

NAME

CURLOPT_NOPROXY – disable proxy use for specific hosts

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_NOPROXY, char *noproxy);
```

DESCRIPTION

Pass a pointer to a null-terminated string. The string consists of a comma separated list of hostnames that do not require a proxy to get reached, even if one is specified. The only wildcard available is a single `*` character, which matches all hosts, and effectively disables the proxy. Each name in this list is matched as either a domain which contains the hostname, or the hostname itself. For example, "ample.com" would match ample.com, ample.com:80, and www.ample.com, but not www.example.com or ample.com.org.

Setting the *noproxy* string to "" (an empty string) explicitly enables the proxy for all hostnames, even if there is an environment variable set for it.

Enter IPv6 numerical addresses in the list of hostnames without enclosing brackets:

```
"example.com,::1,localhost"
```

IP addresses specified to this option can be provided using CIDR notation: an appended slash and number specifies the number of "network bits" out of the address to use in the comparison. For example "192.168.0.0/16" would match all addresses starting with "192.168".

To use international hostnames in this list, add the punycode version of the hostname.

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.

Environment variables

If there is an environment variable called **no_proxy** (or **NO_PROXY**), it is used if the *CURLOPT_NOPROXY(3)* option is not set. It works exactly the same way.

DEFAULT

NULL

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        /* accept various URLs */
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/");
        /* use this proxy */
        curl_easy_setopt(curl, CURLOPT_PROXY, "http://proxy.example:80");
        /* ... but make sure this hostname is not proxied */
        curl_easy_setopt(curl, CURLOPT_NOPROXY, "www.example.com");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

}

HISTORY

CIDR format support was added in 7.86.0.

AVAILABILITY

Added in curl 7.19.4

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_PROXY(3), CURLOPT_PROXYAUTH(3), CURLOPT_PROXYTYPE(3)