

# OSPF (Open Shortest Path First)

## Описание

Продолжаем знакомиться с протоколами динамической маршрутизации.

Следующий на очереди - **OSPF** (Open Shortest Path First)

# Оглавление

1. Протокол OSPF
2. Поднимаем OSPF на оборудовании Cisco
3. Поднимаем OSPF в Cisco Packet Tracer
4. Протокол маршрутизации OSPF: LSA, области и виртуальные ссылки
5. Расширенные возможности OSPF: Области
6. OSPF: создание конкретных типов областей
7. Ручная фильтрация маршрутов OSPF
8. Поиск и устранение проблем в OSPF
9. Траблшутинг OSPF

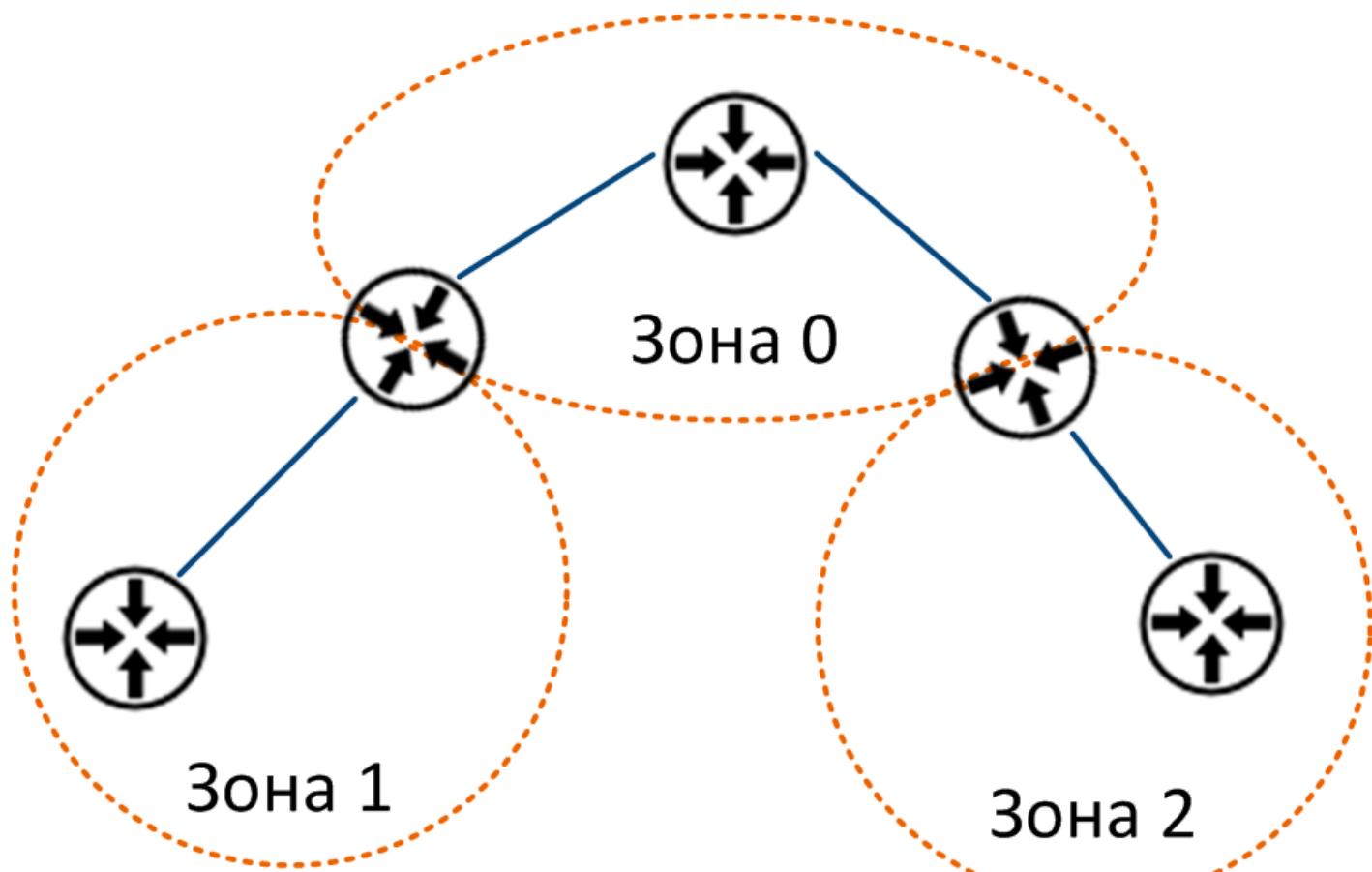
# 1. Протокол OSPF

**OSPF (Open Shortest Path First)** – дословно переводится как «Сперва открытый короткий путь» – надежный протокол внутренней маршрутизации с учетом состояния каналов (Interior gateway protocol, **IGP**). Как правило, данный протокол маршрутизации начинает использоваться тогда, когда протокола RIP уже не хватает по причине усложнения сети и необходимости в её легком масштабировании.

OSPF наиболее широко используемый протокол внутренней маршрутизации. Когда идёт речь о внутренней маршрутизации, то это означает, что связь между маршрутизаторами устанавливается в одном домене маршрутизации, или в одной автономной системе. Представьте компанию среднего масштаба с несколькими зданиями и различными департаментами, каждое из которых связано с другим с помощью канала связи, которые дублируются с целью увеличения надежности. Все здания являются частью одной автономной системы. Однако при использовании OSPF, появляется понятие «площадка», «зона» (Area), которое позволяет сильнее сегментировать сеть, к примеру, разделение по «зонам» для каждого отдельного департамента.

Для понимания необходимости данных «зон» при проектировании сети, необходимо понять, как OSPF работает. Есть несколько понятий, связанных с этим протоколом, которые не встречаются в других протоколах и являются уникальными:

- **Router ID:** Уникальный 32-х битный номер, назначенный каждому маршрутизатору. Как правило, это сетевой адрес с интерфейса маршрутизатора, обладающий самым большим значением. Часто для этих целей используется loopback интерфейс маршрутизатора.
- **Маршрутизаторы-соседи:** Два маршрутизатора с каналом связи между ними, могут посыпать друг другу сообщения.
- **Соседство:** Двухсторонние отношения между маршрутизаторами-соседями. Соседи не обязательно формируют между собой соседство.
- **LSA: Link State Advertisement** – сообщение о состоянии канала между маршрутизаторами.
- **Hello сообщения:** С помощью этих сообщений маршрутизаторы определяют соседей и формируют LSA
- **Area (Зона):** Некая иерархия, набор маршрутизаторов, которые обмениваются LSA с остальными в одной и той же зоне. Зоны ограничивают LSA и стимулируют агрегацию роутеров.



OSPF – протокол маршрутизации с проверкой состояния каналов. Представьте себе карту сети – для того, чтобы ее сформировать, OSPF совершает следующие действия:

1. Сперва, когда протокол только запустился на маршрутизаторе, он начинает посыпать hello-пакеты для нахождения соседей и выбора **DR** (designated router, назначенный маршрутизатор). Эти пакеты включают в себя информацию о соседях и состоянии каналов. К примеру, OSPF может определить соединение типа «точка-точка», и после этого в протоколе данное соединение «поднимается», т.е. становится активным. Если же это распределенное соединение, маршрутизатор дожидается выбора DR перед тем как пометить канал активным.
2. Существует возможность изменить **Priority ID** для, что позволит быть уверенным в том, что DR-ом станет самый мощный и производительный маршрутизатор. В противном случае, победит маршрутизатор с самым большим IP-адресом. Ключевая идея **DR** и **BDR (Backup DR)**, заключается в том, что они являются единственными устройствами, генерирующими LSA и они обязаны обмениваться базами данных состояния каналов с другими маршрутизаторами в подсети. Таким образом, все не-DR маршрутизаторы формируют соседство с DR. Весь смысл подобного дизайна в поддержании масштабируемости сети. Очевидно, что единственный способ убедиться в том, что все маршрутизаторы оперируют одной и той же информацией о состоянии сети – синхронизировать БД между ними. В противном случае, если бы в сети было 35 маршрутизаторов, и требовалось бы добавить еще одно устройство, появилась бы необходимость в установлении 35 процессов соседства. Когда база централизована (т.е существует центральный, выбранный маршрутизатор - DR) данный процесс упрощается на несколько порядков.
3. **Обмен базами данных** – крайне важная часть процесса по установлению соседства, после того как маршрутизаторы обменялись hello-пакетами. При отсутствии синхронизированных баз данных могут появиться ошибки, такие как петли маршрутизации и т.д. Третья часть установления соседства – обмен LSA. Это понятие будет разобрано в следующей статье, главное, что необходимо знать – нулевая зона (Area 0) особенная, и при наличии нескольких зон, все они должны быть соединены с Area 0. Так же это называется магистральной зоной.

## ТИПЫ МАРШРУТИЗАТОРОВ OSPF

Разберем различные типы маршрутизаторов при использовании протокола OSPF:

- **ABR**

Area Border Router – маршрутизатор внутри нулевой зоны, через который идет связь с остальными зонами

- **DR, BDR**

Designated Router, Backup Designated Router – этот тип маршрутизаторов обсуждался выше, это основной и резервирующий маршрутизаторы, которые ответственны за базу данных маршрутизаторов в сети. Они получают и посыпают обновления через Multicast остальным маршрутизаторам в сети.

- **ASBR**

Autonomous System Boundary Router – этот тип маршрутизаторов соединяет одну или несколько автономных систем для осуществления возможного обмена маршрутами между ними.

## Видео: протокол OSPF (Open Shortest Path First) за 8 минут



## ПОДВЕДЕМ ИТОГИ

- OSPF является быстро сходящимся протоколом внутренней маршрутизации с контролем состояния каналов
- Процесс соседства формируется между соседними маршрутизаторами через DR и BDR, используя LSA

- Зоны в данном протоколе маршрутизации используются для ограничения LSA и суммирования маршрутов. Все зоны подключаются к магистральной зоне.

## 2. Поднимаем OSPF на оборудовании Cisco

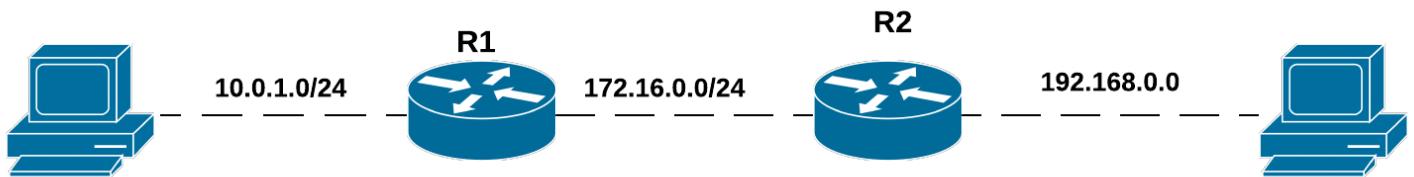
Настройка **OSPF (Open Shortest Path First)** довольно проста и состоит из двух основных шагов:

- включения протокола глобальной командой `router ospf PROCESS_NUMBER`;
- выбора сетей, которые протокол будет «вещать», для чего используется команда(ы) `network 255.255.255.255 0.0.0.255 AREA_NUMBER`.

Как сразу заметно, в OSPF появляется указание «зоны» - area. Первая команда включения говорит сама за себя, но поясним про **PROCESS\_NUMBER** и **AREA\_NUMBER** – это номер процесса и номер зоны соответственно. Для установления соседства номер процесса OSPF не должен быть одинаковым, но обязательно должен совпадать номер зоны. Интерфейсы и сети указываем через обратную маску.

### Пример настройки OSPF

В нашей топологии у маршрутизаторов R1 и R2 есть напрямую подключенные подсети.



Нам нужно включить данные подсети в процесс динамической маршрутизации OSPF. Для этого нам сначала нужно включить OSPF на обоих маршрутизаторах и затем «вещать» данные сети с помощью команды **network**. На маршрутизаторах переходим в глобальный режим конфигурации и вводим следующие команды, в соответствии с нашей схемой:

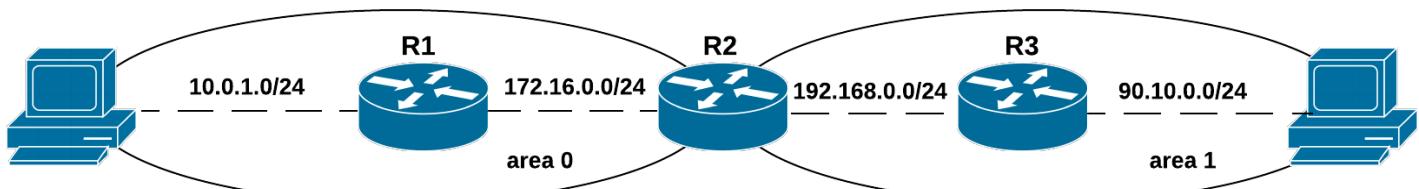
```
router ospf 1
network 10.0.1.0 0.0.0.255 area 0
network 172.16.0.0 0.0.0.255.255 area 0
```

```
router ospf 1
network 192.168.0.0 0.0.0.255 area 0
network 172.16.0.0 0.0.0.255.255 area 0
```

Далее нам нужно проверить, заработала ли динамическая маршрутизация, и для этого используем команды `show ip ospf neighbors` и `show ip route`. Вот и все – также просто, как и настроить RIP: главное не забывать указывать одинаковый номер автономной системы. Первая команда должна показать «соседа» – на обоих маршрутизаторах убедитесь, что там указан адрес другого маршрутизатора в выводе данной команды. Вторая команда выведет таблицу маршрутизации, и, маршруты, получаемые по OSPF, будут отмечены буквой **O**.

### Второй сценарий настройки OSPF

По первому примеру видно, что настройка OSPF довольно проста. Однако, этот протокол маршрутизации имеет довольно много разнообразных фич, которые сильно усложняют процесс настройки, но и делают OSPF очень гибким протоколом. В нашем примере мы настроим мультизонный (**multiarea**) OSPF с некоторыми дополнительными функциями.



В нашем примере у нас есть две зоны OSPF, area 0 и area 1. Как видно на схеме, маршрутизаторы R1 и R2 находятся в зоне 0, и R2 и R3 в зоне 1. Так как R2 соединяет две зоны, он становится **ABR** – Area Border Router (границальным маршрутизатором). Нашей задачей является вещание подсетей, напрямую подключенных к R1 и R3. Для этого, на R1 введем следующую команду:

```
router ospf 1
network 10.0.1.0 0.0.0.255 area 0
network 172.16.0.0 0.0.255.255 area 0
router-id 1.1.1.1
```

Мы вручную указали идентификатор маршрутизатора, и теперь процесс OSPF будет использовать данный RID при общении с другими OSPF соседями.

Так как R1 подключен только к R2, нам необходимо установить соседство с R2 и вещать напрямую подключенные сети через OSPF. Настройки на R3 выглядят такими же, как на R1, но с другим номером зоны.

```
router ospf 1
network 192.168.0.0 0.0.0.255 area 1
network 90.10.0.0 0.0.0.255 area 1
router-id 3.3.3.3
```

Теперь перейдем к настройке R2 – так как он является граничным маршрутизатором, необходимо установить соседство и с R1 и с R3. Для этого, нам необходимо настроить отдельное соседство для каждой зоны – 0 для R1 и 1 для R2.

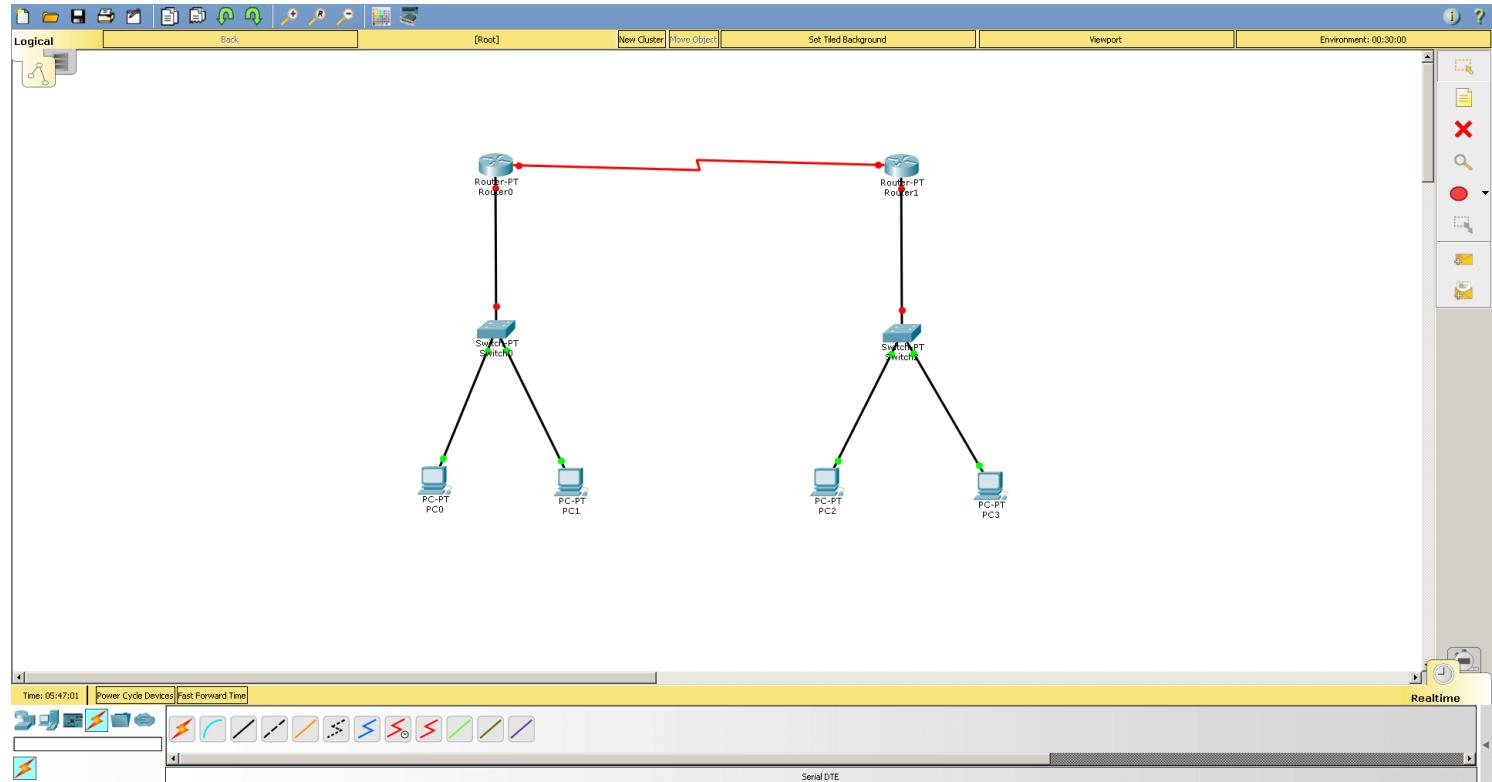
```
router ospf 1
network 172.16.0.0 0.0.255.255 area 0
network 192.168.0.0 0.0.0.255 area 1
router-id 2.2.2.2
```

Для проверки используем команды `show ip ospf neighbor` и `show ip route ospf` на маршрутизаторах R1 и R3. Буквы **IA** означают, что данные маршруты находятся в разных зонах.

Так как R1 и R3 находятся в разных зонах, между ними никогда будет соседства.

### 3. Поднимаем OSPF в Cisco Packet Tracer

Создадим такую схему – два маршрутизатора, соединенные друг с другом посредством серийного интерфейса, к каждому подключено по коммутатору, а уже к ним подключаются компьютеры.



## Настройка сети

Начнем с настройки интерфейсов на роутере. Сначала настроим интерфейс **FastEthernet 0/0**, который смотрит в сторону коммутатора.

Используем команду **int fa 0/0** для входа в режим конфигурации интерфейса, **ip address [ip\_адрес][маска]** для того чтобы назначить ему IP адрес, и команду **no shutdown** для того чтобы включить интерфейс. После этого мы увидим сообщение что интерфейс и протокол теперь находятся в состоянии **UP**.

Physical | Config | CLI | Attributes |

IOS Command Line Interface

```

PT2005 processor: part number 0, mask 01
Bridging software.
X.25 software, Version 3.0.0.
4 FastEthernet/IEEE 802.3 interface(s)
2 Low-speed serial(sync/async) network interface(s)
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)

--- System Configuration Dialog ---

Continue with configuration dialog? [yes/no]: n

Press RETURN to get started!

Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int fa 0/0
Router(config-if)#ip address 192.168.1.10 255.255.255.0
Router(config-if)#no sh

Router(config-if)#
*LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
*LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

```

Top

```

en
conf t
int fa 0/0
ip address 192.168.1.10 255.255.255.0
no sh

```

Затем настроим интерфейс **Serial 2/0**.

Тут все тоже самое, но поскольку это интерфейс serial, то мы еще выполняем команду **clock rate**, чтобы задать скорость в 64000 бита в секунду. Эту команду мы должны выполнить на маршрутизаторе, который является **DCE (Data Communication Equipment)**. Это зависит от того, каким концом подключен serial кабель, так как он не симметричный. Чтобы посмотреть, какая роль у данного порта нужно использовать команду **show controllers [интерфейс]**, где мы увидим, является он **DCE** или **DTE**.

```

Router(config)#
Router(config)#int serial 2/0
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#clock rate 64000
Router(config-if)#no sh
Router(config-if)#exit

```

```

en
conf t
int serial 2/0
ip address 192.168.2.1 255.255.255.0
clock rate 64000
no sh
exit

```

Аналогичные настройки проведем для другого маршрутизатора, за исключением того, что он является **DTE (Data Terminal Equipment)** и использовать команду **clock rate** не нужно.

```

IOS Command Line Interface
Continue with configuration dialog? (yes/no): no

Press RETURN to get started!

Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 192.168.3.10 255.255.255.0
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

Router(config-if)#exit
Router(config)#int serial 2/0
Router(config-if)#ip address 192.168.2.2 255.255.255.0
Router(config-if)#no sh

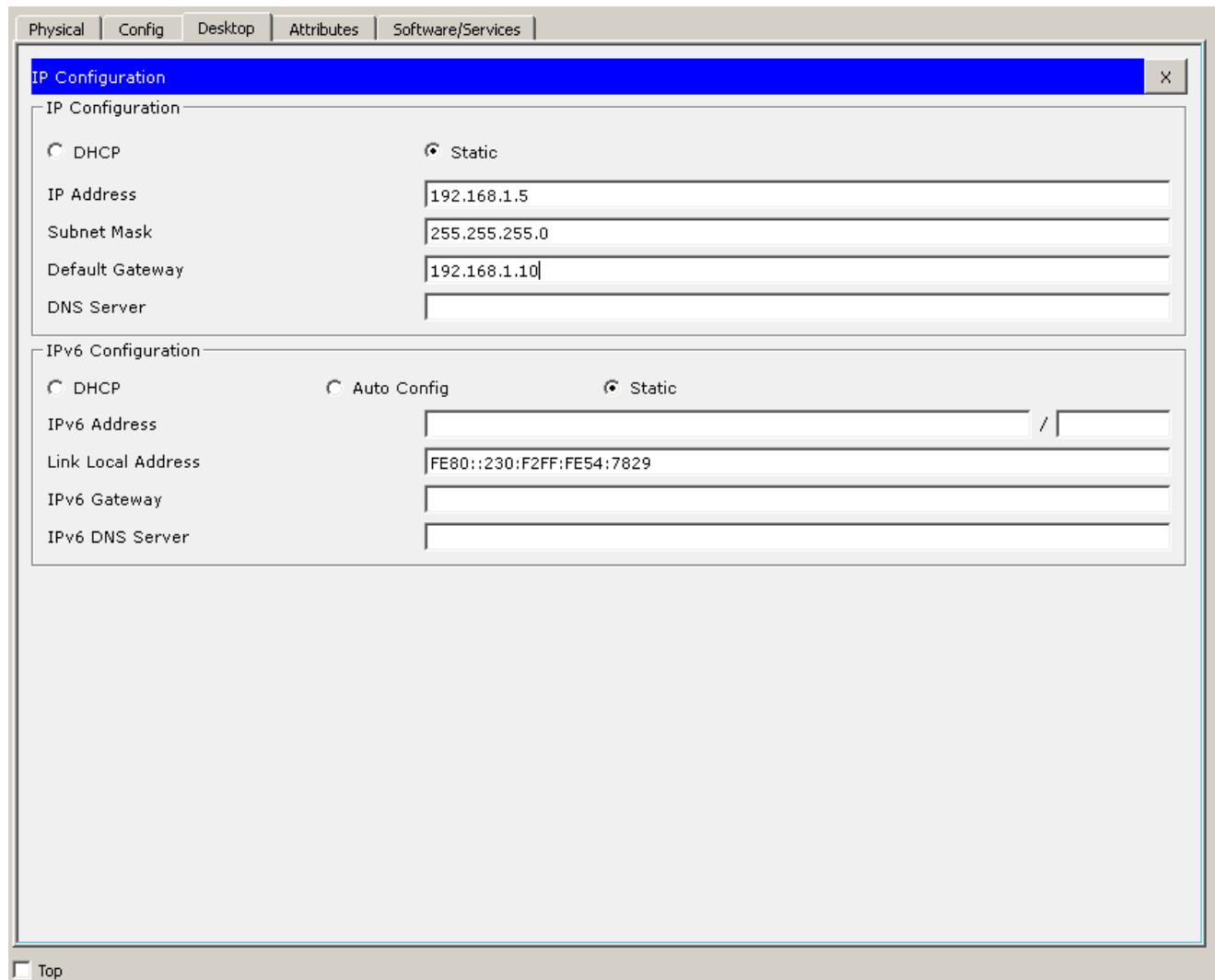
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0,
changed state to up

```

После этого все интерфейсы должны находиться в состоянии **UP** и стать зелеными.

Теперь присвоим IP адреса компьютерам. Для этого дважды кликнем иконку компьютера, перейдем во вкладку **Desktop** и нажмем на **IP Configuration**. В открывшемся окне выбираем опцию **static** в блоке IP Configuration и задаем необходимый IP адрес, маску и шлюз, в соответствии с нашей схемой.



То же самое проделываем и для остальных ПК.

После этого можно проверить доступность, используя команду **Ping**, зайдя на ПК во вкладку **Desktop – Command Prompt**.

## Command Prompt

X

```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:

Reply from 192.168.1.10: bytes=32 time=14lms TTL=255
Reply from 192.168.1.10: bytes=32 time=1lms TTL=255
Reply from 192.168.1.10: bytes=32 time<1ms TTL=255
Reply from 192.168.1.10: bytes=32 time=1lms TTL=255

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 14ms, Average = 38ms

C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Reply from 192.168.2.1: bytes=32 time<1ms TTL=255
Reply from 192.168.2.1: bytes=32 time=16ms TTL=255
Reply from 192.168.2.1: bytes=32 time<1ms TTL=255
Reply from 192.168.2.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 16ms, Average = 4ms

C:\>ping 192.168.3.10

Pinging 192.168.3.10 with 32 bytes of data:

Reply from 192.168.1.10: Destination host unreachable.

Ping statistics for 192.168.3.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

 Top

Как видно на скриншоте у нас есть доступ до шлюза, который находится в нашей подсети 192.168.1.0/24, есть доступ до интерфейса роутера, находящегося в подсети 192.168.2.0/24, потому что он находится на роутере и поэтому попадает в таблицу маршрутизации, но при этом до другого адреса в этой подсети доступа у нас нет. И у нас нет доступа до подсети 192.168.3.0/24.

На данный момент таблица маршрутизации левого роутера выглядит вот так

```
Router#
Router#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
      B - BGP
          D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter
          area
          N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external
          type 2
          E1 - OSPF external type 1, E2 - OSPF external type 2, E -
          BGP
          i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
          IS-IS inter area
          * - candidate default, U - per-user static route, o - ODR
          P - periodic downloaded static route

Gateway of last resort is not set

C      192.168.1.0/24 is directly connected, FastEthernet0/0
C      192.168.2.0/24 is directly connected, Serial2/0
```

Чтобы это исправить и получить доступ до подсети 192.168.3.0/24 настроим протокол маршрутизации OSPF.

## Настройка OSPF

Запускаем процесс OSPF на первом маршрутизаторе. Для этого используем команду **router OSPF [номер процесса]** для запуска протокола, и команду **network [ip\_адрес\_сети][ wildcard\_маска][зона]**, в которой мы указываем все подсети, для которых будет работать OSPF.

Посчитать обратную (**wildcard**) маску можно на нашем калькуляторе подсетей. Просто введите обычный адрес и обычную маску, а калькулятор покажет вам wildcard :)

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#[
```

```
en
conf t
router ospf 1
network 192.168.1.0 0.0.255 area 0
network 192.168.2.0 0.0.255 area 0
network 192.168.3.0 0.0.255 area 0
exit
```

То же самое проделываем на втором маршрутизаторе. Номер зоны должен быть таким же, как и на первом роутере.

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
21:15:12: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.2.1 on Serial2/0
from LOADING to FULL, Loading Done
```

Как видно на скриншоте, процесс OSPF уже сразу заработал. Теперь можно проверить таблицу маршрутизации.

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
      B - EGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter
      area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external
      type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E -
      EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
      IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

      Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, Serial2/0
O    192.168.3.0/24 [110/65] via 192.168.2.2, 00:00:49, Serial2/0

Router#
```

Как мы видим, теперь все сети появились в таблице маршрутизации. Теперь снова проверим сетевую доступность между подсетями 192.168.1.0/24 и 192.168.3.0/24, используя утилиту Ping.

```
C:\>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Reply from 192.168.3.5: bytes=32 time=31ms TTL=126
Reply from 192.168.3.5: bytes=32 time=31ms TTL=126
Reply from 192.168.3.5: bytes=32 time=31ms TTL=126
Reply from 192.168.3.5: bytes=32 time=32ms TTL=126

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 31ms, Maximum = 32ms, Average = 31ms
```

Все пингуется, а это значит, что мы успешно настроили протокол OSPF!

## 4. Протокол маршрутизации OSPF: LSA, области и виртуальные ссылки

Прежде чем перейти к более сложным темам, мы завершим эту серию статей об основах **OSPF**. Здесь мы рассмотрим типы LSA, типы областей и виртуальные ссылки (**LSA types**, **area types**, и **virtual links**).

### OSPF LSA ТИПЫ

**Link State Advertisements (LSA)** — [Объявления](#) состояния канала — это основа работы сетей на OSPF. Наполнение этих обновлений позволяют сети OSPF создать карту сети. Это происходит с помощью алгоритма кратчайшего пути Дейкстры.

Не все LSA OSPF равны. Ниже представлен каждый из них:

- **Router (Type 1) LSA** — мы начинаем с того, что многие называют «фундаментальным» или «строительным блоком» Link State Advertisements. Type 1 LSA (также известный как **Router LSA**) определен в пределах области. Он описывает интерфейсы локального маршрутизатора, которые участвуют в OSPF, и соседей, которых установил локальный спикер OSPF.
- **Network (Type 2) LSA** — вспомните, как OSPF функционирует на (широковещательном) Ethernet сегменте. Он выбирает **Designated Router (DR)** and **Backup Designated Router (BDR)**, чтобы уменьшить количество смежностей, которые должны быть сформированы, и хаос, который будет результатом пересечением этих отношений. Type 2 LSA отправляется назначенным маршрутизатором в локальную область. Этот LSA описывает все маршрутизаторы, которые подключены к этому сегменту.
- **Summary (Type 3) LSA** — напоминаем вам, что Type 1 LSA и Type 2 LSA ретранслируются в пределах области. Мы называем их **intra-area LSA**. Теперь пришло время для первого из наших inter-area LSA. Summary (Type 3) LSA используется для [объявления](#) префиксов, полученных из Type 1 LSA и Type 2 LSA в другой области. Маршрутизатор границы области (**ABR**) — это устройство OSPF, которое разделяет области, и именно это устройство рекламирует Type 3 LSA.

Изучите топологию OSPF, показанную на рисунке 1 ниже.



Рис. 1: Пример многозональной топологии OSPF

Область 1 ABR будет посылать Type 3 LSA в область 0. ABR, соединяющий область 0 и область 2, отправил эти Type 3 LSA в область 2, чтобы обеспечить полную достижимость в домене OSPF. Type 3 LSA остаются Type 3 LSA во время этой пересылки.

- **ASBR Summary (Type4) LSA** — есть особая роль маршрутизатора OSPF, который называется пограничный маршрутизатор автономной системы **Autonomous System Boundary Router (ASBR)**. Задача этого маршрутизатора заключается в том, чтобы ввести внешнюю префиксную информацию из другого домена маршрутизации. Для того чтобы сообщить маршрутизаторам в различных областях о существовании этого специального маршрутизатора, используется Type 4 LSA. Эта Summary LSA предоставляет идентификатор маршрутизатора ASBR. Таким образом, еще раз, Area Border Router отвечает за пересылку этой информации в следующую область, и это есть еще один пример inter-area LSA.
- **External (Type 5) LSA** — Таким образом, ASBR — это устройство, которое приносит префиксы из других доменов маршрутизации. Type 4 LSA описывает это устройство. Но какой LSA используется для реальных префиксов, поступающих из другого домена? Да, это Type 5 LSA. OSPF ASBR создает эти LSA, и они отправляются к Area Border Routers для пересылки в другие области.

- **NSSA External (Type 7) LSA** - в OSPF есть специальный тип области, называемый **Not So Stubby Area (NSSA)**. Эта область может выступать заглушкой, но она также может вводить внешние префиксы из ASBR. Эти префиксы передаются как Type 7 LSA. Когда ABR получает эти Type 7 LSA, он отправляет по одному в другие области, такие как Type 5 LSA. Таким образом, обозначение Type 7 предназначено только для этой специальной области NSSA.

Другие типы LSA. Существуют ли другие типы LSA? Да. Но мы не часто сталкиваемся с ними. Например, Type 6 LSA используется для многоадресной (**Multicast**) передачи OSPF, и эта технология не прижилась, позволяя **Protocol Independent Multicast** передаче победить. Для завершения ниже показан полный список всех возможных типов LSA:

- LSA Type 1: Router LSA
- LSA Type 2: Network LSA
- LSA Type 3: Summary LSA
- LSA Type 4: Summary ASBR LSA
- LSA Type 5: Autonomous system external LSA
- LSA Type 6: Multicast OSPF LSA
- LSA Type 7: Not-so-stubby area LSA
- LSA Type 8: External Attribute LSA for BGP
- LSA Type 9, 10, 11: "Opaque" LSA типы, используемые для конкретных прикладных целей

## OSPF ТИПЫ LSA И ТИПЫ AREA

Одна из причин, по которой вы должны освоить различные типы LSA, заключается в том, что это поможет вам полностью понять потенциальную важность **multi-area**, особенно такого, который может включать специальные области. Ключом к важности специальных типов областей в OSPF является тот факт, что они инициируют автоматическую фильтрацию определенных LSA из определенных областей.

Например, подумайте о области 1, присоединенной к основной области области 0. Type 1 LSA заполнил область 1. Если у нас есть широковещательные сегменты, мы также имеем Type 2 LSA, циркулирующие в этой области. Area Border Router посылает LSA Type 3 в магистраль для суммирования префиксной информации из области 1.

Этот ABR также принимает эту информацию от магистралей для других областей, которые могут существовать. Если где-то в домене есть ASBR, наша область 1 получит LSA Type 4 и LSA Type 5, чтобы узнать местоположение этого ASBR и префиксы, которыми он делится с нами. Обратите внимание, что это является потенциальной возможностью для обмена большим количеством информации между областями. Именно поэтому у нас есть специальные типы зон!

## OSPF LSAS И STUB AREA (ЗАГЛУШКА)

Для чего предназначена **Stub Area**? Мы не хотим слышать о тех префикса, которые являются внешними для нашего домена OSPF. Помните, у нас был такой случай? Конечно, это LSA Type 5. На самом деле, мы даже не хотим слышать о тех LSA Type 4, которые используются для вызова ASBR в сети. Таким образом, Stub Area заполнена LSA Type 1, Type 2 и Type 3. На самом деле, как эта область могла бы добраться до одного из этих внешних префиксов, если бы это было необходимо? Для этого мы обычно используем очень специальный LSA Type 3. Этот LSA представляет маршрут по умолчанию `(0.0.0.0 / 0)`. Именно этот удобный маршрут позволяет устройствам в этой области добраться до всех этих внешних объектов. На самом деле именно этот маршрутизатор используется для получения любого префикса, специально не определенного в базе данных маршрутизации (**RIB**).

## OSPF LSA И TOTALLY STUBBY AREA (ПОЛНОСТЬЮ КОРОТКАЯ ОБЛАСТЬ)

Использование этой области имеют малые перспективы LSA. Использование этой области имеет смысл тогда, когда мы хотим снова заблокировать Type 4 и 5, но в данном случае мы блокируем даже LSA Type 3, которые описывают информацию префикса из других областей в нашем домене OSPF. Однако здесь имеется одно большое исключение. Нам нужен LSA Type 3 для маршрута по умолчанию, чтобы мы действительно могли добраться до других префиксов в нашем домене.

## OSPF LSAS И NOT SO STUBBY AREA И TOTALLY NOT SO STUBBY AREA

Запомните, что **Not So Stubby Area** должна иметь LSA Type 7. Эти LSA Type 7 допускают распространение тех внешних префиксов, которые входят в ваш домен OSPF благодаря этой созданной вами области NSSA. Очевидно, что эта область также имеет Type 1, Type 2 и Type 3 внутри нее. Type 4 и Type 5 будут заблокированы для входа в эту область, как и ожидалось. Вы также можете создать **Totally Not So Stubby Area**, ограничив Type 3 из этой области.

# VIRTUAL LINKS

Вспомните из нашего более раннего обсуждения OSPF, что все области в автономной системе OSPF должны быть физически связаны с основной областью (область 0). Там, где это невозможно, вы можете использовать виртуальную связь (**virtual link**) для подключения к магистрали через область, не являющуюся магистралью.

Учтите следующие факты о virtual link:

- Они никогда не должны рассматриваться в качестве цели проектирования в ваших сетях. Они являются временным "исправлением" нарушения работы OSPF.
- Вы также можете использовать virtual link для соединения двух частей секционированной магистрали через область, не являющуюся магистралью.
- Область, через которую вы настраиваете virtual link, называемую транзитной областью, должна иметь полную информацию о маршруте.
- Транзитная зона не может быть stub area (заглушкой).

Вы создаете virtual link с помощью команды в режиме конфигурации OSPF:

```
area 1 virtual-link 3.3.3.3
```

Эта команда создает virtual link через область 1 с удаленным устройством OSPF с идентификатором маршрутизатора (Router ID) 3.3.3.3. Вы также настраиваете это удаленное устройство OSPF с помощью команды virtual-link. Например, если наше локальное устройство OSPF находится в Router ID 1.1.1.1, то соответствующая удаленная команда будет:

```
area 1 virtual-link 1.1.1.1
```

Примечание: virtual link — это всего лишь один из способов наладки нарушений в работе OSPF. Вы также можете использовать туннель GRE для исправления сбоев в работе OSPF.

## 5. Расширенные возможности OSPF: Области

Пришло время заняться некоторыми более продвинутыми и интересными функциями протокола маршрутизации **Open Shortest Path First**. Мы начинаем с изучения конфигурации и проверки различных областей **OSPF**. Это упражнение является не только забавным, но и действительно может закрепить знания о том, как эти области функционируют и почему они существуют.

### OSPF LSA TYPES

Области (**Areas**) - это фундаментальная концепция OSPF. Это то, что делает протокол маршрутизации иерархическим, как мы любим говорить.

Существует основная магистральная область (**область 0**), которая соединяется с нормальными, не магистральными областями. Магистраль может также соединяться с особыми типами областей, которые мы подробно рассмотрим в этой группе статей. Такая иерархическая природа конструкции помогает гарантировать, что протокол является очень масштабируемым. Мы можем легко уменьшить или исключить ненужные потоки трафика маршрутизации и связи между областями, если это необходимо.

### МАГИСТРАЛЬНАЯ И НЕ МАГИСТРАЛЬНАЯ ОБЛАСТЬ (BACKBONE AND NON-BACKBONE AREAS)

Вернемся немного назад к нашим предыдущим сообщениям в статьях об OSPF. На рисунке 1 показана простая многозонная сеть. Сейчас я настрою эту сеть, используя мой любимый подход к конфигурации, команду конфигурации уровня интерфейса `ip ospf`. Пример 1 показывает конфигурацию всех трех устройств.

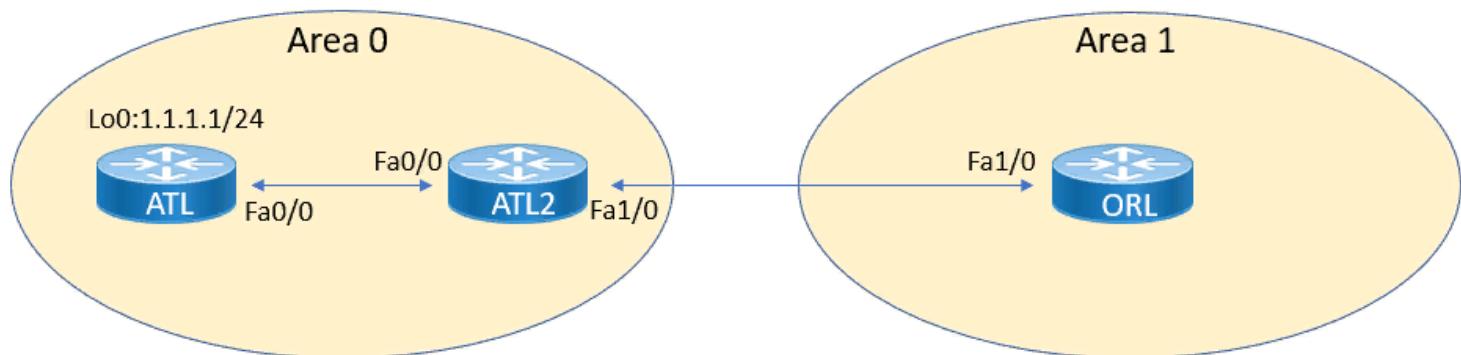


Рисунок 1: Магистральная и не магистральная область (**Backbone и Non-Backbone Areas**)

Пример 1: Настройка магистральных и не магистральных областей

```
ATL Router:
```

```
ATL#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ATL(config)#interface fa0/0
ATL(config-if)#ip ospf 1 area 0
ATL(config-if)#interface lo0
ATL(config-if)#ip ospf 1 area 0
ATL(config-if)#end
ATL#
```

```
ATL2 Router:
```

```
ATL2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ATL2 (config)#interface fa0/0
ATL2 (config-if)#ip ospf 1 area 0
ATL2 (config-if)#interface
*Mar 27 22 :03 :27.815 : %OSPF-5-ADJCHG : Process 1, Nbr 1 .1.1 .1 on
FastEthernet0/0 from LOADING to FULL, Loading Done
ATL2 (config-if)#interface fa1/0
ATL2 (config-if)#ip ospf 1 area 1
ATL2 (config-if)#end
ATL2#
```

```
ORL Router:
```

```
ORL# conf t
Enter configuration commands, one per line. End with CNTL/Z.
ORL( config )#interface fa1/0
ORL(config-if)#ip ospf 1      area 1
ORL(config-if)#end
ORL#
*Mar 27 22 :04:21.515: %OSPF-5-ADJCHG: Process 1, Nbr 10.23.23.2
on FastEthernet1/0 from LOADING to FULL , Loading Done
```

Обратите внимание на простоту этой конфигурации, даже если мы настраиваем довольно сложный протокол маршрутизации. **Area Border Router (ABR)** находится в ATL2 с одним интерфейсом в магистральной и одним в не магистральной области.

Обратите также внимание, как мы получаем некоторые «бонусные» проверки. Когда мы настраиваем интерфейсы, мы можем видеть, что OSPF-соседства формируются между устройствами. Это избавляет нас от необходимости проверять их «вручную» с помощью следующей команды:

```
ATL2# show ip ospf neighbor
```

**ATL2#show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
1.1.1.1	1	FULL/DR	00:00:31	10.12.12.1	FastEthernet0/0
10.23.23.3	1	FULL/BDR	00:00:38	10.23.23.3	FastEthernet1/0

**ATL2#**

Интересной проверкой для нас здесь является проверка префикса 1.1.1.0/24 с устройства ATL (а также удаленной связи между ATL и ATL2). Мы проверяем это на ORL, чтобы проверить многозональную конфигурацию OSPF. Поскольку это «нормальная» область, все LSA должны быть разрешены в этой области, и мы должны видеть, что префикс появляется как межзонный маршрут OSPF.

```
show ip route ospf
```

ORL#**show ip route ospf**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets

O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:06:32,  
FastEthernet1/0

10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks

O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:06:32,  
FastEthernet1/0

ORL#

Хотя это не часто требуется при устранении неполадок, но мы можем изучить базу данных OSPF, чтобы увидеть различные типы LSA.

**show ip ospf database**

```
ORL#show ip ospf database
```

OSPF Router with ID (10.23.23.3) (Process ID 1)

Router Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Link count				
10.23.23.2	10.23.23.2	561	0x80000002	0x003117
1				
10.23.23.3	10.23.23.3	560	0x80000001	0x002E19
1				

Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.23.23.2	10.23.23.2	561	0x80000001	0x00F15E

Summary Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
1.1.1.1	10.23.23.2	609	0x80000001	0x00AE4E
10.12.12.0	10.23.23.2	609	0x80000001	0x003BA4

ORL#

Записи состояния соединения маршрутизатора являются **Type 1 LSA**. Это конечные точки в нашей локальной области 1.

Записи состояния net link-это **Type 2 LSA**. Здесь мы видим идентификатор маршрутизатора назначенного маршрутизатора (DR). Наконец, суммарные состояния сетевых ссылок — это **Type 3 LSA**. Это префиксы, которые ABR посылает в нашу область. Конечно же, это loopback (1.1.1.0) и удаленная сеть (10.12.12.0).

Примечание: интерфейс обратной связи (**loopback interface**) объявлен как хост-маршрут 32-разрядной версии. Чтобы изменить это, вы можете просто использовать команду `ip ospf network point-to-point` на интерфейсе loopback. Это изменяет тип сети от типа loopback для OSPF и вызывает объявление маски в том виде, в каком она настроена.

Теперь пришло время добавить к этой истории еще и другое. Давайте настроим некоторые внешние префиксы и введем их в домен OSPF. Это просто благодаря loopback interfaces. Мы создадим некоторые из них на маршрутизаторе ATL, запустим EIGRP на них, а затем перераспределим их в OSPF.

```
ATL#conf t
Enter configuration commands, one per line. End with CNTL/Z .
ATL (config)#interface lo10
ATL (config-if)#ip address 192.168.10.1 255.255.255.0
ATL (config-if)#interface loopback 20
ATL (config-if)#ip address 192.168.20.1 255.255.255.0
ATL (config if)#router eigrp 100
ATL (config-router)#network 192 .168.10.1 0.0.0.0
ATL (config-router)#network 192.168.20.1 0.0.0.0
ATL (config-router)#router ospf 1
ATL (config-router)#redistribute eigrp 100 subnets metric 1000
ATL (config-router)#end
ATL#
```

Теперь у нас есть еще более интересные проверки на устройстве ORL. Во-первых, таблица маршрутизации:

```
show ip route ospf
```

ORL#**show ip route ospf**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets

O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:21:59,  
FastEthernet1/0

10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks

O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:21:59,  
FastEthernet1/0

O E2 192.168.10.0/24 [110/1000] via 10.23.23.2, 00:01:30,  
FastEthernet1/0

O E2 192.168.20.0/24 [110/1000] via 10.23.23.2, 00:01:30,  
FastEthernet1/0

ORL#

Обратите внимание, что удаленные префиксы перечислены как маршруты E2. Это значение по умолчанию для внешних маршрутов OSPF типа 2. Это означает, что метрика остается неизменной, поскольку префикс течет от ASBR (автономного системного пограничного маршрутизатора) к внутреннему спикеру OSPF. Вы можете изменить тип на Type 1, если хотите, когда вы выполняете перераспределение. Возможно, больший интерес представляет база данных OSPF:

**show ip ospf database**

ORL#**show ip ospf database**

OSPF Router with ID (10.23.23.3) (Process ID 1)

Router Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Link count				
10.23.23.2	10.23.23.2	1457	0x80000002	0x003117
1				
10.23.23.3	10.23.23.3	1456	0x80000001	0x002E19
1				

Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.23.23.2	10.23.23.2	1457	0x80000001	0x00F15E

Summary Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
1.1.1.1	10.23.23.2	1506	0x80000001	0x00AE4E
10.12.12.0	10.23.23.2	1506	0x80000001	0x003BA4

Summary ASB Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
1.1.1.1	10.23.23.2	229	0x80000001	0x009666

Type-5 AS External Link States

Link ID	ADV Router	Age	Seq#	Checksum
Tag				
192.168.10.0	1.1.1.1	236	0x80000001	0x00153C
0				
192.168.20.0	1.1.1.1	236	0x80000001	0x00A6A0
0				
ORL#				

Обратите внимание, как мы подбираем Type 4 LSA (summary ASB link state), который является идентификатором маршрутизатора (1.1.1.1) ASBR (ATL). Мы также получаем Type 5 LSA, которые являются внешними префиксами.

На этом мы завершим ПЕРВУЮ часть нашей продвинутой серии блогов OSPF. В следующий раз мы рассмотрим создание stubby areas, totally stubby areas, not so stubby areas (NSSA), и totally NSSA.

## 6. OSPF: создание конкретных типов областей

Всем привет! Мы продолжаем рассказывать про протокол маршрутизации OSPF. В этой статье мы рассмотрим создание конкретных типов областей. Подробнее про области можно прочитать в предыдущей статье.

### КОРОТКАЯ ОБЛАСТЬ (STUBBY AREA)

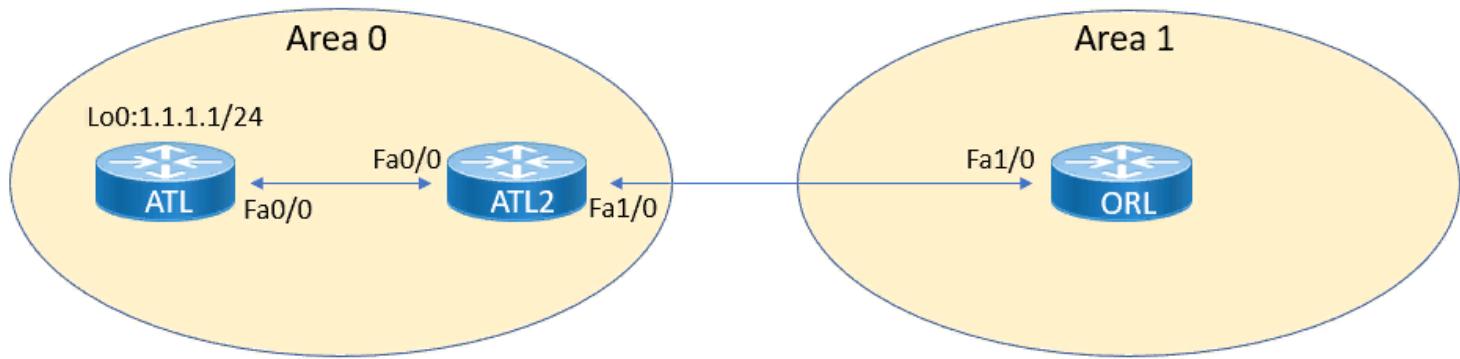


Рисунок 1 - топология OSPF

Пришло время сделать нашу область 1 из рисунка 1 короткой областью. Это внесение небольших настроек в конфигурацию. На каждом устройстве в этой области нам нужно установить область 1 в качестве заглушки. Вот наша конфигурация:

```
ATL2# configuration terminal
Enter configuration commands, one per line. End with CNTL/Z.
ATL2 (config)#router ospf 1
ATL2 (config-router)#area 1 stub
ATL2 (config-router)#end
ATL2#
ORL# configuration terminal
Enter configuration commands, one per line. End with CNTL/Z .
ORL(config)#router ospf 1
ORL(config-router)#area 1 stub
ORL(config-router)#end
ORL#
```

Это вызовет сброс соседства. После внесения изменений настало время просмотреть таблицу маршрутизации и базу данных OSPF.

```
show ip route ospf
```

```
ORL#show ip route ospf
```

Codes: L - local, C - connected, S - static, R - RIP, M -  
mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter  
area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external  
type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 -  
IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-  
user static route

o - ODR, P - periodic downloaded static route, + -  
replicated route

Gateway of last resort is 10.23.23.2 to network 0.0.0.0

O\*IA 0.0.0.0/0 [110/2] via 10.23.23.2, 00:01:46,

FastEthernet1/0

    1.0.0.0/32 is subnetted, 1 subnets

O IA       1.1.1.1 [110/3] via 10.23.23.2, 00:01:56,

FastEthernet1/0

    10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks

O IA       10.12.12.0/24 [110/2] via 10.23.23.2, 00:01:56,

FastEthernet1/0

ORL#

Как мы и надеялись, теперь таблица маршрутизации стала меньше! Больше нет детализации внешних префиксов из ASBR. Вместо этого у нас есть маршрут по умолчанию, автоматически генерируемый ABR. Этот маршрут по умолчанию конечно необходим, потому что маршрутизаторы в области 1 все еще должны иметь возможность доступа к удаленным префиксам (если это необходимо).

Теперь пришло время изучить базу данных OSPF. Это именно то, что мы ожидали бы увидеть в области заглушки:

```
show ip ospf database
```

ORL#**show ip ospf database**

OSPF Router with ID (10.23.23.3) (Process ID 1)

Router Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Link count				
10.23.23.2	10.23.23.2	123	0x80000005	0x0053F2
1				
10.23.23.3	10.23.23.3	117	0x80000004	0x0050F4
1				

Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.23.23.3	10.23.23.3	122	0x80000001	0x00FB54

Summary Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
0.0.0.0	10.23.23.2	134	0x80000001	0x00F013
1.1.1.1	10.23.23.2	134	0x80000003	0x00C834
10.12.12.0	10.23.23.2	134	0x80000003	0x00558A

ORL#

Type 4 LSA и Type 5 LSA фильтруются, и теперь существует Type 3 LSA для маршрута по умолчанию.

## ПОЛНОСТЬЮ КОРОТКАЯ ОБЛАСТЬ (TOTALLY STUBBY AREA)

Если мы хотим быть еще более эффективными в нашем примере, мы можем преобразовать область 1 в полностью короткую область. Это устранит Type 3 LSA, которые используются для [объявления](#) Loopback 0 на ATL и связей между ATL и AT2. Конечно, все равно будет объявлен маршрут по умолчанию, потому что теперь он нужен больше, чем когда-либо! Вот такая конфигурация и верификация:

```
ATL2# configuration terminal
Enter configuration commands , one per line . End with CNTL/Z.
ATL2 (config)#router ospf 1
ATL2 (config-router )#area 1 stub no-summary
ATL2 (config-router )#end
ATL2#
show ip route ospf
```

```
ORL#show ip route ospf
```

Codes: L - local, C - connected, S - static, R - RIP, M -  
mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter  
area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external  
type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 -  
IS-IS level-2  
ia - IS-IS inter area, \* - candidate default, U - per-  
user static route  
o - ODR, P - periodic downloaded static route, + -  
replicated route

Gateway of last resort is 10.23.23.2 to network 0.0.0.0

O\*IA 0.0.0.0/0 [110/2] via 10.23.23.2, 00:04:57,  
FastEthernet1/0

```
show ip ospf database
```

```
ORL#show ip ospf database
```

OSPF Router with ID (10.23.23.3) (Process ID 1)

Router Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Link count				
10.23.23.2	10.23.23.2	941	0x80000005	0x0053F2
1				
10.23.23.3	10.23.23.3	936	0x80000004	0x0050F4
1				

Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.23.23.3	10.23.23.3	941	0x80000001	0x00FB54

Summary Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
0.0.0.0	10.23.23.2	301	0x80000003	0x00EC15

```
ORL#
```

## NOT SO STUBBY AREA (NSSA)

Если вам необходимо ввести внешние префиксы в область заглушки, вы должны сделать ее Not So Stubby Area (NSSA). Это позволяет внешние префиксы, которые будут пересыпаться через зону-заглушки, определять как LSA типа 7. Затем ABR преобразует их в LSA типа 5 для распространения через домен OSPF (потенциально).

Ниже настройка для нашей схемы:

```
ATL2# configuration terminal
Enter configuration commands , one per line. End with CNTL/Z.
ATL2 (config)#router ospf 1
ATL2 (config-router)#no area 1 stub
ATL2 (config-router)#area 1 nssa
ATL2 (config-router)#end
ATL2#

ORL# configuration terminal
Enter configuration commands , one per line . End with CNTL/Z.
ORL(config)#router ospf 1
ORL(config-router)#no area 1 stub
ORL(config-router)#area 1 nssa
ORL(config)#interface loopback10
ORL(config-if)#ip address 172.16.10.3 255.255.255.0
ORL(config)#interface loopback20
ORL(config-if)#ip address 172.16.20.3 255.255.255.0
ORL(config-if)#exit
ORL(config)#router eigrp 200
ORL(config-router)#network 172.16.10.3 0.0.0.0
ORL(config-router)#network 172.16.20.3 0.0.0.0
ORL(config-router)#exit
ORL(config)#router ospf 1
ORL(config-router)#redistribute eigrp 200 subnets metric 1000
ORL(config-router)#end
ORL#
```

Интересно просмотреть результаты нашей настройки. Давайте начнем с изучения маршрутов OSPF и базы данных OSPF на ORL:

```
show ip route ospf
```

```
ORL#show ip route ospf
```

Codes: L - local, C - connected, S - static, R - RIP, M -  
mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter  
area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external  
type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 -  
IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-  
user static route

o - ODR, P - periodic downloaded static route, + -  
replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnet

O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:06:29,  
FastEthernet1/0

10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks

O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:06:29,  
FastEthernet1/0

```
show ip ospf database
```

ORL#**show ip ospf database**

OSPF Router with ID (10.23.23.3) (Process ID 1)

Router Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Link count				
10.23.23.2	10.23.23.2	14	0x80000009	0x00D85F
1				
10.23.23.3	10.23.23.3	205	0x80000007	0x00D760
1				

Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.23.23.3	10.23.23.3	419	0x80000003	0x007FC6

Summary Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
1.1.1.1	10.23.23.2	440	0x80000002	0x0052A3
10.12.12.0	10.23.23.2	440	0x80000002	0x00DEF9

Type-7 AS External Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Tag				
172.16.10.0	10.23.23.3	205	0x80000001	0x00DDA3
172.16.20.0	10.23.23.3	206	0x80000001	0x006F08

ORL#

Из таблицы маршрутизации видно, что мы снова изучаем [объявления](#) Type 3 из области 0 (1.1.1.1 и 10.12.12.0). База данных OSPF доказывает, что NSSA работает так, как объявлено на данный момент. Мы можем видеть внешние префиксы, внесенные в область 1 как Type 7 в базе данных. Давайте быстро проверим ATL, чтобы увидеть, появляются ли они там как Type 5 LSA, как мы ожидаем.

`show ip route ospf`

```
ATL#show ip route ospf
```

Codes: L - local, C - connected, S - static, R - RIP, M -  
mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter  
area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external  
type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 -  
IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-  
user static route

o - ODR, P - periodic downloaded static route, + -  
replicated route

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks

O IA 10.23.23.0/24 [110/2] via 10.12.12.2, 01:12:08,  
FastEthernet0/0

172.16.0.0/24 is subnetted, 2 subnets

O E2 172.16.10.0 [110/1000] via 10.12.12.2, 00:07:58,  
FastEthernet0/0

O E2 172.16.20.0 [110/1000] via 10.12.12.2, 00:07:58,  
FastEthernet0/0

```
show ip ospf database
```

```
ATL#show ip ospf database
```

OSPF Router with ID (1.1.1.1) (Process ID 1)

Router Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
Link count				
1.1.1.1	1.1.1.1	1183	0x80000005	0x005376
10.23.23.2	10.23.23.2	304	0x80000006	0x00551A

Net Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
10.12.12.1	1.1.1.1	415	0x80000003	0x00C909

Summary Net Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
10.23.23.0	10.23.23.2	304	0x80000003	0x00398E

Type-5 AS External Link States

Link ID	ADV Router	Age	Seq#	Checksum
Tag				
172.16.10.0	10.23.23.2	491	0x80000001	0x007814
172.16.20.0	10.23.23.2	491	0x80000001	0x00A78
192.168.10.0	1.1.1.1	1184	0x80000002	0x00133D
192.168.20.0	1.1.1.1	1184	0x80000002	0x00A4A1

ATL#

Да, выводимая информация дает нам полное понимание, как работает NSSA. Префиксы существуют как Type 5s, и мы видим это в таблице маршрутизации.

Примечание: область NSSA не имеет динамически генерируемого маршрута по умолчанию без настройки этой функции. Так, наша сеть не будет работать! Чтобы создать маршрут по умолчанию, просто используйте следующую команду на маршрутизаторе ATL2:

```
area 1 nssa default-information-originate
```

## TOTALLY NSSA

Поскольку вы уже освоили Totally Stubby, скорее всего вы уже понимаете, что происходит с Totally NSSA. Здесь мы блокируем дополнительные типы LSA из этой области. К ним относятся Type 3 LSA. И снова для нас автоматически создается маршрут по умолчанию.

Вот настройки и результирующие проверки:

```
ATL2# configuration terminal
Enter configuration commands , one per line . End with CNTL/Z.
ATL2 (config)#router ospf 1
ATL2 (config-router )#area 1 nssa no-summary
ATL2 (config-router )#end
ATL2#
```

```
show ip route ospf
```

ORL#**show ip route ospf**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is 10.23.23.2 to network 0.0.0.0

O\*IA 0.0.0.0/0 [110/2] via 10.23.23.2, 00:03:09,

FastEthernet1/0

ORL#

**show ip ospf database**

ORL#**show ip ospf database**

OSPF Router with ID (10.23.23.3) (Process ID 1)

Router Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Link count				
10.23.23.2	10.23.23.2	1761	0x80000009	0x00D85F
10.23.23.3	10.23.23.3	1716	0x80000008	0x00D561

Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
10.23.23.3	10.23.23.3	1716	0x80000004	0x007DC7

Summary Net Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
0.0.0.0	10.23.23.2	135	0x80000001	0x007883

Type-7 AS External Link States (Area 1)

Link ID	ADV Router	Age	Seq#	Checksum
Tag				
172.16.10.0	10.23.23.3	1952	0x80000001	0x00DDA3
172.16.20.0	10.23.23.3	1952	0x80000001	0x006F08

ORL#

Как видно из выводимой информации, что у нас невероятно скатая таблица маршрутизации (для OSPF) на роутере ORL.

Как вы можете видеть, OSPF отлично справляется с автоматической фильтрацией маршрутов за счет использования специальных областей и типов LSA. В следующий раз мы рассмотрим варианты ручной фильтрации маршрутов в OSPF.

## 7. Ручная фильтрация маршрутов OSPF

В нашей прошлой статье мы рассмотрели, как **OSPF** может автоматически фильтровать маршруты с помощью специальных областей и типов **LSA**. Но как насчет вариантов ручной фильтрации маршрутов в OSPF? В этой статье мы рассмотрим методы, которые можно использовать в различных точках топологии.

### ФИЛЬТРАЦИЯ НА ASBR

Одним из простых и эффективных методов фильтрации на **ASBR** является использование распределенного списка. Здесь мы определяем правила идентификации маршрута со списком доступа, а затем ссылаемся на этот список доступа в списке распространения.

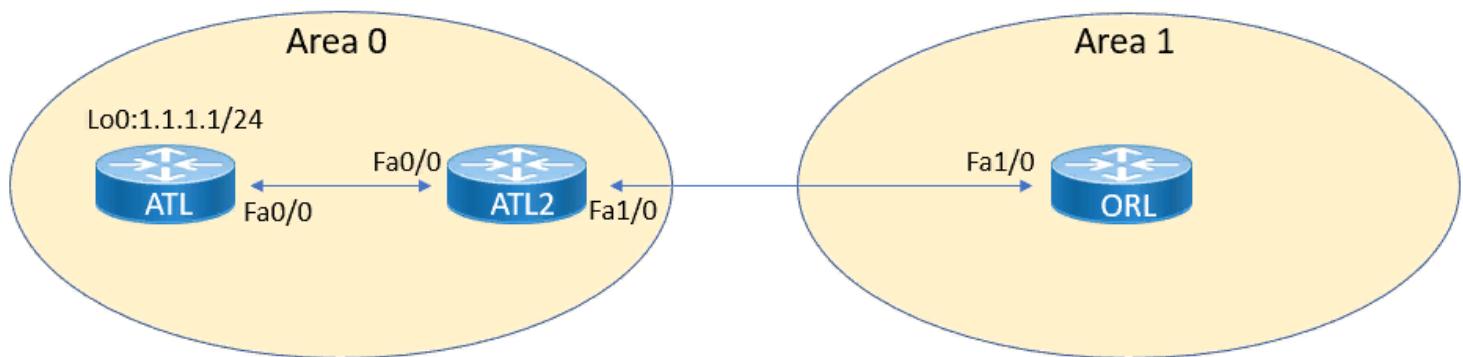


Рисунок 1. Топология OSPF

В этом примере наша область 1 настроена как нормальная, не являющаяся магистральной областью. Вы можете увидеть это, при просмотре таблицы маршрутизации на ORL.

```
show ip route
```

ORL#**show ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets

O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:05:12,

FastEthernet1/0

10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks

O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:05:12,

FastEthernet1/0

C 10.23.23.0/24 is directly connected, FastEthernet1/0

L 10.23.23.3/32 is directly connected, FastEthernet1/0

172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks

C 172.16.10.0/24 is directly connected, Loopback10

L 172.16.10.3/32 is directly connected, Loopback10

C 172.16.20.0/24 is directly connected, Loopback20

L 172.16.20.3/32 is directly connected, Loopback20

O E2 192.168.10.0/24 [110/1000] via 10.23.23.2, 00:00:33,

FastEthernet1/0

O E2 192.168.20.0/24 [110/1000] via 10.23.23.2, 00:05:12,

FastEthernet1/0

ORL#

Обратите внимание на два префикса (E2) 192.168.10.0 и 192.168.20.0. Давайте отфильтруем 192.168.10.0 на ASBR ATL.

```
ATL# configuration terminal
Enter configuration commands , one per line . End with CNTL/Z .
ATL(config)#access-list 1 deny 192.168.10 .0 0.0.0.255
ATL(config)#access-list 1 permit any
ATL(config)#router ospf 1
ATL(config-router)#distribute-list 1 out eigrp 100
ATL(config-router)#end
ATL#
```

Обратите внимание, насколько проста эта конфигурация. Давайте посмотрим, сработало ли это, еще раз изучив таблицу маршрутов ORL:

```
show ip route
```

ORL#**show ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2  
ia - IS-IS inter area, \* - candidate default, U - per-user static route  
o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets  
O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:12:57,  
FastEthernet1/0  
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks  
O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:12:57,  
FastEthernet1/0  
C 10.23.23.0/24 is directly connected, FastEthernet1/0  
L 10.23.23.3/32 is directly connected, FastEthernet1/0  
172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks  
C 172.16.10.0/24 is directly connected, Loopback10  
L 172.16.10.3/32 is directly connected, Loopback10  
C 172.16.20.0/24 is directly connected, Loopback20  
L 172.16.20.3/32 is directly connected, Loopback20  
O E2 192.168.20.0/24 [110/1000] via 10.23.23.2, 00:12:57,  
FastEthernet1/0

ORL#

Конфигурация работает отлично, и 192.168.10.0 больше не доступен на ORL.

Другой простой метод - использовать команду **summary-address** на ASBR и использовать ключевое слово **not-advertise**.

Вот пример из нашей топологии. Обратите внимание, что был удален предыдущий список рассылки из конфигурации ATL до этой настройки здесь:

```
ATL#conf t
Enter configuration commands, one per line. End with CNTL/Z .
ATL(config)#router ospf 1
ATL(config-router)#summary-address 192 .168.10.0 not-advertise
ATL(config-router)#end
ATL#
```

Проверка на ORL доказывает успешную фильтрацию сети 192.168.10.0.

```
show ip route
```

ORL#**show ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2  
ia - IS-IS inter area, \* - candidate default, U - per-user static route  
o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets  
O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:18:44,  
FastEthernet1/0  
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks  
O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:18:44,  
FastEthernet1/0  
C 10.23.23.0/24 is directly connected, FastEthernet1/0  
L 10.23.23.3/32 is directly connected, FastEthernet1/0  
172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks  
C 172.16.10.0/24 is directly connected, Loopback10  
L 172.16.10.3/32 is directly connected, Loopback10  
C 172.16.20.0/24 is directly connected, Loopback20  
L 172.16.20.3/32 is directly connected, Loopback20  
O E2 192.168.20.0/24 [110/1000] via 10.23.23.2, 00:18:44,  
FastEthernet1/0

ORL#

Нет ничего удивительного в том, что вы можете использовать подход **route map** для фильтрации в ASBR. Ведь route map невероятно полезны и гибки.

Здесь мы определим правила со списком доступа (еще раз) и используем их в логике route map:

```
ATL#conf t
Enter configuration commands, one per line . End with CNTL/Z .
ATL(config)#access-list 1 deny 192.168.10.0 0.0.0.255
ATL(config)#access-list 1 permit any
ATL(config)#route-map MYMAP permit 10
ATL(config-route-map)#match ip address 1
ATL(config-route-map)#router ospf 1
ATL(config-router)#redistribute eigrp 100 metric 1000 route-map MYMAP subnets
ATL(config-router)#end
ATL#
```

Как вы можете догадаться, проверка на ORL показывает отличную работу.

```
show ip route
```

```
ORL#show ip route
```

Codes: L - local, C - connected, S - static, R - RIP, M -  
mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter  
area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external  
type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 -  
IS-IS level-2  
ia - IS-IS inter area, \* - candidate default, U - per-  
user static route  
o - ODR, P - periodic downloaded static route, + -  
replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets  
O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:28:07,  
FastEthernet1/0  
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks  
O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:28:07,  
FastEthernet1/0  
C 10.23.23.0/24 is directly connected, FastEthernet1/0  
L 10.23.23.3/32 is directly connected, FastEthernet1/0  
172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks  
C 172.16.10.0/24 is directly connected, Loopback10  
L 172.16.10.3/32 is directly connected, Loopback10  
C 172.16.20.0/24 is directly connected, Loopback20  
L 172.16.20.3/32 is directly connected, Loopback20  
O E2 192.168.20.0/24 [110/1000] via 10.23.23.2, 00:28:07,  
FastEthernet1/0

ORL#

## ФИЛЬТРАЦИЯ НА ABR

Вы также можете фильтровать на **ABR**. Наиболее распространенным методом является использование списка префиксов, как показано здесь:

```
ATL2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ATL2 (config)#ip prefix-list 1 deny 192.168.10.0/24
ATL2 (config)#ip prefix-list 1 permit 0.0.0.0/0
ATL2 (config)#router ospf 1
ATL2 (config-router )#area 1 filter-list prefix 1 out
ATL2 (config-router )#end
ATL2#
```

```
show ip route
```

ORL#**show ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2  
ia - IS-IS inter area, \* - candidate default, U - per-user static route  
o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets  
O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:00:12,  
FastEthernet1/0  
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks  
O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:00:12,  
FastEthernet1/0  
C 10.23.23.0/24 is directly connected, FastEthernet1/0  
L 10.23.23.3/32 is directly connected, FastEthernet1/0  
172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks  
C 172.16.10.0/24 is directly connected, Loopback10  
L 172.16.10.3/32 is directly connected, Loopback10  
C 172.16.20.0/24 is directly connected, Loopback20  
L 172.16.20.3/32 is directly connected, Loopback20  
O E2 192.168.20.0/24 [110/1000] via 10.23.23.2, 00:00:12,  
FastEthernet1/0

ORL#

Мы фильтруем префикс 192.168.10.0, но мы делаем это на ABR, и мы фильтруем по Type 3. Это контрастирует с фильтрацией типа 5 (для того же префикса!) Мы уже делали это раньше в ASBR.

## ФИЛЬТРАЦИЯ В РОУТЕРЕ

Имейте в виду, что вы можете легко фильтровать на любом спикере OSPF внутри самого маршрутизатора. Например, вы можете настроить подход к распределению списка и фильтровать входящие сообщения с его помощью.

В этом примере мы еще раз остановимся на 192.168.10.0. Мы заблокируем его в ACL и будем использовать этот ACL в списке рассылки.

Обратите внимание, что мы находимся на ORL:

```
ORL#conf t
Enter configuration commands , one per line . End with CNTL/Z .
ORL(config) #access-list 1 deny 192.168.10.0 0.0.0.255
ORL(config) #access-list 1 permit any
ORL(config)#router ospf 1
ORL(config-router)#distribute-list 1 in
ORL(config- router)#end
ORL#
```

И снова наш желаемый результат проверки:

```
show ip route
```

ORL#**show ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

1.0.0.0/32 is subnetted, 1 subnets

O IA 1.1.1.1 [110/3] via 10.23.23.2, 00:00:10,  
FastEthernet1/0

10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks

O IA 10.12.12.0/24 [110/2] via 10.23.23.2, 00:00:10,  
FastEthernet1/0

C 10.23.23.0/24 is directly connected, FastEthernet1/0

L 10.23.23.3/32 is directly connected, FastEthernet1/0

172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks

C 172.16.10.0/24 is directly connected, Loopback10

L 172.16.10.3/32 is directly connected, Loopback10

C 172.16.20.0/24 is directly connected, Loopback20

L 172.16.20.3/32 is directly connected, Loopback20

O E2 192.168.20.0/24 [110/1000] via 10.23.23.2, 00:00:10,  
FastEthernet1/0

ORL#

## 8. Поиск и устранение проблем в OSPF

В этой статье рассматриваются OSPF и все проблемы, которые могут возникнуть с этим протоколом. OSPF отличается от EIGRP протоколом состояния канала, но общим для них является то, что оба протокола маршрутизации устанавливают соседство до обмена информацией о маршрутизации. В случае OSPF мы обмениваемся LSA (объявление о состоянии канала), чтобы создать LSDB (база данных о состоянии канала). Наилучшая информация из LSDB будет скопирована в таблицу маршрутизации.

В этой части мы начнем с устранения неполадок соседей OSPF. Как только у нас есть рабочее соседство OSPF, мы рассмотрим другие проблемы, такие как отсутствующие маршруты.

```
R1#
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.12.2 on FastEthernet0/0 from LOADING
to FULL, Loading Done

R1#show ip ospf neighbor
Neighbor ID      Pri  State            Dead Time     Address          Interface
192.168.12.2      1    FULL/BDR        00:00:39   192.168.12.2  FastEthernet0/0
```

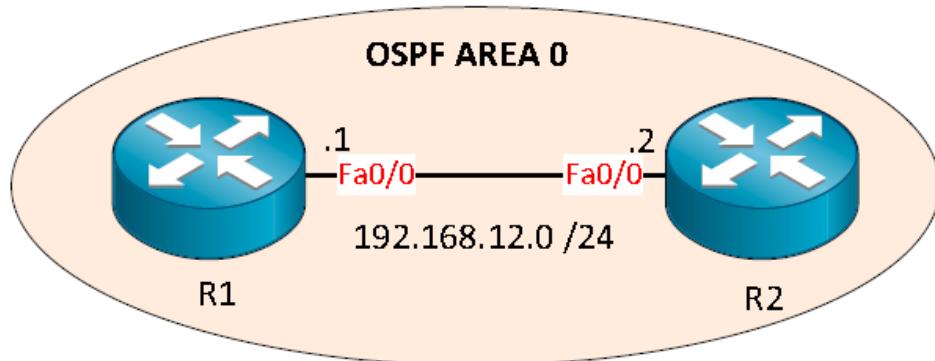
При просмотре соседства **OSPF**, мы видим, что оно сообщает нам **Full**. Необходимо больше информации для понимания состояния Full.

Если смежность соседства OSPF не полная, мы рассматриваем одно из следующих состояний:

- Соседей нет вообще
- Оно "залипло" в ATTEMPT.
- Оно "залипло" в INIT.
- Оно "залипло" в 2-WAY.
- Оно "залипло" в EXSTART/EXCHANGE.
- Оно "залипло" в LOADING.

Давайте начнем и рассмотрим разные ситуации, которые могут возникнуть с соседством OSPF!

### Урок 1



Мы начнем со сценариев, когда OSPF вообще **не имеет соседства**. В приведенном выше примере у нас есть 2 маршрутизатора.

```
R1#show ip ospf neighbor
R1#
```

Как вы можете видеть у нас нет никакого OSPF соседства, что может быть не так?

```
R1#show ip ospf interface fastEthernet 0/0
%OSPF: OSPF not enabled on FastEthernet0/0

R2#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.2/24, Area 0
  Process ID 1, Router ID 192.168.12.2, Network Type BROADCAST, Cost: 1
```

Можно было просто посмотреть на текущую конфигурацию и посмотреть, что не так, но мы не ищем простых путей. Мы используем другие полезные команды OSPF. Сначала используем команду **show ip ospf interface**. Мы видим, что OSPF не включен на интерфейсе FastEthernet 0/0 R1, но он работает на R2.

```
R1#show run | section ospf
router ospf 1
  log-adjacency-changes
  network 192.168.21.0 0.0.0.255 area 0
```

Кто-то допустил ошибку с командой **network** и набрал неверный сетевой адрес ... простая ошибка, но такие вещи случаются.

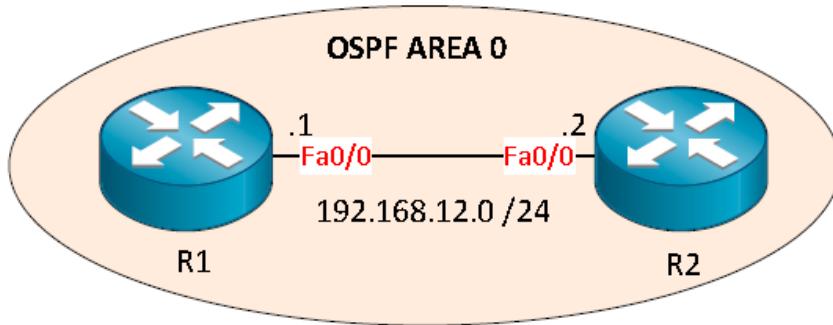
```
R1(config)#router ospf 1
R1(config-router)#no network 192.168.21.0 0.0.0.255 area 0
R1(config-router)#network 192.168.12.0 0.0.0.255 area 0
```

Настройка правильного сетевого адреса и обратной маски устраниет эту ошибку.

```
R1#show ip ospf neighbor
Neighbor ID      Pri  State            Dead Time     Address          Interface
192.168.12.2      1    FULL/DR        00:00:31     192.168.12.2   FastEthernet0/0
```

**Проблема решена.** Соседство OSPF установлено. Это было легкое начало ...**Итог урока:** проверьте правильность настройки сетевого адреса, обратной маски и области.

## Урок 2



Очередная проблема. Схема аналогичная: 2 маршрутизатора, но проблема другая.

```
R1#show ip ospf neighbor
R1#
```

Как вы видите, **нет никакого соседства OSPF**.

```
R1#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
  Process ID 1, Router ID 192.168.12.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State WAITING, Priority 1
  No designated router on this network
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    No Hellos (Passive interface)
R2#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.2/24, Area 0
  Process ID 1, Router ID 192.168.12.2, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.12.2, Interface address 192.168.12.2
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
```

Протокол OSPF был включен на интерфейсе обоих маршрутизаторов, поэтому мы знаем, что был использован правильный тип сети. Однако если вы внимательно посмотрите на R1, то увидите, что на нем написано "**No Hellos (Пассивный интерфейс)**". Если вы настроите пассивный интерфейс, то сеть на интерфейсе все равно будет объявлена, но она не будет отправлять приветственные пакеты OSPF. Таким образом, невозможно создать соседство OSPF.

```
R1#show run | section ospf
router ospf 1
  log adjacency-changes
  passive-interface FastEthernet0/0
  network 192.168.12.0 0.0.0.255 area 0
```

Вот она проблема.

```
R1(config)#router ospf 1
R1(config-router) #no passive-interface Fe0/0
```

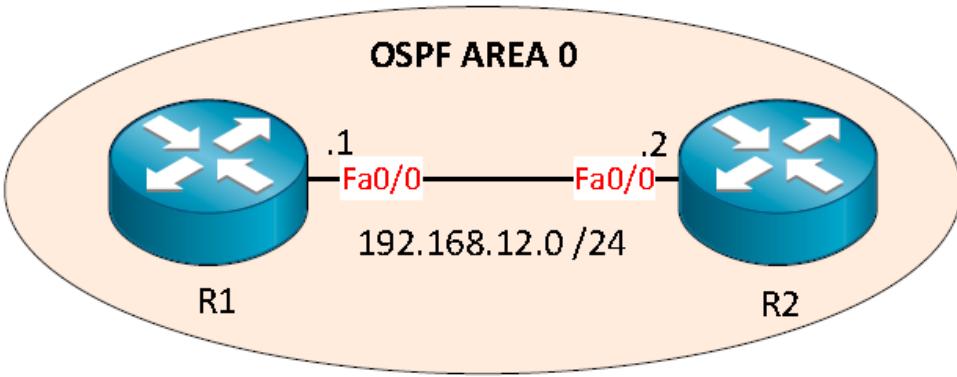
Удалим пассивный интерфейс.

```
R1#show ip ospf neighbor
Neighbor ID      Pri   State        Dead Time     Address          Interface
192.168.12.2      1     FULL/DR    00:00:31     192.168.12.2      FastEthernet0/0
```

Соседство OSPF работает ... проблема устранена!

**Итог урока:** проверьте, что OSPF отправляет приветственные пакеты на интерфейс, поскольку в противном случае вы не сможете создать соседство.

## Урок 3



Следующий сценарий ... те же маршрутизаторы, другая проблема.

```
R1#show ip ospf neighbor
Neighbor ID      Pri   State        Dead Time     Address          Interface
192.168.12.2      1     INIT/DROTHER  00:00:31     192.168.12.2      FastEthernet0/0
R2#show ip ospf neighbor
R2#
```

Интересно... R1 показывает, что наш сосед OSPF находится в состоянии INIT, а R2 ничего не показывает.

```
R1#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
  Process ID 1, Router ID 192.168.12.1, Network Type BROADCAST, Cost: 1

R2#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.2/24, Area 0
  Process ID 1, Router ID 192.168.12.2, Network Type BROADCAST, Cost: 1
```

Как мы видим, в примере выше OSPF был правильно настроен на обоих интерфейсах.

Поскольку R1 показывает состояние INIT, мы можем сделать вывод, что он получает что-то от R2. R2 ничего не показывает, поэтому, вероятно, ничего не получает от R1. OSPF использует пакеты приветствия для установления соседства OSPF, и они отправляются с использованием многоадресного адреса 224.0.0.5.

```
R1#ping 224.0.0.5
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.0.0.5, timeout is 2 seconds:
R2#ping 224.0.0.5
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.0.0.5, timeout is 2 seconds:
```

Рекомендуется проверить, можем ли мы пропинговать адрес многоадресной рассылки, который OSPF использует для пакетов приветствия. Мы видим, что R1 и R2 оба не получают ответа.

```

R1#ping 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/5/8 ms

R2#ping 192.168.12.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/8 ms

```

Отправка эхо-запросов друг другу проходят без проблем. Так что может вызвать проблемы с отправкой и получением многоадресного трафика OSPF? Как насчет списка доступа?

```

R1#show ip interface fastEthernet 0/0 | include access list
  Outgoing access list is not set
  Inbound   access list is not set

R2#show ip interface fastEthernet 0/0 | include access list
  Outgoing access list is not set
  Inbound   access list is BLOCKSTUFF

```

Мы на что-то нашли. И это то, что на R2 имеется входящий список доступа с именем **BLOCKSTUFF**.

```

R2#show access-lists
Extended IP access list BLOCKSTUFF
  10 permit tcp any any
  20 permit udp any any
  30 permit icmp any any (45 matches)

```

Список доступа разрешает только TCP, UDP и ICMP трафик. OSPF не использует TCP или UDP, и он удаляется этим списком доступа из-за **deny any**. Мы этого не видим в верхнем листинге, но в нижней части access-list имеется данный запрет.

```

R2(config)#ip access-list extended BLOCKSTUFF
R2(config-ext-nacl)#5 permit ospf any any

```

```

R2#show access-lists
Extended IP access list BLOCKSTUFF
  5 permit ospf any any (12 matches)
  10 permit tcp any any
  20 permit udp any any
  30 permit icmp any any (45 matches)

```

Проведем коррекцию access-list, чтобы был разрешен трафик OSPF.

```

R1#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time     Address          Interface
192.168.12.2      1    FULL/DR      00:00:37     192.168.12.2    FastEthernet0/0

```

Проблема решена, теперь она отображается как Full.

```

R1#ping 224.0.0.5
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.0.0.5, timeout is 2 seconds:
Reply to request 0 from 192.168.12.2, 16 ms

```

```

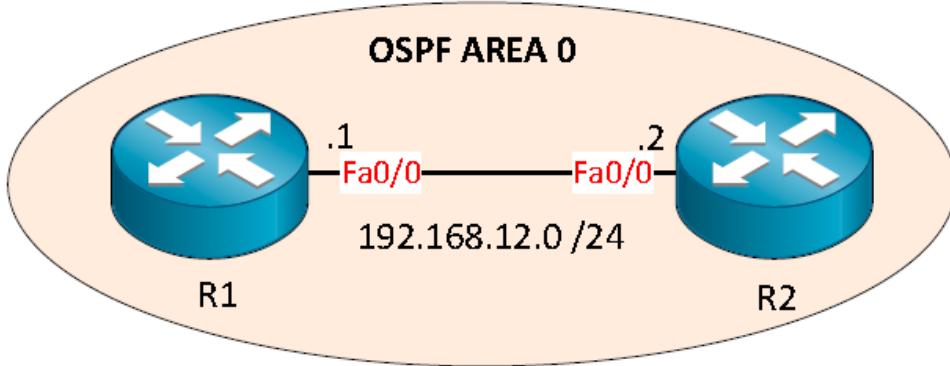
R2#ping 224.0.0.5
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.0.0.5, timeout is 2 seconds:
Reply to request 0 from 192.168.12.1, 4 ms

```

Ну что, теперь можно пинговать адрес многоадресной рассылки 224.0.0.5 OSPF. Мы видим ответ с другой стороны.

**Итог урока:** не блокируйте многоадресные адреса OSPF 224.0.0.5 и 224.0.0.6 (DR / BDR).

## Урок 4



Это еще не все! Тот же сценарий, другая проблема:

```
R1#show ip ospf neighbor      R2#show ip ospf neighbor

R1#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
  Process ID 1, Router ID 192.168.12.1, Network Type BROADCAST, Cost: 1

R2#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.2/25, Area 0
  Process ID 1, Router ID 192.168.12.2, Network Type BROADCAST, Cost: 1
```

Соседство OSPF отсутствует, но мы видим, что OSPF был включен на интерфейсе.

```
R1#ping 224.0.0.5

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.0.0.5, timeout is 2 seconds:
Reply to request 0 from 192.168.12.2, 16 ms

R2#ping 224.0.0.5

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.0.0.5, timeout is 2 seconds:
Reply to request 0 from 192.168.12.1, 4 ms
```

Пинг на адреса многоадресной рассылки проходит, так что это уже хорошо. Это хороший момент для включения отладки, чтобы узнать, что происходит:

```
R1#debug ip ospf hello
OSPF hello events debugging is on
```

Это очень полезная отладка, которая позволяет увидеть, что происходит за кулисами.

```
R1#clear ip ospf process
Reset ALL OSPF processes? [no]: yes
```

Мы сбросим процесс OSPF, чтобы ускорить отладку. Имейте в виду, что вы также можете сбросить только 1 соседство OSPF. Это лучшая идея, если это применяется в производственной сети (сети предприятия или организации).

```
R1#
OSPF: Mismatched hello parameters from 192.168.12.2
OSPF: Dead R 40 C 40, Hello R 10 C 10 Mask R 255.255.255.128 C 255.255.255.0
```

Теперь нам есть с чем работать. R1 говорит, что он получил пакет **hello**, но у нас есть не соответствующие параметры **hello**. R означает то, что мы получили, а C-то, что мы настроили.

Как мы видим, что существует несоответствие в маске подсети. R1 настроен с маской подсети 255.255.255.0, в то время как R2 имеет маску подсети 255.255.255.128. OSPF будет сравнивать маску подсети только в том случае, если вы используете широковещательный тип сети.

```
R1#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
    Process ID 1, Router ID 192.168.12.1, Network Type BROADCAST, Cost: 1
```

```
R2#show ip ospf interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.12.2/25, Area 0
    Process ID 1, Router ID 192.168.12.2, Network Type BROADCAST, Cost: 1
```

Можно использовать команду `show ip ospf interface` для проверки типа сети, и видно, что она **broadcast**.

R1#show run interface fastEthernet 0/0 Building configuration...  Current configuration : 97 bytes ! interface FastEthernet0/0 ip address 192.168.12.1 255.255.255.0 duplex auto speed auto	R2#show run interface fastEthernet 0/0 Building configuration...  Current configuration : 130 bytes ! interface FastEthernet0/0 ip address 192.168.12.2 255.255.255.128 duplex auto speed auto
---	--

Здесь мы видим, что R2 имеет другую маску подсети. Необходимо это исправить!

```
R2(config)#interface Fe0/0
R2(config-if)#ip address 192.168.12.2 255.255.255.0
```

Достаточно просто ...

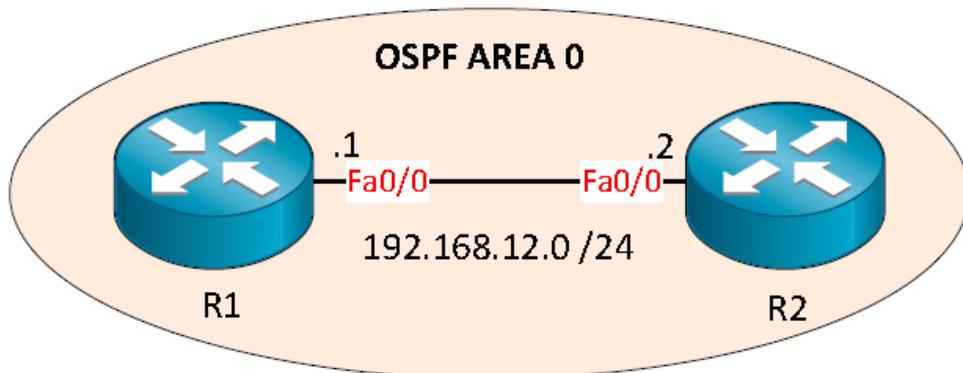
```
R1#
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.12.2 on FastEthernet0/0 from LOADING
to FULL, Loading Done

R2#
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.12.1 on FastEthernet0/0 from LOADING
to FULL, Loading Done
```

Теперь мы видим, соседство OSPF работает.

**Итог урока:** проверьте правильность использования одинаковых масок подсетей на маршрутизаторах, которые напрямую связаны друг с другом.

## Урок 5



Давайте продолжим со следующей ошибкой. Та же топология, и у нас очередная проблема с пакетами `hello`. Сразу перейдем к отладочной части:

```
R1#debug ip ospf hello
OSPF hello events debugging is on

R1#
OSPF: Send hello to 224.0.0.5 area 0 on FastEthernet0/0 from 192.168.12.1
OSPF: Rcv hello from 192.168.12.2 area 0 from FastEthernet0/0 192.168.12.2
OSPF: Mismatched hello parameters from 192.168.12.2
OSPF: Dead R 11 C 24, Hello R 10 C 6 Mask R 255.255.255.0 C 255.255.255.0
```

Эта проблема похожа на наш последний сценарий. Есть часть параметров, которые должны совпадать в hello пакете, чтобы создать соседство OSPF. dead interval на R1 сконфигурирован на 24 секунды, а на R2 на 11 секунд. hello interval настроен на 10 секунд на R2 и 6 секунд на R1. Поменяем настройки параметров:

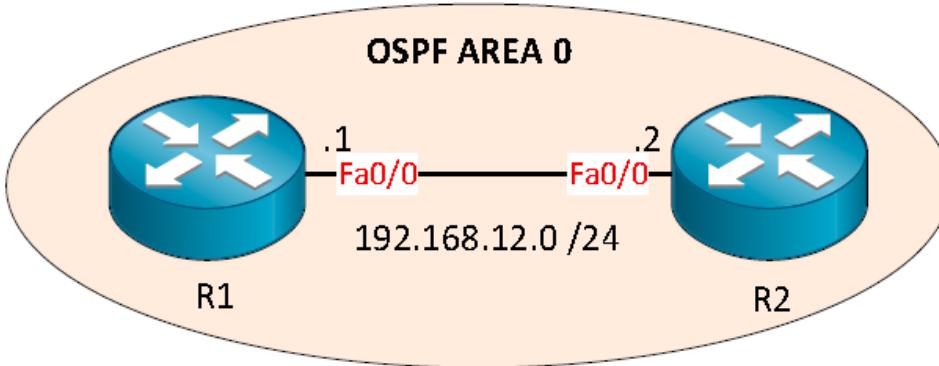
```
R1(config)#interface Fe0/0
R1(config-if)#ip ospf hello-interval 10
R1(config-if)#ip ospf dead-interval 11
```

Нам нужно изменить это на уровне интерфейса.

```
R1#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address          Interface
192.168.12.2      1    FULL/DR     00:00:04   192.168.12.2  FastEthernet0/0
```

Введенные команды с новыми параметрами **решает нашу проблему**. Соседство OSPF работает.

## Урок 6



Еще одна проблема, с которой нам, возможно, придется столкнуться, это аутентификация. OSPF предлагает 3 метода аутентификации:

- без аутентификации
- Plaintext
- MD5 аутентификация

```
R1#show ip ospf neighbor    R2#show ip ospf neighbor
```

Как мы видим, что у нас нет соседей OSPF. Давайте используем debug:

```
R1#debug ip ospf adj
OSPF adjacency events debugging is on
OSPF: Rcv pkt from 192.168.12.2, FastEthernet0/0 : Mismatch Authentication
type. Input packet specified type 2, we use type 0
```

**Debug ip ospf adj** поможет нам решить эти неполадки . Видно, что мы получаем пакет с аутентификацией типа 2, а используется тип 0. Вот что это значит:

- Type 0: нет аутентификации.
- Type 1: plaintext аутентификация.
- Type 2: MD5 аутентификация.

Соответственно - R1 сконфигурирован без аутентификации, а R2 сконфигурирован на использование аутентификации MD5.

```
R2#show ip ospf interface fastEthernet 0/0 | include authentication
Message digest authentication enabled
```

Мы также можем посмотреть информацию OSPF для каждого интерфейса, чтобы увидеть, включена ли аутентификация или нет.

```
R2#show run interface fastEthernet 0/0 | include ospf
  ip ospf authentication message-digest
  ip ospf message-digest-key 1 md5 MYKEY
```

Это то, что настроено на интерфейсе R2.

```
R1(config)#interface FastEthernet0/0
R1(config-if)#ip ospf authentication message-digest
R1(config-if)#ip ospf message-digest-key 1 md5 MYKEY
```

Мы копируем и вставляем его в R1.

```
R1#show ip ospf neighbor
Neighbor ID      Pri   State        Dead Time    Address          Interface
192.168.12.2     1     FULL/DR      00:00:34     192.168.12.2  FastEthernet0/0
```

Проблема устранена! Если вам интересно ... вот что вы увидите, если у вас неправильный пароль на одном из ваших маршрутизаторов:

```
R1(config)#interface FastEthernet0/0
R1(config-if)#no ip ospf message-digest-key 1 md5 MYKEY
R1(config-if)#ip ospf message-digest-key 1 md5 WRONGKEY
```

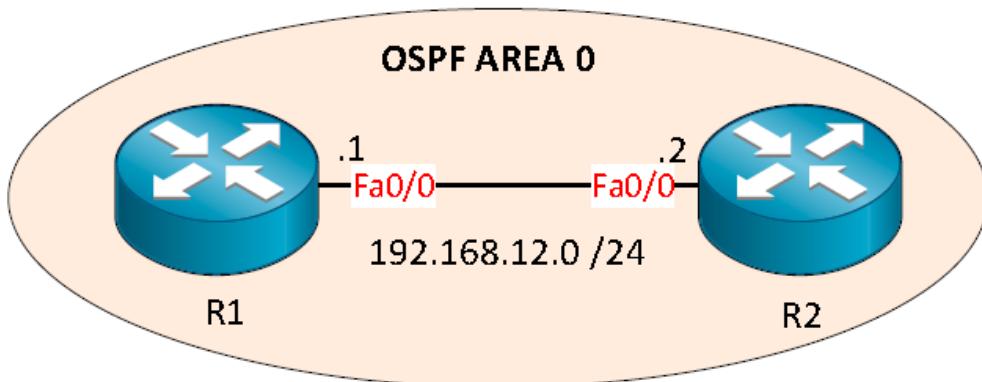
Сначала мы поменяем ключ.

```
R1#debug ip ospf adj
OSPF adjacency events debugging is on
OSPF: Rcv pkt from 192.168.12.2, FastEthernet0/0 : Mismatch Authentication
Key - Message Digest Key 1
```

Наш отладчик говорит нам, что мы используем неправильный ключ между нашими маршрутизаторами.

**Извлеченный урок:** убедитесь, что вы используете один и тот же тип аутентификации OSPF и пароль между маршрутизаторами.

## Урок 7



Что еще может пойти не так? Кажется, что нет никаких проблем, связанных с соседством OSPF! Тот же сценарий ... другая проблема:

```
R1#show ip ospf neighbor  R2#show ip ospf neighbor
```

Соседство отсутствует OSPF.

```
R1# %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from
backbone area must be virtual-link but not found from 192.168.12.2,
FastEthernet0/0
```

Вышло сообщение на одном из наших маршрутизаторов. Сообщение не требует объяснений, похоже, у нас есть несоответствие в номере области.

```
R1#show ip ospf interface fastEthernet 0/0 | include Area
Internet Address 192.168.12.1/24, Area 1
```

```
R2#show ip ospf interface fastEthernet 0/0 | include Area
Internet Address 192.168.12.2/24, Area 0
```

R1 настроен для области 1, а R2 настроен для области 0. Исправляем

```
R1#show run | section ospf
router ospf 1
  log-adjacency-changes
  network 192.168.12.0 0.0.0.255 area 1
```

```
R1(config)#router ospf 1
R1(config-router)#no network 192.168.12.0 0.0.0.255 area 1
R1(config-router)#network 192.168.12.0 0.0.0.255 area 0
```

Мы используем команду **network**, чтобы задать правильный номер области.

```
R1#show ip ospf neighbor
Neighbor ID      Pri   State        Dead Time    Address          Interface
192.168.12.2     1     FULL/DR      00:00:31     192.168.12.2  FastEthernet0/0
```

Ура, все работает!

**Итог урока:** убедитесь, что ваши маршрутизаторы OSPF согласовывают один и тот же номер области.

## Урок 8

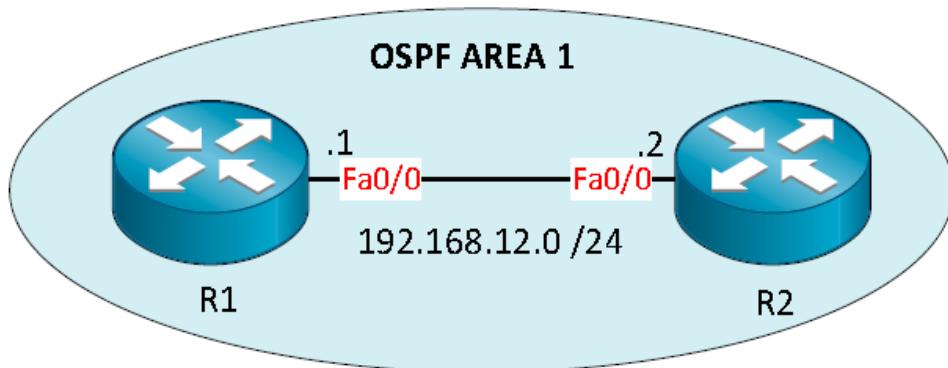


Рисунок выше слегка отличается от предыдущего. На этот раз R1 и R2 находятся в одной зоне 1.

```
R1#show ip ospf neighbor  R2#show ip ospf neighbor
```

Вот так сюрприз...нет соседей! Запускаем отладку:

```
R1#debug ip ospf hello
OSPF hello events debugging is on
OSPF: Rcv hello from 192.168.12.2 area 1 from FastEthernet0/0 192.168.12.2
OSPF: Hello from 192.168.12.2 with mismatched Stub/Transit area option bits
```

Очень, интересно!? Существует несоответствие в опции stub/transit area. OSPF имеет различные типы областей, и оба маршрута должны согласовываться с типом области (**stub, nssa, totally stub и totally nssa**).

```
R1#show ip protocols | include area
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 192.168.12.0 0.0.0.255 area 1
```

R1, по-видимому, настроен на использование normal area.

```
R2#show ip protocols | include area
Number of areas in this router is 1. 0 normal 1 stub 0 nssa
 192.168.12.0 0.0.0.255 area 1
```

R2, похоже, настроен на использование stub area. Несоответствие в типе области означает, что мы не можем установить соседство OSPF.

```
R2#show running-config | section ospf
router ospf 1
  log-adjacency-changes
  area 1 stub
    network 192.168.12.0 0.0.0.255 area 1
    permit ospf any any
```

На листинге выше мы видим, что R2 имеет команду area 1 stub. Удалим ее.

```
R2(config)#router ospf 1
R2(config-router)#no area 1 stub
```

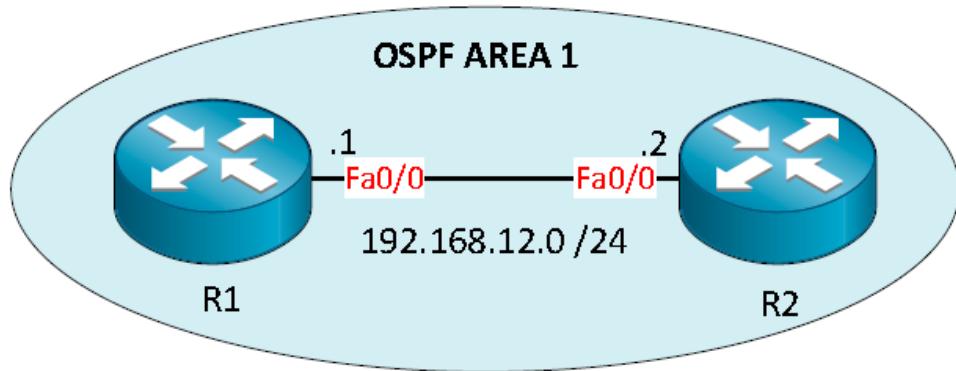
Изменим область 1 на normal area для R2.

```
R1# %OSPF-5-ADJCHG: Process 1, Nbr 192.168.12.2 on FastEthernet0/0 from
LOADING to FULL, Loading Done
```

```
R2# %OSPF-5-ADJCHG: Process 1, Nbr 192.168.12.1 on FastEthernet0/0 from
LOADING to FULL, Loading Done
```

**Итог урока:** убедитесь, что ваши маршрутизаторы OSPF используют один и тот же тип области.

## Урок 9



Очередная ситуация с неполадками с OSPF, которая на первый взгляд кажется очень запутанной. Давайте посмотрим на конфигурацию OSPF обоих маршрутизаторов:

```
R1#show run | section router ospf
router ospf 1
  log adjacency-changes
  network 192.168.12.0 0.0.0.255 area 0
R2#show run | section router ospf
router ospf 1
  log adjacency-changes
  network 192.168.12.0 0.0.0.255 area 0
```

Это простая конфигурация.

```
R1#show ip ospf neighbor
Neighbor ID      Pri      State            Dead Time      Address          Interface
192.168.12.2    0        2WAY/DROTHER      00:00:36       192.168.12.2    FastEthernet0/0

R2#show ip ospf neighbor
Neighbor ID      Pri      State            Dead Time      Address          Interface
192.168.12.1    0        2WAY/DROTHER      00:00:30       192.168.12.1    FastEthernet0/0
```

У нас таблица соседства OSPF не пустая, но оба маршрутизатора "застряли" в состоянии 2WAY. Помимо поиска нужного нам слова "**FULL**", обратить внимание на две вещи, отображаемые командой show:

- Оба маршрутизатора показывают друг друга как **DROTHER**.
- Приоритет для обоих маршрутизаторов равен 0.

В multi-access сети, такой как Ethernet, OSPF будет выполнять выборы **DR/BDR**, если тип сети broadcast или non-broadcast.

Проверяем тип сети:

```
R1#show ip ospf interface | include Network Type
Process ID 1, Router ID 192.168.12.1, Network Type BROADCAST, Cost: 1

R2#show ip ospf interface | include Network Type
Process ID 1, Router ID 192.168.12.2, Network Type BROADCAST, Cost: 1
```

Оба интерфейса настроены для типа сети broadcast. Это значение по умолчанию для интерфейсов Ethernet. Это означает, что у нас есть выборы DR / BDR, но оба маршрутизатора настроены на приоритет 0, а это означает, что они не будут участвовать в выборах DR / BDR. По этой причине они застряли в состоянии 2WAY. Необходимо это исправить:

```
R1(config)#interface fastEthernet 0/0
R1(config-if)#ip ospf priority 1
```

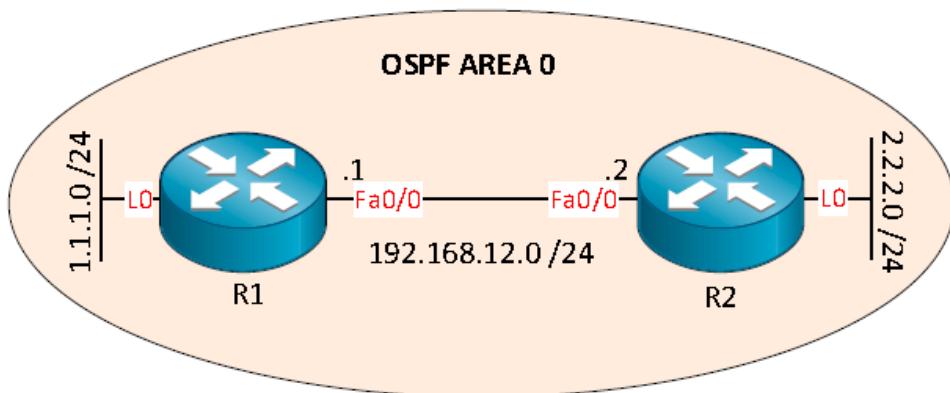
Мы изменим приоритет на одном из маршрутизаторов.

```
R1#show ip ospf neighbor
Neighbor ID      Pri      State            Dead Time      Address          Interface
192.168.12.2    0        FULL/DROTHER      00:00:38       192.168.12.2    FastEthernet0/0

R2#show ip ospf neighbor
Neighbor ID      Pri      State            Dead Time      Address          Interface
192.168.12.1    1        FULL/DR          00:00:37       192.168.12.1    FastEthernet0/0
```

Все работает. Мы видим, что R1 был выбран для DR, потому что он имеет приоритет 1. **Итог урока:** Типы широковещательной и не вещательной сети требуют выбора DR / BDR. Убедитесь, что один из маршрутизаторов выбран.

## 9. Траблшутинг OSPF



**Первая проблема.** Два роутера работают с одной областью OSPF, и каждый роутер имеет loopback интерфейс, объявленный в OSPF. Вот вывод таблиц маршрутизации:

```
R1#show ip route
C      192.168.12.0/24 is directly connected, FastEthernet0/0
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
O      10.2.2.0/24 [110/2] via 192.168.12.2, 00:00:14, FastEthernet0/0
C      10.1.1.0/24 is directly connected, Loopback0

R2#show ip route
C      192.168.12.0/24 is directly connected, FastEthernet0/0
      10.0.0.0/24 is subnetted, 1 subnets
C      10.2.2.0 is directly connected, Loopback0
```

Как мы можем наблюдать, что роутер R1 узнал о сети 10.2.2.0/24 от роутера R2, но в таблице маршрутизации роутера R2 пусто. Что не так?

```
R2#show ip ospf interface loopback 0
Loopback0 is up, line protocol is up
Internet Address 10.2.2.2/24, Area 0
Process ID 1, Router ID 10.2.2.2, Network Type LOOPBACK, Cost: 1
Loopback interface is treated as a stub Host

R1#show ip ospf interface loopback 0
%OSPF: OSPF not enabled on Loopback0
```

Видно, что OSPF не включен на интерфейсе loopback0 роутера R1, так что же мы тогда объявляем в сетях?

```
R1#show ip protocols | begin Networks
Routing for Networks:
  10.10.1.1 0.0.0.0 area 0
  192.168.12.0 0.0.0.255 area 0
```

Похоже, мы объявляем сеть **10.10.1.0/24**, но эта сеть не настроена ни на одном интерфейсе... Сеть 10.1.1.0/24 настроена на интерфейсе loopback0 роутера R1.

```
R1#show run | section router ospf
router ospf 1
  log-adjacency-changes
  network 10.10.1.1 0.0.0.0 area 0
  network 192.168.12.0 0.0.0.255 area 0
```

Здесь вы видите неправильно введенную команду **network**. Удалим ее.

```
R1(config)#router ospf 1
R1(config-router)#no network 10.10.1.1 0.0.0.0 area 0
R1(config-router)#network 10.1.1.0 0.0.0.255 area 0
```

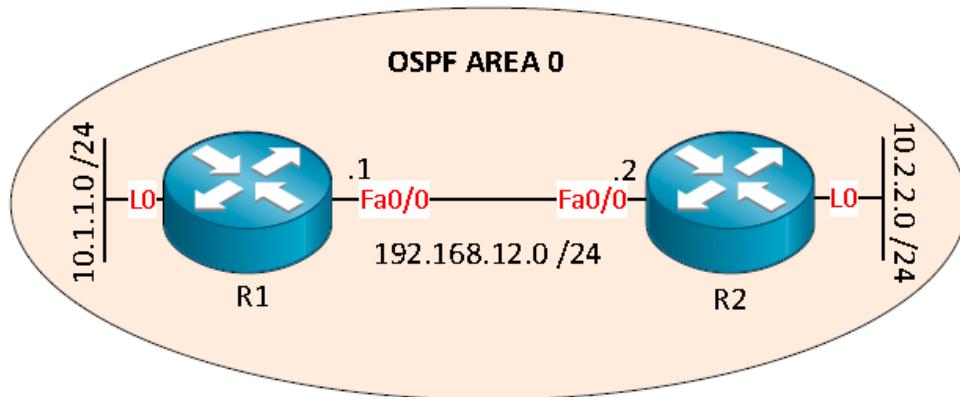
Давайте удостоверимся, что команда network настроена правильно.

```
R2#show ip route | include 10.1.1.
O      10.1.1.0/24 [110/2] via 192.168.12.1, 00:01:29, FastEthernet0/0
```

**Проблема устранена!** Эта проблема может показаться не серьезной, но использование неправильных сетевых операторов - это то, что происходит постоянно. Особенно если мы используем меньшие подсети (например, /27 или /28 или аналогичные), люди склонны делать ошибки с обратными маскам.

**Итог урока:** убедитесь, что вы настроили правильный сетевой адрес, обратную маску и область.

## УРОК №2



Очередная возможная ситуация. Опять два роутера, но другая проблема. Вот таблицы маршрутизации:

```
R1#show ip route
C      192.168.12.0/24 is directly connected, FastEthernet0/0
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
O      10.2.2.2/32 [110/2] via 192.168.12.2, 00:00:14, FastEthernet0/0

R2#show ip route
C      192.168.12.0/24 is directly connected, FastEthernet0/0
      10.0.0.0/24 is subnetted, 1 subnets
C          10.2.2.0 is directly connected, Loopback0
```

В очередной раз роутер R2 не увидел сеть 10.1.1.0/24. Что интересно, что роутер R1 не имеет сети 10.1.1.0/24 в своей таблице маршрутизации как непосредственно подключенной.

```
R1#show ip protocols | begin Networks
Routing for Networks:
  10.1.1.0 0.0.0.255 area 0
  192.168.12.0 0.0.0.255 area 0
```

Мы можем проверить, что роутер R1 использует правильную настройку команды `network`. Поскольку R1 даже не имеет сети в своей таблице маршрутизации, предположим, что проблема с интерфейсом.

```
R1#show ip interface brief
Interface          IP-Address      OK? Method Status           Protocol
FastEthernet0/0    192.168.12.1   YES manual up            up
Loopback0          10.1.1.1       YES manual administratively down
```

Кажется, кто-то забыл применить команду `'no shutdown'` на интерфейсе.

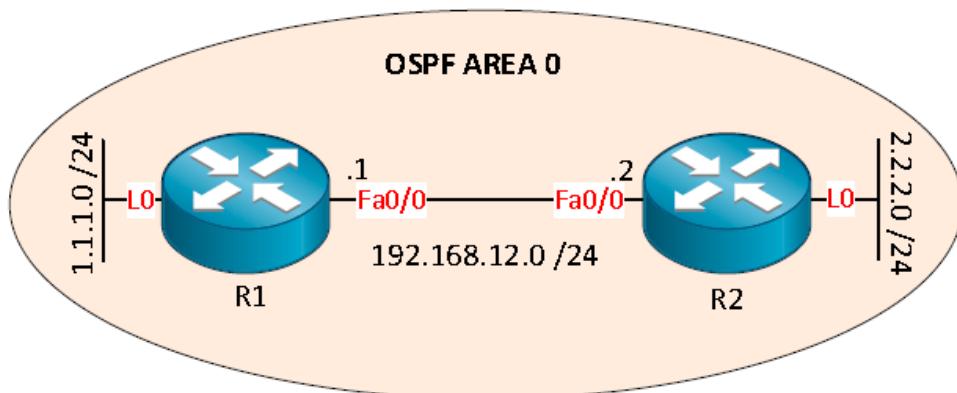
```
R1(config)#interface loopback 0
R1(config-if)#no shutdown
```

Давайте включим интерфейс.

```
R2#show ip route | include 10.1.1
O      10.1.1.1/32 [110/2] via 192.168.12.1, 00:00:32, FastEthernet0/0
```

И теперь он появляется в таблице маршрутизации роутера R2. Итог урока: нельзя объявлять то, чего у тебя нет!

## УРОК №3



Новый урок! Одна область, опять два роутера... мы хотели бы иметь "**full connectivity**", но не работает OSPF ... вот вывод таблиц маршрутизации:

```
R2#show ip route ospf
R1#show ip route ospf      1.0.0.0/32 is subnetted, 1 subnets
                           O      1.1.1.1 [110/2] via 192.168.12.1, 00:08:46, FastEthernet0/0
```

Роутер R1 не показывает никаких маршрутов OSPF, R2 показывает ... Необходимо выяснить, что не так:

```
R2#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 2.2.2.2
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    2.2.2.0 0.0.0.255 area 0
    192.168.12.0 0.0.0.255 area 0
```

Быстро взглянем на роутер R2, чтобы убедиться, что он действительно объявляет правильную сеть(и). Да это так и есть.

```
R1#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is 1
  Router ID 1.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    1.1.1.0 0.0.0.255 area 0
    192.168.12.0 0.0.0.255 area 0
  Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway          Distance      Last Update
    2.2.2.2           110          00:07:23
    192.168.12.2     110          00:24:54
  Distance: (default is 110)
```

Вывод роутера R1 более интересен ... видно, что у него настроен **distribute-list**:

```
R1#show access-lists
Standard IP access list 1
  10 deny 2.2.2.0, wildcard bits 0.0.0.255 (2 matches)
  20 permit any
```

В этом заключается наша проблема. Давайте удалим **distribute-list**.

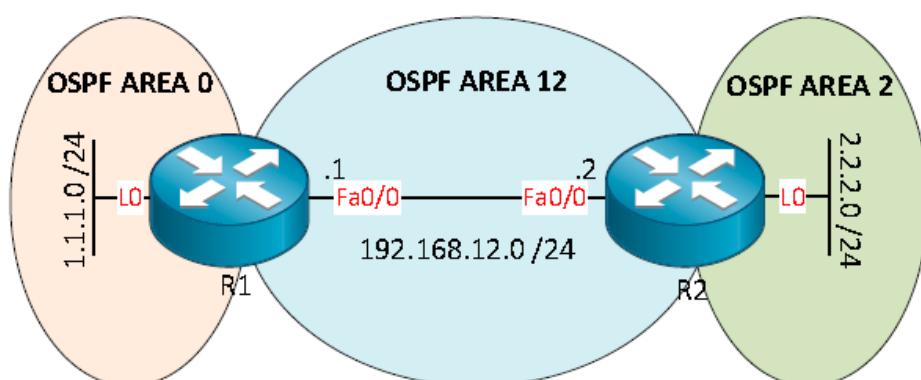
```
R1(config)#router ospf 1
R1(config-router)#no distribute-list 1 in
```

Эта команда отключит его.

```
R1#show ip route ospf
*Mar 1 01:48:39.683: #SYS-5-CONFIG_I: Configured from console by console
2.0.0.0/32 is subnetted, 1 subnets
O      2.2.2.2 [110/2] via 192.168.12.2, 00:00:24, FastEthernet0/0
```

**Задача решена! Итог урока:** знать о **distribute-list**, запрещающий объявление и / или установку префиксов в таблице маршрутизации.

## УРОК №4



Взглянем на более сложные проблемы OSPF. На изображении выше мы имеем роутер R1 и роутер R2, но на этот раз мы имеем конфигурацию OSPF с несколькими областями. Вот конфигурация OSPF этих роутеров:

```
R1#show run | section router ospf          R2#show run | section ospf
router ospf 1                                router ospf 1
  log adjacency-changes
  area 12 virtual-link 192.168.12.2
  network 1.1.1.0 0.0.0.255 area 0
  network 192.168.12.0 0.0.0.255 area 12      log adjacency-changes
                                                area 12 virtual-link 192.168.12.1
                                                network 2.2.2.0 0.0.0.255 area 2
                                                network 192.168.12.0 0.0.0.255 area 12
```

Видно, что все сети были объявлены. Область 2 не связана напрямую с областью 0, поэтому была создана виртуальная связь.

```
R1#show ip route
C   192.168.12.0/24 is directly connected, FastEthernet0/0
  1.0.0.0/24 is subnetted, 1 subnets
    C     1.1.1.0 is directly connected, Loopback0
```

```
R2#show ip route
C   192.168.12.0/24 is directly connected, FastEthernet0/0
  1.0.0.0/32 is subnetted, 1 subnets
O IA  1.1.1.1 [110/2] via 192.168.12.1, 00:07:34, FastEthernet0/0
  2.0.0.0/24 is subnetted, 1 subnets
C   2.2.2.0 is directly connected, Loopback0
```

Роутер R1, однако, не увидел сеть **2.2.2.0/24** от роутера R2, но роутер R2 увидел сеть 1.1.1.0/24. Лучше всего начать с виртуальной линии здесь:

```
R1#show ip ospf virtual-links          R2#show ip ospf virtual-links
Virtual Link OSPF_VL1 to router 192.168.12.2 is down  Virtual Link OSPF_VL0 to router 192.168.12.1 is down
```

Хм, это выглядит не очень хорошо. Виртуальная связь отключена. Обратите внимание на IP-адреса, которые мы видим здесь, это IP-адреса, настроенные на интерфейсах FastEthernet обоих маршрутизаторов.

```
R1#show ip ospf neighbor
Neighbor ID  Pri  State        Dead Time      Address      Interface
2.2.2.2       1    FULL/DR    00:00:31      192.168.12.2  FastEthernet0/0

R2#show ip ospf neighbor
Neighbor ID  Pri  State        Dead Time      Address      Interface
1.1.1.1       1    FULL/BDR   00:00:38      192.168.12.1  FastEthernet0/0
```

Всякий раз, когда мы настраиваем виртуальное соединение, нам нужно настроить идентификатор маршрутизатора OSPF другой стороны, а не IP-адрес другой стороны!

```
R1#show run | incl virtual-link          R2#show run | incl virtual-link
  area 12 virtual-link 192.168.12.2      area 12 virtual-link 192.168.12.1
```

Вот ошибка, так что давайте исправим ее.

```
R1(config)#router ospf 1
R1(config-router)#no area 12 virtual-link 192.168.12.2
R1(config-router)#area 12 virtual-link 2.2.2.2

R2(config)#router ospf 1
R2(config-router)#no area 12 virtual-link 192.168.12.1
R2(config-router)#area 12 virtual-link 1.1.1.1
```

Вот так должна выглядеть **virtual-link**, настроенная между идентификаторами маршрутизаторов OSPF.

```
R1# %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on OSPF_VL2 from LOADING to
FULL, Loading Done

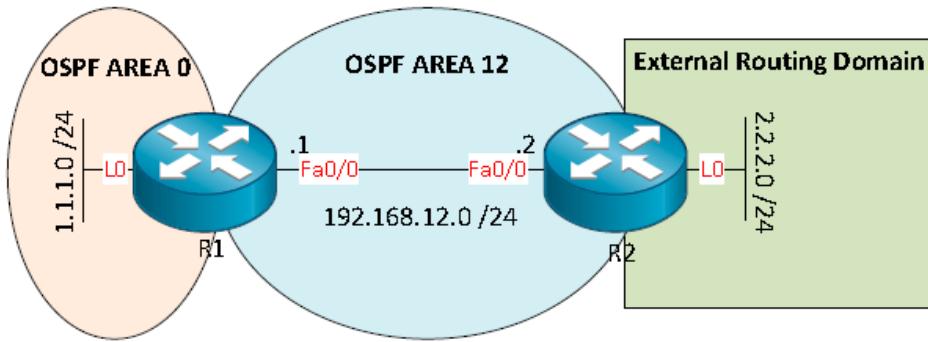
R2# OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on OSPF_VL1 from LOADING to FULL,
Loading Done
```

Сразу после ввода правильных команд появятся данные сообщения в консоли.

```
R1#show ip route ospf
  2.0.0.0/32 is subnetted, 1 subnets
O IA 2.2.2.2 [110/2] via 192.168.12.2, 00:01:38, FastEthernet0/0
```

Запись OSPF для сети 2.2.2.0/24 появилась.

# УРОК №5



Другая проблема. Те же роутеры, но появился "домен внешней маршрутизации". Это может быть другой протокол маршрутизации, такой как RIP или EIGRP, который мы будем распространять в OSPF. R2 перераспределяет сеть 2.2.2.0 / 24 в OSPF, но по какой-то причине она не отображается на R1. Чтобы было интересно, мы не будем просматривать конфигурацию OSPF на роутерах.

```
R1#show ip route
C    192.168.12.0/24 is directly connected, FastEthernet0/0
     1.0.0.0/24 is subnetted, 1 subnets
C      1.1.1.0 is directly connected, Loopback0
```

Нет сети 2.2.2.0/24 на роутере R1, поэтому давайте изучим роутер R2.

```
R2#show ip route | include 2.2.2
C 2.2.2.0 is directly connected, Loopback0
```

Как мы можем видеть, сеть находится в таблице маршрутизации роутера R2 как **directly connected**.

```
R2#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 2.2.2.2
  It is an autonomous system boundary router
  Redistributing External Routes from, connected, includes subnets in
  redistribution
```

Как мы можем видеть роутер R2 был настроен для перераспределения напрямую подключенных сетей. Это должно включать сеть 2.2.2.0/24 на интерфейса loopback0.

```
R2#show ip ospf database external
      OSPF Router with ID (2.2.2.2) (Process ID 1)
```

Однако в базе данных OSPF пусто? Что может быть причиной этого? Возможно, вы помните правила различных типов областей OSPF. Давайте выясним, что это за область!

```
R2#show ip protocols | include areas
  Number of areas in this router is 1. 0 normal 1 stub 0 nssa
```

Вот и объяснение, это stub area! Stub area не допускают LSA type 5 (внешние маршруты). Мы можем либо превратить эту область в normal area или NSSA. Давайте переведем в NSSA.

```
R1(config)#router ospf 1
R1(config-router)#no area 12 stub
R1(config-router)#area 12 nssa

R2(config)#router ospf 1
R2(config-router)#no area 12 stub
R2(config-router)#area 12 nssa
```

Изменим тип области на обоих маршрутизаторах. Область NSSA допускает внешние маршруты с помощью LSA type 7.

```
R2#show ip ospf database nssa-external
```

```
OSPF Router with ID (2.2.2.2) (Process ID 1)
```

#### Type-7 AS External Link States (Area 12)

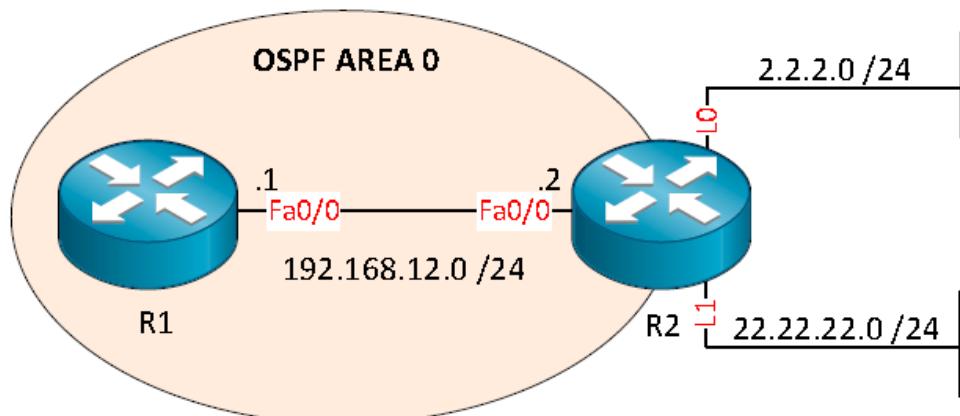
```
LS age: 122  
Options: (No TOS-capability, Type 7/5 translation, DC)  
LS Type: AS External Link  
Link State ID: 2.2.2.0 (External Network Number )  
Advertising Router: 2.2.2.2  
LS Seq Number: 80000001  
Checksum: 0xB759  
Length: 36  
Network Mask: /24
```

Наша сеть 2.2.2.0 / 24 теперь в базе данных OSPF маршрутизатора R2.

```
R1#show ip route | include 2.2.2.2  
o N2 2.2.2.0 [110/20] via 192.168.12.2, 00:03:35, FastEthernet0/0
```

**Итог урока:** Stub area не допускают внешних префиксов (LSA Type 5). Либо измените область на NSSA, либо прекратите перераспределение.

## УРОК №6



Очередная проблема. Проблема **default route OSPF**. На рисунке имеются роутер R1 и роутер R2, и сеть 192.168.12.0 /24 объявленная в OSPF. Loopback интерфейсы роутера R2 не объявляется в OSPF, но мы используем default route, чтобы роутер R1 мог добраться до них. Здесь представлены конфигурации OSPF:

```
R1#show run | section router ospf  
router ospf 1  
log-adjacency-changes  
network 192.168.12.0 0.0.0.255 area 0
```

```
R2#show run | section router ospf  
router ospf 1  
log-adjacency-changes  
network 192.168.12.0 0.0.0.255 area 0  
default-information originate
```

Видно, что в выводе роутера R2 присутствует команда **default-information originate** для **объявления** default route.

```
R1#show ip route  
C 192.168.12.0/24 is directly connected, FastEthernet0/0
```

Увы, но мы не видим **default route** на роутере R1. Будем искать неполадки в настройке. Давайте проверим роутер R2:

```
R2#show ip route  
C 192.168.12.0/24 is directly connected, FastEthernet0/0  
2.0.0.0/24 is subnetted, 1 subnets  
C 2.2.2.0 is directly connected, Loopback0  
22.0.0.0/24 is subnetted, 1 subnets  
C 22.22.22.0 is directly connected, Loopback1
```

В таблице маршрутизации роутера R2 не виден **default route**. Чтобы OSPF объявлял default route, можно использовать два варианта:

- Убедитесь, что у вас есть default route в routing table (невозможно объявлять то, чего нет);
- Примените команду **default-information originate always**. Она объявит default route, даже если он не прописан.

```
R2(config)#ip route 0.0.0.0 0.0.0.0 null 0
```

Выше первый метод решения проблемы. Мы создадим default route на роутере R2. Обычно указывается default route на ISP роутере, но сейчас другого роутера нет. Мы укажем **default route** для интерфейса null0, и он будет внесен в routing table.

```
R1#show ip route ospf  
O *E2 0.0.0.0/0 [110/1] via 192.168.12.2, 00:01:02, FastEthernet0/0
```

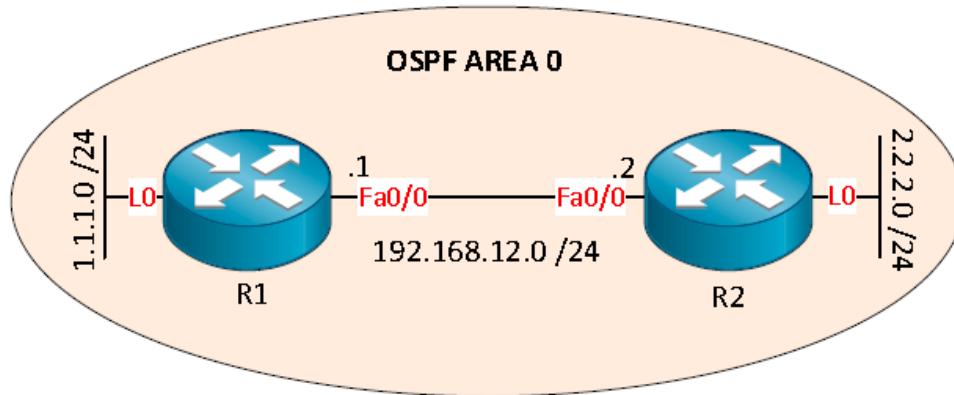
Правило работает!

```
R2(config)#no ip route 0.0.0.0 0.0.0.0 null 0  
R2(config)#router ospf 1  
R2(config-router)#default-information originate always
```

```
R1#show ip route ospf  
O *E2 0.0.0.0/0 [110/1] via 192.168.12.2, 00:00:38, FastEthernet0/0
```

**Итог урока:** что бы объявить default route с помощью OSPF, вам нужно иметь default route в таблице маршрутизации или использовать ключевое слово "always".

## УРОК №7



Немного сложнее проблема... те же два роутера , все в зоне 0. Вот настройки OSPF:

```
R1#show run | section router ospf  
router ospf 1  
    log adjacency-changes  
    network 1.1.1.0 0.0.0.255 area 0  
    network 192.168.12.0 0.0.0.255 area 0  
R2#show run | section router ospf  
router ospf 1  
    log adjacency-changes  
    network 2.2.2.0 0.0.0.255 area 0  
    network 192.168.12.0 0.0.0.255 area 0
```

Ничего особенного, все сети объявлены, и мы используем одну область.

```
R1#show ip route  
C      192.168.12.0/24 is directly connected, FastEthernet0/0  
      1.0.0.0/24 is subnetted, 1 subnets  
C      1.1.1.0 is directly connected, Loopback0  
  
R2#show ip route  
C      192.168.12.0/24 is directly connected, FastEthernet0/0  
      2.0.0.0/24 is subnetted, 1 subnets  
C      2.2.2.0 is directly connected, Loopback0
```

Увы ... таблицы маршрутизации пусты! По крайней мере, никакой отсутствует информация о OSPF ... Настройки network выглядят хорошо, так что это хороший момент вникнуть поглубже в OSPF LSDB. Давайте сначала проверим идентификаторы маршрутизатора OSPF:

```
R1#show ip ospf neighbor  
  
Neighbor ID      Pri      State      Dead Time      Address          Interface  
192.168.12.2      0        FULL/ -      00:00:30      192.168.12.2      FastEthernet0/0  
  
R2#show ip ospf neighbor  
  
Neighbor ID      Pri      State      Dead Time      Address          Interface  
1.1.1.1          1        FULL/BDR     00:00:32      192.168.12.1      FastEthernet0/0
```

Здесь мы видим OSPF router ID. Если вы внимательно посмотрите на информацию выше, вы заметите что-то необычное. State full, но роутер R1 не выбрал DR / BDR, а роутер R2 выбрал роутер R1 в качестве BDR.

```
R1#show ip ospf database router 192.168.12.2
```

```
OSPF Router with ID (1.1.1.1) (Process ID 1)
```

```
Router Link States (Area 0)
```

**Adv Router is not-reachable**

```
LS age: 527
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 192.168.12.2
Advertising Router: 192.168.12.2
LS Seq Number: 80000003
Checksum: 0x1E1C
Length: 48
Number of Links: 2
```

```
Link connected to: a Stub Network
```

```
(Link ID) Network/subnet number: 2.2.2.2
(Link Data) Network Mask: 255.255.255.255
Number of TOS metrics: 0
TOS 0 Metrics: 1
```

```
Link connected to: a Transit Network
```

```
(Link ID) Designated Router address: 192.168.12.2
(Link Data) Router Interface address: 192.168.12.2
Number of TOS metrics: 0
TOS 0 Metrics: 1
```

Мы можем использовать команду `show ip ospf database router` для поиска информации от определенного соседа OSPF.

Роутер R1 говорит нам, **adv router is not-reachable**. Это плохо.

```
R2#show ip ospf database router 1.1.1.1
```

```
OSPF Router with ID (192.168.12.2) (Process ID 1)
```

```
Router Link States (Area 0)
```

**Adv Router is not-reachable**

```
LS age: 776
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
Advertising Router: 1.1.1.1
LS Seq Number: 80000003
Checksum: 0xE8B7
Length: 60
Number of Links: 3
```

```
Link connected to: a Stub Network
```

```
(Link ID) Network/subnet number: 1.1.1.1
(Link Data) Network Mask: 255.255.255.255
Number of TOS metrics: 0
TOS 0 Metrics: 1
```

**Link connected to: another Router (point-to-point)**

```
(Link ID) Neighboring Router ID: 192.168.12.2
(Link Data) Router Interface address: 192.168.12.1
Number of TOS metrics: 0
TOS 0 Metrics: 1
```

```
Link connected to: a Stub Network
```

```
(Link ID) Network/subnet number: 192.168.12.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
```

Роутер R2 также сообщает нам, что роутер R1 недоступен, и если вы посмотрите внимательно, то увидите, что он видит связь как point-to-point. Мы не видим этого в выводе на роутере R1. Это, вероятно, означает, что роутер R1 и роутер R2 используют другой тип сети OSPF, что приводит к разнице в LSDB. Это не позволит нашим роутерам устанавливать маршруты в таблицу маршрутизации!

```
R1#show ip ospf interface fastEthernet 0/0 | include Network Type
```

**Process ID 1, Router ID 1.1.1.1, Network Type POINT\_TO\_POINT, Cost: 1**

```
R2#show ip ospf interface fastEthernet 0/0 | include Network Type
```

**Process ID 1, Router ID 192.168.12.2, Network Type BROADCAST, Cost: 1**

Теперь мы кое-что выяснили. Тип сети отличается ... широковещательная передача на роутере R2 и точка-точка на роутере R1. Нам действительно удалось установить соседство OSPF с этим, но возникает разница в LSDB.

Произведем исправления.

```
R1(config)#interface fa0/0
R1(config-if)#ip ospf network broadcast
```

Изменение типа сети на роутере R1 сделает свое дело.

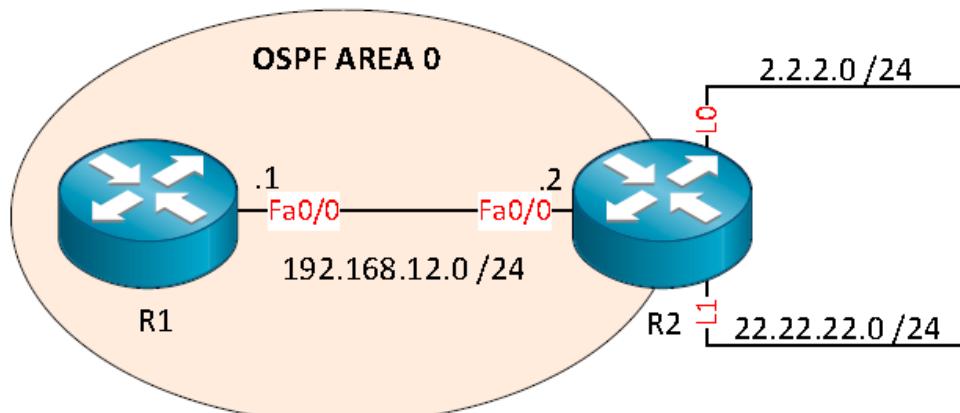
```
R1#show ip route ospf
 2.0.0.0/32 is subnetted, 1 subnets
 O      2.2.2.2 [110/2] via 192.168.12.2, 00:00:18, FastEthernet0/0

R2#show ip route ospf
 1.0.0.0/32 is subnetted, 1 subnets
 O      1.1.1.1 [110/2] via 192.168.12.1, 00:00:49, FastEthernet0/0
```

Наконец "O" появляется в наших таблицах маршрутизации...проблема решена!

**Итог урока:** убедитесь, что вы используете правильный тип сети OSPF на обоих роутерах.

## УРОК №8



Очередная внештатная ситуация. OSPF настроено между роутерами R1 и R2, но не все сети объявлены. Loopback интерфейсы роутера R2 перераспределяются в OSPF. Вот настройки обоих роутеров:

```
R1#show run | section router ospf
router ospf 1
  log-adjacency-changes
  network 192.168.12.0 0.0.0.255 area 0

R2#show run | section router ospf
router ospf 1
  log-adjacency-changes
  redistribute connected
  network 192.168.12.0 0.0.0.255 area 0
```

Мы наблюдаем команду **redistribute connected** на роутере R2, которая должна перераспределить сети на интерфейсах обратной связи в OSPF.

```
R1#show ip route
C    192.168.12.0/24 is directly connected, FastEthernet0/0
```

Однако здесь ничего нет ...

```
R1#show ip protocols | include filter
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set

R2#show ip protocols | include filter
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
```

Обычно было бы неплохо проверить, есть ли distribute list или нет.

```
R2#show run | include redistribute
  redistribute connected
```

Ключ к решению этой проблемы - эта команда. Если вы наберете redistribute connected OSPF будет распространять только classful networks.

```
R2(config)#router ospf 1
R2(config-router)#redistribute connected subnets
```

Нам нужно добавить параметр "**subnets**", позволяющий заставить его выполнять redistribute subnet основных сетей.

```
R1#show ip route ospf
 2.0.0.0/24 is subnetted, 1 subnets
O E2      2.2.2.0 [110/20] via 192.168.12.2, 00:00:52, FastEthernet0/0
22.0.0.0/24 is subnetted, 1 subnets
O E2      22.22.22.0 [110/20] via 192.168.12.2, 00:00:51, FastEthernet0/0
```

Ну вот, наша маршрутная таблица заполнена.

**Итог урока:** добавьте параметр " subnets " при использовании перераспределения или перераспределяются только classful networks.