

**NAME**

curl\_mime\_init - create a mime handle

**SYNOPSIS**

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

```
curl_mime *curl_mime_init(CURL *easy_handle);
```

**DESCRIPTION**

*curl\_mime\_init(3)* creates a handle to a new empty mime structure. This mime structure can be subsequently filled using the mime API, then attached to some easy handle using option *CURLOPT\_MIMEPOST(3)* within a *curl\_easy\_setopt(3)* call or added as a multipart in another mime handle's part using *curl\_mime\_subparts(3)*.

*easy\_handle* is used for part separator randomization and error reporting. Since 7.87.0, it does not need to be the final target handle.

Using a mime handle is the recommended way to post an HTTP form, format and send a multi-part email with SMTP or upload such an email to an IMAP server.

**EXAMPLE**

```
int main(void)
{
    CURL *easy = curl_easy_init();
    curl_mime *mime;
    curl_mimepart *part;

    /* Build an HTTP form with a single field named "data", */
    mime = curl_mime_init(easy);
    part = curl_mime_addpart(mime);
    curl_mime_data(part, "This is the field data", CURL_ZERO_TERMINATED);
    curl_mime_name(part, "data");

    /* Post and send it. */
    curl_easy_setopt(easy, CURLOPT_MIMEPOST, mime);
    curl_easy_setopt(easy, CURLOPT_URL, "https://example.com");
    curl_easy_perform(easy);

    /* Clean-up. */
    curl_easy_cleanup(easy);
}
```

curl\_mime\_init(3)

libcurl

curl\_mime\_init(3)

```
    curl_mime_free(mime);  
}
```

#### **AVAILABILITY**

As long as at least one of HTTP, SMTP or IMAP is enabled. Added in 7.56.0.

#### **RETURN VALUE**

A mime struct handle, or NULL upon failure.

#### **SEE ALSO**

**curl\_mime\_addpart(3)**, **curl\_mime\_free(3)**, **curl\_mime\_subparts(3)**, **CURLOPT\_MIMEPOST(3)**