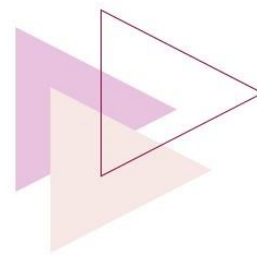


OSI модель



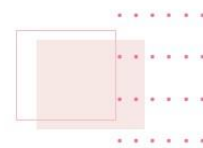
Савчук Анастасия



Что такое OSI?

- аббревиатура от английского Open Systems Interconnection model
- Сетевая модель взаимодействия открытых систем (эталонная модель)

OSI - сетевая модель - это модель взаимодействия сетевых протоколов. А протоколы в свою очередь, это стандарты, которые определяют каким образом, будут обмениваться данными различные программы.



Что такое OSI?

- На английском звучит как Open Systems Interconnection model или же на русский можно перевести как сетевая модель взаимодействия открытых систем (эталонная модель)

OSI - это скелет, фундамент и база всех сетевых сущностей, она представляет собой схему работы (или план действий по обмену данными) для сетевых устройств. Также OSI играет роль в создании новых сетевых протоколов, так как служит эталоном взаимодействия.

Проще говоря OSI - сетевая модель - это модель взаимодействия сетевых протоколов. А протоколы в свою очередь, это стандарты, которые определяют каким образом, будут обмениваться данными различные программы.

Так что сначала надо разобраться с тем, что такое протокол.

Допустим, вы с Васей (уже с другим, а может и тем же самым) живете в соседних домах. Телефон или интернет еще не изобрели, а передавать самые важные новости друг другу как-то надо. Что делать? Как

обмениваться информацией? Вы решаете, что будете махать друг другу разноцветными тряпками:

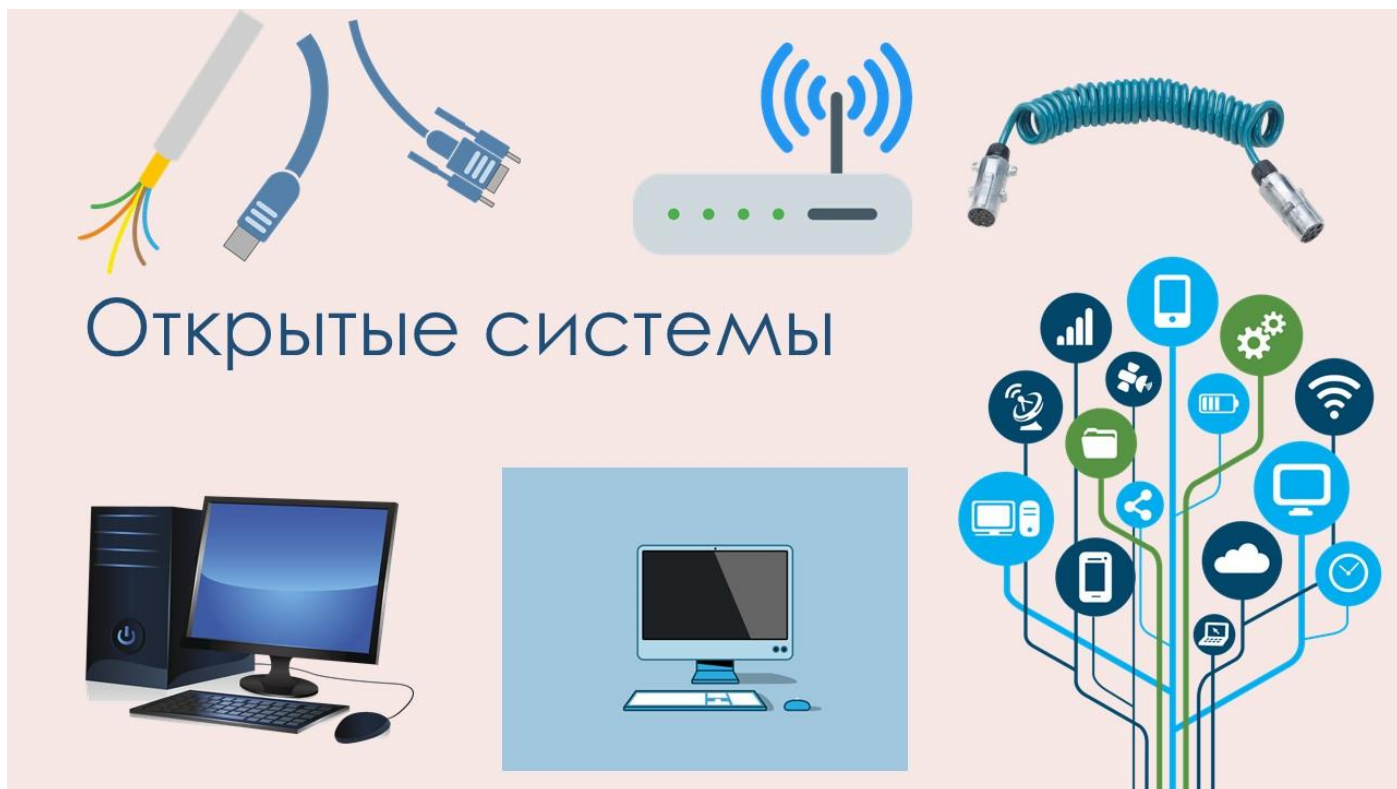
красная означает, что опасность рядом;

синяя означает, что по телевизору показывают что-то интересное;

зеленая, самая важная, означает, что скоро выйдет новый сезон Рика и Морти!

Этот свод правил, по сути, протокол вашего общения - набор правил, по которым вы обмениваетесь данными. Если завтра Вася будет махать фиолетовой тряпкой, вы его не поймете и это закономерно. А вот если зеленой... в общем, ура!

Именно протокол является стандартом, который определяет способ общения программ и серверов. OSI - это набор протоколов, каждый из которых необходим для решения отдельно поставленной задачи. Если хотя бы один исключить, что-то пойдет не так.



- Сетевая модель взаимодействия открытых систем
- Что такое открытые системы можете спросить вы?

Открытая **система** - это **система**, которая состоит из компонентов, взаимодействующих друг с другом через стандартные интерфейсы. То есть это те системы, которые могут обмениваться с окружающей средой веществом (а также энергией и импульсом). В нашем случае под открытыми системами можно понимать сетевое оборудование (компьютеры с сетевыми картами, коммутаторы, маршрутизаторы).

Зачем нужна OSI?

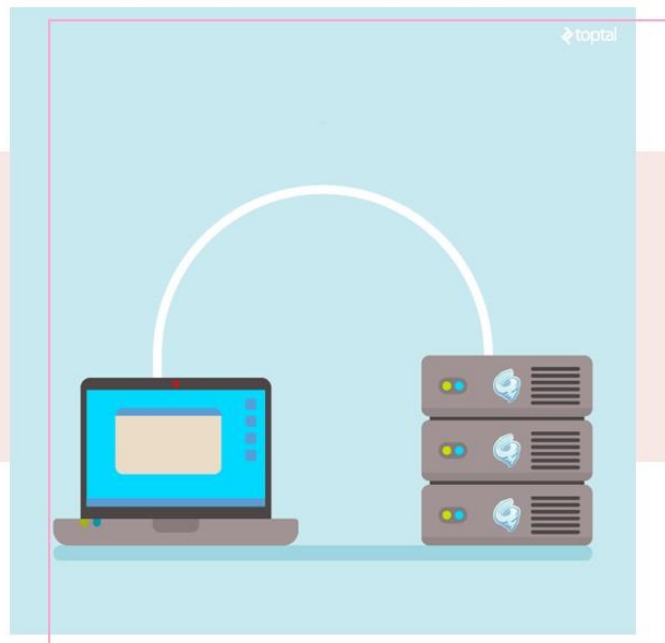
Она описывает как работают сетевые устройства



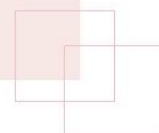
Зачем нужна OSI?

Она полностью описывает как работают сетевые устройства и она совершенно необходима для понимания того как работают компьютерные сети.

OSI - это процесс передачи данных

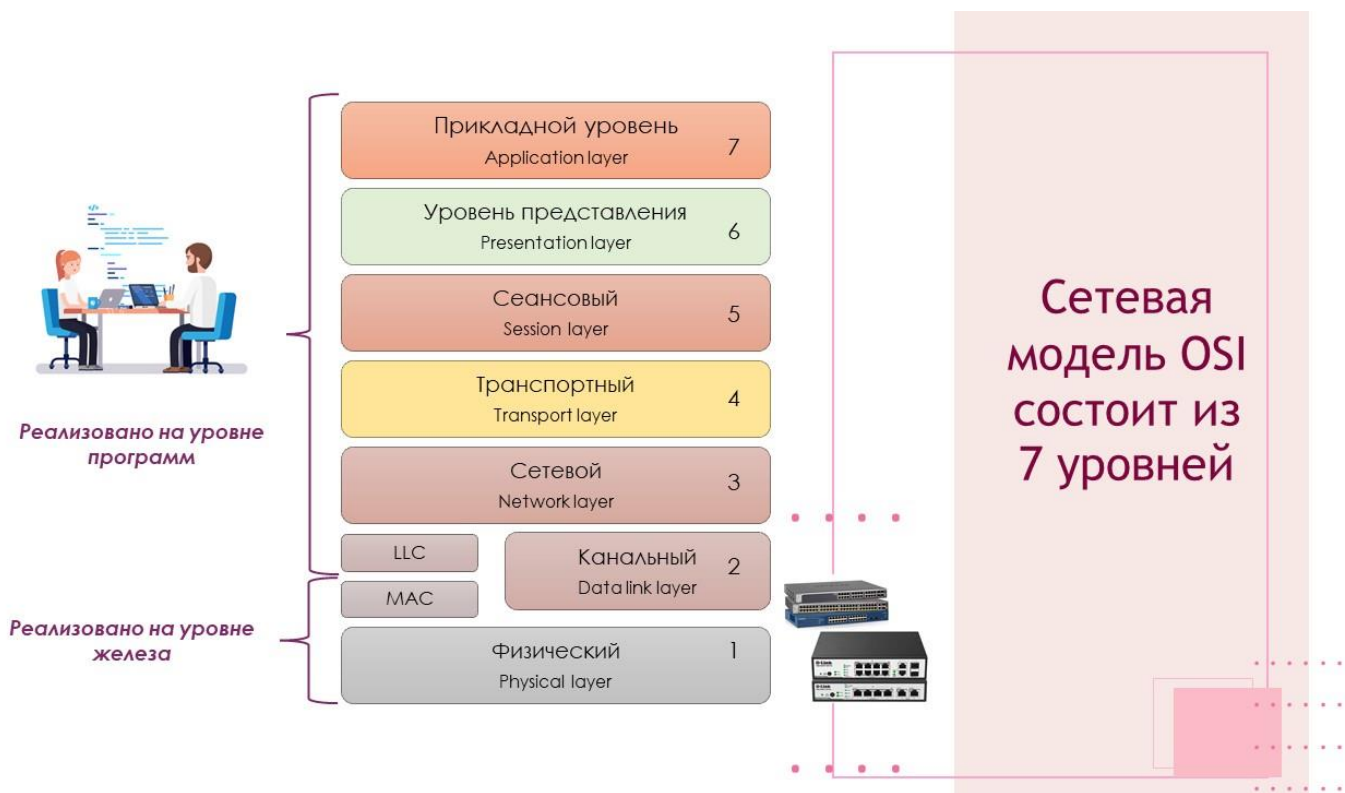


OSI проще говоря это процесс передачи данных



Поясню на примере: открывая любую страничку в интернете, сервер (где находится открываемая страничка) пересылает в Ваш браузер данные (гипертекстовый документ) по протоколу HTTP. Благодаря протоколу HTTP Ваш браузер, получая данные с сервера, знает, как их требуется обработать, и успешно обрабатывает их, показывая Вам запрашиваемую страничку.

Любой текст на веб-страничке заключен в специальные теги, которые указывают браузеру какой размер текста использовать, его цвет, расположение на странице (слева, справа или по центру). Это касается не только текста, но и картинок, форм, активных элементов и вообще всего контента, т.е. того, что есть на страничке. Браузер, обнаруживая теги, действует согласно их предписанию, и показывает Вам обработанные данные, которые заключены в эти теги.



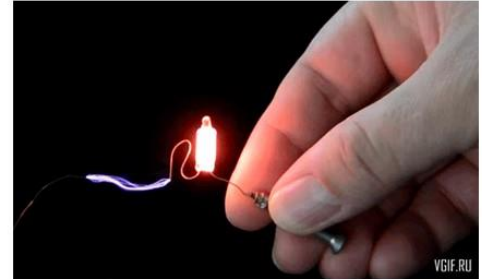
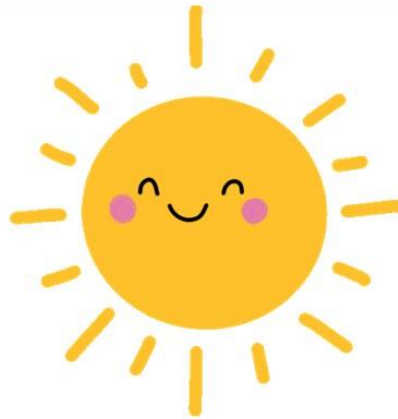
OSI проще говоря это процесс передачи данных

который разделили на 7 частей, каждый из которых отвечает за выполнение определенной функции

Сетевая модель OSI состоит из 7 уровней, причем принято начинать отсчёт с нижнего.

Нижние уровни отвечают за физические параметры передачи, такие как электрические сигналы. Да – да, сигналы в проводах передаются с помощью представления в токи :) Токи представляются в виде последовательности единиц и нулей (1 и 0), затем, данные декодируются и маршрутизируются по сети. Более высокие уровни охватывают запросы, связанные с представлением данных. Условно говоря, более высокие уровни отвечают за сетевые данные с точки зрения пользователя.

Физический уровень / physical layer



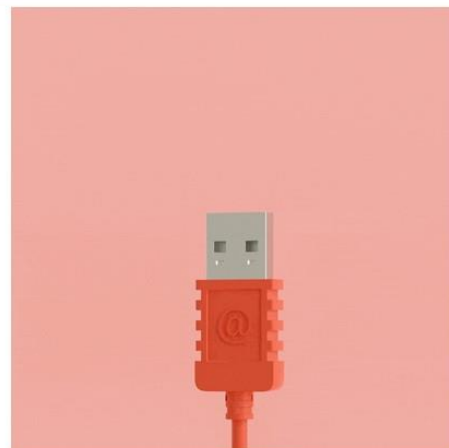
Самый нижний уровень, непосредственно осуществляющий передачу потока данных.
Основная единица - бит

Физический уровень (physical layer) – самый нижний уровень, непосредственно осуществляющий передачу потока данных.

На первом уровне **модели OSI** происходит передача физических сигналов (токов, света, радио) от источника к получателю.

Если убрать этот уровень, никакая информация от одного источника к другому поступать не будет.

Он описывает способы передачи отдельных бит через линии связи, соединяющие сетевые устройства. Кроме единиц и нулей физический уровень не знает ничего. На этом уровне решаются вопросы, связанные с синхронизацией, избавлением от помех, скоростью передачи данных и так далее. Короче, тут течет ток...



На этом уровне мы оперируем кабелями, контактами в разъемах, кодированием единиц и нулей, модуляцией сигнала и так далее.

На этом уровне мы оперируем кабелями, контактами в разъемах, кодированием единиц и нулей, модуляцией сигнала и так далее. Говоря простым языком это уровень проводов.

Среди технологий, которые живут на первом уровне, можно выделить самый основной стандарт -

Ethernet



Среди технологий, которые живут на первом уровне, можно выделить самый основной стандарт - Ethernet. Он есть сейчас в каждом доме. Он

описывает как должны эти сигналы кодироваться, передаваться по проводам и так далее.

Также на этом уровне перебивается Bluetooth, Wi-Fi и инфракрасный порт или IRDA



Также на этом уровне перебивается Bluetooth, Wifi и инфракрасный порт

Отметим, что в качестве носителя данных могут выступать не только электрические токи. Радиочастоты, световые или инфракрасные волны используются также повсеместно в современных сетях.

Сетевые устройства, которые относят к первому уровню это концентраторы и репитеры – то есть «глупые» железки, которые могут просто работать с физическим сигналом, не вникая в его логику (не декодируя).

Канальный уровень / data link layer



Ethernet



00:16:52:00:1f:03

MAC адрес



Коммутаторы

Второй уровень решает проблему адресации при передаче информации. Канальный уровень получает биты и превращает их в кадры (frame, также «фреймы»). Задача здесь — сформировать кадры с адресом отправителя и получателя, после чего отправить их по сети.

Когда два пользователя находятся в одной сети, состоящей только из двух устройств — это идеальный случай. Но что если этих устройств больше?

Представьте, мы получили физический сигнал с первого уровня — физического. Это набор напряжений разной амплитуды, волн или радиочастот. При получении, на втором уровне проверяются и исправляются ошибки передачи. На втором уровне мы оперируем понятием «фрейм», или как еще говорят «кадр».

Второй уровень решает проблему адресации при передаче информации. Канальный уровень получает биты и превращает их в кадры (frame, также «фреймы»). Задача здесь — сформировать кадры с адресом отправителя и получателя, после чего отправить их по сети.

Наверное, все слышали о MAC-адресе, вот он является физическим адресом. Они состоят из 48 бит и выглядят примерно так: 00:16:52:00:1f:03.

Устройства канального уровня – коммутаторы. их задача — передать сформированные кадры от одного устройства к другому, используя в качестве адресов только физические MAC-адреса.

Кстати! Стандарт Ethernet тоже тут. Он уютно расположился на первом и втором (1 и 2) уровнях модели OSI.



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, занимается проверкой и исправлением данных, управляет их передачей

MAC – управление доступом к передающей среде (Media Access Control), создан для взаимодействия с нижним уровнем, отвечает за присвоение физических MAC-адресов

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.

Идем вверх!



Сетевой уровень - третий уровень сетевой модели OSI. Он нужен для определения маршрута передачи данных. На этом уровне появляется новое понятие — маршрутизация. Для этой задачи были созданы устройства третьего уровня — маршрутизаторы (их еще называют роутерами). Маршрутизаторы получают MAC-адрес от коммутаторов с предыдущего уровня и занимаются построением маршрута от одного устройства к другому с учетом всех потенциальных неполадок в сети.

Протоколы, применяемые на этом этапе: интернет протокол IP; IPX, необходимый для маршрутизации пакетов в сетях и др.

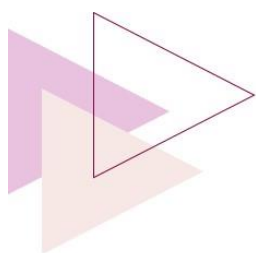
Именно на этом уровне происходит маршрутизация трафика, как таковая. Если мы хотим попасть на сайт **ksendzov.com**, то мы отправляем DNS – запрос, получаем ответ в виде IP – адреса и подставляем его в пакет. Да пакеты.

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

Именно он понимает по указанному IP-адресу где физически расположен адресат и как до него доставить данные.

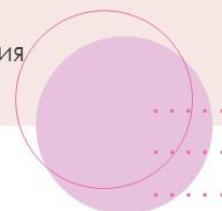
Без этого уровня данные просто заблудятся в интернете или попадут не туда, куда следует.

Кстати, для преобразования IP – адресов в MAC – адреса и обратно используется **протокол ARP** (Address Resolution Protocol — протокол определения адреса). С помощью него 64-битные MAC-адреса преобразуются в 32-битные IP-адреса и наоборот, тем самым обеспечивается инкапсуляция и декапсуляция данных.



Проблема масштабируемости - пакет

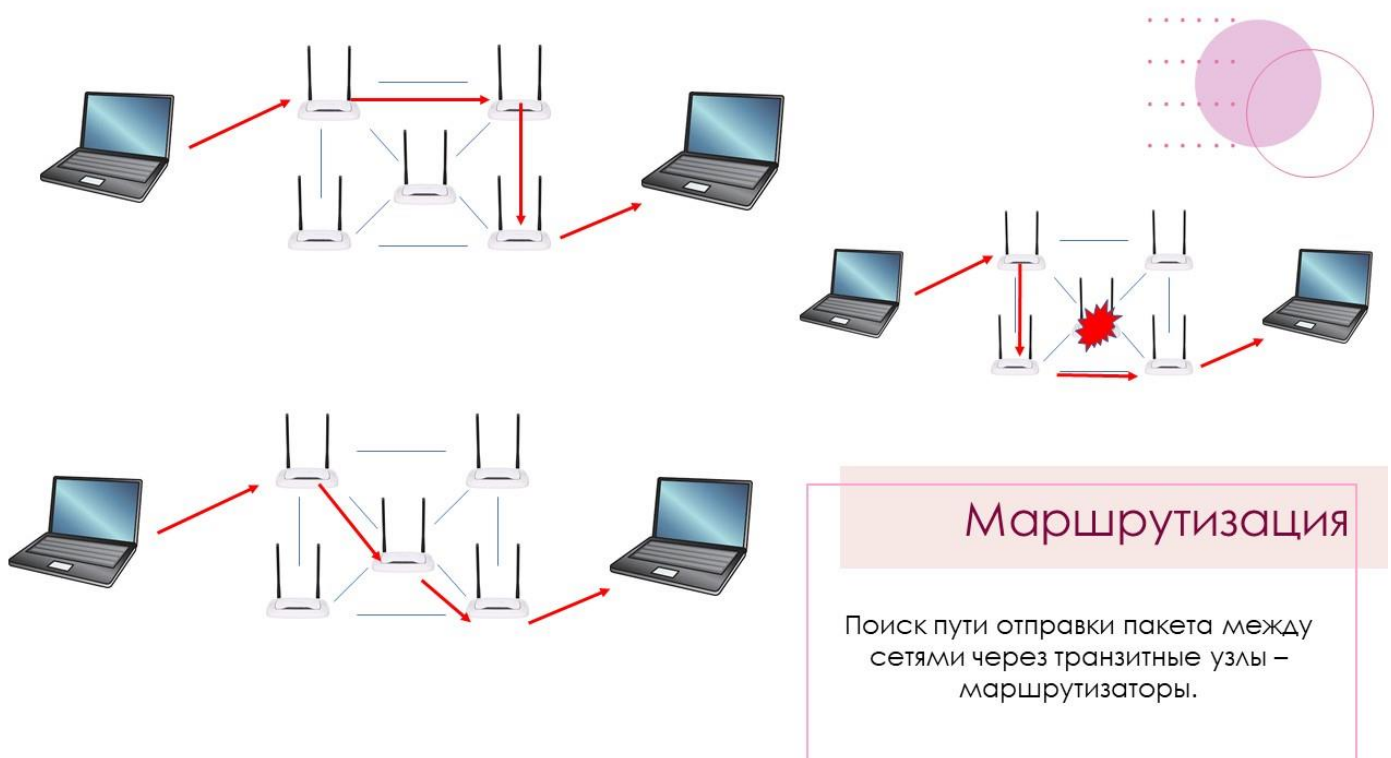
Работа ведется не с отдельными адресами, как на канальном уровне, а с блоками адресов. Пакеты, для которых не известен путь следования отбрасываются, а не пересылаются обратно на все порты. И существенное отличие от канального, возможность нескольких соединений между устройствами сетевого уровня и все эти соединения будут активными.



Мы можем передавать информацию от одного компьютера к другому через Ethernet и Wi-Fi, тогда зачем нужен еще один уровень? У технологии канального уровня (КУ) есть две проблемы, во-первых, технологии КУ отличаются друг от друга, во-вторых, есть ограничение по масштабированию, он ограничивается по расстоянию и максимальным размерам кадров. Wi-Fi просто гарантирует доставку сообщения, а Ethernet нет.

Как решается проблема масштабируемости на сетевом уровне?

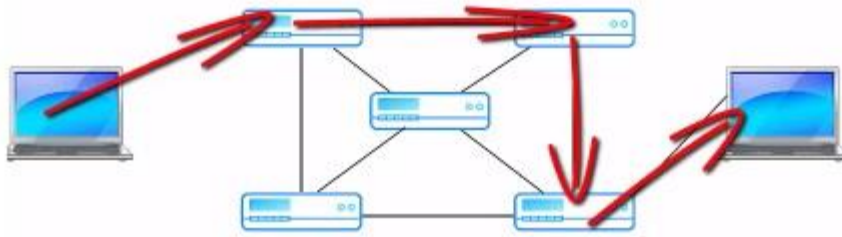
Работа ведется не с отдельными адресами, как на канальном уровне, а с блоками адресов. Пакеты, для которых не известен путь следования отбрасываются, а не пересылаются обратно на все порты. И существенное отличие от канального, возможность нескольких соединений между устройствами сетевого уровня и все эти соединения будут активными.



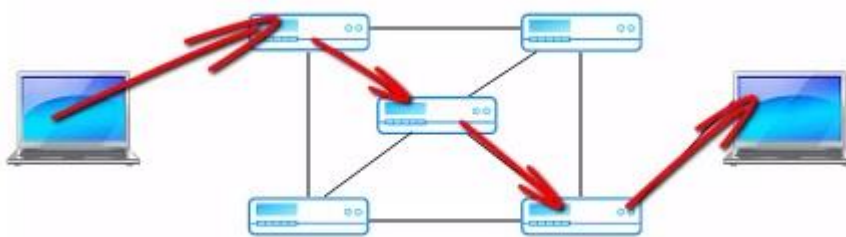
Из устройств здесь живет его величество маршрутизатор :)

Маршрутизация

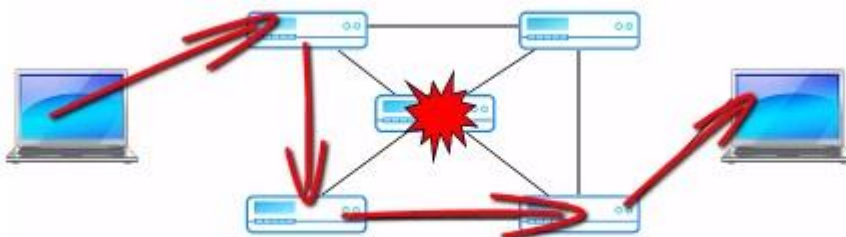
Поиск пути отправки пакета между сетями через транзитные узлы – маршрутизаторы. Рассмотрим пример выполнения маршрутизации. Схема состоит из 5 маршрутизаторов и двух компьютеров. Как могут передаваться данные от одного компьютера к другому? Как видите маршрутизаторы соединены в сеть. В этот раз данные передадутся таким путем



В следующий раз данные могут быть отправлены другим путем.



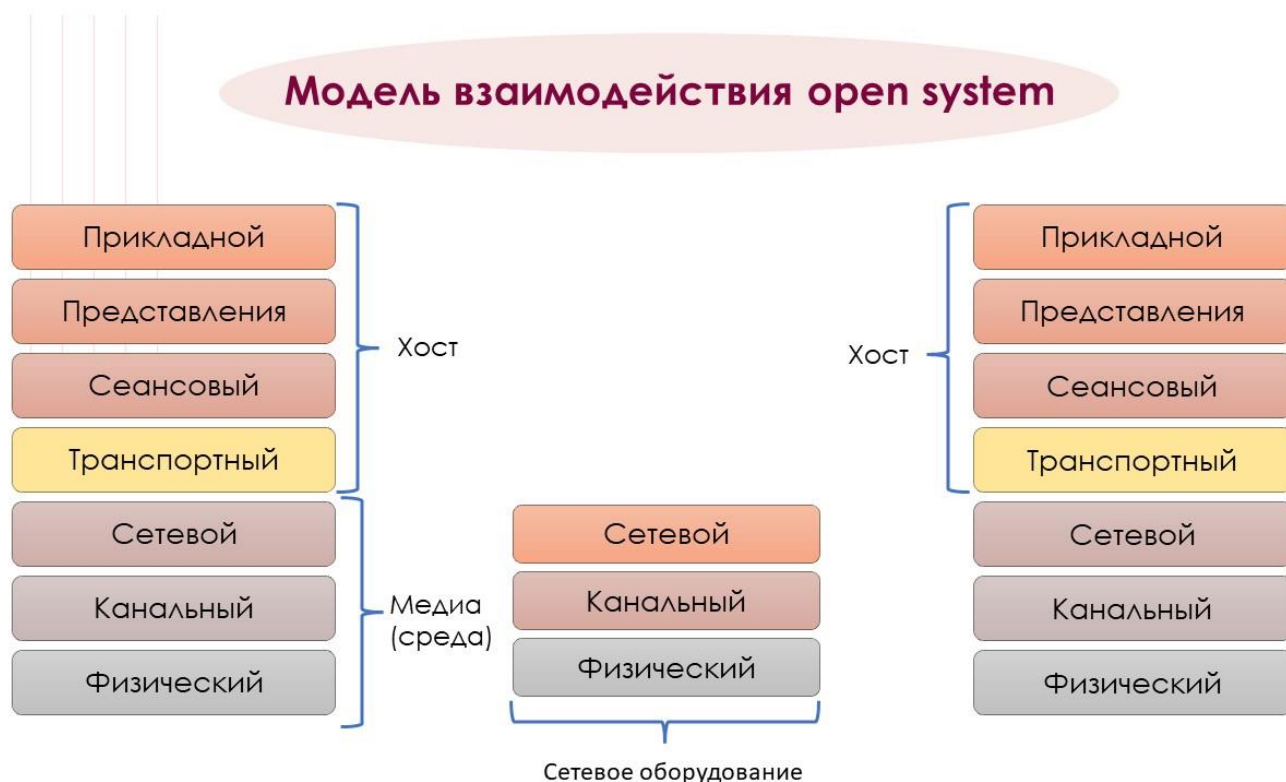
В случае поломки одного из маршрутизатора, ничего страшного не произойдет, можно найти путь в обход сломанного маршрутизатора.



Протоколы, применяемые на этом этапе: интернет протокол IP; IPX, необходимый для маршрутизации пакетов в сетях и др.

Процесс, когда данные передаются с верхних уровней на нижние называется **инкапсуляцией** данных, а когда наоборот, наверх, с первого,

физического к седьмому, то этот процесс называется **декапсуляцией** данных



Модель взаимодействия open system

Все семь уровней модели OSI можно условно разделить на две группы:

- Media layers (уровни среды),
- Host layers (уровни хоста).

Уровни группы Media Layers (L1, L2, L3) занимаются передачей информации (по кабелю или беспроводной сети), используются сетевыми устройствами, такими как коммутаторы, маршрутизаторы и т.п. Уровни группы Host Layers (L4, L5, L6, L7) используются непосредственно на устройствах, будь то стационарные компьютеры или портативные мобильные устройства.

Четвертый уровень — это посредник между Host Layers и Media Layers, относящийся скорее к первым, чем к последним, его главной задачей является транспортировка пакетов.

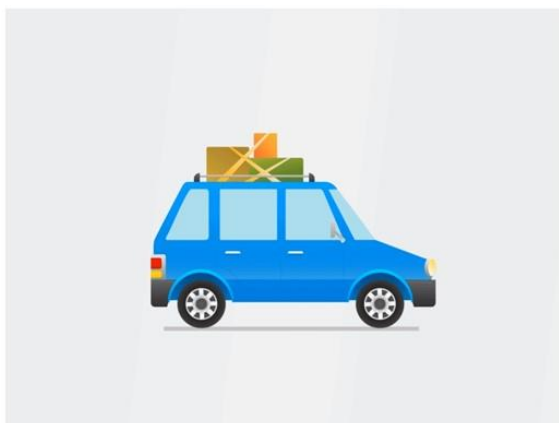
Хосты — это устройства где функционируют полезные пользовательские программы и сетевое оборудование, например, коммутаторы, маршрутизаторы.

Особенностью транспортного уровня является прямое взаимодействие одного компьютера с транспортным уровнем на другом компьютере, на остальных уровнях взаимодействие идет по звеньям цепи.

Такой уровень обеспечивает сквозное соединение между двумя взаимодействующими хостами. Данный уровень независим от сети, он позволяет скрыть от разработчиков приложений детали сетевого взаимодействия.

Для адресации на транспортном уровне используются порты, это числа от 1 до 65 535. Порты записываются вот так: 192.168.1.3:80 (IP адрес и порт).

Транспортный / transport layer



Четвертый уровень сетевой модели OSI. На этом уровне мы говорим о портах и протоколах TCP и UDP, которые отвечают за прямую связь между приложениями и за надежность доставки информации.

Для адресации на транспортном уровне используются порты, это числа от 1 до 65 535. Порты записываются вот так: 192.168.1.3:80 (IP адрес и порт).

Транспортный уровень (transport layer) – этот уровень обеспечивает надёжность передачи данных от отправителя к получателю.

Есть следующая задача, на компьютер, который соединен с составной сетью приходит пакет, на компьютере работает много сетевых приложений (веб-браузер, скайп, почта), нам необходимо понять какому

приложению нужно передать этот пакет. Взаимодействием сетевых приложений занимается транспортный уровень.

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 загрузке 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 сужение) букв не допустима, и когда скачиваем программы – тоже все проходит без ошибок.



Трафик чувствителен к потерям ?

TCP
(Transmission Control Protocol)!
Он обеспечивает контроль за передачей данных

для текста, программ, паролей, архивов и т.п.

Немного потеряем – не страшно?

UDP
(User Datagram Protocol)
вам подойдет.

для музыки, видео, видеоконференций и звонков используем UDP (передаем данные без проверки и без задержек)



Здесь две основных рок – звезды – TCP и UDP. Разница в том, что различный транспорт применяется для разной категории трафика. Принцип такой:

- **Трафик чувствителен к потерям** - нет проблем, TCP (Transmission Control Protocol)! Он обеспечивает контроль за передачей данных;
- а для текста, программ, паролей, архивов и т.п. – TCP (передача данных с подтверждением о получении, затрачивается больше времени).
- **Немного потеряем – не страшно** - по факту, когда вы читаете какую-то статью, пару пакетов могло и потеряться. Но это не чувствуется для вас, как для пользователя. UDP (User Datagram Protocol) вам подойдет. А если бы это была телефония? Потеря пакетов там критична, так как голос в реальном времени начнет попросту «квакать»;
- Следовательно, для музыки, видео, видеоконференций и звонков используем UDP (передаем данные без проверки и без задержек)

Надежность	Подтверждение получения данных, повторная отправка в случае необходимости, а также использование такого инструмента как тайм-аут. TCP гарантирует доставку пакетов	UDP ничего не гарантирует, потому что при получении отправленные данные могут
------------	--	---

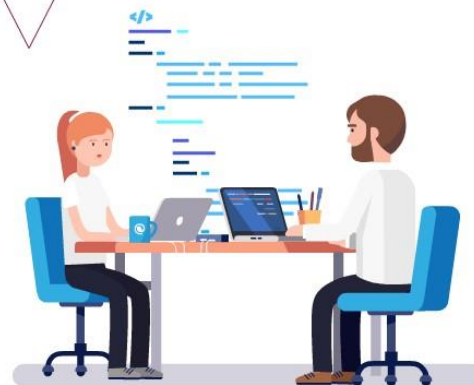
	неизменных последовательности и без потерь	виде, приходится полностью; не
Скорость	Более тяжеловесному TCP-протоколу будет требоваться больше времени для установки соединения, подтверждения получения, повторной отправки данных и т.д. ;	UDP обеспечивает более высокую скорость передачи данных.
Упорядоченность	Опять будет предпочтительнее TCP, поскольку этот протокол гарантирует передачу пакетов данных именно в том порядке, в котором они были отправлены.	В случае с UDP такой порядок не соблюдается;
Метод передачи данных	в случае с TCP данные передаются потоково, границы фрагментов данных не имеют обозначения.	В случае с UDP данные передаются в виде датаграмм – проверка пакетов на целостность осуществляется принимающей стороной только в случае получения сообщения. Также пакеты данных имеют определенные обозначения границ;

Сравнивая оба протокола, очевидно, что протокол TCP – это, можно сказать, «снайпер». Прицелился, выстрелил, зафиксировал попадание, ищет следующую цель. UDP – это, скорее, «пулеметчик» - выставил ствол в направлении врага и начал долбить очередями, не слишком заботясь о точности. Как в войсках важны обе эти воинские специальности, так и в интернете важны оба этих протокола. TCP применяется там, где требуется точная и подтверждаемая передача данных – например, отправка фотографий, или переписка между пользователями. UDP, в свою очередь, нужен для общения в голосовом формате, или при передаче потокового видео, например, с веб-камер или IP-камер.

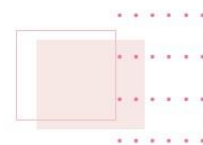
Сеансовый / Session

Пятый уровень сетевой модели OSI. Он отвечает за поддержку сеансов связи. Именно благодаря нему приложения могут взаимодействовать между собой по сети длительное время.

Без этого уровня не было бы онлайн-игр, видеочатов и прочих полезных вещей.



Протоколы
RPC, PAP, L2TP



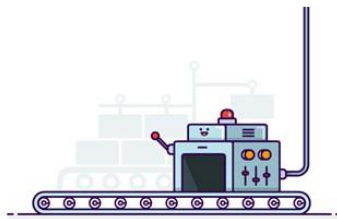
Попросите любого сетевого инженера объяснить вам сеансовый уровень. Ему будет трудно это сделать, инфа 100%. Дело в том, что в повседневной работе, сетевой инженер взаимодействует с первыми четырьмя уровнями – физическим, канальным, сетевым и транспортным. Остальные, или так называемые «верхние» уровни относятся больше к работе разработчиков софта

Сеансовый уровень или уровень сессий(session layer) – как видно из названия, он организует сеанс связи между компьютерами.

Пятый уровень оперирует чистыми данными; помимо пятого, чистые данные используются также на шестом и седьмом уровне.

Примером работы пятого уровня может служить видеозвонок по сети. На этом уровне устанавливается, каким кодеком будет кодироваться сигнал, причем этот кодек должен присутствовать на обеих машинах.

Во время видеосвязи необходимо, чтобы два потока данных (аудио и видео) шли синхронно. Когда к разговору двоих человек прибавится третий — получится уже конференция. Задача пятого уровня — сделать так, чтобы собеседники могли понять, кто сейчас говорит.



Уровень представления / Presentation



Работает с протоколами (стандартами)

изображений (JPEG, GIF, PNG, TIFF),
кодировок (ASCII, EBDIC),
музыки и видео (MPEG)
Шифрование TLS (Transport Layer Security)/SSL
(secure sockets layer)



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.

Помимо перечисленного, шестой уровень занимается шифрованием данных, когда при передаче их необходимо защитить

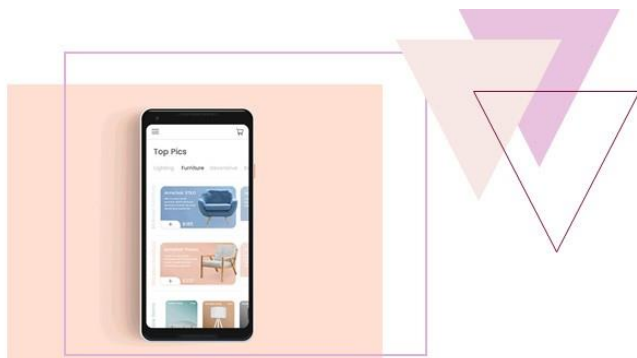
Функции – представить данные, передаваемых между прикладными процессами, в необходимой форме.

Для описания этого уровня, используют автоматический перевод в сети с различных языков. Например, Вы набираете номер телефона, говорите на русском, сеть автоматом переводит на французский язык, передает информацию в Испанию, там человек поднимает трубку и слышит Ваш вопрос на испанском языке. Это задача, пока не реализована.

Для защиты отправляемых данных по сети используется шифрование: secure sockets layer, а также transport layer security, эти технологии позволяют шифровать данные которые отправляются по сети.



Протоколы прикладного уровня используют TLS/SSL и их можно отличить по букве s в конце. Например, https, ftps и другие. Если в браузере Вы видите, что используется протокол https и замок, это значит, что производится защита данных по сети при помощи шифрования.

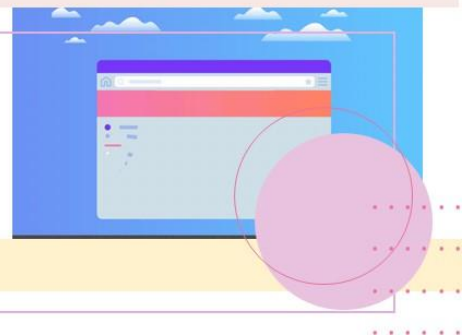


На нем работают все протоколы высокого уровня:

HTTP, POP, SMTP, RDP и так далее.

Прикладной / application layer

Протоколы здесь не должны задумываться о маршрутизации или гарантии доставки информации (этим занимаются предыдущие уровни). На седьмом уровне необходима лишь реализации конкретных действий. Без этого уровня наши компьютеры не могли бы получать html-код, email-сообщения или сообщения в чате.



Седьмой уровень иногда еще называют уровень приложений, но чтобы не запутаться можно использовать оригинальное название — application layer. Прикладной уровень — это то, с чем взаимодействуют пользователи, своего рода графический интерфейс всей модели OSI, с другими он взаимодействует по минимуму.

Все услуги, получаемые седьмым уровнем от других, используются для доставки данных до пользователя. Протоколам седьмого уровня не требуется обеспечивать маршрутизацию или гарантировать доставку данных, когда об этом уже позаботились предыдущие шесть. Задача седьмого уровня — использовать свои протоколы, чтобы пользователь увидел данные в понятном ему виде.

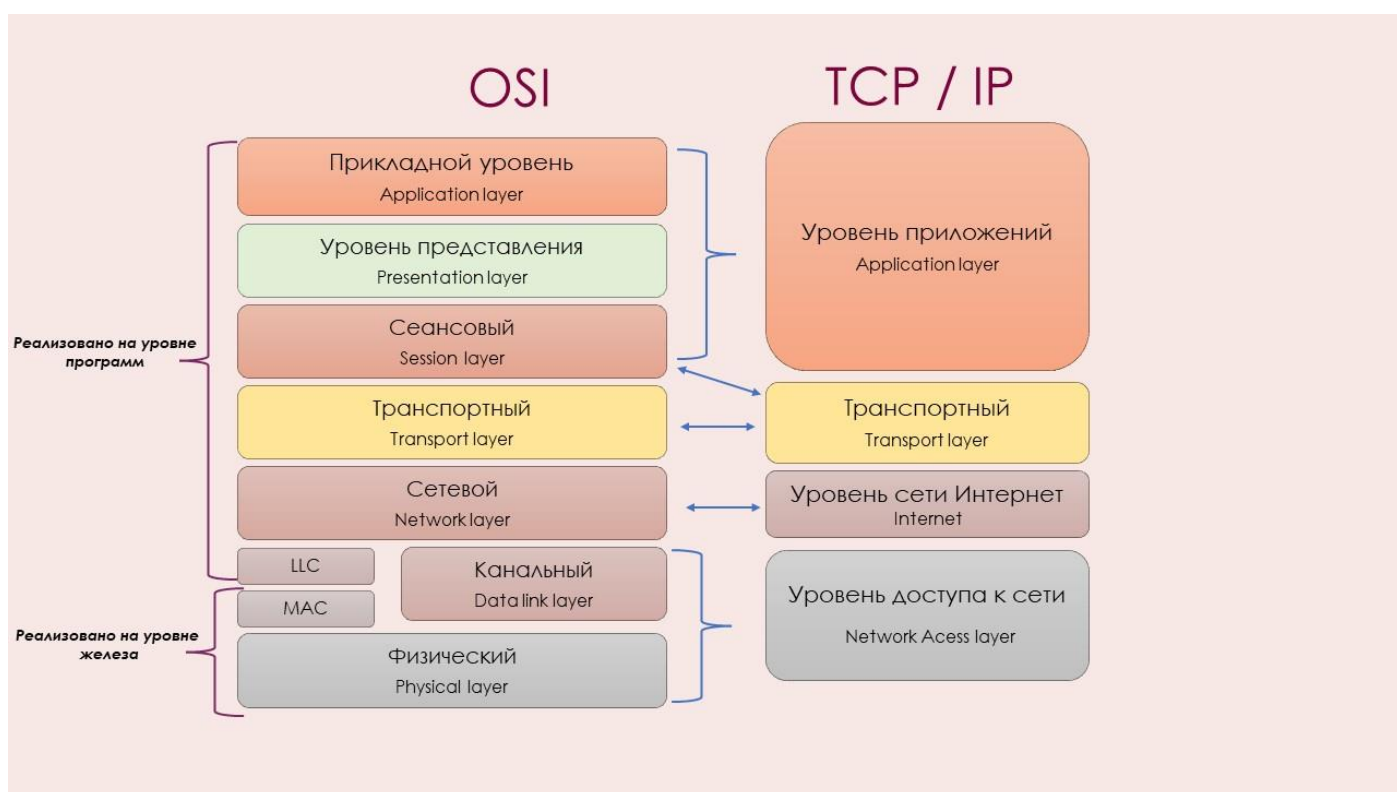
Он осуществляет связь пользовательских приложений с сетью. Эти приложения нам всем знакомы: просмотр веб-страниц (HTTP), передача и приём почты (SMTP, POP3), приём и получение файлов (FTP, TFTP), удаленный доступ (Telnet) и т.д.

Без этого уровня наши компьютеры не могли бы получать html-код, email-сообщения или сообщения в чате.



Уровни OSI

Уровень			Тип данных	Функции	
Host	7	Прикладной Application	Данные (сообщения)	Доступ к сетевым службам	HTTP, FTP, POP3, WebS ocket
	6	Представления Presentation		Представление и шифрование данных	ASCII, EBCDIC
	5	Сеансовый Session		Управление сеансом связи	RPC, PAP, L2TP
	4	Транспортный Transport	Сегменты или дейтаграммы	Прямая связь между конечными пунктами и надежность	TCP, UDP, SCTP, PORTS
Media	3	Сетевой Network	Пакеты	Определение маршрута и логическая адресация	IPv4, IPv6, IPsec, Apple Talk
	2	Канальный Data link	Биты / frame / кадры	Физическая адресация	PPP, IEEE 802.22, Ethernet, DSL, ARP, сетевая карта.
	1	Физический Physical	Биты	Работа со средой передачи, сигналами и двоичными данными	USB, кабель («витая пара»), коаксиальный, оптоволоконный), радиоканал



Критика модели OSI

Семиуровневая модель была принята в качестве стандарта ISO/IEC 7498, действующего по сей день, однако, модель имеет свои недостатки.

Среди основных недостатков говорят о неподходящем времени, плохой технологии, поздней имплементации, неудачной политике.

Первый недостаток — это неподходящее время. На разработку модели было потрачено неоправданно большое количество времени, но разработчики не уделили достаточное внимание существующим в то время стандартам. В связи с этим модель обвиняют в том, что она не отражает действительность. В таких утверждениях есть доля истины, ведь уже на момент появления OSI другие компании были больше готовы работать с получившей широкое распространение моделью TCP/IP.

Вторым недостатком называют плохую технологию. Как основной довод в пользу того, что OSI — это плохая технология, приводят распространенность стека TCP/IP. Протоколы OSI часто дублируют друг друга, функции распределены по уровням неравнозначно, а одни и те же задачи могут быть решены на разных уровнях.

Разделение на семь уровней было скорее политическим, чем техническим. При построении сетей в реальности редко используют уровни 5 и 6, а часто можно обойтись только первыми четырьмя. Даже изначальное описание архитектуры в распечатанном виде имеет толщину в один метр.

Кроме того, в отличие от TCP/IP, OSI никогда не ассоциировалась с UNIX. Добиться широкого распространения OSI не получилось потому, что она проектировалась как закрытая модель, продвигаемая Европейскими телекоммуникационными компаниями и правительством США. Стек протоколов TCP/IP изначально был открыт для всех, что позволило ему набрать популярность среди сторонников открытого программного кода.

Даже несмотря на то, что основные проблемы архитектуры OSI были политическими, репутация была запятнана и модель не получила распространения. Тем не менее, в сетевых технологиях, при работе с коммутацией даже сегодня обычно используют модель OSI.

Сравнивая слои TCP/IP-модели, и модели OSI, прикладной уровень протокола TCP/IP-модели аналогичен комбинации слоев 5, 6, 7 модели OSI, но TCP/IP-модель не имеет отдельного уровня представления и сеансового уровня. Транспортный уровень протокола TCP/IP включает в себя функции транспортного уровня OSI и некоторые функции сеансового уровня модели OSI. Уровень доступа сети модели TCP/IP охватывает канальный и физический уровни модели OSI. Обратите внимание, что сетевой уровень TCP/IP не использует преимущества служб последовательности и подтверждения, которые могут присутствовать на канальном уровне передачи данных модели OSI. Это ответственность транспортного уровня в модели TCP/IP.

Учитывая значения двух моделей, модель OSI является концептуальной моделью. Она в основном используется для описания, обсуждения и понимания отдельных сетевых функций. Однако, TCP/IP в первую очередь сконструирована для того чтобы разрешить специфический круг проблем, а не действовать как описание поколения для всех сетевых взаимодействий как модель OSI. Модель OSI является общей, независимой от протокола, но большинство протоколов и систем придерживаются ее, в то время как модель TCP/IP основана на стандартных протоколах, которые разработал интернет. Другой момент, который следует отметить в модели OSI заключается в том, что не все уровни используются в более простых приложениях. В то время как уровни 1, 2, 3 являются обязательными для любой передачи данных, приложение может использовать какой-то уникальный интерфейс уровня вместо обычных верхних уровней в модели.

Заключение

Модель TCP/IP и модель OSI являются концептуальными моделями, используемыми для описания всех сетевых коммуникаций, в то время как TCP/IP сама по себе также является важным протоколом, используемым

во всех операциях Интернета. Как правило, когда мы говорим об уровне 2, уровне 3 или уровне 7, в котором работает сетевое устройство, мы имеем в виду модель OSI. Модели TCP/IP используется как для моделирования текущей архитектуры Интернета и обеспечивают набор правил, которым следуют все формы передачи по сети.

Основа для сравнения	Модель TCP / IP	Модель OSI
Расширяется до	TCP / IP - протокол управления передачей / интернет-протокол	OSI - Открытая система Interconnect
Имея в виду	Это модель клиент-сервер, используемая для передачи данных через Интернет.	Это теоретическая модель, которая используется для вычислительной системы.
Количество слоев	4 слоя	7 слоев
Разработан	Министерство обороны (DoD)	ISO (Международная организация по стандартизации)
осязаемый	да	нет
использование	В основном используется	Никогда не использовался

Основные различия между TCP / IP и моделью OSI

1. TCP / IP - это модель клиент-сервер, т. Е. Когда клиент запрашивает сервис, он предоставляется сервером. Принимая во внимание, что OSI является концептуальной моделью.

2. TCP / IP - это стандартный протокол, используемый для каждой сети, включая Интернет, тогда как OSI - это не протокол, а эталонная модель, используемая для понимания и проектирования архитектуры системы.

3. TCP / IP - это четырехуровневая модель, тогда как OSI имеет семь уровней.

4. TCP / IP следует вертикальному подходу. С другой стороны, модель OSI поддерживает горизонтальный подход.

5. TCP / IP является материальным, а OSI - нет.

6. TCP / IP следует принципу «сверху вниз», а модель OSI - подходу «снизу вверх».

Заключение

Мы можем заключить, что модель TCP / IP является надежной по сравнению с моделью OSI, TCP / IP используется для сквозного соединения, чтобы передавать данные через Интернет. TCP / IP является надежным, гибким, осязаемым, а также предлагает способ передачи данных через Интернет. Транспортный уровень модели TCP / IP проверяет, поступили ли данные по порядку, есть ли у них ошибка или нет, потерянные пакеты отправлены или нет, подтверждение получено или нет и т. Д.

Источники

<https://www.learnqa.ru/tpost/9rgayjrhk-pro-model-osi-bistro-i-prosto>

<https://vladislaveremeev.github.io/> QABible

http://infocisco.ru/network_model_osi.html

http://infocisco.ru/network_model_tcpip.html

<https://zvondozone.ru/tehnologii/model-osi>

<https://lanmarket.ua/stats/modeli-OSI---posobie-dlya-nachinayushchih/>

<https://selectel.ru/blog/osi-for-beginners/>

<https://wiki.merionet.ru/seti/23/tcp-i-udp-v-chem-raznica/>

<https://ru.gadget-info.com/difference-between-tcp-ip>