

Цель работы. Освоение набора системных вызовов для создания сокетных соединений различных типов, для обмена данными на хостах и по сети.

Последовательность выполнения работы:

1. Проанализируйте набор системных вызовов для серверной и клиентской сторон при организации соединений на сокетах под ОС Linux, принимая во внимание возможности различных видов сокетов и семейств адресации.

2. Скомпилируйте и выполните программу `socketpair.cpp`, иллюстрирующую создание простейшего вида сокета и обмен данными двух родственных процессов.

Проанализируйте вывод на консоль. Существует ли зависимость обмена от различных соотношений величин временных задержек (в вызовах `sleep()`) в процессе-родителе и в процессе-потомке?

3. Скомпилируйте программы `echo_server.cpp` и `echo_client.cpp`, задавая им при компиляции разные имена.

Запустите программы сервера и клиента на разных терминалах. Введите символьную информацию в окне клиента и проанализируйте вывод. Какой разновидности принадлежат сокеты, используемые в данном примере клиент-серверного взаимодействия?

С чем связано создание специального файла в текущем каталоге во время исполнения программ?

4. Скомпилируйте с разными именами программы `sock_c_i_srv.cpp` и `sock_c_i_clt.cpp` (в них используется общий `include` файл `local_c_i.h`). Запустите программы сервера и клиента на разных терминалах. При запуске клиента указывайте в качестве параметра командной строки имя хоста `localhost`. Введите символьную информацию в окне клиента и поясните вывод.

Какой разновидности принадлежат сокеты, используемые в данном примере клиент-серверного взаимодействия?

5. Модифицируйте программу `echo_server.cpp` так, чтобы при ответе на запросы клиента что-либо выводилось в окне сервера.

Испытайте работу эхо-сервера при работе с несколькими клиентами.