

Пример Конфигурации WRED и MDRR на интернет-маршрутизаторах Cisco 12000 Series с Mix of Unicast, Multicast и Voice Traffic

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Условные обозначения](#)

[Общие сведения](#)

[Классы приоритета, используемые при проверке](#)

[Цветная маркировка IP-пакетов](#)

[Ожидаемые результаты](#)

[Настройка сети](#)

[3-портовый модуль GbE 2 Реализация запросов](#)

[Алгоритм WRED модуля Engine 2 в программном обеспечении Cisco IOS](#)

[Конфигурация QoS](#)

[Rx CoS](#)

[Tx QoS](#)

[Сопоставление интерфейса](#)

[Запуск без сброса](#)

[Четыре потока данных по 151 Мбит каждый](#)

[Четыре потока данных по 160 МБ каждый](#)

[Четыре потока данных по 167 мегабит каждый](#)

[Четыре потока данных по 191Мбит каждый](#)

[Четыре потока данных по 244 Мбит каждый](#)

[Все QoS удалены](#)

[Четыре потока данных по 153 МБ каждый](#)

[Четыре потока данных по 158Мбит каждый](#)

[Добавление нагрузки входа](#)

[12-RP#show interfaces g 6/0](#)

[Изменение размера потоков](#)

[Проверка с помощью платы 10-Port Gigabit Ethernet Engine с поддержкой 4 линий](#)

[Дополнительные сведения](#)

Введение

Этот документ объясняет, как настроить линейную карту Серии Cisco 12000 для

Взвешенного произвольного раннего обнаружения (WRED), описанного в [RFC 2309](#), в мультисервисной среде.

Предварительные условия

Требования

Ознакомление с этим документом требует наличия следующих знаний:

- [Общие сведения и настройка MDRR и WRED на маршрутизаторе Интернет серии Cisco 12000](#)
- [Чтение выходных данных команд "show controller frfab" и "tofab queue" на Internet-маршрутизаторе Cisco серии 12000](#)
- [Тип сервиса в Наборе протоколов интернета, приоритеты \(RFC-1349\)](#)
- [Взвешенное произвольное раннее обнаружение](#)

Используемые компоненты

Сведения в этом документе основаны на версиях оборудования и программного обеспечения, указанных ниже:

- Любые Cisco IOS® software release, поддерживающие IP-маршрутизаторы серии Cisco 12000. Обычно это выпуски 12.0S и 12.0ST.
- В этом документе описаны все платформы Cisco 12000. Они включают 12008, 12012, 12016, 12404, 12406, 12410, и 12416.

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.

Условные обозначения

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

Общие сведения

Серия Cisco 12000 является одной из большинства распространенных платформ, используемых для построения базовой IP - сети высокой пропускной способности. Эта платформа предлагает эксклюзивную возможность для настройки Качества обслуживания (QoS).

Так как более распространено смешать различные типы IP - трафика (такого как Передача голоса по IP - VoIP - и передать в многоадресном режиме) в той же сети, требования для приоритизации и управляемого понижающегося поведения становятся чрезвычайно важными, и, во многих случаях, являются разницей между успехом и сбоем при запуске нового сервиса, такого как VoIP.

Требования к сети для различных типов IP - трафика выходят за рамки этого документа. Этот документ фокусируется на лабораторных испытаниях, выполненных для обнаружения конфигурации, которая применима на других линейных картах, включая Серию Cisco 12000, 3 порта Gigabit Ethernet (GbE с 3 портами) линейная карта. Результаты этих тестов показывают, что линейная карта Engine 2 GbE с 3 портами является подходящей для сетевой среды, включающей соединение голоса, данных и многоадресного трафика. Также оказывается, что настройка QoS вносит реальные изменения в перегруженной сети.

Классы приоритета, используемые при проверке

Значения приоритета, которые назначены на другие классы, должны быть тем же во всей сети. Необходимо определить общую политику.

Класс	Precedence	Трафик
Злой трафик		Весь pop, определенный от сетевого трафика (вне сети)
В сети---в сети	1	Трафик, который остается в (сетевой) сети провайдеров услуг
Сервисы интернет-провайдера (ISP)	2	Трафик интернет-провайдера, SMTP, POP, FTP, DNS, Telnet, SSH, www, https
SME (малые и средние предприятия)	3	Корпоративные клиенты, золотой сервис
В реальном времени, голос pop	4	TV, игры в реальном времени
Речь	5	Трафик VoIP RTP
Сообщения управления сетью	6-7	Протокол BGP и другие управляющие сообщения

Цветная маркировка IP-пакетов

Если QoS должно быть внедрено в ядре сети, предпосылка - то, что Поставщик услуг полностью управляет значением приоритета *всех* пакетов IP, транспортируемых в сети. Единственный способ сделать это должно отметить все пакеты, когда они вводят сеть, не делая различия, прибывают ли они от клиента/стороны конечного пользователя или из Интернета. Никакая маркировка или окраска не должны быть сделаны в ядре.

Ожидаемые результаты

Цель с этим дизайном состоит в том, чтобы иметь истинное поведение WRED в классах 0-3. Это означает, что мы хотели бы иметь ситуацию, где мы начинаем отбрасывать приоритеты 0 пакетов во время перегрузки. После этого, если перегрузка продолжается, и затем также

приоритеты 2 и 3, мы должны также начать отбрасывать приоритеты 1. Это все описано в графике ниже.

У вас должна быть самая низкая задержка, возможная для голосовых пакетов и никаких отбрасываний вообще для голоса и многоадресного трафика.

Настройка сети

Чтобы протестировать и оценить конфигурацию, мы использовали Cisco 12410 вместе с мастером создания пакетов от Agilent. Маршрутизатор Cisco 12000 выполняет версию для разработки на основе программного обеспечения Cisco IOS версии 12.0(21)S1.

3-портовый модуль GbE 2 Реализация запросов

Карты engine 2 обычно имеют восемь очередей fromfab и одну очередь с низкой задержкой и восемь очередей tofab на конечный слот. Существует также отдельная очередь групповой адресации tofab. На Плате GbE с 3 портами существует только одна очередь fromfab на физический порт. В тесте конфигурация, которая была применена, задает большие очереди. Результаты показывают, что Плата GbE с 3 портами принимает эту конфигурацию, и очереди автоматически сопоставлены с очередями, которые доступны.

Алгоритм WRED модуля Engine 2 в программном обеспечении Cisco IOS

Необходимо полностью понять алгоритм, используемый для WRED в линейной карте Engine 2 при настройке значений минимума и Maximum Queue Depth. Код не заботится о настроенном минимальном значении; вместо этого, это использует свою собственную формулу (на основе настроенного максимального значения) для установки минимального значения.

Формула:

Минимальное значение = Максимальное значение - (самое высокое значение 2, который не генерирует отрицательный результат),

Значения, используемые в этом тесте, привели к следующим минимальным значениям, запрограммированным к Специализированной интегральной схеме (ASIC):

Precedence	Настроенный минимум	Настраиваемое максимальное значение	Самое высокое значение 2	Минимальное значение в ASIC
0	50	5000	4096	5000-4096=904
1	60	6000	4096	6000-4096=1904
2	70	7000	4096	7000-

				4096=2904
3	80	8000	4096	8000-4096=3904

Использование этой формулы для вычисления минимального значения означает, что можно закончить с поведением обработки неверного пакета, если вы не принимаете это во внимание при настройке параметров WRED. Это показано на следующем примере:

Precedence	Настроенный минимум	Настраиваемое максимальное значение	Самое высокое питание 2	Минимальное значение в ASIC
0	50	150	128	150-128=22
1	75	225	128	225-128=97
2	100	300	256	300-256=44
3	125	375	256	375-256=119

Это означает, что, даже при том, что значения настроены, чтобы начать понижаться согласно правилу 0 сначала, тогда 1, 2 и наконец 3 (выше), когда значения записаны в ASIC, вы фактически начинаете отбрасывать приоритеты 0, затем приоритеты 2, затем приоритеты 1, и наконец приоритеты 3. Нет никакого способа видеть, какие значения были настроены в ASIC на карте Engine 2. При применении конфигурации на карту Механизма 3 значения, обнаруживающиеся в конфигурации, будут действительными значениями (перерасчетное минимальное значение).

[Конфигурация QoS](#)

Для получения дополнительной информации о конфигурации QoS, считайте [Понимание и MDRR Настройки и WRED на Интернет-маршрутизаторе Cisco 12000 серии](#).

[Rx CoS](#)

```
rx-cos-slot 2 B2-Table
rx-cos-slot 3 B2-Table
rx-cos-slot 6 B2-Table
```

В большинстве случаев можно использовать команду **rx-cos-slot all**. В нашем контрольном примере у нас были некоторые карты, которые не поддерживали организацию очереди ToFab, таким образом, мы не могли всегда использовать команду **rx-cos-slot all**. Вместо этого мы назначили нашу таблицу слота на линейные карты, используемые в тесте.

```
!
slot-table-cos B2-Table
destination-slot all B2
multicast B2 !--- If you don't fulfill the steps above, you will not be able to reach the goal of 0 drops for Multicast traffic. With no rx-cos configured, multicast will be treated in the default queue, meaning it will drop as soon as there is congestion. !
```

[Tx QoS](#)

Теперь можно настроить tx-потому что. Выберите название для своего qos tx, такого как "B2 cos-queue-group".

Каждое значение приоритета, что вы хотите настроить поведение отбрасывания для потребностей, которые будут сопоставлены с отдельным Random-detect-label.

```
precedence 0 random-detect-label 0
```

```
precedence 1 random-detect-label 1
```

```
precedence 2 random-detect-label 2
```

```
precedence 3 random-detect-label 3
```

Для Modified Deficit Round Robin (MDRR) сопоставьте каждые приоритеты с очередью MDRR. В нашем примере мы сопоставили приоритеты 0-3 с той же очередью MDRR для резервирования полосы пропускания для видео (многоадресный трафик). Это сопоставление предоставляет запрошенное состояние.

```
precedence 0 queue 0
```

```
precedence 1 queue 0
```

```
precedence 2 queue 0
```

```
precedence 3 queue 0
```

```
precedence 4 queue 4
```

Голос отмечен приоритетами 5, который является, почему приоритеты 5 сопоставлены с очередью с низкой задержкой.

```
precedence 5 queue low-latency
```

```
precedence 6 queue 6
```

```
precedence 7 queue 6
```

Теперь необходимо настроить понижающееся поведение для каждого из Random-detect-label. Во время тестирования были изменены эти номера, пока значения не были найдены, что это дало желаемый образец отбрасывания. Для получения дополнительной информации посмотрите раздел [Ожидаемых результатов](#). Глубина очереди измерена на физической очереди, а не на MDRR или очереди Метки КРАСНОГО.

```
random-detect-label 0 50 5000 1
```

```
random-detect-label 1 60 6000 1
```

```
random-detect-label 2 70 7000 1
```

```
random-detect-label 3 80 8000 1
```

На Cisco 12000 возможно создать на основе классов, Обслуживание очередей на основе равнодоступности (CBWFQ), поведение, путем предоставления другого MDRR помещает вес в очередь. Вес по умолчанию 10 на очередь. Количество байтов передало каждый цикл MDRR, функция значения веса. Значение 1 средства 1500 байтов каждый цикл. Значение 10 средств $1500 + (9 * 512)$ байты на цикл."

```
queue 0 20
```

```
queue 4 20
```

```
queue 6 20
```

```
!
```

Сопоставление интерфейса

Каждый интерфейс должен быть настроен для WRED. Это сделано с помощью команд:

- `configure terminal`
- **интерфейсный концерт 6/0**
- **B2 tx-потому что**

Запуск без сброса

Сформированный поток использует следующие значения, пока что-то еще не сообщается:

```
MTU all three data streams 300byte, MTU voice 80byte, MTU MC 1500byte
```

```
126Mb MC, 114Mb voip
```

Это приводит к фоновому потоку 240 МБ (VoIP и групповая адресация).

Мы тогда добавляем четыре потока данных одинакового размера, но с приоритетами 0-3 (одно значение приоритета на поток).

Четыре потока данных по 151 Мбит каждый

Эта конфигурация дает общую пропускную способность 844 МБ. График ниже показывает, что существует 0 отбрасывания пакета, и очень низкая задержка (приблизительно 50 нас - микросекунды - для каждого потока, включая Голос).

Четыре потока данных по 160 МБ каждый

Эта конфигурация дает общую пропускную способность 880 МБ. График ниже показывает, что пакеты начинают отбрасывать от приоритетов 0 классов и задержку для Голоса, увеличился до 6 мс - миллисекунды.

Четыре потока данных по 167 мегабит каждый

Эта конфигурация дает общую пропускную способность 908 МБ. Отбрасывания теперь запускают для приоритетов 1 класс также. Задержка Голосового трафика является все еще тем же.

Примечание: Поток не был остановлен прежде чем быть увеличенным, таким образом, различие между количеством падений потока 0 и 1 кумулятивно.

Четыре потока данных по 191Мбит каждый

Когда общая пропускная способность увеличивается, пакеты начинают отбрасывать от приоритетов 2 очереди также. Общая пропускная способность, которой мы пытаемся достигнуть для Интерфейса Gigabit Ethernet, - теперь 1004 МБ. Это проиллюстрировано в счетчиках ошибки нумерации в графике ниже.

Четыре потока данных по 244 Мбит каждый

Ошибки нумерации для приоритетов 3 начинают увеличиваться также. Это - первый знак, что отбрасывания запустятся с той очереди. Максимальная пропускная способность, что мы пытаемся отослать интерфейс GbE, - теперь 1216 Мбит. Обратите внимание на то, что отбрасывания на групповой адресации (MC) и Речевая очередь являются все еще нулем, и задержка Речевой очереди не увеличилась.

Остановка и начало

Все потоки были остановлены и начали генерировать график, который очистил счетчики. Это показывает, как это посмотрит во время сильной перегрузки. Как вы можете видеть ниже, поведение является желаемым.

Все QoS удалены

Чтобы доказать, что QoS действительно улучшает производительность во время перегрузки, QoS было теперь удалено, и интерфейс был переполнен. Результаты ниже (сформированный поток использует следующие значения, пока что-то еще не сообщается).

MTU all three data streams 300byte, MTU voice 80byte, MTU MC 1500byte

126Mb MC, 114Mb VoIP

Это приводит к фоновому потоку 240 МБ (VoIP и групповая адресация).

Мы тогда добавляем четыре потока данных одинакового размера, но с приоритетами 0-3 (одно значение приоритета на поток).

Четыре потока данных по 153 МБ каждый

Это дает в общей сложности 852 МБ. Существует 0 отбрасываний и задержка менее тогда 50 нас.

Четыре потока данных по 158Мбит каждый

Мы начинаем понижаться при приблизительно том же использовании как тогда, когда WRED применен (872 МБ). Различие теперь - то, что существует задержка Голосовых пакетов 14 мс (более двух раз так же как в тесте WRED), и это понижается, происходят одинаково от всех классов, включая VoIP и групповую адресацию.

Добавление нагрузки входа

До сих пор все тесты только включали передачу через Интерфейсы Gigabit Ethernet. Чтобы проверить, как интерфейс реагирует в ситуации, где мы также переполняем интерфейс в другом направлении, следующие тесты были сделаны.

Для этого теста мы загрузили Интерфейс Gigabit Ethernet общим количеством 1056 Мбит. Это привело к отбрасываниям на приоритетах 0-2, никаким отбрасываниям на приоритетах 3 трафика. (MC и VOIP остались тем же, т.е. никакими отбрасываниями). Мы тогда добавляемый трафик в другом направлении, столько трафика, сколько мастер создания пакетов смог отослать через Интерфейс Gigabit Ethernet. Результат является довольно впечатляющим: перегрузка получения не вмешивается вообще в очередь передачи, и задержка для трафика получения чрезвычайно низка, меньше чем 13 нас для Голоса.

[12-RP#show interfaces g 6/0](#)

Можно контролировать загрузку на Гигабитной ссылке с помощью команды **show interfaces**:

```
Router#show interfaces gig 6/0 GigabitEthernet6/0 is up, line protocol is up Hardware is GigMac
3 Port GigabitEthernet, address is 0004.de56.c264 (bia 0004.de56.c264) Internet address is
178.6.0.1/24 MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec, rely 255/255, load 232/255
Encapsulation ARPA, loopback not set Keepalive set (10 sec) Full-duplex mode, link type is
force-up, media type is SX output flow-control is unsupported, input flow-control is off ARP
type: ARPA, ARP Timeout 04:00:00 Last input 00:00:05, output 00:00:05, output hang never Last
clearing of "show interface" counters 08:52:40 Queueing strategy: random early detection (WRED)
Output queue 0/40, 2174119522 drops; input queue 0/75, 0 drops 30 second input rate 838969000
bits/sec, 792079 packets/sec 30 second output rate 910819000 bits/sec, 464704 packets/sec
7584351146 packets input, 1003461987270 bytes, 0 no buffer Received 0 broadcasts, 0 runts, 0
giants, 0 throttles 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored 0 watchdog, 514
multicast, 0 pause input 11167110605 packets output, 2241229569668 bytes, 0 underruns 0 output
errors, 0 collisions, 0 interface resets 0 babbles, 0 late collision, 0 deferred 0 lost carrier,
0 no carrier, 0 pause output 0 output buffer failures, 0 output buffers swapped out
```

[Изменение размера потоков](#)

Чтобы проверить, что результаты тестирования не происходят из-за пропускной способности, являющейся тем же для всех потоков, мы изменили потоки так, чтобы они передавали другие объемы данных. Мы также пытались изменить максимальный размер передаваемого блока данных (MTU), таким образом, это было другим для каждого потока. С настроенными значениями очереди результатом было все еще то же, отбрасывая приоритеты 0 первых, тогда 1, тогда 2, и наконец приоритеты 3.

[Проверка с помощью платы 10-Port Gigabit Ethernet Engine с поддержкой 4 линий](#)

Так как задержка очереди VoIP (Очередь с низкой задержкой) в тесте была довольно высока, мы выполнили тот же тест с Механизмом С 10 портами Gigabit Ethernet 4 линейных карты. Как ожидалось результат с этой линейной картой намного лучше расценивал задержку в очереди с низкой задержкой (LLQ). Результатами было то же относительно того, как произошло отбрасывание. Задержка для LLQ была вокруг 10us, который является 1:1000 задержки в линейной карте Engine 2 С 3 портами Gigabit Ethernet.

[Дополнительные сведения](#)

- [Общие сведения и настройка MDRR и WRED на маршрутизаторе Интернет серии Cisco 12000](#)
- [Чтение выходных данных команд "show controller frfab" и "tofab queue" на Internet-маршрутизаторе Cisco серии 12000](#)
- [Тип сервиса в Наборе протоколов интернета, приоритеты \(RFC-1349\)](#)
- [Взвешенное произвольное раннее обнаружение](#)
- [Cisco Systems – техническая поддержка и документация](#)