

Лекция 1. МОДЕЛЬ OSI

Основные положения

1. Понятие сетевого стека

При обеспечении взаимодействия двух и более систем (под системой здесь мы понимаем любую программно-аппаратную сущность – компьютер, виртуальную машину, мобильное устройство и т. п.) необходимо решить множество задач. Например, согласовать физическое кодирование, логическое кодирование, обеспечить непротиворечивую адресацию на разных уровнях сетевой иерархии, обеспечить поиск маршрута, целостность передаваемых данных, подтверждения приема и т. д.

Классическим инженерным подходом для решения комплексных задач является декомпозиция задачи. Этот подход применяют и для сетевой коммуникации, когда общая задача коммуникации разделена на подзадачи. Подзадачи группируются по признаку близости целей. Подзадачи выполняются в определенной последовательности. Каждая из подзадач реализуется отдельным модулем. Каждый модуль описывается как черный ящик, то есть для него определены: формат входных и выходных данных, общие правила обработки данных. На взаимодействующих через сеть системах эти соглашения для идентичных модулей должны полностью совпадать. Такие соглашения и называются **протоколом**. Группа протоколов (модулей), выполняющих близкие по смыслу задачи называются уровнем стека. Упорядоченная последовательность уровней (и протоколов) называются **сетевым стеком**.

Модули протоколов взаимодействуют только с ближайшими соседями по стеку. Формализованные правила взаимодействия соседних уровней называются **интерфейсами**.

Современный подход предполагает, что в стеке формализуются архитектура стека, интерфейсы и протоколы, но не формализуется алгоритмическое обеспечение протоколов.

2. Предпосылки появления модели OSI

Модель OSI (The Open Systems Interconnection model) разрабатывалась с конца 70-х годов 20-го века комитетом ISO (International Organization for Standardization). Предпосылками появления модели явилось несовместимость существующих сетевых стеков (IBM SNA, ранние реализации TCP/IP, DecNet). Такая несовместимость, и связанные с ними патентные ограничения, ставили пользователей, и, в том числе правительственных заказчиков, в зависимость от одного поставщика, препятствовали развитию технологий и не позволяли совместно использовать наиболее удачные решения разных поставщиков.

3. Модель OSI – основные принципы

Модель OSI строилась в описанной выше идеологии сетевого стека. При этом:

- новые уровни выделились только при необходимости нового уровня абстракции,
- каждый новый уровень должен решать определенную функцию,
- границы между уровнями выбираются так, чтобы поток данных через интерфейс был минимальным,
- при выделении уровней учитывалась необходимость стандартизации международных протоколов,
- модель не должна описывать протоколы каждого уровня,
- модель описывает интерфейсы.

4. Архитектура модели

Модель OSI состоит из 7 уровней:

7. Прикладной (application)
6. Представления (presentation)
5. Сеансовый (session)
4. Транспортный (transport)
3. Сетевой (network)
2. Канальный (data link)
1. Физический (physical)

Каждый из уровней решает свои задачи. Каждый из уровней, кроме физического, добавляет к полученным данным свой служебный заголовок. Канальный уровень добавляет еще признак конца кадра – data trailer.

5. Передача данных по стеку

Данные попадают в стек от пользовательского процесса или непосредственно пользователя через консольный ввод сверху, через сетевой уровень.

Далее прикладной уровень добавляет свой заголовок (application header) и передает данные с заголовком вниз по стеку.

Уровень представления обрабатывает данные и заголовок предыдущего совместно, добавляя при этом заголовок уровня представления (presentation header). Таким образом происходит инкапсуляция сообщения вышестоящего уровня в сообщение нижестоящего.

Так происходит до канального уровня. Тот кроме заголовка добавляет в конец сообщения признак конца кадра. До канального уровня обработка идет внутри одной системы и интерфейсы предполагают надежное ограничение конца сообщения по его объему. Но сообщение канального уровня далее передается в виде физического сигнала по физическому уровню и необходимо надежно обнаруживать конец сообщения на принимающей системе.

Данные, содержащиеся в заголовках уровней, обрабатываются только аналогичным уровнем на принимающей системе, то есть каждый уровень работает так, как будто ниже по стеку к нему подключен аналогичный уровень передающей системы. Таким образом в стеке OSI реализовано горизонтальное логическое взаимодействие, тогда как физический поток данных вертикальный.

6. Сегментация на транспортном уровне

Объем данных при передаче от уровня к уровню растет за счет служебных заголовков. Однако на транспортном уровне происходит сегментация потока, когда поступившее от уровня сеансов сообщение делится на часть – сегменты. Каждый из сегментов снабжается отдельным заголовком транспортного уровня и передается отдельно дальше по стеку. Потом, на принимающей системе транспортный уровень соберет исходное сообщение из сегментов обратно. Такой механизм сегментации реализован для:

- a. разделения канала во времени,
- b. обеспечения надежности доставки.

Разделение канала во времени предполагает, что один физический канал связи, в рамках одного физического кодирования может передавать одновременно несколько потоков данных, если фрагменты потоков небольшого размера будут передаваться по очереди. Если не проводить сегментацию на транспортном уровне, то один поток будет передаваться до его исчерпания, в этом случае канал будет монополизирован одним соединением.

Надежность доставки обеспечивается проверкой кода обнаружения ошибки в заголовке транспортного уровня и квитованием (отправкой подтверждения приёма). В случае, если при проверке будет обнаружена ошибка, то проще и

быстрее переслать данные начиная с поврежденного сегмента, а не все исходное сообщение целиком.

7. Уровни модели

Рассмотрим основные функции уровней модели OSI. Для наглядности будем приводить примеры реальных протоколов из стека TCP/IP, функции которых близки функциям уровней OSI. Соответствие, с одной стороны, будет не полным, так как в стеке TCP/IP меньше уровней, но, с другой стороны, полезным, так как соединит модель OSI с повседневной практикой.

7. Прикладной уровень (application layer)

Основные функции: передача служебной информации приложений, предоставляет приложениям информацию об ошибках.

Примеры протоколов: FTP (File Transfer Protocol), Telnet (TErminaL NETwork), HTTP (HyperText Transfer Protocol), POP3 (Post Office Protocol Version 3), SMTP (Simple Mail Transfer Protocol).

6. Уровень представления данных (presentation layer)

Основные функции: сжатие данных, шифрование данных, перекодировка данных

Примеры протоколов: SSL (Secure Socket Layer), RDP — Remote Desktop Protocol

5. Сеансовый уровень (session layer)

Основные функции: обеспечение сеанса связи между приложениями (установление, поддержание и завершение сеанса) с целью взаимодействия приложений между собой длительное время.

Примеры протоколов: NetBIOS (Network Basic Input Output System), RPC (Remote Procedure Call Protocol)

4. Транспортный уровень (transport layer)

Основные функции: обеспечение надежной доставки данных, подтверждение приема и сегментацию потока, получаемого от сеансового уровня.

Примеры протоколов: TCP (Transmission Control Protocol), UDP (User Datagram Protocol)

3. Сетевой уровень (network layer)

Основные функции: решение задачи доставки данных по составной сети, межсетевую адресацию, маршрутизацию, трансляцию физических адресов в сетевые.

Примеры протоколов: IPv4/IPv6 (Internet Protocol), IPX (Internetwork Packet Exchange), IPsec (Internet Protocol Security), ICMP (Internet Control Message Protocol), RIP (Routing Information Protocol), OSPF (Open Shortest Path First), ARP (Address Resolution Protocol).

2. Канальный уровень (data link layer)

Основные функции: доставка данных по локальной сети, обеспечение формирования фреймов (frames) — кадров, передаваемых через физический уровень, контроль ошибок и управление потоком данных (data flow control), логическое кодирование данных.

Примеры протоколов: ATM, Ethernet, EAPS (Ethernet Automatic Protection Switching), FDDI (Fiber Distributed Data Interface), MPLS (Multiprotocol Label Switching), PPP (Point-to-Point Protocol), SLIP (Serial Line Internet Protocol)

1. Физический уровень (physical layer)

Основные функции: обеспечивающий физическое кодирование бит кадра в электрические, оптические или радиосигналы и передачу их по линиям связи. Определяет тип кабелей и разъемов, назначение контактов и формат физических сигналов.

Примеры протоколов: IEEE 802.15 (Bluetooth), IRDA, EIA RS-232, EIA-422, IEEE 802.3 (Ethernet), DSL, IEEE 802.11.

8. Взаимодействие сетевого и канального уровней

В модели OSI два уровня сетевой и канальный обеспечивают доставку данных между узлами сети. Одни по составной сети, второй по локальной. Рассмотрим процесс передачи сообщения между двумя узлами по составной сети, ограничившись описанием взаимодействия этих двух уровней. Под составной сетью будем понимать сеть, состоящую из локальных сетей, объединенных между собой маршрутизаторами, то есть через общий сетевой уровень.

Дадим необходимые соглашения и условные обозначения.

Введем два вида адресов канального уровня (аналог MAC-адресов). Адрес первого типа будет формироваться из трех строчных букв латинского алфавита, адрес второго – из трех прописных букв. Наличием двух разных типов адресов мы указываем на то, что составная сеть может состоять из локальных сетей с разными канальными протоколами. Если адрес состоит из трех букв "z", то это будет широковещательный адрес канального уровня и кадр, отправленный на этот адрес, принимают все узлы в локальной сети.

Общий для составной сети сетевой протокол будет иметь адреса (аналоги IP адресов), состоящие из двух цифр разделенных тире. Первая цифра указывает на адрес сети, вторая на адрес узла. Причем если в поле адреса узла стоит ноль, то это адрес сети целиком. При конфигурации узла будем указывать адрес шлюза в круглых скобках.

На рисунке1 приведем условные обозначения (a - узел сети, b - сеть, c – маршрутизатор, d - сетевое сообщение с адресом отправителя и адресом получателя, e - пример инкапсуляции сетевого сообщения (пакета) в сообщение канального уровня (в кадр).

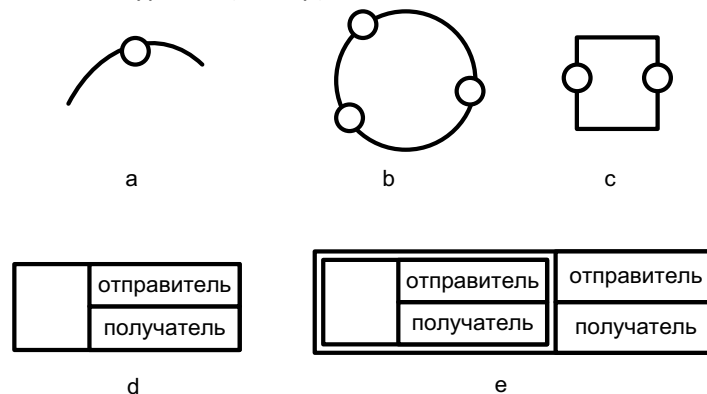


Рисунок 1

Примем упрощенные таблицы маршрутизации, в которых указывается адрес сети назначения, порт и шлюз. При передаче сообщения маршрутизатор по адресу назначения, содержащегося в заголовке пакета, определяет адрес сети назначения и по таблице маршрутизации определяет, через какой порт и на какой шлюз необходимо передавать его на следующем этапе маршрута.

На рисунке 2 показана составная сеть с адресной информацией.

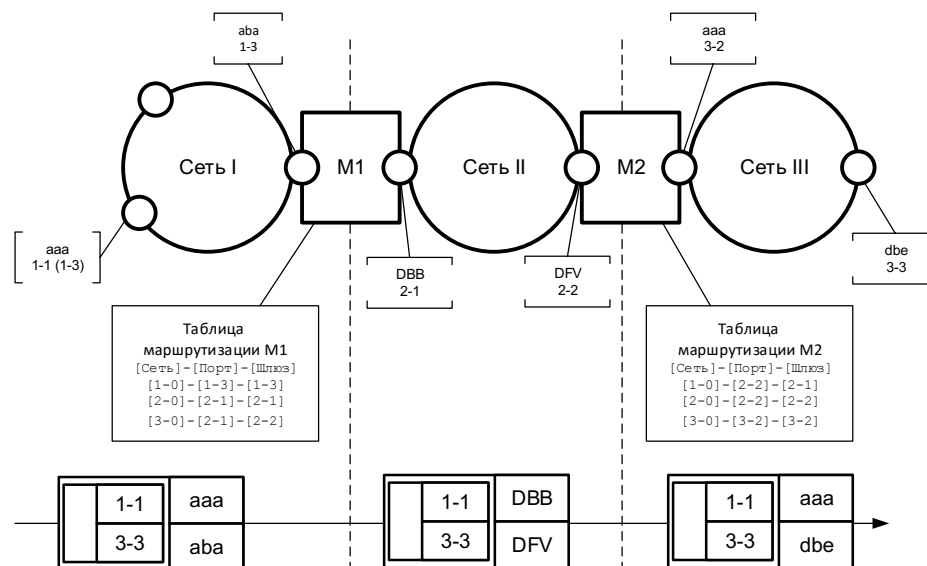


Рисунок 2

Опишем этапы передачи.

- Перед началом передачи сетевой уровень передающей стороны сформирует пакет с адресом отправителя 1-1 и адресом получателя 3-3. Оставим за рамками рассмотрения откуда узел 1-1 "узнал" сетевой адрес получателя. Обычно такие задачи решаются с помощью систем, подобных DNS. Перед инкапсуляцией сетевого пакета в кадр канального уровня сетевой уровень устанавливает, что адрес назначения лежит в другой локальной сети и передавать пакет надо через шлюз, указав его канальный адрес в поле адреса назначения кадра канального уровня.
- В конфигурации узла адрес шлюза (1-3) дан в виде сетевого адреса, поэтому узел 1-1 генерирует широковещательное сообщение на канальном уровне, адресованное на адрес "zzz" с запросом "у кого адрес 1-3?". Это сообщение получают все узлы сети 1, но отвечает на него только узел 1-3 со своего адреса канального уровня. Так узел 1-1 определяет канальный адрес назначения для первого шага.
- Сетевой пакет инкапсулируется в кадр канального уровня, где в поле адреса отправителя стоит "aaa", а в поле получателя – канальный адрес шлюза "aba". Этот кадр приходит на порт маршрутизатора-шлюза M1. Его канальный уровень принимает кадр для обработки, деинкапсулирует пакет сетевого уровня и передает его на свой сетевой уровень.
- Сетевой уровень решает задачу маршрутизации. Сначала определяется адрес сети назначения по адресу назначения в сетевом пакете (адрес сети 3-0). По таблице маршрутизации по адресу сети назначения определяется порт, через который надо передать пакет и сетевой адрес следующего шлюза.
- Сетевой пакет инкапсулируется в кадр канального уровня сети 2. При этом канальный адрес отправителя будет соответствовать адресу порта ("DBB"), а канальный адрес шлюза определяется по его сетевому адресу так же, как и в п. b.
- Сетевой пакет, инкапсулированный в новый кадр канального уровня, попадает на маршрутизатор M2. принимается им и обрабатывается так же, как в п. d и e. С той разницей, что M2 определяет, что он непосредственно подключен к сети с адресом 3-0 (сеть 3) и определяет канальный адрес получателя не для следующего шлюза, а для узла назначения 3-3. Сетевой пакет инкапсулируется в новый кадр канального уровня в сети 3 и отправляется уже на узел 3-3.
- Канальный уровень узла назначения принимает кадр, так как в адресе назначения стоит его адрес, деинкапсулирует пакет сетевого уровня и передает его выше по

стеку на сетевой уровень для обработки. Так как сетевой адрес назначения соответствует собственному адресу узла, то пакет принимается, деинкапсулируется вложенное сообщение и передается выше по стеку.

Обратите внимание, что в сетях 1 и 3 есть узлы с одинаковыми адресами канального уровня. Это возможно, так как область действия адресации канального уровня – локальная сеть. В составной сети адреса сетевого уровня из одной локальной сети должны иметь одинаковую сетевую часть. Это нужно для решения задачи маршрутизации. В составной сети адреса сетевого уровня должны быть уникальными. И, самое важное, за счет введения отдельных сетевого и канального уровней, а также процедуры инкапсуляции, межсетевое взаимодействие не зависит от природы канальных протоколов в локальных сетях и составная сеть может организовываться из локальных сетей с разными канальными протоколами.

9. Выводы

- a. В модели OSI были сформулированы основные принципы сетевой архитектуры как многоуровневой системы.
- b. Были введены понятия сетевого стека, уровня, интерфейса и протокола в их современном виде.
- c. Были выделены семь уровней со специфическими задачами. Названия этих уровней используются для описания современных сетевых протоколов с других стеков.
- d. Модель OSI подвергается справедливой критике, как избыточная и политически ангажированная, кроме того, попытки ее практической реализации не привели к успеху и от них быстро отказались.
- e. Однако ценность знания модели OSI для современного инженера состоит в понимании идеологии функционирования сетевого стека и владения специфической сетевой терминологией.

Основные термины

1. Сетевой стек – набор сетевых протоколов, работающих в определенной последовательности с целью передачи данных по компьютерной сети между двумя и более системами (возможен и частный случай взаимодействия различных приложений через сетевой стек в пределах одной системы).
2. Уровень – группа сетевых протоколов, решающих одну подгруппу задач в рамках общей задачи сетевого взаимодействия.
3. Протокол – соглашение об обработке данных и интерфейсах ввода-вывода, используемое для реализации модуля, решающего подзадачу сетевой коммуникации в рамках общей задачи сетевого взаимодействия.
4. Инкапсуляция – помещение сообщения вышестоящего уровня в поле данных сообщения нижестоящего уровня.
5. Деинкапсуляция – процесс, обратный инкапсуляции.
6. Заголовок – структура данных, которой модуль протокола снабжает сообщение. Информация в заголовке имеет смысл для аналогичного модуля протокола на принимающей стороне.
7. Сегмент – сообщение транспортного уровня, полученное в результате сегментации потока от сеансного уровня. В рамках одного потока все сегменты содержат последовательные фрагменты потока, из которых на принимающей стороне восстанавливается исходный поток.

8. Дейтаграмма – сообщение, не требующее подтверждения.
9. Кадр - сообщение канального уровня.
10. Локальная сеть – сеть, узлы которой взаимодействуют через один и тот же протокол канального уровня, без маршрутизации.
11. Составная сеть – сеть, полученная объединением локальных сетей посредством маршрутизации.
12. Маршрутизатор – программно-аппаратное устройство, передающее сообщения между локальными сетями на основании информации в заголовке сетевого уровня.
13. Аппаратный адрес – адрес устройства, используемый для адресации на канальном уровне.
14. Адрес сети – адрес сетевого уровня для группы узлов (относящихся к одной локальной сети).
15. Адрес узла – адрес сетевого уровня для одного узла.
16. Адрес шлюза – адрес сетевого уровня, указывающий на маршрутизатор, подключенный к той же локальной сети и используемый для выхода во внешнюю сеть.

Дополнительная литература

1. Олифер В., Олифер Н. Компьютерные сети. Принципы, технологии, протоколы. Юбилейное издание. СПб.: Питер, 2020 г.
2. Таненбаум Э., Уэзеролл Д. Компьютерные сети. СПб.: Питер, 2019 г.