

# CVIMのコンピュータノードでメモリ使用量が高い場合のトラブルシューティング

## 内容

---

[はじめに](#)

[前提条件](#)

[要件](#)

[HugePagesとは](#)

[事象の説明](#)

[分析](#)

[トラブルシュート](#)

---

## はじめに

このドキュメントでは、Cisco Virtualized Infrastructure Manager(CVIM)コンピュータノードの高メモリ使用率に関連する問題を分析する手順について説明します。

## 前提条件

### 要件

Linuxのメモリ管理とHugePagesに関する知識があることが推奨されます。

### HugePagesとは

HugePagesを有効にすると、オペレーティングシステムは既定 ( 通常4 KB ) より大きいメモリページをサポートできます。非常に大きなページサイズを使用すると、ページテーブルエントリへのアクセスに必要なシステムリソースが減り、システムパフォーマンスが向上します。その結果、HugePagesは通常、メモリ遅延を軽減するために使用されます。

## 事象の説明

CVIMがアラートをトリガーしていないにもかかわらず、CVIMコンピューティングノードでメモリ使用量が多いアラートが発生する。メモリ使用率に関連するアラートは、サードパーティ製のモニタリングツールまたはモニタリングダッシュボードを使用して生成できます。

### 分析

OSのメモリ使用率が高くなるのが `free` と `sar` コマンドの出力をLinuxで表示します。

```
[root@cvim-computex ~]# free -m
              total        used         free       shared  buff/cache   available
Mem:           385410    365882       7602         3621         11925         8411
Swap:           2047         0         2047
```

```
[root@cvim-computex ~]# sar -r
Linux 4.18.0-193.81.1.el8_2.x86_64 (pod1-compute4.mx2) 08/24/2023 _x86_64_ (112 CPU)
```

Time	kbmemfree	kbavail	kbmemused	%memused	kbbuffers	kbcached	kbcommit	%commit	kbactive	kbinact	kbmemswp	%memswp	kbmemswpd	%memswpd	kbmemfree	kbavail	kbmemused	%memused	kbbuffers	kbcached	kbcommit	%commit	kbactive	kbinact	kbmemswp	%memswp	kbmemswpd	%memswpd
12:00:46 AM																												
12:10:34 AM	7493576	7871200	387166528	98.10	4240	9334356	12893752	3.25	4891940	6325076																		
12:20:11 AM	7503208	7883396	387156896	98.10	4240	9337364	12872708	3.24	4885008	6328096																		
12:30:34 AM	7485648	7869540	387174456	98.10	4240	9340556	12902748	3.25	4892948	6331276																		
12:40:46 AM	7494396	7880940	387165708	98.10	4240	9343636	12866964	3.24	4886908	6334364																		
12:50:34 AM	7479616	7869772	387180488	98.10	4240	9346720	12905156	3.25	4892408	6337444																		
01:00:46 AM	7490304	7883016	387169800	98.10	4240	9349832	12860152	3.24	4885308	6340500																		
01:10:34 AM	7472248	7868672	387187856	98.11	4240	9352836	12896932	3.25	4892604	6343556																		
01:20:46 AM	7484308	7883276	387175796	98.10	4240	9355948	12867972	3.24	4885172	6346676																		
01:30:34 AM	7475092	7869596	387185012	98.11	4240	9350840	12904328	3.25	4892448	6341556																		
01:40:46 AM	7485436	7882508	387174668	98.10	4240	9353932	12864252	3.24	4885148	6344660																		
01:50:34 AM	7468840	7869520	387191264	98.11	4240	9357036	12907464	3.25	4893552	6347752																		
02:00:46 AM	7479076	7882428	387181028	98.10	4240	9360124	12861892	3.24	4886044	6350844																		

ps コマンドを発行して、メモリ使用量が最も高いプロセスを特定します。

```
[root@cvim-computex ~]# ps -aux --sort -rss
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      328199 1207  0.2 541893584 ?    Rl    Mar12 2948779:31 /usr/bin/vpp -c /etc/vpp/vpp.conf
root       1829  0.0  0.0 379024 227692 ?    Ss    Mar12 14:21 /usr/lib/systemd/systemd-journald
```

CLIで統計情報をチェックして、コンテナのメモリ使用量を確認します。 podman または docker コマンドを発行します。

```
[root@cvim-computex ~]# podman stats
ID          NAME                CPU % MEM USAGE / LIMIT MEM % NET IO  BLOCK IO          PIDS
2f8fdc4b63a4 fluentd_31902      --    301.2MB / 404.1GB 0.07% -- / -- 9.265MB / 89.68GB 75
34d806a30733 novalibvirt_31902 --    42.16MB / 404.1GB 0.01% -- / -- 589.8kB / 22.13MB 44
48292d2fa956 novassh_31902     --    5.882MB / 404.1GB 0.00% -- / -- 475.1kB / 167.3MB 2
7b2ce84e86b3 novacompute_31902 --    231.8MB / 404.1GB 0.06% -- / -- 761.9kB / 2.43GB 49
89c01c14ef3f neutron_vpp_31902 --    1.209GB / 404.1GB 0.30% -- / -- 0B / 7.66MB 35
```

提供された出力によると、高いメモリ使用率を示すプロセスはないように思われます。さらに、コンテナは少量のメモリを使用しているようです。

「 free コマンドを実行しても、高いメモリ使用率が表示されます。

```
root@cvim-computex ~]# free -m
              total        used         free       shared    buff/cache   available
Mem:           385410      366751       7310         3496         11348     7696
Swap:            2047           5         2042
[root@cvim-computex ~]#
```

## トラブルシューティング

このメモリ使用率を理解するには、HugePageメモリの知識が不可欠です。

ポッドがHugePagesで有効になっている場合は、システムメモリがVMの起動に使用されないように、適切なフレーバーを使用するように注意する必要があります。VMのシステムメモリの使用は、ワークロードとインフラストラクチャの両方がインフラストラクチャ用に予約されたリソースを奪い合うため、CVIMが不安定になる可能性があります。

HugePagesを確認します。

```
[root@cvim-computex ~]# tail /sys/devices/system/node/node0/hugepages/hugepages-2048kB/nr_hugepages
90001
[root@cvim-computex ~]# tail /sys/devices/system/node/node0/hugepages/hugepages-1048576kB/nr_hugepages
0
[root@cvim-computex ~]# tail /sys/devices/system/node/node1/hugepages/hugepages-2048kB/nr_hugepages
90000
[root@cvim-computex ~]# tail /sys/devices/system/node/node1/hugepages/hugepages-1048576kB/nr_hugepages
0
[root@cvim-computex ~]#
```

`nr_hugepages` は、HugePagesの総数です。

$(90001 + 90000) \times 2M = 360GB$ はHugePage用に予約されています。

また、HugePageが100%構成されている場合でも、物理メモリ全体の5%のメモリが通常のメモリページ(4KB)用に予約され、OSが使用されることに注意してください。

$385GB$  (合計フリー) -  $360GB$  (HugePage用に予約) =  $25GB$ は通常のページ用に予約されています。

したがって、メモリ使用率が高くなります。 `sar` と `free` コマンドが必要です。

上記のコマンドを使用して、実際のメモリ使用量を確認します。

<#root>

```
[root@mgmt-node ~]# ip -br -4 a s br_api
br_api UP 10.x.x.x/24
```

```
[root@mgmt-node ~]# curl -sS -g -u admin:password --cacert /var/www/mercury/mercury-ca.crt https://10.x
```

sample output:

```
{
"status": "
success
",
"data": {
"resultType": "vector",
"result": [
    {
"metric": {
"host": "cvim-computex",
"instance": "10.x.x.x:9273",
"job": "telegraf",
"node_type": "compute"
    },
"value": [
1693479719.383,
"
76.16486394450624
" --> Actual available memory percentage.
    ]
    },
{
"metric": {
"host": "cvim-computey",
"instance": "10.x.x.x:9273",
"job": "telegraf",
"node_type": "compute"
    },
"value": [
1693479719.383,
"76.63431887455388"
```

CVIMは、使用可能なメモリが10 %未満の場合にのみアラートをトリガーします。

アラート名 : mem\_available\_percent

使用可能なシステムメモリが10 %未満である。通常の4Kページのメモリは、システムとOpenStackインフラストラクチャサービスの両方で使用され、大きなページは含まれません。このアラートは、システムまたはインフラストラクチャによるRAMの不足または異常なメモリ使用を示す可能性があります。

## 翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。