

在CPS中配置自定义数据库分段阈值百分比

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[配置](#)

[配置](#)

[OpenStack中托管的CPS方法](#)

[VMWare环境下的CPS托管方法](#)

[验证](#)

[故障排除](#)

简介

本文档介绍如何在思科策略套件(CPS)中配置自定义数据库分段阈值百分比。

先决条件

要求

Cisco 建议您了解以下主题：

- Linux
- CPS
- MongoDB

注意：思科建议您对CPS CLI拥有根用户访问权限。

使用的组件

本文档中的信息基于以下软件和硬件版本：

- CPS 20.2
- 统一计算系统(UCS)-B
- MongoDB v3.6.17

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

CPS使用MongoDB构建其基本数据库(DB)结构。

分段是MongoDB的特征。通过此警报，您可以主动监控MongoDB分段，从而避免由于MongoDB而可能提高资源（磁盘和内存）利用率。

当MongoDB分段百分比超过指定值时，CPS会生成简单网络管理协议(SNMP)警报。

此 `/etc/collectd.d/dbMonitorList.cfg` sessionmgr虚拟机(VM)上存在的文件包含数据库及其各自的分段阈值百分比值的列表。

配置

配置

OpenStack中托管的CPS方法

步骤1:从群集管理器虚拟机中，运行此命令可创建当前配置文件的备份。

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

第二步：从集群管理器运行此命令，从sessionmgr虚拟机获取当前配置（比较和验证更改后）。

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

示例输出：

```
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
```

第三步：将默认阈值(40)修改为建议值（例如；60）。从集群管理器运行此命令。

注意：此命令会更改所有数据库的阈值。如果要求更新特定数据库的阈值，请手动更新文件。

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

第四步：运行此命令以比较集群管理器中的文件以验证更改。

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

示例输出：

```
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
```

第五步：运行此命令以在Cluster Manager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 ~]#
```

第六步：从集群管理器运行此命令以应用会话管理器虚拟机中的更改。

```
[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 ~]#
```

步骤 7.等待木偶完成。从集群管理器运行此命令以查看Puppet配置的进度。

```
#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环境下的CPS托管方法

步骤1:更新 /var/qps/config/deploy/csv/Configuration.csv 集群管理器中的文件，其中包含所需的数据库名称及其各自的阈值百分比。提供自定义阈值百分比值的格式如下(其中XX是百分比的数值.....例如；60)。

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

配置示例：

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

第二步：运行以下命令以更新 `/etc/collectd.d/dbMonitorList.cfg` 文件，使其具有来自Configuration.csv文件的新阈值：

```
[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 ~]#
```

第三步：从集群管理器运行此命令以应用会话管理器虚拟机中的更改。

```
[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 ~]#
```

第四步：等待木偶完成。从集群管理器运行此命令以查看Puppet配置的进度。

```
#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
```

验证

使用本部分可确认配置能否正常运行。

验证sessionmgr虚拟机中的最新配置，并将其与步骤2的输出进行比较。从集群管理器运行此命令

。

```
[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 ~]#
```

故障排除

本部分提供了可用于对配置进行故障排除的信息。

此MongoDB分段警报是在20.1中引入的，在早期版本中未测量。默认情况下，分段阈值为40%。此阈值必须根据部署规模、流量模式（呼叫模式）和其他流量模式因素进行更改。否则，CPS会抛出不需要/不需要的超过数据库分段阈值的警报。

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。