# 

# Лабораторная работа №10. Протоколы динамической маршрутизации IGP

Продолжим изучать протоколы которые работают внутри нашей организации (IGP).



* EGP - exterior gateway protocol (протокол внешнего шлюза)
* IGP - interior gateway protocol (протокол внутреннего шлюза)

## Типы протоколов динамической маршрутизации

Дистанционно-векторные протоколы (Distance Vector Protocols)

* RIP v2
* EIGRP

Протоколы состояния каналов связи (Link-State Protocols)

* OSPF
* IS-IS

Протоколы выбора маршрута по пути (Path Vector Protocols)

* BGP

В прошлых лабораторных мы рассматривали **дистанционно-векторные протоколы** на примере EIGRP. Эти протоколы работаю через систему обмена сообщениями между соседними роутерами, передавая метрики и доступные сети. Каждый роутер ничего не знает о том как устроена сеть, он не может гарантировать что маршрут не содержит петель, у него просто есть информация что вот до этой сети можно добраться через этот и через этот роутер и говорят, какая метрика, он не знает как дальшей пакет пойдёт.

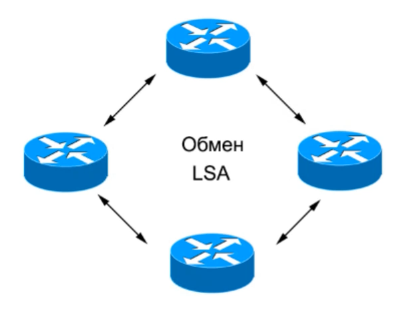
Протоколы класса **состояния каналов связи** работают по другой логике:

1. Запускается роутер, узнает какие у него есть коннектед сетки;
2. После этого начинает всем рассказывать о том какие у него коннектед сетки есть;
3. Так поступает каждый роутер;
4. В итоге все роутеры знают всё обо всех;
5. После того как все узнали обо всех, каждый роутер строит карту топологий, независимую друг от друга;
6. Каждый роутер независимо от других видит сеть, он знает где у кого какие сетки, кто с кем соединен;
7. Каждый роутер самостоятельно просчитывает кратчайший маршрут, это избавляет нас от возможности появления петель.

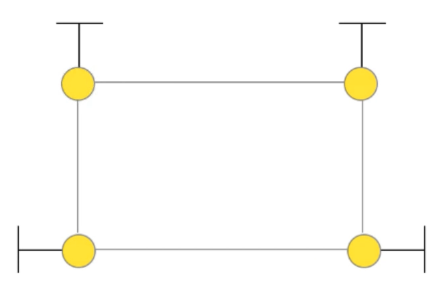
Протоколы состояния каналов связи гораздо быстрее реагируют на изменения в сети, тот же EIGRP после потери связи начинает обмениваться Query и Reply и пытаться выяснить как же добраться. В протоколах Link-State роутеры знают всю карту сети и моментально реагируют на изменение.

## Алгоритм построения маршрутов OSPF (Open Shortest Path First)

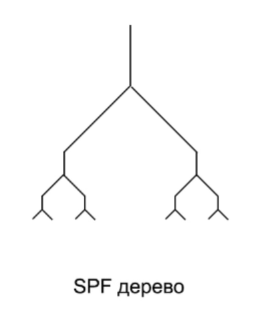
1. Роутеры с помощью служебных пакетов LSA обмениваются информацией о своих connected сетях



2. Каждый роутер после обмена LSA создает таблицу топологии



3. По этой карте до всех сетей входящих в топологию считается самой короткий маршрут с помощью алгоритма DIJKSTRA (Дейкстра) и строится дерево маршрутов (Shortest Path First дерево)



4. Из SPF дерева выбирается лучший маршрут, который попадает в таблицу маршрутизации



Иными словами **при Distance Vector** роутеры **общаются только со своими соседями**, а **при Link-State** все Роутеры общаются **со всеми** и строят у себя **полную картину сети** и мы получаем более высокую сходимость сети.

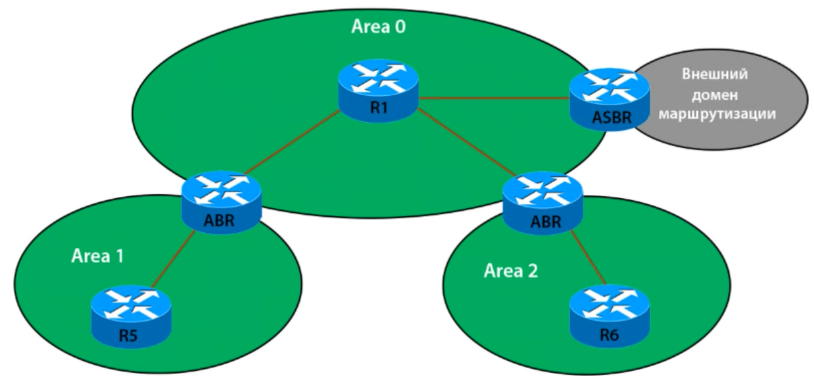
Сходимость сети - насколько быстро сеть вернется в рабочее состояние после сбоя.

У протокола **EIGRP** есть огромный минус, он долгое время являлся **проприетарным** и его можно было использовать только на оборудовании CISCO. Сейчас его уже можно использовать на другом оборудовании, однако другие вендоры пока не подхватили эту идею.

Протокол **OSPF мультивендорный**, открытый и поддерживается всеми вендорами. Наверняка он будет **чаще всего использовать в корпоративной сети**, его настройка может занимать порядка 10 минут. Но мы посмотрим его поглубже, чтобы вы изучили более тонкие настройки.

Почему протоколы назвали **Distance Vector**? Потому что роутер выбирает **направление (вектор)** и **знает дистанцию (метрику)**. **Link-State** так называются потому что он постоянно **смотрит на состояние каналов своей связи**. Для понимания работы с OSPF введем понятия регионов.

## Регионы OSPF



Роутерам мы можем **назначать различные регионы**, это сделано для крупных сетей, чтобы **снизить нагрузку на роутер**, мы понимаем что просчеты топологий и содержание карты маршрутов внутри роутера требует ресурсов, если сеть очень большая (более 100 роутеров) - роутер может не справиться.

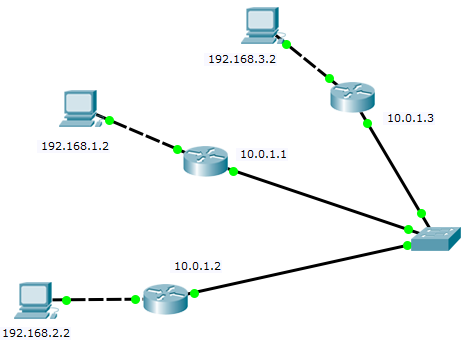
Роутеры в одном регионе строят карту сети по честному, но роутер из Региона 1 не знает о маршрутах в Регионе 2. Если ему нужно обратиться к сети из региона , от отправит на **ABR роутер** пакет, а тот уже разберется куда его отправить.

Заметим что все регионы соединены через нулевой регион, это сделано для защиты от петель, именно так должна строиться сеть. Как вы уже наверно догадались, наши **ABR роутеры** будут работать **как Distance Vector**.

Есть еще такие роутеры как **ASBR**, это роутер который сидит на границе двух автономных систем, как правило **на границе с Интернетом**. Он сообщает нам о маршрутах которые зародились во внешней сети.

## Перейдём к практике

1. Создадим в CPT 3 роутера с двумя сетевыми интерфейсами и соединим их через свитч, пропишем адреса для этих роутеров и добавим по одному ПК к каждому роутеру.



1. Для того чтобы нам комфортнее было работать в консоли IOS, пропишем exec-timeout 0 0 т.е. время работы в консоли у нас будет неограничено.



1. Попробуёте пропинговать с ПК (192.168.2.2) на ПК (192.168.3.2), пинг у нас не пойдет, роутеры пока не владеют маршрутами.
2. Настроим маршруты, для этого перейдите в консоль роутера и введите:



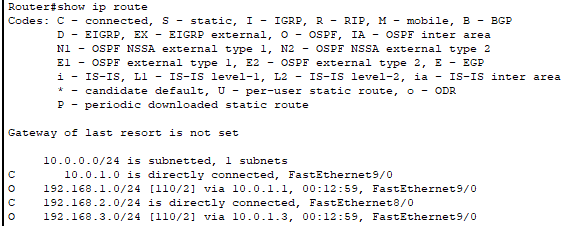
при этом экземпляры на роутерах могут не совпадать.

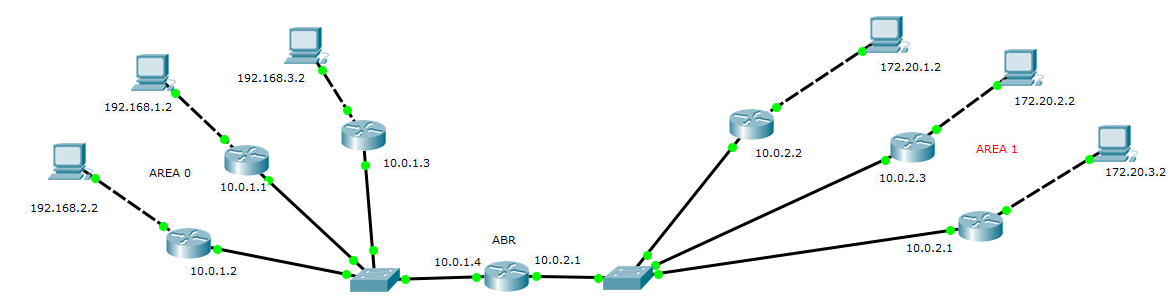
1. Далее мы анонсируем сетку, например мы на Роутере (10.0.1.1) анонсируем сетку 192.168.1.0, тогда получится команда:



указание региона (area) обязательно.

1. Не забываем что нужно анонсировать вторую сеть (10.0.1.0) к которой подключен роутер. Регион тот же.
2. Проведите анонс сетей для других роутеров, для дополнительной информации воспользуйтесь [маном](https://drive.google.com/drive/u/0/folders/1CAmrI3KvKim5xwdLWiSv9nMQLmno4S2s).
3. После проведения анонса введите на любом роутере команду **show ip route**, мы видим маршруты полученные по OSPF, теперь пинг будет работать.

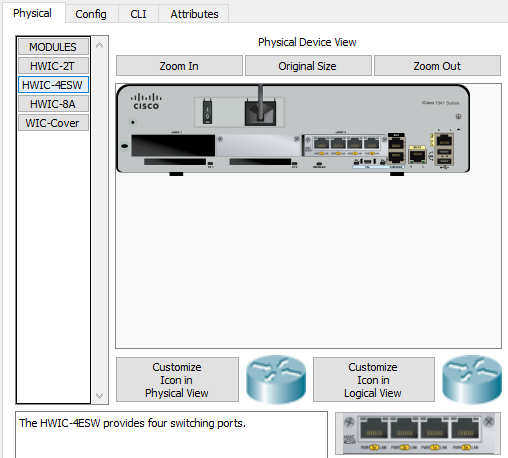


1. Теперь давайте создадим еще один регион и пограничный ABR роутер, должна получится вот такая сеть: 
2. Проведите настройку первого региона, так же как вы настроили нулевой

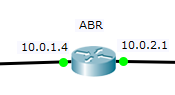
Немного отвлечемся, все узнали что это?



Это знаменитый роутер DIR-300. Вопрос, к нему так же можно подключить 4 или даже 5 разных канальных сред и он будет их маршрутизировать? На самом деле нет, это роутер с функцией свитча. В CTP тоже такой есть в роутере 1941 мы можем установить модуль свитча, тогда бы мы могли обойтись, например без одного из свитчей на нашей схеме.



1. Перейдем к конфигурированию ABR роутера, вводим команды



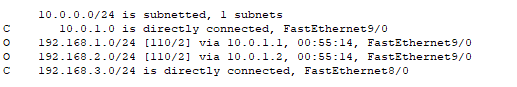






Мы настроили наш пограничный роутер, у которого интерфейсы смотрят в разные регионы.

1. Давайте теперь проверим таблицу маршрутизации на любом роутере, почему то мы не видим сети в другом регионе,



1. Перейдём на ABR роутер, давайте посмотрим какая конфигурация анонсирования записана на нём, введем



Мы видим что всё в норме:





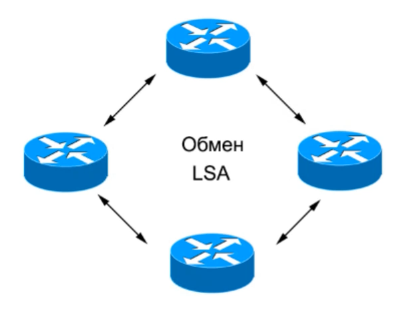


1. Давайте посмотрим какие роутеры видны по соседству, введем команду



вроде всё тоже верно, в чём же проблема?

1. На самом деле у нас не верно построено взаимодействие между роутерами. Мы говорили что все роутеры обмениваются со всеми роутерами с помощью LSA пакетов



1. На самом деле это не совсем так, для экономии траффика LSA сообщения идут не каждый на каждого, на самом деле обмен идет через главный(DR) роутер. Как он выбирается? Конечно же сети это демократическая среда, по этому происходят выборы :) на выбранный роутер отправляют LSA сообщения а он всем отвечает.
2. У нас 4 роутера в одной канальной среде, и им надо выбрать DR, в нашем случае им должен быть выбран именно пограничный роутер, который смотрит на два региона. Видимо DR у нас выбран другим роутером, это не правильно.
3. Давайте вычислим кто сейчас является DR, для этого посмотрим таблицу соседей и мы сразу видим кто у нас является DR:





1. Для того чтобы наш ABR роутер стал DRом, надо изменить его приоритет, по умолчанию он равен 1. Давайте просто поставим нашему роутеру приоритет 100.
2. Приоритет назначается за регион, для этого надо назначить его **на интерфейс**, заходим **на каждый интерфейс** и прописываем приоритет = 100.





Тоже самое надо сделать для второго интерфейса данного роутера.

1. Теперь необходимо перезагрузить наш ABR роутер, введём команду



и подтвердить, ввести “y”.

1. После этого произойдут перевыборы и DRом стал шар ABR роутер. Теперь каждый роутер знает маршрут до другого роутера. Маршруты которые зародились в другом регионе помечаются как IA.



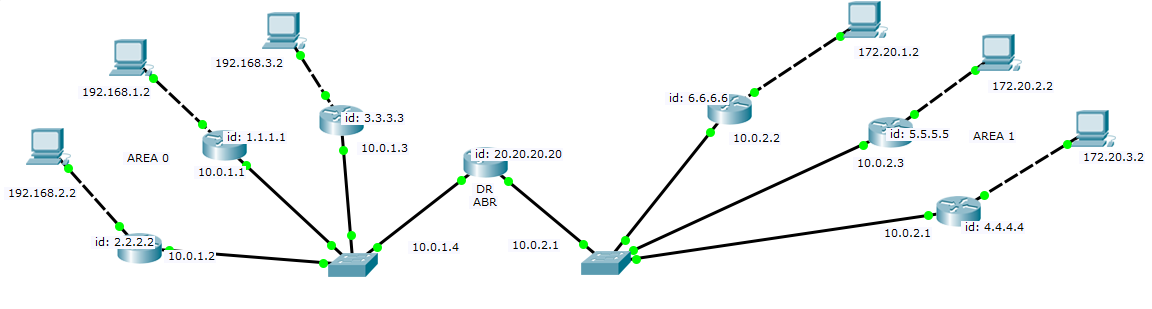
1. У каждого роутера есть Neighbor ID, когда роутер просыпается он придумывает себе Neighbor ID, он берёт его со своих интерфейсов, выбирает тот у которого самый большой IP адрес. Это не очень удобно, мы можем задать этот ID для роутера вручную.
2. Давайте зайдем в консоль нашего DR роутера и пропишем ему ID = 20.20.20.20 (хорошим тоном считается задавать для DR самый большой ID в сети)





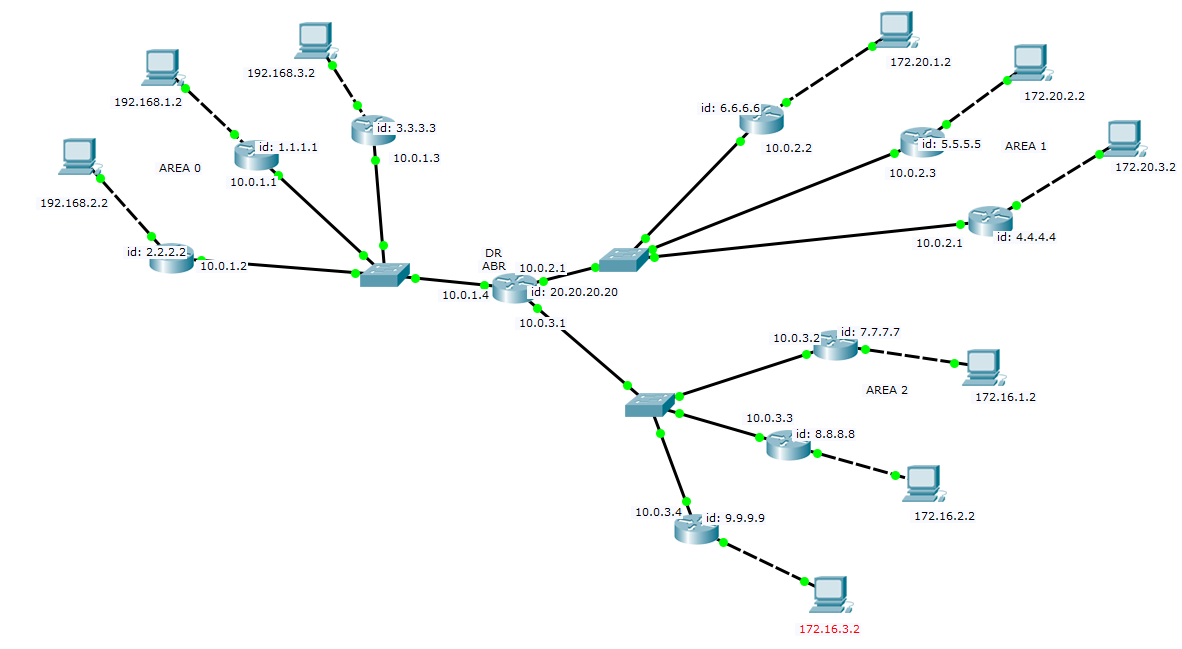
Это **НЕ IP адрес**, это просто число.

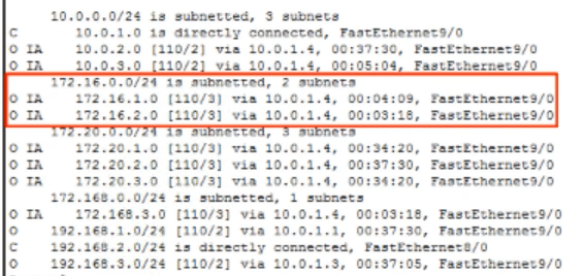
1. Задайте такие ID каждому роутеру:



1. Теперь мы видим что у каждого роутера есть осмысленный ID



1. Интересный факт. Если в сети приоритет у всех роутеров одинаковый, то при проведении выборов, DR-ом станет тот роутер, у которого ID самый высокий.
2. Кроме DR в сети еще есть BDR - это запасной DR, который станет DR-ом если основной перестанет функционировать, он также принимает LSA и накапливает их, но не раскидывает, это делается для того чтобы сеть могла функционировать сразу после сбоя у DR. BDR выбирается так же по приоритету или по номеру ID. Есть еще DROTHER - это обычный роутер.
3. Мы можем выставить приоритет = 0, тогда роутер никогда не будет ни BDR ни DR.
4. Есть еще тип взаимодействия, FULL и 2WAY. Если FULL - то данный роутер будет обмениваться LSA пакетами, если 2WAY - обмен не происходит, они просто соседи.
5. Теперь давайте создадим второй регион, AREA 2 и настроим его, как показано на схеме ниже:
6. Произведем анонсирование второго региона на всех роутерах второго региона.
7. Теперь давайте проверим что знает роутер(id: 2.2.2.2) про AREA 2
8. Наберём на нём #show ip route, мы видим что данный роутер знает о сетях зародившихся во втором регионе



1. Как еще можно работать с этой таблицей? Например мы хотим посмотреть только коннектед сети, тогда на этом же роутере наберите команду #show ip route connected и вы получите таблицу только с коннектед сетками:



1. Можете проверить работоспособность сети, отправив команду ping с компьютера с IP адресом 172.16.3.2 на компьютер с IP адресом 192.168.2.2.

Мы завершили базовую настройку, познакомились с регионами, понятиями DR и BDR, посмотрели как работают протоколы класса **Link-State** и чем они отличаются от **Distance Vector**