

Risoluzione dei problemi relativi agli avvisi con carichi elevati e alle soluzioni consigliate in CPS

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Premesse](#)

[Problema](#)

[Risoluzione dei problemi relativi a HighLoad](#)

[Soluzione alternativa](#)

Introduzione

Questo documento descrive l'indagine sugli allarmi con carico elevato e le soluzioni consigliate in Cisco Policy Suite (CPS).

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Linux
- CPS

Cisco consiglia inoltre di disporre dell'accesso privilegiato alla CLI di CPS root.

Componenti usati

Le informazioni fornite in questo documento si basano su CPS 19.4

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

La media del carico è il carico medio del sistema su un server Linux per un periodo di tempo definito. In altre parole, è la richiesta della CPU di un server che include la somma dei thread attivi e inattivi.

La misurazione del carico medio è fondamentale per comprendere le prestazioni dei server; in caso di sovraccarico, è necessario terminare o ottimizzare i processi che utilizzano grandi quantità di risorse oppure fornire più risorse per bilanciare il carico di lavoro.

In genere, il comando **top** o **uptime** fornisce la media di carico del server con un output simile al seguente:

```
[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
```

Questi numeri rappresentano le medie del carico del sistema su un periodo di uno, cinque e 15 minuti.

Prima di procedere, è importante comprendere queste due frasi importanti in tutti i sistemi Unix:

Carico di sistema/Carico della CPU: misura della sovrautilizzazione o sottoutilizzazione della CPU in un sistema Linux; il numero di processi eseguiti dalla CPU o in stato di inattività.

Carico medio: il carico medio di sistema calcolato in un determinato periodo di tempo di 1, 5 e 15 minuti.

Problema

Ogni volta che la media di carico di una VM CPS supera la soglia definita, viene generato **HighLoadAlert**. Il valore di soglia per l'avviso HighLoad è definito come $1.5 * \text{No Of CPUs in VM}$. Questa configurazione è disponibile in `/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
```

Avviso HighLoad di esempio:

```
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::laErrMsg.1 = STRING: 1 min Load Average too  
high (= 64.84)
```

Risoluzione dei problemi relativi a HighLoad

Prima di ulteriori indagini, verificare che la VM interessata disponga del numero di CPU secondo lo standard. A tale scopo, è possibile utilizzare la rispettiva guida all'installazione di CPS in cui viene indicato il numero di CPU richiesto per ciascuna VM.

L'unico comando Linux che, combinato, fornisce la media del carico e l'utilizzo della CPU da parte dei processi, è **top** command. Per identificare il processo che causa HighLoad, il comando **top** deve essere eseguito nella VM interessata a intervalli regolari per una determinata durata che copre l'istanza HighLoad. Questo comando fornisce un output **superiore** ogni 3 secondi, per un numero di volte pari a 15000 (è possibile modificare il numero in base allo 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
```

Mettere in relazione l'istanza HighLoadAlert e confrontarla con l'output del comando **top**, identificare il processo che al momento dell'alert è a uso intensivo della CPU.

Per raccogliere ulteriori informazioni sul processo, eseguire questo comando:

```
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 -
```

```
Xrunjdpw: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 ~]#
```

Soluzione alternativa

Una volta identificato il processo che causa HighLoadAlert, è possibile prendere in considerazione le soluzioni seguenti:

Passaggio 1. Riavviare il processo.

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

Passaggio 2. Se il processo include il logback, verificare tutti i logger con il livello di log di debug e modificare il livello di log dei logger da debug a warn/error.

Passaggio 3. Se il Passaggio 1. e il Passaggio 2. non funzionano, sottoporre a tuning il rispettivo file di configurazione, se necessario con l'aiuto del team di sviluppo.