

NAME

CURLOPT_DNS_SERVERS – DNS servers to use

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_DNS_SERVERS, char *servers);
```

DESCRIPTION

Pass a char pointer that is the list of DNS servers to be used instead of the system default. The format of the dns servers option is:

```
host[:port][,host[:port]]...
```

For example:

```
192.168.1.100,192.168.1.101,3.4.5.6
```

The application does not have to keep the string around after setting this option.

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

DEFAULT

NULL

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/foo.bin");
        curl_easy_setopt(curl, CURLOPT_DNS_SERVERS,
            "192.168.1.100:53,192.168.1.101");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

NOTES

This option requires that libcurl was built with a resolver backend that supports this operation. The c-ares backend is the only such one.

AVAILABILITY

Added in curl 7.24.0

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_DNS_CACHE_TIMEOUT(3), **CURLOPT_DNS_LOCAL_IP4(3),** **CURLOPT_DNS_LOCAL_IP6(3)**