

Waarschuwingen voor hoge lading en aanbevolen wortels in CPS voor probleemoplossing

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Probleem](#)

[Hoge lading voor probleemoplossing](#)

[Werken](#)

Inleiding

In dit document wordt het onderzoek naar een hoge lading en de aanbevolen opdrachten in Cisco Policy Suite (CPS) beschreven.

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- Linux
- CPS

Cisco raadt ook aan dat u voorkeurstoegang hebt tot CPS CLI.

Gebruikte componenten

De informatie in dit document is gebaseerd op CPS 19.4

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk levend is, zorg er dan voor dat u de mogelijke impact van om het even welke opdracht begrijpt.

Achtergrondinformatie

Het laadgemiddelde is de gemiddelde systeemplading op een Linux-server voor een bepaalde tijd. Met andere woorden, het is de CPU vraag van een server die de som van de actieve en de

ondoorzichtige draden bevat.

Meting van het belastingsgemiddelde is van cruciaal belang om te begrijpen hoe de servers functioneren; als u te veel hebt geladen, moet u de processen doden of optimaliseren die grote hoeveelheden middelen verbruiken, of meer middelen verschaffen om de werklast in evenwicht te brengen.

Meestal levert de **top** of de **uptime** opdracht het laadgemiddelde van uw server met uitvoer die er als volgt uitziet:

```
[root@cps-194-aio-mob ~]# uptime
11:41:08 up 6 days, 5:20, 2 users, load average: 0.71, 0.35, 0.24
[root@cps-194-aio-mob ~]#
```

```
[root@cps-194-aio-mob ~]# top
top - 12:17:26 up 6 days, 5:56, 2 users, load average: 0.09, 0.12, 0.13
Tasks: 185 total, 1 running, 183 sleeping, 0 stopped, 1 zombie
%Cpu(s): 0.8 us, 0.8 sy, 0.0 ni, 98.4 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 12137348 total, 4128956 free, 5219860 used, 2788532 buff/cache
KiB Swap: 4194300 total, 4194300 free, 0 used. 6586848 avail Mem
```

```
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
7070 root 5 -15 8263680 1.3g 21728 S 12.5 11.6 561:38.74 java
1 root 20 0 191384 4320 2620 S 0.0 0.0 3:11.17 systemd
```

Deze getallen zijn de gemiddelden van de systeembelasting over een periode van één, vijf en 15 minuten.

Voordat u verder gaat, moeten we deze twee belangrijke zinnen in alle Unix-achtige systemen begrijpen:

Systeemplading/CPU-lading - is een meting van CPU-over- of onderbenutting in een Linux-systeem; het aantal processen dat door de CPU of in ongebruikte staat wordt uitgevoerd.

Laadgemiddelde - is de gemiddelde systeembelasting berekend over een bepaalde periode van 1, 5 en 15 minuten.

Probleem

Wanneer het belastingsgemiddelde van een CPS VM de vastgestelde drempel overschrijdt, wordt HighloadAlert gegenereerd. De drempelwaarde voor de melding bij hoge lading wordt gedefinieerd als $1,5 \times \text{Aantal CPU's in de VM}$. Deze configuratie is te vinden op `/etc/snmp/snmpd.conf`:

```
load 12 12 12
```

```
# 1, 5 and 15 Minute Load Averages (UCD-SNMP-MIB la)
proxy -v 2c -c broadhop localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4 .1.3.6.1.4.1.2021.10.1.5.1
proxy -v 2c -c broadhop localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5 .1.3.6.1.4.1.2021.10.1.5.2
proxy -v 2c -c broadhop localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6 .1.3.6.1.4.1.2021.10.1.5.3
proxy -v 2c -c broadhop localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4.0 .1.3.6.1.4.1.2021.10.1.5.1
proxy -v 2c -c broadhop localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5.0 .1.3.6.1.4.1.2021.10.1.5.2
proxy -v 2c -c broadhop localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6.0 .1.3.6.1.4.1.2021.10.1.5.3
```

Waarschuwingen bij hoge lading voorbeeldweergave:

```
2021-10-31T14:25:36.572711+05:30 XXXXX-lb01 snmptrapd[5717]: 2021-10-31 14:25:36 pcrfclient01
[UDP: [XX.XX.XX.XX]:46046->[XX.XX.XX.XX]:162]:#012DISMAN-EVENT-MIB::sysUpTimeInstance =
99307800#011SNMPv2-MIB::snmpTrapOID.0 = OID: DISMAN-EVENT-MIB::mteTriggerFired#011DISMAN-EVENT-
MIB::mteHotTrigger.0 = STRING: HighLoadAlert#011DISMAN-EVENT-MIB::mteHotTargetName.0 = STRING:
#011DISMAN-EVENT-MIB::mteHotContextName.0 = STRING: #011DISMAN-EVENT-MIB::mteHotOID.0 = OID:
UCD-SNMP-MIB::laErrorFlag.1#011DISMAN-EVENT-MIB::mteHotValue.0 = INTEGER: 1#011UCD-SNMP-
MIB::laNames.1 = STRING: Load-1#011UCD-SNMP-MIB::laErrorMessage.1 = STRING: 1 min Load Average too
high (= 64.84)
```

Hoge lading voor probleemoplossing

Zorg er voor dat de betrokken VM volgens de norm de CPU-telling heeft. Dit kan worden gedaan met de desbetreffende CPS-installatiehandleiding, waarin de CPU-telling wordt vermeld die voor elke VM vereist is.

De enige Linux-opdracht die gecombineerd het laadgemiddelde en het CPU-gebruik door processen biedt, is de **bovenste** opdracht. Om het proces te identificeren dat hoge belasting veroorzaakt, moet de **hoogste** opdracht in de aangetaste VM met regelmatige tussenpozen worden uitgevoerd voor een bepaalde duur die het geval van hoge belasting bestrijkt. Deze opdracht biedt een **bovenste** uitvoer voor elke 3 seconden, voor 15.000 keer (u kunt het nummer wijzigen volgens uw scenario):

```
#top -b -n15000 >> top.txt &
```

```
[root@cps-194-aio-mob ~]# top
top - 09:32:11 up 7 days, 3:11, 3 users, load average: 0.13, 0.16, 0.15
Tasks: 184 total, 1 running, 182 sleeping, 0 stopped, 1 zombie
%Cpu(s): 0.8 us, 0.8 sy, 0.0 ni, 98.4 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 12137348 total, 3911352 free, 5262096 used, 2963900 buff/cache
KiB Swap: 4194300 total, 4194300 free, 0 used. 6520076 avail Mem
```

```
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
7014 redis 20 0 147356 2372 1184 S 6.7 0.0 48:15.15 redis-server
7070 root 5 -15 8263688 1.4g 21744 S 6.7 11.8 645:12.88 java
1 root 20 0 191384 4320 2620 S 0.0 0.0 3:38.65 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.12 kthreadd
3 root 20 0 0 0 0 S 0.0 0.0 0:04.51 ksoftirqd/0
5 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/0:0H
7 root rt 0 0 0 0 S 0.0 0.0 0:01.76 migration/0
8 root 20 0 0 0 0 S 0.0 0.0 0:00.00 rcu_bh
9 root 20 0 0 0 0 S 0.0 0.0 11:53.47 rcu_sched
```

Vergelijk het HoogLaadAlert-exemplaar nauw met de **bovenste** opdrachtoutput, identificeer het proces dat op het moment van de waarschuwing veel gebruikte CPU bevat.

Start vervolgens deze opdracht om meer informatie over het proces te verzamelen:

```
Command Template:
#ps -ef | grep {PID}
```

Sample command:

```
[root@cps-194-aio-mob ~]# ps -ef | grep 7070
root 7070 1 6 Dec02 ? 12:17:06 /usr/bin/java -server -XX:+UnlockDiagnosticVMOptions -
XX:+UnsyncloadClass -Xms2048m -Xmx2048m -javaagent:/opt/broadhop/qns-1/bin/jmxagent.jar -
Dqns.config.dir=/etc/broadhop/pcrf -Dqns.instancenum=1 -
Dlogback.configurationFile=/etc/broadhop/logback.xml -Djmx.port=9045 -
Dorg.osgi.service.http.port=8080 -Dsnmp.port=1161 -Dcom.broadhop.run.systemId=lab -
Dcom.broadhop.run.clusterId=cluster-1 -Dcom.broadhop.run.instanceId=cps-194-aio-mob-1 -
```

```
Dcom.broadhop.config.url=http://pcrfclient01/repos/run/ -
Dcom.broadhop.repository.credentials.isEncrypted=true -Dcom.broadhop.repository.credentials=qns-
svn/3300901EA069E81CE29D4F77DE3C85FA@pcrfclient01 -
Dcom.broadhop.referencedata.local.location=/var/broadhop/checkout -DdisableJms -
DrefreshOnChange=true -DenableRuntimePolling=true -DdefaultNasIp=127.0.0.1 -Xdebug -
Xrunjwp:transport=dt_socket,server=y,suspend=n,address=1044 -Dua.version.2.0.compatible=true -
Denable.compression=true -Denable.dictionary.compression=true -DuseZlibCompression=true -
DenableBestCompression=true -DenableQueueSystem=false -
Dredis.keystore.connection.string=lb01:lb01:6379:6379 -
DbrokerUrl=failover:(tcp://lb01:61616,tcp://lb02:61616)?randomize=false -
DjmsFlowControlHost=lb02 -DjmsFlowControlPort=9045 -Dosgi.framework.activeThreadType=normal -jar
/opt/broadhop/qns-1/plugins/org.eclipse.equinox.launcher_1.1.0.v20100507.jar -console cps-194-
aio-mob:9091 -clean -os linux -ws gtk -arch x86_64
root 7846 7587 0 11:00 pts/0 00:00:00 grep --color=auto 7070
[root@cps-194-aio-mob ~]#
```

Werken

Zodra het proces dat HighloadAlert veroorzaakt is geïdentificeerd, kunnen deze werkpunten worden overwogen:

Stap 1. Start het proces opnieuw.

```
#monit stop {Process Name}
Wait for 10 secs
#monit start {Process Name}
```

Stap 2. Als het proces logback omvat, verifieer dan elke logger met debug logniveau en wijzig het logniveau van bloggers van debug om te waarschuwen/fout.

Stap 3. Als Stap 1 en Stap 2. niet werken, stem dan het respectievelijke configuratiebestand af met de hulp van het ontwikkelingsteam indien nodig.