

2. Основы маршрутизации.

2.1. Маршрутизация.

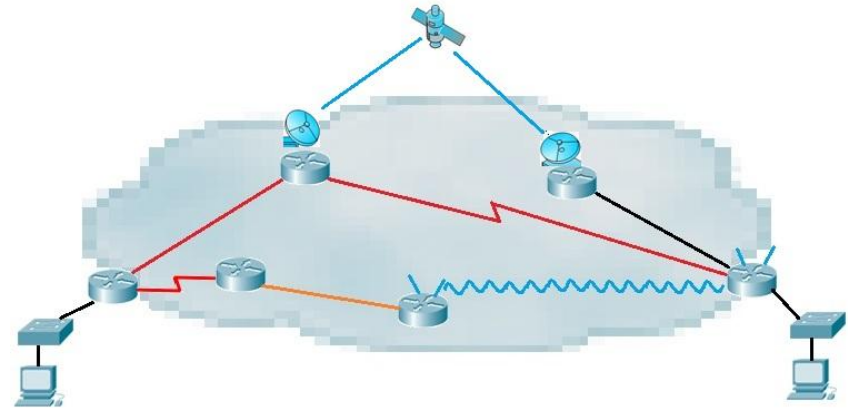
2.1.1. Задачи маршрутизации.

Для передачи данных между различными сетями используется маршрутизатор. Данное сетевое устройство работает на сетевом уровне модели OSI. Основными задачами маршрутизатора являются выбор наилучшего маршрута до сети назначения и коммутация пакета для передачи.

Использование маршрутизаторов предполагает, что сеть будет иметь сложную структуру с различными технологиями передачи данных на канальном и физическом уровнях и множеством маршрутов, в том числе с избыточными связями.

Исходя из этого, можно сказать следующее:

1. Маршрутизатор строит таблицу маршрутизации на основании информации, полученной через свои интерфейсы, от соседних маршрутизаторов.
2. Маршрутизатор принимает решение о передаче пакета, основываясь только на информации, которая содержится в его таблице маршрутизации.
3. В сети, построенной на маршрутизаторах, информация, которая передается к месту назначения по одному маршруту, обратно может передаваться по другому маршруту.



2.1.2. Таблицы маршрутизации

Цель маршрутизации – это выбор оптимального маршрута доставки данных из одной сети в другую.

Для обеспечения маршрутизации строятся специальные таблицы, которые называются таблицами маршрутизации.

На рисунке представлен пример таблицы маршрутизации маршрутизатора фирмы “Cisco”.

Давайте рассмотрим ее содержимое.

В верхней части, выделенная зеленым, находится справочная информация об источниках маршрутной информации (как запись о маршруте попала в таблицу).

Строка, выделенная желтым цветом, указывает на наличие маршрута по умолчанию.

Остальные строки указывают номера «известных» маршрутизатору сетей с определением способа получения маршрутной информации и параметрами маршрута.

```
R1#show 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 - 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

R 10.0.0.0/8 [120/1] via 192.168.1.2, 00:00:12, Serial0/0/0
R 11.0.0.0/8 [120/1] via 192.168.2.2, 00:00:02, Serial0/0/1
172.1.0.0/24 is subnetted, 2 subnets
C 172.1.1.0 is directly connected, FastEthernet0/0
C 172.1.2.0 is directly connected, FastEthernet0/1
C 192.168.1.0/24 is directly connected, Serial0/0/0
C 192.168.2.0/24 is directly connected, Serial0/0/1
R 192.168.3.0/24 [120/1] via 192.168.1.2, 00:00:12, Serial0/0/0
[120/1] via 192.168.2.2, 00:00:02, Serial0/0/1

R1#
```

Рис 2.1.2.

2.1.3. Маршрутная информация

Информация о маршруте, которая находится в таблице маршрутизации, содержит следующие данные:

1. Источник получения маршрутной информации.
2. Номер (*IP*-адрес) сети назначения и префикс (маску) .
3. Параметры маршрута (метрика и административная дистанция).

Административная дистанция определяет степень доверия к источнику маршрутной информации.

Метрика определяет стоимость маршрута до сети назначения и формируется источником маршрутной информации.

Если один источник маршрутной информации получит несколько маршрутов до сети назначения, то в таблицу маршрутизации будет занесен маршрут с наименьшей метрикой.

Если до сети назначения будет получено несколько маршрутов от разных источников, то в таблицу маршрутизации будет занесен маршрут с наименьшим значением данного параметра.

4. *IP*-адрес интерфейса соседнего маршрутизатора по пути к сети назначения (*next hop*).
5. Время, прошедшее с момента обновления информации по данному маршруту.
6. Имя выходного интерфейса данного маршрутизатора для передачи пакета к сети назначения.

```
R1#show 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 - 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

R 10.0.0.0/8 [120/1] via 192.168.1.2, 00:00:12, Serial0/0/0
R 11.0.0.0/8 [120/1] via 192.168.2.2, 00:00:02, Serial0/0/1
172.1.0.0/24 is subnetted, 2 subnets
C 172.1.1.0 is directly connected, FastEthernet0/0
C 172.1.2.0 is directly connected, FastEthernet0/1
C 192.168.1.0/24 is directly connected, Serial0/0/0
C 192.168.2.0/24 is directly connected, Serial0/0/1
R 192.168.3.0/24 [120/1] via 192.168.1.2, 00:00:12, Serial0/0/0
[120/1] via 192.168.2.2, 00:00:02, Serial0/0/1

R1#
```

Рис 2.1.3.

2.1.4. Источники маршрутной информации

Можно выделить три пути, по которым маршрутная информация попадает в таблицу:

В первую очередь, в таблицу маршрутизации заносятся записи о сетях, которые непосредственно подключены к интерфейсам маршрутизатора. Они определяются как «C» - *connected* (выделено зеленым). Указывается, что данная сеть является непосредственно подключенной (*is directly connected*).

Обратите внимание на то, что в этих маршрутах не указаны значения метрики и административной дистанции. Для маршрутов данного типа значения этих параметров равны нулю.

Второй путь – с помощью статической маршрутизации (выделено желтым).

Маршрутизатор узнает о маршруте, когда администратор настраивает маршрут вручную (статически). В этом случае при изменении топологии сети записи о маршрутах должны обновляться вручную.

Здесь так же не указаны значения метрики и административной дистанции. Для маршрутов данного типа значение метрики – «0», а административной дистанции (по умолчанию) – «1».

Третий путь – с помощью динамической маршрутизации (выделено голубым).

Маршрутизатор динамически изучает маршруты, после того как на нем настроен протокол маршрутизации. Протокол маршрутизации автоматически обновляет маршруты при получении информации о смене сетевой топологии. Маршрутизатор изучает и поддерживает маршруты к удаленным сетям, обмениваясь обновлениями маршрутной информации с другими маршрутизаторами в сети.

```
R1#show 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 - 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

R 10.0.0.0/8 [120/1] via 192.168.1.2, 00:00:04,
Serial0/0/0
R 11.0.0.0/8 [120/1] via 192.168.2.2, 00:00:06,
Serial0/0/1
150.160.0.0/24 is subnetted, 1 subnets
S 150.160.10.0 is directly connected, Serial0/0/0
172.1.0.0/24 is subnetted, 2 subnets
C 172.1.1.0 is directly connected, FastEthernet0/0
C 172.1.2.0 is directly connected, FastEthernet0/1
C 192.168.1.0/24 is directly connected, Serial0/0/0
C 192.168.2.0/24 is directly connected, Serial0/0/1
R 192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:06,
Serial0/0/1
[120/1] via 192.168.1.2, 00:00:04, Serial0/0/0
R1#
```

Рис 2.1.4.

2.1.5. Формирование таблиц маршрутизации

Теперь рассмотрим, в каком порядке устанавливаются записи о маршрутах в таблице маршрутизации.

Все записи устанавливаются в порядке возрастания номера сети назначения.

При этом, если префикс (маска) номера сети соответствует классу сети, то такой маршрут записывается в таблицу без дополнений.

В нашем примере это маршруты в сети 10.0.0.0, 11.0.0.0 и 192.168.3.0.

Если префикс больше классовой маски сети, то в таблице создается дополнительная запись с классовой маской и указанием количества подсетей, которые «известны маршрутизатору» и принадлежат данному классу. Такую запись можно назвать родительский маршрут.

В нашем примере это 150.160.0.0/24 и 172.1.0.0/16.

Те маршруты, номера сетей которых являются подсетями для родительских, называются – наследники.

В префиксе родительского маршрута указывается значение префикса наследников (подсетей), если у всех наследников префиксы равны. Если у наследников префиксы разные, то родителю записывается классовый префикс. В нашем примере посмотрите на фрагмент, выделенный зеленым цветом.

Если до сети назначения известно несколько маршрутов с одинаковой метрикой, то они записываются друг за другом. В нашем примере посмотрите на фрагмент, выделенный желтым цветом.

```
R1#show 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 - 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

R 10.0.0.0/8 [120/1] via 192.168.1.2, 00:00:02, Serial0/0/0
R 11.0.0.0/8 [120/1] via 192.168.2.2, 00:00:16, Serial0/0/1
150.160.0.0/24 is subnetted, 1 subnets
S 150.160.10.0 is directly connected, Serial0/0/0
172.1.0.0/16 is variably subnetted, 2 subnets, 2 masks
C 172.1.1.0/26 is directly connected, FastEthernet0/0
C 172.1.2.0/24 is directly connected, FastEthernet0/1
C 192.168.1.0/24 is directly connected, Serial0/0/0
C 192.168.2.0/24 is directly connected, Serial0/0/1
R 192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:16, Serial0/0/1
[120/1] via 192.168.1.2, 00:00:02, Serial0/0/0

R1#
```

Рис 2.1.5.

2.1.6. Поиск маршрутов в таблице маршрутизации

Как же маршрутизатор выбирает маршрут из таблицы маршрутизации?

Когда на интерфейс маршрутизатора приходит информация, то он распаковывает ее до уровня пакета и извлекает адрес назначения.

После этого начинается побитное сравнение этого адреса с номерами сетей в таблице маршрутизации по порядку сверху вниз.

Как только находится строка, в которой номер сети полностью совпадает с адресом назначения в пределах, определяемых префиксом этой сети, то пакет будет передан на соответствующий выходной интерфейс.

Если нет ни одного совпадения, то при наличии маршрута по умолчанию, пакет будет передан на соответствующий выходной интерфейс. Если маршрут по умолчанию не настроен, то пакет будет уничтожен.

```
R1#show 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 - 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

R 10.0.0.0/8 [120/1] via 192.168.1.2, 00:00:02, Serial0/0/0
R 11.0.0.0/8 [120/1] via 192.168.2.2, 00:00:16, Serial0/0/1
  150.160.0.0/24 is subnetted, 1 subnets
S 150.160.10.0 is directly connected, Serial0/0/0
  172.1.0.0/16 is variably subnetted, 2 subnets, 2 masks
C 172.1.1.0/26 is directly connected, FastEthernet0/0
C 172.1.2.0/24 is directly connected, FastEthernet0/1
C 192.168.1.0/24 is directly connected, Serial0/0/0
C 192.168.2.0/24 is directly connected, Serial0/0/1
R 192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:16, Serial0/0/1
  [120/1] via 192.168.1.2, 00:00:02, Serial0/0/0

R1#
```

Рис 2.1.6.

2.1.7. Оптимизация поиска маршрутов в таблице маршрутизации

На предыдущем слайде мы рассмотрели принцип обработки маршрутизатором таблицы маршрутизации.

Но что, если в таблице маршрутизации будет много записей? Это приведет к значительным задержкам при обработке таблицы маршрутизации.

Но тут срабатывает механизм «родитель-наследник».

Если произошло совпадение адреса назначения с номером сети в строке «родителя», то дальше выполняется проверка по строкам наследникам, и если совпадения не было ни с одним из «наследников», то пакет сразу передается на маршрут по умолчанию, если он есть. Если маршрута по умолчанию нет, то пакет будет отброшен.

```
R1#show 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 - 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

R 10.0.0.0/8 [120/1] via 192.168.1.2, 00:00:02, Serial0/0/0
R 11.0.0.0/8 [120/1] via 192.168.2.2, 00:00:16, Serial0/0/1
  150.160.0.0/24 is subnetted, 1 subnets
  S 150.160.10.0 is directly connected, Serial0/0/0
  172.1.0.0/16 is variably subnetted, 2 subnets, 2 masks
  C 172.1.1.0/26 is directly connected, FastEthernet0/0
  C 172.1.2.0/24 is directly connected, FastEthernet0/1
  C 192.168.1.0/24 is directly connected, Serial0/0/0
  C 192.168.2.0/24 is directly connected, Serial0/0/1
R 192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:16, Serial0/0/1
  [120/1] via 192.168.1.2, 00:00:02, Serial0/0/0

R1#
```

Рис 2.1.7.

2.2. Статическая маршрутизация.

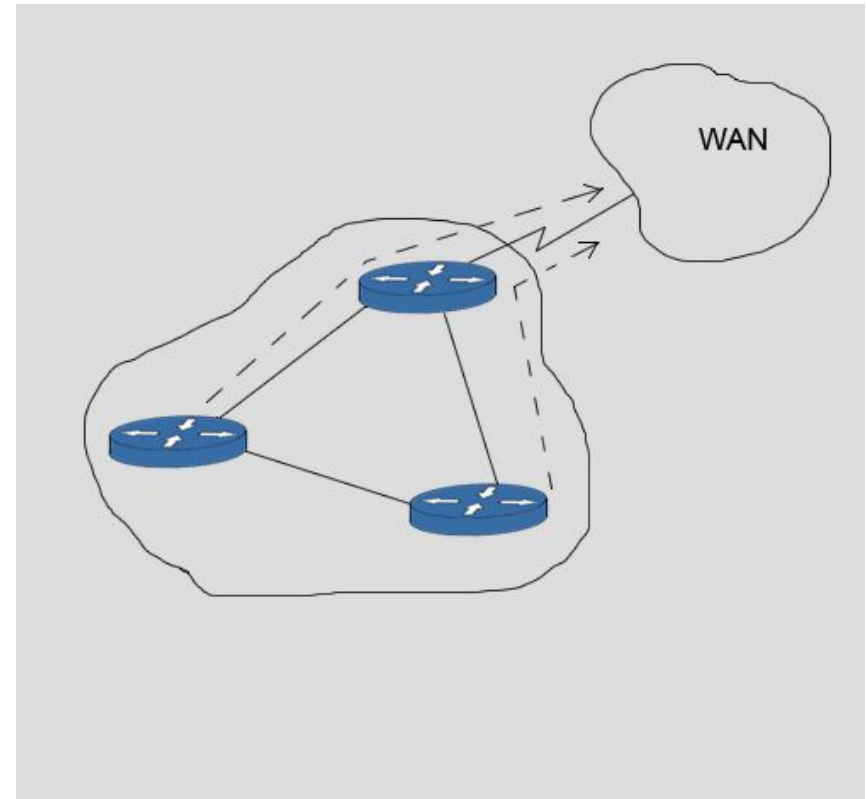
2.2.1. Статические маршруты.

Статические маршруты обычно используются, когда необходимо обеспечить маршрут в сеть, достижимую только по одному маршруту.

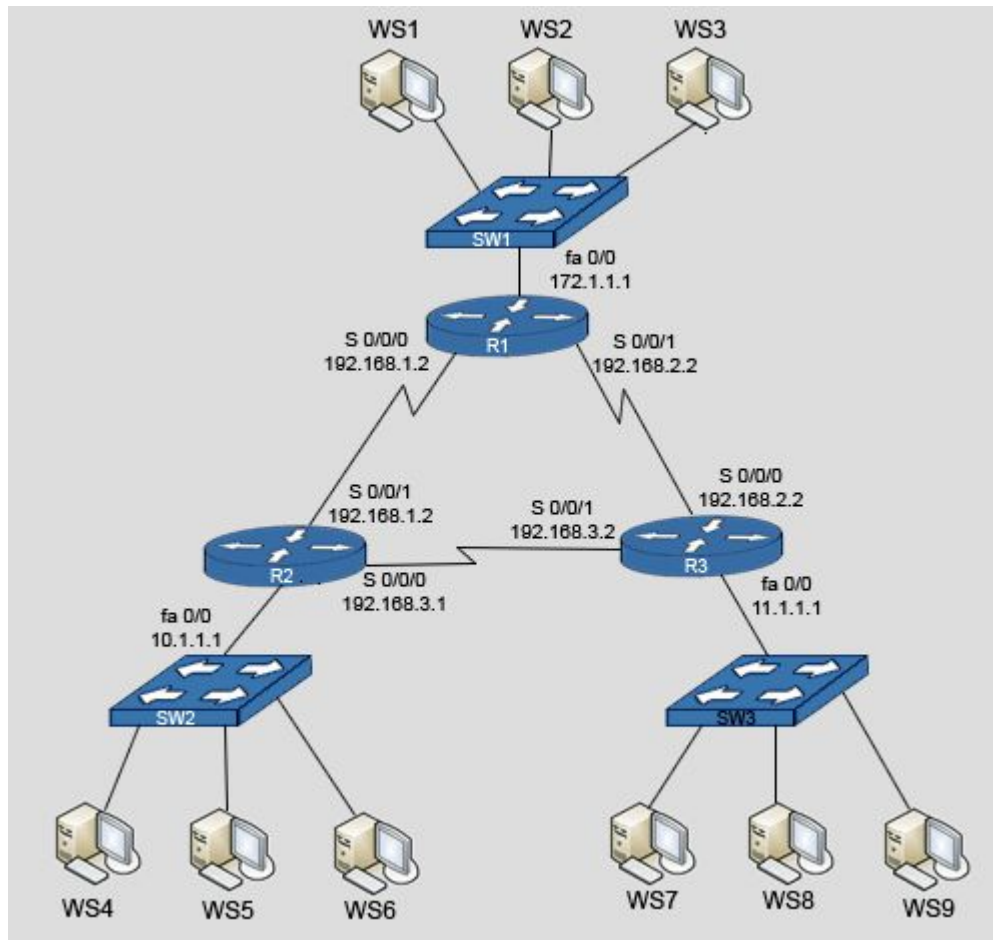
Статические маршруты также используются для определения «маршрута по умолчанию» (*default gateway*) на граничном маршрутизаторе.

Так же статическая маршрутизация может использоваться для формирования транзитного маршрута через сеть в целях обеспечения безопасности в данной сети.

Статический маршрут настраивается для соединения с удаленными сетями непосредственно не подключенными к маршрутизатору. Для двухстороннего взаимодействия статический маршрут должен быть настроен в обоих направлениях



2.2.2. Настройка маршрутизаторов для работы в сети.



Настроим маршрутизаторы в соответствии со схемой
Для просмотра настройки маршрутизаторов нажмите кнопки R1 R2 R3

```
R1#config t
R1(config)#interface fa0/0
R1(config-if)#ip address 172.1.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#interface serial 0/0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#clock rate 64000
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#interface serial 0/0/1
R1(config-if)#ip address 192.168.2.1 255.255.255.0
R1(config-if)#no shutdown
```

```
R1#show run
Building configuration...
!
version 12.4
!
hostname R1
!
interface FastEthernet0/0
ip address 172.1.1.1 255.255.255.0
!
interface Serial0/0/0
ip address 192.168.1.1 255.255.255.0
clock rate 64000
!
interface Serial0/0/1
ip address 192.168.2.1 255.255.255.0
...
end
```

```
R2#config t
R2(config)#interface fa0/0
R2(config-if)#ip address 10.1.1.1 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#interface serial 0/0/0
R2(config-if)#ip address 192.168.3.1 255.255.255.0
R2(config-if)#clock rate 64000
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#interface serial 0/0/1
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#no shutdown
```

```
R2#show run
Building configuration...
!
version 12.4
!
hostname R2
!
interface FastEthernet0/0
ip address 10.1.1.1 255.255.255.0
!
interface Serial0/0/0
ip address 192.168.3.1 255.255.255.0
clock rate 64000
!
interface Serial0/0/1
ip address 192.168.1.2 255.255.255.0
...
end
```

```
R3#config t
R3(config)#interface fa0/0
R3(config-if)#ip address 11.1.1.1 255.255.255.0
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#interface serial 0/0/0
R3(config-if)#ip address 192.168.2.2 255.255.255.0
R3(config-if)#clock rate 64000
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#interface serial 0/0/1
R3(config-if)#ip address 192.168.3.2 255.255.255.0
R3(config-if)#no shutdown
```

```
R3#show run
Building configuration...
!
version 12.4
!
hostname R3
!
interface FastEthernet0/0
ip address 11.1.1.1 255.255.255.0
!
interface Serial0/0/0
ip address 192.168.2.2 255.255.255.0
clock rate 64000
!
interface Serial0/0/1
ip address 192.168.3.2 255.255.255.0
...
end
```

2.2.3. Начальное состояние таблиц маршрутизации.

Пока на маршрутизаторах не настроены маршруты, то таблицы маршрутизации будут только те сети, которые непосредственно к ним подсоединены.

```
R1#show ip route
```

```
Codes: C - connected,...
```

```
...
```

```
172.1.0.0/24 is subnetted, 1 subnets
```

```
C 172.1.1.0 is directly connected, FastEthernet0/0
```

```
C 192.168.1.0/24 is directly connected, Serial0/0/0
```

```
C 192.168.2.0/24 is directly connected, Serial0/0/1
```

```
R2#show ip route
```

```
Codes: C - connected,...
```

```
...
```

```
10.0.0.0/24 is subnetted, 1 subnets
```

```
C 10.1.1.0 is directly connected, FastEthernet0/0
```

```
C 192.168.1.0/24 is directly connected, Serial0/0/1
```

```
C 192.168.3.0/24 is directly connected, Serial0/0/0
```

```
R3#show ip route
```

```
Codes: C - connected, ...
```

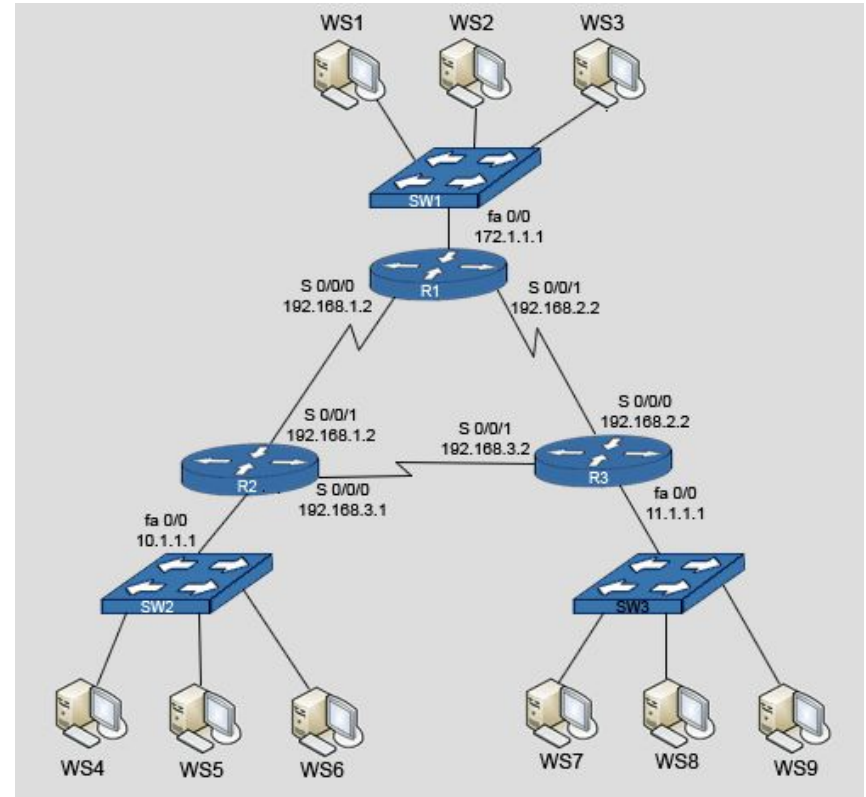
```
...
```

```
11.0.0.0/24 is subnetted, 1 subnets
```

```
C 11.1.1.0 is directly connected, FastEthernet0/0
```

```
C 192.168.2.0/24 is directly connected, Serial0/0/0
```

```
C 192.168.3.0/24 is directly connected, Serial0/0/1
```



2.2.4. Проверка связи с соседним маршрутизатором.

С маршрутизатора *R1* выполним команду *ping* *192.168.1.2*. Так как эта сеть непосредственно подсоединена к *R1*, то на *ICMP echo request* получены отклики *ICMP echo replay*.

```
R1#ping 192.168.1.2
```

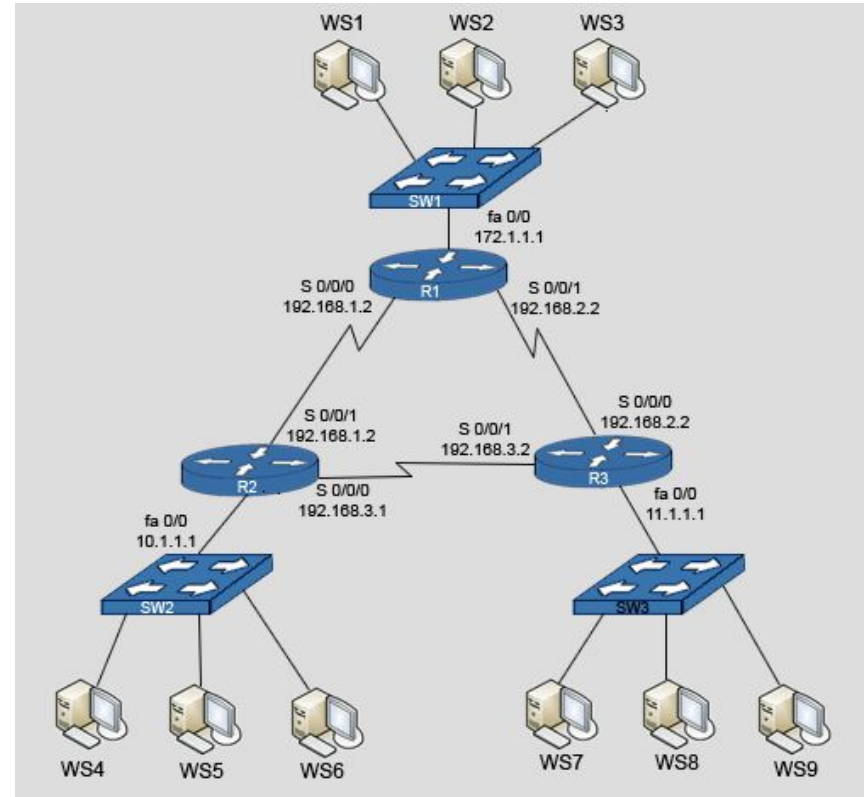
Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 32/33/36 ms

```
R1#
```



2.2.5. Проверка связи с удаленной сетью.

Теперь с *WS1* выполним команду *ping 10.1.1.1*. Так как эта сеть непосредственно не подсоединена к *R1*, то на *ICMP echo request* отклики получены не были.

```
R1#ping 10.1.1.1
```

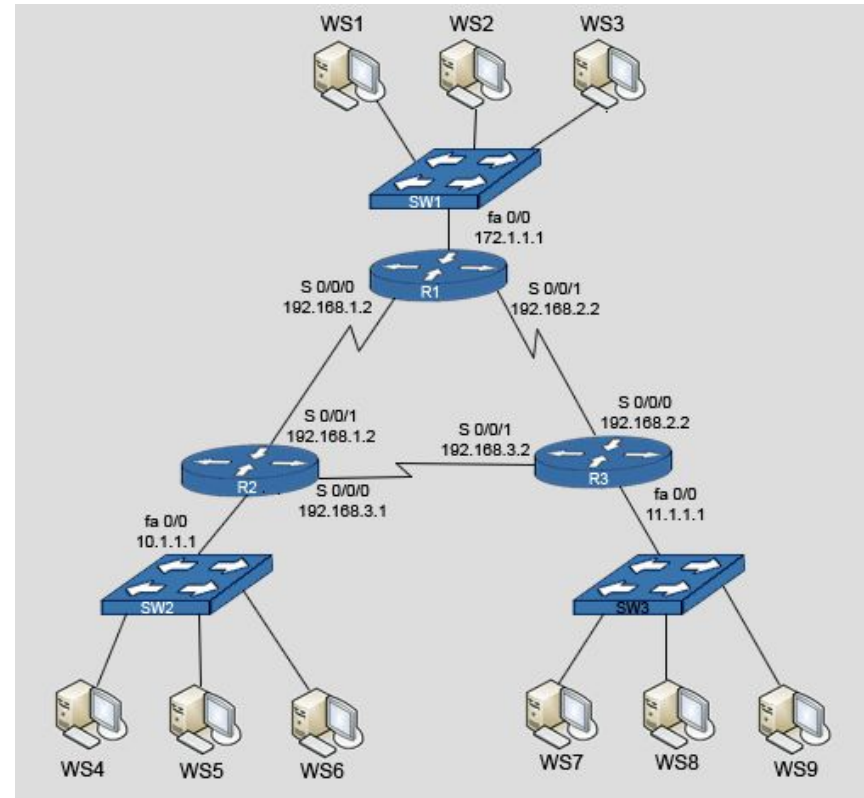
Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

```
R1#
```



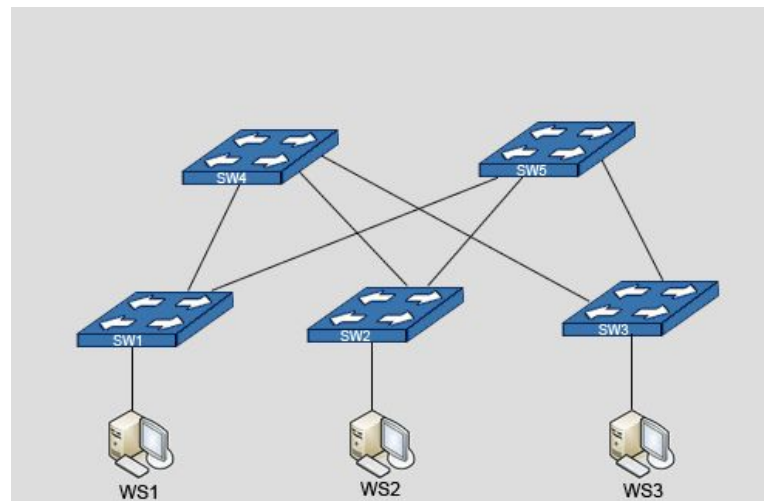
2.2.6. Конфигурирование статического маршрута.

Для настройки статического маршрута необходимо ввести команду *ip route* в режиме глобальной конфигурации.

Router(config)#ip route network [mask] {address|interface} [distance] [permanent]

В поле *{address|interface}* указывается *ip*-адрес интерфейса соседнего маршрутизатора, на который надо передавать пакет для достижения сети назначения. Для ускорения процесса анализа таблиц маршрутизации предпочтительно использовать вместо адреса тип исходящего интерфейса текущего маршрутизатора. В соединении точка-точка проблем не возникает.

Одним из параметров статического маршрута является административная дистанция. Значением административной дистанции являются целые числа в диапазоне от 0 до 255. Маршрут с меньшим значением административной дистанции заслуживает большего доверия, чем с большим значением.



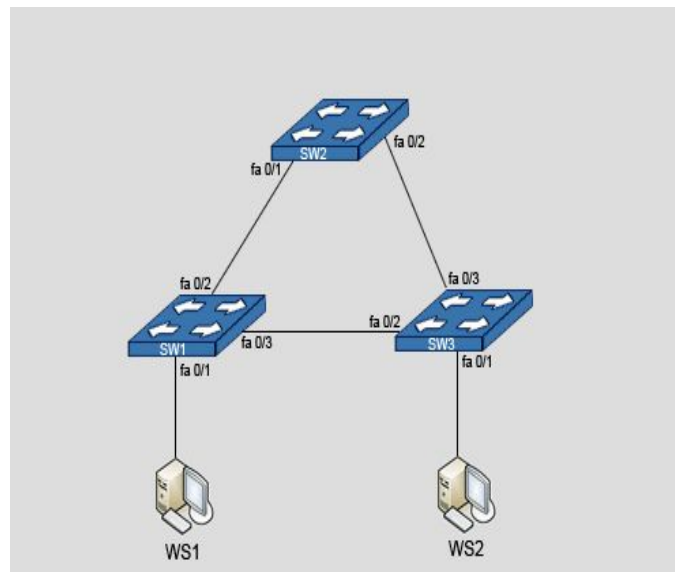
Параметры команды ip route	Описание
<i>network</i>	Сеть, подсеть или хост назначения.
<i>mask</i>	Маска подсети.
<i>address</i>	IP-адрес следующего интерфейса следующего маршрутизатора (<i>next hop</i>).
<i>interface</i>	Если используется соединение типа точка-точка, то можно указывать имя интерфейса для достижения сети назначения. Команда не будет корректно работать при подключении через интерфейс множественного доступа (<i>Ethernet</i>).
<i>distance</i>	Определение административной дистанции (необязателен).
<i>permanent</i>	Определяет, что маршрут не удалится, даже если отключить интерфейс (необязателен).

2.2.7. Статический маршрут с исходящим интерфейсом.

Если используется технология множественного доступа (*Ethernet*), то для передачи данных необходим физический адрес, который определяется с помощью протокола разрешения адреса (*address resolution protocol – ARP*) по *ip*-адресу.

В таком случае, необходимо использовать команду формирования статического маршрута следующего вида

```
Router(config)#ip route network [mask] interface  
address [distance] [permanent]
```



Параметры команды ip route	Описание
<i>network</i>	Сеть, подсеть или хост назначения.
<i>mask</i>	Маска подсети.
<i>address</i>	IP-адрес следующего интерфейса следующего маршрутизатора (next hop).
<i>interface</i>	Имя исходящего интерфейса для достижения сети назначения.
<i>address</i>	<i>ip</i> -адрес интерфейса соседнего маршрутизатора
<i>distance</i>	Определение административной дистанции (необязателен).
<i>permanent</i>	Определяет, что маршрут не удалится, даже если отключить интерфейс (необязателен).

2.2.8. Демонстрационный пример конфигурирования статических маршрутов.

Настроим на маршрутизаторе *R1* статические маршруты из сети 172.1.1.0 в сети 10.1.1.0 и 11.1.1.0 (маска 255.255.255.0). Так как маршрутизаторы соединены между собой по последовательным интерфейсам, то один маршрут настроим, используя *ip*-адрес следующего маршрутизатора (*next hop*), а другой с использованием имени исходящего интерфейса.

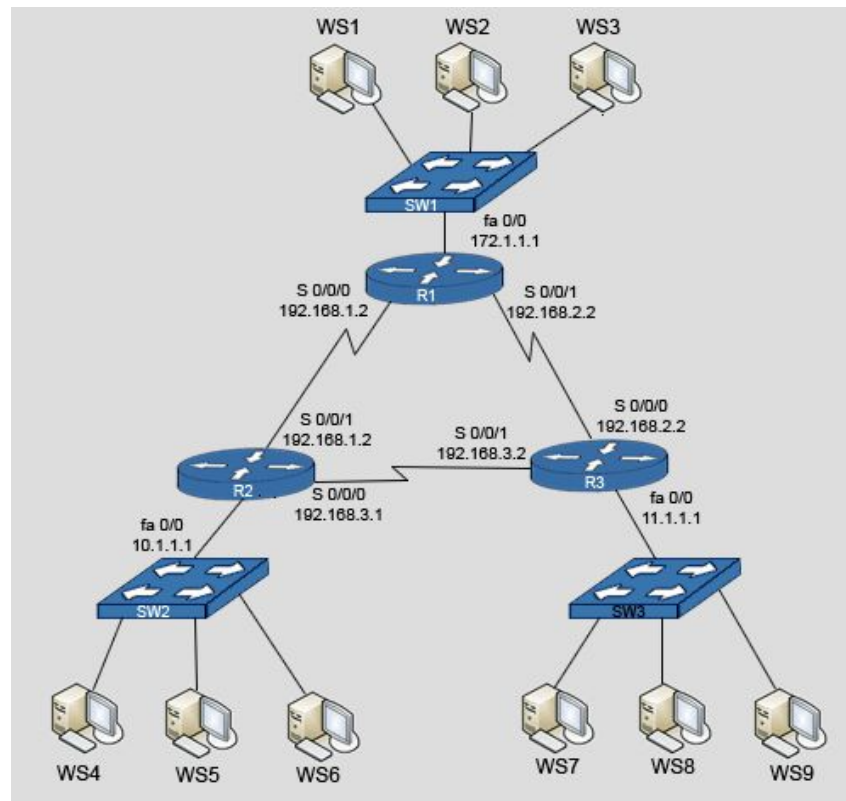
```
R1(config)#ip route 10.1.1.0 255.255.255.0 192.168.1.2
R1(config)#ip route 11.1.1.0 255.255.255.0 serial 0/0/1
R1(config)#^Z
```

```
R1#show 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 - 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

```
10.0.0.0/24 is subnetted, 1 subnets
S   10.1.1.0 [1/0] via 192.168.1.2
11.0.0.0/24 is subnetted, 1 subnets
S   11.1.1.0 is directly connected, Serial0/0/1
172.1.0.0/24 is subnetted, 1 subnets
C   172.1.1.0 is directly connected, FastEthernet0/0
C   192.168.1.0/24 is directly connected, Serial0/0/0
C   192.168.2.0/24 is directly connected, Serial0/0/1
R1#
```



2.2.9. Проверка связи с удаленной сетью после настройки R1.

Теперь с *WS1* выполним команду *ping* на *WS4* с *ip*-адресом 10.1.1.2.

```
C:\>ping 10.1.1.2
```

Обмен пакетами с 192.168.2.2 по 32 байт:

Превышен интервал ожидания для запроса.
Превышен интервал ожидания для запроса.
Превышен интервал ожидания для запроса.
Превышен интервал ожидания для запроса.

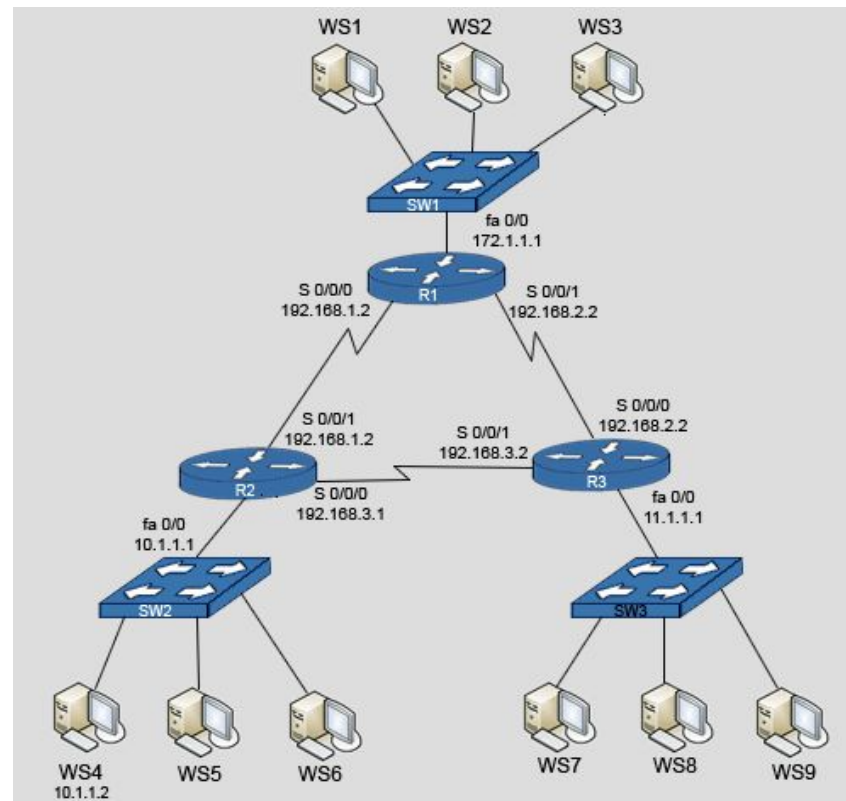
Статистика Ping для 10.1.1.2:

Пакетов: отправлено = 4, получено = 0, потеряно = 4 (100% потеря),

Ответ получен не был, так как на маршрутизаторе *R2* не был прописан обратный маршрут с *WS4* к *WS1*, и он не знал, на какой интерфейс отправлять пакеты для сети 172.1.1.0.

Для R1

S0/0/0	10.1.1.0
S0/0/1	11.1.1.0
Fa0/1	172.1.1.0
S0/0/0	192.168.1.0
S0/0/1	192.168.2.0



2.2.10 Целостность маршрутной информации.

Теперь настроим на маршрутизаторе R2 маршрут в сеть 172.1.1.0:

```
R2(config)#ip route 172.1.1.0 255.255.255.0 serial 0/0/1
R2(config)#^Z
R2#show ip route
```

```
...
10.0.0.0/24 is subnetted, 1 subnets
C 10.1.1.0 is directly connected, FastEthernet0/0
172.1.0.0/24 is subnetted, 1 subnets
S 172.1.1.0 is directly connected, Serial0/0/1
C 192.168.1.0/24 is directly connected, Serial0/0/1
C 192.168.3.0/24 is directly connected, Serial0/0/0
R2#
```

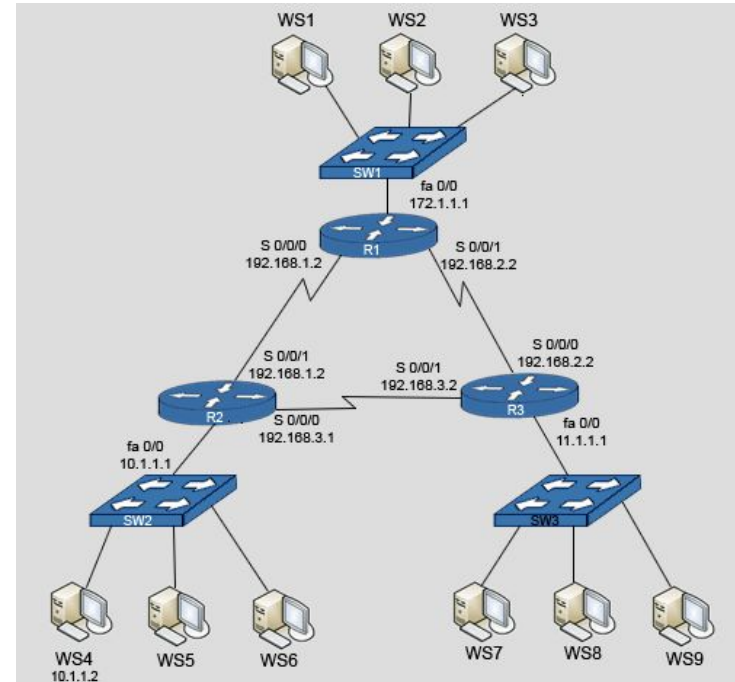
После этого опять выполним команду ping с WS1 на WS4

```
C:\>ping 10.1.1.2
Обмен пакетами с 10.1.1.2 по 32 байт:
Ответ от 10.1.1.2: число байт=32 время=20мс TTL = 126
Ответ от 10.1.1.2: число байт=32 время=20мс TTL=126
Ответ от 10.1.1.2: число байт=32 время=20мс TTL=126
Ответ от 10.1.1.2: число байт=32 время=20мс TTL=126
```

Статистика Ping для 10.1.1.2:
 Пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
 Приблизительное время передачи и приема:
 наименьшее = 20мс, наибольшее = 20мс, среднее = 20мс

C:\>

В результате был получен ответ. Теперь рабочие станции из сетей 172.1.1.0 и 10.1.1.0 могут обмениваться информацией.



Для R1	
S0/0/0	10.1.1.0
S0/0/1	11.1.1.0
Fa0/1	172.1.1.0
S0/0/0	192.168.1.0
S0/0/1	192.168.2.0

Для R2	
Fa0/1	10.1.1.0
S0/0/1	172.1.1.0
S0/0/1	192.168.1.0
S0/0/0	192.168.3.0

2.2.11. Конфигурирование маршрута по умолчанию.

Маршрут по умолчанию (*default route*) применяется в ситуациях, когда маршрут от источника к приемнику не известен или когда невозможно поддерживать на маршрутизаторе много маршрутов в таблице маршрутизации.

Часто такой маршрут используется на граничном маршрутизаторе сети при подключении к Интернет сервис провайдеру.

Для конфигурирования маршрута по умолчанию так же используется команда *ip route*. Но в полях адрес назначения и маска записывается 0.0.0.0.

Параметры команды <i>ip route</i>	Описание
<i>ip route</i>	Определяет команду описания статического маршрута.
0.0.0.0	Все маршруты, не представленные в таблице маршрутизации сети.
0.0.0.0	Особая маска, указывающая на маршрут по умолчанию.
<i>Address</i>	IP-адрес следующего маршрутизатора, используемый по умолчанию для передачи пакетов.

Рис 2.2.11 (3.12)

2.2.12 Демонстрационный пример конфигурации маршрута по умолчанию.

Сконфигурируем на маршрутизаторе R1 виртуальный интерфейс `loopback 0` с адресом `190.1.1.1`. Это интерфейс будем как имитацию для выхода в WAN.

```
R1(config)#interface loopback 0
R1(config-if)#ip address 190.1.1.1
255.255.255.252
R1(config)#ip route 0.0.0.0 0.0.0.0 loopback 0
```

Теперь на маршрутизаторе R2 сконфигурируем маршрут по умолчанию в WAN через R1.

```
R2(config)#ip route 0.0.0.0 0.0.0.0 serial 0/0/1
```

Для проверки статических маршрутов необходимо ввести команду `show ip route` и посмотреть статические маршруты обозначенные "S". Звездочка (*) указывает на последний маршрут, использованный для передачи пакетов (маршрут по умолчанию). И появится запись - `Gateway of last resort is 0.0.0.0 to network 0.0.0.0`.

На маршрутизаторе R2 сеть `172.1.1.0` достижима через тот же интерфейс, где настроен маршрут по умолчанию. Это позволяет удалить статический маршрут в эту сеть, уменьшив размер таблицы маршрутизации, т.к. в любом случае пакет будет передан на маршрутизатор R1.

```
R1#show 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 - 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 0.0.0.0 to network 0.0.0.0
```

```
10.0.0.0/24 is subnetted, 1 subnets
S 10.1.1.0 [1/0] via 192.168.1.2
11.0.0.0/24 is subnetted, 1 subnets
S 11.1.1.0 is directly connected, Serial0/0/1
172.1.0.0/24 is subnetted, 1 subnets
C 172.1.1.0 is directly connected, FastEthernet0/0
190.1.0.0/30 is subnetted, 1 subnets
C 190.1.1.0 is directly connected, Loopback0
C 192.168.1.0/24 is directly connected, Serial0/0/0
C 192.168.2.0/24 is directly connected, Serial0/0/1
S* 0.0.0.0/0 is directly connected, Loopback0
R1#
```

```
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
...
```

```
Gateway of last resort is 0.0.0.0 to network 0.0.0.0
```

```
10.0.0.0/24 is subnetted, 1 subnets
C 10.1.1.0 is directly connected, FastEthernet0/0
172.1.0.0/24 is subnetted, 1 subnets
S 172.1.1.0 is directly connected, Serial0/0/1
C 192.168.1.0/24 is directly connected, Serial0/0/1
C 192.168.3.0/24 is directly connected, Serial0/0/0
S* 0.0.0.0/0 is directly connected, Serial0/0/1
R2#
```

Рис 2212 (3.13)

2.2.13. Команда *ip classless*.

Маршрутизатор может обрабатывать адреса назначения пакетов двумя способами: используя «классовую маску» или маску переменной длины.

В режиме *no ip classless* поиск в таблице маршрутизации основывается на классовой адресации. Когда на маршрутизатор приходит пакет, то адрес назначения сравнивается с адресами в таблице маршрутизации. Если совпадение в пределах классовой маски найдено, то выполняется поиск на уровне подсетей, если ионии есть. Если такого маршрута нет, то пакет будет отброшен независимо от того настроен маршрут по умолчанию или нет. А так же не анализируются записи с суперсетями (суперсети «*supernets*» - сети которые имеют маску с длиной меньше классовой для данного адреса).

Для возможности использования маршрута по умолчанию и супернеттинга необходимо включить режим *ip classless*.

На маршрутизаторе режим *ip classless* включен по умолчанию.

ip classless изменяется в режиме глобальной конфигурации.

```
R1#show 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 - 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 0.0.0.0 to network 0.0.0.0
```

```
10.0.0.0/24 is subnetted, 1 subnets
S 10.1.1.0 [1/0] via 192.168.1.2
11.0.0.0/24 is subnetted, 1 subnets
S 11.1.1.0 is directly connected, Serial0/0/1
172.1.0.0/24 is subnetted, 1 subnets
C 172.1.1.0 is directly connected, FastEthernet0/0
190.1.0.0/30 is subnetted, 1 subnets
C 190.1.1.0 is directly connected, Loopback0
S 191.16.0.0/16 is directly connected, Serial0/0/1
C 192.168.1.0/24 is directly connected, Serial0/0/0
C 192.168.2.0/24 is directly connected, Serial0/0/1
S* 0.0.0.0/0 is directly connected, Loopback0
R1#
```

Рис 2213 (3.14)

2.3. Динамические дистанционно векторные протоколы маршрутизации.

2.3.1. Основные концепции дистанционно векторных протоколов.

При использовании дистанционно векторных протоколов маршрутизации, маршрутизаторы строят таблицы маршрутизации путем периодических обменов своими таблицами маршрутизации с соседями (соседними маршрутизаторами).

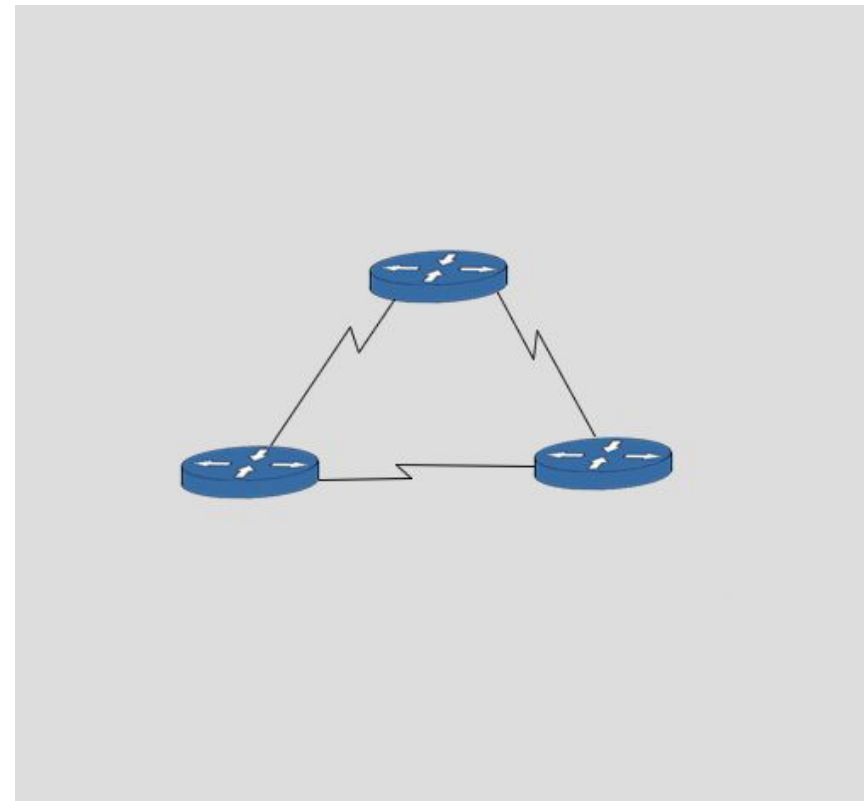
Соседние маршрутизаторы - это маршрутизаторы, интерфейсы которых физически находятся в одной сети. При этом наиболее часто используются широковещательные адреса.

Маршрутизаторы, на которых работают дистанционно векторные протоколы, отправляют периодические обновления, даже если не было изменения топологии сети.

После получения таблицы маршрутизации от соседа, маршрутизатор проверяет все известные маршруты и делает изменения в собственной таблице маршрутизации, основываясь на информации в обновлении.

Маршрутизаторы добавляют непосредственно подключенные сети в свои таблицы маршрутизации даже без включения протокола маршрутизации.

В обновление маршрутной информации включается как о непосредственно подсоединенных сетях, так и информацию о сетях полученную, от других маршрутизаторов. Поэтому каждый маршрутизатор знает обо всех сетях даже, если он связан с ними не непосредственно, а через другие маршрутизаторы.



2.3.2. Метрики, используемые дистанционно векторными протоколами маршрутизации.

Маршрутная информация включает номер сети и метрику. Если маршрутизатор имеет информацию о нескольких маршрутах в одну сеть, то он выбирает маршрут с наименьшей метрикой.

Для каждой сети маршрутизатор устанавливает интерфейс того маршрутизатора, с которого он получил информацию об этой сети.

- Счетчик хопов (*hop count*): число раз, которое пакет проходит через исходящий порт маршрутизаторов до сети назначения.

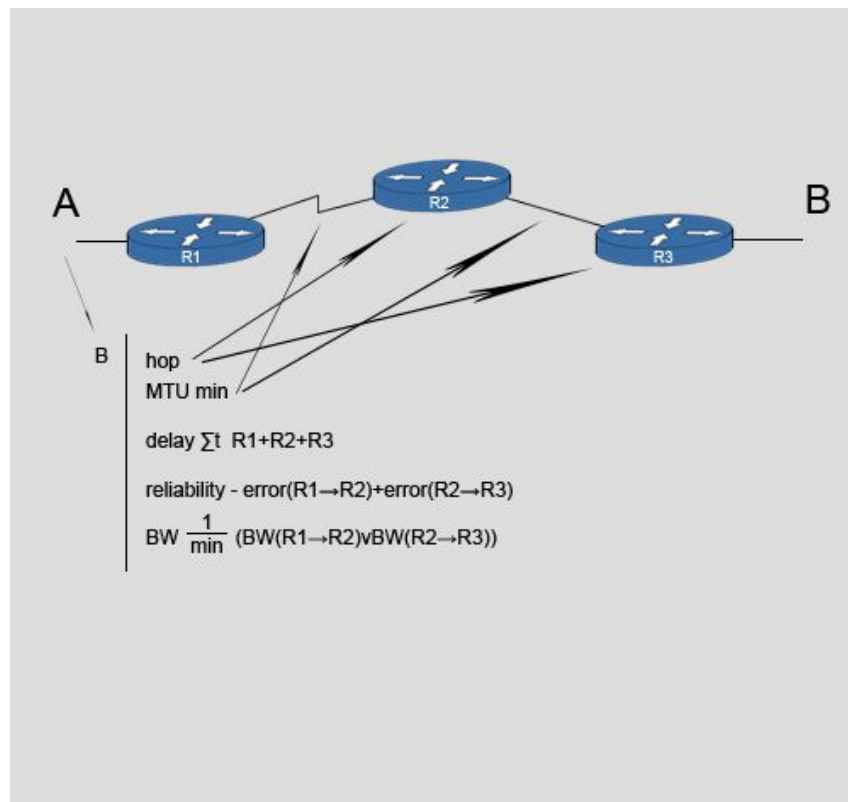
- Полоса пропускания (*bandwidth - BW*): скорость передачи данных по сегменту сети. Например, *Ethernet* предпочтительнее, чем 64-kbps выделенная линия.

- Задержка (*delay*): время, необходимое для перемещения пакета от источника к приемнику.

- Загрузка (*load*): объем загруженности сетевого ресурса, например, маршрутизатора или соединения на пути от источника к приемнику.

- Надежность (*reliability*): оценка количества ошибок на каждом сегменте сети на пути от источника к приемнику.

- *Maximum Transmission Unit (MTU)*: Максимальный размер сообщения в октетах, который возможен на всех соединениях на пути от источника к приемнику.



2.3.3. Пример работы дистанционно векторного протокола.

В начальный момент времени все маршрутизаторы «знают» только о сетях, которые подключены непосредственно к их интерфейсам. Сеть, непосредственно подключенная к интерфейсу маршрутизатора, имеет метрику 0.

Таблицы маршрутизации будут выглядеть следующим образом:

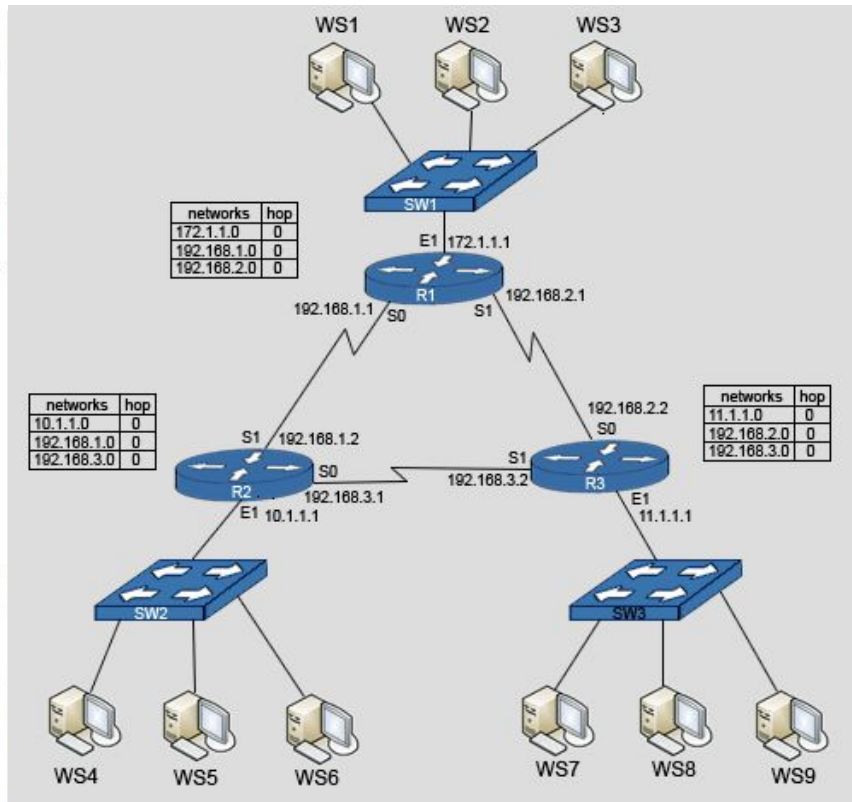
R1		R2		R3	
networks	Hop	networks	Hop	networks	Hop
172.1.1.0	0	10.1.1.0	0	11.1.1.0	0
192.168.1.0	0	192.168.1.0	0	192.168.2.0	0
192.168.2.0	0	192.168.3.0	0	192.168.3.0	0

Затем каждый маршрутизатор пошлет свою таблицу маршрутизации всем своим соседям.

Так как алгоритм вектора расстояния представляет собой процесс последовательного изучения сети, то маршрутизаторы обнаруживают наилучший путь к сетям непосредственно к ним не подключенным, основываясь на суммировании метрик, полученных от соседей. В результате в таблицу маршрутизации каждый маршрутизатор добавит тот маршрут, которого у него нет.

При передаче таблицы маршрутизации соседям, метрики всех маршрутов, записанных в ней, увеличиваются на 1.

В результате все маршрутизаторы имеют информацию о всех сетях.



Таблицы маршрутизации примут следующий вид:

R1		R2		R3	
networks	Hop	networks	Hop	networks	Hop
172.1.1.0	0	10.1.1.0	0	11.1.1.0	0
92.168.1.0	0	192.168.1.0	0	192.168.2.0	0
92.168.2.0	0	192.168.3.0	0	192.168.3.0	0
10.1.1.0	1	172.1.1.0	1	172.1.1.0	1
92.168.3.0	1	192.168.2.0	1	192.168.1.0	1
11.1.1.0	1	11.1.1.0	1	10.1.1.0	1

2.3.4. Поддержка маршрутной информации.

Таблица маршрутизации должна обновляться, когда происходят изменения топологии сети.

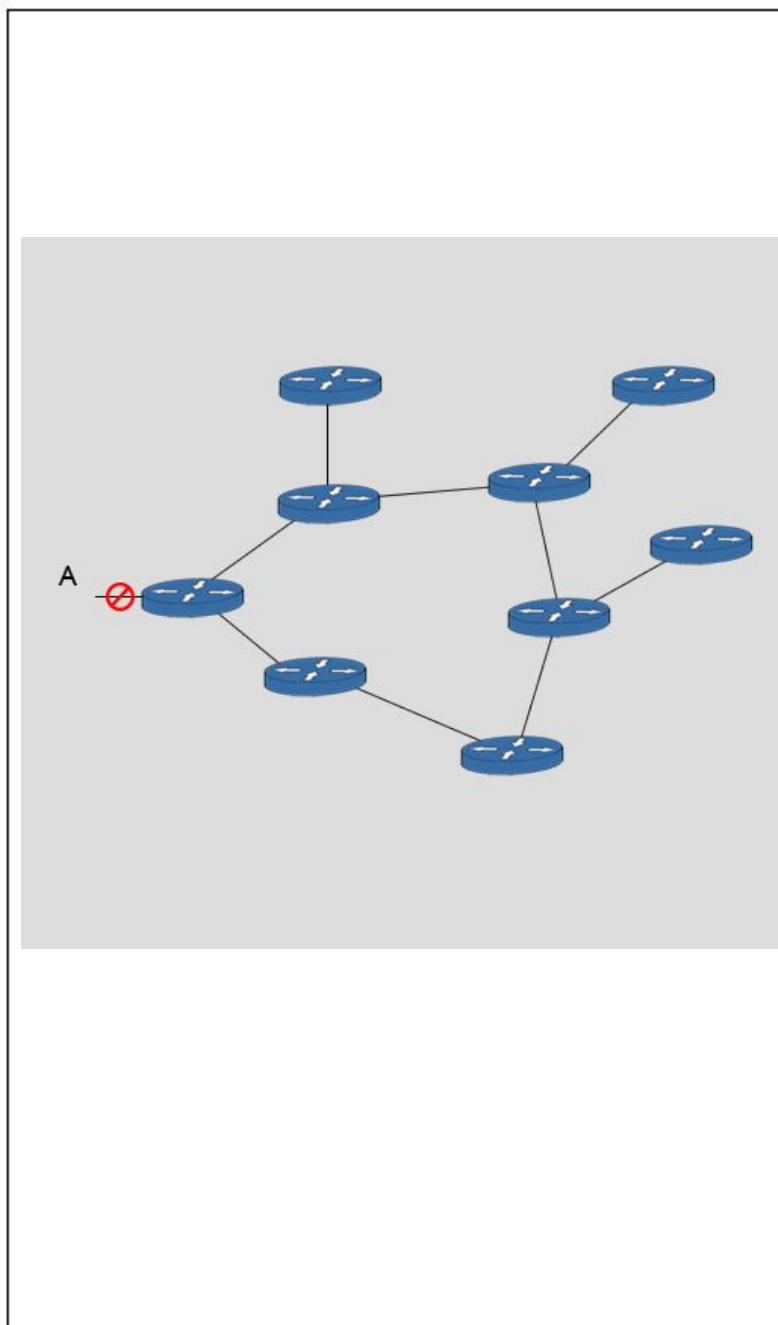
Подобно процессу изучения сети, обновления топологии происходят шаг за шагом от маршрутизатора к маршрутизатору.

Все маршрутизаторы отправляют записи таблицы маршрутизации каждому маршрутизатору-соседу.

Когда маршрутизатор получает обновления от соседнего маршрутизатора, он сравнивает информацию в обновлении с собственной таблицей маршрутизации. Для расчета новой метрики маршрутизатор добавляет стоимость достижения соседнего маршрутизатора к стоимости пути, полученной от соседа.

Если маршрутизатор узнает от соседа лучший маршрут (с меньшей метрикой) к сети, он обновляет собственную таблицу маршрутизации.

Каждая запись в таблице маршрутизации включает информацию об общей стоимости (определяемой метрикой в таблице маршрутизации) и логическом адресе первого маршрутизатора на пути в каждую сеть, которая прописана в таблице маршрутизации.



2.3.5. Нарушение целостности маршрутизации в дистанционно векторных протоколах маршрутизации

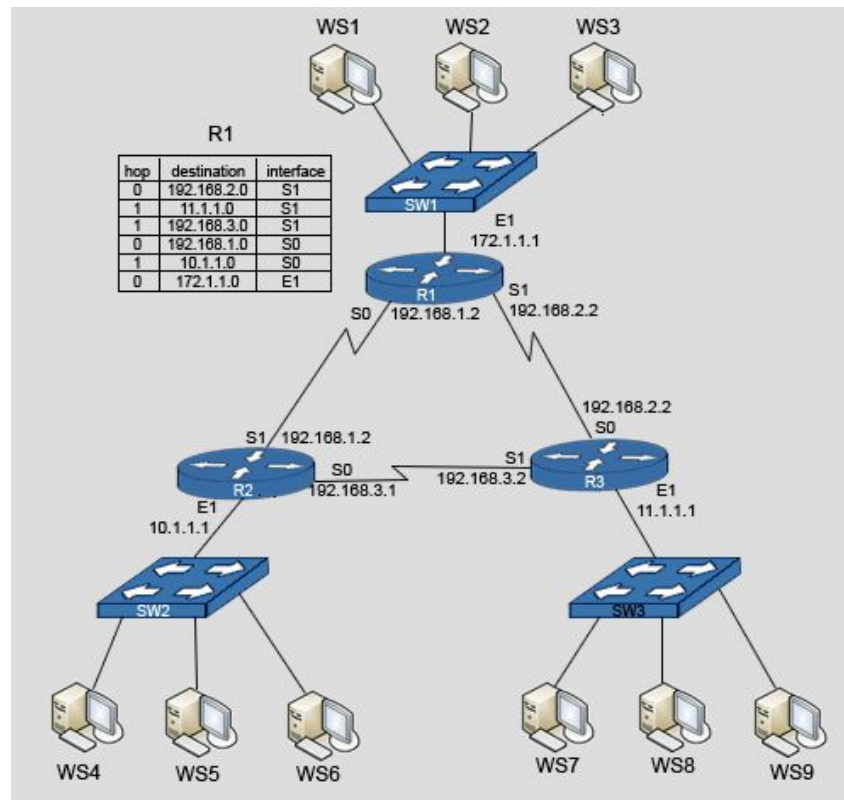
При работе дистанционно векторных протоколов маршрутизации может возникнуть нарушение целостности маршрутизации, когда из-за низкой скорости сходимости этих протоколов появляются некорректные записи о маршрутах.

До отказа какой либо сети все маршрутизаторы находились в состоянии *конвергенции* (на всех маршрутизаторах была полная и достоверная информация о всех сетях).

Затем произошел отказ сети 172.1.1.0.

Маршрутизатор *R1* был непосредственно подключен к сети 172.1.1.0 с метрикой 0 и исключил из таблицы маршрутизации маршрут через fa0/0. Для маршрутизаторов *R2* и *R3* путь к сети 172.1.1.0 лежит через маршрутизатор *R1* с числом хопов равным 1.

Но маршрутизаторы *R2* и *R3* еще не получили информации об отказе, и *R2* рассылает периодическое обновление. *R1* обнаруживает, что сеть 172.1.1.0 достижима через *R2* с метрикой 2 и заносит эту запись в свою таблицу маршрутизации. Затем *R1* отправляет обновление, в котором указывает, что сеть 172.1.1.0 достижима с метрикой 2. Маршрутизаторы *R2* и *R3* изменяют записи в своих таблицах маршрутизации о сети 172.1.1.0. Когда приходит время посылки обновления маршрутизатором *R3*, то он передает информацию о достижимости сети 172.1.1.0 с метрикой 3 для *R1* и *R2*. Таким образом, обновления таблиц маршрутизации будут продолжаться, и счетчик хопов будет расти (**счет до бесконечности**). Пакеты, предназначенные для сети 172.1.1.0, никогда не достигнут пункта назначения. Они будут перемещаться между маршрутизаторами (**петля маршрутизации**).

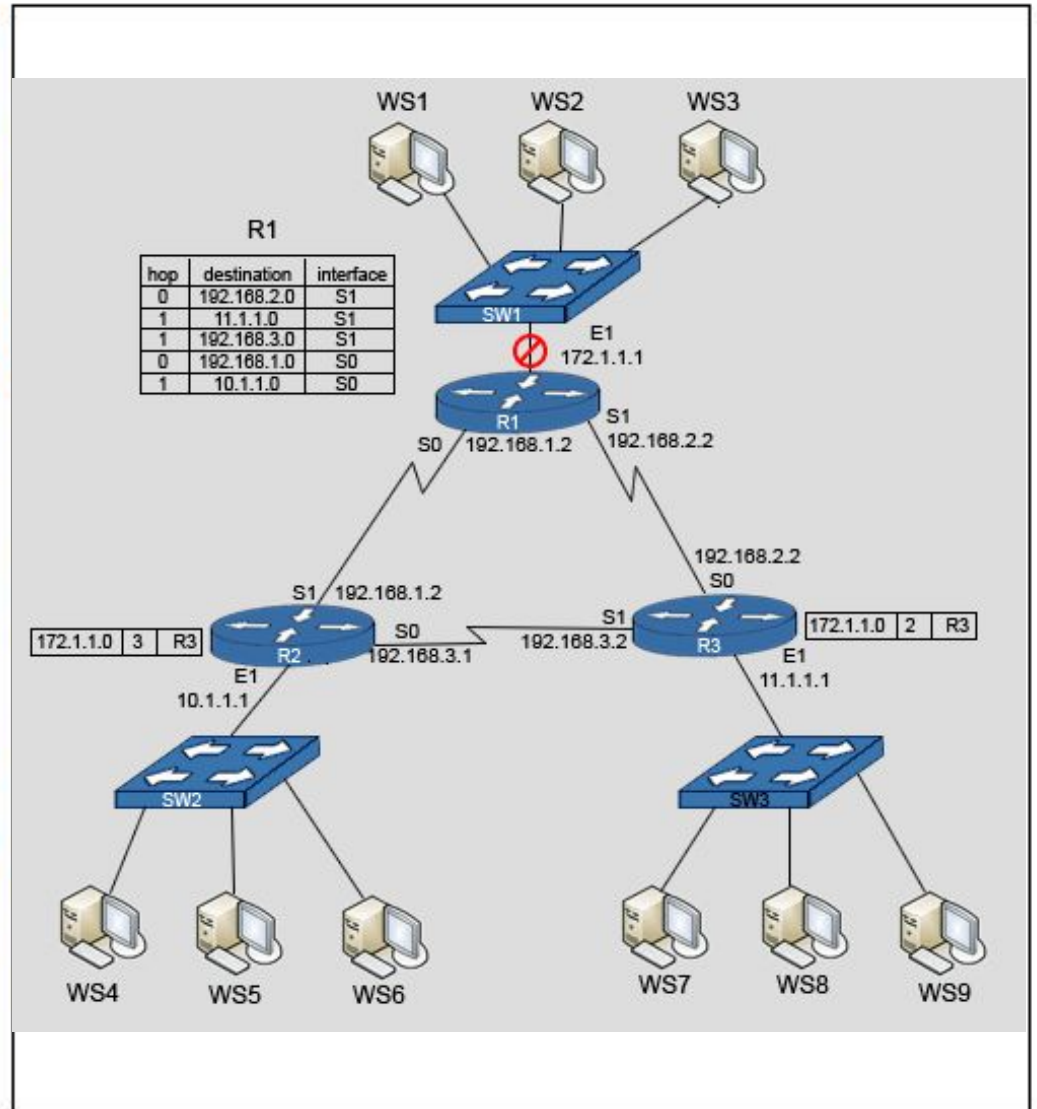


2.3.6. Петля маршрутизации.

Если из сети 11.1.1.0 будет послан пакет в сеть 172.1.1.0, то в соответствии с таблицей маршрутизации R3 он будет передан на R1, а в соответствии с таблицей маршрутизации R1 он будет передан обратно. Таким образом, пакет будет перемещаться между маршрутизаторами R1 и R3 в петле маршрутизации.

Если бы ip-пакеты не имели механизма самоуничтожения, то сеть перестала бы работать из-за перегрузки.

Механизм самоуничтожения ip-пакета – поле заголовка, определяющее время жизни (*time to live* – *TTL*). Когда пакет проходит на маршрутизатор, то *TTL* уменьшается на 1. Тот маршрутизатор, на котором значение *TTL* станет равным 0, уничтожит этот пакет.

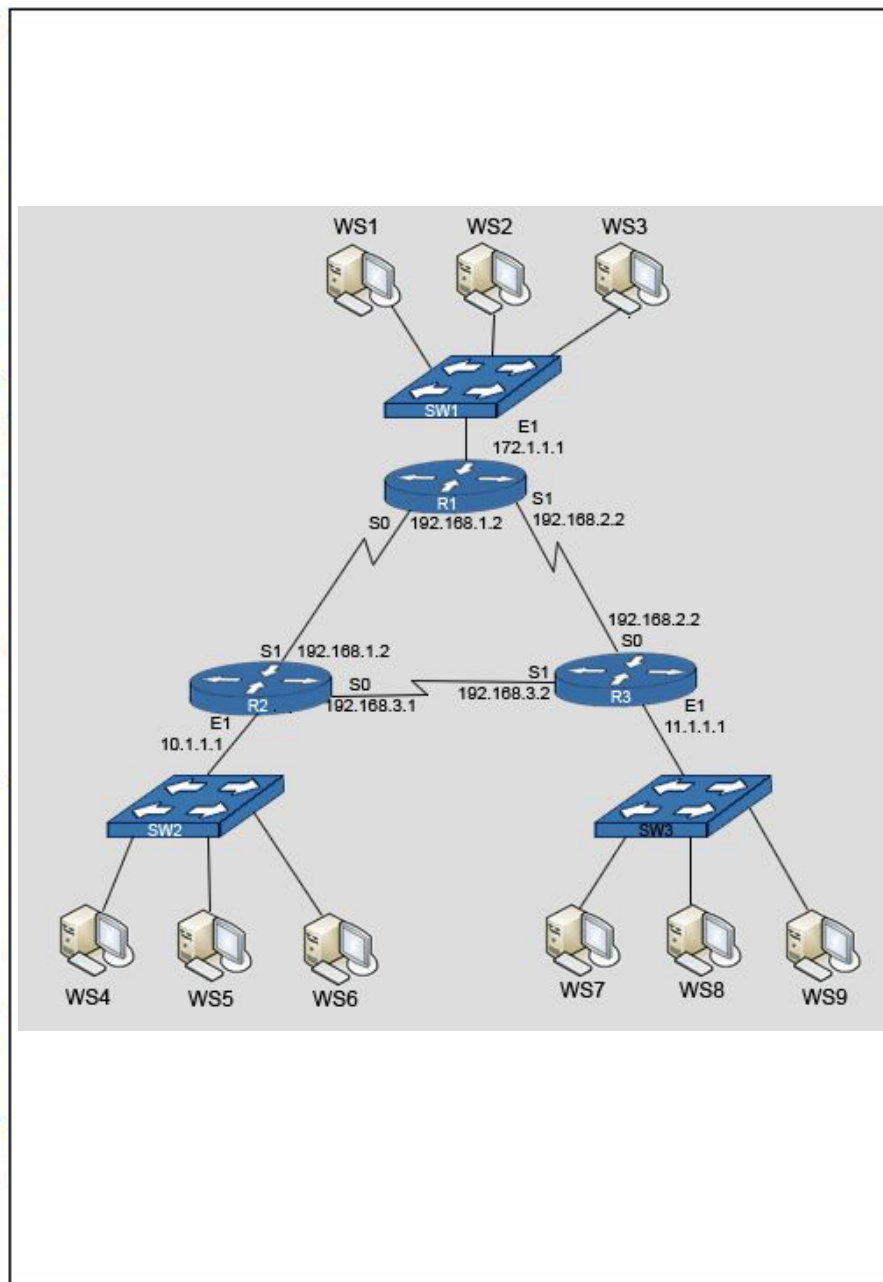


2.3.7. Механизмы удаления петель маршрутизации. Отравление маршрута.

Отравление маршрута (*Route Poisoning*) предназначено для удаления петель маршрутизации, которые вызваны противоречивыми обновлениями.

Если сеть становится недоступной, то маршрутизатор оставляет запись о маршруте в таблице, но присваивает ему максимальную метрику, которая свидетельствует о недостижимости данного маршрута. В текущем обновлении будет содержаться информация о том, что маршрут недостижим. Это сохраняет целостное состояние сети, пока другие маршрутизаторы постепенно проходят процесс корректного схождения при изменениях топологии.

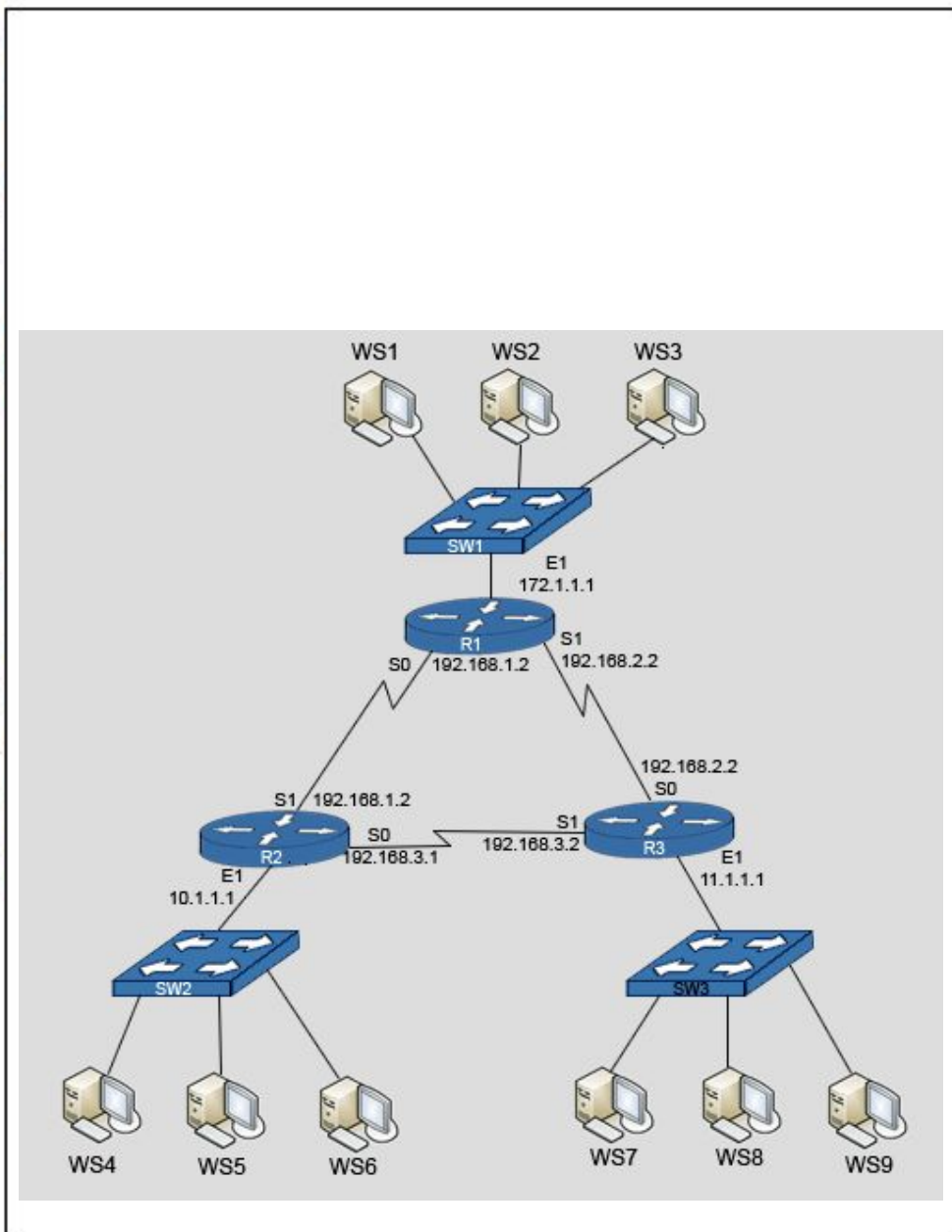
Пусть произошел отказ сети 172.1.1.0. Тогда маршрутизатор *R1* присвоит маршруту в эту сеть максимальную метрику и при передаче обновления, маршрутизаторы *R2* и *R3* получат достоверную информацию о том, что сеть 172.1.1.0 недостижима.



2.3.8. Недостаток механизма «Отравление маршрута».

Если до того как маршрутизатор пошлет обновление о недостижимости сети к нему придет обновление от соседнего маршрутизатора о том, что эта сеть достижима, то он воспримет это обновление, поскольку оно имеет лучшую метрику, и укажет новый маршрут, что опять приведет к возникновению петли маршрутизации и счету до максимального значения метрики.

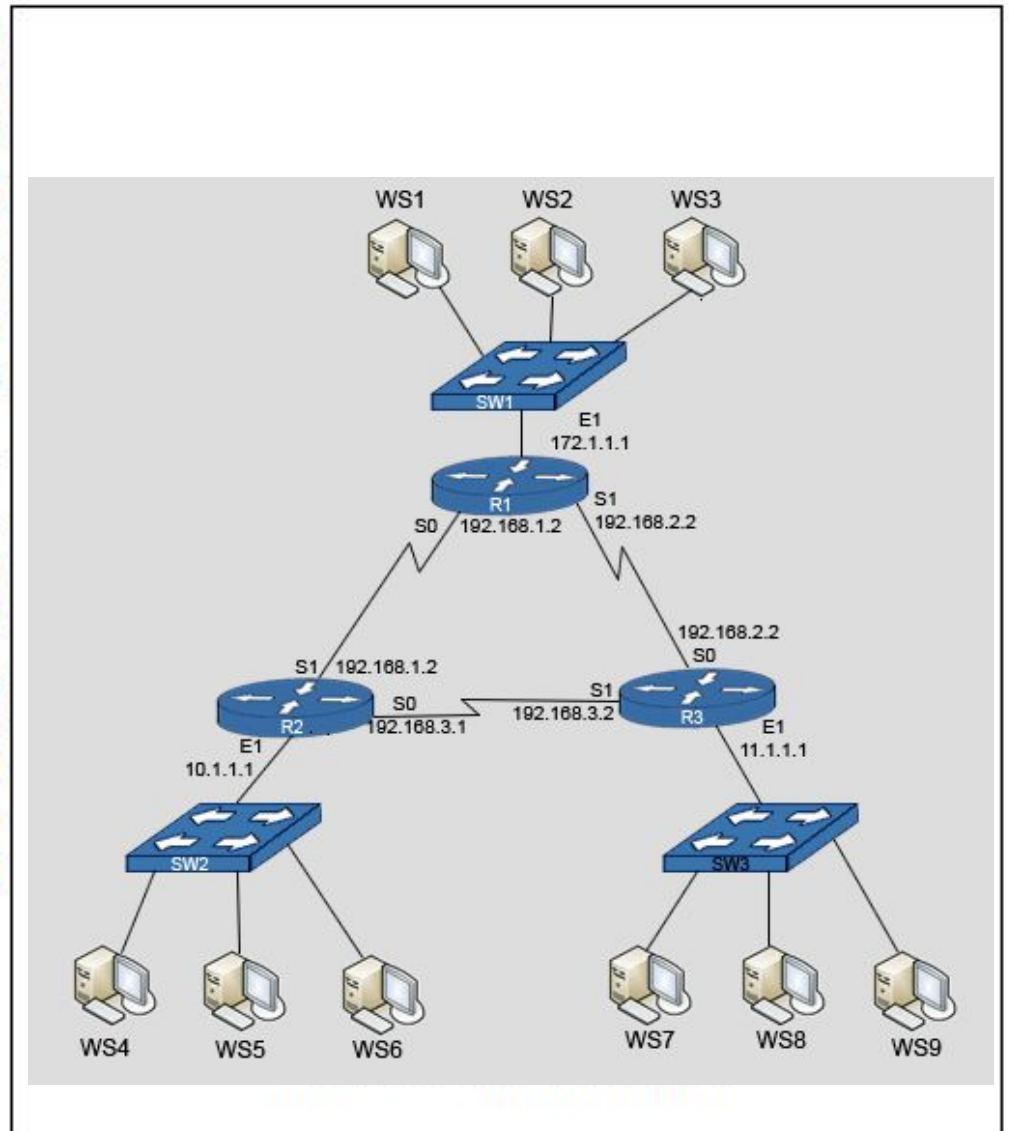
Если произошел отказ сети 172.1.1.0, то маршрутизатор *R1* присвоил маршруту в эту сеть максимальную метрику, но маршрутизатор *R2* послал обновление раньше, чем это сделал *R1*. В таком случае *R1* и *R3* получают ложную информацию о том, что сеть 172.1.1.0 достижима через *R2*.



2.3.9. Механизмы удаления петель маршрутизации. Расщепление горизонта.

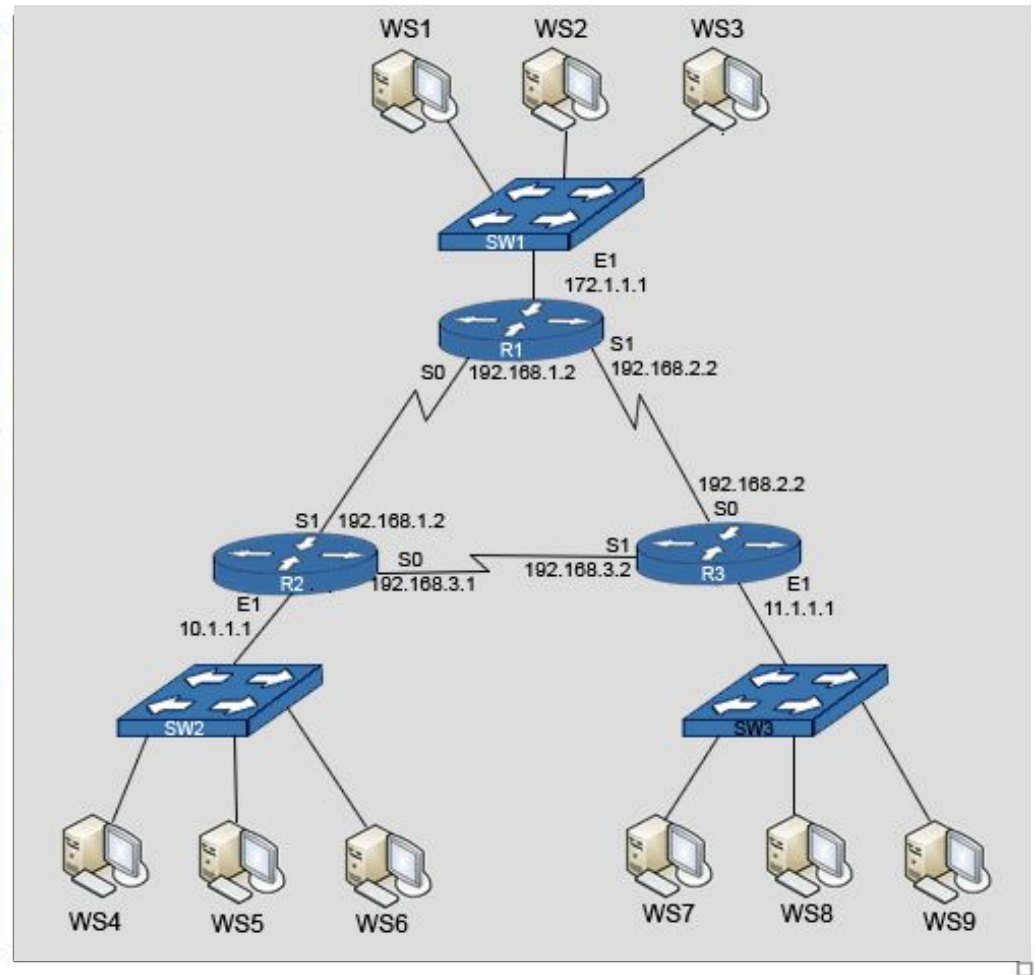
Один из способов удаления петли маршрутизации и ускорения сходимости - механизм расщепления горизонта. **Правило расщепления горизонта (split horizon) запрещает отправку информации о маршруте через тот интерфейс, с которого она пришла.**

Маршрутизатор *R1* получает информацию о сети 10.1.1.0 через интерфейс *S0/0/0*, а о сети 11.1.1.0 через интерфейс *S0/0/1*. Поэтому он никогда не будет передавать информацию о сети 10.1.1.0 через интерфейс *S0/0/0*, а о сети 11.1.1.0 через интерфейс *S0/0/1*. Аналогично обрабатывают информацию о маршрутах маршрутизаторы *R2* и *R3*.



2.3.10. Пример работы механизма «Расщепление горизонта».

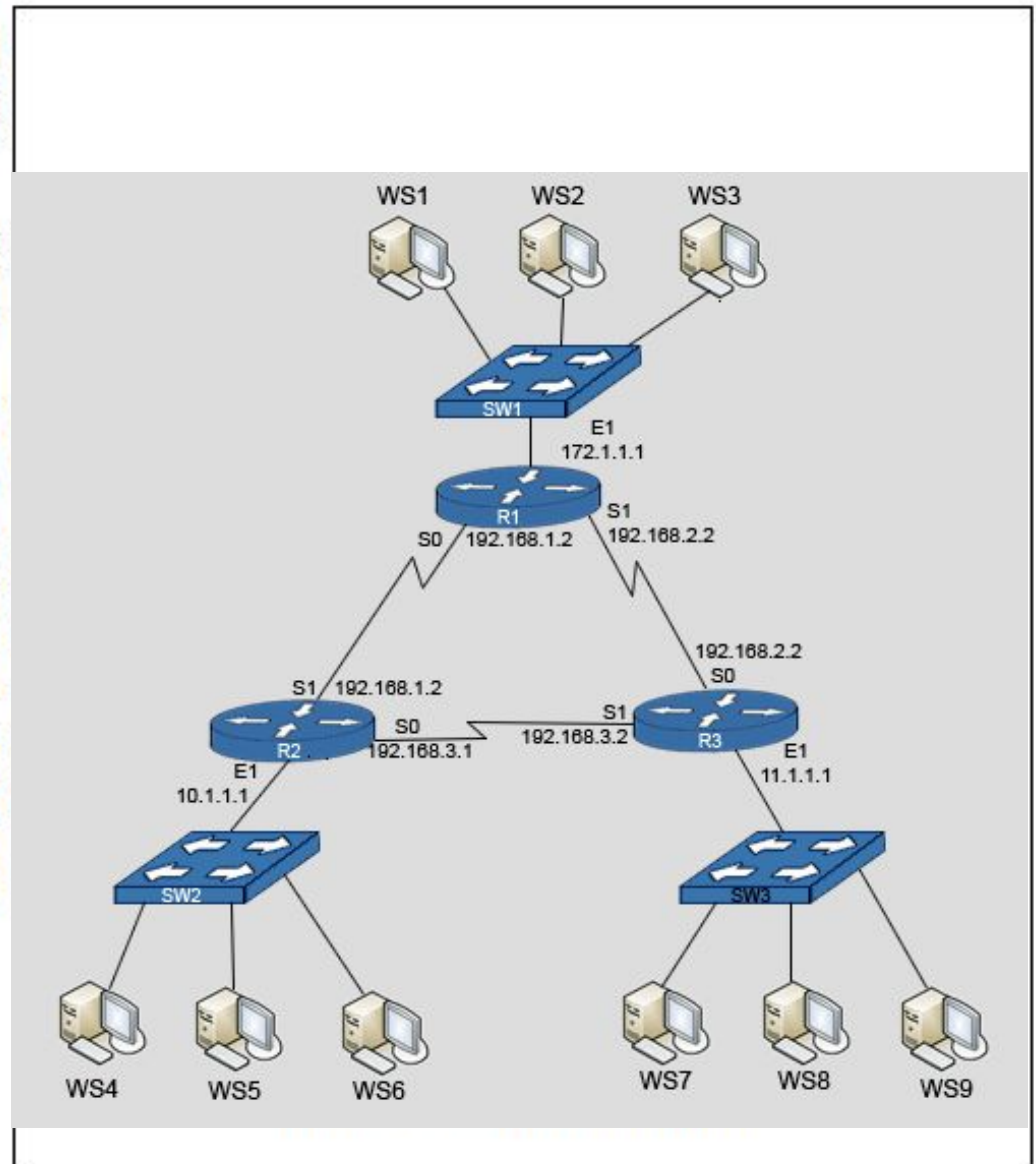
Когда на маршрутизаторе *R1* отключилась сеть *172.1.1.0*, то маршрутизаторы *R2* и *R3* не могут отправить обновление о маршруте *172.1.1.0* маршрутизатору *R1*. Если в нескольких последующих циклах обновлений от *R1* будет отсутствовать запись о маршруте в данную сеть, то маршрутизаторы *R2* и *R3* удалят этот маршрут из своих таблиц маршрутизации. Маршрутизатор *R1* не будет использовать маршрутизатор *R2* или *R3* при попытке достичь сети *172.1.1.0*.



2.3.11. Недостаток механизма «Расщепление горизонта».

Механизм расщепления горизонта может быть неэффективным в сетях с избыточными связями.

Когда пропало соединение с сетью 172.1.1.0, маршрутизатор *R1* удаляет информацию об этом маршруте из своих обновлений. Но, маршрутизатор *R2* имеет в своей таблице маршрутизации запись о сети 172.1.1.0 через *R1*, и в своем текущем обновлении отправит информацию об этой сети на *R3*. Т. к. обновление о сети 172.1.1.0 пришло с *R2*, то маршрутизатор *R3* объявит о нем в своем обновлении маршрутизатору *R1*. Таким образом, маршрутизатор *R1* будет считать, что сеть 172.1.1.0 доступна через *R3*, а это неверно.

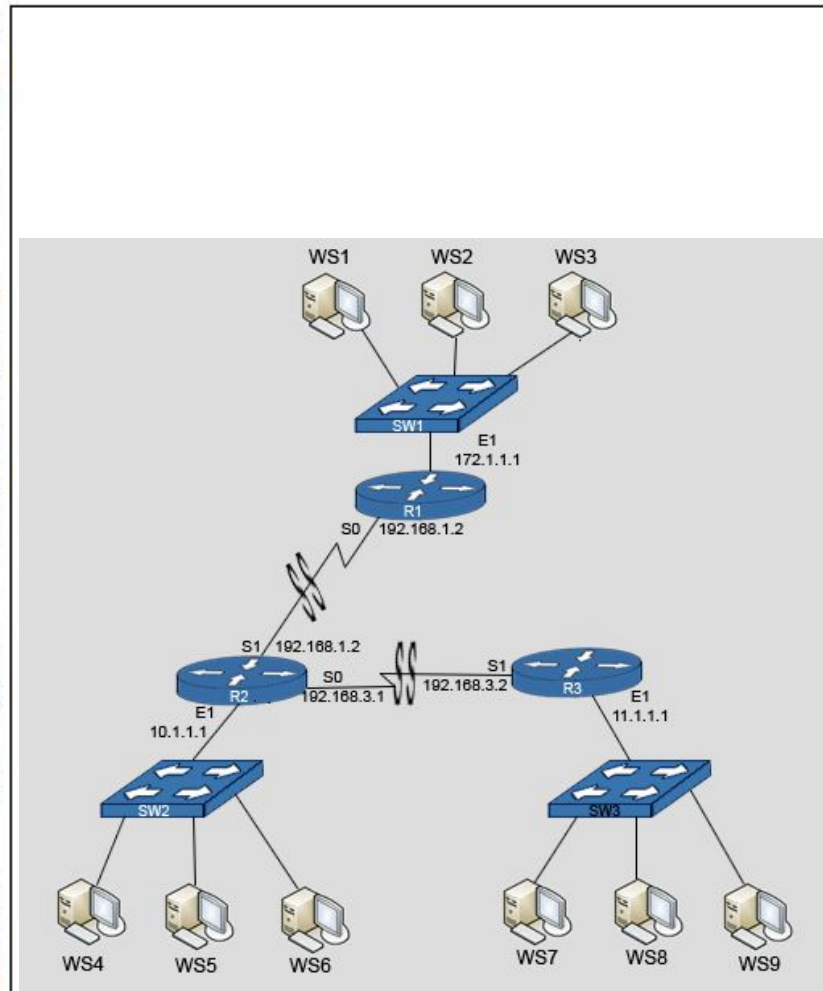


2.3.12. Совместное использование механизмов расщепления горизонта и отравления маршрута.

Для ускорения процесса сходимости реализуется процедура отправки назад отравленного маршрута (*Poison Reverse*).

Когда маршрутизатор *R2* видит, что в полученном обновлении, метрика в сеть 172.1.1.0 изменилась на бесконечную, он отправляет обновление, называемое отправкой назад отравленного маршрута, на маршрутизатор *R1*. Отправка назад отравленного маршрута устанавливает сеть 172.1.1.0 в состояние недоступности. Отправка назад отравленного маршрута — специфичный случай, когда не действует правило расщепленного горизонта. Это сделано для гарантии того, что маршрутизатор *R1* не воспримет некорректное обновление относительно сети 172.1.1.0.

Совместное использование механизмов расщепленного горизонта и отравления маршрута исключает возникновение петель маршрутизации в сетях, не имеющих кольцевой топологии. Но если сеть имеет избыточные связи, то возникновение петли маршрутизации возможно.

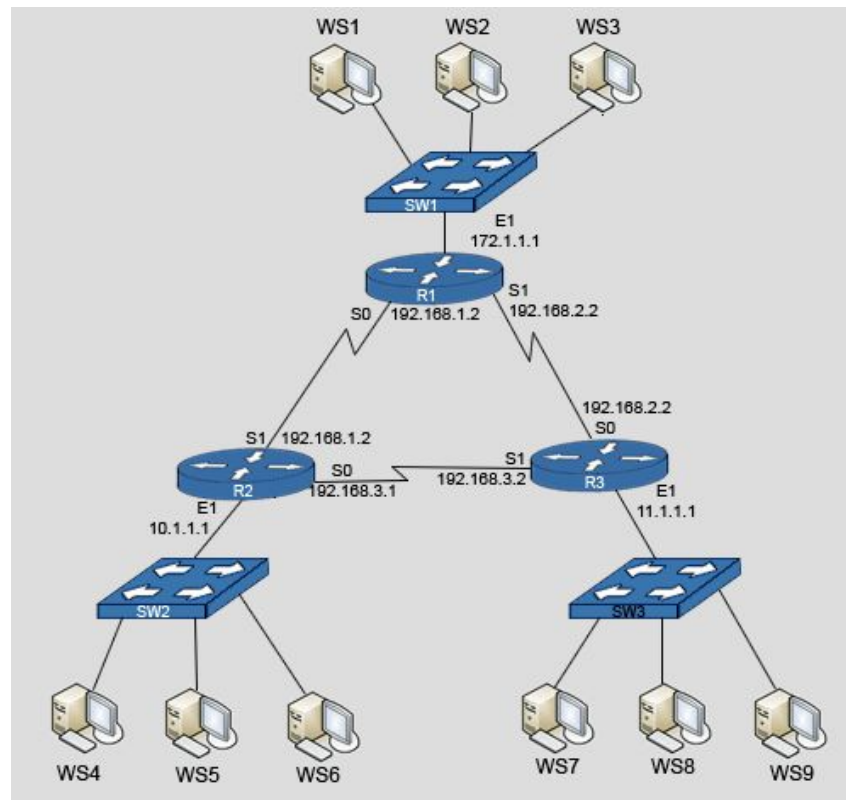


2.3.13. Механизмы удаления петель маршрутизации. Счетчик ожидания сходимости.

Счетчик ожидания сходимости (*Hold-Down Timer*) предотвращают использование информации из регулярного обновления, которая может содержать неверные данные относительно недоступного маршрута и запрещает маршрутизаторам выполнять изменения относительно маршрутов некоторый период времени.

Счетчики ожидания сходимости работают следующим образом:

- Когда маршрутизатор получает обновление от соседа, которое указывает, что ранее доступная сеть больше недоступна, то он помечает этот маршрут как “возможно недоступный” и запускает счетчик ожидания сходимости.
- В период работы счетчика ожидания сходимости, маршруты показываются в таблице маршрутизации как “возможно недоступные”. Маршрутизатор, тем не менее, будет пытаться отправлять пакеты в такие сети (возможно сеть имеет проблемы с соединением к маршрутизатору и то работает, то нет).
- Если, в любой момент до истечения срока действия счетчика, получено обновление от другого соседа с той же или худшей метрикой, обновление игнорируется. Игнорирование обновлений с такой же или худшей метрикой в процессе работы счетчика ожидания сходимости дает время, необходимое для того, чтобы информация об изменении топологии распространилась по всей сети.
- Если, в любой момент до истечения срока действия счетчика, приходит обновление от соседнего маршрутизатора с лучшей метрикой, чем в исходной записи для этой сети в таблице маршрутизации, то счетчик останавливается и маршрутизатор помечает сеть как “доступную”.

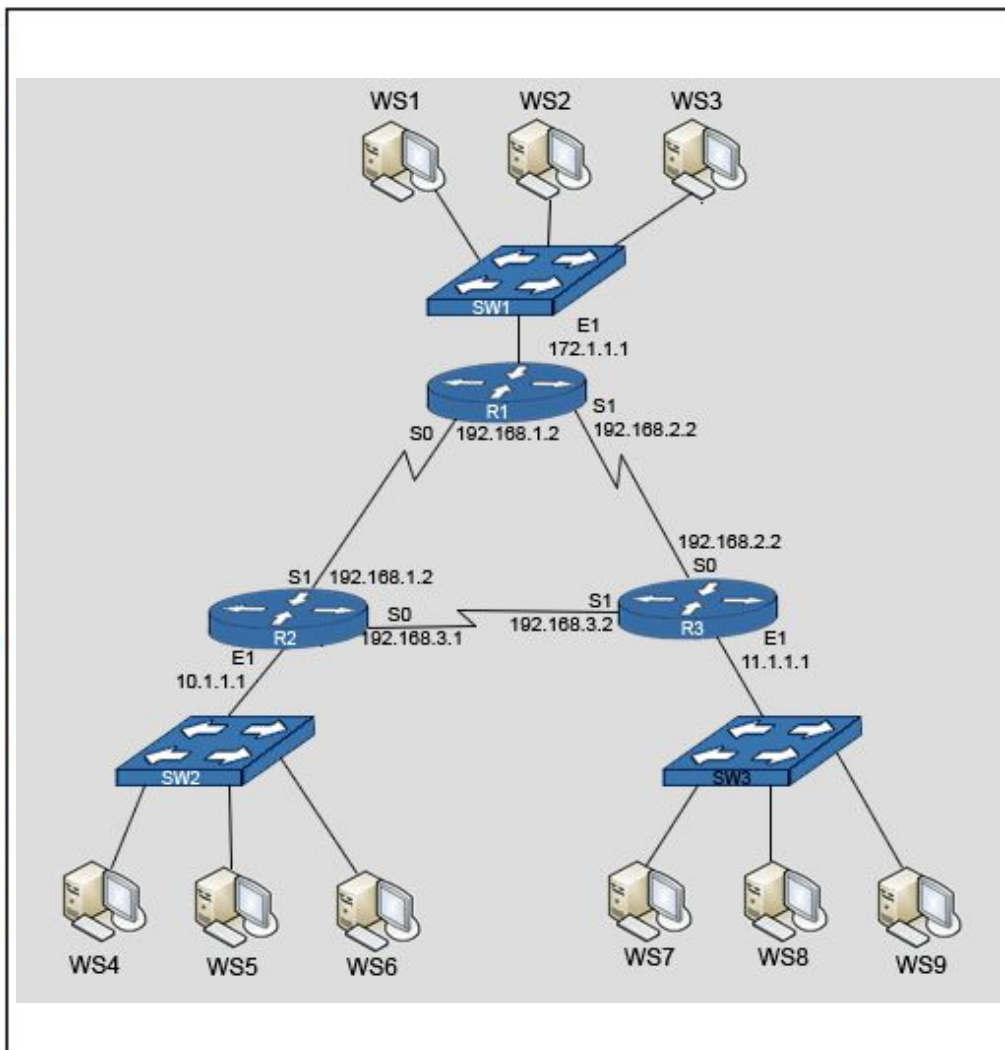


2.3.14. Механизмы удаления петель маршрутизации. Триггерные обновления.

Обычно, обновления таблицы маршрутизации отправляются соседям через регулярные интервалы времени. В результате возникает проблема большого времени сходимости.

Для уменьшения времени сходимости используются триггерные обновления (*Triggered Updates*) — обновления таблицы маршрутизации, отправляемые немедленно в ответ на изменения.

Маршрутизатор, который обнаружил изменения, немедленно отправляет обновление соседним маршрутизаторам, которые также генерируют триггерные обновления для оповещения об изменении своих соседей. Эта волна извещений будет распространяться через всю сеть.



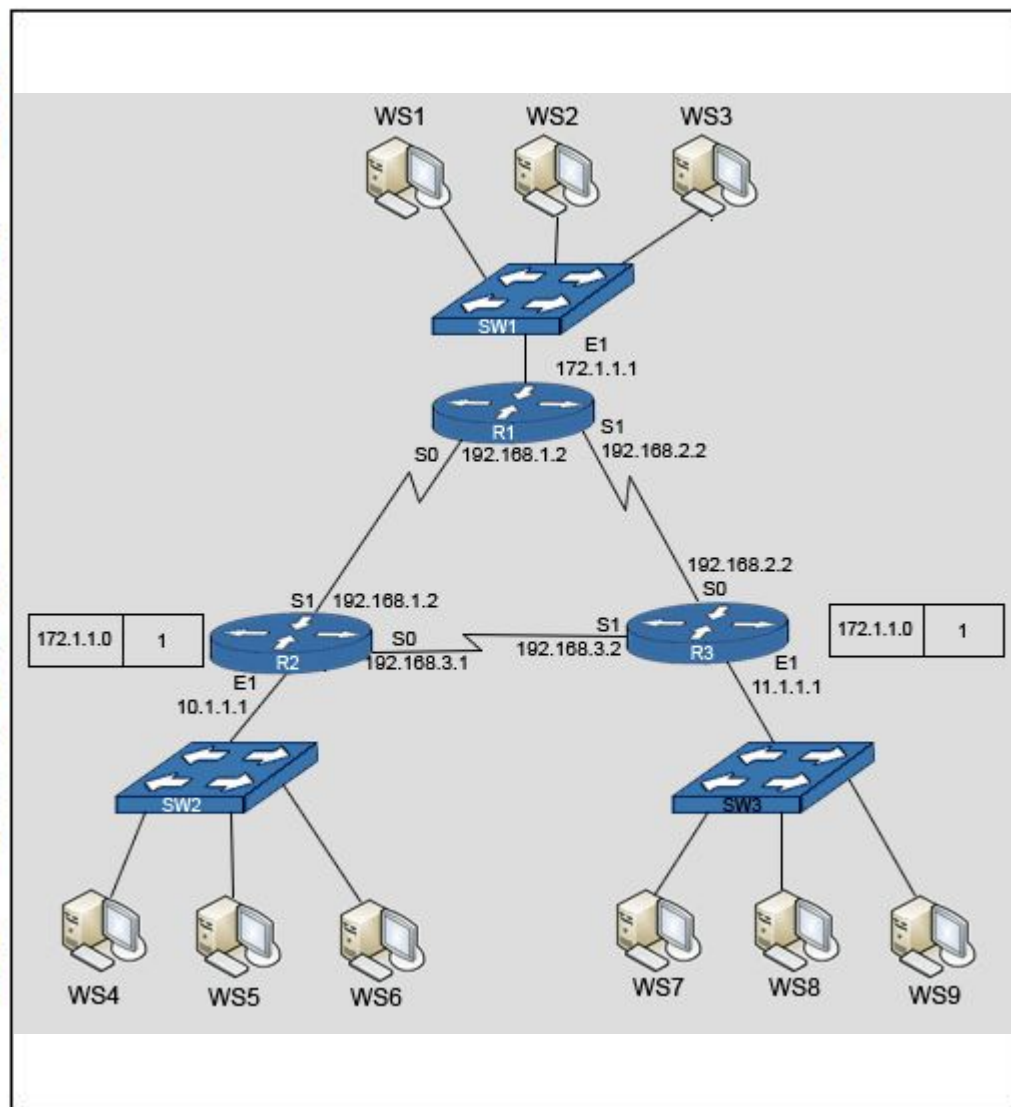
2.3.15. Совместное использование триггерных обновлений и счетчиков ожидания сходимости.

Триггерные обновления должны гарантированно прийти к каждому маршрутизатору. Однако существуют две проблемы:

- Пакеты, содержащие обновления могут быть отброшены или испорчены при прохождении некоторых соединений в сети.

- Триггерные обновления не происходят мгновенно. Существует вероятность того, что маршрутизатор, который еще не получил такое обновление будет отправлять регулярные обновления еще некоторое время, вызывая вставку неверного маршрута в таблицы соседей, которые это обновление уже получили.

Совместная работа триггерного обновления и счетчика ожидания сходимости решает эти проблемы.



2.3.16. Как работают механизмы удаления петель маршрутизации?

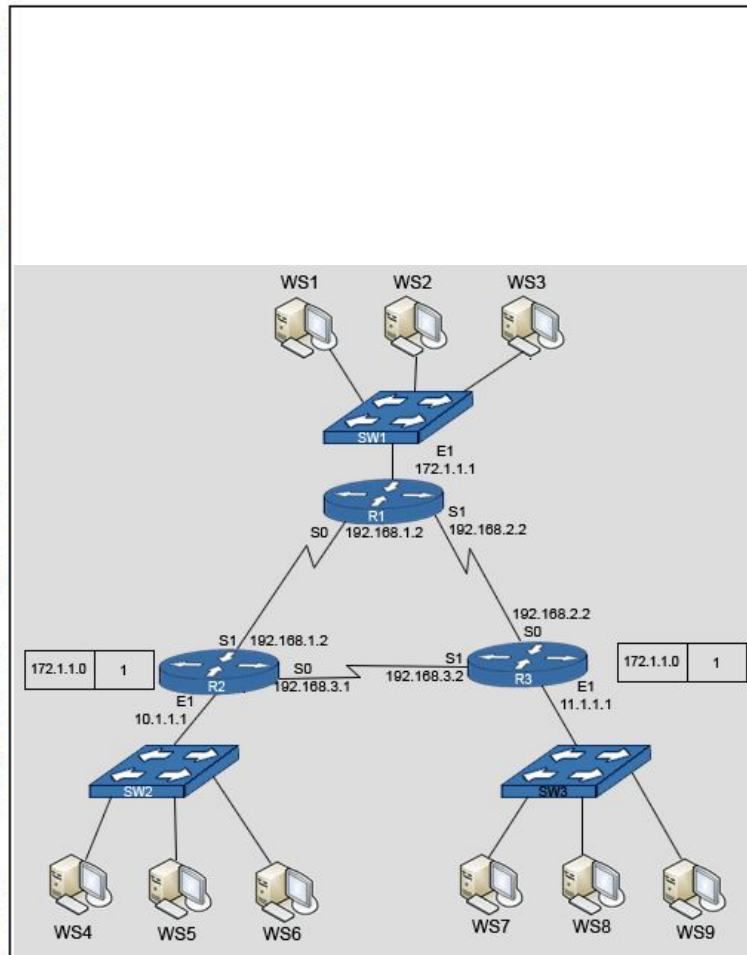
Когда пропало соединение с сетью 172.1.1.0, маршрутизатор *R1* удаляет маршрут в эту сеть и отправляет триггерное обновление на маршрутизаторы *R2* и *R3*, отправляя маршрут к сети 172.1.1.0 путем указания для нее бесконечной метрики.

Маршрутизаторы *R2* и *R3* получают триггерное обновление, запускают свои счетчики ожидания сходимости и помечают сеть 172.1.1.0 как возможно недоступную. Маршрутизаторы *R2* и *R3* отправляют триггерные обновления друг другу, указывая на возможную недоступность сети 172.1.1.0.

Маршрутизаторы *R2* и *R3* отправляют назад на маршрутизатор *R1* отравленный маршрут. Это обновление говорит маршрутизатору *R1*, что сеть 172.1.1.0 остается недоступной.

Маршрутизаторы *R2* и *R3* остаются в ожидании сходимости сети, пока не произойдет одно из следующих событий:

- Истечет время действия счетчика ожидания сходимости.
- Будет получено обновление, указывающее на новый маршрут с лучшей метрикой.
- По окончании счета маршрут удалится из таблицы маршрутизации.

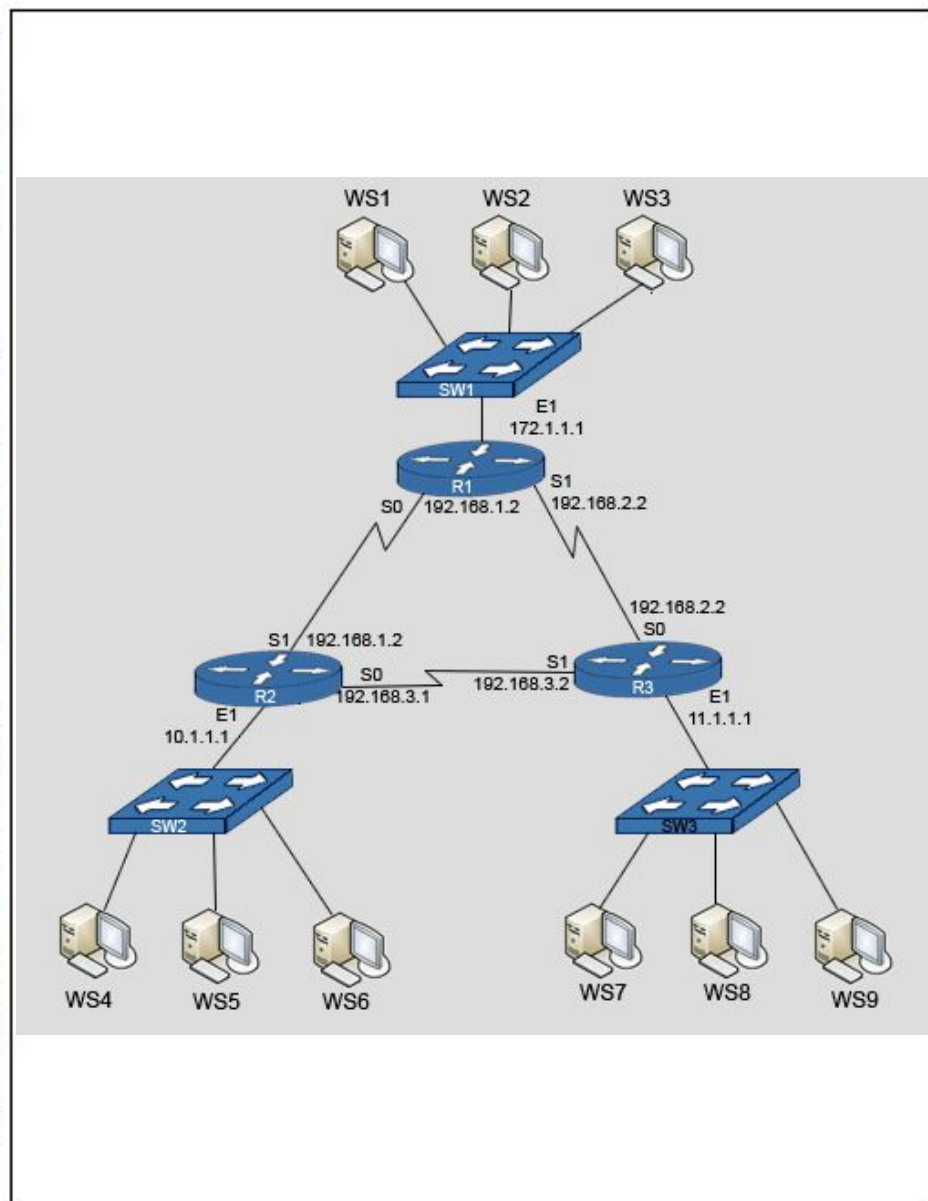


2.3.17. Передача данных при изменении топологии.

Пока работают счетчики ожидания сходимости, маршрутизаторы *R2* и *R3* присваивают сети статус - «возможно недоступная» и пытаются отправлять пакеты в сеть 172.1.1.0.

Если в это время из сети 10.1.1.0 будет попытка передачи данных в сеть 172.1.1.0, то пакеты достигнут маршрутизатора *R1*. Однако маршрутизатор *R1* не имеет маршрута в сеть 172.1.1.0, поэтому он отбросит пакеты и отправит назад сообщение *Internet Control Message Protocol (ICMP)* "network unreachable".

Когда сеть 172.1.1.0 снова станет доступной, маршрутизатор *R1* отправит триггерное обновление на маршрутизаторы *R2* и *R3*, которое сообщит им, что соединение с сетью вновь доступно. После истечения времени работы счетчиков ожидания сходимости, маршрутизаторы *R2* и *R3* изменят состояние маршрута в сеть 172.1.1.0 с «возможно недоступна» на «доступна».



2.3.18. Конфигурирование динамической маршрутизации.

Последовательность действий, необходимых для включения динамического протокола маршрутизации на маршрутизаторе *Cisco*.

Для включения динамического протокола маршрутизации необходимо выполнить следующие действия:

1. Выберите протокол маршрутизации: *RIP*, *EIGRP* или *OSPF*.
2. Назначьте номер IP-сети без определения значения подсети (за исключением *OSPF*).

Для включения динамической маршрутизации на маршрутизаторе *Cisco* используется команда режима глобальной конфигурации ***router***.

Обязательна команда ***network***, которая определяет, какие интерфейсы будут участвовать в отправке и получении обновлений маршрутной информации.

Проверка настройки протокола маршрутизации выполняется с помощью группы команд ***show*** и ***debug***.

<i>router</i>	Команда глобального режима конфигурации
<i>Protocol</i>	<i>RIP</i> , <i>OSPF</i> или <i>EIGRP</i>
<i>Keyword</i>	Например, <i>autonomous system</i> , когда эти протоколы требуют для настройки номер автономной системы (<i>IGRP</i> и <i>EIGRP</i>); также может определять локальный идентификатор процесса, который используется протоколом <i>OSPF</i> .
<i>network</i>	Команда подрежима конфигурации протокола маршрутизации.
<i>network - number</i>	Определяет, непосредственно подключенную сеть, через которую должен работать протокол маршрутизации.
<i>Show ip route [ip-address [mask]] [protocol [process-id]]</i>	Показывает всю таблицу маршрутизации или часть маршрутов, если введены дополнительные параметры.
<i>Show ip protocols</i>	Показывает протоколы маршрутизации и текущие значения их таймеров.
<i>Debug ip [protocol]</i>	Показывает записи по каждому обновлению.

Рис 3-32