

# Оглавление

<b>Предисловие от издательства .....</b>	<b>11</b>
<b>Отзывы .....</b>	<b>12</b>
<b>Аннотация ко второму изданию .....</b>	<b>14</b>
<b>Предисловие .....</b>	<b>15</b>
<b>Глава 1. Интернет .....</b>	<b>17</b>
1.1. Основные операции.....	17
1.1.1. Хосты, маршрутизаторы, каналы связи .....	17
1.1.2. Коммутация пакетов .....	18
1.1.3. Адресация.....	19
1.1.4. Маршрутизация .....	19
1.1.5. Обнаружение ошибок.....	21
1.1.6. Повторная передача ошибочных пакетов .....	22
1.1.7. Управление перегрузками.....	22
1.1.8. Управление потоком.....	23
1.2. DNS, HTTP и www .....	23
1.2.1. DNS.....	23
1.2.2. HTTP и www.....	23
1.3. Краткое изложение .....	24
1.4. Задачи .....	24
1.5. Ссылки .....	25
<b>Глава 2. Принципы .....</b>	<b>26</b>
2.1. Совместное использование .....	26
2.2. Метрики .....	27
2.2.1. Скорость канала связи.....	28
2.2.2. Широкополосный канал и его производительность .....	28
2.2.3. Задержка.....	29
2.2.4. Пропускная способность .....	30
2.2.5. Джиттер .....	32
2.2.6. Очередь M/M/1 .....	33
2.2.7. Закон Литтла.....	35
2.2.8. Fairness, или индексы справедливости в сети .....	37
2.3. Масштабируемость .....	38
2.3.1. Адресация на основе местоположения .....	38
2.3.2. Двухуровневая маршрутизация .....	39
2.3.3. Негарантированная доставка .....	40
2.3.4. Сквозной (end-to-end ) принцип и маршрутизаторы stateless .....	41
2.3.5. Иерархическое именование.....	42
2.4. Приложения и технологическая независимость .....	42
2.4.1. Уровни модели OSI .....	43
2.5. Топология приложений .....	44
2.5.1. Клиент/сервер.....	44
2.5.2. P2P .....	45

2.5.3. Облачные вычисления .....	45
2.5.4. Распределение контента .....	46
2.5.5. multicast/anycast .....	46
2.5.6. push/pull .....	46
2.5.7. Подбор контента по местоположению .....	47
2.6. Резюме .....	47
2.7. Задачи .....	47
2.8. Ссылки .....	51
<b>Глава 3. Ethernet .....</b>	<b>52</b>
3.1. Типовая установка .....	52
3.2. История Ethernet .....	52
3.2.1. Сеть Aloha (Alohanet) .....	53
3.2.2. Кабельная сеть Ethernet .....	54
3.2.3. Хаб Ethernet .....	56
3.2.4. Коммутируемый Ethernet .....	57
3.3. Адреса .....	57
3.4. Кадр .....	57
3.5. Физический уровень .....	58
3.6. Коммутируемый Ethernet .....	59
3.6.1. Пример .....	59
3.6.2. Обучение (learning) .....	59
3.6.3. Протокол spanning tree .....	60
3.7. Aloha .....	62
3.7.1. Версия с тайм-слотами .....	62
3.8. Aloha без слотов .....	63
3.9. Хаб Ethernet .....	63
3.9.1. Максимальное время обнаружения коллизии .....	63
3.10. Приложение: теория вероятности .....	65
3.10.1. Вероятность .....	65
3.10.2. Аддитивность эксклюзивных событий .....	65
3.10.3. Независимые события .....	66
3.10.4. Слотовая Aloha .....	67
3.10.5. Неслотовая Aloha .....	67
3.10.6. Ожидание успеха .....	69
3.10.7. Хаб Ethernet .....	69
3.11. Резюме .....	70
3.12. Задачи .....	70
3.13. Ссылки .....	73
<b>Глава 4. Wi-Fi .....</b>	<b>74</b>
4.1. Основные операции .....	74
4.2. Управление доступом к среде (MAC) .....	75
4.2.1. Протокол MAC .....	75
4.2.2. Усовершенствования для доступа к среде передачи .....	78
4.2.3. MAC-адреса .....	79
4.3. Физический уровень .....	80
4.4. Анализ эффективности протокола MAC .....	81

4.4.1. Одно устройство .....	81
4.4.2. Несколько устройств.....	82
4.5. Последние достижения.....	87
4.5.1. IEEE 802.11n – введение MIMO в Wi-Fi.....	87
4.5.2. IEEE 802.11ad – Wi-Fi в миллиметровом диапазоне .....	88
4.5.3. IEEE 802.11ac – введение MU-MIMO в Wi-Fi .....	88
4.5.4. IEEE 802.11ah – Wi-Fi для IoT и M2M .....	89
4.5.5. Одноранговая Wi-Fi.....	90
4.6. Приложение: цепи Маркова.....	91
4.7. Резюме.....	94
4.8. Задачи .....	95
4.9. Ссылки .....	97
<b>Глава 5. Маршрутизация .....</b>	<b>98</b>
5.1. Домены и двухуровневая маршрутизация.....	98
5.1.1. Масштабируемость .....	99
5.1.2. Транзит и пиринг.....	99
5.2. Междоменная маршрутизация .....	100
5.2.1. Алгоритм вектора пути .....	101
5.2.2. Возможные колебания.....	102
5.2.3. Мультивыходные дискриминаторы .....	103
5.3. Внутридоменная маршрутизация по кратчайшему пути.....	103
5.3.1. Алгоритм дейкстры и состояние связи .....	103
5.3.2. Алгоритм Беллмана–Форда и вектор расстояния .....	105
5.4. Anycast, multicast.....	107
5.4.1. Anycast .....	108
5.4.2. Multicast.....	108
5.4.3. Прямая коррекция ошибок (FEC) .....	109
5.4.4. Сетевое кодирование.....	111
5.5. Сети ad hoc.....	113
5.5.1. AODV.....	113
5.5.2. OLSR.....	114
5.5.3. Муравьиная маршрутизация .....	114
5.5.4. Географическая маршрутизация .....	114
5.5.5. Маршрутизация обратного давления .....	114
5.6. Резюме .....	114
5.7. Задачи .....	115
5.8. Ссылки .....	118
<b>Глава 6. Работа в сети интернет.....</b>	<b>119</b>
6.1. Цель.....	119
6.2. Основные компоненты: маска, шлюз, ARP .....	120
6.2.1. Адреса и подсети .....	121
6.2.2. Шлюз.....	121
6.2.3. DNS-сервер.....	122
6.2.4. ARP .....	122
6.2.5. Конфигурация .....	122
6.3. Примеры .....	122
6.3.1. Одна и та же подсеть.....	123

6.3.2. Разные подсети .....	123
6.3.3. Поиск IP-адресов .....	124
6.3.4. Фрагментация .....	124
6.4. DHCP .....	125
6.5. NAT .....	125
6.6. Резюме .....	127
6.7. Задачи .....	127
6.8. Ссылки .....	128
<b>Глава 7. Транспорт .....</b>	<b>129</b>
7.1. Транспортные услуги .....	129
7.2. Транспортный заголовок .....	130
7.3. Состояния TCP .....	131
7.4. Контроль ошибок .....	132
7.4.1. Stop-and-wait .....	132
7.4.2. Go Back N .....	133
7.4.3. Выборочные подтверждения .....	134
7.4.4. Таймеры .....	135
7.5. Управление перегрузками .....	135
7.5.1. AIMD .....	136
7.5.2. Усовершенствования: быстрая ретрансляция и быстрое восстановление .....	137
7.5.3. Регулировка производительности .....	138
7.5.4. Размер окна TCP .....	139
7.5.5. Терминология .....	140
7.6. Управление потоком .....	140
7.7. Альтернативные схемы управления перегрузками .....	141
7.8. Резюме .....	142
7.9. Задачи .....	143
7.10. Ссылки .....	148
<b>Глава 8. Модели .....</b>	<b>149</b>
8.1. Графы .....	149
8.1.1. Max-flow, min-cut .....	150
8.1.2. Раскраска графа и протоколы MAC .....	151
8.2. Очереди .....	153
8.2.1. Очередь M/M/1 .....	154
8.2.2. Сети Джексона .....	155
8.2.3. Очередь против коммуникационных сетей .....	156
8.3. Роль уровней .....	158
8.4. Управление перегрузками .....	159
8.4.1. Справедливость против производительности .....	159
8.4.2. Распределенное управление перегрузками .....	162
8.4.3. И снова о TCP .....	165
8.5. Динамические маршрутизация и управление перегрузками .....	167
8.6. Беспроводная связь .....	169
8.7. Приложение: обоснование теоремы двойственности линейного программирования .....	172

8.8. Резюме .....	173
8.9. Задачи .....	174
8.10. Ссылки .....	177
<b>Глава 9. LTE.....</b>	<b>179</b>
9.1. Сотовая сеть.....	179
9.2. Технологическая эволюция .....	182
9.3. Ключевые аспекты LTE .....	183
9.3.1. Архитектура системы LTE .....	185
9.3.2. Физический уровень.....	187
9.3.3. Поддержка требований QoS .....	190
9.3.4. Планировщик.....	190
9.4. LTE-Advanced .....	192
9.4.1. Агрегация несущих.....	193
9.4.2. Поддержка расширенного MIMO.....	193
9.4.3. Узлы ретрансляции (RN).....	193
9.4.4. Координированная многоточечная работа (CoMP).....	194
9.5. 5G.....	195
9.6. Резюме .....	196
9.7. Задачи .....	196
9.8. Ссылки .....	198
<b>Глава 10. QOS.....</b>	<b>199</b>
10.1. Обзор.....	199
10.2. Формирование трафика.....	200
10.2.1. Механизм leaky bucket.....	200
10.2.2. Границы задержки .....	201
10.3. Планирование .....	202
10.3.1. GPS .....	202
10.3.2. WFQ.....	204
10.4. Регулируемые потоки и WFQ .....	205
10.5. Сквозное QOS .....	206
10.6. Управление сквозным пропуском .....	207
10.7. Сетевой нейтралитет.....	207
10.8. Резюме .....	208
10.9. Задачи .....	208
10.10. Ссылки .....	210
<b>Глава 11. Физический уровень сети.....</b>	<b>211</b>
11.1. Как передавать биты? .....	211
11.2. Характеристики каналов связи .....	212
11.3. Проводные и беспроводные каналы связи .....	213
11.3.1. Схемы модуляции: BPSK, QPSK, QAM.....	213
11.3.2. Межсотовые помехи и OFDM .....	216
11.4. Оптические линии связи .....	218
11.4.1. Работа оптоволокна .....	218
11.4.2. Модуляция OOK .....	219
11.4.3. Мультиплексирование с разделением по длине волны.....	220

11.4.4. Оптическая коммутация .....	220
11.4.5. Пассивная оптическая сеть .....	221
11.5. Резюме .....	222
11.6. Ссылки .....	223
<b>Глава 12. Дополнительные темы .....</b>	<b>224</b>
12.1. Коммутаторы.....	224
12.1.1. Модульные коммутаторы.....	224
12.1.2. Матричные коммутаторы каналов.....	227
12.2. Оверлейные сети.....	229
12.2.1. Примеры сетей: CDN и P2P .....	231
12.2.2. Маршрутизация в оверлейных сетях.....	232
12.3. Как работают популярные протоколы P2P.....	232
12.3.1. Первое поколение: на базе сервер-клиент .....	232
12.3.2. Второе поколение: централизованный каталог .....	233
12.3.3. Третье поколение: полностью распределенный протокол.....	233
12.3.4. Появление иерархического оверлея – суперузлов .....	233
12.3.5. Продвинутый распределенный совместный доступ к файлам: BitTorrent.....	234
12.4. Сенсорные сети .....	235
12.4.1. Вопросы проектирования .....	235
12.5. Распределенные приложения .....	238
12.5.1. Алгоритм маршрутизации Беллмана–Форда .....	238
12.5.2. Регулировка мощности.....	239
12.6. Византийское соглашение.....	241
12.6.1. Соглашение при ненадежном канале связи .....	241
12.6.2. Консенсус в присутствии противников.....	242
12.7. Сжатие источника .....	244
12.8. SDN и NFV .....	244
12.8.1. Архитектура SDN .....	245
12.8.2. Новые услуги, предоставляемые SDN.....	246
12.8.4. Фреймворк управления для NFV .....	250
12.9. Интернет вещей (IoT) .....	252
12.9.1. Парадигмы удаленных вычислений и хранения данных .....	252
12.10. Резюме .....	253
12.11. Задачи .....	254
12.12. Ссылки .....	256
<b>Об авторах.....</b>	<b>258</b>
<b>Библиография.....</b>	<b>259</b>
<b>Предметный указатель .....</b>	<b>266</b>

# Предисловие от издательства

## Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге, – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв или оставить комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу [dmkpress@gmail.com](mailto:dmkpress@gmail.com); при этом укажите название книги в теме письма.

Если вы являетесь экспертом в какой-либо области и заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу [http://dmkpress.com/authors/publish\\_book/](http://dmkpress.com/authors/publish_book/) или напишите в издательство по адресу [dmkpress@gmail.com](mailto:dmkpress@gmail.com).

## Список опечаток

Хотя мы приняли все возможные меры для того, чтобы обеспечить высокое качество наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в основном тексте или программном коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от недопонимания и поможете нам улучшить последующие издания этой книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу [dmkpress@gmail.com](mailto:dmkpress@gmail.com), и мы исправим это в следующих тиражах.

## Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательства ДМК Пресс и Morgan & Claypool Publishers очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконной публикацией какой-либо из наших книг, пожалуйста, пришлите нам ссылку на интернет-ресурс, чтобы мы могли применить санкции.

Ссылку на подозрительные материалы можно прислать по адресу электронной почты [dmkpress@gmail.com](mailto:dmkpress@gmail.com).

Мы высоко ценим любую помощь по защите наших авторов, благодаря которой мы можем предоставлять вам качественные материалы.

# ОТЗЫВЫ

*Эта книга является желанным дополнением к литературе по коммуникационным сетям. В ней рассматривается с инновационной и современной точки зрения множество тем, отражающих быстрое развитие, которое претерпела эта область. Она информативна, познавательна и полезна для студентов, преподавателей и практиков, работающих в этой области. Эта книга просто необходима!*

Энтони Эфремидес, Мэрилендский университет

*Компьютерные сети являются чрезвычайно сложными системами, и учебники университетского уровня часто содержат пространные описания, которые жертвуют базовым концептуальным пониманием в пользу подробных операционных объяснений. Вальран и Парекс отходят от этого подхода, предлагая краткое и освежающее изложение, фокусирующееся на фундаментальных принципах, которые студенты могут (и должны) использовать. Издание обновлено с учетом последних достижений и представляет большую ценность для преподавания курса по компьютерным сетям.*

Массимо Франческетти, Калифорнийский университет, Сан-Диего

*В книге в четкой и ясной форме представлены наиболее важные принципы проектирования коммуникационных сетей с акцентом на интернет. Рассматриваются вопросы физического уровня, вплоть до ключевых распределенных приложений. Книга станет ценным пособием для студентов, преподавателей и практиков на долгие годы.*

Брюс Хайек, Иллинойский университет, Урбана-Шампейн

*Концептуальная ясность, простота объяснения и краткость – душа этой книги. Она охватывает очень широкий круг современных тем, разлагает сложные системы на их абсолютно базовые составляющие и объясняет каждую идею четко и лаконично. Она является образцом того, каким должен быть учебник. Жаль, что такой книги не было, когда я изучал коммуникационные сети.*

П. Р. Кумар, Техасский университет A&M

*Эта книга посвящена основным принципам, лежащим в основе проектирования и функционирования интернета. В ней дается целостное описание этой критически важной, но сложной инфраструктуры и четко и ясно объясняются основные идеи, не скрываясь за несущественными деталями реализации или аналитики. Это лучшее введение в сетевые технологии, на основе которого можно продолжить рассмотрение различных более специализированных тем.*

Стивен Лоу, Калифорнийский технологический институт (Caltech)



*«Коммуникационные сети: Краткое введение» Жана Вальрана и Шьяма Парекса – замечательная книга. Авторы имеют уникальный опыт благодаря фундаментальному вкладу, который они внесли в эту область в ходе многолетнего преподавания этого курса в Калифорнийском университете в Беркли. Книга охватывает множество важных тем – от архитектуры интернета до современных беспроводных технологий и таких новых тем, как SDN и IoT. Книга фокусируется на ключевых принципах и основных концепциях каждой проблемы, а также представляет краткое обсуждение того, насколько эти принципы необходимы для масштабируемых и надежных сетей связи. Математические инструменты, такие как цепи Маркова и теория графов, вводятся на уровне, довольно простом, но в то же время достаточном для моделирования и анализа ключевых компонентов коммуникационных сетей. Всесторонний охват основных понятий коммуникационных сетей и интуитивно-принципиальный подход делают эту книгу лучшим учебником для вводного курса по коммуникационным сетям для тех студентов, кто заинтересован в проведении исследований в этой области. Эта книга, безусловно, необходима для студентов и исследователей в данной области.*

Лей Инг, Университет штата Аризона)

# Аннотация

## ко второму изданию

Эта книга является результатом многолетнего преподавания курса по коммуникационным сетям на факультете EECS Калифорнийского университета в Беркли. Ее создание было продиктовано потребностью в легкодоступном учебнике, в котором основное внимание уделяется основным концепциям, лежащим в основе современных сетей следующего поколения связи. После краткого обзора того, как работает современный интернет, и обсуждения основных принципов его архитектуры, мы поговорим о ключевых идеях, лежащих в основе Ethernet, сетей WiFi, маршрутизации, интернетизации и TCP. Чтобы сделать книгу как можно более самодостаточной, в приложения включены краткое изложение вероятности и концепции цепей Маркова. Затем следует обсуждение математических моделей, которые дают представление о работе сетевых протоколов. Далее излагаются основные идеи нового поколения беспроводных сетей на основе LTE, а также понятие QoS. Также кратко обсуждаются технологии физического уровня, лежащие в основе различных сетей. Наконец, представлена выборка тем, которые могут оказать существенное влияние на будущую эволюцию сетей, включая оверлейные сети, такие как сети доставки контента и одноранговые сети, сенсорные сети, рассматриваются распределенные алгоритмы, задача византийских генералов<sup>\*1</sup> (сжатие исходного кода, SDN и NFV, а также Internet of Things – IoT (Интернет вещей).

### Ключевые слова

Интернет, Ethernet, WiFi, маршрутизация, алгоритм Беллмана–Форда, алгоритм Дейкстры, TCP, управление перегрузкой, управление потоком передачи, QoS, LTE, одноранговые сети, SDN, NFV, IoT

---

<sup>1</sup> В вычислительной технике и криптологии задача взаимодействия нескольких удаленных абонентов, которые получили приказы из одного центра в мысленный эксперимент, призванный проиллюстрировать проблему синхронизации состояния систем в случае, когда коммуникации считаются надежными, а процессоры – нет. – *Прим. ред.*

# Предисловие

Эти конспекты лекций основаны на курсе по коммуникационным сетям для старших курсов, который авторы читали на факультете электротехники и вычислительной техники Калифорнийского университета в Беркли.

За 30 лет, что мы преподаем этот курс, сети прошли путь от раннего Agranet и экспериментальных версий Ethernet до глобального интернета с широкополосным беспроводным доступом и новыми приложениями – от социальных до сенсорных сетей.

За эти годы мы пользовались множеством учебников. Цель этой книги – быть ближе к фактическому материалу, который мы излагаем. В рамках курса, рассчитанного на один семестр, невозможно охватить 800-страничную книгу. Поэтому в преподаваемом нами курсе и в этих заметках мы сосредоточились на ключевых принципах, которые, по нашему мнению, студенты должны понять.

Мы хотели бы показать, как лес, так и деревья. Сетевые технологии продолжают развиваться. Нашим студентам не придется заново изобретать TCP/IP. Им нужно концептуальное понимание, чтобы продолжать создавать будущее.

Помимо исправления найденных ошибок и добавления некоторых разъяснений, основные изменения во втором издании следующие. *Глава 4*, посвященная Wi-Fi, была обновлена с учетом последних достижений. *Глава 7*, посвященная транспортным протоколам, включает обсуждение альтернативных схем борьбы с заторами в сети. *Глава 8* о моделях была расширена разделами, посвященными графикам и очередям. Кроме того, в этой главе теперь объясняется формулировка TCP и совместного использования беспроводной связи для оптимизации сетей. *В главе 9*, посвященной LTE, теперь обсуждаются основы сотовых сетей и более подробное излагается ряд ключевых аспектов LTE. Она также включает презентации LTE-Advanced и 5G. Обсуждение WiMAX было удалено в свете подавляющего признания LTE. *В главе 12* «Дополнительные темы» мы добавили следующие разделы: «Коммутаторы», «SDN и NFV», а также «IoT».

У нас есть соответствующий веб-сайт <https://bit.ly/2zPXD13>, который мы планируем использовать для будущих обновлений материала этих лекций. В последние годы наш курс в Беркли также включает исследовательский проект, в рамках которого студенты применяют фундаментальные концепции из курса к широкому спектру тем, связанных с сетевыми технологиями. Заинтересованные читатели могут найти расширенные аннотации этих исследовательских проектов на сайте.

Многие коллеги по очереди преподают курс в Беркли. Такая ротация позволяет сохранить свежесть и широту охвата материала. С удовольствием мы отмечаем важный вклад в представленный здесь материал Кевина Фолла, Рэнди Катца, Стива МакКанна, Абхая Парекса, Верна Паксон, Сильвии Ратнасами, Скотта Шенкер, Иона Стоика, Дэвида Тсе и Адама Волич. Мы также благодарим многочисленных ассистентов, помогавших нам на протяжении многих лет, и любознательных студентов Беркли, которые всегда держат нас в напряжении.

Мы благодарны рецензентам ранних черновиков этого материала. В частности, Ассане Гуйе, Либин Цзян, Дживун Ли, Стивену Лоу, Джону Мусаккио, Дженнифер Рексфорд и Нихил Шетти, которые предоставили полезные конструктивные комментарии. Мы благодарим Майка Моргана из Morgan & Claurool за его поддержку и помощь в рецензировании и публикации этого текста.

Жан Вальран получил частичную поддержку от NSF и грант MURI от ARO во время написания этой книги. Шьям Парекс благодарен своим коллегам и руководству AT&T Labs Research за их поддержку. В частности, он благодарен Гагану Чоудхури, Мазину Гилберту, Кэролин Джонсон, Кэти Майер-Хеллстерн и Крису Райсу.

Самое главное, как всегда, – мы глубоко признательны нашим семьям за их неизменную поддержку.

*Жан Вальран и Шьям Парекс  
Октябрь, 2017 г.*

# Глава 1

## Интернет

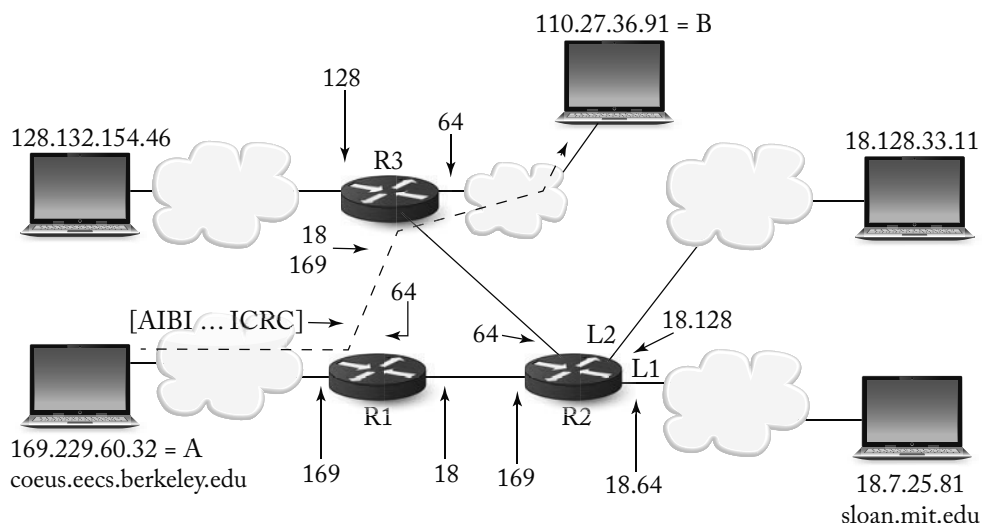
В конце 1960-х гг. интернет вырос из небольшого эксперимента в сеть, которая объединяет около 3,7 млрд пользователей (данные за июнь 2017 г.) и которая стала главной коммуникационной системой современного общества. Этот феноменальный рост коренится в архитектуре интернета, которая делает его масштабируемым, гибким и расширяемым, а также обеспечивает замечательную экономию информационных ресурсов. В этой главе мы объясним, как работает интернет.

### 1.1. Основные операции

Интернет доставляет информацию, сначала превращая и упорядочивая ее в пакеты. В этом разделе рассказывается, как пакеты попадают в пункт назначения, как сеть исправляет ошибки и управляет перегрузкой.

#### 1.1.1. Хосты, маршрутизаторы, каналы связи

Интернет состоит из хостов и маршрутизаторов, соединенных друг с другом каналами связи. Хосты являются источниками или получателями информации. Название «хосты» указывает на то, что на этих устройствах размещаются (host) приложения, генерирующие или использующие информацию, которой они обмениваются по сети. К хостам относятся компьютеры, принтеры, серверы, веб-камеры и т. д. *Маршрутизаторы* получают информацию и пересылают ее следующему маршрутизатору или хосту в сети. Канал связи передает биты между двумя маршрутизаторами или хостами. Каналы бывают оптическими, проводными (из разных видов кабеля) или беспроводными. Некоторые сетевые соединения являются более сложными и включают в себя коммутаторы, которые мы изучим позже. На рис. 1.1 показано несколько хостов и маршрутизаторов, соединенных каналами связи. Облачные сети представляют собой другие наборы маршрутизаторов и соединений.



**Рис. 1.1.** Хосты, маршрутизаторы и каналы связи. Каждый хост имеет отдельный 32-битный IP-адрес, основанный на его местоположении. Заголовок пакета содержит адреса источника и назначения, а также *контрольную сумму ошибки*. Маршрутизаторы ведут таблицы маршрутизации, в которых указан выход для самого длинного префикса, совпадающего с адресом назначения

### 1.1.2. Коммутация пакетов

Первоначальной мотивацией для создания интернета было создание сети, которая была бы устойчива к атакам на некоторые из ее частей. Первоначальная идея заключалась в том, что, если часть сети будет выведена из строя, маршрутизаторы будут автоматически перенаправлять информацию по альтернативным путям. Такая гибкая маршрутизация основана на механизме *коммутации пакетов*. Суть коммутации пакетов состоит в том, что сеть передает биты, сгруппированные в *пакеты*. Пакет – это строка битов, расположенных в соответствии с заданным форматом. Интернет-пакет содержит адреса источника и назначения. На рис. 1.1 показан пакет с адресом источника *A* и адресом назначения *B*. Коммутация заключается в выборе каналов, по которым пакет следует от своего источника к месту назначения. Коммутация пакетов означает, что маршрутизаторы делают этот выбор индивидуально для каждого пакета. В отличие от этого в телефонной сети используется *коммутация каналов*, когда выбор каналов производится только один раз для всего телефонного разговора, и на время разговора необходимая скорость передачи данных на выбранном канале резервируется. Существует также промежуточное решение, использующее концепцию *виртуальных цепей*, где выбор каналов для передачи информации также производится один раз, как

и при коммутации каналов, но требуемая скорость передачи данных в этих каналах не резервируется. Виртуальные цепи также передают данные с помощью пакетов, но отправляют пакеты данного соединения по одному и тому же набору каналов. Для того чтобы отличить коммутацию пакетов от коммутации на основе виртуальных цепей часто используется термин «*датаграммы*» для версии коммутации пакетов, когда все маршрутизаторы выбирают следующий канал связи индивидуально для каждого пакета. *Многопротокольная коммутация по меткам* (Multi-Protocol Label Switching, MPLS)<sup>1</sup> и *асинхронный режим передачи* (Asynchronous Transfer Mode, ATM) являются примерами технологий на основе виртуальных каналов.

### 1.1.3. Адресация

В четвертой версии интернет-протокола (называемой IPv4) каждый компьютер или другой узел, подключенный к интернету, имеет уникальный адрес, указанный в 32-битной строке, называемой его *IP-адресом*, что означает «адрес протокола интернета». Адреса условно записываются в виде a.b.c.d, где a, b, c, d – десятичные значения четырех байтов. Например, 169.229.60.32 соответствует четырем байтам 10101001.11100101.00111100.00100000. Более современная версия протоколов, называемая IPv6, использует 128-битные адреса, но совместима с IPv4.

### 1.1.4. Маршрутизация

Каждый маршрутизатор определяет из адреса назначения, куда должен отправиться пакет в следующий раз (следующий хоп или транзитный участок сети). При продвижении к месту назначения в сети, находящейся под контролем общего администратора, пакеты в основном следуют по кратчайшему пути<sup>2</sup>. Маршрутизаторы регулярно вычисляют эти кратчайшие пути и записывают их в *таблицы маршрутизации*<sup>3</sup>. Таблица маршрутизации определяет следующий переход (или прыжок) для каждого адреса назначения, как показано на рис. 1.2.

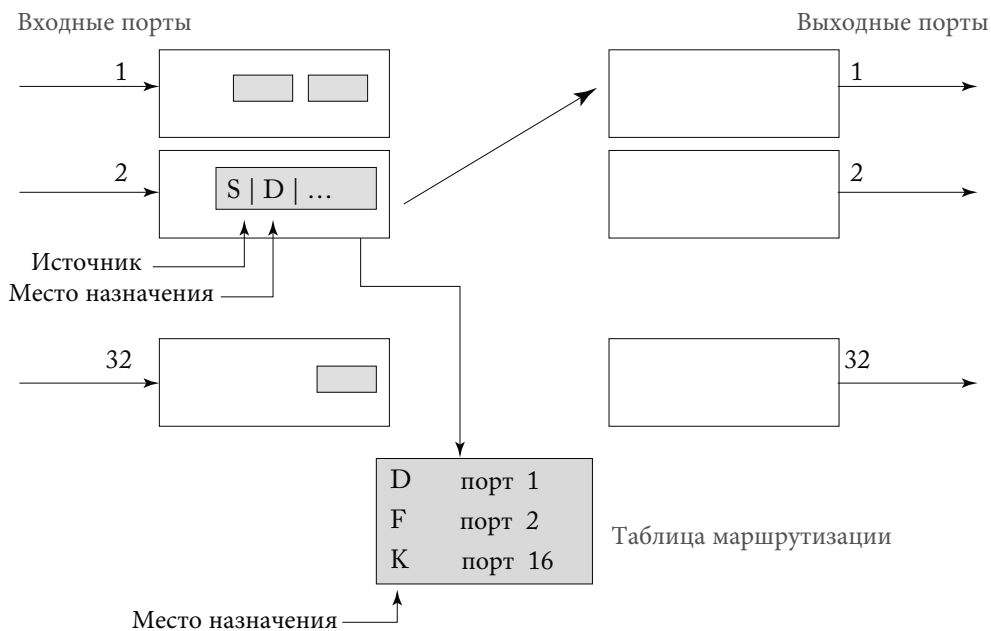
Чтобы упростить таблицы маршрутизации, администраторы сети назначают IP-адреса хостам, основываясь на их местоположении. Например, маршрутизатор R1 на рис. 1.1 отправляет все пакеты с адресом

<sup>1</sup> В сети, основанной на MPLS, пакетам данных присваиваются метки. Решение о дальнейшей передаче пакета данным другому узлу сети осуществляется только на основании значения присвоенной метки без необходимости изучения самого пакета данных. За счет этого возможно создание сквозного виртуального канала, независимого от среды передачи и использующего любой протокол передачи данных. – *Прим. ред.*

<sup>2</sup> Пакеты обычно проходят через набор сетей, принадлежащих различным организациям. Маршрутизаторы выбирают этот набор в соответствии с правилами, которые мы обсудим в главе «Маршрутизация».

<sup>3</sup> Точнее говоря, маршрутизатор обращается к таблице пересылки, в которой указан выходной порт пакета. Однако это различие не является существенным.

назначения, первый байт которого имеет десятичное значение 18, на маршрутизатор R2, а все пакеты с адресом назначения, первый байт которого имеет десятичное значение 64, – на маршрутизатор R3. Вместо того чтобы иметь одну запись для каждого возможного адреса назначения, маршрутизатор имеет одну запись для набора адресов с общей начальной битовой строкой (*префиксом*). Если бы можно было назначить адреса так, чтобы все пункты назначения с одинаковыми начальными 5 битами были доступны из одного и того же порта 32-портового маршрутизатора, то в таблице маршрутизации потребовалось бы только 32 записи по 5 бит: каждая запись указывала бы начальные 5 бит, соответствующие каждому порту. На практике присвоение адресов не является идеальным, но тем не менее оно значительно уменьшает размер таблиц маршрутизации. Это очень похоже на организацию телефонных номеров [код страны, код области, зона, номер]. Например, номер 1 510 642 1529 соответствует телефонному аппарату в США (1), в Беркли (510), зона кампуса Беркли (642).



**Рис. 1.2.** На рисунке изображен маршрутизатор с 32 входными портами (присоединенными каналами) и 32 выходными портами. Пакеты содержат адреса источника и места назначения. Таблица маршрутизации определяет для каждого пакета выходной порт, соответствующий адресу назначения

Общий подход к использованию адресации на основе местоположения состоит в том, чтобы найти самую длинную общую начальную строку битов (*префикс*) в адресах, которые получают посредством од-



ного и того же следующего хопа. Эта схема, называемая бесклассовой междоменной маршрутизацией (Classless Inter-Domain Routing, CIDR), позволяет нам гибко распределять адреса по подгруппам, определяемым префиксами. Основное различие со схемой телефонной нумерации заключается в том, что в CIDR длина префикса не задана заранее, что обеспечивает большую гибкость.

В качестве иллюстрации *маршрутизации с наибольшим совпадением префиксов* рассмотрим рис. 1.1, на котором показано, как маршрутизатор R2 выбирает, куда отправлять пакеты. Адрес назначения, начинающийся с битов 000100101, соответствует первым 9 битам префикса 18.128 = 00010010'10000000 выходного канала L2, но только первым 8 битам префикса 18,64 = 00010010'01000000 выходного канала L1. Следовательно, пакет с адресом назначения 18.128.33.11 покидает R2 по каналу L2. Аналогично пакет с адресом назначения 18.7.25.81 покидает R2 по каналу L1. Подводя итог, маршрутизатор в своей таблице маршрутизации находит префикс, который имеет самое длинное совпадение с адресом назначения 18.7.25.81. Этот префикс определяет выходной порт пакета.

### 1.1.5. Обнаружение ошибок

Каждый узел отправляет биты пакета следующему узлу, предварительно преобразуя их в электрические или оптические сигналы. Принимающий узел преобразует сигналы обратно в биты. В этом процессе возможны ошибки, вызванные случайными флуктуациями сигналов. Таким образом, иногда случается, что некоторые биты в пакете повреждаются, что соответствует *ошибке передачи*.

Простая схема обнаружения ошибок заключается в том, что источник добавляет один бит, называемый *битом четности*, в пакет, чтобы число единиц было четным. Например, если пакет равен 00100101, то отправляющий узел добавляет *бит четности*<sup>4\*</sup>, равный 1, чтобы пакет стал 001001011 и имел четное число единиц. Если получатель получает пакет с нечетным числом единиц, скажем, 001101011, он знает, что произошла ошибка передачи. Однако эта схема обнаружения ошибок не может определить, были ли изменены два или любое четное количество битов во время передачи. Вот почему в интернете используется более надежный способ обнаружения ошибок в заголовках пакетов – *контрольная сумма заголовка*.

При использовании контрольной суммы заголовка отправляющий узел вычисляет контрольную сумму (обычно длиной в 16 бит) других полей в заголовке. Принимающий узел выполняет то же вычисление и сравнивает результат с контрольной суммой; если они отличаются,

<sup>4</sup> Обычно размещается в 7-м бите пакета. – Прим. ред.

принимающий узел знает, что в заголовке пакета произошла какая-то ошибка, и он отбрасывает поврежденный пакет<sup>5</sup>.

### 1.1.6. Повторная передача ошибочных пакетов

В дополнение к удалению пакетов, заголовок которых поврежден из-за ошибок передачи, маршрутизатор может отбрасывать поступающие пакеты, когда у него заканчивается память, для временного хранения их перед пересылкой. Это может происходить, когда пакеты поступают на маршрутизатор быстрее, чем он может переслать их дальше. Такие потери пакетов происходят не из-за ошибок передачи, а из-за перегрузки.

Для обеспечения надежной доставки источник и пункт назначения используют механизм, гарантирующий, что источник повторно передает пакеты, которые не достигают места назначения без ошибок. Такая схема называется *запросом на автоматическую повторную передачу* (automatic retransmission request, ARQ). Основная идея этого механизма заключается в том, что получатель подтверждает все правильные пакеты, которые он получает, а источник повторно передает пакеты, которые получатель не подтвердил в течение определенного промежутка времени. Допустим, источник посылает пакеты 1, 2, 3, 4, и в пункте назначения не получен пакет 2. Через некоторое время источник замечает, что получатель не подтвердил пакет 2, и повторно посылает копию этого пакета. Мы обсудим конкретную реализацию этого механизма в интернете в главе, посвященной транспортному протоколу. Обратите внимание, что повторную передачу организуют хосты источника и назначения, а не маршрутизаторы.

### 1.1.7. Управление перегрузками

Представьте себе, что множество хостов отправляют пакеты, которые проходят через общий сетевой канал. Если хосты посылают пакеты слишком быстро, канал не может обработать их все, и маршрутизатор этого исходящего канала должен отбросить некоторые пакеты. Если хосты не получают подтверждения, то они замедляются для предотвращения чрезмерного количества отброшенных пакетов. То есть когда хост должен повторно передать пакет, подтверждение которого не пришло, он предполагает, что потеря произошла из-за перегрузки, и замедляет скорость отправки пакетов.

В конце концов перегрузка в сети спадает, и потери прекращаются. До тех пор, пока узлы получают подтверждения своевременно, они медленно увеличивают скорость передачи пакетов, приближаясь к макси-

<sup>5</sup> В главе 7 «Транспорт» мы показываем, что пакет может также содержать контрольную сумму всего пакета, и пункт назначения сравнивает ее с расчетной, чтобы убедиться, что он не игнорирует ошибки в остальной части пакета.

мальной, которая может быть поддержана существующими условиями сети. Эта схема, называемая *управлением перегрузкой*, автоматически регулирует передачу пакетов таким образом, чтобы сетевые каналы использовались бы полностью, ограничивая потери при передаче.

### 1.1.8. Управление потоком

Если устройство очень быстро посылает пакеты более медленному устройству, последнее может быть перегружено. Чтобы предотвратить это явление, принимающее устройство в каждом подтверждении, которое оно посылает обратно источнику, указывает количество свободного буферного пространства, имеющегося для приема дополнительных битов. Если это свободное пространство меньше количества битов, уже отправленных источником, а приемник еще их получение не подтвердил – передача прекращается.

Источник объединяет *схему управления потоком* со схемой управления перегрузкой, рассмотренной ранее. Обратите внимание, что управление потоком предотвращает переполнение буфера устройства получателя, в то время как управление перегрузкой предотвращает переполнение буфера сетевого маршрутизатора.

## 1.2. DNS, HTTP и www

### 1.2.1. DNS

Хосты, подключенные к интернету, помимо IP-адреса, имеют *имя*, ведь имена легче запомнить (например, google.com). Чтобы отправить пакеты на хост, источник должен знать IP-адрес этого узла. В интернете существует автоматическая система каталогов, называемая *Системой доменных имен* (Domain Name), или DNS, которая преобразует имя в IP-адрес. DNS – это распределенная система каталогов соответствия IP-адресов и имен хостов. Интернет разделен на *зоны*, и в каждой зоне особый *сервер DNS* поддерживает адреса хостов. Например, факультет EECS в Беркли обслуживает сервер каталога хостов в зоне сети eecs.berkeley.edu. Сервер DNS для этой зоны отвечает на запросы об IP-адресах узлов этой зоны. Следовательно, если добавить хост в сеть нашего факультета, необходимо обновить только этот DNS-сервер.

### 1.2.2. HTTP и www

*Всемирная паутина* организована как коллекция ресурсов с гиперссылками, таких как веб-страницы, видео и музыкальные файлы. Ресурсы идентифицируются с помощью *унифицированного указателя ресурсов*, или URL; он определяет компьютер и файл на этом компьютере вместе с протоколом, который должен доставить файл.

Например, URL <http://www.eecs.berkeley.edu/~wlr.html> указывает домашнюю страницу в компьютере с именем *www.eecs.berkeley.edu* и протоколом HTTP.

HTTP (*Hyper Text Transfer Protocol*), *протокол передачи гипертекста*, определяет правила запроса/ответа для передачи файла от сервера клиенту. По сути, протокол устанавливает соединение между сервером и клиентом, затем запрашивает определенный файл и, наконец, разрывает соединение, когда передача данных завершена.

### 1.3. Краткое изложение

- Интернет состоит из хостов, которые отправляют и/или получают информацию, маршрутизаторов и каналов связи.
- Каждый хост имеет 32-разрядный IP-адрес (в IPv4; 128-разрядный в IPv6) и имя. DNS – это система распределенных каталогов, которая преобразует имя в IP-адрес.
- Хосты упорядочивают информацию в пакеты, представляющие собой группы битов заданного формата. Пакет включает в себя адреса источника и получателя, а также биты обнаружения ошибок.
- Маршрутизаторы вычисляют самые короткие пути (по сути) к местам назначения и хранят их в таблицах маршрутизации. Чтобы уменьшить размер этих таблиц, IP-адреса основаны на местоположении хостов с использованием самого длинного совпадения префиксов.
- Источник управляет объемами своих передач в сеть, чтобы избежать переполнения буфера назначения (управление потоком) и буферов маршрутизаторов (управление перегрузкой).
- Хосты устраняют потери при передаче и перегрузке с помощью подтверждений, тайм-аутов и повторных передач.

### 1.4. Задачи

- P1.1. Сколько хостов может быть в интернете, если каждому из них нужен отдельный IPv4-адрес?
- P1.2. Если адреса были распределены произвольно, сколько записей должно быть в таблице маршрутизации?
- P1.3. Представьте, что все маршрутизаторы имеют 16 портов. При наилучшем распределении адресов какой размер таблицы маршрутизации требуется для каждого маршрутизатора?
- P1.4. Предположим, что хост *A* в Беркли отправляет поток пакетов хосту *B* в Бостоне. Предположим также, что все каналы работа-

ют со скоростью 100 Мбит/с и что требуется 130 мс, чтобы вернулось подтверждение после отправки первого пакета. Допустим,  $A$  отправляет один пакет объемом 1 Кбайт, а затем ожидает подтверждения перед отправкой следующего пакета и т. д. Каков долгосрочный средний битрейт (Long-Term Average Bitrate) соединения? Предположим теперь, что  $A$  отправляет  $N$  пакетов, прежде чем дождется первого подтверждения, и что  $A$  отправляет следующий пакет каждый раз, когда получает подтверждение. Выразите долгосрочный средний битрейт соединения в зависимости от  $N$ .

*Примечание:* 1 Мбит/с =  $10^6$  бит в секунду;  
1 мс = 1 миллисекунда =  $10^{-3}$  с.

- P1.5. Предположим, что хост  $A$  в Беркли отправляет пакеты размером 1 Кбайт со скоростью передачи 100 Мбит/с хосту  $B$  в Бостоне. Однако  $B$  считывает биты только со скоростью 1 Мбит/с. Предположим также, что устройство в Бостоне использует буфер, который может хранить 10 пакетов. Изучите механизм управления потоком и предоставьте график передач.

## 1.5. Ссылки

Коммутация пакетов была независимо изобретена в начале 1960-х гг. Полом Бараном [14], а также Дональдом Дэвисом и Леонардом Клейнроком, которые в своей диссертации MIT [56] отметили, что сети с коммутацией пакетов можно анализировать с помощью теории массового обслуживания. Боб Кан и Винт Серф изобрели базовую структуру TCP/IP в 1973 г. [52]. Ваном Джейкобсоном (Jacobson), анализом Чиу и Джейна [25] и [26], а также стабильностью линейных систем было исправлено в 1988 г. управление перегрузкой в TCP [49]. Пол Мокапетрис изобрел DNS в 1983 г. CIDR описан в [34]. Тим Бернерс-Ли изобрел WWW в 1989 г. Смотрите [40] для обсуждения автономных систем.