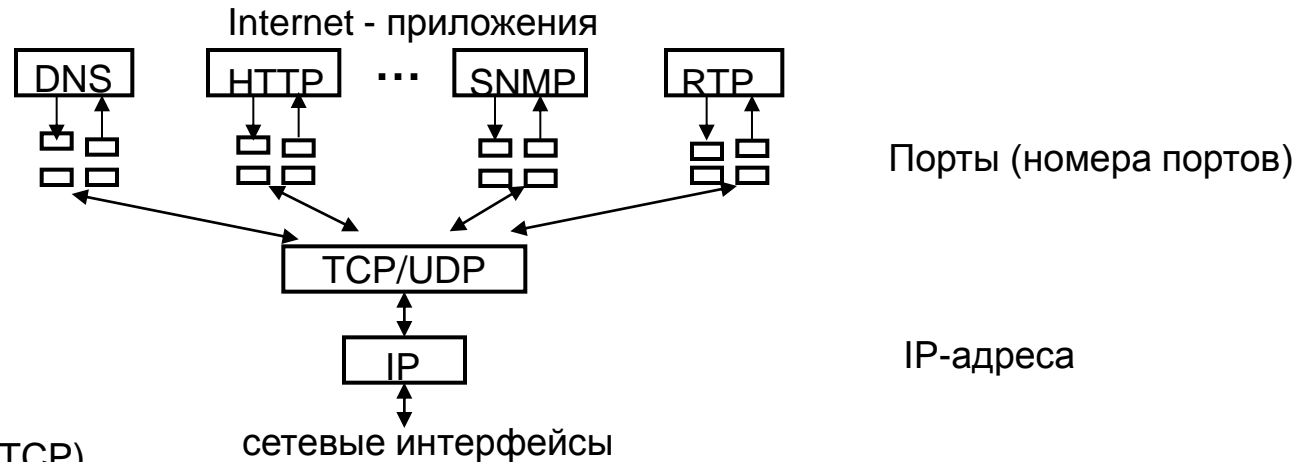


Часть 9. Транспортный уровень Internet

1. Услуги транспортного уровня Internet

Ключевая роль транспортного уровня Internet в протокольном стеке сети – изоляция приложений от деталей организации, технологий и несовершенства лежащих под ними уровня интернета (IP-уровня) и сетевых интерфейсов (базовых глобальных и локальных сетей).



Transport Control Protocol (TCP)

Развитый многофункциональный протокол с обеспечением соединения для формирования надежного дуплексного сквозного потока байтов по ненадежной интерсети: данные доставляются в последовательности, без искажений и ошибок.

User Datagram Protocol (UDP)

Простой протокол без установления соединения

IP-дейтаграмма в заголовке (тип протокола) обозначает данные – TCP-сегмент или UDP-дейтаграмма.

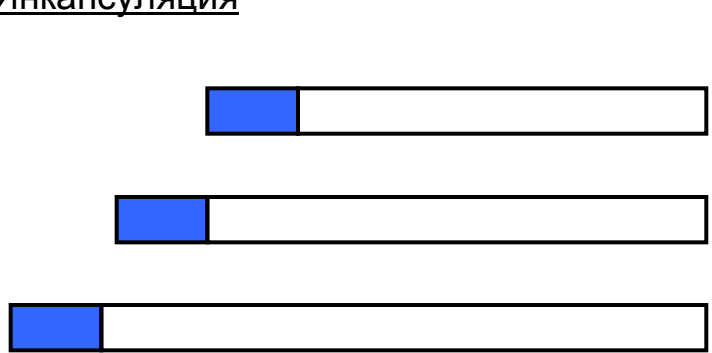
Порты

Порт, связывающий TCP/UDP с прикладным процессом, реализуется как дуплексный, двухточечный интерфейс с двумя очередями (буферами) – приема и передачи, размер буферов UDP ограничен.

TCP и UDP- порты имеют независимую нумерацию: хорошо известные (well known) номера задаются стандартами Internet (например, эл.почта -25б, HTTP –8), динамические номера выделяются ОС по запросу приложений, они уникальны в пределах компьютера.

Проблема возникает, когда: две копии одного приложения со стандартными номерами портов содержатся в одном компьютере. Поэтому каждой копии присваивается свой IP-адрес, что позволяет создавать уникальные точки доступа к услугам (SAP), сокеты, в виде пары <IP-адрес, номер порта>.

Инкапсуляция



■ - заголовки
 UDP – дейтаграмма
 TCP-сегмент

IP – дейтаграмма с тегами инкапсуляции в поле протокола заголовка: 17 (UDP), 6(TCP); поле данных – до 65515 байтов

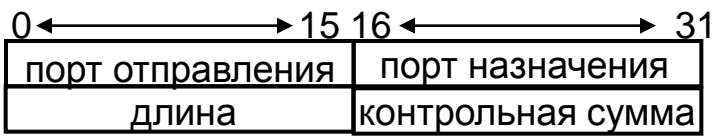
кадр базовой сети, типично Ethernet

2. User Datagram Protocol

Транспорт сетевых приложений, не представляющих высоких требований к качеству услуг (надежности пересылки трафика), например, запрос-ответ при взаимодействии с базой данных или DNS-сервером.

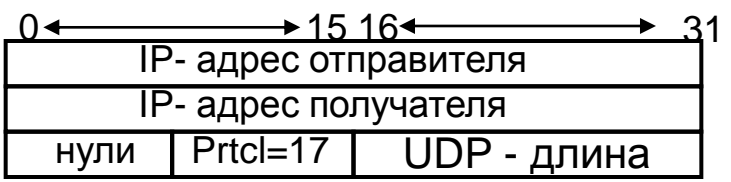
Зарезервированы номера стандартных портов для сетевых услуг тестирования, отладки и измерений, реализуемых с использованием UDP, например, Echo, Character Generator (ответ на любое сообщение, включающее определенное число байтов), Daytime и других.

Заголовок UDP



длина – суммарная длина заголовка и данных в байтах

Контрольная сумма

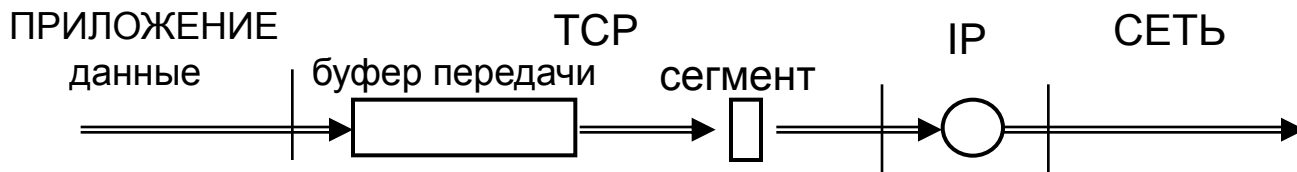


Контрольная сумма формируется алгоритмом обработки данных псевдо-заголовка IP, который включает поле длины из заголовка UDP и адресные поля IP- дейтаграммы, в которую он инкапсулирован.

Псевдо-заголовок IP

3. Протокол TCP

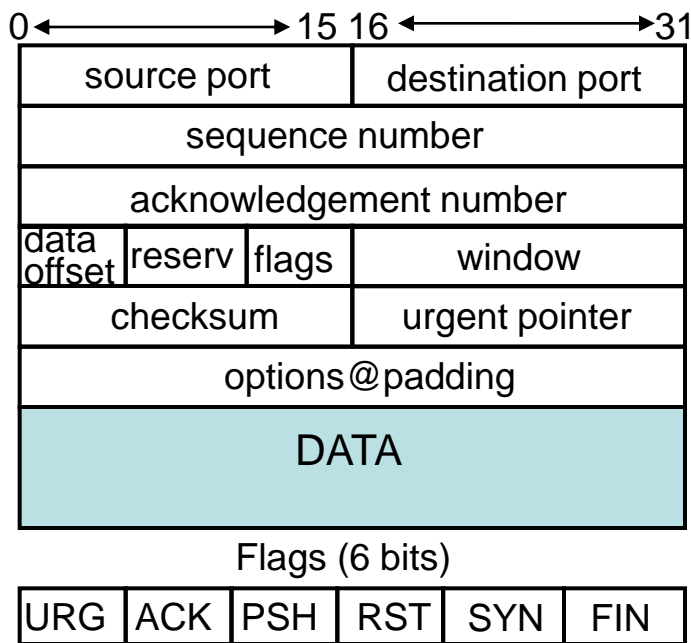
3.1 Этапы и режимы пересылки данных



Режимы формирования сегментов:

- TCP ждет, чтобы заполнение буфера данными способствовало эффективному использованию сети
- Push – данные пересылаются сразу после поступления в буфер (напр., конец диалоговой команды)
- Urgent – срочная информация (например, новый log-in)

3.2 Сегмент TCP



- sequence number (SEC) – номер первого октета в сегменте
- acknowledge number (ACK) – номер следующего ожидаемого октета
- window – окно приема
- urgent pointer – указатель на начало срочных данных в сегменте
- data offset – длина заголовка в 32-битовых словах (4 бита)
- options - может содержать макс. размер сегмента, который инициатор хочет получить
- padding – заполнение между заголовком и данными
- URG, PSH – режимы формирования сегмента
- SYN – бит синхронизации дуплексного потока сегментов
- ACK – бит подтверждения
- RST – завершение соединения (в ответном сообщении)
- FIN – разрыв соединения

3.2 Механизмы TCP

3.2.1 Процедуры сетевого соединения на транспортном уровне

Проблемы надежности

Гарантия доставки управляющего TCP-сегмента при открытии соединения и разъединении

Эта доставка гарантируется подтверждением, получение которого партнером также не гарантировано. Проблема решается выбором практически достаточной последовательности подтверждений, сопровождающей посылку одного управляющего сегмента. В TCP принят принцип «тройного рукопожатия (three-way-handshake)».

Потеря или неприемлемая задержка управляющего сегмента

Для отправителя определен тайм-аут, в течение которого должно поступить подтверждение. По истечении тайм-аута (например, при перегрузке в сети) управляющий сегмент считается потерянным и отправляется дубликат, или процедура завершается.

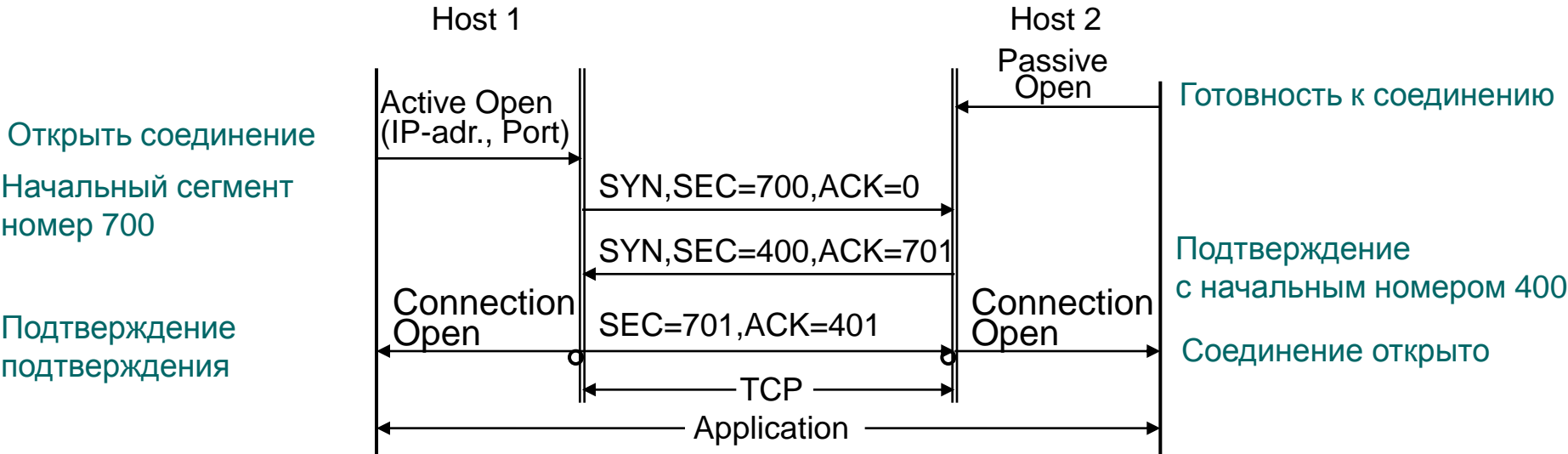
Поступление старых дубликатов

Старые дубликаты опознаются по заданной в них нумерации сегментов, которая не соответствует текущей нумерации в хосте, принявшем дубликат.

Нумерация сегментов

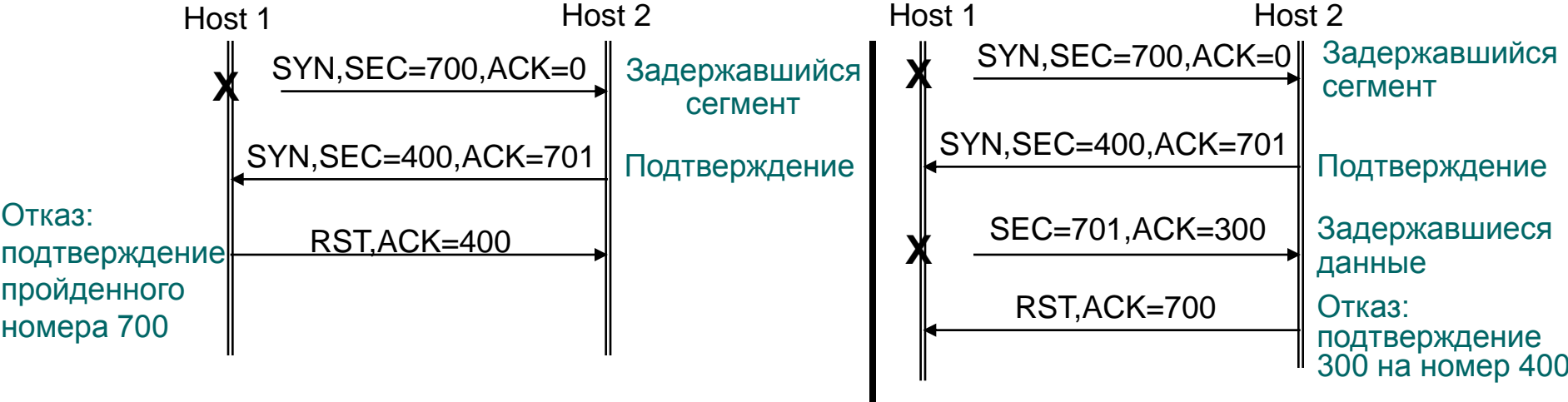
Номер задается внутренним счетчиком, продолжающим работу при отключении хоста.

Нормальное открытие соединения

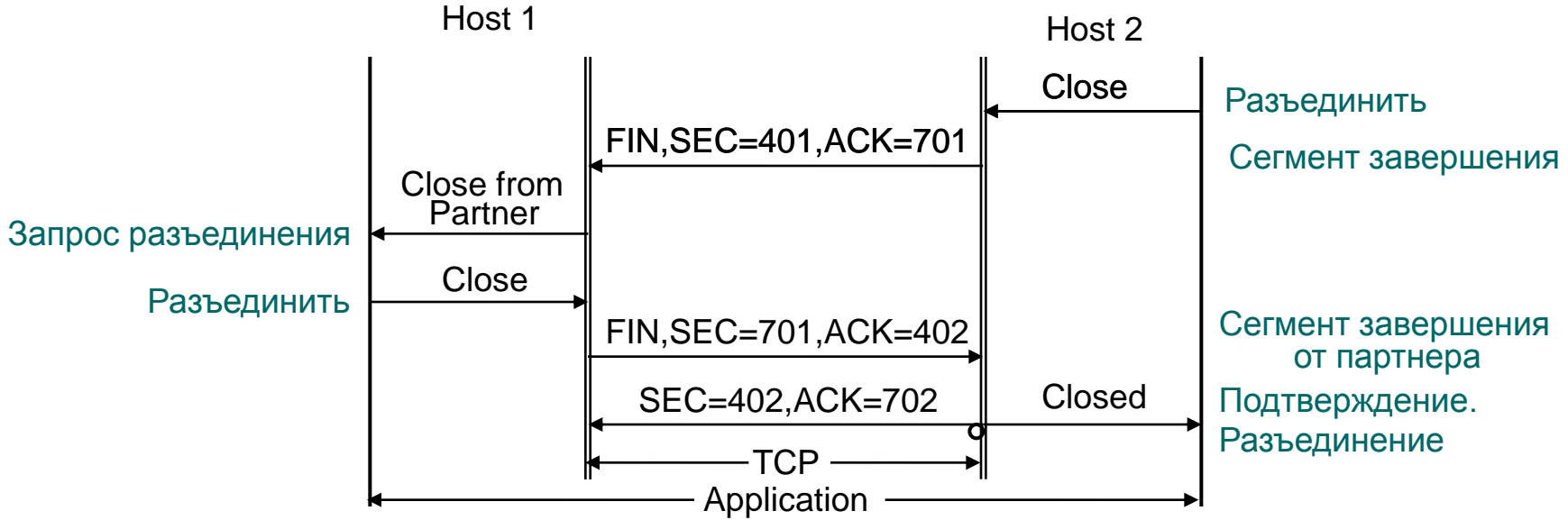


Ликвидация устаревших заявок на открытие

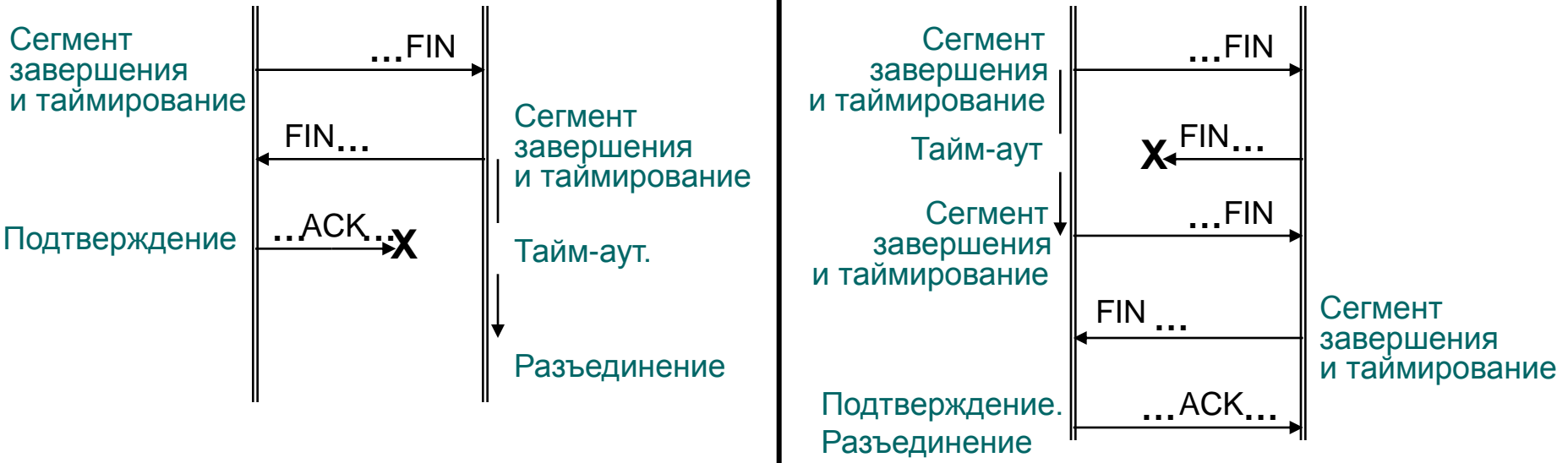
При перегрузке посланные по тайм-аутам и задержавшиеся в сети начальные сегменты могут спровоцировать несанкционированное открытие.



Нормальное разъединение (three-way handshake)



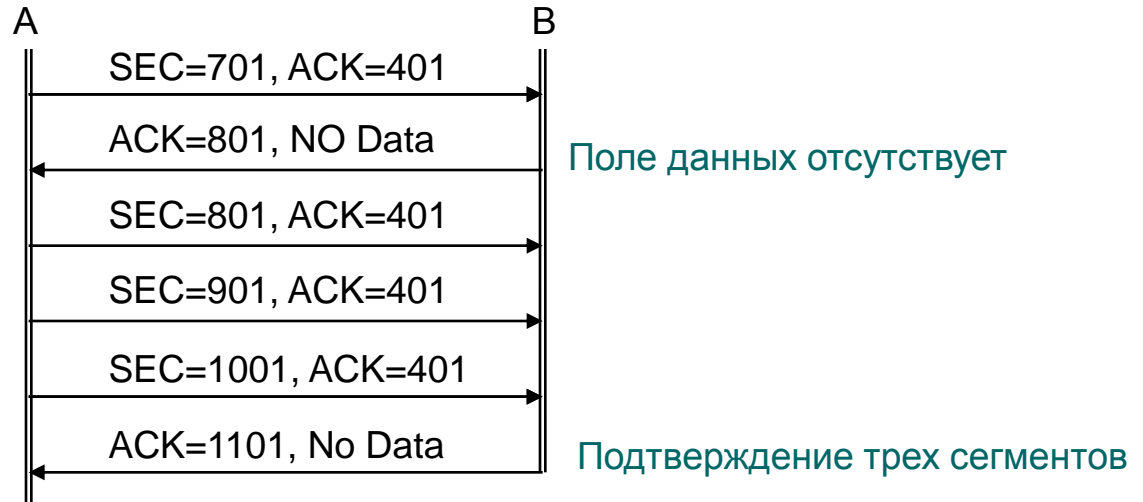
Разъединение по тайм-ауту



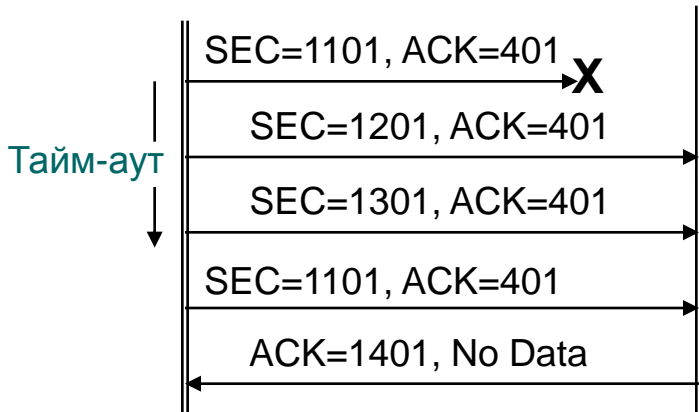
3.2.2 Передача данных

Предшествовало нормальное открытие соединения с начальными номерами 700 (от А к В) , 400 (от В к А).
Передача сегментов по 100 байтов.

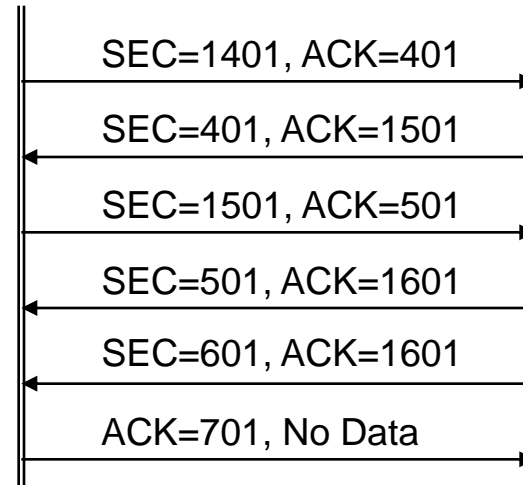
Односторонняя передача



Потеря и повторная передача данных



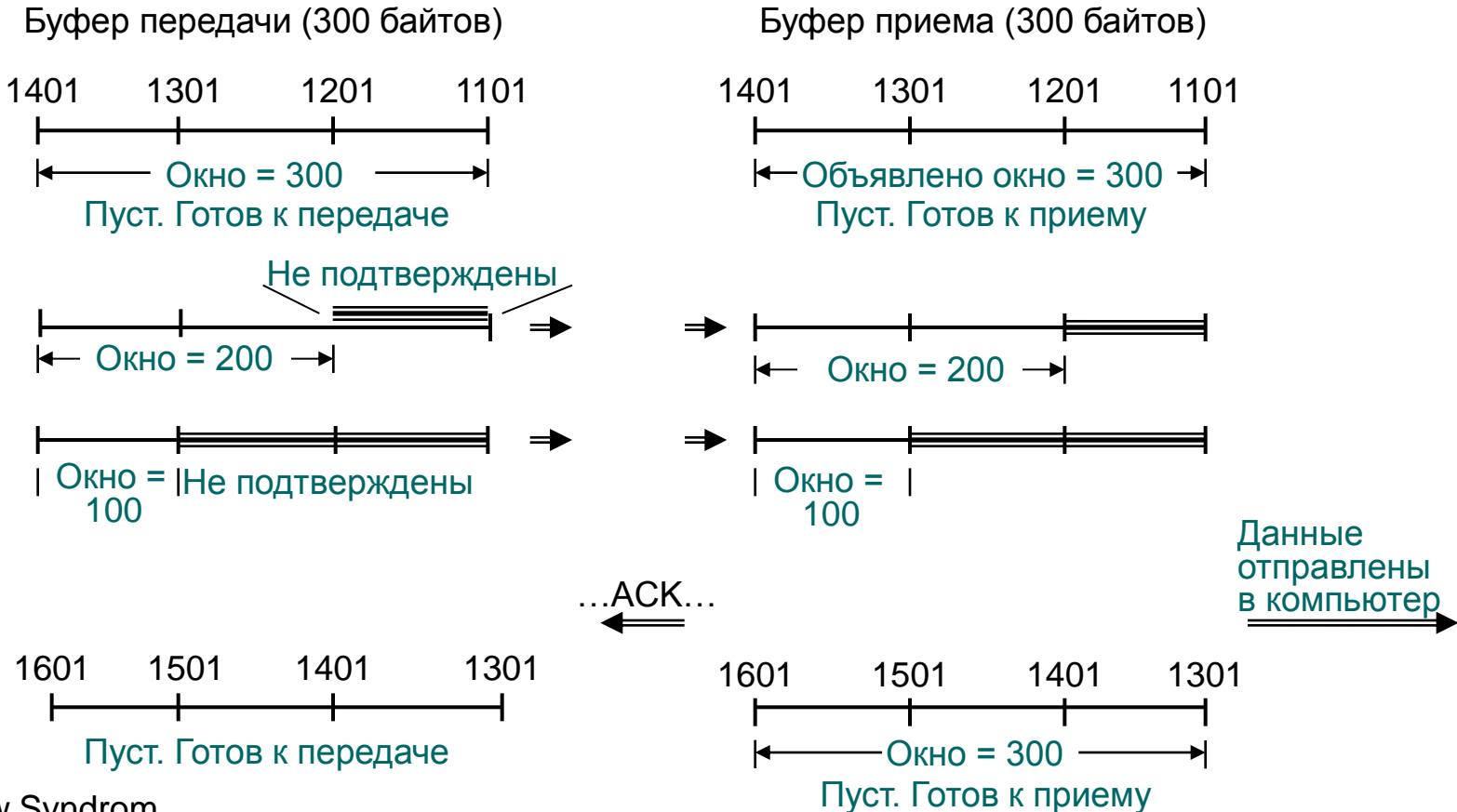
Двусторонняя передача данных



3.2.3 Управление потоком: окно приема и окно передачи

Окно приема – длина (в байтах) части буфера, не занятой данными.

Окно передачи – длина части буфера между последним посланным, но еще не подтвержденным, байтом и концом текущего окна передачи.



Silly Window Syndrom

Приложение удаляет из буфера приема малые порции данных, приемник объявляет малые окна, передатчик посылает малые сегменты, плохо используя соединение. Рекомендуется: не освобождать приемный буфер малыми порциями – надо выбирать минимально $1/2$ Receive Buffer Size, максимально Maximum Segment Size, то-есть приемник должен «обманывать» - не посылать передатчику малые окна, пока не накопится нужное количество данных для удаления из буфера и произойдет их пересылка в компьютер.

3.2.4 Тайминг

Основные параметры

Сглаженное суммарное время передачи TCP-сегмента и подтверждения (Smoothed Round Trip Time, SRTT)

Определяется на основании регулярных измерений интервала времени между отправлением и возвращением сообщения *echo*.

$$\text{New SRTT} = (1-\alpha) \times \text{Old SRTT} + \alpha \times (\text{Latest Round Trip Time}).$$

где обычно берется $\alpha = 1/8$, а начальный $\text{SRTT} = 0$.

Девияция SRTT (DEV)

$$\text{DEV} = | \text{Latest Round Trip Time} - \text{Old SRTT} |$$

Сглаженная девиация (SDEV)

$$\text{New SDEV} = 3/4 \times \text{Old SDEV} + 1/4 \times \text{Latest DEV}$$

Тайм-аут передачи

$$\text{TIMEOUT} = \text{SRTT} + 2 \times \text{SDEV}$$

Алгоритм Карна

Если SRTT возрастает (свидетельство вероятной перегрузки), то значение тайм-аута пересчитывается

$$\text{New TIMEOUT} = \text{Factor (typically 2)} \times \text{Old TIMEOUT}$$

- Тайм-аут увеличивается до определенного предела, затем оставаясь неизменным.
- Количество повторных передач минимизируется. Если лимит исчерпан, соединение ликвидируется.
- Как только подтверждения начинают прибывать до истечения тайм-аута, выполняется возвращение к нормальному тайм-ауту.

Окно перегрузки (Congestion Window)

Увеличение SRTT помогает не всегда. Congestion Window – окно, ограничивающее число передач.

В нормальном режиме $CW = \text{Send Buffer}$, при перегрузке $CW = \text{maximum (1/2 current CW, single segm. size)}$, при повторном пересчете уменьшает размер буфера до одного сегмента.

Обратный путь: с получением одного ACK размер CW увеличивается на один сегмент.