## http://www.escotal.com/Images/Network%20parts/osi.gif Прикладной уровень

Прикладной уровень или уровень приложений(application layer) – это самый верхний уровень модели. Он осуществляет связь пользовательских приложений с сетью. Эти приложения нам всем знакомы: просмотр веб-страниц (HTTP), передача и приём почты (SMTP, POP3), приём и получение файлов (FTP, TFTP), удаленный доступ (Telnet) и т.д.

**Представительский уровень**

Представительский уровень или уровень представления данных (presentation layer) – он преобразует данные в соответствующий формат. На примере понять проще: те картинки (все изображения) которые вы видите на экране, передаются при пересылке файла в виде маленьких порций единиц и ноликов (битов). Так вот, когда Вы отправляете своему другу фотографию по электронной почте, протокол Прикладного уровня SMTP отправляет фотографию на нижний уровень, т.е. на уровень Представления. Где Ваша фотка преобразуется в удобный вид данных для более низких уровней, например в биты (единицы и нолики).

Именно таким же образом, когда Ваш друг начнет получать Ваше фото, ему оно будет поступать в виде все тех же единиц и нулей, и именно уровень Представления преобразует биты в полноценное фото, например JPEG.

Вот так и работает этот уровень с протоколами (стандартами) изображений (JPEG, GIF, PNG, TIFF), кодировок (ASCII, EBDIC), музыки и видео (MPEG) и т.д.

**Сеансовый уровень**

Сеансовый уровень или уровень сессий(session layer) – как видно из названия, он организует сеанс связи между компьютерами. Хорошим примером будут служить аудио и видеоконференции, на этом уровне устанавливается, каким кодеком будет кодироваться сигнал, причем этот кодек должен присутствовать на обеих машинах. Еще примером может служить протокол SMPP (Short message peer-to-peer protocol), с помощью него отправляются хорошо известные нам СМСки и USSD запросы. И последний пример: PAP (Password Authentication Protocol) – это старенький протокол для отправки имени пользователя и пароля на сервер без шифрования.

Больше про сеансовый уровень ничего не скажу, иначе углубимся в скучные особенности протоколов. А если они (особенности) Вас интересуют, пишите письма мне или оставляйте сообщение в комментариях с просьбой раскрыть тему более подробно, и новая статья не заставит себя долго ждать ;)

**Транспортный уровень**

Транспортный уровень (transport layer) – этот уровень обеспечивает надёжность передачи данных от отправителя к получателю. На самом деле всё очень просто, например вы общаетесь с помощью веб-камеры со своим другом или преподавателем. Нужна ли здесь надежная доставка каждого бита переданного изображения? Конечно нет, если потеряется несколько битов из потокового видео Вы даже этого не заметите, даже картинка не изменится (м.б. изменится цвет одного пикселя из 900000 пикселей, который промелькнет со скоростью 24 кадра в секунду).

А теперь приведем такой пример: Вам друг пересылает (например, через почту) в архиве важную информацию или программу. Вы скачиваете себе на компьютер этот архив. Вот здесь надёжность нужна 100%, т.к. если пару бит при закачке архива потеряются – Вы не сможете затем его разархивировать, т.е. извлечь необходимые данные. Или представьте себе отправку пароля на сервер, и в пути один бит потерялся – пароль уже потеряет свой вид и значение изменится.

Таким образом, когда мы смотрим видеоролики в интернете, иногда мы видим некоторые артефакты, задержки, шумы и т.п. А когда мы читаем текст с веб-страницы – потеря (или скжение) букв не допустима, и когда скачиваем программы – тоже все проходит без ошибок.

На этом уровне я выделю два протокола: UDP и TCP. UDP протокол (User Datagram Protocol) передает данные без установления соединения, не подтверждает доставку данных и не делает повторы. TCP протокол (Transmission Control Protocol), который перед передачей устанавливает соединение, подтверждает доставку данных, при необходимости делает повтор, гарантирует целостность и правильную последовательность загружаемых данных.

Следовательно, для музыки, видео, видеоконференций и звонков используем UDP (передаем данные без проверки и без задержек), а для текста, программ, паролей, архивов и т.п. – TCP (передача данных с подтверждением о получении, затрачивается больше времени).

**Сетевой уровень**

Сетевой уровень (network layer) – этот уровень определяет путь, по которому данные будут переданы. И, между прочим, это третий уровень Сетевой модели OSI, а ведь существуют такие устройства, которые как раз и называют устройствами третьего уровня – маршрутизаторы.

Все мы слышали об IP-адресе, вот это и осуществляет протокол IP (Internet Protocol). IP-адрес – это логический адрес в сети.

На этом уровне достаточно много протоколов и все эти протоколы мы разберем более подробно позже, в отдельных статьях и на примерах. Сейчас же только перечислю несколько популярных.

Как об IP-адресе все слышали и о команде ping – это работает протокол ICMP.

Те самые маршрутизаторы (с которыми мы и будет работать в дальнейшем) используют протоколы этого уровня для маршрутизации пакетов (RIP, EIGRP, OSPF).

Вся вторая часть курса CCNA (Exploration 2) о маршрутизации.

**Канальный уровень**

Канальный уровень (data link layer) – он нам нужен для взаимодействия сетей на физическом уровне. Наверное, все слышали о MAC-адресе, вот он является физическим адресом. Устройства канального уровня – коммутаторы, концентраторы и т.п.

IEEE (Institute of Electrical and Electronics Engineers - Институт инженеров по электротехнике и электронике) определяет канальный уровень двумя подуровнями: LLC и MAC.

LLC – управление логическим каналом (Logical Link Control), создан для взаимодействия с верхним уровнем.

MAC – управление доступом к передающей среде (Media Access Control), создан для взаимодействия с нижним уровнем.

Объясню на примере: в Вашем компьютере (ноутбуке, коммуникаторе) имеется сетевая карта (или какой-то другой адаптер), так вот для взаимодействия с ней (с картой) существует драйвер. Драйвер – это некоторая **программа** - верхний подуровень канального уровня, через которую как раз и можно связаться с нижними уровнями, а точнее с микропроцессором (**железо**) – нижний подуровень канального уровня.

Типичных представителей на этом уровне много. PPP (Point-to-Point) – это протокол для связи двух компьютеров напрямую. FDDI (Fiber Distributed Data Interface) – стандарт передаёт данные на расстояние до 200 километров. CDP (Cisco Discovery Protocol) – это проприетарный (собственный) протокол принадлежащий компании Cisco Systems, с помощью него можно обнаружить соседние устройства и получить информацию об этих устройствах.

Вся третья часть курса CCNA (Exploration 3) об устройствах второго уровня.

**Физический уровень**

Физический уровень (physical layer) – самый нижний уровень, непосредственно осуществляющий передачу потока данных. Протоколы нам всем хорошо известны: Bluetooth, IRDA (Инфракрасная связь), медные провода (витая пара, телефонная линия), Wi-Fi, и т.д.

Подробности и спецификации ждите в следующих статьях и в курсе CCNA. Вся первая часть курса CCNA (Exploration 1) посвящена модели OSI.

**HTTP** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *HyperText Transfer Protocol* — «протокол передачи [гипертекста](https://ru.wikipedia.org/wiki/%D0%93%D0%B8%D0%BF%D0%B5%D1%80%D1%82%D0%B5%D0%BA%D1%81%D1%82)») — [протокол](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%BE%D0%B9_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB) [прикладного уровня](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D1%8B_%D0%BF%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B3%D0%BE_%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D1%8F) передачи данных (изначально — в виде гипертекстовых документов в формате [HTML](https://ru.wikipedia.org/wiki/HTML), в настоящий момент используется для передачи произвольных данных). Основой HTTP является [технология «клиент-сервер»](https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B8%D0%B5%D0%BD%D1%82-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80)

|  |  |
| --- | --- |
| **Название:** | Hypertext Transfer Protocol |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Прикладной |
| **Семейство:** | [TCP/IP](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B5%D0%BA_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D0%BE%D0%B2_TCP/IP) |
| **Создан в:** | [1992](https://ru.wikipedia.org/wiki/1992) г. |
| **Порт/ID:** | 80/[TCP](https://ru.wikipedia.org/wiki/TCP) |

В настоящее время выделено пять классов кодов состояния.

[**1xx Informational**](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#1xx)**(«Информационный»)**

В этот класс выделены коды, информирующие о процессе передачи. В HTTP/1.0 сообщения с такими кодами должны игнорироваться. В HTTP/1.1 клиент должен быть готов принять этот класс сообщений как обычный ответ, но ничего отправлять серверу не нужно. Сами сообщения от сервера содержат только стартовую строку ответа и, если требуется, несколько специфичных для ответа полей заголовка. [Прокси-серверы](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%BA%D1%81%D0%B8-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) подобные сообщения должны отправлять дальше от сервера к клиенту.

[**2xx Success**](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#2xx)**(«Успех»)**

Сообщения данного класса информируют о случаях успешного принятия и обработки запроса клиента. В зависимости от статуса сервер может ещё передать заголовки и тело сообщения.

[**3xx Redirection**](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#3xx)**(«Перенаправление»)**

Коды класса 3xx сообщают клиенту что для успешного выполнения операции необходимо сделать другой запрос (как правило по другому URI). Из данного класса пять кодов [301](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#301), [302](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#302), [303](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#303), [305](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#305) и [307](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#307) относятся непосредственно к перенаправлениям (редирект). Адрес, по которому клиенту следует произвести запрос, сервер указывает в заголовке Location. При этом допускается использование фрагментов в целевом URI.

[**4xx Client Error**](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#4xx)**(«Ошибка клиента»)**

Класс кодов 4xx предназначен для указания ошибок со стороны клиента. При использовании всех методов, кроме HEAD, сервер должен вернуть в теле сообщения гипертекстовое пояснение для пользователя.

[**5xx Server Error**](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP#5xx)**(«Ошибка сервера»)**

Коды 5xx выделены под случаи неудачного выполнения операции по вине сервера. Для всех ситуаций, кроме использования метода HEAD, сервер должен включать в тело сообщения объяснение, которое клиент отобразит пользователю.

## Структура протокола

Каждое HTTP-сообщение состоит из трёх частей, которые передаются в указанном порядке:

1. Стартовая строка ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Starting line*) — определяет тип сообщения;
2. Заголовки ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Headers*) — характеризуют тело сообщения, параметры передачи и прочие сведения;
3. Тело сообщения ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Message Body*) — непосредственно данные сообщения. Обязательно должно отделяться от заголовков пустой строкой.

Заголовки и тело сообщения могут отсутствовать, но стартовая строка является обязательным элементом, так как указывает на тип запроса/ответа. Исключением является версия 0.9 протокола, у которой сообщение запроса содержит только стартовую строку, а сообщения ответа только тело сообщения.

### Методы**[**[**править**](https://ru.wikipedia.org/w/index.php?title=HTTP&veaction=edit&vesection=8)**|**[**править вики-текст**](https://ru.wikipedia.org/w/index.php?title=HTTP&action=edit&section=8)**]**

Метод HTTP ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *HTTP Method*) — последовательность из любых символов, кроме управляющих и разделителей, указывающая на основную операцию над ресурсом. Обычно метод представляет собой короткое английское слово, записанное заглавными буквами. Обратите внимание, что название метода чувствительно к регистру.

Каждый сервер обязан поддерживать как минимум методы GET и HEAD. Если сервер не распознал указанный клиентом метод, то он должен вернуть статус 501 (Not Implemented). Если серверу метод известен, но он неприменим к конкретному ресурсу, то возвращается сообщение с кодом 405 (Method Not Allowed). В обоих случаях серверу следует включить в сообщение ответа заголовок Allow со списком поддерживаемых методов.

Кроме методов GET и HEAD, часто применяется метод POST.

## Compare GET vs. POST

<http://www.w3schools.com/tags/ref_httpmethods.asp> - !!!

**• GET - запрашивает данные из указанного ресурса**

Некоторые другие заметки о запросах GET:

GET запросы могут кэшироваться

GET запросы остаются в истории браузера

GET запросы могут быть закладкой

не получать запросы не должны использоваться при работе с конфиденциальными данными

GET запросы имеют ограничения длины

Некоторые другие замечания по запросам POST:

**• POST - Отправляет данные, подлежащие обработке для указанного ресурса**

POST запросы никогда не кэшируются

POST запросы не остаются в истории браузера

запросы POST не может быть закладкой

POST запросы не имеют ограничений по длине данных

**HTTPS** (аббр. от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA" \o "Английский язык) *HyperText Transfer Protocol Secure*) — расширение [протокола](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB_%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85" \o "Протокол передачи данных) [HTTP](https://ru.wikipedia.org/wiki/HTTP), поддерживающее [шифрование](https://ru.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5" \o "Шифрование). Данные, передаваемые по протоколу HTTPS, «упаковываются» в криптографический протокол [SSL](https://ru.wikipedia.org/wiki/SSL) или [TLS](https://ru.wikipedia.org/wiki/TLS). В отличие от HTTP, для HTTPS по умолчанию используется [TCP](https://ru.wikipedia.org/wiki/TCP)-[порт](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%80%D1%82_(TCP/IP)) 443.

|  |  |
| --- | --- |
| **Название:** | HyperText Transfer Protocol Secure |
| **Уровень (по [модели OSI](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI" \o "Сетевая модель OSI)):** | Прикладной |
| **Семейство:** | [TCP/IP](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B5%D0%BA_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D0%BE%D0%B2_TCP/IP) |
| **Создан в:** | [2000](https://ru.wikipedia.org/wiki/2000) г. |
| **Порт/ID:** | 443/[TCP](https://ru.wikipedia.org/wiki/TCP) |
| **Назначение протокола:** | Безопасное соединение с сервером |

**FTP** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *File Transfer Protocol* — [протокол передачи](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D1%8B_%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) [файлов](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB)) — стандартный [протокол](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%BE%D0%B9_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB), предназначенный для передачи файлов по TCP-сетям (например, Интернет). Использует 21-й порт. FTP часто используется для загрузки сетевых страниц и других документов с частного устройства разработки на открытые [сервера](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5)) [хостинга](https://ru.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3).

|  |  |
| --- | --- |
| **FTP** | |
| **Название:** | File Transfer Protocol |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Прикладной |
| **Семейство:** | TCP/IP |
| **Создан в:** | 1971 г. |
| **Порт/ID:** | 21/TCP для команд, 20/TCP для данных, 49152-65534/TCP динамически |
| **Назначение протокола:** | Передача файлов |

**SSH** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) ***S****ecure****Sh****ell* — «безопасная оболочка»[[1]](https://ru.wikipedia.org/wiki/SSH#cite_note-1)) — [сетевой протокол прикладного уровня](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D1%8B_%D0%BF%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B3%D0%BE_%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D1%8F), позволяющий производить [удалённое управление](https://ru.wikipedia.org/wiki/%D0%A3%D0%B4%D0%B0%D0%BB%D1%91%D0%BD%D0%BD%D0%BE%D0%B5_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5) [операционной системой](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) и [туннелирование](https://ru.wikipedia.org/wiki/%D0%A2%D1%83%D0%BD%D0%BD%D0%B5%D0%BB%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_(%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D1%8B%D0%B5_%D1%81%D0%B5%D1%82%D0%B8)) [TCP](https://ru.wikipedia.org/wiki/TCP)-соединений (например, для передачи файлов). Схож по функциональности с протоколами [Telnet](https://ru.wikipedia.org/wiki/Telnet) и [rlogin](https://ru.wikipedia.org/wiki/Rlogin), но, в отличие от них, [шифрует](https://ru.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) весь [трафик](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%BE%D0%B9_%D1%82%D1%80%D0%B0%D1%84%D0%B8%D0%BA), включая и передаваемые [пароли](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%80%D0%BE%D0%BB%D1%8C). SSH допускает выбор различных [алгоритмов](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC) шифрования. SSH-[клиенты](https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B8%D0%B5%D0%BD%D1%82_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) и SSH-[серверы](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5)) доступны для большинства сетевых операционных систем.

|  |  |
| --- | --- |
| **SSH** | |
| **Название:** | Secure Shell |
| **Уровень (по [модели OSI](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI" \o "Сетевая модель OSI)):** | [Прикладной](https://ru.wikipedia.org/w/index.php?title=%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B9&action=edit&redlink=1) |
| **Семейство:** | [TCP/IP](https://ru.wikipedia.org/wiki/TCP/IP) |
| **Порт/ID:** | 22/[TCP](https://ru.wikipedia.org/wiki/TCP) |
| **Назначение протокола:** | [Удалённый доступ](https://ru.wikipedia.org/wiki/%D0%A3%D0%B4%D0%B0%D0%BB%D1%91%D0%BD%D0%BD%D1%8B%D0%B9_%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF) |

**SMTP** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Simple Mail Transfer Protocol* — простой протокол передачи почты) — это широко используемый [сетевой протокол](https://ru.wikipedia.org/wiki/IP), предназначенный для передачи [электронной почты](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BE%D1%87%D1%82%D0%B0) в сетях TCP/

|  |  |
| --- | --- |
| **SMTP** | |
| **Название:** | Simple Mail Transfer Protocol |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Прикладной |
| **Семейство:** | TCP/IP |
| **Порт/ID:** | 25/TCP 465/TCP, 587/TCP (SMTP over SSL) |
| **Назначение протокола:** | Отправка электронной почты |

**DNS** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Domain Name System* — система доменных имён) — компьютерная [распределённая система](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D1%91%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) для получения информации о [доменах](https://ru.wikipedia.org/wiki/%D0%94%D0%BE%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%B8%D0%BC%D1%8F). Чаще всего используется для получения IP-адреса по имени [хоста](https://ru.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82) (компьютера или устройства), получения информации о маршрутизации почты, обслуживающих узлах для протоколов в домене ([SRV-запись](https://ru.wikipedia.org/wiki/SRV-%D0%B7%D0%B0%D0%BF%D0%B8%D1%81%D1%8C)).

|  |  |
| --- | --- |
| **DNS** | |
| **Название:** | Domain Name System |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Прикладной |
| **Семейство:** | TCP/IP |
| **Порт/ID:** | 53/[TCP](https://ru.wikipedia.org/wiki/TCP), 53/[UDP](https://ru.wikipedia.org/wiki/UDP) |
| **Назначение протокола:** | Разрешение доменных имён |

**WHOIS** (от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *who is* — «кто это?») — [сетевой протокол](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%BE%D0%B9_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB) [прикладного уровня](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI), базирующийся на протоколе [TCP](https://ru.wikipedia.org/wiki/TCP) ([порт](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%80%D1%82_(TCP/UDP)) 43). Основное применение — получение регистрационных данных о владельцах [доменных имён](https://ru.wikipedia.org/wiki/%D0%94%D0%BE%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%B8%D0%BC%D1%8F), [IP-адресов](https://ru.wikipedia.org/wiki/IP-%D0%B0%D0%B4%D1%80%D0%B5%D1%81) и [автономных систем](https://ru.wikipedia.org/wiki/%D0%90%D0%B2%D1%82%D0%BE%D0%BD%D0%BE%D0%BC%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_(%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82)).

**POP3** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Post Office Protocol Version 3* — [протокол](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D1%8B_%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) почтового отделения, версия 3) — стандартный [интернет-протокол прикладного уровня](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D1%8B_%D0%BF%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B3%D0%BE_%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D1%8F), используемый [клиентами электронной почты](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%87%D1%82%D0%BE%D0%B2%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0) для получения почты с удаленного [сервера](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%87%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) по [TCP/IP](https://ru.wikipedia.org/wiki/TCP/IP)-соединению.

|  |  |
| --- | --- |
| **POP3** | |
| **Название:** | Post Office Protocol 3 |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Прикладной |
| **Семейство:** | TCP/IP |
| **Порт/ID:** | 110/TCP |
| **Назначение протокола:** | Получение электронной почты |

**RDP** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Remote Desktop Protocol* — протокол удалённого рабочего стола) — протокол прикладного уровня, позаимствованный [Microsoft](https://ru.wikipedia.org/wiki/Microsoft) из купленной у PictureTel (ныне известной как Polycom) телекоммуникационной программы Liveshare Plus (названной впоследствии NetMeeting), использующийся для обеспечения удалённой работы пользователя с сервером, на котором запущен сервис [терминальных подключений](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF). Клиенты существуют практически для всех версий Windows (включая [Windows CE](https://ru.wikipedia.org/wiki/Windows_CE), [Phone](https://ru.wikipedia.org/wiki/Windows_Phone) и [Mobile](https://ru.wikipedia.org/wiki/Windows_Mobile)), [Linux](https://ru.wikipedia.org/wiki/Linux), [FreeBSD](https://ru.wikipedia.org/wiki/FreeBSD), [Mac OS X](https://ru.wikipedia.org/wiki/Mac_OS_X), [iOS](https://ru.wikipedia.org/wiki/IOS), [Android](https://ru.wikipedia.org/wiki/Android), [Symbian](https://ru.wikipedia.org/wiki/Symbian). По умолчанию используется порт TCP 3389.   
  
**IPv4** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Internet Protocol version 4*) — четвёртая версия интернет протокола (IP). Первая широко используемая версия. Протокол описан в [RFC 791](https://tools.ietf.org/html/rfc791)

|  |  |
| --- | --- |
| **IPv4** | |
| **Название:** | Internet Protocol version 4 |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Сетевой |
| **Семейство:** | [TCP/IP](https://ru.wikipedia.org/wiki/TCP/IP) |
| **Назначение протокола:** | Адресация |

IPv4 использует 32-[битные](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D1%82) (четырёх[байтные](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B9%D1%82)) адреса, ограничивающие [адресное пространство](https://ru.wikipedia.org/wiki/IP-%D0%B0%D0%B4%D1%80%D0%B5%D1%81) 4 294 967 296 (232) возможными уникальными адресами.

Традиционной формой записи [IPv4 адреса](https://ru.wikipedia.org/wiki/IP-%D0%B0%D0%B4%D1%80%D0%B5%D1%81" \o "IP-адрес) является запись в виде четырёх [десятичных чисел](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%81%D1%8F%D1%82%D0%B8%D1%87%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%81%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F) (от 0 до 255), разделённых точками. Через дробь указывается длина [маски подсети](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%81%D0%BA%D0%B0_%D0%BF%D0%BE%D0%B4%D1%81%D0%B5%D1%82%D0%B8).

**IPv6** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Internet Protocol version 6*) — новая версия [протокола](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%BE%D0%B9_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB) [IP](https://ru.wikipedia.org/wiki/IP), призванная решить проблемы, с которыми столкнулась предыдущая версия ([IPv4](https://ru.wikipedia.org/wiki/IPv4)) при её использовании в [Интернете](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82), за счёт использования длины адреса 128 [бит](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D1%82) вместо 32. 

|  |  |
| --- | --- |
| **IPv6** | |
| **Название:** | Internet Protocol version 6 |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Сетевой |
| **Семейство:** | [TCP/IP](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B5%D0%BA_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D0%BE%D0%B2_TCP/IP) |
| **Создан в:** | 1996 г. |
| **Назначение протокола:** | Адресация |

**MAC-адрес** (от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Media Access Control* — управление доступом к среде, также **Hardware Address**) — уникальный идентификатор, присваиваемый каждой единице активного оборудования или некоторым их интерфейсам в [компьютерных сетях](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D1%81%D0%B5%D1%82%D1%8C) Ethernet.  
Адреса вроде MAC-48 наиболее распространены; они используются в таких технологиях, как [Ethernet](https://ru.wikipedia.org/wiki/Ethernet), [Token ring](https://ru.wikipedia.org/wiki/Token_ring), [FDDI](https://ru.wikipedia.org/wiki/FDDI), [WiMAX](https://ru.wikipedia.org/wiki/WiMAX) и др. Они состоят из 48 [бит](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D1%82), таким образом, адресное пространство MAC-48 насчитывает 248 (или 281 474 976 710 656) адресов. Согласно подсчётам IEEE, этого запаса адресов хватит по меньшей мере до [2100 года](https://ru.wikipedia.org/wiki/2100_%D0%B3%D0%BE%D0%B4).

**TCP** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) ***t****ransmission****c****ontrol****p****rotocol* — протокол управления передачей) — один из основных [протоколов передачи данных](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB_%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) интернета, предназначенный для управления [передачей данных](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). Сети и подсети, в которых совместно используются протоколы TCP и [IP](https://ru.wikipedia.org/wiki/IP) называются сетями [TCP/IP](https://ru.wikipedia.org/wiki/TCP/IP).

|  |  |
| --- | --- |
| **TCP** | |
| **Название:** | Transmission Control Protocol |
| **Уровень (по**[**модели OSI**](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI)**):** | Транспортный |
| **Семейство:** | TCP/IP |

**UDP** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *User Datagram Protocol* — протокол пользовательских [датаграмм](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%B9%D1%82%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0)) — один из ключевых элементов [TCP/IP](https://ru.wikipedia.org/wiki/TCP/IP), набора сетевых протоколов для [Интернета](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82). С UDP компьютерные приложения могут посылать сообщения (в данном случае называемые [датаграммами](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%B9%D1%82%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0)) другим хостам по [IP-сети](https://ru.wikipedia.org/wiki/IP) без необходимости предварительного сообщения для установки специальных каналов передачи или путей данных. 

|  |  |
| --- | --- |
| **UDP** | |
| **Название:** | User Datagram Protocol |
| **Уровень (по [модели OSI](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_OSI" \o "Сетевая модель OSI)):** | [Транспортный](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D0%BD%D1%81%D0%BF%D0%BE%D1%80%D1%82%D0%BD%D1%8B%D0%B9_%D1%83%D1%80%D0%BE%D0%B2%D0%B5%D0%BD%D1%8C) |
| **Семейство:** | [TCP/IP](https://ru.wikipedia.org/wiki/TCP/IP) (иногда называют UDP/IP) |
| **Порт/ID:** | 17 (в IP) |

**NAT** (от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Network Address Translation* — «преобразование сетевых адресов») — это механизм в [сетях](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D1%81%D0%B5%D1%82%D1%8C) [TCP/IP](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B5%D0%BA_%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D0%BE%D0%B2_TCP/IP), позволяющий преобразовывать [IP-адреса](https://ru.wikipedia.org/wiki/IP-%D0%B0%D0%B4%D1%80%D0%B5%D1%81) транзитных [пакетов](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D0%BA%D0%B5%D1%82_(%D1%81%D0%B5%D1%82%D0%B5%D0%B2%D1%8B%D0%B5_%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D0%B8)). Также имеет названия *IP Masquerading*, *Network Masquerading* и*Native Address Translation*.  
NAT выполняет три важных функции.

1. **Позволяет сэкономить**[**IP-адреса**](https://ru.wikipedia.org/wiki/IP-%D0%B0%D0%B4%D1%80%D0%B5%D1%81) (только в случае использования NAT в режиме PAT), транслируя несколько внутренних IP-адресов в один внешний публичный IP-адрес (или в несколько, но меньшим количеством, чем внутренних). По такому принципу построено большинство сетей в мире: на небольшой район домашней сети местного провайдера или на офис выделяется 1 публичный (внешний) IP-адрес, за которым работают и получают доступ интерфейсы с приватными (внутренними) IP-адресами.
2. Позволяет предотвратить или ограничить обращение снаружи ко внутренним хостам, оставляя возможность обращения изнутри наружу. При инициации соединения изнутри сети создаётся трансляция. Ответные пакеты, поступающие снаружи, соответствуют созданной трансляции и поэтому пропускаются. Если для пакетов, поступающих снаружи, соответствующей трансляции не существует (а она может быть созданной при инициации соединения или статической), они не пропускаются.
3. Позволяет скрыть определённые внутренние сервисы внутренних хостов/серверов. По сути, выполняется та же указанная выше трансляция на определённый [порт](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%80%D1%82_(TCP/IP)), но возможно подменить внутренний порт официально зарегистрированной службы (например, 80-й порт [TCP](https://ru.wikipedia.org/wiki/TCP) ([HTTP](https://ru.wikipedia.org/wiki/HTTP)-сервер) на внешний 54055-й). Тем самым, снаружи, на внешнем IP-адресе после трансляции адресов на сайт (или форум) для осведомлённых посетителей можно будет попасть по адресу http://example.org:54055, но на внутреннем сервере, находящемся за NAT, он будет работать на обычном 80-м порту. Повышение безопасности и скрытие «непубличных» ресурсов.

**DMZ** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Demilitarized Zone* — демилитаризованная зона, ДМЗ) — [сегмент сети](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D0%B3%D0%BC%D0%B5%D0%BD%D1%82_%D1%81%D0%B5%D1%82%D0%B8), содержащий общедоступные [сервисы](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D1%8B%D0%B5_%D1%81%D0%B5%D1%80%D0%B2%D0%B8%D1%81%D1%8B) и отделяющий их от частных[[1]](https://ru.wikipedia.org/wiki/DMZ_(%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D1%8B%D0%B5_%D1%81%D0%B5%D1%82%D0%B8)#cite_note-pit-1). В качестве общедоступного может выступать, например, [веб-сервис](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%BB%D1%83%D0%B6%D0%B1%D0%B0): обеспечивающий его[сервер](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80_(%D0%B0%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5)), который физически размещён в локальной сети ([Интранет](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D1%80%D0%B0%D0%BD%D0%B5%D1%82" \o "Интранет)), должен отвечать на любые запросы из внешней сети ([Интернет](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82)), при этом другие локальные ресурсы (например, файловые серверы, рабочие станции) необходимо изолировать от внешнего доступа.

Клиент сервер архитектура

https://www.youtube.com/watch?v=CEbFC-joa4s