Network Interfaces SNMP (windows)

Sommaire

Si vous êtes intéressé par ce pack, veuillez nous contacter pour son téléchargement. Vous pourrez ensuite suivre ce tutoriel pour l'installation de votre pack. Si vous rencontrez des problèmes, nous vous accompagnerons lors de l'installation de ce pack sur votre plateforme.

Contexte

Cette page a pour but de vous guider pas à pas dans l'installation du pack linux_by_ssh.

Elle vous accompagnera dans le déploiement du pack sur votre plateforme, dans la configuration de vos connexions SSH, dans la mise à jour du package d'OpenSSH et dans la mise en place de pré-requis pour certains checks.

Procédure de mise en place du 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. Pour supprimer ces dossiers utilisez la commande suivante :

```
rm -rf /var/lib/shinken-user/libexec/linux_by_ssh; rm -rf /etc/shinken/packs/linux_by_ssh/
```

Une fois que vous avez nettoyé les dossiers qui contenaient l'ancien code du pack et que vous avez téléchargé le nouveau pack au format .tar.gz, vous devez le transférer sur votre machine et vous rendre dans le répertoire où le pack est situé.

Puis décompressez le fichier en utilisant commande et en remplaçant **"nom_du_pack_version.tar.gz"** par le nom du pack qui vous est livré :

```
tar xvzf nom_du_pack_version.tar.gz -C ./
```

Voici un exemple plus concret :

```
tar xvzf linux_by_ssh_V02.00-RC001.tar.gz -C ./
```

Maintenant que le pack est décompressé est accessible sur votre machine utilisez les commandes suivantes :

```
mkdir /etc/shinken/packs/linux_by_ssh; cp -r ./linux_by_ssh/configuration/* /etc/shinken/packs/linux_by_ssh/
chown -R shinken:shinken /etc/shinken/packs/linux_by_ssh/
mkdir /var/lib/shinken-user/libexec/linux_by_ssh; cp -r ./linux_by_ssh/libexec/* /var/lib/shinken-user
/libexec/linux_by_ssh/
```

Une fois ces commandes exécutées, vous n'avez plus qu'à importer les nouveaux éléments depuis votre interface de configuration.

Vous pouvez également supprimer le dossier qui a été créer 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.

Import des modifications suite à une mise à jour de Shinken

Après avoir effectué une mise à jour de Shinken et en cas de modification du pack linux_by_ssh, les mises à jour seront mises à votre disposition sous la forme d'un fichier tar.gz,.

Vous devrez suivre les mêmes étapes que lors de la mise en place de ce pack.

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 commandes sont réalisées par les Pollers, en tant qu'utilisateur "**shinken**". Comme pour tous les serveurs hébergeant Shinken, cet utilisateur est un utilisateur sans mot de passe par défaut. (les connexions SSH vers les serveurs Shinken via cet utilisateur ne sont donc possibles qu'avec une clé SSH)

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.

Remarque

Si vous utilisez le pack linux_by_ssh pour superviser vos serveurs hébergeant Shinken, vous pouvez utiliser l'utilisateur déjà créé et utilisé par Shinken Entreprise : **shinken.**

Si vous choisissez cet utilisateur, vous n'aurez pas besoin de données particulières pour vos modèles d'hôte "linux_by_ssh", "linux_by_ssh_advanced" et "linux_by_ssh_extra" car les valeurs par défaut à l'installation de shinken suffiront (voir le tableau de données plus bas dans cette page).

Il faudra par contre réaliser les autorisations manuelles via clé SSH.

Cas Particulier

Par défaut, vos serveurs Shinken autorisent les connexions SSH émises par l'utilisateur **shinken** de leur propre serveur.

Donc dans le cas d'une installation rapide, le Poller pourra exécuter avec succès les requêtes SSH envoyées sur lui même, sans que vous ne fassiez de manipulations avec les clés.

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 user-service-shinken mais vous pouvez créer un autre utilisateur)

```
              [root@linux-1 ~]# adduser -m -r user-service-shinken
[FACULTATIF] : [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 "**user-service-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 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: XXXXXXXXXXX
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 vm2) :

```
cat /var/lib/shinken/.ssh/id_rsa.pub | ssh root@vm2 "cat >> /var/lib/shinken/.ssh/authorized_keys"
```

Ici la connexion se fait via l'utilisateur root du serveur vm2 (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 **/var/lib/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]$ less .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 "**remote-host**" 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
```

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, ces 4 données seront par la suite utilisées par tous les checks.

Contrairement aux autres données, les valeurs par défaut de celles-ci sont configurées dans un certain fichier en central (serveur hébergeant l'Arbiter) /etc/shinken/resource.d/ssh.cfg.

Donnée	Description	Valeur par défaut	Valeur par défaut à l'installation de shinken
SSH_KEY	Répertoire de la clé générée sur votre serveur hébergeant le démon Poller	SSH_KEY	~/.ssh/id_rsa
SSH_KEY_PASSPHRASE	Mot de passe utilisé pour l'authentification de l'utilisateur ou pour utiliser la clé privée ("Passphrase") si nécessaire	$SSH_KEY_PASSPHRASE$	''
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 intégrale 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

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.

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 :

Note

Les commandes peuvent s'étendre à d'autres distributions non répertoriées

Centos 7 et Redhat

```
yum update openssh
```

Debian et Ubuntu

```
apt-get upgrade openssh
```

ArchLinux et autres

```
pacman -Syu openssh
```

Sur Centos 6.6

Voici les différentes commandes : (un exemple ici avec la version OpenSSH 7.6 Officielle, mais vous pouvez prendre la dernière version disponible sur le site officiel)

Installation de quelques paquets

```
yum install rpm-build gcc make wget openssl-devel krb5-devel pam-devel libX11-devel xmkmf libXt-devel
```

Téléchargement

```
wget https://mirrors.ircam.fr/pub/OpenBSD/OpenSSH/portable/openssh-7.6p1.tar.gz
```

Extractions et copies

```
tar xvf openssh-7.6p1.tar.gz
mkdir -p /root/rpmbuild/{SOURCES,SPECS}
cp ./openssh-7.6p1/contrib/redhat/openssh.spec /root/rpmbuild/SPECS/
cp openssh-7.6p1.tar.gz /root/rpmbuild/SOURCES/
```

Paramétrage Specs

```
cd /root/rpmbuild/SPECS/
sed -i -e "s/%define no_gnome_askpass 0/%define no_gnome_askpass 1/g" openssh.spec
sed -i -e "s/%define no_x11_askpass 0/%define no_x11_askpass 1/g" openssh.spec
sed -i -e "s/BuildPreReq/BuildRequires/g" openssh.spec
```

Build du RPM et installation

```
rpmbuild -bb openssh.spec

cd /root/rpmbuild/RPMS/x86_64/

rpm -Uvh *.rpm
```

Redémarrez votre service sshd :

```
service sshd restart
```

Vous pouvez vérifier votre version avec :

```
rpm -qa | grep openssh
```

Si vous avez quelconques problèmes vous pouvez revenir sur l'ancienne version avec :

```
yum downgrade openssh-server
```

Si vous utilisez l'utilisateur "shinken" (par défaut) avec une connexion via clé RSA, il se peut que suite à la mise à jour, vos scripts affichent un message de problème d'authentification (**[ERROR]** Connection failed 'Authentification failed'), sur le serveur sur lequel vous avez mis à jour SSH, veuillez réinitialiser l'utilisateur en lui supprimant son mot de passe (par défaut) avec la commande :

passwd -d shinken

Pré-requis pour certains checks

Certains checks requièrent un accès spécifique à des fichiers. Pour se faire une commande est à votre disposition. Cette commande permettra 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 "Unknown".

> **Remarque**
>
> Cette commande 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.

Fonctionnement

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.

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.

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

Pour donner un accès en lecture seule au fichier /var/log/btmp et au fichier /etc/ssh/sshd_config au groupe **shinken**, en root depuis le serveur à superviser, exécutez :

Utilisation

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