

Identification des checks les plus consommateurs à l'aide de la commande shinken-scheduler-export-data

Sommaire

- Économiser des CPUs en identifiant les checks les plus consommateurs
- Création du tableau récapitulatif sur la consommation totale des temps CPU des royaumes respectifs
 - Consolidation des données: utilisation d'un tableau croisé dynamique
 - Création du tableau croisé dynamique
 - Sélection des commandes en tant que lignes de notre tableau
 - Obtenir la consommation CPU totale par check
 - Passer du nombre de lignes avec "cpu_time" à une vraie somme des temps CPU
 - Rendre le titre de la colonne explicite
 - Obtenir la répartition de la consommation CPU par check
 - Passer la seconde colonne en somme de temps, comme la première
 - Passer du nombre total de temps CPU consommé à la répartition de la consommation CPU par check
 - Rendre le titre de la colonne explicite
 - Obtenir la répartition de la consommation CPU moyenne pour une exécution (en s)
 - Passer du nombre total de temps CPU consommé à la consommation CPU moyenne pour une exécution de check
 - Rendre le titre de la colonne explicite
- Analyse des résultats
 - Analyse du cas présent: quel check doit être optimisé en priorité pour économiser du CPU?
 - Quel check est le plus consommateur?
 - Exercice d'exemple: quel check optimiser afin de limiter la consommation CPU à un seul CPU?

Contexte

Pour utiliser la connexion SNMPv1/2, le pack switch vous offrent deux modèles d'hôtes.

- *switch-SNMPv1v2*
- *switch-SNMPv1v2-detailed*

switch-SNMPv1v2	switch-SNMPv1v2-detailed
<ul style="list-style-type: none">• Ces modèles vous offrent une vue d'ensemble pour chaque check sur l'état général de vos interfaces.• Mis à part la mise en place du protocole SNMP, il ne nécessite aucune configuration.• Ces modèles sont donc conseillés si vous voulez une vue sur l'ensemble de vos interfaces en un seul résultat,<ul style="list-style-type: none">◦ Mais attention, si un problème est rencontré sur une seule des interfaces, alors le résultat indiquant l'erreur, sera noyée par la masse d'informations renvoyées par le check◦ De même, si un second problème venait à apparaître, il en serait alors, encore plus difficilement repérable.	<ul style="list-style-type: none">• Ces modèles vous offrent une vue éclatée, c'est-à-dire un résultat interface par interface pour chaque check que vous allez effectuer.• En plus de configurer SNMP, il sera nécessaire de configurer le nom de toutes les interfaces dans l'interface de configuration Shinken.• Ces modèles sont conseillés si vous voulez voir une description interface par interface,<ul style="list-style-type: none">◦ Cela vous demandera une configuration, mais si un problème vient à apparaître, il vous indiquera simplement l'interface qui pose un problème.

Vérifier que la connexion SNMP entre le serveur de supervision et l'hôte supervisé est fonctionnelle avant d'appliquer un modèle d'hôte

Test de connexion

Vous pouvez tester la bonne configuration du service SNMP de votre switch depuis votre serveur Poller en fonction du SNMP utilisé.

En replacer dans la commande ci-dessous :

- le mot *COMMUNAUTE* par la communauté paramétrée sur votre switch,
- et *IP-SWITCH* par l'adresse IP de votre switch.

```
[root@shinken-poller ~]# snmpwalk -v2c -c COMMUNAUTE IP-SWITCH
```

Une liste de valeur doit défiler à l'écran pour valider la bonne connexion, comme l'exemple ci-dessous.

```
$ snmpwalk -v2c -c public 192.168.1.4
SNMPv2-MIB::sysDescr.0 = STRING: Cisco Internetwork Operating System SoftwareIOS (tm) MSFC Software (C6MSFC-
JS-M), Version 12.0(7)XE1, EARLY DEPLOYMENT RELEASE SOFTWARE (fc1)TAC:Home:SW:IOS:Specials for infoCopyright
(c) 1986-2000 by cisco Systems, Inc.Compiled Thu 03-Feb-00 23:
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.1.258
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (22061) 0:03:40.61
SNMPv2-MIB::sysContact.0 = STRING: admin
SNMPv2-MIB::sysName.0 = STRING: CISCOROUTER
SNMPv2-MIB::sysLocation.0 = STRING: server-room
SNMPv2-MIB::sysServices.0 = INTEGER: 78
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
IF-MIB::ifNumber.0 = INTEGER: 13
IF-MIB::ifIndex.2 = INTEGER: 2
...
```