NAME

CURLOPT_RESOLVE − provide custom host name to IP address resolves

SYNOPSIS

```
#include <curl/curl.h>

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_RESOLVE,
                          struct curl_slist *hosts);
```

DESCRIPTION

Pass a pointer to a linked list of strings with host name resolve information to use for requests with this handle. The linked list should be a fully valid list of **struct curl_slist** structs properly filled in. Use *curl_slist_append(3)* to create the list and *curl_slist_free_all(3)* to clean up an entire list.

Each single name resolve string should be written using the format HOST:PORT:ADDRESS where HOST is the name libcurl will try to resolve, PORT is the port number of the service where libcurl wants to connect to the HOST and ADDRESS is the numerical IP address. If libcurl is built to support IPv6, ADDRESS can of course be either IPv4 or IPv6 style addressing.

This option effectively pre-populates the DNS cache with entries for the host+port pair so redirects and everything that operations against the HOST+PORT will instead use your provided ADDRESS. Addresses to set with *CURL_RESOLVE* will not time-out from the DNS cache like ordinary entries.

You can remove names from the DNS cache again, to stop providing these fake resolves, by including a string in the linked list that uses the format "-HOST:PORT". The host name must be prefixed with a dash, and the host name and port number must exactly match what was already added previously.

DEFAULT

NULL

PROTOCOLS

All

EXAMPLE

```
CURL *curl;
struct curl_slist *host = NULL;
host = curl_slist_append(NULL, "example.com:80:127.0.0.1");

curl = curl_easy_init();
if(curl) {
  curl_easy_setopt(curl, CURLOPT_RESOLVE, host);
  curl_easy_setopt(curl, CURLOPT_URL, "http://example.com");
  res = curl_easy_perform(curl);

  /* always cleanup */
  curl_easy_cleanup(curl);
}

curl_slist_free_all(host);
```

AVAILABILITY

Added in 7.21.3

RETURN VALUE

Returns CURLE_OK if the option is supported, and CURLE_UNKNOWN_OPTION if not.

SEE ALSO

CURLOPT_IPRESOLVE(3), **CURLOPT_DNS_CACHE_TIMEOUT**(3),