Configuration du linux supervisé par le pack linux-by-SSH__shinken

Sommaire

Contexte

Cette page a pour but de décrire la mise en place d'une configuration minimale nécessaire pour un Linux supervisé par le pack **linux-by-SSH__shinken**

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 ~]# useradd -m -r -s /bin/bash user-service-shinken
[FACULTATIF] : [root@linux-1 ~]# passwd user-service-shinken
```

- '-m' : Permet la création d'un répertoire /home/user-service-shinken.
  Necessaire afin que le fichier '/home/user-service-shinken/.ssh/authorized_keys' existe et autorise les connexions.
- '-r' : Spécifie que l'utilisateur créé sera un compte de système.
  Cela n'a pas d'influence directe sur le comportement du pack, mais permet de catégoriser le compte créé.
- '-s /bin/bash' : Spécifie le shell à exécuter par défaut lorsque l'utilisateur se connecte. Notamment lors d'une connexion SSH.
  Bash est choisi comme shell par défaut pour s'assurer d'une exécution standard des checks du pack.

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

Par défaut, l'utilisateur "**shinken**" sur le serveur Poller possède une clef RSA : "**/var/lib/shinken/.ssh/id_rsa**".

Si vous générez de nouvelles clefs SSH, voici la liste non exhaustive des clefs supportées ou non :

Clefs supportées

Les clefs SSH suivantes sont supportées :

- ssh-rsa
- sh-ed25519
- ecdsa-sha2-nistp521
- ecdsa-sha2-nistp384
- ecdsa-sha2-nistp25

⚠ Les ssh-dss (DSA) ne sont pas supportés par la sonde pour des raisons de sécurités. Elles sont dépréciées par OpenSSH depuis Janvier 2024, et ne sont officiellement plus supportés depuis Janvier 2025. Article sur le sujet : https://lwn.net/Articles/958048/

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**" pour l'utilisateur "user-service-shinken" (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: XXXXXXXXXX
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 -i .ssh/id_rsa user-service-shinken@linux-1
```

La connexion doit s'établir avec succès.

Problème de connexion : regarder dans les logs

Si la connexion échoue, les logs du service "**sshd**" peuvent donner des indications précieuses sur la cause de l'échec. La méthode de consultation de ces informations dépend de la distribution utilisée et de son ancienneté.

Ci-dessous, la liste non exhaustive des méthodes connues pour consulter les logs, sur les distributions officiellement supportées par Shinken :

Pour Redhat / Almalinux / Rockylinux / Centos :

- la commande journactl permet d'afficher les logs du service :

  ```
  journalctl -xf -u sshd
  ```

- Le fichier de log "secure" contient les informations concernant les dernières connexions :

  ```
  cat /var/log/secure
  ```

Pour Debian :

- la commande journactl permet d'afficher les logs du service (le nom de service utilisé n'est pas le même que pour les autres distributions) :

  ```
  journalctl -xf -u ssh
  ```

Ces logs permettent, entre autres, de savoir :

- Si le problème de connexion provient du client ou du serveur.
- Si les algorithmes de chiffrement utilisés sont cohérents entre les serveurs et permettent l'établissement de la connexion.
- Si les droits attribués aux répertoires contenant les clés d'authentification sont les bons.

Exemple :

Dans l'exemple ci-dessous, les droits n'ont pas été correctement appliqués au répertoire de Shinken contenant les clés SSH. La connexion n'a pas pu s'établir

```
[root@alpachouette-shinken01 shinken]# journalctl -xf -u sshd
-- Logs begin at Wed 2025-09-03 14:44:35 CEST. --
Sep 04 15:32:21 shinken-server sshd[1372868]: Authentication refused: bad ownership or modes for directory
/var/lib/shinken/.ssh
Sep 04 15:32:21 shinken-server sshd[1372868]: Connection closed by authenticating user shinken 172.17.0.37
port 48152 [preauth]
Sep 04 15:32:25 shinken-server sshd[1372968]: Authentication refused: bad ownership or modes for directory
/var/lib/shinken/.ssh
Sep 04 15:32:25 shinken-server sshd[1372968]: Connection closed by authenticating user shinken 172.17.0.37
port 48160 [preauth]
Sep 04 15:32:25 shinken-server sshd[1372971]: Authentication refused: bad ownership or modes for directory
/var/lib/shinken/.ssh
```

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*).

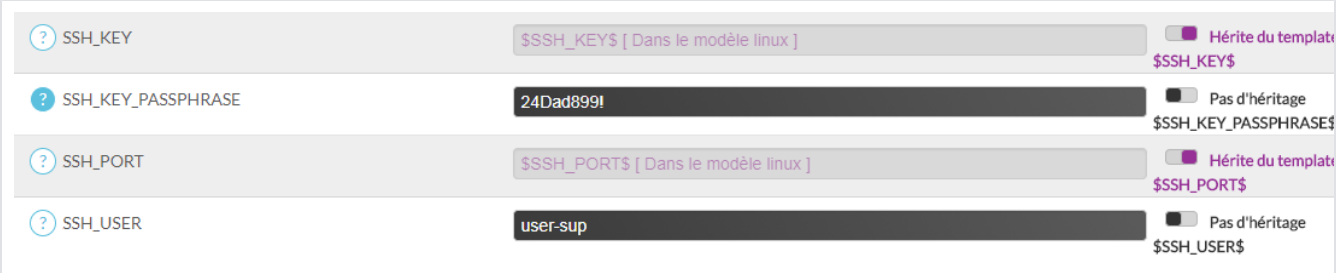
Donnée	Description	Valeur par défaut	Valeur par défaut à l'installation de shinken
SSH_KEY	Chemin vers la clé générée sur votre serveur hébergeant le démon Poller	SSH_KEY	~/.ssh/id_rsa
SSH_KEY_PASSPHRASE	Phrase secrète utilisé pour déverrouiller la clé privée de l'utilisateur (si celle-ci est protégée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

```
service-shinken restart
```

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



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** (*exemple : user-service-shinken*) pour votre supervision Shinken d'avoir un accès :
  - (*en lecture seule*) au fichier **/var/log/btmp** (*pour le check Connections Failed SSH*)
  - (*en lecture seule*) au fichier **/etc/ssh/sshd_config** (*pour le check Security SSH*), fichiers comportant vos logs de connexions échouées et votre configuration SSH.
  - (*en lecture, écriture, exécution*) au dossier **/tmp/shinken** (*pour les checks Stats Disk SSH, Stats Kernel SSH, Stats Net SSH, Stats NFS SSH, Security SSH*)
- Sans ces accès, ces checks ne fonctionneront pas et vous renverront le statut **INCONNU**.

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.

Il faut aussi vérifier que le groupe *'user-service-shinken'* existe avec la commande :

```
getent group user-service-shinken
```

Si vide, créer le groupe et y ajouter l'utilisateur :

```
groupadd user-service-shinken
usermod -aG user-service-shinken user-service-shinken
```

Check Connection Failed by SSH

Commandes à éxecuter :

```
sed -i -e "s/btmp 0600/btmp 0640/g" /usr/lib/tmpfiles.d/var.conf
chmod 640 /var/log/btmp
usermod -aG utmp user-service-shinken
```

1. La commande **sed -i -e "s/btmp 0600/btmp 0640/g" /usr/lib/tmpfiles.d/var.conf** modifie les droits par défaut du fichier **/var/log/btmp** dans le fichier de configuration des fichiers temporaires.

- Cette modification garantit que, même après un redémarrage, les permissions de **btmp** resteront correctes (*lecture pour le groupe*).
- **Note :** Si le fichier **/usr/lib/tmpfiles.d/var.conf** n'existe pas sur votre système, une erreur "no such file or directory" peut apparaître. Cela n'affecte en rien l'application de la commande.

2. La commande **chmod 640 /var/log/btmp** applique immédiatement les droits nécessaires sur le fichier.

- Les utilisateurs du groupe pourront lire les journaux des tentatives de connexion échouées.

3. La commande **usermod -aG utmp user-service-shinken** ajoute l'utilisateur **user-service-shinken** au groupe **utmp**, qui a la responsabilité des logs système.

- Cela permet à l'utilisateur de supervision de lire le fichier **/var/log/btmp**.

Check Security by SSH

Commandes à éxecuter :

```
sed -i -e "s/create 0600/create 0640/g" /etc/logrotate.conf
chmod 640 /etc/ssh/sshd_config
chown root:user-service-shinken /etc/ssh/sshd_config
```

1. La commande **sed -i -e "s/create 0600/create 0640/g" /etc/logrotate.conf** modifie les droits par défaut dans le fichier de configuration de **logrotate.**

   - Cela garantit que, lors de la rotation des fichiers logs (*par défaut, chaque mois*), les permissions de lecture sur **/etc/ssh/sshd_config** pour le groupe ne seront pas rétablies à des niveaux plus restrictifs.

2. La commande **chmod 640 /etc/ssh/sshd_config** applique immédiatement les droits nécessaires.

   - Le fichier de configuration SSH devient lisible par le groupe.

3. La commande **chown root:user-service-shinken /etc/ssh/sshd_config** modifie le groupe du fichier.

   - Le propriétaire reste **root**, mais le groupe est désormais **user-service-shinken**. Cela permet à l'utilisateur de supervision d'accéder au fichier en lecture seule.

Stats Disk by SSH, Stats Kernel by SSH, Stats NET by SSH, Stats NFS by SSH, Security by SSH

Commandes à éxecuter :

```
shinken_tmp_dirname="shinken"
mkdir --parents /tmp/$shinken_tmp_dirname
chown root:user-service-shinken /tmp/$shinken_tmp_dirname
chmod g+rwx /tmp/$shinken_tmp_dirname
```

1. La commande **mkdir --parents /tmp/$shinken_tmp_dirname** crée un récursivement un répertoire.

   - Le répertoire créé est **/tmp/shinken**.
   - Si vous voulez changer le dossier de stockage des fichiers temporaire, modifiez la première ligne : **shinken_tmp_dirname="NouveauDossier"** ainsi que la donnée **SHINKEN_TMP_DIRNAME** attaché au modèle d'hôte.

2. La commande **chown root:user-service-shinken /tmp/shinken** modifie le groupe du dossier **/tmp/shinken**.

   - Cela garantit que des droits peuvent être appliqués au groupe shinken sur ce dossier.

3. La commande **chmod g+rwx /tmp/shinken** applique immédiatement les droits nécessaires au dossier **/tmp/shinken** pour le groupe **user-service-shinken**.

   - Les droits de lecture, d'écriture et d'exécution sont ajoutés au dossier. Cela permet aux sondes de créer et lire des fichiers dans le dossier **/tmp/shinken**.