

# Nexus 3000 durchschnittliche Speichernutzung

## Inhalt

[Einführung](#)

[Hintergrund](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Tipps zur Fehlerbehebung](#)

## Einführung

Dieses Dokument erläutert die durchschnittliche Speichernutzung auf dem Nexus 3000.

Unterstützt von Narottama Sahoo, Santhosh Shankar, Cisco TAC Engineers

## Hintergrund

Die Speichernutzung im Nexus 3000 variiert wahrscheinlich je nach der auf dem Gerät ausgeführten Version.

Bei neueren Versionen ist die Speicherauslastung im Vergleich zu älteren Versionen wahrscheinlich hoch.

## Voraussetzungen

### Anforderungen

Cisco empfiehlt, über Kenntnisse in folgenden Bereichen zu verfügen:

- Nexus Befehlszeilenschnittstelle
- Die Funktionsweise von Flash- und RAM-Speicher in der Computerwelt

### Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf den folgenden Software- und Hardwareversionen:

- Nexus 3000-Plattform
- Codeversion 6.x und 7.x

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

# Tipps zur Fehlerbehebung

Im Folgenden sind einige Beispielausgaben aus meinem Labor (minimal konfiguriertes N3548) aufgeführt, die die Auslastung des Basisspeichers auf der Plattform veranschaulichen.

```
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--
```

Alle ausgeführten Prozesse benötigen einen gewissen Arbeitsspeicher für Text/Heap/Stack/Daten/Shared-Lib. Die obige CLI meldet die proportionale Speichernutzung pro Prozess (berücksichtigt Speicherseiten, die von Prozessen gemeinsam genutzt werden). Durch Hinzufügen der proportionalen Speichernutzung aller Prozesse erhalten wir eine Darstellung der Speichernutzung von Benutzerarbeitsprozessen. Auf dem Laborschalter betrug die Auslastungsgrundlage für alle laufenden Prozesse etwa 42 % der Gesamtauslastung (943276 kB).

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

NX-OS Binärdateien / Konfigurationen / etc werden während des Bootvorgangs aus Kickstart-/System-Images extrahiert. Diese befinden sich in temporären Dateisystemen, die nicht durch persistente Speicher gesichert sind und von linux als ShmFS verbucht werden.

Neben der ShmFS-Nutzung speichert Linux temporär Dateiinhalte, die von Prozessen gelesen/referenziert werden. Der Cached-Zähler ist eine Summe aus ShmFS- und zwischengespeicherten Dateien. Aus der obigen Ausgabe können Sie sehen, dass die Nutzung im Cache ca. 52 % der Gesamtauslastung ausmacht (1174236 kB).

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

Schließlich verwendet der Kernel auch Speicher für interne Datenstrukturen. Slab berücksichtigt die Datenstrukturen von Kernel und kernel\_module. Aus der oben genannten Ausgabe entfallen ca. 4 % der Gesamtnutzung auf Slab (85376 kB).

Es gibt einige zusätzliche Speichernutzung, die nicht vom Linux-Kernel ausgebrochen ist, aber nur ca. 50 MB beträgt. Weitere Informationen finden Sie in der Linux-Architekturdokumentation.

Im Verhältnis zur Auslastung bei den Basiswerten sind temporäre Dateisysteme der größte Speicherabnehmer. Dieses Verhalten wird von allen NX-OS-basierten Plattformen erwartet.

Für NX-OS mit 5.x beträgt die durchschnittliche Auslastung etwa 40 %.  
Bei NX-OS mit 6.x liegt die durchschnittliche Auslastung zwischen 60 und 70 %.  
Für NX-OS mit 7.x beträgt die durchschnittliche Auslastung ca. 80 %.

Daher ist die erhöhte Speicherauslastung wahrscheinlich ein Faktor für neue Funktionen in jeder Version und eine größere Grundfläche für das Betriebssystem.

Beispielausgabe ohne beliebige/Standardkonfiguration auf dem Gerät erfasst

Version: 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)
```

Version: 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)
```

**Empfehlungen:**

Lesen Sie immer die Versionshinweise, um die Systemanforderungen für jede Software zu verstehen. Die Hardware- und Softwareanforderungen unterscheiden sich insbesondere beim Upgrade auf eine neuere Version des Codes.

Nützliche Links:

[Versionshinweise](#)