

NAME

CURLMOPT_PIPELINING – enable HTTP multiplexing

SYNOPSIS

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

```
CURLMcode curl_multi_setopt(CURLM *handle, CURLMOPT_PIPELINING, long bitmask);
```

DESCRIPTION

Pass in the correct value in the **bitmask** parameter to instruct libcurl to enable multiplexing for this multi handle.

With multiplexing enabled, libcurl attempts to do multiple transfers over the same connection when doing parallel transfers to the same hosts.

CURLPIPE_NOHING (0)

Make no attempts at multiplexing.

CURLPIPE_HTTP1 (1)

This bit is deprecated and has no effect since version 7.62.0.

CURLPIPE_MULTIPLEX (2)

If this bit is set, libcurl tries to multiplex the new transfer over an existing connection if possible. This requires HTTP/2 or HTTP/3.

DEFAULT

CURLPIPE_MULTIPLEX

PROTOCOLS

This functionality affects http only

EXAMPLE

```
int main(void)
{
    CURLM *m = curl_multi_init();
    /* try HTTP/2 multiplexing */
    curl_multi_setopt(m, CURLMOPT_PIPELINING, CURLPIPE_MULTIPLEX);
}
```

HISTORY

The multiplex support bit was added in 7.43.0. HTTP/1 Pipelining support was disabled in 7.62.0.

Since 7.62.0, **CURLPIPE_MULTIPLEX** is enabled by default.

Before that, default was **CURLPIPE_NOHING**.

AVAILABILITY

Added in curl 7.16.0

RETURN VALUE

curl_multi_setopt(3) returns a CURLMcode indicating success or error.

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

SEE ALSO

CURLMOPT_CHUNK_LENGTH_PENALTY_SIZE(3), **CURLMOPT_CONTENT_LENGTH_PENALTY_SIZE(3),** **CURLMOPT_MAXCONNECTS(3),** **CURLMOPT_MAX_HOST_CONNECTIONS(3),** **CURLMOPT_MAX_PIPELINE_LENGTH(3),** **CURLMOPT_PIPELINING_SITE_BL(3)**