Принцип коммутации пакетов

(Смелянский Р.Л. Компьютерные сети т.1 стр. 30 – 36, 188-193)

Введение в компьютерные сети

чл.-корр. РАН Смелянский Р.Л.

Кафедра АСВК ф-т ВМК МГУ

Содержание

- Коммутация каналов
- Коммутация пакетов
- Почему в Интернет используется коммутация пакетов

Коммутация каналов

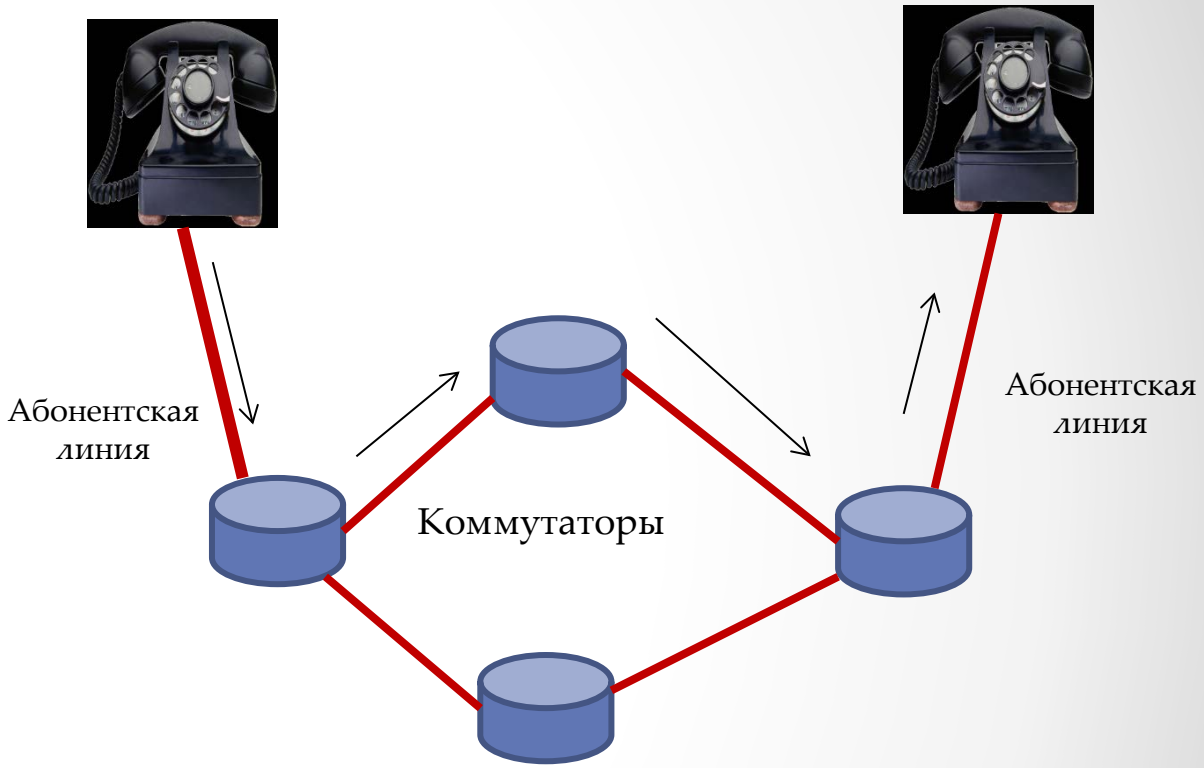
Абонентская линия

Абонентская линия

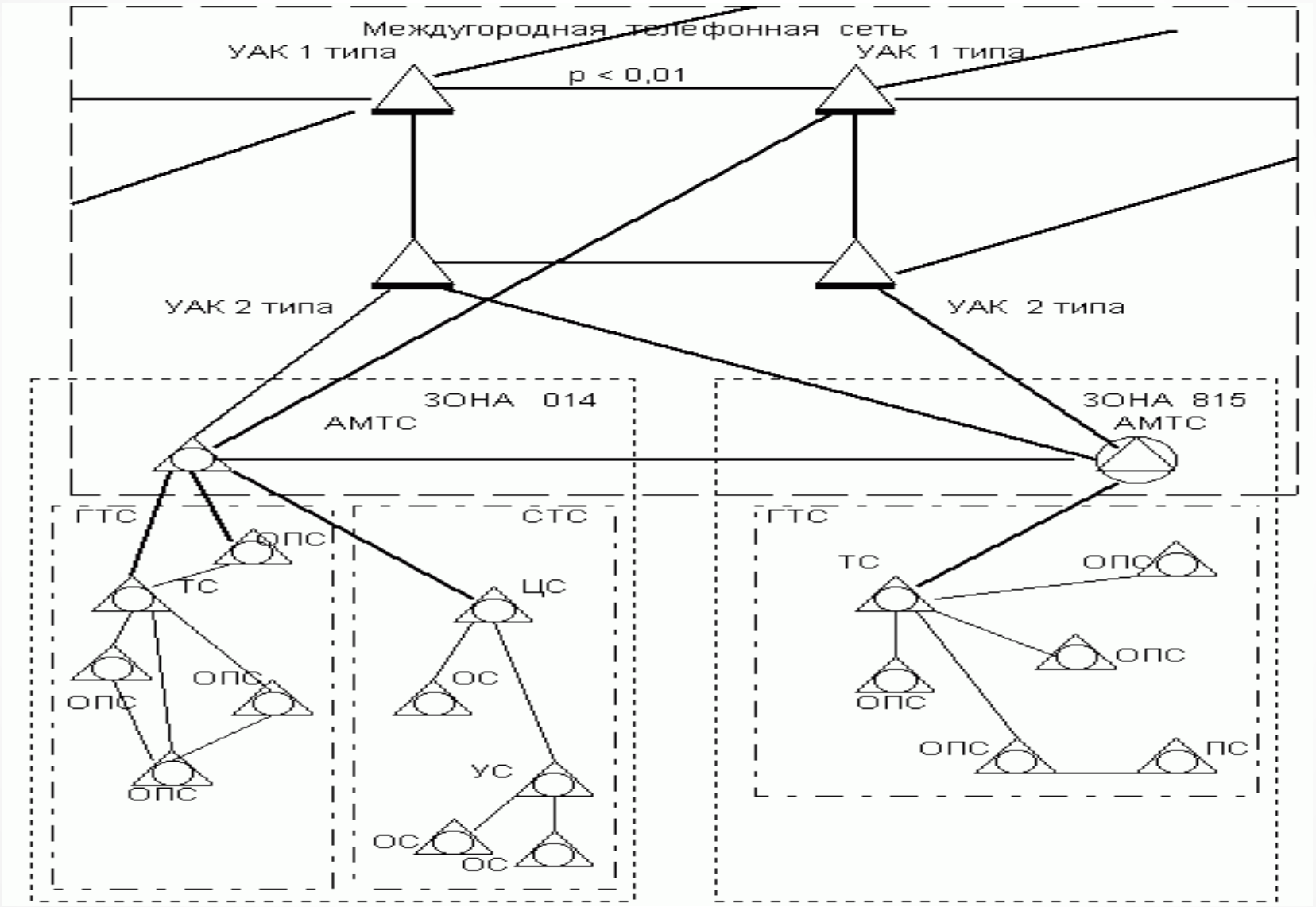
Коммутация каналов

Коммутация каналов: проблемы

- Неэффективность: взаимодействие между компьютерами в сети ***очень не равномерно*** (набор текста по ssh соединению и просмотр web страницы). Если каждая коммуникация будет использовать выделенное соединение, это будет крайне ***не эффективно***.
- Большой разброс скоростей взаимодействия (набираю текст или просмотрю видео с web)
- Управление состоянием взаимодействия: все коммутаторы, задействованные в соединении должны поддерживать согласованно состояние взаимодействия



Коммутация каналов



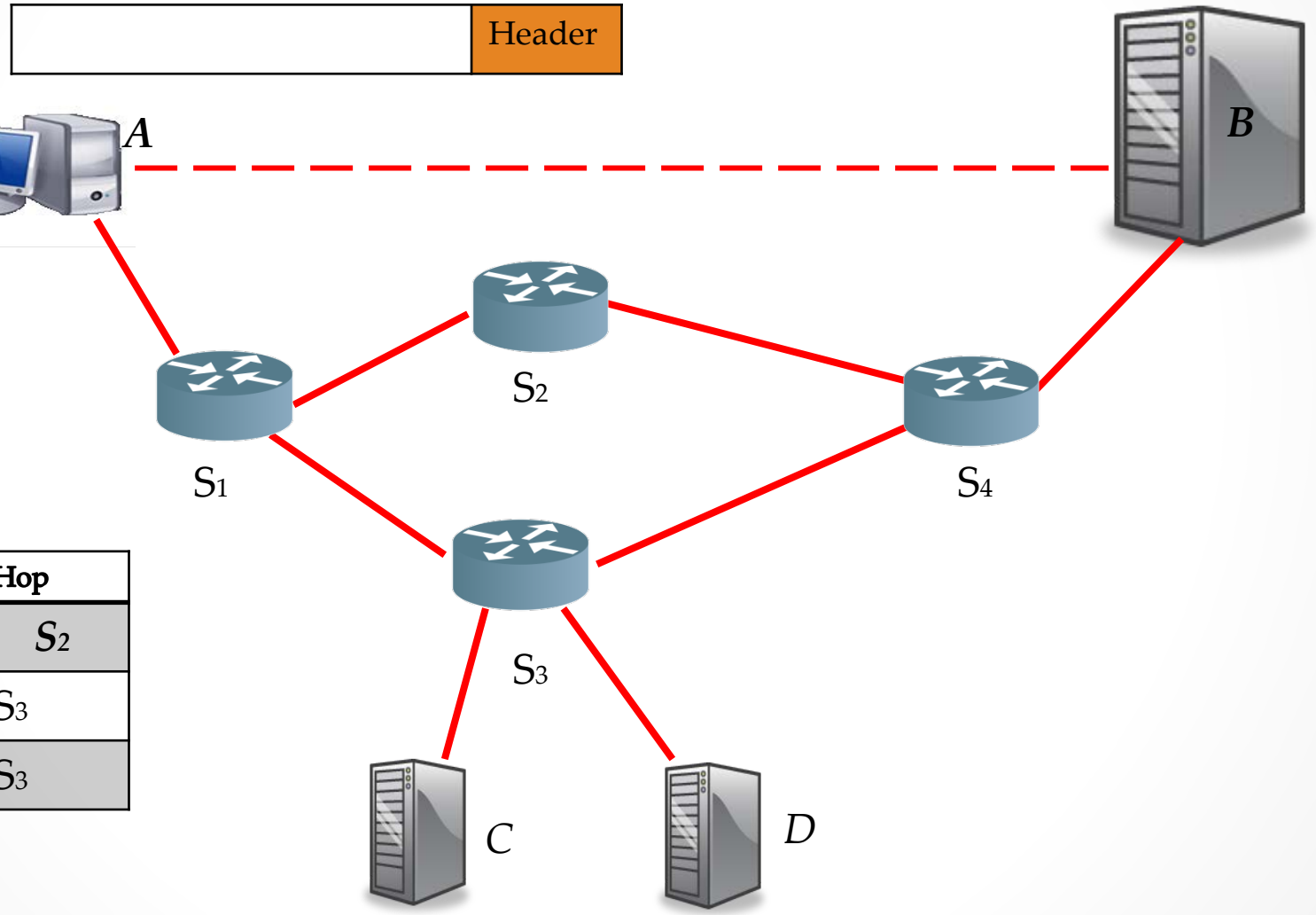
Коммутация каналов

- У каждого соединения точка-точка своя гарантированная, пропускная способность, доступная тому кто передает по ней
- Каждый звонок состоит из трех фаз:
  - Установление соединения точка-точка
  - Передача
  - Разрыв соединения
- Во времена Александра Белла соединение представляло собой провод
- Сегодня, соединение – это виртуальный (псевдо) индивидуальный провод

Что такое коммутация пакетов?

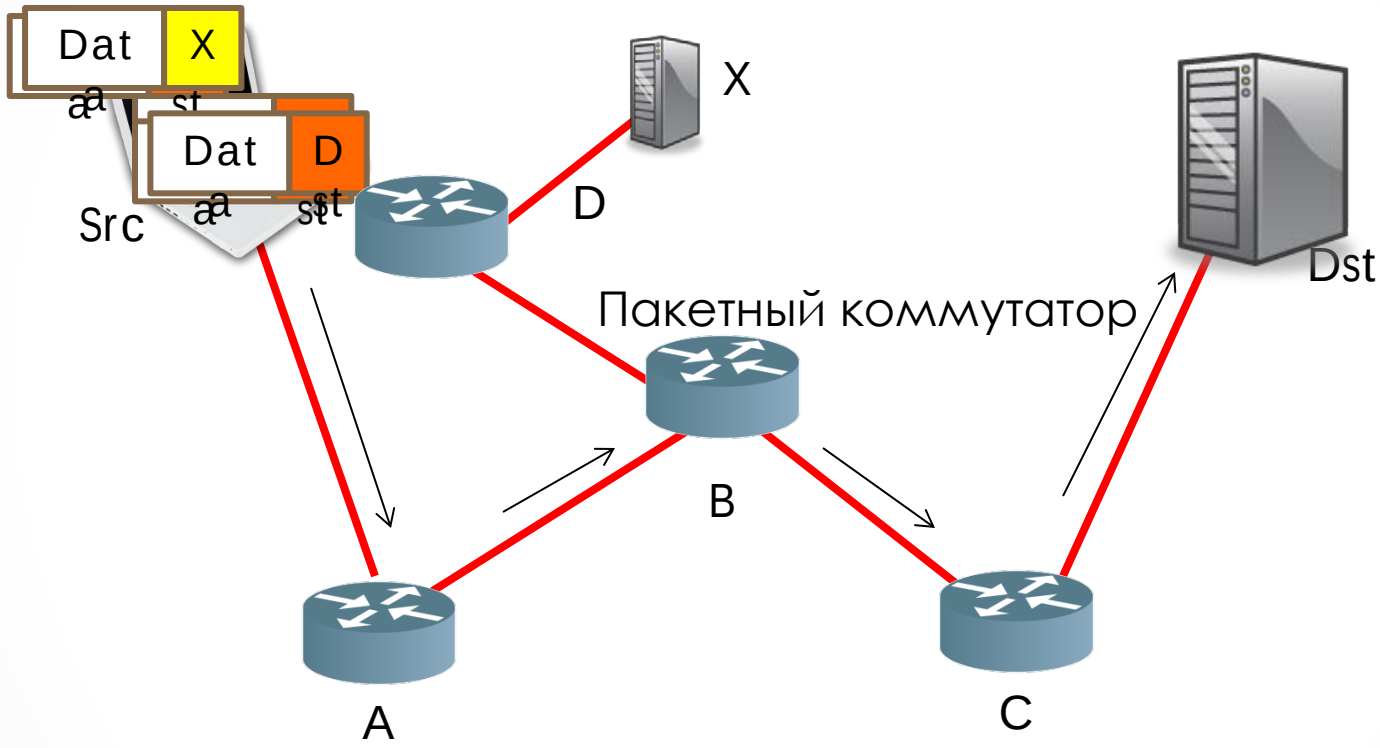
- **Пакет** – единица данных, несущая достаточно информации, чтобы быть доставленной к месту назначения
- **Коммутация пакетов:** для каждого поступающего пакета независимо выбирается канал для отправки. Если канал свободен, то пакет отправляют, если нет – буферизуют пока канал не освободится

Коммутация пакетов

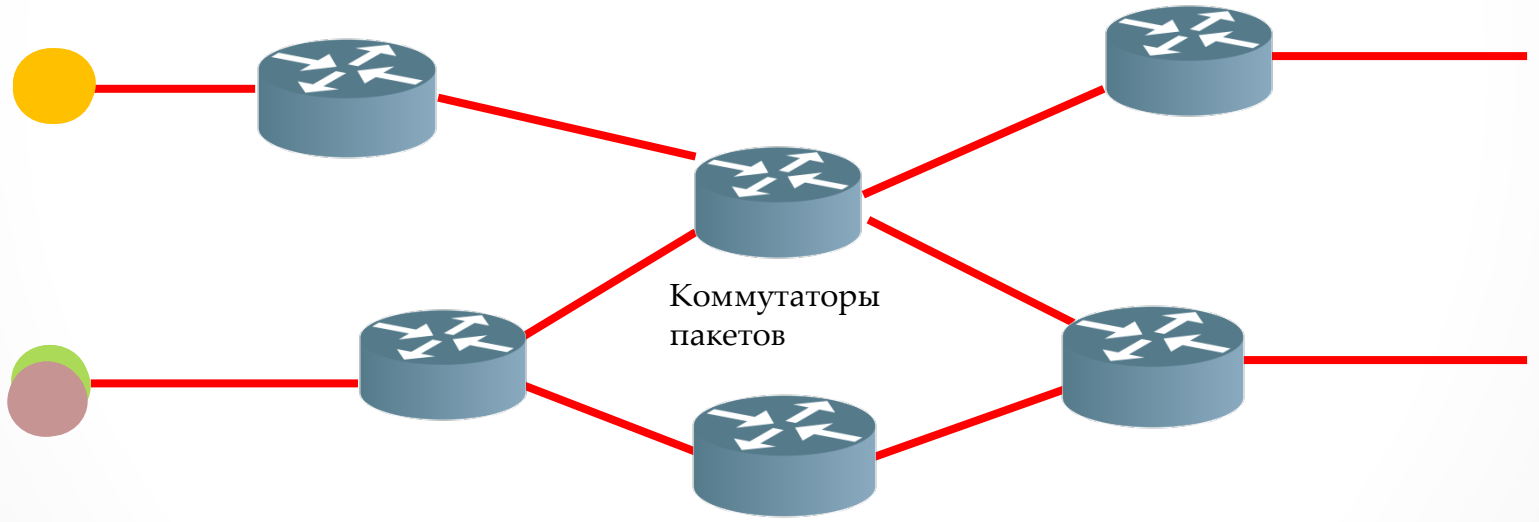


Address	Hop
B	S_2
C	S_3
D	S_3

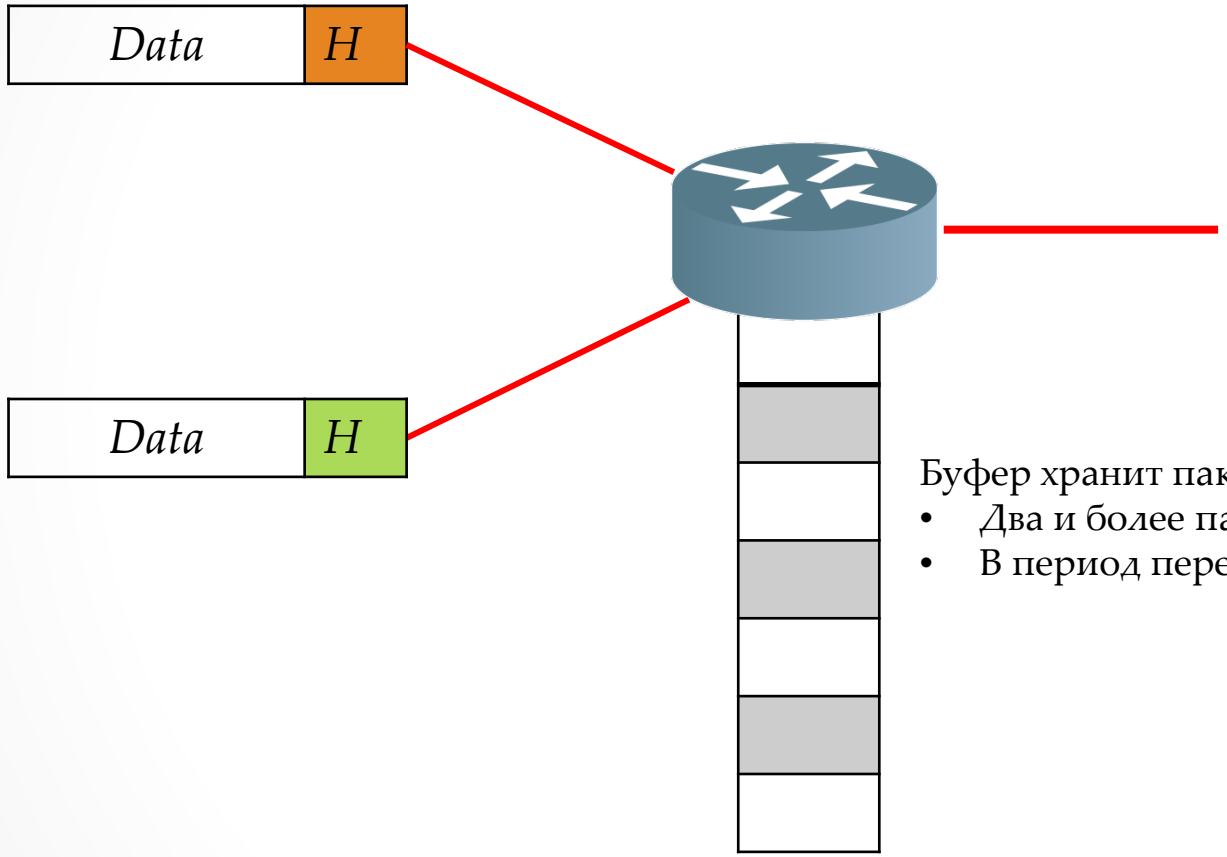
Коммутация пакетов



Коммутация пакетов



Коммутация пакетов: буферизация



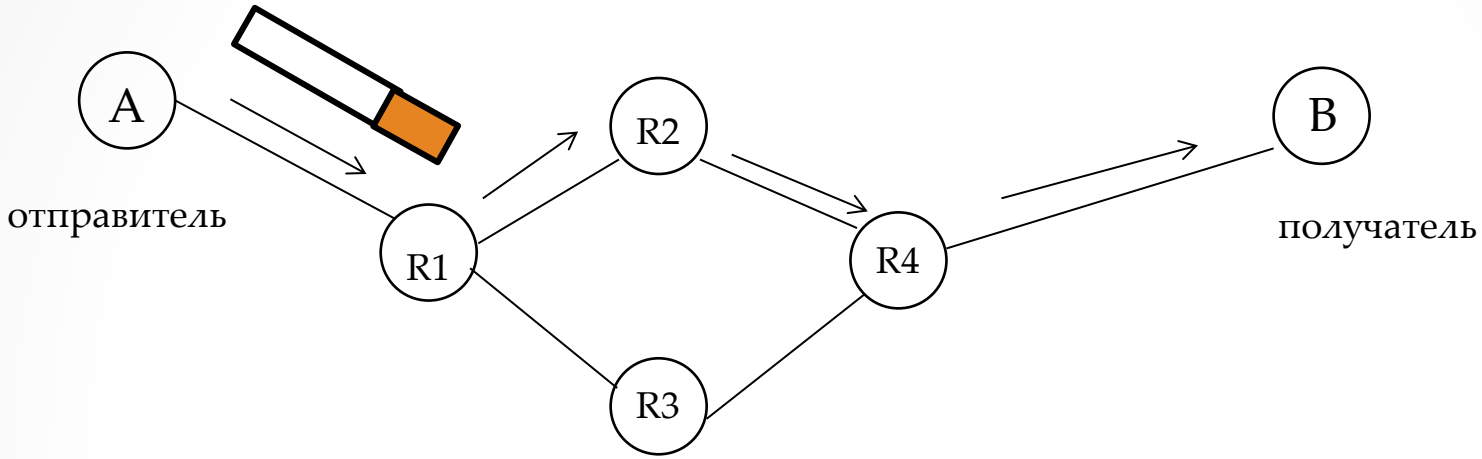
Буфер хранит пакеты:

- Два и более пакеты пришли одновременно
- В период перегрузки

Не требуется состояния для потока

- каждый пакет самодостаточен
- не надо поддерживать/хранить информацию о состоянии потока
- в случае ошибки нет изменений состояния потока

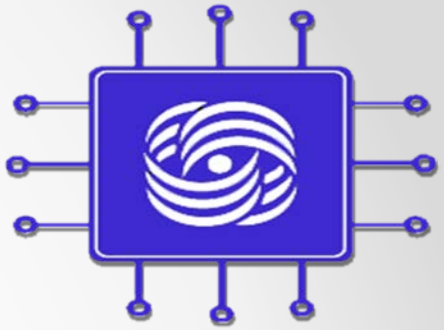
Коммутация пакетов



- Пакеты маршрутизируют индивидуально согласно адресам локальной таблицы маршрутизации
- Одна и та же линия может быть использована пакетами разных потоков в разное время
- Все пакеты используют полную пропускную способность линии
- Маршрутизаторы не поддерживают состояния каждого взаимодействия
- Статистическое мультиплексирование/уплотнение канала
- Ели какой-то канал или маршрутизатор будет разрушен/выйдет из строя, то утерянную дейтаграмму всегда можно послать еще раз другим маршрутом

Заключение

- коммутация пакетов обеспечивает более эффективное использование ресурсов, чем коммутация каналов
- сеть с коммутацией пакетов более живучая и устойчивая к повреждениям
- пакет самодостаточен, каждый пакет маршрутизируют индивидуально на основе таблицы маршрутизатора, которую формирует сам коммутатор
- каждый коммутатор работает автономно от других
- состояние соединений, проходящих через коммутатор, не хранится
- буфер коммутатора разделяют проходящие через него потоки пакетов динамически



Принцип: имена и адреса

Введение в компьютерные сети

чл.-корр. РАН Смелянский Р.Л.
Кафедра АСВК ф-т ВМК МГУ

Имена vs Адреса

- Имя указывает что есть объект именования
  - Кабинет Смелянского
  - pluto.lvk.cs.msu.su
  - list_srt
- Адрес указывает где расположен объект
  - 119899, Москва, Ленинские горы, МГУ, 2-ой учебный корпус, ком.764
  - IP: 171.66.3.9
  - 0x0040005080
- Телефонный номер – это адрес или имя?

Имена

- Структура имени:
  - одноуровневое:
    - имя бирж (РТС, ММВБ, NSDQ), названия аэропортов (SVO, DMV)
    - сервисы – http, ftp, VPN
    - Apple ID, Skype ID
  - набор (БД): пол: муж.; ФИО: Смелянский Р.Л.; должность: профессор;
  - иерархия:
    - lvk.cs.msu.su
    - Смелянский Руслан Леонидович, профессор ф-та ВМК, Московский Гос. Университет

Адреса

- Структура адреса влияет на то, как мы можем указывать на объект
  - одно ранговый адрес: номер порта, адрес основной памяти
  - набор: координаты (x,y,z); (55°45′с.ш.:37°36′в.д.)
  - иерархические адреса
    - адрес сегмента данных на диске (цилиндр, дорожка, сегмент)
    - страна, город, улица, дом, квартира
    - косвенный адрес
    - <код страны> <код региона> < номер абонента в регионе>

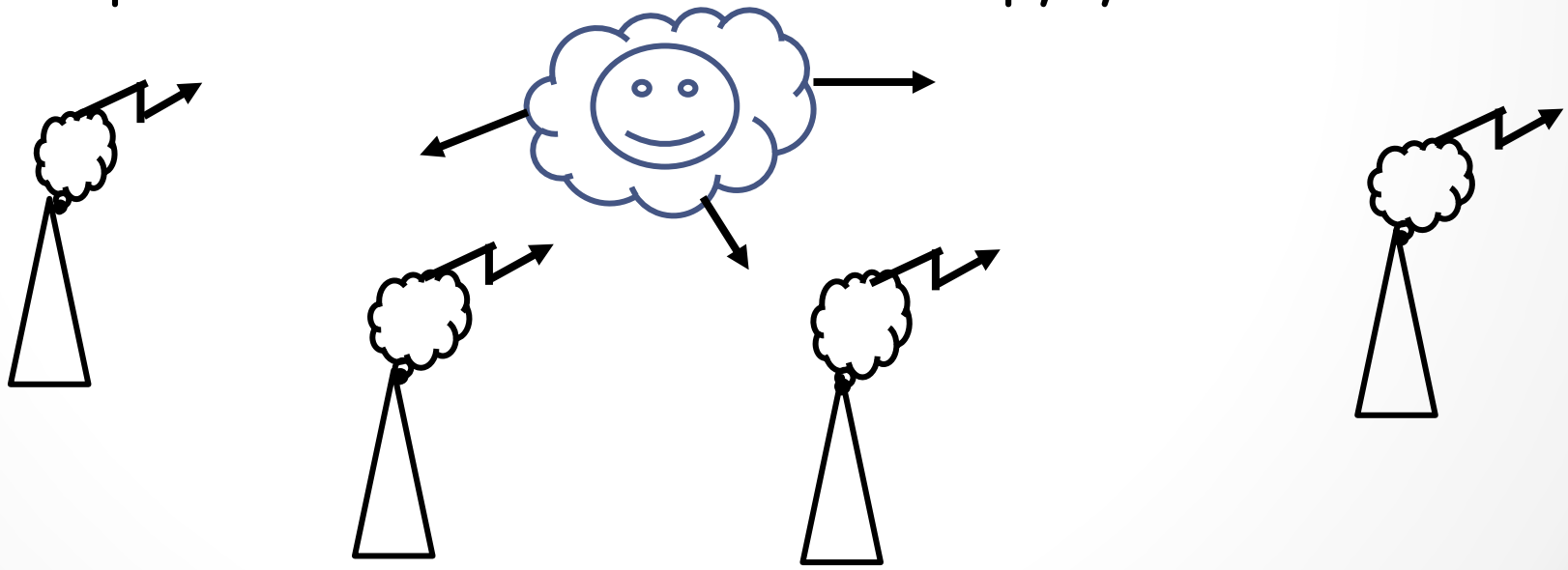
Пример: загрузка файла

- Как работает ссылка на файл?
- Адрес: http://lvk.cs.msu.su/publications/~smel/pubs.html
  - указывает на хост, где файл расположен
  - указывает где файл расположен на хосте
- Имя: обращение через hash-функцию
  - запрашиваем файл через hash-функцию (изменяется файл, меняет значение hash-функции, включая место расположения) 0x27de2b6939d7fb4b0573dbd6dbe2c740

Примеры

- http://lvk.cs.msu.su/~smel vs http://171.64.73.43/~smel

- пользователь перемещаться из одной сети в другую



адреса и имена в Internet

- Internet адреса: 32 IPv4; 128 IPv6

  95.182.74.02

- Internet имена: система доменных имен (DNS): lvk.cs.msu.su

- Имена и адреса на более высоких уровнях
  - имя сервиса (http), имя порта (80)
  - SIP идентификатор (pal@a.com), e-адрес (pal@a.com)

- Internet Corporation for Assigned Names and Numbers (ICANN)
  - Internet Assigned Number Authority (IANA)

Заключение

- То как Вы именуете и адресуете что-то, имеет огромное влияние на организацию сети, ее протоколов, сложность алгоритмов работы с информационными объектами

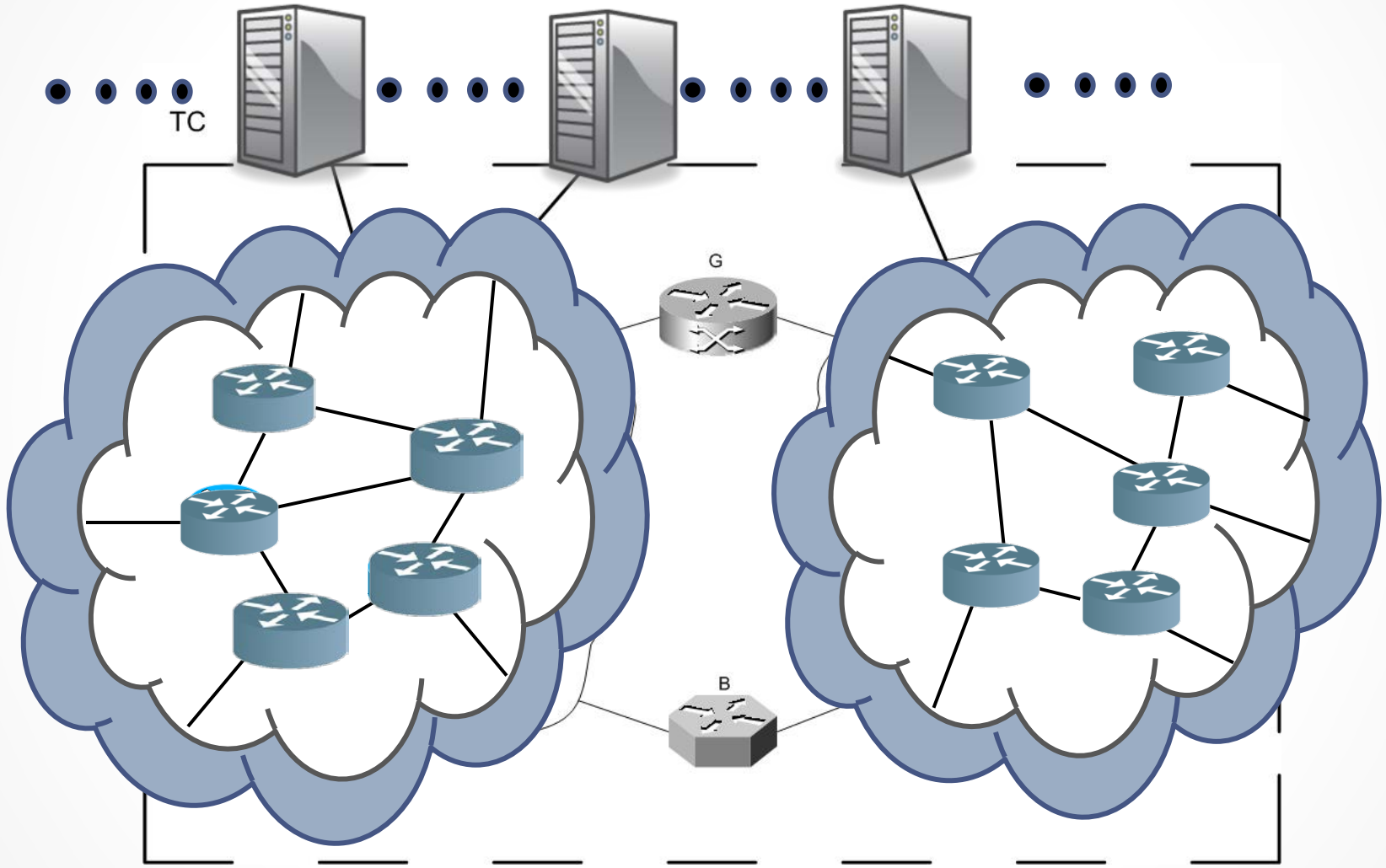
- Структура имен и адресов также имеет большое влияние на простоту и удобство интеграции сетей и протоколов

Принцип E2E

Введение в компьютерные сети

чл.-корр. РАН, проф. Смелянский Р.Л.
Кафедра АСВК ф-т ВМК МГУ

Сеть с точки зрения приложения



End-To-End Принцип

The function in question can completely and correctly be implemented only with the knowledge and help of the application standing at the end points of the communication system. Therefore, providing that questioned function as a feature of the communication system itself is not possible. (Sometimes an incomplete version of the function provided by the communication system may be useful as a performance enhancement.) We call this line of reasoning. . . "the end-to-end argument."

- Saltzer, Reed, and Clark,
End-to-end Arguments in System Design, 1984

Заключение

The network's job is to transmit datagrams as efficiently and flexibly as possible. Everything else should be done at the fringes. . .

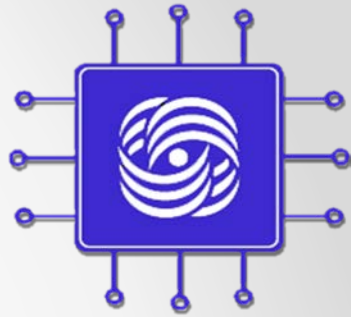
– [RFC 1958]

Задача сети - передавать датаграммы так эффективно и гибко, насколько это возможно. Все остальное должно быть сделано на периферии

Вводя сервис в сеть, необходимо тщательно проанализировать то, как он может повлиять на работу приложений, архитектуру сети, ее протоколы.

Базовые принципы Интернета

- ***Принцип уровневости***
- ***Принцип инкапсуляции данных***
- ***Принцип коммутации пакетов***
- ***Принцип доменного именования и адресации***
- ***E2E принцип***



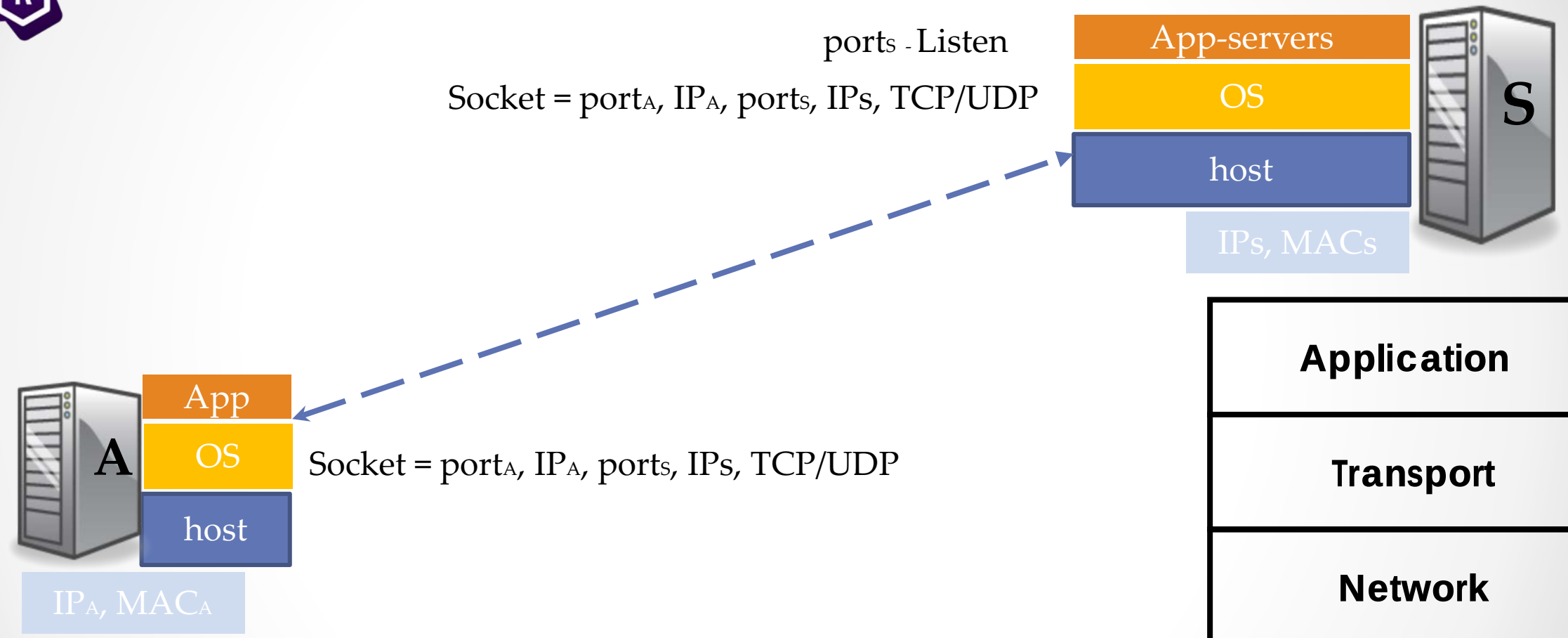
Интернет: модель сервисов сетевого и транспортного уровней

Введение в компьютерные сети

чл.-корр. РАН. Смелянский Р.Л.

Кафедра АСВК ф-т ВМК МГУ

Соединение на прикладном уровне



Application
Transport
Network
Link

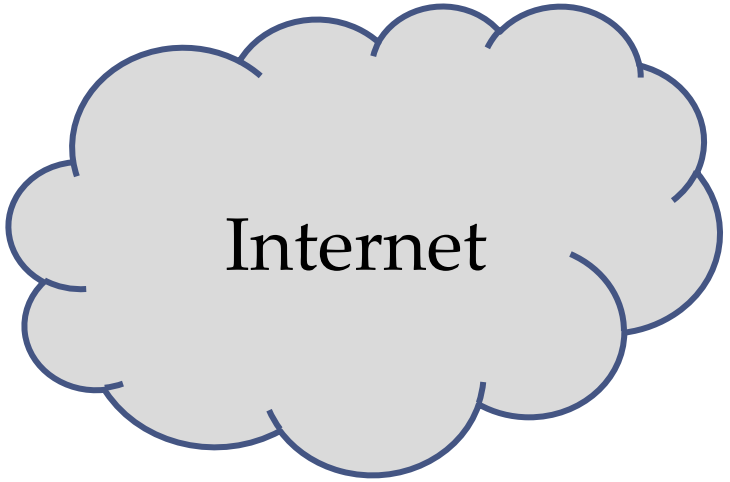
Работа с номерами портов

- Приложению доступно динамически назначаемый ОС уникальный порт (ephemeral port)
- Файл /etc/services – список известных портов в TCP/IP UNIX
- TCP и UDP разные пулы портов
- По умолчанию приложению выдается порт с произвольным номером
- При необходимости приложение может запросить конкретный номер порта.
- Определение номера порта отправителя (получателя ответа):
  - при подключении по TCP определяется по заголовку протокола
  - использование обратных номеров портов в UDP зависит от реализации

Поток байтов в TCP

address:171.76.76.157
Port: 23946

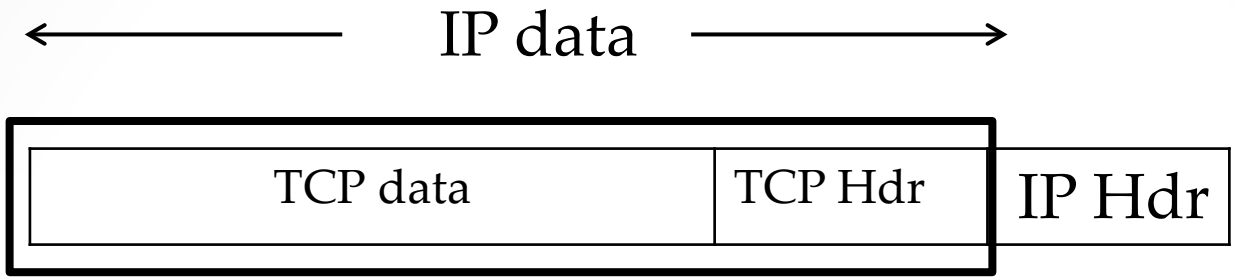
Client

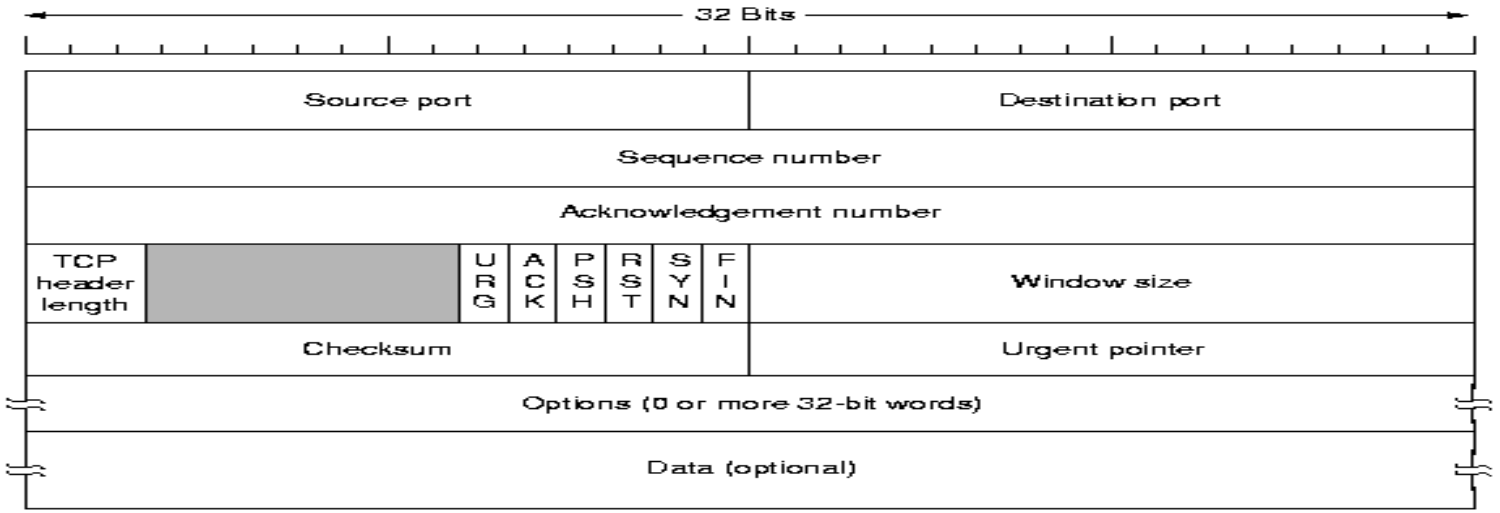


address:74.125.127.103
Port: 80

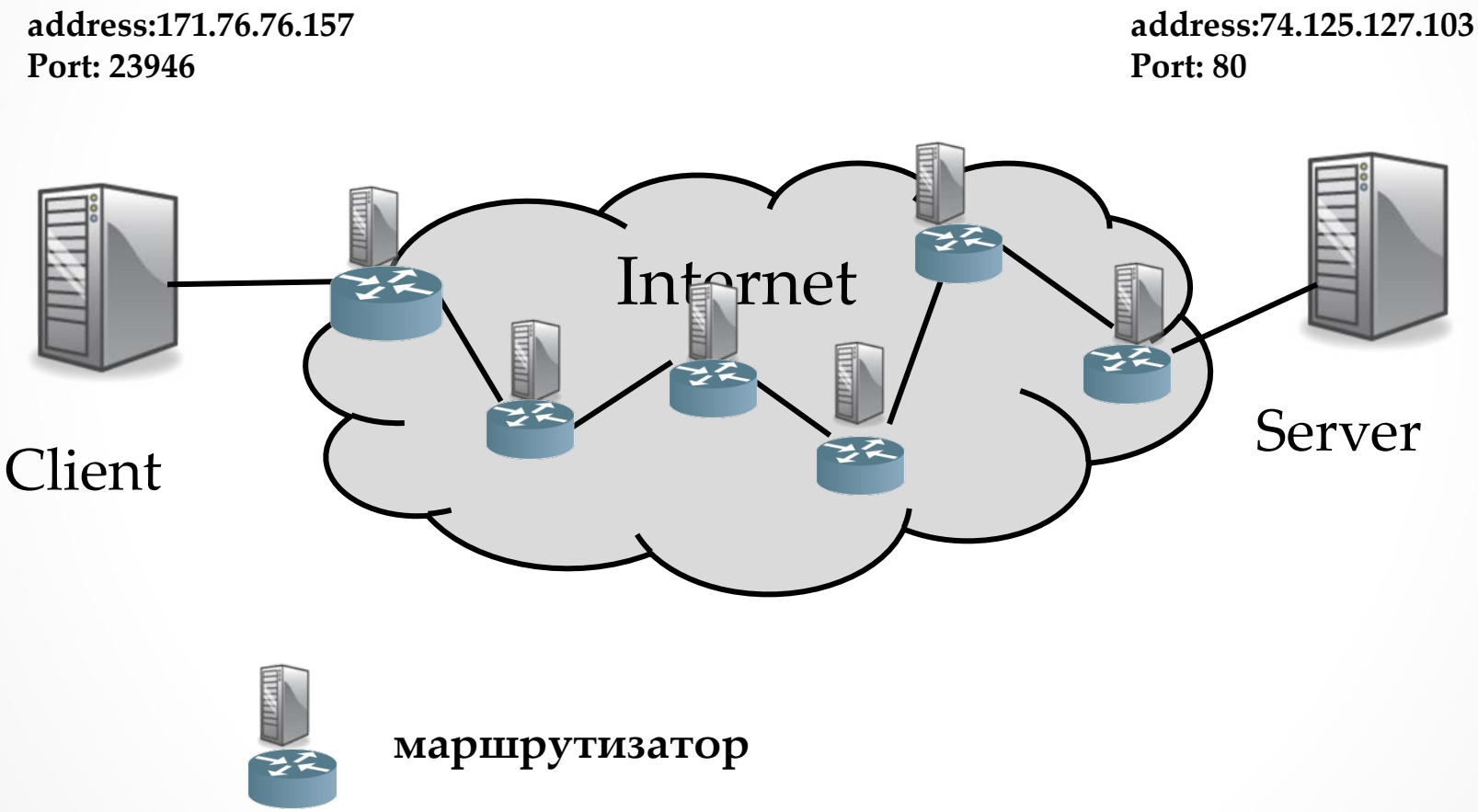
Server

TCP сегмент

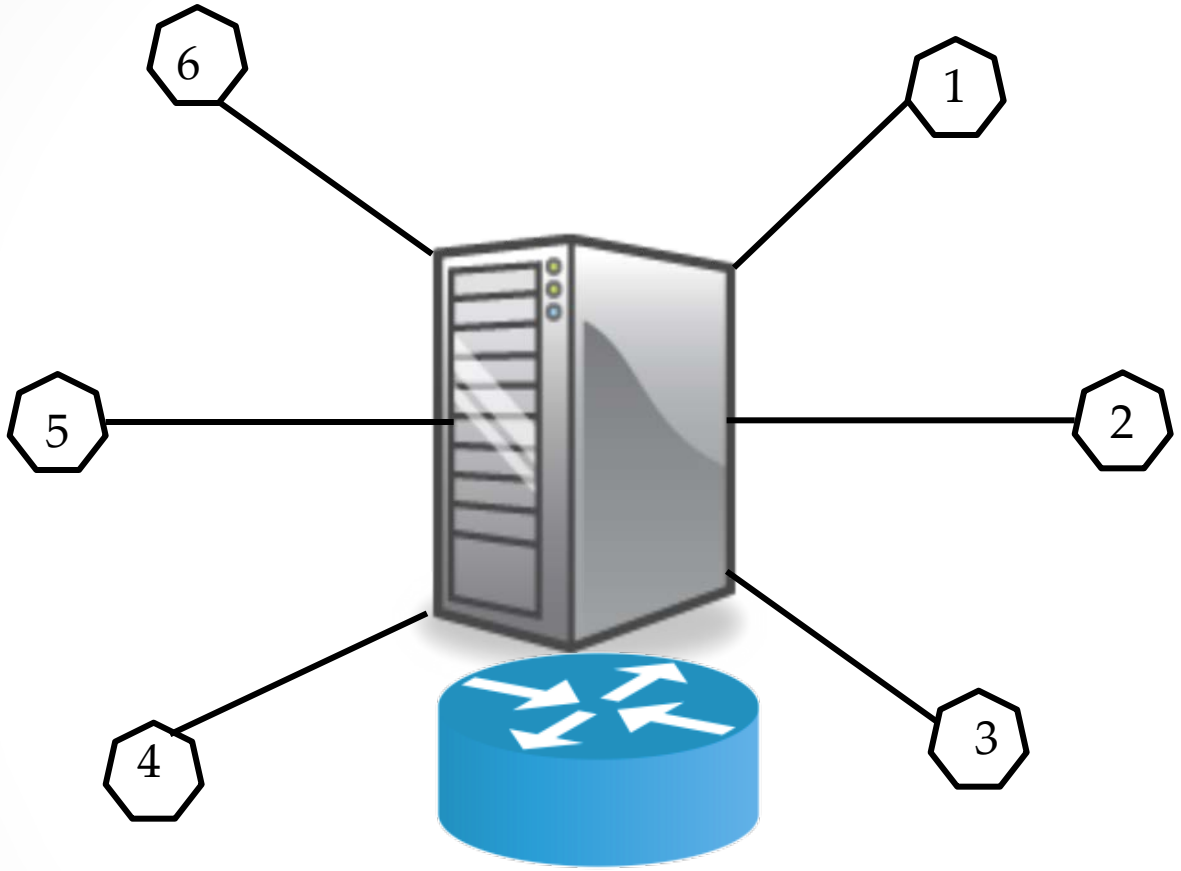

IP data
TCP data
TCP Hdr
IP Hdr


32 Bits
Source port
Destination port
Sequence number
Acknowledgement number
TCP header length
URG
ACK
PSH
RST
SYN
FIN
Window size
Checksum
Urgent pointer
Options (0 or more 32-bit words)
Data (optional)

Внутри потока



Внутри каждого перехода (hope)

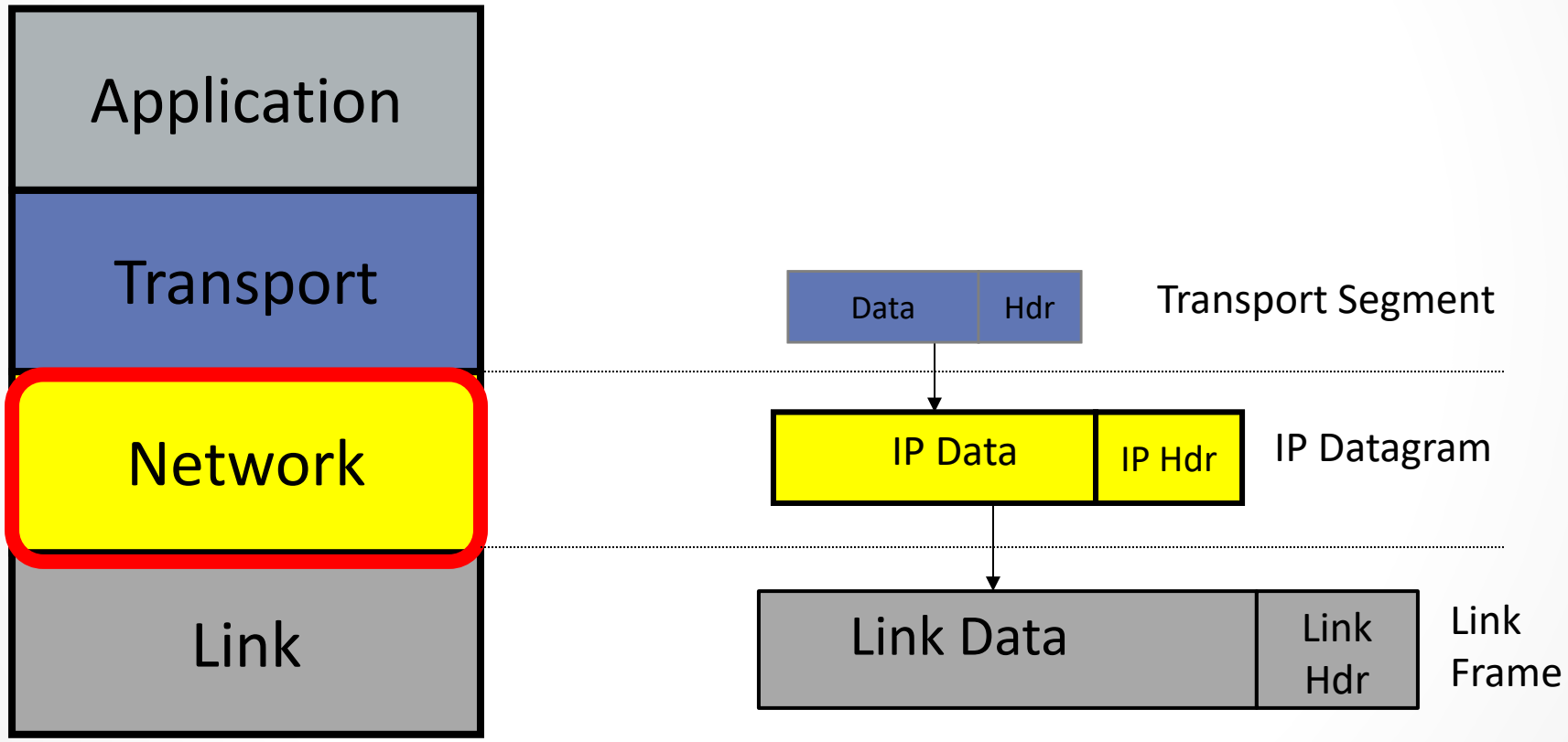


address	link
default	1
23.x.x.x	5
28.33.5.x	2
171.32.x.x	4
171.33.x.x	2
67.x.x.x	6
216.x.x.x	1

Модель IP сервиса

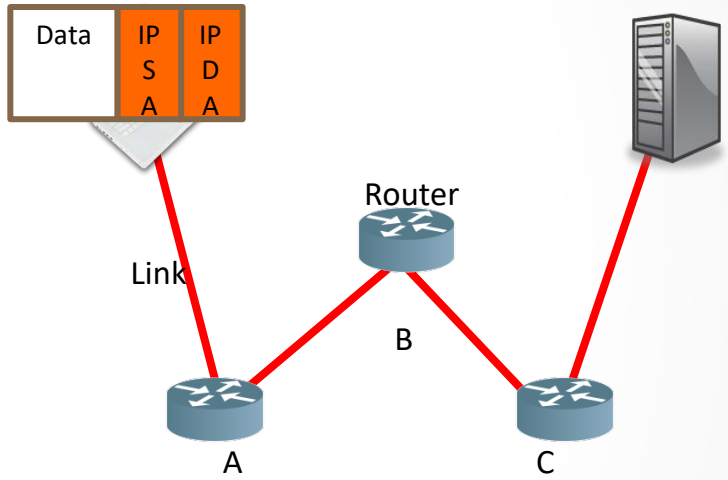
- IP работает над любой физической средой
- Над IP работают различные транспортные протоколы
- Поверх транспорта работают прикладные протоколы

IP протокол


Application
Transport
Network
Link
Data
Hdr
Transport Segment
IP Data
IP Hdr
IP Datagram
Link Data
Link Hdr
Link Frame

модель IP сервиса

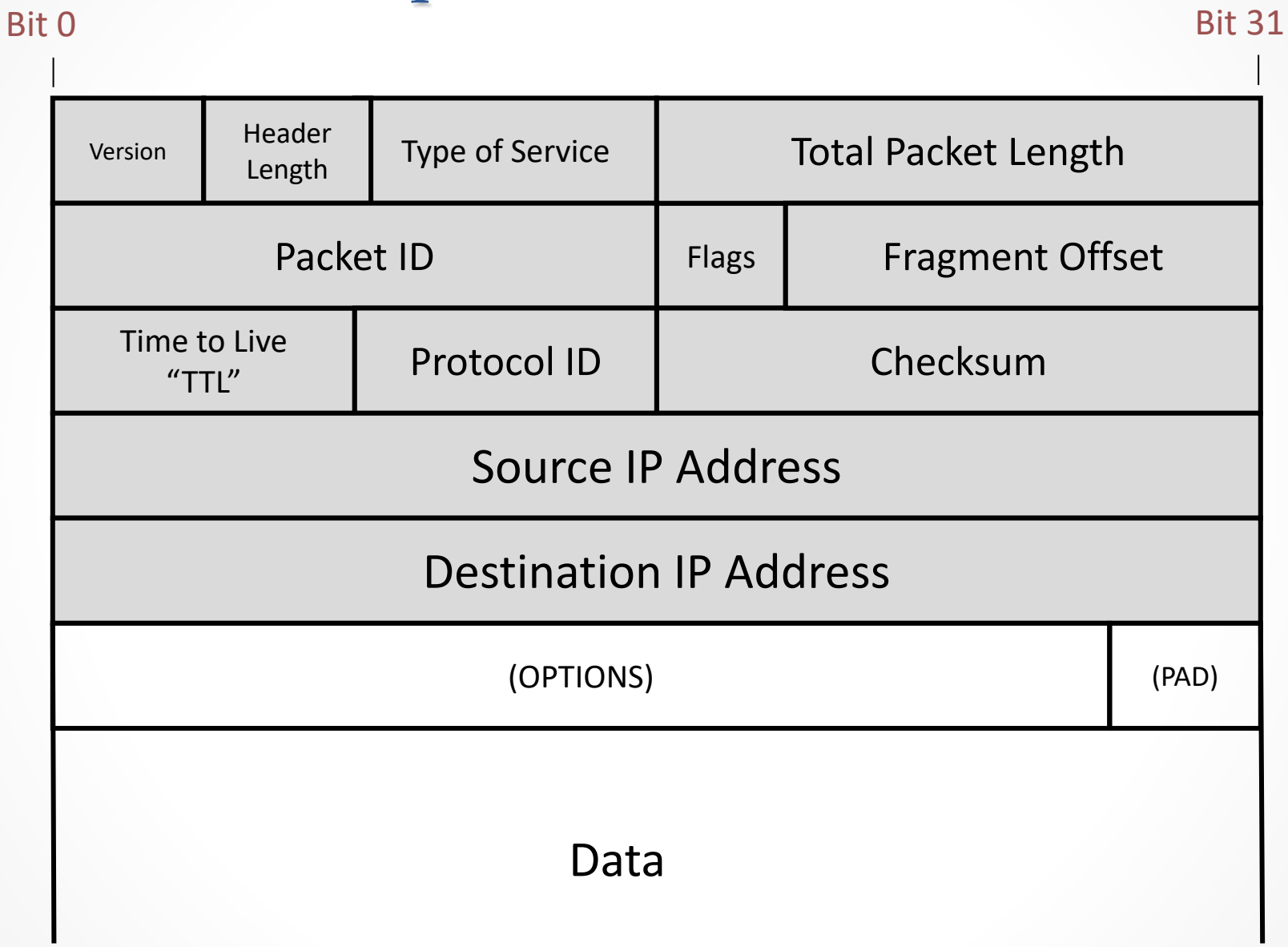
Свойство	Поведение
Datagram	Индивидуально маршрутизируемый скачками пакет
Ненадежный	Пакет может быть сброшен
По возможности (best effort)	…. но только при необходимости
Без соединения	Нет состояния у потока Последовательность пакетов может быть нарушена



модель IP сервиса

- Предотвращает «зацикливание» пакетов
- Фрагментирует сегменты если они слишком длинные
- Использует контрольную сумму, чтобы сократить возможность доставки в неправильное место назначения
- Две версии
  - *IPv4 с 32 битным адресом*
  - *IPv6 с 128 битным адресом*
- Позволяет добавлять новые опции к заголовку
- Stateless

Формат пакета IPv4



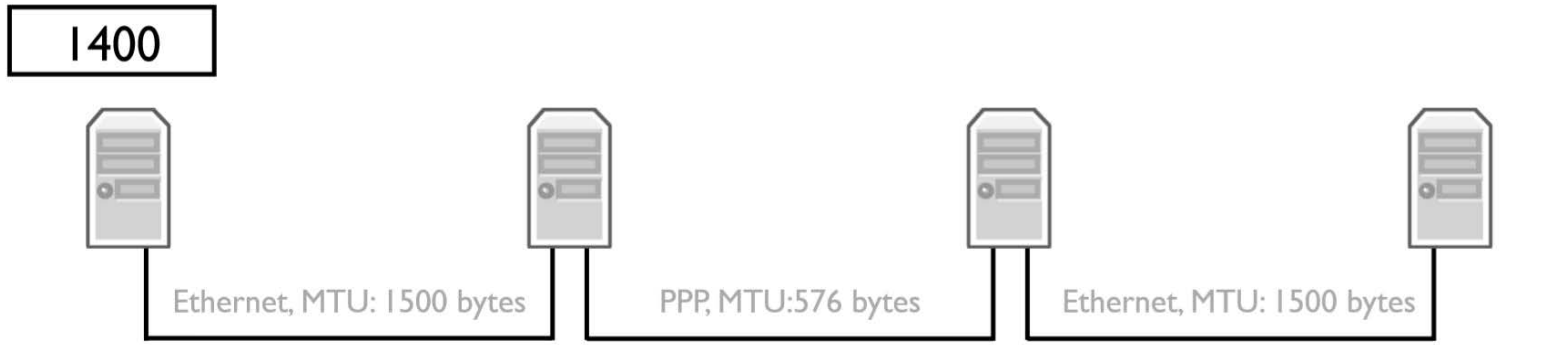
Options

Options	Значение
Security	Уровень секретности и способ шифрования
Strict source routing	Задан маршрут (Rtrs), по которому должен пройти пакет
Loose source routing	Заданы Rtrs, через которые обязан пройти пакет
Record route	Каждый Rtr, через который проходит пакет, отмечает себя
Time stamp	Каждый Rtr отмечает свой адрес и время прохождения

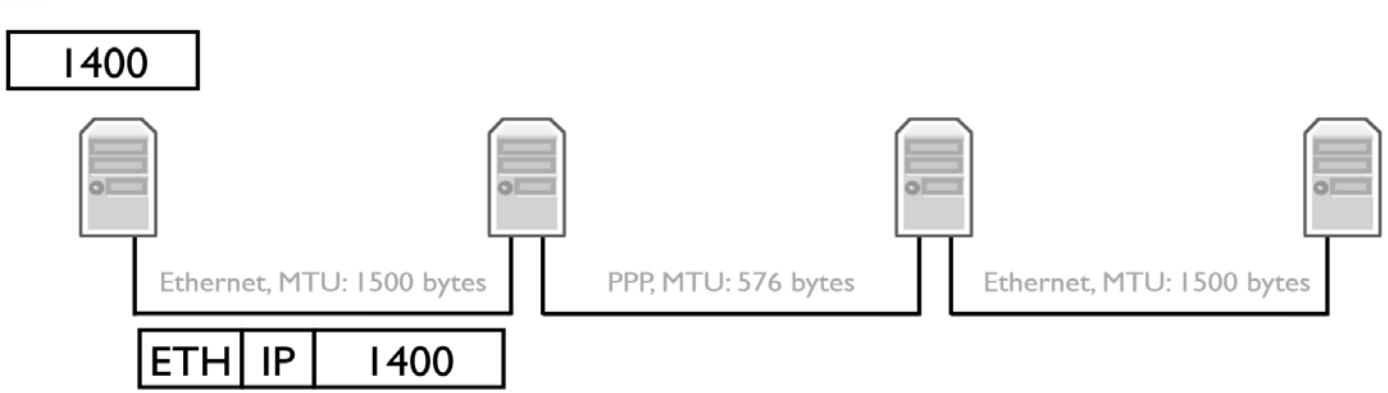
Фрагментация и сборка

- *Проблема возникает когда PDU верхнего уровня превышает PDU на нижних уровнях*
- *Фрагментация: получаем PDU большого размера и разбиваем его на короткие порции*
- *Сборка: получаем набор коротких порций данных и собираем их в нужном порядке.*
- *Пример:*
  - *Транспорт: TCP получает поток байт и разделает его на TCP сегменты*
  - *Сетевой: IP пакет может оказаться слишком коротким для TCP сегмента*
  - *Канальный: IP пакет длиннее кадра (Ethernet кадр – 1,5КБ, IP packet – 64КБ)*

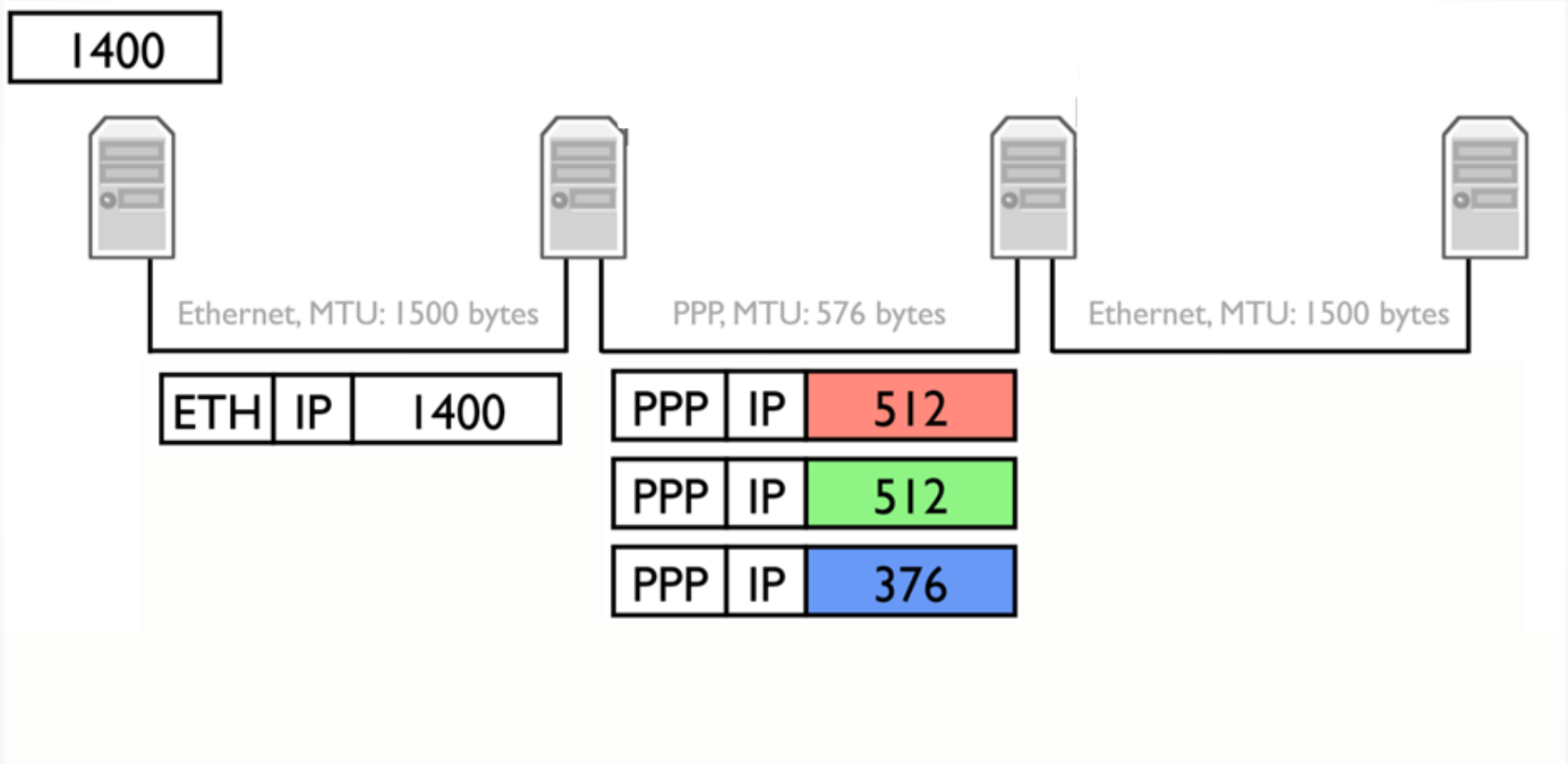
Фрагментация: пример на IP



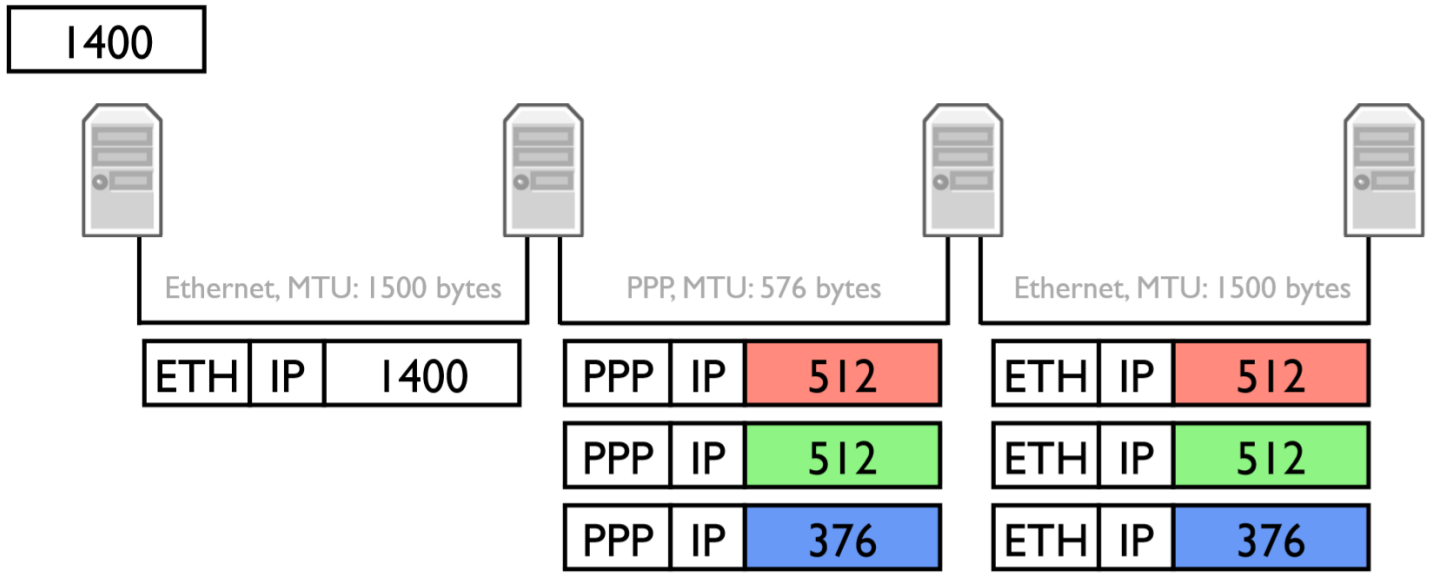
Фрагментация: пример на IP



Фрагментация: пример на IP



Фрагментация: пример на IP



IP фрагментация

- *IP адреса + поле ident идентифицируют фрагменты пакета*
- *Флаг MF =1 у всех фрагментов кроме последнего*
- *Поле offset указывает смещение фрагмента (8 байтовыми порциями)*
- *Длины всех фрагментов кроме последнего должны быть кратны 8*

начало заголовка
ident= *x* | | | | 0 offset = *0*
продолжение заголовка
1400 байт

После фрагментации

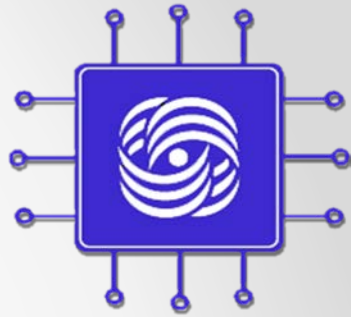
начало заголовка
ident= *x* | | 1 offset = *0*
продолжение заголовка
512 байт

начало заголовка
ident= *x* | | 1 offset = 64
продолжение заголовка
512 байт

начало заголовка
ident= *x* | | *0* offset = *128*
продолжение заголовка
376 байт

Фрагментация: пример TCP

- *Избегай IP фрагментации всегда, когда можно*
- *TCP может выбирать размер сегмента так, чтобы избежать фрагментации*
- *Чтобы определить MTU, используй DF флаг (запрет фрагментации), и смотри получишь ли ICMP сообщение (подробности в RFC 1191)*



Интернет: адресация на сетевом уровне

Введение в компьютерные сети

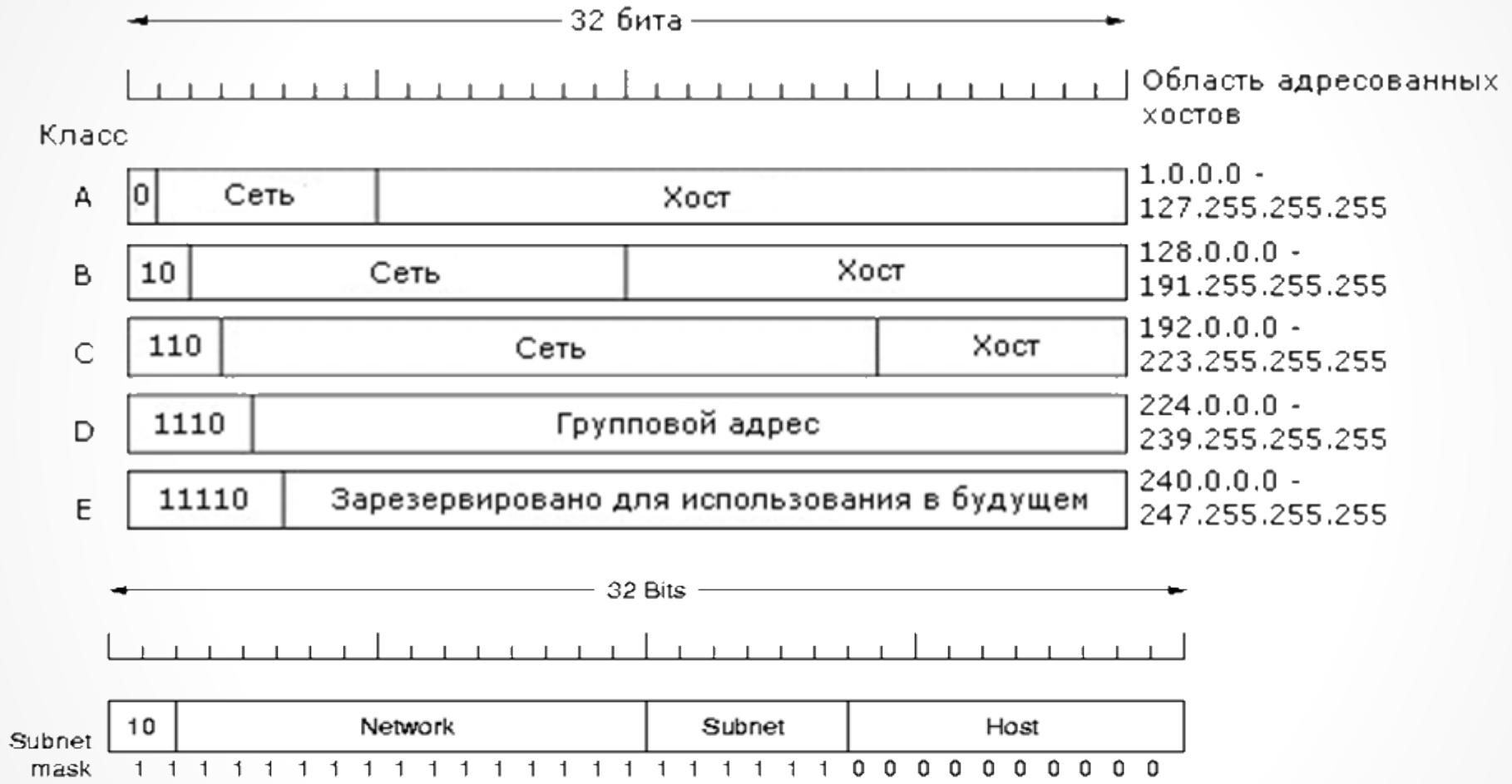
чл.-корр. РАН. Смелянский Р.Л.

Кафедра АСВК ф-т ВМК МГУ

Адресация

- Адресация на транспортном уровне
- Адресация на сетевом уровне
- Адресация на канальном уровне (Ethernet)
- Протоколы преобразования адресов
- Что такое протокол?

Классы IP адресов



Групповая адресация в Internet

- 224.0.0.1 – все хосты в данной сети.
- 224.0.0.2 – все маршрутизаторы в данной сети.
- 224.0.0.5 – все OSPF-маршрутизаторы в данной сети.
- 224.0.0.6 – все выделенные OSPF маршрутизаторы в данной сети.

CIDR - бесклассовая маршрутизация внутри домена

- 194.0.0.0 - 195.255.255.255 - Европа
- 198.0.0.0 - 199.255.255.255 - Северная Америка
- 200.0.0.0 - 201.255.255.255 - Центральная и Южная Америка
- 202.0.0.0 - 203.255.255.255- Азия и Тихий Океан

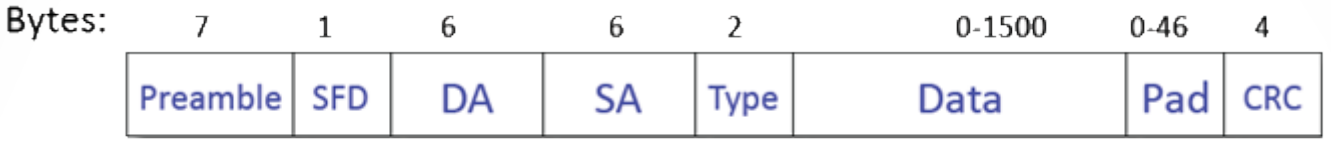
- каждый регион получил 32 миллиона адресов

IPv6
(том 2 стр.90-97)

- **Расширено адресное пространство:** в IPv6 используется 128 разрядный адрес (увеличивает количество адресов в 2^{96} раз, что позволяет иметь $6{,}7 \times 10^{23}$ адресов на один квадратный метр поверхности земли!)
- **Расширен механизм вариантов (опций)**
- **Автоматическое назначение адресов**
- **Улучшена гибкость групповой адресации.**
- **Поддержка распределения ресурсов:** вместо поля Тип сервиса в IPv4, IPv6 позволяет отмечать пакеты, принадлежащие к конкретному потоку данных, и для которых отправитель требует специальной обработки.

заголовок	расширение	. . .	расширение	TPDU

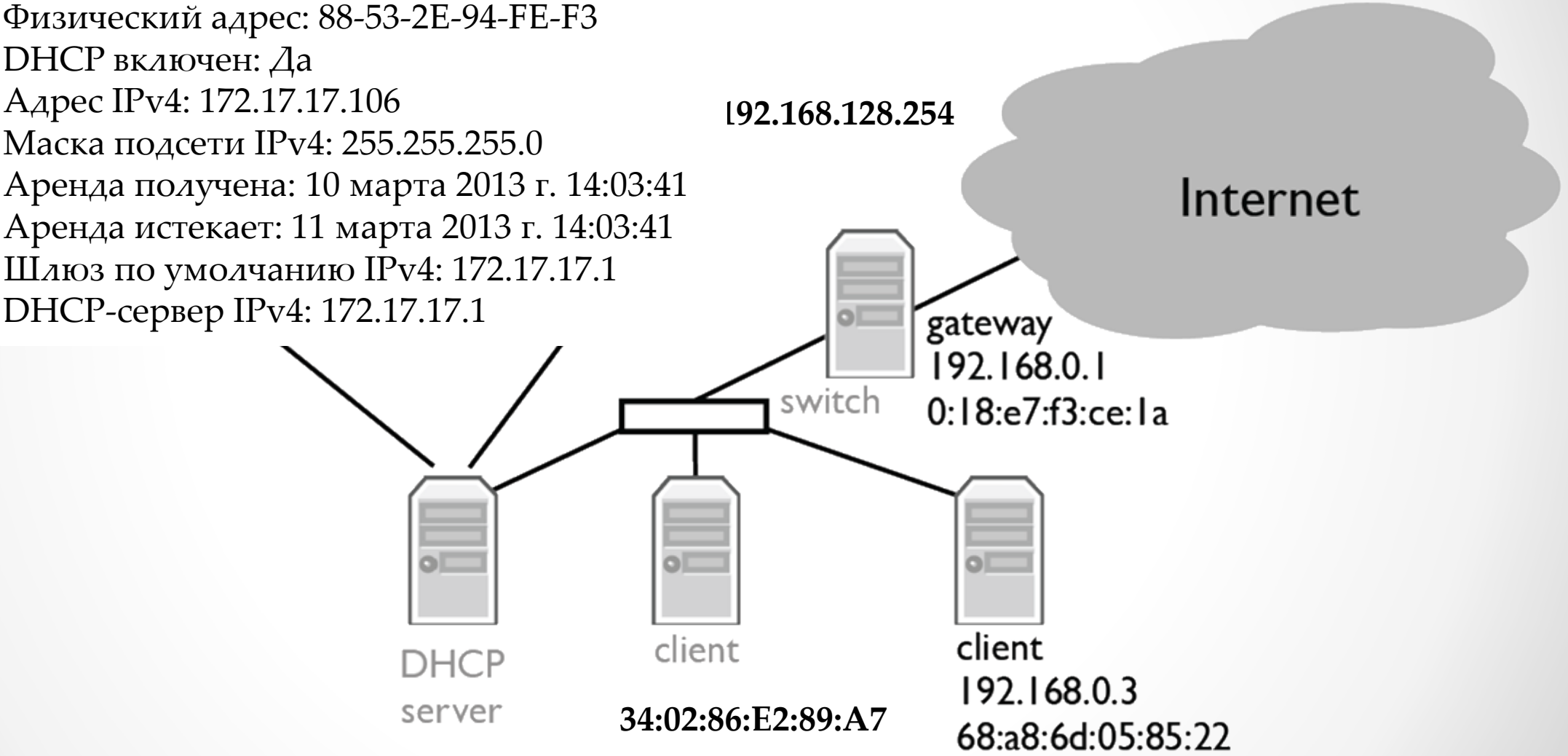
Адрес на канальном уровне – Ethernet адресация



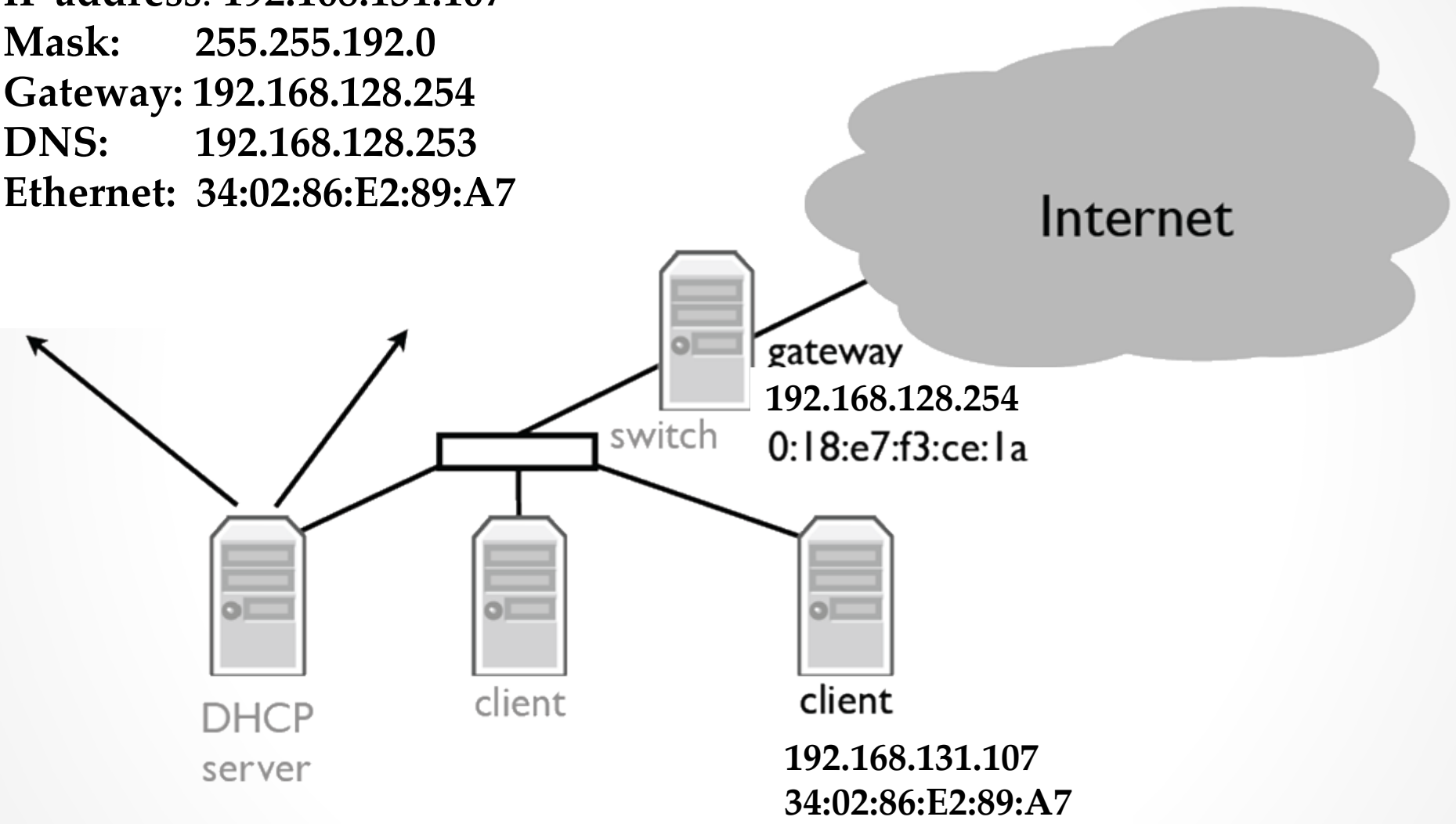
- *Это формат Ethernet кадра (подробно рассмотрим позднее*
- *Адрес 48 разрядный уникальный (присваивается производителем)*
  - *48-й разряд – признак локальности*
  - *47-й разряд – признак групповой адресации*
- *Type – тип протокола верхнего уровня*

Пример проблемы

Описание: Intel(R) Centrino(R) Advanced-N 6230
Физический адрес: 88-53-2E-94-FE-F3
DHCP включен: Да
Адрес IPv4: 172.17.17.106
Маска подсети IPv4: 255.255.255.0
Аренда получена: 10 марта 2013 г. 14:03:41
Аренда истекает: 11 марта 2013 г. 14:03:41
Шлюз по умолчанию IPv4: 172.17.17.1
DHCP-сервер IPv4: 172.17.17.1



IP address: 192.168.131.107
Mask: 255.255.192.0
Gateway: 192.168.128.254
DNS: 192.168.128.253
Ethernet: 34:02:86:E2:89:A7



Address Resolution Protocol

- *Отображает адреса уровня 3 на адреса уровня 2*
- *Принцип работы – «запрос-ответ»*
  - *«У кого сетевой адрес X?»*
  - *«У меня сетевой адрес X и вот мой адрес уровня 2»*
- *Запрос рассылается лавиной на канальном уровне*
- *Ответ шлют непосредственно тому кто запрашивал*

Динамические vs статтические записи в ARP cache
arp -s <IP-адрес> <MAC-адрес>

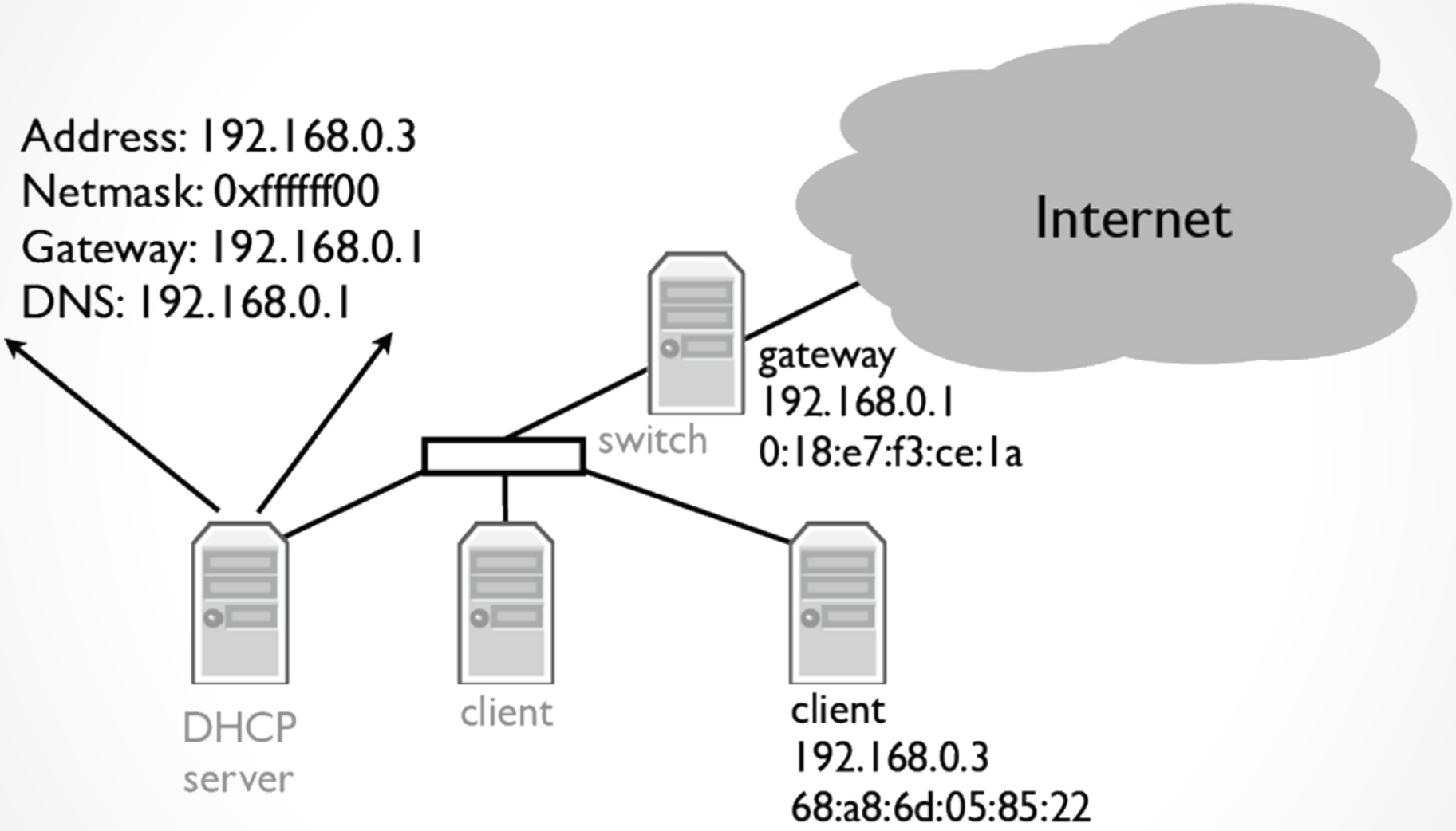
Формат ARP пакета

+	Bits 0 — 7	8 — 15	16 — 31
0	Hardware type = 0x0001(Ethernet)		Protocol type = 0x0800 (IPv4)
32	Hardware length = 6	Protocol length = 4	Operation = 1
64	Source Hardware Address (first 32 bits)		
96	SHA (last 16 bits)		SPA (first 16 bits)
128	SPA (last 16 bits)		DHA (first 16 bits)
160	Destination Hardware Address (last 32 bits)		
192	Destination Protocol Address		

Структура ARP пакета

Hardware (HTYPE)
Каждый протокол уровня 2 имеет свой номер, который хранится в этом поле. Например, Ethernet имеет номер 0x0001.
Protocol (PTYPE)
Код протокола уровня 3. Например, для IPv4 будет записано 0x0800.
Hardware length (HLEN)
Длина адреса уровня 2 в байтах. Ethernet адреса имеют длину 6 байт.
Protocol length (PLEN)
Длина адреса уровня 3 в байтах. IPv4 адреса имеют длину 4 байта.
Operation
Код операции отправителя: 1 в случае запроса и 2 в случае ответа.
Source hardware address (SHA)
Адрес уровня 2отправителя.
Source protocol address (SPA)
Адрес уровня 3 отправителя.
Destination hardware address (THA)
Адрес уровня 2 получателя. Поле пусто при запросе.
Destination protocol address (TPA)
Адрес уровня 3 получателя.

ARP запрос


Address: 192.168.0.3
Netmask: 0xffffff00
Gateway: 192.168.0.1
DNS: 192.168.0.1
Internet
gateway
192.168.0.1
0:18:e7:f3:ce:1a
switch
DHCP
server
client
client
192.168.0.3
68:a8:6d:05:85:22

Протоколы преобразования адресов

RARP - протокол сетевого уровня модели OSI, выполняет обратное отображение адресов, то есть преобразует физический адрес запросившего в его IP-адрес.

BOOTP - (англ. сокращение от Bootstrap Protocol) — сетевой протокол, используемый для автоматического получения клиентом IP-адреса; позволяет бездисковым рабочим станциям получать IP-адрес прежде, чем будет загружена полноценная операционная система.

DHCP - (англ. Dynamic Host Configuration Protocol — протокол динамической конфигурации узла) — это сетевой протокол, позволяющий компьютерам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети TCP/IP. Данный протокол работает по модели «клиент-сервер».

Примеры

- Запросить www.arccn.ru
- Используя wireshark, рассмотреть поток байтов, установленный TCP протоколом

(Wireshark можно скачать тут: http://www.wireshark.org/download.html)

- Используя traceroute, рассмотреть поток и маршрут пакетов в Интернете

- Везде идет обмен пакетами. Взаимодействие между приложениями в Интернете реализуется через обмен пакетами!

Заключение

- Каждый уровень – своя адресация
- TCP – надежный с состояниями
- IP ненадежный, без состояний
- Это очень простой сервис
  - Datagram
  - Ненадежный
  - Best-effort
  - Без соединения
  - Управление сервисами через заголовок
  - Управление передачей и собственно передачи не отделимы