

# Лабораторная работа 1

## Тема: Создание клиент-серверного приложения на базе транспортного протокола SCTP.

### Задание:

Требуется разработать клиент-серверные программы передачи данных на базе транспортного протокола SCTP [1,2]. Обеспечить передачу информации по нескольким потокам в одном соединении или ассоциации (особенность протокола SCTP).

1. Клиентская программа посылает в одном соединении или ассоциации: текстовое сообщение в первом потоке; файл с картинкой во втором потоке.
2. Серверная программа принимает в одном соединении или ассоциации от каждого из клиентов: текстовое сообщение с первого потока и выводит на экран; со второго потока принимает данные и сохраняет в файл.
3. В серверной программе реализовать (на выбор) параллельную или псевдопараллельную обработку клиентов.
4. Реализация - на языке C/C++, консольные приложения.

Серверная программа должна находить номер свободного порта и выводить его на экран. При запуске клиентской программы задавать со строки IP адрес сервера и порт (см. , например, [3], стр. 338-342, функции bind, getsockname).

### Полезные ссылки:

1. Протокол SCTP <http://rfc.com.ru/rfc2960.htm>
2. Стивенс У.Р., Феннер Б., Рудофф Э. М. UNIX: разработка сетевых приложений. - 3-е изд. - СПб. : ПИТЕР, 2007. - 1038с.
3. Фейт С. TCP/IP: Архитектура, протоколы, реализация (включая IP версии 6 и IP Security). – М.: Лори, 2000. – 424 с.

# Лабораторная работа 2

## Тема: Разработка сетевой программы – «снифер».

### Задание:

Используя библиотеку PCAP написать сетевую программу «снифер». По заданному правилу (регулярное выражение в стиле bpf (Berkley Packet Filter) обеспечить фильтрацию пакетов в сети. В выбранном пакете найти определенную информацию. Реализация - на языке C/C++, консольное приложение.

### Например:

Фильтровать все пакеты по протоколу TCP на порт 21. Определить IP адрес отправителя и получателя пакета или найти пакет содержащий слово USER.

Фильтр задавать либо при запуске программы (со строки) либо в диалоговом режиме.

## Лабораторная работа 3

**Тема:** Разработка клиент-серверной программы на основе сетевой библиотеки Boost.Asio.

### Задание:

Используя сетевую библиотеку Boost.Asio разработать программы асинхронного сервера, клиента на протоколе TCP [1]. Реализация - на языке C/C++, консольные приложения.

**Замечание.** Серверная программа должна находить номер свободного порта и выводить его на экран. При запуске клиентской программы задавать со строки IP адрес сервера и порт (см. , например, [2], стр. 338-342, функции bind, getsockname).

### Полезные ссылки:

1. Boost.Asio <http://habrahabr.ru/post/192284/>
2. Фейт С. TCP/IP: Архитектура, протоколы, реализация (включая IP версии 6 и IP Security). – М.: Лори, 2000. – 424 с.

### Вариант первый.

Клиент отправляет текстовое сообщение. В ответе сервер возвращает то же самое преобразованное сообщение, в котором каждое слово «перевернуто» относительно исходного, но сохранен порядок их расположения.

### Вариант второй.

При запуске клиента пользователь задает число  $i$  от 1 до 10. Клиент передает серверу в цикле это число с задержкой в  $i$  секунд между передачей. Сервер возвращает клиенту квадрат числа  $i$ .

Например:

1-й клиент посылает число 1 в цикле с задержкой в 1 сек.

2-ой клиент посылает число 2 с задержкой в 2 сек.

3-й клиент посылает число 3 в цикле с задержкой в 3 сек.

Сервер отображает информацию полученную от клиентов. Если у Вас правильно организован асинхронный ввод/вывод, то на экран со стороны сервера будет выводиться с чередованием числа 1, 2 и 3. Причем частота появления определенного числа будет зависеть от задержки по времени его передачи.

## Лабораторная работа 4

**Тема:** Разработка параллельных программ.

**Задание:**

Используя библиотеку MPI написать параллельную программу перемножения матрицы на вектор.

Исследовать эффективность распараллеливания на кластерной вычислительной системе - рассчитать коэффициент ускорения вычислений параллельной программы в зависимости от числа элементарных машин (вычислительных ядер) системы.

**Полезные ссылки:**

MPI

[http://parallel.ru/tech/tech\\_dev/mpi.html](http://parallel.ru/tech/tech_dev/mpi.html)

Центр Параллельных Вычислительных Технологий (ЦПВТ) СибГУТИ

<http://cpct.sibsutis.ru/index.php/Main/HomePage>

Руководство пользователя вычислительного кластера Jet ЦПВТ

<http://cpct.sibsutis.ru/uploads/Main/jet-cluster-howto>

Вычислительные ресурсы ЦПВТ

<http://cpct.sibsutis.ru/index.php/Main/Resources>