

2 Agent

Aperçu

L'agent Zabbix est déployé sur une cible de surveillance pour superviser activement les ressources locales et les applications (disques durs, mémoire, statistiques de processeur, etc.).

L'agent rassemble les informations opérationnelles localement et transmet les données au serveur Zabbix pour un traitement ultérieur. En cas d'échec (par exemple, un disque dur plein ou un service en panne), le serveur Zabbix peut alerter activement les administrateurs de la machine particulière qui a signalé la panne.

Les agents Zabbix sont extrêmement efficaces en raison de l'utilisation d'appels système natifs pour collecter des informations statistiques.

Vérifications passives et actives

Les agents Zabbix peuvent effectuer des vérifications passives et actives.

Dans une [vérification passive](#), l'agent répond à une demande de données. Le serveur Zabbix (ou le proxy) demande des données, par exemple, la charge du processeur, et l'agent Zabbix renvoie le résultat.

Les [vérifications actives](#) nécessitent un traitement plus complexe. L'agent doit d'abord récupérer une liste d'éléments du serveur Zabbix pour un traitement indépendant. Ensuite, il enverra périodiquement de nouvelles valeurs au serveur.

Que ce soit pour effectuer des vérifications passives ou actives, cela se configure en sélectionnant un [type d'élément](#). L'agent Zabbix traite les éléments de type 'Agent Zabbix' ou 'Agent Zabbix (actif)'.

Plateformes supportées

L'agent Zabbix est supporté sur :

- Linux
- IBM AIX
- FreeBSD
- NetBSD
- OpenBSD
- HP-UX
- Mac OS X
- Solaris : 9, 10, 11
- Windows : toutes les versions bureautique et serveurs depuis XP

Agent sur les systèmes de type Unix

L'agent Zabbix sur les systèmes de type Unix s'exécute sur l'hôte qui est surveillé.

Installation

Voir la section [installation des packages](#) pour voir les instructions d'installation de l'agent Zabbix sous forme de package.

Sinon référez-vous aux instructions d'[installation manuelle](#) si vous ne voulez pas utiliser de packages.

En général, les agents Zabbix 32bits fonctionnent sur les systèmes 64bits, mais cela peut ne pas fonctionner dans certains cas.

Si installé comme package

L'agent Zabbix s'exécute comme un processus démon. L'agent peut être démarré en exécutant :

```
shell> service zabbix-agent start
```

Cela fonctionnera sur la plupart des systèmes GNU/Linux. Sur les autres systèmes vous devrez exécuter la commande suivante :

```
shell> /etc/init.d/zabbix-agent start
```

De même, pour arrêter/redémarrer/afficher l'état de l'agent Zabbix, utilisez les commandes suivantes :

```
shell> service zabbix-agent stop  
shell> service zabbix-agent restart  
shell> service zabbix-agent status
```

Démarrage manuel

Si les commandes précédentes ne fonctionnent pas, il faudra démarrer l'agent manuellement. Trouvez le chemin des binaires `zabbix_agentd` puis exécutez :

```
shell> zabbix_agentd
```

Agent sur les systèmes Windows

L'agent Zabbix sur Windows s'exécute en tant que service Windows.

Préparation

L'agent Zabbix est distribué sous forme d'archive au format zip. Après le téléchargement de l'archive vous devrez la décompresser. Vous pouvez choisir n'importe quel dossier pour stocker l'agent Zabbix et le fichier de configuration. ex :

C:\zabbix

Copiez les fichiers bin\win64\zabbix_agentd.exe et conf\zabbix_agentd.win.conf vers la destination choisie précédemment.

Editez le fichier c:\zabbix\zabbix_agentd.win.conf selon vos besoins en prenant soin de spécifier correctement le paramètre "Hostname".

Installation

Une fois que les étapes précédentes sont réalisées, exécutez la commande suivante pour installer l'agent Zabbix en tant que service :

```
C:\> c:\zabbix\zabbix_agentd.exe -c c:\zabbix\zabbix_agentd.win.conf -i
```

Maintenant vous devriez être capable de configurer le service "Agent Zabbix" comme les autres services Windows.

Voir [plus de détails](#) sur l'installation et l'exécution de l'agent Zabbix sur Windows.

Autres options de l'agent

Il est possible d'exécuter plusieurs instances de l'agent sur un hôte. Une seule instance peut utiliser le fichier de configuration par défaut ou un fichier de configuration spécifié dans la ligne de commande. Dans le cas d'instances multiples, chaque instance d'agent doit posséder son propre fichier de configuration (l'une des instances peut utiliser le fichier de configuration par défaut).

Les paramètres de ligne de commande suivants peuvent être utilisés avec l'agent Zabbix :

Paramètre	Description
Agent UNIX et Windows	
-c --config <config-file>	Chemin vers le fichier de configuration. Vous devez utiliser cette option pour spécifier le fichier de configuration qui n'est pas celui par défaut. Sur UNIX, le fichier par défaut est /usr/local/etc/zabbix_agentd.conf ou comme appliqué par les variables compile-time --sysconfdir ou --prefix Sur Windows, le fichier par défaut est c:\zabbix_agentd.conf
-p --print	Affiche les éléments connus et quitte. <i>Note:</i> Pour retourner également les résultats des paramètres utilisateur , vous devez spécifier le fichier de configuration (s'il n'est pas à l'emplacement par défaut).
-t --test <item key>	Teste l'élément spécifié et quitte. <i>Note:</i> Pour retourner également les résultats des paramètres utilisateur , vous devez spécifier le fichier de configuration (s'il n'est pas à l'emplacement par défaut).
-h --help	Affiche les informations d'aide
-V --version	Affiche le numéro de version
Agent UNIX uniquement	

Paramètre	Description
Agent UNIX et Windows	
-R --runtime-control <option>	Effectue des fonctions administratives. Voir runtime control .
Agent Windows uniquement	
-m --multiple-agents	Utilise plusieurs instances d'agent (avec les fonctions -i,-d,-s,-x). Pour distinguer les noms des services des noms des instances, chaque nom de service inclura la valeur Hostname du fichier de configuration spécifié.
Agent Windows uniquement (fonctions)	
-i --install	Installe l'agent Windows Zabbix en tant que service
-d --uninstall	Désinstalle le service de l'agent Windows Zabbix
-s --start	Démarre le service de l'agent Windows Zabbix
-x --stop	Arrête le service de l'agent Windows Zabbix

Exemples spécifiques d'utilisation des paramètres de ligne de commande :

- affichage de tous les éléments d'agent intégrés avec des valeurs
- tester un paramètre utilisateur avec la clé "mysql.ping" définie dans le fichier de configuration spécifié
- installation d'un service "Agent Zabbix" pour Windows en utilisant le chemin par défaut du fichier de configuration c:\zabbix_agentd.conf
- installation d'un service "Agent Zabbix [Hostname]" pour Windows en utilisant le fichier de configuration zabbix_agentd.conf situé dans le même dossier que l'exécutable de l'agent et rend le nom de service unique en l'étendant par la valeur 'Hostname' du fichier config

```
shell> zabbix_agentd --print
shell> zabbix_agentd -t "mysql.ping" -c /etc/zabbix/zabbix_agentd.conf
shell> zabbix_agentd.exe -i
shell> zabbix_agentd.exe -i -m -c zabbix_agentd.conf
```

Runtime control

Avec les options de runtime control, vous pouvez changer le niveau de journalisation des processus de l'agent.

Option	Description	Cible
log_level_increase[=<target>]	Augmente le niveau de journalisation. Si la cible n'est pas spécifiée, tous les processus sont affectés.	La cible peut être spécifiée de la manière suivante : : process type - tous les processus du type spécifié (ex : listener) Voir tous les types de processus de l'agent . process type,N - type de processus et numéro (ex : listener,3)
log_level_decrease[=<target>]	Diminue le niveau de journalisation. Si la cible n'est pas spécifiée, tous les processus sont affectés.	pid - identifiant de processus (1 à 65535). Pour les valeurs plus large spécifier la cible comme 'process-type,N'.

Notez que la plage utilisable de PID pour modifier le niveau de journalisation d'un processus d'un seul agent est comprise entre 1 et 65535. Sur les systèmes dotés de PIDs > 65535, la cible <process type, N> peut être utilisée comme solution de contournement pour modifier le niveau de journalisation d'un seul processus.

Exemples :

- augmente le niveau de journalisation de tous les processus
- augmente le niveau de journalisation du troisième processus listener
- augmente le niveau de journalisation du processus avec le PID 1234
- diminue le niveau de journalisation de tous les processus nommés "active checks"

```
shell> zabbix_agentd -R log_level_increase
shell> zabbix_agentd -R log_level_increase=listener,3
shell> zabbix_agentd -R log_level_increase=1234
shell> zabbix_agentd -R log_level_decrease="active checks"
```

Runtime control n'est pas supporté sur OpenBSD, NetBSD et Windows.

Types de processus Agent

- □active checks - processus qui effectue les vérifications actives
- □collector - processus pour la collecte des données
- □listener - processus à l'écoute des vérifications passives

Le fichier journal de l'agent peut être utilisé pour observer ces types de processus

Processus utilisateur

L'agent Zabbix sous UNIX est conçu pour s'exécuter en tant qu'utilisateur non root. Il fonctionnera comme n'importe quel utilisateur non-root. Vous pouvez donc exécuter l'agent comme n'importe quel utilisateur non root sans aucun problème.

Si vous essayez de l'exécuter en tant que 'root', il passera à un utilisateur 'zabbix' codé en dur, qui doit être présent sur votre système. Vous pouvez uniquement exécuter l'agent en tant que 'root' si

vous modifiez le paramètre 'AllowRoot' dans le fichier de configuration de l'agent associé.

Fichier de configuration

Pour plus de détails sur la configuration de l'agent Zabbix, voir les options du fichier de configuration pour [zabbix_agentd](#) ou l'[agent Windows](#).

Environnement local

Notez que l'agent requiert un environnement local UTF-8 afin que certains éléments textuels d'agent puissent renvoyer le contenu attendu. La plupart des systèmes modernes de type Unix ont un paramètre régional UTF-8 par défaut, cependant, certains systèmes peuvent avoir besoin d'être définis spécifiquement.

Code de sortie

Avant la version 2.2, l'agent Zabbix renvoyait 0 en cas de sortie réussie et 255 en cas d'échec. A partir de la version 2.2 et supérieure, l'agent Zabbix renvoie 0 en cas de réussite et 1 en cas d'échec.

From:

<https://www.zabbix.com/documentation/4.0/> - **Zabbix Documentation 4.0**

Permanent link:

<https://www.zabbix.com/documentation/4.0/fr/manual/concepts/agent>

Last update: **2019/01/30 13:03**

