

# 在CPS中配置自定義資料庫碎片閾值百分比

## 目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[設定](#)

[組態](#)

[OpenStack中託管的CPS方法](#)

[VMWare環境下的CPS託管方法](#)

[驗證](#)

[疑難排解](#)

## 簡介

本文說明如何在Cisco Policy Suite(CPS)中設定自訂資料庫分段閾值百分比。

## 必要條件

### 需求

思科建議您瞭解以下主題：

- Linux
- CPS
- MongoDB

**注意：** 思科建議您具有對CPS CLI的超級使用者訪問許可權。

### 採用元件

本文中的資訊係根據以下軟體和硬體版本：

- CPS 20.2
- 整合運算系統(UCS)-B
- MongoDB v3.6.17

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

## 背景資訊

CPS使用MongoDB構建其基本資料庫(DB)結構。

分段是MongoDB的一個特徵。通過此警報，您可以主動監控MongoDB分段，從而避免由於MongoDB而可能提高的資源（磁碟和記憶體）利用率。

當MongoDB分段百分比超過指定值時，CPS將生成簡單網路管理協定(SNMP)警報。

其 `/etc/collectd.d/dbMonitorList.cfg` `sessionmgr` 虛擬機器(VM)上存在的檔案包含資料庫及其各自的分段閾值百分比值的清單。

## 設定

### 組態

### OpenStack中託管的CPS方法

步驟1.從群集管理器虛擬機器，運行此命令以建立當前配置檔案的備份。

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

步驟2.從群集管理器運行此命令，從`sessionmgr` VM獲取當前配置（比較和驗證更改後）。

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

輸出示例：

```
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
```

步驟3.將預設閾值(40)修改為建議值（例如；60）。從群集管理器運行此命令。

**注意：**此命令更改所有資料庫的閾值。如果要求更新特定資料庫的閾值，請手動更新檔案。

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

步驟4.運行此命令以比較群集管理器中的檔案以驗證更改。

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

輸出示例：

```
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
```

步驟5.運行此命令以在群集管理器中生成更改。

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

步驟6.從群集管理器運行此命令以應用會話管理器虛擬機器中的更改。

```
[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 ~]#
```

步驟7.等待木偶完成。從群集管理器運行此命令以檢視Puppet配置的進度。

```
#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 dc1-sessionmgr02...
[Tue Nov 08 06:32:30 +0000 2022] - NTP sync started, check the logs in vm-init.log
```

## VMWare環境下的CPS託管方法

步驟1.更新 `/var/qps/config/deploy/csv/Configuration.csv` 檔案，其中包含所需的資料庫名稱及其各自的閾值百分比。提供自定義閾值百分比值的格式如下(其中XX是百分比的數值.....例如；60)。

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

示例配置：

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

步驟2.運行以下命令以更新 `/etc/collectd.d/dbMonitorList.cfg` 檔案，使其具有來自Configuration.csv檔案的新閾值：

```
[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: 'dc1-lb01' (aka 'lb01')
```

```
Rebuilding facts for: 'dc1-sessionmgr01' (aka 'sessionmgr01')
```

```
Rebuilding facts for: 'dc1-lb02' (aka 'lb02')
```

```
Rebuilding facts for: 'dc1-qns01' (aka 'qns01')
```

```
Rebuilding facts for: 'dc1-qns02' (aka 'qns02')
```

```
Rebuilding facts for: 'dc1-pcrfclient01' (aka 'pcrfclient01')
```

```
Rebuilding facts for: 'dc1-sessionmgr02' (aka 'sessionmgr02')
```

```
Rebuilding facts for: 'dc1-pcrfclient02' (aka 'pcrfclient02')
```

```
No file for VipProxyConfiguration found
```

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

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

```
[root@installer ~]#
```

步驟3.從群集管理器運行此命令以應用會話管理器虛擬機器中的更改。

```
[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 ~]#
```

步驟4.等待木偶完成。從群集管理器運行此命令以檢視Puppet配置的進度。

```
#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): Unsheduling 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
```

## 驗證

使用本節內容，確認您的組態是否正常運作。

驗證sessionmgr VM中的最新配置，並將其與步驟2的輸出進行比較。從群集管理器運行此命令。

```
[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 ~]#
```

## 疑難排解

本節提供的資訊可用於對組態進行疑難排解。

此MongoDB分段警報是在20.1版中引入的，在早期版本中沒有對其進行測量。預設情況下，分段閾值為40%。此閾值必須根據您的部署大小、流量模式（呼叫模式）和其他流量模式因素進行更改。否則，CPS將引發不需要/不需要的資料庫分段閾值突破警報。

## 關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。