

Comment interpréter les informations et les erreurs courantes du Healthcheck

Sommaire

Le démon est configuré avec l'adresse "localhost"
Le démon est injoignable
Le démon est en cours d'arrêt
Le démon n'a pas encore reçu sa configuration de l'Arbiter
Le démon a chargé une configuration enregistrée
Le démon et son Arbiter ont des versions différentes
Conflit d'Arbiter sur un démon
La dernière connexion de l'Arbiter remonte à trop longtemps
La configuration de l'Arbiter n'a pas été trouvée
Les serveurs ne sont pas à la même heure
Un démon est bloqué et doit être redémarré
Un démon est configuré en tant que Spare
 Un démon master a un démon spare de désigné (seulement pour les Brokers)
 Un démon master autorise son spare à ne pas avoir la même liste de modules
 Un démon master n'a pas de démon spare de désigné (seulement pour les Brokers)
 Un démon spare qui n'a pas de démon master (seulement pour les Brokers)

Informations et détections dans le cadre de virtualisation Vmware
 Détection de la présence des "VMWare tools"
 Détection d'un pourcentage élevé de "CPU Stolen"
 Le démon Gatherer n'est pas disponible

Un module a redémarré de manière imprévue
Erreur de communication entre les démons
Erreur d'encodage dans les fichiers de configuration d'un démon
Un scheduler n'a pas de broker ou de poller ou de réagir
Un module a mis trop de temps à répondre aux demandes d'informations/statistiques
Erreurs de configuration concernant les données de météorologie
 Les erreurs d'écriture
 Cas 1 : Les données d'un royaume ne sont pas sauvegardées
 Cas 2 : Plusieurs brokers du même royaume écrivent dans une même base graphite
 Cas 3 : Le module graphite sauvegarde uniquement des royaumes différents de celui du broker
 Cas 4 : Le paramètre realm_store_only contient un royaume qui n'existe pas
 Cas 5 : Le port de Graphite-Perfdata est invalide
 Cas 6 : Le nom de l'hôte de Graphite-Perfdata est invalide
 Les erreurs de lecture
 Cas 1 : La configuration du module webui.cfg précise un royaume qui n'existe pas
 Cas 2 : La configuration du module webui.cfg précise un royaume non géré par le broker
 Cas 3 : Le serveur graphite précisé dans la configuration de webui ne gère pas les données de ce royaume
 Cas 4 : Le serveur graphite précisé dans la configuration de webui n'est pas un serveur graphite connu
 Cas 5 : Le broker gère des royaumes qui ne sont pas configurés dans les graphite_backends de la webui
 Cas 6 : Le broker rencontre des erreurs pour lire les données sur les serveurs graphite
 Cas 7 : Le port de Graphite n'est pas correct
 Cas 8 : L'adresse locale des serveurs de lecture ou d'écriture de Graphite n'a pas pu être résolu

Erreurs de format dans le paramètre graphite_backends de la WebUI
 Cas 1 : Pas de royaume dans le backend
 Cas 2 : Trop de séparateur de royaume
 Cas 3 : Pas de protocole
 Cas 4 : Pas de port HTTP
 Cas 5 : Pas de nom d'hôte ou d'adresse IP

Erreurs dans le contenu de la valeur de la clé graphite_backend
 Cas 1 : Port invalide
 Cas 2 : Protocole non supporté

Le démon a bloqué une tentative de chargement d'objet malveillant

Contexte

Cette page a pour but de vous décrire la mise en place du pack **linux_by_ssh** :

- Le déploiement du pack sur votre plateforme,
- La configuration de vos connexions SSH,
- La mise à jour du package d'OpenSSH (*si nécessaire*),
- La mise en place de pré-requis pour certains checks.

Procédure de mise en place du pack

En préambule, il faut :

1. récupérer la dernière version du pack auprès de l'équipe Shinken ou de votre partenaire,
2. Transférer le pack sur votre Synchronizer et votre/vos Pollers.

En utilisant l'installateur "install.sh" (sur Poller / Synchronizer)

Commun à tout Poller / Synchronizer

Que ce soit sur un Synchronizer où sur un Pollers :

1. Désarchiver le pack :

```
tar --no-same-owner -xf NOM_DU_PACK.tar.xz
```

2. Entrée dans le répertoire extrait de l'archive et lancer le "install.sh"

```
cd NOM_DU_PACK/  
./install.sh
```

3. L'installateur dépose les éléments suivants :

- Le fichier de configuration de la source "**shinken-additional-packs-import**" qui servira à importer la définition du pack dans le **Sync hronizer**.
- Les fichiers de définitions des éléments du pack, à importer dans le **Synchronizer**.
- Les sondes du pack.

Exemple de mise en place :

```
=====
INSTALL PACK [ shinken-linux_by_ssh ]
=====

-----
|   COLLECTOR [ shinken-additional-packs-import ]

=> ADDING Collector cfg file
    - Path : /etc/shinken/sources/shinken-additional-packs-import.cfg
=> CREATING Collector DATA space
    - Path : /etc/shinken-user/source-data/source-data-shinken-additional-packs-
import
=> ADDING Definition File [ definition_shinken-additional-packs-import.cfg ]
    - Path : /etc/shinken-user/source-data/source-data-shinken-additional-
packs-import/definition_shinken-additional-packs-import.cfg
    - Changing owner ( to "shinken" user )
    - Setting rights

-----
|   MONITORING DEFINITION:

=> Installing definition files in source-data folder ( Collector => shinken-additional-
packs-import )
    - Path : /etc/shinken-user/source-data/source-data-shinken-additional-packs-
import
    - Changing owner ( to "shinken" user )
    - Setting rights

-----
|   PROBES:

=> Deleting previous probes
=> Installing probes
    - Path : /var/lib/shinken-user/libexec/shinken-additional-packs/shinken-
linux_by_ssh
    - Setting owner ( to "shinken" user )
    - Setting rights
```

En suivant, spécifiquement sur le Synchronizer

1. Si le collecteur "**shinken-additional-packs-import**" n'est pas présent dans votre Synchronizer (*par exemple : première fois que vous utilisez un installateur de pack Shinken*) :

Ajouter le nom de la nouvelle source au **Synchronizer** en modifiant le paramètre **sources** du fichier **/etc/shinken/synchronizers/synchronizer-master.cfg**.

```
define synchronizer {
    [...]
    sources          Source 1, Source 2, Source 3, shinken-additional-packs-import
    [...]
}
```

et en suivant, redémarrez le Synchronizer pour que le collecteur soit pris en compte :

```
service shinken-synchronizer restart
```

2. Lancer l'import du collecteur "**shinken-additional-packs-import**":

? Unknown Attachment

- En suivant, vous devrez peut-être accepter ou des nouveautés ou des différences qui seraient liés à ce pack (*en fonction de l'évolution du pack*).

Manuellement

Commun à tout Poller / Synchronizer

Si vous avez déjà installé le pack

Nous vous conseillons de supprimer les anciens dossiers dans lesquels le pack était présent, pour être sûr d'avoir la dernière version de tous les fichiers livrés.

Pour supprimer ces dossiers, utilisez la commande suivante (*si vous avez suivi à la lettre la procédure d'installation manuelle*) :

```
rm -rf /var/lib/shinken-user/libexec/linux_by_ssh
```

Extraire le pack

Transférer sur votre machine et décompressez le fichier contenant le pack à l'aide de la ligne de commande ci-dessous (en remplaçant "**PACK__shinken__NON-DU-PACK__VERSION__OS.tar.gz**" par le nom du pack qui vous est livré.

- Pour Linux (*le paramètre --no-same-owner permet de ne pas conserver les attributs d'utilisateurs/roles présent de la création du pack*) :

```
tar -xjfv --no-same-owner PACK__shinken__NON-DU-PACK__VERSION__linux.tar.xz -C ./
```

- Pour Windows (*Rôle de Poller*) :

```
tar -xvzf PACK__shinken__NON-DU-PACK__VERSION__windows.tar.gz -C ./
```

Entrez dans le répertoire créer :

```
cd NOM_DU_PACK/
```

Les sondes (mises en place)

On place les sondes :

```
probe_path="/var/lib/shinken-user/libexec/shinken-additional-packs/shinken-linux_by_ssh"
mkdir --parents $probe_path
cp -r ./probes/* $probe_path
chown -R shinken:shinken $probe_path
chmod -R 750 $probe_path
```

Sur le Synchronizer

Ensuite, plaçons maintenant la définition des éléments de supervision du pack au bon endroit :

- Les fichiers de configuration :
 - Nous vous conseillons de créer un Collecteur de type (*cfg-file-import*) qui permet d'importer des définitions d'éléments Shinken depuis les fichiers de définitions présents dans ce pack.
 - Vous trouverez la procédure de mise en place dans la page de documentation suivante ([Collecteur de type \(cfg-file-import \) - Import depuis des fichiers au format .cfg](#))
 - Pour la suite des explications, nous considérerons que ce collecteur s'appelle "**shinken-additional-packs-import**"
 - Une fois le collecteur mis en place, nous allons déployer le pack au bon endroit :
 - En premier, les fichiers de définition des éléments (*modèles d'hôtes, checks, commandes, ...*) du pack :

```
shinken_collector_data_path="/etc/shinken-user/source-data/source-data-shinken-
additional-packs-import"
cp -r ./monitoring-definitions/packs/* $shinken_collector_data_path
chown -R shinken:shinken $shinken_collector_data_path
chmod -R 744 $shinken_collector_data_path
```

- Ensuite, les données globales

```
shinken_pack_globaldata_path="/etc/shinken/resource.d/shinken-additional-packs-import"
mkdir --parents $shinken_pack_globaldata_path
cp -r ./monitoring-definitions/global-data" "$shinken_pack_globaldata_path"
chown -R "$user:$user" "$shinken_pack_globaldata_path"
chmod -R 744 "$shinken_pack_globaldata_path"
```

- Redémarrer shinken

```
service shinken-synchronizer restart
```

- Lancer l'import du collecteur "**shinken-additional-packs-import**",

? Unknown Attachment

- En suivant, vous devrez peut-être accepter les nouveautés / différences qui seraient liés à ce pack (*en fonction de l'évolution du pack*).

Vous pouvez également supprimer le dossier qui a été créé suite à la décompression du pack en utilisant la commande suivante :

```
rm -rf ./linux_by_ssh
```



Attention à ne pas supprimer le tar.gz, connaître la version exacte du pack qui vous a été livré vous sera utile en cas de problème avec votre pack.

Si vous avez déjà installé une version précédente de ce pack, il vous faudra supprimer les anciens dossiers dans lesquels le pack était présent (*afin d'éviter de garder d'ancien fichier en cas d'évolution de la structure du pack*).

Pour supprimer ces dossiers, utilisez la commande suivante :

```
rm /etc/shinken/sources/shinken-additional-packs-import.cfg
rm -rf /etc/shinken-user/source-data/source-data-shinken-additional-packs-import/
rm -rf /var/lib/shinken-user/libexec/shinken-linux_by_ssh
```

Comment configurer la connexion SSH ?

Pour l'exécution correcte des commandes du pack linux_by_ssh, vous aurez besoin d'une connexion SSH.

Quelques informations au préalable sont nécessaires pour la bonne compréhension de cette partie.

- D'une part, du côté de l'**architecture Shinken**, l'exécution des checks (*plus exactement les sondes*) sont réalisées par les Pollers, en tant qu'utilisateur "**shinken**".
 - Donc l'utilisateur "**shinken**" devra avoir accès aux clefs SSH que vous utiliserez pour la connexion SSH sur les serveurs distants monitorés.
- D'autre part, du côté des **machines Linux supervisées**,
 - un **nom d'utilisateur**, et une **clé SSH** ou **mot de passe** sont requis.
 - Dans le modèle linux_by_ssh, des données sont prévues à cet effet.

Nous conseillons l'utilisation d'un utilisateur spécifique (*pour le service de supervision*) ainsi que l'utilisation d'une connexion via clé SSH, afin d'éviter l'utilisation du super utilisateur root qui n'est pas requis par les checks.

Côté client (machine ou serveur Linux supervisé)

Si votre utilisateur de supervision n'est pas déjà créé sur votre linux à superviser, depuis un terminal de la machine supervisée "**linux-1**" (en root), il faut créer un nouvel utilisateur local avec mot de passe.

- dans cet exemple, nous utilisons "**user-service-shinken**" mais vous pouvez créer un autre utilisateur.

```
[root@linux-1 ~]# adduser -m -r user-service-shinken
[FACTULTATIF] : [root@linux-1 ~]# passwd user-service-shinken
```



Notez que la mise en place d'un mot de passe pour cet utilisateur n'est pas obligatoire, mais il vous faudra copier la clé SSH via la **méthode manuelle** expliquée plus bas, car la commande automatique ssh-copy-id requiert un mot de passe pour l'utilisateur du système de destination.

Côté serveur Poller

Copie de la clé SSH de votre utilisateur de supervision "**shinken**" depuis le serveur Poller "**shinken-poller**" (pour cet exemple), vers le serveur supervisé "**linux-1**" (dans cet exemple, IP : 192.168.1.19)

Copie clé SSH via commande ssh-copy-id

Soit via la méthode "automatique" via la commande ssh-copy-id en se connectant au préalable via l'utilisateur shinken sur le ou les serveurs pollers :

```
[root@shinken-poller ~]# su - shinken
[shinken@shinken-poller ~]# ssh-copy-id -i ~/.ssh/id_rsa.pub user-service-shinken@linux-1
The authenticity of host '192.168.1.19 (192.168.1.19)' can't be established.
RSA key fingerprint is 00:ff:ee:dd:cc:bb:aa:d6:d3:79:1d:f6:93:47:80:27.
Are you sure you want to continue connecting (yes/no)? yes
user-service-shinken@linux-1's password: XXXXXXXXXXXX
Now try logging into the machine, with "ssh 'user-service-shinken@linux-1'", and check in:
 .ssh/authorized_keys
to make sure we haven't added extra keys that you weren't expecting.
```

Copie clé SSH via commande ssh

Soit via une commande SSH depuis le serveur Poller, il s'agit d'ajouter la clé publique au fichier "authorized_keys" du serveur supervisé (ici linux-1) :

```
cat /var/lib/shinken/.ssh/id_rsa.pub | ssh root@linux-1 "cat >> /home/user-service-shinken/.ssh/authorized_keys"
```

Ici la connexion se fait via l'utilisateur root du serveur linux-1 (mais vous pouvez utiliser votre propre utilisateur), le but étant de rajouter, en une commande SSH, la clé de l'utilisateur shinken du Poller `/var/lib/shinken/.ssh/id_rsa.pub` à la fin du fichier `/home/user-service-shinken/.ssh/authorized_keys` du serveur supervisé.

Copie clé SSH manuellement

Soit via méthode "manuelle" via rajout de la clé dans le fichier `authorized_keys`

- Récupérez la clé publique de l'utilisateur qui va établir la connexion SSH, et la copier

```
[root@shinken-poller ~]# su - shinken
[-bash-4.1]$ cat .ssh/id_rsa.pub

-> copiez la clé
```

- Connectez-vous sur le serveur linux supervisé avec votre utilisateur de supervision et collez cette clé dans le fichier "authorized_keys" de l'utilisateur de supervision :

```
[root@linux-1 ~]# su - user-service-shinken
[-bash-4.1]$ vi .ssh/authorized_keys

-> collez la clé
```

Test de connexion

Test de connexion au serveur "linux-1" en tant qu'utilisateur "user-service-shinken" via l'utilisateur du Poller (shinken) :

```
[root@shinken-poller ~]# su - shinken
[shinken@shinken-poller ~]# ssh user-service-shinken@linux-1 -i .ssh/id_rsa.pub
```

La connexion doit s'établir avec succès.

Côté interface de configuration

Dans chaque hôte héritant du modèle d'hôte "linux_by_ssh", "linux_by_ssh_advanced" ou "linux_by_ssh_extra",

- vous aurez 4 données concernant la connexion SSH (`SSH_KEY`, `SSH_KEY_PASSPHRASE`, `SSH_PORT`, `SSH_USER`)
- Ces 4 données seront par la suite utilisées par tous les checks.
 - Par défaut, ces données sont configurées pour utiliser des variables globales défini par défaut dans le fichier `/etc/shinken/resource.d/ssh.cfg` (*sur le serveur central hébergeant l'Arbiter*).
 - Si vous souhaitez les changer globalement,
 - vous pouvez modifier le fichier `/etc/shinken/resource.d/ssh.cfg`,
 - ou faire vos propres modèles qui héritent des modèles d'hôtes proposés par ce pack en surchargeant ces 4 valeurs (*ainsi vous aurez vos propres valeurs par défaut*).

Donnée	Description	Valeur par défaut	Valeur par défaut à l'installation de shinken
SSH_KEY	Chemin vers la clé générée sur votre serveur hébergeant le démon Poller	<code>\$\$SSH_KEY\$</code>	<code>~/.ssh/id_rsa</code>
SSH_KEY_PASSPHRASE	Phrase secrète utilisé pour déverrouiller la clé privée de l'utilisateur (si celle-ci est protégée par une passphrase). La clé privée déverrouillée est ensuite utilisée pour authentifier l'utilisateur.	<code>\$\$SSH_KEY_PASSPHRASE\$</code>	"
SSH_PORT	Port de connexion SSH	<code>\$\$SSH_PORT\$</code>	22
SSH_USER	Utilisateur pour la connexion SSH	<code>\$\$SSH_USER\$</code>	shinken

Remarque

- Toutes les valeurs par défaut renvoient à une globale (voir la page [LES VARIABLES \(Remplacement dynamique de contenu - Anciennement les MACROS \)](#)) qui sont modifiables dans le fichier `/etc/shinken/resource.d/ssh.cfg`, attention cependant, la modification dans le fichier direct entraînera une modification sur tous les hôtes utilisant ces globales.
- La modification des valeurs par défaut présentes dans le fichier du serveur (`/etc/shinken/resource.d/ssh.cfg`) nécessite un redémarrage intégral du service shinken (`service shinken restart`).

Par exemple, voici le paramétrage d'une connexion via clé SSH par défaut :

? Unknown Attachment

? Unknown Attachment

Par exemple, voici le paramétrage d'une connexion via Utilisateur/Mot de passe :

? Unknown Attachment

Desinstaller le pack avec "uninstall.sh"

Entrée dans le répertoire extrait de l'archive et lancer le "uninstall.sh"

```
cd PACK__shinken__linux-by-ssh__VX.X.X-RCXXX__Linux
./uninstall.sh
```

Ce mécanisme va seulement enlever les fichiers mis en place lors de l'installation (*Fichier du collecteur d'import, définition des éléments de supervision, et les sondes proposées*), mais ne touchent pas aux éléments que vous avez absorbés dans le **Synchronizer**.

```
=====
DELETE PACK [ shinken-linux_by_ssh ]
=====

-----
|   COLLECTOR [ shinken-additional-packs-import ]
|
|   => Deleting collector configuration file
|       - Path : /etc/shinken/sources/shinken-additional-packs-import.cfg
|   => Deleting collector data folder
|       - Path : /etc/shinken-user/source-data/source-data-shinken-additional-packs-import
|
-----
|   MONITORING DEFINITION:
|
|   => Deleting pack globaldata folder
|       - Path : /etc/shinken/resource.d/shinken-additional-packs-import/
|
-----
|   PROBES:
|
|   => Deleting probes
|       - Path : /var/lib/shinken-user/libexec/shinken-additional-packs/shinken-linux_by_ssh
```



À noter : Vous devrez retirer les éléments importés dans le **Synchronizer** (modèles d'hôtes, checks, ...) via les actions de masses.

Mise à jour d'OpenSSH

Ce script peut ne pas fonctionner correctement avec les versions d'OpenSSH antérieure à la 6, dû à une impossibilité de modifier les droits des fichiers et donc de faire fonctionner le script hors root lors des accès à la commande "lastb" à distance.

- Nous vous conseillons donc de mettre à jour votre version d'OpenSSH, ce qui garantira également une meilleure sécurité sur votre environnement.
- Attention, par précaution, assurez-vous d'avoir une session console au serveur sur lequel vous souhaitez réaliser la mise à jour.
 - Si vous rencontrez des soucis dans la mise à jour, avec cette console, vous aurez la possibilité d'intervenir sur le serveur.

En général

Sur la plupart des distributions encore à jour, les versions d'OpenSSH 6 ou supérieures se trouvent déjà dans les dépôts officiels, il vous suffit donc de réaliser votre commande de mise à jour, quelques exemples :

Centos 7 et Redhat 7, 8

```
yum update openssh
```

AlmaLinux 8

```
dnf update openssh
```

RockyLinux 8

```
dnf update openssh
```

Debian et Ubuntu

```
apt-get upgrade openssh
```

ArchLinux et autres

```
pacman -Syu openssh
```

Pré-requis pour certains checks

Certains checks requièrent un accès spécifique à des fichiers.

- Pour ce faire, nous vous mettons à disposition une série de commandes.
 - Ces commandes permettront au groupe de l'utilisateur choisi pour votre supervision Shinken d'avoir un accès (*en lecture seule*) au fichier **/var/log/btmp** (*pour le check Connections Failed SSH*) et au fichier **/etc/ssh/sshd_config** (*pour le check Security SSH*), fichiers comportant vos logs de connexions échouées et votre configuration SSH.
- Sans cet accès, les sondes ne fonctionneront pas et vous renverront le statut **INCONNU**.

Explication

1. La commande modifie le fichier **/usr/lib/tmpfiles.d/var.conf** qui est chargé de rétablir les droits au redémarrage de la machine.
 - Ce fichier n'est pas disponible sur toutes les distributions Linux, vous pourrez alors avoir une erreur, "no such file or directory", cela n'affecte en rien l'application de la commande.
2. Ensuite, le fichier **/etc/logrotate.conf** sera modifié de la même façon pour qu'à la rotation des logs (*tous les mois par défaut*) les droits ne soient pas rétablis.
3. Pour finir, nous changeons donc les droits des fichiers **/var/log/btmp** et **/etc/ssh/sshd_config** pour permettre au groupe utilisé pour la supervision (*et donc son utilisateur*) de les lire.

Exécution de la commande



Remarque

Cette série de commandes ne peut être effectuée qu'en ayant les droits root.

Donc en étant connecté au compte root ou en ayant fait la commande "su" au préalable.

Pour donner un accès en lecture seule au fichier **/var/log/btmp** et au fichier **/etc/ssh/sshd_config** au groupe **shinken**, exécutez :

Utilisation

```
sed -i -e "s/btmp 0600/btmp 0640/g" /usr/lib/tmpfiles.d/var.conf
sed -i -e "s/create 0600/create 0640/g" /etc/logrotate.conf
chmod 640 /var/log/btmp /etc/ssh/ssh_config
chown root:user-service-shinken /etc/ssh/ssh_config
usermod -aG utmp user-service-shinken
```