

PROCEDURE D'INSTALLATION:

ZABBIX – HA SERVEUR



Introduction

- **Objectif :** L'objectif du HA (High Availability) pour Zabbix, c'est d'assurer que ton système de supervision reste disponible et opérationnel en continu, même en cas de panne d'un composant (serveur, base de données, etc.).

Prérequis

- **Système d'exploitation**
 - **Debian 12, Ubuntu 22.04, CentOS, RHEL, AlmaLinux, Rocky Linux**
- **Ressources matérielles (minimum recommandé)**
 - **2 serveurs zabbix en fonction**
 - **Cluster Galera avec HAPROXY en frontale opérationnelle.**
 - **Proxy zabbix (Facultatif)**
 - **Frontend (obligatoire pour la gestion des 2 serveurs zabbix**

Zabbix HA Nom du nœud

Laissez's jeter un oeil à la *HANodeName* paramètre. C'est le paramètre de configuration le plus important – il est obligatoire de le spécifier si vous souhaitez exécuter votre serveur Zabbix en mode haute disponibilité.

- Ce paramètre est utilisé pour spécifier le nom du mode de cluster particulier
- Si le *HANodeName* n'est pas spécifié, le serveur Zabbix ne démarre pas en mode cluster
- Le nom du nœud doit être unique sur chacun de vos nœuds

```
# nano /etc/zabbix/zabbix_server.conf
```

```
## Option: HANodeName
# The high availability cluster node name.
# When empty server is working in standalone mode.
HANodeName=zbx-node1
```



Dans notre exemple, nous pouvons observer un cluster à deux nœuds, où zbx-node1 est le nœud actif et zbx-node2 est le nœud de secours. Ces deux nœuds seront **envoient leurs battements de cœur au backend de la base de données Zabbix** toutes les 5 secondes. Si un nœud cesse d'envoyer son rythme cardiaque, un autre nœud prendra le relais.

Zabbix HA Adresse du nœud

Le deuxième paramètre que vous devrez également spécifier est le NodeAdresse paramètre.

```
## Option: NodeAddress
#     IP or hostname to define hoe frontend should connect to the server.
#     Format: <address>[:port]
NodeAddress=node1.example.com
```

Dans notre exemple, nous utilisons l'adresse node1.exemple.com. Le but de ce paramètre est de indiquer à l'interface Zabbix l'adresse du serveur Zabbix actuellement actif étant donné que le composant frontal Zabbix communique également en permanence avec le composant serveur Zabbix. Si ce paramètre n'est pas spécifié, le frontend Zabbix risque de ne pas pouvoir se connecter au nœud de serveur Zabbix actif.

Configuration frontale zabbix

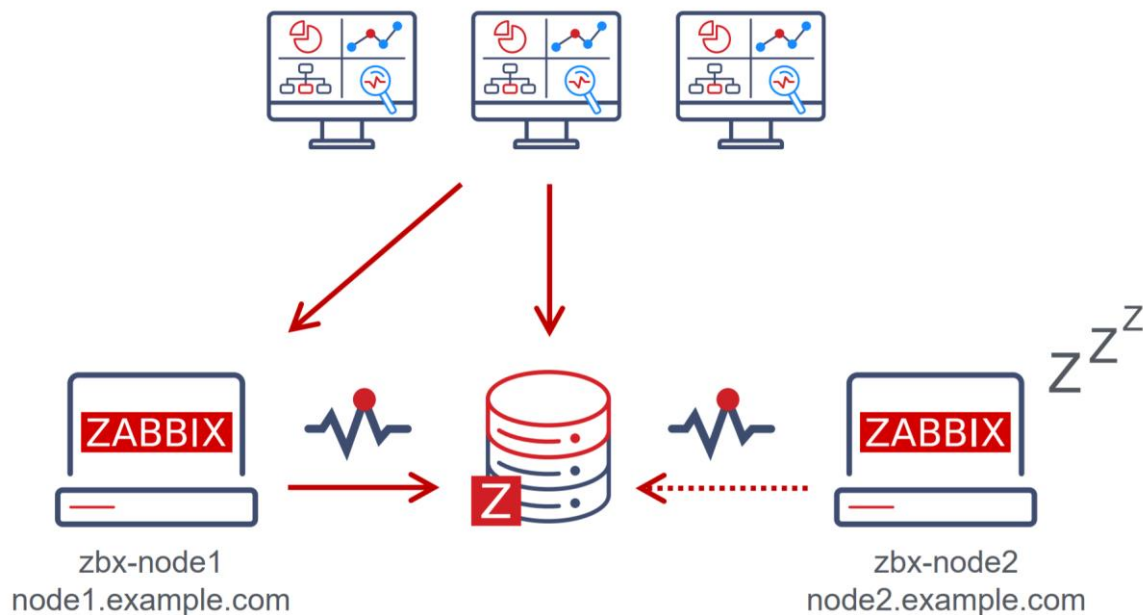
Les utilisateurs chevronnés de Zabbix savent peut-être que le frontend Zabbix a son propre fichier de configuration, qui contient généralement l'adresse du serveur Zabbix et le port serveur Zabbix pour établir des connexions du frontend Zabbix au serveur Zabbix.

```
# nano /etc/zabbix/web/zabbix.conf.php
```

```
// Uncomment and set to desired values to override Zabbix hostname/IP and port.
// $ZBX_SERVER                = '';
// $ZBX_SERVER_PORT            = '';
```

Si vous utilisez le cluster de haute disponibilité Zabbix, alors **vous devrez commenter ces paramètres** comme au lieu d'être statiques, ils dépendent maintenant du nœud de serveur Zabbix actuellement actif et seront obtenus à partir de la base de données backend Zabbix.

Mettre tout cela ensemble



Dans l'exemple ci-dessus, nous pouvons voir que nous avons deux nœuds – **zbx-node1**, qui est actuellement actif et **zbx-node2**. Ces nœuds peuvent être accessibles en utilisant les adresses externes – `node1.example.com` et `node2.example.com` pour **zbx-node1** et **zbx-node2** respectivement. Nous pouvons voir que nous avons également déployé plusieurs frontends. Chacun de ces nœuds frontaux se connectera à la base de données backend Zabbix, lira l'adresse du nœud actuellement actif et procédera à la connexion à ce nœud.

Types de nœuds Zabbix HA

Les nœuds de cluster de haute disponibilité du serveur Zabbix peuvent avoir l'un des statuts multiples suivants:

- **Active** – Le nœud actuellement actif. Un seul nœud peut être actif à la fois
- **Standby** – Le nœud est en cours d'exécution en mode veille. Plusieurs nœuds peuvent avoir ce statut
- **Arrêt** – Le nœud a déjà été détecté, mais il a été fermé avec grâce

- Unreachable – Node a été précédemment détecté mais a été perdu de manière inattendue sans arrêt. Cela peut être causé par de nombreuses raisons différentes, par exemple – le nœud se bloque ou ayant des problèmes de réseau

Dans des circonstances normales, vous aurez un nœud actif et un ou plusieurs nœuds de secours. Les nœuds en mode arrêt sont également attendus si, par exemple, vous effectuez certaines tâches de maintenance sur ces nœuds. D'autre part, si un nœud actif devient inaccessible, c'est à ce moment qu'un des nœuds de secours prendra le relais.

Statut du nœud Zabbix HA

Le widget Informations système a reçu quelques modifications dans Zabbix 6.0 LTS. Il est maintenant capable d'afficher l'état de votre cluster de haute disponibilité de serveur Zabbix et de ses nœuds individuels.

System information			
Parameter	Value	Details	
Zabbix server is running	Yes	zbx-node1:10051	
Number of hosts (enabled/disabled)	1	1 / 0	
Number of templates	288		
Number of items (enabled/disabled/not supported)	97	88 / 0 / 9	
Number of triggers (enabled/disabled [problem/ok])	55	55 / 0 [1 / 54]	
Number of users (online)	2	1	
Required server performance, new values per second	1.42		
High availability cluster	Enabled	Fail-over delay: 1 minute	
Name	Address	Last access	Status
zbx-node1	zbx-node1:10051	5s	Active
zbx-node3	zbx-node3:10051	3s	Stopped
zbx-node2	zbx-node2:10051	4s	Standby

Le widget peut afficher le **mode cluster actuel**, qui est activé dans notre exemple et **fournit une liste de tous les nœuds de cluster**. Dans notre exemple, nous pouvons voir que nous avons 3 nœuds – 1 nœud actif, 1 nœud arrêté et 1 nœud en mode veille. De cette façon, nous ne pouvons pas seulement voir le **statut de nos nœuds mais aussi leurs noms, adresses et derniers temps d'accès**.

Commutation du nœud Zabbix HA

La commutation entre les nœuds se fait manuellement. Une fois que vous arrêtez le nœud de serveur Zabbix actuellement actif, un autre nœud prendra automatiquement le relais. Bien sûr, vous devez avoir au moins un autre nœud en état de veille, afin qu'il puisse prendre le relais du nœud actif défaillant.

```
# systemctl stop zabbix-server.service
```

```
systemctl stop zabbix-server
```



Comment fonctionne le basculement ?

Tous les nœuds signalent leur statut toutes les 5 secondes. Chaque fois que vous arrêtez un nœud, il passe dans un état d'arrêt et en 5 secondes, un autre nœud prendra le relais. Mais si un nœud échoue, le flux de travail est un peu différent. C'est là que quelque chose s'appelle un **retard de basculement** est pris en compte. **Par défaut, ce délai de basculement est de 1 minute.** Le nœud de secours attendra une minute pour que le nœud actif défaillant mette à jour son statut et si, dans une minute, le nœud actif n'est toujours pas visible, le nœud de secours prendra le relais.

Connexion de proxys Zabbix à votre cluster

La configuration du proxy est très similaire à la configuration de l'agent. Encore une fois – nous pouvons avoir un proxy en cours d'exécution soit en mode passif ou en mode actif.

Pour les proxys Zabbix passifs, nous devons **listez nos nœuds de cluster sous le *Serveur* paramètre dans le fichier de configuration du proxy**. Ces nœuds doivent être spécifiés dans un **liste séparée par des virgules**. De cette façon, les proxys accepteront les connexions de n'importe quel nœud de serveur Zabbix.

```
### Option: Server
# List of comma delimited IP addresses or DNS names of Zabbix server.
# Incoming connections will be accepted only from the addresses listed here.
Server=zbx-node01,zbx-node02
```

En ce qui concerne les proxies Zabbix actifs –, nous devons à nouveau répertorier nos nœuds sous le *Serveur* paramètre, mais cette fois les noms de nœuds seront séparés par des points-virgules.

```
### Option: Server
# IP address or DNS name (address:port) or cluster (address:port;address2:port) of
# Zabbix server to get configuration data from and send data to.
# Cluster nodes need to be separated by semicolon.
Server=zbx-node01;zbx-node02
```