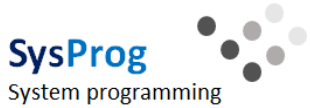
Московский Государственный Технический Университет им. Н.Э. Баумана

СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ

к.т.н.
Папулин Сергей Юрьевич
papulin_bmstu@mail.ru

Москва, 2021

Сетевые протоколы



Модель ISO

Модель ISO

А

Б

7	Уровень приложений		Уровень приложений
6	Уровень представлений		Уровень представлений
		Данные (data)	
5	Сеансовый уровень		Сеансовый уровень
		Сегменты (segments)	
4	Транспортный уровень		Транспортный уровень
		Пакеты (packets)	
3	Сетевой уровень		Сетевой уровень
		Фреймы (frames)	
2	Канальный уровень		Канальный уровень
		Биты (bits)	
1	Физический уровень		Физический уровень

Примеры протоколов

Уровень приложений (Application)	Доступ к сетевым ресурсам (HTTP, FTP, SMTP)
Уровень представлений (Presentation)	Трансляция, сжатие, шифрование данных (ASCII, JPEG)
Сеансовый уровень (Session)	Установка, управление и остановка сеансов (RPC, PAP)
Транспортный уровень (Transport)	Надежная доставка, коррекция ошибок (TCP, UDP)
Сетевой уровень (Network)	Логическая адресация, доставка пакетов от источника к месту назначения (IPv4, IPv6, IPsec)
Канальный уровень (Data link)	Организует байты в кадры; MAC адресация и доставка (PPP, Ethernet, L2TP, ARP)
Физический уровень (Physical)	Передача битов по линии связи; спецификации линий

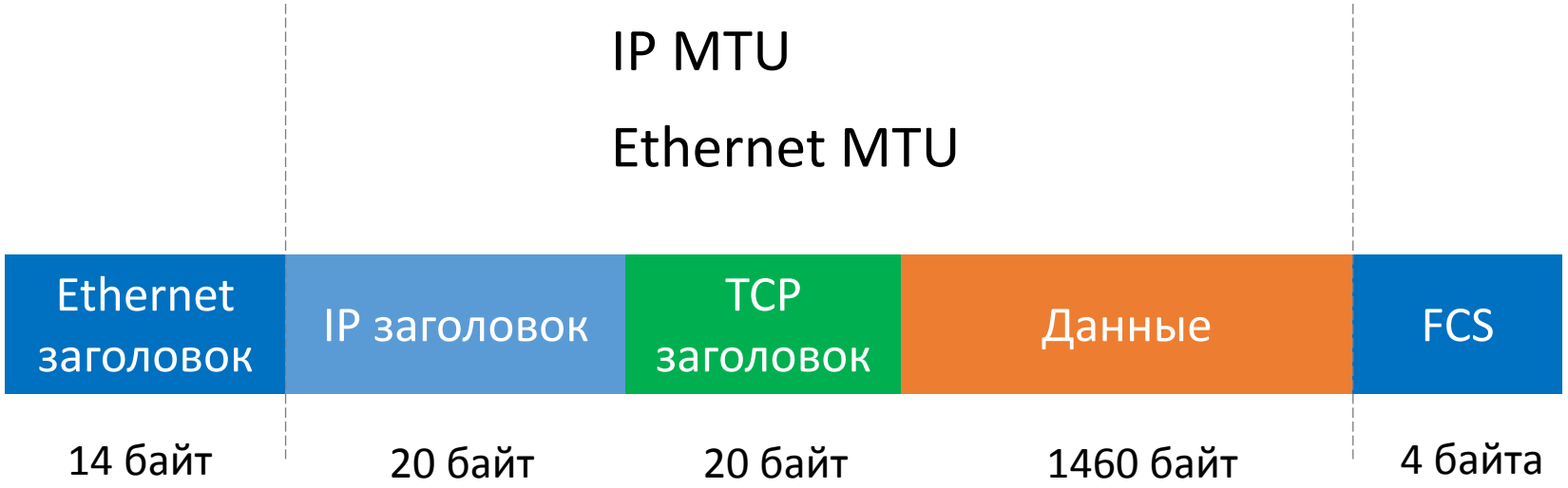
TCP/IP


Уровень приложений (Application)
Уровень представлений (Presentation)
Сеансовый уровень (Session)
Прикладной уровень
Транспортный уровень (Transport)
Транспортный уровень
Сетевой уровень (Network)
Сетевой уровень
Канальный уровень (Data link)
Физический уровень (Physical)
Уровень сетевых интерфейсов
Формирование кадров Ethernet 802.3, 802.11
Интерфейс среды передачи

Передача данных

Передача данных в сети

Максимальная единица передачи (Maximum transmission unit - MTU) – максимальный размер пакета, измеряемых в байтах, который можно передать по сети. Сообщение, которое больше чем MTU, перед отправкой разбивается на меньшие пакеты, что затормаживает передачу данных

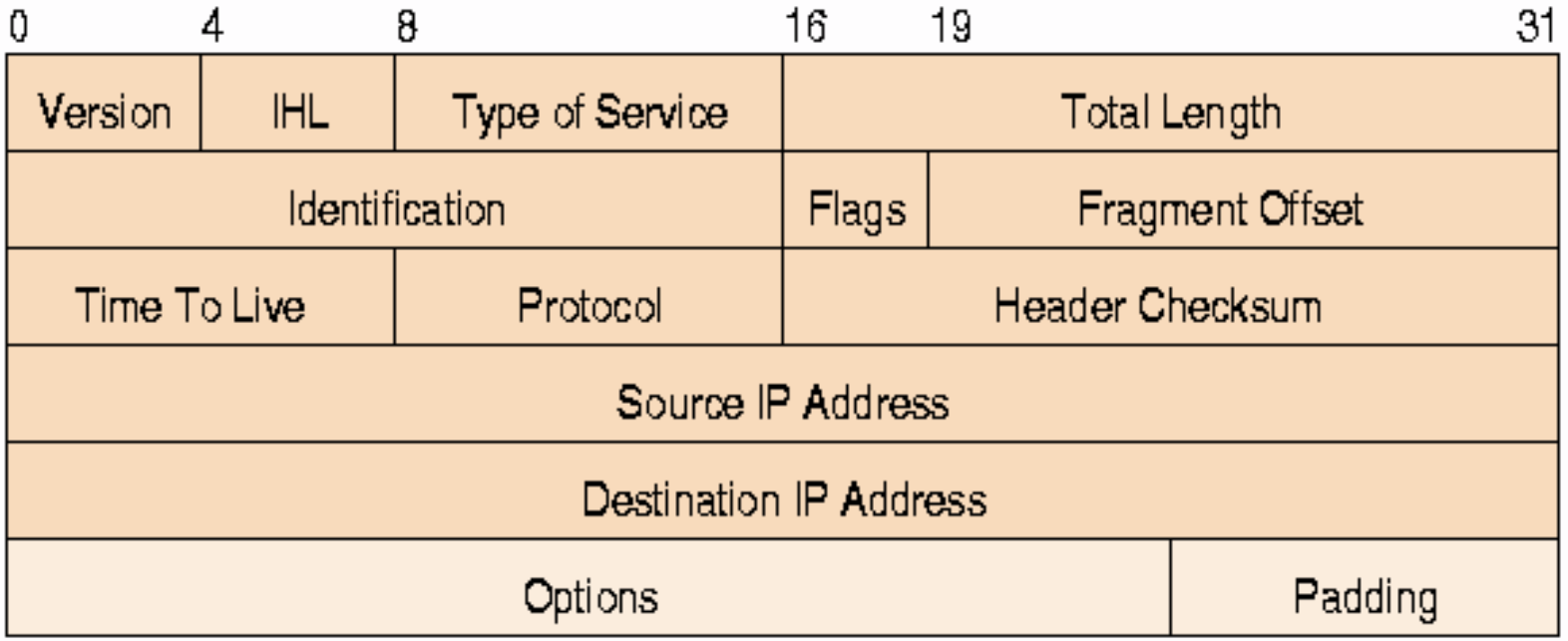


MTU:
Ethernet – 1500 байт
ATM – 48 байт
PPP – 500-2000 байт

Протокол сетевого уровня

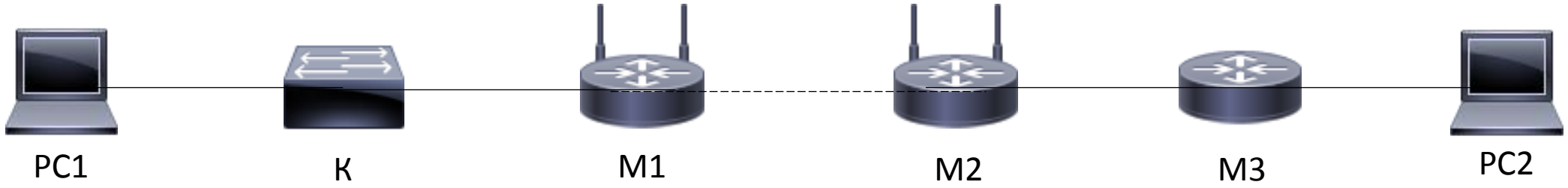
Интернет протокол (IP) обеспечивает передачу данных через различные сети, маршрутизацию (выбор оптимального пути), контроль ошибок, приоритетов и размеров пакетов

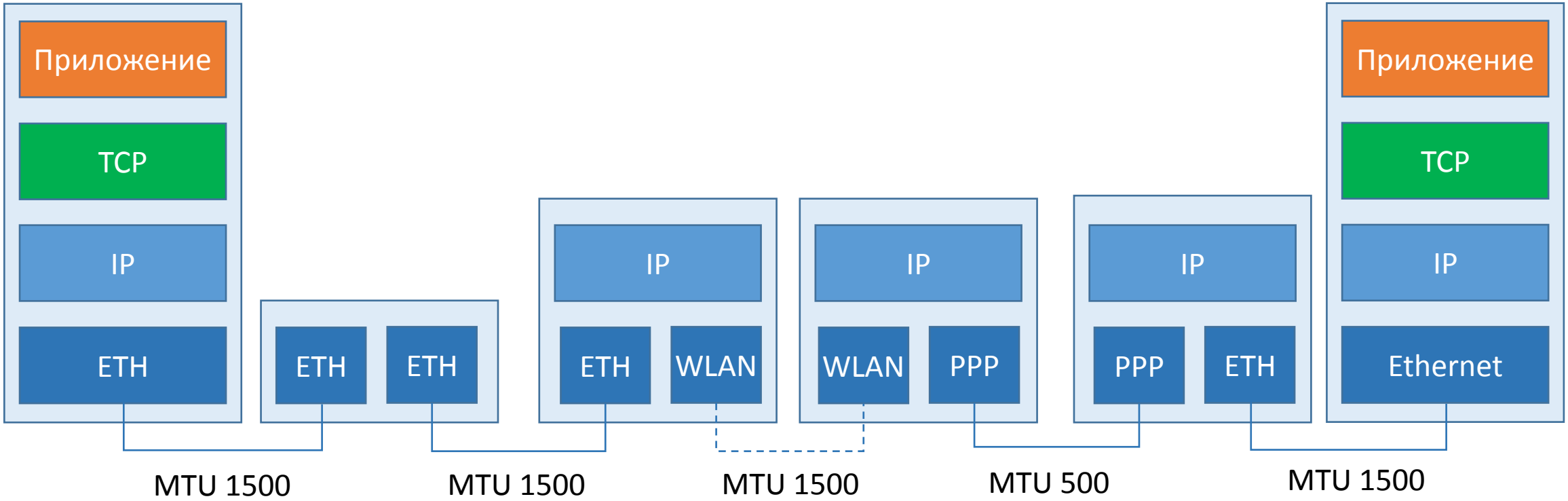
Заголовок 20 байт



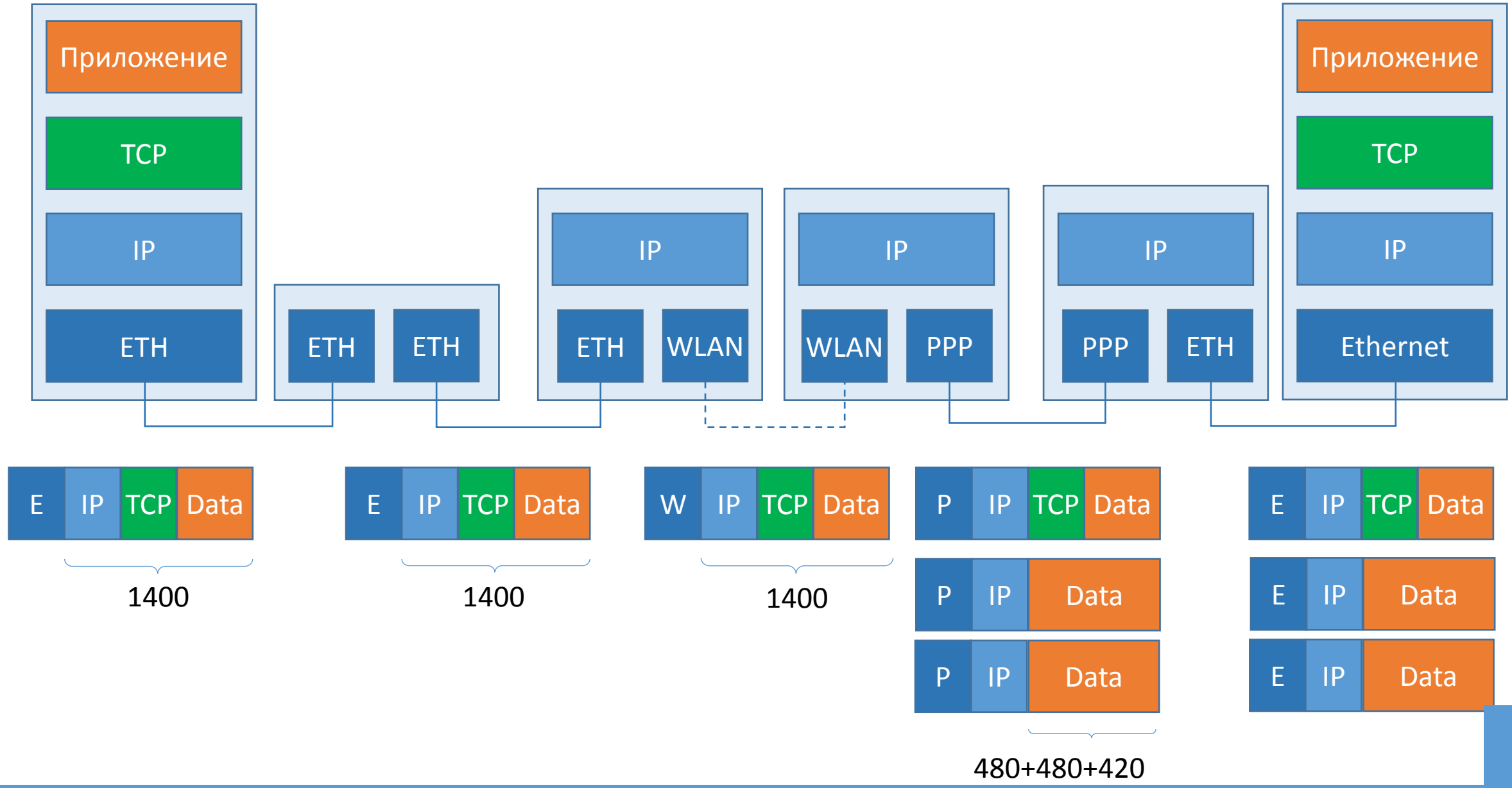
Максимальный размер пакета $2^{16} - 1 = 65{,}535$ байт

Передача данных в сети

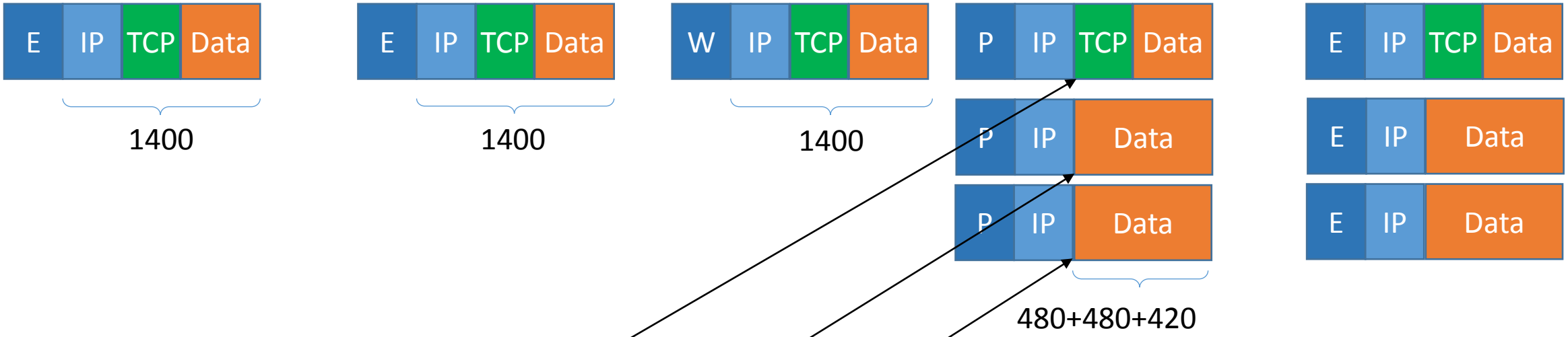




Передача данных в сети



Передача данных в сети



Flag -> bit 2 -> MF=1 (More Fragments)

Flag -> bit 2 -> MF=1 (More Fragments); Fragment offset

Flag -> bit 2 -> MF=0 (Last Fragments); Fragment offset

IPv4 vs IPv6

IPv4: 32-битный адрес ($2^{32} \approx 4.3 \cdot 10^9$)

IPv6: 128-битный адрес ($2^{128} \approx 3.4 \cdot 10^{38}$)

Максимальная единица передачи в соединении (Link Maximum Transfer Unit - LMTU) – максимальная размер пакета, который может быть передан через сетевое соединение.

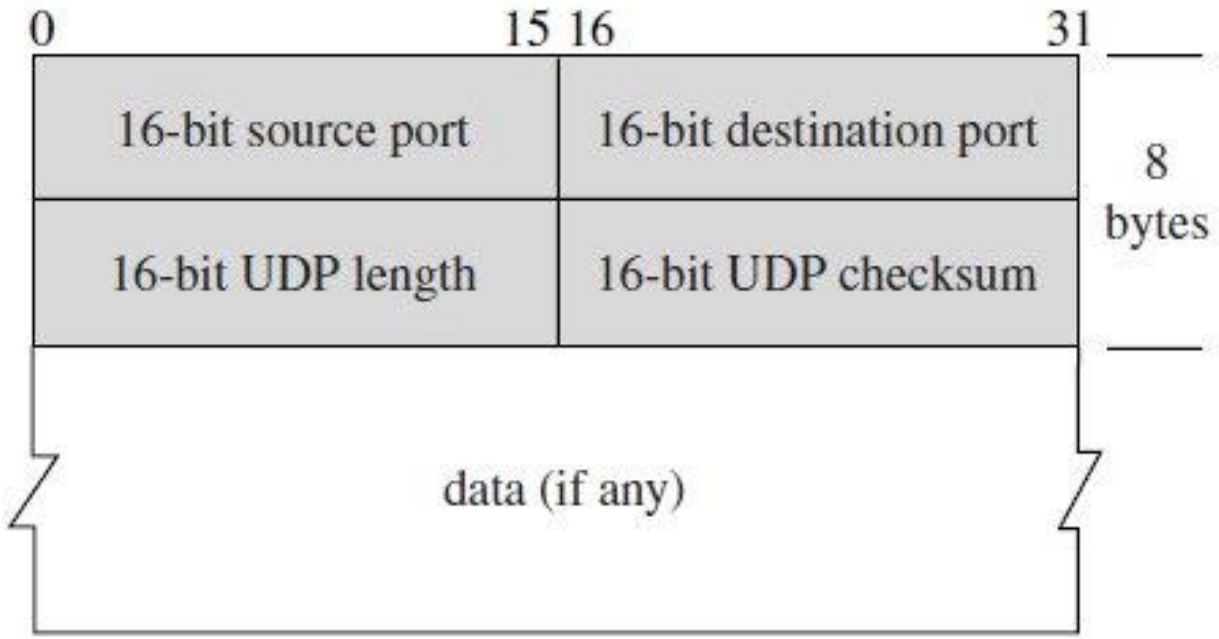
Максимальная единица передачи в маршруте (Path Maximum Transfer Unit - PMTU) – минимальная LMTU из всех сетевых соединений на маршруте между сетевыми устройствами

Методы PMTUD (Path MTU Discovery)

Протоколы транспортного уровня

Протоколы транспортного уровня. UDP

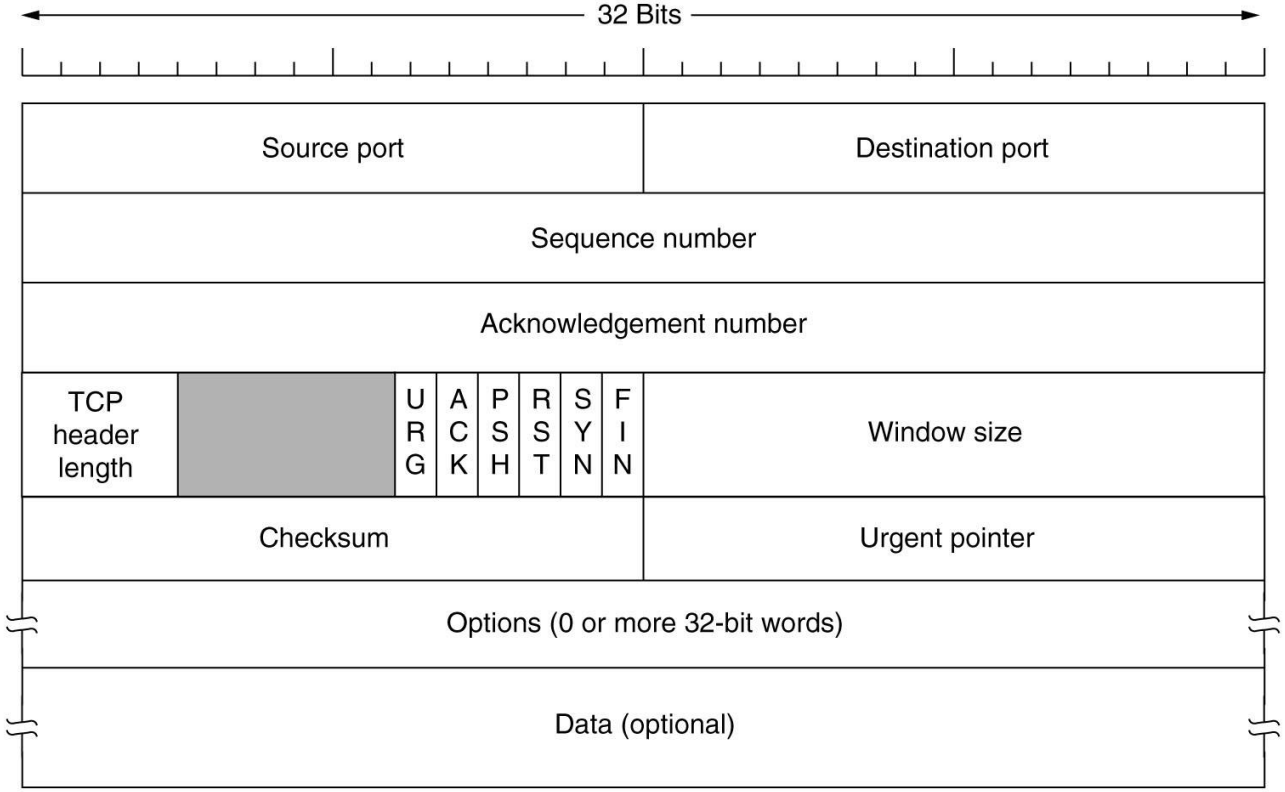
Протокол пользовательских датаграмм (User Datagram Protocol – UDP) позволяет передавать данные между хостами без установки сессии, предоставляет порт для уровня приложения в модели TCP/IP



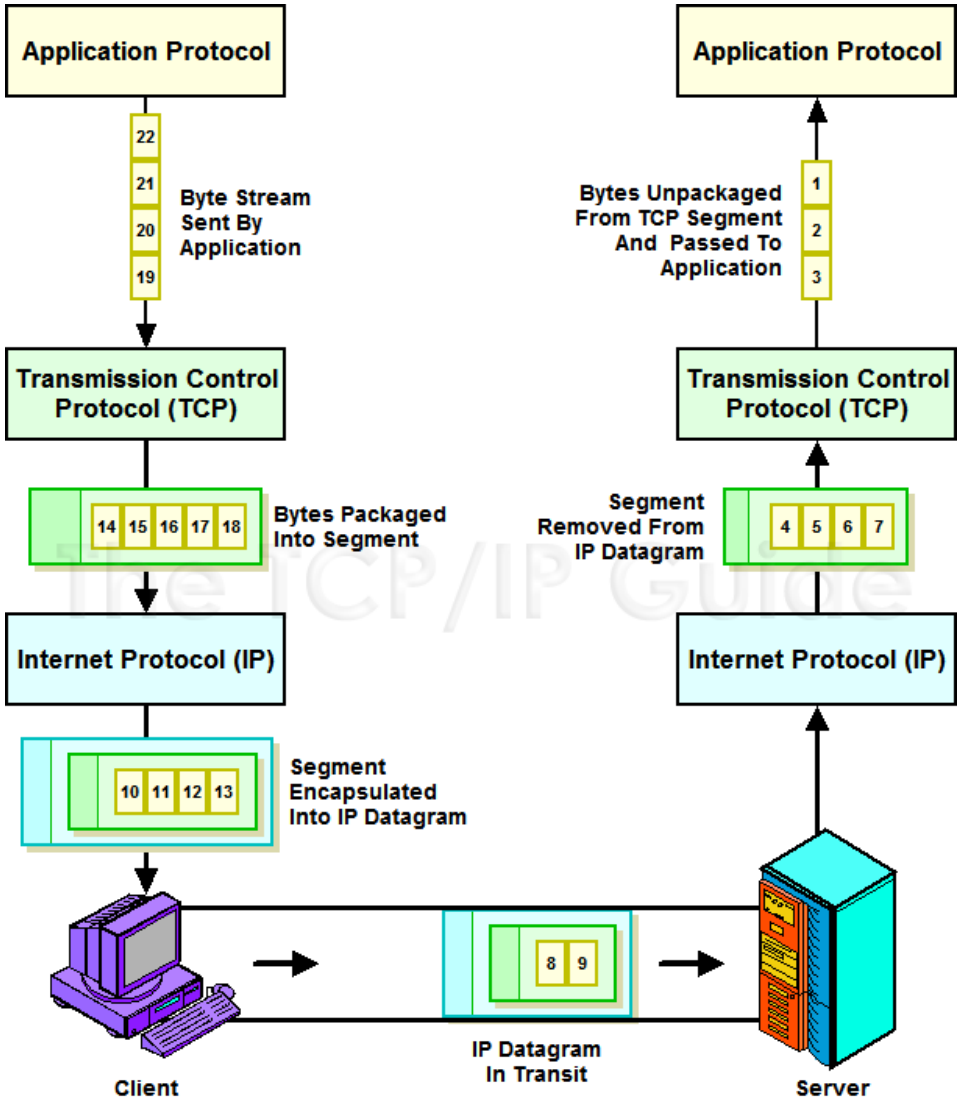
Протоколы транспортного уровня. TCP

Протокол управления передачей (Transmission Control Protocol) устанавливает сессию между узлами, обеспечивает порт доставки для вышележащих уровней, организацию сегментов данных, управляет потоком и перегрузками.

Минимальный размер заголовка 20 байт



Протоколы транспортного уровня. TCP



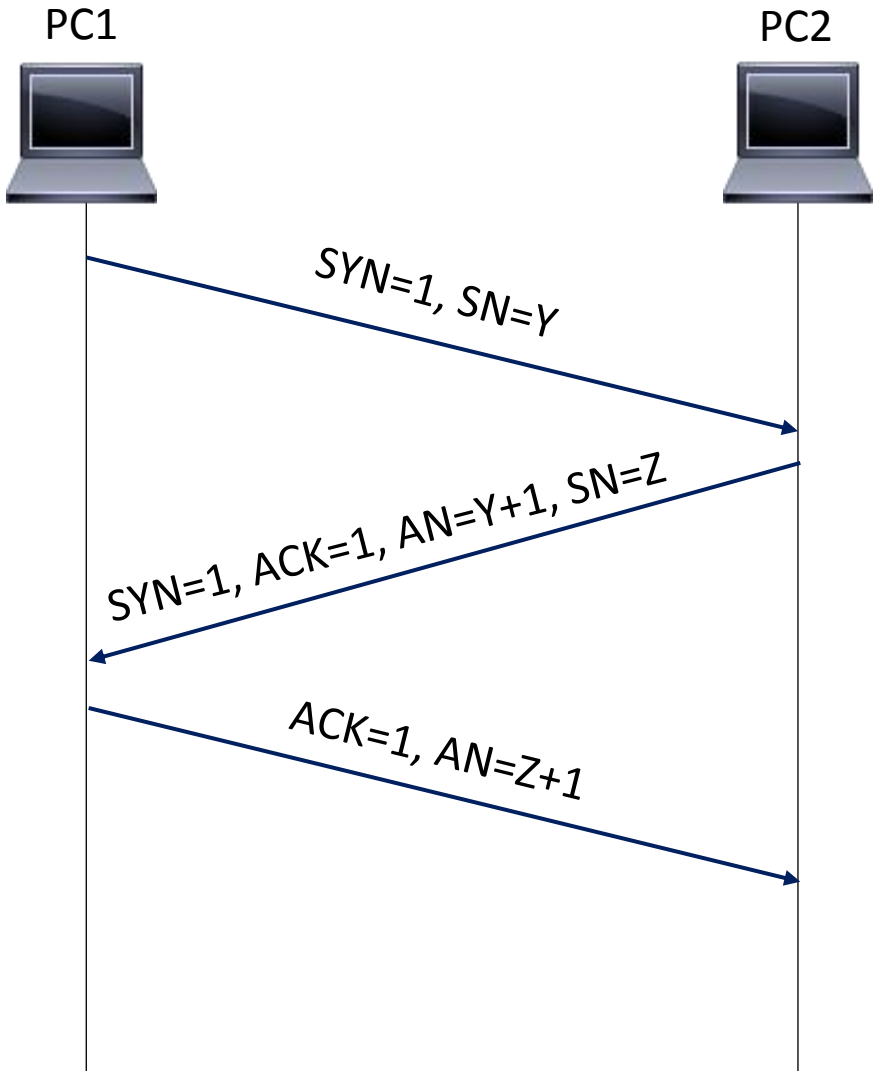
http://www.tcpipguide.com/free/t_TCPDataHandlingandProcessingStreamsSegmentsandSequ-2.htm

Установка соединения TCP

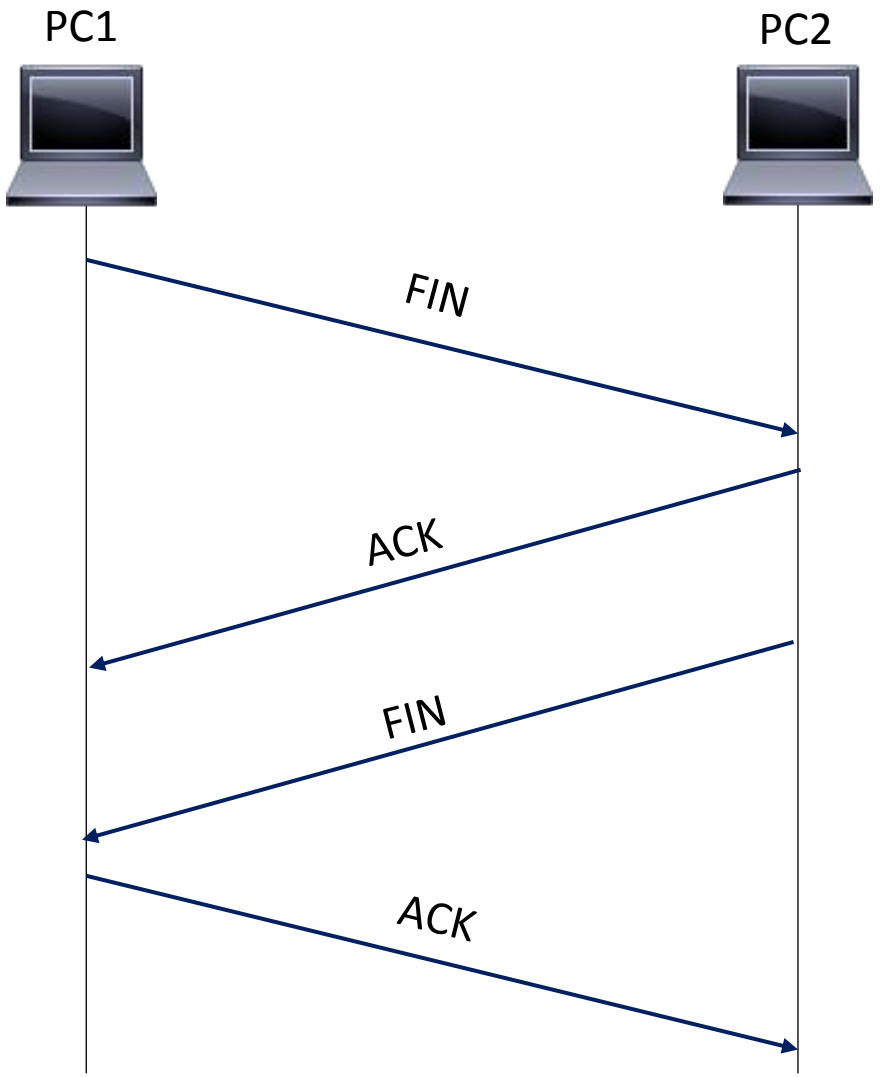
Шаг 1. PC1 -> PC2
Флаг SYN устанавливается равным 1, генерируется SN (Sequence Number). Выбирается 32-битное число Y

Шаг 2. PC2 -> PC1
Флаг SYN устанавливается равным 1, флаг подтверждения ACK устанавливается равным 1, назначается подтверждающее число AN = Y +1, генерируется SN = Z

Шаг 3. PC2 -> PC1
Флаг подтверждения ACK устанавливается равным 1, назначается подтверждающее число AN = Z +1



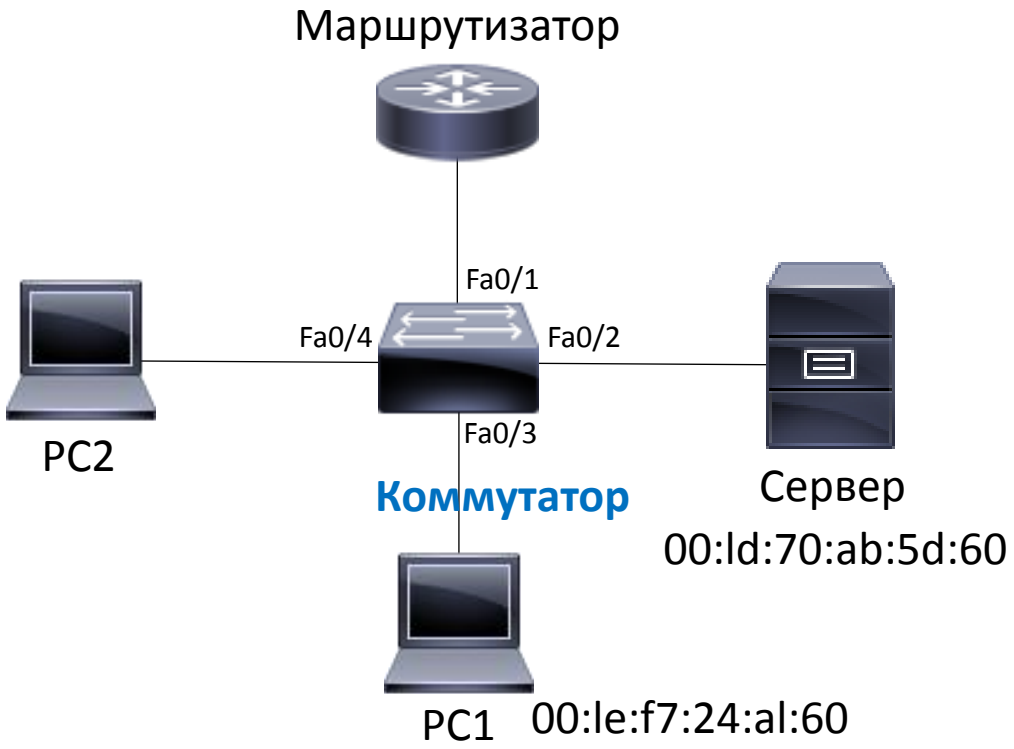
Завершение соединения TCP



Протоколы канального уровня

Коммутация

Коммутация – процесс соединения сетевых устройств. При этом используются коммутаторы, которые обеспечивают передачу данных на канальном уровне между различными сетевыми устройства.



Коммутатор

Коммутаторы на основе Ethernet

Фрейм/кадр – единица передачи данных в сетях Ethernet.

Преамбула	SFD	Адрес назначения	Адрес отправителя	Тип Ethernet	Данные	FCS	Gap
7 байт	1 байт	6 байт	6 байт	2 байт	42 - 1500 байт	4 байт	12 байт

Коммутаторы на основе Ethernet

MAC-адрес (Media Access Control) – уникальный идентификатор сетевого устройства на базе Ethernet.

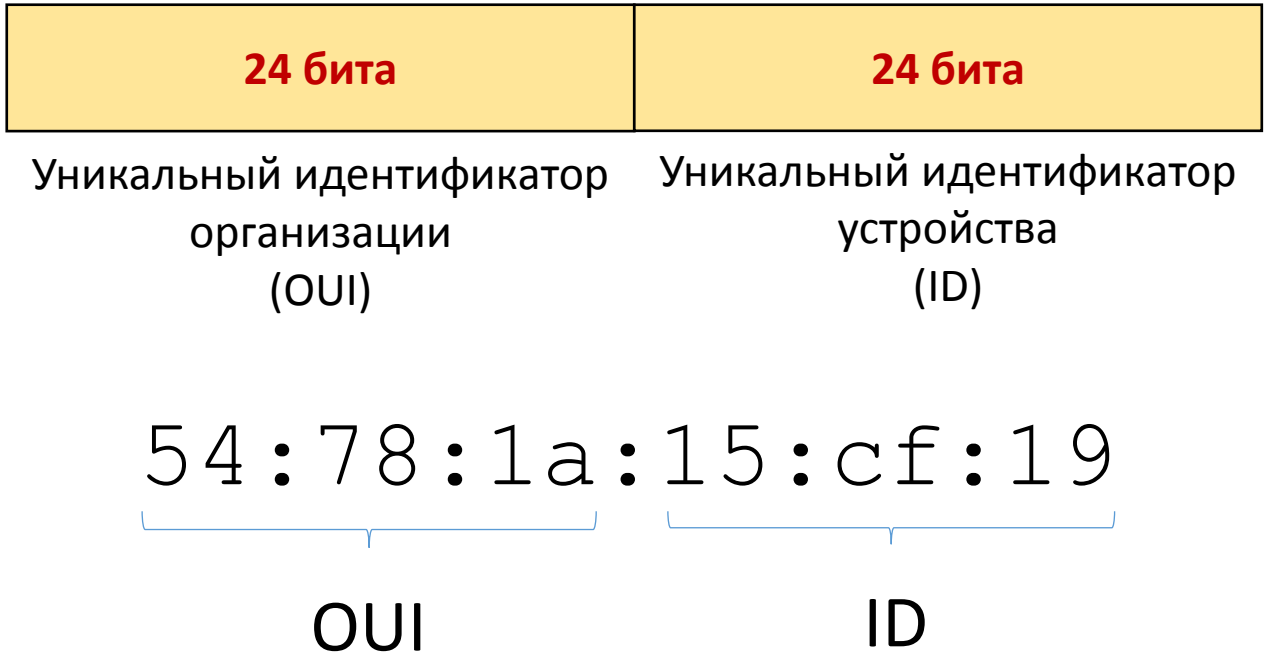
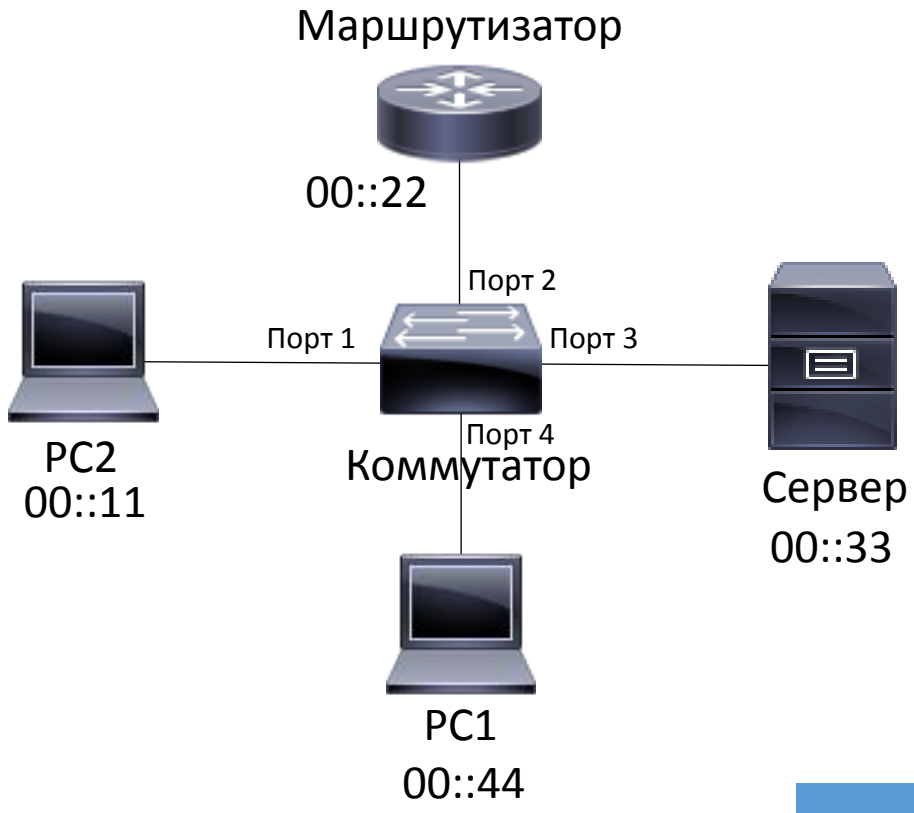


Таблица MAC-адресов

Таблица MAC-адресов (MAC address table) используется в Ethernet коммутаторах для определения адресата передачи трафика в локальной сети.

Port	Mac Address
1	00:00:00:00:00:11
2	00:00:00:00:00:22
3	00:00:00:00:00:33
4	00:00:00:00:00:44

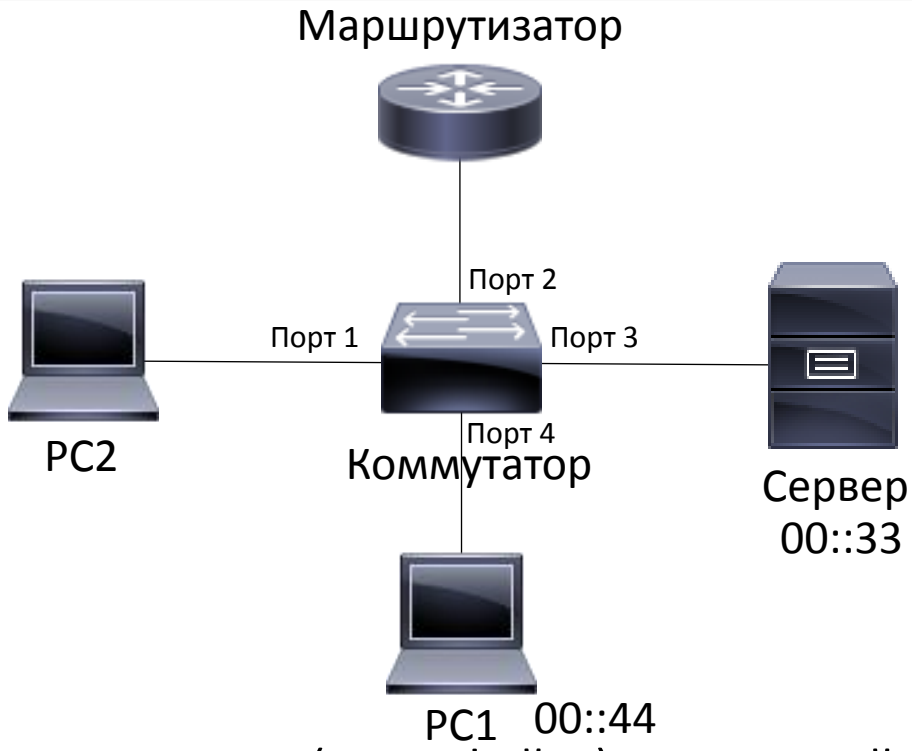


Формирование MAC-таблицы (1)

Шаг 1. Все устройства включены, но не обмениваются данными

Шаг 2. PC1 намеревается передать фрейм серверу с известным MAC

Шаг 3. При получении фрейма коммутатор создаст новое поле в таблице MAC-адресов для MAC-адреса PC1 (PC1 -> порт 4)



Шаг 4. Коммутатор просмотрит таблицу адресов с целью поиска порта (интерфейса), на который необходимо направить трафик.

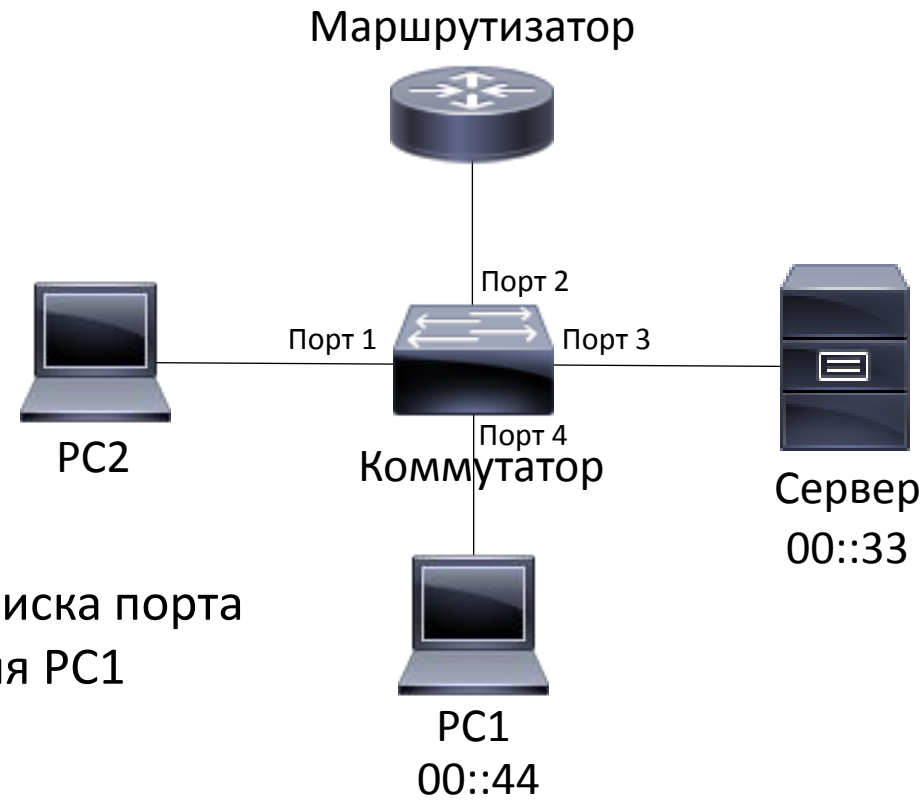
- Если MAC-адрес назначения не обнаружен в таблице, коммутатор подает фрейм на все свои порты (интерфейсы) (кроме PC1)
- Переданный PC1 фрейм получат все устройства, включая Сервер

Формирование MAC-таблицы (2)

Шаг 5. Если Сервер решит ответить PC1, то отправит новый фрейм, который получит коммутатор

Шаг 6. При получении фрейма коммутатор создаст новое поле в таблице MAC-адресов для MAC-адреса Сервера (Сервер -> порт 3)

Шаг 4. Коммутатор просмотрит таблицу адресов с целью поиска порта (интерфейса), на который необходимо направить трафик для PC1



- Так как таблица уже содержит MAC-адрес и порт, к которому подключён PC1, коммутатор отправит фрейм от Сервера на порт соответствующий PC1

Процесс повторяется пока устройства пересылают данные друг другу.

Формирование MAC-таблицы (3)

Таблица MAC-адресов для рассмотренного примера

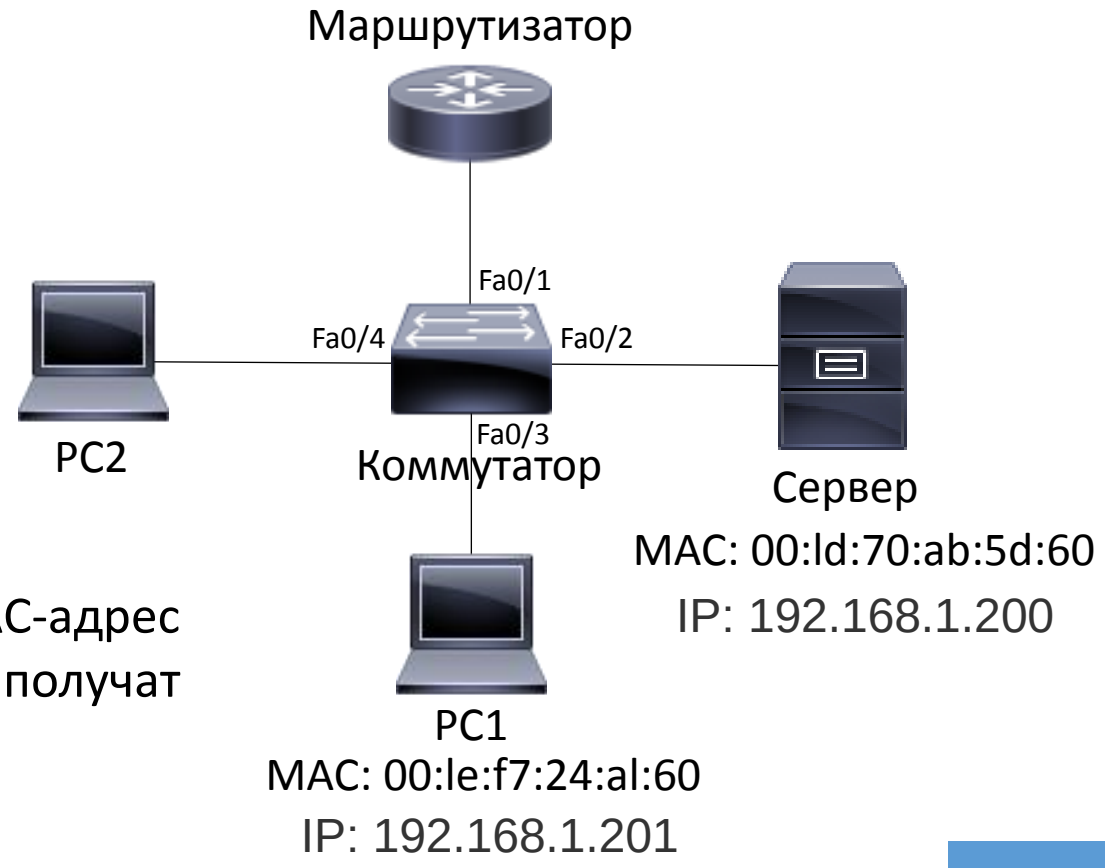
Vlan	Mac Address	Type	Ports
1	00:00:00:00:00:33	DYNAMIC	3
1	00:00:00:00:00:44	DYNAMIC	4

Строка хранится в таблице MAC-адресов в течение небольшого времени (по умолчанию в Cisco 5 мин.), затем строка будет удалена

Протокол определения адреса ARP (1)

Протокол определения адреса (Address Resolution Protocol – ARP) – сетевой протокол, который используется для определения MAC-адреса по IP-адресу. ARP используется для связывания сетевого и канального уровней

Если PC1 намеревается передать данные Серверу и знает его IP-адрес, то для определения его MAC-адреса используется ARP.



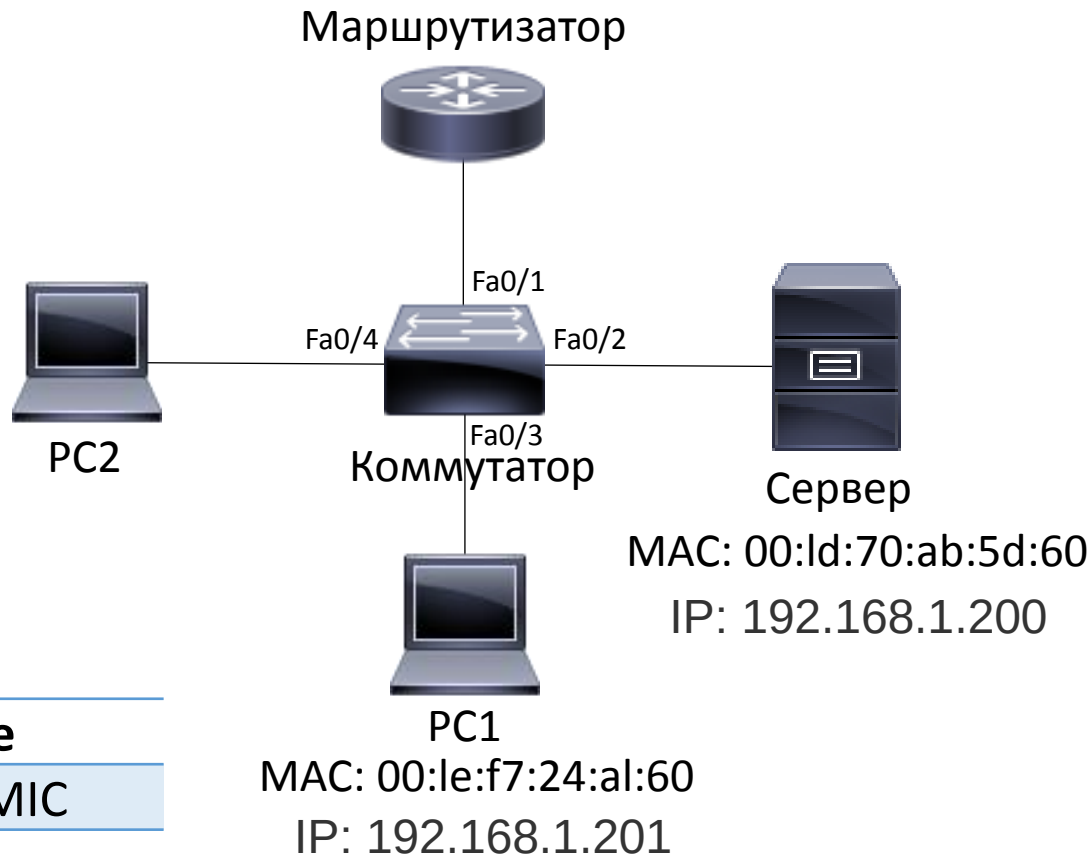
Шаг 1. Для этого PC1 формирует ARP-запрос, в который размещает IP-адрес Сервера.

При этом используется широковещательный MAC-адрес (ff.ff.ff.ff.ff.ff). Таким образом все локальные устройства получат ARP запрос от PC1.

Протокол определения адреса ARP (2)

Шаг 2. Сервер, получив ARP-запрос от PC1, формирует ARP-ответ, в который включает свой MAC-адрес.

Шаг 3. После того как получен ARP-ответ от Сервера, PC1 записывает его в ARP-таблицу и может использовать MAC-адрес Сервера для передачи данных.



Internet Address	Mac Address	Type
192.168.1.200	00:ld:70:ab:5d:60	DYNAMIC

Время жизни записи в ARP-таблице значительно больше, чем в таблице MAC-адресов, и по умолчанию составляет в Cisco 4 часа.

Протокол определения адреса

ARP-запроса

```
⊞ Frame 299 (42 bytes on wire, 42 bytes captured)
⊞ Ethernet II, Src: Ibm_43:49:97 (00:11:25:43:49:97), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊟ Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    [Is gratuitous: False]
    Sender MAC address: Ibm_43:49:97 (00:11:25:43:49:97)
    Sender IP address: 192.168.1.1 (192.168.1.1)
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.1.254 (192.168.1.254)
```

ARP-ответ

```
⊞ Frame 300 (60 bytes on wire, 60 bytes captured)
⊞ Ethernet II, Src: Cisco_35:1a:d0 (00:19:55:35:1a:d0), Dst: Ibm_43:49:97 (00:11:25:43:49:97)
⊟ Address Resolution Protocol (reply)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (0x0002)
    [Is gratuitous: False]
    Sender MAC address: Cisco_35:1a:d0 (00:19:55:35:1a:d0)
    Sender IP address: 192.168.1.254 (192.168.1.254)
    Target MAC address: Ibm_43:49:97 (00:11:25:43:49:97)
    Target IP address: 192.168.1.1 (192.168.1.1)
```

Протоколы сетевого уровня

Маршрутизация

- Маршрутизатор соединяет одну сеть с другой и обеспечивает передачу пакетов между ними

- Основные функции маршрутизатора:
  - определение наилучшего пути для оправки пакета
  - передача пакета получателю

- Маршрутизаторы передают пакет, используя информацию в таблице маршрутизации. Записи в таблице маршрутизации могут быть получены двумя способами:
  - статическим (ручным)
  - динамическим

Маршрутизация

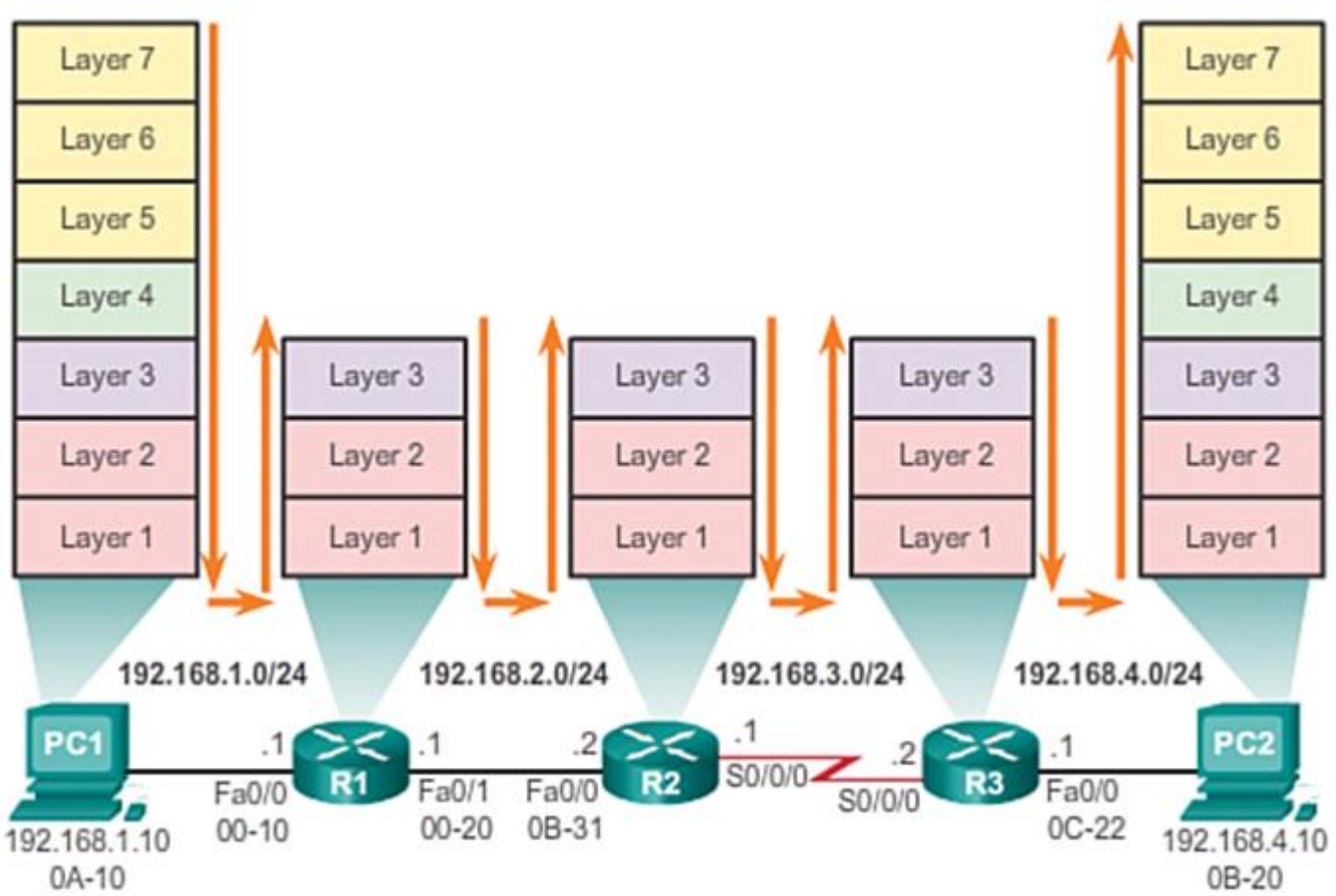
При получении пакета маршрутизатор:

1. Деинкапсулирует пакет (уровень 3), удаляя заголовок кадра (уровень 2)

2. Проверяет IP-адрес назначения, чтобы найти наилучший путь в таблице маршрутизации

3. Если найден путь, то пакет инкапсулируется в кадр и передается на соответствующий выходной интерфейс

Маршрутизация

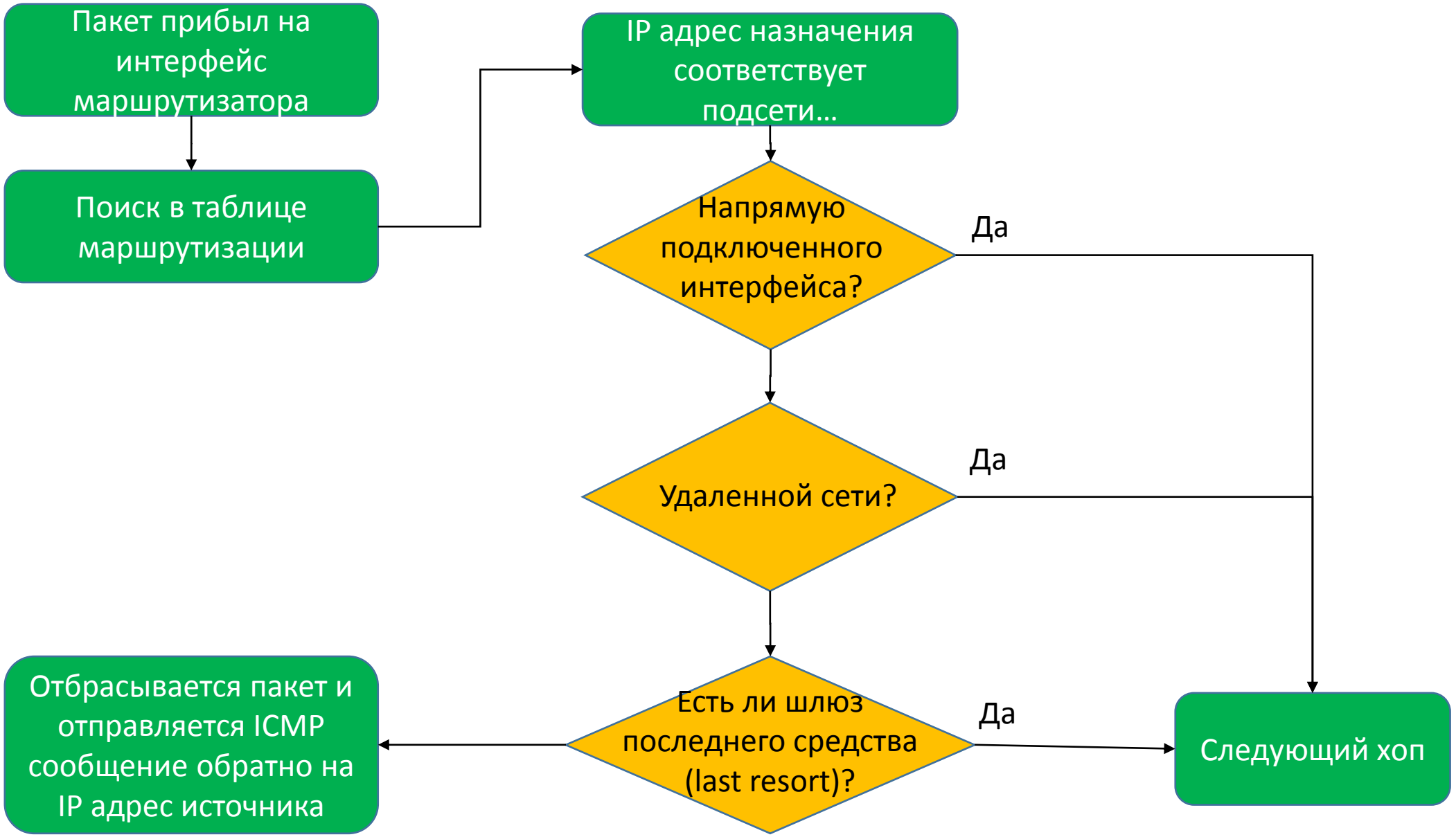


Проверка IP-адреса назначения:

1. Текущая сеть
2. Удаленная сеть
3. Путь не определен (Gateway of Last Resort)

	Запись	Next-hop	Выходной интерфейс
R1	192.168.4.0/24	192.168.2.2	Fa0/1
R2	192.168.4.0/24	192.168.3.2	S0/0/0
R3	192.168.4.10	192.168.4.10	Fa0/0

Маршрутизация



Статическая маршрутизация

Преимущества

- Просто выполнить в небольших сетях
- Безопасный
- Предсказуемый
- Не использует дополнительные вычислительные ресурсы

Недостатки

- Подходит для простых топологий
- Если связь пропадает, нет возможности автоматически её восстановить

Функции динамических протоколов

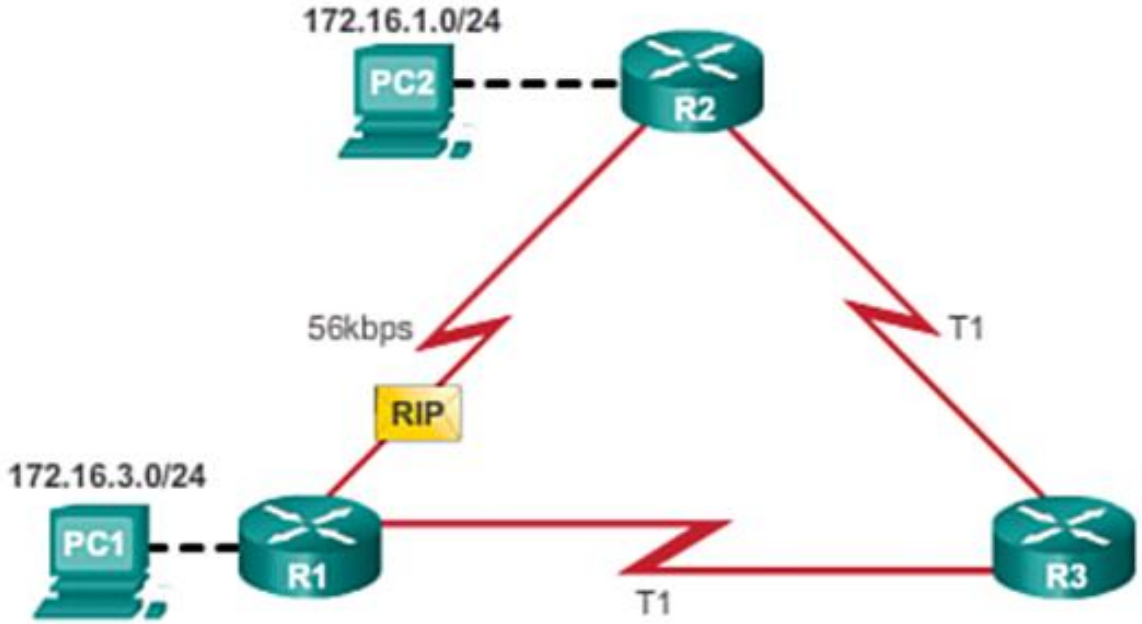
- Обнаружение удаленных сетей
- Поддержка актуальной информации о маршрутах
- Выбор наилучшего пути
- Поиск нового лучшего пути, если топология изменилась

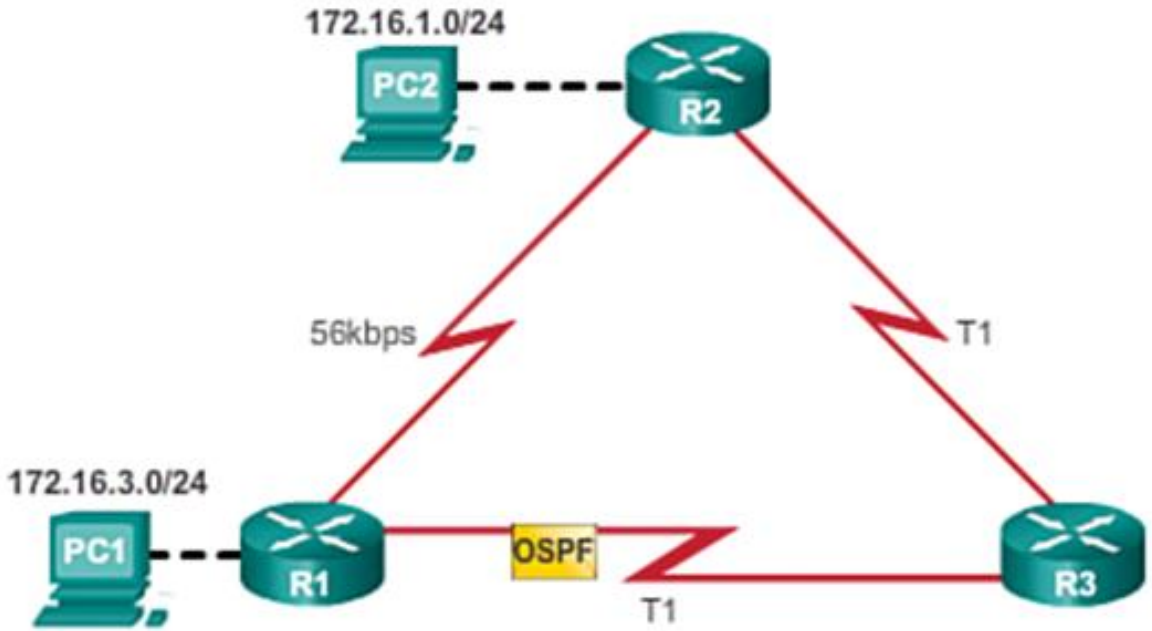
Классификация динамических протоколов

Протокол на базе вектора расстояния VS состояния линии

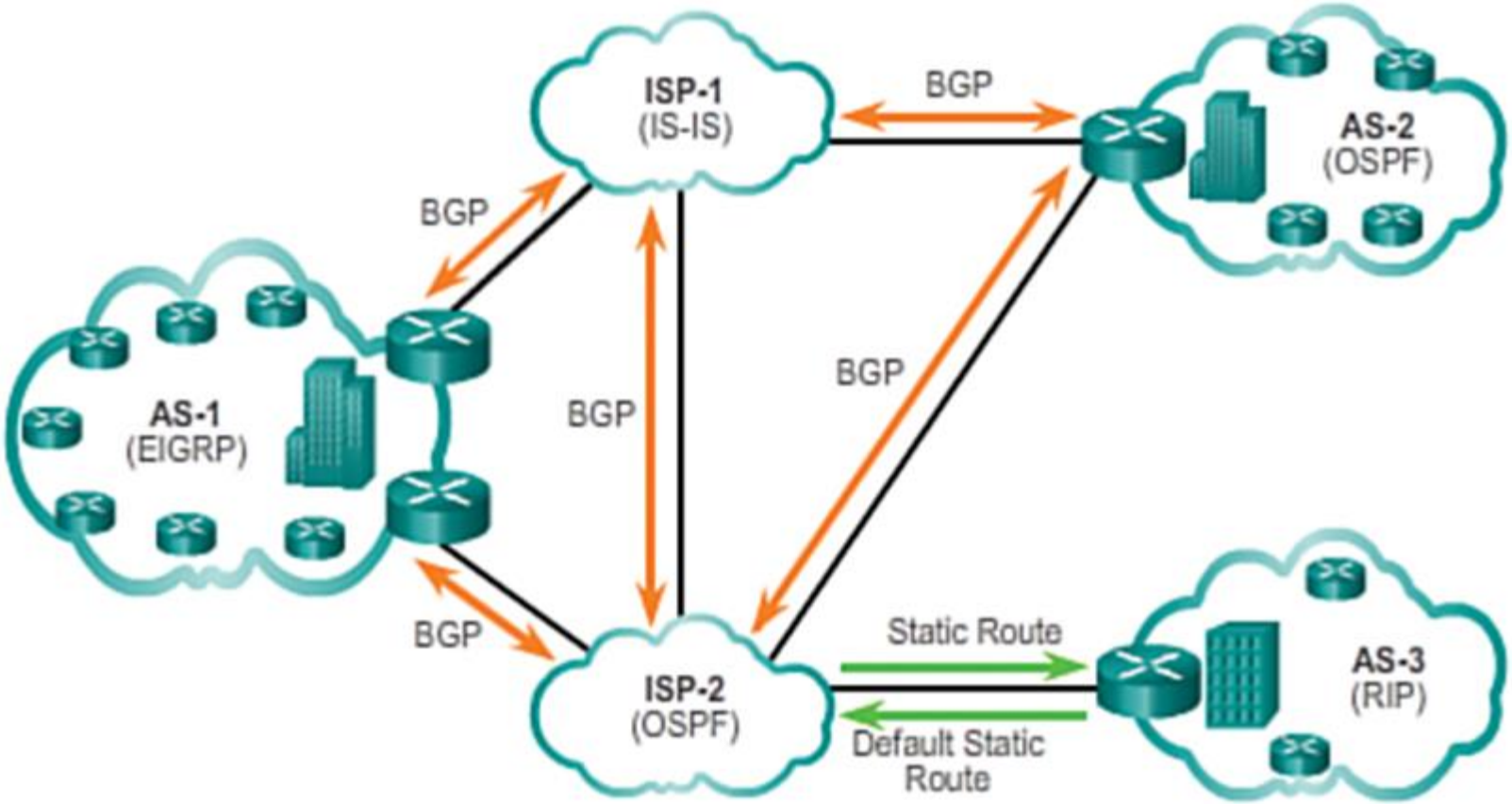
- Маршрутизаторы, работающие с протоколом с **вектором расстояния** (distance vector), не знают о полном пути до пункта назначения, но имеют две базовые характеристики:
  - **Расстояние** (distance): определяет как далеко до назначения за счет метрики, такой как количество хопов, ширины пропускания, задержки и др.
  - **Вектор** (vector): определяет направление передачи (следующий хоп, выходной интерфейс (порт))
- Маршрутизаторы с **состоянием линии** (link-state), как правило, создают полную сетевую топологию за счет сбора информации от всех других маршрутизаторов. Подходит для больших иерархических сетей, существует необходимость быстрой сходимости.
- Маршрутизаторы с **вектором расстояния** периодически отправляют сообщения об имеющихся маршрутах соседям.
- Маршрутизаторы с **состоянием линии** после того, как построили сетевую топологию, отправляют сообщения, только если произошли изменения в топологии.

RIP vs OSPF


172.16.1.0/24
PC2
R2
56kbps
T1
RIP
172.16.3.0/24
PC1
R1
R3
T1


172.16.1.0/24
PC2
R2
56kbps
T1
172.16.3.0/24
PC1
R1
OSPF
R3
T1

Пример использования динамических протоколов



Динамические протоколы маршрутизации

Преимущества

- Подходит для всех топологий, где используются несколько маршрутизаторов
- В целом, не зависит от количество сетей
- Автоматически адаптируется под изменения топологии сети

Недостатки

- Менее безопасный из-за необходимости обмениваться сообщениями
- Маршрут зависит от текущей топологии
- Требует дополнительные вычислительные и сетевые ресурсы

Преобразование сетевых адресов (NAT)

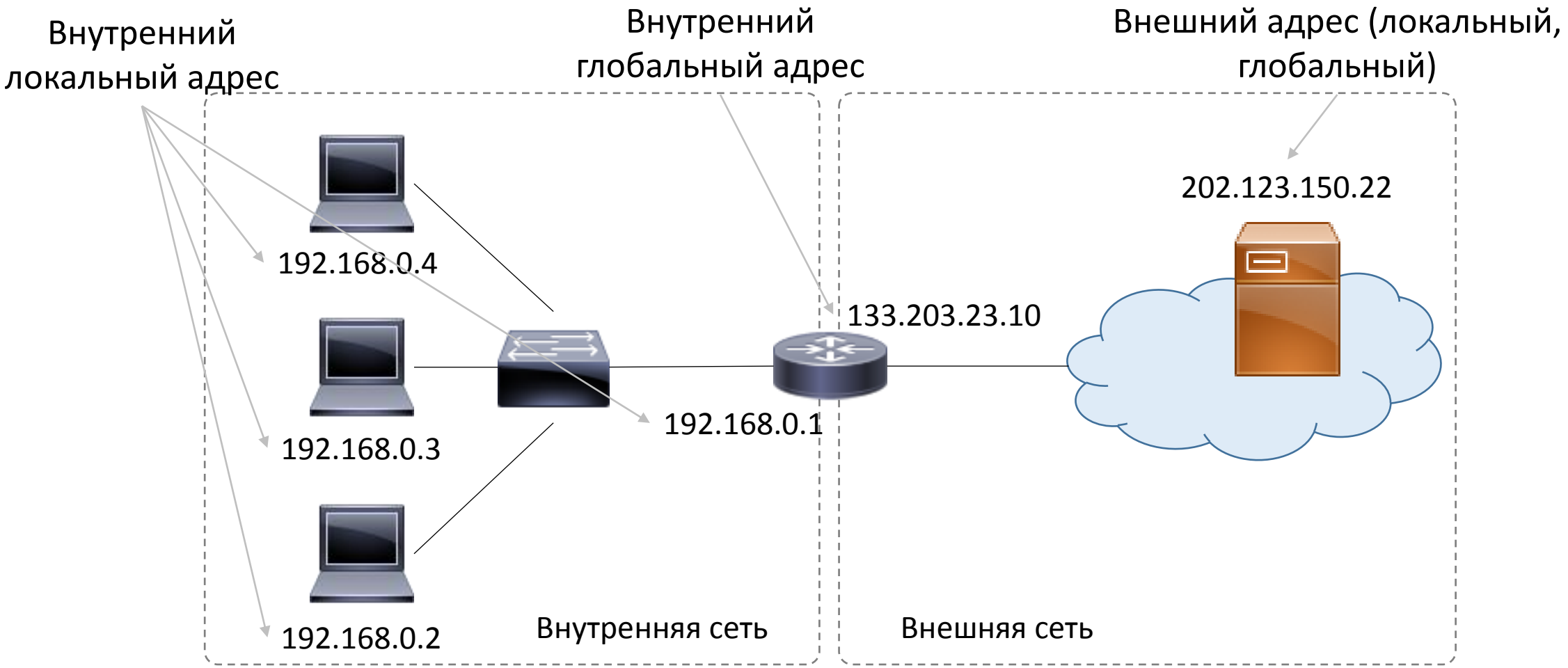
Преобразование сетевых адресов

- **Преобразование сетевых адресов** (Network Address Translation – NAT) обеспечивает взаимосвязь IP адресов пересылаемых пакетов во внутренней и внешней сетях.

- Два основных процесса:
  - Перевод реального IP адреса внутренней сети исходящего пакета в IP адрес для глобальной адресации
  - Преобразованием IP адреса входящих пакетов с глобальной адресацией в IP адрес внутренней сети

- Виды NAT:
  - Стандартный NAT (статический и динамический)
  - Трансляция порт-адрес PAT (статический и динамический)

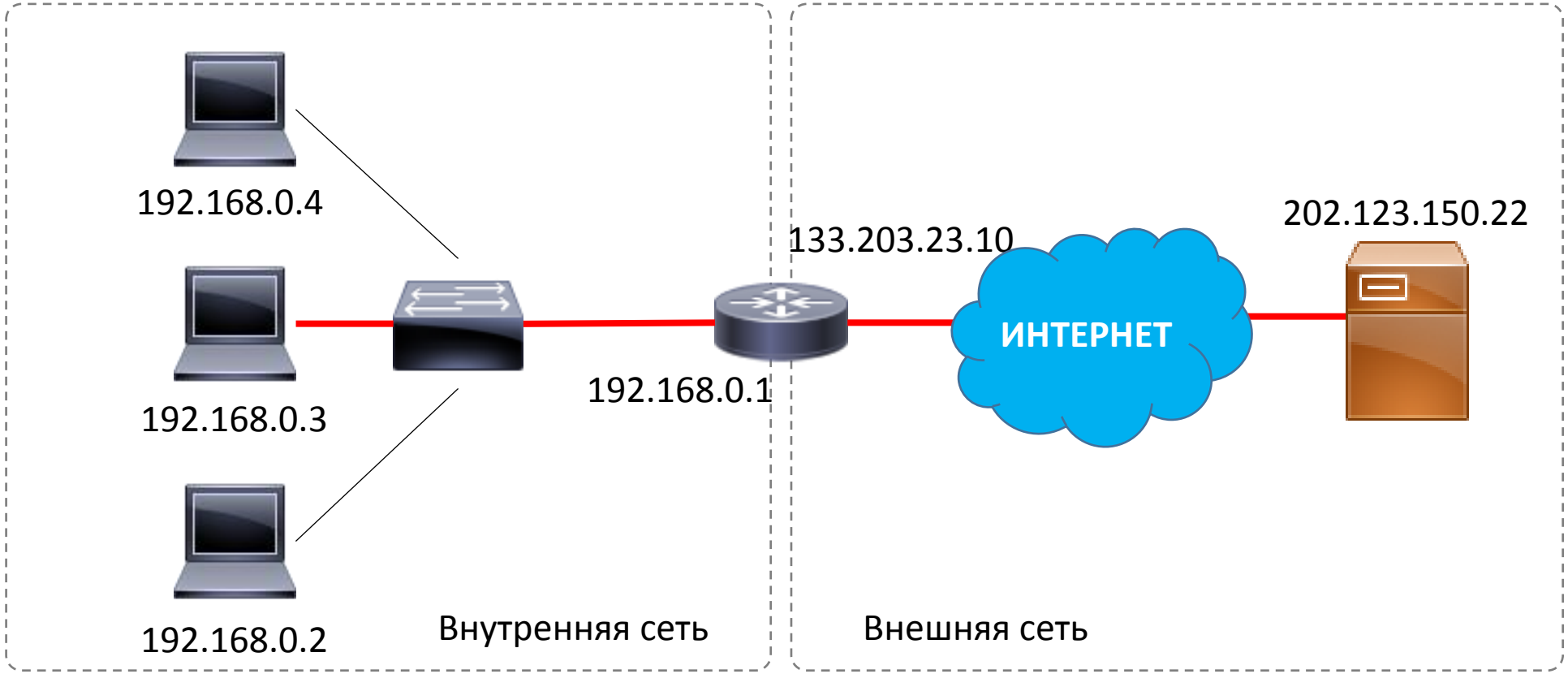
Термины NAT


Внутренний локальный адрес
Внутренний глобальный адрес
Внешний адрес (локальный, глобальный)
202.123.150.22
192.168.0.4
192.168.0.3
192.168.0.2
192.168.0.1
133.203.23.10
Внутренняя сеть
Внешняя сеть

Статический NAT

NAT-таблица

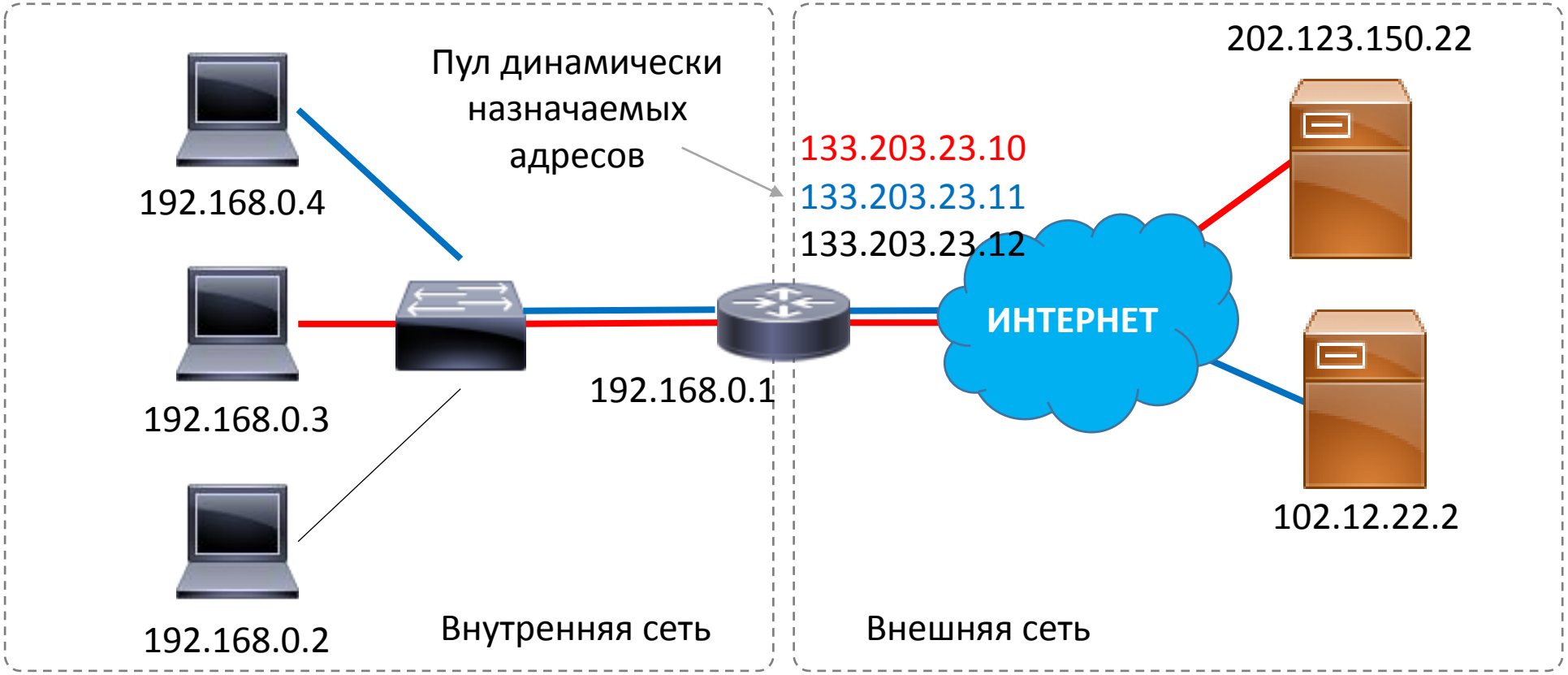
Внутренний локальный адрес	Внутренний глобальный адрес	Внешний глобальный адрес
192.168.0.3	133.203.23.10	202.123.150.22



Динамический NAT

NAT-таблица

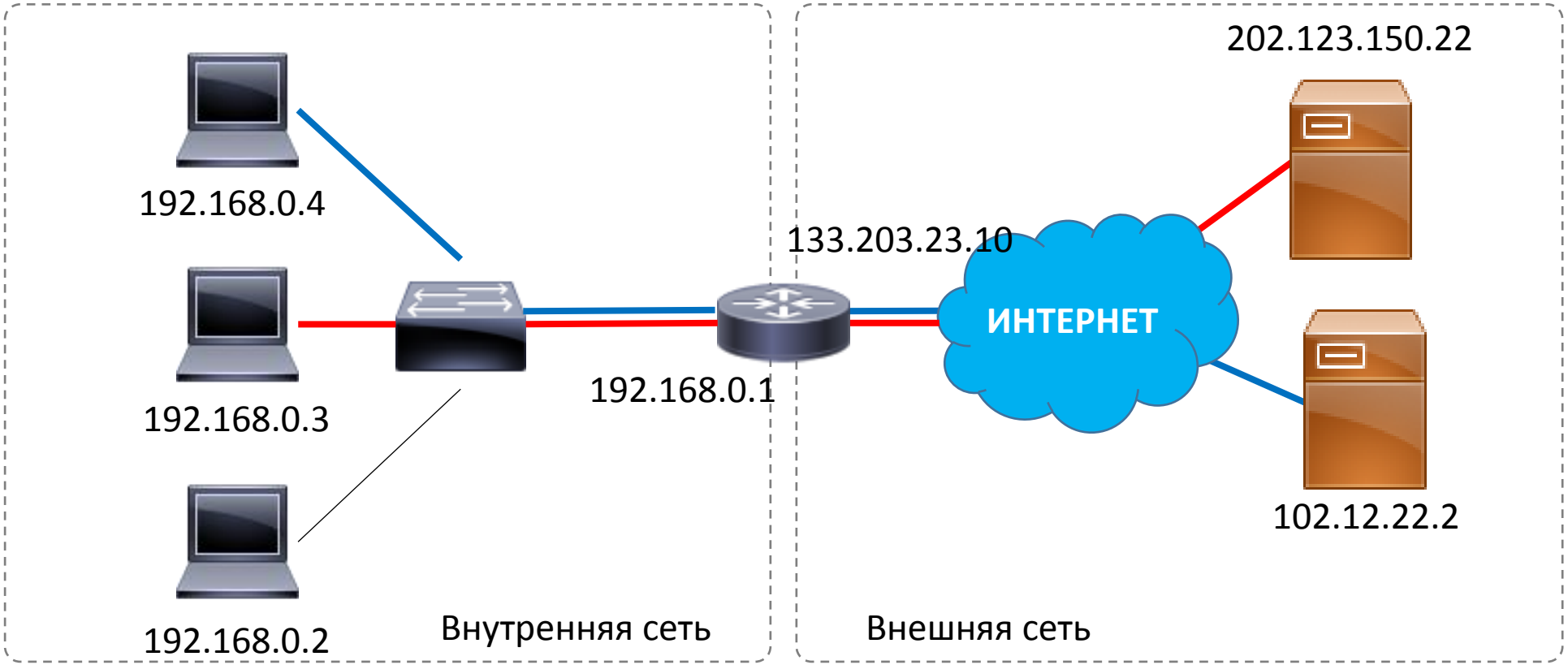
Внутренний локальный адрес	Внутренний глобальный адрес	Внешний глобальный адрес
192.168.0.3	133.203.23.10	202.123.150.22
192.168.0.4	133.203.23.11	102.12.22.2



Трансляция порт-адрес (PAT)

PAT-таблица

Внутренний локальный адрес: порт	Внутренний глобальный адрес: порт	Внешний глобальный адрес: порт
192.168.0.3**:7480**	133.203.23.10**:7480**	202.123.150.22**:80**
192.168.0.4**:7480**	133.203.23.10**:5839**	102.12.22.2**:80**
192.168.0.3**:8732**	133.203.23.10**:8700**	102.12.22.2**:80**



Система доменных имен (DNS)

Система доменных имен

Система доменных имен (Domain Name System – DNS) обеспечивает взаимосвязь IP адресов и доменных имен

Виды DNS:

- Итеративная
- Рекурсивная

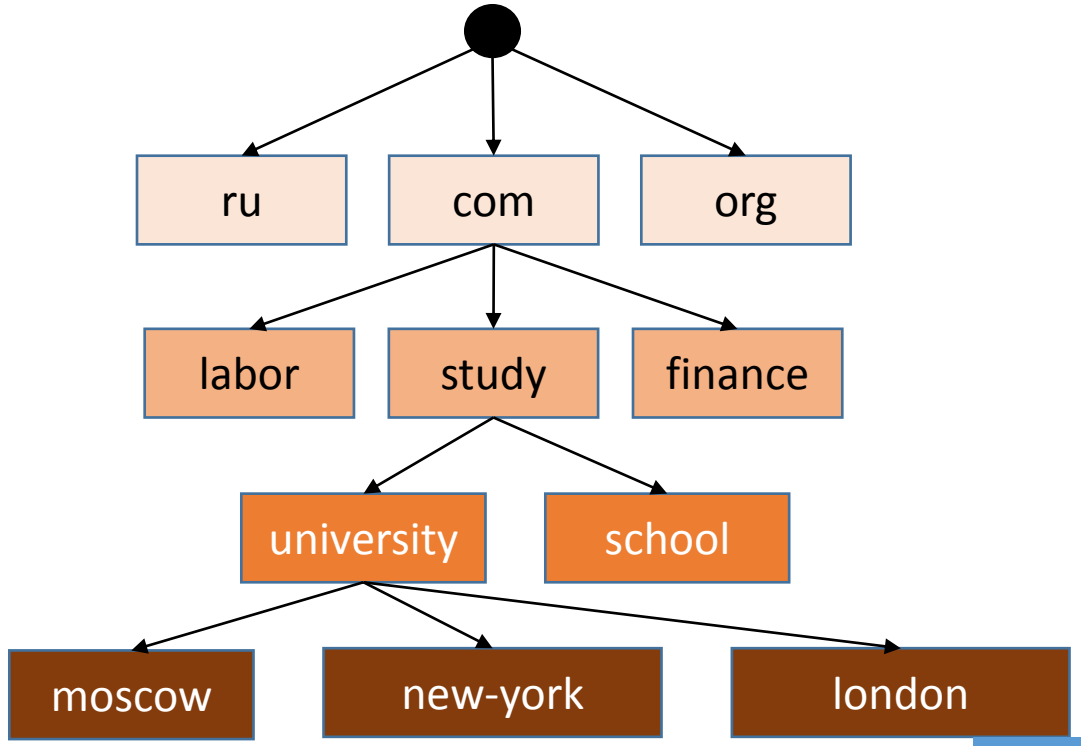
Уровни DNS

DNS на компьютере пользователя

DNS в локальной сети

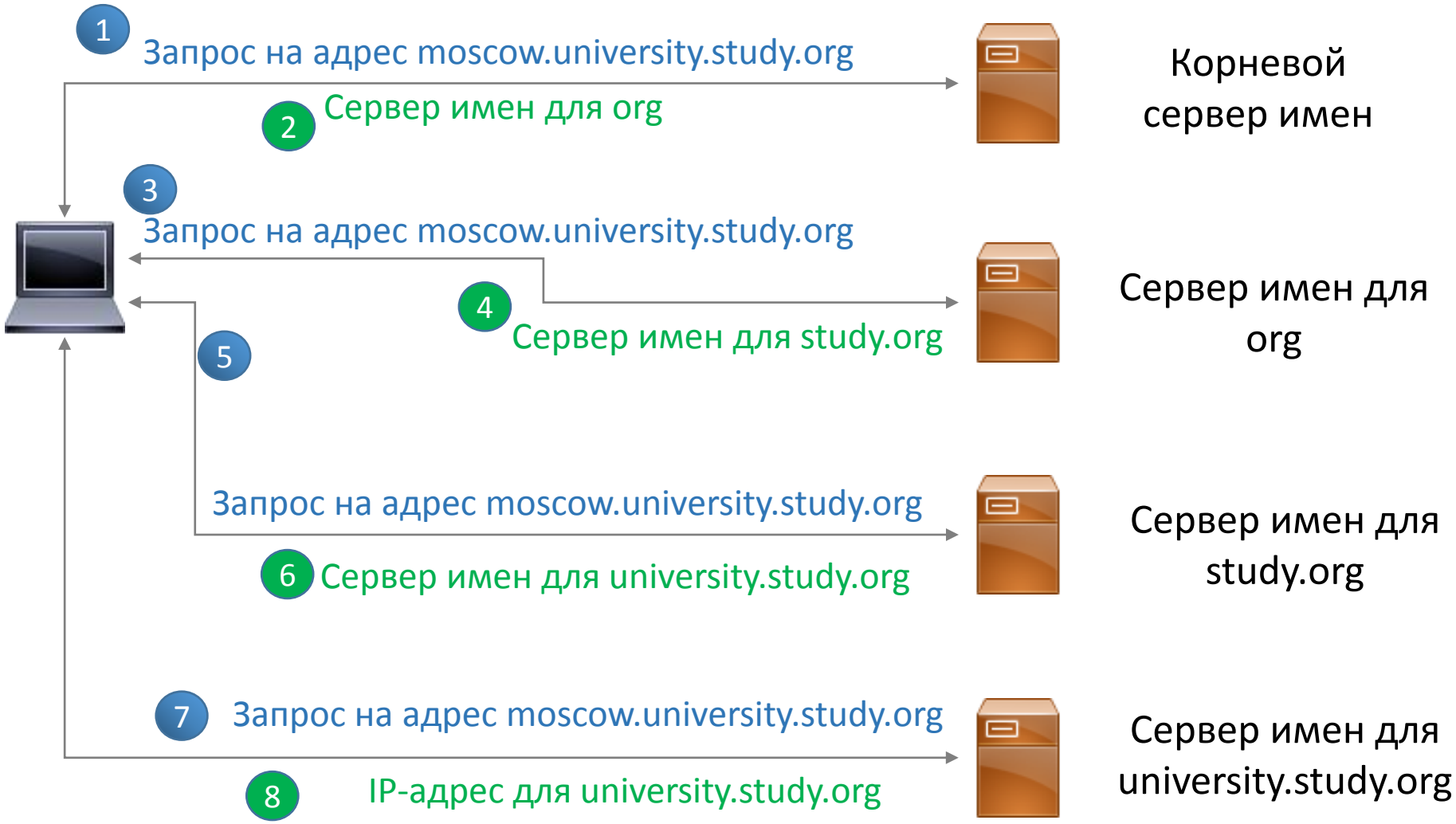
Внешний DNS

Пространство доменных имен
moscow.university.study.org



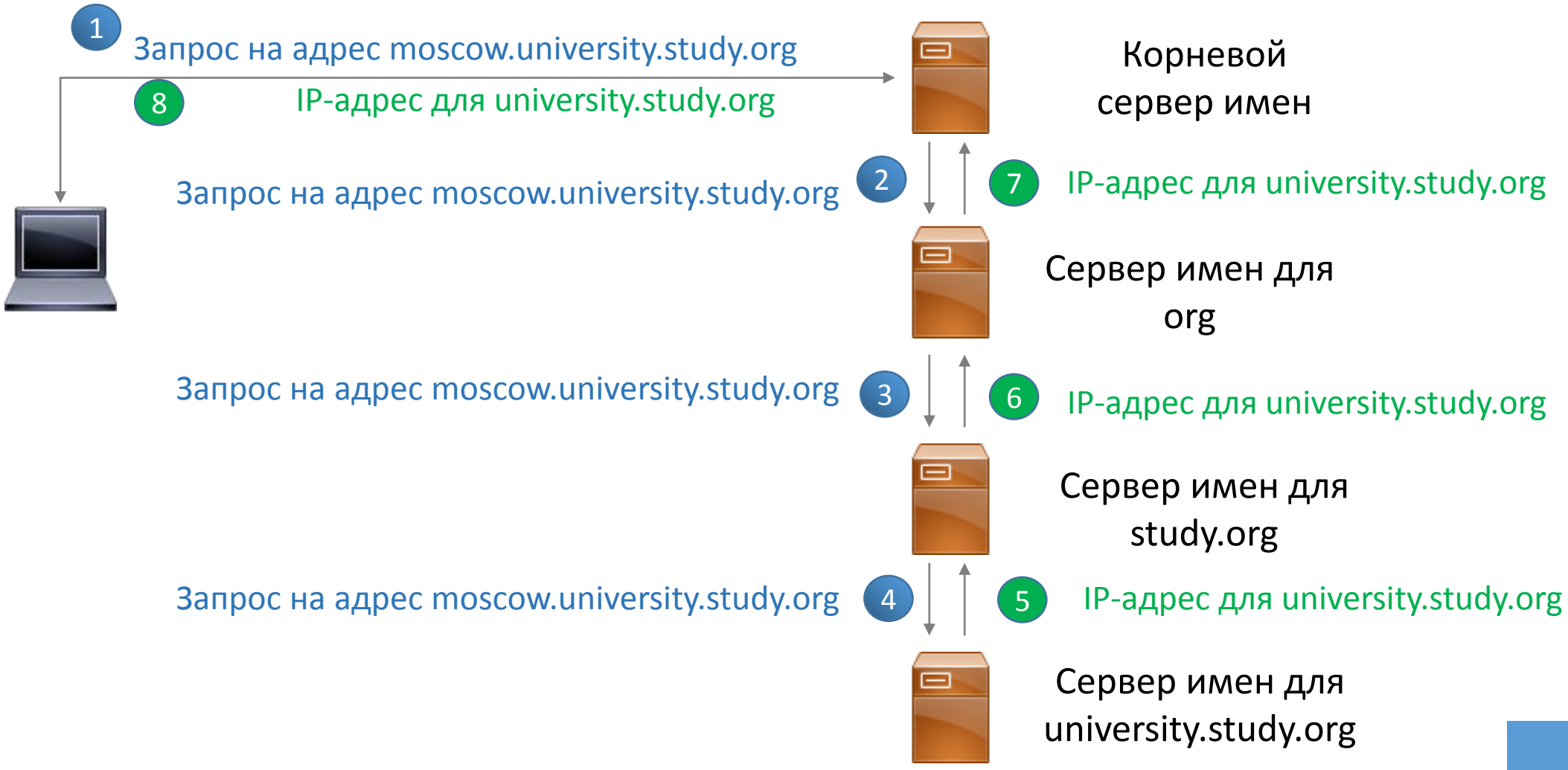
Итеративная DNS

moscow.university.study.org



Рекурсивная DNS

moscow.university.study.org



Протокол динамической настройки узлов (DHCP)

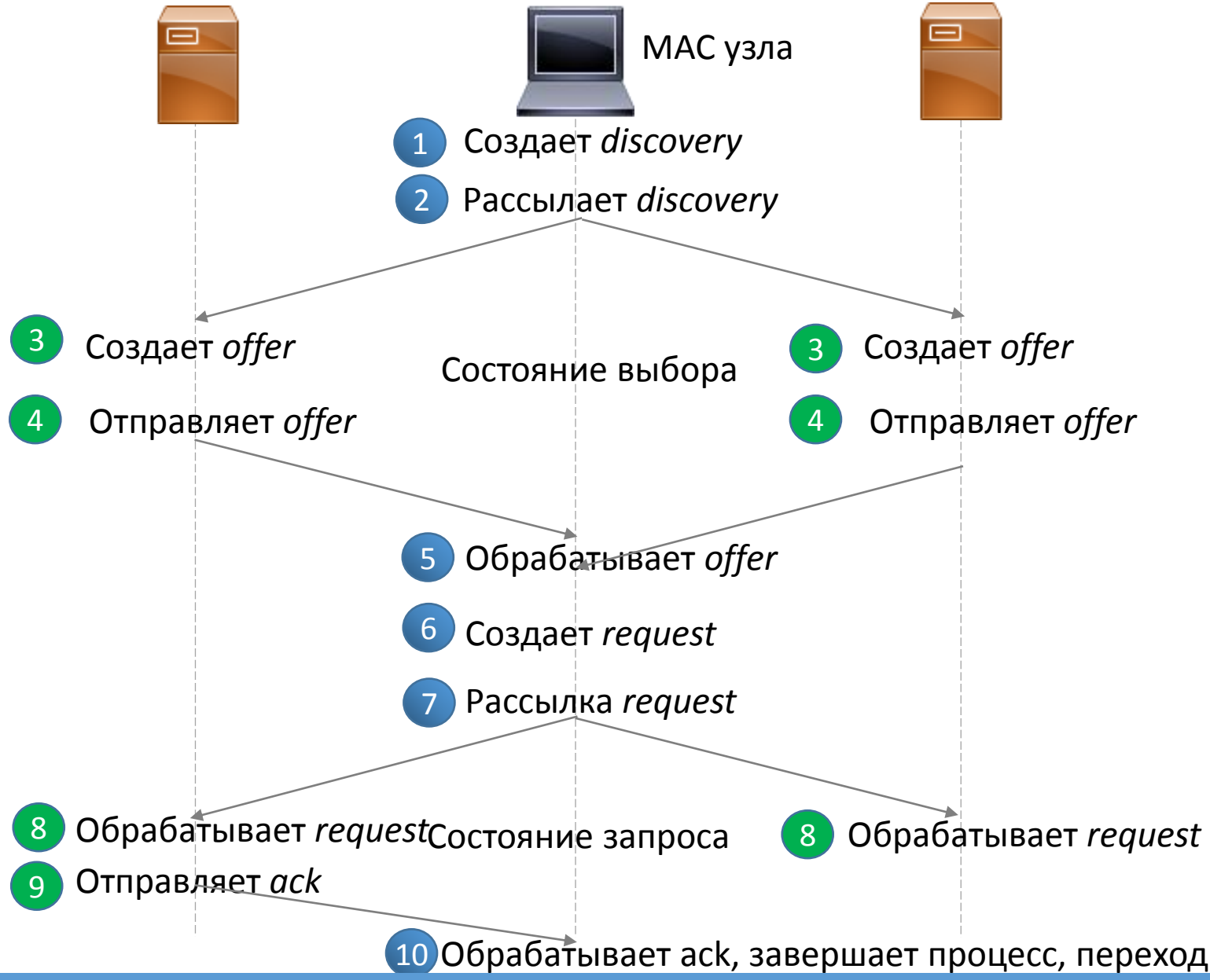
Протокол динамической настройки узлов (1)

- **Протокол динамической настройки узлов** (Dynamic Host Configuration Protocol – DHCP) используется для динамического назначения IP адресов узлам, а также для передачи другой важной информации (маска сети, адрес шлюза, DNS сервера и др.) необходимых для обмена данными в сети.

- Конфигурация предоставляется узлу на определенное время, время аренды (lease time). В процессе работы узел может обновить время аренды до его окончания. После завершения времени аренды данные об узле удаляются с DHCP сервера, и освободившийся IP может быть использован для других узлов (DHCP клиентов)

- Используются UDP порты 67 (сервер) и 68 (клиент)

Протокол динамической настройки узлов (2)



Discovery-сообщение:
MAC, ID транзакции,
*требуемый IP,
*время аренды

Offer-сообщение:
Предлагаемый IP адрес,
время аренды,
другие параметры
ID сервера,
ID транзакции из discovery

Request-сообщение:
ID выбранного сервера,
Предложенный IP

Источники

- Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы
- The TCP/IP Guide
- Basic Data Transmission in Networks: MAC Tables and ARP Tables
- Routing Protocols Companion Guide
- DNS Basic Name Resolution Techniques: Iterative and Recursive Resolution
- DHCP Lease Allocation Process