

# Основы регулировки пропускной способности

## Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Условные обозначения](#)

[Общие сведения](#)

[Коммутация уровней процесса и прерывания](#)

[Коммутация путей](#)

[Коммутация в контексте процесса](#)

[Быстрая коммутация](#)

[Оптимальная коммутация](#)

[Cisco Express Forwarding \(CEF\)](#)

[Распределенная быстрая/оптимальная коммутация](#)

[Распределенная функция CEF](#)

[Коммутация NetFlow](#)

[Распределенные службы](#)

[Выбор пути коммутации](#)

[Мониторинг маршрутизатора](#)

[Дополнительные сведения](#)

## Введение

В этом документе представлен изложенный на профессиональном уровне обзор проблем, влияющих на производительность маршрутизатора. Кроме того, раздел содержит ссылки на другие документы, в которых эти проблемы рассмотрены более подробно.

## Предварительные условия

### Требования

Для этого документа отсутствуют особые требования.

### Используемые компоненты

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

- Операционная система Cisco IOS® версии 12.1.

## Условные обозначения

[Дополнительные сведения об условных обозначениях см. в документе Условные обозначения технических терминов Cisco.](#)

## Общие сведения

На эффективность обработки пакетов могут влиять настройки маршрутизатора. Маршрутизаторам, обрабатывающим большие объемы трафика, в целях оптимизации производительности стоит знать, что какое действие выполняет устройство, как оно это делает и сколько времени это у него займет. Эти сведения представлены в файле конфигурации. Конфигурация отражает способ прохода пакетов через маршрутизатор. Не совсем оптимальная конфигурация может удерживать пакет в маршрутизаторе дольше, чем это необходимо. Если поддерживается высокий уровень нагрузки, это может привести к медленным ответам, перегрузке и увеличению времени ожидания подключения.

Основная цель настройки маршрутизатора - сократить время пребывания пакета в маршрутизаторе. То есть сведение к минимуму периода времени, необходимого маршрутизатору для пересылки пакета с входящего на исходящий интерфейс, а также, по возможности, устранение буферизации и перегрузки. Каждая настраиваемая функция в конфигурации должна способствовать быстрейшему достижению входящим пакетом порта назначения.

Два основных ресурса, за сохранением которых нужно следить, - это время использования CPU и память. CPU маршрутизатора должен быть всегда доступным для обработки задач при пиковой и обычной нагрузке. При любом использовании CPU на 99% в течение длительного времени на стабильности сети может быть оказано серьезное воздействие. Тот же принцип применим и для доступности памяти: память должна быть всегда доступна. Если память маршрутизатора использована почти полностью, места в системных буферных пулах нет. Это означает, что пакеты, которым необходима обработка со стороны процессора (перенаправляемые пакеты), сбрасываются сразу же после поступления на маршрутизатор. Легко представить, что бы случилось, если бы сбрасываемые пакеты содержали параметры активного интерфейса или важные обновления маршрута.

## Коммутация уровней процесса и прерывания

В IP-сетях решения о пересылках в маршрутизаторах принимаются на основе содержимого таблицы маршрутизации. [При поиске в таблице маршрутизации маршрутизатор ищет самое длинное совпадение для префикса IP-адреса узла назначения. Это осуществляется "на уровне процессов" \(т.н. коммутация процессов\), то есть поиск является просто одним из процессов, поставленных в очередь для обработки CPU.](#) В результате время поиска становится непредсказуемым и поиск может продолжаться очень долго. Для решения этой проблемы операционная система Cisco IOS располагает целым рядом методов коммутации, основанных на поиске точного соответствия.

Основная выгода поиска точного соответствия в том, что время поиска является детерминированным и очень коротким. Время, необходимое маршрутизатору для принятия решения о пересылке, уменьшено, что позволяет это выполнить на "уровне прерывания". Коммутация на уровне прерывания означает, что при прибытии пакета инициируется прерывание, по которому процессор приостанавливает выполнение остальных задач и выполняет обработку пакета. Традиционный метод для пересылки пакетов (поиск

наибольшего совпадения в таблице маршрутизации) невозможно применить на уровне прерываний, поэтому его необходимо использовать на уровне процессов. По ряду причин, некоторые из которых рассмотрены ниже, невозможно полностью отказаться от метода поиска самого длинного совпадения, поэтому эти два метода поиска сосуществуют в маршрутизаторах Cisco. Эта стратегия обобщена и применяется к протоколам IPX и AppleTalk.

Для выполнения поиска точного соответствия на уровне прерывания таблица маршрутизации должна быть преобразована, чтобы использовать структуру памяти, удобную для этого типа поиска. Различные пути маршрутизации используют различную структуру памяти. Архитектура этой так называемой структуры существенно влияет на время поиска, превращая выбор наиболее подходящего пути маршрутизации в очень важную задачу. Для принятия маршрутизатором решения о перенаправлении пакета необходимы (как минимум) сведения об адресе следующего узла и исходящем интерфейсе. Кроме того, необходимы сведения об инкапсуляции исходящего интерфейса. В зависимости от масштабируемости, сведения об инкапсуляции исходящего интерфейса могут храниться в той же самой или отдельной структуре памяти.

Следующая процедура предназначена для выполнения коммутации на уровне прерываний:

1. Просмотрите структуру памяти для того, чтобы определить адрес следующего узла и исходящий интерфейс.
2. Выполните перезапись на втором уровне модели взаимодействия открытых систем (OSI), также называемой MAC-перезаписью, которая означает изменение инкапсуляции пакета для согласования с исходящим интерфейсом.
3. Поместите пакет в кольцо передачи или очередь выхода исходящего интерфейса.
4. Обновите подходящие структуры памяти (сбросьте таймеры в кэше, обновите счетчики и т.д.).

Прерывание, которое возникло при приеме пакета с сетевого интерфейса, называется "RX interrupt". Этот сигнал прерывания прекращается, только когда выполнены все изложенные выше инструкции. Если хотя бы один из трех вышеизложенных пунктов не может быть выполнен, то пакет пересылается на следующий уровень коммутации. Если следующий уровень коммутации является уровнем коммутации на уровне обработки, то пакет помещается во входную очередь входящего интерфейса для коммутации на уровне обработки, а подача сигнала прерывания прекращается. Так как прерывания не могут быть прерваны прерываниями того же уровня, а все интерфейсы поднимают прерывания того же уровня, ни один пакет не может быть обработан до обработки текущего прерывания RX.

Различающиеся пути маршрутизации прерывания могут быть организованы в иерархическую структуру, позволяющую изменять скорость поиска от самой быстрой до самой медленной. Последней пересортировкой, используемой для обработки пакетов, всегда является коммутация на уровне обработки. Не все интерфейсы и типы пакетов поддерживаются в каждом пути маршрутизации прерывания. В общем случае, только те из них, которые требуют анализа и изменений, ограниченных заголовком пакета, могут быть коммутируемы при прерывании. Коммутация на уровне прерываний будет невозможна, если перед пересылкой необходим анализ полезной информации пакетов. Для некоторых путей маршрутизации прерывания могут существовать особые ограничения. Кроме того, если подключение второго уровня через исходящий интерфейс должно быть надежным (именно поэтому это подключение включает в себя поддержку возможности повторной передачи), то пакеты не могут обрабатываться на уровне прерывания.

Примерами пакетов, для которых не может использоваться коммутация на уровне

прерываний, являются следующие пакеты:

- Трафик, направляемый на маршрутизатор (трафик протокола маршрутизации, простой протокол сетевого управления (SNMP), простейший протокол передачи файлов (TFTP) Telnet, ping и т.д.). Управляющий трафик может быть исходной точкой и направляться на маршрутизатор. В них содержатся процессы, относящиеся к отдельным задачам.
- Ориентированные на подключение инкапсуляции второго уровня модели OSI (например, стандарт X.25). Некоторые задачи слишком сложны для программирования в коммутируемом при прерывании пути, так как содержат слишком много исполняемых инструкций, или требуют большого количества таймеров и окон. Примером этого являются функция шифрования, трансляция LAT-протокола и протокол Data-Link Switching Plus (DLSW+).

## Коммутация путей

Путь прохождения пакета внутри маршрутизатора определяется активным алгоритмом пересылки. Их также называют "алгоритмы коммутации" или "маршруты коммутации". Современные платформы обычно используют более мощные алгоритмы пересылки по сравнению с платформами низшего класса, однако часто они не активизируются по умолчанию. Некоторые алгоритмы продвижения данных реализуются в аппаратных средствах, некоторые - в программном обеспечении, некоторые - в обоих, но целью ставится отправить пакеты так быстро, как это возможно.

В маршрутизаторах Cisco доступны такие алгоритмы коммутации:

Алгоритм пересылки	Команда (Проблема От режима настройки интерфейса)
<a href="#">Быстрая коммутация</a>	<a href="#">ip route-cache</a>
<a href="#">Коммутация одного интерфейса</a>	<a href="#">ip route-cache same-interface</a>
<a href="#">Автономная коммутация</a> (только 7000 платформ)	<a href="#">ip route-cache cbus</a>
<a href="#">Коммутация с использованием кремниевых устройств</a> (7000 платформ с SSP, установленным только)	<a href="#">sse кэша IP- маршрутов</a>
<a href="#">Распределенная коммутация</a> (Только ПЛАТФОРМЫ С ПОДДЕРЖКОЙ VIP)	<a href="#">ip route-cache distributed</a>
<a href="#">Оптимальная коммутация</a> (только высокопроизводительные маршрутизаторы)	<a href="#">ip route-cache optimum</a>
<a href="#">Коммутация NetFlow</a>	<a href="#">ip route-cache flow</a>
<a href="#">Cisco Express Forwarding (CEF)</a>	<a href="#">ip cef</a>
<a href="#">Распределенная функция CEF</a>	<a href="#">ip cef distributed</a>

Вот является краткое описание каждого коммутируемых путями, сортированными в порядке производительности. Автономная коммутация и коммутация на основе кремниевых процессоров не рассматриваются, поскольку они относятся к последним технологиям аппаратных средств.

## Коммутация в контексте процесса

Коммутация процесса – самый главный способ обработки пакета. Этот пакет помещен в очередь, относящуюся к протоколу уровня 3, а затем планировщиком запланирован соответствующий процесс. **Этот процесс является одним из процессов, которые отображаются в результате выполнения команды `show processes cpu` (а именно, "ip input" для IP-пакета).** На этом шаге пакет остается в очереди до тех пор, пока планировщик предоставляет соответствующему процессу ресурсы CPU. Время ожидания зависит от количества процессов, которые ожидают выполнения, а также от количества пакетов, которые нужно обработать. Затем на основе таблицы маршрутизации принимается решение о маршрутизации. Инкапсуляция пакета изменяется для согласования с исходящим интерфейсом, пакет помещается в исходящую очередь соответствующего исходящего интерфейса.

## Быстрая коммутация

В режиме быстрой коммутации CPU принимает решение о пересылке на уровне прерываний. Сведения, взятые из таблицы маршрутизации, и сведения об инкапсуляции исходящих интерфейсов объединяются для формирования кэша быстрой коммутации. Каждая запись кэша состоит из IP-адреса назначения, идентификации исходящего интерфейса и сведений перезаписи MAC. Кэш быстрой коммутации имеет структуру двоичного дерева.

Если для определенного узла назначения в кэше быстрой коммутации отсутствуют какие-либо записи, то текущий пакет должен быть поставлен в очередь для коммутации на уровне обработки. Когда подходящий процесс принимает решение о пересылке этого пакета, он создает запись в кэше быстрой коммутации, а все последующие пакеты для того же самого узла назначения могут быть пересланы на уровне прерывания.

Поскольку этот кэш основан на сведениях об узлах назначения, то распределение нагрузки осуществляется только между узлами назначения. Даже если таблица маршрутизации содержит два одинаковых по затратности пути к сети назначения, то существует только одна запись в кэше быстрой коммутации для каждого узла.

## Оптимальная коммутация

Оптимальная коммутация аналогична быстрой коммутации, но для нее используется многомерное дерево с 256 листьями (mtree) вместо двоичного, что означает большую нагрузку на память и необходимость более быстрого поиска в кэше. [С дополнительными сведениями о структурах деревьев и быстрой, оптимальной и CEF-коммутации можно ознакомиться в документе под названием Выбор наилучшего пути маршрутизации в сети.](#)

## Cisco Express Forwarding (CEF)

Основные недостатки предыдущих алгоритмов коммутации:

1. Первый пакет для конкретного адресата всегда коммутируется процессом, чтобы инициализировать быстрое кэширование.
2. Быстрый кэш может стать очень большим. [Например, если имеются несколько маршрутов с одинаковой стоимостью к одной и той же сети, то скоростная кэш-память наполняется записями о хостах, а не записями о сети, как было описано выше.](#)
3. Не существует прямой связи между быстрым кэшем и таблицей протокола разрешения адресов (ARP-таблица). Если в ARP-кэше запись становится неверной, то в быстром кэше не существует способа объявления ее недействительной. Во избежание этой проблемы каждую минуту произвольным образом сбрасывается 1/20-ая часть кэша. Эта недействительность/перезагрузка кэша может значительно нагружать CPU в очень больших сетях.

CEF обращается к этим проблемам с помощью двух таблиц: таблица FIB (база данных для переадресации) и таблица смежности. Таблица соседства индексируется слоем 3 (L3) адресов и содержит соответствующие данные слоя 2 (L2), необходимые для пересылки пакета. Она заполняется, когда маршрутизатор обнаруживает смежные узлы. Таблица FIB - этоmtree, индексированное по адресам L3. Он создан на основе таблицы маршрутизации и указывает на таблицу смежности.

Еще одно преимущество CEF заключается в том, что структура базы данных позволяет распределять нагрузку для каждого места назначения или для каждого пакета. [Ресурс, посвященный CEF-маршрутизации, содержит дополнительные сведения о технологии CEF.](#)

## [Распределенная быстрая/оптимальная коммутация](#)

Распределенная быстрая/оптимальная коммутация снимает нагрузку с CPU (процессор маршрутизации и коммутации [RSP]) благодаря тому, что перемещает решение о маршрутизации на процессоры интерфейсов (IP). Это возможно только на современных платформах, которые могут иметь выделенные CPU на каждом интерфейсе (многоцелевые интерфейсные процессоры [VIP], линейные платы [LC]). В этом случае fast cache просто выгружается в VIP. При получении пакета VIP пытается принять решение о маршрутизации на основе этой таблицы. Если это удастся, то пакет непосредственно ставится в очередь исходящего интерфейса. При отказе пакет ставится в очередь на следующий конфигурированный путь коммутации (оптимальная коммутация ->быстрая коммутация -> коммутация процессов).

При использовании распределенной коммутации списки доступа копируются на VIP-адреса, а это значит, что VIP может проверить пакет на вхождение в список доступа без вмешательства RSP.

## [Распределенная функция CEF](#)

Распределенный CEF (dCEF) подобен распределенной коммутации, но между таблицами существует меньше синхронизирующих проблем. dCEF является единственным методом распределенной коммутации, доступным от программного обеспечения Cisco IOS версии 12.0. Важно знать, что если распределенная коммутация разрешена на маршрутизаторе, то таблицы FIB/смежности выгружаются на все VIP в маршрутизаторе вне зависимости от CEF/dCEF-настройки интерфейса.

С помощью dCEF-коммутации VIP также обрабатывает списки контроля доступа, данные маршрутизации на основе политик и правила скорости передачи, которые все содержатся в VIP-карте. Netflow можно включить вместе с dCEF для улучшения обработки списка доступа

процессорами VIP.

В таблице ниже показано для каждой платформы, какой путь коммутации с какой версии программного обеспечения Cisco IOS поддерживается.

Путь коммутации	Ниже нижнего уровня (1)	Низкий /Средний Конец (2)	Cisco AS 58 50	Cisco 70 00 w/ RS P	Cisco 72x x/71 xx	Cisco 75 xx	GS R Cisco 12 xxx	Комментарии
Коммутация в контексте процесса	ВСЕ	ВСЕ	ВСЕ	ВСЕ	ВСЕ	ВСЕ	Нет	Инициализирует кэш коммутации
Быстро	Нет	ВСЕ	ВСЕ	ВСЕ	ВСЕ	ВСЕ	Нет	По умолчанию для всех за исключением IP верхнего предела
Оптимальная коммутация	Нет	Нет	Нет	ВСЕ	ВСЕ	ВСЕ	Нет	По умолчанию для высокой производительности для IP прежде 12.0
Netflow Switching (3)	Нет	12.0 (2), 12.0T И 12.0S	ВСЕ	11. 1C A, 11. 1C C, 11. 2, 11. 2P, 11. 3, 11. 3T, 12.	11. 1C A, 11. 1C C, 11. 2, 11. 2P, 11. 3, 11. 3T, 12.	11. 1C A, 11. 1C C, 11. 2, 11. 2P, 11. 3, 11. 3T, 12.	12. 0 (6) S	

				0, 12. 0T, 12. 0S	0, 12. 0T, 12. 0S	0, 12. 0T, 12. 0S		
<b>Коммутация распределенного оптимума</b>	Нет	Нет	Нет	Нет	Нет	11.1, 11.1C, 11.1C, А, 11.2, 11.2P, 11.3 & 11.3T	Нет	Использование VIP2-20,40,50, Не доступно от 12.0.
<b>CEF</b>	Нет	12.0 (5)T	ВСЕ	11.1C, 12.0 & 12.0x	11.1C, 12.0 & 12.0x	11.1C, 12.0 & 12.0x	Нет	По умолчанию при высокой производительности для IP от 12.0
<b>dCEF</b>	Нет	Нет	ВСЕ	Нет	Нет	11.1C, 12.0 & 12.0x	11.1C, 12.0 & 12.0x	Только на 75xx+VIP и на GSR

(1) Включает 801-805.

(2) Включает 806 и выше, 1000, 1400, 1600, 1700, 2600, 3600, 3700, 4000, AS5300, AS5350, AS5400 и AS5800 серии.

(3) Поддержка NetFlow Export версий 1, 5 и 8 на платформах 1400, 1600 и 2500 предназначена для операционной системы Cisco IOS версии 12.0(4)T. Поддержка NetFlow для этих платформ не доступна в операционной системе Cisco IOS основной версии 12.0.

(4) Влияние производительности использования UHP на этих платформах: RSP720-3C/MSFC4, RSP720-3CXL/MSFC4, 7600-ES20-GE3CXL/7600-ES20-D3CXL, SUP720-3BXL/MSFC3 Явный пустой указатель, который вызывает рециркуляцию, и уменьшите производительность в PE. Через уменьшен до 12 Mpps с 20 Mpps на RSP720-3C/MSFC4, RSP720-3CXL/MSFC4, и SUP720-3BXL/MSFC3, и 7600-ES20-GE3CXL/7600-ES20-D3CXL имеет пониженную пропускную способность к 25 Mpps от 48 Mpps.



## Коммутация NetFlow

Коммутация NetFlow (в данном случае) - неправильное употребление термина, усугубленное тем фактом, что она настраивается также, как и коммутируемый путь. Фактически, коммутация NetFlow не является путём маршрутизации, так как NetFlow-кэш не содержит или не указывает на данные, необходимые для перезаписи второго уровня. Решение о коммутации должно быть принято с помощью активного пути маршрутизации.

При коммутации по NetFlow маршрутизатор классифицирует трафик в каждом потоке. Поток - это однонаправленная последовательность пакетов между заданным источником и конечными точками. Для определения потока маршрутизатор использует адреса источника и получателя, номера портов транспортного уровня, тип IP-протокола, тип услуги (ToS) и исходный интерфейс. Этот способ классификации трафика позволяет маршрутизатору обрабатывать только первый пакет потока при работе с такими ресурсоемкими объектами, как большие списки доступа, очереди, политики учета, биллинговые системы и т. п. [Ресурс, посвященный NetFlow-маршрутизации, содержит дополнительные сведения о технологии NetFlow.](#)

## Распределенные службы

Современные платформы позволяют перемещать несколько задач с высокой загрузкой CPU (не только алгоритмы коммутации пакетов) от главного процессора на распределенные процессоры, например, на процессоры плат VIP (7500). Некоторые из данных заданий могут быть экспортированы с универсального процессора на конкретные адаптеры порта или сетевые модули, на которых реализованы функции выделенного аппаратного обеспечения.

Обычно, по возможности, задачи переносят из главного процессора в процессоры VIP. Это высвобождает ресурсы и увеличивает производительность маршрутизатора. Некоторые процессы, которые можно перенести: сжатие пакетов, шифрование пакетов, взвешенная организация очередей. Дополнительные сведения о задачах, которые можно разгрузить, см. в следующей таблице. [Полное описание доступных служб может быть найдено в документе под названием Распределенные службы на маршрутизаторе Cisco 7500.](#)

Сервис	Функции
Базовая коммутация	Fast EtherChannel Фрагментации ip Скоростной маршрутизации Cisco
VPN	ACL - расширились и турбо туннельная IP-безопасность Общей инкапсуляции маршрута (GRE) Шифрования Cisco (IPSec) туннели Протокола туннелирования Уровня 2 (L2TP)
QoS	Формирование трафика NBAR (dTS), Определяющий политику (CAR) Предотвращение перегрузки (DWRED) Гарантированная минимальная пропускная способность (dCBWFQ) Распространение политики через Маршрутизацию по политикам BGP
Многофункциональность	Протокол PPP Сжатия заголовка RTP организации очереди с малой

ь	задержкой за 11/12 FRF с фрагментацией и чередованием данных в канале
Учет	Учет выходных данных приоритета экспорта NetFlow и учет MAC-адресов
Распределение нагрузки	Многоканальный протокол двухточечного соединения выравнивания нагрузки CEF
Кэширование	WCCP V1 WCCP V2
Сжатие	L2 SW и HW компрессия L3 SW и HW компрессия
Групповая адресация	Многоадресная распределенная коммутация

## Выбор пути коммутации

Основное правило состоит в том, что следует выбирать лучший из доступных коммутируемых путей (от самого быстрого к самому медленному): dCEF, CEF, оптимальный и быстрый. Включение CEF или dSEF дает лучшие характеристики. Включение коммутации NetFlow позволяет увеличивать или уменьшать производительность в зависимости от конфигурации. Если списки доступа очень велики, или если требуется ведение некоторого учета, или и то и другое, рекомендуется коммутация NetFlow. Обычно NetFlow включается на оконечных коммутаторах с питанием центрального процессора и многими функциями. Если в одном интерфейсе настроено несколько путей коммутации (например пути быстрой коммутации и CEF), маршрутизатор последовательно пытается использовать все настроенные пути, начиная с наилучшего (CEF) и заканчивая коммутацией процессов.

## Мониторинг маршрутизатора

Используйте следующие команды, чтобы видеть, используется ли коммутируемый путь эффективно и насколько загруженный маршрутизатор.

**show ip interfaces:** Эта команда предоставляет обзор коммутируемого маршрута, примененного к конкретному интерфейсу.

```
Router#show ip interfacesEthernet0/0 is up, line protocol is up Internet address is
10.200.40.23/22 Broadcast address is 255.255.255.255 Address determined by setup command MTU is
1500 bytes Helper address is not set Directed broadcast forwarding is disabled Outgoing access
list is not set Inbound access list is not set Proxy ARP is enabled Security level is default
Split horizon is enabled ICMP redirects are always sent ICMP unreachables are always sent ICMP
mask replies are never sent IP fast switching is enabled IP fast switching on the same interface
is disabled IP Flow switching is disabled IP CEF switching is enabled IP Fast switching turbo
vector IP Normal CEF switching turbo vector IP multicast fast switching is enabled IP multicast
distributed fast switching is disabled IP route-cache flags are Fast, CEF Router Discovery is
disabled IP output packet accounting is disabled IP access violation accounting is disabled
TCP/IP header compression is disabled RTP/IP header compression is disabled Probe proxy name
replies are disabled Policy routing is disabled Network address translation is disabled WCCP
Redirect outbound is disabled WCCP Redirect inbound is disabled WCCP Redirect exclude is
disabled BGP Policy Mapping is disabled
```

Из вышеприведенного результата выполнения команды можно видеть, что быстрая коммутация и коммутация CEF разрешены, а коммутация NetFlow отключена.

[show processes cpu](#): Эта команда отображает полезные сведения о загрузке процессора. [Дополнительные сведения см. в документе Устранение неполадок, связанных с высокой загрузкой CPU на маршрутизаторах Cisco.](#)

```
Router#show processes cpuCPU utilization for five seconds: 0%/0%; one minute: 0%; five minutes: 0%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process 1 28 396653 0 0.00% 0.00% 0.00% 0
Load Meter 2 661 33040 20 0.00% 0.00% 0.00% 0 CEF Scanner 3 63574 707194 89 0.00% 0.00% 0.00% 0
Exec 4 1343928 234720 5725 0.32% 0.08% 0.06% 0 Check heaps 5 0 1 0 0.00% 0.00% 0.00% 0 Chunk
Manager 6 20 5 4000 0.00% 0.00% 0.00% 0 Pool Manager 7 0 2 0 0.00% 0.00% 0.00% 0 Timers 8 100729
69524 1448 0.00% 0.00% 0.00% 0 Serial Backgroun 9 236 66080 3 0.00% 0.00% 0.00% 0 Environmental
mo 10 94597 245505 385 0.00% 0.00% 0.00% 0 ARP Input 11 0 2 0 0.00% 0.00% 0.00% 0 DDR Timers 12
0 2 0 0.00% 0.00% 0.00% 0 Dialer event 13 8 2 4000 0.00% 0.00% 0.00% 0 Entity MIB API 14 0 1 0
0.00% 0.00% 0.00% 0 SERIAL A'detect 15 0 1 0 0.00% 0.00% 0.00% 0 Critical Bkgnd 16 130108 473809
274 0.00% 0.00% 0.00% 0 Net Background 17 8 327 24 0.00% 0.00% 0.00% 0 Logger 18 573 1980044 0
0.00% 0.00% 0.00% 0 TTY Background [...]
```

[show memory summary](#): Первые строки, отображаемые в результате выполнения этой команды, дают полезную информацию об использовании памяти маршрутизатора и о памяти/буфере.

```
Router#show memory summary Head Total(b) Used(b) Free(b) Lowest(b) Largest(b)Processor 8165B63C
6965700 4060804 2904896 2811188 2884112 I/O 1D00000 3145728 1770488 1375240 1333264 1375196[...]
```

[show interfaces stat](#) и [show interfaces switching](#): Эти две команды показывают, какой путь использует маршрутизатор и как коммутируется трафик.

```
Router#show interfaces stat Ethernet0 Switching path Pkts In Chars In Pkts Out Chars Out
Processor 52077 12245489 24646 3170041 Route cache 0 0 0 0 Distributed cache 0 0 0 0 Total 52077
12245489 24646 3170041Router#show interfaces switching Ethernet0 Throttle count 0 Drops RP 0 SP
0 SPD Flushes Fast 0 SSE 0 SPD Aggress Fast 0 SPD Priority Inputs 0 Drops 0 Protocol Path Pkts
In Chars In Pkts Out Chars Out Other Process 0 0 595 35700 Cache misses 0 Fast 0 0 0 0 Auton/SSE
0 0 0 0 IP Process 4 456 4 456 Cache misses 0 Fast 0 0 0 0 Auton/SSE 0 0 0 0 IPX Process 0 0 2
120 Cache misses 0 Fast 0 0 0 0 Auton/SSE 0 0 0 0 Trans. Bridge Process 0 0 0 0 Cache misses 0
Fast 11 660 0 0 Auton/SSE 0 0 0 0 DEC MOP Process 0 0 10 770 Cache misses 0 Fast 0 0 0 0
Auton/SSE 0 0 0 0 ARP Process 1 60 2 120 Cache misses 0 Fast 0 0 0 0 Auton/SSE 0 0 0 0 CDP
Process 200 63700 100 31183 Cache misses 0 Fast 0 0 0 0 Auton/SSE 0 0 0 0
```

## [Дополнительные сведения](#)

- [Решение проблемы высокой загрузки CPU на маршрутизаторах Cisco](#)
- [Команда show processes](#)
- [Cisco Systems – техническая поддержка и документация](#)