

# Reactionner - \$KEY\$ - Running Well

## Introduction


Cette propriété est utilisé pour générer plusieurs checks dans l'arbitre avec une seule définition de check dans le Synchronizer.

Pour cela, la fonctionnalité "duplicate foreach" lie la donnée d'un hôte ou cluster à un check.

Cette donnée lié a alors un structure particulière.

## Structure de la donnée duplicate Foreach:

- **KEY**: Correspond à la clé, appelé \$KEY\$.
- **VALUE**: Correspond à un paramètre, appelé \$VALUE\$ si il y en a qu'un ou \$VALUEx\$, avec **x** allant de 1 à 16.

 **remarque:** L'utilisation cette structure est possible dans une commande, il faut utiliser le délimiteur \$ au début et fin.

## Exemple des structures possible:

### Structure simple (sans paramètre):

Nombre de génération	Structure simple	Exemple
1	KEY	DB1
2	KEY-name1,KEY-name2	DB1, DB2
n+1	KEY-name-n	DB1, DB2, DB3

**Remarque:** Cette structure peu être répéter autant de fois que nécessaire. Chaque structure correspond à une génération de check. (Cf. Structure avancée ci-dessous)

### Structure avancée:

Nombre de génération	Structure avancée	Exemple
1	KEY\$(value1)\$	DB1\$(2222)\$
2	KEY-name1\$(value1)\$,KEY-name2\$(value1,value2)\$	DB1\$(2222)\$,DB2\$(3333,ssl)\$

n+1	KEY-name1\$(value1)\$,KEY-name2\$(value1)\$, KEY-name3\$(value1)\$	DB1\$(22)\$,DB2\$(3000)\$, DB2\$(1521)\$
-----	---	---

## Description

La fonctionnalité duplicate foreach s'active directement depuis le check que l'on souhaite générer plusieurs fois. Deux champs sont à renseigner:

### Duplicate for Each:

**Ce champ correspond au nom de la donnée de l'hôte ou cluster à utiliser.**

Une fois le nom de la donnée défini, Shinken interprète au niveau de l'hôte ou cluster: *Crée un check pour chaque valeur de cette donnée, sous la forme KEY1,KEY2,KEY3*

**remarque:** Le nom du check doit contenir un \$KEY\$ pour l'identifier de façon unique.

### Duplicate for Each default:

**Ce champs correspond à la valeur par défaut des arguments de la donnée utiliser pour le "duplicate foreach", soit le nom de la donnée renseigné dans le champ "duplicate foreach".**

**remarque:** Si il y a plusieurs arguments par défaut, utiliser la virgule comme séparateur

## Utilisation

En partant du postulat d'avoir 3 base de données, **DB1**, **DB2** et **DB3** avec un port par défaut à **1521** et une connexion sécurisée en **ssl**. Le nom de la donnée de l'hôte/cluster choisi pour le "dupciate foreach" est **DATABASES**.

### Sans les valeurs des arguments par default:

Page d'édition d'un check:

(1) Renseignez le champ "dupciate foreach" avec **DATABASES**.

Page d'édition d'un hôte/cluster:

(2) Ajouter la donnée **DATABASES** avec pour valeur: **DB1\$(1521,ssl)\$,DB2\$(1521,ssl)\$,DB3\$(1521,ssl)\$**

### Avec les valeurs des arguments par default:

Page d'édition d'un check:

(1) Renseignez le champ "dupciate foreach" avec **DATABASES**.

(2) Renseignez la valeur par default des arguments du champ "dupciate foreach" avec **1521,ssl**.

Page d'édition d'un hôte/cluster:

(3) Ajouter la donnée **DATABASES** avec pour valeur: **DB1,DB2,DB3**

**Remarque:** Définir un port particulier sur DB2 donnerait: Ajouter la donnée **DATABASES** avec pour valeur **DB1, DB2\$(2222)\$,DB3**