Contexte

Simple Network Management Protocol (abrégé **SNMP**), est un protocole de communication qui permet aux administrateurs réseau de gérer les équipements du réseau, de superviser et de diagnostiquer des problèmes réseaux et matériels à distance.

Une requête SNMP est un datagramme UDP envoyée par le manager à destination du port 161 de l'agent. L'agent répond alors au manager avec la valeur demandée.

Les traps SNMP, elles, sont émises depuis les agents SNMP vers une destination (un serveur de supervision par exemple), qui entendra ces requêtes. Pour les comprendre, ce serveur devra disposer de bases de données avec l'ensemble des informations (OID et Descripteurs) des constructeurs, ces bases sont appelées MIB. Les valeurs pourront alors être interprétées par le serveur de supervision. Ce procédé est souvent utilisé dans les routeurs pour par exemple, avertir qu'un lien vient de tomber sur l'une de ses interfaces. L'intérêt des traps SNMP est donc d'envoyer des « alertes » dès qu'une panne apparaît sans attendre que le serveur de supervision le détecte de lui même pendant une vérification dans le cadre d'un monitoring actif.

Il se peut donc que vous souhaitiez paramétrer Shinken pour récupérer et interpréter ces traps, nous vous proposons deux moyens, via le module WS Arbiter de Shinken, ou via le module Named Pipe (aussi utilisé de manière historique dans Nagios avec le fichier nagios.cmd).

WS Arbiter

En cours de Documentation...

Module Named Pipe

Le module Named Pipe, via le fichier "passe plat" FIFO nagios.cmd, va permettre à Shinken de récupérer les entrées ou commandes externes.

Mise en place du module

- S'il n'existe pas, copier le répertoire named-pipe (dans le zip ici) dans le répertoire /var/lib/shinken/modules
- Il faut ensuite rajouter la définition du module :
  - Dans le répertoire /etc/shinken/modules/
  - Copier le fichier name-pipe.cfg
- Ensuite on va accrocher le module au receiver
  - Ouvrir le fichier /etc/shinken/receivers/receiver-master.cfg
  - Ajouter named-pipe à la ligne modules
    - *modules named-pipe*
- Redémarrer Shinken via la commande *service shinken restart*

Création de l'élément de supervision en configuration

- Dans l'interface de configuration, il faut à présent créer le modèle de check, en passif et volatile, car si nous recevons une deuxième interruption pour le même hôte avant que la première ait été remise à OK, nous voulons être notifié à nouveau.
  - Voici un exemple avec la syntaxe via un fichier cfg :

```
define service{
        name                    snmptrap-service
        service_description     TRAP
        check_command           check_host_alive
        is_volatile             1
        max_check_attempts      1
        check_interval          1
        retry_interval          1
        passive_checks_enabled  1
        check_period            24x7
        notification_period     24x7
        notification_interval   86400
        notification_options    w,c,r
        contact_groups          admins
        register                0
}
```

- Il faut ensuite rattacher le service à un hôte, voici un exemple via fichier cfg :

```
define service{
        use                     snmptrap-service
        host_name               localhost
        service_description     TRAP
}
```

- Une fois les éléments passés en production, le check doit donc apparaître dans l'interface de visualisation comme un ping du fait de la check_command saisie dans la configuration. C'est elle qui va s'occuper de refaire passer le service en « Ok » après une réception de trap qui l'aurait fait passer en « Critical ».

Script interpréteur des traps

- Maintenant il faut un script (plugin) qui va se charger d'interpréter les futures traps SNMP reçues pour les envoyer à Shinken (à travers le module named-pipe et le fichier nagios.cmd).
- Ajouter le script suivant que l'on appellera submit_check_result dans le dossier des plugins nagios (/usr/lib64/nagios/plugins/):

```
#!/bin/bash

# Arguments:
# $1 = host_name (Short name of host that the service is
# associated with)
# $2 = svc_description (Description of the service)
# $3 = return_code (An integer that determines the state
# of the service check, 0=OK, 1=WARNING, 2=CRITICAL,
# 3=UNKNOWN).
# $4 = plugin_output (A text string that should be used
# as the plugin output for the service check)
#

echocmd="/bin/echo"

CommandFile="/var/lib/shinken/nagios.cmd"

# get the current date/time in seconds since UNIX epoch
datetime=`date +%s`

# create the command line to add to the command file
cmdline="[$datetime] PROCESS_SERVICE_CHECK_RESULT;$1;$2;$3;$4"

# append the command to the end of the command file
`$echocmd $cmdline >> $CommandFile`
```

- On le rend exécutable et on le donne à l'utilisateur shinken.

- Enfin, tu suis le tutorial https://mespotesgeek.fr/fr/capture-de-trap-snmp-sous-shinken/ à partir de « Ajouter le script suivant que l'on appellera submit_check_res »