

Prozentsatz des Grenzwerts für die benutzerdefinierte Datenbankfragmentierung in CPS konfigurieren

Inhalt

[Einleitung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Konfigurieren](#)

[Konfigurationen](#)

[Ansatz für in OpenStack gehostete CPS](#)

[Ansatz für CPS gehostet in VMware](#)

[Überprüfung](#)

[Fehlerbehebung](#)

Einleitung

In diesem Dokument wird beschrieben, wie Sie in Cisco Policy Suite (CPS) einen benutzerdefinierten Schwellenwert für die Datenbankfragmentierung konfigurieren.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- Linux
- CPS
- MongoDB

Hinweis: Cisco empfiehlt, dass Sie über die Berechtigung Root-Benutzerzugriff auf die CPS CLI verfügen.

Verwendete Komponenten

Die Informationen in diesem Dokument basierend auf folgenden Software- und Hardware-Versionen:

- CPS 20.2
- Unified Computing System (UCS)-B
- MongoDB v3.6.17

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle verstehen.

Hintergrundinformationen

CPS verwendet MongoDB, um seine grundlegende Datenbankstruktur (DB) zu bilden.

Fragmentierung ist ein Merkmal von MongoDB. Mit dieser Warnung überwachen Sie die Fragmentierung von MongoDB proaktiv und vermeiden so eine höhere Ressourcenauslastung (Festplatte und Arbeitsspeicher) durch MongoDB.

CPS generiert einen Simple Network Management Protocol (SNMP) Alarm, wenn der MongoDB Fragmentierungsprozentsatz einen festgelegten Wert überschreitet.

Die Fehlermeldung `/etc/collectd.d/dbMonitorList.cfg` -Datei, die auf den virtuellen Maschinen (VMs) von `sessionmgr` vorhanden ist, enthält die Liste der Datenbanken und die entsprechenden prozentualen Fragmentierungsschwellenwerte.

Konfigurieren

Konfigurationen

Ansatz für in OpenStack gehostete CPS

Schritt 1: Führen Sie auf der virtuellen Cluster Manager-Maschine diesen Befehl aus, um eine Sicherung der aktuellen Konfigurationsdatei zu erstellen.

```
#cp /etc/puppet/modules/qps/templates/collectd_worker/collectd.d/dbMonitorList.cfg  
/etc/puppet/modules/qps/templates/collectd_worker/collectd.d/dbMonitorList.cfg.bkp
```

Schritt 2: Führen Sie diesen Befehl im Cluster Manager aus, um die aktuelle Konfiguration von `sessionmgr` VMs abzurufen (zum Vergleichen und Validieren nach der Änderung).

```
#for host in $(hosts-all.sh | grep 'sessionmgr'); do echo checking in $host; ssh $host "cat  
/etc/collectd.d/dbMonitorList.cfg"; done
```

Beispiel für das Ergebnis:

```
checking in sessionmgr01  
session_cache|session|40  
sk_cache|secondary_key|40  
diameter|endpoints|40  
spr|subscriber|40  
balance_mgmt|account|40  
checking in sessionmgr02  
session_cache|session|40  
sk_cache|secondary_key|40  
diameter|endpoints|40  
spr|subscriber|40  
balance_mgmt|account|40
```

Schritt 3: Ändern Sie den Standard-Schwellenwert (40) auf den empfohlenen Wert (z. B. 60). Führen Sie diesen Befehl im Cluster-Manager aus.

Hinweis: Mit diesem Befehl wird der Grenzwert für alle DBs geändert. Wenn der Schwellenwert für eine bestimmte Datenbank aktualisiert werden muss, muss die Datei manuell aktualisiert werden.

```
#sed -i 's/40/60/g'  
/etc/puppet/modules/qps/templates/collectd_worker/collectd.d/dbMonitorList.cfg
```

Schritt 4: Führen Sie diesen Befehl aus, um die Dateien im Cluster-Manager zu vergleichen und die Änderung zu überprüfen.

```
#diff /etc/puppet/modules/qps/templates/collectd_worker/collectd.d/dbMonitorList.cfg  
/etc/puppet/modules/qps/templates/collectd_worker/collectd.d/dbMonitorList.cfg.bkp
```

Beispiel für das Ergebnis:

```
4c4  
<session_cache|session|60  
---  
>session_cache|session|40  
9c9  
<sk_cache|secondary_key|60  
---  
>sk_cache|secondary_key|40  
14c14  
<diameter|endpoints|60  
---  
>diameter|endpoints|40  
19c19  
<spr|subscriber|60  
---  
>spr|subscriber|40  
24c24  
<balance_mgmt|account|60  
---  
>balance_mgmt|account|40
```

Schritt 5: Führen Sie diesen Befehl aus, um die Änderung im Cluster-Manager zu erstellen.

```
[root@installer ~]# /var/qps/bin/build/build_puppet.sh  
Copying /etc/puppet to /var/qps/images/puppet.tar.gz...  
Creating MD5 Checksum...  
[root@installer ~]#
```

Schritt 6: Führen Sie diesen Befehl im Cluster Manager aus, um die Änderung auf die virtuellen Sitzungsmgr-Rechner anzuwenden.

```
[root@installer ~]# for host in $(hosts-all.sh | grep 'sessionmgr'); do echo starting vm-init in  
$host; ssh $host "/etc/init.d/vm-init > /dev/null 2>&1 &"; done  
starting vm-init in sessionmgr01  
starting vm-init in sessionmgr02  
[root@installer ~]#
```

Schritt 7. Warten Sie, bis die Puppe fertig ist. Führen Sie diesen Befehl im Cluster Manager aus, um den Fortschritt der Puppet-Konfiguration anzuzeigen.

```
#for host in $(hosts-all.sh | grep 'sessionmgr' | tail -1); do echo checking in $host; ssh $host  
"tail -f /var/log/puppet.log"; done
```

```
2022-11-08 06:32:23 +0000 Service[whisper](provider=cps) (info): whisper will be managed using  
monit.
```

```
2022-11-08 06:32:23 +0000 Service[whisper](provider=cps) (info): whisper will be managed using  
monit.
```

```
2022-11-08 06:32:23 +0000 /Stage[main]/Whisper/Service[whisper] (notice): Triggered 'refresh'  
from 1 event
```

```
2022-11-08 06:32:27 +0000 Stage[main] (info): Unsheduling all events on Stage[main]
```

```
2022-11-08 06:32:28 +0000 Puppet (notice): Applied catalog in 83.52 seconds
```

```
[Tue Nov 08 06:32:30 +0000 2022] * Completed puppet configuration for dcl-sessionmgr02...
```

```
[Tue Nov 08 06:32:30 +0000 2022] - NTP sync started, check the logs in vm-init.log
```

Ansatz für CPS gehostet in VMware

Schritt 1: Aktualisieren Sie `/var/qps/config/deploy/csv/Configuration.csv` in Cluster Manager mit dem erforderlichen Datenbanknamen und dem entsprechenden prozentualen Schwellenwert abgelegt. Das Format für den benutzerdefinierten Schwellenwert ist wie folgt (wobei XX der numerische Wert von Prozentsatz ist...Beispiel: 60).

```
session_cache,XX,  
sk_cache,XX,  
diameter,XX,  
spr,XX,  
balance_mgmt,XX,
```

Konfigurationsbeispiel:

```
session_cache,60,  
sk_cache,60,  
diameter,60,  
spr,60,  
balance_mgmt,60,
```

Schritt 2: Führen Sie diese Befehle aus, um `/etc/collectd.d/dbMonitorList.cfg`, sodass sie über die neuen Schwellenwerte aus der Datei Configuration.csv verfügt:

```
[root@installer ~]# /var/qps/install/current/scripts/import/import_deploy.sh  
Filenames that will be processed
```

```
AdditionalHosts.csv Configuration.csv DBConfigServer.csv Definitions.csv Hosts.csv  
ReplicationSets.csv SessionCache.csv VLANs.csv VMSpecification.csv SecureConfig.csv  
VipProxyConfiguration.csv DSCPConfig.csv CriticalFiles.csv
```

```
The CSV files in /var/qps/config/deploy/csv are converted to json files in  
/var/qps/config/deploy/json..
```

```
build the hosts file to /var/www/html/hosts...
```

```
build the /etc/hosts file from the json configuration... /etc/hosts is backed to /etc/hosts.back
```

```
Skipping backup of '/etc/hosts' -- no changes detected.
```

```
Redis by default disabled -DenableQueueSystem=false in /etc/broadhop/qns.conf
```

```
Removing feature configs moved to core
```

```
Removing ws feature from pb and pcrf feature file
```

```
Building /etc/broadhop...
```

```
Copying to /var/qps/images/etc.tar.gz...
```

```
Creating MD5 Checksum...
```

```
Generating /etc/broadhop/servers.all
```

```
Rebuilding facts for: 'installer' (aka 'installer')
```

```
Creating md5sum for hosts file to validate later
Rebuilding facts for: 'dcl-lb01' (aka 'lb01')
Rebuilding facts for: 'dcl-sessionmgr01' (aka 'sessionmgr01')
Rebuilding facts for: 'dcl-lb02' (aka 'lb02')
Rebuilding facts for: 'dcl-qns01' (aka 'qns01')
Rebuilding facts for: 'dcl-qns02' (aka 'qns02')
Rebuilding facts for: 'dcl-pcrfclient01' (aka 'pcrfclient01')
Rebuilding facts for: 'dcl-sessionmgr02' (aka 'sessionmgr02')
Rebuilding facts for: 'dcl-pcrfclient02' (aka 'pcrfclient02')
No file for VipProxyConfiguration found
Copying /etc/puppet to /var/qps/images/puppet.tar.gz...
Creating MD5 Checksum...
[root@installer ~]#
```

Schritt 3: Führen Sie diesen Befehl im Cluster Manager aus, um die Änderung auf die virtuellen Sitzungsmgr-Rechner anzuwenden.

```
[root@installer ~]# for host in $(hosts-all.sh | grep 'sessionmgr'); do echo starting vm-init in
$host; ssh $host "/etc/init.d/vm-init > /dev/null 2>&1 &"; done
starting vm-init in sessionmgr01
starting vm-init in sessionmgr02
[root@installer ~]#
```

Schritt 4: Warten Sie, bis die Puppe fertig ist. Führen Sie diesen Befehl im Cluster Manager aus, um den Fortschritt der Puppet-Konfiguration anzuzeigen.

```
#for host in $(hosts-all.sh | grep 'sessionmgr' | tail -1); do echo checking in $host; ssh $host
"tail -f /var/log/puppet.log"; done

2022-11-08 06:48:34 +0000 Service[whisper](provider=cps) (info): whisper will be managed using
monit.
2022-11-08 06:48:34 +0000 Service[whisper](provider=cps) (info): whisper will be managed using
monit.
2022-11-08 06:48:34 +0000 /Stage[main]/Whisper/Service[whisper] (notice): Triggered 'refresh'
from 1 event
2022-11-08 06:48:39 +0000 Stage[main] (info): Unscheduling all events on Stage[main]
2022-11-08 06:48:40 +0000 Puppet (notice): Applied catalog in 93.27 seconds
[Tue Nov 08 06:48:42 +0000 2022] * Completed puppet configuration for dcl-sessionmgr02...
[Tue Nov 08 06:48:42 +0000 2022] - NTP sync started, check the logs in vm-init.log
```

Überprüfung

Verwenden Sie diesen Abschnitt, um zu überprüfen, ob Ihre Konfiguration ordnungsgemäß funktioniert.

Validieren Sie die aktuelle Konfiguration in den virtuellen Session-Manager-Systemen, und vergleichen Sie sie mit der Ausgabe aus Schritt 2. Führen Sie diesen Befehl im Cluster-Manager aus.

```
[root@installer ~]# for host in $(hosts-all.sh | grep 'sessionmgr'); do echo checking in $host;
ssh $host "cat /etc/collectd.d/dbMonitorList.cfg"; done
checking in sessionmgr01
session_cache|session|60
sk_cache|secondary_key|60
diameter|endpoints|60
spr|subscriber|60
balance_mgmt|account|60
checking in sessionmgr02
```

```
session_cache|session|60
sk_cache|secondary_key|60
diameter|endpoints|60
spr|subscriber|60
balance_mgmt|account|60
[root@installer ~]#
```

Fehlerbehebung

Dieser Abschnitt enthält Informationen, die Sie zur Fehlerbehebung bei Ihrer Konfiguration verwenden können.

Dieser MongoDB-Fragmentierungsalarm wurde in 20.1 eingeführt und in früheren Versionen nicht gemessen. Standardmäßig beträgt der Fragmentierungsschwellenwert 40 %. Dieser Schwellenwert muss in Abhängigkeit von der Bereitstellungsgröße, den Datenverkehrsmustern (Anrufmodellen) und anderen Datenverkehrsmusterfaktoren geändert werden. Andernfalls gibt CPS Warnungen über Schwellenwertverletzungen bei der Datenbankfragmentierung aus (keine/nicht erwünschte).

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.