

# PROMETHEUS GRAFANA

La « data visualisation » open source



## SOMMAIRE

1. PROMETHEUS ET GRAFANA : C'EST QUOI ?
2. INSTALLATION DE PROMETHEUS SUR DEBIAN 11.6
3. INSTALLATION DE GRAFANA OSS SUR DEBIAN 11.6
4. PARAMETRAGE LIAISON GRAFANA/PROMETHEUS
5. INSTALLATION DE NODE\_EXPORTER DANS PROMETHEUS
6. REALISATION D'UN DASHBOARD DANS GRAFANA
7. MONITORING GRAFANA DEPUIS UN SMARTPHONE

© [tutos-info.fr](https://tutos-info.fr) - 01/2023



DIFFICULTE



UTILISATION COMMERCIALE INTERDITE

# 1 – PROMETHEUS ET GRAFANA : LE MONITORING ACTIF

**Prometheus** est un logiciel libre de surveillance informatique et générateur d'alertes. Il enregistre des métriques en temps réel dans une base de données (avec une capacité d'acquisition élevée) en se basant sur le contenu de point d'entrée exposé à l'aide du protocole HTTP. Ces métriques peuvent ensuite être interrogées à l'aide d'un langage de requête simple : le **PromQL** et peuvent également servir à générer des alertes. Le projet est écrit en **Go** et est disponible sous licence Apache 2.

Prometheus n'est pas conçu pour faire de la restitution d'informations sous la forme de tableau de bord bien qu'il dispose d'une solution pour le faire. Une bonne pratique est de faire appel à un outil comme **Grafana**.

**Grafana** est un outil open source de monitoring informatique orienté data visualisation. Il est conçu pour générer des dashboards sur la base de métriques et données temporelles.

## Grafana, qu'est-ce que c'est ?

Portée par l'américain Grafana Labs, **Grafana** est une plateforme open source, orientée observabilité, taillée pour la surveillance, l'analyse et la visualisation des métriques IT. Elle est livrée avec un serveur web (écrit en Go) permettant d'y accéder via une API HTTP. Sous licence Apache 2.02, Grafana génère ses graphiques et tableaux de bord à partir de bases de données de séries temporelles.

Véritable éditeur de dashboards informatiques, Grafana permet également de les partager sous forme de snapshot (ou instantanés) avec d'autres utilisateurs. L'outil intègre par ailleurs un système de gestion des droits d'accès et protège les tableaux de bord des modifications accidentelle.



Pour réaliser ce tutoriel, nous utiliserons une machine Debian 11.6 (version actuelle) et nous présenterons la méthode d'installation classique (sans docker). Un autre tutoriel sera consacré à l'installation via Docker Compose.

Afin de tester le bon fonctionnement de votre monitoring, il pourrait être judicieux d'avoir d'autres machines afin d'obtenir des métriques les plus diversifiées.

## 2 – INSTALLATION DE PROMETHEUS SUR UNE MACHINE DEBIAN 11.6

Pour la réalisation de ce tutoriel, assurez-vous d'avoir une **machine Debian 11.6** fonctionnelle avec un accès à Internet. Ici, nous travaillerons avec l'utilisateur « root » car nous sommes dans un environnement laboratoire mais il est conseillé de travailler avec un utilisateur ayant les droits « sudo » lorsque vous travaillez sur une machine en production.

### Note :

Pensez, si vous travaillez avec un utilisateur faisant partie du groupe « sudoers », à saisir la commande « sudo » devant les commandes présentées ci-après.


**Un serveur LAMP doit être installé sur votre machine Debian.**

A ce jour, la version, la version de Prometheus utilisée est la 2.41.0 (décembre 2022). Vous trouverez le lien de téléchargement directement sur le site officiel de Prometheus : <https://prometheus.io/download/>

Lien de téléchargement de Prometheus (pour une architecture Linux Debian amd64) au format « tar.gz » : <https://github.com/prometheus/prometheus/releases/download/v2.41.0/prometheus-2.41.0.linux-amd64.tar.gz>

Operating system **linux** Architecture **amd64**

**prometheus**

The Prometheus monitoring system and time series database.  prometheus/prometheus

**2.41.0 / 2022-12-20** Release notes

File name	OS	Arch	Size	SHA256 Checksum
<b>prometheus-2.41.0.linux-amd64.tar.gz</b>	linux	amd64	86.15 MiB	9c1486f5f3e03d56433112594740ca84c6e59db061d096eeb3ea971ba25441c9

- Téléchargez les dernières sources de Prometheus (ici nous les téléchargeons depuis le dossier « home » du root » mais vous pouvez procéder avec un utilisateur « sudoer ») à l'aide de la commande « wget » :

```
wget https://github.com/prometheus/prometheus/releases/download/v2.41.0/prometheus-2.41.0.linux-amd64.tar.gz
```

Patientez le temps du téléchargement (fichier de 86 Mo environ) :

```
prometheus-2.41.0.linux-amd64.t 100%[=====>] 86,15M 30,9MB/s ds 2,8s
2022-12-30 17:14:02 (30,9 MB/s) - « prometheus-2.41.0.linux-amd64.tar.gz » sauvegardé [90331753/90331753]
```

- Décompressez le fichier téléchargé au format « tar.gz » avec la commande « tar xzf » :

```
tar xzf prometheus-2.41.0.linux-amd64.tar.gz
```

Si vous saisissez la commande « ls » vous constaterez qu'un dossier décompressé est maintenant présent :

```
root@debian:~# ls
prometheus-2.41.0.linux-amd64 prometheus-2.41.0.linux-amd64.tar.gz
```

- Déplacez le dossier décompressé dans « /usr/share/prometheus » à l'aide de la commande « mv » :

```
mv prometheus-2.41.0.linux-amd64/ /usr/share/prometheus
```

- Créez un utilisateur « prometheus » à l'aide de la commande « useradd » :

```
useradd prometheus
```

- Spécifiez les dossiers de stockage des données (ici nous avons créé un dossier « prometheus » et un sous-dossier « data » dans « /var/lib ») :

```
mkdir -p /var/lib/prometheus/data
```

- Indiquez que l'utilisateur « prometheus » est le propriétaire du chemin complet « /var/lib/prometheus/data » (emplacement de stockage des données) à l'aide de la commande « chown » :

```
chown prometheus:prometheus /var/lib/prometheus/data
```

- Indiquez que l'utilisateur « prometheus » est le propriétaire, de manière récursive « -R » du chemin complet « /usr/share/prometheus » (sources du logiciel), à l'aide de la commande « chown » :

```
chown -R prometheus:prometheus /usr/share/prometheus
```

Logiquement, Prometheus est installé ! Vérifiez son bon fonctionnement avec la configuration par défaut. Pour cela, déplacez-vous, avec la commande « cd », dans « /usr/share/prometheus » et saisissez « ls -lat ». Vous constaterez qu'un fichier « prometheus.yml » est présent :

```
root@debian: /usr/share/prometheus# ls -lat
total 223032
drwxr-xr-x 85 root      root      4096 30 déc.  17:19 ..
drwxr-xr-x  4 prometheus prometheus 4096 20 déc.  12:09 .
drwxr-xr-x  2 prometheus prometheus 4096 20 déc.  12:05 console_libraries
drwxr-xr-x  2 prometheus prometheus 4096 20 déc.  12:05 consoles
-rw-r--r--  1 prometheus prometheus 11357 20 déc.  12:05 LICENSE
-rw-r--r--  1 prometheus prometheus  3773 20 déc.  12:05 NOTICE
-rw-r--r--  1 prometheus prometheus   934 20 déc.  12:05 prometheus.yml
-rwxr-xr-x  1 prometheus prometheus 110033446 20 déc.  11:46 promtool
```

- Lancez un test en saisissant :

```
/usr/share/prometheus/prometheus --config.file=/usr/share/prometheus/prometheus.yml
```

Vous devriez voir s'afficher des résultats sous la forme ci-dessous ;

```
ts=2022-12-30T16:37:45.971Z caller=head.go:683 level=info component=tsdb msg="WAL segment loaded" segment=0 maxSegment=1
ts=2022-12-30T16:37:45.971Z caller=head.go:683 level=info component=tsdb msg="WAL segment loaded" segment=1 maxSegment=1
ts=2022-12-30T16:37:45.971Z caller=head.go:720 level=info component=tsdb msg="WAL replay completed" checkpoint_replay_duration=
77.648µs wal_replay_duration=950.933µs wbl_replay_duration=137ns total_replay_duration=1.151721ms
ts=2022-12-30T16:37:45.973Z caller=main.go:1014 level=info fs_type=EXT4_SUPER_MAGIC
ts=2022-12-30T16:37:45.973Z caller=main.go:1017 level=info msg="TSDB started"
ts=2022-12-30T16:37:45.973Z caller=main.go:1197 level=info msg="Loading configuration file" filename=/usr/share/prometheus/prom
etheus.yml
ts=2022-12-30T16:37:45.973Z caller=main.go:1234 level=info msg="Completed loading of configuration file" filename=/usr/share/pr
ometheus/prometheus.yml totalDuration=456.641µs db_storage=833ns remote_storage=1.552µs web_handler=244ns query_engine=643ns sc
rape=156.006µs scrape_sd=16.739µs notify=22.135µs notify_sd=9.958µs rules=1.049µs tracing=5.545µs
ts=2022-12-30T16:37:45.973Z caller=main.go:978 level=info msg="Server is ready to receive web requests."
ts=2022-12-30T16:37:45.973Z caller=manager.go:953 level=info component="rule manager" msg="Starting rule manager..."
```

Pour stopper l'affichage des données, saisissez « CTRL » + « C » : Prometheus est fonctionnel !

On va maintenant créer un script « **SystemD** » pour que Prometheus soit démarré en tant que service avec la machine Debian qui héberge Prometheus.

A l'aide de l'éditeur nano, nous allons créer le fichier « prometheus.service » en saisissant :

```
nano /etc/systemd/system/prometheus.service
```

Complétez le fichier en saisissant les indications suivantes :

```
GNU nano 5.4 /etc/systemd/system/prometheus.service *
[Unit]
Description=Prometheus Server
Documentation=https://prometheus.io/docs/introduction/overview/
After=network-online.target

[Service]
User=prometheus
Restart=on-failure
WorkingDirectory=/usr/share/prometheus
ExecStart=/usr/share/prometheus/prometheus --config.file=/usr/share/prometheus/prometheus.yml

[Install]
WantedBy=multi-user.target
```

On déclare le service Prometheus en reprenant les paramètres d'installation (chemin d'accès, utilisateur).

- Relancez le « daemon » Debian afin que le nouveau service Prometheus soit pris en compte

## systemctl daemon-reload

- Activez Prometheus en tant que service avec la commande « systemctl enable prometheus » :

```
root@debian:/usr/share/prometheus# systemctl enable prometheus
Created symlink /etc/systemd/system/multi-user.target.wants/prometheus.service → /etc/systemd/system/prometheus.service.
```

- Faites démarrer le service Prometheus à l'aide de la commande « systemctl start prometheus » :

## systemctl start prometheus

- Vérifiez que Prometheus fonctionne à l'aide de la commande « systemctl status prometheus » :

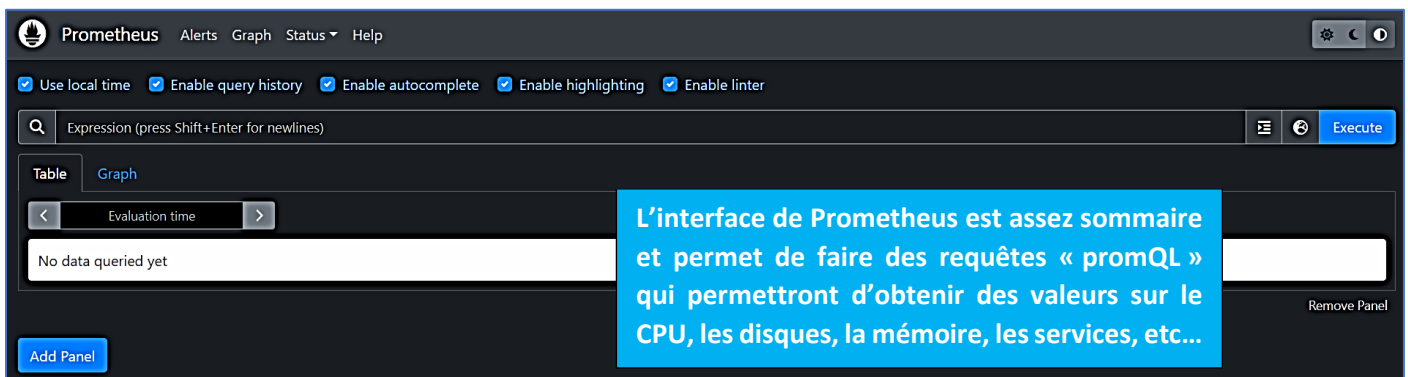
```
● prometheus.service - Prometheus Server
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2022-12-31 11:21:35 CET; 2min 50s ago
     Docs: https://prometheus.io/docs/introduction/overview/
    Main PID: 1730 (prometheus)
      Tasks: 7 (limit: 2337)
     Memory: 32.5M
        CPU: 152ms
    CGroup: /system.slice/prometheus.service
            └─1730 /usr/share/prometheus/prometheus --config.file=/usr/share/prometheus/prometheus.yml

déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.433Z caller=head.go:683 level=info compo
déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.433Z caller=head.go:720 level=info compo
déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.434Z caller=main.go:1014 level=info fs_t
déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.434Z caller=main.go:1017 level=info msg=
déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.434Z caller=main.go:1197 level=info msg=
déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.435Z caller=main.go:1234 level=info msg=
déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.435Z caller=main.go:978 level=info msg="
déc. 31 11:21:35 debian prometheus[1730]: ts=2022-12-31T10:21:35.436Z caller=manager.go:953 level=info co
```

Pour accéder à l'interface web de Prometheus, ouvrez un navigateur et saisissez l'adresse suivante :

[http://votre\\_domaine:9090](http://votre_domaine:9090) (pensez à bien indiquer le port « 9090 » à la fin de l'adresse et **ouvrez le port 9090 dans votre pare-feu ou créez une règle dans votre routeur le cas échéant**).

## INTERFACE D'ACCUEIL DE PROMETHEUS



## 3 – INSTALLATION DE GRAFANA (OSS) SUR UNE MACHINE DEBIAN 11.6

Prometheus étant installé (voir point précédent), il est conseillé d'installer Grafana qui vous permettra d'obtenir des tableaux de bord (« dashboards ») complets. Grafana fournira des données sous forme de graphiques notamment et peut être installé sur un smartphone pour un monitoring en temps réel !

Il est possible d'installer Grafana soit à partir des sources (voir site web officiel), soit à partir du référentiel « apt ». Nous vous proposons d'installer Grafana depuis le référentiel APT dans les pages suivantes :

- Commencez par installer l'utilitaire « gnupg2 » et divers modules :

```
apt install gnupg2 curl software-properties-common dirmngr apt-transport-https lsb-release ca-certificates -y
```

- Ajoutez la clé GPG de Grafana au système :

```
curl https://packages.grafana.com/gpg.key | apt-key add -
```

- Ajouter le « repository » de Grafana à votre système :

```
add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
```

- Mettez à jour les dépôts de votre système :

```
apt update
```

- Lancez l'installation de Grafana :

```
apt -y install grafana
```

Une fois l'installation terminée, un message s'affiche et vous demande d'exécuter diverses commandes :

1. Relancer le « daemon » avec la commande « **systemctl daemon-reload** »
  2. Activer le serveur Grafana avec la commande « **systemctl enable grafana-server** »
  3. Démarrer le serveur Grafana avec la commande « **systemctl start grafana-server** »
- Exécutez ces commandes dans l'ordre ! Si vous travaillez en « root », enlevez « sudo » :

```
### NOT starting on installation, please execute the following statements to configure grafana to start automatically using systemd
sudo /bin/systemctl daemon-reload
sudo /bin/systemctl enable grafana-server
### You can start grafana-server by executing
sudo /bin/systemctl start grafana-server
```

Lancez l'activation du serveur Grafana avec la commande « **systemctl enable grafana-server** » :

```
root@debian:/etc/systemd/system/multi-user.target.wants# systemctl enable grafana-server
Synchronizing state of grafana-server.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service → /lib/systemd/system/grafana-server.service.
```

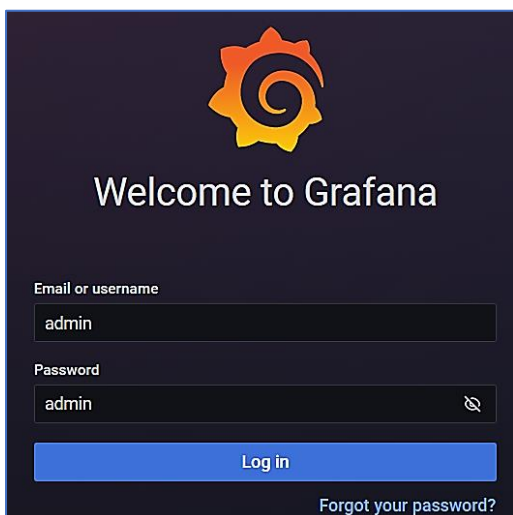
Faites démarrer le serveur Grafana avec la commande « **systemctl start grafana-server** » :

```
systemctl start grafana-server
```

Contrôler le statut Grafana avec la commande « **systemctl status grafana-server** » :

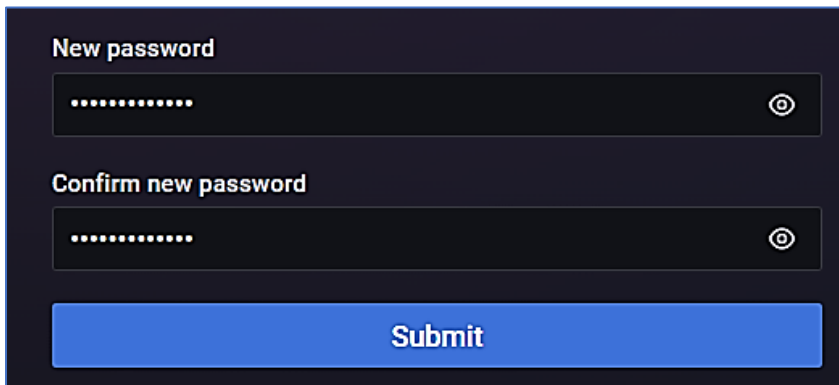
```
root@debian:/etc/systemd/system/multi-user.target.wants# systemctl start grafana-server
root@debian:/etc/systemd/system/multi-user.target.wants# systemctl status grafana-server
● grafana-server.service - Grafana instance
   Loaded: loaded (/lib/systemd/system/grafana-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2022-12-31 14:35:38 CET; 1min 11s ago
     Docs: http://docs.grafana.org
   Main PID: 2747 (grafana-server)
    Tasks: 15 (limit: 2337)
   Memory: 60.9M
      CPU: 1.397s
   CGroup: /system.slice/grafana-server.service
           └─2747 /usr/sbin/grafana-server --config=/etc/grafana/grafana.ini --pidfile=/run/graf
déc. 31 14:36:16 debian grafana-server[2747]: logger=infra.usagestats.collector t=2022-12-31T14:36:16.145954133+01:00
déc. 31 14:36:16 debian grafana-server[2747]: logger=server t=2022-12-31T14:36:16.145954133+01:00
déc. 31 14:36:16 debian grafana-server[2747]: logger=provisioning.alerting t=2022-12-31T14:36:16.145954133+01:00
déc. 31 14:36:16 debian grafana-server[2747]: logger=provisioning.alerting t=2022-12-31T14:36:16.145954133+01:00
```

Votre installation de Grafana est maintenant prête. Pour accéder à l'écran d'administration, lancez votre navigateur et saisissez une adresse de type [http://votre\\_domaine:3000](http://votre_domaine:3000) (pensez à bien ajouter le port « 3000 » à la fin de l'adresse et à **autoriser l'accès dans votre pare-feu ou créer une règle dans votre routeur**). Au premier lancement, une fenêtre d'identification s'ouvre et vous demande de saisir l'identifiant « admin » et le mot de passe « admin » par défaut :



Lors de la première connexion à Grafana, vous devez saisir les identifiants par défaut « admin » / « admin ».

Une seconde fenêtre vous demande de créer un nouveau mot de passe ; faites la modification du mot de passe initial et cliquez le bouton « Submit » :

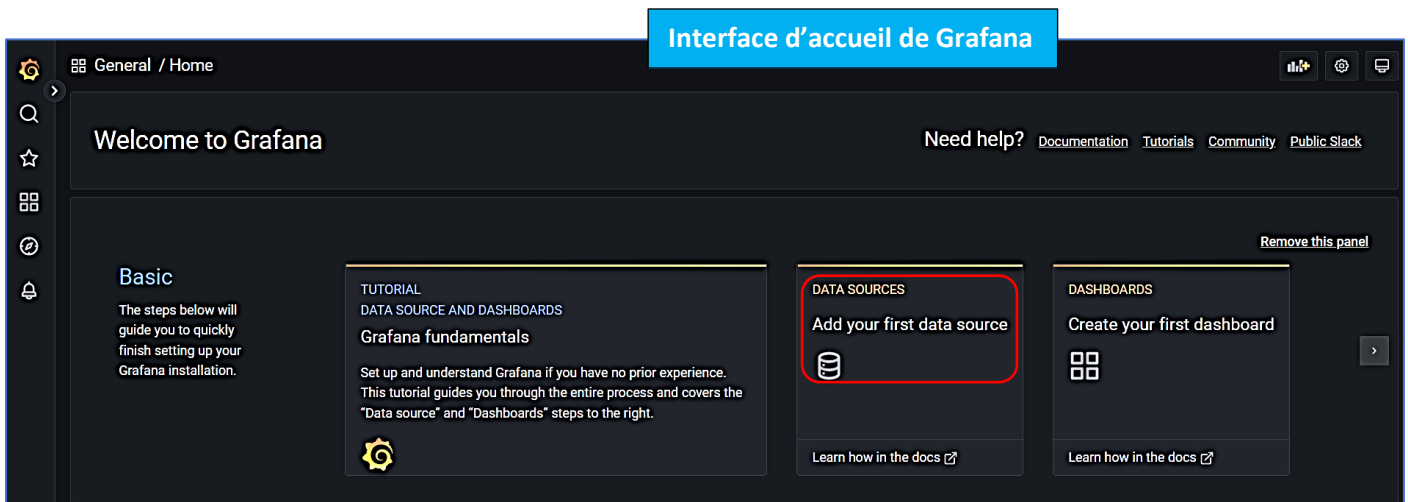


Modifiez le mot de passe par défaut en saisissant, ici, le nouveau mot de passe de l'utilisateur admin de Grafana.

Si vous travaillez en production, il est conseillé de sécuriser l'accès avec un reverse proxy.

## INTERFACE D'ACCUEIL DE GRAFANA

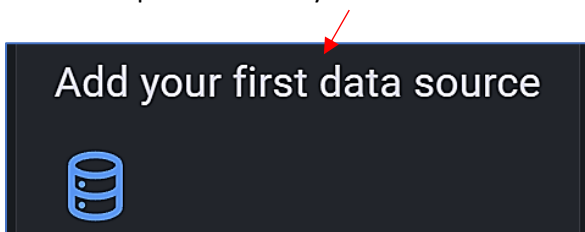
L'écran d'accueil de Grafana est maintenant affiché.



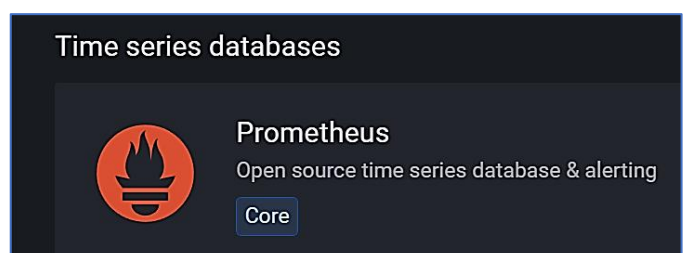
## 4 – PARAMETRAGE DE LA LIAISON ENTRE GRAFANA ET PROMETHEUS

Maintenant que Prometheus et Grafana sont installés, nous allons les connecter entre eux. En effet, Grafana doit utiliser la base de données de Prometheus pour pouvoir afficher des relevés en temps réel sur différents critères.

- Lancez Grafana depuis votre navigateur ([http://votre\\_domaine:3000](http://votre_domaine:3000)) et connectez-vous à l'interface
- Cliquez sur « Add your first data source »

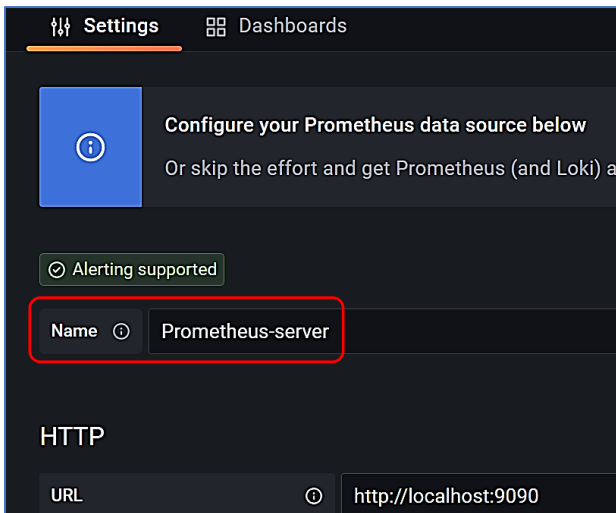


- Dans la liste des sources affichées, cliquez sur « Prometheus »

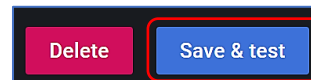




- Une fenêtre de paramétrage s'affiche ; commencez par saisir un nom de pour cette source (ici nous avons saisi « Prometheus-server par exemple ») et saisissez également l'adresse du serveur Prometheus :

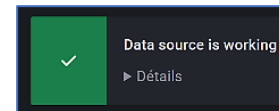


Saisissez un nom pour votre serveur Prometheus qui servira de source de données.



- Cliquez, dans le bas de la fenêtre, sur « Save and test »

Si la connexion est validée, le message « Data source is working » doit s'afficher :



La liaison entre Prometheus et Grafana est maintenant effective. Nous allons pouvoir préparer le monitoring est la gestion des tableaux de bord dans Grafana.

## 5 – CONFIGURATION DE NODE\_EXPORTER DANS PROMETHEUS

La liaison entre Prometheus et Grafana étant maintenant activée, nous pouvons préparer notre premier tableau de bord afin d'assurer le monitoring en temps réel d'une machine. Nous allons commencer le monitoring à partir de la machine Debian qui nous sert de serveur.

Node\_exporter Full est en fait un modèle (« template ») de tableau de bord très complet qui nous permettra d'assurer le suivi en temps réel d'une ou plusieurs machines du réseau. Il est également possible d'effectuer un monitoring sur des services (par exemple avec « apache exporter »).

**1<sup>ère</sup> étape** : installation de node\_exporter sur la machine serveur Debian (en console)

- Téléchargez l'archive de node\_exporter :

```
root@debian:~# curl -s https://api.github.com/repos/prometheus/node_exporter/releases/latest | grep browser_download_url | grep linux-amd64 | cut -d '"' -f 4 | wget -qi -
```

- Décompressez l'archive :

```
tar -xvf node_exporter*.tar.gz
```

- Ouvrez et copiez le dossier node\_exporter dans « /usr/local/bin » :

```
cd node_exporter*/
```

```
cp node_exporter /usr/local/bin
```

- Vérifiez que `node_exporter` s'est installé en saisissant la commande `node_exporter --version`

Logiquement, vous devriez voir s'afficher les informations sur la version installée :

```
node_exporter, version 1.5.0 (branch: HEAD, revision: 1b48970ffcf5630534fb00bb0687d73c66d1c959)
build user:   root@6e7732a7b81b
build date:   20221129-18:59:09
go version:   go1.19.3
platform:    linux/amd64
```

**2<sup>ème</sup> étape** : création d'un service « `node_exporter` » :

- Créez un nouveau service pour `node_exporter` à l'aide de l'éditeur `nano` :

```
nano /etc/systemd/system/node_exporter.service
```

- Saisissez les indications suivantes et enregistrez le fichier :

```
GNU nano 5.4
[Unit]
Description=Node Exporter
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
ExecStart=/usr/local/bin/node_exporter

[Install]
WantedBy=default.target
```

On déclare le service `node_exporter` en reprenant les paramètres d'installation (chemin d'accès, utilisateur).

- Rechargez le « daemon », démarrez le service `node_exporter` et activez-le en tant que service au démarrage :

```
systemctl daemon-reload
```

```
systemctl start node_exporter
```

```
systemctl enable node_exporter
```

Si tout se passe bien, vous devriez avoir le message ci-dessous qui stipule que le service a bien été créé :

```
Created symlink /etc/systemd/system/default.target.wants/node_exporter.service → /etc/systemd/system/node_exporter.service.
```

- Vérifiez le statut du service avec la commande « `systemctl status node_exporter` » :

```
root@debian:~/node_exporter-1.5.0.linux-amd64# systemctl status node_exporter.service
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2022-12-31 21:08:43 CET; 2min 17s ago
     Main PID: 2559 (node_exporter)
       Tasks: 4 (Limit: 2337)
      Memory: 4.7M
         CPU: 8ms
    CGroup: /system.slice/node_exporter.service
            └─2559 /usr/local/bin/node_exporter
```

**3<sup>ème</sup> étape** : ajout du service `node_exporter` dans le fichier de configuration du serveur Prometheus :

- Editez le fichier « `prometheus.yml` » avec l'éditeur `nano` :

```
nano /usr/share/prometheus/prometheus.yml
```

- Ajoutez, dans la section « `scrape_configs` », un nouveau « `job_name` » nommé « `node_exporter` » :

```
scrape_configs:
# The job name is added as a
- job_name: "prometheus"
- job_name: "node_exporter"
```

On crée, ici, un nouveau « `job_name` » dans le fichier « `prometheus.yml` ». Ce `job_name` correspond à l'instance « `node_exporter` » que l'on souhaite activer.

- Ajoutez, dans la rubrique « `static_configs` » une nouvelle « `target` ». Ici nous utiliserons le port 9100 pour `node_exporter` :

```
static_configs:
- targets: ["localhost:9090"]
- targets: ["localhost:9100"]
```

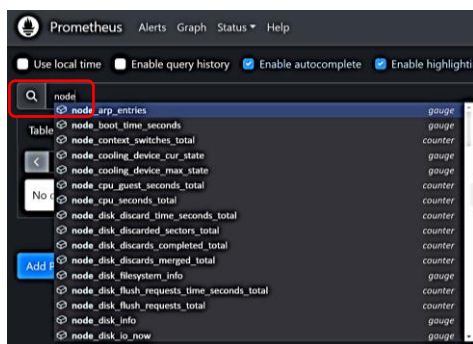
On définit, ici, le port d'écoute de `node_exporter` (en l'occurrence 9100 dans notre cas).

- Enregistrez les modifications et relancez Prometheus :

```
systemctl restart prometheus
```

Retournez dans l'interface de Prometheus et saisissez « `node` » dans la barre de recherche située à gauche ; une liste de « `metrics` » est proposée. Vous pouvez lancer un monitoring sur de nombreux composants !

Exemple de « `metrics` » proposés par `node_exporter` :



Une fois que `node_exporter` a été déclaré comme nouveau « `job_name` », on peut tester son fonctionnement en saisissant, dans la barre de recherche de Prometheus, `node...`. Un ensemble de requêtes liées à `node_exporter` s'affiche : il ne reste plus qu'à en choisir une et l'exécuter pour afficher les valeurs relatives à cette requête.

Si vous sélectionnez une mesure et que vous cliquez, sur la droite, le bouton « Exécute » vous verrez s'afficher un rapport détaillé sur la mesure demandée. Ce rapport peut être visualisé sous forme de graphe mais vous constaterez rapidement que Prometheus n'est pas conçu pour cela.

Si vous saisissez dans votre navigateur [http://votre\\_domaine:9100/metrics](http://votre_domaine:9100/metrics) vous obtenez ceci :

```
# 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"} 2.0153e-05
go_gc_duration_seconds{quantile="0.25"} 3.388e-05
go_gc_duration_seconds{quantile="0.5"} 3.9533e-05
go_gc_duration_seconds{quantile="0.75"} 3.9849e-05
go_gc_duration_seconds{quantile="1"} 8.4846e-05
go_gc_duration_seconds_sum 0.003987141
go_gc_duration_seconds_count 86
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 9
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.19.3"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 2.042952e+06
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed.
```

On voit, ici, que Prometheus n'est pas spécialement conçu pour faciliter la lecture des données recueillies. Il est donc nécessaire de le coupler avec un éditeur de graphes et de tableaux de bord comme Grafana (voir pages suivantes).

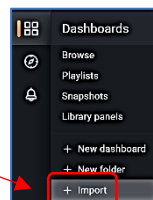
Le rôle premier de Prometheus est de collecter les « metrics » et Grafana intervient ensuite pour présenter les mesures collectées de manière visuelle, claire et ludique (les fameux « dashboards »). La création de tableaux de bord est réalisée dans Grafana qui servira d'interface graphique pour les « metrics » obtenus.

## 6 – REALISATION D'UN DASHBOARD NODE EXPORTER DANS GRAFANA

Une fois que node\_exporter a été installé en tant que service dans Prometheus, l'intérêt est de pouvoir réaliser des tableaux de bord clairs, modernes, détaillés (et ludiques) dans Grafana.

Sur Grafana Labs, de nombreux modèles de tableaux de bord sont proposés dans la rubrique « Nodes Dashboard ». Dans Grafana Labs choisissez un modèle de dashboard et repérez son « ID » : ici, nous allons importer le tableau de bord « node\_exporter Full » qui possède l'ID **1860** :

- Connectez-vous à l'interface Grafana depuis votre navigateur
- Cliquez, dans le menu de gauche, sur « Dashboard » - « Import »



- Saisissez le numéro ID **1860** qui correspond au modèle de dashboard souhaité et cliquez le bouton « Load » :

Import via grafana.com

1860

L'importation d'un tableau de bord peut se faire directement avec le numéro ID (voir sur Grafana Labs).

Load

- Indiquez la source des données (votre serveur Prometheus) et cliquez le bouton « Import » :

Name

Node Exporter Full

Folder

General

Unique identifier (UID)

The unique identifier (UID) of a dashboard can be used for uniquely identify a dashboard between multiple Grafana installs. The UID allows having consistent URLs for accessing dashboards so changing the title of a dashboard will not break any bookmarked links to that dashboard.

rYdddIPWk

prometheus

Prometheus-server

Import

Cancel

Lorsque la recherche du numéro ID a été lancée, on obtient le nom du tableau de bord (Node Explorer Full). Celui-ci sera enregistré dans le dossier « General » de Grafana.

Connectez votre source de données, en l'occurrence votre serveur Prometheus, afin que le tableau de bord importé puisse afficher les valeurs essentielles et cliquez le bouton « Import » pour lancer la procédure d'importation du template.

Le tableau de bord est importé et affiche l'ensemble des mesures envoyées par Prometheus :

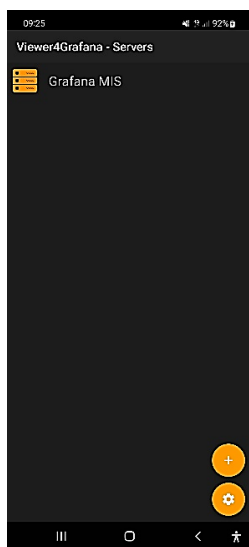


De multiples possibilités de monitoring s'offrent à vous. Il ne reste plus qu'à tester !

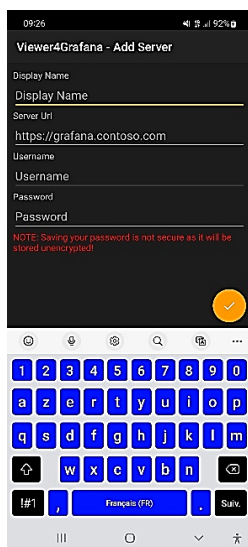
## 7 – MONITORING GRAFANA DEPUIS UN SMARTPHONE

Il est également possible de « monitorer » depuis votre smartphone ! Pour cela, connectez-vous au store de votre appareil et installez l'application « **Viewer4Grafana** ».

- Lancez l'application Viewer4Grafana
- Cliquez sur le « + » en bas à droite de l'écran
- Saisissez les identifiants de connexion à votre serveur et validez
- Appuyez sur le nom de votre session Grafana
- Appuyez sur « Node Exporter Full » : votre tableau de bord s'affiche en temps réel !



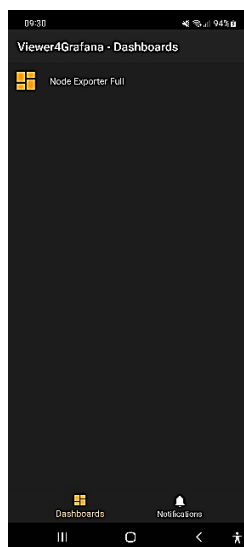
1



2



3



4



5

Fichier de configuration de Prometheus (« prometheus.yml »)

Prometheus : définitions et configuration

+ - fichier yaml :

```
./prometheus --config.file=prometheus.yml
```

+ - configuration :

- + - global : configurations qui s'appliquent en général
  - + - scrape\_interval (ou job) : intervalle de récupération des données
  - + - evaluation\_interval (ou job) : réévaluation des règles (alertes)
  - + - scrape\_timeout (ou job) : timeout lors du scraping
- + - rule\_files: configuration des alertes
- + - scrape\_configs : configuration particulière du scraping
  - + - job\_name : nom de bloc
  - + - metrics\_path: route de scraping (/metrics)
  - + - static\_config:
    - + - labels: définition de labels (important pour grafana / standardiser)
    - + - targets : url/ip:ports

A propos des « job\_name » et instance

Prometheus : définitions et configuration

+ - Définitions : jobs et instances

- + - job\_name = job
- + - instance = target (ip:port)

```
up{job="<job-name>", instance="<instance-id>"}
up{instance="192.168.62.3:9100", job="node_exporter", service="myapp"}
```

+ - Définitions : metrics

- + - ex : node\_cpu\_seconds\_total{cpu="0", mode="iowait"} 0.3
- + - <prefix\_app>\_<nom>\_<unit>\_<calcul>{<label>="<valeur>", ...} <valeur\_metrics>

Source : chaîne Youtube XAVKI

**POUR ALLER PLUS LOIN**

Une playlist exhaustive de tout ce qu'il faut savoir sur Prometheus/Grafana sur l'excellente chaîne Youtube de Xavki

Lien de la chaîne :

[https://www.youtube.com/playlist?list=PLn6POgplwWo3\\_2pj5Jflqwla62P5OI8n](https://www.youtube.com/playlist?list=PLn6POgplwWo3_2pj5Jflqwla62P5OI8n)