Architecture Distribuée

Sommaire

Description

Shinken peut, au travers de cette source, importer sa configuration depuis des fichiers plats. La syntaxe d'import de ces fichiers est la même que la syntaxe Nagios.

Il est donc possible d'importer des fichiers de configuration Nagios dans Shinken, et ainsi migrer sa configuration Nagios vers Shinken Entreprise facilement.

Aussi, les fichiers de configuration Shinken sont évidemment supportés et importables dans Shinken Entreprise.

Activation du module

La source ne peut s'activer que sur le Synchronizer.

- L'activation de la source s'effectue en ajoutant le nom de cette source dans le fichier de configuration du démon Synchronizer.
- Pour ce faire, ouvrez le fichier de configuration à l'emplacement **/etc/shinken/synchronizers/synchronizer-master.cfg**, et ajouter le nom de votre collecteur "cfg-file-import".

Exemple: par défaut, nous livrons une source dont le nom est "cfg-file-sample":

```
define synchronizer {
    [...]
    sources                  Source 1, Source 2, Source 3, cfg-file-sample
    [...]
}
```

Pour prendre en compte le changement de configuration, redémarrer le Synchronizer:

```
service shinken-synchronizer restart
```

Configuration

La configuration du module se trouve par défaut dans le fichier **/etc/shinken/sources/cfg-file-sample-example.cfg**

- Vous trouverez aussi systématiquement un exemple dans **/etc/shinken-user-example/configuration/daemons/synchronizers/sources/cfg-file-import/cfg-file-sample-example.cfg**

Exemple de fichier de configuration

```
#===============================================================================
# cfg-file-sample
#===============================================================================
# Daemons that can load this source:
# - synchronizer
# This source import the cfg-config sample from Shinken Enterprise update.
#===============================================================================


define source {
    source_name                             cfg-file-sample-example
    enabled                     0
    order                                19
    import_interval                   5
    module_type                     cfg-file-import
    cfg_path                            /etc/shinken-user/source-data/source-data-cfg-sample/definition-
source-data-cfg-sample.cfg
    description                             This source is about loading the default Shinken Enterprise
packs

    # The list of properties to be used as sync_keys in addition to the item name. Properties not managed
by Shinken can be added here.
    # properties_used_as_synckey        address

    # Properties which can be defined in the items from the source but which Shinken will not import.
    # not_stored_properties

    # With this option if item in cfg hasn't a SE_UUID, the source will ask the synchronizer to search a
match in staging or in working area. If found, it will be inserted in the file.
    # update_cfg_with_staging_se_uuid        0
}
```

Détails des sections composant le fichier de configuration

Identification de la source

Il est possible de définir plusieurs instances de module de type **`cfg-file-import`** dans votre architecture Shinken.

- Chaque instance devra avoir un nom unique.

Nom	Type	Unité	Défaut	Commentaire
`source_name`	Texte	---	**cfg-file-sample-example**	**Nous vous conseillons de choisir un nom en fonction de l'utilisation du module pour que votre configuration soit simple à maintenir.** Chaîne de caractères composée de lettres, de chiffres et des caractères _ et - . • Doit être unique. • Doit commencer par une lettre. • D'une longueur maximum à 40 caractères. • Ne doit pas contenir le caractère "$".
`module_type`	Texte	---	**cfg-file-import**	Ne peut être modifié

Interval d'import et ordre de la source

Nom	Type	Unité	Défaut	Commentaire
`import_interval`	Entier positif	minute	**0**	Délai écoulé entre les imports automatiques de la source. Si 0, l'import de la source ne sera jamais exécuté automatiquement. (uniquement manuellement)
`order`	Entier positif	---	**2**	L'ordre de la source dans l'interface de configuration (*A un impact dans la fusion des données lors des imports de sources*). Voir la page du Synchronizer pour plus d'information au sujet des fusions. **Remarque** *Si vous changez l'ordre depuis l'interface (page d'accueil), le fichier .cfg sera mis à jour.*

Emplacement des fichiers de configuration

Afin de récupérer les éléments Shinken, contenus dans des fichiers .cfg, il faut spécifier un unique fichier de définition de la source qui contiendra la liste des chemins vers les fichiers à importer. (Voir Importer ses propres fichiers CFG pour plus de détail)

Exemple de définition pour la source cfg-file-sample

```
...
        cfg_path /etc/shinken-user/source-data/source-data-cfg-sample/definition-source-data-cfg-sample.cfg
...
```

Nom	Type	Unité	Défaut	Commentaire
`cfg_path`	Texte	---	**---**	Emplacement du fichier de définition de la source. Ce fichier servira à définir où sont les fichiers .cfg à importer.

Clés de synchronisation (sync_key)

Défini la liste des propriétés qui seront utilisées pour générer les clés de synchronisation.

À noter : On ne peut pas supprimer les valeurs par défaut, mais on peut les compléter.

Nom	Type	Unité	Défaut	Commentaire
`properties_used_as_synckey`	Texte	---	**_SE_UUID, host_name**	Permet de compléter la clé de synchronisation déjà existante.

Propriétés non récupérées

Il est possible de définir des propriétés que la source ne devra pas récupérer. Ceci est utile si vos instances Shinken distantes utilisent des propriétés propres à leur fonctionnement.

Nous vous conseillons de laisser la valeur "_SE_UUID" et d'en rajouter d'autres si vous le souhaitez. Si vous supprimez la valeur "_SE_UUID" vos hôtes possèderont les mêmes "_SE_UUID" que sur le Shinken sur lesquels ils ont été récupérés.

Nom	Type	Unité	Défaut	Commentaire
`not_stored_properties`	Texte	---	**_SE_UUID**	Empêche la récupération de certaines propriétés des hôtes récoltés

Ajout des SE_UUID dans les fichiers de configuration

Si un élément importé dans un des fichiers .cfg ne contient pas pas de SE_UUID, Shinken va lui en générer un. Il est alors possible de faire en sorte que cet SE_UUID soit retranscrit dans le fichier .cfg grâce à ce paramètre.

```
...
        # With this option if item in cfg hasn't a SE_UUID, the source will ask the synchronizer to search
a match in staging or in working area. If found, it will be inserted in the file.
        # update_cfg_with_staging_se_uuid 0
...
```

Nom	Type	Unité	Défaut	Commentaire
`update_cfg_with_staging_se_uuid`	Booléen	---	**0**	• 0 : N'ajoute pas les SE_UUID des éléments importés dans les fichiers .cfg • 1 : Ajout les SE_UUID des éléments importé dans les fichiers .cfg