

NAME

CURLOPT_CAINFO_BLOB – Certificate Authority (CA) bundle in PEM format

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_CAINFO_BLOB,  
                          struct curl_blob *stblob);
```

DESCRIPTION

Pass a pointer to a `curl_blob` structure, which contains information (pointer and size) about a memory block with binary data of PEM encoded content holding one or more certificates to verify the HTTPS server with.

If the blob is initialized with the `flags` member of struct `curl_blob` set to `CURL_BLOB_COPY`, the application does not have to keep the buffer around after setting this.

If `CURLOPT_SSL_VERIFYPEER(3)` is zero and you avoid verifying the server's certificate, `CURLOPT_CAINFO_BLOB(3)` is not needed.

This option overrides `CURLOPT_CAINFO(3)`.

DEFAULT

NULL

PROTOCOLS

This functionality affects all TLS based protocols: HTTPS, FTPS, IMAPS, POP3S, SMTPS etc.

This option works only with the following TLS backends: GnuTLS, OpenSSL, Rustls, Schannel, mbedTLS and wolfSSL

EXAMPLE

```
#include <string.h>  
  
int main(void)  
{  
    char *strpem = "PEMDATA"; /* strpem must point to a PEM string */  
    CURL *curl = curl_easy_init();  
    if(curl) {  
        CURLcode result;  
        struct curl_blob blob;  
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com/");  
        blob.data = strpem;  
        blob.len = strlen(strpem);  
        blob.flags = CURL_BLOB_COPY;  
        curl_easy_setopt(curl, CURLOPT_CAINFO_BLOB, &blob);  
        result = curl_easy_perform(curl);  
        curl_easy_cleanup(curl);  
    }  
}
```

HISTORY

This option is supported by the mbedTLS (since 7.81.0), Rustls (since 7.82.0), wolfSSL (since 8.2.0), GnuTLS (since 8.18.0), OpenSSL and Schannel backends.

AVAILABILITY

Added in curl 7.77.0

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_CAINFO(3), CURLOPT_CAPATH(3), CURLOPT_SSL_VERIFYHOST(3), CURLOPT_SSL_VERIFYPEER(3)