

CURLOPT\_CHUNK\_BGN\_FUNCTION(3) curl\_easy\_setopt options CURLOPT\_CHUNK\_BGN\_FUNCTION(3)

## NAME

CURLOPT\_CHUNK\_BGN\_FUNCTION – callback before a transfer with FTP wildcardmatch

## SYNOPSIS

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

```
long chunk_bgn_callback(const void *transfer_info, void *ptr,  
                        int remains);
```

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_CHUNK_BGN_FUNCTION,  
                          chunk_bgn_callback);
```

## DESCRIPTION

Pass a pointer to your callback function, which should match the prototype shown above.

This callback function gets called by libcurl before a part of the stream is going to be transferred (if the transfer supports chunks).

The *transfer\_info* pointer will point to a struct `curl_fileinfo` with details about the file that is about to get transferred.

This callback makes sense only when using the `CURLOPT_WILDCARDMATCH(3)` option for now.

The target of *transfer\_info* parameter is a "feature depended" structure. For the FTP wildcard download, the target is `curl_fileinfo` structure (see *curl/curl.h*). The parameter *ptr* is a pointer given by `CURLOPT_CHUNK_DATA(3)`. The parameter *remains* contains number of chunks remaining per the transfer. If the feature is not available, the parameter has zero value.

Return `CURL_CHUNK_BGN_FUNC_OK` if everything is fine, `CURL_CHUNK_BGN_FUNC_SKIP` if you want to skip the concrete chunk or `CURL_CHUNK_BGN_FUNC_FAIL` to tell libcurl to stop if some error occurred.

## DEFAULT

NULL

## PROTOCOLS

FTP

## EXAMPLE

TODO

## AVAILABILITY

This was added in 7.21.0

## RETURN VALUE

Returns `CURLE_OK` if the option is supported, and `CURLE_UNKNOWN_OPTION` if not.

## SEE ALSO

`CURLOPT_CHUNK_END_FUNCTION(3)`, `CURLOPT_WILDCARDMATCH(3)`,