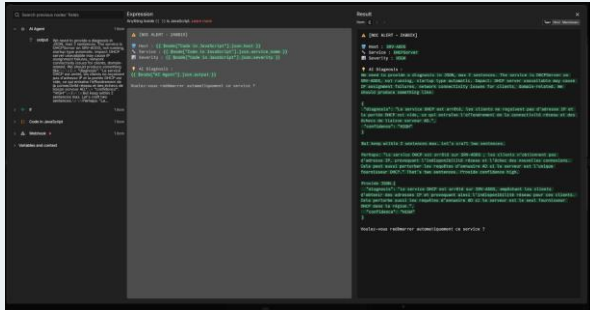
Le noeud AI Agent envoie au modele local une alerte normalisee. Le resultat est converti en proposition lisible, puis envoie a l'opérateur. Le workflow reste suspendu jusqu'a la decision ou jusqu'au timeout configure.

SECURITY GATE

Aucune action corrective n'est exécutée sans approbation explicite.

Remediation idempotente et tracabilite

Executer uniquement l'action necessaire, verifier, puis conserver la preuve.



```
sqlite> SELECT id, timestamp, host, service_name, severity, final_status
FROM incidents
ORDER BY id DESC
LIMIT 10;
id timestamp          host      service_name severity final_status
--
8 2026-05-06 08:51:30 SRV-ADDS DHCPServer HIGH Fixed
7 2026-04-07 15:10:45 SRV-ADDS DNS HIGH Fixed
6 2026-04-07 14:54:38 SRV-ADDS DHCPServer HIGH Fixed
5 2026-04-07 14:54:03 SRV-ADDS iphlpsvc HIGH Fixed
4 2026-04-07 06:47:15 SRV-ADDS DHCPServer HIGH Fixed
3 2026-04-03 23:58:51 SRV-ADDS DHCPServer HIGH Fixed
1 2026-04-01          SRV-ADDS dns high running
sqlite>
```

- Ansible controle l'etat du service avant modification.
- Une deuxieme execution sur un etat sain retourne OK avec 0 changed.
- Une verification post-remediation declenche la branche d'echec si le service reste arrete.
- SQLite conserve l'horodatage, l'hote, le service, la severite, le statut final et le MTTR.

Mesures de controle et gestion des erreurs

La securite est integree a la conception de la chaine d'automatisation.

Validation humaine

Blocage du workflow avant chaque remediation critique.

IA locale

Absence d'appel a une API IA cloud pour le diagnostic.

Credentials n8n

References conservees dans le gestionnaire de credentials,
pas dans le workflow public.

Validation de commande

Controle du format attendu avant execution.

Timeout

Fin controlee et notification si aucune decision n'est recue.

Fallback

Branche d'echec en cas d'indisponibilite ou de remediation
incomplete.

PUBLIC VERSION

Les chat IDs, adresses IP, tokens, credential IDs et
URLs internes ont ete supprimees ou remplacees.

Methodologie et scenario de test

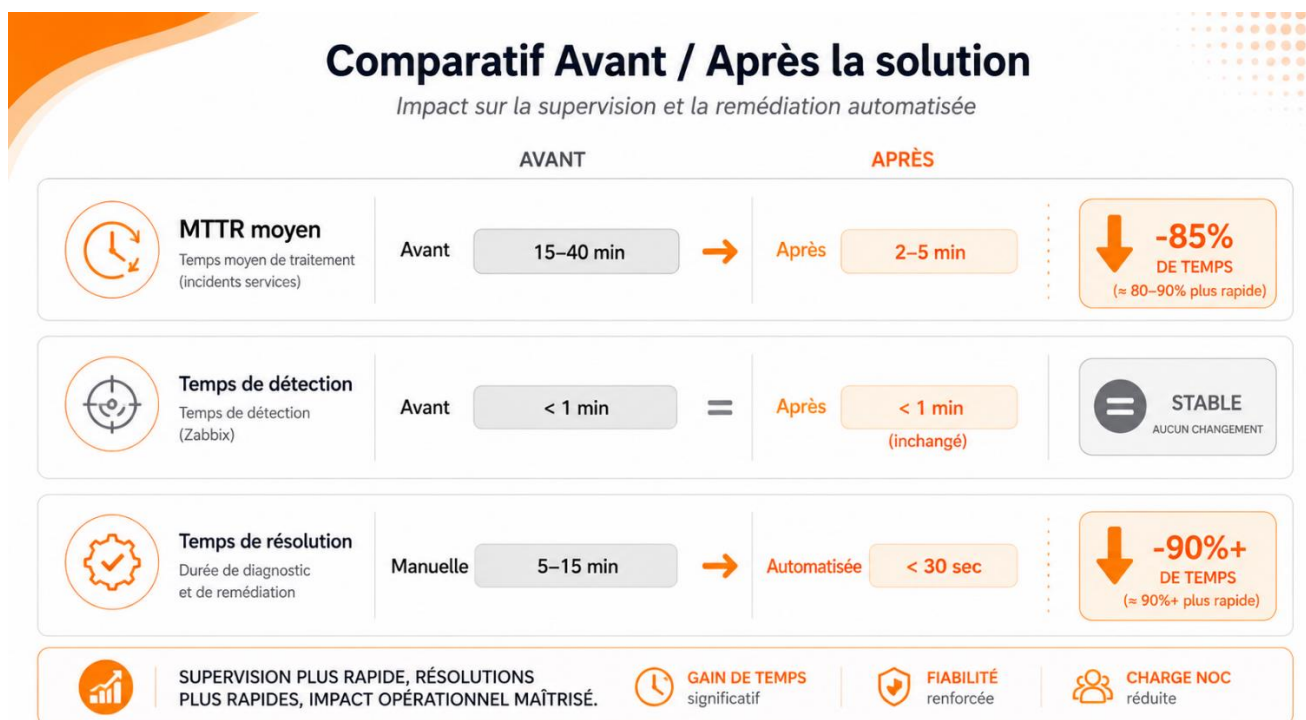
Un incident reproductible permet de valider chaque transition du workflow.

Scenario principal: arret controle du service DHCP Server sur SRV-ADDS.

ETAPE	CONTROLE	CRITERE
1	Simulation	Arret du service cible dans le laboratoire.
2	Detection	Observation du trigger Zabbix et du webhook n8n.
3	Parsing	Controle de l'hote, du service, de la severite et du flag de remediation.
4	Validation	Reception du diagnostic et reprise du workflow apres approbation.
5	Remediation	Execution du playbook et verification de l'etat final.
6	Idempotence	Deuxieme execution lorsque le service est deja sain.
32 s	< 2 s	0 changed
Alerte vers webhook	Notification Telegram	Seconde execution

Comparaison avant / apres

La detection reste rapide; le principal gain concerne le diagnostic et la remediation.



- MTTR moyen ramene de 15–40 minutes à 2–5 minutes.
- Réduction estimée de 85–90% sur les incidents de type service arrêté.
- Disponibilité continue de la chaîne de traitement dans le laboratoire.
- Historique SQLite exploitable pour l'audit et le suivi des performances.

INTERPRETATION

Le projet accélère la réaction sans automatiser aveuglément la décision critique.

Limites et perspectives

Un laboratoire fonctionnel qui peut evoluer vers une plateforme industrialisee.

LIMITE	EVOLUTION PROPOSEE
Dependance au poste Ollama	Deployer le modele sur un serveur dedie ou ajouter un fallback.
Perimetre limite	Ajouter des playbooks pour le disque, la replication AD, le reseau et les fichiers.
Boucle Zabbix incomplete	Notifier ou cloturer l'incident via l'API Zabbix apres remediation.
URL de reprise Telegram	Ajouter une signature cryptographique et renforcer le controle d'accès.
Gestion multi-environnements	Introduire AWX / Ansible Tower pour les permissions et l'historique.
Pilotage KPI	Creer un dashboard Grafana ou Zabbix alimente par les donnees d'incidents.
VISION	Conserver une architecture modulaire afin d'etendre les incidents couverts sans reconstruire toute la chaine.

Conclusion generale

Une approche AIOPS open-source, locale et controlee peut produire un gain operationnel mesurable.

Le projet a permis de concevoir et de valider une chaine complete de supervision proactive et d'auto-remediation. L'architecture associe le monitoring Zabbix, l'orchestration n8n, l'analyse locale Ollama, la validation Telegram, l'execution Ansible et la tracabilite SQLite.

La valeur principale ne repose pas sur un outil isole, mais sur l'integration coherente des composants et sur la place donnee a l'operateur humain. Le systeme automatise les etapes repetitives tout en conservant une barriere de validation avant toute action sensible.

85–90% Reduction MTTR estimee	Local Diagnostic IA	Human Decision critique
---	-------------------------------	-----------------------------------

Competences mobilisees

- Supervision avancee avec Zabbix et webhooks.
- Orchestration de workflows n8n et logique event-driven.
- Prompt engineering et integration d'un LLM local.
- Automatisation Windows avec Ansible et controle d'idempotence.
- Administration Linux, SSH, journalisation SQLite et tests de validation.

AUTEUR	Achraf El Mouchtari — Infrastructure Digitale, Systemes et Reseaux.
---------------	--