

NAME

CURLOPT_FTPPORT – make FTP transfer active

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_FTPPORT, char *spec);
```

DESCRIPTION

Pass a pointer to a null-terminated string as parameter. It specifies that the FTP transfer should be made actively and the given string is used to get the IP address to use for the FTP PORT instruction.

The PORT instruction tells the remote server to do a TCP connect to our specified IP address. The string may be a plain IP address, a hostname, a network interface name (under Unix) or a '-' symbol to let the library use your system's default IP address. Default FTP operations are passive, and does not use the PORT command.

The address can be followed by a ':' to specify a port, optionally followed by a '-' to specify a port range. If the port specified is 0, the operating system picks a free port. If a range is provided and all ports in the range are not available, libcurl reports CURLE_FTP_PORT_FAILED for the handle. Invalid port/range settings are ignored. IPv6 addresses followed by a port or port range have to be in brackets. IPv6 addresses without port/range specifier can be in brackets.

Examples with specified ports:

```
eth0:0
192.168.1.2:32000-33000
curl.se:32123
[::1]:1234-4567
```

We strongly advise against specifying the address with a name, as it causes libcurl to do a blocking name resolve call to retrieve the IP address. That name resolve operation does **not** use DNS-over-HTTPS even if *CURLOPT_DOH_URL*(3) is set.

Using anything else than "-" for this option should typically only be done if you have special knowledge and confirmation that it works.

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. You disable PORT again and go back to using the passive version by setting this option to NULL.

DEFAULT

NULL

PROTOCOLS

This functionality affects ftp only

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL,
            "ftp://example.com/old-server/file.txt");
        curl_easy_setopt(curl, CURLOPT_FTPPORT, "-");
    }
}
```

```
    result = curl_easy_perform(curl);
    curl_easy_cleanup(curl);
}
```

AVAILABILITY

Added in curl 7.1

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_ACCEPTTIMEOUT_MS(3), **CURLOPT_FTP_USE_EPRT(3),** **CUR-**
LOPT_FTP_USE_EPSV(3)