**Rapport des incidents de monitoring :**

Voici le **récapitulatif complet affiché directement**, structuré en étapes logiques, à partir du moment où tu **as cessé d’utiliser pfSense** comme cible Prometheus, en privilégiant un usage actif et sécurisé de **Grafana + Prometheus**, sans violer la politique de sécurité de ton environnement :

**CONTEXTE DE RESTRICTION INTERNE**

**On ne peut pas** :

* **Scanner pfSense (ni nmap, ni netstat, ni sniffing),**
* **Interroger les IP locales,**
* **Forcer les requêtes entrantes/sortantes sur les ports sensibles de la DMZ.**

**Ou ne peut uniquement :**

* Interroger Prometheus et Grafana **en HTTP sur IP publique**,
* Afficher des métriques **déjà exposées** par Node Exporter, Telegraf, Blackbox Exporter,
* Activer/désactiver des services en root ou sudo **sur l’instance Ubuntu Jitsi**.

**✅ OBJECTIF : UTILISATION ACTIVE DE GRAFANA**

1. **Utilisation concrète de Grafana**
   * Tu veux exploiter **des dashboards utiles**, pas seulement ceux liés à up{} ou prometheus\_tsdb\_\*.
   * Tu avais déjà importé un dashboard JSON de base avec 3 panels (Scrape Duration, Active Targets, TSDB series).

**ÉTAPES OPÉRATIONNELLES**

**1. Analyse initiale**

* Prometheus tournait en tant que service, **mais les cibles ne répondaient pas** (health: unknown).
* La configuration /etc/prometheus/prometheus.yml était correcte, mais Prometheus ne lisait pas ce fichier.
* Prometheus lançait son propre fichier /opt/prometheus/prometheus.yml via le service systemd.

**2. Résolution par correction du service**

[Service]

ExecStart=/opt/prometheus/prometheus \

--config.file=/etc/prometheus/prometheus.yml \

--web.listen-address=0.0.0.0:9091

* **Redémarrage via :**

sudo systemctl daemon-reexec

sudo systemctl daemon-reload

sudo systemctl restart prometheus

* **Résultat :** Prometheus tournait bien (PID actif), mais certains targets n’étaient pas accessibles (ex. node\_exporter invisible).

**3. Debug des targets :**

**Commandes utiles utilisées** :

curl http://localhost:9091/api/v1/targets | jq '.data.activeTargets[] | {job: .labels.job, instance, health, lastError}'

* node\_exporter, telegraf, blackbox\_http → visibles ou "up" parfois
* pfsense\_exporter → non utilisable (refus interne ou port bloqué)

**4. Redémarrage manuel réussi**

sudo /opt/prometheus/prometheus \

--config.file=/etc/prometheus/prometheus.yml \

--web.listen-address="0.0.0.0:9091"

**Résultat :**

* Prometheus fonctionne bien si lancé manuellement.
* Le port 9091 doit être **libre**, sinon message : bind: address already in use.

**5. Node Exporter ne remonte pas**

* Il est bien lancé (ps aux | grep node\_exporter),
* Il écoute sur \*:9100 (ss -tulpn | grep 9100),
* Port autorisé (sudo ufw allow 9100),
* Présent dans le YAML : targets: ['37.156.46.238:9100'],
* **Mais toujours non détecté dans Grafana.**

**Hypothèses :**

* node\_exporter lancé avec l’utilisateur node\_exporter → pas de droits sur certaines métriques ?
* Problème de pare-feu GCP ou règles réseau non testées ?
* Prometheus ne lit pas encore /etc/prometheus/prometheus.yml si c’est le service qui est actif, pas la commande manuelle.

**RECOMMANDATIONS POUR GRAFANA**

**1. Filtrage utile dans PromQL :**

* up{job="node\_exporter"} → pour tester disponibilité
* node\_cpu\_seconds\_total → charge CPU
* node\_memory\_MemAvailable\_bytes → RAM libre
* node\_filesystem\_avail\_bytes → espace disque

**2. Exemple de panels à ajouter :**

* 📈 CPU par core :
* rate(node\_cpu\_seconds\_total{mode!="idle"}[1m])
* 📊 RAM disponible :
* node\_memory\_MemAvailable\_bytes / node\_memory\_MemTotal\_bytes \* 100
* 💾 Disque root :
* node\_filesystem\_avail\_bytes{mountpoint="/"}

**✅ PROCHAINES ACTIONS**

1. Vérifier **le fichier YAML utilisé réellement** par Prometheus :
2. sudo systemctl cat prometheus
3. Repasser en lancement manuel avec logs visibles :
4. sudo /opt/prometheus/prometheus \
5. --config.file=/etc/prometheus/prometheus.yml \
6. --web.listen-address="0.0.0.0:9091"
7. Revoir la configuration réseau externe (GCP, infomaniak ?) :
   * S’assurer que les ports 9100 (node), 9273 (telegraf), etc. sont **accessibles** publiquement ou via tunnel SSH local.
8. Ajouter un dashboard complet Grafana avec panels ci-dessus (JSON prêt possible sur demande).