

NAME

CURLOPT_CAPATH – specify directory holding CA certificates

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_CAPATH, char *capath);
```

DESCRIPTION

Pass a char * to a zero terminated string naming a directory holding multiple CA certificates to verify the peer with. If libcurl is built against OpenSSL, the certificate directory must be prepared using the openssl c_rehash utility. This makes sense only when used in combination with the *CURLOPT_SSL_VERIFYPEER(3)* option. If *CURLOPT_SSL_VERIFYPEER(3)* is zero, *CURLOPT_CAPATH(3)* need not even indicate an accessible path. The *CURLOPT_CAPATH(3)* function apparently does not work in Windows due to some limitation in openssl.

DEFAULT

NULL

PROTOCOLS

All TLS based protocols: HTTPS, FTPS, IMAPS, POP3, SMTPS etc.

EXAMPLE

TODO

AVAILABILITY

This option is OpenSSL-specific and does nothing if libcurl is built to use GnuTLS. NSS-powered libcurl provides the option only for backward compatibility.

RETURN VALUE

Returns CURLE_OK if TLS enabled, and CURLE_UNKNOWN_OPTION if not, or CURLE_OUT_OF_MEMORY if there was insufficient heap space.

SEE ALSO

CURLOPT_STDERR(3), CURLOPT_DEBUGFUNCTION(3),