NAME

   ne_addr_resolve, ne_addr_result, ne_addr_first, ne_addr_next, ne_addr_error, ne_addr_destroy -
   functions to resolve hostnames to addresses

SYNOPSIS

   #include <ne_socket.h>

   ne_sock_addr *ne_addr_resolve(const char ***hostname, int** flags**);**

   int ne_addr_result(const ne_sock_addr ***addr);**

   const ne_inet_addr *ne_addr_first(ne_sock_addr ***addr);**

   const ne_inet_addr *ne_addr_next(ne_sock_addr ***addr);**

   char *ne_addr_error(const ne_sock_addr ***addr, char ***buffer**, size_t** bufsiz**);**

   void ne_addr_destroy(ne_sock_addr ***addr);**

DESCRIPTION

   The **ne_addr_resolve** function resolves the given *hostname*, returning an **ne_sock_addr** object
   representing the address (or addresses) associated with the hostname. The *flags* parameter is currently
   unused, and must be passed as 0.

   The *hostname* passed to **ne_addr_resolve** can be a DNS hostname (e.g. "www.example.com") or an
   IPv4 dotted quad (e.g. "192.0.34.72"); or, on systems which support IPv6, an IPv6 hex address, which
   may be enclosed in brackets, e.g. "[::1]".

   To determine whether the hostname was successfully resolved, the **ne_addr_result** function is used,
   which returns non-zero if an error occurred. If an error did occur, the **ne_addr_error** function can be
   used, which will copy the error string into a given *buffer* (of size *bufsiz*).

   The functions **ne_addr_first** and **ne_addr_next** are used to retrieve the Internet addresses associated
   with an address object which has been successfully resolved. **ne_addr_first** returns the first address;
   ne_addr_next returns the next address after the most recent call to **ne_addr_next** or **ne_addr_first**, or
   NULL if there are no more addresses. The **ne_inet_addr** pointer returned by these functions can be
   passed to **ne_sock_connect** to connect a socket.

   After the address object has been used, it should be destroyed using **ne_addr_destroy**.

RETURN VALUE

   ne_addr_resolve returns a pointer to an address object, and never NULL. **ne_addr_error** returns the
   buffer parameter .

EXAMPLES

   The code below prints out the set of addresses associated with the hostname www.google.com.

```
ne_sock_addr *addr;
char buf[256];

addr = ne_addr_resolve("www.google.com", 0);
if (ne_addr_result(addr)) {
  printf("Could not resolve www.google.com: %s\n",
      ne_addr_error(addr, buf, sizeof buf));
} else {
  const ne_inet_addr *ia;
  printf("www.google.com:");
  for (ia = ne_addr_first(addr); ia != NULL; ia = ne_addr_next(addr)) {
    printf(" %s", ne_iaddr_print(ia, buf, sizeof buf));
  }
  putchar('\n');
}
ne_addr_destroy(addr);
```

SEE ALSO

   ne_iaddr_print

AUTHOR

   Joe Orton <neon@lists.manyfish.co.uk>
       Author.

COPYRIGHT