

Livrable

Projet de réseau LAB
Intégration d'un outil de
supervision.

Raphaël FRELAUT

DSI EAU17

Stage du 27/05/2024 au 28/06/2024

Sommaire :

Introduction :	2
Benchmarks :	2
Dossier technique :	3
1. Installation :	3
a. Installation de grafana :	3
b. Installation de prometheus :	4
c. Installation de prometheus-node-exporter :	4
d. Lancement des services.	4
2. Ajout d'une machine :	4
Transfert du dossier /var sur un nouveau disque :	7
Sources :	8

Introduction :

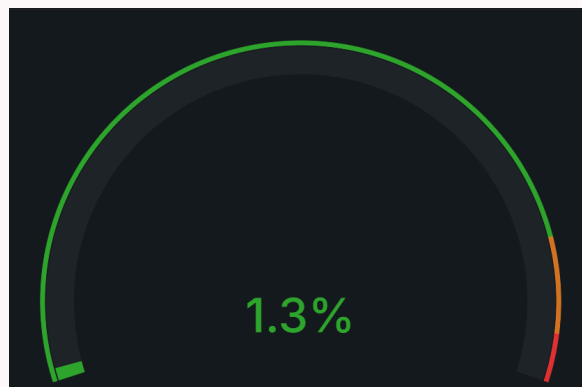
L'outil de supervision choisi est Grafana. Cet outil ne servant qu'à l'affichage des données sur son interface, il est couplé à d'autres outils pour fonctionner.

- **Grafana** : interface utilisateurs ou son affiché les informations.
- **prometheus** : récupère les informations
- **victoria-metrics** : stock ces données
- **prometheus-node-exporter** : envoie les données depuis la machine supervisée.

Ces outils sont essentiels pour le bon fonctionnement de la supervision.

Benchmarks :

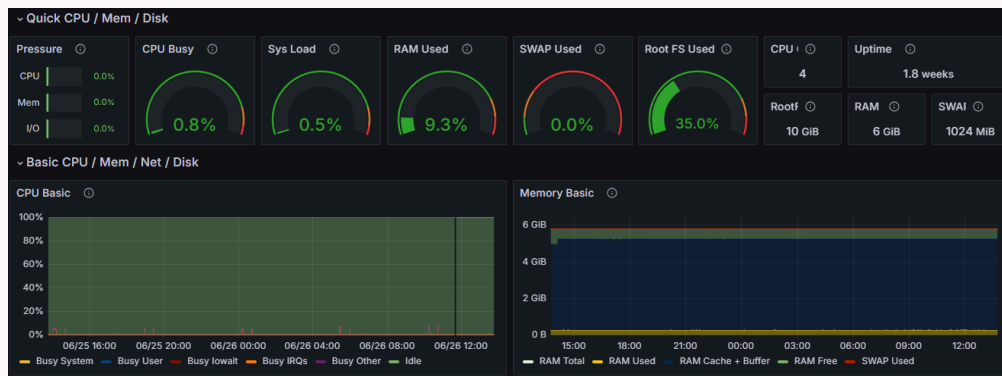
Une fois installé, l'outil de supervision offre un très large choix d'informations détaillées , de l'utilisation du CPU au nombre d'erreurs TCP en passant par le débit d'écriture de chaque disque.



Chaque information est affichée sous forme de panel.

Un panel peut être appelé en une ligne de code, pour avoir l'utilisation du système par exemple :

```
scalar(node_load1{instance="$node",job="$job"}) * 100 /  
count(count(node_cpu_seconds_total{instance="$node",job="$job"}) by (cpu))
```



Ces panels sont regroupés dans des dashboards. Ils peuvent être déplacés et librement classés. Par défaut, les informations sont actualisées toutes les 15sec au niveau de prometheus pour des soucis de stockage. Il est possible de changer la fréquence d'actualisation coté grafana allant de 5 sec à 1 jour. La plage d'information peut aussi être personnalisée dans le coin supérieur droit.

Dossier technique :

1. Installation :

a. Installation de grafana :

```
apt-get install -y apt-transport-https software-properties-common wget
```

```
mkdir -p /etc/apt/keyrings/  
wget -q -O - https://apt.grafana.com/gpg.key | gpg --dearmor | tee  
/etc/apt/keyrings/grafana.gpg > /dev/null
```

```
echo "deb [signed-by=/etc/apt/keyrings/grafana.gpg] https://apt.grafana.com stable  
main" | tee -a /etc/apt/sources.list.d/grafana.list
```

```
apt-get install grafana
```

b. Installation de prometheus :

```
apt-get install prometheus
```

c. Installation de prometheus-node-exporter :

```
apt-get install prometheus-node-exporter
```

d. Lancement des services

```
systemctl start prometheus.service
```

```
systemctl start grafana-server.service
```

```
systemctl start prometheus-node-exporter.service
```

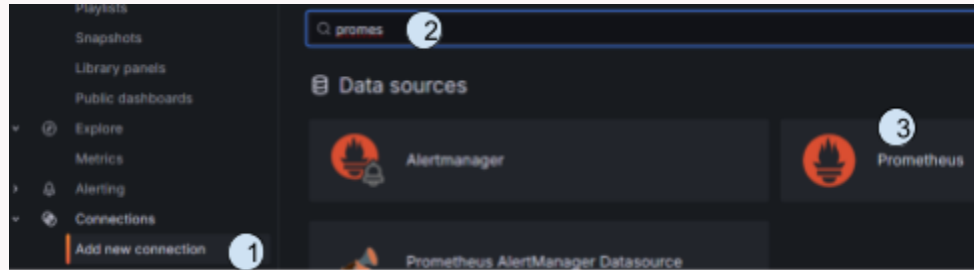
2. Ajout d'une machine :

Pour ajouter une machine :

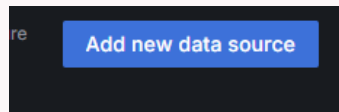
1. Bien installer prometheus-node-exporter sur la machine à superviser
 - a. Linux (voir 1. Installation)
 - b. Windows : [Github](#)
2. Ajouter la machine dans le fichier prometheus.yml (/etc/prometheus/prometheus.yml)

```
- job_name: "Serveur_Windows"
  scrape_interval: 15s
  static_configs:
    - targets: ["IP.Cible:9182"]
```

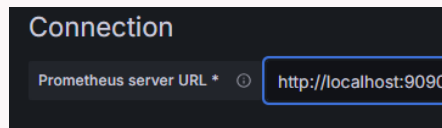
- **Job_name** : correspond à la catégorie dans laquelle le serveur sera supervisé sur grafana. Il est conseillé de mettre les serveurs linux et windows dans des catégories différentes.
 - **Scrape_interval** : temps entre chaque récupération d'informations (envoyé par node-exporter)
 - **static_configs** : sous paramètre ou seront rangés les configurations propre aux serveurs du "job"
 - **targets** : ip du serveur supervisé avec le port par défaut de node-exporter sur windows.
 - pour ajouter plusieurs serveurs ; mettre une virgule (["IP1" , "IP2"])
 - si DNS, possibilité de mettre le nom de la machine.
3. relier grafana et prometheus



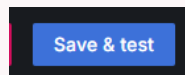
a.



b.



c.

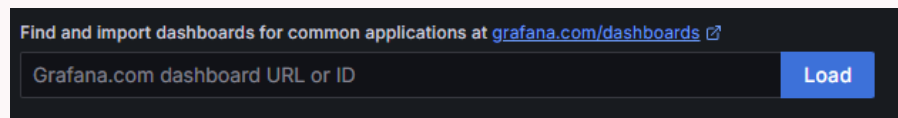


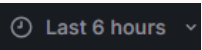
d.

4. Importer un dashboard

- Pour que toutes les informations soient affichées au même endroit, Grafana utilise un système de dashboard. Un dashboard peut être fait sous forme de .JSON mais peut aussi être importé à partir des créations des autres utilisateurs [ici](#).
- deux dashboard qui ont fonctionnés :
 - windows : 14451
 - Linux : 1860
- Pour les importer :

- dans dashboard :  >> import



Et voila le dashboard est bien importé et les informations sont censées bien s'afficher sinon régler la date d'affichage des informations. 

5. Vérification du bon fonctionnement.

- vérification que les données sont bien envoyés

ipduserveur:9182/metrics

```
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 0
go_gc_duration_seconds{quantile="0.25"} 0
go_gc_duration_seconds{quantile="0.5"} 0
go_gc_duration_seconds{quantile="0.75"} 0.0005155
go_gc_duration_seconds{quantile="1"} 0.0130316
go_gc_duration_seconds_sum 0.5173469
go_gc_duration_seconds_count 2779
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 12
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info(version="go1.21.5") 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 5.011168e+06
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed.
# TYPE go_memstats_alloc_bytes_total counter
go_memstats_alloc_bytes_total 8.3703568e+09
# HELP go_memstats_buck_hash_sys_bytes Number of bytes used by the profiling bucket hash table.
# TYPE go_memstats_buck_hash_sys_bytes gauge
go_memstats_buck_hash_sys_bytes 1.557852e+06
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 1.46702025e+08
# HELP go_memstats_gc_sys_bytes Number of bytes used for garbage collection system metadata.
# TYPE go_memstats_gc_sys_bytes gauge
go_memstats_gc_sys_bytes 3.106992e+06
# HELP go_memstats_heap_alloc_bytes Number of heap bytes allocated and still in use.
# TYPE go_memstats_heap_alloc_bytes gauge
go_memstats_heap_alloc_bytes 5.011168e+06
# HELP go_memstats_heap_idle_bytes Number of heap bytes waiting to be used.
# TYPE go_memstats_heap_idle_bytes gauge
go_memstats_heap_idle_bytes 8.667136e+06
# HELP go_memstats_heap_inuse_bytes Number of heap bytes that are in use.
# TYPE go_memstats_heap_inuse_bytes gauge
go_memstats_heap_inuse_bytes 7.12704e+06
# HELP go_memstats_heap_objects Number of allocated objects.
# TYPE go_memstats_heap_objects gauge
```

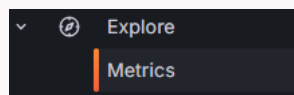
- Vérification que prometheus acquiert et traite bien les informations :

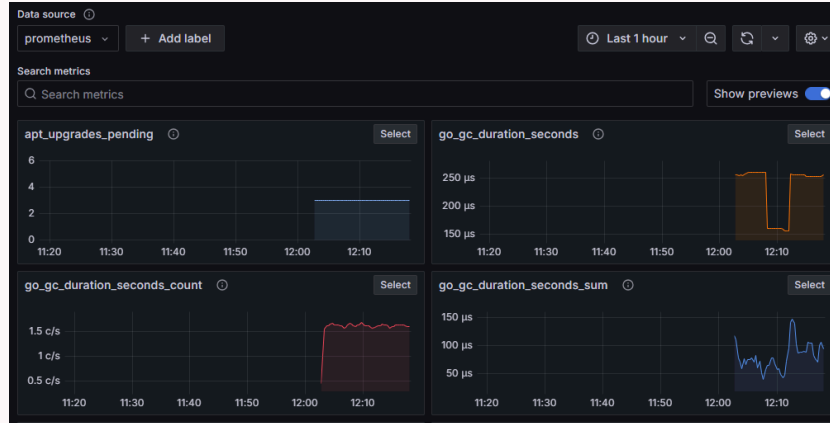
ipduserveur:9090/classic/targets

Serveur_Windows (1/1 up) [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://[redacted]metrics	UP	instance="[redacted]9182" job="Serveur_Windows"	3.539s ago	684.6ms	

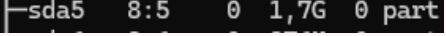
- Vérification que grafana reçoit les informations :





- pour réinitialiser le mdp admin de grafana, avec sqlite3
/var/logs/grafana/grafana.db entrer cette commande :
 - update user set password =
 '59acf18b94d7eb0694c61e60ce44c110c7a683ac6a8f09580d626f90f4a24200
 0746579358d77dd9e570e83fa24faa88a8a6', salt = 'F3FAxVm33R' where login
 = 'admin';

Transfert du dossier /var sur un nouveau disque :

Problématique : D'origine, le dossier /var est monté sur sda5  avec très peu de place (2Go). Les metrics récolté par Prometheus sont stockés dessus, le stockage proposé est donc assez faible et va donc vite poser problème.

Solution retenue : Nous allons donc ajouter un disque et monter /var dessus. 10Go est suffisant étant donné que la data récolté est automatiquement supprimée au bout de 7 jours.

Manipulation :

1. supprimer les données sur le nouveau disque :

```
$ sudo mkfs.ext4 /dev/sdX1
```

2. Créer un point de montage et monter le disque

```
$ sudo mkdir /mnt/newvar  
$ sudo mount /dev/sdX1 /mnt/newvar
```

3. Transférer les données /var vers la nouvelle partition (installer rsync)

```
$ sudo rsync -avxHAX --progress /var/* /mnt/newvar/
```

4. Récupérer l'UUID de la partition

```
$ sudo blkid /dev/sdX1
```


5. Mettez à jour le fichier fstab pour monter la nouvelle partition en tant que /var au démarrage en mettant en commentaire le disque actuellement utilisé.

```
$ echo 'UUID="UUID HERE" /var ext4 defaults 0 2' | sudo tee -a /etc/fstab
```

6. Démonter le dossier temporaire et redémarrer

```
$ sudo umount /mnt/newvar  
$ sudo reboot
```

7. Vérifier

```
$ df -h /var/
```

Sources :

Installation grafana : [How to install&configure Grafana and Prometheus on Debian 12 – D4D Blog](#)

Documentations :

- [grafana](#)
- [prometheus](#)
- [node-exporter](#)

Dashboards Grafana : [Grafana dashboards | Grafana Labs](#)

Montage /var vers un nouveau disque : [How to Move /var to Another Directory on a Full Partition - Linux Tutorials - Learn Linux Configuration](#)