**Сетевые модели. Часть 1. OSI.**

Определенно начинать лучше с теории, и затем, плавно, переходить к практике. Поэтому сначала рассмотрим сетевую модель (теоретическая модель), а затем приоткроем занавес на то, как теоретическая сетевая модель вписывается в сетевую инфраструктуру (на сетевое оборудование, компьютеры пользователей, кабели, радиоволны и т.д.).

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

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.

Если Вы еще не в курсе что из себя представляет страничка в интернете, то объясню в двух словах: 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. Вы и сами можете увидеть теги этой странички (и этот текст между тегами), для этого зайдите в меню вашего браузера и выберите - просмотр исходного кода.

Не будем сильно отвлекаться, "Сетевая модель" нужная тема для тех, кто хочет стать специалистом. Эта статья состоит из 3х частей и для Вас, Я постарался написать не скучно, понятливо и коротко. Для получения подробностей, или получения дополнительного разъяснения отпишитесь в комментариях внизу страницы, и я непременно помогу Вам.

Мы, как и в Сетевой Академии Cisco рассмотрим две сетевые модели: модель OSI и модель TCP/IP (иногда её называют DOD), а заодно и сравним их.

**Эталонная сетевая модель OSI**

OSI расшифровывается как Open System Interconnection. На русском языке это звучит следующим образом: Сетевая модель взаимодействия открытых систем (эталонная модель). Эту модель можно смело назвать стандартом. Именно этой модели придерживаются производители сетевых устройств, когда разрабатывают новые продукты.

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

Перечислим их:

* 7. Прикладной уровень (application layer)
* 6. Представительский уровень или уровень представления (presentation layer)
* 5. Сеансовый уровень (session layer)
* 4. Транспортный уровень (transport layer)
* 3. Сетевой уровень (network layer)
* 2. Канальный уровень (data link layer)
* 1. Физический уровень (physical layer)

[](http://infocisco.ru/articles/OSI.png)

Как говорилось выше, сетевая модель – это модель взаимодействия сетевых протоколов (стандартов), вот на каждом уровне и присутствуют свои протоколы. Перечислять их скучный процесс (да и не к чему), поэтому лучше разберем все на примере, ведь усваиваемость материала на примерах гораздо выше ;)

**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.

**Заключение**

Вот мы и разобрали сетевую модель OSI. В следующей части приступим к Сетевой модели TCP/IP, она меньше и протоколы те же. Для успешной сдачи тестов CCNA надо провести сравнение и выявить отличия, что и будет сделано.