

Guide de dépannage de la mémoire des routeurs de la gamme ASR 1000

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Présentation de la mémoire ASR](#)

[Allocation de mémoire sous le pool lsmpl io](#)

[Utilisation de la mémoire](#)

[Vérifier l'utilisation de la mémoire sur IOS-XE](#)

[Vérifier l'utilisation de la mémoire sur IOSd](#)

[Vérifier l'utilisation de TCAM sur un ASR1K](#)

[Vérification de l'utilisation de la mémoire sur QFP](#)

Introduction

Ce document décrit comment vérifier la mémoire système et résoudre les problèmes liés à la mémoire sur les routeurs à services d'agrégation de la gamme Cisco ASR 1000 (ASR1K).

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Logiciel Cisco IOS-XE
- CLI ASR

Note: Vous pourriez avoir besoin d'une licence spéciale pour vous connecter au shell Linux sur le routeur de la gamme ASR 1001.

Components Used

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Toutes les plates-formes ASR1K
- Toutes les versions du logiciel Cisco IOS-XE qui prennent en charge la plate-forme ASR1K

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Présentation de la mémoire ASR

Avec la plupart des plates-formes de routeurs basées sur des logiciels, la majorité des processus logiciels internes sont exécutés dans la mémoire Cisco IOS[®]. La plate-forme ASR1K introduit une architecture logicielle distribuée qui déplace de nombreuses responsabilités de système d'exploitation (OS) hors du processus IOS. L'IOS dans cette architecture, qui était auparavant responsable de presque toutes les opérations internes, fonctionne maintenant comme l'un des nombreux processus Linux. Cela permet aux autres processus Linux de partager la responsabilité du fonctionnement du routeur.

L'ASR1K exécute IOS-XE, pas l'IOS traditionnel. Dans IOS-XE, un composant Linux exécute le noyau, et l'IOS fonctionne comme un démon, qui est ci-après appelé IOSd (IOS-Daemon). Cela crée une exigence de division de la mémoire entre le noyau Linux et l'instance IOSd.

La mémoire partagée entre IOSd et le reste du système est fixe au démarrage et ne peut pas être modifiée. Pour un système de 4 Go, IOSd est alloué à environ 2 Go et pour un système de 8 Go, IOSd est alloué à environ 4 Go (avec redondance logicielle désactivée).

Comme l'ASR1K possède une architecture 64 bits, tout pointeur qui se trouve dans chaque structure de données du système consomme deux fois plus de mémoire que les plates-formes à processeur unique traditionnelles (8 octets au lieu de 4 octets). L'adressage 64 bits permet à IOS de surmonter la limite de mémoire adressable de 2 Go de l'IOS, ce qui lui permet de passer à des millions de routes.

Note: Assurez-vous que vous disposez de suffisamment de mémoire avant d'activer de nouvelles fonctionnalités. Cisco recommande d'avoir au moins 8 Go de DRAM si vous recevez la table de routage BGP (Border Gateway Protocol) complète lorsque la redondance logicielle est activée afin d'éviter l'épuisement de la mémoire.

Allocation de mémoire sous le pool lsmapi_io

Le pool de mémoire LSMPI (Linux Shared Memory Punt Interface) est utilisé afin de transférer des paquets du processeur de transfert vers le processeur de routage. Ce pool de mémoire est découpé lors de l'initialisation du routeur en mémoires tampon préallouées, par opposition au pool de processeurs, où IOS-XE alloue des blocs de mémoire de manière dynamique. Sur la plate-forme ASR1K, le pool lsmapi_io a peu de mémoire libre - généralement moins de 1000 octets - ce qui est normal. Cisco recommande que vous désactiviez la surveillance de la réserve de LSMPI par les applications d'administration réseau afin d'éviter des alarmes fausses.

```
ASR1000# show memory statistics
      Head      Total(b)      Used(b)      Free(b)      Lowest(b)      Largest(b)
Processor 2C073008  1820510884  173985240  1646525644  1614827804  1646234064
```

```
lsmpi_io 996481D0 6295088 6294120 968 968 968
```

S'il y a des problèmes dans le chemin LSMPI, le compteur **d'échec de sortie de périphérique** apparaît incrémenté dans cette sortie de commande (certains résultats sont omis) :

```
ASR1000-1# show platform software infrastructure lsmpi driver
```

```
LSMPI Driver stat ver: 3
```

```
Packets:
```

```
    In: 674572
```

```
    Out: 259861
```

```
Rings:
```

```
    RX: 2047 free    0    in-use    2048 total
```

```
    TX: 2047 free    0    in-use    2048 total
```

```
    RXDONE: 2047 free    0    in-use    2048 total
```

```
    TXDONE: 2047 free    0    in-use    2048 total
```

```
Buffers:
```

```
    RX: 7721 free    473 in-use    8194 total
```

```
Reason for RX drops (sticky):
```

```
    Ring full      : 0
```

```
    Ring put failed : 0
```

```
    No free buffer  : 0
```

```
    Receive failed  : 0
```

```
    Packet too large : 0
```

```
    Other inst buf  : 0
```

```
    Consecutive SOPs : 0
```

```
    No SOP or EOP   : 0
```

```
    EOP but no SOP  : 0
```

```
    Particle overrun : 0
```

```
    Bad particle ins : 0
```

```
    Bad buf cond    : 0
```

```
    DS rd req failed : 0
```

```
    HT rd req failed : 0
```

```
Reason for TX drops (sticky):
```

```
    Bad packet len  : 0
```

```
    Bad buf len     : 0
```

```
    Bad ifindex     : 0
```

```
    No device       : 0
```

```
    No skbuff       : 0
```

```
    Device xmit fail : 0
```

```
    Device xmit retry : 0
```

```
    Tx Done ringfull : 0
```

```
    Bad u->k xlation : 0
```

```
    No extra skbuff  : 0
```

```
<snip>
```

Utilisation de la mémoire

L'ASR1K comprend les éléments fonctionnels suivants dans son système :

- ASR 1000 Series Route Processor (RP)
- ASR 1000 Series Embedded Services Processor (ESP)
- ASR 1000 Series SPA Interface Processor (SIP)

Il est donc nécessaire de surveiller l'utilisation de la mémoire par chacun de ces processeurs dans un environnement de production.

Les processeurs de contrôle exécutent le logiciel Cisco IOS-XE qui se compose d'un noyau Linux et d'un ensemble commun de programmes d'utilitaire au niveau du système d'exploitation, qui inclut Cisco IOS qui s'exécute en tant que processus utilisateur sur la carte RP.

Vérifier l'utilisation de la mémoire sur IOS-XE

Entrez la commande **show platform software status control-processor brief** afin de surveiller l'utilisation de la mémoire sur le RP, l'ESP et le SIP. L'état du système doit être identique, en ce qui concerne des aspects tels que la configuration des fonctionnalités et le trafic, tandis que vous comparez l'utilisation de la mémoire.

```
ASR1K# show platform software status control-processor brief
<snip>
```

```
Memory (kB)
Slot Status   Total      Used (Pct)   Free (Pct)  Committed (Pct)
RP0 Healthy  3907744    1835628 (47%)  2072116 (53%)  2614788 (67%)
ESP0 Healthy  2042668    789764 (39%)   1252904 (61%)  3108376 (152%)
SIP0 Healthy  482544     341004 (71%)   141540 (29%)   367956 (76%)
SIP1 Healthy  482544     315484 (65%)   167060 (35%)   312216 (65%)
```

Note: La mémoire allouée est une estimation de la quantité de mémoire vive dont vous avez besoin pour garantir que le système n'est jamais en panne de mémoire (OOM) pour cette charge de travail. Normalement, le noyau survalide la mémoire. Par exemple, lorsque vous exécutez un malloc de 1 Go, rien ne se passe vraiment. Vous ne recevez une vraie mémoire à la demande que lorsque vous commencez à utiliser cette mémoire allouée, et seulement autant que vous utilisez.

Chaque processeur répertorié dans la sortie précédente peut indiquer l'état **Sain**, **Avertissement** ou **Critique**, qui dépend de la quantité de mémoire disponible. Si l'un des processeurs affiche l'état **Avertissement** ou **Critique**, entrez la commande **monitor platform software process<slot>** afin d'identifier le contributeur principal.

```
ASR1K# monitor platform software process ?
0 SPA-Inter-Processor slot 0
1 SPA-Inter-Processor slot 1
F0 Embedded-Service-Processor slot 0
F1 Embedded-Service-Processor slot 1
FP Embedded-Service-Processor
R0 Route-Processor slot 0
R1 Route-Processor slot 1
RP Route-Processor
<cr>
```

Vous pouvez être invité à définir le type de terminal avant de pouvoir exécuter la commande **monitor platform software process** :

```
ASR1K# monitor platform software process r0
Terminal type 'network' unsupported for command
Change the terminal type with the 'terminal terminal-type' command.
```

Le type de terminal est défini sur **réseau** par défaut. Afin de définir le type de terminal approprié, entrez la commande **terminal-type** :

```
ASR1K#terminal terminal-type vt100
```

Une fois le type de terminal correct configuré, vous pouvez entrer la commande **monitor platform**

software process (certains résultats sont omis) :

```
ASR1000# monitor platform software process r0
```

```
top - 00:34:59 up 5:02, 0 users, load average: 2.43, 1.52, 0.73
Tasks: 136 total, 4 running, 132 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.8%us, 2.3%sy, 0.0%ni, 96.8%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 2009852k total, 1811024k used, 198828k free, 135976k buffers
Swap: 0k total, 0k used, 0k free, 1133544k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
25956	root	20	0	928m	441m	152m	R	1.2	22.5	4:21.32	linux_iosd-imag
29074	root	20	0	106m	95m	6388	S	0.0	4.9	0:14.86	smand
24027	root	20	0	114m	61m	55m	S	0.0	3.1	0:05.07	fman_rp
25227	root	20	0	27096	13m	12m	S	0.0	0.7	0:04.35	imand
23174	root	20	0	33760	11m	9152	S	1.0	0.6	1:58.00	cmmand
23489	root	20	0	23988	7372	4952	S	0.2	0.4	0:05.28	emd
24755	root	20	0	19708	6820	4472	S	1.0	0.3	3:39.33	hman
28475	root	20	0	20460	6448	4792	S	0.0	0.3	0:00.26	psd
27957	root	20	0	16688	5668	3300	S	0.0	0.3	0:00.18	plogd
14572	root	20	0	4576	2932	1308	S	0.0	0.1	0:02.37	reflector.sh

<snip>

Note: Pour trier la sortie par ordre décroissant d'utilisation de la mémoire, appuyez sur **Maj + M**.

Vérifier l'utilisation de la mémoire sur IOSd

Si vous remarquez que le processus **linux_iosd-imag** contient une quantité exceptionnellement importante de mémoire dans la sortie de commande **du processus logiciel de surveillance rp active**, concentrez vos efforts de dépannage sur l'instance IOSd. Il est probable qu'un processus spécifique dans le thread IOSd ne libère pas la mémoire. Dépannez les problèmes de mémoire dans l'instance IOSd de la même manière que vous dépannez les plates-formes de transfert logicielles, telles que les gammes Cisco 2800, 3800 ou 3900.

```
ASR1K# monitor platform software process rp active
```

```
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
25794 root 20 0 2929m 1.9g 155m R 99.9 38.9 1415:11 linux_iosd-imag
23038 root 20 0 33848 13m 10m S 5.9 0.4 30:53.87 cmmand
9599 root 20 0 2648 1152 884 R 2.0 0.0 0:00.01 top
<snip>
```

Entrez la commande **show process memory sorted** afin d'identifier le processus de problème :

```
ASR1000# show process memory sorted
```

```
Processor Pool Total: 1733568032 Used: 1261854564 Free: 471713468
lsmpi_io Pool Total: 6295088 Used: 6294116 Free: 972
```

PID	TTY	Allocated	Freed	Holding	Getbufs	Retbufs	Process
522	0	1587708188	803356800	724777608	54432	0	BGP Router
234	0	3834576340	2644349464	232401568	286163388	15876	IP RIB Update
0	0	263244344	36307492	215384208	0	0	*Init

Note: Ouvrez un dossier TAC si vous avez besoin d'aide pour dépanner ou identifier si

l'utilisation de la mémoire est légitime.

Vérifier l'utilisation de TCAM sur un ASR1K

La classification du trafic est l'une des fonctions les plus élémentaires des routeurs et des commutateurs. De nombreuses applications et fonctionnalités exigent que les périphériques d'infrastructure fournissent ces services différenciés à différents utilisateurs en fonction des exigences de qualité. Le processus de classification du trafic doit être rapide, de sorte que le débit du périphérique ne soit pas considérablement dégradé. La plate-forme ASR1K utilise la 4^e génération de mémoire TCAM4 (Ternary Content Addressable Memory) à cette fin.

Afin de déterminer le nombre total de cellules TCAM disponibles sur la plate-forme, ainsi que le nombre d'entrées libres restantes, entrez cette commande :

```
ASR1000# show platform hardware qfp active tcam resource-manager usage
```

```
Total TCAM Cell Usage Information
```

```
-----
```

```
Name                : TCAM #0 on CPP #0
Total number of regions : 3
Total tcam used cell entries : 65528
Total tcam free cell entries : 30422
Threshold status      : below critical limit
```

Note: Cisco vous recommande de toujours vérifier l'état du seuil avant d'apporter des modifications aux politiques de listes d'accès ou de qualité de service (QoS), afin que le TCAM dispose de suffisamment de cellules libres disponibles pour programmer les entrées.

Si le processeur de transfert fonctionne à un faible niveau critique sur les cellules TCAM libres, l'ESP peut générer des journaux similaires à ceux présentés ci-dessous et peut se bloquer. S'il n'y a pas de redondance, cela entraîne une interruption du trafic.

```
%CPPTCAMRM-6-TCAM_RSRC_ERR: SIP0: cpp_sp: Allocation failed because of insufficient TCAM resources in the system.
```

```
%CPPOSLIB-3-ERROR_NOTIFY: SIP0: cpp_sp:cpp_sp encountered an error -  
Traceback=1#s7f63914d8ef12b8456826243f3b60d7 errmsg:7EFFF525C000+1175
```

Vérification de l'utilisation de la mémoire sur QFP

En plus de la mémoire physique, il y a également de la mémoire attachée à l'ASIC QFP (Quantum Flow Processor) qui est utilisée pour transférer des structures de données, qui incluent des données telles que les stratégies FIB (Forwarding Information Base) et QoS. La quantité de DRAM disponible pour l'ASIC QFP est fixe, avec des plages de 256 Mo, 512 Mo et 1 Go, dépendant du module ESP.

Entrez la commande **show platform hardware qfp active infrastructure exmem statistics** afin de déterminer l'utilisation de la mémoire **exmem**. La somme de la mémoire IRAM et DRAM utilisée donne la mémoire QFP totale utilisée.

BGL.I.05-ASR1000-1# **show platform hardware qfp active infra exmem statistics user**

Type: Name: IRAM, CPP: 0

Allocations	Bytes-Alloc	Bytes-Total	User-Name
-------------	-------------	-------------	-----------

1	115200	115712	CPP_FIA
---	--------	--------	---------

Type: Name: DRAM, CPP: 0

Allocations	Bytes-Alloc	Bytes-Total	User-Name
-------------	-------------	-------------	-----------

4	1344	4096	P/I
9	270600	276480	CEF
1	1138256	1138688	QM RM
1	4194304	4194304	TCAM
1	65536	65536	Qm 16

La mémoire IRAM est la mémoire d'instruction du logiciel QFP. En cas d'épuisement de la DRAM, vous pouvez utiliser la mémoire IRAM disponible. Si la mémoire IRAM est extrêmement faible, le message d'erreur suivant s'affiche :

%QFPOOR-4-LOWRSRC_PERCENT: F1: cpp_ha: QFP 0 IRAM resource low - 97 percent depleted

%QFPOOR-4-LOWRSRC_PERCENT: F1: cpp_ha: QFP 0 IRAM resource low - 98 percent depleted

Afin de déterminer le processus qui consomme la plus grande partie de la mémoire, entrez la commande **show platform hardware qfp active infra exmem statistics user** :

ASR1000# **show platform hardware qfp active infra exmem statistics user**

Type: Name: IRAM, CPP: 0

Allocations	Bytes-Alloc	Bytes-Total	User-Name
-------------	-------------	-------------	-----------

1	115200	115712	CPP_FIA
---	--------	--------	---------

Type: Name: DRAM, CPP: 0

Allocations	Bytes-Alloc	Bytes-Total	User-Name
-------------	-------------	-------------	-----------

4	1344	4096	P/I
9	270600	276480	CEF
1	1138256	1138688	QM RM
1	4194304	4194304	TCAM
1	65536	65536	Qm 16