Zabbix et OpenVMS

Zabbix et OpenVMS

- Cette présentation sera découpée en plusieurs parties :
  - Qu'est-ce que Zabbix
  - L'intégration de systèmes OpenVMS dans Zabbix
  - Exemples d'implémentation chez plusieurs clients
  - Questions / réponses

L'intégration d'OpenVMS dans Zabbix

Cette présentation est une courte introduction à ce qu'est Zabbix

L’intégration d’OpenVMS dans Zabbix

Cette présentation décrit les outils Zabbix pour OpenVMS : l’agent, le ‘sender’ commande DCL ou shareable, etc…

Zabbix et OpenVMS

- SysGroup développe depuis plus de 10 ans un environnement Zabbix (client) pour OpenVMS
- Cet environnement comprend :
  - Un agent lancé sous la forme d'un process détaché
  - une commande DCL zabbix_sender
  - une routine `zabbix_send` contenue dans la *shareable* `zabbix_sender_shr32.exe`
- Il est, donc, possible d'envoyer des alarmes Zabbix à partir d'un script DCL ou à partir d'une API dans tout programme.
- Exemples :
  - `$ zabbix_sender/key=keya/value=1/host=hostn/change`
  - `zabbix_send('keya', '1')`

Zabbix et OpenVMS

- Fonctionne sur Itanium/Alpha, une version supportant moins d'items existe pour VAX
- L'agent ZABBIX pour OpenVMS s'exécute dans un process détaché
- Plus de 70 items sont supportés
  - Paramétrages/Informations systèmes, par exemple
    - Taille de la mémoire, nombre de CPUs
    - Nombre maximum de process concurrents
    - Nombre de bufio/dirio
    - Nombre de process par type
    - Consommation CPU, moyenne sur 1, 5, 15 minutes
    - Etc...

Zabbix et OpenVMS

- Items supportés, suite
  - Informations « process »
    - Présence d'un process
    - Nom de l'image
    - Quotas : limite et consommation
    - Etc…
  - Informations « device »
    - Nombre d'erreurs
  - Informations disque
    - Espace disponible
    - Nombre de membres présents dans un « shadow »

Zabbix et OpenVMS

- Items supportés, suite
  - Informations fichier
    - Présence d'un fichier
    - Nombre, allocation, taille
    - Il est possible d'utiliser des « wildcards »
  - Information « queue manager »
    - Status du « queue manager »
    - Nœud sur lequel tourne le « queue manager »
    - « job_limit » de la « queue »
    - Status d'une « queue »
    - Nombre de jobs dans une « queue » pour un status

Zabbix et OpenVMS

- Items supportés, suite
  - Informations réseau
    - Accessibilité d'un système à partir de la machine VMS
    - État et utilisation d'une interface réseau
  - Informations sur les « logicals »
    - Présence d'un nom logique
    - Valeur d'un nom logique

Zabbix et OpenVMS

- L'item vms.generic.discovery
  - Permet à partir de fichier JSON de générer automatiquement des items/triggers/graphes,...
  - Exemple, vérifier la présence ou la valeur de noms logiques :

```
$ type ZABBIX_LOCAL_DIR:LOGICAL_DISCOVERY_GEN.JSON
{"data": [
  {"{#LNAME001}":"LOGICAL1",
   "{#LTABLE001}":"LNM$SYSTEM_TABLE"},
  {"{#LNAME002}":"LOGICAL2",
   "{#LTABLE002}":"APPLI_TABLE",
   "{#LVALUE002}":"1"}]
}
$
```

Zabbix et OpenVMS

- L'item vms.generic.discovery, suite
  - Exemple, suite :
    - Ajout d'une « discovery rule » vms.generic.discovery[LOGICAL_DISCOVERY_GEN]
    - Définition des prototypes d'items
      `logical.exists[{#LNAME001},{#LTABLE001}]`
      `logical.exists[{#LNAME002},{#LTABLE002}]`
      `logical.value[{#LNAME002},{#LTABLE002}]`
    - Définition des prototypes de triggers :
      - `{mytemplate VMS:logical.exists[{#LNAME001},{#LTABLE001}].last()}=0`
      - `{mytemplate VMS:logical.exists[{#LNAME002},{#LTABLE002}].last()}=0`
        `or`
        `{mytemplate VMS:logical.value[{#LNAME002},{#LTABLE002}].last()}<>{#LVALUE002}`

Zabbix et OpenVMS

- L'envoi d'informations à Zabbix à partir de DCL
  - L'exécutable `ZABBIX_SENDER` supporte les qualifieurs suivants :
    - `/key=nom_item`
    - `/value=valeur_item`
    - `/host=nom_host`
    - `/change`
      - N'envoie la valeur à Zabbix que si celle-ci a changé
      - Le valeur est stockée dans un nom logique dans la table `zabbix_table_mon`

Zabbix et OpenVMS

- L'envoi d'information à Zabbix à partir d'un programme
  - La fonction `ZABBIX_SEND` incluse dans la « shareable » `ZABBIX_SENDER_SHR32.EXE`
  - Reçoit les mêmes arguments que la commande DCL
    - Nom de l'item
    - Valeur de l'item
    - Nom du « host »
    - Non envoi si la valeur n'a pas changé

Retours d’utilisateurs

Zabbix chez quelques utilisateurs OpenVMS

Zabbix et OpenVMS

- Supervision métier
  - Plusieurs centaines d'items surveillés
- Exemple de tableau de bord :

Zabbix et OpenVMS

- Autres implémentations
  - Supervision « classique » OS, SAN, réseau, exemples :
    - Erreurs matériels
    - Charge CPU, disques pleins, …
    - Changement de path SAN pour un disque
  - Supervision de l'exploitation, exemples :
    - Tuxedo
      - Nombre de process actifs par services
    - CFT
      - Transferts échoués
    - L'ordonnanceur sched
      - sched en erreur ou en cours d'exécution
    - Rdb
      - Présence de bugcheck

Zabbix et OpenVMS

- Obtenir plus d'informations :
  - https://zabbix.sysgroup.fr/
  - mailto:zabbix-openvms@sysgroup.fr

Zabbix et OpenVMS

Zabbix questions