

# Устранение проблем утечек памяти MallocLite

## Содержание

[Введение](#)

[Общие сведения](#)

[Устранение неполадок](#)

[Определите приложение, ответственное за утечку](#)

[Декодируйте ПК программы распределения](#)

[Исследуйте статистику памяти MallocLite](#)

[Отключите MallocLite](#)

## Введение

Этот документ описывает, как устранить неполадки утечек памяти MallocLite на Cisco платформы <sup>IOS@software</sup>.

Это также задает информацию, которую необходимо собрать, прежде чем вы откроете Центр технической поддержки Cisco (TAC) случай или повторно загрузите устройство. Соберите выходные данные, упомянутые в этом документе, и подключите их к кэйсу TAC (Центра технической поддержки), чтобы помочь ускорить устранение проблемы.

## Общие сведения

MallocLite используется диспетчером памяти для выделения маленькой, части фиксированного размера памяти, известной как блоки, для выделений, меньше чем или равных 128 байтам. Маленькие распределения памяти не имеют издержек заголовка блока для каждого выделения. Эта функция поддерживается для пулов памяти процессора только.

Каждый заголовок блока памяти берет приблизительно 48 байтов памяти, и самый маленький блок берет приблизительно 24 байта. Даже если необходимо выделить только 8 байтов реальных данных, с традиционным подходом в программном обеспечении Cisco IOS для каждого выделения вы использовали бы по крайней мере 72 (48 + 24) байты памяти.

С MallocLite эти издержки могут быть уменьшены при помощи блоков. Существуют все еще некоторые издержки, потому что нужно управлять блоками. Однако, так как блоки являются фиксированным размером, ими управляют по-другому, чем блоки, и издержки меньше.

Это - ответственность приложений, которые используют память MallocLite для освобождения его должным образом. MallocLite маскирует пользователя памяти.

## Устранение неполадок

**Примечание:** [Cisco CLI Анализатор \(только зарегистрированные клиенты\)](#)

поддерживает некоторые команды `show`. Используйте Cisco CLI Анализатор для

просматривания аналитику выходных данных команды show.

## Определите приложение, ответственное за утечку

Если вы ищете только *mallocite* ключевым словом, обычно трудно определить существующий дефект.

Данный пример показывает, что \*MallocLite\* процесс держит аварийное количество памяти:

```
#show processes memory sorted
```

```
Processor Pool Total: 1614282720 Used: 1544726580 Free: 69556140  
I/O Pool Total: 313524224 Used: 115564032 Free: 197960192
```

```
PID TTY Allocated      Freed   Holding   Getbufs   Retbufs Process  
0 0          0          0 1476043512      0          0 *MallocLite*
```

Необходимо определить точное приложение, которое ответственно за утечку. Три возможных метода идентификации:

- Декодируйте ПК программы распределения.
- Исследуйте статистику памяти MallocLite.
- Отключите MallocLite.

## Декодируйте ПК программы распределения

Даже с включенным MallocLite, можно обычно видеть то, что функция попросила памяти. Выходные данные от команды **show memory allocating-process totals** могли бы показать другие значения ПК даже при том, что название сообщило, MallocLite:

```
#show processes memory sorted
```

```
Processor Pool Total: 1614282720 Used: 1544726580 Free: 69556140  
I/O Pool Total: 313524224 Used: 115564032 Free: 197960192
```

```
PID TTY Allocated      Freed   Holding   Getbufs   Retbufs Process  
0 0          0          0 1476043512      0          0 *MallocLite*
```

Специалист службы технической поддержки Cisco может декодировать значения ПК от вершины списка (с самым высоким общим количеством). Это помогает определять приложение, которое имеет утечку памяти.

## Исследуйте статистику памяти MallocLite

Среди усовершенствований, добавленных в Cisco IOS Software Release 15.1T, был новый CLI, который отображает сводку памяти MallocLite, выделенной каждым ПК. Команда

**облегченных блоков show memory** может помочь вам определять приложения, которые используют большое количество блоков MallocLite.

```
#show processes memory sorted
```

```
Processor Pool Total: 1614282720 Used: 1544726580 Free: 69556140  
I/O Pool Total: 313524224 Used: 115564032 Free: 197960192
```

```
PID TTY Allocated      Freed   Holding   Getbufs   Retbufs Process  
0 0          0          0 1476043512    0          0 *MallocLite*
```

См. [Справочник по командам](#) для подробных данных команды **облегченных блоков show memory**.

```
#show processes memory sorted
```

```
Processor Pool Total: 1614282720 Used: 1544726580 Free: 69556140  
I/O Pool Total: 313524224 Used: 115564032 Free: 197960192
```

```
PID TTY Allocated      Freed   Holding   Getbufs   Retbufs Process  
0 0          0          0 1476043512    0          0 *MallocLite*
```

Примеры вывода от этой команды включают:

```
#show processes memory sorted
```

```
Processor Pool Total: 1614282720 Used: 1544726580 Free: 69556140  
I/O Pool Total: 313524224 Used: 115564032 Free: 197960192
```

```
PID TTY Allocated      Freed   Holding   Getbufs   Retbufs Process  
0 0          0          0 1476043512    0          0 *MallocLite*
```

Снова, инженер ТАС может декодировать значения ПК с самым высоким общим количеством и определить приложение, которое пропускает память.

## Отключите MallocLite

Опция MallocLite активирована по умолчанию. Для исследования утечки MallocLite можно отключить MallocLite:

```
#show processes memory sorted
```

```
Processor Pool Total: 1614282720 Used: 1544726580 Free: 69556140
I/O Pool Total: 313524224 Used: 115564032 Free: 197960192
```

```
PID TTY Allocated      Freed   Holding   Getbufs   Retbufs Process
0 0          0          0 1476043512      0          0 *MallocLite*
```

Пропущенная память будет все еще находиться под MallocLite до следующей повторной загрузки; однако, можно начать контролировать дальнейшие утечки с **сортированным show processes memory** и команды **show memory allocating-process totals**. Утечки теперь появятся под реальном существующим процесс.

Если устройство очень исчерпывает ресурсы на памяти, необходимо сохранить конфигурацию и повторно загрузить устройство для выпуска памяти:

```
#show processes memory sorted
```

```
Processor Pool Total: 1614282720 Used: 1544726580 Free: 69556140
I/O Pool Total: 313524224 Used: 115564032 Free: 197960192
```

```
PID TTY Allocated      Freed   Holding   Getbufs   Retbufs Process
0 0          0          0 1476043512      0          0 *MallocLite*
```

Память могла бы истощать снова в течение долгого времени, так используйте **сортированного show processes memory** и команды **show memory allocating-process totals** чтобы к использованию памяти монитора от той точки вперед.

**Примечание:** Если вы эффективно отключите MallocLite ни с какой командой **memory lite** и перезагрузитесь, то устройство, выведенное от команды **облегченных блоков show memory**, будет пусто.

См. [Справочник по командам](#) для подробных данных команды **memory lite**.