

# REFERENCE DES METRIQUES

## PRÉSENTATION

Le packaging Tick par BlueMind apporte la surveillance d'un grand nombre de données (appelées *métriques*). Certaines données surveillées sont brutes mais d'autres sont issues de pré-traitements afin d'offrir une plus grande pertinence et de faciliter leur interprétation et analyse.

Chaque métrique comporte une arborescence pouvant contenir :

- **dataLocation** : nom du serveur
- **host** : nom ou IP de l'hôte
- **meterType** : le type de donnée
  - *gauge* : mesure instantanée
  - *counter* : compteur à incrémentation
  - *distsum* : couple de données comportant un compteur et une quantité  
par exemple :
    - `bm-lmtpd.emailSize` = (nombre d'emails , taille totale des emails)
    - `bm-lmtpd.emailRecipients` = (nombre d'emails , nombre de destinataires)
  - *timer* : identique au *distsum* mais la quantité est ici toujours exprimée en nanosecondes
- **status** : selon le type de données il peut s'agir d'un statut ok/failed (requête aboutie/échouée par exemple), success/failure (authentification réussie/échouée par exemple), etc.

## DONNÉES COMMUNES

De façon générale, les métriques sont regroupées par composant.

### JVM

Pour chaque composant, vous pourrez trouver les métriques concernant la JVM :

- **bm-<composant>.hprof** : le nombre de fichier hprof existant sur la machine, permet de constater s'il y a eu un plantage
- **bm-<composant>.jvm.\*** : toutes les informations concernant la jvm pour ce composant (consommation mémoire courante, maximum, etc.)

### Heartbeat

Dans chaque composant ayant des interactions avec le core, on retrouvera les métriques suivantes qui servent à surveiller que les données de l'état de santé (*heartbeat* - littéralement « battement de cœur ») du core sont bien reçues par le composant :

Nom de la métrique	type	contenu	Informations complémentaires
heartbeat.receiver.age	Gauge	âge de la réception du dernier <i>heartbeat</i>	Correspond à la durée entre 2 <i>heartbeat</i> .  Le core est censé envoyer son état de santé toutes les 4 secondes, des durées supérieures, voire dépassant les 8 secondes, doivent donc alerter sur le bon état de fonctionnement de celui-ci
heartbeat.receiver.failures	Counter	nombre d'échecs de réception	
heartbeat.receiver.latency	Gauge	durée de réception du <i>heartbeat</i>	durée entre l'émission du <i>heartbeat</i> par le core et la réception par le composant
heartbeat.receiver.latencyMax	Gauge	durée de réception maximum	
heartbeat.receiver.received	Counter	nombre de réceptions OK	

### Hazelcast

Les serveurs membres du cluster hazelcast comportent la métrique suivante :

Nom de la métrique	type	contenu	Informations complémentaires
cluster.members	Gauge		la valeur de cette métrique doit être '3'

## LES MÉTRIQUES

Nom de la métrique	type	contenu	Informations complémentaires
agent.metricsGathered	Counter	nombre de métriques collectées par l'agent	cette métriques sert principalement à vérifier que l'agent est toujours en fonctionnement : l'absence de données indique que l'agent ne collecte plus rien, donc qu'il ne fonctionne plus
agent.vmware*		données concernant la machine hébergeant l'agent	L'agent est actif uniquement si les <i>vmware tools</i> sont détectés sur les machines hébergeant BlueMind. Dans cas les métriques du " <i>vSphere Guest SDK</i> " sont extraites et historisées.  Ces métriques permet de diagnostiquer des problèmes liés à la façon dont serait virtualisé BlueMind sur vmware.
bluemind.cluster			
bluemind.cluster.partitions			
<b>bm-core</b>	<b>moteur principal de BlueMind</b>		
callsCount	Counter	nombre d'appels reçus par le core	
dirVersion	Gauge		
directory.cluster.events	Counter		
handlingDuration	Timer	durée de traitement des requêtes	
heartbeat.broadcast	Counter		
heartbeat.maxPeriod	Gauge		
heartbeat.period	Gauge		
<b>bm-eas</b>	<b>service de connexion mobile</b>		
executionTime	Timer		
responseSize	DistSum		
<b>bm-hps</b>	<b>service d'authentification</b>		
authCount	Counter	nombre de demandes de connexions à BlueMind	<ul style="list-style-type: none"> <li>● status success : connexion réussie</li> <li>● status failed : connexion échouée (mauvais identifiant et/ou mot de passe)</li> </ul>
fitTemplates.requests	Counter	nombre de pages générées demandées	
requestsCount	Counter	nombre de requêtes servies par hps	<ul style="list-style-type: none"> <li>● kind : maintenance - affichage de la page de maintenance</li> <li>● kind : protected - affichage d'une page protégée</li> </ul> <p>Permet notamment de vérifier le nombre d'affichage effectifs de la page de maintenance. Trop de requêtes de type "maintenance" peut être le signe d'un problème.</p>
staticFile.requests	Counter	nombre de pages statiques demandées	exemple : page de login
upstreamRequestSize	DistSum	taille des requêtes	
upstreamRequestTime	Timer	durée de traitement des requêtes	
upstreamRequestsCount	Counter	nombre de requêtes	
<b>bm-ips</b>	<b>traçage des opérations IMAP</b>		
activeConnections	Gauge	nombre de connexions ips actives	
<b>bm-lmtpd</b>	<b>service de remise des emails</b>		
activeConnections	Gauge	nombre de connexions actives	
connectionCount	Counter		
deliveries	Counter		

emailRecipients	DistSum	nombre de destinataires par email	
emailSize	DistSum	taille des emails	
sessionDuration	Timer	durée de session	
traffic.transportLatency	Timer		
<b>bm-locator</b>	<b>localisation des services</b>		
executionTime	Timer	temps d'exécution des requêtes	
requestsCount	Counter	nombre de requêtes reçues par le service	<ul style="list-style-type: none"> <li>● origin : composant ayant émis la requête</li> <li>● statusCode : code retour http</li> </ul>
<b>bm-milter</b>	<b>Analyse et modification des emails au niveau SMTP</b>		
connectionsCount	Counter		
sessionDuration	Timer		
traffic.class	Counter		
traffic.size	Counter		
<b>bm-webserver</b>	<b>serveur d'application web</b>		
appCache.requestTime	Timer		
appCache.requests	Counter		
filTemplates.requests	Counter	nombre de requêtes d'affichage de pages générées par le webserver	
staticFile.requests	Counter	nombre de requêtes d'affichage de pages statiques	
<b>bm-xmpp</b>	<b>service de messagerie instantanée</b>		
packetsCount	Counter	nombre de paquets transmis par le service	sert à voir à quel point le service de messagerie est utilisé ainsi que s'il fonctionne correctement ou a cessé de fonctionner
<b>bm-ysnp</b>	<b>service de validation des données</b>		
authCount	Counter	nombre de demandes traitées	<ul style="list-style-type: none"> <li>● status ok : demandes validées (par exemple authentification acceptée pour le login/pass entré par une personne)</li> <li>● status failed : validation rejetée (par exemple authentification échouée car mot de passe incorrect)</li> </ul>
<b>Autres</b>			
cpu		données d'utilisation des processeurs	permet de contrôler l'utilisation et la répartition des processeurs
disk		données de gestion de l'espace disque	permet de contrôler l'espace disque utilisé/libre/total/etc. par disque, partition, chemin, etc.
diskio		nombre d'octets écrits/lus en temps réel	permet de voir si le disque travaille correctement ou trop
elasticsearch*		données concernant ElasticSearch	pour plus d'informations et de détails sur les métriques ES, voir la documentation dédiée <a href="https://github.com/influxdata/telegraf/tree/master/plugins/inputs/elasticsearch">https://github.com/influxdata/telegraf/tree/master/plugins/inputs/elasticsearch</a>
imapd.process			
influxdb*		données de la base de stockage des métriques	
kapacitor*		données concernant l'outil lui-même	
kernel			
kernel_vmstat			
mem			
memcached			
net			
netstat			
nginx			
phpfpm			

postfix_queue			
postgresql		informations sur la base de données de BlueMind	
processes			
swap			
syslog			
system			