Sécuriser les communications vers l'Arbiter

Sommaire

- Contexte
- Paramétrage du SSL
  - Fichier /etc/shinken/shinken.cfg
  - Fichier /etc/shinken/arbiters/arbiter-master.cfg

Contexte

Afin de sécuriser les communications de l'Arbiter, il est possible d'activer le SSL sur le démon, via des paramètres dédiés à cet effet.

Attention, à la différence des autres démons, le Synchronizer et l'Arbiter n'ont pas de fichiers "ini" dans le répertoire /etc/shinken/daemons. En effet, l'Arbiter utilise le fichier /etc/shinken/shinken.cfg et le Synchronizer utilise le fichier /etc/shinken/synchronizer.cfg, fichiers qui contiennent à la fois des paramètres globaux mais aussi leurs propres paramètres de configuration.

Important

Attention à ne pas confondre le protocole utilisé pour la communication des démons **ET** le protocole utilisé pour l'accès des utilisateurs /administrateurs aux interfaces Shinken via leurs navigateurs Internet (interface de configuration et interface de visualisation).
Nous traitons ici le paramétrage du protocole de communication du démon.

Paramétrage du SSL

Fichier /etc/shinken/shinken.cfg

Pour activer le SSL, il faut tout d'abord modifier le fichier /etc/shinken/shinken.cfg qui contient les paramètres du démon Arbiter. (ce fichier est à modifier sur le serveur "central", donc sur le serveur qui héberge l'Arbiter)

Ce fichier contient un bloc concernant le paramétrage des ports d'écoutes du démon :

```
#======== Arbiter daemon HTTP(S) listening  =========

                                                # If enabled, the arbiter daemon will listen in HTTPS
instead of HTTP protocol.
                                                # Note: default pem/cert and key files are for sample
only. You need to generate
                                                # your own with your PKI.
                                                # by default: 0 (disabled)

use_ssl=0
ca_cert=/etc/shinken/certs/ca.pem
server_cert=/etc/shinken/certs/server.cert
server_key=/etc/shinken/certs/server.key


                                                # Should the arbiter connections will force the HTTPS
certificates name checks
                                                # If enabled and a distant certificate is not the
same as the daemon address, then
                                                # the connection will be refused.
hard_ssl_name_check=0


                                                # Which HTTP backend to start the listening daemon
with.
                                                # Currently only auto is managed
http_backend=auto



                                                # Which addr to bind for the arbiter daemon
                                                # by default: 0.0.0.0 (means all interfaces)
bind_addr=0.0.0.0
```

Ce fichier contient :

- le paramètre **use_ssl** à passer à 1 pour activer le SSL.
- les paramètres **server_cert** et **server_key**:
    - Les certificats utilisés par défaut sont auto-signés et donc fournis à titre d'exemple, ils ne sont en aucun cas approuvés par une autorité de certification.
    - Il faut donc que vous placiez vos propres certificats dans le répertoire **/etc/shinken/certs/** et modifiez alors les chemins si besoin.
- Le paramètre **hard_ssl_name_check** permet de forcer la vérification que l'adresse de connexion est strictement identique a ce qui a été déclaré dans le certificat SSL.

Dans le cas d'un Spare, il faut aussi modifier, de la même manière, le fichier /etc/shinken/shinken.cfg de la machine hébergeant l'arbiter-spare

Fichier /etc/shinken/arbiters/arbiter-master.cfg

Il faut déclarer le démon Arbiter dans l'architecture de supervision, pour que les autres démons qui en ont besoin reçoivent les informations nécessaires

- Pour cela, le fichier utilisé est **/etc/shinken/arbiters/arbiter-master.cfg** sur le serveur central.
    - La variable use_ssl permet de signaler en central, que pour contacter l'Arbiter, il faut utiliser une connexion SSL.
    - Il faut donc passer le paramètre **use_ssl** à 1.