NAME

CURLOPT_MAIL_RCPT − list of SMTP mail recipients

SYNOPSIS

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

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_MAIL_RCPT,
                          struct curl_slist *rcpts);
```

DESCRIPTION

Pass a pointer to a linked list of recipients to pass to the server in your SMTP mail request. 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.

libcurl does not copy the list, it needs to be kept around until after the transfer has completed.

When performing a mail transfer, each recipient should be specified within a pair of angled brackets (<>). Should you not use an angled bracket as the first character, libcurl assumes you provided a single email address and encloses that address within brackets for you.

In order to specify DSN parameters (as per RFC 3461), the address has to be written in angled brackets, followed by the parameters.

When performing an address verification (**VRFY** command), each recipient should be specified as the username or username plus domain (as per Section 3.5 of RFC 5321).

When performing a mailing list expand (**EXPN** command), each recipient should be specified using the mailing list name, such as *Friends* or *London−Office*.

Using this option multiple times makes the last set list override the previous ones. Set it to NULL to disable its use again.

DEFAULT

NULL

PROTOCOLS

This functionality affects smtp only

EXAMPLE

```
int main(void)
{
 CURL *curl = curl_easy_init();
 if(curl) {
  CURLcode result;
  struct curl_slist *list;
  list = curl_slist_append(NULL, "root@localhost");
  list = curl_slist_append(list, "person@example.com");
  list = curl_slist_append(list, "<other@example.com> NOTIFY=SUCCESS");
  curl_easy_setopt(curl, CURLOPT_URL, "smtp://example.com/");
  curl_easy_setopt(curl, CURLOPT_MAIL_RCPT, list);
  result = curl_easy_perform(curl);
  curl_slist_free_all(list);
  curl_easy_cleanup(curl);
 }
}
```

AVAILABILITY

Added in curl 7.20.0

RETURN VALUE

curl_easy_setopt(3) returns a CURLcode indicating success or error.

CURLE_OK (0) means everything was OK, non−zero means an error occurred, see *libcurl−errors(3)*.

SEE ALSO

CURLOPT_MAIL_AUTH(3), **CURLOPT_MAIL_FROM**(3)