NAME

inet_ntop − convert IPv4 and IPv6 addresses from binary to text form

LIBRARY

Standard C library (*libc*, *−lc*)

SYNOPSIS

```
#include <arpa/inet.h>

const char *inet_ntop(int af, const void *restrict src,
                      char dst[restrict .size], socklen_t size);
```

DESCRIPTION

This function converts the network address structure *src* in the *af* address family into a character string. The resulting string is copied to the buffer pointed to by *dst*, which must be a non-null pointer. The caller specifies the number of bytes available in this buffer in the argument *size*.

inet_ntop() extends the **inet_ntoa**(3) function to support multiple address families, **inet_ntoa**(3) is now considered to be deprecated in favor of **inet_ntop**(). The following address families are currently supported:

AF_INET
: *src* points to a *struct in_addr* (in network byte order) which is converted to an IPv4 network address in the dotted-decimal format, "*ddd.ddd.ddd.ddd*". The buffer *dst* must be at least **INET_ADDRSTRLEN** bytes long.

AF_INET6
: *src* points to a *struct in6_addr* (in network byte order) which is converted to a representation of this address in the most appropriate IPv6 network address format for this address. The buffer *dst* must be at least **INET6_ADDRSTRLEN** bytes long.

RETURN VALUE

On success, **inet_ntop**() returns a non-null pointer to *dst*. NULL is returned if there was an error, with *errno* set to indicate the error.

ERRORS

EAFNOSUPPORT
: *af* was not a valid address family.

ENOSPC
: The converted address string would exceed the size given by *size*.

ATTRIBUTES

For an explanation of the terms used in this section, see **attributes**(7).

Interface	Attribute	Value
inet_ntop()	Thread safety	MT-Safe locale

STANDARDS

POSIX.1-2001, POSIX.1-2008. Note that RFC 2553 defines a prototype where the last argument *size* is of type *size_t*. Many systems follow RFC 2553. glibc 2.0 and 2.1 have *size_t*, but 2.2 and later have *socklen_t*.

BUGS

AF_INET6 converts IPv4-mapped IPv6 addresses into an IPv6 format.

EXAMPLES

See **inet_pton**(3).

SEE ALSO

getnameinfo(3), **inet**(3), **inet_pton**(3)