NAME

netinet/in.h − Internet address family

SYNOPSIS

#include <netinet/in.h>

DESCRIPTION

The <*netinet/in.h*> header shall define the following types:

in_port_t

Equivalent to the type **uint16_t** as defined in <*inttypes.h*> .

in_addr_t

Equivalent to the type **uint32_t** as defined in <*inttypes.h*> .

The **sa_family_t** type shall be defined as described in <*sys/socket.h*> .

The **uint8_t** and **uint32_t** type shall be defined as described in <*inttypes.h*>. Inclusion of the <*netinet/in.h*> header may also make visible all symbols from <*inttypes.h*> and <*sys/socket.h*>.

The <*netinet/in.h*> header shall define the **in_addr** structure that includes at least the following member:

in_addr_t s_addr

The <*netinet/in.h*> header shall define the **sockaddr_in** structure that includes at least the following members (all in network byte order):

sa_family_t	**sin_family**	AF_INET.
in_port_t	**sin_port**	Port number.
struct in_addr	**sin_addr**	IP address.

The **sockaddr_in** structure is used to store addresses for the Internet address family. Values of this type shall be cast by applications to **struct sockaddr** for use with socket functions.

The <*netinet/in.h*> header shall define the **in6_addr** structure that contains at least the following member:

uint8_t s6_addr[16]

This array is used to contain a 128-bit IPv6 address, stored in network byte order.

The <*netinet/in.h*> header shall define the **sockaddr_in6** structure that includes at least the following members (all in network byte order):

sa_family_t	**sin6_family**	AF_INET6.
in_port_t	**sin6_port**	Port number.
uint32_t	**sin6_flowinfo**	IPv6 traffic class and flow information.
struct in6_addr	**sin6_addr**	IPv6 address.
uint32_t	**sin6_scope_id**	Set of interfaces for a scope.

The **sockaddr_in6** structure shall be set to zero by an application prior to using it, since implementations are free to have additional, implementation-defined fields in **sockaddr_in6**.

The *sin6_scope_id* field is a 32-bit integer that identifies a set of interfaces as appropriate for the scope of the address carried in the *sin6_addr* field. For a link scope *sin6_addr*, the application shall ensure that *sin6_scope_id* is a link index. For a site scope *sin6_addr*, the application shall ensure that *sin6_scope_id* is a site index. The mapping of *sin6_scope_id* to an interface or set of interfaces is implementation-defined.

The <*netinet/in.h*> header shall declare the following external variable:

const struct in6_addr in6addr_any

This variable is initialized by the system to contain the wildcard IPv6 address. The <*netinet/in.h*> header also defines the IN6ADDR_ANY_INIT macro. This macro must be constant at compile time and can be used to initialize a variable of type **struct in6_addr** to the IPv6 wildcard address.

The <*netinet/in.h*> header shall declare the following external variable:

const struct in6_addr in6addr_loopback

This variable is initialized by the system to contain the loopback IPv6 address. The <*netinet/in.h*> header also defines the IN6ADDR_LOOPBACK_INIT macro. This macro must be constant at compile time and can be used to initialize a variable of type **struct in6_addr** to the IPv6 loopback address.

The <*netinet/in.h*> header shall define the **ipv6_mreq** structure that includes at least the following members:

struct in6_addr **ipv6mr_multiaddr** IPv6 multicast address.
unsigned **ipv6mr_interface** Interface index.

The <*netinet/in.h*> header shall define the following macros for use as values of the *level* argument of *getsockopt*() and *setsockopt*():

IPPROTO_IP
Internet protocol.

IPPROTO_IPV6
Internet Protocol Version 6.

IPPROTO_ICMP
Control message protocol.

IPPROTO_RAW
Raw IP Packets Protocol.

IPPROTO_TCP
Transmission control protocol.

IPPROTO_UDP
User datagram protocol.

The <*netinet/in.h*> header shall define the following macros for use as destination addresses for *connect*(), *sendmsg*(), and *sendto*():

INADDR_ANY
IPv4 local host address.

INADDR_BROADCAST
IPv4 broadcast address.

The <*netinet/in.h*> header shall define the following macro to help applications declare buffers of the proper size to store IPv4 addresses in string form:

INET_ADDRSTRLEN
16. Length of the string form for IP.

The *htonl*(), *htons*(), *ntohl*(), and *ntohs*() functions shall be available as defined in <*arpa/inet.h*> . Inclusion of the <*netinet/in.h*> header may also make visible all symbols from <*arpa/inet.h*> .

The <*netinet/in.h*> header shall define the following macro to help applications declare buffers of the proper size to store IPv6 addresses in string form:

INET6_ADDRSTRLEN
46. Length of the string form for IPv6.

The *<netinet/in.h>* header shall define the following macros, with distinct integer values, for use in the *option_name* argument in the *getsockopt*() or *setsockopt*() functions at protocol level IPPROTO_IPV6:

IPV6_JOIN_GROUP
Join a multicast group.

IPV6_LEAVE_GROUP
Quit a multicast group.

IPV6_MULTICAST_HOPS

Multicast hop limit.

IPV6_MULTICAST_IF
Interface to use for outgoing multicast packets.

IPV6_MULTICAST_LOOP

Multicast packets are delivered back to the local application.

IPV6_UNICAST_HOPS
Unicast hop limit.

IPV6_V6ONLY
Restrict AF_INET6 socket to IPv6 communications only.

The *<netinet/in.h>* header shall define the following macros that test for special IPv6 addresses. Each macro is of type **int** and takes a single argument of type **const struct in6_addr ***:

IN6_IS_ADDR_UNSPECIFIED

Unspecified address.

IN6_IS_ADDR_LOOPBACK

Loopback address.

IN6_IS_ADDR_MULTICAST

Multicast address.

IN6_IS_ADDR_LINKLOCAL

Unicast link-local address.

IN6_IS_ADDR_SITELOCAL

Unicast site-local address.

IN6_IS_ADDR_V4MAPPED

IPv4 mapped address.

IN6_IS_ADDR_V4COMPAT

IPv4-compatible address.

IN6_IS_ADDR_MC_NODELOCAL

Multicast node-local address.

IN6_IS_ADDR_MC_LINKLOCAL

Multicast link-local address.

IN6_IS_ADDR_MC_SITELOCAL

Multicast site-local address.

IN6_IS_ADDR_MC_ORGLOCAL

Multicast organization-local address.

IN6_IS_ADDR_MC_GLOBAL

Multicast global address.

The following sections are informative.

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

Host and Network Byte Orders , *<arpa/inet.h>* , *<inttypes.h>* , *<sys/socket.h>* , the System Interfaces volume of IEEE Std 1003.1-2001, *connect*(), *getsockopt*(), *htonl*(), *htons*(), *ntohl*(), *ntohs*(), *sendmsg*(), *sendto*(), *setsockopt*()

COPYRIGHT