

CPU_Stats SSH

Contexte

Le check CPU Stats SSH va utiliser des commandes à travers votre serveur SSH pour rechercher des informations sur votre CPU, tel que le nombre de cœurs, la fréquence, le pourcentage d'utilisation du processeur et pour finir, si votre utilisation du processeur est trop importante, les X processus utilisant le plus gros du processeur (X étant la donnée CPU_MAX_PROCESS modifiable dans la configuration) avec une notation "étoile" permettant de voir quels processus sont les plus consommateurs.

Le seuil choisi pour la consommation d'un processus sera la valeur du processus le plus consommateur et fera office de base pour trouver quels processus sont les plus consommateurs. En effet nous n'affichons pas le pourcentage de CPU utilisé par chaque processus, car ceux-ci ne représentent que l'utilisation globale à un instant T. Pour essayer de trouver les processus les plus consommateurs, nous essayons alors d'analyser une tendance.

Nous utilisons donc le pourcentage global du CPU consommé par les processus. Nous faisons une moyenne de ce pourcentage selon le nombre de CPU de votre machine pour ensuite réaliser une échelle "d'alarme" montrant intuitivement quels sont les processus à réellement surveiller (Plus d'infos dans l'exemple ci-dessous).



Note : Le check CPU Stats SSH utilise le paquet SysStat. Si votre serveur Linux ne dispose pas de cette librairie, il faudra l'installer (yum install sysstat)

Exemple

Exemple de résultat

[WARNING] CPU usage is 12%.

Top CPU consuming processes :

Processes	CPU usage
python	★★★★★
sshd	★★★★★
python	★★★★★
python	★★★★★
shinken-scheduler	★★★★★
python	★★★★★
python	★★★★★
/usr/bin/mongod	★★★★★
shinken-broker	★★★★★

Number of cores : 4
Frequency : 3292MHz

Cas spécifique

Si le check a le statut "Unknown" :

Installez le paquet sysstat (via la commande yum install sysstat, apt-get install sysstat, pacman -S sysstat ou yaourt -S sysstat, selon votre OS).

Données et métriques

Données

Donnée	Description	Valeur par défaut
CPU_CRIT	Définit le pourcentage d'utilisation du processeur à partir duquel le check passe en critique	90
CPU_WARN	Définit le pourcentage d'utilisation du processeur à partir duquel le check passe en avertissement	80
CPU_MAX_PROCESS	Définit le nombre de processus alarmant à afficher au maximum	10

Métriques

Nom de la métrique	Description
--------------------	-------------

cpu_(numerocpu)_sys	Pourcentage de temps CPU utilisé par le noyau Linux
cpu_(numerocpu)_usr	Pourcentage de temps CPU utilisé par les processus en espace utilisateur
cpu_(numerocpu)_soft	Pourcentage de temps CPU utilisé par les interruptions logicielles
cpu_(numerocpu)_guest	Pourcentage de temps CPU utilisé par les processeurs virtuels
cpu_(numerocpu)_idle	Pourcentage de temps CPU inactif
cpu_(numerocpu)_irq	Pourcentage de temps CPU utilisé par les interruptions matérielles
cpu_(numerocpu)_nice	Pourcentage de temps CPU utilisé par les processus en espace utilisateur dont le nice a été changé
cpu_(numerocpu)_iowait	Pourcentage de temps CPU utilisé pour effectuer des attentes d'entrées/sorties
cpu_(numerocpu)_steal	Pourcentage de temps CPU utilisé par une attente involontaire imposée par un hyperviseur
cpu_(numerocpu)_used	Pourcentage de temps pendant lequel le CPU est utilisé (de n'importe qu'elle manière)

Cas particulier

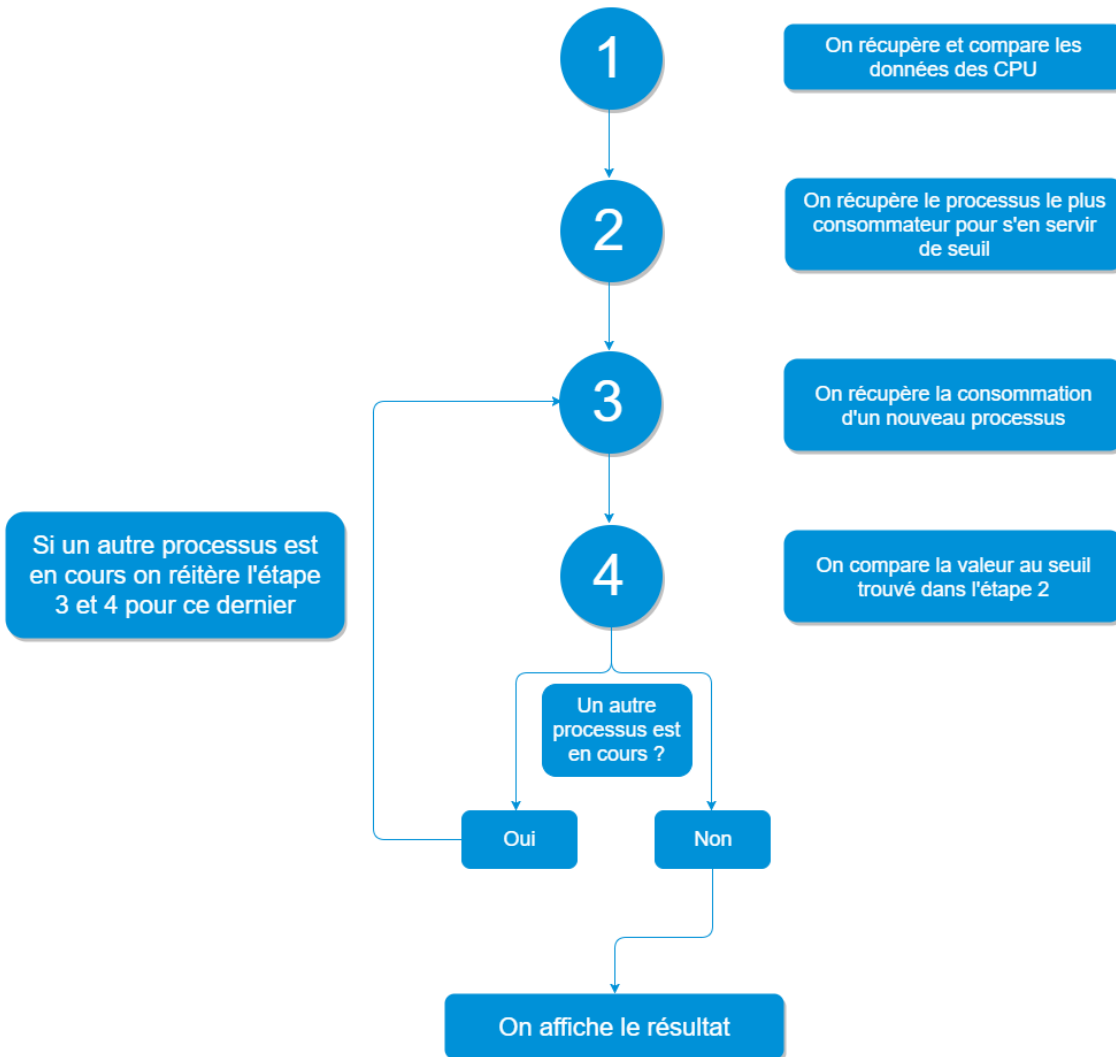
"No processes found"

Il se peut que votre tableau ne vous affiche que le message "No processes found", dans ce cas votre machine est configurée au cas par cas et donc il nous est impossible de trouver les processus utilisés par les autres utilisateurs que celui utilisé pour la supervision.

Veuillez nous contacter en nous détaillant votre configuration.

Détails de l'algorithme

Récapitulatif

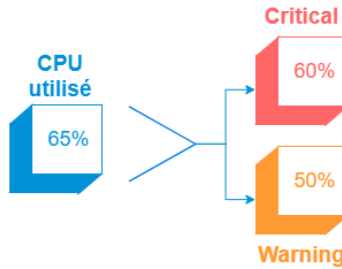


1

Détection et analyse du CPU global

Les outils système de linux nous indique que le CPU est à 65% de sa capacité maximale.

Dans cet exemple nous utiliserons ces données :
- **Warning** = 50%
- **Critical** = 60%
Le serveur servant d'exemple possède 4 CPU



Si le CPU dépasse le seuil **Warning** ou **Critical** nous passons à l'étape 2, sinon le check est terminé. Ici, le check retourne un statut **Critical**, nous passons à l'étape 2.

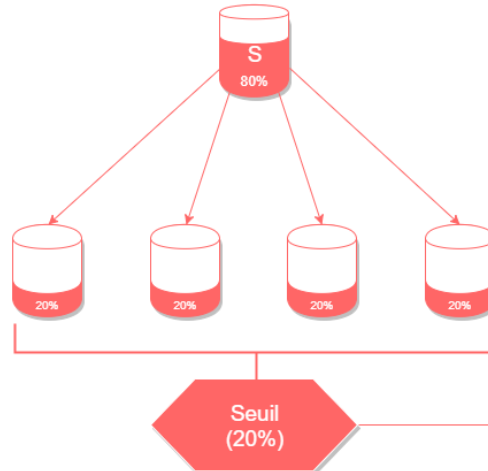
2

Recherche du seuil d'alerte pour les processus

Les outils système linux nous indiquent que le processus S consommant le plus de CPU utilise 80% de celui-ci.

La valeur du seuil équivaut au processus le plus consommateur de CPU.

Un processus sera donc toujours à 5 étoiles, les autres dépendront de leurs différences avec ce premier.



Dans l'optique d'obtenir des résultats les plus fidèles à la réalité possible nous évaluons ce pourcentage sur l'ensemble des CPU disponibles. Nous utiliserons la moyenne de ce processus comme référence "Seuil".

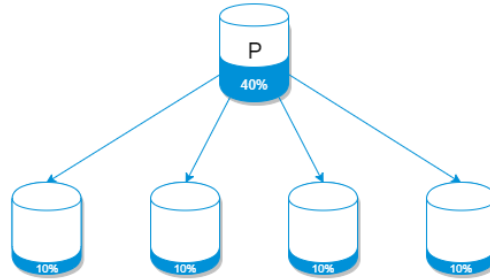
3

Analyse des processus un à un

Maintenant nous nous intéressons aux processus un à un.

Comme dans l'étape 2 nous établissons une moyenne de la valeur système obtenue.

Les outils système linux nous indiquent que le processus P utilise 40% du CPU

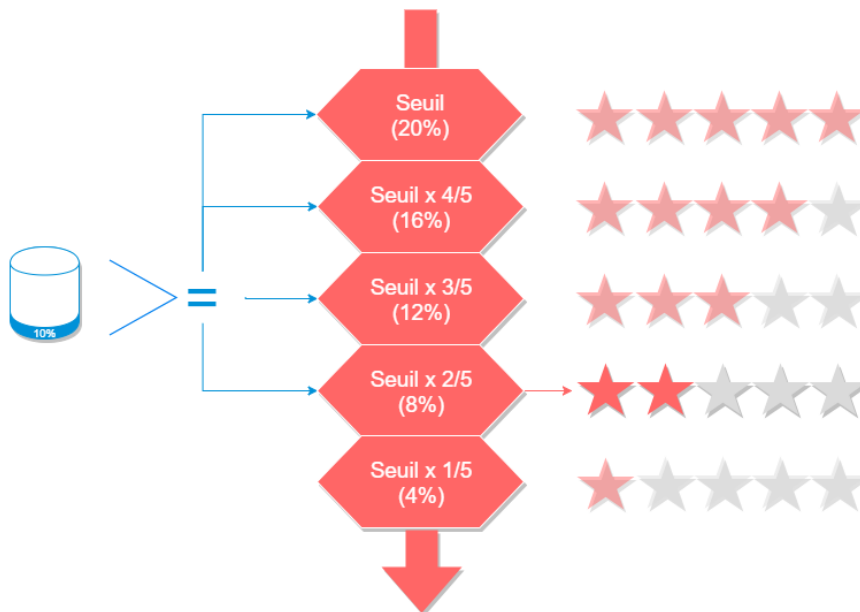


Dans l'optique d'obtenir des résultats les plus fidèles à la réalité possible nous évaluons ce pourcentage sur l'ensemble des CPU disponibles

4

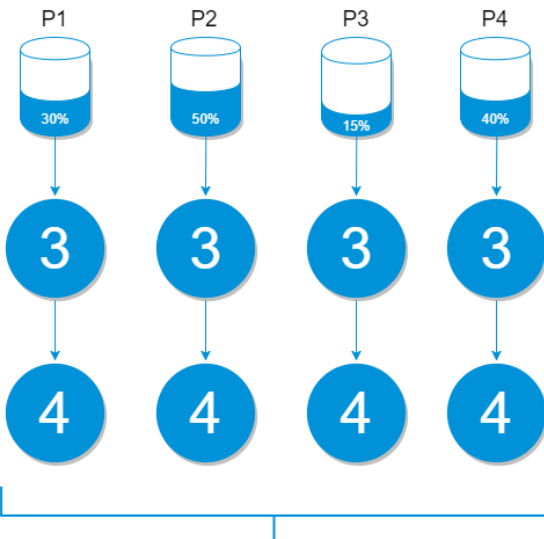
Comparaison des analyses avec le seuil

Pour évaluer la consommation du processus nous comparons la moyenne obtenue précédemment avec une échelle basée sur le seuil S, 1 étoile sur 5 représentera 1/5 du seuil S.



Ici comme $P > (\text{Seuil} \times 2/5)$ nous obtenons 2 étoiles

Exemple global pour un total de 6 processus (comprenant ceux des étapes précédentes)



Réitération des étapes 3 et 4 pour chaque nouveau processus



Le check affichera donc ceci dans un tableau :

Le processus P3 a disparu car il n'a atteint aucun seuil et donc n'est pas alarmant