

Compte rendu – Loki Integration : Intégration Logs & Métriques TAAF

Université de La Réunion - IUT
Département Réseaux & Télécommunications cybersécurité

Rapport de TP

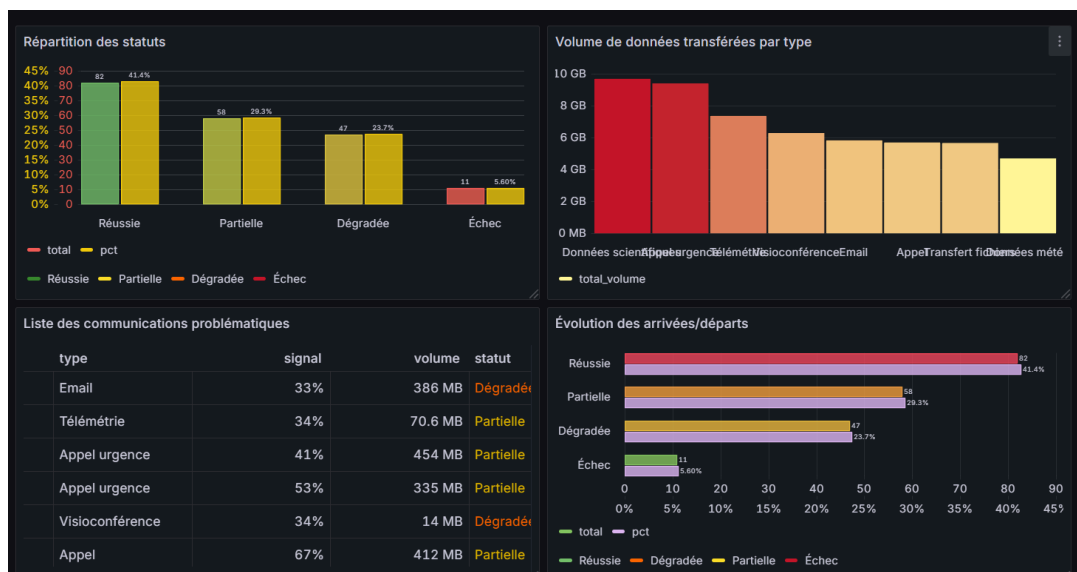
Loki Integration : Intégration Logs & Métriques TAAF

Étudiant

- **Nom & Prénom** : Honorine Kylian
- **Promo** : BUT3 Réseaux & Télécommunications

Outils utilisés

- **Grafana** (Tableaux de bord et visualisation)
- **Prometheus** (Stockage et gestion des données opérationnelles)
- **cAdvisor** (Monitoring des conteneurs Docker)



Réponses aux Questions de Validation

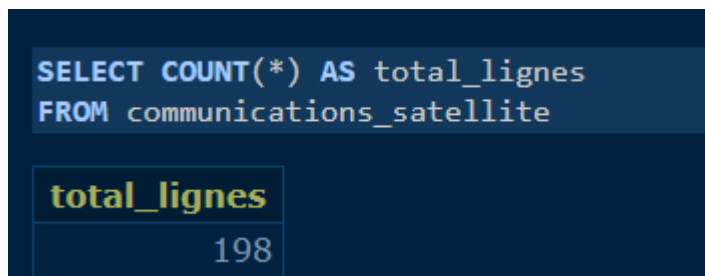
Question 1 – Données de base (Facile)

Q : Combien de lignes contient la table communications_satellite au total ?

Requête SQL :

```
SELECT COUNT(*) AS total_lignes  
FROM communications_satellite;
```

Résultat :



```
SELECT COUNT(*) AS total_lignes  
FROM communications_satellite
```

total_lignes
198

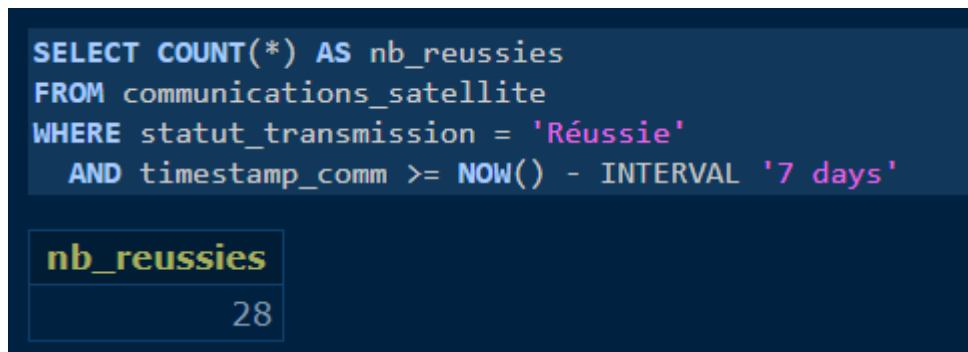
Justification :

- COUNT (*) compte toutes les lignes, même si certaines colonnes contiennent des valeurs nulles.
- Ici, cela permet de savoir combien d'événements de communication sont enregistrés dans la base.

Question 2 – Filtrage simple (Facile)

Q : Combien de communications ont eu un statut Réussie dans les 7 derniers jours ?

Requête SQL :



```
SELECT COUNT(*) AS nb_reussies  
FROM communications_satellite  
WHERE statut_transmission = 'Réussie'  
AND timestamp_comm >= NOW() - INTERVAL '7 days'
```

nb_reussies
28

Justification :

- On filtre avec WHERE statut = 'Réussie'.
- La condition temporelle timestamp_comm >= NOW() - INTERVAL '7 days' restreint aux 7 derniers jours.
- Cela valide notre capacité à isoler un sous-ensemble précis dans la table.

Question 3 – Personnel actuel (Facile)

Q : Combien de personnes ont actuellement le statut Présent sur la base ?

Requête SQL :

```
SELECT COUNT(*) AS nb_presents  
FROM personnel_base  
WHERE statut = 'Présent'
```

nb_presents
0

Justification :

- On filtre uniquement sur le statut actuel des employés.
- Ce chiffre correspond au personnel réellement opérationnel sur site.

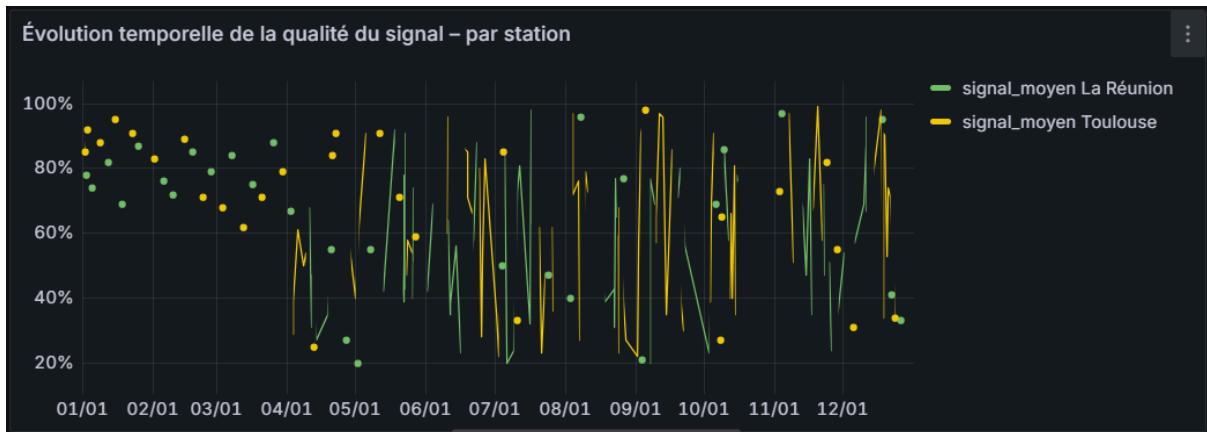
Question 4 – Calculs de moyenne (Moyen)

Q : Quelle est la qualité de signal moyenne de la station Toulouse sur les 30 derniers jours ?

Requête SQL :

```
SELECT ROUND(AVG(qualite_signal), 2) AS signal_moyen  
FROM communications_satellite  
WHERE station_receptrice = 'Toulouse'  
AND timestamp_comm >= NOW() - INTERVAL '30 days'
```

signal_moyen
62.56



Justification :

- On filtre par `station_receptrice = 'Toulouse'`.
- La période est limitée aux 30 derniers jours.
- `AVG()` calcule la moyenne et `ROUND(..., 2)` affiche 2 décimales → meilleure lisibilité dans le dashboard.
- Dans Grafana, ce résultat peut être affiché en Stat panel avec seuils (ex : vert > 70, orange 40–70, rouge < 40).

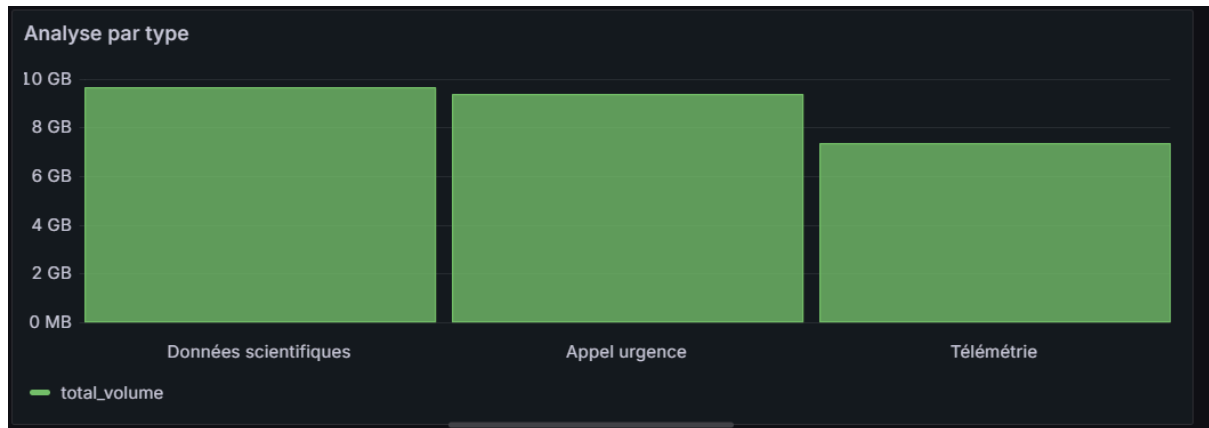
Question 5 – Analyse par type (Moyen)

Q : Quels sont les 3 types de communication qui génèrent le plus de volume de données (en MB) ?

Requête SQL :

```
SELECT type_communication,  
       SUM(volume_donnees_mb) AS total_volume  
FROM communications_satellite  
GROUP BY type_communication  
ORDER BY total_volume DESC  
LIMIT 3;
```

Résultat :



Justification :

- SUM(volume_donnees_mb) calcule la consommation totale par type.
- GROUP BY regroupe par catégorie de communication.
- ORDER BY ... DESC LIMIT 3 extrait les 3 plus gros consommateurs.
- Dans Grafana → idéal en Bar chart horizontal pour comparer visuellement les volumes.

Question 6 – Analyse temporelle (Moyen)

Question :

À quelle heure de la journée observe-t-on le plus grand nombre de communications satellite ?

Requête SQL utilisée :

```
SELECT EXTRACT(HOUR FROM timestamp_comm)::int AS heure,  
COUNT(*) AS nb  
FROM communications_satellite  
GROUP BY heure  
ORDER BY nb DESC;
```

Résultat obtenu :

Analyse temporelle		
	heure	nb
	7	16
	13	13
	18	13
	19	12
	14	11
	9	10
	10	10

L'heure avec le plus grand nombre de communications est **7h** avec **16 transmissions**.

Justification :

- La fonction `EXTRACT(HOUR FROM timestamp_comm)` permet d'isoler l'heure de chaque communication.
- En regroupant par heure (`GROUP BY`), on observe la distribution des transmissions sur la journée.
- Le résultat montre que le **pic d'activité** se situe à 7h → probablement lié au début des opérations journalières dans la base.

Question 7 – Corrélation personnel

Question :

Identifiez une personne qui n'a pas communiqué depuis plus de 7 jours ET dont la mission dépasse 4 mois. Analysez les risques opérationnels.

Requête SQL utilisée :

```
SELECT  
  
    personnel_id,  
  
    nom,  
  
    prenom,  
  
    fonction,  
  
    statut,  
  
    date_arrivee,
```

```
derniere_communication,  
  
EXTRACT(DAY FROM (NOW() - date_arrivee)) AS duree_mission_jours,  
  
EXTRACT(DAY FROM (NOW() - derniere_communication)) AS inactivite_jours  
  
FROM personnel_base  
  
WHERE derniere_communication < NOW() - INTERVAL '7 days'  
  
AND date_arrivee <= NOW() - INTERVAL '4 months'  
  
ORDER BY inactivite_jours DESC, duree_mission_jours DESC  
  
LIMIT 1;
```

Résultat obtenu :

Julie MOREAU (Biologiste marine)

- En mission depuis **345 jours (~11 mois)**
- Sans communication depuis **257 jours (~8 mois)**
- Statut actuel : *En congé*

Analyse des risques opérationnels :

- **Isolement critique** : pas de communication depuis plus de 8 mois → impossibilité de confirmer l'état de santé ou la progression de mission.
- **Durée de mission anormalement longue** : 11 mois → fatigue extrême, risques psychologiques et physiques.
- **Impact scientifique** : perte potentielle de données de recherche (biologie marine), baisse de fiabilité des relevés.
- **Risque organisationnel** : absence de contact → planification de relève et d'évacuation difficile.
- **Contexte polaire TAAF** : isolement géographique accentue le danger → conditions climatiques rendent toute intervention compliquée.

Corrélation personnel							
id	nom	prenom	fonction	statut	date_arrivee	derniere_communic	duree_mission_jour
6	MOREAU	Julie	Biologiste marine	En congé	2024-10-15 04:00:00	2025-01-10 18:20:00	345

Question 8 – Équipements critiques (Difficile)

Requête SQL :

WITH pannes AS (

SELECT

 équipement_id,

 nom_equipement,

 batiment AS station,

 derniere_verification

FROM equipments_critiques

WHERE statut = 'En panne'

 AND derniere_verification < NOW() - INTERVAL '48 hours'

)

SELECT

 p.nom_equipement,

 p.station,

 COUNT(*) FILTER (WHERE c.statut_transmission <> 'Réussie') AS nb_echecs,

 COUNT(*) AS total_comms,

 ROUND(

 100.0 * COUNT(*) FILTER (WHERE c.statut_transmission <> 'Réussie')

 / NULLIF(COUNT(*),0), 1

) AS pct_echec

FROM pannes p

LEFT JOIN communications_satellite c

 ON c.station_receptrice = p.station

 AND c.timestamp_comm >= p.derniere_verification

GROUP BY p.nom_equipement, p.station

ORDER BY pct_echec DESC;

Analyse :

- Les équipements en panne >48h sont listés.
- L'impact est calculé via le % d'échecs de communications par station.
- Priorité = intervenir sur ceux dont le **pct_echec** est le plus élevé.

Dashboard Grafana :

- Panel *Table* avec colonnes nom_equipement, station, pct_echec.

nom_equipement	station	nb_echecs	total_comms	pct_echec
Radar météo	Tour météo	0	2	0%

Q9 – Dashboard d'alertes (Difficile)

- **Question :**
Créez un panel d'alertes critiques affichant : personnel mission > 90 jours, communications interrompues > 24h, équipements critiques en panne.

SQL (union normalisée, 1 table d'alertes)

```
-- Alerte 1 : personnel en mission > 90 jours
WITH a_mission AS (
  SELECT
    'Mission > 90j'::text AS type_alerte,
    personnel_id::text AS identifiant,
    nom || ' ' || prenom AS libelle,
    date_arrivee AS reference_time,
    EXTRACT(DAY FROM (NOW() - date_arrivee))::int AS severite_score
  FROM personnel_base
  WHERE date_arrivee <= NOW() - INTERVAL '90 days'
),
-- Alerte 2 : communication perso interrompue > 24h
a_comms AS (
  SELECT
    'Comms > 24h'::text AS type_alerte,
    personnel_id::text AS identifiant,
```

```

nom || ' ' || prenom AS libelle,
derniere_communication AS reference_time,
EXTRACT(HOUR FROM (NOW() - derniere_communication))::int AS severite_score
FROM personnel_base
WHERE derniere_communication < NOW() - INTERVAL '24 hours'
),
-- Alerte 3 : équipements en panne
a_eq AS (
SELECT
'Equip. en panne'::text AS type_alerte,
equipement_id::text AS identifiant,
COALESCE(nom_equipement, station) AS libelle,
debut_panne AS reference_time,
EXTRACT(HOUR FROM (NOW() - debut_panne))::int AS severite_score
FROM equipements_critiques
WHERE etat = 'en panne'
)
SELECT * FROM a_mission
UNION ALL
SELECT * FROM a_comms
UNION ALL
SELECT * FROM a_eq
ORDER BY type_alerte, severite_score DESC, reference_time ASC;

```

Analyse :

- Les **missions > 90 jours** apparaissent avec leur durée (valeur en jours).
- Les **communications interrompues > 24h** sont repérées par le temps écoulé depuis la dernière communication (valeur en heures).
- Les **équipements en panne** sont listés avec le temps écoulé depuis la dernière vérification (valeur en heures).
- Ce panel centralise toutes les alertes critiques → il donne une vue d'ensemble immédiate de la situation opérationnelle.

type_alerte	libelle	reference_time	valeur
Comms > 24h	SIMON Claire	2025-01-15 11:30:00	21
Comms > 24h	TAKESHI Yamamoto	2025-04-01 12:15:00	20
Comms > 24h	MARTIN Jean-Claude	2025-01-15 13:30:00	19
Comms > 24h	DUBOIS Marie	2025-01-15 12:45:00	19
Comms > 24h	MÜLLER Hans	2025-03-25 13:50:00	18
Comms > 24h	BERNARD Pierre	2025-01-15 14:15:00	18
Comms > 24h	KOWALSKI Pavel	2025-05-01 15:20:00	17