Учреждение образования

Белорусский государственный университет

информатики и радиоэлектроники

Кафедра электронных вычислительных машин

Лабораторная работа №6

**МЕТОДЫ ВЗАИМОДЕЙСТВИЯ В ЗВЕНЕ СЕТИ ПЕРЕДАЧИ ДАННЫХ**

Выполнил: Проверил:

ст. гр. №750502 Глоба А. А.

Альховик Д. Г.

Минск, 2019

**1**. **Виртуальный канал.**

При формировании полномасштабной сети, то есть при объединении разрозненных физических сегментов в СПД той или иной сложности, возникает ряд специфических задач, направленных на оптимизацию взаимодействия между абонентами.

Упомянутые задачи решают на третьем и четвертом уровнях модели OSI.

Новые задачи обусловлены серьезными отличиями процессов передачи-приема пакета в пределах сегмента и между сегментами. Основные отличия заключаются в необходимости ретрансляций пакетов, а так же в возможном наличии альтернативных путей.

Одним из ключевых терминов транспортного уровня является термин соединение (connection). По сути дела, понятие соединения связано с понятием готовности. Если абоненты находятся в состоянии «нормальной готовности» передавать или принимать данные, то считают, что между ними установлено соединение. С учетом абстрагирования от более низких уровней модели OSI и инкапсуляции, соединение может быть выражено неявно.

Нужно отличать виртуальные соединения (virtual connections) от физических соединений (physical connections). Абоненты-программы физически (явно) соединены быть не могут. Следовательно, применительно к ним, соединения являются сугубо виртуальными.

Следует также учитывать, что нормальная готовность может рассматриваться в двух ракурсах:

1. Организация взаимодействия абонентов-программ.

2. Настройка задействованного промежуточного оборудования.

В первом случае речь идет о собственно виртуальных соединениях транспортного уровня, во втором -- о виртуальных цепях (virtual circuits) сетевого или канального уровней.

В свою очередь, виртуальные цепи бывают:

1. PVCs (Permanent Virtual Circuits) -- выделенные виртуальные цепи.

2. SVCs (Switched Virtual Circuits) -- коммутируемые виртуальные цепи (в

отечественной литературе иногда называют виртуальными вызовами).

Термин виртуальный канал (virtual channel) может в равной степени подходить как к виртуальным соединениям, так и к виртуальным цепям.

При разговоре о соединениях невозможно обойти стороной вопрос о надежности.

Существуют два способа организации взаимодействия:

1. Без гарантированной доставки -- в СПД предпринимаются определенные усилия по доставке пакетов, но при этом ничего не гарантируется (при необходимости, соответствующий контроль возлагается на программы-абоненты).

2. С гарантированной доставкой -- алгоритм работы транспортной службы гарантирует доставку пакетов (программы-абоненты могут не контролировать наличие и очередность пакетов). Однако, соединение без гарантированной доставки практического смысла не имеет.

Поэтому наличие соединения как правило говорит о надежности.

В общем случае, контроль передачи информации посредством СПД предотвращает не только потерю пакетов, но и искажение их содержимого. Отсутствие соединения не означает, что защита от сбойных пакетов отсутствует.

Простейшим подходом к обеспечению контроля доставки информационных пакетов является применение метода, который обобщенно можно назвать методом запросов- подтверждений (requests/acknowledges). Метод предполагает некоторое

разнообразие и заключается в том, что вводят специальные служебные пакеты двух типов. Пакет-запрос используется при получении права принять или передать полезные данные, а также собственно при запросе данных. Пакет-подтверждение (в отечественной литературе часто называют квитанцией) передается в ответ на пакет- запрос или после приема полезных данных.

**2**. **TCP.**

Классической реализацией оконного метода является оконный механизм протокола транспортного уровня TCP (Transmission Control Protocol) (основное RFC -- RFC 793). Протокол обеспечивает установление надежного соединения между сугубо пользовательскими или другими видами приложений, то есть доставка данных вправильном порядке гарантируется.

В стандарте TCP описано динамическое скользящее окно.

TCP соответствует клиент-серверной модели.

Сокет (socket) -- это «привязка» к виртуальному каналу, соединяющему между собой два взаимодействующих сетевых процесса, с точки зрения одного (любого) из этих процессов, причем с учетом всех трех уровней адресации.



Рисунок -- Структура системы TCP

Применительно к каждому TCP-соединению нужно выделять приложение, производящее или потребляющее сетевые данные, и TCP-процесс, предоставляющий коммуникационные услуги (например, специальный драйвер ОС).

Синхронизировать работу приложения и TCP-процесса можно только с помощью буферизации. TCP-интерфейс, которым пользуется приложение, состоит из примитивов для работы с буфером, позволяющих контролируя записывать или

считывать данные. Доступ к буферу имеет и TCP-процесс, который отслеживает наполнение буфера и, используя ресурсы более низких уровней, организует прием или передачу данных.

Предназначенное для передачи сообщение разбивается на сегменты.

Минимальной учитываемой в окне единицей данных является октет, то есть байт. Все байты сообщения последовательно нумеруются так называемыми последовательными номерами -- SNs (Sequence Numbers). Нумерация начинается с некоторого начального последовательного номера -- ISN (Initial Sequence Number), который как правило не равен нулю, а генерируется реализациями случайно (например, на основе текущего времени) для того чтобы лучше управлять соединениями (например, после их ненормальных завершений). Принято, что сам ISN в нумерацию байтов не включается, то есть номер первого байта сообщения больше ISN на единицу.

Номером сегмента является SN первого байта данных в нем. По разным понятным причинам длина сегмента может варьировать, но она имеет ограничение. Поэтому важное значение имеет конфигурационный параметр MSS (Maximum Segment Size) -- максимальная длина сегмента (по умолчанию 536 байтов).



Рисунок -- Пример сегментации TCP-сообщения



Рисунок -- Формат заголовка TCP-сегмента

Поля:

1. Source Port -- программный порт источника.

2. Destination Port -- программный порт назначения.

3. Sequence Number (SN) -- последовательный номер (сегмента).

4. Acknowledgment Number (AN) -- подтверждающий номер.

5. Data Offset -- смещение данных (в 32-ухбитных словах).

6. Reserved -- зарезервировано (должно равняться нулю).

7. URG (URGent Pointer field significant) -- флаг значимости указателя на экстренные данные.

8. ACK (ACKnowledgment field significant) -- флаг значимости подтверждающего номера.

9. NS (Nonce Sum) -- флаг -- контрольная сумма для проверки правильности кодов явных уведомлений о заторах (связан с QoS, связан с IP-заголовком) (RFC 3540).

10. CWR (Congestion Window Reduced) -- флаг уменьшения окна затора при явном уведомлении о заторе (RFC 3168).

11. ECE (Explicit Congestion Notification Echo) -- флаг подтверждения явного уведомления о заторе (RFC 3168).

12. PSH (PuSH Function) -- флаг принудительной доставки данных (без буферизации).

13. RST (ReSeT the connection) -- флаг разрыва соединения (например, из-за сбоя на одной из взаимодействующих сторон).

14. SYN (SYNchronize sequence numbers) -- флаг синхронизации последовательных номеров.

15. FIN (No more data from sender) -- флаг последних данных.

16. Window (W) -- предлагаемое окно.

17. Checksum -- контрольная сумма.

18. Urgent Pointer -- указатель на экстренные данные (RFC 6093).

19. Options -- опции (например, MSS).

20. Padding -- наполнитель.

**Задание.**

Передать от сервера клиенту 154 Б при MSS = 33 Байт. Показать работу fast retransmit.

