

12.4 IPv6 Address-Testing Macros

There is a small class of IPv6 applications that must know whether they are talking to an IPv4 peer. These applications need to know if the peer's address is an IPv4-mapped IPv6 address. The following 12 macros are defined to test an IPv6 address for certain properties.

```
#include <netinet/in.h>

int IN6_IS_ADDR_UNSPECIFIED(const struct in6_addr *aptr);
int IN6_IS_ADDR_LOOPBACK(const struct in6_addr *aptr);
int IN6_IS_ADDR_MULTICAST(const struct in6_addr *aptr);
int IN6_IS_ADDR_LINKLOCAL(const struct in6_addr *aptr);
int IN6_IS_ADDR_SITELocal(const struct in6_addr *aptr);
int IN6_IS_ADDR_V4MAPPED(const struct in6_addr *aptr);
int IN6_IS_ADDR_V4COMPAT(const struct in6_addr *aptr);
int IN6_IS_ADDR_MC_NODELOCAL(const struct in6_addr *aptr);
int IN6_IS_ADDR_MC_LINKLOCAL(const struct in6_addr *aptr);
int IN6_IS_ADDR_MC_SITELocal(const struct in6_addr *aptr);
int IN6_IS_ADDR_MC_ORGLOCAL(const struct in6_addr *aptr);
int IN6_IS_ADDR_MC_GLOBAL(const struct in6_addr *aptr);
```

All return: nonzero if IPv6 address is of specified type, zero otherwise

The first seven macros test the basic type of IPv6 address. We show these various address types in [Section A.5](#). The final five macros test the scope of an IPv6 multicast address ([Section 21.2](#)).

IPv4-compatible addresses are used by a transition mechanism that has fallen out of favor. You're not likely to actually see this type of address or need to test for it.

An IPv6 client could call the `IN6_IS_ADDR_V4MAPPED` macro to test the IPv6 address returned by the resolver. An IPv6 server could call this macro to test the IPv6 address returned by `accept` or `recvfrom`.

As an example of an application that needs this macro, consider FTP and its `PORT` command. If we start an FTP client, log in to an FTP server, and issue an FTP `dir` command, the FTP client sends a `PORT` command to the FTP server across the control connection. This tells the server the client's IP address and port, to which the server then creates a data connection. (Chapter 27 of TCPv1 contains all the details of the FTP application protocol.) But, an IPv6 FTP client must know whether the server is an IPv4 server or an IPv6 server, because the former requires a command of the form `PORT a1,a2,a3,a4,p1,p2` where the first four numbers (each between 0 and 255) form the 4-byte IPv4 address and the last two numbers form the 2-byte port number. An IPv6 server, however, requires an `EPRT` command (RFC 2428 [Allman, Ostermann, and Metz 1998]), containing an address family, text format address, and text format port. [Exercise 12.1](#) gives an example of IPv4 and IPv6 FTP protocol behavior.