NAME

ares_gethostbyaddr − Initiate a host query by address

SYNOPSIS

#include <ares.h>

typedef void (*ares_host_callback)(void *_arg_, int _status_, int _timeouts_, struct hostent *_hostent_)

void ares_gethostbyaddr(ares_channel _channel_, const void *_addr_, int _addrlen_, int _family_, ares_host_callback _callback_, void *_arg_)

DESCRIPTION

The **ares_gethostbyaddr** function initiates a host query by address on the name service channel identified by *channel*. The parameters *addr* and *addrlen* give the address as a series of bytes, and *family* gives the type of address. When the query is complete or has failed, the ares library will invoke *callback*. Completion or failure of the query may happen immediately, or may happen during a later call to *ares_process(3)*, *ares_destroy(3)* or *ares_cancel(3)*.

The callback argument *arg* is copied from the **ares_gethostbyaddr** argument *arg*. The callback argument *status* indicates whether the query succeeded and, if not, how it failed. It may have any of the following values:

ARES_SUCCESS The host lookup completed successfully.

ARES_ENOTIMP The ares library does not know how to look up addresses of type *family*.

ARES_ENOTFOUND
The address *addr* was not found.

ARES_ENOMEM Memory was exhausted.

ARES_ECANCELLED
The query was cancelled.

ARES_EDESTRUCTION
The name service channel *channel* is being destroyed; the query will not be completed.

The callback argument *timeouts* reports how many times a query timed out during the execution of the given request.

On successful completion of the query, the callback argument *hostent* points to a **struct hostent** containing the name of the host returned by the query. The callback need not and should not attempt to free the memory pointed to by *hostent*; the ares library will free it when the callback returns. If the query did not complete successfully, *hostent* will be **NULL**.

SEE ALSO

ares_process(3), **ares_gethostbyname**(3)

AUTHOR

Greg Hudson, MIT Information Systems