Red Hat Enterprise Linux Release 9.2 Manual Pages on 'getipnodebyaddr.3' command

$ man getipnodebyaddr.3

```
GETIPNODEBYNAME(3)       Linux Programmer's Manual       GETIPNODEBYNAME(3)

NAME

       getipnodebyname,  getipnodebyaddr,  freehostent - get network hostnames

       and addresses

SYNOPSIS

       #include <sys/types.h>

       #include <sys/socket.h>

       #include <netdb.h>

       struct hostent *getipnodebyname(const char *name, int af,

                                       int flags, int *error_num);

       struct hostent *getipnodebyaddr(const void *addr, size_t len,

                                       int af, int *error_num);

       void freehostent(struct hostent *ip);

DESCRIPTION

       These functions are deprecated (and unavailable in glibc).  Use  getad?

       drinfo(3) and getnameinfo(3) instead.

       The  getipnodebyname() and getipnodebyaddr() functions return the names

       and addresses of a network host.  These functions return a  pointer  to

       the following structure:

           struct hostent {

               char  *h_name;

               char **h_aliases;

               int    h_addrtype;
```

```
        int    h_length;
        char **h_addr_list;
    };
```

These functions replace the gethostbyname(3) and gethostbyaddr(3) func?
tions, which could access only the IPv4 network address family. The
getipnodebyname() and getipnodebyaddr() functions can access multiple
network address families.

Unlike the gethostby functions, these functions return pointers to dy?
namically allocated memory. The freehostent() function is used to re?
lease the dynamically allocated memory after the caller no longer needs
the hostent structure.

getipnodebyname() arguments

The getipnodebyname() function looks up network addresses for the host specified by the name argument. The af argument specifies one of the following values:

AF_INET

The name argument points to a dotted-quad IPv4 address or a name of an IPv4 network host.

AF_INET6

The name argument points to a hexadecimal IPv6 address or a name of an IPv6 network host.

The flags argument specifies additional options. More than one option can be specified by bitwise OR-ing them together. flags should be set to 0 if no options are desired.

AI_V4MAPPED

This flag is used with AF_INET6 to request a query for IPv4 ad?
dresses instead of IPv6 addresses; the IPv4 addresses will be
mapped to IPv6 addresses.

AI_ALL This flag is used with AI_V4MAPPED to request a query for both IPv4 and IPv6 addresses. Any IPv4 address found will be mapped to an IPv6 address.

AI_ADDRCONFIG

This flag is used with AF_INET6 to further request that queries for IPv6 addresses should not be made unless the system has at least one IPv6 address assigned to a network interface, and that queries for IPv4 addresses should not be made unless the system has at least one IPv4 address assigned to a network interface. This flag may be used by itself or with the AI_V4MAPPED flag.

AI_DEFAULT

This flag is equivalent to (AI_ADDRCONFIG | AI_V4MAPPED).

getipnodebyaddr() arguments

The getipnodebyaddr() function looks up the name of the host whose net?work address is specified by the addr argument. The af argument speci?fies one of the following values:

AF_INET

The addr argument points to a struct in_addr and len must be set to sizeof(struct in_addr).

AF_INET6

The addr argument points to a struct in6_addr and len must be set to sizeof(struct in6_addr).

RETURN VALUE

NULL is returned if an error occurred, and error_num will contain an error code from the following list:

HOST_NOT_FOUND

The hostname or network address was not found.

NO_ADDRESS

The domain name server recognized the network address or name, but no answer was returned. This can happen if the network host has only IPv4 addresses and a request has been made for IPv6 in?formation only, or vice versa.

NO_RECOVERY

The domain name server returned a permanent failure response.

TRY_AGAIN

The domain name server returned a temporary failure response. You might have better luck next time.

A successful query returns a pointer to a hostent structure that con?

tains the following fields:

h_name This is the official name of this network host.

h_aliases

This is an array of pointers to unofficial aliases for the same host. The array is terminated by a null pointer.

h_addrtype

This is a copy of the af argument to getipnodebyname() or getipnodebyaddr(). h_addrtype will always be AF_INET if the af argument was AF_INET. h_addrtype will always be AF_INET6 if the af argument was AF_INET6.

h_length

This field will be set to sizeof(struct in_addr) if h_addrtype is AF_INET, and to sizeof(struct in6_addr) if h_addrtype is AF_INET6.

h_addr_list

This is an array of one or more pointers to network address structures for the network host. The array is terminated by a null pointer.

CONFORMING TO

RFC 2553.

NOTES

These functions were present in glibc 2.1.91-95, but were removed again. Several UNIX-like systems support them, but all call them dep?recated.

SEE ALSO

getaddrinfo(3), getnameinfo(3), inet_ntop(3), inet_pton(3)

COLOPHON

This page is part of release 5.10 of the Linux man-pages project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at https://www.kernel.org/doc/man-pages/.

Linux 2017-09-15 GETIPNODEBYNAME(3)