

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Обзор схемы памяти ASR](#)

[Распределение памяти под пулом `lsmpri.io`](#)

[Использование памяти](#)

[Проверьте использование памяти на XE IOS](#)

[Проверьте использование памяти на IOSd](#)

[Проверьте использование TCAM на ASR1K](#)

[Проверьте загруженность памяти на QFP](#)

Введение

Этот документ описывает как к памяти системы контроля и связанным проблемам памяти устранения неполадок на сервисных маршрутизаторах агрегации Cisco ASR серии 1000 (ASR1K).

Предварительные условия

Требования

Cisco рекомендует иметь базовые знания об этих темах:

- Программное обеспечение Cisco IOS XE
- CLI ASR

Примечание: Вам, возможно, понадобилась бы специальная лицензия для регистрации к оболочке Linux на series маршрутизаторе ASR 1001.

Используемые компоненты

Сведения, содержащиеся в данном документе, касаются следующих версий программного обеспечения и оборудования:

- Все платформы ASR1K
- Все версии программного обеспечения Cisco IOS XE, которые поддерживают платформу ASR1K

Сведения, представленные в этом документе, были получены от устройств, работающих в

специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

Обзор схемы памяти ASR

С большинством программных платформ маршрутизатора большинство процессов внутреннего программного обеспечения выполнено в памяти Cisco IOS®. Платформа ASR1K представляет архитектуру распределенных программных средств, которая перемещает много обязанностей по Операционной системе (OS) из процесса IOS. IOS в этой архитектуре, которая была ранее ответственна за почти все внутренние операции, теперь выполняется как один из многих процессов Linux. Это позволяет другим процессам Linux распределять ответственность для использования маршрутизатора.

ASR1K выполняет XE IOS, не традиционный IOS. В XE IOS компонент Linux выполняет ядро и выполнения IOS как демон, который после этого отнесен как IOSd (Демон IOS). Это создает требование что память быть разделенным между Ядром Linux и экземпляром IOSd.

Память, которая разделена между IOSd и остатком системы, исправлена при запуске и не может модифицироваться. Для системы на 4 ГБ IOSd выделен приблизительно 2 ГБ, и для системы на 8 ГБ, IOSd выделен приблизительно 4 ГБ (с отключенной программной избыточностью).

Так как ASR1K имеет 64-разрядную архитектуру, любой указатель, который находится в каждой структуре данных в системе, использует дважды количество памяти когда по сравнению с традиционными одними CPU платформами (8 байтов вместо 4 байтов). 64 - битная адресация позволяет IOS преодолеть 2GB адресуемое ограничение памяти IOS, который позволяет ему масштабироваться к миллионам маршрутов.

Примечание: Гарантируйте, что вы имеете достаточно памяти в наличии перед активацией любых новых характеристик. Cisco рекомендует иметь DRAM на по крайней мере 8 ГБ при получении всей таблицы маршрутизации Протокола BGP, когда программная избыточность включена для предотвращения исчерпания памяти.

Распределение памяти под пулом lsmpi_io

Пул памяти Интерфейса избыточного направления совместно используемой памяти Linux (LSMPI) используется для передачи пакетов от процессора переадресации до процессора маршрута. Этот пул памяти вырезан в инициализации маршрутизатора в предварительно выделенные буферы, в противоположность пулу процессора, где XE IOS выделяет блоки памяти динамично. На платформе ASR1K пул lsmpi_io имеет мало доступной памяти? обычно меньше чем 1000 байтов? который обычен. Cisco рекомендует отключить мониторинг пула LSMPI приложениями для управления сетью во избежание ошибочных сигналов тревоги.

```
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
```

Если существуют какие-либо проблемы в пути LSMPI, Устройство xmit счетчик сбоя, кажется, инкрементно увеличивается в этих выходных данных команды (некоторые опущенные выходные данные):

```
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 rtry : 0
```

```
    Tx Done ringfull : 0
```

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

```
    No extra skbuff  : 0
```

```
<snip>
```

Использование памяти

ASR1K включает эти функциональные элементы в своей системе:

- ASR процессор маршрута (RP) серии 1000
- ASR Встроенный процессор сервисов (ESP) серии 1000
- ASR Интерфейсный процессор SPA (SIP) серии 1000

Также, это требуется, чтобы контролировать загруженность памяти каждым из этих процессоров в производственной среде.

Управляющие процессоры выполняют программное обеспечение Cisco IOS XE, которое состоит из основанного на Linux ядра и единого набора служебных программ уровня ОС, который включает Cisco IOS, которая выполняется как пользовательский процесс на карте RP.

Проверьте использование памяти на XE IOS

Введите команду `show platform software status control-processor brief` для мониторинга использования памяти на RP, ESP и SIP. Состояние системы должно быть идентичным, в отношении аспектов, таких как конфигурация функции и трафик при сравнении использования памяти.

```
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%)

Примечание: Зафиксированная память является оценкой того, в каком количестве ОЗУ вы нуждаетесь, чтобы гарантировать, что система никогда не является Из памяти (OOM) для этой рабочей нагрузки. Обычно, ядро сверхпередает память. Например, при выполнении `malloc` на 1 ГБ ничто действительно не происходит. Вы только получаете истинную память по требованию, когда вы начинаете использовать, который выделил память, и только так, как вы используете.

Каждый процессор, перечисленный в предыдущих выходных данных, мог бы сообщить о статусе как **Здоровом**, **Предупреждение**, или **Важный**, который зависит от количества свободной памяти. Если какой-либо из процессоров отображает статус как **Предупреждение** или **Важный**, введите команду `<slot> monitor platform software process` для определения главного участника.

```
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>
```

Вам можно было бы предложить установить `terminal-type`, прежде чем можно будет выполнить команду `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.
```

Тип "терминал" собирается **передать** по умолчанию. Для установки соответствующего типа "терминал" введите команду `terminal terminal-type`:

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

Как только корректный тип "терминал" настроен, можно ввести команду `monitor platform software process` (некоторые опущенные выходные данные):

```
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 cmand
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>
```

Примечание: Для сортировки выходных данных в порядке убывания использования памяти, нажмите **Shift + M**.

Проверьте использование памяти на IOSd

Если вы замечаете, что процесс **linux_iosd-imag** держится необычно большое количество памяти в **армированном пластике monitor platform software process активные** выходные данные команды, фокусируйте свои работы по устранению проблем на экземпляре IOSd. Вероятно, что определенный процесс в потоке IOSd не освобождает память. Связанные проблемы памяти устранения неполадок в экземпляре IOSd тот же способ, которым вы устраняете неполадки любых программных передающих платформ, таких как Cisco 2800, 3800 или серии 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 cmand
9599 root 20 0 2648 1152 884 R 2.0 0.0 0:00.01 top
<snip>
```

Войдите **show process memory** сортировал команду для определения процесса задач:

```
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
```

Примечание: Откройте кэйс ТАС (Центра технической поддержки), если вы требуете помощи, чтобы устранить неполадки или определить, если использование памяти легитимно.

Проверьте использование TCAM на ASR1K

Классификация трафика является одной из наиболее базовых функций, найденных в

маршрутизаторах и коммутаторах. Много приложений и функций требуют, чтобы устройства, относящиеся к инфраструктуре предоставили эти дифференцированные сервисы для других пользователей на основе требований к уровню качества. Процесс классификации трафика должен быть быстрым, так, чтобы была не значительно ухудшена пропускная способность устройства. Платформа ASR1K использует четвертое поколение Ternary Content Addressable Memory (TCAM4) для этой цели.

Для определения общего числа ячеек TCAM, доступных на платформе, и количество пустых полей, которые остаются, введите эту команду:

```
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
```

Примечание: Cisco рекомендует всегда проверять пороговый статус перед внесением любых изменений в политику Access-lists или Качества обслуживания (QoS) так, чтобы TCAM имел достаточные свободные ячейки в наличии для программирования записей.

Если процессор переадресации критически исчерпывает ресурсы на свободных ячейках TCAM, ESP мог бы генерировать журналы, подобные показанным ниже, и мог бы завершиться катастрофическим отказом. Если нет никакого резервирования, это приводит к traffic разрушению.

```
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
```

Проверьте загруженность памяти на QFP

В дополнение к физической памяти существует также память, подключенная к ASIC процессора Quantum Flow (QFP), который используется для передачи структур данных, который включает данные, такие как База данных переадресации (FIB) и политики QoS. Сумма DRAM, доступного для ASIC QFP, исправлена, с диапазонами 256 МБ, 512 МБ и 1 ГБ, зависящего от модуля ESP.

Введите команду `show platform hardware qfp active infrastructure exmem statistics` для определения использования памяти `ex-mem`. Сумма памяти для IRAM и DRAM, который используется, дает общую память QFP, которая используется.

```
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
```

IRAM является памятью инструкции для программного обеспечения QFP. Если DRAM исчерпан, доступный IRAM может использоваться. Если IRAM критически исчерпывает ресурсы на памяти, вы могли бы видеть это сообщение об ошибках:

```
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
```

Для определения процесса, который использует большую часть памяти, введите **show platform hardware qfp активная инфра команда user** статистики экс-мем:

```
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
```