**3.16. Применение символических имен компьютеров**

В предыдущем разделе разбирался механизм поиска серверного компьютера с помощью использования широковещательных адресов. При наличии специальной службы в сети способной разрешить адрес компьютера по его символическому имени (например, DNS или некоторые протоколы, работающие поверх TCP/IP) проблему можно решить с помощью функции gethostbyname (рисунок 3.16.1). При этом предполагается, что известно символическое имя компьютера, на котором находится программа сервера.

Такое решение достаточно часто применяется разработчиками распределенных систем. Связав набор программ-серверов с определенными стандартными именами компьютеров, распределенное приложение становится не зависимым от адресации в сети. Естественно при этом необходимо позаботиться, чтобы существовала служба, разрешающая адреса компьютеров по имени. Установка таких служб, как правило, возлагается на системного администратора сети.

Помимо функции gethostbyname в составе Winsock2имеется функция gethostbyaddr (рисунок 3.16.2)**,**  назначение которой противоположно**:** получение символического имени компьютера по сетевому адресу. Обе функции используют структуру hosten (рисунок 3.16.3), содержащуюся в Winsock2.h.

// **-- получить адрес хоста по его имени**

// **Назначение:** функция для получения информации о хосте по

// его символическому имени

**hostent\* gethostbyname**

**(**

**const char\* name,** // [in] символическое имя хоста

**);**

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

// возвращает указатель на структуру **hosten**,

// иначе значение **NULL**

// **Примечание:** допускается в качестве символического имени,

// указать символическое обозначение адреса

// хоста в виде n.n.n.n

Рисунок 3.16.1. Функция gethostbyname

//-- **получить имя хоста по его адресу**

// **Назначение:** функция для получения информации о хосте по

// его символическому имени

**hostent\* gethostbyaddr**

**(**

**const char\* addr,** // [in] адрес в сетевом формате

**int la,** // [in] длина адреса в байтах

**int ta** // [in] тип адреса: для TCP/IP **AF\_INET**

**);**

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

// возвращает указатель на структуру **hosten**,

// иначе возвращается значение **NULL**

Рисунок 3.16.2 Функция gethostbyaddr

**typedef struct hostent {** // структура hostent

**char FAR\* h\_name;** // имя хоста

**char FAR FAR\*\* h\_aliases;** // список алиасов

**short h\_addrtype**; // тип адресации

**short h\_length;** // длина адреса

**char FAR FAR\*\* h\_addr\_list;** // список адресов

**} hostent;**

Рисунок 3.16.3 Структура hostent

Следует отметить, что символическое имя ***localhost***  является зарезервированным именем и предназначено для обозначения собственного имени компьютера. Если с помощью функции gethostbyname получить адрес компьютера с именем localhost, то в будет собственный получен IP-адрес компьютера или адрес INADDR\_LOOPBACK.

Кроме того, для получения действительного собственного имени компьютера (NetBIOS-имени или DNS-имени) можно использовать функцию gethostname (рисунок 3.16.4).

// **-- получить имя хоста**

// **Назначение:** функция для получения собственного имени хоста

**int gethostname**

**(**

**char\* name ,** // [out] имя хоста

**int ln** // [in] длина буфера **name**

**);**

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

// возвращает нуль, иначе возвращается значение

// **SOCKET\_ERROR**

Рисунок 3.16.4 Функция gethostname