Лекция 2. Протоколы Internet

Семиуровревая модель OSI ISO предполагает вертикальное взаимодействие уровней на основе вложения пакетов верхнего уровня в качестве данных в пакет последующего уровня. Кроме этого, реальная практика сложилась так, что три верхних уровня этой модели пока не реализовались в полной мере как самостоятельные программные решения. То есть, приложения сами занимаются и вопросами организации сеансов, и вопросами представления (шифрования, перекодировки, подбора подходящих параметров отображения видео или графическоой информации). Поэтому мы пока не будем подробно рассматривать эти вопросы.

Нижние уровни в большой степени завязаны на аппаратуру (физическую или сетевую), поэтому протоколы этих уровней также имеют достаточно технический характер.

А вот протоколы сетевого и транспортного уровня — это то с чем непосредственно сталкивается программист при реализации программ, выполняющих сетевое взаимодействие. Итак, далее кратко познакомимся с некоторыми протоколами этих жвух уровней.

Знакомство с протоколами.

Объем курса не позволяет много времени уделить на весь стек протоколов.

Помним про стек протоколов (вложенность пакетов) и горизонтальное и вертикальное взаимодействие.

Канальный уровень — Ethernet, WiFi, ARP, RARP — пропускаем.

Сетевой уровень. IP — Internet Protocol. (Есть и другие сетевые решения, кроме Internet, но они более специфичны и используются для специальных целей, поэтому тоже их не смотрим.)

Протокол решает два круга задач:

- передача пакетов от станции к станции в рамках глобальной сети
  (базовые средства для маршрутизации,
  фрагментация, сборка, обслуживание
  транспортного уровня)

- идентификация в рамках глобальной сети
  IP address.

IP сети (Internet Protocol)

В настоящее время существуют и работают версии 4 и 6.

Версия 4 отражает представления, сложившиеся около 40 лет тому назад. Версия 6 корректирует круг задач, решаемых протоколом, в соответствии с развитием сетей к настоящему времени.

IP v.4, адресация — 4 байта

точечная нотация 192.168.14.38

Все множество адресов делится на 5 классов.

класс	маска: сеть | станция	диапазон
A	`0.......|........ ........ ........`	`1.0.0.1 - 126.255.255.254`
B	`10...... ........|........ ........`	`128.1.0.1 - 191.254.255.254`
C	`110..... ........ ........|........`	`192.0.1.1 - 223.255.254.254`
D	`1110.... ........ ........ ........`	`224.0.0.0 - 239.255.255.255`
E	`11110... ........ ........ ........`	не используется

D — мультикастинг (групповые адреса для групп маршрутизаторов)

E — резерв

сеть 0...0 — данная сеть (сообшение распространяется только в своей локальной сети).

станция 1...1 — широковещательный (сообщение получают все станции).

127.0.0.1 — адрес обратной связи (сообщение не выходит наружу, но полностью обрабатывается сетевыми программами данной станции).

маска подсети `1111...111000...000` — расширет возможности назначения адресов сети и станций по сравнению с исходными классами A, B, C.

Задача протокола — доставить данные с учетом возможных отказов по пути прохождения пакета. Основные причины отказов: порча пакета в результате помех, отсутствие пути к получателю, отказ по MTU (maximal transfer unit — характеристика канального уровня, обычно 1.5–2.5 килобайта), истечение времени жизни павета и т.д. При возникновении различных отказов, маршрутизатор посылает отправителю информационное сообщение при помощи ICMP протокола.

Протокол 4 версии может разбивать (фрагментировать) пакет по пути его распространиния (чтобы влезть в MTU) и далее передавать исходный пакета серией пакетов, содержащих его отдельные части. Эти части собираются в единое целое уже на станции-получателе.

Формат пакета отражает в своей структуре информацию, которую протокол использует при решение описанных выще задач.

IP v.4 Формат пакета ⇒ принципы работы

битов	назначение	примечание
4	версия	=4
4	длина заголовка	=5 (в 32-битных словах)
8	тип сервиса	приоритеты/рекомендации маршрутизатору
16	полная длина	всего пакета: заголовок + данные
16	идентификатор	
3	флаги	0, more fragments, don't fragment
13	смещение фрагмента	в 8-байтовых словах
8	время жизни	=30
8	протокол	данных 1-ICMP 2-IGMP 4-IP 6-TCP 17-UDP
16	контрольная сумма	заголовка
32	SA	IP адрес отправителя
32	DA	IP адрес получателя
32	опции+заполнитель	реально не используются
—	данные	

Поясним некоторые поля.

— тип сервиса реально не используется;

— "уникальный" идентивикатор последовательно назначается каждому пакету, выпущенному отправителем, в случае разбиения пакета на части маршрутизатором, этот идентификатор наследуется всеми пакетами-частями.

— смещение фрагмента определяет в каком месте эта часть находилась в исходном пакете (разбивается на части кратно 8 байтам)

— флаги — more fragments = 1 — эта часть лежала в середине исходного пакета, more fragments = 0 — эта часть последняя в разбиении.

— don't fragment = 1 — пакет нельзя фрагментировать;

— время жизни — количество допустимых маршрутизацийЮ уменьшается на 1 при прохождении через каждый маршрутизатор, если стало 0, пакет уничтожается (и отсылается ICMP сообщение отправителю).

— протокол — какому протоколу отдать данные пакета для дальнейшей обработки.

— контрольная сумма — циклическая сумма 2-х байтовых слов для контроля возможных искажений пакета;

ICMP, Internet Control Message Protocol

битов	назначение	примечание
8	тип	
8	код	
16	контрольная сумма	всего пакета
...	содержимое IP сообщения	для контр. сообщ. [20+64] байта IP пакета

— тип, код соответствуют различным событиям или идентифицируют "команду" содержащуюся в сообщении.

Пример: PING — запрос/ответ с меткой времени Отправляется ICMP сообщение с меткой времени данной станции на другую станцию. Та станция, получив это сообщение, немедленно отправляет его обратно (просто переставив адреса отправителя и получателя местами). Первая станция полчает ответ и по текушему времени может понять сколько времени данное сообщение путешествовало туда и обратно — проверка загруженности сети и достижимости станции.

Другие сообщения и команды касаются:
проверки работоспособность сети
(адресат недостижим — сеть, станция, порт, протокол)
Необходимость переадресации
Объявление и запрос маршрутизатора
Ошибки в IP пакете, истекло время жизни, отказ по MTU и т.д.

IP v.6, адресация — 16 байтов

точечная нотация 19D7:FF39:2371:356B:44C3:1681:438A:56DE
:: для 0000:0000:0000, например 235F::1739:4DF2
::0 не специфицирован
::1 адрес обратной связи
::FFFF:ab:cd вложение v.4 в v.6 — a.b.c.d

возможная структуризация адреса (128 бит)
010 | registryID | providerId | subscriberID | subnetID | interfaceID

IP v.6 Формат пакета ⇒ принципы работы

битов	назначение	примечание
4	версия	=6
8	traffic class	“приоритет”
20	flow label	единый ID потока
16	payload length	длина данных
8	next header	какой пакет в данных
8	hop limit	max кол-во маршрутизаций
128	SA	IP адрес отправителя
128	DA	IP адрес получателя
—	данные	

+ новый ICMP (формат тот же, но события другие)

— flow label — этот идентивикатор устанавливается единым для серии пакетов, которые идут друг за другим в рамках “потоковой передачи”, маршрутизатор анализирует только первый пакет в серии, а все остальные передает точно так же, как предыдущие (не тратит время на их анализ); например, передача видео.

IP v.6 Цели и задачи

— длинные и структурированные адреса
— разгрузка маршрутизаторов
— все опции включаются в заголовки расширения *
— нет контрольных сумм (не надо считать и проверять)
— нет фрагментации на маршрутизаторах **

* next header = 0 — маршрутизация “по старому”

* next header = 43 — матрешка вложенных IP пакетов с заранее прописанным маршрутом

** next header = 44 — в данных выполнена предварительная фрагментация по типу v.4

Взаимодействие IP v.6 и IP v.4

Вложение 4 в 6
Тунелирование 6 в 4

Использование расширений IP v.4

— сети с локальными адресами 192.168.xxx.xxx
— тунели (Point to Point Tunnelling Protocol)
— виртуальные сети (VPN, Virtual Private Network)