

Сетевая модель OSI

Сетевая модель OSI (базовая эталонная модель взаимодействия открытых систем, англ. Open Systems Interconnection Basic Reference Model) — абстрактная сетевая модель для коммуникаций и разработки сетевых протоколов. Представляет уровневый подход к сети. Каждый уровень обслуживает свою часть процесса взаимодействия. Благодаря такой структуре совместная работа сетевого оборудования и программного обеспечения становится гораздо проще и прозрачнее.

В настоящее время основным используемым семейством протоколов является TCP/IP, разработка которого не была связана с моделью OSI.

Уровни модели OSI

Модель состоит из 7-ми уровней, расположенных друг над другом. Уровни взаимодействуют друг с другом (по «вертикали») посредством интерфейсов, и могут взаимодействовать с параллельным уровнем другой системы (по «горизонтали») с помощью протоколов. Каждый уровень может взаимодействовать только со своими соседями и выполнять отведённые только ему функции. Подробнее можно посмотреть на рисунке.

Модель OSI

Данные	Прикладной доступ к сетевым службам
Данные	Представления представление и кодирование данных
Данные	Сеансовый Управление сеансом связи
Блоки	Транспортный безопасное и надёжное соединие точка-точка
Пакеты	Сетевой Определение пути и IP (логическая адресация)
Кадры	Канальный MAC и LLC (Физическая адресация)
Биты	Физический кабель, сигналы, бинарная передача данных

Рис.1. Модель OSI.

Уровень приложений (англ. Application layer)

Верхний уровень модели, обеспечивает взаимодействие пользовательских приложений с сетью. Этот уровень позволяет приложениям использовать сетевые службы, такие как удалённый доступ к файлам и базам данных, пересылка электронной почты. Также отвечает за передачу служебной информации, предоставляет приложениям информацию об ошибках и формирует запросы к уровню представления. Пример: HTTP, POP3, SMTP, FTP, XMPP, OSCAR, BitTorrent, MODBUS, SIP

Уровень представления (англ. Presentation layer)

Этот уровень отвечает за преобразование протоколов и кодирование/декодирование данных. Запросы приложений, полученные с уровня приложений, он преобразует в формат для передачи по сети, а полученные из сети данные преобразует в формат, понятный приложениям. На этом уровне может осуществляться сжатие/распаковка или кодирование/декодирование данных, а также перенаправление запросов другому сетевому ресурсу, если они не могут быть обработаны локально.

Уровень 6 (представлений) эталонной модели OSI обычно представляет собой промежуточный протокол для преобразования информации из соседних уровней. Это позволяет осуществлять обмен между приложениями на разнородных компьютерных системах прозрачным для приложений образом. Уровень представлений обеспечивает форматирование и преобразование кода. Форматирование кода используется для того, чтобы гарантировать приложению поступление информации для обработки, которая имела бы для него смысл. При необходимости этот уровень может выполнять перевод из одного формата данных в другой. Уровень представлений имеет дело не только с форматами и представлением данных, он также занимается структурами данных, которые используются программами. Таким образом, уровень 6 обеспечивает организацию данных при их пересылке.

Чтобы понять, как это работает, представим, что имеются две системы. Одна использует для представления данных расширенный двоичный код

обмена информацией EBCDIC, например, это может быть мэйнфрейм компании IBM, а другая — американский стандартный код обмена информацией ASCII (его используют большинство других производителей компьютеров). Если этим двум системам необходимо обмениваться информацией, то нужен уровень представлений, который выполнит преобразование и осуществит перевод между двумя различными форматами.

Другой функцией, выполняемой на уровне представлений, является шифрование данных, которое применяется в тех случаях, когда необходимо защитить передаваемую информацию от приема несанкционированными получателями. Чтобы решить эту задачу, процессы и коды, находящиеся на уровне представлений, должны выполнить преобразование данных. На этом уровне существуют и другие подпрограммы, которые сжимают тексты и преобразовывают графические изображения в битовые потоки, так что они могут передаваться по сети.

Стандарты уровня представлений также определяют способы представления графических изображений. Для этих целей может использоваться формат PICT — формат изображений, применяемый для передачи графики QuickDraw между программами для компьютеров Macintosh и PowerPC. Другим форматом представлений является тэгированный формат файлов изображений TIFF, который обычно используется для растровых изображений с высоким разрешением. Следующим стандартом уровня представлений, который может использоваться для графических изображений, является стандарт, разработанный Объединенной экспертной группой по фотографии (Joint Photographic Expert Group); в повседневном пользовании этот стандарт называют просто JPEG.

Существует другая группа стандартов уровня представлений, которая определяет представление звука и кинофрагментов. Сюда входят интерфейс электронных музыкальных инструментов MIDI (Musical Instrument Digital Interface) для цифрового представления музыки, разработанный Экспертной

группой по кинематографии стандарт MPEG, используемый для сжатия и кодирования видеороликов на компакт-дисках, хранения в оцифрованном виде и передачи со скоростями до 1,5 Мбит/с, и QuickTime — стандарт, описывающий звуковые и видео элементы для программ, выполняемых на компьютерах Macintosh и PowerPC.

Пример: AFP - Apple Filing Protocol, ASCII - American Standard Code for Information Interchange, EBCDIC - Extended Binary Coded Decimal Interchange Code, ICA - Independent Computing Architecture, LPP - Lightweight Presentation Protocol, NCP - NetWare Core Protocol, NDR - Network Data Representation RDP - Remote Desktop Protocol, XDR - eXternal Data Representation, X.25 PAD - Packet Assembler/Disassembler Protocol

Сеансовый уровень (англ. Session layer)

5-й уровень модели отвечает за поддержание сеанса связи, позволяя приложениям взаимодействовать между собой длительное время. Уровень управляет созданием/завершением сеанса, обменом информацией, синхронизацией задач, определением права на передачу данных и поддержанием сеанса в периоды неактивности приложений. Синхронизация передачи обеспечивается помещением в поток данных контрольных точек, начиная с которых возобновляется процесс при нарушении взаимодействия.

Пример: ADSP (AppleTalk Data Stream Protocol), ASP (AppleTalk Session Protocol), H.245 (Call Control Protocol for Multimedia Communication), ISO-SP (OSI Session Layer Protocol (X.225, ISO 8327)), iSNS (Internet Storage Name Service), L2F (Layer 2 Forwarding Protocol), L2TP (Layer 2 Tunneling Protocol), NetBIOS (Network Basic Input Output System), PAP (Password Authentication Protocol), PPTP (Point-to-Point Tunneling Protocol), RPC (Remote Procedure Call Protocol), RTCP (Real-time Transport Control Protocol), SMPP (Short Message Peer-to-Peer), SCP (Secure Copy Protocol), SSH (Secure Shell), ZIP (Zone Information Protocol), SDP (Sockets Direct Protocol).

Транспортный уровень (англ. Transport layer)

4-й уровень модели предназначен для доставки данных без ошибок, потерь и дублирования в той последовательности, как они были переданы. При этом не важно, какие данные передаются, откуда и куда, то есть он предоставляет сам механизм передачи. Блоки данных он разделяет на фрагменты, размер которых зависит от протокола, короткие объединяет в один, а длинные разбивает. Пример: TCP, UDP.

Существует множество классов протоколов транспортного уровня, начиная от протоколов, предоставляющих только основные транспортные функции (например, функции передачи данных без подтверждения приема), и заканчивая протоколами, которые гарантируют доставку в пункт назначения нескольких пакетов данных в надлежащей последовательности, мультиплексируют несколько потоков данных, обеспечивают механизм управления потоками данных и гарантируют достоверность принятых данных.

Некоторые протоколы сетевого уровня, называемые протоколами без установки соединения, не гарантируют, что данные доставляются по назначению в том порядке, в котором они были посланы устройством-источником. Некоторые транспортные уровни справляются с этим, собирая данные в нужной последовательности до передачи их на сеансовый уровень. Мультиплексирование (multiplexing) данных означает, что транспортный уровень способен одновременно обрабатывать несколько потоков данных (потоки могут поступать и от различных приложений) между двумя системами. Механизм управления потоком данных — это механизм, позволяющий регулировать количество данных, передаваемых от одной системы к другой. Протоколы транспортного уровня часто имеют функцию контроля доставки данных, заставляя принимающую данные систему отправлять подтверждения передающей стороне о приеме данных.

Описать работу протоколов с установкой соединения можно на примере работы обычного телефона. Протоколы этого класса начинают

передачу данных с вызова или установки маршрута следования пакетов от источника к получателю. После чего начинают последовательную передачу данных и затем по окончании передачи разрывают связь.

Протоколы без установки соединения, которые посылают данные, содержащие полную адресную информацию в каждом пакете, работают аналогично почтовой системе. Каждое письмо или пакет содержит адрес отправителя и получателя. Далее каждый промежуточный почтамт или сетевое устройство считывает адресную информацию и принимает решение о маршрутизации данных. Письмо или пакет данных передается от одного промежуточного устройства к другому до тех пор, пока не будет доставлено получателю. Протоколы без установки соединения не гарантируют поступление информации получателю в том порядке, в котором она была отправлена. За установку данных в соответствующем порядке при использовании сетевых протоколов без установки соединения отвечают транспортные протоколы.

Пример: ATP (AppleTalk Transaction Protocol), CUDP (Cyclic UDP), DCCP (Datagram Congestion Control Protocol), FCP (Fiber Channel Protocol), IL (IL Protocol), NBF (NetBIOS Frames protocol), NCP (Netware Core Protocol), SCTP (Stream Control Transmission Protocol), SPX (Sequenced Packet Exchange), SST (Structured Stream Transport), TCP (Transmission Control Protocol), UDP (User Datagram Protocol).

Сетевой уровень (англ. Network layer)

3-й уровень сетевой модели OSI предназначен для определения пути передачи данных. Отвечает за трансляцию логических адресов и имён в физические, определение кратчайших маршрутов, коммутацию и маршрутизацию, отслеживание неполадок и заторов в сети. На этом уровне работает такое сетевое устройство, как маршрутизатор.

Протоколы сетевого уровня маршрутизируют данные от источника к получателю.

Пример: IP/IPv4/IPv6 (Internet Protocol), IPX (Internetwork Packet Exchange, протокол межсетевого обмена), X.25 (частично этот протокол реализован на уровне 2) CLNP (сетевой протокол без организации соединений), IPsec (Internet Protocol Security), ICMP (Internet Control Message Protocol), RIP (Routing Information Protocol), OSPF (Open Shortest Path First).

Канальный уровень (англ. Data Link layer)

Этот уровень предназначен для обеспечения взаимодействия сетей на физическом уровне и контроля за ошибками, которые могут возникнуть. Полученные с физического уровня данные он упаковывает во фреймы, проверяет на целостность, если нужно исправляет ошибки (посылает повторный запрос поврежденного кадра) и отправляет на сетевой уровень. Канальный уровень может взаимодействовать с одним или несколькими физическими уровнями, контролируя и управляя этим взаимодействием. Спецификация IEEE 802 разделяет этот уровень на 2 подуровня — MAC (Media Access Control) регулирует доступ к разделяемой физической среде, LLC (Logical Link Control) обеспечивает обслуживание сетевого уровня.

На этом уровне работают коммутаторы, мосты.

Протоколы: ARCnet, ATM, Cisco Discovery Protocol (CDP), Controller Area Network (CAN), Econet, Ethernet, Ethernet Automatic Protection Switching (EAPS), Fiber Distributed Data Interface (FDDI), Frame Relay, High-Level Data Link Control (HDLC), IEEE 802.2 (provides LLC functions to IEEE 802 MAC layers), IEEE 802.11 wireless LAN, Link Access Procedures, D channel (LAPD), LocalTalk, Multiprotocol Label Switching (MPLS), Point-to-Point Protocol (PPP), Serial Line Internet Protocol (SLIP) (obsolete), Spanning tree protocol, StarLan, Token ring, Unidirectional Link Detection (UDLD), x.25.

В программировании этот уровень представляет драйвер сетевой платы, в операционных системах имеется программный интерфейс взаимодействия канального и сетевого уровней между собой, это не новый уровень, а просто реализация модели для конкретной ОС. Примеры таких интерфейсов: ODI, NDIS

Физический уровень (англ. Physical layer)

Самый нижний уровень модели предназначен непосредственно для передачи потока данных. Осуществляет передачу электрических или оптических сигналов в кабель или в радиоэфир и, соответственно, их приём и преобразование в биты данных в соответствии с методами кодирования цифровых сигналов. Другими словами, осуществляет интерфейс между сетевым носителем и сетевым устройством.

На этом уровне работают концентраторы (хабы), повторители (ретрансляторы) сигнала и медиаконверторы.

Функции физического уровня реализуются на всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом. К физическому уровню относятся физические, электрические и механические интерфейсы между двумя системами. Физический уровень определяет такие свойства среды сети передачи данных как оптоволокно, витая пара, коаксиальный кабель, спутниковый канал передач данных и т. п. Стандартными типами сетевых интерфейсов, относящимися к физическому уровню, являются: V.35, RS-232C, RS-485, RJ-11, RJ-45, разъемы AUI и BNC.

Протоколы: IRDA, USB, EIA RS-232, EIA-422, EIA-423, RS-449, RS-485, Ethernet (включая 10BASE-T, 10BASE2, 10BASE5, 100BASE-TX, 100BASE-FX, 100BASE-T, 1000BASE-T, 1000BASE-SX и другие), 802.11 Wi-Fi, DSL, ISDN, SONET/SDH, GSM Um radio interface, IEEE 802.15, ITU и ITU-T, Firewire, TransferJet, Etherloop, ARINC 818, G.hn/G.9960.