

瞭解vManage報告的適用於vEdge 5000/2000/1000/100B和vEdge雲平台的高CPU利用率

目錄

[簡介](#)

[瞭解vEdge 5000/2000/1000/100B和vEdge雲平台報告的高CPU利用率](#)

[說明](#)

[使用fp-um進程的CPU使用率高](#)

[結論](#)

簡介

本文檔描述為什麼在vEdge 5000/2000/1000/100B和vEdge雲平台的vManage中報告了CPU使用率較高，儘管這些平台的效能正常，並且沒有報告高CPU的情況，如頂部所示。

瞭解vEdge 5000/2000/1000/100B和vEdge雲平台報告的高CPU利用率

在17.2.x及更高版本中，可以發現vEdge和vEdge雲平台的CPU和記憶體消耗更高。在給定裝置的vManage Dashboard上會注意到這一點。在某些情況下，這也會導致vManage中的警報和警告數量增加。

說明

當裝置在正常、低負載或無負載下正常運行時，報告高CPU使用率的原因是由於用於計算使用率的公式發生更改。對於17.2版，根據vEdge上的show system status的Load Average計算CPU使用率。

vManage顯示裝置的即時CPU利用率。它根據歷史資料拉出1分鐘平均值[min1_avg]和5分鐘平均值[min5_avg]。根據定義，「平均負載」包括各種因素，而不只是影響利用率計算的CPU週期。例如，當您為平台提供此值時，會考慮IO等待時間、進程掛起時間和其他值。在這種情況下，您會忽略vShell中top命令中顯示的CPU狀態和CPU值。

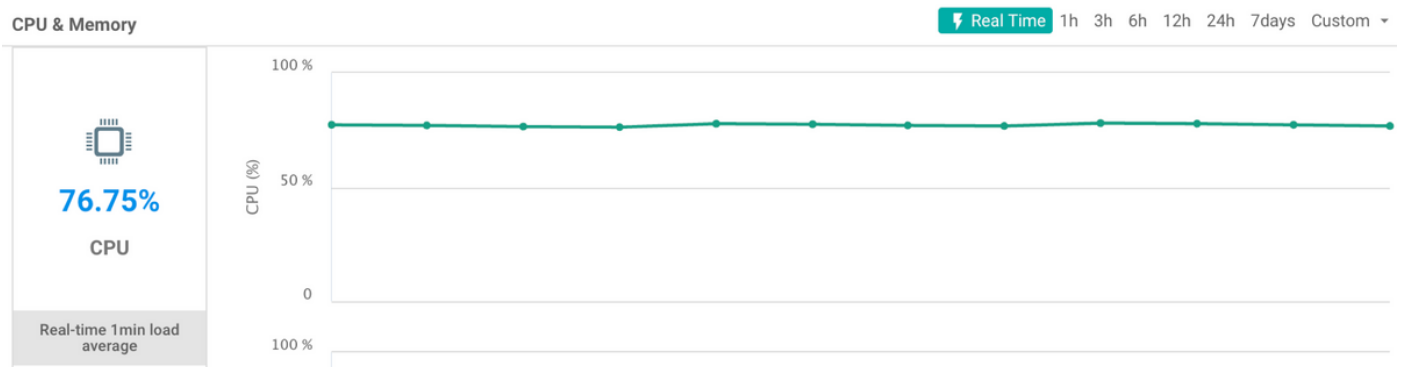
下面是一個示例，介紹如何在vManage控制面板中計算並顯示CPU利用率(實際上是1分鐘負載平均)：

從vEdge CLI檢查負載時，可以看到以下情況：

```
vEdge# show system status | include Load
```

Load average: 1 minute: 3.10, 5 minutes: 3.06, 15 minutes: 3.05
 Load average: 1 minute: 3.12, 5 minutes: 3.07, 15 minutes: 3.06
 Load average: 1 minute: 3.13, 5 minutes: 3.08, 15 minutes: 3.07
 Load average: 1 minute: 3.10, 5 minutes: 3.07, 15 minutes: 3.05

在此案例中，CPU使用率是根據平均負載/核心數(vCPU)計算的。在本示例中，該節點有4個核心。接著負載平均值被轉換100倍，然後除以核心數。當計算所有核心的負載平均值並乘以100時，得到的值為~310。取此值並除以4個產出率，CPU讀數為77.5%，與收集CLI輸出前後在vManage中捕獲的即時圖形中顯示的值一致，如圖所示。



為了檢視系統中的負載平均值和CPU核心數，可以從裝置上的vShell檢視top的輸出。

在此處的示例中，vEdge包含4個vCPU。第一個核心(Cpu0)用於Control (通過較低的使用者利用率看到)，而其餘3個核心用於Data:

```
top - 01:14:57 up 1 day, 3:15, 1 user, load average: 3.06, 3.06, 3.08
Tasks: 219 total, 5 running, 214 sleeping, 0 stopped, 0 zombie
Cpu0 : 1.7%us, 4.0%sy, 0.0%ni, 94.3%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu1 : 56.0%us, 44.0%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu2 : 54.2%us, 45.8%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu3 : 59.3%us, 40.7%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 7382664k total, 2835232k used, 4547432k free, 130520k buffers
Swap: 0k total, 0k used, 0k free, 587880k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
978	root	20	0	3392m	664m	127m	R	100	9.2	1635:21	fp-um-2
692	root	20	0	3392m	664m	127m	R	100	9.2	1635:18	fp-um-1
979	root	20	0	3392m	664m	127m	R	100	9.2	1634:51	fp-um-3
694	root	20	0	1908m	204m	131m	S	1	2.8	15:29.95	ftmd
496	root	20	0	759m	72m	3764	S	0	1.0	1:31.50	confd

為了從vEdge CLI獲取CPU數量，可以使用以下命令：

```
vEdge# show system status | display xml | include total_cpu
<total_cpu_count>4</total_cpu_count>
```

此處提供了計算vEdge 1000上的vManage中所示值的另一個示例。從vShell發出top後，l對於顯示所有核心的負載很感興趣：

```
top - 18:19:49 up 19 days, 1:37, 1 user, load average: 0.55, 0.71, 0.73
```

由於vEdge 1000隻有一個CPU核心可用，因此此處報告的負載為55%(0.55*100)。

使用fp-um進程的CPU使用率高

您有時也會從頂部看到fp-um進程運行率高且顯示多達100%的CPU。在用於資料平面處理的CPU核心上預期會出現這種情況。

從前面引用的top命令中，3個核心以100% CPU運行，1個核心顯示正常利用率：

```
top - 01:14:57 up 1 day, 3:15, 1 user, load average: 3.06, 3.06, 3.08
Tasks: 219 total, 5 running, 214 sleeping, 0 stopped, 0 zombie
Cpu0 : 1.7%us, 4.0%sy, 0.0%ni, 94.3%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu1 : 56.0%us, 44.0%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu2 : 54.2%us, 45.8%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu3 : 59.3%us, 40.7%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 7382664k total, 2835232k used, 4547432k free, 130520k buffers
Swap: 0k total, 0k used, 0k free, 587880k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
   978 root        20   0 3392m 664m 127m R   100   9.2   1635:21 fp-um-2
   692 root        20   0 3392m 664m 127m R   100   9.2   1635:18 fp-um-1
   979 root        20   0 3392m 664m 127m R   100   9.2   1634:51 fp-um-3
...
```

第一個核心(Cpu0)用於Control，其餘三個核心用於Data。正如您在進程清單中看到的，fp-um進程使用這些資源。

fp-um是一個使用輪詢模式驅動程式的進程，這意味著它持續坐下並輪詢資料包的基礎埠，以便接收幀後可以立即對其進行處理。此進程處理轉發，相當於vEdge 1000、vEdge 2000和vEdge 100中的快速路徑轉發。英特爾使用此輪詢模式體系結構來實現基於資料平面開發工具包(DPDK)框架的高效資料包處理。由於資料包轉發是在緊密環路中實施的，因此CPU始終保持或接近100%。雖然這樣做了，但是不會通過這些CPU引入延遲，因為這是預期行為。

有關DPDK輪詢的背景資訊可在[此處](#)找到。

vEdge Cloud和vEdge 5000平台使用相同的轉發架構，並在這方面表現出相同的行為。下面是從頂部輸出中抽取的vEdge 5000的示例。它有28個核心，其中2個（Cpu0和Cpu1）用於Control（與vEdge 2000一樣），26個用於Data。

```

top - 02:18:30 up 1 day, 7:33, 1 user, load average: 26.24, 26.28, 26.31
Tasks: 382 total, 27 running, 355 sleeping, 0 stopped, 0 zombie
Cpu0  :  0.7%us,  1.3%sy,  0.0%ni, 98.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu1  :  0.7%us,  1.3%sy,  0.0%ni, 98.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu2  : 79.4%us, 20.6%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu3  : 73.4%us, 26.6%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu4  : 73.4%us, 26.6%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu5  :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu6  :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu7  :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu8  :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu9  :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu10 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu11 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu12 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu13 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu14 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu15 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu16 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu17 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu18 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu19 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu20 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu21 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu22 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu23 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu24 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu25 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu26 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu27 :100.0%us,  0.0%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem: 32659508k total, 10877980k used, 21781528k free, 214788k buffers
Swap:      0k total,      0k used,      0k free, 1039104k cached

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2028	root	20	0	12.1g	668m	124m	R	100	2.1	1897:21	fp-um-3
2029	root	20	0	12.1g	668m	124m	R	100	2.1	1897:22	fp-um-4
2030	root	20	0	12.1g	668m	124m	R	100	2.1	1897:12	fp-um-5
2031	root	20	0	12.1g	668m	124m	R	100	2.1	1897:22	fp-um-6
2032	root	20	0	12.1g	668m	124m	R	100	2.1	1897:22	fp-um-7
2034	root	20	0	12.1g	668m	124m	R	100	2.1	1897:22	fp-um-9
2035	root	20	0	12.1g	668m	124m	R	100	2.1	1897:21	fp-um-10
2038	root	20	0	12.1g	668m	124m	R	100	2.1	1897:21	fp-um-13
2040	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-15
2041	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-16
2043	root	20	0	12.1g	668m	124m	R	100	2.1	1897:22	fp-um-18
2045	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-20
2052	root	20	0	12.1g	668m	124m	R	100	2.1	1897:18	fp-um-27
2033	root	20	0	12.1g	668m	124m	R	100	2.1	1897:21	fp-um-8
2036	root	20	0	12.1g	668m	124m	R	100	2.1	1897:21	fp-um-11
2037	root	20	0	12.1g	668m	124m	R	100	2.1	1897:21	fp-um-12
2039	root	20	0	12.1g	668m	124m	R	100	2.1	1897:09	fp-um-14
2042	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-17
2044	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-19
2046	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-21
2047	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-22
2048	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-23
2049	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-24
2050	root	20	0	12.1g	668m	124m	R	100	2.1	1897:22	fp-um-25
2051	root	20	0	12.1g	668m	124m	R	100	2.1	1897:23	fp-um-26
1419	root	20	0	116m	5732	2280	S	0	0.0	0:02.00	chmgrd
1323	root	20	0	753m	70m	3764	S	0	0.2	1:51.20	confd
1432	root	20	0	1683m	172m	134m	S	0	0.5	0:58.91	fpm

這裡，平均負載始終很高，因為fp-um進程導致28個處理器中的26個運行率為100%。

結論

在17.2.7之前的17.2.x版本中，vManage中報告的CPU使用率不是實際CPU使用率，而是根據平均負載計算。這可能會導致在理解報告值時產生混淆，並導致與高CPU相關的錯誤警報，而平台在正常、低或無實際流量/網路負載的情況下正常運行。

此行為隨17.2.7和18.2版本而更改/修改，這樣，CPU讀數現在可以根據cpu_user的頂部讀數進行準確讀取。

[17.2發行說明中也提到了此問題。](#)

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）。