shinken-protected-fields-encryption-enable

Contexte

Lorsque vous installez Shinken enterprise, un certain nombre de modèles et de commandes sont inclus dans votre configuration.

Le pack "snmp_checks", comme son nom l'indique, permet de superviser des hôtes sur lesquels est installé un système d'exploitation basé sur Linux ou Windows (serveur ou client) via le **protocole SNMP**.

Il contient 17 commandes, 17 modèles de checks dédiés à 2 modèles d'hôte spécifiques (nommés "linux" et "linux-advanced").

Toutes les commandes de ce pack se basent sur des scripts présents dans le répertoire des scripts shinken **/var/lib/shinken/libexec** (ou **$PLUGINSDIR$** depuis l'interface de configuration).

Le protocole SSH (Secure Shell) est utilisé par chacun des 15 scripts du pack linux. Les scripts communiqueront avec votre machine directement par un invite de commande après s'être connecté avec les identifiants SSH que vous aurez paramétré.

Nous allons ici détailler ces checks associés au modèle de ce pack.

Sommaire

Sommaire des checks

2 modèles d'hôtes sont inclus à ce pack, le modèle **linux_by_snmp** (pour les OS Linux) et le modèle **windows_by_snmp** (pour les OS Windows).

Modèle linux_by_snmp

Check Name	Description
CPU	Récupère et vérifie le Load Average du CPU
Disks	Récupère et vérifie les informations de taille des disques
Memory	Récupère et vérifie les informations concernant la RAM
Process	Récupère et vérifie les informations concernant les processus du système

Modèle windows_by_snmp

Check Name	Description
CPU	Récupère et vérifie le Load Average du CPU
Disks	Récupère et vérifie les informations de taille des disques
Memory	Récupère et vérifie les informations concernant la RAM
Process	Récupère et vérifie les informations concernant les processus du système
Windows Services	Récupère et vérifie les informations concernant les services Windows du système

Les modèles d'hôtes et leurs données héritées

Les modèles d'hôtes **windows_by_snmp** et **linux_by_snmp**, sur lesquels sont accrochés les différents checks dédiés, contiennent des données (locales) qui seront utilisés par les checks. Ces données seront invoquées par les checks et commandes via **$_HOST** suivi du nom de la variable.

Exemple : **$_HOSTSNMP_PROCESS$** utilisera la donnée nommée **SNMP_PROCESS** (quelle soit locale ou héritée d'un modèle).

Pour un hôte qui hérite par exemple du modèle windows_by_snmp ou linux_by_snmp de notre pack, ces données seront donc héritées également, mais elles pourront aussi être surchargées directement sur l'hôte (attention aux conflits de nom des données).

Si vous souhaitez modifier de manière globale ces données, ou en rajouter, faites le directement sur le modèle voulu, ceci s'appliquera alors à tous vos hôtes utilisant ce modèle.

Pour plus d'information, veuillez consulter la page sur les Remplacement dynamique de contenu.

? Unknown Attachment

? Unknown Attachment

Comment utiliser le pack snmp_checks

Le pack **snmp_checks** peut être utilisé en appliquant le modèle souhaité à un hôte. Il existe deux manières de procéder :

En utilisant l'interface de Configuration

Dans l'interface de Configuration, créez ou éditez un Hôte, et ajoutez le modèle **windows_by_snmp** ou **linux_by_snmp** grâce au menu déroulant.

En éditant les fichiers de configuration

Dans un fichier de configuration, créez ou éditez votre définition d'hôte en ajoutant, dans le propriété "**use",** la valeur "**windows_by_snmp "** ou **"linux _by_snmp"** selon les besoins.

Le fichier de configuration devra alors être importé avec une source (plus d'information sur cette page: Import de fichier de configuration).

Configuration de la connexion SNMP

Pour l'exécution correcte des commandes, vous aurez besoin du service SNMP sur l'hôte supervisé.

Côté client (machine ou serveur supervisé)

Linux

Sur votre serveur supervisé avec l'OS Linux, il vous faut installer les paquets **net-snmp** et **net-snmp-utils** :

```
yum -y install net-snmp net-snmp-utils
```

Ensuite, par précaution, faîtes une copie puis éditez le fichier de configuration de snmpd :

```
cp /etc/snmp/snmpd.conf /etc/snmp/snmpd.conf.bak
vim /etc/snmp/snmp.conf
```

La ligne suivante permet de changer la communauté *public* vers une communauté propre à votre réseau et plus ou moins complexe (remplacez *public* par la chaîne de caractères que vous souhaitez):

```
####
# First, map the community name "public" into a "security name"

#       sec.name  source          community
com2sec notConfigUser  default       public
```

Par défaut, le fichier de configuration associe ensuite (étape 3 dans le fichier) le nom de sécurité ("*notConfigUser"*) à une vue d'accès restreintes à certains OID ("*systemview"*).
Pour un accès sur l'ensemble des OIDs du système, utilisez une nouvelle vue "*all"*:

```
####
# Third, create a view for us to let the group have rights to:
# Make at least  snmpwalk -v 1 localhost -c public system fast again.
#       name           incl/excl     subtree         mask(optional)
#view    systemview    included   .1.3.6.1.2.1.1
#view    systemview    included   .1.3.6.1.2.1.25.1.1
view    all    included   .1
```

Et par conséquent, remplacez la vue "*systemview"* par "*all"* dans la dernière étape de configuration du fichier :

```
#       group          context sec.model sec.level prefix read   write  notif
access  notConfigGroup ""      any       noauth    exact  all none none
```

Vous pouvez maintenant démarrer le démon SNMPD :

```
service snmpd start
```

Pensez à redémarrer le service snmpd à chaque modification du fichier de configuration snmpd.conf.

Pour un démarrage du service snmpd à chaque démarrage de votre machine, utilisez la commande :

```
chkconfig snmpd on
```

Vous pouvez tester votre service snmpd avec la commande snmpwalk (changez la communauté si besoin) :

```
snmpwalk -v 1 -c public localhost
```

Windows

Côté serveur Poller

Les scripts sont exécutés par le ou les serveurs Poller. Les commandes sont basées sur des scripts PERL.

Les prérequis sont déjà installés avec Shinken Enterprise, donc aucune action n'est censée être requise.

Pour information, les librairies Perl ainsi que net-snmp-utils et net-snmp-libs sont nécessaires pour le bon fonctionnement.