# Haute disponibilité des démons de Shinken Entreprise

## Haute disponibilité pour la supervision

La plateforme de supervision est souvent un élément très important d'un parc informatique, vous permettant de rester proactif sur votre infrastructure. Il faut pouvoir avoir pleinement confiance en cette plateforme, en sa capacité à informer des erreurs et problèmes potentiels du périmètre des machines supervisées.

Il est donc intéressant d'avoir une architecture hautement disponible pour la supervision afin que celle ci soit fonctionnelle même lors d'un problème sur une partie de votre réseau.

Shinken Entreprise permet de configurer les démons qui composent une installation de manière à obtenir une supervision plus fiable et résistante aux pannes.

**Sommaire**


- Haute disponibilité pour la supervision
- Comment configurer Shinken Entreprise pour avoir une supervision hautement disponible
- Visualiser l'état d'un spare
- Conseils d'utilisation


## Comment configurer Shinken Entreprise pour avoir une supervision hautement disponible

### Le principe des spares

Chaque démon dans Shinken Entreprise peut être secondé par un démon de même type configuré en tant que spare. Le principe de spares permet d'assurer une haute disponibilité en assurant un service continu même lorsqu'un démon cesse de fonctionner.

On classe alors les démons dans 2 catégories: les démons maîtres et les démons Spare.

Le cycle de vie d'un spare est le suivant:

- Dans un fonctionnement nominal, le démon maître fonctionne correctement: il reçoit des données des autres démons et effectue son travail. Le démon spare est présent pour pouvoir effectuer la travail du démon maître si celui ci ne peut plus assumer son rôle. Dans un cas de fonctionnement nominal comme celui ci, le démon spare est inactif.
- Le démon maître cesse de fonctionner pour une raison quelconque (erreur inconnue, arrêt volontaire, maintenance du serveur, ...). Il ne peut donc plus assumer son rôle. Ce dysfonctionnement est détecté par l'Arbiter qui va alors activer le démon spare pour qu'il remplace le démon maître et puisse continuer d'assurer le service. Le démon spare agit donc exactement comme le démon maitre: il assure le même service. Ce remplacement de démons est invisible du point de vue de l'utilisateur.
- Le démon maître est à nouveau fonctionnel (intervention suite à une erreur, fin de maintenance du serveur, ...). L'Arbiter remarque que le démon maître peut à nouveau effectuer son travail. Puisqu'il s'agit du démon maître, il va être préféré au démon spare. L'Arbiter

va donc désactiver le démon spare et ordonner au démon maître de reprendre son travail.

Plusieurs remarques peuvent être effectuées sur le fonctionnement des spares.

- L'exemple sur le schéma met en scène un seul démon maître et un seul démon Spare. Il peut en fait exister **plusieurs démons maîtres et plusieurs démons Spares.**
  Par exemple, on peut avoir 3 Schedulers maîtres et 2 Schedulers spares. Dans ce cas, lorsqu'un Scheduler maitre est en erreur, un Scheduler spare sera choisi au hasard pour assurer le rôle du Scheduler maître en erreur.
- Un démon spare ne peut remplacer un démon maître que si ils sont **de même type**.
  Par exemple, un Scheduler Spare ne peut pas remplacer un Poller maître en erreur, mais il pourra remplacer un Scheduler maître.
- Un démon spare ne peut remplacer un démon maître que si les deux démons se trouvent **dans le même royaume ou sous-royaume**.
  Aussi, un démon spare peut remplacer un démon d'un sous-royaume seulement si l'option "*manage_sub_realm*" est activée dans le fichier de configuration du démon. Un Poller spare d'un royaume "France" ne pourra pas remplacer un Poller maître d'un royaume "Allemagne".

## Procédure de configuration d'un démon Spare

L'ajout d'un démon Spare s'effectue en 4 étapes:

- Déclarer le démon comme n'importe quel autre démon
- Activer la propriété Spare dans le démon
- S'assurer que le démon en question est bien activé sur la machine du démon Spare.
- Redémarrer Shinken afin que la nouvelle configuration soit prise en compte par les démons.

Pour illustrer la configuration d'un démon spare, on prend l'exemple d'un Scheduler qu'on va configurer en tant que spare.

La configuration d'un Spare, comme pour la configuration d'un démon classique, s'effectue sur la machine hébergeant le démon Arbiter. Aucune modification au niveau des fichiers de configuration n'est à effectuer sur la machine hébergeant le Scheduler spare.

- **Déclaration du scheduler**
  La première étape est donc de déclarer le nouveau Scheduler dans **/etc/shinken/schedulers/scheduler-spare.cfg**. Dans ce fichier, on va donc mettre le contenu suivant (copié depuis **/etc/shinken/schedulers/scheduler-master.cfg**).

**/etc/shinken/schedulers/scheduler-spare.cfg**

```
define scheduler {
    scheduler_name               scheduler-spare
    address                            adresse-scheduler-spare (ip ou nom dns)
    port                               7768
    use_ssl                            0

    timeout                            3
    data_timeout                       120
    max_check_attempts                 3
    check_interval                     60

    modules                            MongodbRetention

    realm                              All
    spare                              0

    enabled                            1
}
```

- **Mise en place du paramètre spare**
  On prend soin ensuite de passer la valeur de l'option **spare** à 1

**/etc/shinken/schedulers/scheduler-spare.cfg**

```
define scheduler{
    ...
    spare 1
    ...
}
```

- **Activation du scheduler sur la machine distante**

  Enfin, sur la machine hébergeant le Scheduler spare, il faut vérifier que le Scheduler est bien activé. Cette vérification va se faire avec les commandes de Manipulation des démons Shinken.
  On vérifie d'abord que le démon Scheduler est activé (sur la machine du scheduler):

```
$ shinken-daemons-list
```

  Si le Scheduler n'est pas activé sur la machine spare, on peut l'activer grâce à la commande *shinken-daemons-enable*:

```
$ shinken-daemons-enable scheduler
```

  Aussi, seuls les démons utilisés doivent être activés sur la machine de spare, sous peine d'avoir des comportements hasardeux en cas de mauvaise configuration. Les démons inutilisés peuvent être désactivés grâce à la commande *shinken-daemons-disable*:

```
$ shinken-daemons-disable demon1 demon2 ...
```

## Cas particulier de l'Arbiter/Synchronizer

Nous avons vu que dans Shinken Entreprise, l'ajout d'un démon spare n'était pas plus difficile que l'ajout de n'importe quel autre démon:

- Déclaration du démon
- Activation du paramètre spare
- Activation du démon sur la machine concernée

Deux démons échappent à cette règle de configuration pour le cas des spares: l'Arbiter et le Synchronizer

En effet, ces deux démons sont spéciaux dans le sens ou ils ne peuvent pas être répliqués facilement:

- Le Synchronizer est unique dans une installation Shinken Entreprise. Il permet de définir la configuration et contient les informations sur les éléments supervisés dans une base de données locales. Le dupliquer ou l'utiliser en tant que spare est bien plus complexe que les autres démons car il faut également répliquer la base de données de configuration.
- L'Arbiter est le démon central de Shinken Entreprise. Il distribue la configuration à tous les autres démons et nécessite des étapes de configuration supplémentaires pour être utilisé en tant que spare.

**Important**

Il ne peut y avoir qu'un seul Arbiter master et un seul Arbiter spare par infrastructure.

Compte tenu de son rôle central, la configuration d'un Arbiter spare nécessite donc des étapes supplémentaires:

- Déclaration d'un nouvel Arbiter dans la configuration
- Activation du paramètre spare
- Mise en place du paramètre host_name dans la configuration des 2 Arbiters
- Activation du démon sur la machine concernée
- Activation du paramètre spare sur la machine hébergeant l'Arbiter spare

- **Déclaration du nouvel Arbiter dans la configuration**

  **/etc/shinken/arbiters/arbiter-spare.cfg**

```
define arbiter {
    arbiter_name         arbiter-spare

    host_name            hostname_machine_spare
    address              adresse-arbiter-spare

    port                 7770
    use_ssl›             0

    timeout              3
    data_timeout         120
    max_check_attempts   3
    check_interval       60

    spare                0

    modules              synchronizer-import

    enabled              1
}
```

- **Activation du paramètre spare**
  Dans le fichier de configuration de l'Arbiter spare, on passe ensuite le paramètre spare à 1

**/etc/shinken/arbiters/arbiter-spare.cfg**

```
define arbiter {
    ...
    spare                1
    ...
}
```

- **Mise en place du paramètre host_name sur les 2 Arbiters**
  Chaque Arbiter dans la configuration doit pouvoir s'identifier. Puisqu'il ne peut y avoir qu'un seul Arbiter par machine, on choisit d'identifier chaque Arbiter par le hostname de la machine sur laquelle il est installé.
  Pour cela, on spécifie le nom du hostname de la machine dans le paramètre host_name de chaque Arbiter.
  Par exemple, sur la machine de l'Arbiter spare, on cherche le hostname de la machine:

```
$ hostname
hostname_machine_spare
```

  Sur la machine hébergeant l'Arbiter master, donc la machine centrale sur laquelle s'effectue la configuration des démons, on spécifie pour chaque Arbiter le paramètre host_name (à effectuer pour les 2 Arbiters).

**/etc/shinken/arbiters/arbiter-spare.cfg**

```
define arbiter {
    ...
    host_name    hostname_machine_spare
    ...
}
```

- **Activation du démon sur la machine concernée**
  Comme pour les autres démons, il faut s'assurer sur la machine hébergeant le démon que celui-ci est bien activé. Cette opération se fait avec les commandes de Manipulation des démons Shinken: *shinken-daemons-enable*, *shinken-daemons-disable* et *shinken-daemons-list*.

- **Activation du paramètre spare sur la machine hébergeant l'Arbiter spare**
  En règle générale, la configuration de tous les démons se fait sur la machine centrale de l'installation Shinken (celle qui héberge l'Arbiter maître).
  Le cas de l'Arbiter spare est un cas particulier. Puisqu'il a un rôle central, lorsqu'il démarre, il agit automatiquement comme arbitre avec les autres démons. Dans le cas d'un spare, on veut par contre qu'il reste inactif et n'essaye pas d'envoyer de configuration aux autres démons.
  Pour cela, il faut, sur la machine du spare, spécifier que le démon est en spare.

**Machine spare > /etc/shinken/arbiters/arbiter-master.cfg**

```
define arbiter {
    arbiter_name    arbiter-spare
    ...
    spare           1
    ...
}
```

## Visualiser l'état d'un spare

Le fait qu'un démon soit en spare est une information importante lorsqu'on veut déterminer le bon fonctionnement d'une plateforme de supervision Shinken Entreprise. La propriété spare d'un démon peut être vue à 2 endroits:

- Lors d'un vérification du fonctionnement de l'installation avec le shinken-healthcheck
- Lors de la supervision de Shinken avec les checks du pack Shinken

## Shinken-healthcheck

Dans le Shinken-healthcheck, les démons spare sont accompagnés d'une annotation "SPARE", qui permet de déterminer facilement si un démon est un démon maître un ou démon spare.

```
- 172.16.0.2 (172.16.0.2):
^^^^^^^^^^^^^^^^^^^^^^^^^^^^
    [broker: broker-corse-spare] (SPARE)
        OK:     Configuration seems valid
        OK:     Connection to daemon is OK at port 7772
        OK:     Daemon version is: 02.04.01-003_BUILD07.fr
        OK:     Correct connection from arbiter "arbiter-vm3" ( and no time shift )
        Talk to:
```

Lors d'une prise de relais, une annotation "RUNNING" est rajoutée à la mention "SPARE".

```
- 192.168.1.139 (192.168.1.139):
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
    [scheduler: scheduler-spare] (SPARE) (RUNNING)
        OK:     Configuration seems valid
        OK:     Connection to daemon is OK at port 7768
        OK:     Daemon version is: 02.04.01-005_BUILD11.fr
        OK:     Correct connection from arbiter "arbiter-master" ( and no time shift )
        Modules:
            OK:     Name: MongodbRetention        Type: mongodb_retention
```

## Checks Shinken

Les checks du pack Shinken permettent de superviser les démons Shinken. Ces checks affichent une pastille spare lorsque le démon supervisé est un démon Spare.

| | | | | | |
|---|---|---|---|---|---|
| | Check | LAB VM2 | Poller - Performance | [OK] SPARE<br>Poller statistics :<br>• This poller is idle: no relevant statistics to show | This poller is currently idle because it is configured as a Spare.<br>It will take over another poller when a main poller stop workin |

Dans le cas d'une défaillance du démon maître, lorsque le Spare prend le relais, une pastille "RUNNING" est rajoutée afin de montrer que le Spare est en cours de fonctionnement :

| | | | | | | |
|---|---|---|---|---|---|---|
| | Check | spare | Scheduler - Performance | ★★★ | All | [OK] SPARE RUNNING<br>Scheduler performance :<br>- Average scheduler CPU usage: 0%<br>- Average time before a check is started in a poller 1.1s<br>Scheduler Satellite : |

| poller name | realm | tags | checks todo | checks done | CPU | | | RAM |
|---|---|---|---|---|---|---|---|---|
| | | | | | CPU available | Metrics: CPU used by the poller | Metrics: CPU Running queue on the poller | % used RAM on the server |
| poller-master | All | None | 11.86/s | 0.14/s | Resources available | 2% (of 2 core) | normal<br>1 Processes in the queue (limit : 8) | normal<br>72% (limit :95%) |
| Total | | | 11.86/s | 0.14/s | | | | |

# Conseils d'utilisation

## Paramétrage des délais de vérification

Selon votre infrastructure, nous vous conseillons d'ajuster les propriétés **timeout**, **max_check_attempts** et **check_interval**, pour permettre à votre architecture en Haute disponibilité d'être la plus efficace possible. (Voir aussi les paramètres de configuration des Démons)

En effet, avec les valeurs par défaut, votre Arbiter vérifie vos différents démons toutes les 60 secondes (**check_interval**). Il considérera le nœud comme "éteint/inaccessible" s'il ne répond pas au bout de 3 secondes (**timeout**) après 3 tentatives de connexion (**max_check_attempts**).

Si vous souhaitez une forte réactivité dans le passage Maître/Spare, vous pourriez par exemple passer votre **check_interval** à 10 secondes et votre **max_check_attempts** à 1. Les délais de bascule seraient alors réduits. Par contre, attention dans un réseau subissant des coupures très régulières ou des fortes latences, car avec ce paramétrage, il se peut que vous basculiez de manière répétée de maître à spare et de spare à maître, et le système de haute disponibilité entraînera de l'instabilité au niveau des communications entres les démons (multiplication des bascules). Il en est de même sur de très grosses installations avec des Schedulers et Brokers qui peuvent être surchargés ou subir de fortes latences réseaux. Les temps de réponse de disponibilité sont alors plus longs, il faudra adapter vos valeurs.

Il s'agit donc de bien choisir les valeurs selon:

- les latences réseaux entre les démons de votre architecture Shinken
- votre politique de PRA (Plan de Reprise d'Activité)

## Pollers spares

**Remarque sur les Poller spares**

Il est possible comme pour les autres démons d'utiliser des Pollers spare. Cependant dans le cas du Poller, il serait plus efficace d'utiliser plusieurs Pollers sans utiliser de spare. En effet, les Pollers se répartissent les checks à effectuer de manière transparente. Un Poller spare inactif peut donc être substitué par un Pollers actif, ce qui est plus intéressant au niveau des performances.