

Nexus 3000の平均メモリ使用率

内容

[概要](#)

[背景](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[トラブルシューティングのヒント](#)

概要

このドキュメントは、Nexus 3000デバイスの平均メモリ使用率を理解するのに役立ちます。

著者：Cisco TACエンジニア、Narotama Sahoo、Santhosh Shankar

背景

Nexus 3000のメモリ使用率は、デバイスで実行されているバージョンによって異なる可能性があります。

新しい機能の追加による新しいバージョンでは、古いバージョンに比べてメモリ使用率が高くなる可能性があります。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- Nexusコマンドラインインターフェイス
- コンピュータの世界におけるフラッシュおよびRAMメモリの動作の理解

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- Nexus 3000シリーズプラットフォーム
- コードバージョン6.xおよび7.x

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

トラブルシューティングのヒント

次に、ラボのサンプル出力を示します (最小構成のN3548)。プラットフォームのベースラインメモリ使用率を示します。

```
N3K-3548.42# sh system resources | egrep "used"
Memory usage: 4117860K total, 2250220K used, 1867640K free
```

```
N3K-3548.42# sh system internal kernel memory global | be NAME | cut -c 1-42
NAME          |TOTAL          |
mtc_usd       | 462744 231212 223972 |
fwm           | 277132 97788 84907 |
afm           | 218180 53060 40829 |
netstack      | 380252 45592 38952 |
clis          | 219208 41924 32060 |
m2rib         | 182944 35812 31502 |
--More--
```

```
//// CLI 42 %(943276 kB)
```

```
N3K-3548.42# sh system internal kernel meminfo | egrep "ShmFS|^Cached"
: 1174236 kB
ShmFS:1169840 kB
```

NX-OSのバイナリ/設定などは、ブート中にキックスタート/システムイメージから抽出されます。これらは、パーシステントストレージにバックアップされていない一時ファイルシステムに存在し、ShmFSとしてLinuxで認識されます。

ShmFSの使用に加えて、linuxはプロセスが読み取り/参照しているファイルの内容を一時的にキャッシュします。Cachedカウンタは、ShmFSとキャッシュファイルの合計です。上記の出力から、キャッシュ使用率が全体の約52%(1174236 kB)であることがわかります。

```
N3K-3548.42# sh system internal kernel meminfo | egrep "Slab"
: 85376 kB
```

最後に、カーネルは内部データ構造にメモリを使用します。Slabは、kernelおよびkernel_moduleデータ構造体を対象としています。上記の出力から、スラブの使用率は全体の約4%(85376 kB)になります。

追加のメモリ使用率はlinuxカーネルでは割り出されていませんが、50 mb以下しか占めていません。詳細は、Linuxアーキテクチャのドキュメントを参照してください。

ベースライン使用率の割合として、一時ファイル・システムは最大のメモリ・コンシューマです。これは予想される動作であり、すべてのNX-OSベースのプラットフォームで共通です。

- 5.Xを実行するNX-OSの平均使用率は約40%になる
- 6.Xを実行するNX-OSの平均使用率は60 ~ 70%です。
- 7.Xの平均使用率を実行するNX-OSの場合は約80%になる

したがって、メモリ使用量の増加は、各リリースの新機能の要因となり、OSのベースフットプリントが大きくなる可能性があります。

デバイスで何も/デフォルト設定なしで収集された出力例

バージョン : 6.0(2)U3(7)

```
switch# show system resources module all
```

CPU Resources:

```
-----  
CPU utilization: Module 5 seconds 1 minute 5 minutes  
-----
```

```
1 7 6 9  
-----
```

Processor memory: Module Total(KB) Free(KB) % Used

```
-----  
1 3785276 1450540 61 (Usage when running 6.x version)
```

バージョン : 7.0.3.I2.2a

```
switch# show system resources module all
```

CPU Resources:

```
-----  
CPU utilization: Module 5 seconds 1 minute 5 minutes  
-----
```

```
1 6 8 8  
-----
```

Processor memory: Module Total(KB) Free(KB) % Used

```
-----  
1 3903900 795600 79 (Usage when running 7.x version)
```

推奨事項 :

すべてのソフトウェアのシステム要件を理解するには、必ずリリースノートを参照してください。ハードウェアとソフトウェアの要件は特に、メモリが異なっており、コードの新しいバージョンにアップグレードする場合があります。

便利なリンク :

[リリースノート](#)