Autorisations

Sommaire

Introduction

Grafana est une plateforme permettant de créer des tableaux de bord de visualisation pour les métriques. Dans ces tableaux de bord, la création de différents types de widget et de nombreuses options sont disponibles pour la visualisation des métriques.

Plus de détails sont disponibles sur le site officiel: https://grafana.com/

Les versions suivantes comportent une failles de sécurité critique (CVE-2021-43798). Nous vous déconseillons de les installer :

- 8.0.0 à 8.0.6
- 8.1.0 à 8.1.7
- 8.2.0 à 8.2.6
- 8.3.0

Installation

L'installation de Grafana sous CentOS se fait via un paquet RPM. La version de Grafana testée avec Shinken Entreprise est la **v8.3.2-1**.

Pour l'installer, utiliser la commande suivante:

```
yum install --nogpgcheck https://dl.grafana.com/enterprise/release/grafana-enterprise-8.3.2-1.x86_64.rpm
```

Une fois l'installation terminée, le lancement de Grafana dépend de la version de CentOS / RedHat / AlmaLinux utilisée.

- Sous CentOS 6

```
chkconfig grafana-server on
service grafana-server start
```

- Sous CentOS 7 et RedHat/AlmaLinux 8

```
systemctl enable grafana-server
systemctl start grafana-server
```

Après avoir installé puis lancé Grafana, l'interface sera accessible sur le **port 3000**. Les identifiants par défaut sont admin/admin

Connexion avec Graphite

Pour pouvoir récupérer les métriques générées par Shinken et enregistrées dans Graphite. Grafana doit avoir accès à Graphite. Pour cela, il faut ajouter dans Grafana une source de données Graphite.

Dans l'interface, aller dans la catégorie Data Sources , puis ajouter une nouvelle source de type Graphite, qui sera paramétrée comme suivant:

- **URL**: adresse du serveur hébergeant Graphite, sur le port 80. Par exemple:

```
http://localhost:80
```

- **Version**: La version de Graphite utilisée dépends de votre version de Shinken Enterprise:
  - **Avant** la version 02.08.01.01, il faut choisir **0.9.x**

? Unknown Attachment

  - **Après** la version 02.08.01.01, il faut choisir **1.1.x**

? Unknown Attachment

Si Grafana est installé sur un serveur différent du serveur Graphite, il faudra effectuer une étape de configuration supplémentaire. Par défaut, Graphite autorise seulement les connexions locales.

Pour permettre à des serveurs distants d'accéder à ses données, il faut le fichier de configuration Apache de Graphite **/etc/httpd/conf.d/graphite.conf** :

- Changer la ligne :

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

- En :

```
<VirtualHost 0.0.0.0:80>
```

Cette ligne permet de spécifier les interfaces réseau de la machine sur lesquelles effectuer l'écoute. Spécifier 0.0.0.0 ou * permettent d'écouter sur toutes les interfaces réseau. On peut mettre une seule interface à la place en spécifiant l'IP de l'interface réseau concernée.

- Redémarrer Apache pour prendre en compte les modifications
  Sur CentOS 6:

```
service httpd restart
```

Sur CentOS 7 et RedHat / AlmaLinux 8:

```
systemctl restart httpd
```

L'installation et la connexion de Grafana avec Graphite sont maintenant terminées.

Vous pouvez maintenant créer des tableaux de bord, ajouter des utilisateurs et permettre la visualisation des métriques de Shinken.

Intégration dans Shinken

Par défaut, cette version n'accepte pas d'être intégrée dans un widget page web (*pour plus de détail sur la configuration du widget voir la page : Widget Page web*).

Si l'authentification est activée dans Grafana et que vous souhaitez l'intégrer dans un widget web, il faut que Grafana soit accessible depuis la même adresse IP ou même le nom de domaine (exemple: shinken-solution.com est un nom de domaine) . Sans quoi le blocage CORS des navigateurs bloquera la connexion à Grafana.

Pour l'activer vous devez éditer le fichier de configuration Grafana **/etc/grafana/grafana.ini** dans la section "[security]":

```
allow_embedding = true
```

Puis redémarrer Grafana

Sur CentOS 6:

```
service grafana-server restart
```

Sur CentOS 7 et RedHat / AlmaLinux 8:

```
systemctl restart grafana-server
```

Créer un tableau de bord (dashbord)

Pour créer un tableau de bord,

1. Cliquer sur le "+", "Create", puis "Add new panel".

? Unknown Attachment

2. Définir un nom dans le menu de droite et dans la partie basse de l'écran, vous avez la composition de la requête.
   Pour générer un graphe, il faut cliquer sur "select metric", puis ajouter la métrique souhaitée.

? Unknown Attachment

3. Il faut sélectionner l'intervalle de visualisation.

? Unknown Attachment

Une fois terminé, le bouton "Save" en haut à droite, va sauvegarde le panneau dans le tableau de bord. Le nom inscrit sera le nom du tableau de bord qui peut être composé de plusieurs panneaux.

Récupération de l'URL à intégrer dans Shinken

Pour centraliser la visualisation des éléments supervisés par Shinken, il est possible d'intégrer les tableaux de bord Grafana dans un tableau de bord Shinken en utilisant le widget page web (*voir la page* Widget Page web).

Par exemple :

? Unknown Attachment

1. Allez sur votre panneau et cliquer sur share :

? Unknown Attachment

2. Dans l'onglet "Embed", sélectionnez le line comme ci-dessous :

? Unknown Attachment

   Pour obtenir un graphique évolutif en temps réel, remplacer dans le lien, toute la partie "from=XXXX&to=XXXX" par "from=now-6h&to=now" pour avoir un intervalle de 6h de visualisation.

3. Copier ce lien dans votre widget Page web.

   Exemple :

   ? Unknown Attachment

```
http://adresse_serveur:3000/d-solo/sLWNXIsMk/synchronizer?orgId=1&from=now-6h&to=now&panelId=2
```

Décomposition de l'adresse :

Paramètre	Définition
adresse_serveur	Adresse IP / Nom DNS du serveur
3000	Port par défaut de Grafana
d-solo	Permet de cacher les barres de navigation de Grafana pour n'afficher que les éléments visualisés
synchronizer	Nom du tableau de bord
from=now-6h	Intervalle d'affichage du graphes sur 6h
to=now	Intervalle jusqu'à maintenant
panelId=2	Numéro du panneau dans le tableau de bord

Passage en HTTPS

Dans le fichier de configuration de Grafana (*/etc/grafana/grafana.ini*),

Ajoutez :

```
protocol = https
cert_file = /chemin/vers/server.cert
cert_key = /chemin/vers/server.key
```

Puis redémarrer Grafana :

- Sur CentOS 6 :

```
service grafana-server restart
```

- Sur CentOS 7 et RedHat / AlmaLinux 8:

```
systemctl restart grafana-server
```

Lien vers le mapping nomuuid nécessaire pour Grafana, et suivi des requêtes

Les métriques stockées dans Graphite utilisent l'UUID des éléments comme clé.

L'Interface de Visualisation de Shinken va utiliser les UUID pour ses échanges avec Graphite.

Les outils externes qui échangent avec Graphite comme Grafana vont utiliser le nom des éléments.
Cela implique que Graphite possède une table de correspondance UUID nom pour répondre aux outils externes.

- Cette correspondance est fournie par le serveur d'inventaire (ce serveur est un composant du module de métrologie du Broker de Shinken).
-
  - La configuration de Graphite pour l'accès au serveur d'inventaire se fait via le fichier /opt/graphite/conf/shinken_inventory.conf
- Si la table de correspondance n'est plus à jour et qu'une requête par nom est demandée à Graphite, alors Graphite va faire la demande d'une nouvelle table de correspondance au serveur d'inventaire.
- Lorsqu'une nouvelle configuration de Shinken est déployée, le serveur d'inventaire renvoie une nouvelle table de correspondance.
  - Afin que tous les processus de Graphite/Apache soient mis au courant, le fichier /opt/graphite/storage/whisper/.cacheinvalidation est mis à jour
    - Ce fichier ne doit pas être modifié.
    - En cas de problème, il est recréé, et le cache vidé.

Lors de l'utilisation d'un cluster graphite, la configuration pour gérer la correspondance *UUID nom* doit être faite sur **tous** les serveurs faisant tourner un Carbon-Cache (*noeud de stockage des métriques*)

Les logs de chargement de la table de correspondance *UUID nom* sont disponibles dans le fichier **/opt/graphite/storage/log/webapp/info.log**

Pour suivre la mise à jour du mapping nom uuid et les requêtes pour un hôte particulier, il suffit de remplir le fichier **/opt/graphite/storage/whisper/.apache_graphite_host_filter_log** avec le nom de l'hôte. Les mises à jour dans la table de correspondance le concernant, ainsi que les requêtes de recherches de métriques seront toutes disponibles dans le fichier de log.

Paramètres de connexion aux serveurs d'inventaire

- Graphite se base sur les informations du fichier **/opt/graphite/conf/shinken_inventory.conf** pour aller chercher les informations qui lui permettront d'assurer la correspondance entre les noms et les UUID

Nom	Type	Unité	Défaut	Commentaire
`ENABLE`	Booléen	---	**1**	Permets d'activer ou désactiver la recherche des correspondances entre les ID et les noms (*1 pour activer, 0 pour désactiver*).
`URI`	Liste Texte	---	**http://localhost:52000/inventory/**	URL séparées par des virgules Permet de contacter chacun des modules de métrologie qui fournit des métriques à ce serveur Graphite
`TIMEOUT`	Numérique	---	**10**	Timeout général, utilisé pour les opérations bloquantes comme les tentatives de connexion à un serveur d'inventaire, par exemple. (*secondes*).

Après tous changements du fichier de configuration, penser à redémarrer Apache pour que Graphite prenne les modifications en compte

Commande pour prendre en compte les changements de configuration dans Graphite :

```
service httpd restart
```

Autoriser les connexions aux serveurs d'inventaire

Configurer les modules de métrologie Graphite

Si le serveur Graphite et les Brokers faisant tourner les modules de métrologie Graphite sont sur des machines différentes, il faut configurer le serveur d'inventaire des modules de métrologie Graphite pour écouter sur les IP publiques de leur machine.

Pour cela, sur le serveur de l'Arbiter, éditer les fichiers de configuration des modules Graphite et décommenter la ligne du paramètre

/etc/shinken/modules/graphite.cfg

```
broker__module_graphite_perfdata__inventory_server__address 0.0.0.0
```

(*pour passer sa valeur de **127.0.0.1** à **0.0.0.0***)

Redémarrer l'Arbiter pour appliquer le changement de configuration

Ouvrir le port du serveur d'inventaire sur le firewall (*firewalld*)

Si le serveur d'un Broker qui fait tourner le module de métrologie Graphite dispose d'un firewall (*firewalld par défaut sur les systèmes Redhat et dérivés*), la commande suivante permet d'obtenir la liste des ports autorisés

```
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, le port 52000/tcp (*port par défaut du serveur d'inventaire du module de métrologie Graphite*) n'est pas listé, il est donc bloqué par défaut

Les commandes suivantes, à lancer sur le serveur du Broker, permettent d'autoriser les connexions :

```
firewall-cmd --add-port=52000/tcp
firewall-cmd --runtime-to-permanent
```

Compatibilité historique

En cas d'impossibilité d'accès au serveur d'inventaire des modules de métrologie (*ports bloqués, paramètres par défaut incompatibles avec votre configuration, ...*), Graphite peut utiliser l'ancienne méthode que Shinken avait déployé pour fournir ces informations avec MongoDB.

- L'accès est configuré dans Graphite dans le fichier **/opt/graphite/conf/mongodb.conf**.

L'accès via Mongo est déprécié et est voué à disparaître.

En effet, Graphite ne peut consulter qu'une seule base Mongo pour obtenir les correspondances de noms, il est ainsi obligé d'utiliser la base centrale, qui est souvent aussi la plus chargée.

Configuration de l'accès à MongoDB

Pour se connecter au serveur Mongo, deux méthodes sont disponibles:

- **Connexion directe**: Par défaut, mais non sécurisée.
- **Tunnel SSH**: Shinken se connecte au serveur Mongo au travers d'un tunnel SSH pour plus de sécurité

Connexion directe au serveur Mongo

Par défaut, Graphite se connecte de manière directe au serveur Mongo pour y lire et écrire sa table de correspondance.

Dans la configuration de Graphite, on sait que la connexion se fait de manière directe lorsque le paramètre "USE_SSH_TUNNEL" est à 0.

Cette méthode de connexion a pour avantage d'être facile à configurer au niveau de Shinken. Par contre, elle oblige à permettre l'accès à la base Mongo au monde extérieur, et donc s'exposer à des problèmes de sécurité.

- La sécurisation de la base Mongo est bien sur toujours possible (*voir la page Sécurisation des connexions aux bases MongoDB*) mais bien plus complexe à mettre en place.
- La méthode de connexion par SSH est donc préférable pour des raisons pratiques et de sécurité.

Connexion par SSH au serveur Mongo

Graphite peut également se connecter au serveur mongo par tunnel SSH (*pour des raisons de sécurité*).

- En effet, le paramétrage de mongoDB (*/etc/mongod.conf*) permet de définir sur quelle adresse ce dernier écoute les requêtes.
    - En n'autorisant seulement l'adresse 127.0.0.1, cela évite d'ouvrir la base au monde extérieur.
        - Dans la configuration du serveur Mongo (*/etc/mongod.conf*), assurez-vous que le paramètre "*bind_ip*" est positionné pour n'écouter que sur l'interface locale:
          `bind_ip= 127.0 . 0.1`

Comme toutes les connexions vers MongoDB, il est possible, et même recommandé, de sécuriser la communication via un tunnel chiffré SSH.

Le paramétrage de la connexion à MongoDB depuis Graphite, se fait en éditant les options suivantes (*dans **/opt/graphite/conf/mongodb.conf***):

Nom	Type	Unité	Défaut	Commentaire
`URI`	Texte	---	**mongodb://localhost/?w=1&fsync=false**	URI du serveur Mongo L'adresse de la base Mongo à utiliser est celle configurée dans le Module Graphite-Perfdata
`DATABASE`	Texte	---	**shinken**	Nom de la base contenant les données d'inventaire sur le serveur Mongo
`COLLECTION`	Texte	---	**metrology_inventory**	Nom de la collection contenant les données d'inventaire
`USE_SSH_TUNNEL`	Booléen	---	**0**	Paramètre permettant d'activer ou non l'utilisation d'un tunnel SSH Valeurs possibles : • **0** (*désactivé*) • **1** (*activé*)

`SSH_USER`	Texte	---	**shinken**	Utilisateur sur le serveur Mongo à contacter pour établir la connexion
`SSH_KEYFILE`	Texte	---	**/opt/graphite/conf/id_rsa**	Chemin vers la clé SSH privée utilisée
`SSH_TUNNEL_TIMEOUT`	Nombre	seconde	**5**	Durée du timeout au bout duquel on détermine si l'établissement du tunnel a échoué

Après tous changements du fichier de configuration, penser à redémarrer Apache pour que Graphite prenne les modifications en compte

Commande pour prendre en compte les changements de configuration dans Graphite :

```
service httpd restart
```

Graphite étant hébergé par le service apache, il n'a pas accès au répertoire **/var/lib/shinken** et il n'a donc pas accès à la clé SSH **/var/lib/shinken/.ssh/id_rsa**. C'est pour cette raison que la clé SSH utilisée pour le tunnel est situé dans **/opt/graphite/conf/id_rsa**.

Deux solutions sont disponibles :

- Générer une nouvelle clé SSH pour apache / graphite à l'aide de la page suivante : Création automatique et gestion de la clé SSH de l'utilisateur shinken
  - Lors de la génération de la clé, il est possible de spécifier directement le chemin suivant : **/opt/graphite/conf/id_rsa**
  - Il faudra ajouter cette nouvelle clé publique (*/opt/graphite/conf/id_rsa.pub*) sur le/les serveurs MongoDB (*dans le fichier ~shinken/.ssh/authorized_keys*)
  - Cette clé sera indépendante et non impactée par un changement de clé SSH sur l'utilisateur "shinken".
- Utiliser la clé SSH de l'utilisateur "shinken" présent sur le serveur.
  - La clé publique est sûrement déjà présente sur les serveurs MongoDB.
  - Il faut copier la clé privée et changer les droits pour l'utiliser et la maintenir à jour en cas de changement.

```
cp /var/lib/shinken/.ssh/id_rsa* /opt/graphite/conf/
chown apache:apache /opt/graphite/conf/id_rsa
```

Attention : un lien symbolique entre les deux fichiers ne fonctionnera pas, car l'utilisateur apache n'a pas les droits suffisants pour lire le fichier originel, et SSH refusera d'utiliser une clé dont les droits d'accès sont trop permissifs.

Mise à jour d'une version supérieur à 5.4.0

Il faut simplement lancer la commande suivante :

```
yum update https://dl.grafana.com/enterprise/release/grafana-enterprise-X.X.X.x86_64.rpm
```

Pour la partie intégration avec le widget web de Shinken, si le paramètre "allow_embedding" ne se trouve pas dans le fichier **/etc/grafana/grafana.ini**, vous pouvez l'ajouter dans la section "[security]" de ce fichier :

```
[security]
allow_embedding = true
```

Vous pouvez redémarrer grafana :

Sur CentOS 6:

```
service grafana-server restart
```

Sur CentOS 7 et RedHat / AlmaLinux 8:

```
systemctl restart grafana-server
```

Une fois Grafana mis à jour, il suffit de rafraîchir la page et de s'authentifier

Authentification avec le widget Page Web

Pour fonctionner avec le widget Page web, Grafana va nécessiter un paramétrage spécifique (*voir la page* Widget Page web).

- L'authentification se fait avec des cookies.
- Dans les dernières versions de Chrome/Internet Explorer/Edge (*Firefox n'est pas concerné*), l'utilisation de requête "cross-site" par un cookie n'est plus autorisé.
- Cela signifie que si l'application n'est pas installé sur le même serveur que la WebUI de Shinken, l'authentification depuis le widget Page Web ne fonctionnera pas.

Pour palier à ce problème, nous allons utiliser HAProxy (*Version 1.5.18*) pour récupérer le flux de ce site et simuler sa présence sur le serveur où est installé la WebUI.

Configuration

Pour afficher les graphes d'un Grafana qui n'est pas installé sur le serveur où se trouve la WebUI Shinken. Il faut installer HAProxy sur chacun des serveurs où ces graphes doivent être affiché dans le widget Page Web.

```
yum install haproxy
```

Ajouter une exception dans SELinux pour HAProxy:

```
setsebool -P haproxy_connect_any=1
```

Modifier le fichier de configuration */etc/haproxy/haproxy.cfg* et le remplir avec ces informations :

```
#---------------------------------------------------------------------
# Example configuration for a possible web application.  See the
# full configuration options online.
#
#   http://haproxy.1wt.eu/download/1.4/doc/configuration.txt
#
#---------------------------------------------------------------------

#---------------------------------------------------------------------
# Global settings
#---------------------------------------------------------------------
global
    # to have these messages end up in /var/log/haproxy.log you will
    # need to:
    #
    # 1) configure syslog to accept network log events.  This is done
    #    by adding the '-r' option to the SYSLOGD_OPTIONS in
    #    /etc/sysconfig/syslog
    #
    # 2) configure local2 events to go to the /var/log/haproxy.log
    #   file. A line like the following can be added to
    #   /etc/sysconfig/syslog
    #
    #    local2.*                       /var/log/haproxy.log
    #
    log         127.0.0.1 local2

    chroot      /var/lib/haproxy
    pidfile     /var/run/haproxy.pid
    maxconn     4000
    user        haproxy
    group       haproxy
    daemon

    # turn on stats unix socket
    stats socket /var/lib/haproxy/stats

#---------------------------------------------------------------------
# common defaults that all the 'listen' and 'backend' sections will
# use if not designated in their block
#---------------------------------------------------------------------
defaults
    mode                    http
    log                     global
    option                  httplog
    option                  dontlognull
    option http-server-close
    option forwardfor       except 127.0.0.0/8
    option                  redispatch
    retries                 3
    timeout http-request    10s
    timeout queue           1m
    timeout connect         10s
    timeout client          1m
    timeout server          1m
    timeout http-keep-alive 10s
    timeout check           10s
    maxconn                 3000

listen  grafana
  bind    SERVEUR_WEBUI:3000
  server    SERVEUR_GRAFANA SERVEUR_GRAFANA:3000
```

Il faut remplacer **SERVEUR_WEBUI** et **SERVEUR_GRAFANA** par le nom ou l'adresse IP des serveurs en questions

Démarrer HAProxy

```
systemctl enable haproxy
systemctl start haproxy
```

Vous pouvez ajouter l'URL suivante dans le widget Page Web : http://SERVEUR_WEBUI:3000/XXXXXXX mais pas ~~http://SERVEUR_GRAFANA:3000/XXXXXXX~~ La première authentification est nécessaire.

Log HAProxy

Par défaut HAProxy n'a pas de fichier de log.

Pour en générer un il faut :

- Éditer le fichier */etc/rsyslog.conf* et décommenter les lignes suivantes :

```
$ModLoad imudp
$UDPServerRun 514
```

- Créer et ajouter dans le fichier */etc/rsyslog.d/haproxy.conf :*

```
local2.* /var/log/haproxy.log
```

- Redémarrer rsyslog et haproxy :

```
systemctl restart rsyslog
systemctl restart haproxy
```