



# Сетевой уровень

МАРШРУТИЗАЦИЯ, IP-АДРЕСАЦИЯ, РАЗБИЕНИЕ IP-СЕТЕЙ НА ПОДСЕТИ

# Модель OSI, стек протоколов TCP/IP

Модель OSI	Протоколы	Стек протоколов TCP/IP
Уровень приложений	HTTP, DNS, DHCP, FTP	Уровень приложений
Уровень представления		
Сеансовый уровень		
Транспортный уровень	TCP, UDP	Транспортный уровень
Сетевой уровень	IPv4, IPv6, ICMPv4, ICMPv6	Межсетевой уровень
Канальный уровень	PPP, Frame Relay, Ethernet	Уровень сетевого доступа
Физический уровень		

# Сетевой уровень

- ▶ Сетевой уровень (3 уровень модели OSI) предоставляет сервисы, позволяющие конечным устройствам обмениваться данными по сети.

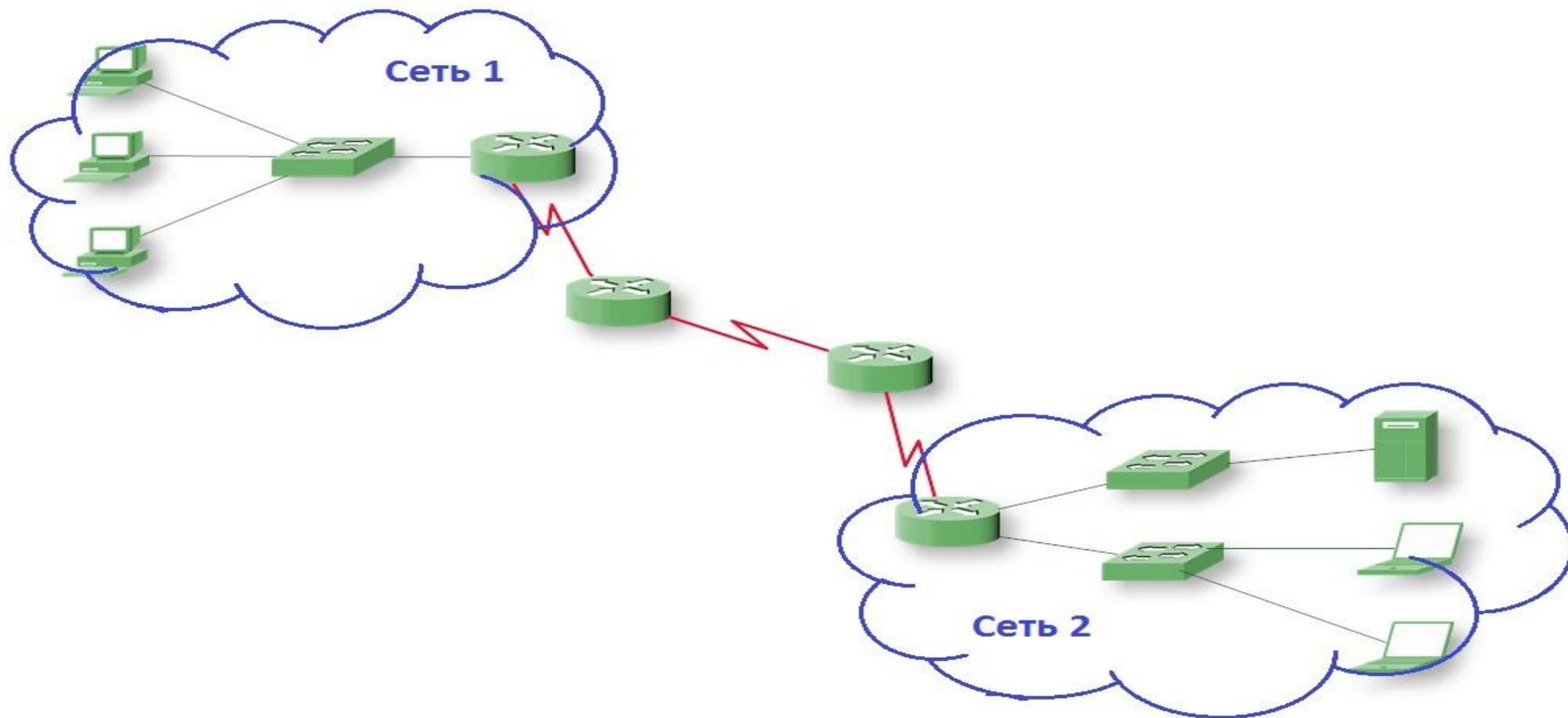
Адресация  
конечных  
устройств

Инкапсуляция

Маршрутизация

Декапсуляция

# Сетевой уровень



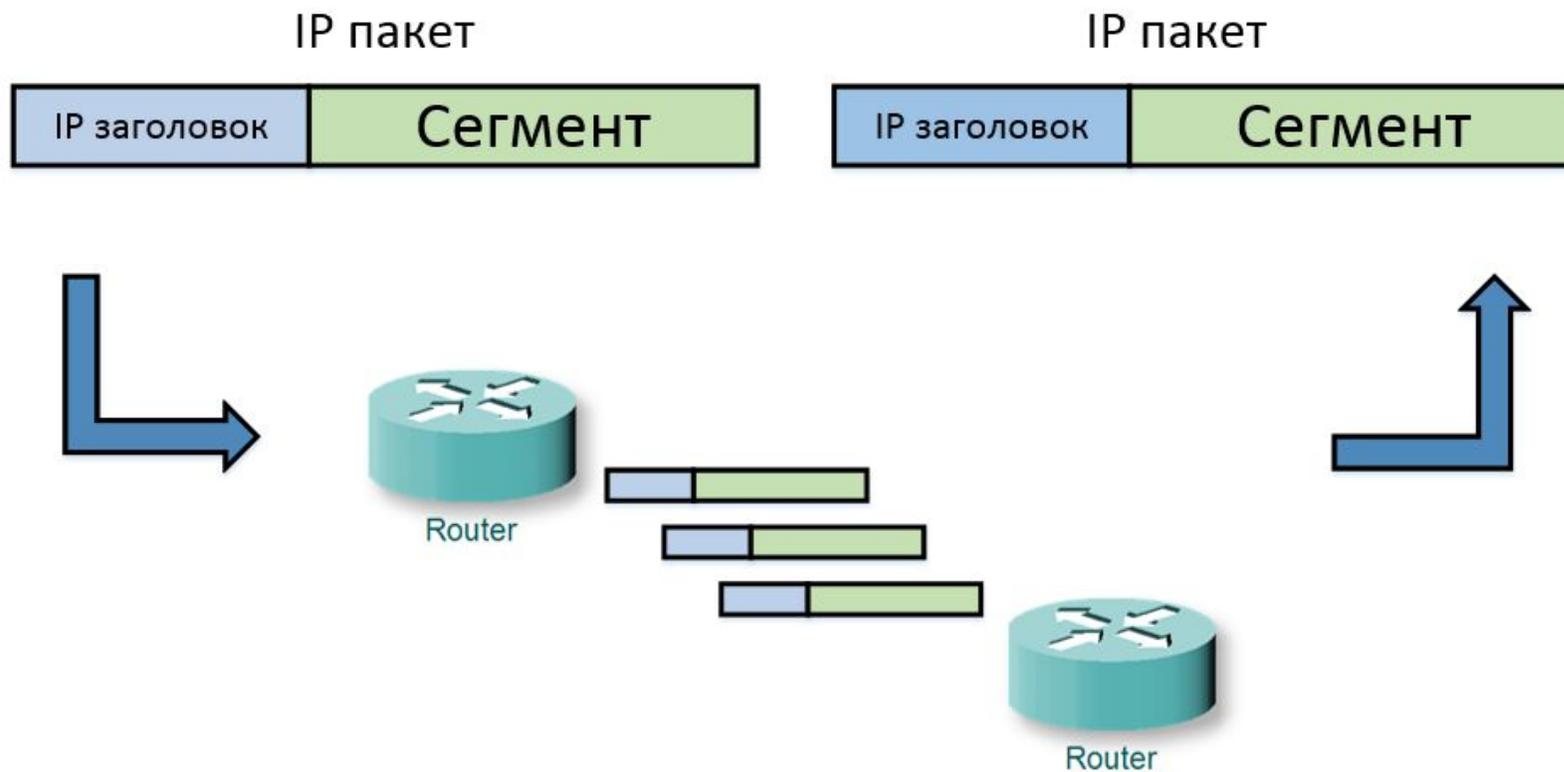
# Протоколы сетевого уровня

- ▶ **Маршрутизируемый** - это любой сетевой протокол, адрес сетевого уровня которого предоставляет достаточное количество информации для доставки пакета от одного сетевого узла другому на основе используемой схемы адресации.
- ▶ **Маршрутизации** - это протокол, который поддерживает маршрутизируемые протоколы и предоставляет механизмы обмена маршрутной информацией.

# Протокол межсетевого взаимодействия

- ▶ **IP (Internet Protocol)** – межсетевой маршрутизируемый протокол сетевого уровня стека TCP/IP. Именно IP стал тем протоколом, который объединил отдельные компьютерные сети во всемирную сеть Интернет. Неотъемлемой частью протокола является *адресация сети*
  - ▶ internet – объединенная сеть
  - ▶ Основа сети Интернет

# Протокол межсетевого взаимодействия



# Характеристики IP

- ▶ Объединение сетей
- ▶ Не зависит от среды
- ▶ Качество обслуживания
- ▶ Передача без установления соединения
  - ▶ Нет гарантии доставки
  - ▶ Произвольный порядок доставки

# Версии IP протокола

## ▶ **Протокол IP версии 4 (IPv4)**

- ▶ Длина IP-адреса 4 байта
- ▶ Используется сейчас

## ▶ **Протокол IP версии 6 (IPv6)**

- ▶ Длина IP-адреса 16 байт
- ▶ Вводится в эксплуатацию

# Пакет IPv4

- ▶ Пакет IPv4 состоит из двух частей:
  - ▶ **заголовок IP**: определяет характеристики пакета;
  - ▶ **полезная нагрузка**: содержит информацию сегмента уровня 4 и пользовательские данные

# Пакет IPv4

Октет	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Версия			IHL			Тип обслуживания				Длина пакета																					
4	Идентификатор								Флаги		Смещение фрагмента																					
8	Время жизни (TTL)				Протокол				Контрольная сумма заголовка																							
12	IP-адрес отправителя																															
16	IP-адрес получателя																															
20	Параметры (от 0 до 10-и 32-х битных слов)																															
	Данные																															

# Пример заголовка IPv4

```
⊕ Frame 1: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface 0
⊕ Ethernet II, Src: IntelCor_24:89:8e (fc:f8:ae:24:89:8e), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊖ Internet Protocol Version 4, Src: 10.1.11.184 (10.1.11.184), Dst: 10.1.11.255 (10.1.11.255)
  Version: 4
  Header Length: 20 bytes
  ⊕ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
  Total Length: 78
  Identification: 0x786a (30826)
  ⊕ Flags: 0x00
  Fragment offset: 0
  Time to live: 128
  Protocol: UDP (17)
  ⊕ Header checksum: 0x967c [validation disabled]
  Source: 10.1.11.184 (10.1.11.184)
  Destination: 10.1.11.255 (10.1.11.255)
  [Source GeoIP: Unknown]
  [Destination GeoIP: Unknown]
⊕ User Datagram Protocol, Src Port: 137 (137), Dst Port: 137 (137)
⊕ NetBIOS Name Service
```

# Адресация

- ▶ Адресация — это основная функция протоколов сетевого уровня, которая позволяет узлам обмениваться данными вне зависимости от того, находятся ли узлы в одной или нескольких сетях.
- ▶ IP-протокол версии 4 (IPv4) и IP-протокол версии 6 (IPv6) обеспечивают иерархическую адресацию пакетов, которые служат для передачи данных.

# IP - адрес

- ▶ IP адрес - уникальный сетевой адрес узла в компьютерной сети, построенной по протоколу IP.
- ▶ IP-адрес (v4) состоит из 32-бит. IP-адрес (v6) состоит из 128-бит.
- ▶ Всего теоретически IPv4-адресов может быть:  
$$2^{32} = 2^{10} * 2^{10} * 2^{10} * 2^2 = 1024 * 1024 * 1024 * 4 \approx 1000 * 1000 * 1000 * 4 = 4 \text{ млрд.}$$
- ▶ IP адрес обычно записывается в виде четырех десятичных номеров, разделенных точками: 192.168.0.8
- ▶ В двоичном представлении IP адрес записывается в виде четырех октетов, разделенных точками 11000000.10101000.00000001.00001000

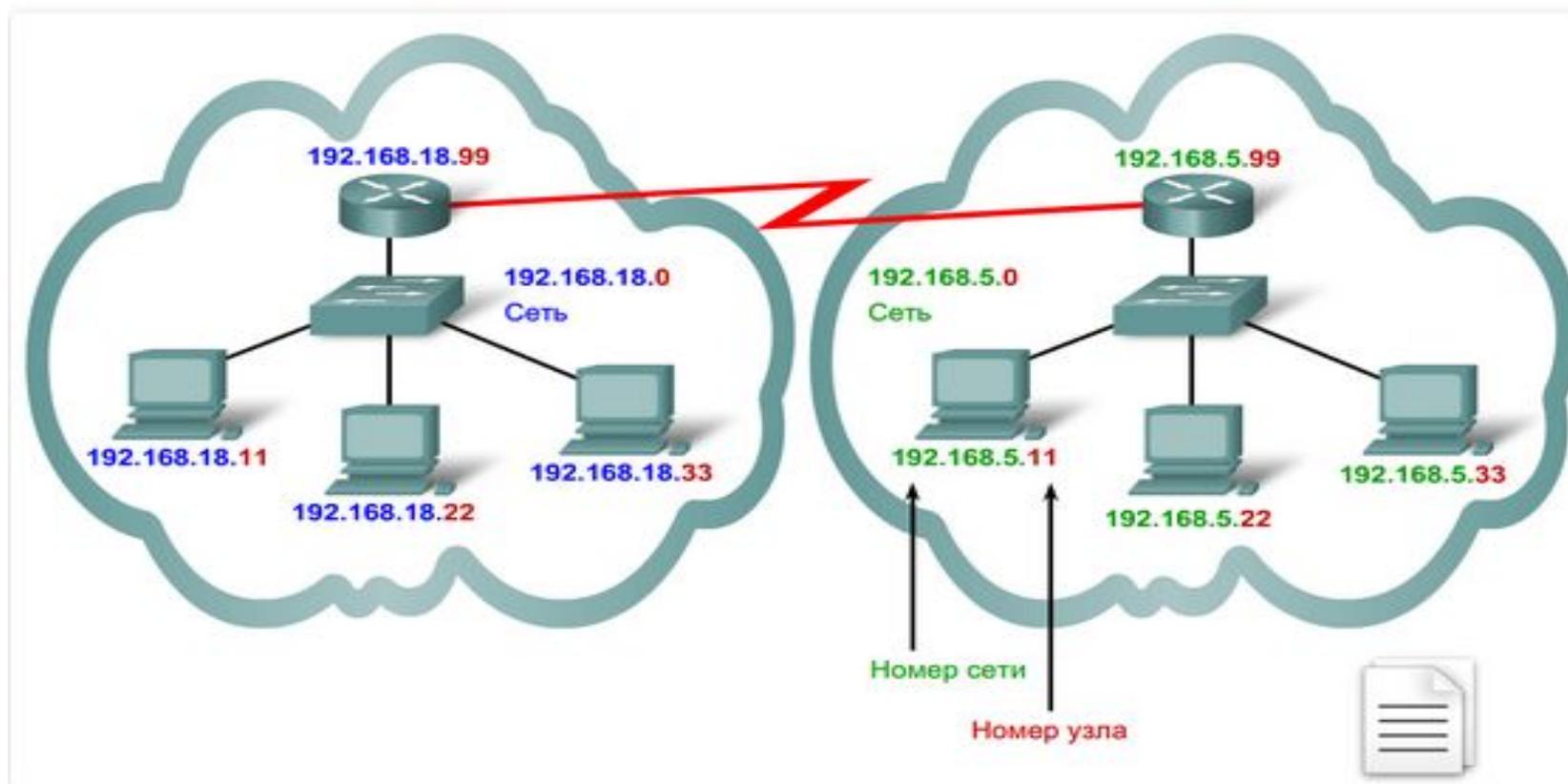
# IP - адрес

- ▶ 1 октет 8 бит
- ▶ Значение в каждом октете может быть от 0 до 255 в десятичном представлении или от 00000000 до 11111111 в двоичном представлении.
- ▶ Преобразование двоичных октетов в десятичное представление:
  - ▶ 1 1 1 1 1 1 1 1
  - ▶ 128 64 32 16 8 4 2 1 (128 + 64 + 32 + 16 + 8 + 4 + 2 + 1)
  
  - ▶ 0 1 0 0 0 0 0 0
  - ▶ 0 64 0 0 0 0 0 1 (0 + 64 + 0 + 0 + 0 + 0 + 0 + 1)

# Формат IP-адреса



# Номер сети, номер узла



# Где номер узла? Где номер сети?

- ▶ **Классовая IP адресация** —используется фиксированная маска подсети, поэтому класс сети всегда можно идентифицировать по первым битам. Нерациональный подход

Класс	Первый байт(2)	Первый байт(10)	Адреса хостов	Маска
A	0xxxxxxx	1-127	10.0.0.1 - 126.255.255.254	255.0.0.0
B	10xxxxxx	128-191	128.0.0.1 - 191.255.255.254	255.255.0.0
C	110xxxxx	192-223	192.0.0.1 - 223.255.255.254	255.255.255.0
D	1110xxxx	224-239	224.0.0.1 - 239.255.255.254	255.255.255.0
E	11110xxx	240-247	240.0.0.1 - 255.255.255.254	255.255.255.0

- ▶ **Бесклассовая IP адресация (*Classless Inter-Domain Routing — CIDR*)** — используются маски подсети переменной длины (*variable length subnet mask — VLSM*). Метод IP-адресации, который позволяет рационально управлять пространством IP адресов.

# Маска подсети

- ▶ Маска подсети определяет границы подсети
- ▶ Маска подсети - 32-бита
- ▶ В отличие от IP-адреса, нули и единицы в маске подсети не могут чередоваться. Всегда сначала идут единицы, потом нули

255.255.248.0=11111111.11111111.11111000.00000000

- ▶ Сначала N единиц, потом 32-N нулей
- ▶ Для записи маски используют число N, называемого длиной маски.

192.168.11.10/21 вместо 192.168.11.10 255.255.248.0 или 11111111.11111111.11111000.00000000

# Определение границ подсети

- ▶ Для определения границ подсети компьютер делает побитовое умножение (логическое И) между IP-адресом и маской, получая на выходе адрес с обнуленными битами в позициях нулей маски

11000000.10101000.00001011.00001010

X

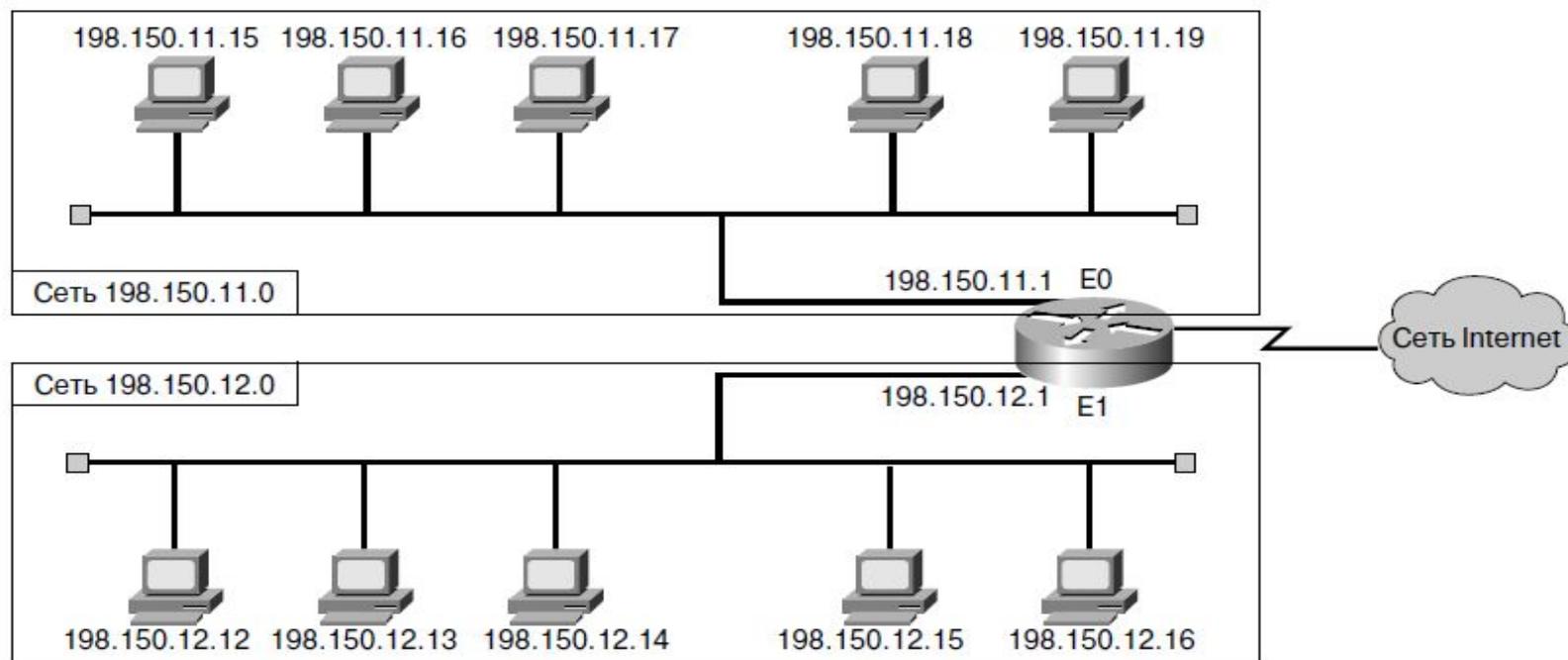
11111111.11111111.11111000.00000000

-----  
11000000.10101000.00001000.00000000 = 192.168.8.0

# Зарезервированные адреса

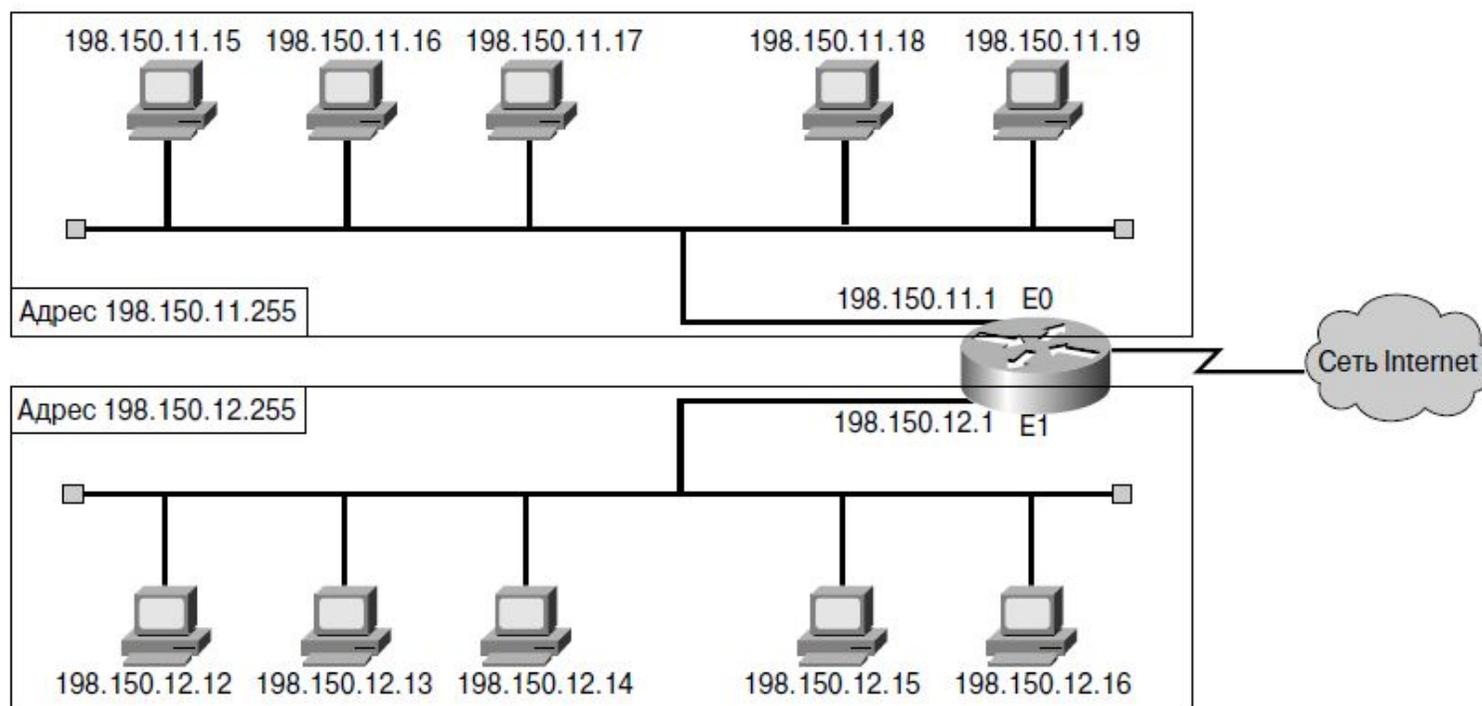
► Адреса, которые не могут быть присвоены сетевым устройствам:

► **1) Адрес сети**

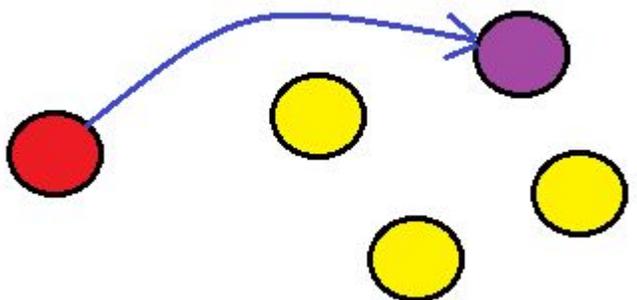
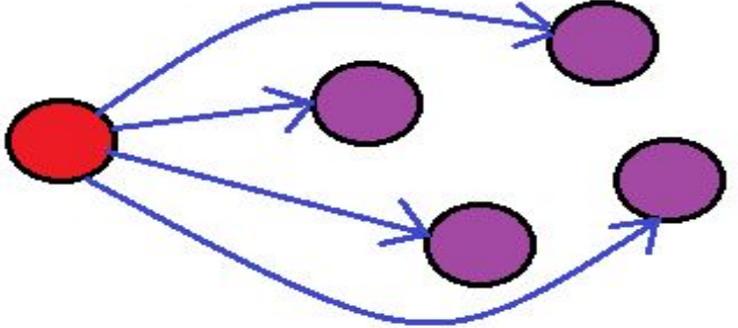
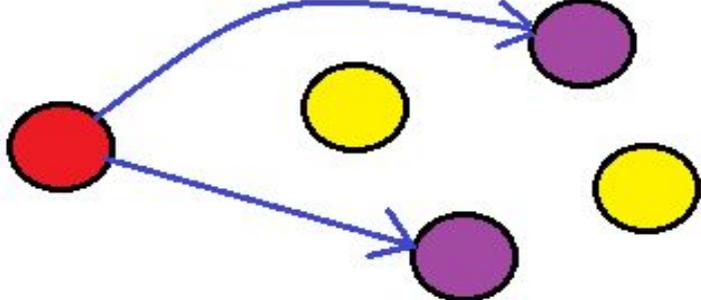


# Зарезервированные адреса

- ▶ Адреса, которые не могут быть присвоены сетевым устройствам:
- ▶ **2) Широковещательный адрес**



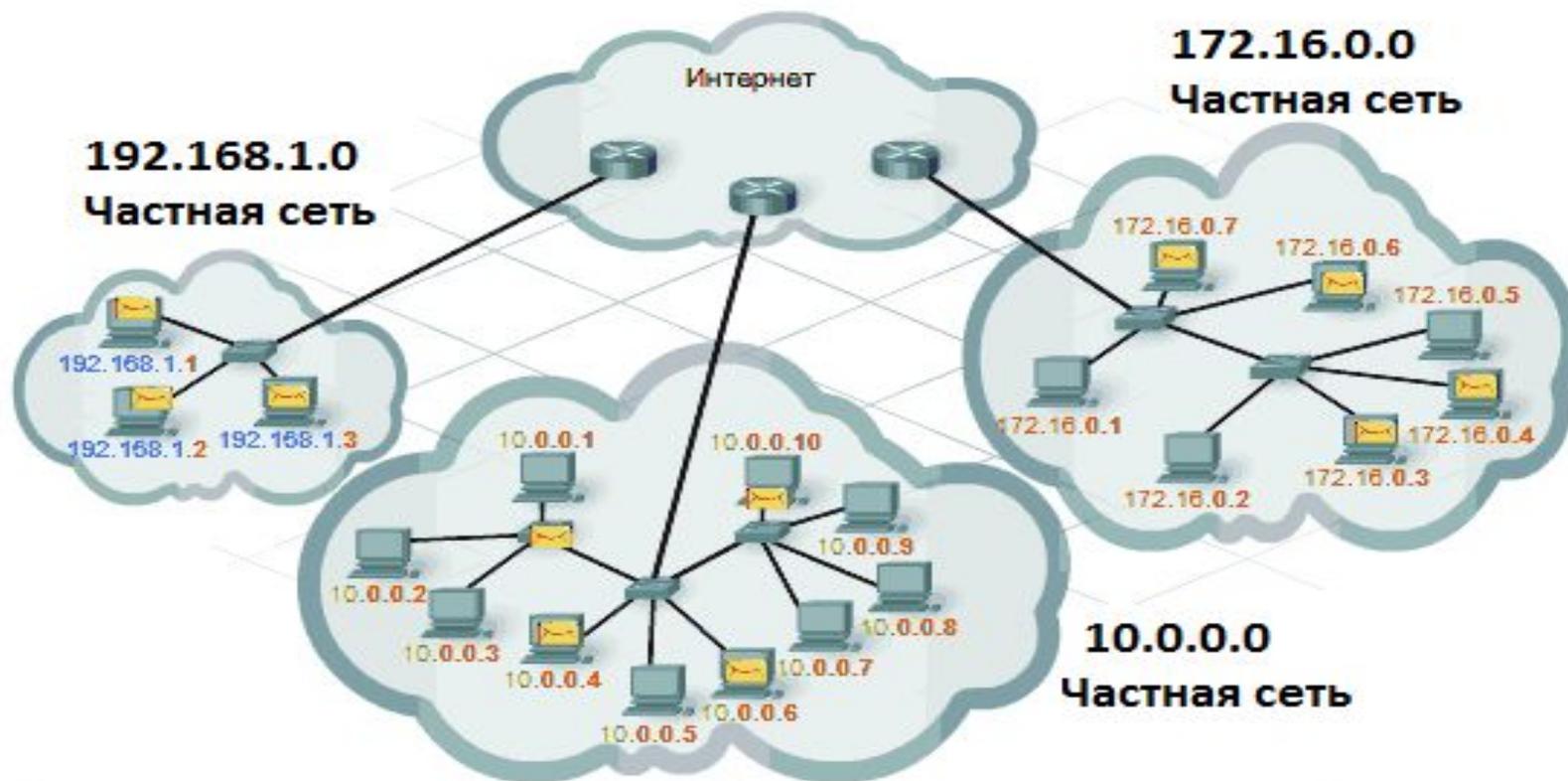
# Типы рассылок

<b>Одноадресная рассылка Unicast</b>	<b>Широковещательная рассылка Multicast</b>	<b>Многоадресная рассылка Anycast</b>
процесс отправки пакета с одного узла на индивидуальный	процесс отправки пакета с одного узла на все узлы в сети	процесс отправки пакета с одного узла выбранной группе узлов, возможно, в различных сетях
		

# Публичные и частные IP-адреса

- ▶ **Публичные IP адреса** - уникальны, используются в глобальном масштабе и подчиняются стандарту.
- ▶ **Частные IP адреса** - уникальны только внутри локальной сети
- ▶ 10.0.0.0 — 10.255.255.255 (маска подсети: 255.0.0.0 или /8)
- ▶ 172.16.0.0 — 172.31.255.255 (маска подсети: 255.240.0.0 или /12)
- ▶ 192.168.0.0 — 192.168.255.255 (маска подсети: 255.255.0.0 или /16)

# Публичные и частные IP-адреса



# IPv4 VS IPv6

IPv4 4 октета	IPv6 16 октетов
11010001.11011100.11001001.01110001	11010001.11011100.11001001.01110001.11010001.11011100. 110011001.01110001.11010001.11011100.11001001. 01110001.11010001.11011100.11001001.01110001
209.156.201.113	A524:72D3:2C80:DD02:0029:EC7A:002B:EA73
4,294,467,295 IP адресов	16 млрд IP адресов

# Присвоение IP - адресов

- ▶ **Статическое присвоение IP – адреса**
- ▶ **Динамическое присвоение IP – адреса**
  - ▶ **Выделение адреса с помощью протокола DHCP**

# Выделение адреса с помощью протокола DHCP

- ▶ **DHCP** - (Dynamic Host Configuration Protocol) – протокол позволяет динамически получить IP - адрес, не прибегая к создаваемым администратором профилям для каждой конкретной машины.
- ▶ Все, что нужно, это назначить диапазон доступных адресов на DHCP - сервере.
- ▶ **Распределение IP-адресов:**
  - ▶ **Ручное распределение** - сетевой администратор сопоставляет аппаратному адресу каждого клиентского компьютера определённый IP-адрес.
  - ▶ **Автоматическое распределение** -каждому компьютеру на постоянное использование выделяется произвольный свободный IP-адрес из определённого администратором диапазона.
  - ▶ **Динамическое распределение** - адрес выдаётся компьютеру не на постоянное пользование, а на определённый срок (*арендой адреса*). По истечении срока аренды IP-адрес вновь считается свободным, и клиент обязан запросить новый.

# Протокол преобразования адресов ARP

- ▶ ARP (Address Resolution Protocol протокол преобразования адресов)-позволяет автоматически получить MAC - адрес, если известен IP-адрес.
- ▶ Для определения MAC-адреса получателя по IP-адресу хост формирует широковещательный Ethernet-кадр, содержащий ARP-запрос (ARP-Request). Запрос содержит MAC и IP отправителя и IP получателя. Хост, обнаруживший свой IP в поле "сетевой адрес получателя", дописывает свой MAC-адрес и отправляет ARP-ответ (ARP-Reply). Получив искомый MAC-адрес, хост заносит его в ARP-кэш.

# Протокол преобразования адресов ARP

**ARP-запрос** отправляется на широковещательный **MAC-адрес ff:ff:ff:ff:ff:ff**.

В теле ARP-запроса поле с неизвестным значением Target MAC Address заполняется нулями.

**ARP-ответ** отправляется на MAC-адрес получателя, отправившего ARP-запрос.

В поле Sender MAC Address указывается запрашиваемый MAC-адрес устройства.

No.	Time	Source	Destination	Protocol	Length	Info
64	2.474396000	Hewlett-_90:7e:7f	Broadcast	ARP	42	who has 198.18.0.5? Tell 198.18.0.250
65	2.475678000	d8:b1:90:2e:e7:41	Hewlett-_90:7e:7f	ARP	60	198.18.0.5 is at d8:b1:90:2e:e7:41

Filter: arp Expression... Clear Apply Save

⊕ Frame 64: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0

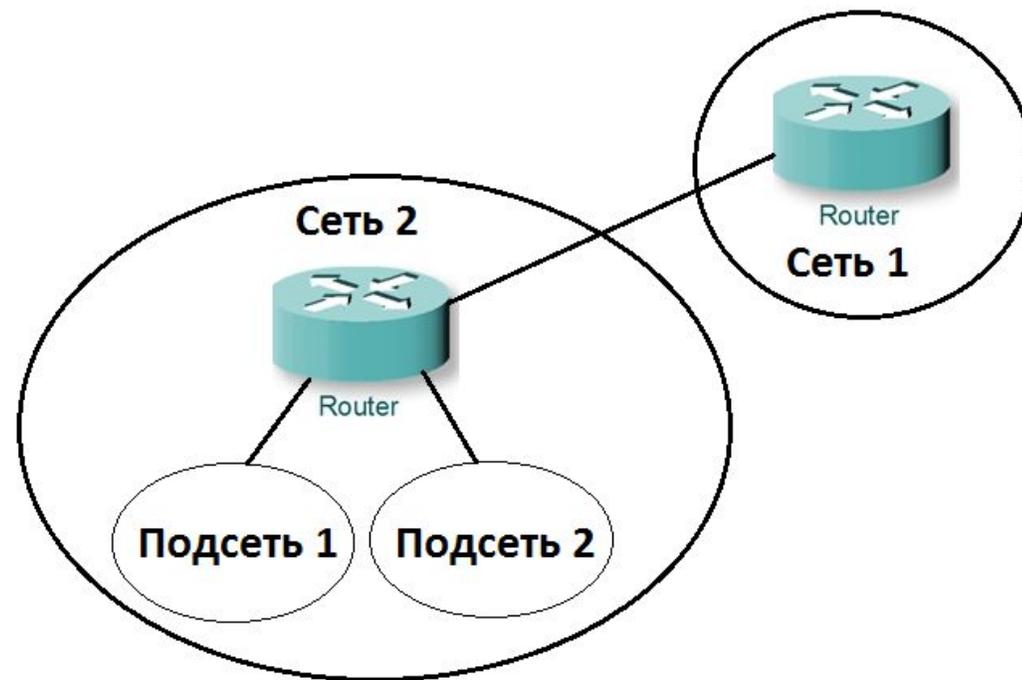
⊕ Ethernet II, Src: Hewlett-\_90:7e:7f (00:1a:4b:90:7e:7f), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

⊖ Address Resolution Protocol (request)

- Hardware type: Ethernet (1)
- Protocol type: IP (0x0800)
- Hardware size: 6
- Protocol size: 4
- Opcode: request (1)
- Sender MAC address: Hewlett-\_90:7e:7f (00:1a:4b:90:7e:7f)
- Sender IP address: 198.18.0.250 (198.18.0.250)
- Target MAC address: 00:00:00\_00:00:00 (00:00:00:00:00:00)
- Target IP address: 198.18.0.5 (198.18.0.5)

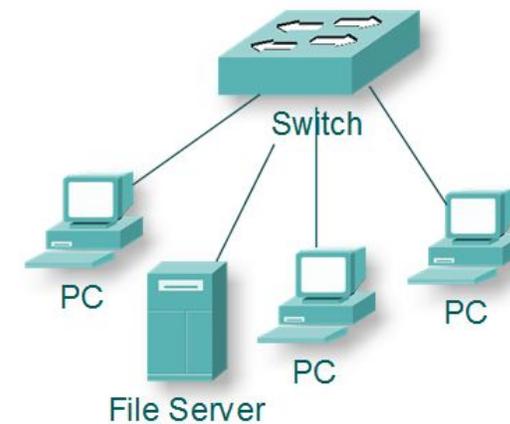
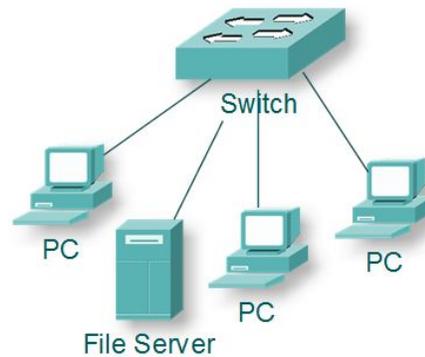
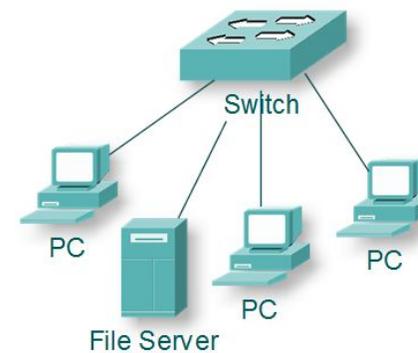
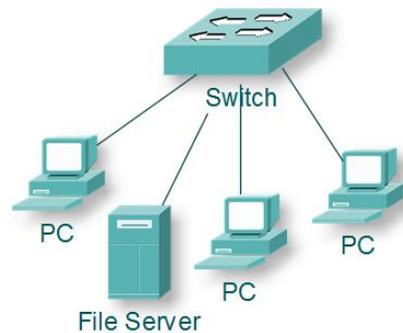
# Сегментация сетей

- ▶ Разделение сетей на подсети позволяет:
  - ▶ Повысить управляемость сетью
  - ▶ Ограничить широковещательные рассылки



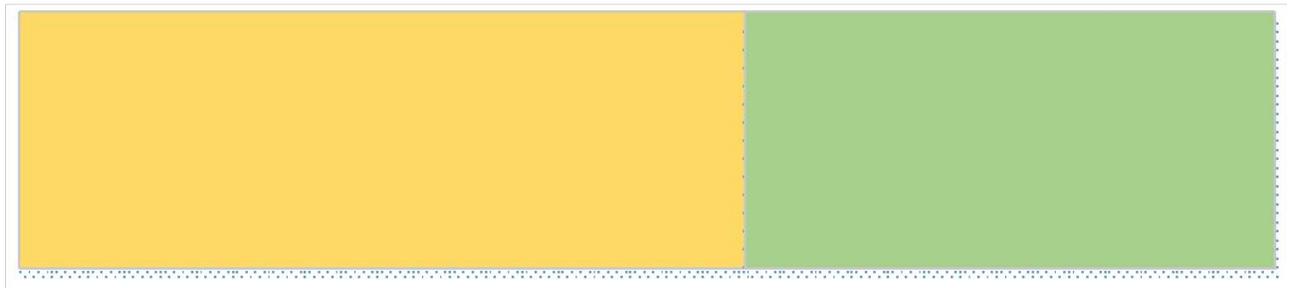
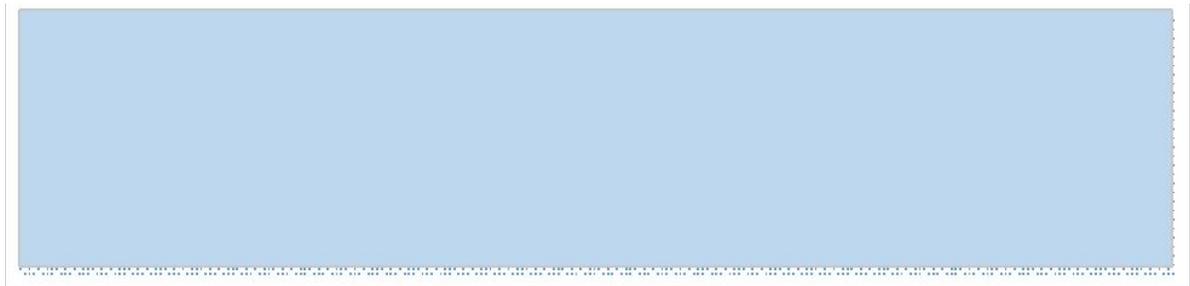
# Планирование сети

- ▶ При планировании подсети необходимо определить:
- ▶ Размер сети
- ▶ Количество узлов в каждой подсети
- ▶ Способы назначения узлам адресов



# Разбиение IPv4 сети на подсети

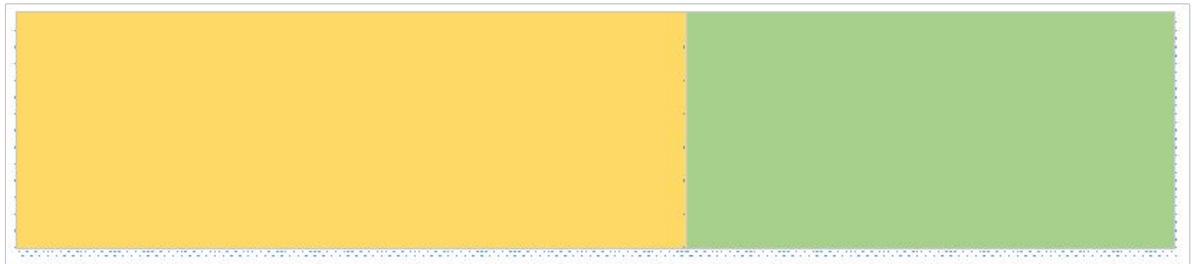
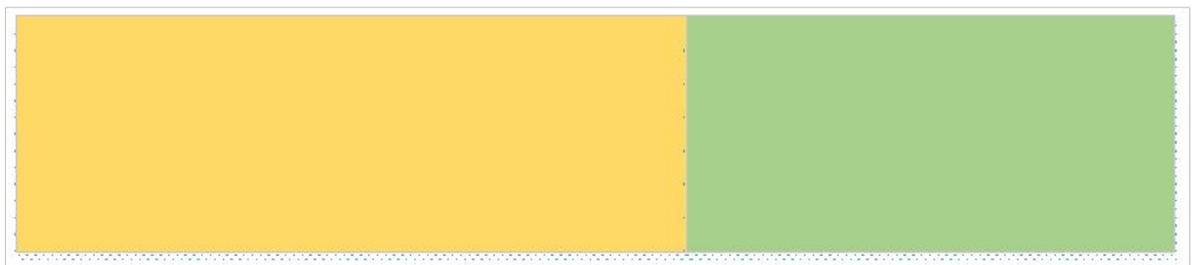
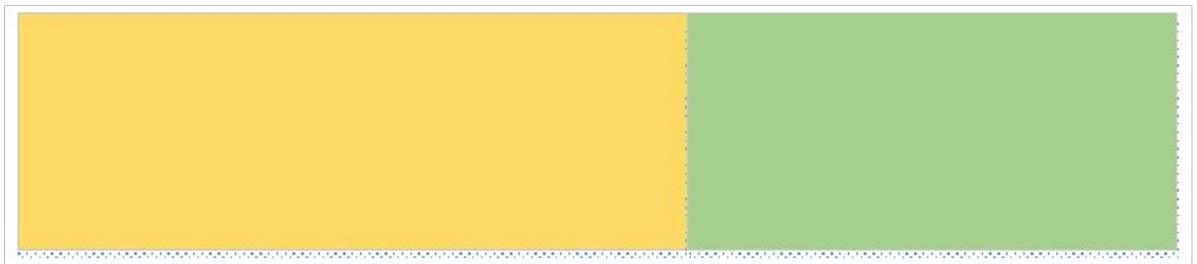
11111111111111111111111111111111



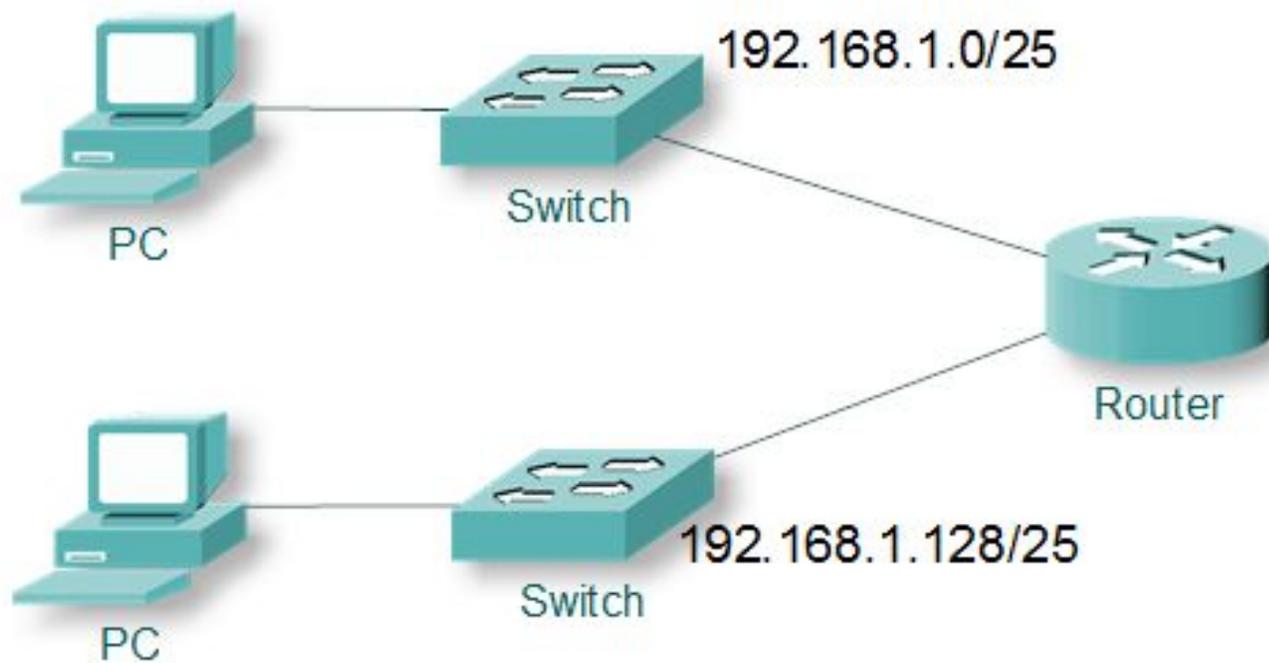


# Разбиение IPv4 сети на подсети

255.255.255.0



# Разбиение IPv4 сети на подсети



# Диапазон адресов подсети 192.168.1.0/25

- ▶ Сетевой адрес
  - ▶ 192.168.1.0 000 0000 = 192.168.1.0
- ▶ Адрес первого узла
  - ▶ 192.168.1.0 000 0001 = 192.168.1.1
- ▶ Адрес последнего узла
  - ▶ 192.168.1.0 111 1110 = 192.168.1.126
- ▶ Широковещательный адрес
  - ▶ 192.168.1.0 111 1111 = 192.168.1.127

# Диапазон адресов подсети 192.168.1.128/25

- ▶ Сетевой адрес
  - ▶  $192.168.1.1\ 000\ 0000 = 192.168.1.128$
- ▶ Адрес первого узла
  - ▶  $192.168.1.1\ 000\ 0001 = 192.168.1.129$
- ▶ Адрес последнего узла
  - ▶  $192.168.1.1\ 111\ 1110 = 192.168.1.254$
- ▶ Широковещательный адрес
  - ▶  $192.168.1.1\ 111\ 1111 = 192.168.1.255$

# Расчет количества подсетей

- ▶ Количество подсетей =  $2^n$ , где  $n$  = заимствованные биты
- ▶ 192.168.1.0 000 0000
  - ▶ 1 бит был заимствован
- ▶  $2^1 = 2$  подсети

# Расчет количества узлов

- ▶ Количество узлов =  $(2^n) - 2$ , где  $n$  = оставшиеся биты
- ▶ 192.168.1.0 000 0000
  - ▶ 7 бит остаются в поле узла
- ▶  $2^7 - 2 = 126$  допустимых узлов в каждой подсети

# УПС, задачка



- ▶ Дана сеть 192.168.1.0/24
- ▶ Создать 4 подсети
- ▶ Рассчитать количество узлов

# Решение

Заимствование 2 бит позволит создать 4 подсети:  $2^2 = 4$  подсети

Исходный адрес: 192.168.1.00 00 0000  
Маска: 255.255.255.00 00 0000

При заимствовании 2 бит создаются 4 подсети:

Сеть 0: 192.168.1.00 00 0000 = 192.168.1.0/26

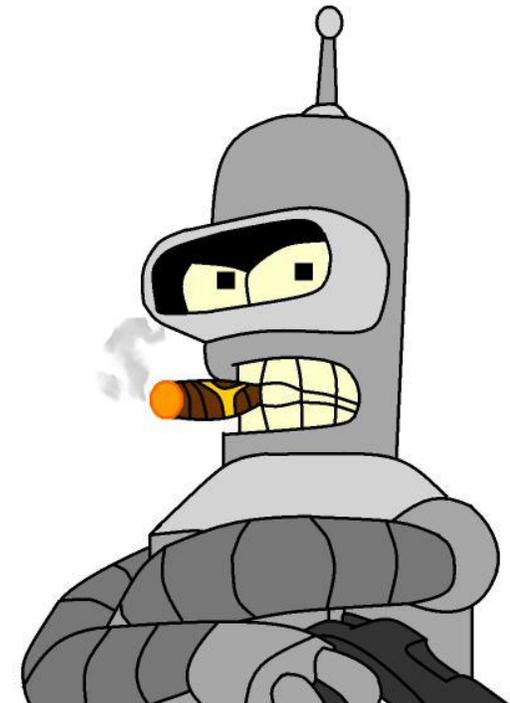
Сеть 1: 192.168.1.01 00 0000 = 192.168.1.64/26

Сеть 2: 192.168.1.10 00 0000 = 192.168.1.128/26

Сеть 3: 192.168.1.11 00 0000 = 192.168.1.192/26

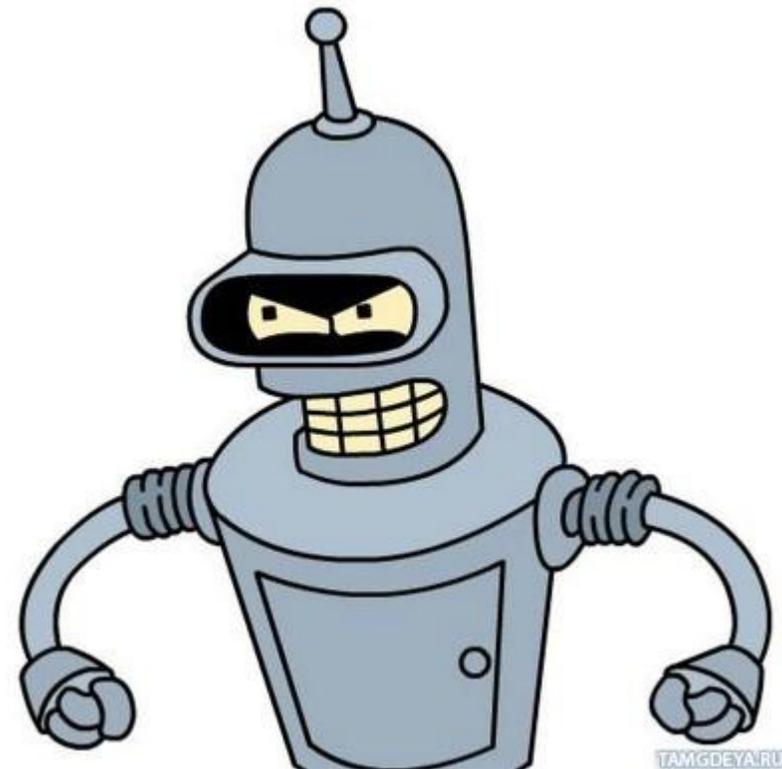
Все 4 подсети имеют одинаковую маску подсети

Маска: 255.255.255.11 00 0000 = 255.255.255.192



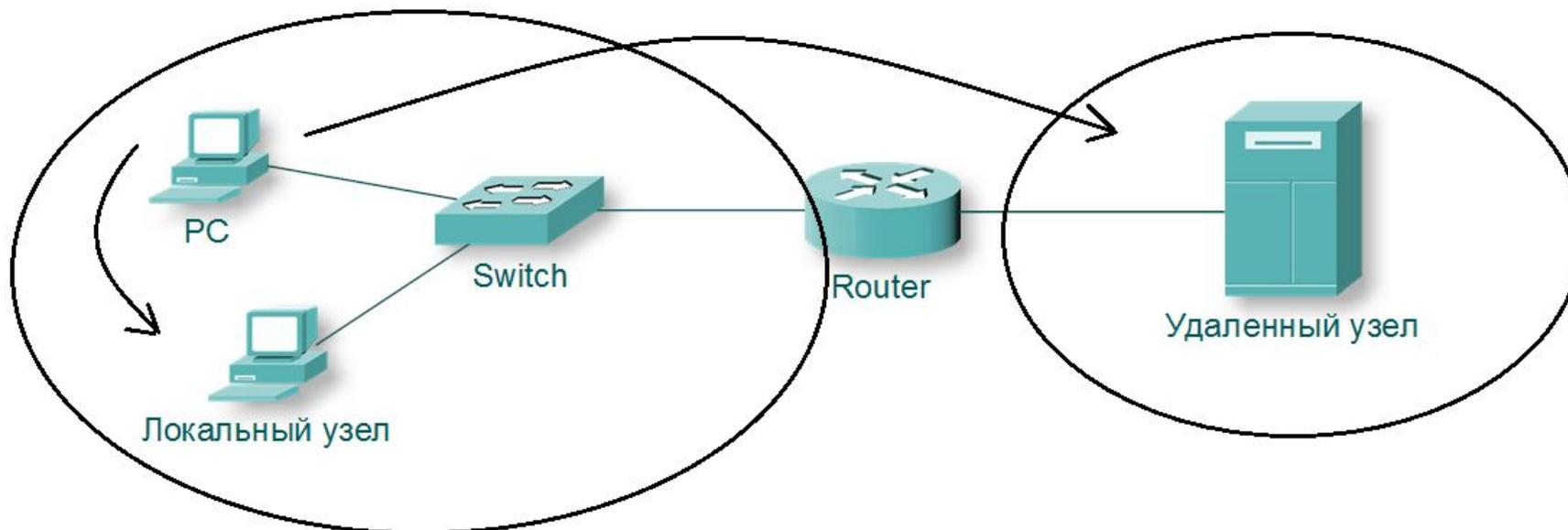
# Решение: расчет количества узлов

- ▶ Количество узлов =  $(2^n) - 2$ , где  $n$  = оставшиеся биты в узловой части
  - ▶ 192.168.1.00 00 0000
    - ▶ 6 бит остаются в поле узла
- ▶  $(2^6) - 2 = 62$  допустимых узла в каждой подсети



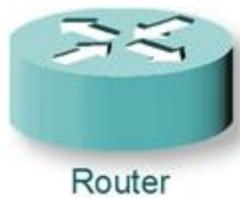
# Маршрутизация

- ▶ **Маршрутизация** (*Routing*) — процесс определения наиболее эффективного пути от одного устройства к другому
  - ▶ Статическая – маршруты задаются администраторами
  - ▶ Динамическая – построение маршрутов автоматически, при помощи протоколов маршрутизации



# Маршрутизатор

- ▶ Маршрутизатор – специализированный сетевой компьютер, имеющий два или более сетевых интерфейсов и пересылающий пакеты данных между различными сегментами сети.
- ▶ Устройство сетевого уровня



# Функции маршрутизатора

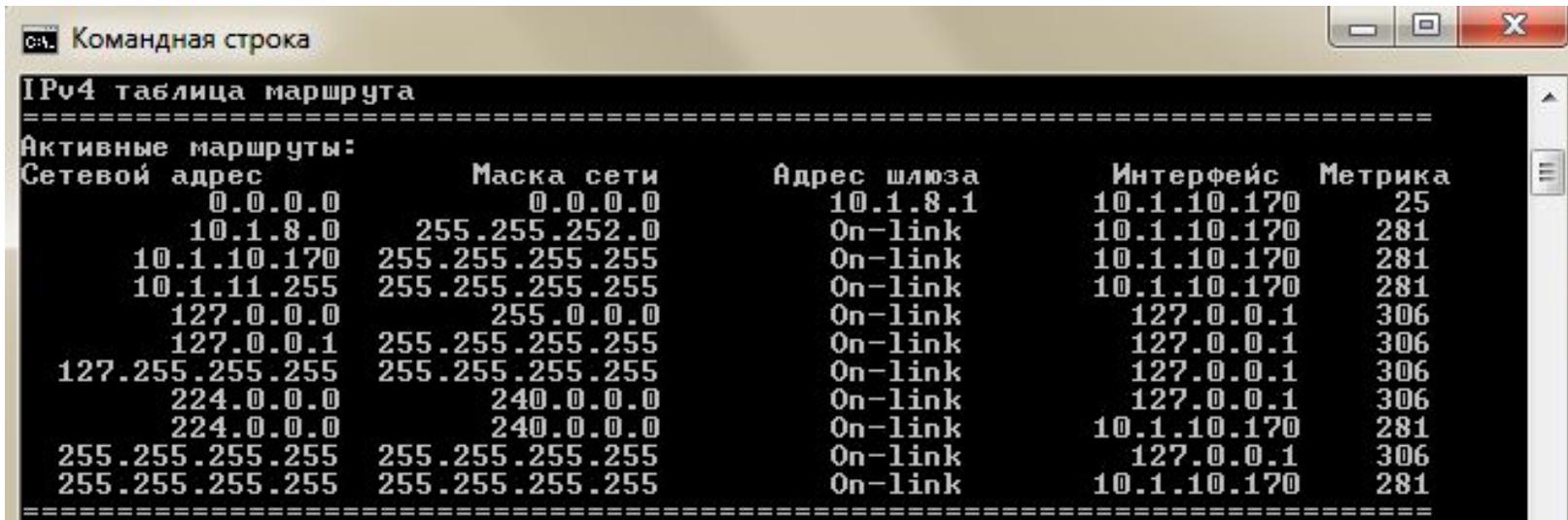
- ▶ Поддерживает таблицы маршрутизации и обменивается информацией об изменениях в топологии сети с другими маршрутизаторами
- ▶ Когда пакеты приходят на один из интерфейсов, маршрутизатор, руководствуясь таблицей маршрутизации, должен определить, куда именно следует отправить пакет
- ▶ Перенаправляет пакеты на выбранный интерфейс
- ▶ Ограничивает широковещательные рассылки

# Таблица маршрутизации

- ▶ **Таблица маршрутизации** — электронная таблица или база данных, хранящаяся на маршрутизаторе или сетевом компьютере, которая описывает соответствие между адресами назначения и интерфейсами, через которые следует отправить пакет данных до следующего маршрутизатора.
- ▶ Содержит:
- ▶ **Сеть назначения**
- ▶ **Метрика, связанная с сетью назначения**
- ▶ **Шлюз для подключения к сети назначения**

# Таблица маршрутизации

- ▶ В Windows для отображения таблицы маршрутизации узла можно использовать команду **route print** или **netstat -r**.



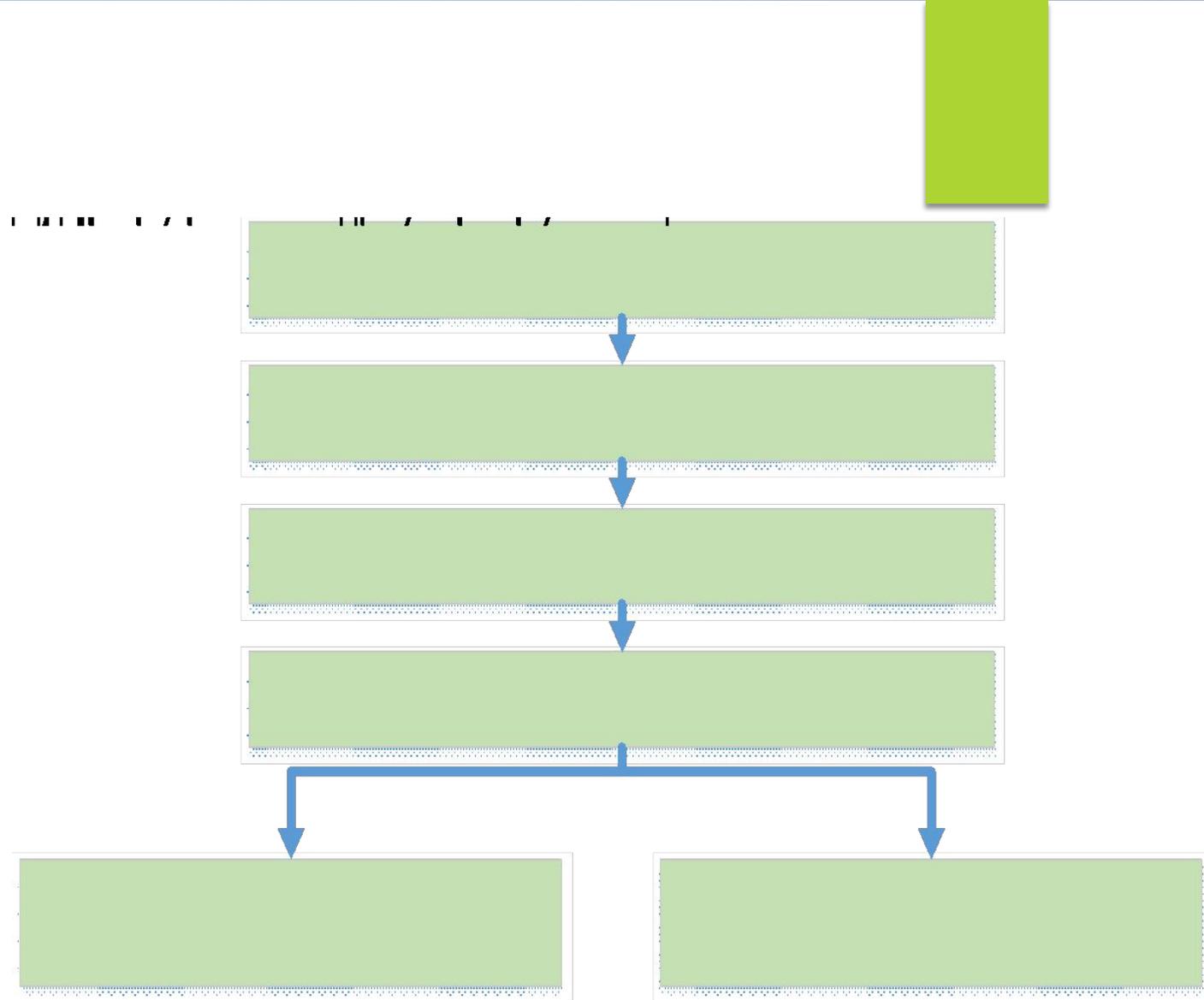
```
Командная строка
IPv4 таблица маршрута
=====
Активные маршруты:
Сетевой адрес      Маска сети        Адрес шлюза       Интерфейс         Метрика
0.0.0.0            0.0.0.0           10.1.8.1          10.1.10.170      25
10.1.8.0           255.255.252.0     On-link           10.1.10.170      281
10.1.10.170        255.255.255.255  On-link           10.1.10.170      281
10.1.11.255        255.255.255.255  On-link           10.1.10.170      281
127.0.0.0          255.0.0.0         On-link           127.0.0.1        306
127.0.0.1          255.255.255.255  On-link           127.0.0.1        306
127.255.255.255    255.255.255.255  On-link           127.0.0.1        306
224.0.0.0          240.0.0.0         On-link           127.0.0.1        306
224.0.0.0          240.0.0.0         On-link           10.1.10.170      281
255.255.255.255    255.255.255.255  On-link           127.0.0.1        306
255.255.255.255    255.255.255.255  On-link           10.1.10.170      281
=====
```

# Определение пути

- ▶ **Сеть с прямым подключением** – Directly connected
- ▶ **Удаленная сеть** – IP-адрес назначения пакета принадлежит удалённой сети
- ▶ **Маршрут не определен** – если IP-адрес назначения пакета не принадлежит подключённой или удалённой сети, маршрутизатору нужно определить, доступен ли шлюз по умолчанию

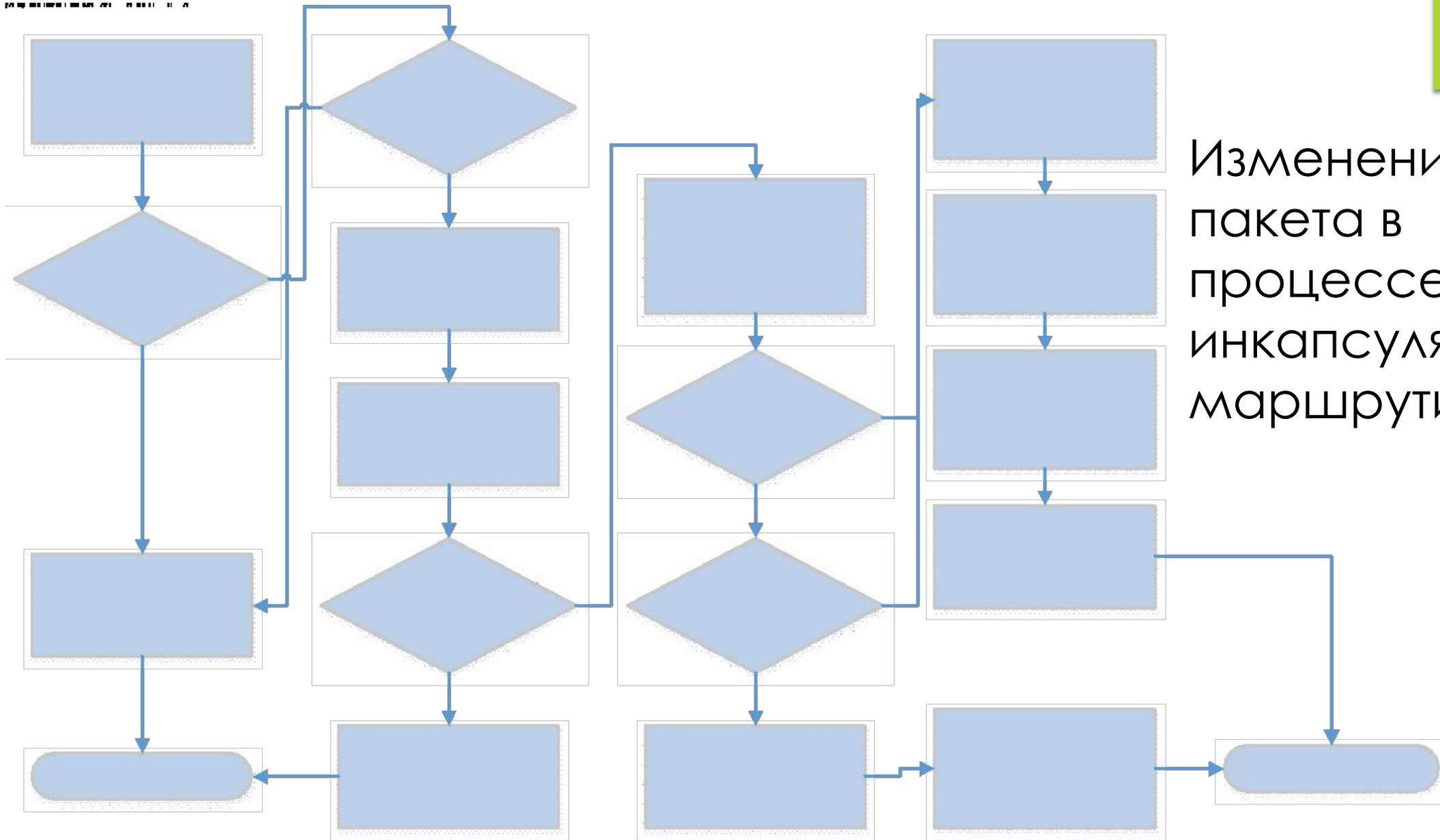
# ШЛЮЗ ПО УМОЛЧАНИЮ

*Default gateway* — сетевой шлюз, на который пакет отправляется в том случае, если маршрут к сети назначения пакета не известен (не задан явным образом в таблице маршрутизации хоста).



# Протокол маршрутизации

- ▶ **Протокол маршрутизации** — это протокол, который предоставляет механизмы обмена маршрутной информацией.
- ▶ Сообщения протокола маршрутизации передаются между маршрутизаторами.
- ▶ Протокол маршрутизации позволяет маршрутизаторам обмениваться информацией друг с другом для обновления записей и поддержки таблиц маршрутизации.
- ▶ Примеры протоколов маршрутизации TCP/IP:
  - ▶ Протокол маршрутной информации (Routing Information Protocol — RIP);
  - ▶ Протокол маршрутизации внутреннего шлюза (Interior Gateway Routing Protocol — IGRP);
  - ▶ Усовершенствованный протокол маршрутизации внутреннего шлюза (Enhanced Interior Gateway Routing Protocol — EIGRP);
  - ▶ Протокол первоочередного обнаружения кратчайших маршрутов (Open Shortest Path First — OSPF).



Изменение пакета в процессе инкапсуляции в маршрутизаторе



# Базовая конфигурация маршрутизатора

- ▶ Присвоить имя маршрутизатору
- ▶ Конфигурация баннера
- ▶ Обеспечить защиту доступа
  - ▶ Настройка пароля для доступа в привилегированный режим
  - ▶ Настройка пароля для доступа через консольный порт
  - ▶ Настройка пароля для доступа через VTY

# Базовая конфигурация маршрутизатора

- ▶ Чтобы обеспечить доступность интерфейса маршрутизатора необходимо:
- ▶ Настроить на интерфейсе адрес с маской подсети
- ▶ Активировать интерфейс

```
R1(config)# interface gigabitethernet 0/0
```

```
R1(config-if)#description Link to LAN 1
```

```
R1(config-if)# ip address 192.168.10.1 255.255.255.0
```

```
R1(config-if)#no shutdown
```

```
R1(config-if)#exit
```

```
R1(config)#
```

# Базовая конфигурация маршрутизатора

- ▶ **Интерфейс loopback** - это логический интерфейс внутри маршрутизатора. Он не назначается физическому порту, поэтому его нельзя подключить к другому устройству. Он считается программным интерфейсом, который автоматически переводится в состояние UP во время работы маршрутизатора.

```
Router(config)# interface loopback number
```

```
Router(config-if)# ip address ip-address subnet-mask
```

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

# Проверка настроек маршрутизатора

- ▶ **show ip interface brief** – отображает краткую информацию обо всех интерфейсах, включая IPv4-адрес интерфейса и текущее рабочее состояние
- ▶ **show ip route** – отображает содержимое таблицы маршрутизации IPv4, которая хранится в ОЗУ. В Cisco IOS 15 активные интерфейсы должны быть указаны в таблице маршрутизации с двумя связанными с ними записями, которые определены кодом «C» (подключён) или «L» (локальный).
- ▶ **show running-config interface *interface-id*** – отображает команды, настроенные на указанном интерфейсе
- ▶ **show interfaces** – отображает информацию об интерфейсе и счётчик потока пакетов для всех интерфейсов на устройстве.
- ▶ **show ip interface** – отображает информацию об IPv4 для всех интерфейсов маршрутизатора.

# Настройка статического маршрута

- ▶ Статические маршруты настраиваются с помощью команды глобальной конфигурации **ip route**.
- ▶ Синтаксис команды:
  - ▶ Router(config)# **ip route** *network-address subnet-mask* {*ip-address* | *exit-intf*}
  - ▶ R1(config)# **ip route 192.168.2.0 255.255.255.0 GigabitEthernet 0/1**
- ▶ Статический маршрут определяется в таблице маршрутизации посредством кода «S».

# Проверка настроек статического маршрута

- ▶ Команды **ping** и **tracert**
- ▶ **show ip route**
- ▶ **show ip route static**
- ▶ **show ip route** *network*

# Маршрут по умолчанию

- ▶ Статический маршрут по умолчанию — это маршрут, которому соответствуют все пакеты.
- ▶ Вместо хранения всех маршрутов ко всем сетям в таблице маршрутизации маршрутизатор может хранить один маршрут по умолчанию, представляющий любую сеть, отсутствующую в таблице маршрутизации.
- ▶ **ip route 0.0.0.0 0.0.0.0 { ip-address | exit-intf }**

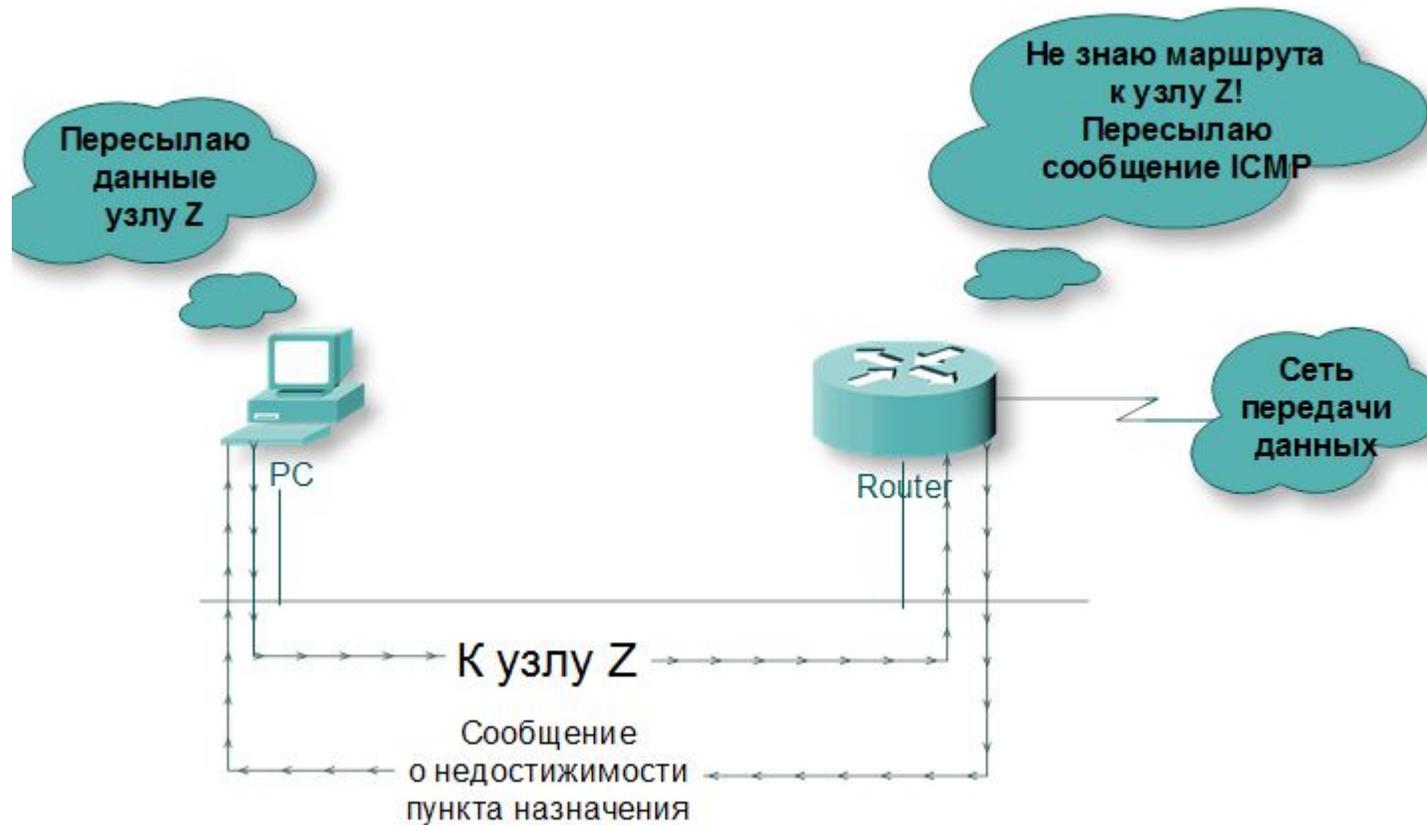
# Протокол ICMP

- ▶ *ICMP (Internet Control Message Protocol) - протокол управляющих сообщений в сети Internet*
  - ▶ Компенсирует неспособность протокола IP гарантированно доставлять данные
- ▶ Является механизмом отправки сообщений об ошибках для протокола IP

# Протокол ICMP

- ▶ **Сообщения о недостижимости** пункта назначения могут включать в себя следующие виды информации:
- ▶ **Сеть недостижима** - это сообщение обычно свидетельствует об ошибках в маршрутизации или адресации;
- ▶ **Узел недостижим** - это сообщение обычно свидетельствует об ошибках при доставке, например, об ошибочной маске подсети;
- ▶ **Протокол недоступен (недостижим)** - это сообщение обычно свидетельствует о том, что пункт назначения не поддерживает протокол верхнего уровня, указанный в пакете;
- ▶ **Порт недостижим** - это сообщение обычно свидетельствует о том, что TCP
- ▶ порт (сокеты) недоступен.

# Протокол ICMP



# Протокол ICMP

- ▶ Использование команды ping для проверки достижимости пункта назначения

```
S3>ping 192.168.99.11
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 192.168.99.11, timeout is 2 seconds:
```

```
....!
```

```
Success rate is 20 percent (1/5), round-trip min/avg/max = 0/0/0 ms
```

```
S3>ping 192.168.99.11
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 192.168.99.11, timeout is 2 seconds:
```

```
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/1 ms
```

```
S3>ping 192.168.99.50
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 192.168.99.50, timeout is 2 seconds:
```

```
.....
```

```
Success rate is 0 percent (0/5)
```

# Протокол ICMP

Команда **traceroute** используется для просмотра пути, по которому пакеты доходят до пункта назначения

```
hou-rtr-1#traceroute 10.32.25.1
Type escape sequence to abort.
Tracing the route to 10.212.25.234
 0 10.2.38.24 [AS 100] 0 msec 0 msec 0 msec
 1 10.2.38.24 [AS 100] 0 msec 0 msec 0 msec
 2 10.212.47.16 [AS 1358] [MPLS: Label 54321] Exp 9] 1 msec 0 msec 1 msec
 3 10.212.247.57 [AS 12345] 12 msec 14 msec 15 msec
 4 10.212.32.66 22 msec 23 msec 24 msec
 5 10.212.25.234 [AS 4321] 40 msec * 36 msec
 6 10.32.25.1 39 msec 38 msec 40 msec
hou-rtr-1#
```



Спасибо за внимание!

