

Scheduler - \$KEY\$ - Running Well

Sommaire

- Contexte
- Paramétrage
 - Données utilisées provenant du modèle
 - Données communes pour les checks du modèle
 - Provenant du modèle shinken
 - Provenant du modèle shinken-daemon
 - Provenant du modèle shinken-scheduler
 - Données spécifiques pour ce check
 - Les données DFE (Duplicate Foreach)
 - Données utilisées provenant du check
 - Données globales
- Résultat
 - Exemple
 - Interprétation
 - Statut
 - Les vérifications communes
 - Les vérifications spécifiques
 - Résultat
 - Résultat Long
 - Description des erreurs
 - Erreur d'un démon bloqué, qui doit être redémarré
 - Un démon spare n'est assigné sur aucun démon master
 - Problème de conflits d'Arbiters
 - Les serveurs ne sont pas à la même heure
 - Erreur de surcharge des disques de logs
 - Le démon a bloqué une tentative de chargement d'objet malveillant
 - Le démon est en cours d'arrêt
 - La dernière connexion de l'Arbiter remonte à trop longtemps
 - Le démon a chargé une configuration enregistrée
 - Erreur de vol de CPU
 - Problèmes réseau
 - Latence réseau importante vers des Pollers passif
 - La récupération des données de connectivité prend trop de temps
 - Le timeout du check est trop court
 - Démons passifs injoignables
 - Schedulers injoignables
 - Problème de conflits d'Arbiters
- Métriques

Contexte

Le modèle **shinken-scheduler** vous permet de superviser un hôte hébergeant le démon Scheduler (voir la page [Le Scheduler](#)).

Le modèle **shinken-scheduler** hérite du modèle **shinken-daemon**.

Afin de superviser le démon Scheduler, le modèle **shinken-scheduler** appliqué à votre hôte, attachera deux checks qui vérifieront la santé et la performance de ce démon.

Sommaire des checks

Les données

Les données communes pour tous les checks

Provenant du modèle shinken-daemon

Les données spécifiques

Les données DFE (Duplicate Foreach)

Comment appliquer un modèle d'hôte à un hôte

Application du modèle via l'interface de Configuration

Application du modèle via un collecteur d'import de fichiers au format .cfg

Checks

- Scheduler - \$KEY\$ - Running Well

Vérifie que le démon Scheduler peut être correctement contacté sur le réseau et affiche son numéro de version.

Affiche également le nombre d'éléments qu'il gère dans un tableau, en distinguant les hôtes, les Clusters et les Checks et le Total des éléments. (*Résultat court*)

Présente la liste des démons auxquels le Scheduler doit se connecter, et l'état des connexions (*Résultat long*).

Donne un état des modules chargés (*Résultat long*).

? Unknown Attachment

? Unknown Attachment

- Scheduler - \$KEY\$ - Performance

Affiche l'utilisation CPU ainsi que les statistiques des Pollers et des Reactionners qu'il gère. Si des limites de CPU ou de RAM sont atteintes sur vos Pollers, alors ces limites seront affichées.

Si certains de vos Pollers sont en spare, ils seront également affichés, avec le texte (*SPARE*) après leurs noms.

Le retour du check Scheduler - Performance contient également un tableau de classification des checks ordonnancés par le Scheduler.

Si jamais le démon Arbiter est en exécution sur une machine virtuelle supervisée par VMware, alors le pourcentage de temps de vol de CPU (*CPU Stolen*) sera affiché.

? Unknown Attachment

? Unknown Attachment

Données du modèle (paramétrage des checks)

Les checks du Scheduler peuvent être configurés via des données fournies par le modèle. Les paramètres SSH sont décrits sur la page [Module MongodBRetention \(Rétenion en base de données centralisée par royaume \)](#)

Les données suivantes sont disponibles pour le Scheduler :

Nom de la donnée	Description	Valeur par défaut	Hérité du modèle d'hôte ou locale
SHINKEN_PROTOCOL	Protocole utilisé pour établir la connexion avec le Scheduler	http	shinken
CHECK_SHINKEN_TIME OUT	Timeout utilisé pour établir la connexion avec le Scheduler, également utilisé par le Scheduler pour tester ses connexions vers les autres démons	3	shinken
SCHEDULER_PORT	Port utilisé pour établir la connexion avec le Scheduler	7768	Locale
SCHEDULER_LIST	Liste de Scheduler (<i>Multi-démon</i>)	scheduler-master\$(\$_HOSTSCHEDULER_PORT)\$	Locale - Duplicate for each (voir la page Dupliquer des checks en fonction d'une liste de valeurs présentes dans la Donnée d'un hôte (duplicate_foreach))
PASSIVE_POLLER_LATE NCY	Latence de connexion (<i>en secondes</i>) au-delà de laquelle le check sort en erreur	0.5	Locale
SCHEDULER__RENTI ON__RENTENTION-IS- TOO-OLD- AFTER_X_MINUTES	Temps (<i>en minutes</i>) additionnel de marge rajouté à l'intervalle de sauvegarde de la rétention avant qu'une rétention ancienne ne soit déclarée trop vieille et retourne en WARNING.	5	Locale

THRESHOLD_CPU_STOLEN_WARNING	Seuil de CPU volé (<i>en pourcentage</i>) sur une machine virtuelle supervisée par VMware avant de déclencher un warning	5	shinken-deamon
THRESHOLD_CPU_STOLEN_CRITICAL	Seuil de CPU volé (<i>en pourcentage</i>) sur une machine virtuelle supervisée par VMware avant de déclencher un critique	10	shinken-deamon

Métriques enregistrées

Les checks du modèle **shinken-scheduler** enregistrent des données de performance, qui peuvent ensuite être affichées dans l'interface de Visualisation sur l'Onglet Graphes ou bien le Widget Graphique. (voir la page [Onglet Graphiques](#) et [Widget Graphique](#))

Nom du check	Nom de la métrique	Explication
Scheduler - \$KEY\$ - Running Well	cpu_stolen__vmware__percent_ready	(<i>Seulement si le démon est situé sur une VM VMWare</i>) Valeur de l'indicateur VMWare %ready (<i>temps de blocage de la VM avant d'avoir accès à ses VCpu, donc temps perdu du point de vue de la VM</i>)
Scheduler - \$KEY\$ - Running Well	nb_checks	Nombre de checks gérés par ce Scheduler.
Scheduler - \$KEY\$ - Running Well	nb_clusters	Nombre de clusters gérés par ce Scheduler.
Scheduler - \$KEY\$ - Running Well	nb_hosts	Nombre de hôte gérés par ce Scheduler.
Scheduler - \$KEY\$ - Running Well	nb_late_checks	Nombre d'exécutions de checks (<i>pour les pollers</i>) en retard de lancement (<i>late</i>) dans ce Scheduler
Scheduler - \$KEY\$ - Running Well	nb_late_notifications	Nombre d'exécutions de notifications (<i>pour les reactionners</i>) en retard de lancement (<i>late</i>) dans ce Scheduler
Scheduler - \$KEY\$ - Running Well	nb_late_event_handlers	Nombre d'exécutions d'event handlers (<i>pour les reactionners</i>) en retard de lancement (<i>late</i>) dans ce Scheduler
Scheduler - \$KEY\$ - Performance	average_scheduler_cpu_estimated_overload	Estimation de la surcharge du Scheduler. <ul style="list-style-type: none"> • Si cette métrique est à zéro alors le Scheduler n'est pas en surcharge. • Si cette métrique est trop souvent supérieure à 0, c'est l'indication qu'il y a besoin d'un Scheduler supplémentaire.
Scheduler - \$KEY\$ - Performance	average_scheduler_cpu_usage	Durée d'un cycle de traitement du Scheduler. <ul style="list-style-type: none"> • Le maximum est à 100. • Plus cette valeur est haute plus cela indique une charge sur le Scheduler.
Scheduler - \$KEY\$ - Performance	checks_todo_by_sec	Nombre de vérifications d'hôtes et de checks générées par seconde dans le Scheduler (<i>moyenne glissante calculée sur 1 min</i>).
Scheduler - \$KEY\$ - Performance	checks_done_by_sec	Nombre de résultats de vérification d'hôtes et de checks donnés par les Pollers par seconde (<i>moyenne glissante calculée sur 1 min</i>).
Scheduler - \$KEY\$ - Performance	notifications_todo_by_sec	Nombre de notifications générées par seconde dans le Scheduler (<i>moyenne glissante calculée sur 1 min</i>).
Scheduler - \$KEY\$ - Performance	event_handlers_todo_by_sec	Nombre d'event handlers générées par seconde dans le Scheduler (<i>moyenne glissante calculée sur 1 min</i>).
Scheduler - \$KEY\$	load_retention_time	Durée en seconde du dernier chargement de rétention

- Performance		
Scheduler - \$KEY\$ - Performance	notifications_and_event_handlers_done_by_sec	Nombre de notifications & event handlers fait par les Reactionners par seconde (<i>moyenne glissante calculée sur 1 min</i>).
Scheduler - \$KEY\$ - Performance	nb_pollers	Nombre de Pollers connectés à ce Scheduler.
Scheduler - \$KEY\$ - Performance	nb_pollers_in_overload	Nombre de Pollers connectés à ce Scheduler en surcharge.
Scheduler - \$KEY\$ - Performance	nb_reactionners	Nombre de Reactionners connectés à ce Scheduler.
Scheduler - \$KEY\$ - Performance	nb_reactionners_in_overload	Nombre de Reactionners connectés à ce Scheduler en surcharge.
Scheduler - \$KEY\$ - Performance	save_retention_time	Durée en seconde de la dernière sauvegarde de rétention

Commandes

Nom du check	Commande du check	Ligne de commande
Scheduler - \$KEY\$ - Performance	check_shinken_scheduler!status!\$VALUE1\$	\$PLUGINSDIR\$/check_shinken -H "\$HOSTADDRESS\$" -p "\$ARG2\$" --shinkenversion "\$SHINKENVERSION\$" -t scheduler -m \$ARG1\$ -l "lck-\$LASTSERVICECHECK\$" --passive_poller_latency "\$_HOSTPASSIVE_POLLER_LATENCY\$" --timeout \$_HOSTCHECK_SHINKEN_TIMEOUT\$ -w \$_HOSTTHRESHOLD_CPU_STOLEN_WARNING\$ -c \$_HOSTTHRESHOLD_CPU_STOLEN_CRITICAL\$ --scheduler_too_old_retention_save_margin \$_HOSTSCHEDULER__RETENTION__RETENTION-IS-TOO-OLD-AFTER_X_MINUTES\$
Scheduler - \$KEY\$ - Running Well	check_shinken_scheduler!alive!\$VALUE1\$	\$PLUGINSDIR\$/check_shinken -H "\$HOSTADDRESS\$" -p "\$ARG2\$" --shinkenversion "\$SHINKENVERSION\$" -t scheduler -m \$ARG1\$ -l "lck-\$LASTSERVICECHECK\$" --passive_poller_latency "\$_HOSTPASSIVE_POLLER_LATENCY\$" --timeout \$_HOSTCHECK_SHINKEN_TIMEOUT\$ -w \$_HOSTTHRESHOLD_CPU_STOLEN_WARNING\$ -c \$_HOSTTHRESHOLD_CPU_STOLEN_CRITICAL\$ --scheduler_too_old_retention_save_margin \$_HOSTSCHEDULER__RETENTION__RETENTION-IS-TOO-OLD-AFTER_X_MINUTES\$

Les modes dépréciés ("-m") :

- api_connection
- late_checks
- latency
- top10_average
- top10_total

Check : Scheduler - \$KEY\$ - Running Well

Statistiques de l'état de santé du Scheduler

Si votre Scheduler est en bon état de fonctionnement, c'est-à-dire qu'il permet d'ordonnancer correctement tous les checks et de recevoir les résultats de ces checks en temps et en heure, alors le statut du retour du check "Scheduler - Running Well" est **OK**.

Résultat court :

Suite à ce retour, les informations suivantes sont disponibles dans le *résultat court* :

- Si le Scheduler est connecté à des Pollers passifs, la latence de la connexion vers chacun d'eux est affichée.
- Un tableau contenant le nombre d'éléments que gère le Scheduler.
- Le royaume géré par le Scheduler.
- Le numéro de version du logiciel.
- Le temps consommé pour récupérer les informations générales
- Le temps utilisé pour récupérer l'état des connexions vers les Schedulers, les Pollers passifs et les Reactionners passifs

? Unknown Attachment

Résultat long :

Les informations suivantes sont également disponibles dans le *résultat long* :

- Une liste des démons (*Schedulers, Pollers passifs, Reactionners passifs*) auxquels le Scheduler doit se connecter, avec pour chacun des démons :
 - Son nom (*suivi des tags gérés pour les Pollers passifs ou pour les Reactionners passifs*)
 - Son type
 - La valeur du paramètre **timeout** renseigné dans le fichier de configuration de ce démon (*qui correspond au délai potentiel maximal requis pour le contacter*)
 - L'état de la connexion
- La liste des modules chargés ainsi que leur état

? Unknown Attachment

Exemple d'un état de santé dégradé du Scheduler

Il peut arriver que vos Pollers ne permettent pas d'absorber tous les checks ordonnancés par le Scheduler, et dans ce cas, certains seront en retard ! Les checks sont considérés en retard s'ils dépassent 10 secondes à partir du moment où ils sont mis à disposition des Pollers par les Schedulers.

L'état de retour du check "Scheduler - \$KEY\$ - Running Well" est alors **WARNING** . Le nombre de checks en retard et le pourcentage des checks en retard par rapport au volume géré par le Scheduler sont affichés.

Le volume géré par le Scheduler est calculé à partir de ces données :

- Le nombre de checks
- Le nombre d'hôtes (*un check est fait pour chaque commande de vérification*)
- Le nombre de clusters (*un check est fait pour chaque définition du cluster*)

Le nombre de checks en retard est ensuite affiché par Poller Tag.

Pour les checks qui ont été exécutés durant les cinq dernières minutes, le temps d'attente moyen sur le Scheduler et sur le Poller est également affiché. Si ce chiffre s'approche des 10 secondes :

- Vos Pollers n'arrivent pas à absorber toute la charge : il peut être nécessaire d'ajouter un nouveau Poller.
- Il peut également s'agir d'un problème de latence réseau ou que l'un des Pollers ne soit plus disponible.

Ces informations sont également disponibles pour les notifications et les événements. Dans ce cas, le démon concerné est le Reactionner

Enfin, le check rappelle à quel royaume appartient le Scheduler.

? Unknown Attachment

Description des erreurs

En attente de la configuration

Lorsque le Scheduler est en attente de la configuration de l'Arbiter, le *résultat court* du check affiche un message d'avertissement pour le signaler. Le *résultat long* du check reste vide.

? Unknown Attachment

Le Scheduler contacté est d'une version incompatible

Lorsque le Scheduler supervisé n'est pas de la même version, le check renvoie une information le signalant, et toute autre information n'est pas disponible dans le *résultat court* ou dans le *résultat long* .

? Unknown Attachment

Problème de surcharge des disques de logs

- **Disque des logs trop lent** :

En cas de disques trop lent sur le volume des logs, le check sera mis en **WARNING** avec l'erreur suivante.

? Unknown Attachment

Erreur d'un démon bloqué, qui doit être redémarré

- Si un démon est dans un état bloqué, il doit être redémarré. Si c'est le cas:
 - les checks seront en **ERROR** avec le message suivant,
 - il faut ouvrir un ticket à votre support pour analyser le blocage

[CRITICAL]

The daemon have a **lock**, it's **not working** and **MUST** be restarted.

Please contact your support to analyse the daemon logs:

- "Main loop" was locked more than 3600s
- Detected at 2021-12-03 08:21:55 [WATCH DOG]

Le démon a bloqué une tentative de chargement d'objet malveillant

Il est possible qu'un démon puisse détecter et bloquer une tentative d'injection d'objet malveillant par le biais de l'une de ses routes.

Un message est remonté :

- le nombre total de ces tentatives que le démon a bloqué ce jour (*le compte commence à minuit*) ;
- pour chacune des tentatives (*maximum 3*) :
 - descriptif de l'objet que l'attaquant essaye de charger,
 - sa provenance de l'attaque, par exemple le nom de la route utilisée, et l'IP à la source de l'attaque,
 - sa date.

? Unknown Attachment

Problèmes réseau

Latence réseau importante vers des Pollers passif

Quand la connexion vers un ou plusieurs Pollers passifs souffre d'une latence réseau trop importante, cette information est remontée dans le *résultat court*

[WARNING] Your scheduler have some issues.

=> The latency is too high between some passive pollers and the scheduler :

- poller-hostdev3 [hostdev3:7771] : 1005.5ms (> 500ms)
- poller-hostdev2 [hostdev2:9771] : 0.42ms

La récupération des données de connectivité prend trop de temps

Quand le check ne parvient pas à récupérer les données de connectivité du Scheduler, la cause est indiquée dans le *résultat court* et le tableau affichant l'état des connexions dans le *résultat long* n'est plus affiché.

[WARNING] Your scheduler have some issues.

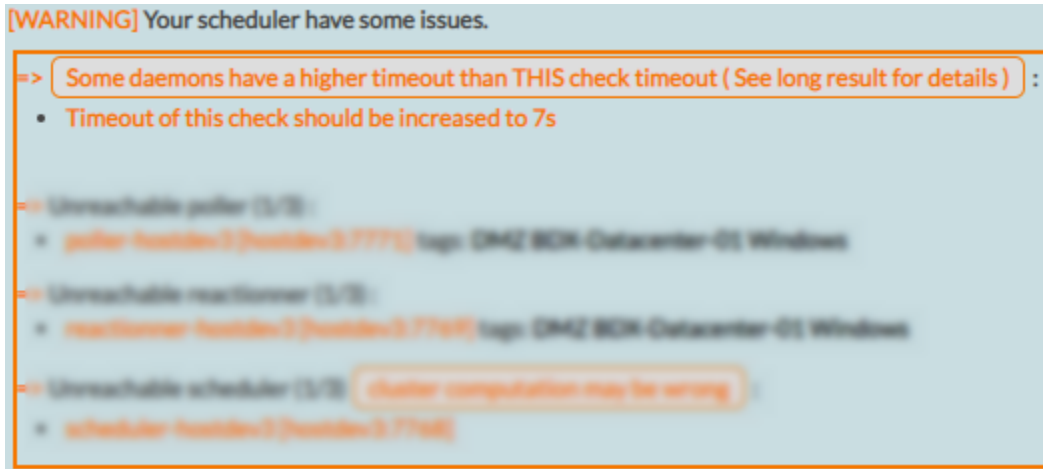
=> No answer from Scheduler, unable to fetch connectivity data with cause : (timed out)

Le timeout du check est trop court

Quand le timeout associé à ce check est inférieur au paramètre timeout renseigné dans la configuration d'un des démons que doit contacter le Scheduler, il se peut que la connexion vers ce démon échoue lors du test de connectivité.

Résultat court:

Un message d'avertissement signale que certains démons nécessitent un timeout plus élevé pour être contacté, et une valeur conseillée est affichée.



Résultat long :

Dans ce cas, le **Status** dans le *résultat long* précise que l'erreur peut être liée au délai trop court accordé pour tester la connexion. Il est alors conseillé d'augmenter le timeout du check pour que le test soit pertinent.

Connectivity status:			
Name	Type	Timeout	Status
poller-hostdev2 (tags: None)	poller (passive)	3s	OK
poller-hostdev3 (tags: DMZ BDX-Datacenter-01 Windows)	poller (passive)	5s	WARNING Unreachable poller at http://hostdev3:7771/ (timed out, BUT the result may not be reliable => [because the check timeout (3s) is shorter than poller-hostdev3 configured timeout (5s)] => Please increase timeout of THIS check to at least 7s)
reactionner-hostdev2 (tags: None)	reactionner (passive)	3s	OK
reactionner-hostdev3 (tags: DMZ BDX-Datacenter-01 Windows)	reactionner (passive)	6s	WARNING Unreachable reactionner at http://hostdev3:7769/ (timed out, BUT the result may not be reliable => [because the check timeout (3s) is shorter than reactionner-hostdev3 configured timeout (6s)] => Please increase timeout of THIS check to at least 7s)
scheduler-master	scheduler	3s	OK
scheduler-hostdev2	scheduler	10s	OK
scheduler-hostdev3	scheduler	2s	WARNING Unreachable scheduler at http://hostdev3:7768/ (timed out)

Démons passifs injoignables

Résultat court :

Quand la connexion vers certains Pollers passifs ou certains Reactionners passifs est impossible, le *résultat court* du check liste les démons injoignables, en précisant pour chacun :

- Le nom
- L'adresse et le port de connexion
- Les tags gérés

Ceux-ci sont regroupés par type (*Poller* ou *Reactionner*), un compteur indique le nombre de passifs injoignables et le nombre total de démons du même type disponible (*passifs* et *actifs*)

[WARNING] Your scheduler have some issues.

=> Unreachable pollers (2/3) :

- poller-hostdev2 [hostdev2:9771] tags: None
- poller-hostdev3 [hostdev3:7771] tags: DMZ BDX-Datacenter-01 Windows

=> Unreachable reactionner (1/3) :

- reactionner-hostdev3 [hostdev3:7769] tags: DMZ BDX-Datacenter-01 Windows

Résultat long :

Le tableau du *résultat long* , indique les problèmes de connectivité dans la colonne **Status**, avec un message précisant leur nature.

Connectivity status:			
Name	Type	Timeout	Status
poller-hostdev2 (tags: None)	poller (passive)	3s	WARNING Unreachable poller at http://hostdev2:9771/ ([Errno 111] Connection refused)
poller-hostdev3 (tags: DMZ BDX-Datacenter-01 Windows)	poller (passive)	5s	WARNING Unreachable poller at http://hostdev3:7771/ ([Errno 111] Connection refused)
reactionner-hostdev2 (tags: None)	reactionner (passive)	3s	OK
reactionner-hostdev3 (tags: DMZ BDX-Datacenter-01 Windows)	reactionner (passive)	6s	WARNING Unreachable reactionner a http://hostdev3:7769/ ([Errno 111] Connection refused)
scheduler-master	scheduler	3s	OK
scheduler-hostdev2	scheduler	10s	OK
scheduler-hostdev3	scheduler	2s	OK

Schedulers injoignables

Résultat court :

Quand le Scheduler ne parvient pas à communiquer avec un ou plusieurs Schedulers du royaume, ceux-ci sont listés avec :

- Leur nom
- Leur adresse
- Leur port de connexion

Un compteur indiquant le nombre de Schedulers injoignables et le nombre total de Schedulers disponibles est également affiché.

L'indisponibilité d'un ou plusieurs Scheduler pouvant perturber le calcul des états de clusters, un message d'avertissement le précisant est également ajouté.

[WARNING] Your scheduler have some issues.

=> Unreachable scheduler (1/3) **cluster computation may be wrong** :

- scheduler-hostdev3 [hostdev3:7768]

Résultat long :

Le tableau listant les connexions du Scheduler, indique les problèmes de connectivité vers les autres Schedulers dans la colonne **Status**, avec un message précisant la nature du problème.

Connectivity status:			
Name	Type	Timeout	Status
poller-hostdev2 (tags: None)	poller (passive)	3s	OK
poller-hostdev3 (tags: DMZ BDZ-Datacenter-01 Windows)	poller (passive)	5s	OK
reactionner-hostdev2 (tags: None)	reactionner (passive)	3s	OK
reactionner-hostdev3 (tags: DMZ BDZ-Datacenter-01 Windows)	reactionner (passive)	6s	OK
scheduler-master	scheduler	3s	OK
scheduler-hostdev2	scheduler	10s	OK
scheduler-hostdev3	scheduler	2s	WARNING Unreachable scheduler at http://hostdev3:7768/ ([Errno 111] Connection refused)

Problème de conflits d'Arbiters

- **Conflits d'Arbiters :**

Si le démon est contacté par des Arbiters qui ne sont pas sur la même architecture (*par exemple un Arbitre de Production et un autre de l'environnement de Testing*), le check sera mis en **CRITICAL** .

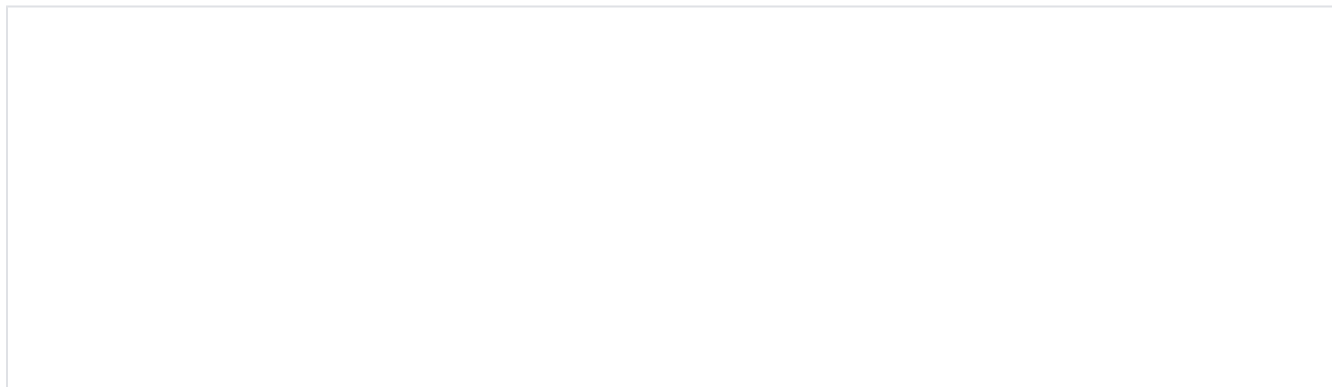
=> Arbiters CONFLICT

Architecture List :

1. Production [**CRITICAL**]
 - o arbiter-master : last connection 11s ago. Defined on the server with uuid d2a358b0-cae3-4ce8-bc3f-edd241627bac (/var/lib/shinken/server.uuid)
2. Testing [**CRITICAL**]
 - o arbiter-master : last connection 12s ago. Defined on the server with uuid 785e0227-455a-449d-b8f1-516897000e3b (/var/lib/shinken/server.uuid)

- **Conflit d'Arbiters qui ont le même nom d'Architecture :**

Comme dans le cas précédent, le démon est contacté par des Arbiters d'architectures différentes, mais qui ont le même nom. On sort également en CRITICAL mais en avertissant que les noms sont identiques, et en indiquant où changer le nom de vos architectures.



=> Arbiters CONFLICT

Architecture List :

1. Production [XXXXXXXXXX]

- o arbiter-master : last connection 33s ago. Defined on the server with uuid d2a358b0-cae3-4ce8-bc3f-edd241627bac (/var/lib/shinken/server.uuid)

2. Production [XXXXXXXXXX]

- o arbiter-master : last connection 10s ago. Defined on the server with uuid 785e0227-455a-449d-b8f1-51689700e3b (/var/lib/shinken/server.uuid)

NOTE:

Some architecture have the same name. We advise you to change it in the configuration of their architecture_export module.

Les serveurs ne sont pas à la même heure

- Si le serveur n'est pas à la même heure que le serveur Arbitre (qui fait office de référence), une erreur **CRITICAL** sera levée, car des temps différents sur des serveurs distincts va avoir des effets **désastreux** sur la cohérence des données de supervision.

? Unknown Attachment

La dernière connexion de l'Arbiter remonte à trop longtemps

- Si la dernière connexion de l'Arbiter remonte à trop de temps, le démon va lever un **WARNING** . Ceci peut être dû à :
 - o Les Arbiters MASTER et SPARE sont réellement éteints.
 - o Les Arbitre MASTER et SPARE sont en train d'envoyer des configurations à d'autres démons, et ne peuvent donc pas contacter ce démon pour l'instant.

? Unknown Attachment



Le temps pris en compte comme limite de dernière connexion est de `check_interval * max_check_attempts` du démon (définis dans sa configuration).

Les valeurs par défauts sont de `60s * 3` (soit 3 minutes)

Le démon est en cours d'arrêt

Lorsque le démon est en cours d'arrêt, le check le signale, et les informations relatives aux modules ne sont plus disponibles

? Unknown Attachment

Check : Scheduler - \$KEY\$ - Performance

La supervision d'un démon Scheduler présente un grand nombre de statistiques de performances qui permettent de visualiser le travail d'ordonnement effectué par le Scheduler, ainsi que les statistiques des Poller qui viennent se connecter à celui-ci.

Le démon Scheduler va effectuer tout le travail d'ordonnement, et c'est à lui que vont s'adresser (en autres) les démons Poller et Reagir pour récupérer les checks et les notifications à effectuer. Son bon fonctionnement est donc vital au bon fonctionnement de votre architecture Shinken. Aussi, pour dimensionner correctement une installation Shinken Entreprise, il est important de pouvoir visualiser combien de checks ses Poller satellites peuvent traiter, ainsi que leurs utilisations CPU et RAM.

Les checks du Scheduler fournis dans le pack Shinken proposent donc un grand nombre de données sur les performances du Scheduler et de ses Pollers.

L'ensemble des informations se retrouve dans le résultat court du check.

Statistiques générales

- La première statistique remontée par le check est le pourcentage CPU moyen utilisé par le démon Scheduler sur le serveur supervisé.

- La deuxième statistique remontée est le temps d'attente moyen d'un check sur le Scheduler avant d'être récupéré par un Poller pour traitement.
- La date et la durée de la dernière sauvegarde réussie de rétention
- La date et la durée du dernier chargement de rétention
- Si votre machine virtuelle est hébergée sur un système VMMare, alors une dernière statistique sera remontée.
 - Elle affiche le taux de CPU %ready (*vol de temps de calcul du CPU votre machine par les autres machines virtuelles de l'hyperviseur*).

? Unknown Attachment

Suivi des chargements/sauvegardes des données de rétention

Les données de rétention sont chargées/sauvegardées par les démons Schedulers. Un affichage permet de voir :

- La date et la durée du dernier chargement de rétention (*lors d'une nouvelle configuration*)
- La date et la durée de la dernière sauvegarde de rétention (*lors d'une nouvelle configuration, ou alors toutes les retention_interval disponibles dans le fichier shinken.cfg*)

? Unknown Attachment

? Unknown Attachment

Si la dernière sauvegarde de rétention est trop vieille, c'est-à-dire plus que `retention_interval + SCHEDULER__RETENTION__RENTENTION-IS-TOO-OLD-AFTER_X_MINUTES`, alors un **WARNING** sera remonté.

? Unknown Attachment

Si le dernier essai en date de sauvegarde de rétention est en **ERROR**, alors un message sera disponible avec le dernier message du module en question.

? Unknown Attachment

Suivie des Pollers Satellites

Informations générales

Suite aux statistiques générales, un premier tableau rassemble les données de performance des satellites du Scheduler de type Poller.

La première partie du tableau (*les trois premières colonnes*) permet d'identifier les Pollers en affichant leurs noms, leurs appartenances à un Royaume, et enfin leurs tags (*None si aucun tag n'est associé au Poller*).

? Unknown Attachment

Statistiques des checks

Les deux colonnes suivantes affichent les performances de traitement des checks des Pollers :

- checks todo : Moyenne du nombre de checks à **traiter** par le Poller (en checks par seconde)
- checks done : Moyenne du nombre de checks **traités** par le Poller (en checks par seconde)

Vous pourrez donc avoir l'information du nombre de checks récupérés et traités par vos différents Pollers rattachés à ce Scheduler et ainsi pouvoir comparer les performances de vos Pollers suivant leur positionnement dans votre architecture réseau, ou suivant leur puissance matérielle.

? Unknown Attachment

Utilisation du CPU

CPU Available

La colonne "CPU available" concerne les performances CPU des Pollers.

Cette information représente la charge du Poller. Il s'agit d'un indicateur général indiquant si le Poller peut encore supporter des checks supplémentaires, ou s'il est chargé au maximum. Cet indicateur n'est pas lié aux autres indicateurs de performances de la machine (*File d'attente CPU, mémoire*)

Une pastille orange précédant la mention "Poller load" signifie que le Poller ne peut plus prendre de checks supplémentaires.

C'est donc un signe indiquant qu'il faudrait ajouter un Poller supplémentaire dans l'architecture Shinken.

Si tous vos Pollers sont en surcharge, alors les checks ne pourront plus être récupérés, et vous aurez des retards visibles dans le retour de votre check "Scheduler - Running Well".

Il vous faudra de toute urgence rajouter des Pollers dans votre royaume.

? Unknown Attachment

Voici par exemple une surcharge d'un Poller.

? Unknown Attachment

CPU used by the poller

La colonne "CPU used by the poller" permet d'afficher la consommation CPU utilisée par le Poller. Comme son nom l'indique, un graphique est associé à ce check et permet d'afficher cette métrique.

Lorsque le Poller utilise le maximum de CPU possible sur le serveur, une information apparaît.

Cette valeur de CPU utilisée par le Poller ne sera jamais à 100%, car le système Linux hébergeant le démon utilise une partie du CPU, comme les applications additionnelles que ce serveur peut utiliser.

Plus il y a d'application sur votre serveur Poller consommant du CPU, moins votre démon pourra utiliser de CPU à ses fins et atteindra rapidement sa charge maximale utilisable (*bien en deçà de 100%*).

? Unknown Attachment

Lorsque la limite est atteinte, voici l'affichage dans le tableau.

? Unknown Attachment

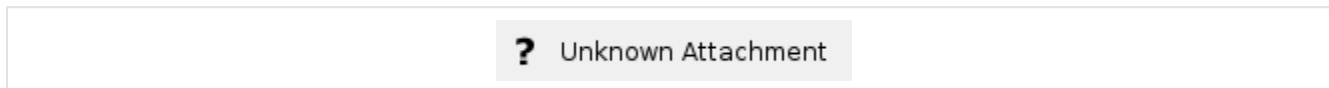
Utilisation de la RAM

% used RAM on the server

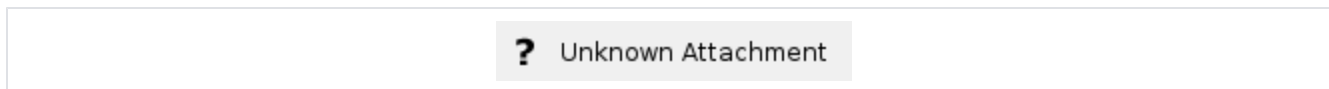
La dernière colonne du tableau représente le pourcentage de RAM utilisé sur le serveur.

Si la valeur détectée est inférieure à la limite définie, alors la consommation est considérée comme normale et la pastille "normal" est affichée.

La limite paramétrée dans le Poller est affichée entre parenthèses.



Si l'utilisation de la mémoire (RAM) sur le serveur dépasse le seuil défini dans la configuration de ce Poller, une pastille rouge de dépassement est affichée, indiquant l'utilisation excessive de la mémoire. Lorsque cet avertissement est affiché, le Poller n'exécute plus de checks supplémentaires tant que l'utilisation de la mémoire est supérieure au seuil défini.

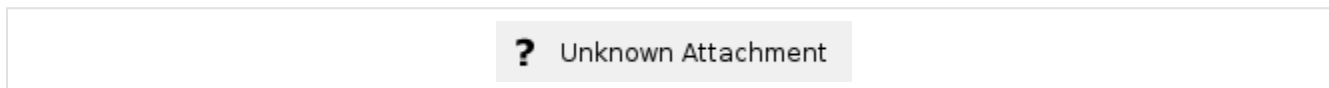


Load

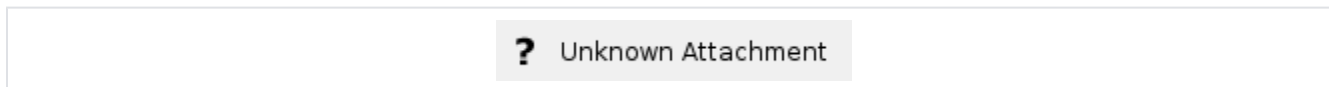
Si par exemple le CPU n'est pas utilisé au maximum de ses performances, mais que sa "running queue" (*file d'attente*) est importante, la limitation de CPU ne peut prévenir ce cas.

Pour s'assurer que le Poller ne tente d'exécuter des checks sur une machine surchargée le Poller se limitera en fonction de l'état de la file d'attente processeur (*représentant la valeur source du load average*).

Dans ce cas, le Poller n'exécutera plus de checks supplémentaires tant que le nombre de processus dans la file d'attente du processeur sera supérieur au seuil choisi. La limite paramétrée dans le Poller est affichée entre parenthèses.



Lorsque la limite est atteinte pour ce Poller, alors le check "Scheduler - Performance" ajoute une pastille rouge vous informant du dépassement de la limite.



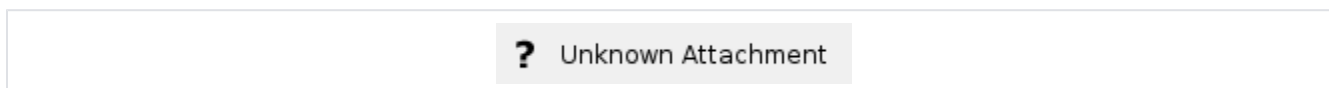
Suivie des Reactionners Satellites

Suite aux statistiques des Satellites de type "Poller", un deuxième tableau rassemble les données de performance des satellites du Scheduler de type Reactionner.

Les trois premières colonnes représentent, comme pour le tableau précédent, les données d'identification des Reactionners venant récupérer les notifications auprès du Scheduler.

Les deux colonnes suivantes permettent d'obtenir les statistiques des notifications à traiter par les Reactionner ainsi que les notifications déjà réalisées (*en nombre de notifications par seconde*).

Enfin les deux dernières colonnes affichent les informations CPU des Reactionners, de la même manière que pour les Pollers.



Type de checks fait par seconde

Le Scheduler est un ordonnanceur de checks.

Cet ordonnancement peut être fait pour différentes raisons qui sont énumérées dans la colonne "Causes" de ce tableau ci-contre :

- **Dependency** : Les checks qui sont demandés, car liés à une dépendance (*checks liés à son hôte ou hôte fils lié à son hôte parent*)

- **Retry** : Les checks qui sont revérifiés pour la confirmation des états, via la propriété "Intervalle de nouvelles tentatives de confirmations d'état" des checks et des hôtes
- **Force** : Les checks qui sont demandés par les utilisateurs depuis l'interface de visualisation (bouton "*Forcer la vérification*")
- **Schedule** : Les checks qui sont ordonnancés de manière régulière via la propriété "intervalle entre les vérifications" des checks et des hôtes (*normalement le plus actif des 4 raisons*)

Pour chaque raison, le nombre de checks par seconde est affiché dans la deuxième colonne.

? Unknown Attachment

Consommation de temps CPU des checks

Le check "Scheduler - Performance" peut également détecter si la commande d'un check prend trop de temps CPU lors de son exécution.

Si le seuil est atteint (*et dans ce cas seulement*), le check passe en état **WARNING** et le tableau ci-contre apparaît dans le résultat du check.

Ce tableau contient le nom des commandes, leurs temps CPU consommés, le seuil fixé pour cette commande et la date de l'exécution.

? Unknown Attachment

Par défaut, le seuil est fixé à 5 secondes. Cette propriété nommée "Seuil d'alerte de l'utilisation CPU (sec)" est modifiable via l'UI de configuration dans les onglets "Supervision" des objets "hôte" et "check" et dans l'onglet "Général" des commandes. La clé d'import est **warning_threshold_cpu_usage**.

Ce paramètre est aussi modifiable globalement dans le fichier `/etc/shinken/shinken.cfg`.

```
# How many seconds a command check (for hosts, clusters and checks) is allowed to consume cpu
# before raising a warning in check scheduler performance
# by default: 5
#warning_threshold_cpu_usage=5
```

Après modification, un redémarrage de l'Arbiter sera ici requis.

Cas Particuliers d'erreur

Si un Poller est détecté comme injoignable (*par exemple s'il y a un problème réseau avec ce démon ou alors qu'il vient juste d'être désactivé depuis l'Arbiter*) alors un message est affiché.

? Unknown Attachment

Description des erreurs

Vol de CPU

Seulement si votre machine virtuelle est hébergé sur un hyperviseur VMWare

- **Votre machine à du vol de CPU** :
 - Si la VM se fait voler trop de temps de calcul (CPU Stolen), le check sera mis en **WARNING** ou en **CRITIQUE** (*en fonction du taux de vol fixé par défaut ou indiqué par l'utilisateur*).
 - Vous pouvez avoir plus d'information sur cet indicateur et comment réduire la part de temps de la VM sur la page [Machine VMWare avec un fort taux de CPU Stolen \(%ready + %costop\)](#)

? Unknown Attachment

? Unknown Attachment

Erreur d'un démon bloqué, qui doit être redémarré

- Si un démon est dans un état bloqué, il doit être redémarré. Si c'est le cas:
 - les checks seront en **ERROR** avec le message suivant,
 - il faut ouvrir un ticket à votre support pour analyser le blocage

[CRITICAL]

The daemon have a **lock**, it's **not working** and **MUST** be restarted.

Please contact your support to analyse the daemon logs:

- "Main loop" was locked more than 3600s
- Detected at 2021-12-03 08:21:55 [WATCH DOG]

Le démon a bloqué une tentative de chargement d'objet malveillant

Il est possible qu'un démon puisse détecter et bloquer une tentative d'injection d'objet malveillant par le biais de l'une de ses routes.

Un message est remonté :

- le nombre total de ces tentatives que le démon a bloqué ce jour (*le compte commence à minuit*) ;
- pour chacune des tentatives (*maximum 3*) :
 - descriptif de l'objet que l'attaquant essaye de charger,
 - sa provenance de l'attaque, par exemple le nom de la route utilisée, et l'IP à la source de l'attaque,
 - sa date.

? Unknown Attachment

Le démon est en cours d'arrêt

Lorsque le démon est en cours d'arrêt, le check le signale, et les informations relatives aux modules ne sont plus disponibles

? Unknown Attachment