

**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.

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, POP3S, SMTPS etc.

**EXAMPLE**

TODO

**AVAILABILITY**

This option is supported by the OpenSSL, GnuTLS and PolarSSL backends. The NSS backend 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\_CAINFO(3)*, *CURLOPT\_STDERR(3)*, *CURLOPT\_DEBUGFUNCTION(3)*,