

# Haute disponibilité de la base de métrologie (Graphite)

## Sommaire

- [Introduction](#)
- [Architecture mise en place](#)
- [Procédure de configuration](#)
  - [Installation de Shinken](#)
  - [Mise en place du démon carbon-relay](#)
  - [Autorisation des connexions à Graphite](#)
    - [RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9](#)
    - [Debian 13](#)
  - [Modification de la configuration Shinken pour l'utilisation du cluster Graphite](#)
    - [Écriture des métriques](#)
    - [Lecture des métriques](#)
  - [Redémarrage de Graphite et Shinken](#)
- [Vérification du firewall \( est-ce que l'accès à carbon-cache / carbon-relay est disponible \)](#)
  - [Liste des ports utilisés par Graphite](#)
  - [Si firewall est présent \( firewall activé par défaut sur RHEL / Alma / Rocky \)](#)
    - [Vérifier que les ports sont ouverts](#)
    - [Sur les serveurs carbon-relay](#)
    - [Sur les serveurs carbon-cache](#)
- [Passer le cluster Graphite en HTTPS](#)
  - [RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9](#)
  - [Debian 13](#)
- [Comportement de Shinken avec un cluster Graphite](#)
- [Outils externes \( Grafana \) et relation UUID nom](#)

## Introduction

La base de métrologie de Shinken peut être fortement sollicitée sur des architectures supervisant un grand nombre d'éléments. Sa disponibilité est également importante puisqu'elle peut être source de données pour des outils externes à Shinken ( *Grafana par exemple* ) pouvant être mis à disposition des utilisateurs.

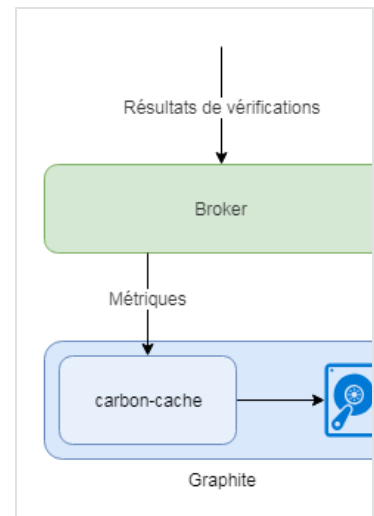
L'objectif de cette page de documentation est de détailler la procédure pour la mise en place d'un cluster Graphite. Ce qui permettra d'augmenter la disponibilité des données et d'éviter la perte de données en cas d'incident sur la machine stockant les métriques.

## Architecture mise en place

Dans une installation Shinken classique, les métriques émises par le Broker pour être stockées dans une base de données Graphite.

Chaque résultat de vérification contenant des données de métrologie est analysé par le Broker. Ces métriques sont envoyées à Graphite par le Broker grâce au module Graphite-Perfdata.

Du côté de Graphite, le nom du démon responsable de l'enregistrement des métriques sur le disque est "carbon-cache".



Pour transformer cette architecture dans le but de la rendre hautement disponible, les données seront répliquées lors de leur écriture.

Un démon "carbon-relay" distribue l'enregistrement des métriques vers plusieurs démons "carbon-cache".

Ces derniers enregistrent chacun les métriques sur une machine différente.

L'architecture obtenue à la fin de la procédure décrite dans cette documentation correspond à celle sur le schéma ci-contre.

Le cluster Graphite est constitué d'au moins 2 machines distinctes ( *idéalement 3 pour répartir les performances* ).

Chaque démon "*carbon-cache*" est sur une machine distincte, afin d'y stocker les métriques.

Le démon "*carbon-relay*" reçoit l'ensemble des métriques envoyées par le Broker. Il se charge ensuite de faire suivre les ordres d'écritures des métriques aux différents démons "*carbon-cache*". Il peut être sur une machine à part, ou sur l'une des deux machines hébergeant un démon "*carbon-cache*".



Le système mis en place ici est de la réplication, et non de la répartition.

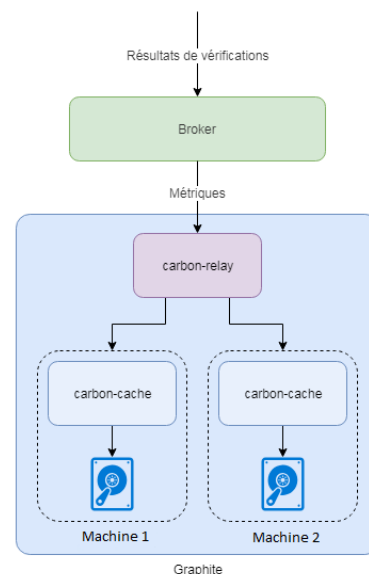
Les démons "*carbon-cache*" stockent chacun l'ensemble des métriques envoyées par le Broker. Il faut donc que chaque machine qui héberge un démon "*carbon-cache*" soit capable d'enregistrer l'intégralité des métriques gérées par le Broker.



Pour une version de Shinken antérieure à la 02.08.02-RC015 :

- Il existe des incompatibilités entre un Graphite installé sur une CentOS 7 ou RHEL 7 et un Graphite installé sur une RHEL 8, Alma 8 ou Rocky 8 ( *du fait de la version de python livrée par le système* ).
- En raison de cette incompatibilité, il n'est pas possible de monter un cluster Graphite mélangeant des nœuds sur un système RHEL 7 ou CentOS 7 avec des nœuds sur un système RHEL 8, Alma 8 ou Rocky 8.
- Il est par contre possible d'avoir un cluster :
  - avec des nœuds qui sont tous sur des systèmes RHEL 7 et CentOS 7
  - ou alors, avec des nœuds qui sont sur des systèmes RHEL 8, Alma 8 et Rocky 8.

Depuis la version 02.08.02-RC015, cette incompatibilité n'existe plus. Les nœuds d'un même cluster peuvent être installés sur n'importe quelle distribution supportée par Shinken.



## Procédure de configuration

Avant de commencer, voici un résumé des différentes étapes nécessaires pour la configuration du cluster Graphite:

- Installation de Shinken.
- Mise en place du démon "*carbon-relay*".
- Autorisation des connexions à Graphite.
- Modification de la configuration Shinken pour utiliser le cluster Graphite.
- Redémarrage de Graphite et Shinken.

## Installation de Shinken

La première étape dans la mise en place d'un cluster Graphite est bien entendu l'installation de Graphite. Comme les autres dépendances de Shinken, il faut que la version de Graphite installée soit celle fournie avec l'installateur Shinken. En effet, certaines modifications ont été faites sur Graphite pour l'intégration avec Shinken, ce qui rend incompatibles les versions de Graphite qui auraient pu être installées auparavant via d'autres installateurs.

Sur chaque machine qui compose le cluster Graphite, il faudra procéder à une installation de Shinken.

La procédure d'installation de Shinken est décrite une page de documentation dédiée ( *voir la page [Guide d'installation et de mise à jour](#)* ).

## Mise en place du démon carbon-relay

Une fois Shinken et Graphite installés, il faut procéder à la configuration de Graphite.

On commence par dire à Graphite qu'on utilise la fonctionnalité de relais via le fichier **`/opt/graphite/conf/relay-rules.conf`** ( *copier le fichier d'exemple `relay-rules.conf.example` si besoin* ).

On configure le relais en précisant les adresses des serveurs hébergeant les démons *carbon-cache* ( *pour l'écriture et la lecture des métriques* ).

### `/opt/graphite/conf/relay-rules.conf`

```
[default]
default = true
destinations = adresse_carbon_cache_1:2004,adresse_carbon_cache_2:2004
read_destinations = adresse_carbon_cache_1:80,adresse_carbon_cache_2:80
```

On configure ensuite le démon carbon-relay pour préciser la liste des carbon-cache constituant le cluster.

Pour cela éditer le fichier `/opt/graphite/conf/carbon.conf` ( copier le fichier d'exemple `carbon.conf.example` si besoin ).

Dans la section "[relay]", on modifie la variable "DESTINATIONS" dans laquelle on décrit la liste des démons **carbon-cache** qui constituent le cluster Graphite.

#### `/opt/graphite/conf/carbon.conf`

```
[relay]
...
<autres options>
...

DESTINATIONS = adresse_carbon_cache_1:2004, adresse_carbon_cache_2:2004
```



Le paramètre **destinations** du fichier `relay-rules.conf` et le paramètre **DESTINATIONS** modifié dans le fichier `carbon.conf` doivent contenir la même information.

Par défaut, l'installateur Shinken met en place un fichier de démarrage pour le démon carbon-relay sans les droits d'exécution. Pour permettre un démarrage en tant que service du système, il faut ajouter le droit d'exécution.

Sur la machine qui va héberger le démon carbon-relay, lancer la commande suivante :

#### Donner les droits d'exécution

```
chmod +x /etc/init.d/carbon-relay
```

À ce stade de la configuration, le relais est correctement configuré en ce qui concerne l'écriture des données.

Il faut maintenant permettre la lecture des métriques.

## Autorisation des connexions à Graphite

Par défaut, Graphite autorise la lecture des métriques seulement depuis l'interface locale ( `127.0.0.1` ), pour des raisons de sécurité.

Pour le fonctionnement en cluster, il faut modifier les réglages d'Apache pour autoriser la lecture depuis le réseau.

## RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9

Sur chaque carbon-relay et carbon-cache, dans le fichier `/etc/httpd/conf.d/graphite.conf` on trouve la ligne suivante:

#### `/etc/httpd/conf.d/graphite.conf`

```
<VirtualHost 127.0.0.1:80>
```

Cette directive indique à Apache qu'il faut écouter les requêtes de lecture des métriques sur l'interface locale ( `127.0.0.1` ) et le port 80 seulement.

Pour écouter ces requêtes sur toutes les interfaces réseau de la machine, on la remplacera par :

#### `/etc/httpd/conf.d/graphite.conf`

```
<VirtualHost *:80>
```

On redémarre ensuite Apache pour prendre en compte les modifications :

```
systemctl restart httpd
```

## Debian 13

Sur chaque carbon-relay et carbon-cache, dans le fichier `/etc/apache2/sites-available/graphite.conf` on trouve la ligne suivante:

#### **/etc/apache2/sites-available/graphite.conf**

```
<VirtualHost 127.0.0.1:80>
```

Cette directive indique à Apache qu'il faut écouter les requêtes de lecture des métriques sur l'interface locale ( *127.0.0.1* ) et le port 80 seulement.

Pour écouter ces requêtes sur toutes les interfaces réseau de la machine, on la remplacera par :

#### **/etc/apache2/sites-available/graphite.conf**

```
<VirtualHost *:80>
```

On redémarre ensuite Apache pour prendre en compte les modifications :

```
systemctl restart apache2
```

## **Modification de la configuration Shinken pour l'utilisation du cluster Graphite**

### **Écriture des métriques**

L'envoi des métriques à Graphite est géré par le module Graphite-Perfdata du Broker.

Pour que les métriques soient envoyées au relais plutôt qu'au démon carbon-cache, il faut modifier la configuration du module Graphite-Perfdata ( *fichier /etc/shinken/modules/graphite.cfg* ) :

#### **/etc/shinken/modules/graphite.cfg**

```
# Metrology server parameters #
# Graphite writer ( carbon ) address
# IP address or FQDN of carbon-cache or carbon-relay instance to send metrics to
#
#           Default : localhost
#
broker__module_graphite_perfdata__writer__host      adresse_carbon_relay

# Graphite writer ( carbon ) port
# tcp port of carbon-cache or carbon-relay instance to send metrics to
#
#           Default : 2003
#
broker__module_graphite_perfdata__writer__port      2013
```

On change ici

- l'adresse du serveur destinataire des métriques à écrire par celle du serveur hébergeant le démon carbon-relay,
- le port 2003 ( *port par défaut du carbon-cache* ) vers le port 2013 ( *port par défaut du carbon-relay* ).

### **Lecture des métriques**

Il faut également préciser à Shinken de passer par le relais pour la lecture des métriques.

Cette modification se fait dans les paramètres du module WebUI, responsable de l'interface de Visualisation. Dans **/etc/shinken/modules/webui.cfg**, on modifie la section suivante :

#### /etc/shinken/modules/webui.cfg

```
[...]  
  
graphite_backends      *=http://adresse_carbon_relay:80  
  
[...]
```

## Redémarrage de Graphite et Shinken

Les configurations de Graphite et de Shinken ayant été modifiées pour utiliser le cluster, il faut maintenant redémarrer les différents démons pour appliquer ces changements.

- On commence par démarrer le démon carbon-relay de Graphite :

```
service carbon-relay start
```

- On fait en sorte que ce démon soit relancé au démarrage du serveur :

```
systemctl enable carbon-relay
```

- On redémarre ensuite les démons carbon-cache sur leurs machines respectives :

```
service carbon-cache restart
```

- Enfin, on redémarre l'Arbiter pour prendre en compte les modifications de configuration de Shinken:

```
service-shinken-arbiter restart
```

## Vérification du firewall ( *est-ce que l'accès à carbon-cache / carbon-relay est disponible* )

En cas de problème de connexion aux démons carbon-cache / carbon-relay vérifier que le port est ouvert dans le firewall.

### Liste des ports utilisés par Graphite

La liste des ports utilisés par Graphite est la suivante :

Port	Serveur hébergeant	Fonctionnalité
2003	carbon-cache	Réception des métriques à écrire, au format texte clair.
2004	carbon-cache	Réception des métriques à écrire, au format binaire ( <i>les données sont sérialisées</i> ).
2013	carbon-relay	Réception des métriques à relayer, au format texte clair.
2014	carbon-relay	Réception des métriques à relayer, au format binaire ( <i>les données sont sérialisées</i> ).
80	carbon-relay et carbon-cache	Lecture des métriques ( <i>protocole http</i> ).
443	carbon-relay et carbon-cache	Lecture des métriques ( <i>protocole https</i> ).

## Si firewalld est présent ( *firewall activé par défaut sur RHEL / Alma / Rocky* )

### Vérifier que les ports sont ouverts

Voici comment vérifier que les ports des démons carbon-relay / carbon-cache sont ouverts sur leurs machines respectives :

```
firewall-cmd --list-ports
```

### Exemple de résultat

```
80/tcp 7763/tcp 7765/tcp 7766/tcp 7767/tcp 7768/tcp 7769/tcp 7770/tcp 7771/tcp 7772/tcp 7773/tcp 7777/tcp  
7780/tcp 50000/tcp
```

Dans cet exemple, aucun des ports de carbon-relay ( *2013/tcp* ) et de carbon-cache ( *2004/tcp* ) ne sont présents, ils sont donc bloqués.

### Sur les serveurs carbon-relay

Si le port du démon **carbon-relay** n'est pas ouvert, pour autoriser le trafic vers ce port, utiliser les commandes suivantes :

```
firewall-cmd --add-port=2013/tcp  
firewall-cmd --runtime-to-permanent
```

### Sur les serveurs carbon-cache

Si le port du **carbon-cache** n'est pas ouvert, pour autoriser le trafic vers ce port, utiliser les commandes suivantes :

```
firewall-cmd --add-port=2004/tcp  
firewall-cmd --runtime-to-permanent
```

## Passer le cluster Graphite en HTTPS

Les étapes suivantes permettent d'activer le HTTPS d'un cluster Graphite :

- Sur chaque nœud du cluster (carbon-relay et carbon-cache), activer le HTTPS de Graphite ( voir la page [Base de métrologie \( Graphite \)](#) section *Activation de HTTPS pour Graphite* ).
- Activer HTTPS pour les communications intra-cluster. Sur la machine du **carbon-relay** dans le fichier de configuration `/opt/graphite/webapp/graphite/local_settings.py` ( copier le fichier d'exemple `local_settings.py.example` si besoin ), dé-commenter la clé `INTRACLUSTER_HTTPS` et la passer à True

#### `/opt/graphite/webapp/graphite/local_settings.py`

```
[ ... ]  
  
INTRACLUSTER_HTTPS = True  
  
[ ... ]
```

- Ne pas oublier de changer, sur le serveur hébergeant le démon **carbon-relay**, les ports à utiliser pour la lecture des métriques ( *443 désormais au lieu de 80* ).

Cette configuration se fait via le fichier `/opt/graphite/conf/relay-rules.conf` ( copier le fichier d'exemple `relay-rules.conf.example` si besoin ) :

#### `/opt/graphite/conf/relay-rules.conf`

```
[default]  
default = true  
destinations = adresse_carbon_cache_1:2004,adresse_carbon_cache_2:2004  
read_destinations = adresse_carbon_cache_1:443,adresse_carbon_cache_2:443
```

- Si ce n'est pas déjà fait, préciser dans le fichier de configuration de la WebUi ( `/etc/shinken/modules/webui.cfg` ) que le cluster Graphite communique en HTTPS. Il faut ensuite redémarrer l'Arbiter pour prendre en compte la modification.

#### `/etc/shinken/modules/webui.cfg`

```
[ ... ]

graphite_backends                               *=https://CARBON_RELAY_SERVER:443

[ ... ]
```

- Ajouter le certificat de l'autorité de certification sur chaque nœud composant le cluster ( *le certificat de l'autorité de certification est celui utilisé pour générer chacun des certificats utilisés sur les différents serveurs* ).

## RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9

- Pour ajouter le certificat de l'autorité de certification dans la chaîne de confiance du système, il faut l'ajouter dans `/etc/ssl/certs/ca-bundle.trust.crt`.
- Les commandes suivantes permettent d'effectuer cette opération :

```
cp your_certificate.crt /etc/pki/ca-trust/source/anchors/
update-ca-trust
```

- Ensuite, sur l'ensemble des machines redémarrer le démon apache :

```
service httpd restart
```

## Debian 13

- Pour ajouter le certificat de l'autorité de certification dans la chaîne de confiance du système,, il faut le copier dans le dossier `/usr/local/share/ca-certificates/` et mettre à jour la chaîne de confiance.
- Les commandes suivantes permettent de réaliser cette opération :

```
cp your_certificate.crt /usr/local/share/ca-certificates/
update-ca-certificates
```

- Ensuite, sur l'ensemble des machines redémarrer le démon apache :

```
systemctl restart apache2
```

- Redémarrer le démon carbon-relay de Graphite:

```
service carbon-relay restart
```

## Comportement de Shinken avec un cluster Graphite

Configurer Shinken pour utiliser un cluster Graphite se fait très simplement, il suffit de changer les adresses utilisées pour la lecture et l'écriture des métriques.

Le cluster Graphite gère la haute disponibilité des données et la réplication, le changement d'architecture est transparent pour les utilisateurs de Shinken.

## Outils externes ( *Grafana* ) et relation UUID nom

En cas d'utilisation d'outils externes ( *comme Grafana par exemple* ) pour consulter les métriques, il faut également :

- configurer la récupération des données de l'inventaire sur les serveurs carbon-cache,
- autoriser les connexions au serveur d'inventaire depuis les serveurs carbon-cache.

( voir la page [Base de métrologie \( Graphite \)](#) section *Correspondance ID Nom de l'élément* )