Note
/20

Compétences	Acquise	Non Acquise
C04 : Analyser un système informatique		
C05 : Concevoir un système informatique		

TP n°6

Supervision et Alerting 4.0 : Visualisation Grafana, Notifications Webhook et Audit IT/OT

FICHE MISSION 3

I – <u>PROBLEMATIQUE</u> : Comment visualiser en temps réel les indicateurs de production et avertir automatiquement les équipes de maintenance sur leur smartphone lors d'une dérive tout en garantissant l'étanchéité du réseau de commande OT ?

II – <u>OBJECTIFS</u> :

- Installer et configurer le serveur de supervision **Grafana** sur la VM Ubuntu (**192.168.2.100:3000**)
- Connecter **Grafana** à la base **InfluxDB v2** en utilisant le langage Flux
- Concevoir un tableau de bord (**Dashboard**) décisionnel (jauges, courbes historiques, indicateurs d'état)
- Définir une règle d'alerte sur seuil (Réflectivité ≤ 20 %) et émettre une notification **Webhook** vers l'extérieur via l'interface **enp0s3** (WAN)
- Auditer le cloisonnement entre le sous-réseau OT (**192.168.1.0/24**) et le sous-réseau IT (**192.168.2.0/24**)

III – <u>PRE-REQUIS</u> :

- Avoir validé le TP4 et le TP5 IT (Base **InfluxDB** alimentée en continu par **Telegraf**)
- Compréhension du fonctionnement d'une **API Web REST** et du protocole **HTTP** (requêtes **POST** / **Webhooks**)
- Notions d'analyse de trames réseau (**tcpdump**, **Wireshark**, **Nmap**)

IV – <u>MISE EN SITUATION</u> : Les étapes précédentes ont permis d'archiver les métriques du capteur OGD592 dans **InfluxDB v2**. Cependant, une base de données brute reste peu exploitable pour les techniciens de maintenance sur la ligne de conditionnement.

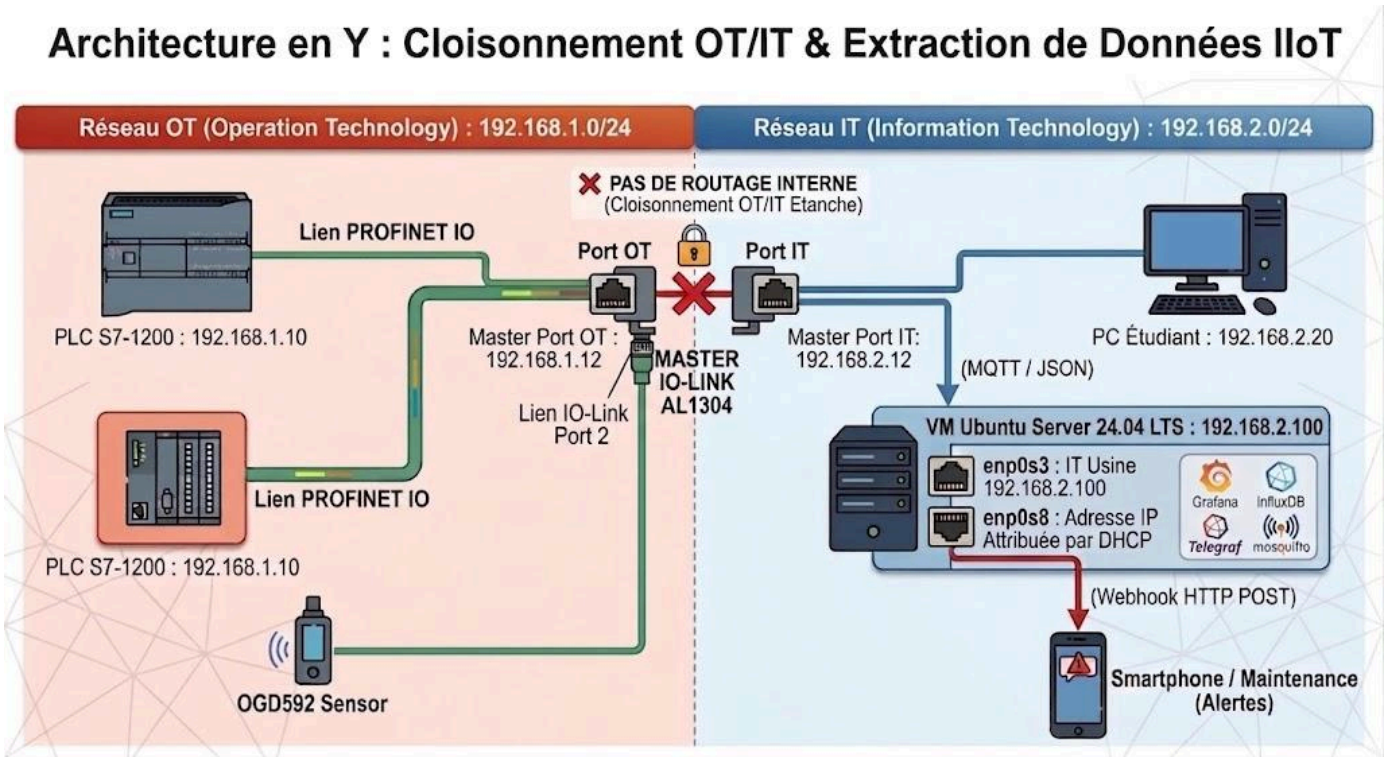
Dans cette dernière mission, vous allez finaliser la supervision 4.0. Vous allez déployer **Grafana** sur la Machine Virtuelle (**192.168.2.100:3000**) pour créer un tableau de bord visuel intégrant des jauges et des courbes d'évolution. Vous paramétrerez un mécanisme d'alerte automatique : en cas d'encrassement critique du capteur (réflectivité ≤ 20 %), **Grafana** générera une notification Webhook instantanée. La Machine Virtuelle utilisera sa seconde interface réseau **enp0s8** (connectée au WAN en Adressage IP Automatique) pour transmettre l'alerte vers le smartphone du technicien. Enfin, vous vérifierez la sécurité du réseau usine en testant le cloisonnement entre l'**IT** et l'**OT**.

V – <u>RESSOURCES MATERIELLES</u> : Vous réaliserez cette mission en travaillant avec un **PC** connecté à **Internet** et un accès à un environnement de travail de type **Workspace** avec les outils bureautiques :

- Traitement de texte.
- Tableur.
- Logiciel de diaporama.
- PC Étudiant (**192.168.2.20**) connecté au réseau IT
- Serveur sur une Machine Virtuelle **Ubuntu 26.04 LTS** (Réseau Interne **192.168.2.100** et Accès par Pont Carte Réseau Sans-Fil)
- Master **IO-Link** IFM (AL1304 / AL1306) avec double interface : IT (**192.168.2.12**) et OT (**192.168.1.12**)
- Capteur photoélectrique **IO-Link** OGD592
- Automate PLC Siemens S7-1200 : OT (**192.168.1.10**)
- Logiciel Moneo Configure
- Clients **MQTT** (mosquitto-clients, MQTT Explorer)
- Logiciels installés sur la VM : Broker **Mosquitto**, **InfluxDB v2**, **Agent Telegraf**, **Wireshark**, **Nmap**
- Application de notification distante : **Discord** / **Slack** / Serveur **Webhook**

VI – <u>RESSOURCES PEDAGOGIQUES</u> : Vous utiliserez le dossier technique fourni ainsi que la documentation technique IFM du Master **IO-Link** et du capteur OGD592.

VII – <u>SCHEMA DE L'INFRASTRUCTURE RESEAU</u> :



VIII – ETUDE PREALABLE DE LA SUPERVISION ET DES WEBHOOKS :

Dans cette nouvelle mission, nous allons nous intéresser à la supervision et à l'envoi de notifications.

Qu'est-ce que **Grafana** et quel est son rôle principal dans une architecture **IIoT** / **Industrie 4.0** ?

Grafana est une plateforme open-source de visualisation et d'analyse de données en temps réel. Dans une architecture **IIoT**, elle sert à centraliser et afficher sous forme de tableaux de bord interactifs (jauges, courbes, indicateurs d'état) les métriques issues des capteurs et systèmes de production.

Pourquoi est-il nécessaire d'associer **Grafana** à une base de données de séries temporelles comme **InfluxDB v2** ?

InfluxDB v2 est optimisée pour le stockage et la requête rapide de volumes massifs de données horodatées mais ne dispose pas d'outils avancés de restitution visuelle. **Grafana** vient compléter la base en fournissant une interface de supervision dynamique, la capacité d'agréger plusieurs sources de données et un moteur d'alerting configurable.

Quelle est la différence entre une **IHM** locale (pupitre tactile connecté au **PLC**) et une plateforme de supervision centralisée comme **Grafana** ?

Une **IHM** locale offre une vue d'exploitation immédiate et dédiée à la conduite en bord de ligne sur le réseau **OT**, tandis que **Grafana** centralise l'ensemble des données multi-sources sur le réseau **IT** pour fournir des tableaux de bord analytiques accessibles à distance par les équipes de gestion et de maintenance.

Expliquer le fonctionnement d'un **Webhook HTTP POST** pour la transmission d'alertes de maintenance.

Un **Webhook** fonctionne comme un mécanisme de rappel automatique déclenché par un événement précis. Dès que **Grafana** détecte un dépassement de seuil, il formule une requête **HTTP POST** contenant les détails du problème qu'il expédie via la carte réseau **WAN** vers un service distant pour notifier immédiatement les techniciens sur leur mobile.

En quoi l'analyse de l'historique de réflectivité permet-elle de basculer d'une maintenance préventive classique à une maintenance prédictive ?

L'étude continue de la pente de dégradation de la réflectivité permet d'évaluer la vitesse d'encrassement de la lentille et de modéliser l'échéance exacte du dysfonctionnement. Cette anticipation permet d'intervenir uniquement lorsque cela est nécessaire, évitant ainsi les arrêts non programmés et les opérations de maintenance systématiques superflues.

IX – INSTALLATION ET CONFIGURATION DE GRAFANA :

1. Installation de Grafana :

Nous allons commencer par installer **Grafana** sur notre machine virtuelle, taper les commandes suivantes pour l'installer :

```
sudo apt-get install -y apt-transport-https software-properties-common wget
sudo mkdir -p /etc/apt/keyrings/
wget -q -O - https://apt.grafana.com/gpg.key | gpg --dearmor | sudo tee /etc/apt/keyrings/grafana.gpg > /dev/null
echo "deb [signed-by=/etc/apt/keyrings/grafana.gpg] https://apt.grafana.com stable main" | sudo tee /etc/apt/sources.list.d/grafana.list
sudo apt-get update && sudo apt-get install grafana -y
sudo systemctl enable --now grafana-server
```

Donner la capture d'écran du bon fonctionnement de **Grafana**.

```
root@Ubuntu:/home/eleve# sudo systemctl status grafana-server
● grafana-server.service - Grafana instance
     Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; preset: enabled)
     Active: active (running) since Sun 2026-09-20 16:56:09 UTC; 2min 12s ago
 Invocation: c2fe701421a143298acfc35ea2c0e569
       Docs: http://docs.grafana.org
   Main PID: 6061 (grafana)
      Tasks: 83 (limit: 1888)
     Memory: 955.1M (peak: 976.6M, swap: 40.5M, swap peak: 44.4M)
        CPU: 17.912s
     CGroup: /system.slice/grafana-server.service
             ├─6061 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini --pidfile=/run/grafana/grafana-server.pid --packaging=deb cfg:defau>
             ├─6079 /var/lib/grafana/plugins-bundled/grafana-postgresql-datasource/gpx_grafana_postgresql_datasource_linux_amd64
             ├─6090 /var/lib/grafana/plugins-bundled/influxdb/gpx_grafana_influxdb_datasource_linux_amd64
             ├─6095 /var/lib/grafana/plugins-bundled/jaeger/gpx_grafana-jaeger-datasource_linux_amd64
             ├─6100 /var/lib/grafana/plugins-bundled/loki/gpx_grafana-loki-datasource_linux_amd64
             ├─6107 /var/lib/grafana/plugins-bundled/mssql/gpx_grafana-mssql-datasource_linux_amd64
             ├─6112 /var/lib/grafana/plugins-bundled/mysql/gpx_grafana-mysql-datasource_linux_amd64
             ├─6152 /var/lib/grafana/plugins/elasticsearch/gpx_grafana_elasticsearch_datasource_linux_amd64
             ├─6157 /var/lib/grafana/plugins/grafana-pyroscope-datasource/gpx_grafana-pyroscope-datasource_linux_amd64
             ├─6163 /var/lib/grafana/plugins/opentsdb/gpx_grafana-opentsdb-datasource_linux_amd64
             ├─6170 /var/lib/grafana/plugins/stackdriver/gpx_grafana_cloudmonitoring_datasource_linux_amd64
             ├─6175 /var/lib/grafana/plugins/prometheus/gpx_grafana-prometheus-datasource_linux_amd64
             ├─6180 /var/lib/grafana/plugins/tempo/gpx_grafana-tempo-datasource_linux_amd64
             └─6186 /var/lib/grafana/plugins/zipkin/gpx_grafana-zipkin-datasource_linux_amd64

sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=provisioning.grafana.app resource=repositories reason=getStats t=2026-09-2>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=provisioning.grafana.app resource=repositories reason=getStats t=2026-09-2>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=provisioning.grafana.app resource=repositories reason=getStats t=2026-09-2>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=playlist.grafana.app resource=playlists reason=getStats t=2026-09-20T16:58>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=playlist.grafana.app resource=playlists reason=getStats t=2026-09-20T16:58>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=playlist.grafana.app resource=playlists reason=getStats t=2026-09-20T16:58>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=folder.grafana.app resource=folders reason=getStats t=2026-09-20T16:58:13.>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=folder.grafana.app resource=folders reason=getStats t=2026-09-20T16:58:13.>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=bleve-backend namespace=default group=folder.grafana.app resource=folders reason=getStats t=2026-09-20T16:58:13.>
sept. 20 16:58:13 ubuntu grafana[6061]: logger=infra.usagestats t=2026-09-20T16:58:13.243450432Z level=info msg="Usage stats are ready to report"
lines 1-35/35 (END)
```

2. Configuration de Grafana :

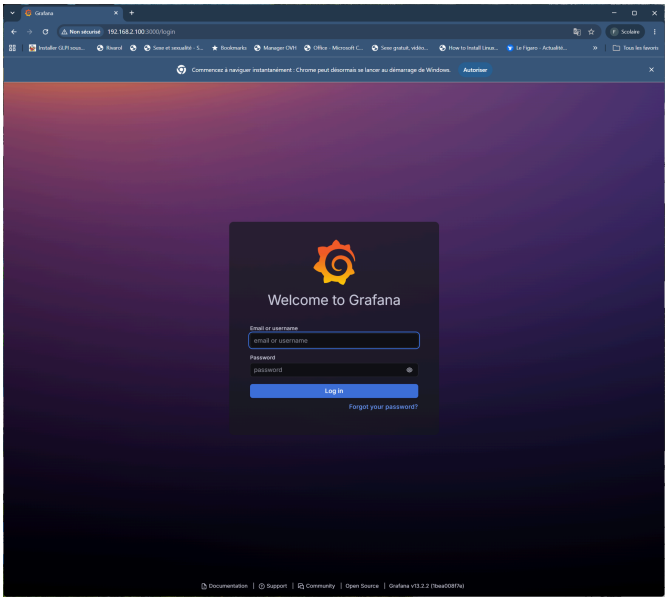
Vous pouvez maintenant accéder à l'interface de configuration de **Grafana** en utilisant un navigateur Internet et en tapant l'**URL** suivante : **http://192.168.2.100:3000**

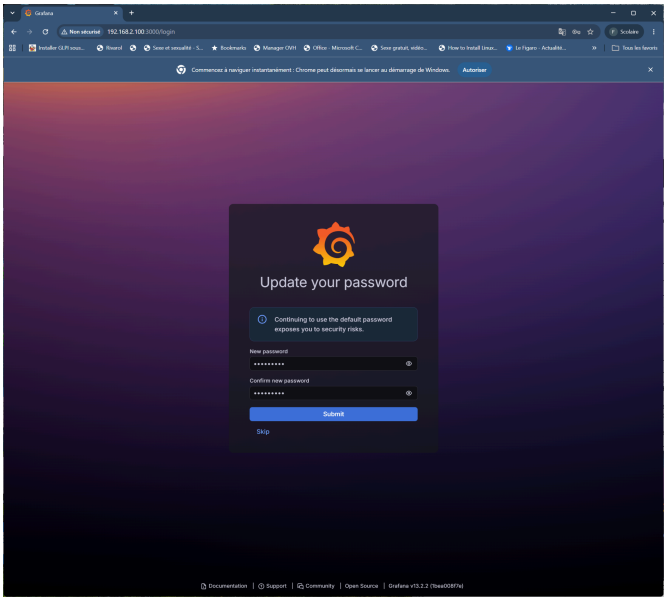
Remplir les informations suivantes en ajoutant une nouvelle source de données dans **Grafana** :

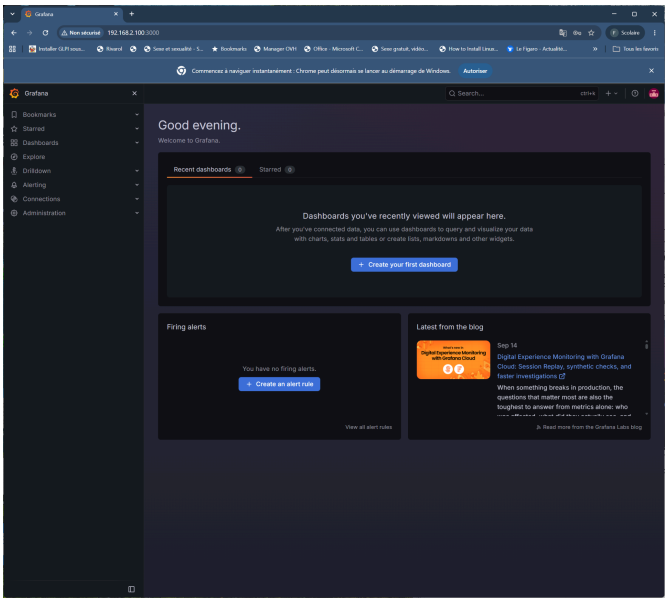
- Query Language : **Flux**
- URL : **http://localhost:8086**
- Organization : **Lycee_Jules_Fil**
- Bucket : **iiot_cassoulet**

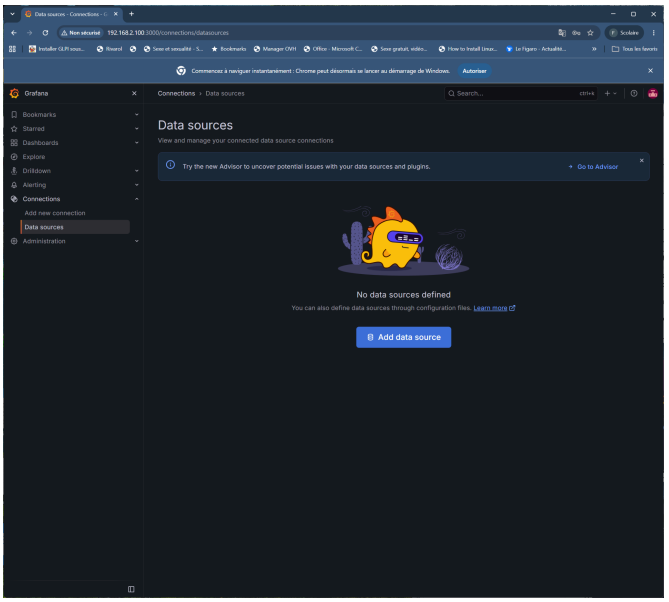
Donner les captures d'écran de la configuration de **Grafana**.

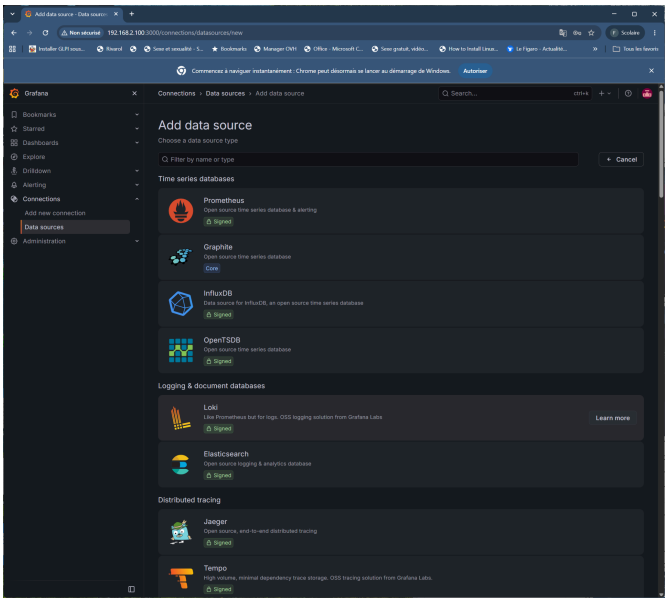
L'ajout d'une source de données dans **Grafana** est dans : **Connections → Data Sources → Add data source → InfluxDB**.

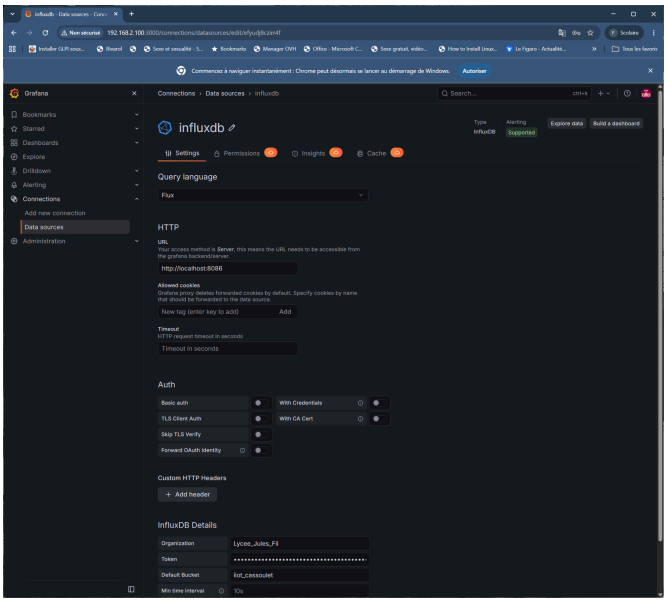




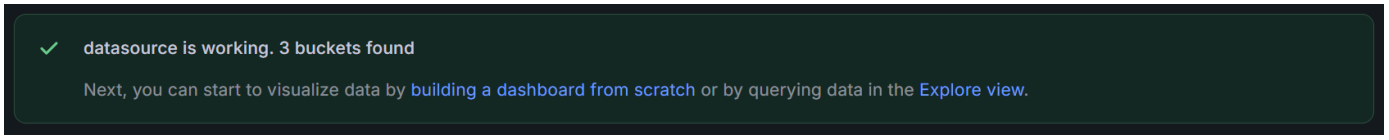








Vous devez avoir ce message si votre configuration est correcte, il faut penser à sauvegarder :



Dans **Data Sources**, votre nouvelle source de données doit apparaître :



3. Création du Dashboard « Supervision Ligne Conditionnement » :

Nous allons maintenant créer trois panneaux sur le tableau de bord :

- Gauge (Réflectivité instantanée) : Vert (> 30 %), Orange (20 à 30 %), Rouge (≤ 20 %)
- Graphique Time-Series : Evolution conjointe de la distance (mm) et de la réflectivité (%)
- Stat Panel : Affichage numérique instantané de la dernière valeur reçue du Master AL1304/AL1306 concernant la distance

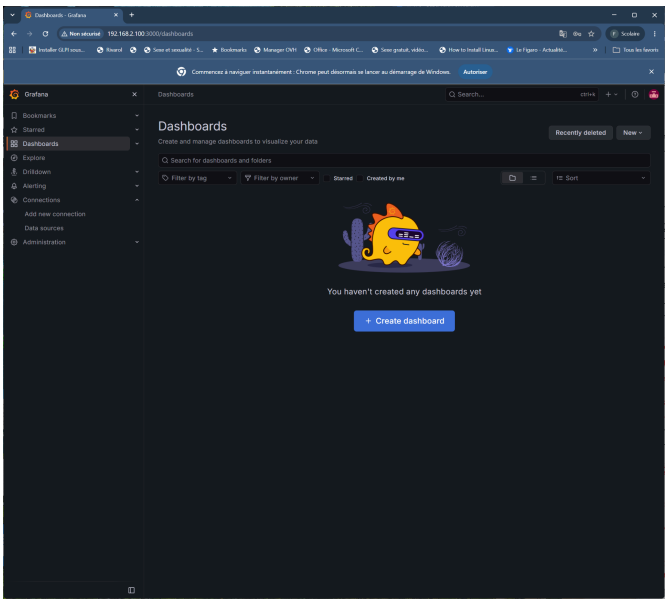
Voici la procédure étape par étape pour créer et configurer le premier panneau du tableau de bord dans **Grafana** à partir de votre source de données **InfluxDB v2**.

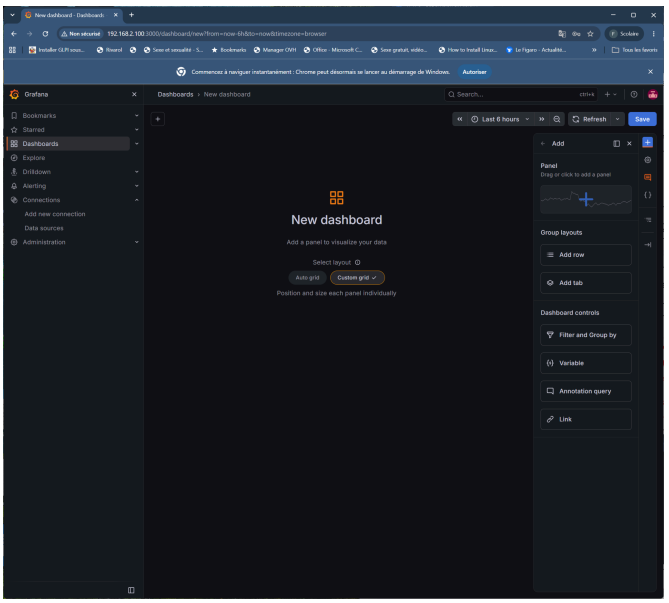
Etape préalable : Création du **Dashboard**

1. Dans le menu de gauche de Grafana, cliquez sur **Dashboards** puis sur le bouton **New** → **New Dashboard**

2. Cliquez sur le bouton **Add visualization**

3. Sélectionnez votre source de données **InfluxDB** (configurée en langage **Flux**)





Panneau 1 : Gauge — Réflectivité instantanée

Ce panneau affiche une jauge dynamique colorée selon l'état d'encrassement du capteur.

1. Choix de la visualisation : En haut à droite, changez le type de panneau pour sélectionner **Gauge**
2. Titre du panneau : Dans la section Panel options à droite, nommez le panneau « **Réflectivité Instantanée** »
3. Requête Flux (Code) : Dans l'onglet Query, collez la requête suivante :

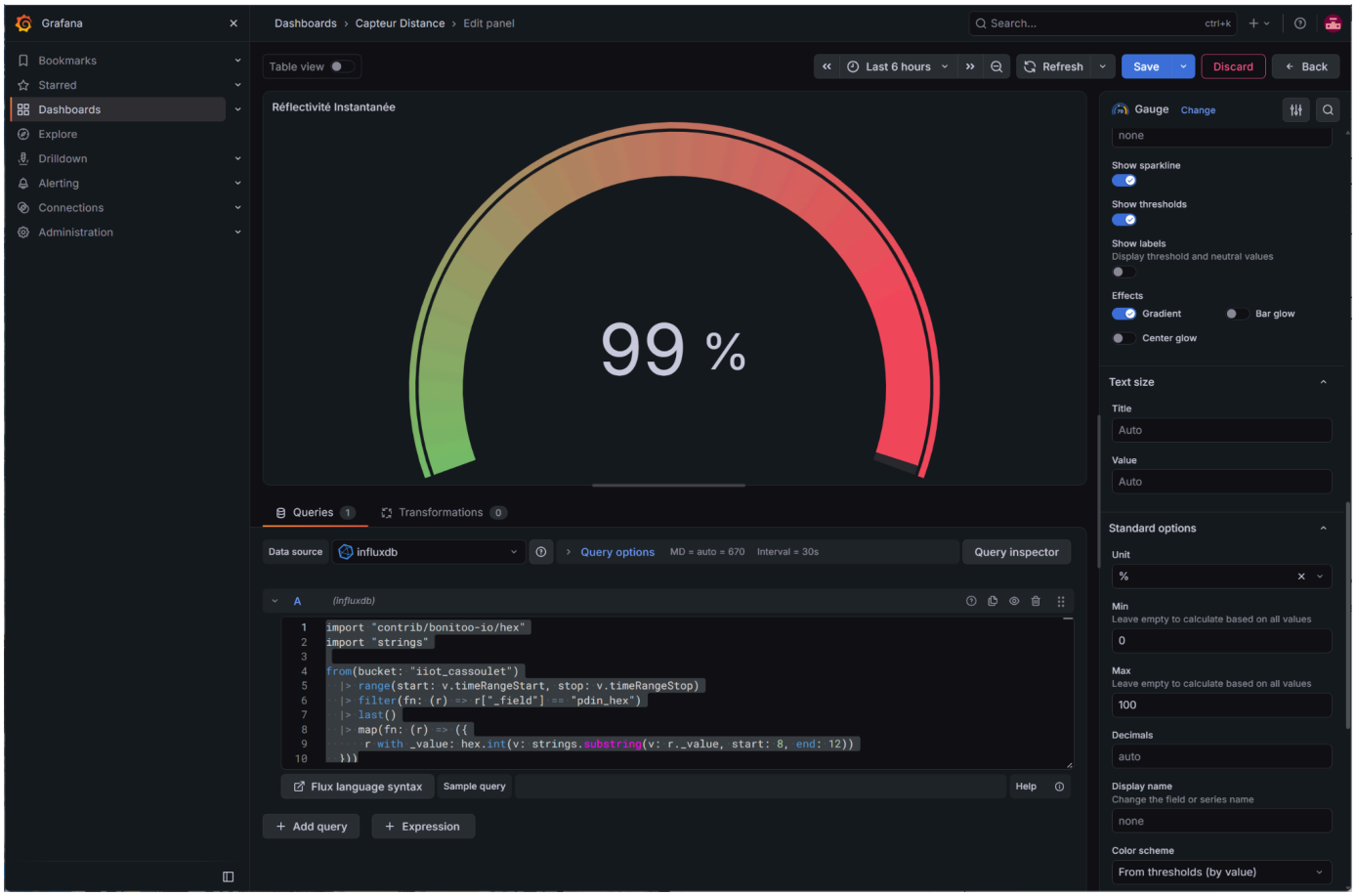
```
import "contrib/bonitoo-io/hex"
import "strings"



from(bucket: "iiot_cassoulet")
 |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
 |> filter(fn: (r) => r["_field"] == "pdin_hex")
 |> last()
 |> map(fn: (r) => ({
     r with _value: hex.int(v: strings.substring(v: r._value, start: 8, end: 12))
 }))
```

4. Configuration de la valeur (Standard options) :

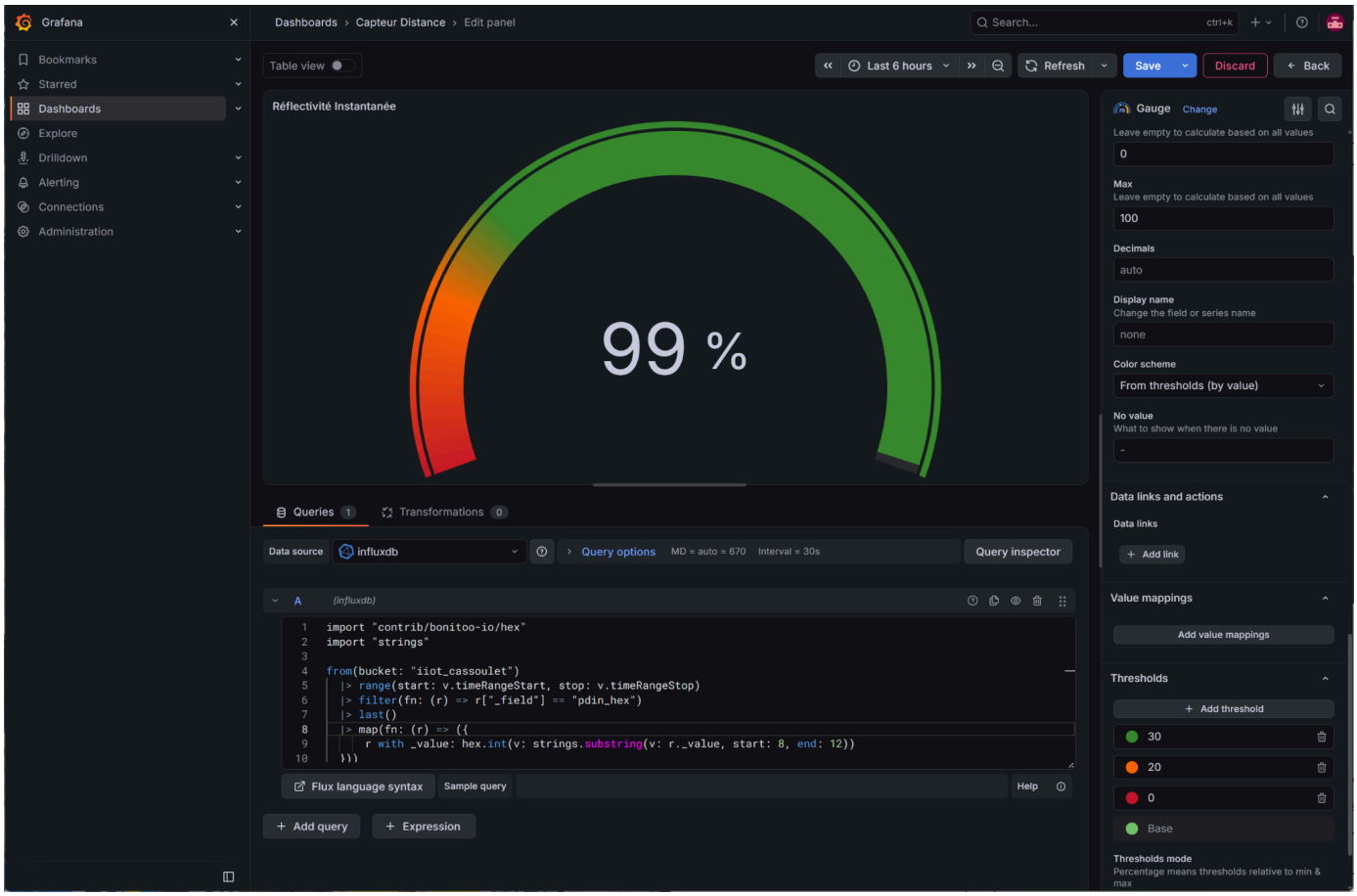
   - Unit : Percent (0-100) ou %
   - Min : 0 / Max : 100.



5. Configuration des seuils (Thresholds) :

Dans le volet de droite, configurer les plages de couleurs :

- Rouge (Base) : 0 (valeur ≤ 20 %)
- Orange : 20 (valeur entre 20 % et 30 %)
- Vert : 30 (valeur > 30 %)



6. Cliquez sur **Save** en haut à droite

Comment faites-vous pour créer les deux autres panneaux en donnant les étapes de configuration ?

Panneau 2 : Time-Series — Evolution conjointe Distance et Réflectivité

Ce graphique permet de suivre l'évolution temporelle simultanée des deux métriques du capteur.

1. Cliquez sur **Add** → **Visualization** pour ajouter un second panneau
2. Choix de la visualisation : Sélectionnez **Time series**
3. Titre du panneau : Nommez-le « **Evolution Distance et Réflectivité** »
4. Requêtes Flux :

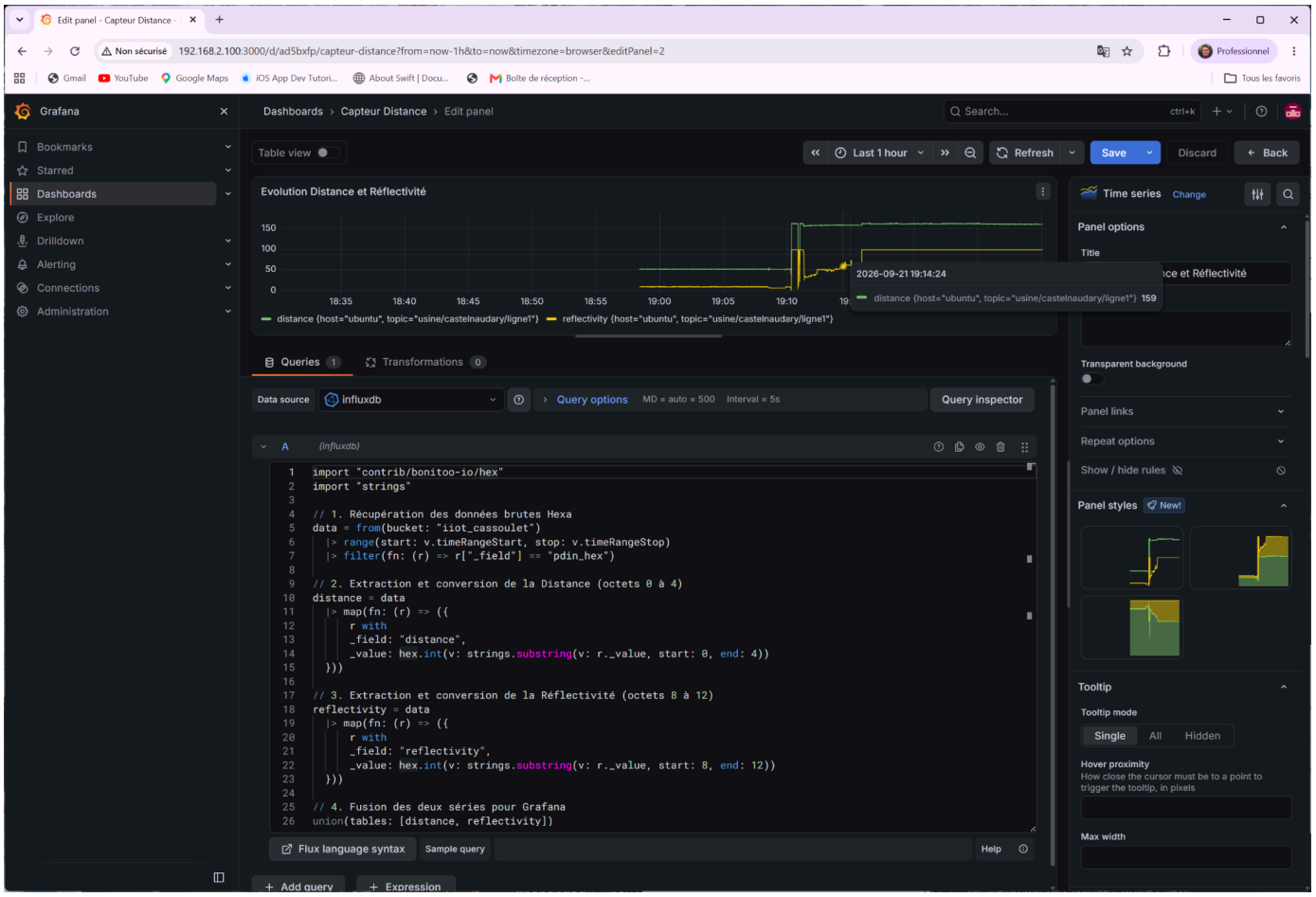
```
import "contrib/bonitoo-io/hex"
import "strings"

// 1. Récupération des données brutes Hexa
data = from(bucket: "iiot_cassoulet")
 |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
 |> filter(fn: (r) => r["_field"] == "pdin_hex")

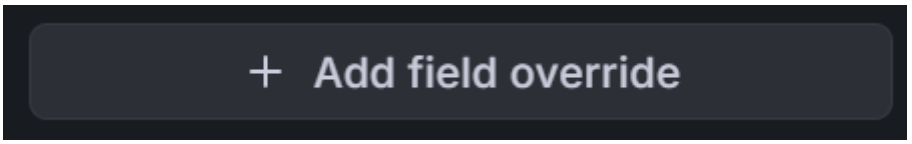
// 2. Extraction et conversion de la Distance (octets 0 à 4)
distance = data
 |> map(fn: (r) => ({
     r with
     _field: "distance",
     _value: hex.int(v: strings.substring(v: r._value, start: 0, end: 4))
 }))

// 3. Extraction et conversion de la Réflectivité (octets 8 à 12)
reflectivity = data
 |> map(fn: (r) => ({
     r with
     _field: "reflectivity",
     _value: hex.int(v: strings.substring(v: r._value, start: 8, end: 12))
 }))

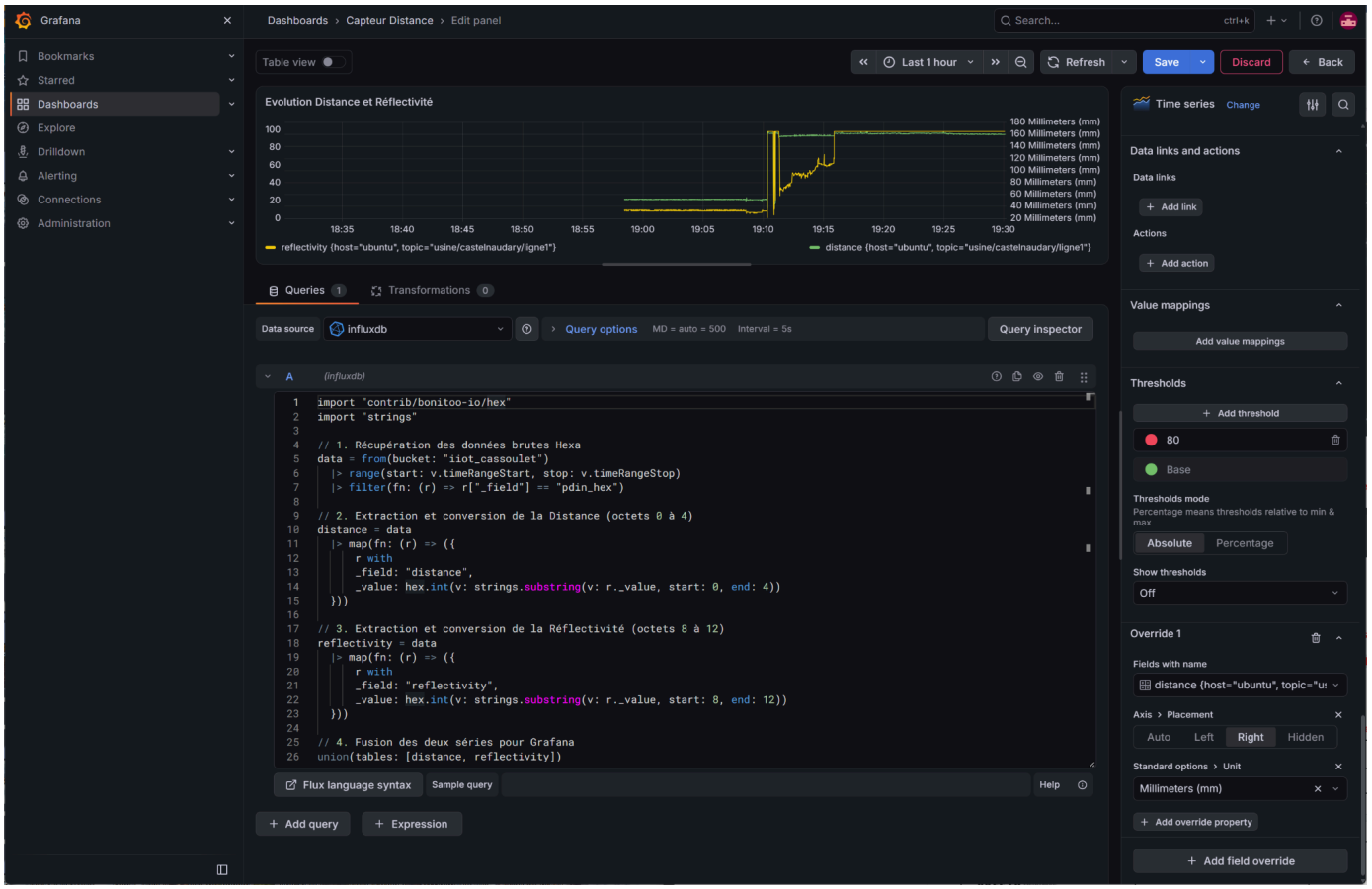
// 4. Fusion des deux séries pour Grafana
union(tables: [distance, reflectivity])
```



5. Configuration du double axe Y (**Overrides**) :
   - Allez dans l'onglet **Overrides** (à côté de **Properties** dans la colonne de droite)



   - Cliquez sur **+ Add field override → Fields with name**
   - Sélectionnez le champ de la **distance (distance_mm)**
   - Cliquez sur **+ Add override property → Axis > Placement** et choisissez **Right**
   - Ajoutez une autre propriété **Standard options > Unit** et choisissez **Millimeters (mm)**
6. Cliquez sur **Apply**


Grafana
Dashboards › Capteur Distance › Edit panel
Search...
Bookmarks
Starred
Dashboards
Explore
Drilldown
Alerting
Connections
Administration
Table view
Last 1 hour
Refresh
Save
Discard
Back
Evolution Distance et Réflectivité
reflectivity {host="ubuntu", topic="usine/castelnaudary/ligne1"}
distance {host="ubuntu", topic="usine/castelnaudary/ligne1"}
Time series
Change
Data links and actions
Data links
Add link
Actions
Add action
Value mappings
Add value mappings
Thresholds
Add threshold
80
Base
Thresholds mode
Percentage means thresholds relative to min & max
Absolute
Percentage
Show thresholds
Off
Override 1
Fields with name
Axis › Placement
Auto
Left
Right
Hidden
Standard options › Unit
Millimeters (mm)
Add override property
Add field override
Queries 1
Transformations 0
Data source
influxdb
Query options
MD = auto = 500
Interval = 5s
Query inspector
A
(influxdb)
import "contrib/bonitoo-io/hex"
import "strings"
// 1. Récupération des données brutes Hexa
data = from(bucket: "iiot_cassoulet")
|> range(start: v.timeRangeStart, stop: v.timeRangeStop)
|> filter(fn: (r) => r["_field"] == "pdin_hex")
// 2. Extraction et conversion de la Distance (octets 0 à 4)
distance = data
|> map(fn: (r) => ({
r with
_field: "distance",
_value: hex.int(v: strings.substring(v: r._value, start: 0, end: 4))
}))
// 3. Extraction et conversion de la Réflectivité (octets 8 à 12)
reflectivity = data
|> map(fn: (r) => ({
r with
_field: "reflectivity",
_value: hex.int(v: strings.substring(v: r._value, start: 8, end: 12))
}))
// 4. Fusion des deux séries pour Grafana
union(tables: [distance, reflectivity])
Flux language syntax
Sample query
Help
Add query
Expression

Panneau 3 : Stat Panel — Affichage numérique instantané

Ce panneau affiche sous forme de grand chiffre la toute dernière mesure brute enregistrée.

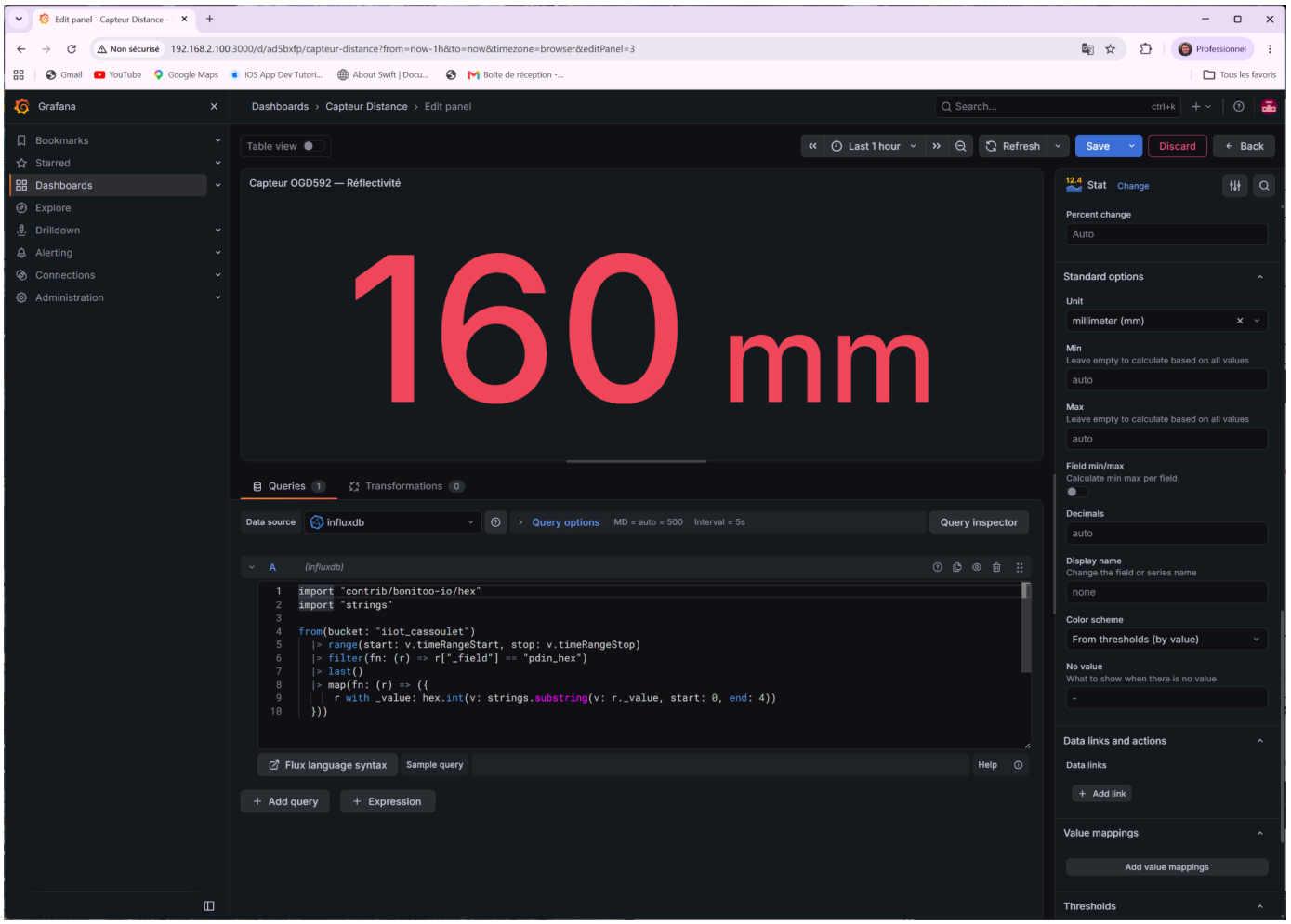
1. Cliquez sur **Add** → **Visualization** pour créer le troisième panneau
2. Choix de la visualisation : Sélectionnez **Stat**
3. Titre du panneau : Nommez-le « **Capteur OGD592 — Réflectivité** »
4. Requête Flux (Code) :

```
import "contrib/bonitoo-io/hex"
import "strings"



from(bucket: "iiot_cassoulet")
 |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
 |> filter(fn: (r) => r["_field"] == "pdin_hex")
 |> last()
 |> map(fn: (r) => ({
     r with _value: hex.int(v: strings.substring(v: r._value, start: 0, end: 4))
 }))
```

5. Configuration d'affichage (Stat options) :
   - Calculation : **Last *** (ou **Last**)
   - Color mode : **Value** (pour que le texte prenne la couleur du seuil actuel) ou **Background**
   - Graph mode : **Area** (pour afficher un mini-historique en arrière-plan) ou **None**
   - Unit : **mm**
6. Cliquez sur Apply



Votre tableau de bord doit ressembler à ceci :



Comment faites-vous pour afficher le tableau de bord sans les menus afin de le partager avec d'autres utilisateurs ? Faites une capture d'écran de ce tableau de bord sans les menus.

1. Masquage des menus via le mode Kiosk

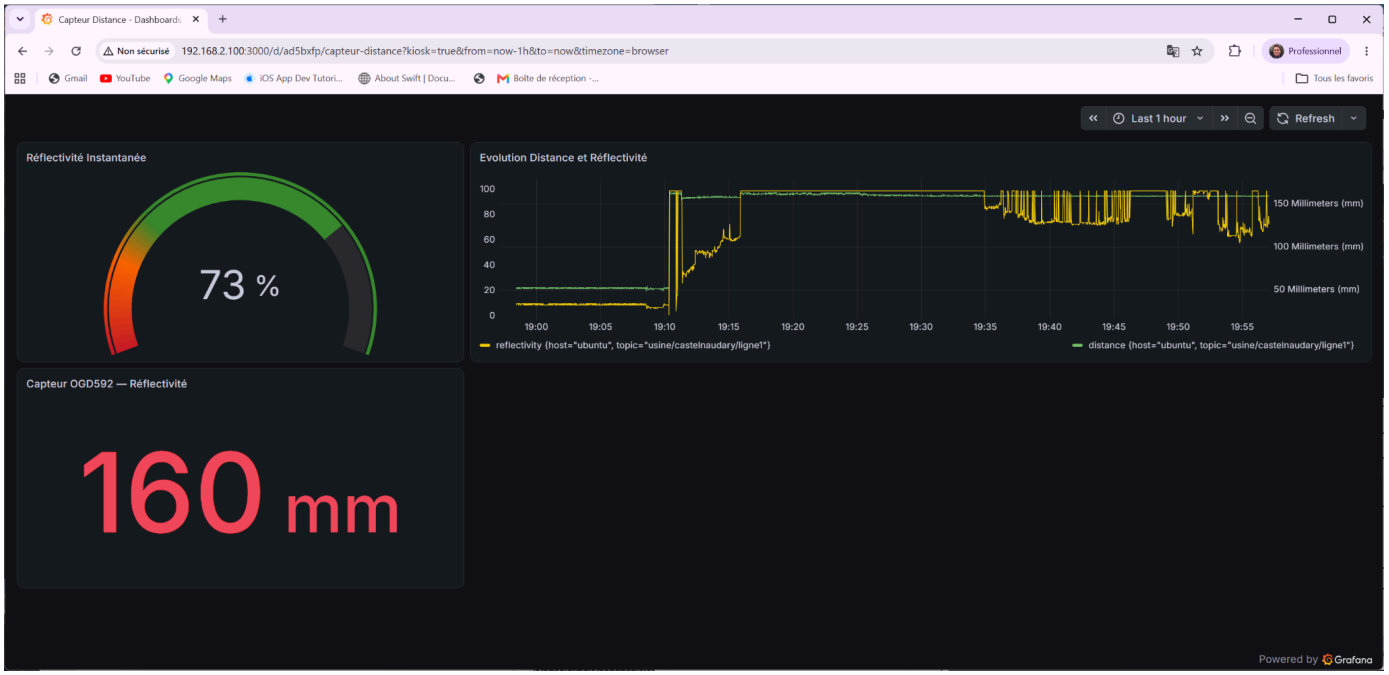
L'ajout du paramètre kiosk à l'URL du tableau de bord désactive la barre latérale de navigation ainsi que les options d'édition. L'affichage est ainsi dédié aux seuls indicateurs de mesure (réflectivité instantanée, distance en mm et évolution temporelle). L'URL d'accès épurée est :
http://192.168.2.100:3000/d/ad5bxfp/capteur-distance?kiosk

Variantes d'affichage :

- ?kiosk : masque l'intégralité des menus et barres d'outils
- ?kiosk=tv : masque la barre latérale tout en conservant le sélecteur de plage temporelle en haut à droite

3. Capture d'écran du tableau de bord partagé

La capture ci-dessous illustre le rendu obtenu avec l'URL en mode kiosk : la barre latérale de gauche a disparu, offrant une interface de supervision propre pour un écran partagé.



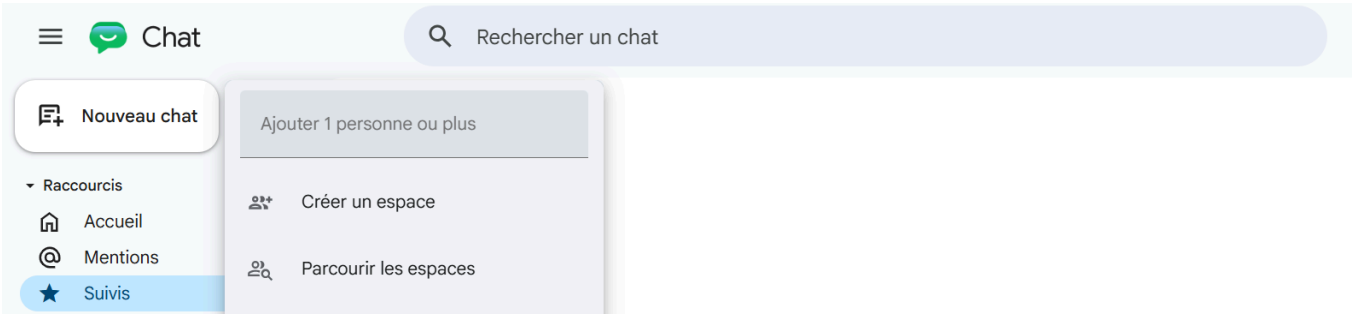
X – CONFIGURATION DES NOTIFICATIONS :

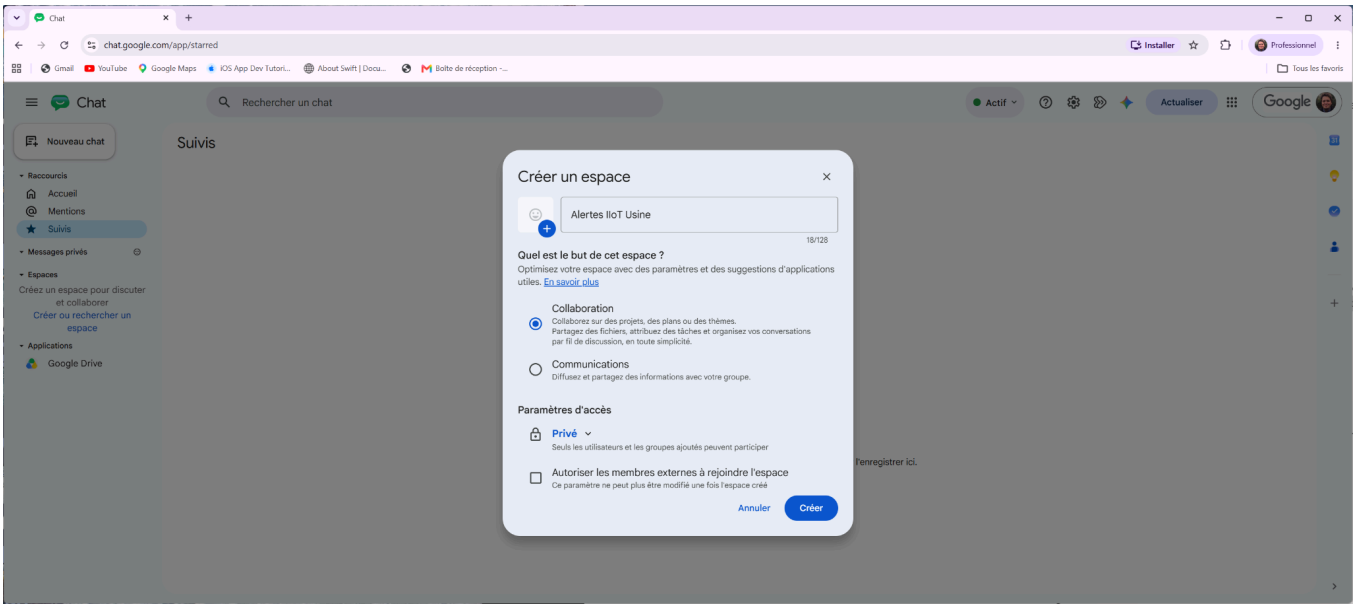
L'objectif de cette partie consiste à prévenir l'encrassement du capteur optique OGD592 grâce à la mise en place d'un service de notification automatique en temps réel. Lorsque des salissures s'accumulent sur la lentille du capteur, la mesure de réflectivité chute progressivement. Pour garantir la continuité d'exploitation, une alerte est transmise automatiquement dans un salon Google Chat dès que la réflectivité descend sous le seuil critique de 20 %.

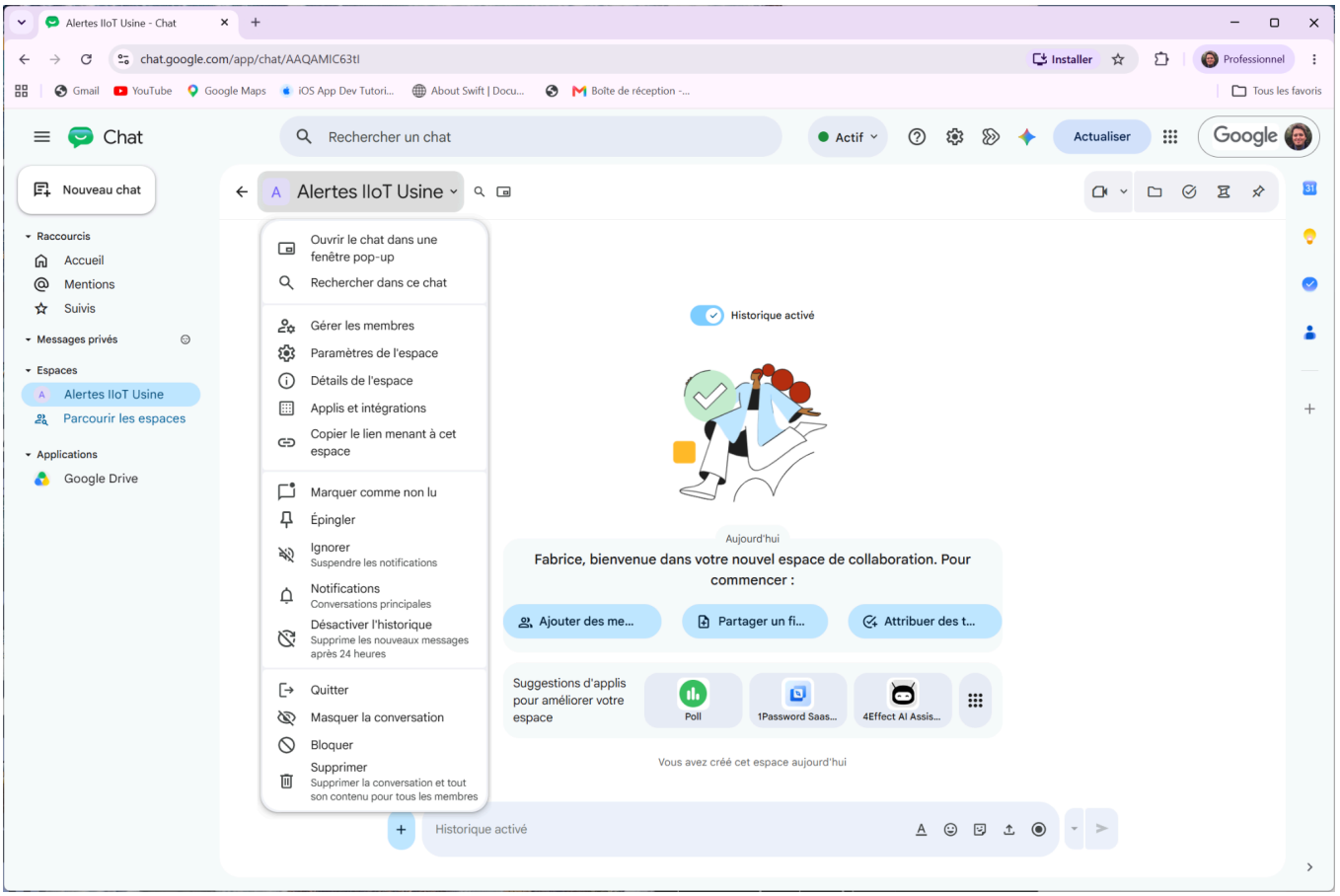
La phase préalable consiste à préparer le salon de réception dans Google Chat :

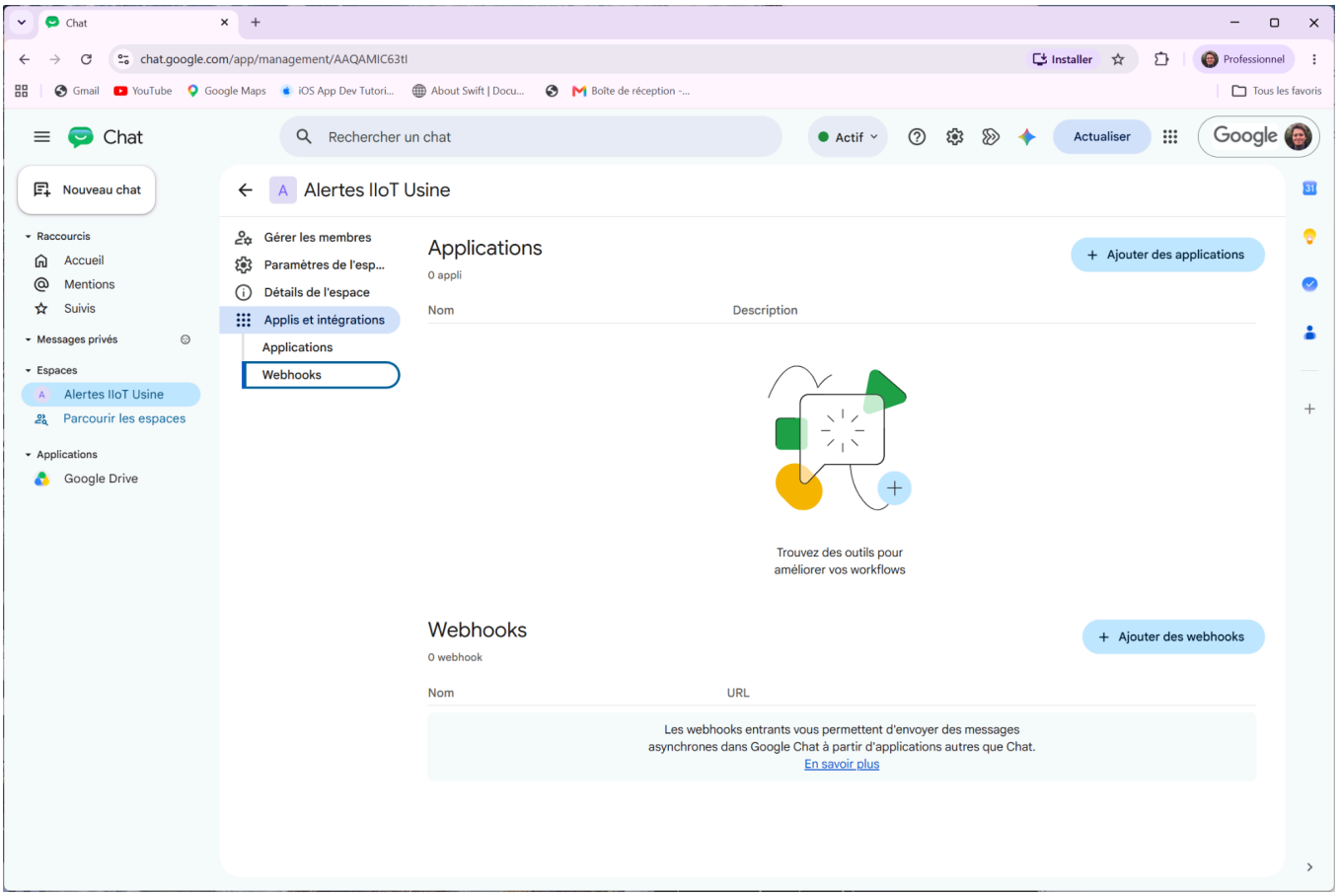
- Création de l'espace : Ouverture d'un nouvel espace de discussion dédié (**Alertes IIoT Usine**) dans **Google Chat**
- Génération du Webhook : Création d'un **Webhook** entrant nommé **Grafana Bot** depuis le menu **Applis et intégrations → Webhooks**
- Récupération du lien : Copie de l'URL unique générée au format **https://chat.googleapis.com/v1/spaces/** ….. afin de la lier à Grafana

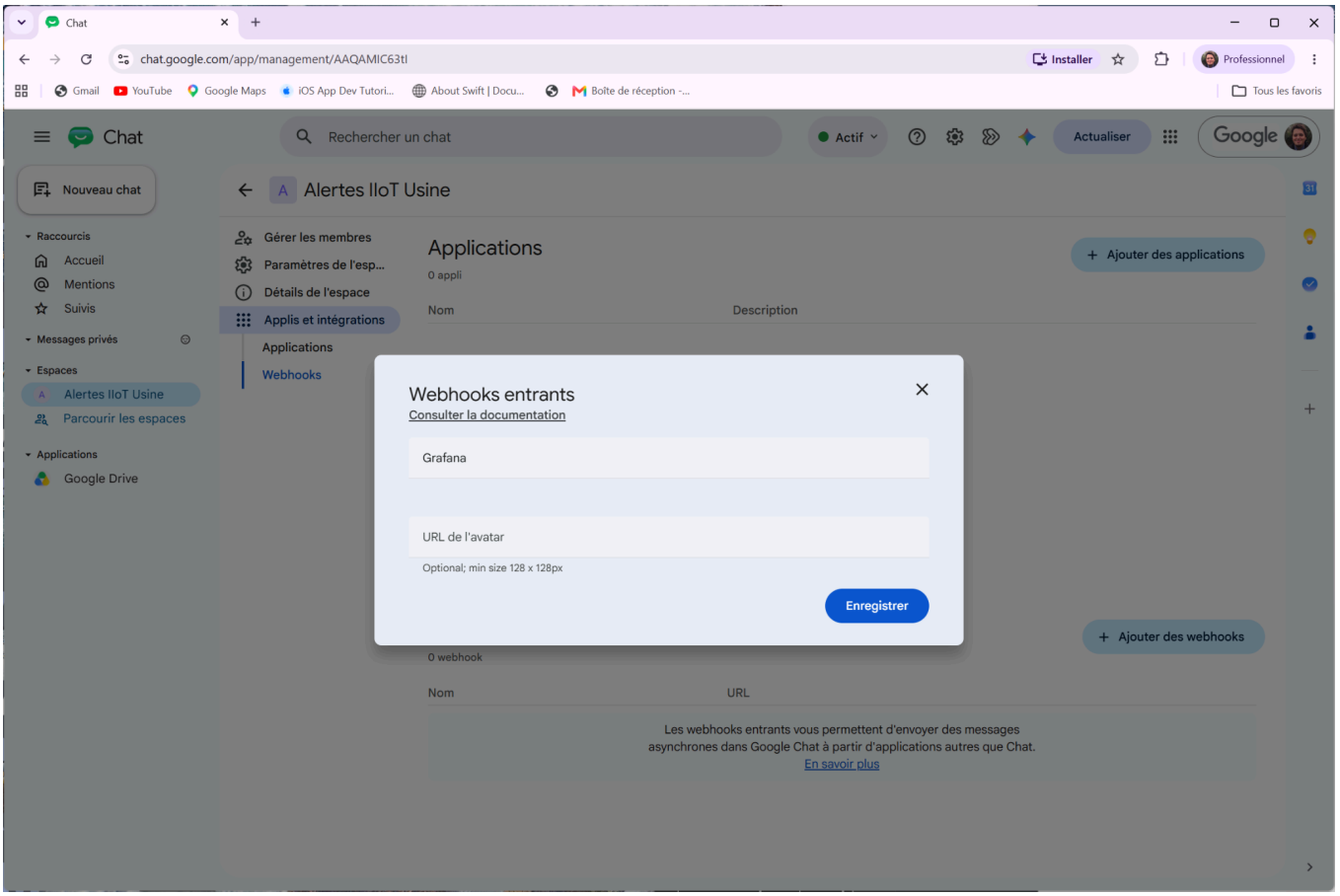
Comment faites-vous pour réaliser cette étape préalable ?

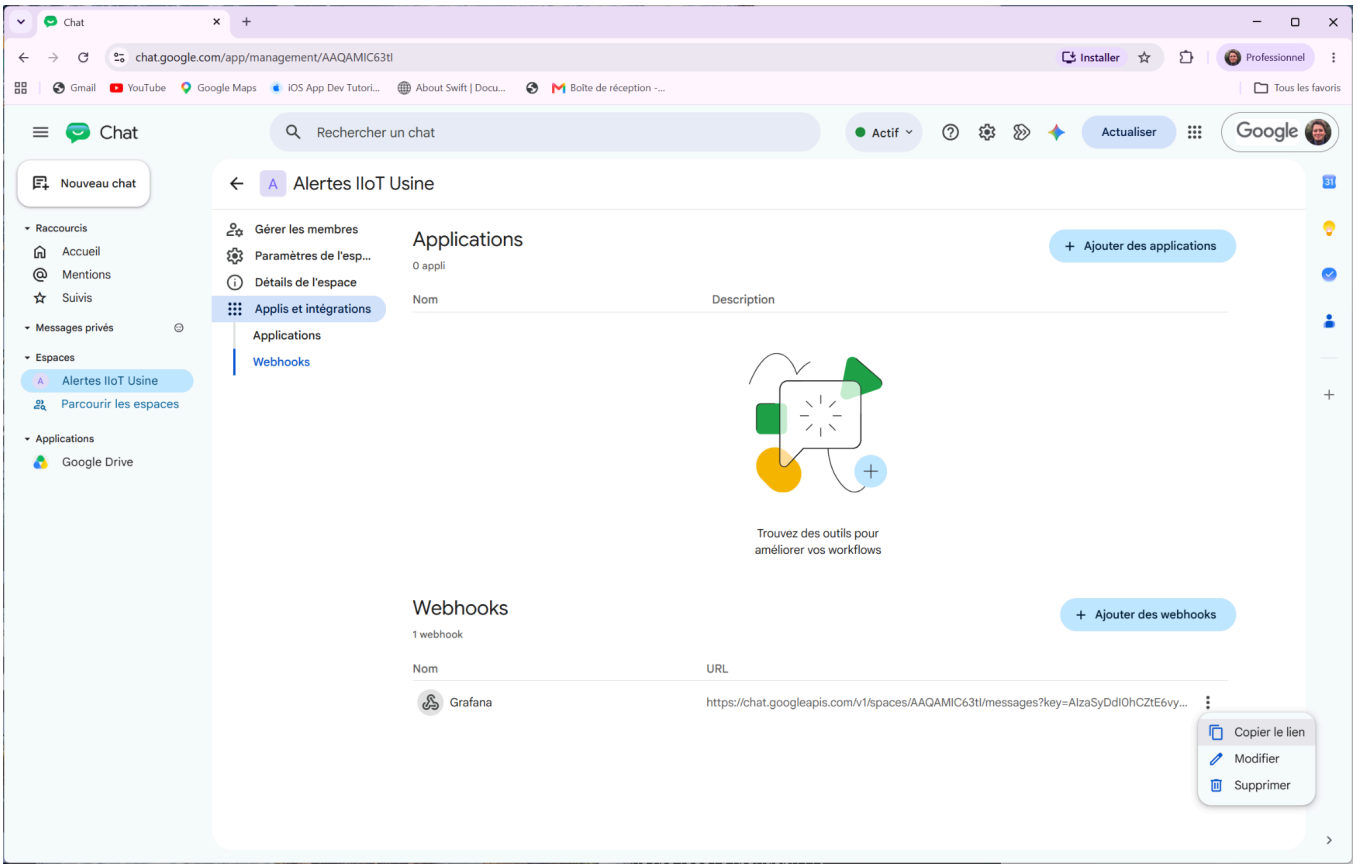





Alertes IIoT Usine - Chat
chat.google.com/app/chat/AAQAMIC63tI
Installer
Professionnel
Gmail
YouTube
Google Maps
iOS App Dev Tutori...
About Swift | Docu...
Boîte de réception -...
Tous les favoris
Chat
Rechercher un chat
Actif
Actualiser
Google
Nouveau chat
Alertes IIoT Usine
Raccourcis
Accueil
Mentions
Suivis
Messages privés
Espaces
Alertes IIoT Usine
Parcourir les espaces
Applications
Google Drive
Ouvrir le chat dans une fenêtre pop-up
Rechercher dans ce chat
Gérer les membres
Paramètres de l'espace
Détails de l'espace
Applis et intégrations
Copier le lien menant à cet espace
Marquer comme non lu
Épingler
Ignorer
Suspendre les notifications
Notifications
Conversations principales
Désactiver l'historique
Supprime les nouveaux messages après 24 heures
Quitter
Masquer la conversation
Bloquer
Supprimer
Supprimer la conversation et tout son contenu pour tous les membres
Historique activé
Aujourd'hui
Fabrice, bienvenue dans votre nouvel espace de collaboration. Pour commencer :
Ajouter des me...
Partager un fi...
Attribuer des t...
Suggestions d'applis pour améliorer votre espace
Poll
1Password Saas...
4Effect AI Assis...
Vous avez créé cet espace aujourd'hui
Historique activé


Chat
chat.google.com/app/management/AAQAMIC63tI
Installer
Professionnel
Gmail
YouTube
Google Maps
iOS App Dev Tutori...
About Swift | Docu...
Boîte de réception -...
Tous les favoris
Chat
Rechercher un chat
Actif
Actualiser
Google
Nouveau chat
Raccourcis
Accueil
Mentions
Suivis
Messages privés
Espaces
Alertes IIoT Usine
Parcourir les espaces
Applications
Google Drive
Alertes IIoT Usine
Gérer les membres
Paramètres de l'esp...
Détails de l'espace
Applis et intégrations
Applications
Webhooks
Applications
0 appli
Ajouter des applications
Nom
Description
Trouvez des outils pour
améliorer vos workflows
Webhooks
0 webhook
Ajouter des webhooks
Nom
URL
Les webhooks entrants vous permettent d'envoyer des messages
asynchrones dans Google Chat à partir d'applications autres que Chat.
En savoir plus


Chat
chat.google.com/app/management/AAQAMIC63tI
Installer
Professionnel
Gmail
YouTube
Google Maps
iOS App Dev Tutori...
About Swift | Docu...
Boîte de réception -...
Tous les favoris
Chat
Rechercher un chat
Actif
Actualiser
Google
Nouveau chat
Raccourcis
Accueil
Mentions
Suivis
Messages privés
Espaces
Alertes IIoT Usine
Parcourir les espaces
Applications
Google Drive
Alertes IIoT Usine
Gérer les membres
Paramètres de l'esp...
Détails de l'espace
Applis et intégrations
Applications
Webhooks
Applications
0 appli
Nom
Description
Ajouter des applications
Webhooks entrants
Consulter la documentation
Grafana
URL de l'avatar
Optional; min size 128 x 128px
Enregistrer
Ajouter des webhooks
0 webhook
Nom
URL
Les webhooks entrants vous permettent d'envoyer des messages asynchrones dans Google Chat à partir d'applications autres que Chat.
En savoir plus



Récupérer l'URL du Webhook :
https://chat.googleapis.com/v1/spaces/AAQAMIC63tI/messages?key=AIzaSyDdI0hCZtE6vy…..

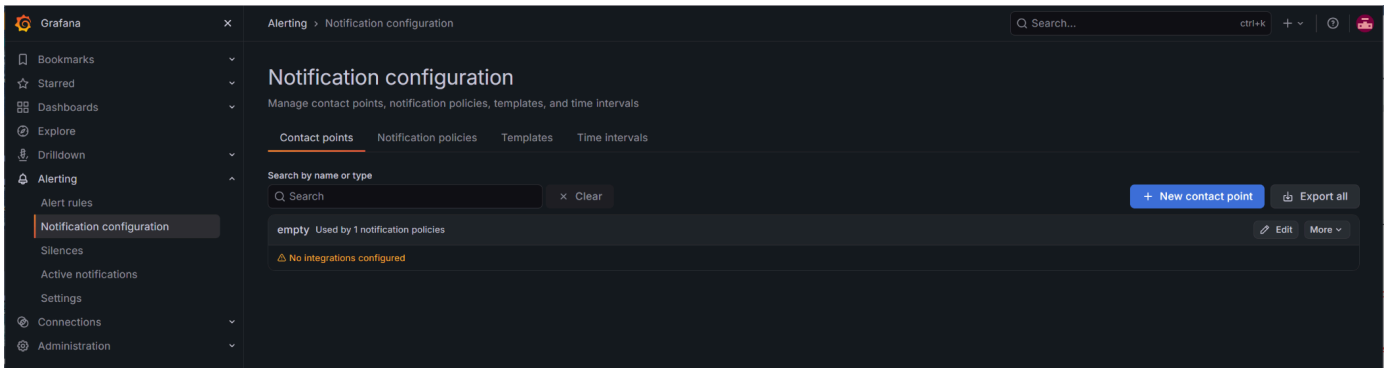
La première phase permet d'interconnecter **Grafana** avec l'espace de discussion **Google Chat** au moyen de cette **URL** :

- Création du point de contact : Déclaration d'un point de contact nommé **Webhook Google Chat** dans le menu **Alerting** de **Grafana** en y insérant l'**URL** générée
- Validation du canal : Vérification de la bonne réception de la notification de test directement sur le salon
- Définition de la destination par défaut : Raccordement du point de contact à la politique de notification par défaut (**Default policy**) pour garantir l'acheminement systématique des messages

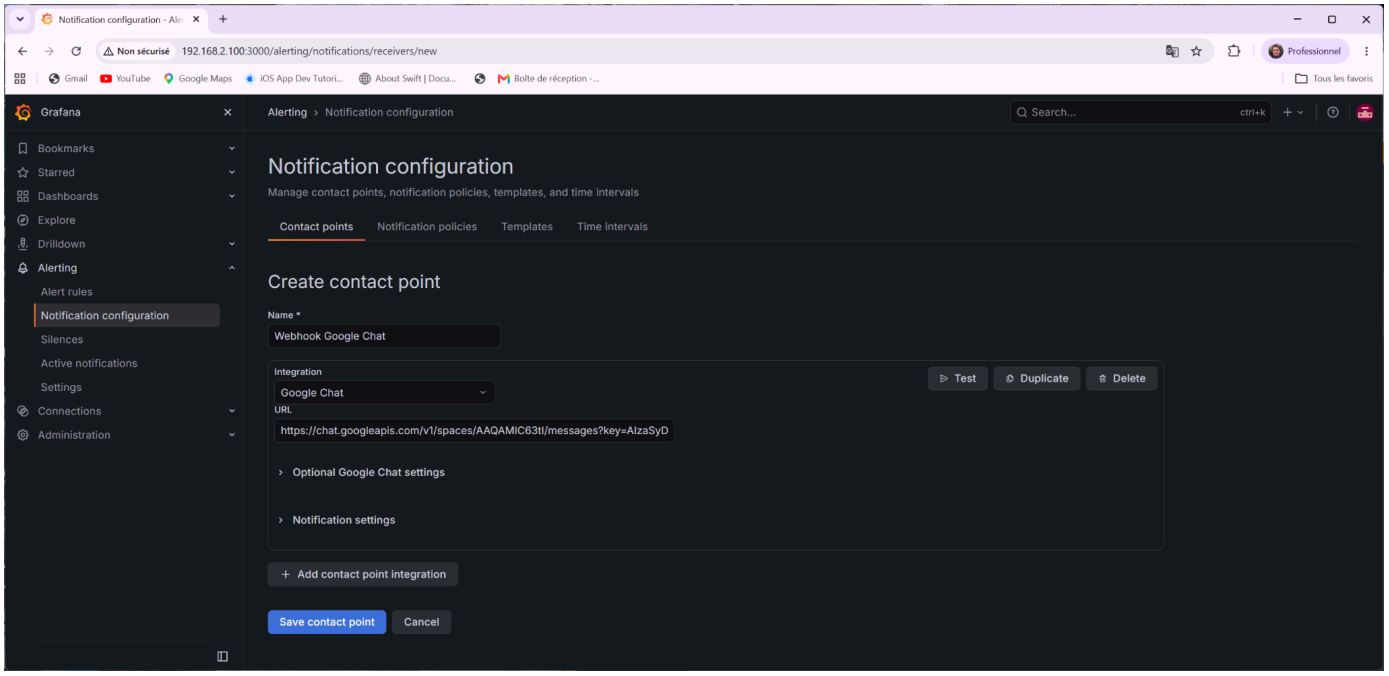
Comment faites-vous pour réaliser cette première phase ?

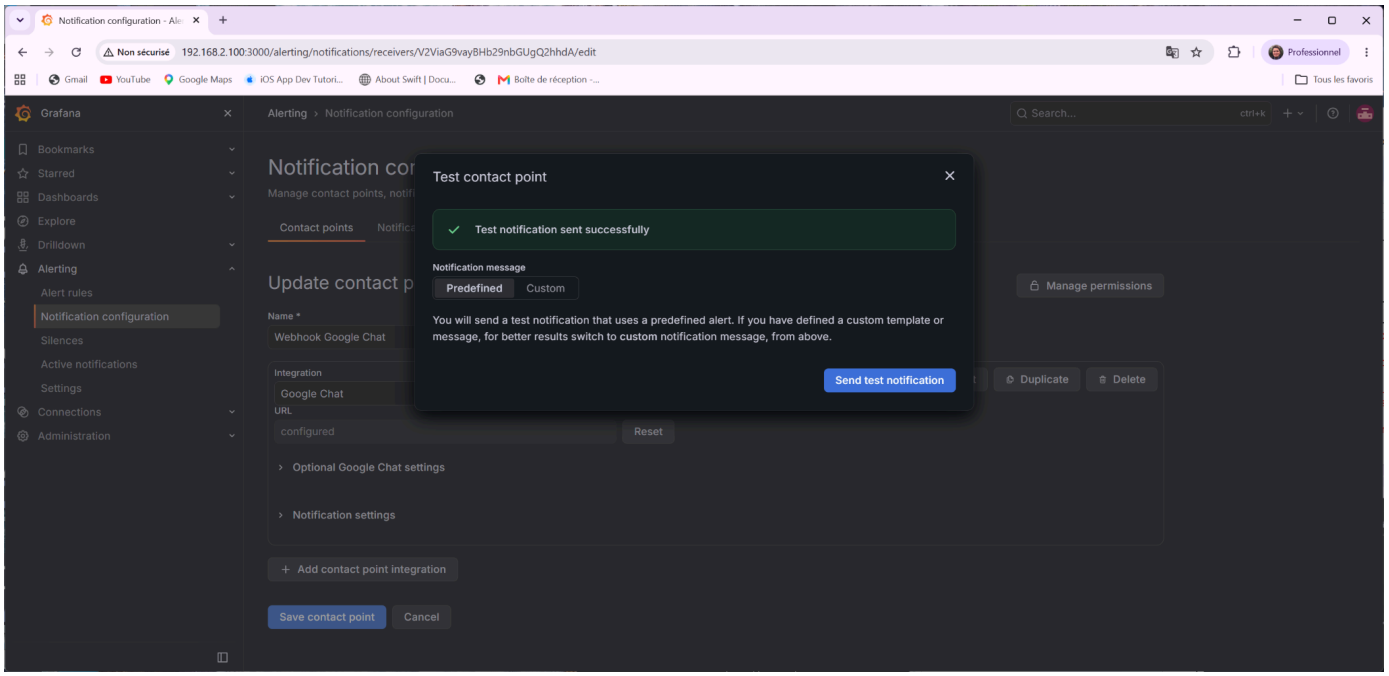
Etape 1 : Créer le Point de Contact (Contact Point)

1. Dans le menu de gauche de Grafana, Aller dans **Alerting → Contact points**
2. Cliquer sur **New contact point**



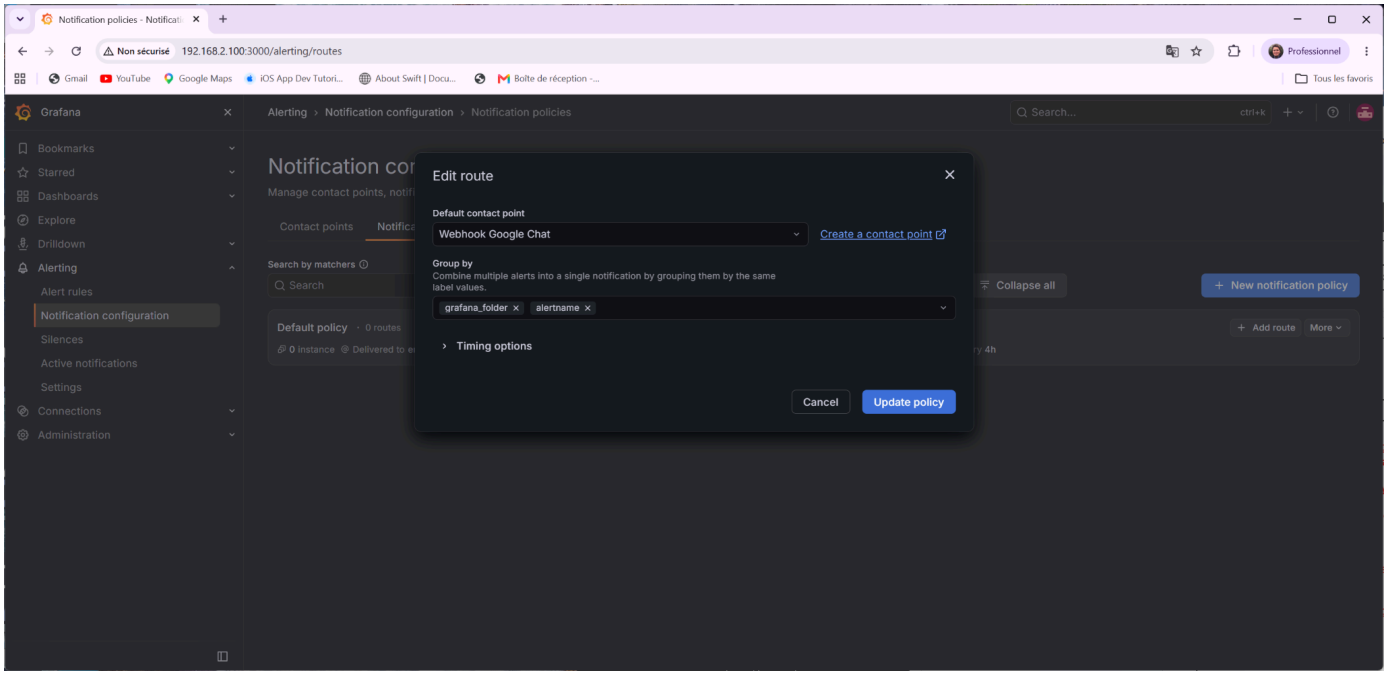
3. Remplir les champs :
    - Name : **Webhook Google Chat**
    - Integration : Choisir **Google Chat** (ou Webhook si on utilise un endpoint générique)
    - URL : Coller l'URL du **Webhook** générée dans le salon Google Chat (ex: [https://chat.googleapis.com/v1/spaces/])
4. Cliquer sur **Test** pour vérifier la bonne réception d'un message de test, puis cliquer sur **Save contact point**



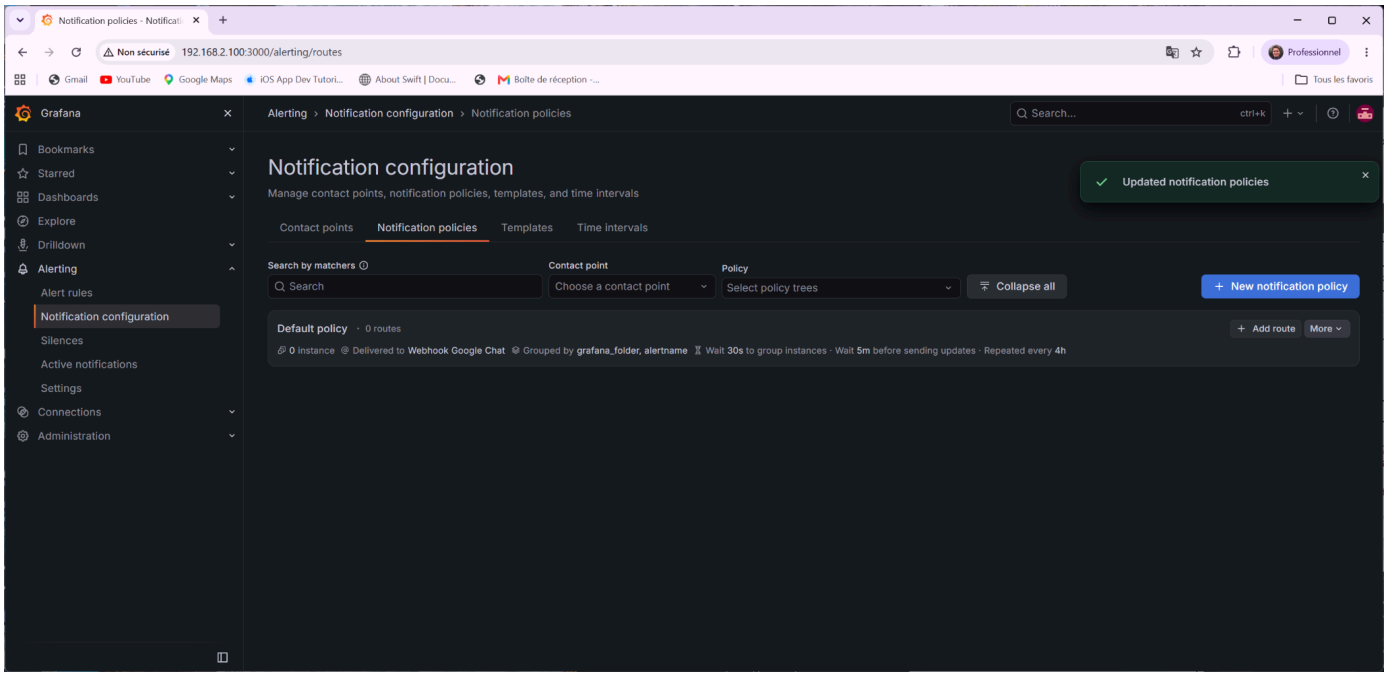


Etape 2 : Définir comme Destination par Défaut (Notification Policy)

1. Aller dans **Alerting → Notification policies**
2. Dans la section **Default policy**, cliquer sur **Edit**
3. Dans le champ **Default contact point**, sélectionner **Webhook Google Chat**



4. Sauvegarder la politique



La seconde phase concerne la création et le paramétrage de la règle d'alerte intitulée Alerte Réflectivité Faible (< 20%) :

- Traitement de la trame brute : Le capteur transmet une valeur sous format hexadécimal (**pdin_hex**). Une requête rédigée en langage Flux extrait puis convertit cette information en un entier décimal directement exploitable.

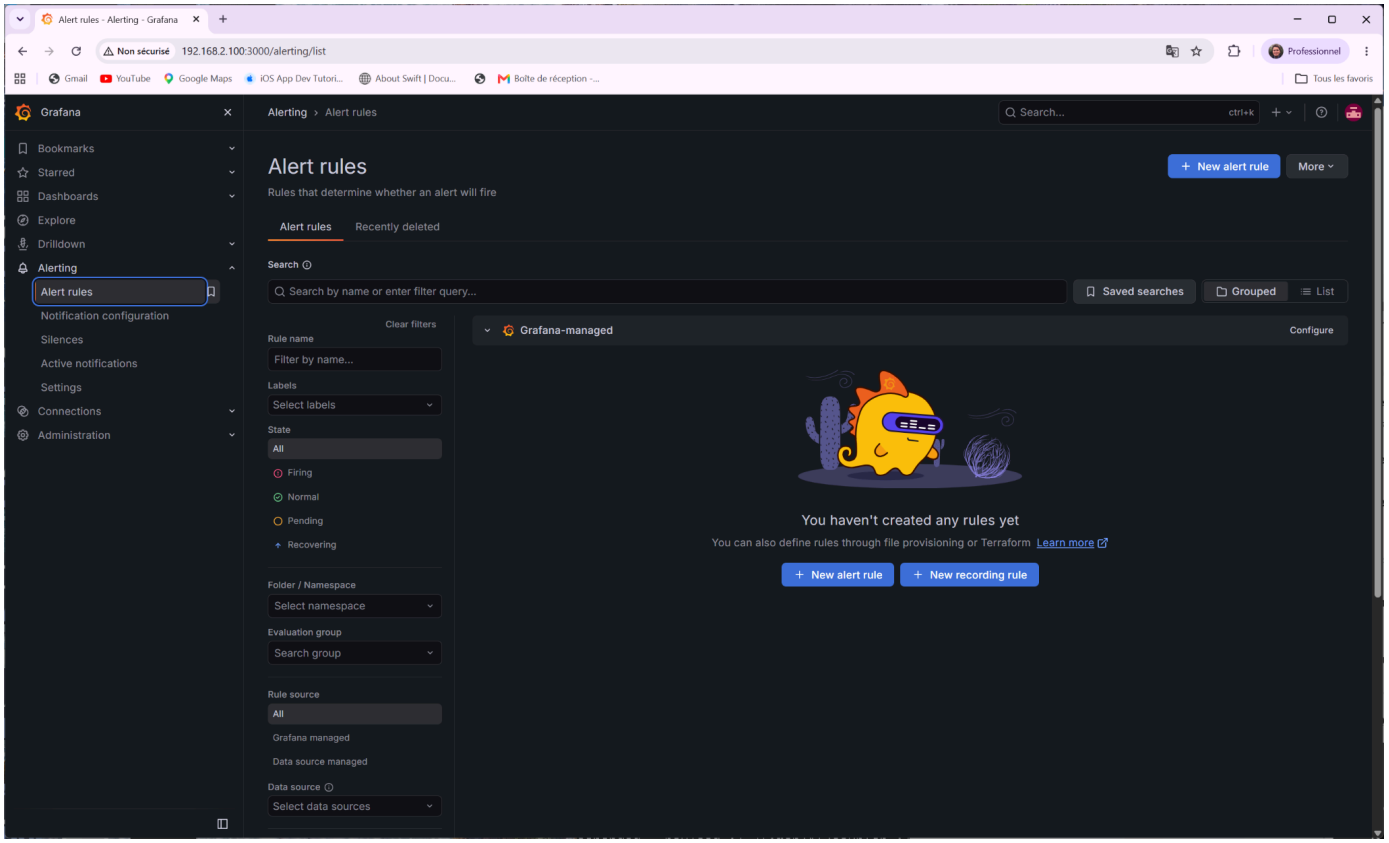
- Configuration du seuil : Application d'une condition de contrôle IS BELOW avec une valeur limite fixée à 20.

- Contrôle et déclenchement : La surveillance s'effectue selon un cycle d'évaluation de 30 secondes. Si la réflectivité reste inférieure à 20 % de manière continue pendant 30 secondes, la règle passe à l'état actif (Firing) et déclenche immédiatement l'envoi de l'alerte sur Google Chat.

Comment faites-vous pour réaliser cette deuxième phase ?

Etape 3 : Exemple de Règle d'Alerte sur la Réflectivité

Dans le menu de gauche, cliquer sur **Alerting** puis sur **Alert rules** :



Cliquer sur le bouton **+ New alert rule** :

Nommer et saisir la requête :

- Rule name : Alerte Réflectivité Faible (< 20%)
- Data source : Choisir influxdb
- Query :

```
import "contrib/bonitoo-io/hex"
import "strings"

from(bucket: "iiot_cassoulet")
 |> range(start: -1m)
 |> filter(fn: (r) => r["_field"] == "pdin_hex")
 |> last()
 |> map(fn: (r) => ({
     r with _value: hex.int(v: strings.substring(v: r._value, start: 8, end: 12))
 }))
```

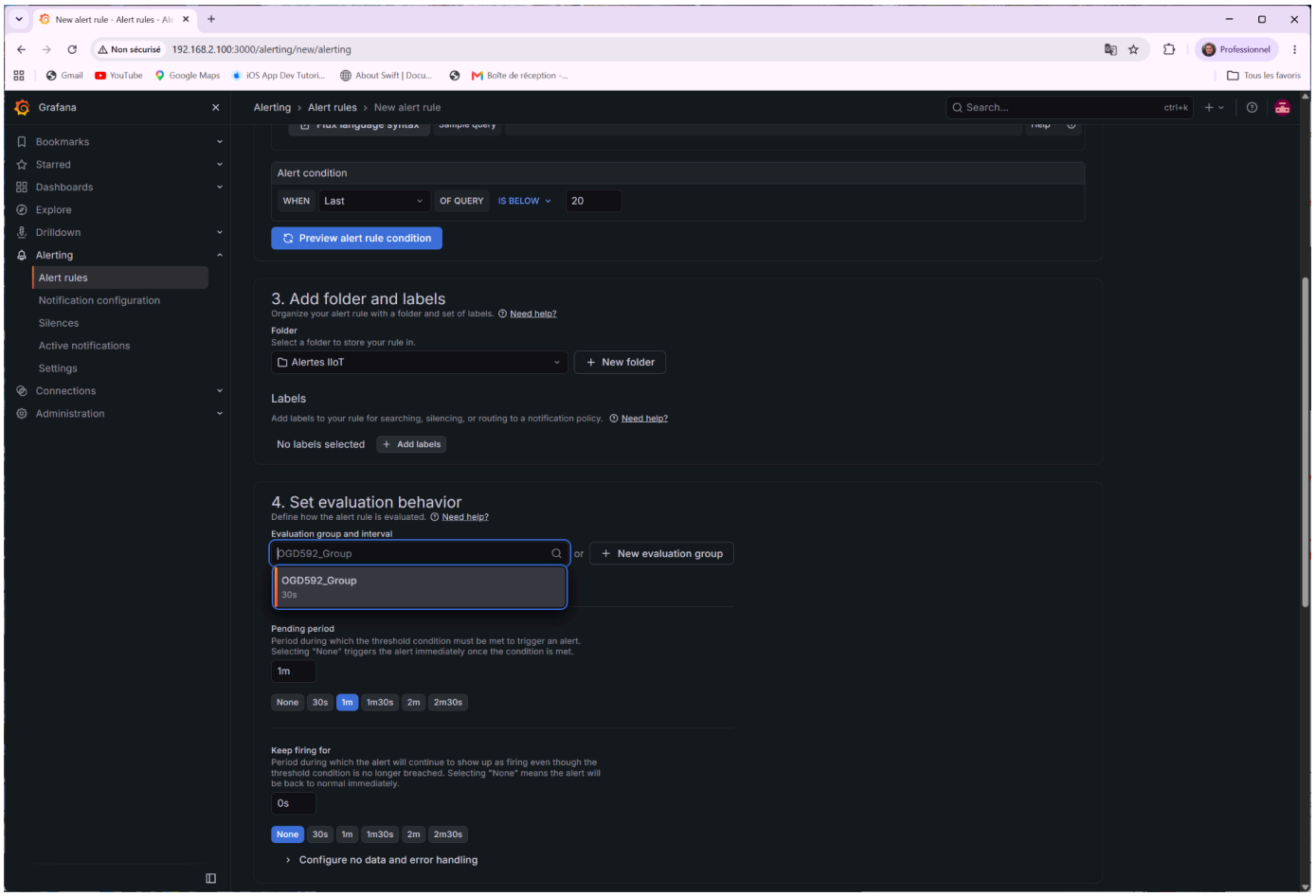
Définir la condition de seuil dans le bloc de condition :

- Condition : Choisir **IS BELOW**
- Valeur : Saisir **20**

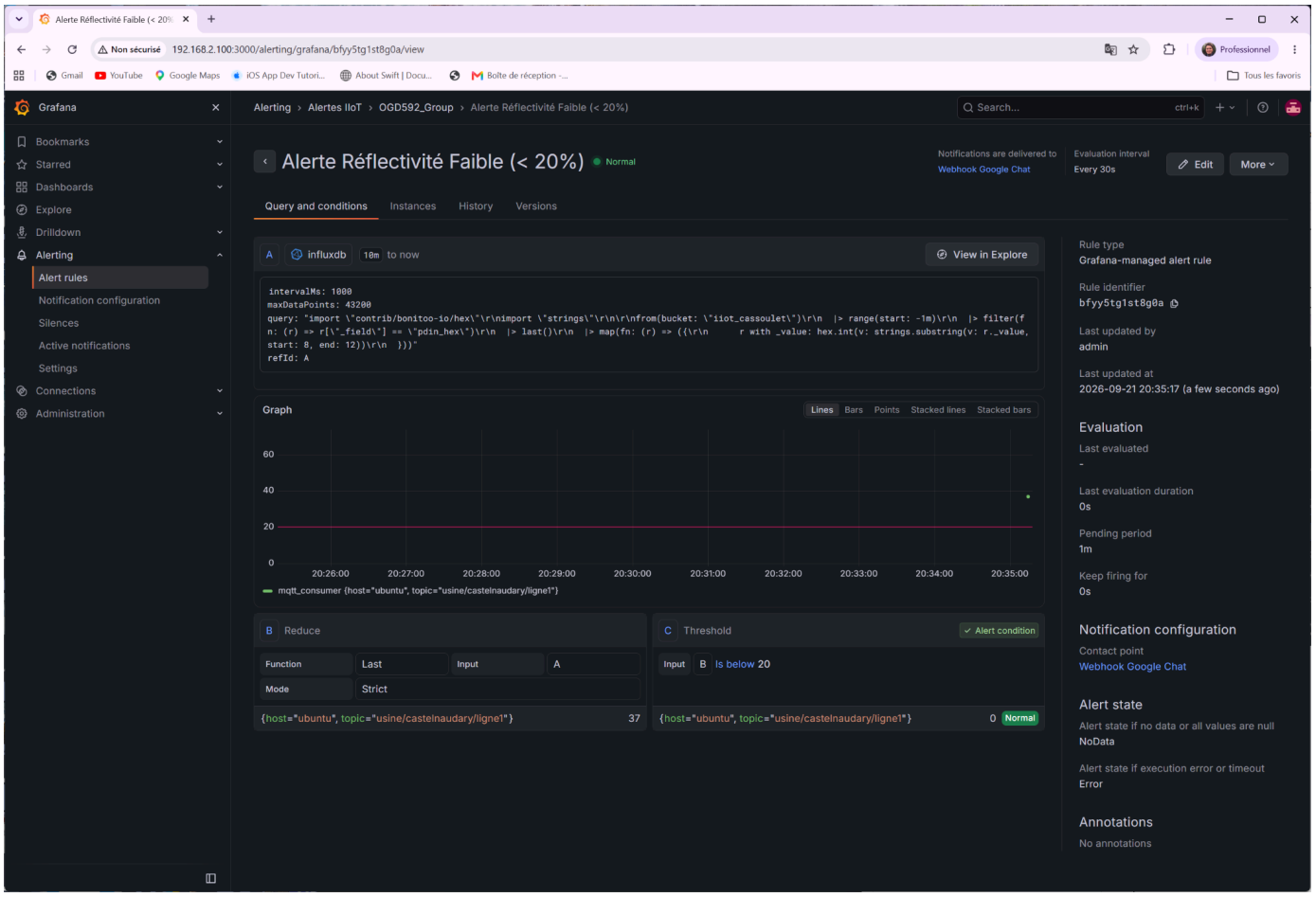
Configurer la fréquence :

Folder : **+ New folder → Alertes IIoT**

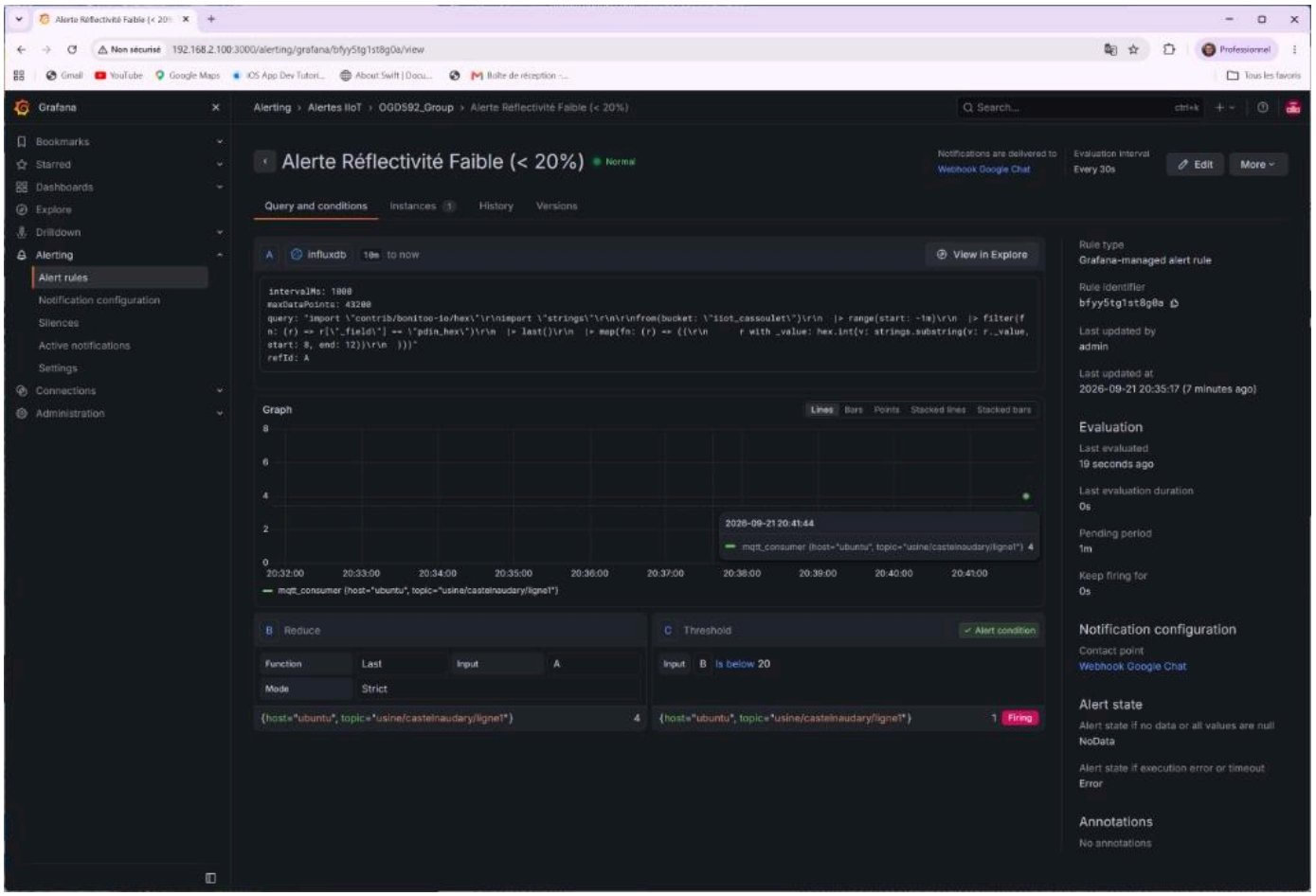
- Evaluation group : **+ New evaluation group → OGD592_Group**
- Evaluate every : 30s
- For : 1m

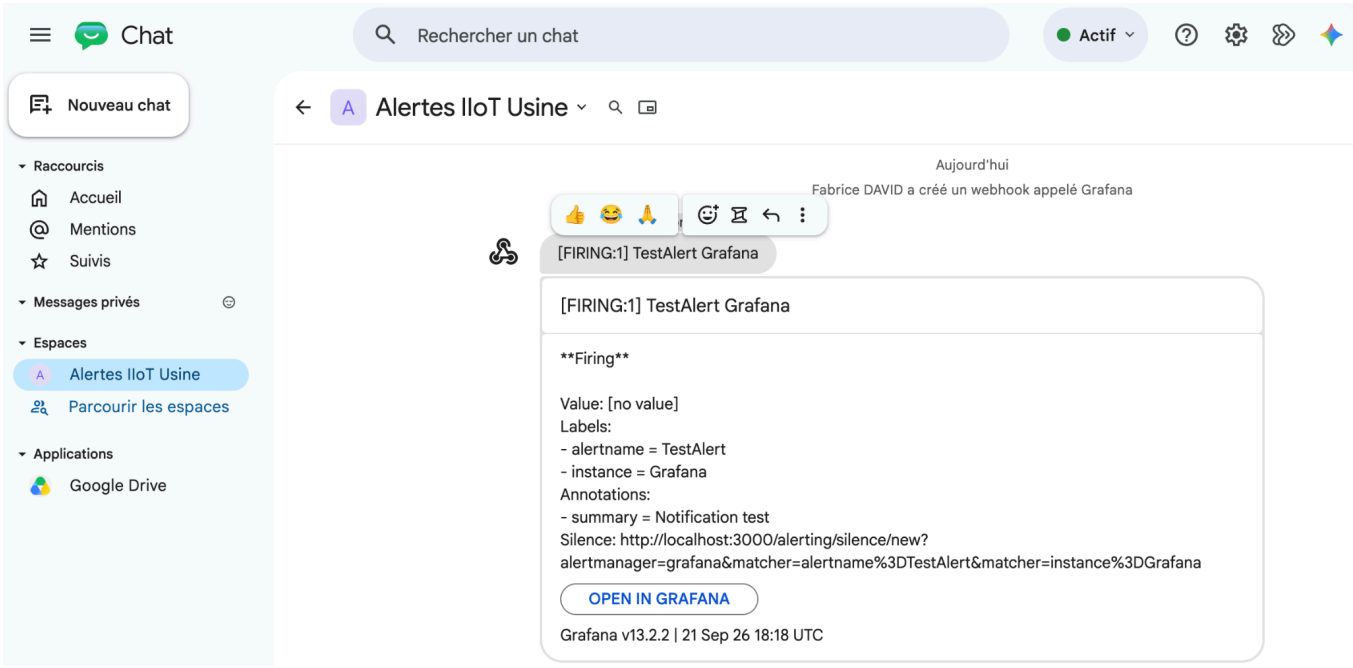


6.6. Enregistrer : En haut à droite


Alerte Réflectivité Faible (< 20%)
192.168.2.100:3000/alerting/grafana/bfyy5tg1st8g0a/view
Grafana
Alerting › Alertes IIoT › OGD592_Group › Alerte Réflectivité Faible (< 20%)
Search...
Bookmarks
Starred
Dashboards
Explore
Drilldown
Alerting
Alert rules
Notification configuration
Silences
Active notifications
Settings
Connections
Administration
Alerte Réflectivité Faible (< 20%)
Normal
Notifications are delivered to
Webhook Google Chat
Evaluation interval
Every 30s
Edit
More
Query and conditions
Instances
History
Versions
A
influxdb
10m to now
View in Explore
intervalMs: 1000
maxDataPoints: 43200
query: "import \"contrib/bonitoo-io/hex\"\r\nimport \"strings\"\r\n\r\nfrom(bucket: \"iiot_cassoulet\")\r\n |> range(start: -1m)\r\n |> filter(fn: (r) => r[\"_field\"] == \"pdin_hex\"]\r\n |> last()\r\n |> map(fn: (r) => ({\r\n r with _value: hex.int(v: strings.substring(v: r._value, start: 8, end: 12))\r\n }))"
refId: A
Graph
Lines
Bars
Points
Stacked lines
Stacked bars
60
40
20
0
20:26:00
20:27:00
20:28:00
20:29:00
20:30:00
20:31:00
20:32:00
20:33:00
20:34:00
20:35:00
mqtt_consumer {host="ubuntu", topic="usine/castelnaudary/ligne1"}
B
Reduce
Function
Last
Input
A
Mode
Strict
{host="ubuntu", topic="usine/castelnaudary/ligne1"}
37
C
Threshold
Alert condition
Input
B
Is below 20
{host="ubuntu", topic="usine/castelnaudary/ligne1"}
0
Normal
Rule type
Grafana-managed alert rule
Rule identifier
bfyy5tg1st8g0a
Last updated by
admin
Last updated at
2026-09-21 20:35:17 (a few seconds ago)
Evaluation
Last evaluated
-
Last evaluation duration
0s
Pending period
1m
Keep firing for
0s
Notification configuration
Contact point
Webhook Google Chat
Alert state
Alert state if no data or all values are null
NoData
Alert state if execution error or timeout
Error
Annotations
No annotations

Réaliser un test en obstruant le capteur et relever les notifications envoyées dans Google Chat.




Chat
Rechercher un chat
Actif
Nouveau chat
Raccourcis
Accueil
Mentions
Suivis
Messages privés
Espaces
Alertes IIoT Usine
Parcourir les espaces
Applications
Google Drive
Alertes IIoT Usine
Aujourd'hui
Fabrice DAVID a créé un webhook appelé Grafana
[FIRING:1] TestAlert Grafana
[FIRING:1] TestAlert Grafana
Firing
Value: [no value]
Labels:
- alertname = TestAlert
- instance = Grafana
Annotations:
- summary = Notification test
Silence: http://localhost:3000/alerting/silence/new?
alertmanager=grafana&matcher=alertname%3DTestAlert&matcher=instance%3DGrafana
OPEN IN GRAFANA
Grafana v13.2.2 | 21 Sep 26 18:18 UTC


Grafana Application 20:43
[FIRING:1] Alerte Réflectivité Faible (< 20%) Alertes IIoT (ubuntu usine/castelnaudary/ligne1)
[FIRING:1] Alerte Réflectivité Faible (< 20%) Alertes IIoT (ubuntu usine/castelnaudary/ligne1)
Firing
Value: B=4, C=1
Labels:
- alertname = Alerte Réflectivité Faible (< 20%)
- grafana_folder = Alertes IIoT
- host = ubuntu
- topic = usine/castelnaudary/ligne1
Annotations:
Source: http://localhost:3000/alerting/grafana/bfyy5tg1st8g0a/view?orgId=1
Silence: http://localhost:3000/alerting/silence/new?
alertmanager=grafana&matcher=__alert_rule_uid__%3Dbfyy5tg1st8g0a&matcher=host%3Dub
untu&matcher=topic%3Dusine%2Fcastelnaudary%2Fligne1&orgId=1
OPEN IN GRAFANA
Grafana v13.2.2 | 21 Sep 26 18:43 UTC


Grafana Application 20:48
[RESOLVED] Alerte Réflectivité Faible (< 20%) Alertes IIoT (ubuntu usine/castelnaudary/ligne1)
[RESOLVED] Alerte Réflectivité Faible (< 20%) Alertes IIoT (ubuntu usine/castelnaudary/ligne1)
Resolved
Value: B=29, C=0
Labels:
- alertname = Alerte Réflectivité Faible (< 20%)
- grafana_folder = Alertes IIoT
- host = ubuntu
- topic = usine/castelnaudary/ligne1
Annotations:
Source: http://localhost:3000/alerting/grafana/bfyy5tg1st8g0a/view?orgId=1
Silence: http://localhost:3000/alerting/silence/new?
alertmanager=grafana&matcher=__alert_rule_uid__%3Dbfyy5tg1st8g0a&matcher=host%3Dub
untu&matcher=topic%3Dusine%2Fcastelnaudary%2Fligne1&orgId=1
OPEN IN GRAFANA
Grafana v13.2.2 | 21 Sep 26 18:48 UTC
Répondre

XI – AUDIT DE CLOISONNEMENT IT/OT ET CYBERSECURITE :

Nous allons réaliser une analyse de l'audit de cloisonnement et de la cybersécurité.

Depuis la Machine Virtuelle (enp0s3 : **192.168.2.100**), tenter un ping vers l'automate PLC OT (**192.168.1.10**).

Résultat attendu : Echec total (**Destination Host Unreachable**). Les réseaux **192.168.1.0/24** et **192.168.2.0/24** ne sont pas routés.

Nous allons faire une capture de trames pour vérifier si on récupère des trames du protocole OT. Taper la commande suivante :

```
sudo tcpdump -i eth0 -n "pn_rt"
```

Résultat attendu : Aucun paquet **PROFINET Real-Time** n'est capturé sur l'interface IT.

Pour s'assurer que la Machine Virtuelle ne sert pas de passerelle entre le WAN (**enp0s8**) et le réseau IT de l'usine (**enp0s3**), que faut-il vérifier dans le fichier de configuration **/etc/sysctl.conf** ?

Pour s'assurer que la Machine Virtuelle ne sert pas de passerelle entre le WAN (**enp0s8**) et le réseau IT de l'usine (**enp0s3**), il faut vérifier dans le fichier de configuration **/etc/sysctl.conf** cette ligne :

```
net.ipv4.ip_forward = 0
```

XII – CONCLUSION :

Pour conclure cette étude et ouvrir sur les perspectives d'optimisation de l'architecture IIoT, l'analyse technique globale permet de dégager trois axes clés concernant le traitement des données et la montée en charge.

Quel serait l'impact sur les performances si l'infrastructure devait intégrer 50 capteurs OGD592 supplémentaires tout en conservant le décodage hexadécimal via **Flux** dans **Grafana** ?

La charge processeur du serveur **InfluxDB/Grafana** augmenterait de manière exponentielle. Chaque rafraîchissement de tableau de bord et chaque cycle d'évaluation d'alerte obligeraient le moteur **Flux** à exécuter les fonctions **hex.int** et **strings.substring** sur des milliers de séries temporelles. Cela provoquerait de la latence sur les graphiques, des lenteurs d'affichage et un risque de retard dans le déclenchement des alertes de sécurité.

Faut-il ne conserver que les valeurs décimales converties ou est-il pertinent d'archiver la trame brute **pdin_hex** dans **InfluxDB** ?

La bonne pratique consiste à traiter la donnée dès l'ingestion pour enregistrer uniquement les métriques décimales nettoyées (**distance** et **réflectivité**) dans le **bucket** principal de production. Pour des besoins d'audit ou de diagnostic matériel approfondi, la trame brute **pdin_hex** peut être conservée dans un **bucket** secondaire dédié, assorti d'une politique de rétention courte (par exemple de 7 à 14 jours), cela évite de surcharger la base de données à long terme.

Quel est l'intérêt de déporter la conversion hexadécimale dans **Telegraf** (via **Starlark**) ou directement au niveau du Master **IO-Link** / de l'automate ?

En déportant le calcul au plus près du terrain (**Edge Computing**), cela permet de séparer clairement les responsabilités. La couche d'acquisition (**Telegraf** ou Master **IO-Link**) prend en charge le découpage protocolaire du capteur tandis que la couche de restitution (**Grafana**) se concentre uniquement sur la visualisation. Cela simplifie la rédaction des requêtes **Flux**, réduit la taille des paquets sur le réseau **MQTT** et garantit des performances constantes, quel que soit le nombre de capteurs déployés.

Comparer le traitement de l'alerte d'encrassement côté **OT** (TP3 — Voyant automate **%Q0.0**) et côté **IT** (TP6 — Notification Grafana). En quoi sont-ils complémentaires ?

Le traitement **OT** déclenche une réaction déterministe immédiate sur la machine en allumant un voyant physique pour sécuriser la production locale, alors que le traitement **IT** traite l'événement de manière asynchrone pour informer la maintenance à distance. Ces deux approches sont complémentaires car l'**OT** protège le procédé en temps réel pendant que l'**IT** assure la traçabilité et la coordination globale des interventions.

Résumer la complémentarité entre la réactivité temps réel industrielle et la centralisation des données de supervision **IT** dans l'usine **4.0**.

L'architecture en **Y** permet de marier la fiabilité et la sécurité de la boucle de commande industrielle **OT** avec la souplesse et la puissance d'analyse des outils informatiques **IT**. Cette étanchéité garantit que l'exploitation massive des données de production n'altère jamais la réactivité et la sécurité du pilotage machine.

Grille de Notation			Validation Compétences
Etude préalable (Supervision, **Grafana** et **Webhooks**)		**/3**	
Installation et Configuration de **Grafana** (Source **InfluxDB v2)**		**/4**	
Tableau de bord décisionnel & partage Mode **Kiosk** (Jauges, Courbes, Stat)		**/3**	**Valide C05**
Configuration du système d'alerte et **Webhook Google Chat**		**/3**	
Conclusion et analyse technique (Stockage Hexadécimal et Décimal, **Starlark**/**Edge**)		**/6**	**Valide C04**
Barème de notation total		**/20**	