

# Le mapping des informations collectées des champs du serveur Open LDAP vers les propriétés et les données Shinken

## Sommaire

[Qu'est-ce qu'une règle de mapping OpenLDAP ?](#)  
[Définition d'une règle de mapping utilisateur](#)  
    [Forcer la valeur des propriétés](#)  
    [Liste des clés d'import pouvant être forcées](#)  
    [Chemin du fichier de définition des règles de mapping](#)  
[Exemple](#)  
[Compatibilité avec les anciennes versions](#)

## Économiser des CPUs en identifiant les sondes les plus consommatrices

Une fois que l'on a un parc supervisé, on peut souhaiter de temps en temps procéder à des optimisations sur sa consommation, et tout particulièrement sur sa consommation CPU car elle a un coût non négligeable. Une fois les sondes les plus consommatrices de CPU identifiées, une optimisation de ces dernières sera la méthode la plus efficace afin de consommer moins de CPUs, et possiblement avoir des serveurs plus petits, ou moins de serveurs.

Pour cela, nous allons extraire des informations issues de la commande **shinken-scheduler-export-data** afin d'avoir :

- le nombre de CPUs utilisés par chaque commande sur l'ensemble des royaumes

Pour cela, nous allons avoir besoin d'un dump de données avec en option une prévision de la charge sur une période, disons par exemple 1 heure.

Il suffit alors de lancer la commande comme ceci :

```
shinken-scheduler-export-data --full --simulation-extra-period=3600
```

La simulation sur une heure sert à é



Avec ce lancement, les noms ( *hôte, check, commandes et royaumes* ) seront présents dans l'export.

- Il est tout à fait possible de faire l'analyse sur un export sans les noms en retirant l'option `--full`.
- Cependant pour ne pas avoir à manipuler des uuid lors de la lecture des résultats nous préconisons un export avec les noms.

L'importation du fichier .csv généré est décrit dans [FOR SHINKEN TEAM - shinken-scheduler-export-data : export des données du Scheduler](#)

## Création du tableau récapitulatif sur la consommation totale des temps CPU des royaumes respectifs

### Consolidation des données : utilisation d'un tableau croisé dynamique

A partir des nombreuses données de checks que nous avons, nous allons devoir procéder à une consolidation afin d'avoir notre résultat facilement exploitable. Pour cela nous allons utiliser un tableau croisé dynamique.

- Un tableau croisé dynamique dans un tableur est un outil de analyse de données qui vous permet de créer une vue synthétique et facile à lire d'une grande quantité de données ( *ici nos exécutions de checks* ).
- On y choisit les données à inclure, comment les organiser et comment les synthétiser, puis filtrer, classer et totaliser les données en fonction de nos besoins.

### Création du tableau croisé dynamique

La création du tableau récapitulatif passe par la création d'un Tableau croisé dynamique.

Depuis votre Feuille d'importation des données, il faut cliquer sur **Insertion Tableau croisé dynamique**, et valider :

? Unknown Attachment

## Sélection des commandes en tant que lignes de notre tableau

Arrivé sur la nouvelle feuille, Excel demande quelles lignes sélectionner, dans le bloc de droite.

Dans notre cas, il faut faire glisser le champ **command\_name** (ou bien **command\_name\_anonymous\_hash** si on a une version anonyme de l'export) vers le bloc "Lignes" :

? Unknown Attachment

## Sélection des "Valeurs": le cpu\_time, le temps consommé par les sondes

A chaque commande, il faut lui assigner une ( *ou plusieurs* ) "Valeurs".

Pour cela, on fait glisser le champ "cpu\_time" vers le bloc "Valeurs" afin d'avoir pour chaque royaume son champs **cpu\_time** associé :

? Unknown Attachment

A noter que par défaut Excel prends le nombre d'occurrences du champ **cpu\_time** comme "Valeur", ce qui n'est pas ce qui est souhaité.

## Passer du nombres de lignes avec "cpu\_time" à une vraie somme des temps CPU

Une modification des "Paramètres des champs de valeurs" est nécessaire sur "Nombre de cpu\_time" qu'il faut changer en Somme pour donner au final "Somme de cpu\_time" :

? Unknown Attachment

## Obtenir le % de consommation d'une sonde par rapport à toutes les autres

S'il est possible de trier par rapport au temps absolu consommé par une sonde et donc avoir les plus consommatrices, il peut être également très intéressant d'avoir leur pourcentage par rapport à la consommation totale. Il sera ainsi possible de savoir de quel ordre de grandeur une optimisation sera bénéfique pour la consommation.

Pour cela, il faut rajouter une seconde fois le champ "cpu\_time" dans les valeurs, ce qui rajoute une seconde colonne :

? Unknown Attachment

## Passer la seconde colonne en somme de temps, comme la première

Ensuite il faut passer, comme pour la première colonne, d'un calcul en **Nombre** à un calcul en "Somme de cpu\_time" :

? Unknown Attachment

## Passer du nombre total de temps CPU consommé au nombre de CPU nécessaires

On peut enfin changer l'affichage dans l'onglet "Afficher les valeurs" en choisissant "% du total général" pour obtenir le résultat final :

? Unknown Attachment

Ici nous avons un exemple simple où une sonde consomme à elle toute seule plus de 87% du temps CPU total.

Elle est donc la cible idéale pour une optimisation qui pourrait, dans le cas idéal, diviser par 5 le nombre de CPUs utilisés.