

# Маршрутизация в информационных сетях

# Коммутация и маршрутизация

**Коммутация** - это организация и поддержка необходимого числа логических каналов между абонентами и узлами за счет мультиплексирования (уплотнения), разделения физического тракта передачи - пространственное разделение (каждому абоненту предоставляется собственный физический подканал на все время работы).

**Маршрутизация** — процесс определения маршрута следования информации в сетях связи.

Маршрутизация означает передвижение информации от источника к пункту назначения через объединенную сеть. При этом, как правило, на пути встречается по крайней мере один узел.

# Компоненты маршрутизации

- **Определение маршрута**

Определение маршрута может базироваться на различных показателях (величинах, результирующих из алгоритмических вычислений по отдельной переменной - например, длина маршрута) или комбинациях показателей. Программные реализации алгоритмов маршрутизации высчитывают показатели маршрута для определения оптимальных маршрутов к пункту назначения.

Для облегчения процесса определения маршрута, алгоритмы маршрутизации инициализируют и поддерживают таблицы маршрутизации, в которых содержится маршрутная информация. Маршрутная информация изменяется в зависимости от используемого алгоритма маршрутизации.

- **Транспортировка пакетов через объединенную сеть (Коммутация)**

Алгоритмы коммутации сравнительно просты и в основном одинаковы для большинства протоколов маршрутизации. В большинстве случаев главная вычислительная машина определяет необходимость отправки пакета в другую главную вычислительную машину. Получив определенным способом адрес роутера, главная вычислительная машина-источник отправляет пакет, адресованный специально в **физический адрес роутера** (уровень MAC), однако с адресом протокола (сетевой уровень) компьютера пункта назначения.

После проверки адреса протокола пункта назначения пакета роутер определяет, знает он или нет, как передать этот пакет к следующему роутеру. Во втором случае (когда роутер не знает, как переслать пакет) пакет, как правило, игнорируется. В первом случае роутер отправляет пакет к следующей роутеру путем замены физического адреса пункта назначения на физический адрес следующего роутера и последующей передачи пакета.

# Алгоритмы маршрутизации

## Цели разработки алгоритмов маршрутизации :

### ▶ *Оптимальность*

Оптимальность, вероятно, является самой общей целью разработки. Она характеризует способность алгоритма маршрутизации выбирать "наилучший" маршрут. Наилучший маршрут зависит от показателей и от "веса" этих показателей, используемых при проведении расчета. Например, алгоритм маршрутизации мог бы использовать несколько пересылок с определенной задержкой, но при расчете "вес" задержки может быть им оценен как очень значительный. Естественно, что протоколы маршрутизации должны строго определять свои алгоритмы расчета показателей.

### ▶ *Простота и низкие производительные затраты*

Алгоритмы маршрутизации разрабатываются как можно более простыми. Другими словами, алгоритм маршрутизации должен эффективно обеспечивать свои функциональные возможности, с минимальными затратами программного обеспечения и коэффициентом использования.

### ▶ *Живучесть и стабильность*

Алгоритмы маршрутизации должны обладать живучестью. Другими словами, они должны четко функционировать в случае неординарных или непредвиденных обстоятельств, таких как отказы аппаратуры, условия высокой нагрузки и некорректные реализации.

### ▶ *Быстрая сходимоть*

Алгоритмы маршрутизации должны быстро сходиться. Сходимость - это процесс соглашения между всеми роутерами по оптимальным маршрутам. Когда какое-нибудь событие в сети приводит к тому, что маршруты или отвергаются, или становятся доступными, роутеры рассылают сообщения об обновлении маршрутизации. Алгоритмы маршрутизации, которые сходятся медленно, могут привести к образованию петель маршрутизации или выходам из строя сети.

### ▶ *Гибкость*

Алгоритмы маршрутизации должны быть также гибкими. Другими словами, алгоритмы маршрутизации должны быстро и точно адаптироваться к разнообразным обстоятельствам в сети. Например, предположим, что сегмент сети отвергнут. Многие алгоритмы маршрутизации, после того как они узнают об этой проблеме, быстро выбирают следующий наилучший путь для всех маршрутов, которые обычно используют этот сегмент. Алгоритмы маршрутизации могут быть запрограммированы таким образом, чтобы они могли адаптироваться к изменениям полосы пропускания сети, размеров очереди к роутеру, величины задержки сети и других переменных.

# Типы алгоритмов

- **Статические алгоритмы** . Распределение *статических* таблиц маршрутизации устанавливается администратором сети до начала маршрутизации. Оно не меняется, если только администратор сети не изменит его. Алгоритмы, использующие статические маршруты, просты для разработки и хорошо работают в окружениях, где трафик сети относительно предсказуем, а схема сети относительно проста.
- **Динамические алгоритмы** маршрутизации подстраиваются к изменяющимся обстоятельствам сети в масштабе реального времени. Они выполняют это путем анализа поступающих сообщений об обновлении маршрутизации. Если в сообщении указывается, что имело место изменение сети, программы маршрутизации пересчитывают маршруты и рассылают новые сообщения о корректировке маршрутизации.
- **Одномаршрутные или многомаршрутные алгоритмы**. Некоторые сложные протоколы маршрутизации обеспечивают множество маршрутов к одному и тому же пункту назначения. Преимущества многомаршрутных алгоритмов очевидны - они могут обеспечить значительно большую пропускную способность и надежность.
- **Внутридоменные или междоменные алгоритмы**

Некоторые алгоритмы маршрутизации действуют только в пределах доменов; другие - как в пределах доменов, так и между ними. Природа этих двух типов алгоритмов различная. Поэтому понятно, что оптимальный алгоритм внутридоменной маршрутизации не обязательно будет оптимальным алгоритмом междоменной маршрутизации.

•

# Типы алгоритмов

Некоторые алгоритмы маршрутизации оперируют в плоском пространстве, в то время как другие используют иерархии маршрутизации. В **одноуровневой системе маршрутизации** все роутеры равны по отношению друг к другу. В иерархической системе маршрутизации некоторые роутеры формируют то, что составляет основу маршрутизации. Пакеты из небазовых роутеров перемещаются к базовым роутерам и пропускаются через них до тех пор, пока не достигнут общей области пункта назначения. Начиная с этого момента, они перемещаются от последнего базового роутера через один или несколько небазовых роутеров до конечного пункта назначения.

Основным преимуществом **иерархической маршрутизации** является то, что она имитирует организацию большинства компаний и следовательно, очень хорошо поддерживает их схемы трафика. Большая часть сетевой связи имеет место в пределах групп небольших компаний (доменов). Внутридоменным роутерам необходимо знать только о других роутерах в пределах своего домена, поэтому их алгоритмы маршрутизации могут быть упрощенными. Соответственно может быть уменьшен и трафик обновления маршрутизации, зависящий от используемого алгоритма маршрутизации.

# Типы алгоритмов

## *Алгоритмы состояния канала или вектора расстояния*

Алгоритмы **состояния канала** направляют потоки маршрутной информации во все узлы объединенной сети. Однако каждый роутер посылает только ту часть маршрутной таблицы, которая описывает состояние его собственных каналов.

**Преимущества:** быстрая сходимость, не склонны к образованию петель маршрутизации.

Алгоритмы **вектора расстояния** ( известные также как алгоритмы Бэлмана-Форда) требуют от каждого роутера посылки всей или части своей маршрутной таблицы, но только своим соседям. Алгоритмы состояния каналов фактически направляют небольшие корректировки по всем направлениям, в то время как алгоритмы вектора расстояний отсылают более крупные корректировки только в соседние роутеры.

**Преимущества:** просты, менее ресурсоемкие.

Вследствие этого, реализация и поддержка алгоритмов состояния канала может быть более дорогостоящей. Несмотря на их различия, оба типа алгоритмов хорошо функционируют при самых различных обстоятельствах.

# Показатели алгоритмов (метрики)

## Длина маршрута

Длина маршрута является наиболее общим показателем маршрутизации. Некоторые протоколы маршрутизации позволяют администраторам сети назначать произвольные цены на каждый канал сети. В этом случае длиной тракта является сумма расходов, связанных с каждым каналом, который был traversирован. Часто используют "количество переходов", т.е. показатель, характеризующий число проходов, которые пакет должен совершить на пути от источника до пункта назначения через роутеры.

## Надежность

Надежность, в контексте алгоритмов маршрутизации, относится к надежности каждого канала сети (обычно описываемой в терминах соотношения бит/ошибка).

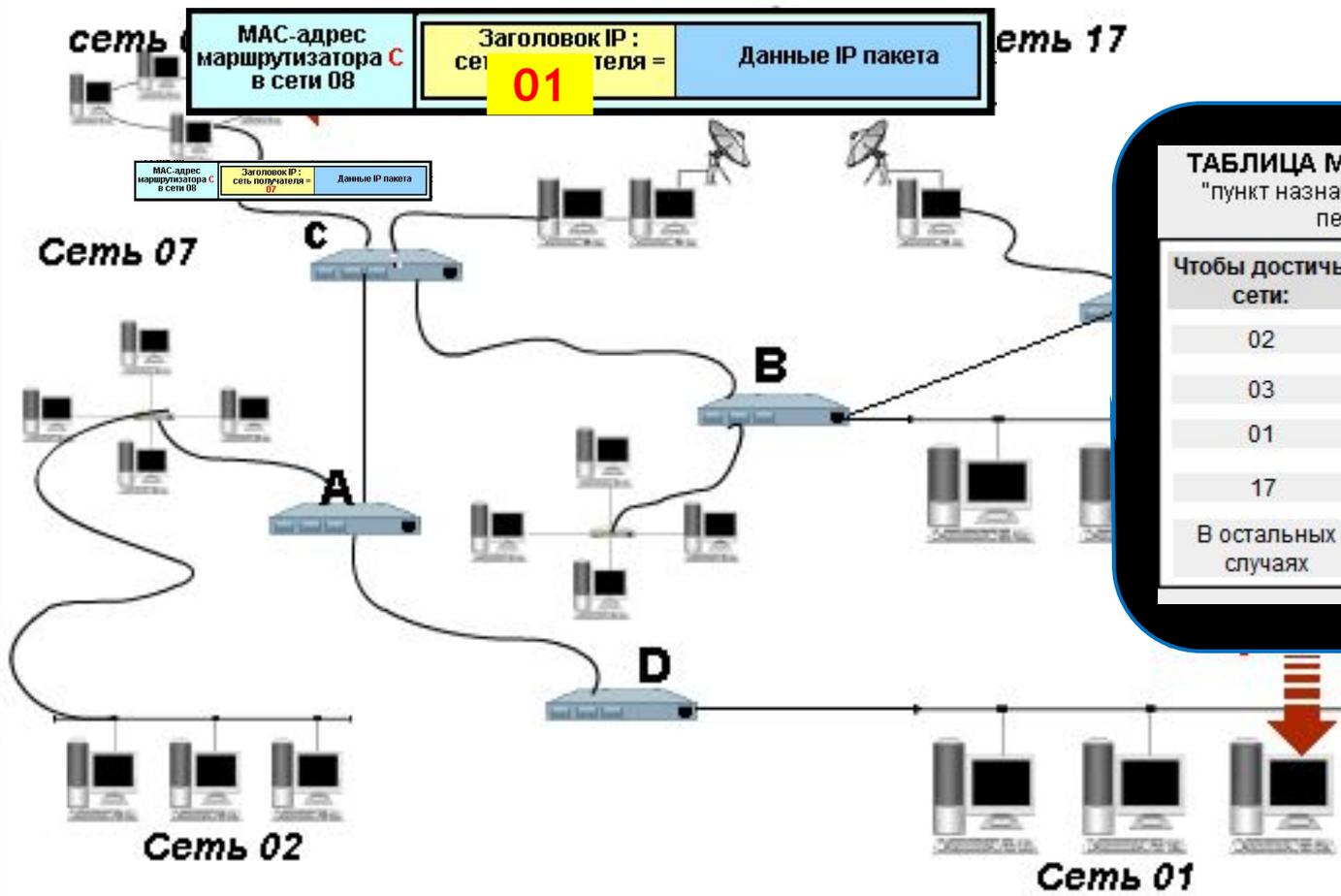
## Задержка

Под задержкой маршрутизации обычно понимают отрезок времени, необходимый для передвижения пакета от источника до пункта назначения через объединенную сеть.

## Полоса пропускания

Полоса пропускания относится к имеющейся мощности трафика какого-либо канала.

# Принцип работы





# Таблицы маршрутизации



# Структура таблицы маршрутизации Windows

- ▶ **адрес сети** или **узла назначения**, либо указание, что маршрут является *маршрутом по умолчанию*
- ▶ **маску сети назначения** (маска /32 (255.255.255.255) позволяет указать единственный узел сети)
- ▶ **шлюз**, обозначающий адрес маршрутизатора в сети, на который необходимо отправить пакет, следующий до указанного адреса назначения
- ▶ **интерфейс**
- ▶ **метрику** — числовой показатель, задающий предпочтительность маршрута. Чем меньше число, тем более предпочтителен маршрут.

# Заполнение таблиц маршрутизации

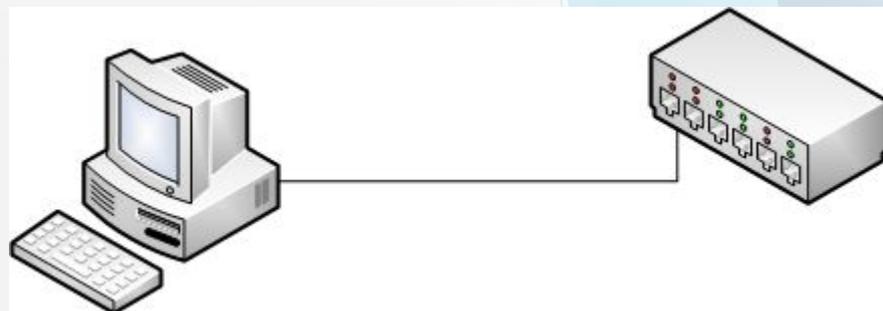
## Источники записей в таблице маршрутизации

- ▶ Записи о непосредственно подключенных сетях и маршрутизаторах по умолчанию,
- ▶ Адреса особого назначения,
- ▶ Записи, введенные принудительно администратором сети,
- ▶ Записи, полученные в результате работы протоколов маршрутизации.

# Типы записей в таблице маршрутизации:

- ▶ маршрут до компьютера,
- ▶ маршрут до сети,
- ▶ маршрут по умолчанию,
- ▶ маршрут на собственный обратный адрес,
- ▶ маршрут для широковещательных пакетов,
- ▶ маршрут для многоадресной рассылки.

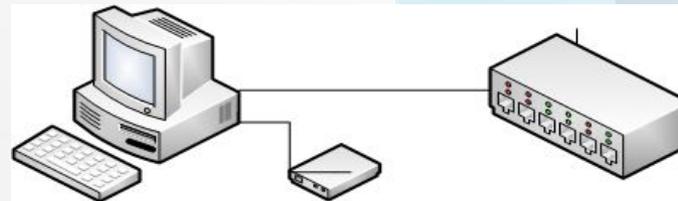
# Пример таблицы маршрутизации



Network Address	Netmask	Gateway Address	Interface	Metric
0.0.0.0	0.0.0.0	192.168.0.1	192.168.0.4	1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
192.168.0.0	255.255.0.0	192.168.0.4	192.168.0.4	2
192.168.0.4	255.255.255.25 5	127.0.0.1	127.0.0.1	1
192.168.0.255	255.255.255.25 5	192.168.0.4	192.168.0.4	1
224.0.0.0	224.0.0.0	192.168.0.4	192.168.0.4	1
255.255.255.255	255.255.255.25 5	192.168.0.4	192.168.0.4	1

Сетевая карта с IP-адресом  
192.168.0.4.

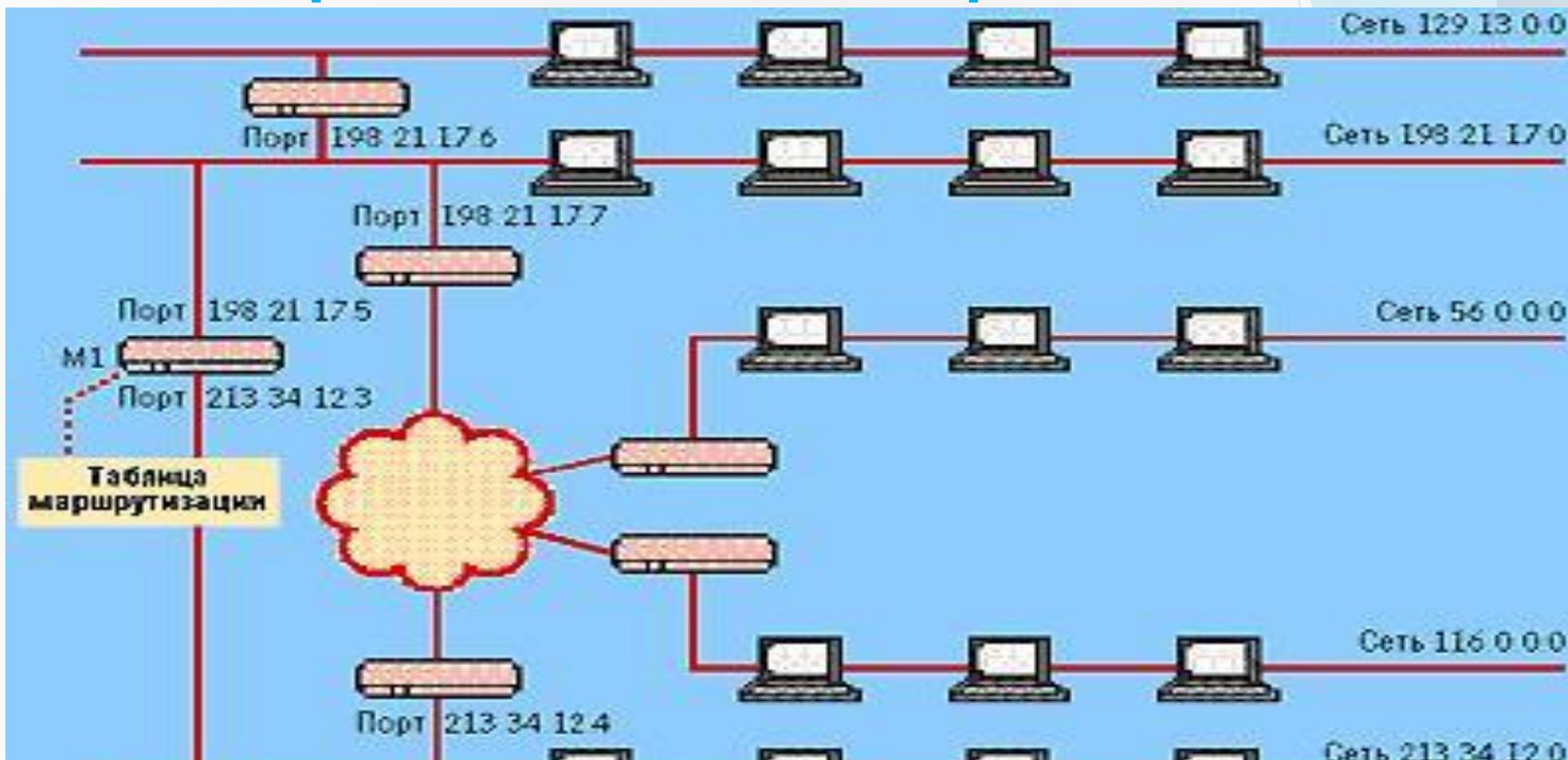
## Пример 2.



Network Address	Netmask	Gateway Address	Interface	Metric
0.0.0.0	0.0.0.0	203.96.10.254	03.96.10.51	1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
192.168.0.0	255.255.0.0	192.168.0.4	192.168.0.4	2
192.168.0.4	255.255.255.255	127.0.0.1	127.0.0.1	1
192.168.0.255	255.255.255.255	192.168.0.4	192.168.0.4	1
203.96.10.0	255.255.255.0	203.96.10.51	203.96.10.51	1
203.96.10.51	255.255.255.255	127.0.0.1	127.0.0.1	1
203.96.10.255	255.255.255.255	203.96.10.51	203.96.10.51	1
224.0.0.0	224.0.0.0	203.96.10.51	203.96.10.5	1
224.0.0.0	224.0.0.0	192.168.0.4	192.168.0.4	1
255.255.255.255	255.255.255.255	192.168.0.4	192.168.0.4	1

Сетевая карта с IP-адресом 192.168.0.4  
Модем с IP-адресом 203.96.10.51

# ТАБЛИЦА МАРШРУТИЗАЦИИ В СЕТЯХ IP



	Network address	Netmask	Gateway Address	Interface	Metric
	Network address	Netmask	Gateway Address	Interface	Metric
1	Network address	Netmask	Gateway Address	Interface	Metric
2	224.0.0.0	224.0.0.0	198.21.17.6	198.21.17.6	1
2	224.0.0.0	224.0.0.0	213.34.12.3	213.34.12.3	1
	213.34.12.0	255.255.255.0	213.34.12.3	213.34.12.3	1

# Таблица маршрутизации Windows

Network address	Netmask	Gateway Address	Interface	Metric
0.0.0.0	0.0.0.0	198.21.17.7	198.21.17.5	1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
56.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	15
116.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	13
129.13.0.0	255.255.0.0	198.21.17.6	198.21.17.5	2
198.21.17.0	255.255.255.0	198.21.17.5	198.21.17.5	1
198.21.17.5	255.255.255.255	127.0.0.1	127.0.0.1	1
198.21.17.255	255.255.255.255	198.21.17.5	198.21.17.5	1
213.34.12.0	255.255.255.0	213.34.12.3	213.34.12.3	1
213.34.12.3	255.255.255.255	127.0.0.1	127.0.0.1	1
213.34.12.255	255.255.255.255	213.34.12.3	213.34.12.3	1
224.0.0.0	224.0.0.0	198.21.17.6	198.21.17.6	1
224.0.0.0	224.0.0.0	213.34.12.3	213.34.12.3	1
255.255.255.255	255.255.255.255	198.21.17.6	198.21.17.6	1

# Настройка таблиц маршрутизации route

Обработка таблиц сетевых маршрутов.

**ROUTE [-f] [-p] [-4|-6] command [destination]  
[MASK netmask] [gateway] [METRIC metric] [IF interface]**

-f Очистка таблиц маршрутов от записей всех шлюзов.

-p При использовании с командой ADD задает сохранение маршрута при перезагрузке системы. По умолчанию маршруты не сохраняются при перезагрузке.

-4 Обязательное использование протокола IPv4.

-6 Обязательное использование протокола IPv6.

command Одна из следующих команд:

PRINT Печать маршрута

ADD Добавление маршрута

DELETE Удаление маршрута

CHANGE Изменение существующего маршрута

destination Адресуемый узел.

netmask Значение маски подсети для записи данного маршрута (по умолчанию 255.255.255.255)

gateway Шлюз.

interface Номер интерфейса для указанного маршрута.

# Примеры

Для команд PRINT и DELETE можно указать узел и шлюз с помощью подстановочных знаков или опустить параметр «шлюз».

Знак '\*' соответствует любой строке, а '?' - одному знаку.

Соответствие шаблону поддерживает только команда PRINT.

```
>route PRINT
```

```
> route PRINT -4
```

```
> route PRINT -6
```

```
> route PRINT 157*      .... Печать только узлов, начинающихся со 157
```

```
> route ADD 157.0.0.0 MASK 255.0.0.0 157.55.80.1 METRIC 3 IF 2
```

узел^

^маска

^шлюз

метрика^

интерфейс^

```
> route CHANGE 157.0.0.0 MASK 255.0.0.0 157.55.80.5 METRIC 2 IF 2
```

```
> route DELETE 157.0.0.0
```

```
> route DELETE 3ffe::/32
```

# Алгоритм работы маршрутизации



# Таблица маршрутизации Unix-маршрутизатора

Destination	Gateway	Flags	Refcnt	Use	Interface
127.0.0.0	127.0.01	UH	1	154	lo0
Default	198.21.17.7	UG	5	43270	le0
198.21.17.0	198.21.17.5	U	35	246876	le0
213.34.12.0	213.34.12.3	U	44	132435	le1
129.13.0.0	198.21.1.7.6	UG	6	16450	le0
56.0.0.0	213.34.12.4	UG	12	5764	le1
116.0.0.0	213.34.12.4	UG	21	23544	le1

Flags:

**U** - показывает, что маршрут активен и работоспособен.

**H** — признак специфического маршрута к определенному хосту. Маршрут ко всей сети, к которой принадлежит данный хост, может отличаться от данного маршрута;

**G** - означает, что маршрут пакета проходит через промежуточный маршрутизатор.

**D** - означает, что маршрут получен из сообщения Redirect протокола ICMP. Такой признак может присутствовать только в таблице маршрутизации конечного узла.

Справочные поля:

**Refcnt** - показывает, сколько раз на данный маршрут ссылались при продвижении пакетов.

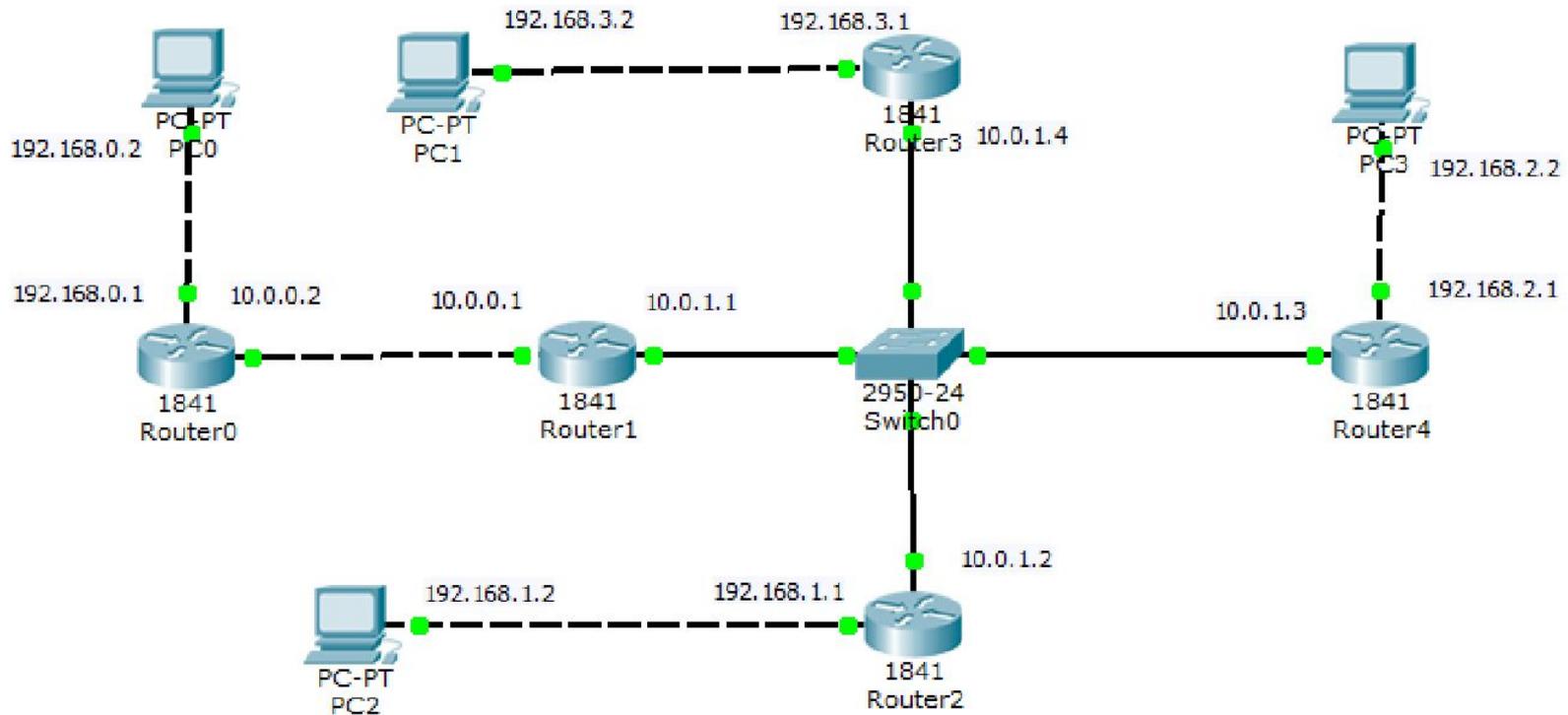
**Use** - отражает количество пакетов, переданных по данному маршруту.



# Статическая маршрутизация Cisco



# Пример сети



# Настройка маршрутизации Router0

The screenshot shows the configuration interface for Router0, specifically the Static Routes section. The interface is divided into three tabs: Physical, Config, and CLI. The Config tab is active, and the left sidebar shows a tree view of configuration categories: GLOBAL, ROUTING, SWITCHING, and INTERFACE. The Static Routes section is currently selected, displaying a table with the following entries:

Network	Mask	Next Hop
0.0.0.0	0.0.0.0	10.0.0.2

Below the table is an "Add" button. A summary box below the table shows the configuration: "Network Address: 0.0.0.0/0 via 10.0.0.2". A "Remove" button is located at the bottom right of the summary box.

At the bottom of the interface, there is a section titled "Equivalent IOS Commands" which is circled in red. It contains the following text:

```
Enter configuration commands, one per line. End with CNTL/Z.  
Router(config)#no ip route 0.0.0.0 0.0.0.0 10.0.0.2  
Router(config)#ip route 0.0.0.0 0.0.0.0 10.0.0.2  
Router(config)#
```

# Просмотр таблицы маршрутизации Router0

Router# en

Router# 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 10.0.0.2 to network 0.0.0.0

10.0.0.0/24 is subnetted, 1 subnets

**C** 10.0.0.0 is directly connected, FastEthernet0/0

**C** 192.168.0.0/24 is directly connected, FastEthernet0/1

**S\*** 0.0.0.0/0 [1/0] via 10.0.0.2

Router#

# Настройка маршрутизации Router1

The screenshot shows the configuration interface for Router1, specifically the Static Routes section. The interface is divided into three tabs: Physical, Config, and CLI. The Config tab is active, showing a tree view on the left with categories: GLOBAL, ROUTING, SWITCHING, and INTERFACE. The ROUTING category is expanded to show Static, RIP, and other options. The main area displays the Static Routes configuration, including a table for adding a new route and a list of existing routes. A red oval highlights the Equivalent IOS Commands section at the bottom.

Router1

Physical Config CLI

**GLOBAL**

Settings

Algorithm Settings

**ROUTING**

Static

RIP

**SWITCHING**

VLAN Database

**INTERFACE**

FastEthernet0/0

FastEthernet0/1

### Static Routes

Network	192.168.3.0
Mask	255.255.255.0
Next Hop	10.0.1.4

Add

Network Address

- 0.0.0.0/0 via 10.0.1.2
- 192.168.0.0/24 via 10.0.0.1
- 192.168.1.0/24 via 10.0.1.2
- 192.168.2.0/24 via 10.0.1.3
- 192.168.3.0/24 via 10.0.1.4

Remove

Equivalent IOS Commands

```
Router(config)#ip route 192.168.1.0 255.255.255.0 10.0.1.2
Router(config)#ip route 192.168.2.0 255.255.255.0 10.0.1.3
Router(config)#ip route 192.168.3.0 255.255.255.0 10.0.1.4
Router(config)#
```

# Просмотр таблицы маршрутизации Router0

Router#en

Router#show ip route

Codes: C - connected, S - static,

.....

\* - candidate default, U - per-user static route, o - ODR

P - periodic downloaded static route

Gateway of last resort is 10.0.1.2 to network 0.0.0.0

10.0.0.0/24 is subnetted, 2 subnets

C 10.0.0.0 is directly connected, FastEthernet0/0

C 10.0.1.0 is directly connected, FastEthernet0/1

S 192.168.0.0/24 [1/0] via 10.0.0.1

S 192.168.1.0/24 [1/0] via 10.0.1.2

S 192.168.2.0/24 [1/0] via 10.0.1.3

S 192.168.3.0/24 [1/0] via 10.0.1.4

S\* 0.0.0.0/0 [1/0] via 10.0.1.2

Router#



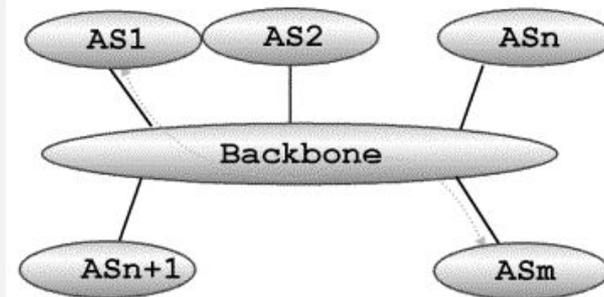
# Динамическая маршрутизация



# Концепция автономных систем

Вся сеть разбивается на отдельные области, в пределах которых управление процессами определения маршрута производится автономно.

Каждая из автономных систем  $AS_N$  может объединять в себе несколько локальных сетей. Взаимодействие между этими локальными сетями в пределах автономной системы производится в соответствии с правилами определения маршрута, которые приняты в данной автономной системе. Внутри каждой автономной системы могут быть использованы различные правила определения маршрута и протоколы маршрутизации. Информационное взаимодействие между компонентами различных автономных систем может быть выполнено только через специальную область, которая предназначена для интеграции всей системы в целом. Такая область называется Backbone Area.



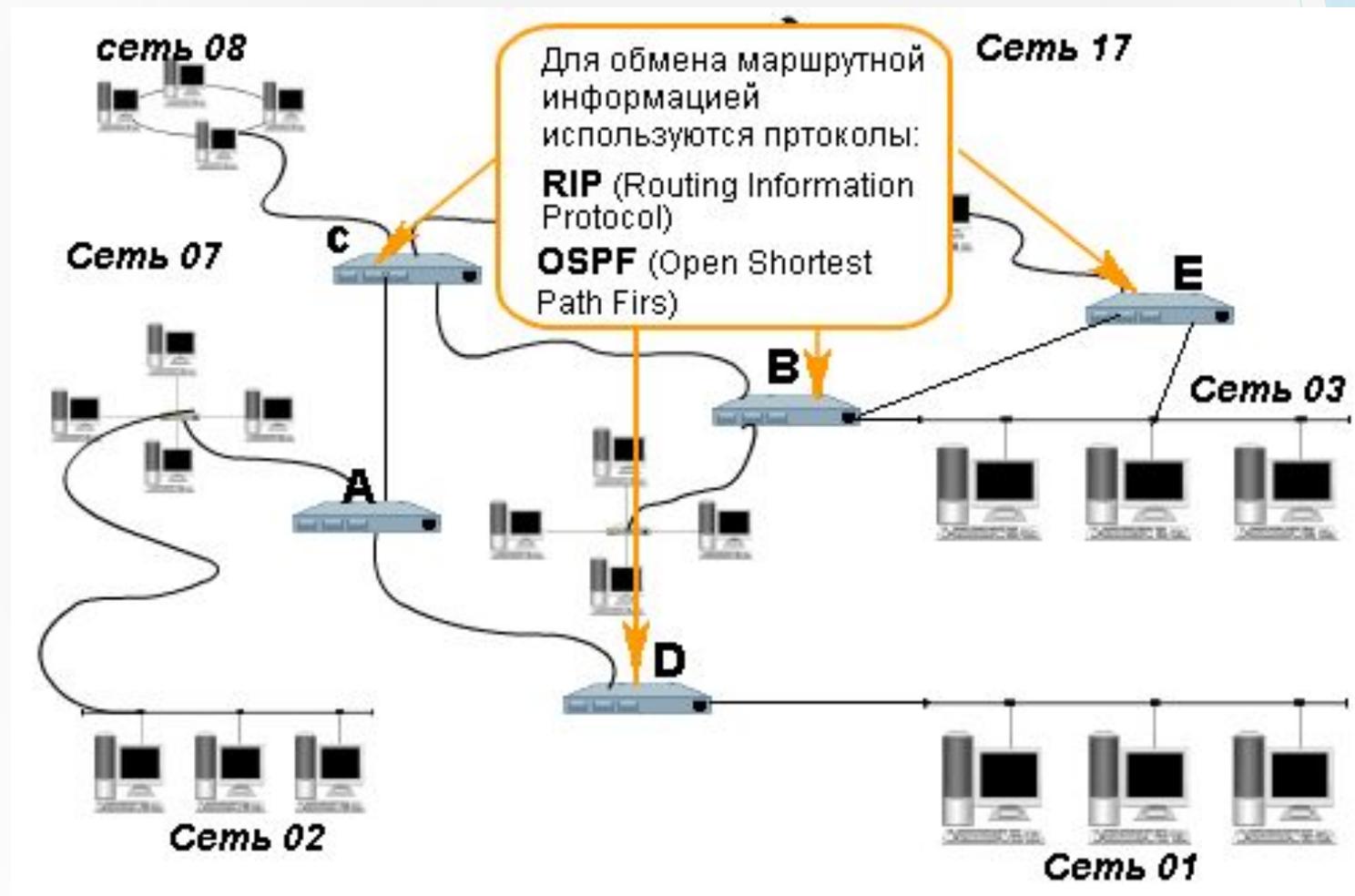
## Процесс определения маршрута:

- определение маршрута внутри автономной системы;
- определение маршрута между автономными системами.

Алгоритмы маршрутизации, которые используются для определения маршрута за пределами автономной системы, относятся к классу **exterior gateway protocol** - **EGP** .

Алгоритмы маршрутизации, которые используются для определения маршрута внутри автономной системы, относятся к классу **interior gateway protocol** - **IGP** .

# Динамическая маршрутизация



Дистанционно-векторная маршрутизация базируется на алгоритме Белмана-Форда. Через определённые моменты времени маршрутизатор передаёт соседним маршрутизаторам всю свою таблицу маршрутизации.



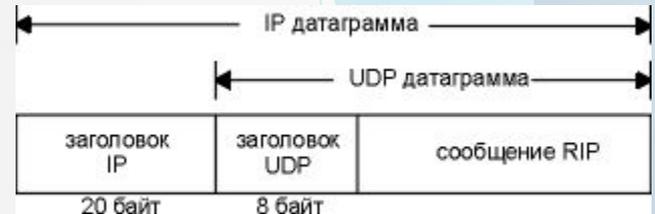
## Протокол RIP



# RIP (Routing Information Protocol)

(Порт/ID:520/UDP)

- ▶ Регулярное обновление маршрутизации. Каждые **30** секунд вся или часть таблицы маршрутизации отправляется каждому соседнему маршрутизатору.
- ▶ Если изменяется показатель маршрута, передается только та запись, которая была изменена.
- ▶ С каждым маршрутом связан тайм-аут. Если система, использующая RIP, определила, что маршрут не был обновлен в течение трех минут, показатель маршрута устанавливается в состояние "бесконечно" (**16**) и помечается для удаления. Это означает, что было пропущено **шесть 30-секундных** обновлений от маршрутизатора, который объявил маршрут. Однако, удаление маршрута из локальной таблицы маршрутизации откладывается еще **на 60 секунд**, чтобы убедиться что маршрут действительно исчез.

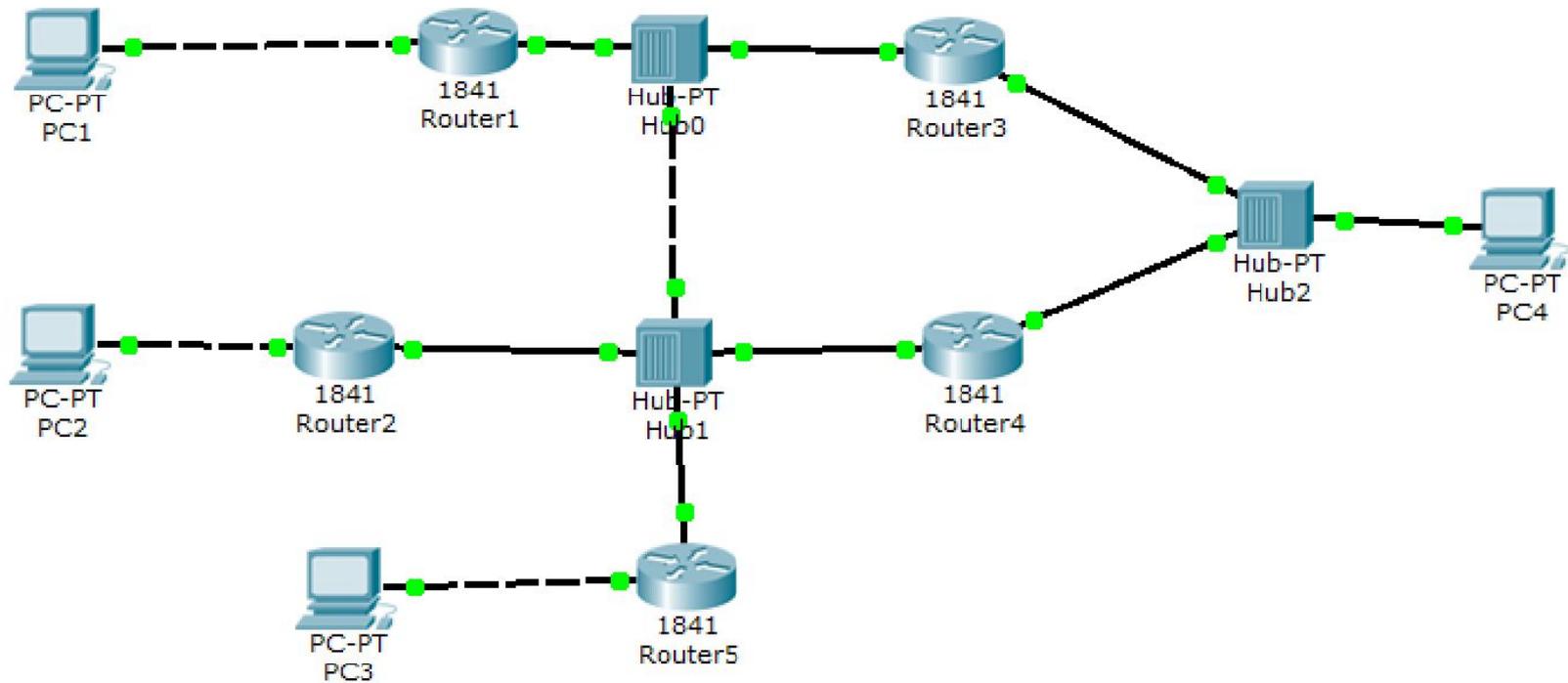


## Ограничения протокола RIP.

- ▶ Протокол RIP предназначен для использования в качестве IGP в сетях небольшого размера.
- ▶ Использование протокола RIP целесообразно в сетях, самый длинный путь в которых составляет не более 15 переходов.
- ▶ Использование протокола RIP может в ряде случаев привести к появлению «зацикленных маршрутов».
- ▶ Для сравнения маршрутов протокол RIP использует достаточно простую "метрику" - число переходов.

**RIP2** - является расширением протокола RIP, обеспечивающим передачу дополнительной маршрутной информации в сообщениях RIP и повышающим уровень безопасности.

# Сеть

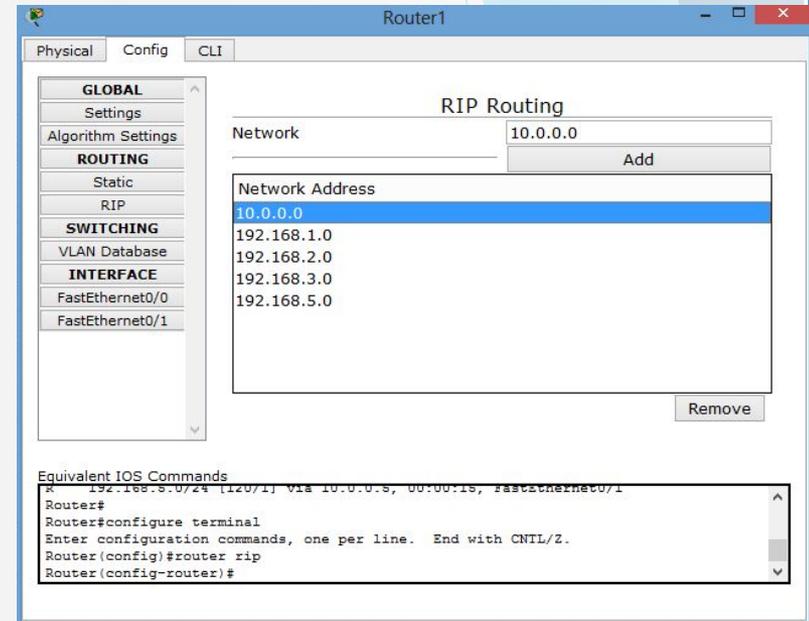


# Настройка маршрутизатора Router1

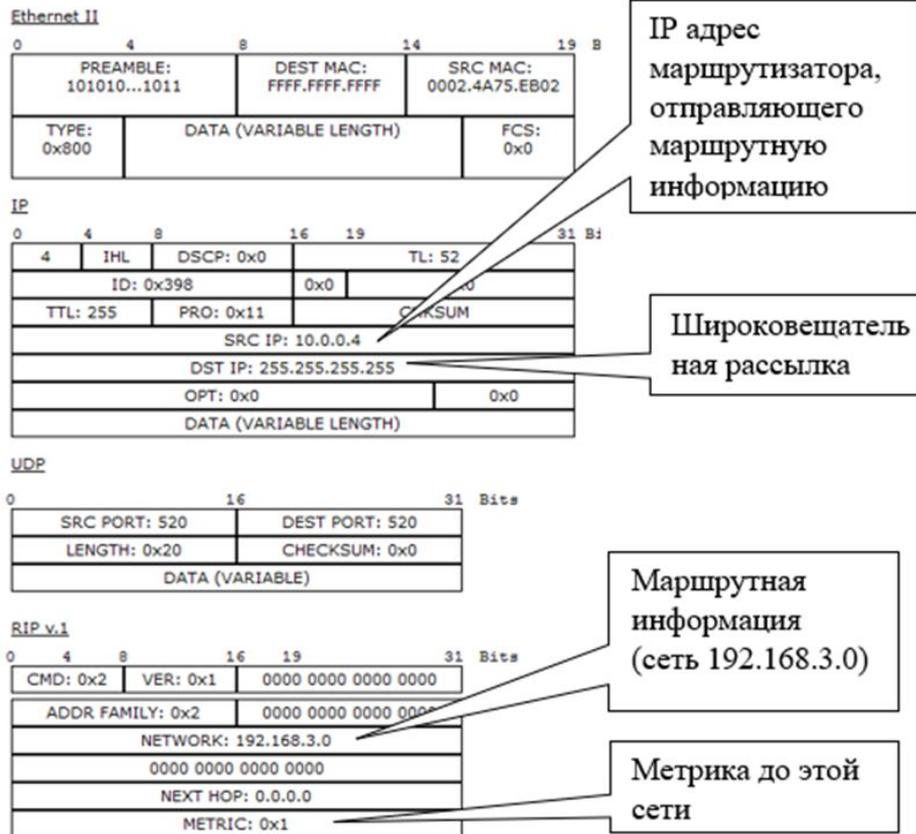
```
Router1>en
Router1>#conf t
Router1(config)#router rip
Router1(config-router)#network 10.0.0.0
```

.....

```
Router1(config-router)#version 2
Router1(config-router)#exit
Router1(config)#exit
Router1>#write memory
```



# Структура пакета RIP



Метрика 16 означает «бесконечно большую метрику» (маршрут недоступен).

# Таблица маршрутизации Cisco

Основные компоненты таблицы маршрутизации

- ▶ **Административная дистанция** - это мера достоверности источника маршрута. Если маршрутизатор изучил о месте назначения более чем от одного протокола маршрутизации, сравнивается административная дистанция и предпочтение отдается маршрутам чья административная дистанция меньше  
(Connected - 0, Static - 1, IGRP - 100, OSPF - 110, RIP - 120)
- ▶ **Метрика** - это мера, используемая протоколом маршрутизации для вычисления наилучшего пути к любому данному месту назначения, если протокол изучает несколько путей к одному и тому же месту назначения. Каждый протокол маршрутизации использует различные метрики.
- ▶ **Длина префикса** - маски подсети. Предпочтение отдается более длинной длине префикса.

# Таблица маршрутизации Router1

Router>en

Router#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

C 10.0.0.0 is directly connected, FastEthernet0/1

C 192.168.1.0/24 is directly connected, FastEthernet0/0

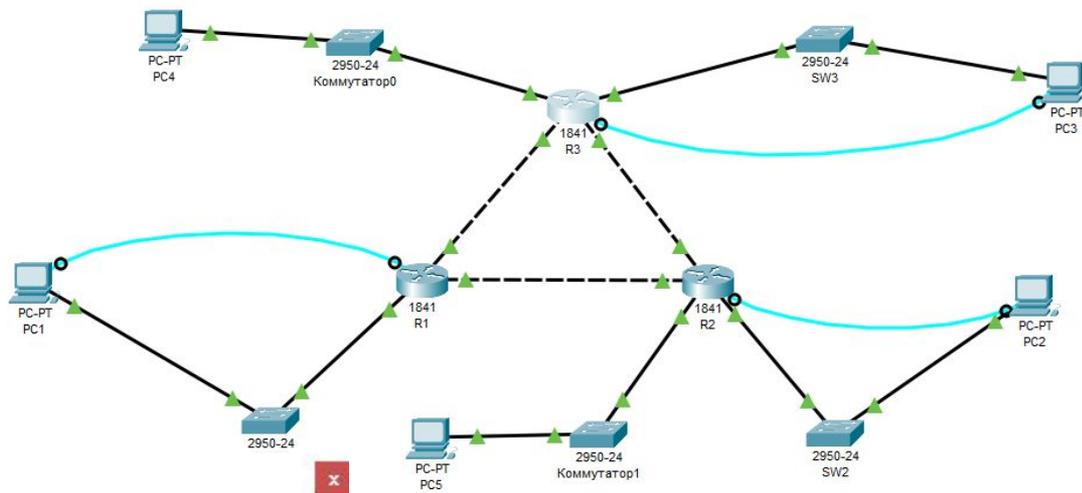
R 192.168.2.0/24 [120/1] via 10.0.0.2, 00:00:21, FastEthernet0/1

R 192.168.3.0/24 [120/1] via 10.0.0.3, 00:00:01, FastEthernet0/1

[120/16] via 10.0.0.4, 00:00:05, FastEthernet0/1

R 192.168.5.0/24 [120/1] via 10.0.0.5, 00:00:25, FastEthernet0/1

# Пример протокол RIP



Routing Table for R3

Type	Network	Port	Next Hop IP	Metric
C	10.1.0.0/30	FastEthernet0/0	---	0/0
C	10.2.0.0/30	FastEthernet0/1	---	0/0
R	10.3.0.0/30	FastEthernet0/0	10.1.0.1	120/1
R	10.3.0.0/30	FastEthernet0/1	10.2.0.2	120/1
R	192.168.10.0/24	FastEthernet0/0	10.1.0.1	120/1
R	192.168.20.0/24	FastEthernet0/1	10.2.0.2	120/1
C	192.168.30.0/24	Ethernet0/1/0	---	0/0
C	192.168.40.0/24	Ethernet0/0/0	---	0/0
R	192.168.50.0/24	FastEthernet0/1	10.2.0.2	120/1

Эти протоколы состояния связи предлагают лучшую масштабируемость и сходимость по сравнению с дистанционно-векторными протоколами. Работа протоколов базируется на алгоритме Дейкстры.



## Протокол OSPF



## OSPF (*Open Shortest Path First*) (Порт/ID:89/UDP)

Протокол динамической маршрутизации, основанный на технологии отслеживания состояния канала (link-state technology) и использующий для нахождения кратчайшего пути Алгоритм Дейкстры .

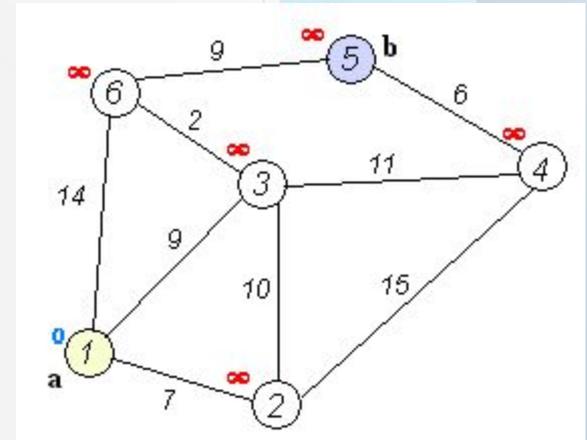
Протокол OSPF представляет собой протокол внутреннего шлюза (Interior Gateway Protocol – IGP). Протокол OSPF распространяет информацию о доступных маршрутах между маршрутизаторами одной автономной системы.

OSPF предлагает решение следующих задач:

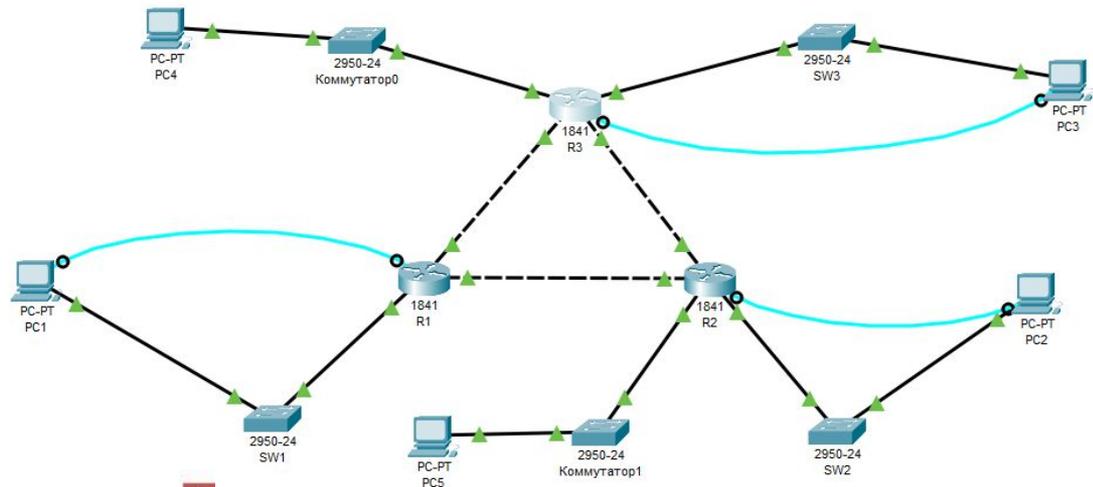
- Увеличение скорости сходимости (в сравнении с протоколом RIP2, так как нет необходимости выжидания многократных тайм-аутов по 30с);
- Достижимость сети (быстро обнаруживаются отказавшие маршрутизаторы, и топология сети изменяется соответствующим образом);
- Оптимальное использование пропускной способности (т.к строится минимальный остовный граф по алгоритму Дейкстры);
- Метод выбора пути.

# Алгоритм Дейкстры

- ▶ Каждой вершине из  $V$  сопоставим метку — минимальное известное расстояние от этой вершины до  $a$ . Алгоритм работает пошагово — на каждом шаге он «посещает» одну вершину и пытается уменьшать метки. Работа алгоритма завершается, когда все вершины посещены.
- ▶ **Инициализация.** Метка самой вершины  $a$  полагается равной 0, метки остальных вершин — бесконечности. Это отражает то, что расстояния от  $a$  до других вершин пока неизвестны. Все вершины графа помечаются как непосещённые.
- ▶ **Шаг алгоритма.** Если все вершины посещены, алгоритм завершается. В противном случае, из ещё не посещённых вершин выбирается вершина  $u$ , имеющая минимальную метку. Мы рассматриваем всевозможные маршруты, в которых  $u$  является предпоследним пунктом. Вершины, в которые ведут рёбра из  $u$ , назовем *соседями* этой вершины. Для каждого соседа вершины  $u$ , кроме отмеченных как посещённые, рассмотрим новую длину пути, равную сумме значений текущей метки  $u$  и длины ребра, соединяющего  $u$  с этим соседом. Если полученное значение длины меньше значения метки соседа, заменим значение метки полученным значением длины. Рассмотрев всех соседей, пометим вершину  $u$  как посещённую и повторим шаг алгоритма.



# Пример протокол OSPF



Routing Table for R3

Type	Network	Port	Next Hop IP	Metric
C	10.1.0.0/30	FastEthernet0/0	---	0/0
C	10.2.0.0/30	FastEthernet0/1	---	0/0
O	10.3.0.0/30	FastEthernet0/0	10.1.0.1	110/11
O	10.3.0.0/30	FastEthernet0/1	10.2.0.2	110/11
O	192.168.10.0/24	FastEthernet0/0	10.1.0.1	110/2
O	192.168.20.0/24	FastEthernet0/1	10.2.0.2	110/2
C	192.168.30.0/24	Ethernet0/1/0	---	0/0
C	192.168.40.0/24	Ethernet0/0/0	---	0/0
O	192.168.50.0/24	FastEthernet0/1	10.2.0.2	110/11

# Преимущества OSPF:

- ▶ Для каждого адреса может быть несколько маршрутных таблиц.
- ▶ Каждому интерфейсу присваивается безразмерная цена, учитывающая пропускную способность, время транспортировки сообщения. Для каждой IP-операции может быть присвоена своя цена (коэффициент качества).
- ▶ При существовании эквивалентных маршрутов OSPF распределяет поток равномерно по этим маршрутам.
- ▶ Поддерживается адресация субсетей.
- ▶ При связи точка-точка не требуется IP-адрес для каждого из концов.
- ▶ Применение мультикастинга вместо широковещательных сообщений снижает загрузку не вовлеченных сегментов.

# Настройка протокола OSPF

```
Router>en
```

```
Router#conf t
```

```
Router(config)#router ospf 1
```

```
Router(config-router)#network 10.0.0.0 255.255.255.0 area 1
```

```
.....
```

```
Router(config-router)#exit
```

```
Router(config)#exit
```

```
Router #write memory
```

1 – идентификатор процесса (уникальное числовое значение для каждого процесса роутинга на маршрутизаторе. Данное значение должно быть больше в интервале от 1 до 65535. В OSPF процессам на роутерах одной зоны принято присваивать один и тот же идентификатор.

# Таблица маршрутизации

Router#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

C 10.0.0.0 is directly connected, FastEthernet0/1

C 192.168.1.0/24 is directly connected, FastEthernet0/0

O 192.168.2.0/24 [110/2] via 10.0.0.2, 00:18:33, FastEthernet0/1

O 192.168.3.0/24 [110/2] via 10.0.0.4, 00:05:06, FastEthernet0/1  
[110/2] via 10.0.0.3, 00:05:06, FastEthernet0/1

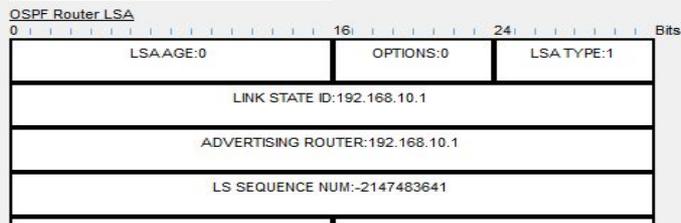
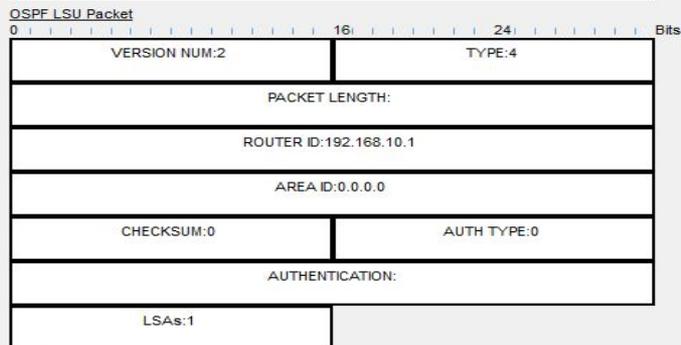
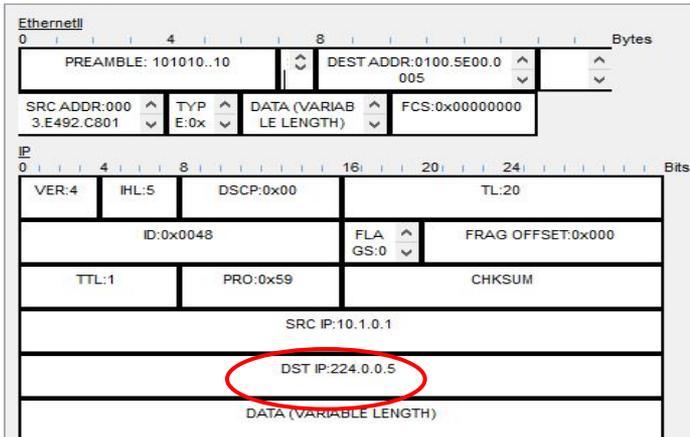
O 192.168.5.0/24 [110/2] via 10.0.0.5, 00:12:03, FastEthernet0/1

# Пакеты OSPF

DU Information at Device: R3

OSI Model Inbound PDU Details

PDU Formats

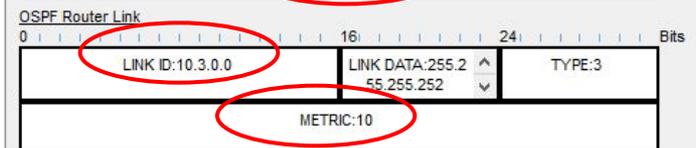
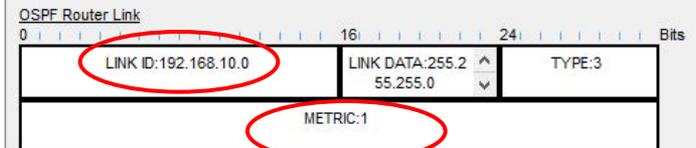
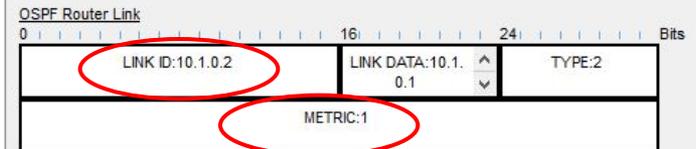
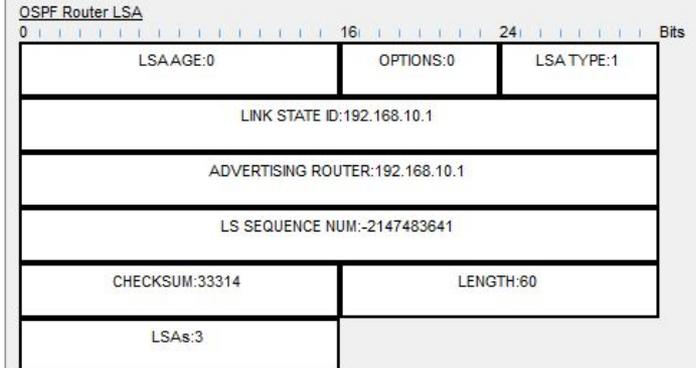
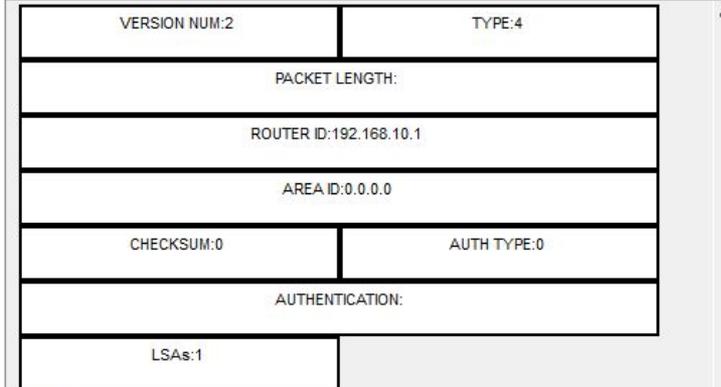


Для того чтобы обозначить недоступную сеть, OSPF использует метрику равную 16777215.

PDU Information at Device: R3

OSI Model Inbound PDU Details

PDU Formats



Протокол BGP предназначен для обмена информацией о достижимости подсетей между автономными системами (АС), то есть группами маршрутизаторов под единым техническим и административным управлением, использующими протокол **внутридоменной** маршрутизации для определения маршрутов внутри себя и протокол **междоменной** маршрутизации для определения маршрутов доставки пакетов в другие АС.

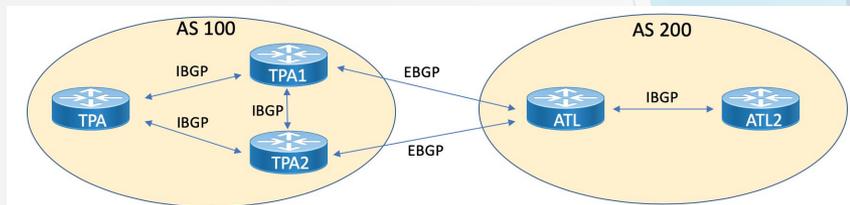


## Протокол BGP



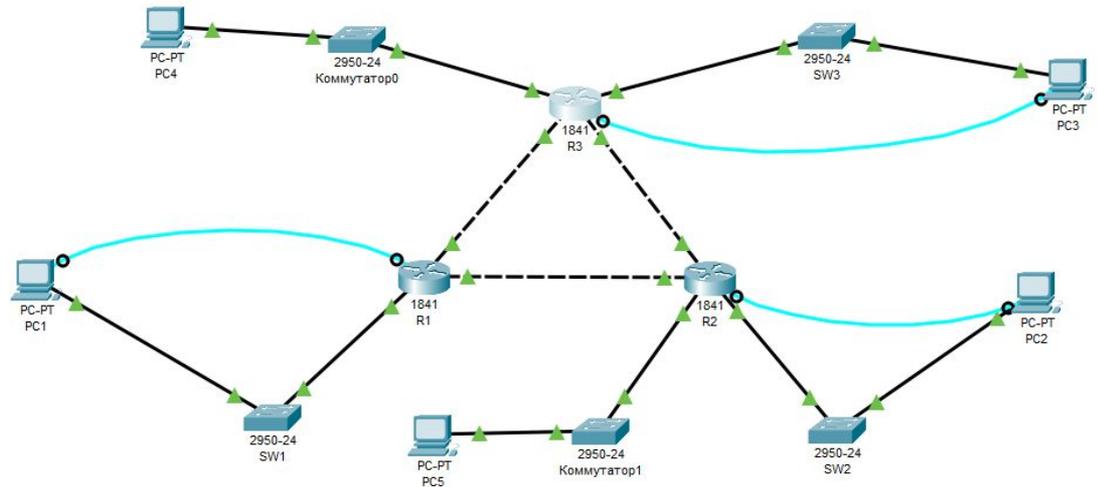
# BGP (Border Gateway Protocol). (Порт/ID:179/TCP)

Описание работы протокола:



1. После включения маршрутизаторов протокол ищет непосредственно подключенных соседей и устанавливает с ними «дружеские» отношения.
2. Затем они обмениваются друг с другом информацией о подключенных и доступных им сетях. То есть они строят карту сети (топологию сети). Данная карта одинакова на всех маршрутизаторах.
3. На основе полученной информации запускается алгоритм SPF (Shortest Path First, «выбор наилучшего пути»), который рассчитывает оптимальный маршрут к каждой сети. Данный процесс похож на построение дерева, корнем которого является сам маршрутизатор, а ветвями — пути к доступным сетям. Данный процесс, то есть конвергенция, происходит очень быстро.

# Пример протокол BGP



Routing Table for R3

Type	Network	Port	Next Hop IP	Metric
C	10.1.0.0/30	FastEthernet0/0	---	0/0
C	10.2.0.0/30	FastEthernet0/1	---	0/0
B	10.3.0.0/30	FastEthernet0/1	10.2.0.2	20/0
B	192.168.10.0/24	FastEthernet0/0	10.1.0.1	20/0
B	192.168.20.0/24	FastEthernet0/1	10.2.0.2	20/0
C	192.168.30.0/24	Ethernet0/1/0	---	0/0
C	192.168.40.0/24	Ethernet0/0/0	---	0/0
B	192.168.50.0/24	FastEthernet0/1	10.2.0.2	20/0

# Пакеты BGP

PDU Information at Device: R1

OSI Model [Inbound PDU Details](#)

PDU Formats

EthernetII

PREAMBLE: 101010..10		DEST ADDR:0001.43C6.C6 EC	
SRC ADDR:0060 .3E0D.8398	TYP E:0x	DATA (VARIABLE LENGTH)	FCS:0x00000000

IP

VER:4	IHL:5	DSCP:0x00	TL:69
ID:0x0008		FLAG S:0x2	FRAG OFFSET:0x000
TTL:255	PRO:0x06	CHKSUM	
SRC IP:10.3.0.2			
DST IP:10.3.0.1			
DATA (VARIABLE LENGTH)			

TCP

SOURCE PORT:1028		DESTINATION PORT:179	
SEQUENCE NUMBER:1			
ACKNOWLEDGEMENT NUMBER:1			
OFFS ET:0x 0	RESE RVED :0	FLAGS:0b00011000	WINDOW:65535
CHECKSUM:0x0000		URGENT POINTER:0x0000	
OPTION			
DATA (VARIABLE LENGTH)		PADDING: 0	

BGP Open

Version:4	AS:2	HT:180
ID:3232248321		
Length	Optional Parameter(VARIABLE LENGTH): 0	

PDU Information at Device: R1

OSI Model [Inbound PDU Details](#)

PDU Formats

SRC ADDR:000  
7.EC43.0B01

TYP  
E:0x

DATA (VARIABLE LENGTH)

FCS:0x00000000

IP

VER:4	IHL:5	DSCP:0x00	TL:73
ID:0x004d		FLAG GS:0	FRAG OFFSET:0x000
TTL:255	PRO:0x06	CHKSUM	
SRC IP:10.1.0.2			
DST IP:10.1.0.1			
DATA (VARIABLE LENGTH)			

TCP

SOURCE PORT:1025		DESTINATION PORT:179	
SEQUENCE NUMBER:1248			
ACKNOWLEDGEMENT NUMBER:1248			
OFFS ET:0x	RESE RVE	FLAGS:0b00011 000	WINDOW:65535
CHECKSUM:0x0000		URGENT POINTER:0x0000	
OPTION			
DATA (VARIABLE LENGTH)		PADDING: 0	

BGP Update

Length:0	
----------	--

BGP Attribute

Flag:	Type:3
-------	--------

BGP Attribute

Flag:	Type:2
-------	--------

BGP Attribute

Flag:	Type:5
-------	--------