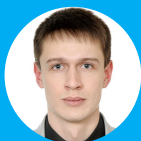


Основы работы с сетью. Модель OSI, TCP, UDP



Григорий
Вахмистров



Григорий Вахмистров

Backend Developer в Tennisi.bet

План занятия

1. [Вспомним прошлый урок](#)
2. [Как доставить сообщение?](#)
3. [Модель OSI](#)
4. [Модель TCP/IP](#)
5. [Практика: кодим сокеты](#)
6. [Итоги](#)
7. [Домашнее задание](#)



Вспоминаем прошлый урок



Зачем нужен Mock (заглушка)?

Поставьте в чате номер **одного** правильного ответа.

1. Заглушить двигатель Java
2. Тестировать заглушки/пустые объекты
3. Тестировать модули, имеющие зависимости на другие сложные модули
4. Определять, когда какой-либо модуль/объект является “заглушкой” и бесполезен

Зачем нужен Mock (заглушка)?

Поставьте в чате номер **одного** правильного ответа.

1. Заглушить двигатель Java
2. Тестировать заглушки/пустые объекты
3. Тестировать модули, имеющие зависимости на другие сложные модули
4. Определять, когда какой-либо модуль/объект является “заглушкой” и бесполезен



Какие есть способы создать Mock?


Поставьте в чате номер **двух** правильных ответов.

1. Воспользоваться Mock из java-core
2. Создать Mock средствами библиотеки Mockito
3. Реализовать класс как заглушку вручную, реализовав интерфейс исходного класса
4. Воспользоваться онлайн-версией

Какие есть способы создать Mock?

Поставьте в чате номер **двух** правильных ответов.

1. Воспользоваться Mock из java-core
2. Создать Mock средствами библиотеки Mockito
3. Реализовать класс как заглушку вручную, реализовав интерфейс исходного класса
4. Воспользоваться онлайн-версией



Есть ли возможность проверить, сколько раз был вызван метод у Mock'а в Mockito?

Поставьте в чате номер **одного** правильного ответа.

1. Нет, это ограничение библиотеки Mockito
2. Нет, такое в принципе невозможно в Java
3. Да, можно воспользоваться Spy объект
4. Да, для этого используется Verify метод

Есть ли возможность проверить, сколько раз был вызван метод у Mock'а в Mockito?

Поставьте в чате номер **одного** правильного ответа.

1. Нет, это ограничение библиотеки Mockito
2. Нет, такое в принципе невозможно в Java
3. Да, можно воспользоваться Spy объект
4. **Да, для этого используется Verify метод**



Почему не рекомендуется использовать Sru?

Поставьте в чате номер **двух** правильных ответов.

1. Есть ограничения со стороны Java
2. Есть ограничения со стороны Mockito
3. Необходимость использовать Sru говорит о плохом дизайне
4. Свой вариант (написать в чате)

Почему не рекомендуется использовать Sru?

Поставьте в чате номер **двух** правильных ответов.

1. Есть ограничения со стороны Java
2. Есть ограничения со стороны Mockito
3. **Необходимость использовать Sru говорит о плохом дизайне**
4. **Свой вариант**



Какие заглушки можно создать в Mockito?

Поставьте в чате номер **одного** правильного ответа.

1. Заглушки из `final` классов
2. Заглушки к `private` методам
3. Заглушки к `static` методам
4. Заглушки к классам из другого пакета
5. Заглушки к конструкторам
6. Заглушки к методам `equals` и `hashCode()`

Какие заглушки можно создать в Mockito?

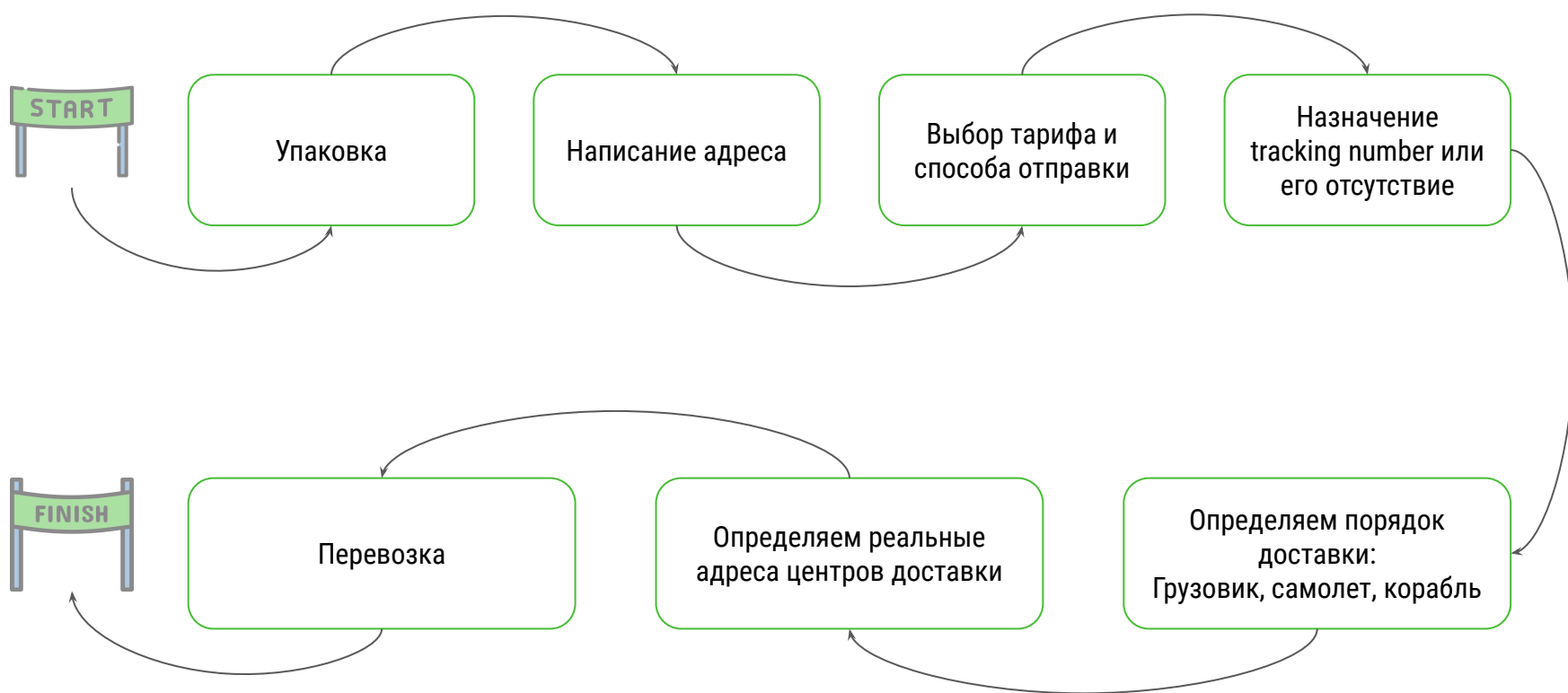
Поставьте в чате номер **одного** правильного ответа.

1. Заглушки из `final` классов
2. Заглушки к `private` методам
3. Заглушки к `static` методам
4. **Зажлушки к классам из другого пэкеджа**
5. Заглушки к конструкторам
6. Заглушки к методам `equals` и `hashCode()`



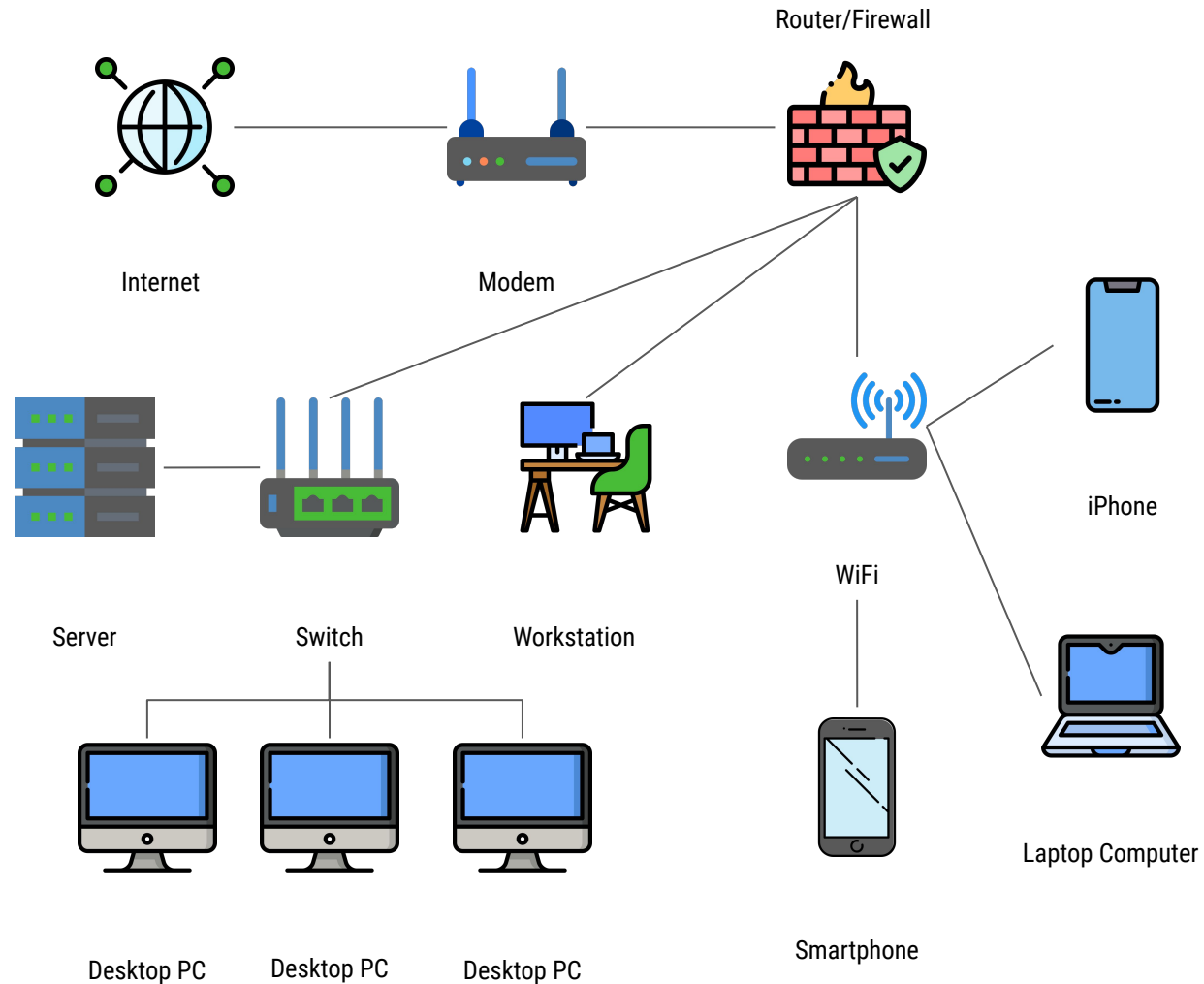
Как доставить сообщение?

Как работает почта



Как передать данные между компьютерами?

Что нам надо для
этого знать?

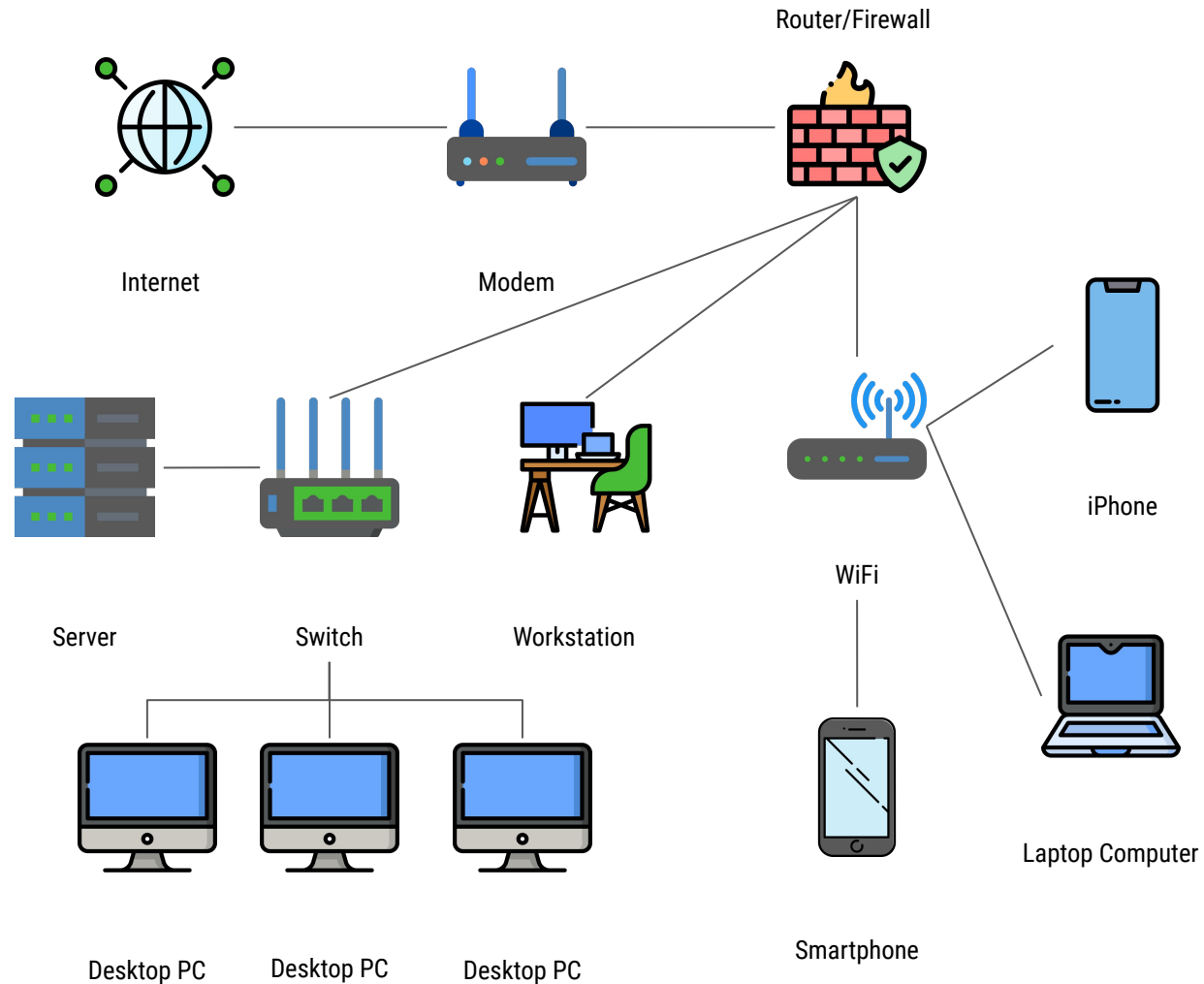


Как передать данные между компьютерами?

Что нам надо для этого знать?

01 Адрес назначения

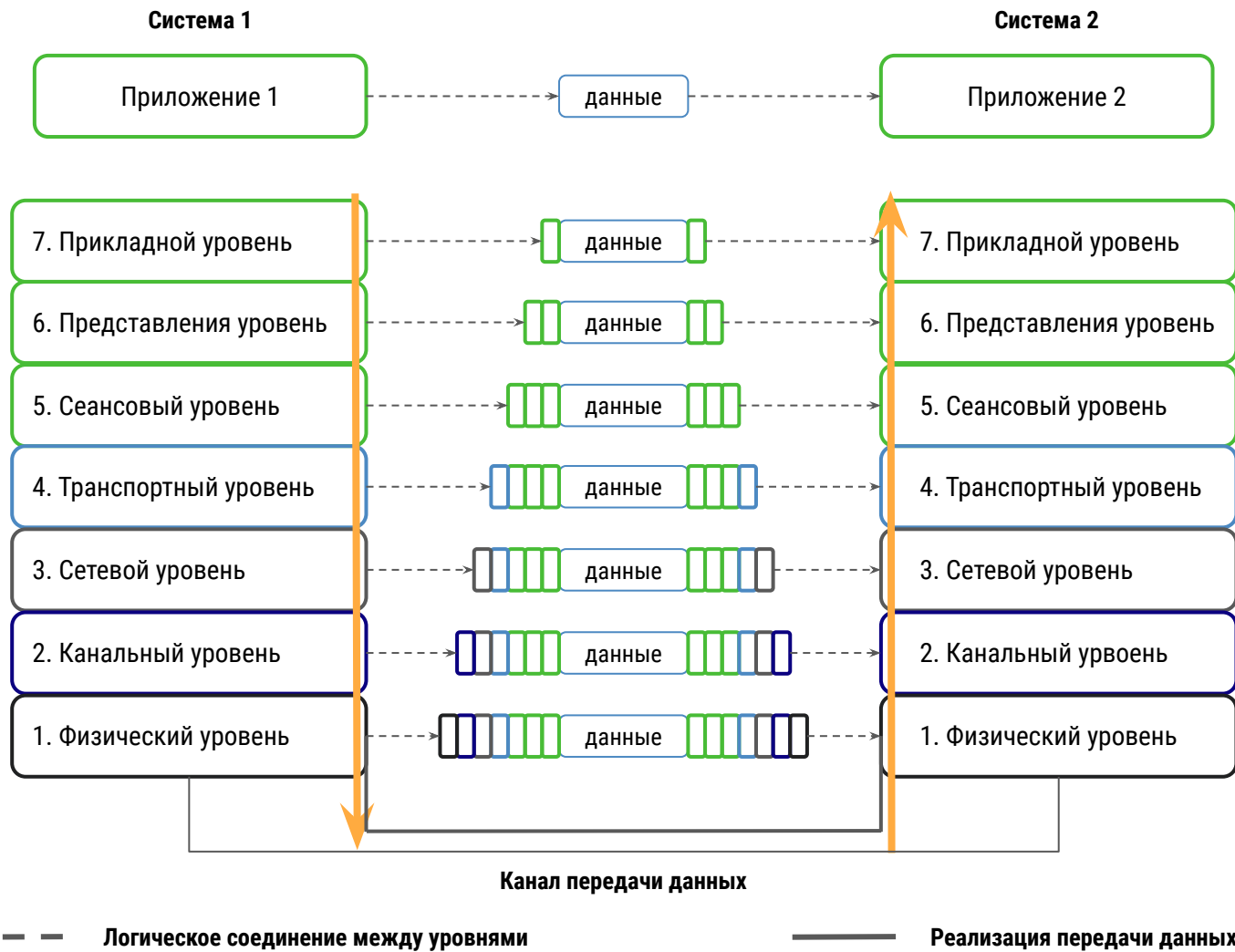
02 Как добраться





Модель OSI

Модель OSI



Уровень приложений

Верхний уровень модели,
обеспечивающий взаимодействие
пользовательских приложений с сетью

- FTP
- TELNET
- HTTP

7. Прикладной уровень

6. Представления уровень

5. Сеансовый уровень

4. Транспортный уровень

3. Сетевой уровень

2. Канальный уровень

1. Физический уровень

Уровень представления

Преобразование форматов сообщений, такое как кодирование, шифрование или сжатие

- SSL
- JPEG

7. Прикладной уровень

6. Представления уровень

5. Сеансовый уровень

4. Транспортный уровень

3. Сетевой уровень

2. Канальный уровень

1. Физический уровень

Сеансовый уровень

Обеспечивает поддержание сеанса связи, позволяя приложениям взаимодействовать между собой длительное время

НЕ БЫЛО НИ ЕДИНОГО
РАЗРЫВА!

7. Прикладной уровень

6. Представления уровень

5. Сеансовый уровень

4. Транспортный уровень

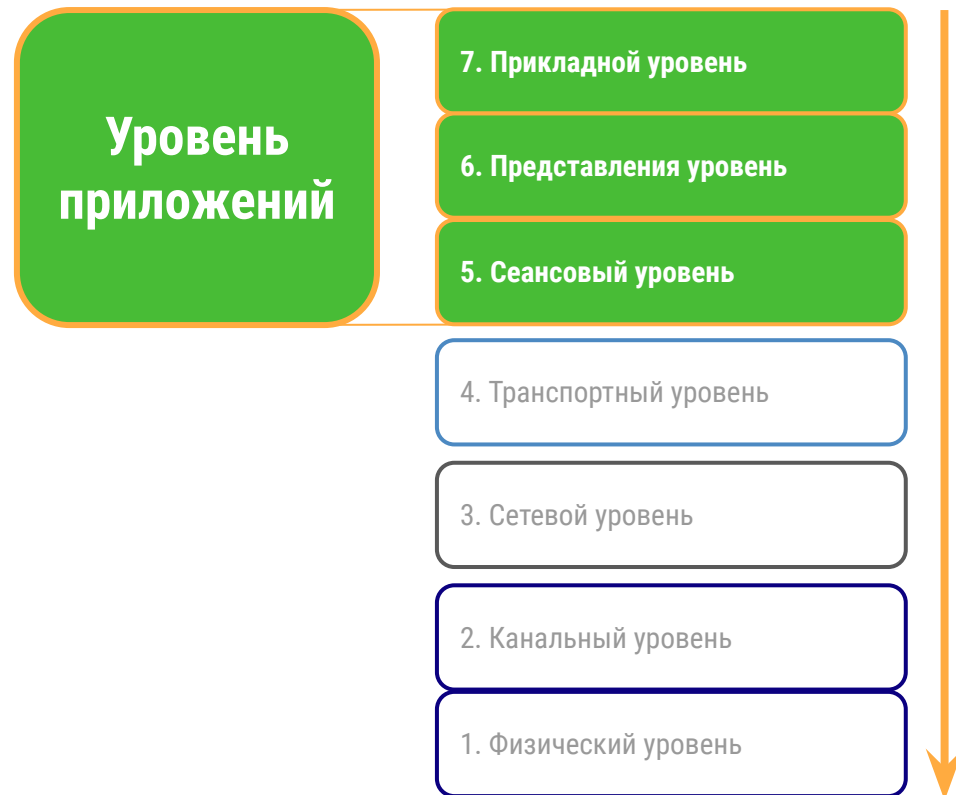
3. Сетевой уровень

2. Канальный уровень

1. Физический уровень

Уровень приложений Application layer

На этом уровне работают большинство сетевых приложений. Самый важный уровень для программиста



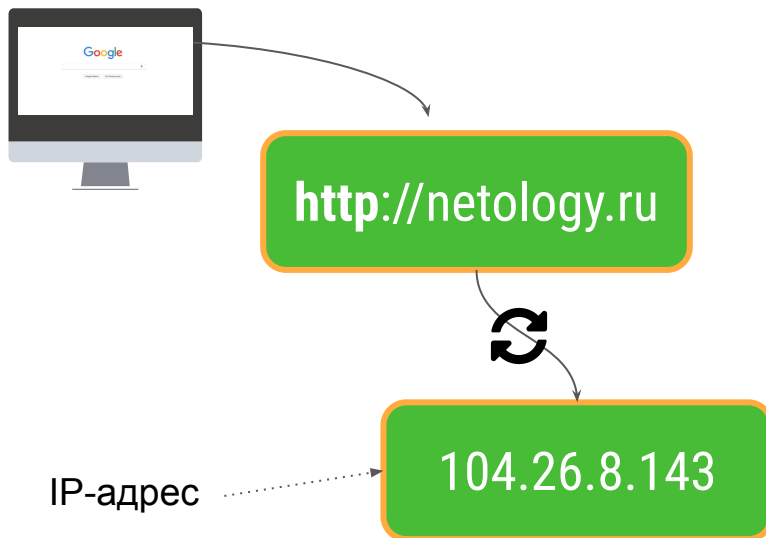
Уровень приложений



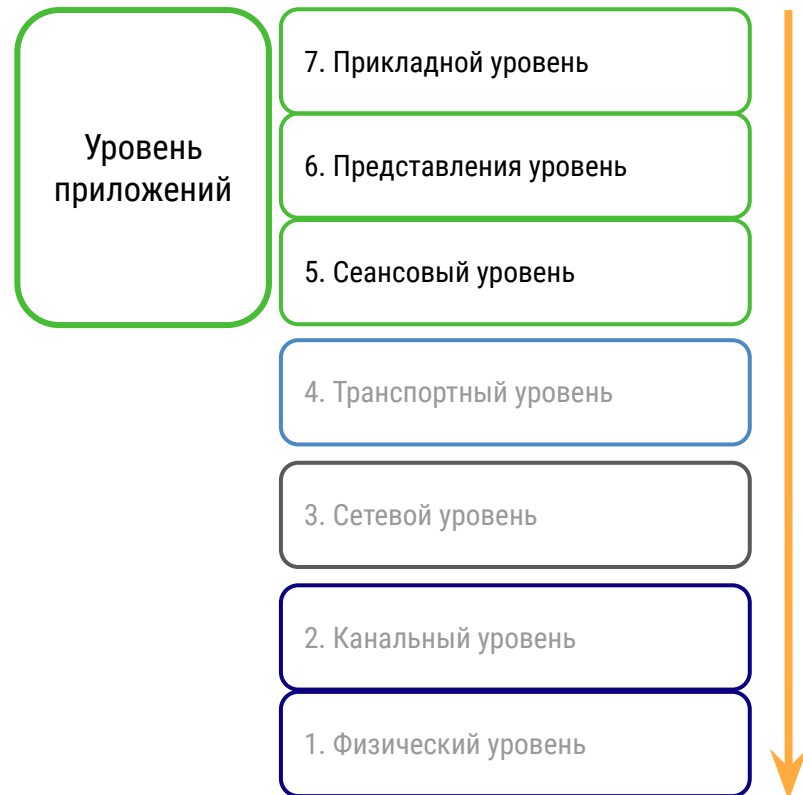
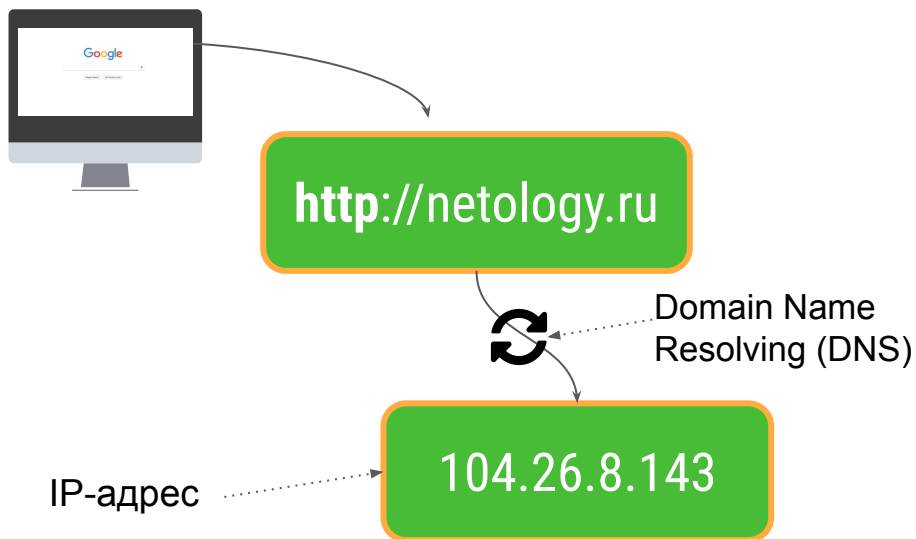
<http://netology.ru>



Уровень приложений

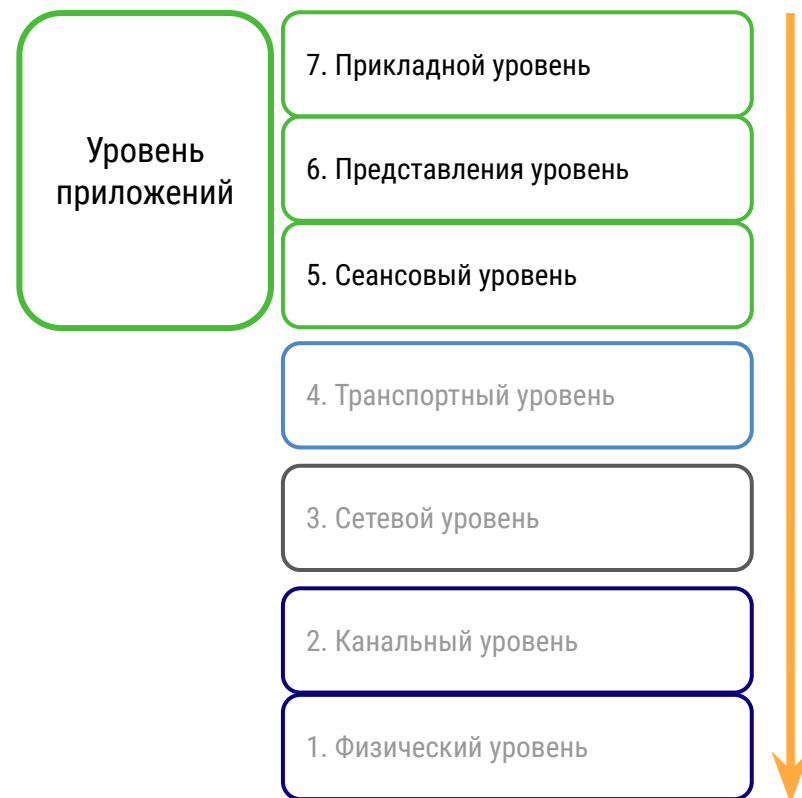


Уровень приложений



Уровень приложений: DNS

DNS (domain name server) - позволяет узнать адрес назначения по имени домена (сайта)



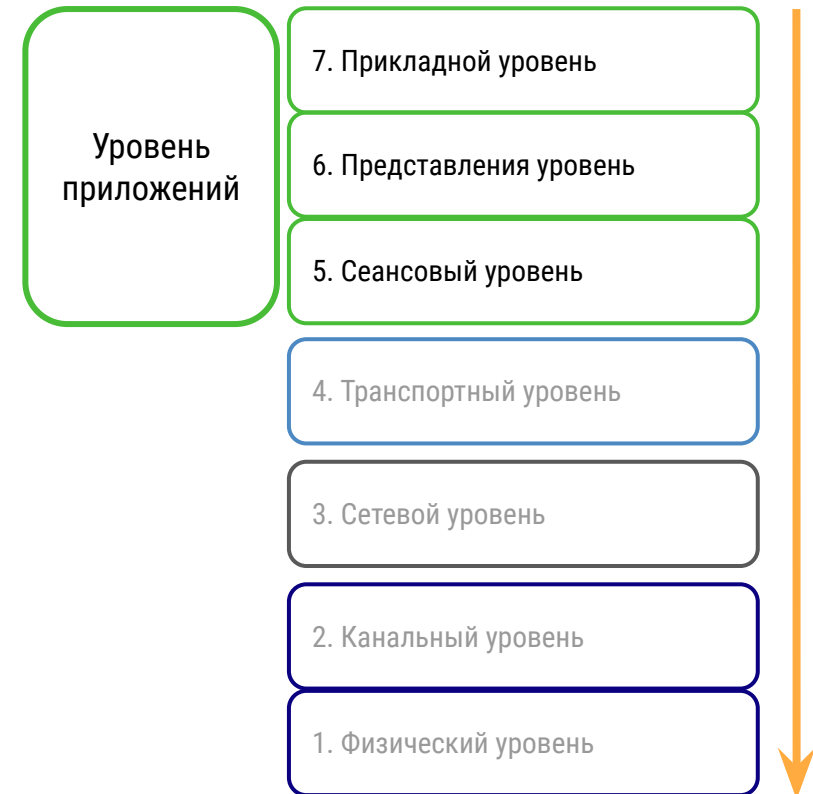
Уровень приложений: DNS

DNS (domain name server) - позволяет узнать адрес назначения по имени домена (сайта)

```
alexey@MacBook-Pro-Alexey ~ % nslookup netology.ru
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:   netology.ru
Address: 104.26.8.143
Name:   netology.ru
Address: 104.26.9.143
Name:   netology.ru
Address: 172.67.75.22

alexey@MacBook-Pro-Alexey ~ %
```



Уровень приложений: DNS

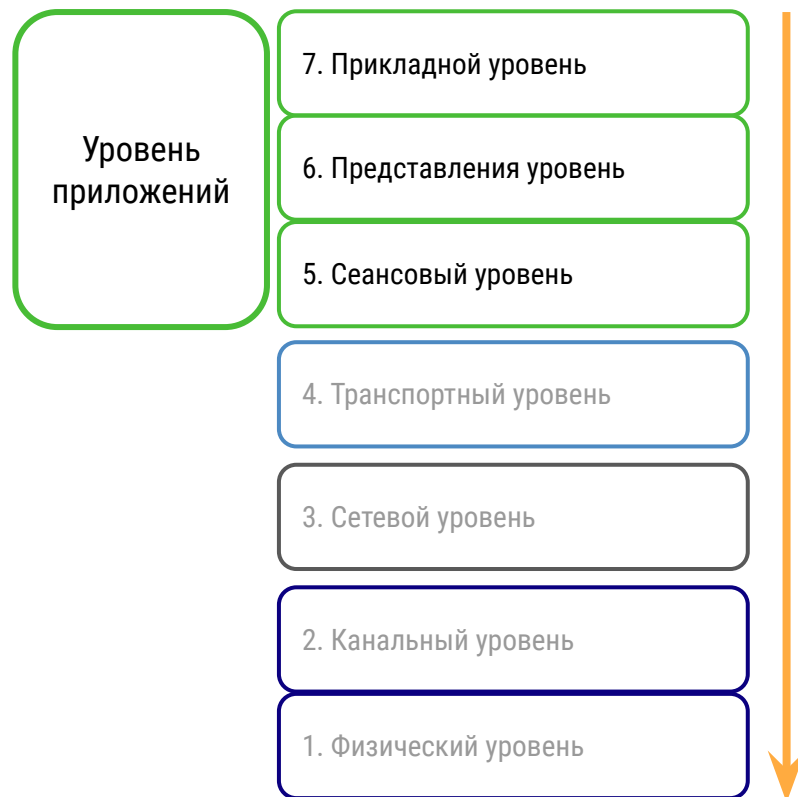
DNS (domain name server) - позволяет узнать адрес назначения по имени домена (сайта)

```
alexey@MacBook-Pro-Alexey ~ % nslookup netology.ru
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:   netology.ru
Address: 104.26.8.143
Name:   netology.ru
Address: 104.26.9.143
Name:   netology.ru
Address: 172.67.75.22

alexey@MacBook-Pro-Alexey ~ %
```

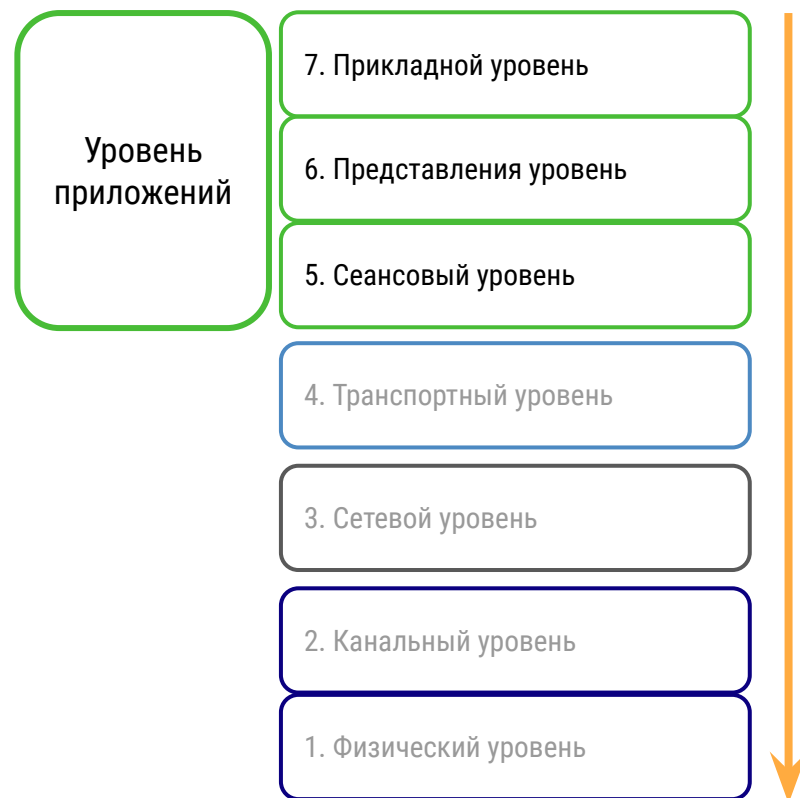
Попробуйте у себя :)



Уровень приложений: DNS

Обмануть DNS?

Например, при локальных тестах...



Уровень приложений: DNS

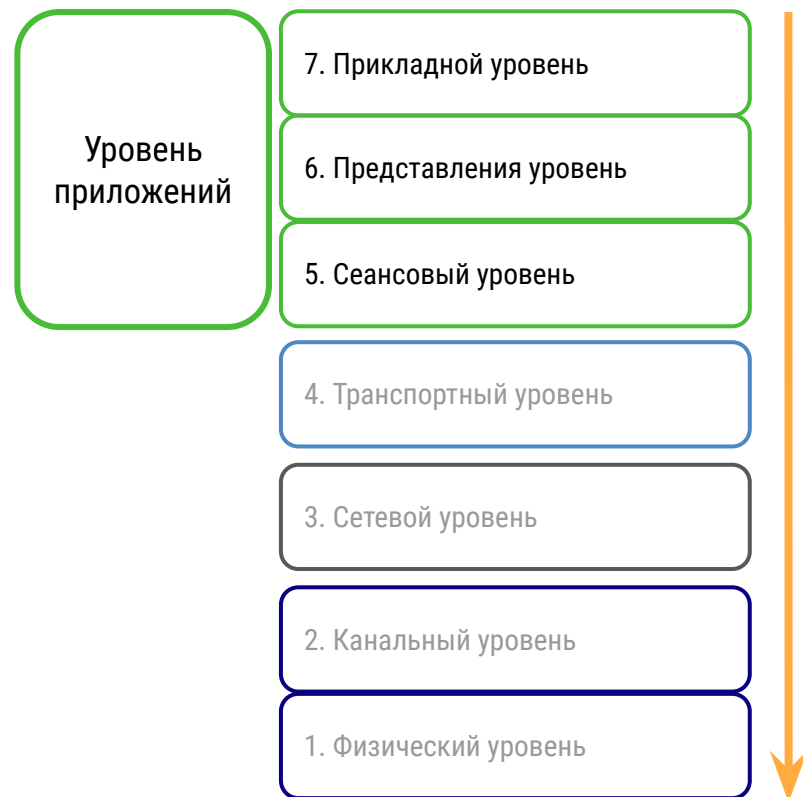
Обмануть DNS?

Например, при локальных тестах...

Windows:

C:\Windows\System32\drivers\etc\hosts

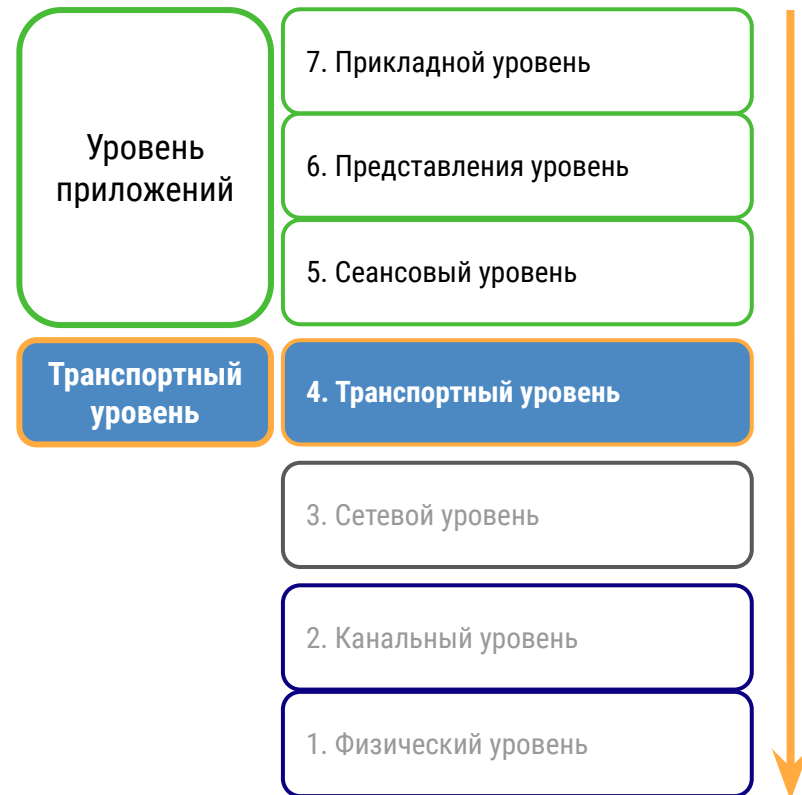
Linux, MacOS: /etc/hosts



[Подробнее >>](#)

Транспортный уровень

Предназначен для обеспечения надежной передачи данных от отправителя к получателю



Транспортный уровень

Предназначен для обеспечения надежной передачи данных от отправителя к получателю

TCP

Когда пакеты данных не должны теряться.
Например, передача файлов



Транспортный уровень

Предназначен для обеспечения надежной передачи данных от отправителя к получателю

TCP

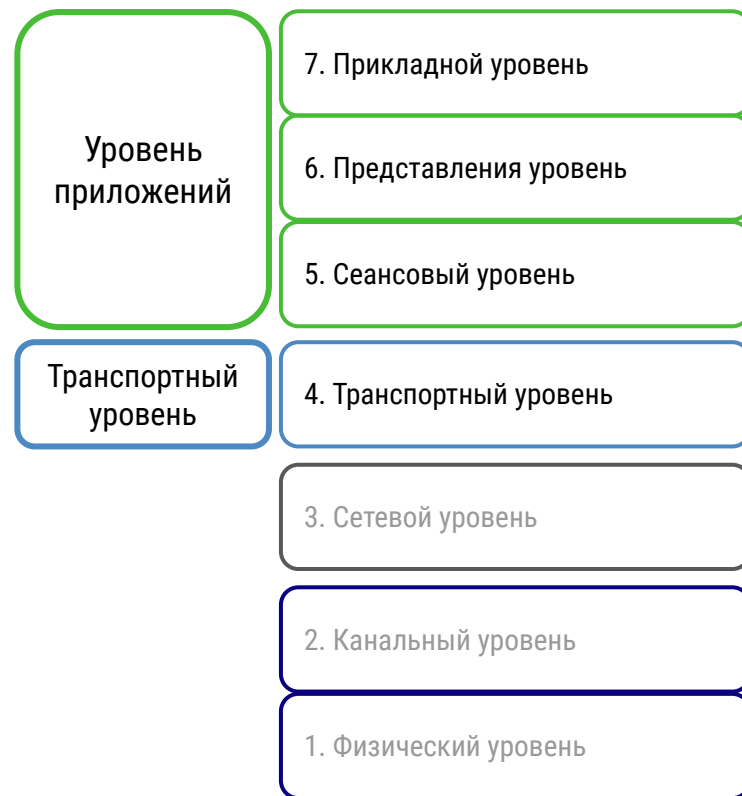
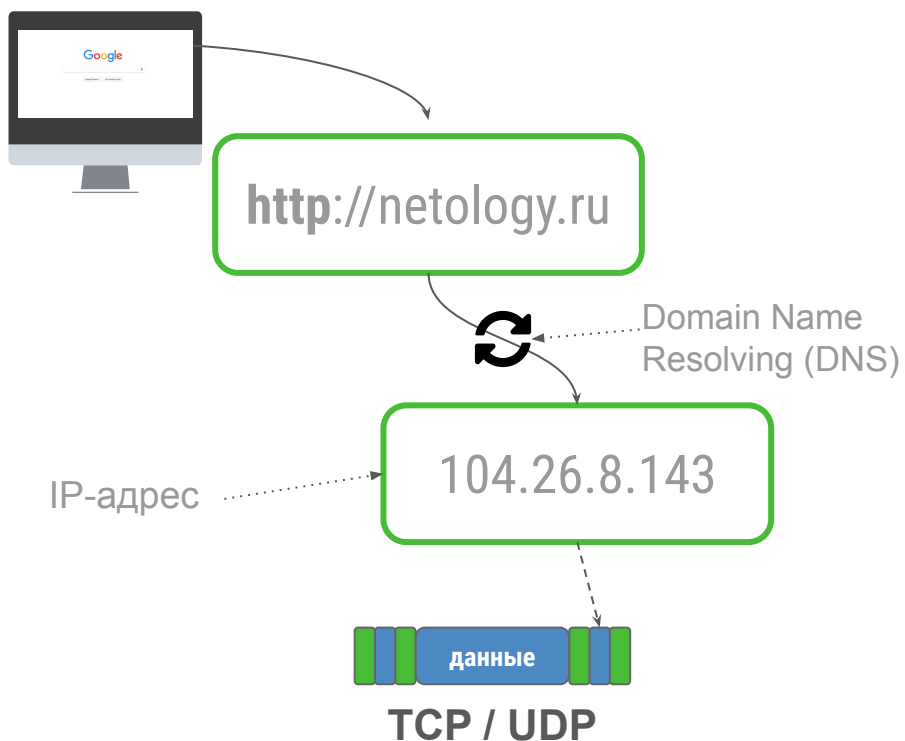
Когда пакеты данных не должны теряться. Например, передача файлов

UDP

Не критично немного потерь. Нет подтверждений получения - работает быстрее. Например, когда вы играете в игру - пускai движения могут иногда дергаться, зато игра будет быстрее.



Транспортный уровень



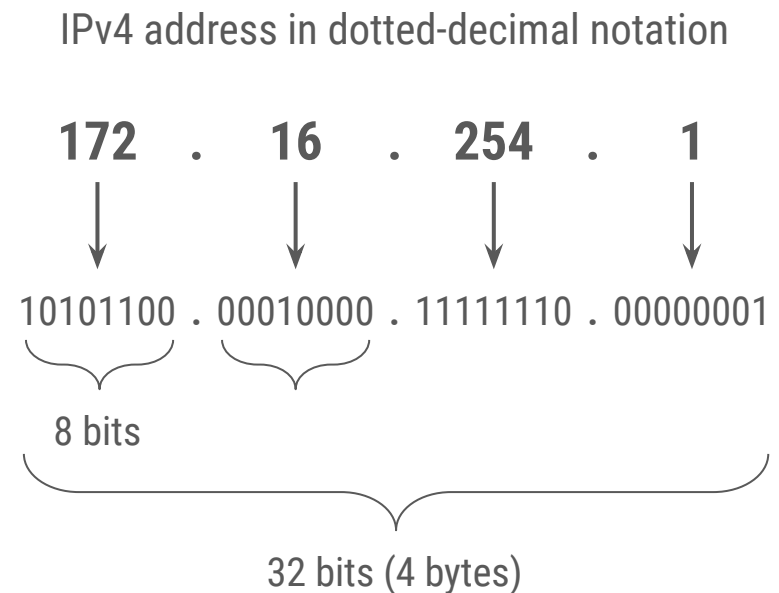
Сетевой уровень

Маршрутизация трафика. Куда и как добраться



Сетевой уровень: IP адрес

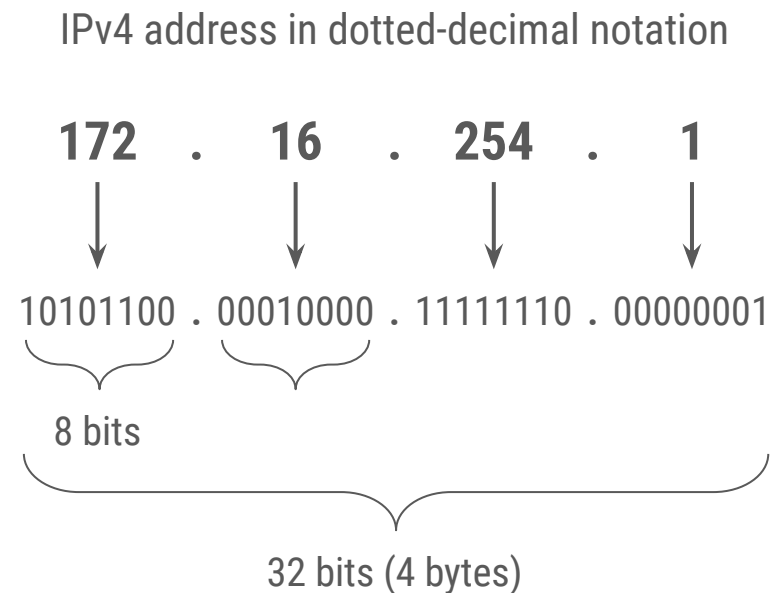
В 4-й версии IP-адрес представляет собой **32-битное** число. Определяется владельцем сети, всего **4 294 967 296**



Сетевой уровень: IP адрес

В 4-й версии IP-адрес представляет собой **32-битное** число. Определяется владельцем сети, всего **4 294 967 296**

- А что делать, когда адреса закончатся?

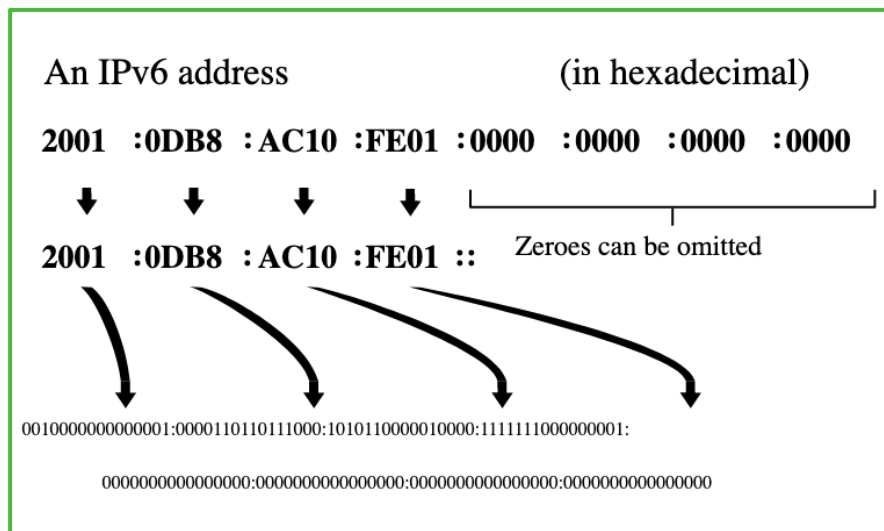


Сетевой уровень: IP адрес

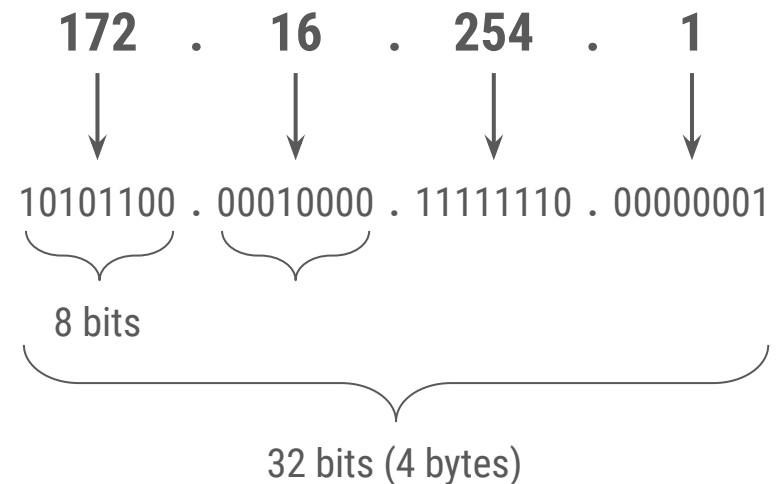
В 4-й версии IP-адрес представляет собой **32-битное** число. Определяется владельцем сети, всего 4 294 967 296

- А что делать, когда адреса закончатся?

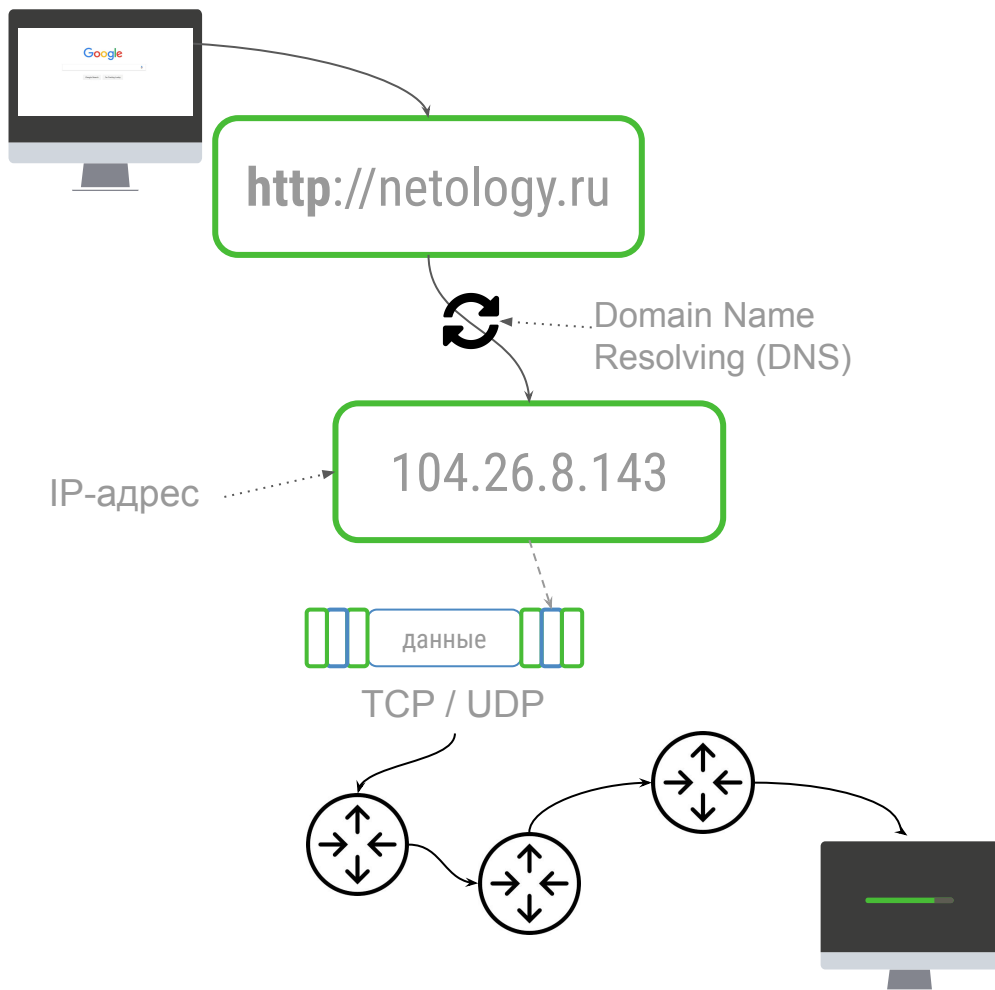
- Просто больше битов!



IPv4 address in dotted-decimal notation



Сетевой уровень



Сетевой уровень

Есть ли сетевой маршрут? Ping

```
alexey@MacBook-Pro-Alexey ~ % ping netology.ru
PING netology.ru (104.26.9.143): 56 data bytes
64 bytes from 104.26.9.143: icmp_seq=0 ttl=57 time=28.872 ms
64 bytes from 104.26.9.143: icmp_seq=1 ttl=57 time=14.577 ms
64 bytes from 104.26.9.143: icmp_seq=2 ttl=57 time=19.179 ms
^C
--- netology.ru ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 14.577/20.876/28.872/5.958 ms
alexey@MacBook-Pro-Alexey ~ %
```



Сетевой уровень

Есть ли сетевой маршрут? Ping

```
alexey@MacBook-Pro-Alexey ~ % ping netology.ru
PING netology.ru (104.26.9.143): 56 data bytes
64 bytes from 104.26.9.143: icmp_seq=0 ttl=57 time=28.872 ms
64 bytes from 104.26.9.143: icmp_seq=1 ttl=57 time=14.577 ms
64 bytes from 104.26.9.143: icmp_seq=2 ttl=57 time=19.179 ms
^C
--- netology.ru ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 14.577/20.876/28.872/5.958 ms
alexey@MacBook-Pro-Alexey ~ %
```

Попробуйте у себя :)



Канальный уровень

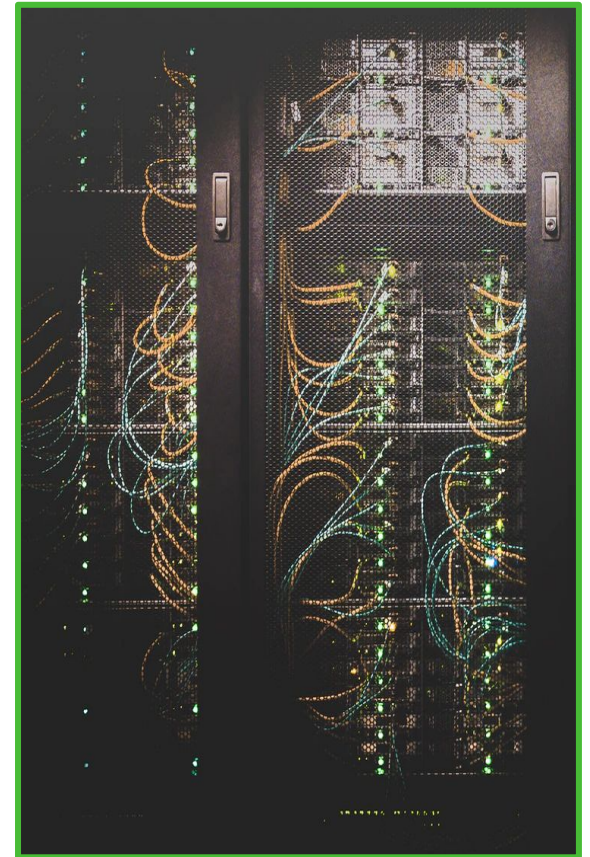
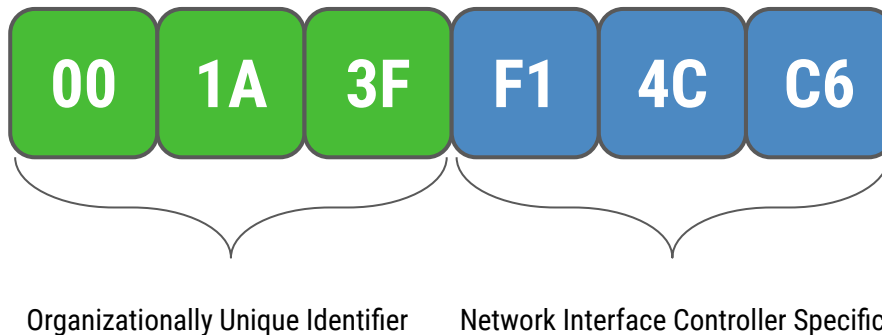
Предназначен для обеспечения взаимодействия сетей на физическом уровне и контроля ошибок, которые могут возникнуть



MAC адрес

Каждое устройство имеет уникальный
MAC адрес, назначается производителе.
Всего адресов 281 000 000 000 000

MAC Media Access Control Address



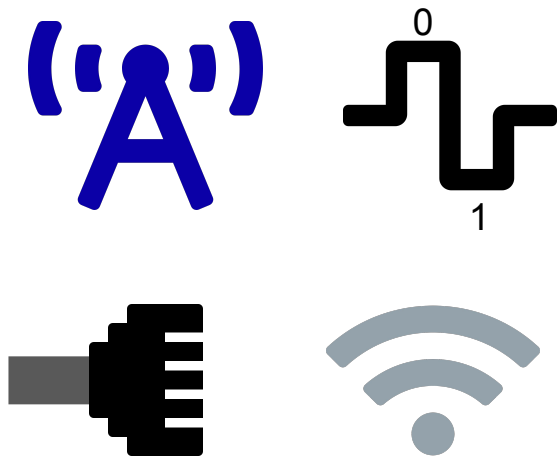
Физический уровень

Нижний уровень модели, который определяет метод передачи данных, представленных в двоичном виде, от одного устройства (компьютера) к другому

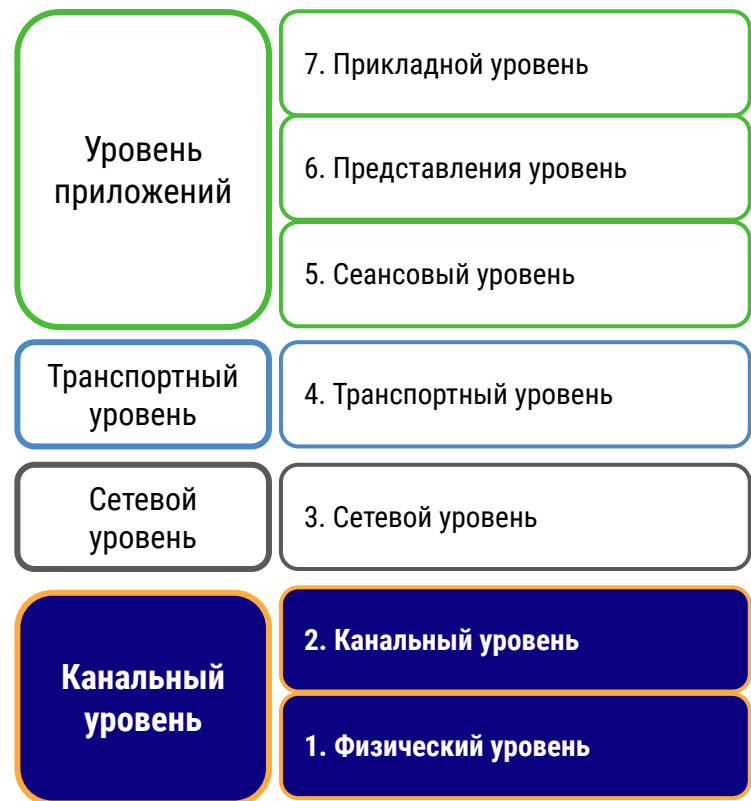
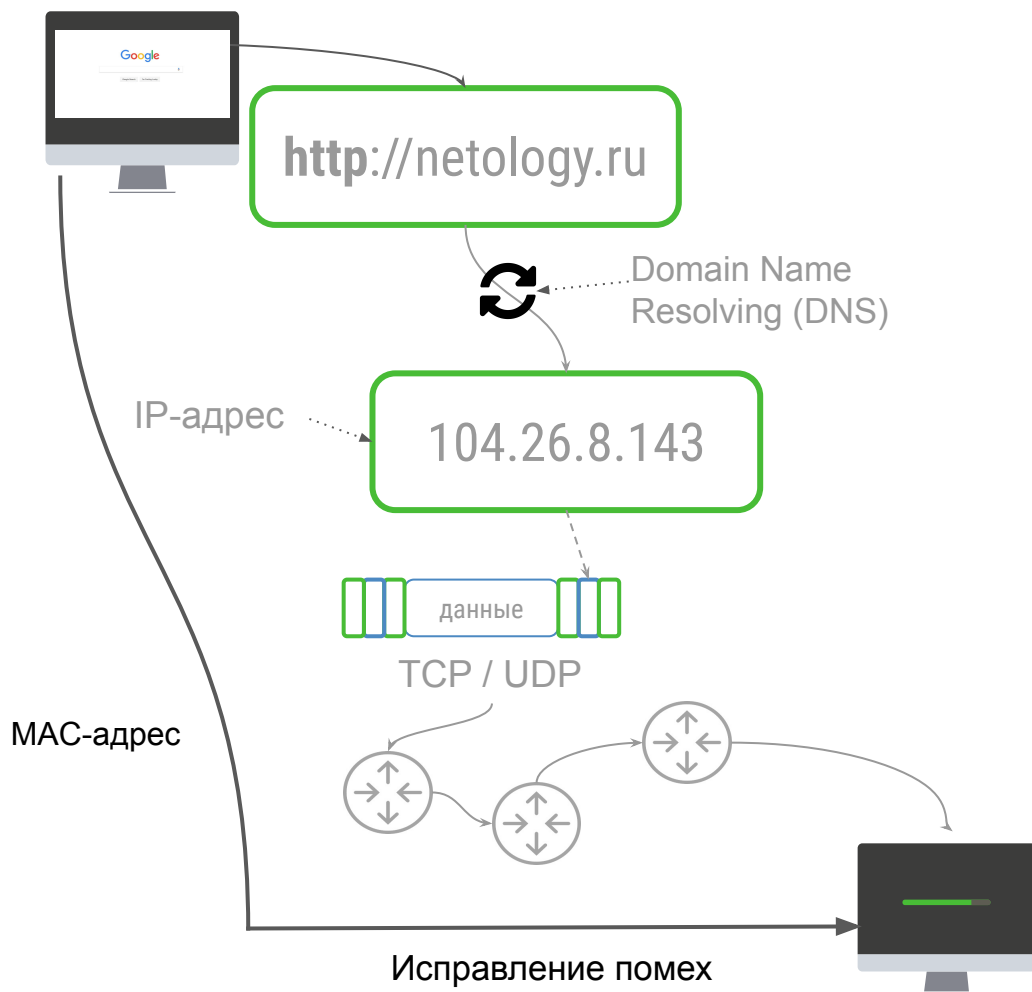


Физический уровень

Нижний уровень модели, который определяет метод передачи данных, представленных в двоичном виде, от одного устройства (компьютера) к другому



Network Access Layer





Модель ТСР/IP

TCP/IP стек

- Проще



OSI

- Теоретическая, концептуальная модель

TCP/IP стек

- Проще
- Большинство реальных протоколов работает по этой модели



OSI

- Теоретическая, концептуальная модель
- Используется для понимания и обсуждения

TCP/IP стек

- Проще
- Большинство реальных протоколов работает по этой модели
- Для разработчика достаточно

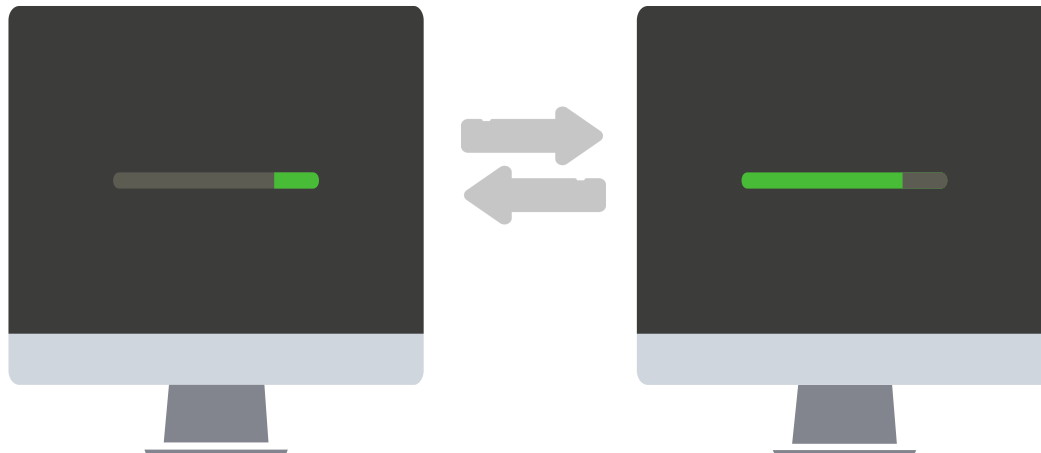


OSI

- Теоретическая, концептуальная модель
- Используется для понимания и обсуждения
- Но на собесе, скорее всего, спросят эту

Как передать данные между компьютерами?

Теперь мы знаем! Ну, хотя бы примерно ;)





Практика: кодим сокеты



Время практики

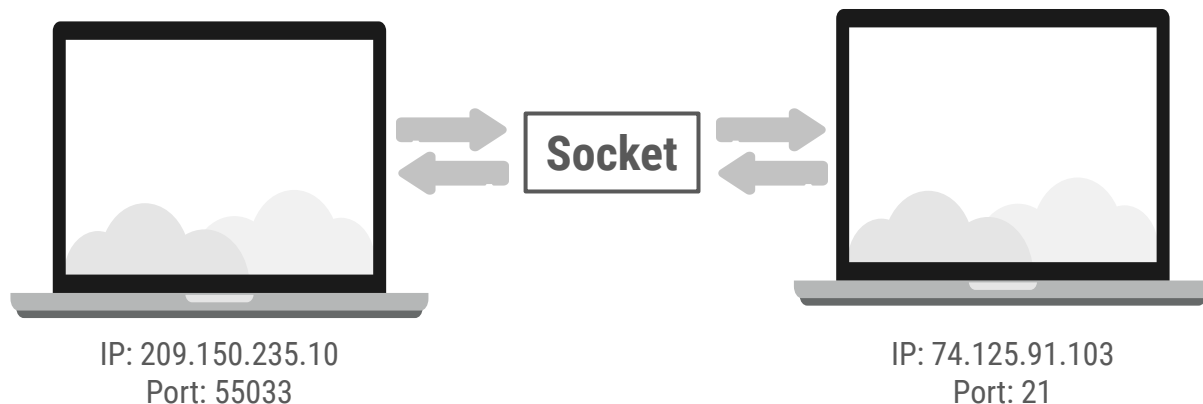
Разработаем программу, которая:

- при сетевом подключении запрашивает строку;
- в ответ отправляет номер порта удаленного хоста.

Socket = IP + Port

Номер порта — целое число в диапазоне от 0 до 65535

В процессе обмена, как правило, используется два сокета — сокет отправителя и сокет получателя. Например, при обращении к серверу на HTTP-порт сокет будет выглядеть так: 194.106.118.30:80, а ответ будет поступать на mmm.nnn.ppp.qq: xxxxx.



Подключимся к сайту

```
String host = "netology.ru";
int port = 80;
try (Socket clientSocket = new Socket(host, port);
    PrintWriter out = new
PrintWriter(clientSocket.getOutputStream(), true);
    BufferedReader in = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream())) {

    out.println("GET / HTTP/1.1\n" +
               "Host: netology.ru\n\n\n");

    String resp = in.readLine();
    System.out.println(resp);
}
```

Определим IP адрес программно

```
String host = "netology.ru";  
InetAddress inetAddress = InetAddress.getByName(host);  
System.out.println(host + ", ip address: " +  
inetAddress.getHostAddress());
```

```
// netology.ru, ip address: 104.26.8.143
```



Итоги



Итоги

- **Модель OSI**
 - *Что помните? (напишите в чат)*



Итоги

- **Модель OSI**
 - Служит для объяснения взаимодействия между системами в виде уровней
 - Содержит 7 уровней



Итоги

- **Модель OSI**

- Служит для объяснения взаимодействия между системами в виде уровней
- Содержит 7 уровней

- **Модель TCP/IP**

- *Что помните? (напишите в чат)*



Итоги

- **Модель OSI**

- Служит для объяснения взаимодействия между системами в виде уровней
- Содержит 7 уровней

- **Модель TCP/IP**

- Протоколы работают по этой модели
- Содержит 4 уровня



Итоги

- **Модель OSI**

- Служит для объяснения взаимодействия между системами в виде уровней
- Содержит 7 уровней

- **Модель TCP/IP**

- Протоколы работают по этой модели
- Содержит 4 уровня

- **Протоколы**

- *Какие помните? (напишите в чат)*



Итоги

- **Модель OSI**

- Служит для объяснения взаимодействия между системами в виде уровней
- Содержит 7 уровней

- **Модель TCP/IP**

- Протоколы работают по этой модели
- Содержит 4 уровня

- **Протоколы**

- TCP
- UDP
- IP
- DNS



Итоги

- **Модель OSI**

- Служит для объяснения взаимодействия между системами в виде уровней
- Содержит 7 уровней

- **Модель TCP/IP**

- Протоколы работают по этой модели
- Содержит 4 уровня

- **Протоколы**

- TCP - пакеты дойдут
- UDP - пакеты могут потеряться
- IP - маршрутизация на сетевом уровне
- DNS - резолвинг доменных имен в IP-адреса



Итоги

- **Модель OSI**

- Служит для объяснения взаимодействия между системами в виде уровней
- Содержит 7 уровней

- **Модель TCP/IP**

- Протоколы работают по этой модели
- Содержит 4 уровня

- **Протоколы**

- TCP - пакеты дойдут
- UDP - пакеты могут потеряться
- IP - маршрутизация на сетевом уровне
- DNS - резолвинг доменных имен в IP-адреса

- **Программирование сокетов**



Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задавайте **в чате** мессенджера Slack.
- Задачи можно сдавать **по частям**.
- Зачёт по домашней работе проставляется после того, как **приняты все задачи**.

Спасибо!

Good

Job

⌘ нетология

**Задавайте вопросы и
пишите отзыв о лекции!**

Григорий Вахмистров