**Использование широковещательных IP-адресов**

До сих пор при разработке распределенного приложения предполагались известными сетевой адрес компьютера, на котором находится программа-сервер и номер порта, прослушиваемый этой программой. В реальности распределенное приложение не должно быть привязано к конкретным параметрам сокетов, т.к. это делает ограниченным его применение.

Для обеспечения независимости приложения от параметров сокета сервера (сетевой адрес и номера порта), как правило, номер порта делают одним из параметров инициализации сервера и хранят в специальных конфигурационных файлах, которые считывается сервером при загрузке (реже номер порта передается в виде параметра в командной строке). Так, например, большинство серверов баз данных в качестве одного из параметров инициализации используют номер порта, а при конфигурации (или инсталляции) клиентских приложений указывается сетевой адрес и порт сервера.

В некоторых случаях удобно возложить поиск сетевого адреса сервера на само клиентское приложение (при условии, что номер порта сервера известен). В этих случаях используются широковещательные сетевые адреса, позволяющие адресовать сообщение о поиске сервера всем компьютерам сети. Предполагается, что сервер (или несколько серверов) должен находиться в состоянии ожидания (прослушивания) на доступном в сети компьютере. При получении сообщения от клиента, сервер определяет параметры сокета клиента и передает клиенту необходимые данные для установки канала связи. В общем случае в сети может находиться несколько серверов, которые откликнутся на запрос клиента. В этом случае алгоритм работы клиента должен предполагать процедуру обработки откликов и выбора подходящего сервера. Сразу следует оговориться, что реально данный метод можно применять только внутри сегмента локальной сети, т.к. широковещательные пакеты, как правило, не пропускаются маршрутизаторами и шлюзами .

Использование широковещательных адресов возможно только в протоколе UDP. Поэтомупри создании дескрипторов сокетов (в программах клиентов и серверов) при вызове функции socket значение параметраtype должно быть SOCK\_DGRAM**,** а для обмена данными этом случае используются функции sendto и recvfrom**.**

// **-- установить опции сокета**

// **Назначение:** функция предназначена для установки режимов

// использования сокета

**int setsockopt (**

**SOCKET s,** // [in] дескриптор сокета

**int level,** // [in] уровень действия режима

**int optname,** // [in] режим сокета для установки

**const char\* optval,** // [in] значение режима сокета

**int fromlen** // [in] длина буфера **optval**

**);**

// **Код возврата:** в случае успешного завершения возвращается

// нуль, иначе функция возвращает значение

// **SOCKET\_ERROR**

// **Примечания:** - поддерживаются два значения параметра **level**:

// **SOL\_SOCKET** и **IPPROTO\_TCP;**

// - для уровня **SOL\_SOCKET** параметр **optval** может

// принимать более десяти различных значений;

// например, **SO\_BROADCAST** - для разрешения

// использования широковещательного адреса**;**

**//** - для уровня **IPPROTO\_TCP** поддерживается одно

// значение параметра **level**:**TCP\_NODELAY**,

// которое позволяет устанавливать или отменять

// использование алгоритма Нейгла (см. TCP/IP);

// - значение **fromlen** всегда **sizeof(int)**;

// - если необходимо установить указанный

// параметр**(optname)** в состояние Enabled**,**

// то в поле **optval** долнжо быть не нулевое

//значение (например, **0x00000001**), еслиже

// параметр устанавливается в состояние

// Disabled, то поле **optval** должно содержать

// **0x00000000**

Рисунок 3.15.1. Функция setsockopt

Стандартный широковещательный адрес в формате TCP/IP задается с помощью константы INADDR\_BROADCAST, которая определена в Winsock2.h. По умолчанию использование стандартного широковещательного адреса не допускается и для его применения необходимо установить специальный режим использования сокета SO\_BROADCASTс помощью функции setsockopt (рисунок 3.15.1). Проверить установленные для сокета режимы можно с помощью функции getsockopt (описание здесь не приводится).

//...........................................................

**SOCKET cC;**

**if ((cC = socket(AF\_INET, SOCK\_DGRAM, NULL))== INVALID\_SOCKET)**

**throw SetErrorMsgText("socket:",WSAGetLastError());**

**int optval = 1;**

**if (setsockopt(cC,SOL\_SOCKET,SO\_BROADCAST,**

**(char\*)&optval,sizeof(int)) == SOCKET\_ERROR)**

**throw SetErrorMsgText("opt:",WSAGetLastError());**

**SOCKADDR\_IN all;** // параметры сокета sS

**all.sin\_family = AF\_INET**; // используется IP-адресация

**all.sin\_port = htons(2000);** //порт 2000

**all.sin\_addr.s\_addr = INADDR\_BROADCAST;** // всем

**char buf[] = "answer anyone!";**

**if ((sendlen = sendto(cC, sendbuf, sizeof(buf), NULL,**

**(sockaddr\*)&all, sizeof(all)))== SOCKET\_ERROR)**

**throw SetErrorMsgText("sendto:",WSAGetLastError());**

//...........................................................

Рисунок 3.15.1. Пример применения setsockopе

На рисунке 3.15.1 приводится фрагмент программы, использующей стандартный широковещательный адрес. Функция setsockopt используется в этом примере для установки опции сокета SO\_BROADCAST, позволяющей использовать адрес INADDR\_BROADCAST.