gethostbyname(3) - Linux man page

Name

gethostbyname, gethostbyaddr, sethostent, gethostent, endhostent, herror, hstrerror - get network host entry

Synopsis

```
#include <netdb.h>
extern int h_errno;

struct hostent *gethostbyname(const char *name);

#include <sys/socket.h>       /* for AF_INET */
struct hostent *gethostbyaddr(const void *addr, int len, int type);

void sethostent(int stayopen);

void endhostent(void);

void herror(const char *s);

const char *hstrerror(int err);

/* System V/POSIX extension */
struct hostent *gethostent(void);

/* GNU extensions */
struct hostent *gethostbyname2(const char *name, int af);

int gethostent_r(
  struct hostent *ret, char *buf, size_t buflen,
  struct hostent **result, int *h_errnop);

int gethostbyname_r(const char *name,
  struct hostent *ret, char *buf, size_t buflen,
  struct hostent **result, int *h_errnop);

int gethostbyname2_r(const char *name, int af,
  struct hostent *ret, char *buf, size_t buflen,
  struct hostent **result, int *h_errnop);
```

Description

The **gethostbyname**() function returns a structure of type *hostent* for the given host *name*. Here *name* is either a host name, or an IPv4 address in standard dot notation, or an IPv6 address in colon (and possibly dot) notation. (See RFC 1884 for the description of IPv6 addresses.) If *name* is an IPv4 or IPv6 address, no lookup is performed and **gethostbyname**() simply copies *name* into the *h_name* field and its *struct in_addr* equivalent into the *h_addr_list[0]* field of the returned *hostent* structure. If *name* doesn't end in a dot and the environment variable **HOSTALIASES** is set, the alias file pointed to by **HOSTALIASES** will first be searched for *name* (see **hostname**(7) for the file format). The current domain and its parents are searched unless *name* ends in a dot.

The **gethostbyaddr**() function returns a structure of type *hostent* for the given host address *addr* of length *len* and address type *type*. Valid address types are **AF_INET** and **AF_INET6**. The host address argument is a pointer to a struct of a type depending on the address type, for example a **struct in_addr *** (probably obtained via a call to *inet_addr*()) for address type AF_INET.

The **sethostent**() function specifies, if *stayopen* is true (1), that a connected TCP socket should be used for the name server queries and that the connection should remain open during successive queries. Otherwise, name server queries will use UDP datagrams.

The **endhostent**() function ends the use of a TCP connection for name server queries.

The (obsolete) **herror**() function prints the error message associated with the current value of *h_errno* on stderr.

The (obsolete) **hstrerror**() function takes an error number (typically *h_errno*) and returns the corresponding message string.

The domain name queries carried out by **gethostbyname**() and **gethostbyaddr**() use a combination of any or all of the name server **named**(8), a broken out line from */etc/hosts*, and the Network Information Service (NIS or YP), depending upon the contents of the *order* line in */etc/host.conf*. The default action is to query **named**(8), followed by */etc/hosts*.

The *hostent* structure is defined in <*netdb.h*> as follows:

```
struct hostent {
      char *h_name;       /* official name of host */
      char **h_aliases;   /* alias list */
      int  h_addrtype;    /* host address type */
      int  h_length;      /* length of address */
      char **h_addr_list; /* list of addresses */
}

#define h_addr  h_addr_list[0]   /* for backward compatibility */
```

The members of the *hostent* structure are:

h_name The official name of the host.

h_aliases An array of alternative names for the host, terminated by a NULL pointer.

h_addrtype The type of address; always **AF_INET** or **AF_INET6** at present.

h_length The length of the address in bytes.

h_addr_list An array of pointers to network addresses for the host (in network byte order), terminated by a NULL pointer.

h_addr The first address in *h_addr_list* for backward compatibility.

Return Value

The **gethostbyname**() and **gethostbyaddr**() functions return the *hostent* structure or a NULL pointer if an error occurs. On error, the *h_errno* variable holds an error number. When non-NULL, the return value may point at static data, see the notes below.

Errors

The variable *h_errno* can have the following values:

HOST_NOT_FOUND The specified host is unknown.

NO_ADDRESS or **NO_DATA** The requested name is valid but does not have an IP address.

NO_RECOVERY A non-recoverable name server error occurred.

TRY_AGAIN A temporary error occurred on an authoritative name server. Try again later.