

shinken-broker

Contexte

Les modèles shinken-broker, shinken-broker-db et shinken-broker-module-sla vous permettent de superviser un hôte hébergeant le démon **Broker**.

Sommaire

- Contexte
- Description du modèle

Description du modèle





Modèle d'hôte correspondant: **shinken-broker** (notez que ce modèle hérite du modèle **shinken** et **shinken-broker-db**)




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

Le modèle d'hôte **shinken-broker-module-sla** attachera un check additionnel qui permettra de superviser le module SLA.

Le modèle hérité **shinken-broker-db** attachera quant à lui, les checks qui permettront de superviser la base utilisée par Shinken pour le démon Broker.

Checks

Nom du Check	Description	Exemple de résultat																														
Broker - \$KEY\$ - Alive	Vérifie que le démon Broker peut être correctement contacté sur le réseau ; la version du démon est affichée également (Résultat court) et que les modules sont opérationnels (Résultat long).	 Broker - broker-master - Alive [OK] The daemon is running well. Version [02.05.00-004_BUILD10.fr]. Connection established in 0.019s. Module info: <table border="1"><thead><tr><th>Name</th><th>Type</th><th>Status</th><th>Restart in the last 2h</th><th>Last restart date</th><th>Submodules</th></tr></thead><tbody><tr><td>sla</td><td>sla</td><td>[OK]</td><td>0</td><td></td><td></td></tr><tr><td>Simple-log</td><td>simple-log</td><td>[OK]</td><td>0</td><td></td><td></td></tr><tr><td>WebUI</td><td>webui</td><td>[OK]</td><td>0</td><td></td><td><ul style="list-style-type: none">• MongoDB: [OK]• Cfg_password: [OK]• webui-enterprise: [OK]• sla: [OK]</td></tr><tr><td>Graphite-perfdata</td><td>graphite_perfdata</td><td>[OK]</td><td>0</td><td></td><td></td></tr></tbody></table>	Name	Type	Status	Restart in the last 2h	Last restart date	Submodules	sla	sla	[OK]	0			Simple-log	simple-log	[OK]	0			WebUI	webui	[OK]	0		<ul style="list-style-type: none">• MongoDB: [OK]• Cfg_password: [OK]• webui-enterprise: [OK]• sla: [OK]	Graphite-perfdata	graphite_perfdata	[OK]	0		
Name	Type	Status	Restart in the last 2h	Last restart date	Submodules																											
sla	sla	[OK]	0																													
Simple-log	simple-log	[OK]	0																													
WebUI	webui	[OK]	0		<ul style="list-style-type: none">• MongoDB: [OK]• Cfg_password: [OK]• webui-enterprise: [OK]• sla: [OK]																											
Graphite-perfdata	graphite_perfdata	[OK]	0																													
Broker - \$KEY\$ - Performance API Connection	Vérifie les temps de connexion vers le Broker	 Broker - broker-master - Performance API Connection [OK] API Connexion is in the good range (0.003 < 2s).																														
Broker - \$KEY\$ - Performance Modules Queues	Vérifie les performances des modules du Broker	 Broker - broker-master - Performance Modules Queues [OK] Modules performance is going well																														
Broker - DB - Connection	Vérifie le temps de connexion à la base de données	 Broker - DB - Connection OK - Connection took 0 seconds																														

Broker - DB - Last Flush Time	Vérifie la durée du dernier "Flush" de la base de données	 Broker - DB - Last Flush Time OK - Last Flush Time: 2770.00ms
Broker - DB - Lock Time Percentage	Vérifie le pourcentage de temps de lock de la base de données	 Broker - DB - Lock Time Percentage OK - Lock Percentage: 0.19%
Broker - DB - Open Connections	Vérifie le nombre de connexions en cours sur la base de données	 Broker - DB - Open Connections OK - 0 percent (152 of 51200 connections) used

Paramétrage des Checks

Les checks du Broker peuvent être configurés via des données fournies par le modèle.

Les données suivantes sont disponibles:

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 Broker	http	shinken
BROKER_PORT	Port utilisé pour l'établissement de la connexion avec Broker	7772	Locale
BROKER_LIST	Liste de Broker (Multi-démon)	broker-master\$(\$_HOSTBROKER_PORTS)\$	Locale - Duplicate For Each
DB_CONNECTION_METHOD	Méthode de connexion au serveur MongoDB ("direct" ou "ssh")	ssh	shinken-broker-db
DB_CONNECTIONS_CRITICAL	Seuil critique du nombre de connexions en cours sur la base de données (en pourcentage)	80	shinken-broker-db
DB_CONNECTIONS_WARNING	Seuil warning du nombre de connexions en cours sur la base de données (en pourcentage)	70	shinken-broker-db
DB_CONNECT_CRIT	Seuil critique pour le temps de connexion à la base de données (en secondes)	4	shinken-broker-db
DB_CONNECT_WARN	Seuil warning pour le temps de connexion à la base de données (en secondes)	2	shinken-broker-db
DB_LAST_FLUSH_TIME_CRITICAL	Seuil critique pour le temps de flush de la base de données (en millisecondes)	10000	shinken-broker-db
DB_LAST_FLUSH_TIME_WARNING	Seuil warning pour le temps de flush de la base de données (en millisecondes)	8000	shinken-broker-db
DB_LOCK_CRIT	Seuil critique du temps de lock de la base de données (en pourcentage)	10	shinken-broker-db
DB_LOCK_WARN	Seuil warning du temps de lock de la base de données (en pourcentage)	5	shinken-broker-db
DB_PORT	Port de la base de données	27017	shinken-broker-db
DB_SSH_KEY	Clé SSH privée permettant la connexion au serveur hébergeant MongoDB	~/.ssh/id_rsa	shinken-broker-db
DB_SSH_USER	Utilisateur distant avec lequel se connecter au serveur hébergeant MongoDB	shinken	shinken-broker-db

Voir la page de [Sécurisation des connexions aux bases MongoDB](#) pour plus d'information sur un paramétrage sécurisé.

Détail des commandes

Nom du check	Commande du check	Ligne de commande
Broker - \$KEY\$ - Alive	check_shinken_broker! alive! VALUE1\$	\$PLUGINDIR\$/check_shinken -H "\$HOSTADDRESS\$" -p "\$ARG2\$" --shinkenversion "\$SHINKENVERSION\$" -t broker -m \$ARG1\$
Broker - \$KEY\$ - Performance API Connection	check_shinken_broker! api_co nnection! VA LUE1\$	\$PLUGINDIR\$/check_shinken -H "\$HOSTADDRESS\$" -p "\$ARG2\$" --shinkenversion "\$SHINKENVERSION\$" -t broker -m \$ARG1\$
Broker - \$KEY\$ - Performance Modules Queues	check_shinken_broker! modul es_queue! V ALUE1\$	\$PLUGINDIR\$/check_shinken -H "\$HOSTADDRESS\$" -p "\$ARG2\$" --shinkenversion "\$SHINKENVERSION\$" -t broker -m \$ARG1\$
Broker - DB - Connection	check_shinken_db_connection	\$PLUGINDIR\$/check_shinkendb.py -D -H "\$HOSTADDRESS\$" -P "\$_HOSTDB_PORT\$" --connection-method "\$_HOSTDB_CONNECTION_METHOD\$" --ssh-user "\$_HOSTDB_SSH_USER\$" --ssh-keyfile "\$_HOSTDB_SSH_KEY\$" -A connect -W "\$_HOSTDB_CONNECT_WARN\$" -C "\$_HOSTDB_CONNECT_CRIT\$"
Broker - DB - Last Flush Time	check_shinken_db_last_flush	\$PLUGINDIR\$/check_shinkendb.py -D -H "\$HOSTADDRESS\$" -P "\$_HOSTDB_PORT\$" --connection-method "\$_HOSTDB_CONNECTION_METHOD\$" --ssh-user "\$_HOSTDB_SSH_USER\$" --ssh-keyfile "\$_HOSTDB_SSH_KEY\$" -A last_flush_time -W "\$_HOSTDB_LAST_FLUSH_TIME_WARN\$" -C "\$_HOSTDB_LAST_FLUSH_TIME_CRIT\$"
Broker - DB - Lock Time Percentage	check_shinken_db_lock_time	\$PLUGINDIR\$/check_shinkendb.py -D -H "\$HOSTADDRESS\$" -P "\$_HOSTDB_PORT\$" --connection-method "\$_HOSTDB_CONNECTION_METHOD\$" --ssh-user "\$_HOSTDB_SSH_USER\$" --ssh-keyfile "\$_HOSTDB_SSH_KEY\$" -A lock -W "\$_HOSTDB_LOCK_WARN\$" -C "\$_HOSTDB_LOCK_CRIT\$"
Broker - DB - Open Connections	check_shinken_db_open_connections	\$PLUGINDIR\$/check_shinkendb.py -D -H "\$HOSTADDRESS\$" -P "\$_HOSTDB_PORT\$" --connection-method "\$_HOSTDB_CONNECTION_METHOD\$" --ssh-user "\$_HOSTDB_SSH_USER\$" --ssh-keyfile "\$_HOSTDB_SSH_KEY\$" -A connections -W "\$_HOSTDB_CONNECTIONS_WARN\$" -C "\$_HOSTDB_CONNECTIONS_CRIT\$"