R5.Cyber.

Modou Diop

RT3

5/ Compte rendu de supervision d'un autre service/équipement **(SAE5.Cyber.03)**

Objectif de la supervision :

Nous souhaitons maintenant étendre cette surveillance à notre base de données MySQL

L'objectif de cette supervision est de garantir la disponibilité, la performance et la sécurité de la base de données MySQL en surveillant des éléments clés tels que l'utilisation des ressources, les erreurs de connexion, les requêtes lentes et les performances globales du service.

Sommaire

Rappel sur MySQL

MySQL est un système de gestion de base de données relationnelle (SGBDR) open-source, très populaire pour le développement d'applications web et la gestion de données. Il fonctionne selon le modèle client-serveur et est basé sur le langage SQL (Structured Query Language) pour la gestion des bases de données.

Pour superviser le service MySQL, on peut utiliser la suite Elastic (Elasticsearch, Logstash, Kibana) ainsi que des outils de monitoring spécifiques à MySQL.

- **Elasticsearch** : Utilisé pour stocker et indexer les logs et les métriques de performance.
- **Logstash** : Utilisé pour collecter les logs générés par MySQL et les envoyer vers Elasticsearch.
- **Kibana** : Pour visualiser et analyser les métriques et logs dans des tableaux de bord personnalisés.

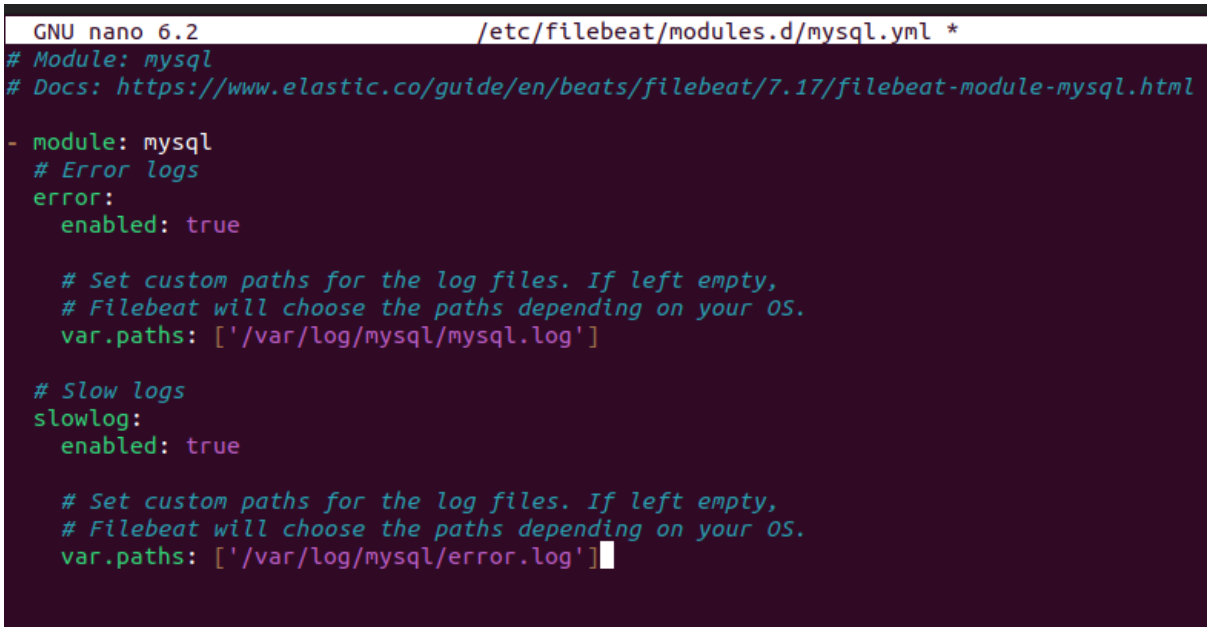
Points clés de la configuration MySQL

1. Activation du module MySQL :

Le fichier `mysql.yml` indique clairement que le module MySQL est activé.

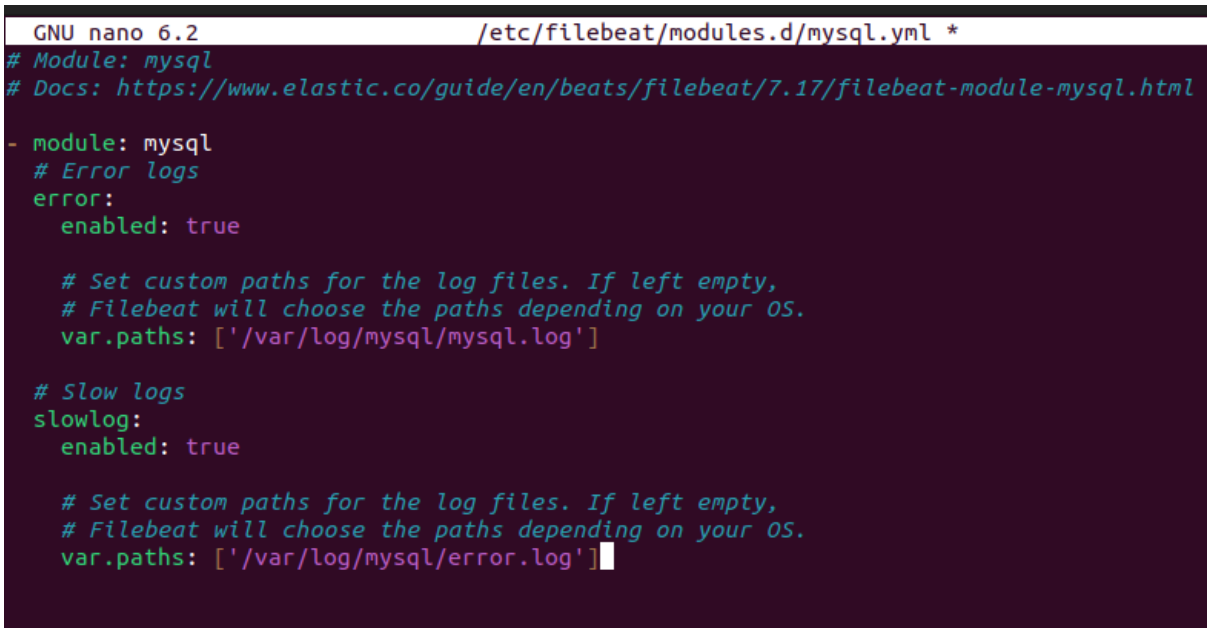
2. Collecte des logs d'erreurs :

Filebeat est configuré pour collecter les logs d'erreurs de MySQL, généralement situés dans `/var/log/mysql/error.log`.



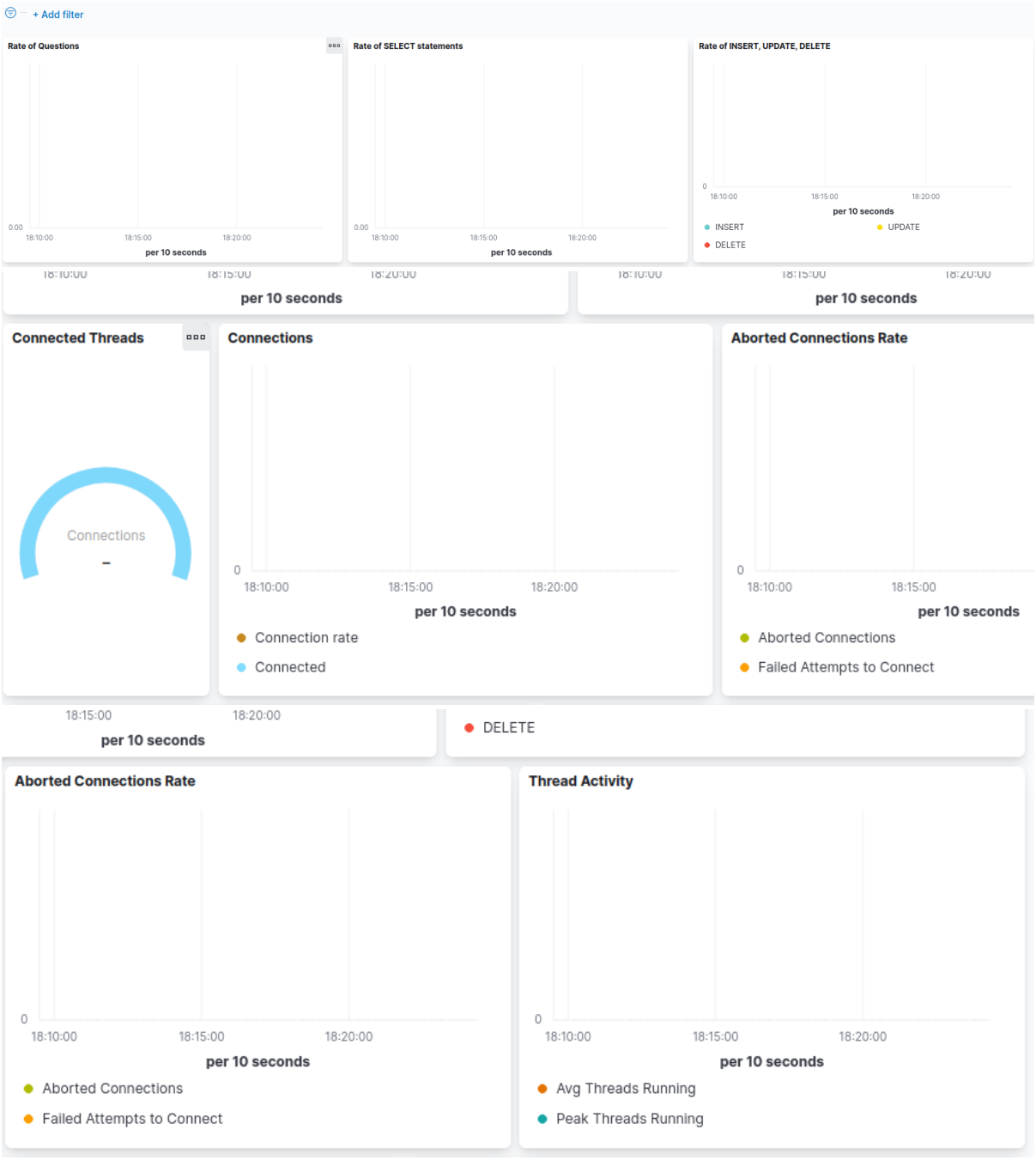
3. Collecte des logs de requêtes lentes

Filebeat est également configuré pour collecter les logs de requêtes lentes, ce qui est crucial pour identifier les goulots d'étranglement dans vos requêtes SQL.



Supervisions

Le tableau de bord que nous avons offre une vue synthétique de l'activité de votre base de données MySQL.


+ Add filter
Rate of Questions
0.00
18:10:00
18:15:00
18:20:00
per 10 seconds
Rate of SELECT statements
0.00
18:10:00
18:15:00
18:20:00
per 10 seconds
Rate of INSERT, UPDATE, DELETE
0
18:10:00
18:15:00
18:20:00
per 10 seconds
INSERT
UPDATE
DELETE
per 10 seconds
per 10 seconds
Connected Threads
Connections
–
Connections
0
18:10:00
18:15:00
18:20:00
per 10 seconds
Connection rate
Connected
Aborted Connections Rate
0
18:10:00
18:15:00
per 10 seconds
Aborted Connections
Failed Attempts to Connect
18:15:00
18:20:00
per 10 seconds
DELETE
Aborted Connections Rate
0
18:10:00
18:15:00
18:20:00
per 10 seconds
Aborted Connections
Failed Attempts to Connect
Thread Activity
0
18:10:00
18:15:00
18:20:00
per 10 seconds
Avg Threads Running
Peak Threads Running

Analyses

Les principaux indicateurs présentés sont :

- **Taux de requêtes :** Il indique le nombre de requêtes (SELECT, INSERT, UPDATE, DELETE) exécutées par seconde.

- **Connexions :** Cet indicateur montre le nombre de connexions actives à la base de données ainsi que le taux de connexion et d'abandon de connexion.

- **Tentatives de connexion échouées :** Ce graphique met en évidence les problèmes d'authentification ou d'accès à la base de données.

Grâces à cette supervision nous pouvons observer des :

- **Peaks d'activité :** Des pics d'activité peuvent indiquer des périodes de forte charge sur votre application, des traitements par lots ou des requêtes lentes.

- **Taux d'erreurs :** Un taux d'erreurs élevé peut signaler des problèmes de configuration, des requêtes mal formulées ou des problèmes de connectivité.

- **Connexions abandonnées :** Un nombre important de connexions abandonnées peut indiquer des problèmes de performance, des fuites de mémoire ou des problèmes de configuration du pool de connexions.

- **Tentatives de connexion échouées :** Des tentatives de connexion échouées peuvent être le signe d'attaques par force brute ou de problèmes d'authentification.

Conclusion

Ce tableau de bord est un bon début pour surveiller notre base de données MySQL. En ajoutant des métriques supplémentaires et en personnalisant les alertes, nous pourrons mieux comprendre le comportement de notre base de données et détecter rapidement les problèmes potentiels.