

NAME

CURLOPT_INTERLEAVEDATA – pointer passed to RTSP interleave callback

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_INTERLEAVEDATA, void *pointer);
```

DESCRIPTION

This is the userdata *pointer* that is passed to *CURLOPT_INTERLEAVEFUNCTION(3)* when interleaved RTP data is received. If the interleave function callback is not set, this pointer is not used anywhere.

DEFAULT

NULL

PROTOCOLS

This functionality affects rtsp only

EXAMPLE

```
struct local {
    void *custom;
};

static size_t rtp_write(void *ptr, size_t size, size_t nmemb, void *userp)
{
    struct local *l = userp;
    printf("my pointer: %p\n", l->custom);
    /* take care of the packet in 'ptr', then return... */
    return size * nmemb;
}

int main(void)
{
    struct local rtp_data;
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_INTERLEAVEFUNCTION, rtp_write);
        curl_easy_setopt(curl, CURLOPT_INTERLEAVEDATA, &rtp_data);

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

AVAILABILITY

Added in curl 7.20.0

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_INTERLEAVEFUNCTION(3), CURLOPT_RTSP_REQUEST(3)