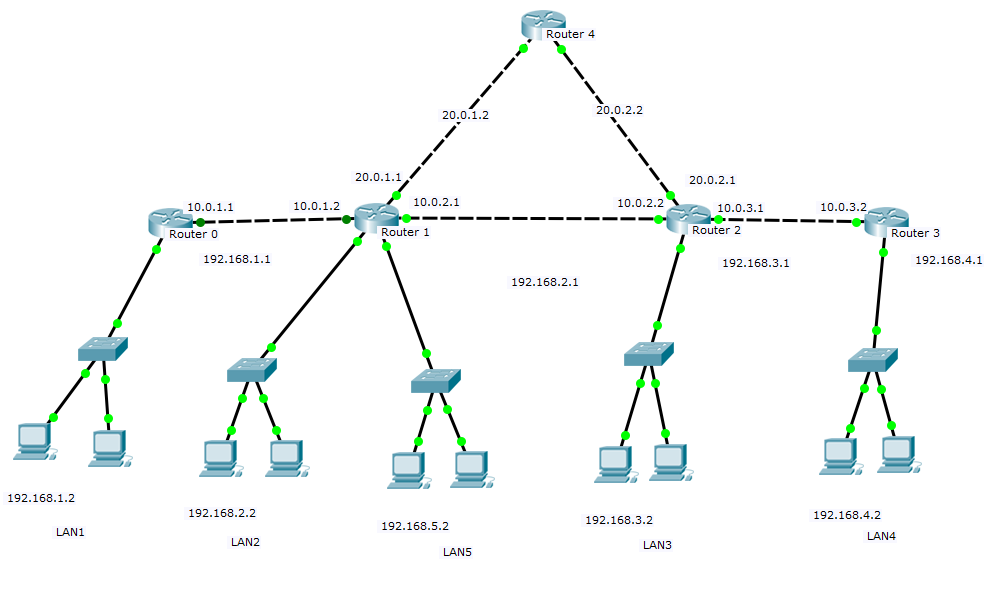
# 

# Лабораторная работа №8. Метрика. Балансировка трафика по маршрутам с метрикой не равной стоимости

За основу возьмём нашу сеть с прошлой лабораторной работы. Нам необходимо получить вот такую сеть:



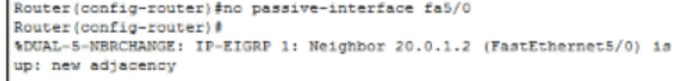
Для этого нам необходимо:

1. Сохранить конфигурацию Роутер 1 и Роутер 2 (команда write).
2. Выключить Роутер 1 и Роутер 2 и добавить в них еще по одному сетевому интерфейсу.
3. Соединить и настроить новый Роутер 4 (добавив адреса).
4. Включите eigrp на этом роутере, командой

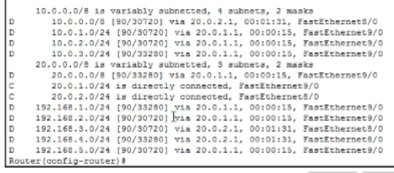
(config-if)# **router eigrp 1**

(config-router)# **network 20.0.2.0 0.0.255.255** - этой командой мы анонсируем сразу две сети, благодаря обратной маске (вспоминаем прошлую лабу); вместо 2 в адресе можно было в принципе написать всё что угодно.

1. Теперь анонсируйте эти сети на Роутер 1 и Роутер 2 самостоятельно.
2. На Роутере 1 мы с вами включали глобальный пассивный интерфейс, наш Роутер 1 применил это правило на новый интерфейс, чтобы это изменить, напишите команду (config-router)# **no passive-interface fa5/0(у вас может быть другой)**. После этой команды вы должны увидеть ответ:

****

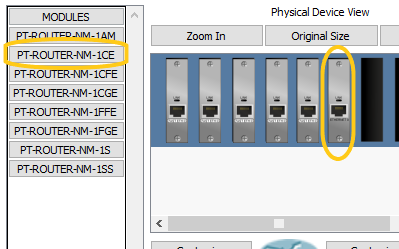
1. Давайте теперь проверим таблицу маршрутизации на Роутер 4, введем команду **show ip route**, отлично наши роутеры “подружились”.

****

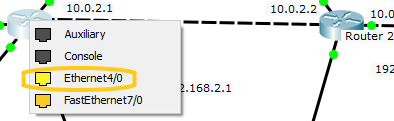
1. Теперь давайте заменим канал между Роутер 1 и Роутер 2, со 100 мегабитного на 10 мегабитный.



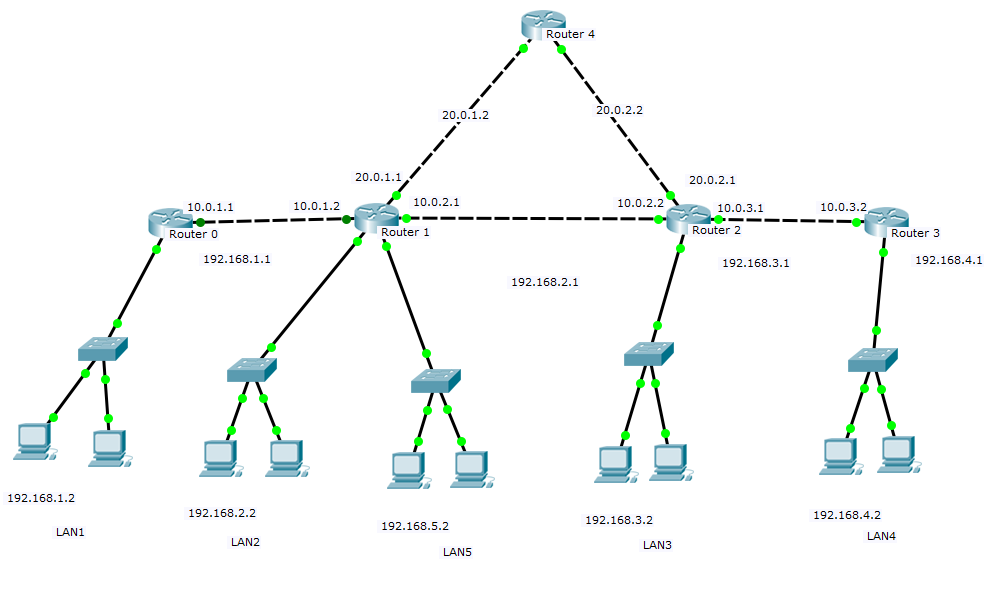
1. Для этого, сохраним конфигурации Роутера 1 и Роутер 2 (команда write), выключим их и добавим 10 мегабитный интерфейс в Роутеры



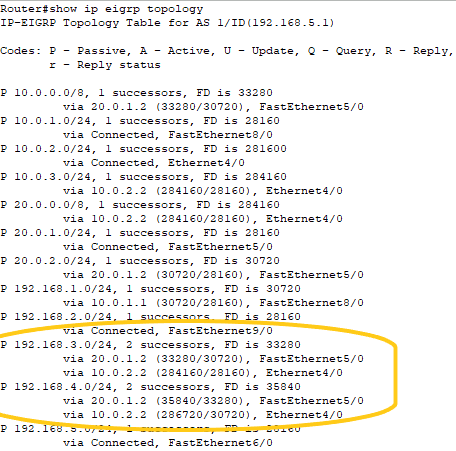
1. Теперь давайте снимем с Роутер 1 и 2 старые IP адреса и выключим не нужный интерфейс. Заходим в интерфейс командой **int fa7/0(у вас может быть другой)** и пишем команду **no ip address** (этой командой мы сняли IP адрес с интерфейса). После этого задаем команду **shut** - она выключит данный интерфейс.
2. Сделайте всё тоже самое с Роутер 2.
3. Создайте подключение между Роутерами, выберите созданный нами Ethernet



1. Настраиваем IP адреса на Роутерах, ставим те же самые что и были.
2. Нам придётся заново эту сеть анонсировать на Роутерах, не забывайте, что Роутер 1 сконфигурирован так что все новые интерфейсы будут пассивными, не забудьте сделать его активным.
3. В итоге у нас такая ситуация, что мы можем попасть на с LAN 3 на LAN 2 и через Роутер 4, и через Роутер 2. Задача роутера какой маршрут выбрать.



1. Давайте введем команду show ip eigrp topology на Роутер 1, и мы увидим, что у нас есть два маршрута как добраться до сети 192.168.3.0 и 192.168.4.0. Это называется **таблица топологий**. Один из этих маршрутов должен попасть в **таблицу маршрутизации**. Роутер должен принять решение, через кого отправлять трафик.

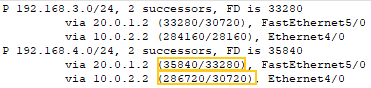


1. Введем команду **show ip route** и посмотрим, какой маршрут выбрал роутер, мы видим, что в таблице маршрутизации, маршрут построен через Роутер 4.

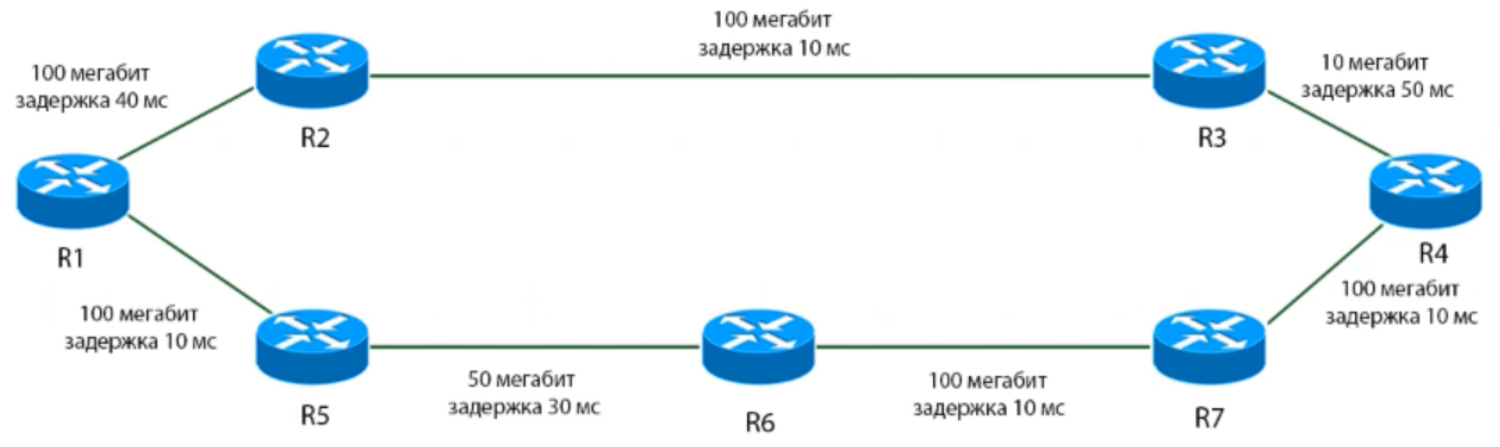




1. Почему Роутер принял такое решение? Есть такое понятие как метрика, это число, которое показывает, насколько маршрут плох или хорош. Чем значение меньше - тем лучше. Трафик всегда направляется по пути с меньшей метрикой. т.е. по тому каналу, который будет быстрее передавать пакеты. На скрине в оранжевой рамке, **первое** - то, как посчитал метрику наш роутер, **второе** то, как посчитал метрику соседний роутер. Т.е. второй роутер сообщает с какой метрикой он может добраться до той или иной сети и так делают все роутеры, в итоге **первое значение** является **конечной (суммарной) метрикой**.



1. Давайте введем такие понятия как **Successor**
2. - маршрут, который выгоден в настоящий момент времени (является маршрутом через который будет проходить трафик) и **Feasible Successor** - маршрут, который является запасным, по нему трафик пойдёт если что-то случится с Successor. Бывают и другие маршруты, например 3,4,5 и т.д. маршруты, которые тоже возможны. Их можно увидеть, написав команду **show ip eigrp topology all-links**
3. Как высчитывается метрика?



**Метрика** = (/minimum bandwith в килобитах)\*256 + ([задержка 1]\*100 + [задержка 2]\*100 + ... + [задержка х]\*100)\*256

**Minimum bandwith** - самое узкое место “трассы” по скорости. На нашем рисунке, у верхней части 10 мб и 50 мб в нижней.

**Задержки** - задержки создают сетевое оборудование (Свичи и Роутеры).

Посчитайте, где пойдет трафик в нашем рисунке?

Протокол EIGRP использует только эти два показателя для расчета метрики, на самом деле можно включить более детальный расчет метрик, тогда параметров будет больше:



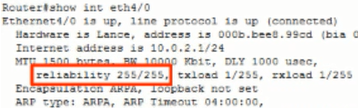
В данной формуле К - это некоторый умножающий коэффициент, которые можно задавать вручную.

Здесь используется показатель **надёжности** и **загрузки** интерфейса. Эти показатели динамичные, метрика будет плавать, возможны частые переключения с одного маршрута на другий. У этого есть один негативный момент, при переключении маршрута, Роутер получит временное отсутствие доступа к сети. Это не желательно, поэтому по дефолту это не используется, но такая возможность есть.

**Загрузка** - насколько загружен Роутер, сколько через него трафика сейчас проходит.

**Надежность** - насколько часто Роутер уходил в отказ за месяц работы.

1. Посмотрим показатели задержки и надёжности в CPT, отправим команду **show int eth4/0** мы увидим надёжность, это параметр reliability, на нашем снимке она максимальна, **250 из 250**. Так же мы видим задержку, в **1000 микросекунд**.





1. А что есть метрика одинаковая для нескольких маршрутов? Тогда выполняется **балансировка трафика**, когда **пакеты отправляются поочередно то по одной то по другой трассе**. В жизни встречается такое довольно часто.
2. Можно включить балансировку трафика, для маршрутов с неравной стоимостью (не равной метрики). Такое можно сделать только в EIGRP. Давайте еще раз проверим в режиме симуляции, что трафик сейчас идёт через Роутер 4. Отправим ICMP (пинг) пакет с LAN 1 на LAN 4.
3. После проверки давайте включим балансировку с неравной метрикой. Для этого заходим на Роутер 1, в конфигурацию eigrp 1 и прописываем команду **variance 128** (этой командой мы устанавливаем во сколько раз наш запасной маршрут может быть хуже, нужно указать целое число от 1 до 128) такой командой мы говорим что запасной маршрут может быть в 128 раз хуже, но балансировка всё равно будет выполнятся, т.е. пакеты будут отправляться поочередно по каждому из маршрутов.
4. Балансировку неравного маршрута можно выполнить только между Successor и Feasible Successor (может быть несколько). Если маршрут настолько плох что он даже не смог стать Feasible Successor, то нельзя будет использовать балансировку.
5. Давайте теперь в режиме симуляции посмотрим как пойду пакеты с LAN 1 до LAN 4. У вас должно получится так, что пакеты поочерёдно будут ходить, через Роутер 4 и напрямую к Роутер 2.
6. Мы можем настроить еще хитрее, можем сделать так, чтобы бОльшая часть трафика шла по лучшему маршруту, а меньшая по худшему. Для этого мы можем использовать команду **traffic-share balanced** (найдите правила её применения и описание в методичке с командами). Но к сожалению **CPT** этой команды не поддерживает, но она есть и она работает на реальном железе.
7. Еще одна очень интересная команда, **metric maximum-hops** 1-255 - она задаёт количество прыжков (роутеров), которое пакет может преодолеть, т.е. есть сеть находится за 256 роутерами, то она считается недостижимой.
8. Балансировка маршрута по дефолту работает максимум на 4 маршрута, чтобы задать большее количество маршрутов, введите команду **maximum-paths 6**. Где 6 - количество маршрутов. Но она опять же не поддерживается в **CPT**.
9. Когда роутер делает метрику, ему надо убедиться что сеть не содержит [петель маршрутизации](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%80%D1%88%D1%80%D1%83%D1%82%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B5%D1%82%D0%BB%D1%8F). Может быть такая ситуация, что Роутер 1 хочет отправить пакет сети X, он знает что до неё есть маршрут через Роутер 2, но Роутер 2 знает маршрут до сети Х через Роутер 1 и они начнут гонять пакеты между собой. Это называется **петля маршрутизации**. Для того чтобы это не происходило, существует правило **split-horizon**, если Роутер узнал о сети через интерфейс Х, то он не будет по этому интерфейсу Х рассказывать что он знает эту сеть. Это правило можно отключить, для специфичных задач. Тогда используется проверка метрик, цена метрики полученная от соседа будет выше, чем метрика просчитанная самим Роутером. Следовательно он сделает вывод что на этот интерфейс посылать пакет не надо.
10. Давайте выключим правило **split-horizon** на Роутер 4. Для этого зайдем на интерфейс соединяющий с Роутер 1 и введем команду **no ip split-horizon**. Теперь Роутер 4 рассказывает Роутер 1 про LAN 1,2,5. Но маршруты не будут нарушены, т.к. действует вспомогательный способ контроля метрик.