

# Gestionnaire d'événements

## Sommaire

### Concept

Mettre en place les gestionnaires d'événements dans Shinken

Activer/Désactiver les gestionnaires d'événements dans Shinken

Activer/Désactiver les logs des gestionnaires d'événements dans Shinken

Comment utiliser le gestionnaire d'événements d'un élément ( Hôtes, Cluster, Checks )

Ajouter une commande au gestionnaire d'événements d'un élément

Logique d'exécution d'une commande par le gestionnaire d'événements

Par défaut ( Seulement quand il passe d'un état non OK à OK, ou inversement )

Exécution lors de n'importe quel changement de statut non-OK confirmé

Exécution pendant une période de maintenance

Individuellement pour un check

Log d'exécution des commandes

## Concept

Le gestionnaire d'événements ( *event handler* ) permet d'exécuter une commande ( *scripts ou exécutables ...* ) lorsqu'un élément ( *cluster, hôte, check* ) change de statut, puis à chaque vérification jusqu'à ce que le statut soit confirmé ( *HARD* ).

L'objectif est de pouvoir invoquer une commande pour réagir à un changement de statut.

Le mode de fonctionnement :

- Le **Scheduler** vérifie si la commande doit être exécutée ou non
- C'est ensuite le **Reactionner** qui exécutera la commande.
  - Dans une architecture distribuée, il est donc essentiel de s'assurer que les scripts ou exécutables à lancer se trouvent sur la machine du **Reactionner** ( *il est crucial de s'assurer que les droits d'accès sont correctement configurés pour la machine du Reactionner* ).
  - L'utilisation d'un **tag de Reactionner** permet de contrôler quel Reactionner exécutera la commande

Exemples d'utilisations :

- Redémarrer un hôte.
- Relancer un service sur un hôte.
- Créer un ticket dans un outil de support.
- Tracer des événements dans une base de données.
- Informer un Shinken distant.

## Mettre en place les gestionnaires d'événements dans Shinken

### Activer/Désactiver les gestionnaires d'événements dans Shinken

Il est possible de rajouter le paramètre **enable\_event\_handlers** dans le fichier de configuration `/etc/shinken/shinken.cfg` ( voir la page [Fichier de configuration \( shinken.cfg \)](#) ) pour activer ou de désactiver les gestionnaires d'événements de **tous les éléments** dans Shinken.



Ce paramètre n'existe pas par défaut dans le fichier **shinken.cfg**, il faudra le rajouter à la suite des autres définitions.

```
enable_event_handlers=1  
  
# Enable or not the event handlers  
# default: 1 (enabled)  
# 0 (disabled)
```

### Activer/Désactiver les logs des gestionnaires d'événements dans Shinken

Le paramètre **log\_event\_handlers** dans le fichier de configuration `/etc/shinken/shinken.cfg` ( voir la page [Fichier de configuration \( shinken.cfg \)](#) ) permet d'activer ou de désactiver les logs du déclenchement des commandes.



Ce paramètre n'existe pas par défaut dans le fichier **shinken.cfg**, il faudra le rajouter à la suite des autres définitions.


```
log_event_handlers=1
# Enable or not the event handlers logs
# default: 1 (enabled)
# 0 (disabled)
```

## Comment utiliser le gestionnaire d'événements d'un élément ( Hôtes, Cluster, Checks )

Le chapitre **Gestionnaire d'événements**, dans l'onglet **expert** de la page d'édition des hôtes, clusters et checks ( voir les pages [Éditer un Hôte](#), [Editer un Cluster](#) et [Editer un check appliqué à un hôte](#) ), permet de configurer le gestionnaire d'événements d'un élément.

Il est possible :

- Activer ou désactiver le gestionnaire d'événements de l'élément
- Spécifier la commande à exécuter et ses paramètres.
- Préciser le tag de Reactionner afin de contrôler quels Reactionners exécuteront la commande.

 Un mécanisme existe pour désactiver tous les gestionnaires d'événements de tous les éléments de Shinken ( voir le chapitre [Activer /Désactiver les gestionnaires d'événements dans Shinken](#) ).

## Ajouter une commande au gestionnaire d'événements d'un élément






La commande sera exécutée par un Reactionner :

- Les commandes exécutées par le gestionnaire d'événements correspondent aux objets commandes de Shinken ( voir la page [Les commandes](#) ).
- Comme pour les notifications, les scripts exécutés auront besoin d'information ( *paramètre* ), notamment pour les connexions à un outil de ticket, ou par exemple pour se connecter à la machine pour supprimer des logs.
- Il est possible d'utiliser des variables ( voir la page [Les Variables \( Remplacement dynamique de contenu - Anciennement les Macros \)](#) ) pour contrôler le comportement de la commande en fonction de l'événement géré.

## Logique d'exécution d'une commande par le gestionnaire d'évènements

### Par défaut ( Seulement quand il passe d'un état non OK à OK, ou inversement )

L'exécution de la commande se fera dans les cas suivants :

- Lors d'un changement de statut **OK** vers statut non-OK ( *CRITIQUE* , *ATTENTION* , *INCONNU* )  
ex :  
- Lors d'un changement statut **non-OK** ( *CRITIQUE* , *ATTENTION* , *INCONNU* ) vers **OK**.  
ex :  
- À la réception de chaque vérification de statut **non-OK** ( *CRITIQUE* , *ATTENTION* , *INCONNU* ), tant que le statut n'est pas confirmé ( *SOFT* ) et une dernière fois lors de la confirmation ( *HARD* ) du statut.  
Ensuite, aucune commande ne sera plus exécutée tant que le statut ne redevient pas **OK**.
- Il n'y a pas d'exécution de commande par le gestionnaire d'évènements lorsqu'un élément est dans une période de maintenance (  ).

### Exécution lors de n'importe quel changement de statut **non-OK confirmé**


En plus des cas précédemment évoqués, la commande peut être exécutée pour chaque changement de statut **non-OK confirmé** ( *HARD* ).

ex :  

Dans le fichier `/etc/shinken-user/configuration/daemons/arbiter/arbiter_cfg_overload.cfg` ( voir la page [Surcharge des paramètres du démon \( arbiter\\_cfg\\_overload.cfg \)](#) ), le paramètre `event_handler_hard_state_trigger_on_any_status_change` permet d'activer ce comportement ( pour tous les éléments de Shinken ).

```
event_handler_hard_state_trigger_on_any_status_change=1
# Activate this option to allow Event Handler to
# execute command on any hard state status change, in addition to its normal behavior.
# 0 : Disabled (Default)
# 1 : Enabled
```

## Exécution pendant une période de maintenance

Il est possible d'effectuer l'exécution de la commande lors d'une période de maintenance (  ).

Dans le fichier de configuration `/etc/shinken-user/configuration/daemons/arbiters/arbiter_cfg_overload.cfg` ( voir la page [Surcharge des paramètres du démon \( arbiter\\_cfg\\_overload.cfg \)](#) ), le paramètre `no_event_handlers_during_downtimes` permet d'activer ce comportement ( pour tous les éléments de Shinken ).

```
downtime.                                     # By default don't launch even handlers during
                                                # Note: the default Nagios behaviour is to allow
them. If you want to keep this                # Nagios behavior, then set this parameter to 0
no_event_handlers_during_downtimes=0
```

## Individuellement pour un check

Pour un check volatil, la commande sera exécutée par le gestionnaire d'événements pour chaque retour de statut **non-OK** ( voir la page [Checks volatiles](#) ).

- Les notifications seront aussi envoyées pour chaque retour de statut non-OK.

## Log d'exécution des commandes

Les logs présents dans le fichier de log du Scheduler ( `/var/log/shinken/Scheduler` ) lors du déclenchement de la commande par le gestionnaire d'événements :

Pour le gestionnaire d'évènements d'un hôte :

```
[YYYY-MM-DD HH:MM:SS] INFO: [ NOM DU SCHEDULER ] SERVICE EVENT HANDLER: NOM DE L'HÔTE; NOM DU SERVICE;
STATUT DU SERVICE; CONFIRMATION DU STATUT; NOMBRE DE TENTATIVES DE CONFIRMATION DE STATUT; NOM DE LA COMMANDE
```

Pour le gestionnaire d'évènements d'un check :

```
[YYYY-MM-DD HH:MM:SS] INFO: [ NOM DU SCHEDULER ] HOST EVENT HANDLER: NOM DE L'HÔTE; STATUT DE L'HÔTE;
CONFIRMATION DU STATUT; NOMBRE DE TENTATIVES DE CONFIRMATION DE STATUT; NOM DE LA COMMANDE
```