

# Nexus Monitor Memoria e CPU con Python Cli

## Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Configurazione](#)

[Verifica](#)

[Pianificazione dello script](#)

[Utilizzo di EEM](#)

[Utilizzo dello scheduler](#)

[Risoluzione dei problemi](#)

## Introduzione

Questo documento descrive come monitorare la CPU e la memoria con l'uso del modulo cli python integrato e pianificare le notifiche syslog.

## Prerequisiti

### Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- [Gestore eventi incorporato](#)
- [Nexus Scheduler](#)

### Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Nexus 3000 - 7.0(3)14(7)
- Nexus 9000 - 7.0(3)17(1)

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.

## Configurazione

Per questo esempio, monitorare il processo PTP per la CPU e la memoria.

Assicurarsi di configurare.

```
feature ptp
```

```
feature scheduler
```

Lo script python viene utilizzato per monitorare la CPU e la memoria per un particolare processo in esecuzione sul dispositivo.

```
import cisco
import sys
from cli import *

def main(process):

mem = ""
cpu = ""

ptp_cpu = cli('sh proc cpu | i '+process)
ptp_mem = cli('sh proc mem | i '+process)

if ptp_cpu:
csplit = ptp_cpu.split()
cpu = csplit[4]
if ptp_mem:
msplit = ptp_mem.split()
mem = msplit[3]

cli('syslog priority notifications msg for '+process+' mem: '+mem+' and cpu '+cpu)

return

if __name__ == "__main__":
main(sys.argv[1])
```

Salvare il file come **ptp\_alert\_mem\_cpu.py**.

## Verifica

Fare riferimento a questa sezione per verificare che la configurazione funzioni correttamente.

Eseguire il file dalla CLI.

Sintassi per l'esecuzione dello script:

**python <file script> <nome processo>**

```
Nexus# python bootflash:ptp_alert_mem_cpu.py ptp
```

```
Nexus# show logg last 5
```

```
2018 Dec 13 10:59:30 Nexus %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configured from vty by admin on vsh.28744
```

```
2018 Dec 13 11:02:30 Nexus %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configured from vty by admin on vsh.28867
```

2018 Dec 13 11:03:37 Nexus %EEM\_ACTION-5-NOTIF: for ptp mem: 649089024 and cpu 0.00%

Come mostrato di seguito, questo avviso syslog viene attivato per il processo PTP e include l'utilizzo della CPU e della memoria.

Lo stesso output viene visualizzato dalla CLI di Nexus.

```
Nexus(config-schedule)# show proc cpu | i ptp
21037          233          4586          50          0.00% ptp
```

```
Nexus(config-schedule)# show proc mem | i ptp
```

```
21037  4669440  1018201484649089024 fff201d0/ffff1e65c ptp
```

Di seguito è riportato un altro esempio che mostra come lo script viene utilizzato per terminare un processo non appena la memoria supera la soglia specificata.

Lo script accetta due input, il nome del processo e la dimensione del membro, dopo di che il processo deve essere terminato.

```
Nexus# python bootflash:Check_mem_kill_process.py ptp 23423
```

```
Nexus# show logg last 5
```

```
2018 Dec 20 07:00:09 BGL14.1-G.17-N3K-C31108PC-1 %EEM_ACTION-5-NOTIF: Killing ptp mem: 691027968
2018 Dec 20 07:00:09 BGL14.1-G.17-N3K-C31108PC-1 %VSHD-5-VSHD_SYSLOG_CMD_EXEC: User:admin
executed the command:run bash
2018 Dec 20 07:00:09 BGL14.1-G.17-N3K-C31108PC-1 %SYSMGR-2-SERVICE_CRASHED: Service "ptp" (PID
29107) hasn't caught signal 6 (core will be saved).
```

## Pianificazione dello script

### Utilizzo di EEM

Questo script EEM verrà attivato ogni minuto e quindi verrà generato il messaggio syslog.

```
Nexus(config)# event manager applet mem_cpu
Nexus(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.1.6.1 get-type exact entry-op le
entry-val 12345 poll-interval 60
Nexus(config-applet)# action 1.0 syslog priority notifications msg Running_mem_cpu_script
Nexus(config-applet)# action 2.0 cli command python bootflash:ptp_alert_mem_cpu.py ptp
```

```
Nexus(config-schedule)# show event manager policy internal mem_cpu
```

```
          Name : mem_cpu
          Policy Type : applet
          Event Specification : event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.1.6.1 get-type exact
entry-op le entry-val 12345 poll-interval 60
          Action : 1.0,sup:syslog priority notifications msg
Running_mem_cpu_script;2.0,vsh,sup:command python bootflash:ptp_alert_mem_cpu.py
          Event Specification active on : Active
```

### Utilizzo dello scheduler

Questo processo dell'utilità di pianificazione verrà attivato ogni minuto, quindi verrà generato il messaggio syslog.

```
Nexus(config)# feature scheduler
Nexus(config)# scheduler job name ptp
Nexus(config-job)# python bootflash:ptp_alert_mem_cpu.py ptp
Nexus(config-job)# exit
```

```
Nexus(config)# scheduler schedule name ptp
Nexus(config-schedule)# job name ptp
Nexus(config-schedule)# time start now repeat 0:0:1
```

Schedule starts from Thu Dec 13 11:21:13 2018

```
Nexus(config-schedule)# show scheduler schedule
```

```
Schedule Name      : ptp
-----
User Name          : admin
Schedule Type      : Run every 0 Days 0 Hrs 1 Mins
Start Time         : Thu Dec 13 11:21:13 2018
Last Execution Time : Thu Dec 13 11:21:13 2018
Last Completion Time: Thu Dec 13 11:21:17 2018
```

```
Execution count    : 1
```

```
-----
Job Name           Last Execution Status
-----
ptp                Success (0)
```

```
Nexus(config-schedule)# sh logg last 5
2018 Dec 13 11:20:19 Nexus %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configured from vty by admin on vsh.29770
2018 Dec 13 11:20:31 Nexus %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configured from vty by admin on vsh.29777
2018 Dec 13 11:21:17 Nexus %EEM_ACTION-5-NOTIF: for ptp mem: 649089024 and cpu 0.00%
```

## Risoluzione dei problemi

Al momento non sono disponibili informazioni specifiche per la risoluzione dei problemi di questa configurazione.