

Сеть и сетевые протоколы: Модель OSI/ISO. Обзор сетевых протоколов



Андрей
Вахутинский



Андрей Вахутинский

Заместитель начальника IT-отдела
АО “ИНТЕКО”



[Андрей Вахутинский](#)

Модуль «Сеть и сетевые протоколы»

Цель модуля:

1. дать навыки работы с сетевым стеком в Linux;
2. объяснить самые распространённые практики;
3. научить настраивать самые распространённые сетевые сервисы.

Структура модуля:

1. Модель OSI/ISO. Обзор сетевых протоколов
2. L2-сеть
3. L3-сеть
4. ...

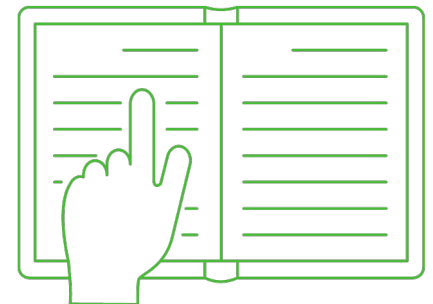
Модуль «Сеть и сетевые протоколы»

4. NAT
5. VPN
6. Firewall
7. Высокоуровневые сетевые протоколы
8. Траблшутинг
9. DHCP, PXE
10. DNS
11. HTTP/HTTPS
12. SMTP/POP3/IMAP.
13. IPv6

Предисловие

Эта лекция содержит **основные теоретические понятия**, которые вам понадобятся, чтобы понимать что происходит на практике.

Зная теорию, вы можете настраивать *любое* сетевое и серверное оборудование (в плане сети) в рамках этой теории, т.к. будут меняться только инструменты, а **принципы остаются неизменными**.



План занятия

1. Основные понятия
2. Сетевая модель
3. Модель OSI
4. Модель TCP/IP
5. Итоги
6. Домашнее задание



Основные понятия

Основные понятия

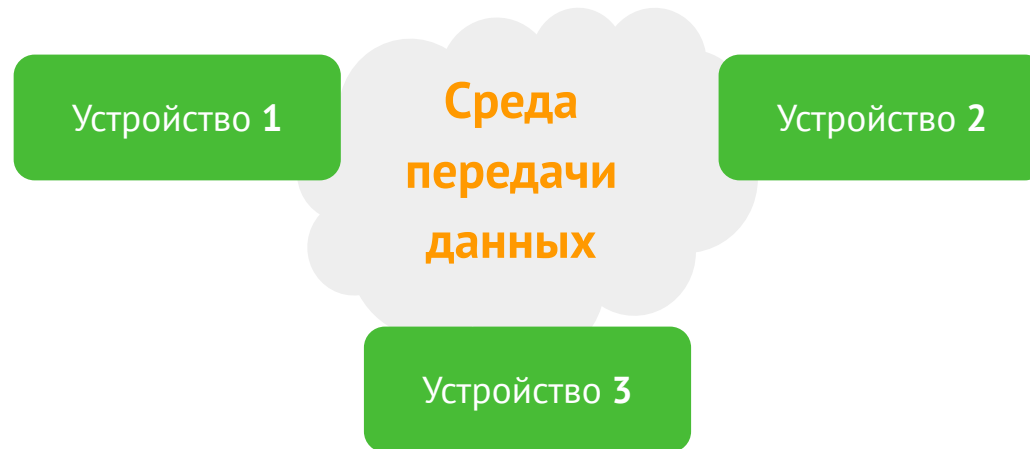
- сеть;
- протоколы передачи данных;
- NAT (Network Address Translations);
- VPN (Virtual Private Network);
- Firewall;
- DNS (Domain Name System);
- DHCP (Dynamic Host Configuration Protocol);
- HTTP/HTTPS;
- SMTP/POP3/IMAP.



Сеть

Что такое сеть?

Сеть – это два и более устройств, способных взаимодействовать друг с другом через использование среды передачи данных (кабель, оптический канал и т.д.).



Необходимо объединить в связанную систему, способную **обмениваться информацией** и **предоставлять сервисы** (например, печать, вычисление, хранение данных).

Прямая аналогия

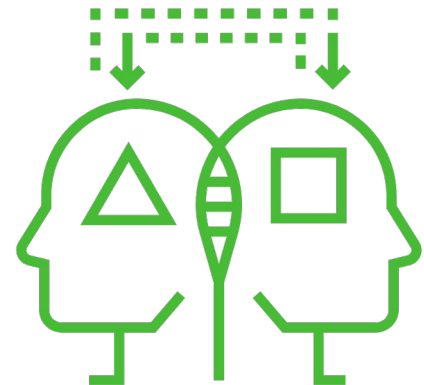
Общение между людьми (голос, зрение) хорошо работает при небольшом количестве участников, но возникают проблемы при большом количестве участников:

- требуется усиливать звук (чтобы все слышали);
- не перебивать друг друга;
- указывать, к кому вы обращаетесь;
- использовать один и тот же язык (либо использовать переводчик);
- даже преобразовывать формат (например, акустические колебания при синхронном переводе преобразуются сначала в электрические, а затем с помощью динамика снова в акустические);
- если хотим обеспечить конфиденциальность, то должны говорить либо очень тихо, либо на таком языке, который никто не поймёт.

Прямая аналогия

Аналогия с человеческим общением – простая и понятная.

Если вы будете использовать её в процессе изучения сетей, то вам будет очень просто понять принципы, на основании которых всё устроено.



Прямая аналогия с почтой

Здесь уместнее провести аналогию с почтой.

The image shows a blank Russian postal envelope template. The word "ПОЧТА" (POST) is printed vertically along the left edge. The template includes the following fields:

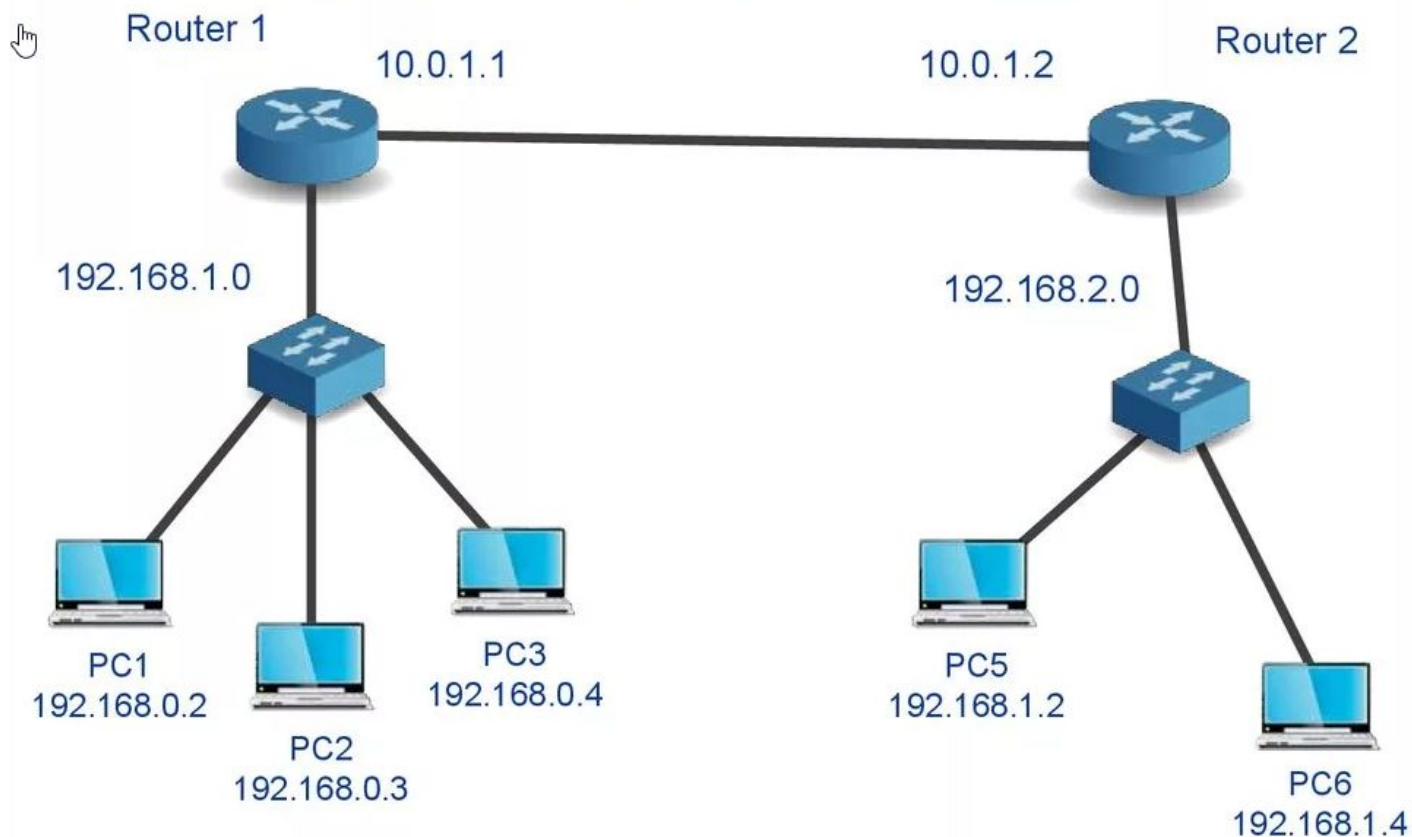
- from**
 - От кого _____
 - Откуда _____
 - _____
 - _____ (почтовый индекс / zip code)
- Вес** _____
- Сумма за вес** _____
- платы за объем, ценность** _____
- _____ (подпись)


There is a large rectangular area for a stamp or photograph in the upper right. The **to** section includes:

- Кому** _____
- Куда** _____
- _____
- _____ (почтовый индекс / zip code)
- _____

Источник изображения - festima.ru

Адреса отправителя и получателя





Протоколы передачи данных

Протоколы передачи данных

Чтобы такое разнообразие «собеседников» понимало друг друга, нужно установить правила общения. Это как с людьми – нужно говорить на одном языке:

- **начало общения** (выбор собеседника, приветствие);
- **завершение общения** (прощание с собеседником);
- **как и когда говорим** (какие фразы и интонации используем, как долго и когда ждём ответа на заданный вопрос);
- **что делаем если что-то пошло не так** (просьба повторить, объяснить другими словами).

Протокол

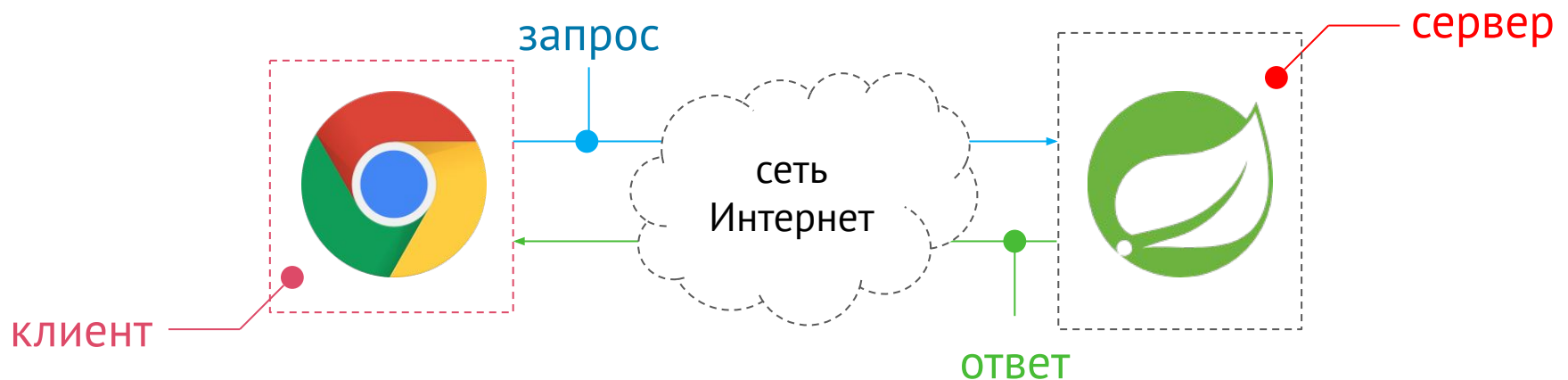
Протокол – это набор соглашений (правил), который определяет обмен данными между различными устройствами.

Протокол определяет:

- **установку соединения** (приветствие);
- **характеристики передачи данных** (скорость обмена, сколько времени ждать ответа, и т.п.);
- **формат данных** (сколько данных передаем за один раз, в каком виде);
- **обработку ошибок** (что делаем, если произошла ошибка);
- **заккрытие соединения** (прощание).

Пример: загрузка веб-сайта

Когда вы вбиваете адрес сайта в строку браузера, то браузер и сервер (программа, работающая на удалённом компьютере) используют общий протокол для обмена информацией:



Пример: загрузка веб-сайта

The screenshot displays a web browser window with the URL `kremlin.ru`. The page content includes several news items, such as "Встреча с министрами иностранных дел стран ШОС" and "Совещание с членами Правительства". The browser's developer tools are open, showing the Network tab. The list of requests includes `kremlin.ru`, `screen.css?07d2237a92`, `xOAtf96UA9mPymtwbvZhRo7qRu9xnJY.jpg`, `app.js?59da6a8407`, `5VFHGCAYrpsAso70EDsgAARApUxy6RLM.jpg`, `pJPWasknkkkSeAzTkhKQJc4TJBa7U0Wh.JPG`, `JB8qyikHebcA7dmdrPyiTyIudBkz6c9Y.jpg`, `print.css`, `smalllogo.svg`, `search.svg`, `select.svg`, `logo_slider.svg`, `circle.svg`, `zoom.svg`, `arrow_up.svg`, `arrow_right.svg`, `footersearch.svg`, `smi.svg`, `special.svg`, `logo.svg`, `all_fonts.css`, `portal_points.svg`, `35d24b68-5d7d-47f1-93c6-2d9f361b3624.woff`, `7b0d9548-bfac-41e0-bba9-6796e1b276f4.woff`, `de8c4c4f-417d-478c-8f88-6422f09187f8.woff`, `b2781bbc-3cc3-47df-be99-5da4e7d6f1e6.woff`, and `31c02786-53fc-47ec-ab0c-d166c6e21711.woff`. The selected request is `kremlin.ru`, and its details are shown in the right pane. The General tab shows the Request URL as `http://kremlin.ru/`, Request Method as `GET`, Status Code as `200 OK`, Remote Address as `95.173.136.71:80`, and Referrer Policy as `no-referrer-when-downgrade`. The Response Headers tab shows `HTTP/1.1 200 OK`, `Server: nginx`, `Date: Wed, 09 Sep 2020 20:05:48 GMT`, `Content-Type: text/html; charset=UTF-8`, `Transfer-Encoding: chunked`, `Connection: keep-alive`, `Keep-Alive: timeout=10`, `Vary: Accept-Encoding`, `X-UA-Compatible: IE=edge`, `Set-Cookie: sid=X62IR19ZNZwy7rA50DRZAg==; path=`, and `Content-Encoding: gzip`. The Request Headers tab shows `GET / HTTP/1.1`, `Host: kremlin.ru`, `Connection: keep-alive`, `Pragma: no-cache`, `Cache-Control: no-cache`, `Upgrade-Insecure-Requests: 1`, and `User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)`.

«ПОД КАПОТОМ»

Протоколы

На каждый протокол должна быть **спецификация** (описание правил).

В зависимости от того, про какой протокол идёт речь, эти спецификации могут выпускаться различными организациями.

Ключевые для нас:

- [IEEE](#) (Institute of Electrical and Electronics Engineers);
- [IETF](#) (Internet Engineering Task Force).

RFC

Например, протоколы сети Интернет описываются в документах, которые называются RFC, выпускаемые IETF.

Request for Comments (дословно: запрос комментария, тема для обсуждения) — документ содержащий технические спецификации и стандарты, используемые в работе сети Интернет.

Где посмотреть:

- tools.ietf.org;
- rfc-editor.org.

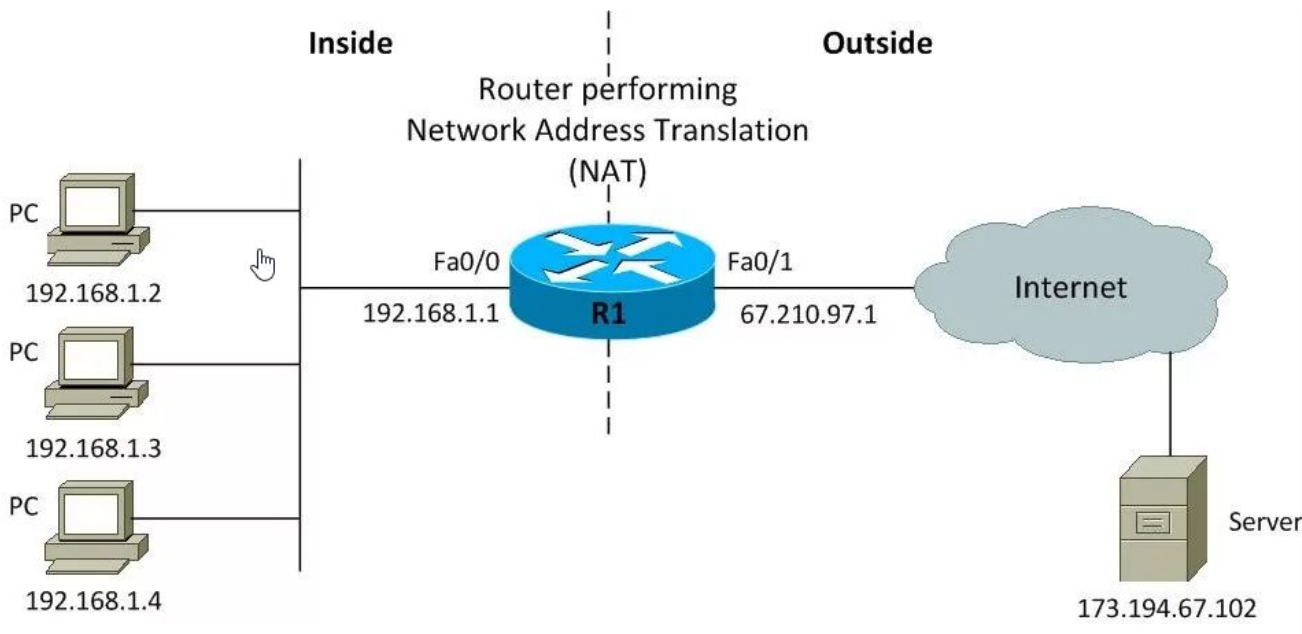


NAT

NAT

NAT (Network Address Translation) – используется для подмены адреса отправителя или адреса получателя.

Пример: локальная сеть – внутри «серые» адреса, выходят через 1 адрес.



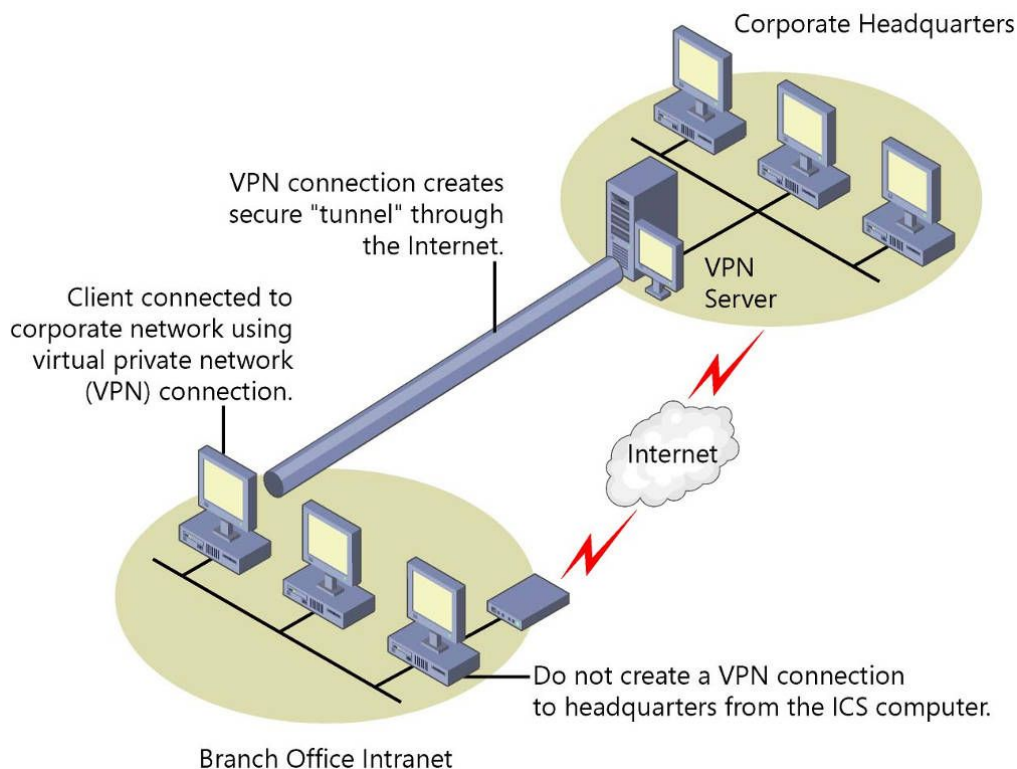


VPN

VPN

VPN (Virtual Private Network) – используется для безопасного соединения 2-х точек через публичные сети.

Пример: соединение 2-х точек в интернете через VPN будто прямым проводом.



Источник изображения -
orbitel.ru



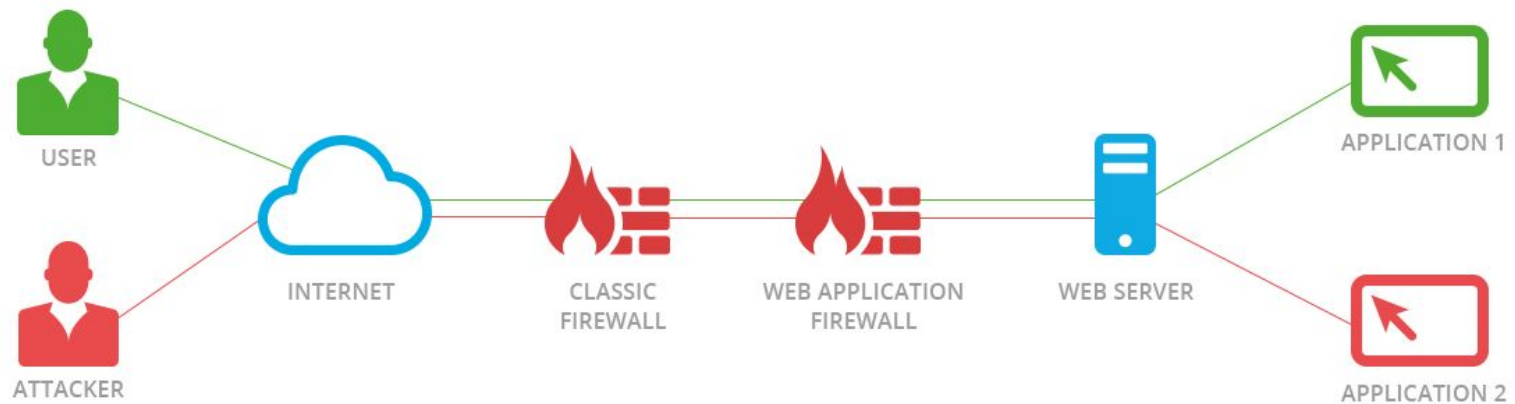
Firewall

Firewall

Firewall (межсетевой экран) – программный или программно-аппаратный элемент компьютерной сети, осуществляющий контроль и фильтрацию проходящего через него трафика.

➡ Пропустить легитимный трафик, не пропустить нелегитимный.

Самое сложное – классифицировать трафик (легитимный/не легитимный).



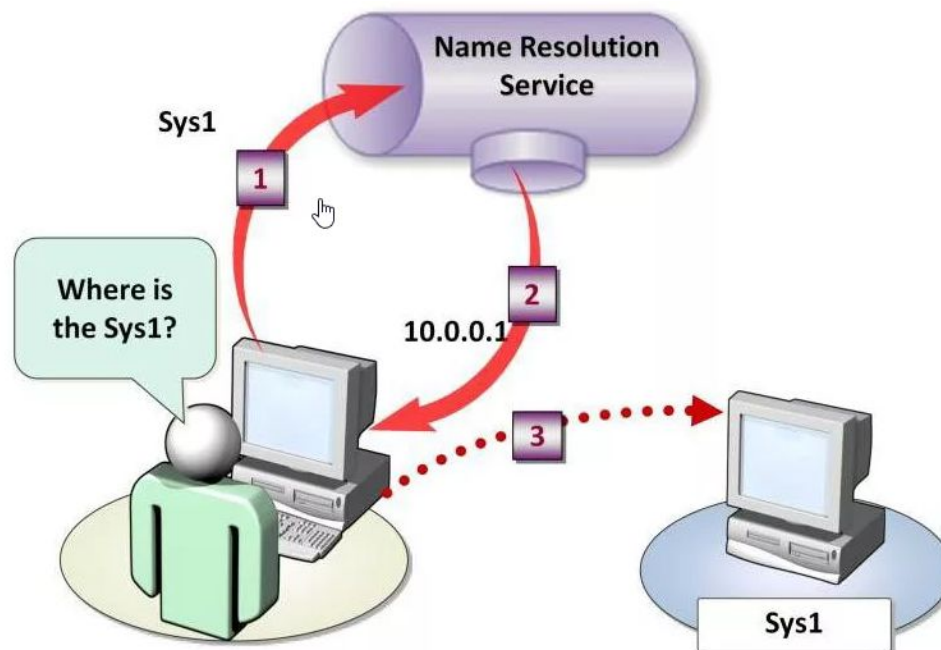


DNS

DNS

DNS (Domain Name System) – система (сервис), предназначенная для получения информации о доменах.

Чаще всего используется для получения IP-адреса по имени хоста (компьютера или устройства), получения информации почтовых серверах.



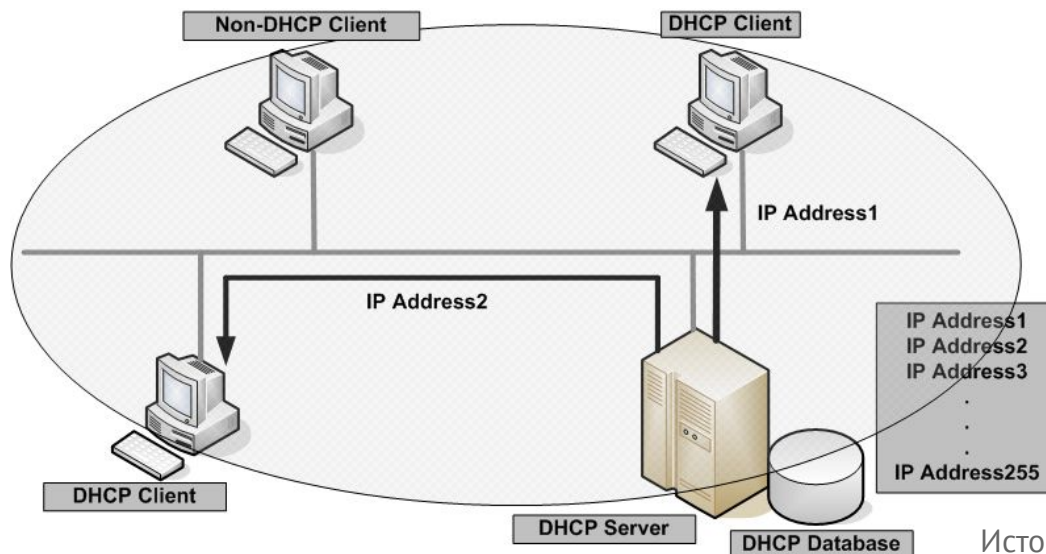


DHCP

DHCP

DHCP (Dynamic Host Configuration Protocol) – протокол, позволяющий сетевым устройствам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети. Данный протокол работает по модели «клиент-сервер»

Пример: WiFi гостевой или домашний, домашний роутер.



Источник изображения - blogspot.com



HTTP/HTTPS

HTTP/HTTPS

HTTP/HTTPS (HyperText Transfer Protocol) – протокол передачи гипертекста (документов, которые могут содержать ссылки, позволяющие организовать переход к другим документам).

Основой HTTP является технология «клиент-сервер», то есть предполагается существование:

- **Потребителей** (клиентов), которые инициируют соединение и посылают запрос;
- **Поставщиков** (серверов), которые ожидают соединения для получения запроса, производят необходимые действия и возвращают сообщение с результатом.

HTTP/HTTPS

HTTP в настоящее время повсеместно используется во Всемирной паутине для получения информации с веб-сайтов.

HTTP**S** – HyperText Transfer Protocol **Secure**.





SMTP/POP3/IMAP

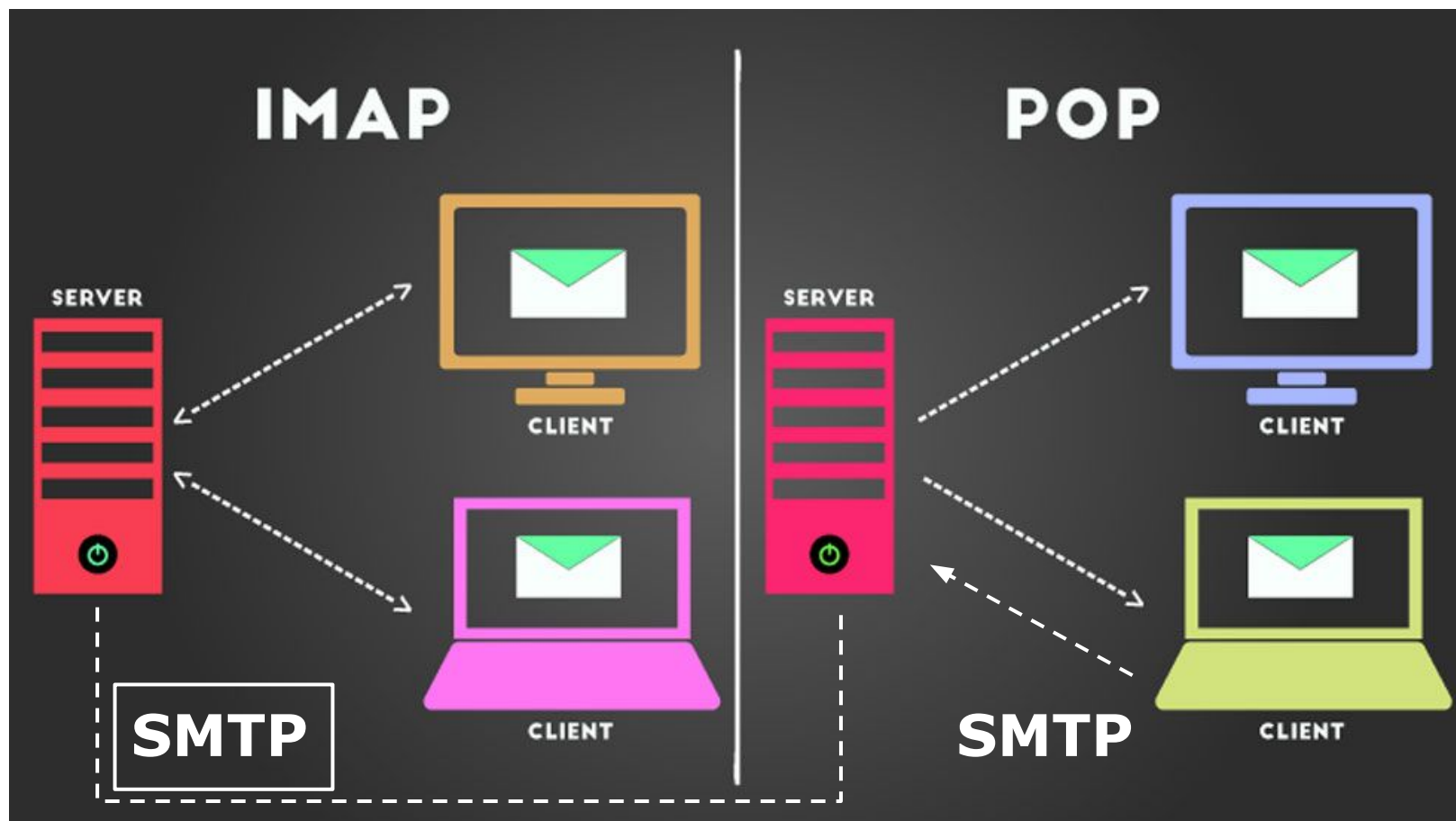
SMTP/POP3/IMAP

SMTP (Simple Mail Transfer Protocol) — простой протокол передачи почты. Широко используемый сетевой протокол, предназначенный для передачи электронной почты в сетях TCP/IP.

POP3 (Post Office Protocol Version 3) — протокол почтового отделения, версия 3. Стандартный интернет-протокол, используемый клиентами электронной почты для получения почты с удалённого сервера по TCP-соединению.

IMAP (Internet Message Access Protocol) — протокол доступа к электронной почте. Задача аналогична POP3, реализация другая. IMAP работает только с сообщениями и не требует каких-либо пакетов со специальными заголовками.

SMTP/POP3/IMAP





Сетевая модель

Сетевая модель

При сетевом взаимодействии удобно строить модель, позволяющую описывать происходящие процессы.

Модель позволяет использовать общие понятия и говорить на одном языке как с производителями оборудования, так и с потребителями.



Сетевая модель

Сетевая модель — описание принципов совместной работы сетевых протоколов.

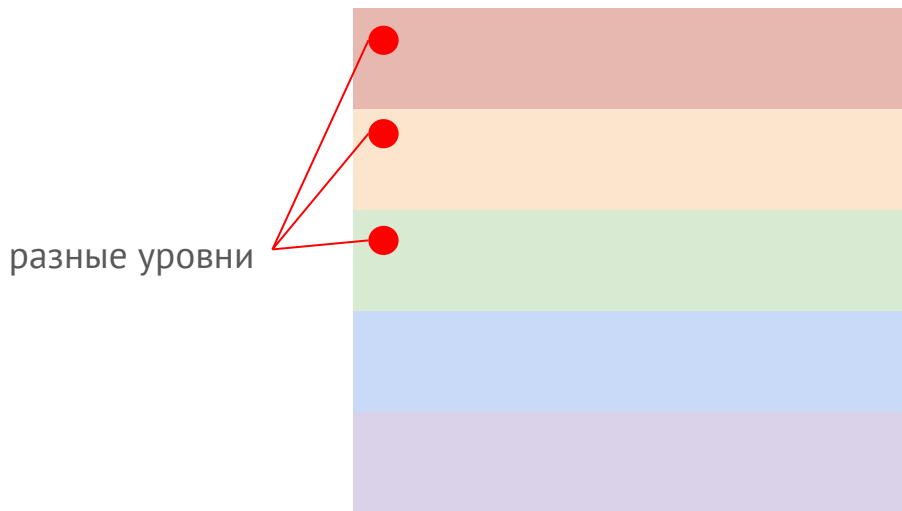
Для удобства разработки модель делится на **уровни**, так, чтобы каждый уровень выполнял какую-то одну задачу.

Инкапсуляция – метод построения модели, когда протоколы вышестоящего уровня используют протоколы нижестоящего уровня.

Модели могут быть:

практическими (использующимися в сетях – TCP/IP) и
теоретическими (показывающими принципы реализации – OSI).

Стек протоколов



Каждый нижележащий уровень предоставляет сервисы вышележащему.

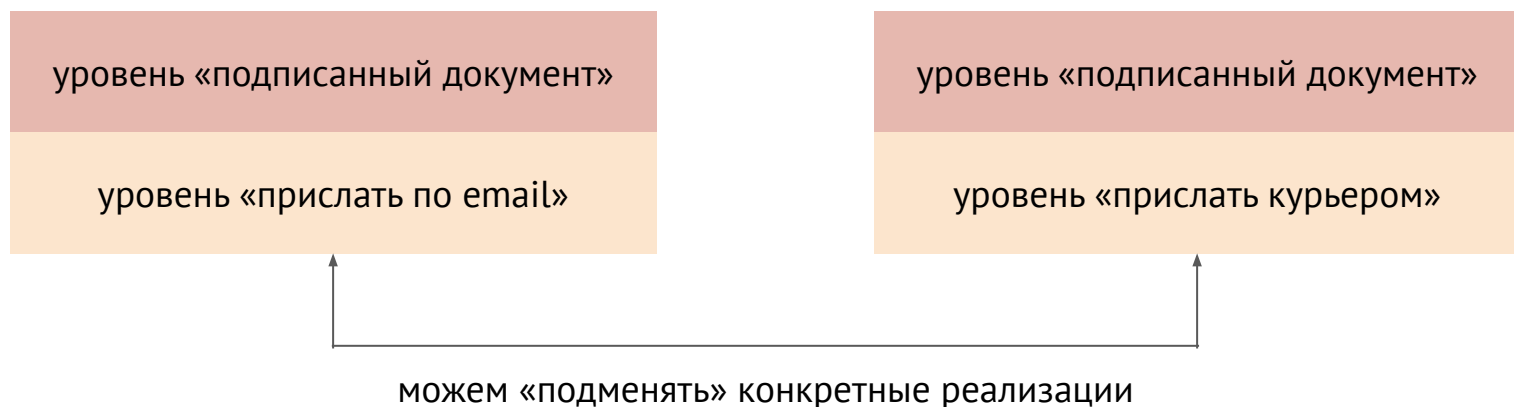
Взаимодействие осуществляется только между двумя «примыкающими» друг к другу уровнями.

Это позволяет заменять реализации конкретных уровней не заменяя всего стека.

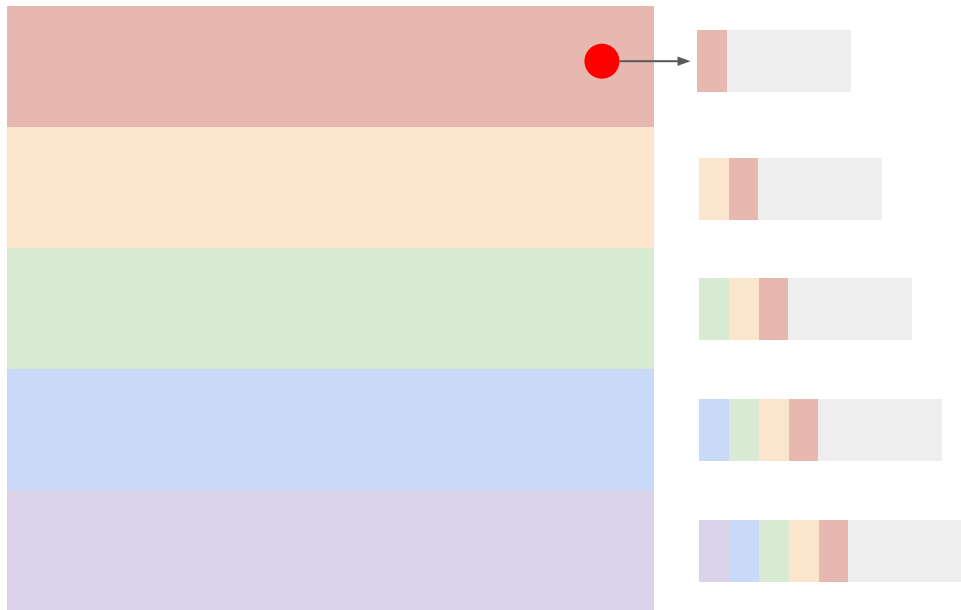
Аналогия

В юридических отношениях две стороны обмениваются документами. Т.е. информацией для них является подписанный документ (самый верхний уровень).

Но документ нужно доставить: для этого мы можем либо послать по почте документ с ЭП, либо прислать его курьером:



Инкапсуляция



Данные, проходя через уровни «заворачиваются» в формат, присущий конкретному уровню (добавляются метаданные: заголовки, при необходимости, сообщение делится на несколько частей).

Например, в Email мы пишем электронный адрес получателя и тему сообщения, а при передаче курьером: имя, физический адрес и телефон.

Примеры сетевых моделей

Ключевые для нас:

- **модель OSI** (Open Systems Interconnection, взаимосвязь открытых систем) — теоретическая сетевая модель, описанная в различных стандартах и используемая как пример для обучения;
- **модель DOD** (модель TCP/IP) — практически используемая сетевая модель, принятая для работы в Интернете.



Модель OSI

Модель OSI

Модель OSI — теоретическая сетевая модель, на практике не используется.

Уровни модели OSI:

7. Прикладной (**Application**)

6. Представления (**Presentation**)

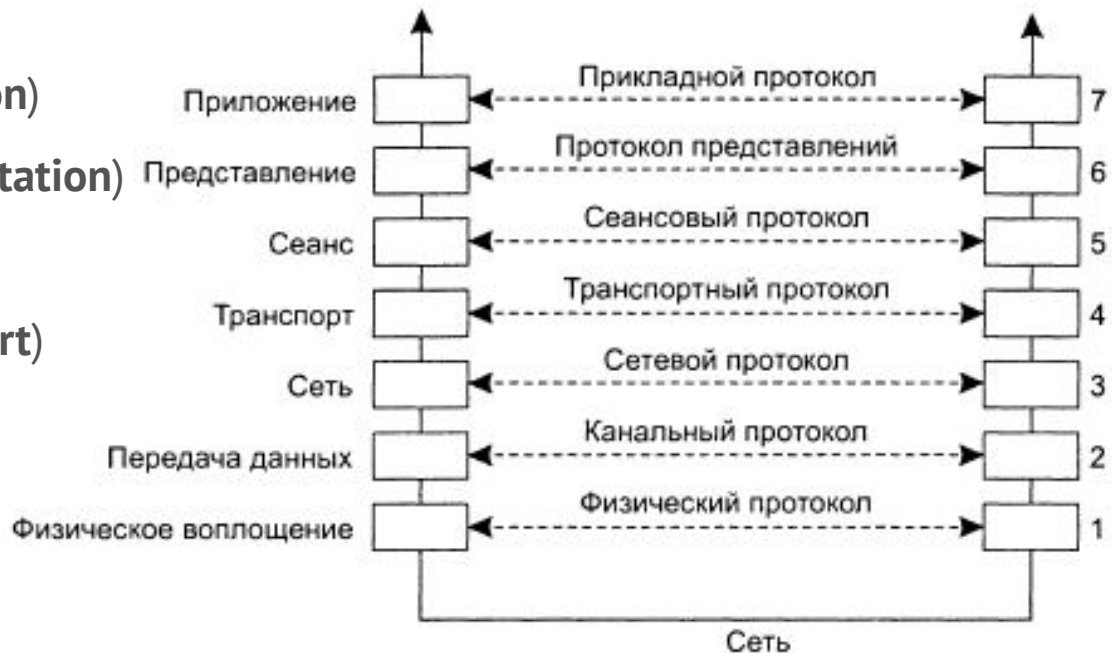
5. Сеансовый (**Session**)

4. Транспортный (**Transport**)

3. Сетевой (**Network**)

2. Канальный (**Data link**)

1. Физический (**Physical**)



Модель OSI

Ключевая задача этой модели – помогать описывать происходящие в сети процессы.



Модель OSI: физический уровень

Физический уровень (physical layer) — определяет способы передачи бит информации через физические среды линий связи (оптический кабель, витая пара).

Основные решаемые проблемы: синхронизация источника и приёмника, избавление от помех, поддержание скорости передачи данных.

Единица данных: бит, бод (изменение среды).

Модель OSI: физический уровень

Пример оборудования:

- витая пара UTP Cat.5 (5e);
- хаб (сетевой концентратор);
- медиаконвертер (преобразователи оптика – медь, Ethernet – RS-485).

Модель OSI: канальный уровень

Канальный уровень (Data Link layer) — определяет способы передачи данных между устройствами, находящимися в одном сегменте сети.

Основные решаемые проблемы: обнаружение ошибок физического уровня, одновременная передача данных разными устройствам (доступ к среде), аппаратная адресация.

Единица данных: кадр, фрейм (**frame**).

Пример оборудование / протокол:

- коммутатор (Ethernet);
- сетевая карта.

Модель OSI: сетевой уровень

Сетевой уровень (Network layer) — определяет способы передачи данных между устройствами, находящимися в разных сетях (сегментах сети).

Основные решаемые проблемы: логическая адресация, построение маршрутов между сетями, диагностика сети.

Единица данных: пакет (packet).

Пример оборудование / протокол:

маршрутизатор (IPv4, IPv6, ICMP).

Модель OSI: транспортный уровень

Транспортный уровень (Transport layer) — определяет способы доставки данных (т.е. определяет сам механизм передачи данных).

Тип взаимодействия: точка – точка.

Основные решаемые проблемы: мультиплексирование (может работать с несколькими потоками данных между двумя устройствами), надежная передача данных, регулирование количества передаваемых данных, контроль доставки данных.

Единица данных: **сегмент** (segment), **дейтаграмма** (datagram).

Пример протокола:

- TCP;
- UDP.

Модель OSI: сеансовый уровень

Сеансовый уровень (Session layer) — определяет способы установления и поддержания сеансов связи.

Основные решаемые проблемы: создание/завершение сеанса, синхронизация/восстановление сеанса, определение прав на передачу данных, поддержание сеанса в периоды неактивности приложений.

Единица данных: **нет** (поток данных).

Пример протокола:

- H.245;
- NetBIOS.

Модель OSI: уровень представления

Уровень представления (Presentation layer) — определяет способы преобразования протоколов и кодирование/декодирование данных.

Основные решаемые проблемы: сжатие и распаковка, кодирование и декодирование данных, перенаправление запросов другому сетевому ресурсу.

Единица данных: **нет** (поток данных).

Пример протокола:

- ASCII;
- EBCDIC.

Модель OSI: прикладной уровень

Прикладной уровень (Application layer) — определяет способы взаимодействия сети и пользователя.

Основные решаемые проблемы: доступ к сетевым службам, передача служебной информации, предоставляет информацию об ошибках.

Единица данных: **нет** (поток данных).


Пример протокола:

- HTTP;
- DNS;
- SSH;
- Telnet.

Модель OSI: подведем итоги



* **сообщение (полезные данные)** также называют **“payload”**



Модель TCP/IP

модель DOD

ТСР/IP: история

Модель DOD (Department of Defense, министерство обороны США) — модель сетевого взаимодействия, разработанная Министерством Обороны США.

ARPANET (Advanced Research Projects Agency Network) — компьютерная сеть, созданная Агентством Министерства Обороны США по перспективным исследованиям (DARPA) — 1969 г.

 Прототип сети Интернет.

TCP/IP: сетевая модель

Модель TCP/IP — сетевая модель передачи данных, описывающая способы передачи данных от источника информации к получателю.

В модели выделено **четыре сетевых уровня***, каждый из которых описывается соответствующими протоколами передачи данных.

Название TCP/IP происходит из двух важнейших протоколов семейства — **Transmission Control Protocol (TCP)** и **Internet Protocol (IP)**, которые были первыми разработаны и описаны в данном стеке.

* В материалах Cisco (законодателя мод в мире сетей) сейчас описывается 5 уровней (Data Link и Physical).

ТСР/IP: сетевая модель

Стек протоколов ТСР/IP включает в себя четыре уровня:

- Прикладной уровень (Application Layer);
- Транспортный уровень (Transport Layer);
- Межсетевой уровень (Internet Layer);
- Канальный уровень (Network Access Layer).



Итоги

Итоги

Сегодня мы узнали:

- базовые понятия в сетях;
- что такое сетевая модель;
- 2 основные сетевых модели:
 - модель OSI
 - модель TCP/IP.



Домашнее задание

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**.

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

Андрей Вахутинский