NAME

curl_easy_setopt – set options for a curl easy handle

SYNOPSIS

#include <curl/curl.h>

CURLcode curl_easy_setopt(CURL *handle, CURLoption option, parameter);

DESCRIPTION

curl_easy_setopt(3) is used to tell libcurl how to behave. By setting the appropriate options, the application can change libcurl's behavior. All options are set with an *option* followed by a *parameter*. That parameter can be a **long**, a **function pointer**, an **object pointer** or a **curl_off_t**, depending on what the specific option expects. Read this manual carefully as bad input values may cause libcurl to behave badly! You can only set one option in each function call. A typical application uses many *curl_easy_setopt(3)* calls in the setup phase.

Options set with this function call are valid for all forthcoming transfers performed using this *handle*. The options are not in any way reset between transfers, so if you want subsequent transfers with different options, you must change them between the transfers. You can optionally reset all options back to internal default with *curl_easy_reset(3)*.

Strings passed to libcurl as 'char *' arguments, are copied by the library; thus the string storage associated to the pointer argument may be overwritten after *curl_easy_setopt(3)* returns. The only exception to this rule is really *CURLOPT_POSTFIELDS(3)*, but the alternative that copies the string *CURLOPT_COPY-POSTFIELDS(3)* has some usage characteristics you need to read up on.

The order in which the options are set does not matter.

Before version 7.17.0, strings were not copied. Instead the user was forced keep them available until libcurl no longer needed them.

The *handle* is the return code from a *curl_easy_init(3)* or *curl_easy_duphandle(3)* call.

BEHAVIOR OPTIONS

CURLOPT_VERBOSE
> Display verbose information. See *CURLOPT_VERBOSE(3)*

CURLOPT_HEADER
> Include the header in the body output. See *CURLOPT_HEADER(3)*

CURLOPT_NOPROGRESS
> Shut off the progress meter. See *CURLOPT_NOPROGRESS(3)*

CURLOPT_NOSIGNAL
> Do not install signal handlers. See *CURLOPT_NOSIGNAL(3)*

CURLOPT_WILDCARDMATCH
> Transfer multiple files according to a file name pattern. See *CURLOPT_WILDCARDMATCH(3)*

CALLBACK OPTIONS

CURLOPT_WRITEFUNCTION
> Callback for writing data. See *CURLOPT_WRITEFUNCTION(3)*

CURLOPT_WRITEDATA
> Data pointer to pass to the write callback. See *CURLOPT_WRITEDATA(3)*

CURLOPT_READFUNCTION
> Callback for reading data. See *CURLOPT_READFUNCTION(3)*

CURLOPT_READDATA
>   Data pointer to pass to the read callback. See *CURLOPT_READDATA(3)*

CURLOPT_IOCTLFUNCTION
>   Callback for I/O operations. See *CURLOPT_IOCTLFUNCTION(3)*

CURLOPT_IOCTLDATA
>   Data pointer to pass to the I/O callback. See *CURLOPT_IOCTLDATA(3)*

CURLOPT_SEEKFUNCTION
>   Callback for seek operations. See *CURLOPT_SEEKFUNCTION(3)*

CURLOPT_SEEKDATA
>   Data pointer to pass to the seek callback. See *CURLOPT_SEEKDATA(3)*

CURLOPT_SOCKOPTFUNCTION
>   Callback for sockopt operations. See *CURLOPT_SOCKOPTFUNCTION(3)*

CURLOPT_SOCKOPTDATA
>   Data pointer to pass to the sockopt callback. See *CURLOPT_SOCKOPTDATA(3)*

CURLOPT_OPENSOCKETFUNCTION
>   Callback for socket creation. See *CURLOPT_OPENSOCKETFUNCTION(3)*

CURLOPT_OPENSOCKETDATA
>   Data pointer to pass to the open socket callback. See *CURLOPT_OPENSOCKETDATA(3)*

CURLOPT_CLOSESOCKETFUNCTION
>   Callback for closing socket. See *CURLOPT_CLOSESOCKETFUNCTION(3)*

CURLOPT_CLOSESOCKETDATA
>   Data pointer to pass to the close socket callback. See *CURLOPT_CLOSESOCKETDATA(3)*

CURLOPT_PROGRESSFUNCTION
>   OBSOLETE callback for progress meter. See *CURLOPT_PROGRESSFUNCTION(3)*

CURLOPT_PROGRESSDATA
>   Data pointer to pass to the progress meter callback. See *CURLOPT_PROGRESSDATA(3)*

CURLOPT_XFERINFOFUNCTION
>   Callback for progress meter. See *CURLOPT_XFERINFOFUNCTION(3)*

CURLOPT_XFERINFODATA
>   Data pointer to pass to the progress meter callback. See *CURLOPT_XFERINFODATA(3)*

CURLOPT_HEADERFUNCTION
>   Callback for writing received headers. See *CURLOPT_HEADERFUNCTION(3)*

CURLOPT_HEADERDATA
>   Data pointer to pass to the header callback. See *CURLOPT_HEADERDATA(3)*

CURLOPT_DEBUGFUNCTION
>   Callback for debug information. See *CURLOPT_DEBUGFUNCTION(3)*

CURLOPT_DEBUGDATA
>   Data pointer to pass to the debug callback. See *CURLOPT_DEBUGDATA(3)*

CURLOPT_SSL_CTX_FUNCTION
>   Callback for SSL context logic. See *CURLOPT_SSL_CTX_FUNCTION(3)*

CURLOPT_SSL_CTX_DATA
>   Data pointer to pass to the SSL context callback. See *CURLOPT_SSL_CTX_DATA(3)*

CURLOPT_CONV_TO_NETWORK_FUNCTION
>   Callback for code base conversion. See *CURLOPT_CONV_TO_NETWORK_FUNCTION(3)*

CURLOPT_CONV_FROM_NETWORK_FUNCTION
> Callback for code base conversion. See *CURLOPT_CONV_FROM_NETWORK_FUNCTION(3)*

CURLOPT_CONV_FROM_UTF8_FUNCTION
> Callback for code base conversion. See *CURLOPT_CONV_FROM_UTF8_FUNCTION(3)*

CURLOPT_INTERLEAVEFUNCTION
> Callback for RTSP interleaved data. See *CURLOPT_INTERLEAVEFUNCTION(3)*

CURLOPT_INTERLEAVEDATA
> Data pointer to pass to the RTSP interleave callback. See *CURLOPT_INTERLEAVEDATA(3)*

CURLOPT_CHUNK_BGN_FUNCTION
> Callback for wildcard download start of chunk. See *CURLOPT_CHUNK_BGN_FUNCTION(3)*

CURLOPT_CHUNK_END_FUNCTION
> Callback for wildcard download end of chunk. See *CURLOPT_CHUNK_END_FUNCTION(3)*

CURLOPT_CHUNK_DATA
> Data pointer to pass to the chunk callbacks. See *CURLOPT_CHUNK_DATA(3)*

CURLOPT_FNMATCH_FUNCTION
> Callback for wildcard matching. See *CURLOPT_FNMATCH_FUNCTION(3)*

CURLOPT_FNMATCH_DATA
> Data pointer to pass to the wildcard matching callback. See *CURLOPT_FNMATCH_DATA(3)*

ERROR OPTIONS

CURLOPT_ERRORBUFFER
> Error message buffer. See *CURLOPT_ERRORBUFFER(3)*

CURLOPT_STDERR
> stderr replacement stream. See *CURLOPT_STDERR(3)*

CURLOPT_FAILONERROR
> Fail on HTTP 4xx errors. *CURLOPT_FAILONERROR(3)*

NETWORK OPTIONS

CURLOPT_URL
> URL to work on. See *CURLOPT_URL(3)*

CURLOPT_PATH_AS_IS
> Disable squashing /../ and /./ sequences in the path. See *CURLOPT_PATH_AS_IS(3)*

CURLOPT_PROTOCOLS
> Allowed protocols. See *CURLOPT_PROTOCOLS(3)*

CURLOPT_REDIR_PROTOCOLS
> Protocols to allow redirects to. See *CURLOPT_REDIR_PROTOCOLS(3)*

CURLOPT_DEFAULT_PROTOCOL
> Default protocol. See *CURLOPT_DEFAULT_PROTOCOL(3)*

CURLOPT_PROXY
> Proxy to use. See *CURLOPT_PROXY(3)*

CURLOPT_PROXYPORT
> Proxy port to use. See *CURLOPT_PROXYPORT(3)*

CURLOPT_PROXYTYPE
> Proxy type. See *CURLOPT_PROXYTYPE(3)*

CURLOPT_NOPROXY
> Filter out hosts from proxy use. *CURLOPT_NOPROXY(3)*

CURLOPT_HTTPPROXYTUNNEL
>   Tunnel through the HTTP proxy. *CURLOPT_HTTPPROXYTUNNEL(3)*

CURLOPT_SOCKS5_GSSAPI_SERVICE
>   Socks5 GSSAPI service name. *CURLOPT_SOCKS5_GSSAPI_SERVICE(3)*

CURLOPT_SOCKS5_GSSAPI_NEC
>   Socks5 GSSAPI NEC mode. See *CURLOPT_SOCKS5_GSSAPI_NEC(3)*

CURLOPT_PROXY_SERVICE_NAME
>   Proxy service name. *CURLOPT_PROXY_SERVICE_NAME(3)*

CURLOPT_SERVICE_NAME
>   SPNEGO service name. *CURLOPT_SERVICE_NAME(3)*

CURLOPT_INTERFACE
>   Bind connection locally to this. See *CURLOPT_INTERFACE(3)*

CURLOPT_LOCALPORT
>   Bind connection locally to this port. See *CURLOPT_LOCALPORT(3)*

CURLOPT_LOCALPORTRANGE
>   Bind connection locally to port range. See *CURLOPT_LOCALPORTRANGE(3)*

CURLOPT_DNS_CACHE_TIMEOUT
>   Timeout for DNS cache. See *CURLOPT_DNS_CACHE_TIMEOUT(3)*

CURLOPT_DNS_USE_GLOBAL_CACHE
>   OBSOLETE Enable global DNS cache. See *CURLOPT_DNS_USE_GLOBAL_CACHE(3)*

CURLOPT_BUFFERSIZE
>   Ask for smaller buffer size. See *CURLOPT_BUFFERSIZE(3)*

CURLOPT_PORT
>   Port number to connect to. See *CURLOPT_PORT(3)*

CURLOPT_TCP_NODELAY
>   Disable the Nagle algorithm. See *CURLOPT_TCP_NODELAY(3)*

CURLOPT_ADDRESS_SCOPE
>   IPv6 scope for local addresses. See *CURLOPT_ADDRESS_SCOPE(3)*

CURLOPT_TCP_KEEPALIVE
>   Enable TCP keep-alive. See *CURLOPT_TCP_KEEPALIVE(3)*

CURLOPT_TCP_KEEPIDLE
>   Idle time before sending keep-alive. See *CURLOPT_TCP_KEEPIDLE(3)*

CURLOPT_TCP_KEEPINTVL
>   Interval between keep-alive probes. See *CURLOPT_TCP_KEEPINTVL(3)*

CURLOPT_UNIX_SOCKET_PATH
>   Path to a Unix domain socket. See *CURLOPT_UNIX_SOCKET_PATH(3)*

NAMES and PASSWORDS OPTIONS (Authentication)

CURLOPT_NETRC
>   Enable .netrc parsing. See *CURLOPT_NETRC(3)*

CURLOPT_NETRC_FILE
>   .netrc file name. See *CURLOPT_NETRC_FILE(3)*

CURLOPT_USERPWD
>   User name and password. See *CURLOPT_USERPWD(3)*

CURLOPT_PROXYUSERPWD
>   Proxy user name and password. See *CURLOPT_PROXYUSERPWD(3)*

CURLOPT_USERNAME
>User name. See *CURLOPT_USERNAME(3)*

CURLOPT_PASSWORD
>Password. See *CURLOPT_PASSWORD(3)*

CURLOPT_LOGIN_OPTIONS
>Login options. See *CURLOPT_LOGIN_OPTIONS(3)*

CURLOPT_PROXYUSERNAME
>Proxy user name. See *CURLOPT_PROXYUSERNAME(3)*

CURLOPT_PROXYPASSWORD
>Proxy password. See *CURLOPT_PROXYPASSWORD(3)*

CURLOPT_HTTPAUTH
>HTTP server authentication methods. See *CURLOPT_HTTPAUTH(3)*

CURLOPT_TLSAUTH_USERNAME
>TLS authentication user name. See *CURLOPT_TLSAUTH_USERNAME(3)*

CURLOPT_TLSAUTH_PASSWORD
>TLS authentication password. See *CURLOPT_TLSAUTH_PASSWORD(3)*

CURLOPT_TLSAUTH_TYPE
>TLS authentication methods. See *CURLOPT_TLSAUTH_TYPE(3)*

CURLOPT_PROXYAUTH
>HTTP proxy authentication methods. See *CURLOPT_PROXYAUTH(3)*

CURLOPT_SASL_IR
>Enable SASL initial response. See *CURLOPT_SASL_IR(3)*

CURLOPT_XOAUTH2_BEARER
>OAuth2 bearer token. See *CURLOPT_XOAUTH2_BEARER(3)*

HTTP OPTIONS

CURLOPT_AUTOREFERER
>Automatically set Referer: header. See *CURLOPT_AUTOREFERER(3)*

CURLOPT_ACCEPT_ENCODING
>Accept-Encoding and automatic decompressing data. See *CURLOPT_ACCEPT_ENCODING(3)*

CURLOPT_TRANSFER_ENCODING
>Request Transfer-Encoding. See *CURLOPT_TRANSFER_ENCODING(3)*

CURLOPT_FOLLOWLOCATION
>Follow HTTP redirects. See *CURLOPT_FOLLOWLOCATION(3)*

CURLOPT_UNRESTRICTED_AUTH
>Do not restrict authentication to original host. *CURLOPT_UNRESTRICTED_AUTH(3)*

CURLOPT_MAXREDIRS
>Maximum number of redirects to follow. See *CURLOPT_MAXREDIRS(3)*

CURLOPT_POSTREDIR
>How to act on redirects after POST. See *CURLOPT_POSTREDIR(3)*

CURLOPT_PUT
>Issue a HTTP PUT request. See *CURLOPT_PUT(3)*

CURLOPT_POST
>Issue a HTTP POST request. See *CURLOPT_POST(3)*

CURLOPT_POSTFIELDS
>Send a POST with this data. See *CURLOPT_POSTFIELDS(3)*

CURLOPT_POSTFIELDSIZE
        The POST data is this big. See *CURLOPT_POSTFIELDSIZE(3)*

CURLOPT_POSTFIELDSIZE_LARGE
        The POST data is this big. See *CURLOPT_POSTFIELDSIZE_LARGE(3)*

CURLOPT_COPYPOSTFIELDS
        Send a POST with this data - and copy it. See *CURLOPT_COPYPOSTFIELDS(3)*

CURLOPT_HTTPPOST
        Multipart formpost HTTP POST. See *CURLOPT_HTTPPOST(3)*

CURLOPT_REFERER
        Referer: header. See *CURLOPT_REFERER(3)*

CURLOPT_USERAGENT
        User-Agent: header. See *CURLOPT_USERAGENT(3)*

CURLOPT_HTTPHEADER
        Custom HTTP headers. See *CURLOPT_HTTPHEADER(3)*

CURLOPT_HEADEROPT
        Control custom headers. See *CURLOPT_HEADEROPT(3)*

CURLOPT_PROXYHEADER
        Custom HTTP headers sent to proxy. See *CURLOPT_PROXYHEADER(3)*

CURLOPT_HTTP200ALIASES
        Alternative versions of 200 OK. See *CURLOPT_HTTP200ALIASES(3)*

CURLOPT_COOKIE
        Cookie(s) to send. See *CURLOPT_COOKIE(3)*

CURLOPT_COOKIEFILE
        File to read cookies from. See *CURLOPT_COOKIEFILE(3)*

CURLOPT_COOKIEJAR
        File to write cookies to. See *CURLOPT_COOKIEJAR(3)*

CURLOPT_COOKIESESSION
        Start a new cookie session. See *CURLOPT_COOKIESESSION(3)*

CURLOPT_COOKIELIST
        Add or control cookies. See *CURLOPT_COOKIELIST(3)*

CURLOPT_HTTPGET
        Do a HTTP GET request. See *CURLOPT_HTTPGET(3)*

CURLOPT_HTTP_VERSION
        HTTP version to use. *CURLOPT_HTTP_VERSION(3)*

CURLOPT_IGNORE_CONTENT_LENGTH
        Ignore Content-Length. See *CURLOPT_IGNORE_CONTENT_LENGTH(3)*

CURLOPT_HTTP_CONTENT_DECODING
        Disable Content decoding. See *CURLOPT_HTTP_CONTENT_DECODING(3)*

CURLOPT_HTTP_TRANSFER_DECODING
        Disable Transfer decoding. See *CURLOPT_HTTP_TRANSFER_DECODING(3)*

CURLOPT_EXPECT_100_TIMEOUT_MS
        100-continue timeout. See *CURLOPT_EXPECT_100_TIMEOUT_MS(3)*

CURLOPT_PIPEWAIT
        Wait on connection to pipeline on it. See *CURLOPT_PIPEWAIT(3)*

SMTP OPTIONS

      CURLOPT_MAIL_FROM

            Address of the sender. See *CURLOPT_MAIL_FROM(3)*

      CURLOPT_MAIL_RCPT

            Address of the recipients. See *CURLOPT_MAIL_RCPT(3)*

      CURLOPT_MAIL_AUTH

            Authentication address. See *CURLOPT_MAIL_AUTH(3)*

TFTP OPTIONS

      CURLOPT_TFTP_BLKSIZE

            TFTP block size. See *CURLOPT_TFTP_BLKSIZE(3)*

FTP OPTIONS

      CURLOPT_FTPPORT

            Use active FTP. See *CURLOPT_FTPPORT(3)*

      CURLOPT_QUOTE

            Commands to run before transfer. See *CURLOPT_QUOTE(3)*

      CURLOPT_POSTQUOTE

            Commands to run after transfer. See *CURLOPT_POSTQUOTE(3)*

      CURLOPT_PREQUOTE

            Commands to run just before transfer. See *CURLOPT_PREQUOTE(3)*

      CURLOPT_APPEND

            Append to remote file. See *CURLOPT_APPEND(3)*

      CURLOPT_FTP_USE_EPRT

            Use EPTR. See *CURLOPT_FTP_USE_EPRT(3)*

      CURLOPT_FTP_USE_EPSV

            Use EPSV. See *CURLOPT_FTP_USE_EPSV(3)*

      CURLOPT_FTP_USE_PRET

            Use PRET. See *CURLOPT_FTP_USE_PRET(3)*

      CURLOPT_FTP_CREATE_MISSING_DIRS

            Create missing directories on the remote server. See *CURLOPT_FTP_CREATE_MISS-ING_DIRS(3)*

      CURLOPT_FTP_RESPONSE_TIMEOUT

            Timeout for FTP responses. See *CURLOPT_FTP_RESPONSE_TIMEOUT(3)*

      CURLOPT_FTP_ALTERNATIVE_TO_USER

            Alternative to USER. See *CURLOPT_FTP_ALTERNATIVE_TO_USER(3)*

      CURLOPT_FTP_SKIP_PASV_IP

            Ignore the IP address in the PASV response. See *CURLOPT_FTP_SKIP_PASV_IP(3)*

      CURLOPT_FTPSSLAUTH

            Control how to do TLS. See *CURLOPT_FTPSSLAUTH(3)*

      CURLOPT_FTP_SSL_CCC

            Back to non-TLS again after authentication. See *CURLOPT_FTP_SSL_CCC(3)*

      CURLOPT_FTP_ACCOUNT

            Send ACCT command. See *CURLOPT_FTP_ACCOUNT(3)*

      CURLOPT_FTP_FILEMETHOD

            Specify how to reach files. See *CURLOPT_FTP_FILEMETHOD(3)*

RTSP OPTIONS

CURLOPT_RTSP_REQUEST
RTSP request. See *CURLOPT_RTSP_REQUEST(3)*

CURLOPT_RTSP_SESSION_ID
RTSP session-id. See *CURLOPT_RTSP_SESSION_ID(3)*

CURLOPT_RTSP_STREAM_URI
RTSP stream URI. See *CURLOPT_RTSP_STREAM_URI(3)*

CURLOPT_RTSP_TRANSPORT
RTSP Transport: header. See *CURLOPT_RTSP_TRANSPORT(3)*

CURLOPT_RTSP_CLIENT_CSEQ
Client CSEQ number. See *CURLOPT_RTSP_CLIENT_CSEQ(3)*

CURLOPT_RTSP_SERVER_CSEQ
CSEQ number for RTSP Server->Client request. See *CURLOPT_RTSP_SERVER_CSEQ(3)*

PROTOCOL OPTIONS

CURLOPT_TRANSFERTEXT
Use text transfer. See *CURLOPT_TRANSFERTEXT(3)*

CURLOPT_PROXY_TRANSFER_MODE
Add transfer mode to URL over proxy. See *CURLOPT_PROXY_TRANSFER_MODE(3)*

CURLOPT_CRLF
Convert newlines. See *CURLOPT_CRLF(3)*

CURLOPT_RANGE
Range requests. See *CURLOPT_RANGE(3)*

CURLOPT_RESUME_FROM
Resume a transfer. See *CURLOPT_RESUME_FROM(3)*

CURLOPT_RESUME_FROM_LARGE
Resume a transfer. See *CURLOPT_RESUME_FROM_LARGE(3)*

CURLOPT_CUSTOMREQUEST
Custom request/method. See *CURLOPT_CUSTOMREQUEST(3)*

CURLOPT_FILETIME
Request file modification date and time. See *CURLOPT_FILETIME(3)*

CURLOPT_DIRLISTONLY
List only. See *CURLOPT_DIRLISTONLY(3)*

CURLOPT_NOBODY
Do not get the body contents. See *CURLOPT_NOBODY(3)*

CURLOPT_INFILESIZE
Size of file to send. *CURLOPT_INFILESIZE(3)*

CURLOPT_INFILESIZE_LARGE
Size of file to send. *CURLOPT_INFILESIZE_LARGE(3)*

CURLOPT_UPLOAD
Upload data. See *CURLOPT_UPLOAD(3)*

CURLOPT_MAXFILESIZE
Maximum file size to get. See *CURLOPT_MAXFILESIZE(3)*

CURLOPT_MAXFILESIZE_LARGE
Maximum file size to get. See *CURLOPT_MAXFILESIZE_LARGE(3)*

CURLOPT_TIMECONDITION
>   Make a time conditional request. See *CURLOPT_TIMECONDITION(3)*

CURLOPT_TIMEVALUE
>   Time value for the time conditional request. See *CURLOPT_TIMEVALUE(3)*

CONNECTION OPTIONS

CURLOPT_TIMEOUT
>   Timeout for the entire request. See *CURLOPT_TIMEOUT(3)*

CURLOPT_TIMEOUT_MS
>   Millisecond timeout for the entire request. See *CURLOPT_TIMEOUT_MS(3)*

CURLOPT_LOW_SPEED_LIMIT
>   Low speed limit to abort transfer. See *CURLOPT_LOW_SPEED_LIMIT(3)*

CURLOPT_LOW_SPEED_TIME
>   Time to be below the speed to trigger low speed abort. See *CURLOPT_LOW_SPEED_TIME(3)*

CURLOPT_MAX_SEND_SPEED_LARGE
>   Cap the upload speed to this. See *CURLOPT_MAX_SEND_SPEED_LARGE(3)*

CURLOPT_MAX_RECV_SPEED_LARGE
>   Cap the download speed to this. See *CURLOPT_MAX_RECV_SPEED_LARGE(3)*

CURLOPT_MAXCONNECTS
>   Maximum number of connections in the connection pool. See *CURLOPT_MAXCONNECTS(3)*

CURLOPT_FRESH_CONNECT
>   Use a new connection. *CURLOPT_FRESH_CONNECT(3)*

CURLOPT_FORBID_REUSE
>   Prevent subsequent connections from re-using this. See *CURLOPT_FORBID_REUSE(3)*

CURLOPT_CONNECTTIMEOUT
>   Timeout for the connection phase. See *CURLOPT_CONNECTTIMEOUT(3)*

CURLOPT_CONNECTTIMEOUT_MS
>   Millisecond timeout for the connection phase. See *CURLOPT_CONNECTTIMEOUT_MS(3)*

CURLOPT_IPRESOLVE
>   IP version to resolve to. See *CURLOPT_IPRESOLVE(3)*

CURLOPT_CONNECT_ONLY
>   Only connect, nothing else. See *CURLOPT_CONNECT_ONLY(3)*

CURLOPT_USE_SSL
>   Use TLS/SSL. See *CURLOPT_USE_SSL(3)*

CURLOPT_RESOLVE
>   Provide fixed/fake name resolves. See *CURLOPT_RESOLVE(3)*

CURLOPT_DNS_INTERFACE
>   Bind name resolves to this interface. See *CURLOPT_DNS_INTERFACE(3)*

CURLOPT_DNS_LOCAL_IP4
>   Bind name resolves to this IP4 address. See *CURLOPT_DNS_LOCAL_IP4(3)*

CURLOPT_DNS_LOCAL_IP6
>   Bind name resolves to this IP6 address. See *CURLOPT_DNS_LOCAL_IP6(3)*

CURLOPT_DNS_SERVERS
>   Preferred DNS servers. See *CURLOPT_DNS_SERVERS(3)*

CURLOPT_ACCEPTTIMEOUT_MS
>   Timeout for waiting for the server's connect back to be accepted. See *CURLOPT_ACCEPTTIME-OUT_MS(3)*

SSL and SECURITY OPTIONS

> CURLOPT_SSLCERT
>> Client cert. See *CURLOPT_SSLCERT(3)*
>
> CURLOPT_SSLCERTTYPE
>> Client cert type. See *CURLOPT_SSLCERTTYPE(3)*
>
> CURLOPT_SSLKEY
>> Client key. See *CURLOPT_SSLKEY(3)*
>
> CURLOPT_SSLKEYTYPE
>> Client key type. See *CURLOPT_SSLKEYTYPE(3)*
>
> CURLOPT_KEYPASSWD
>> Client key password. See *CURLOPT_KEYPASSWD(3)*
>
> CURLOPT_SSL_ENABLE_ALPN
>> Enable use of ALPN. See *CURLOPT_SSL_ENABLE_ALPN(3)*
>
> CURLOPT_SSL_ENABLE_NPN
>> Enable use of NPN. See *CURLOPT_SSL_ENABLE_NPN(3)*
>
> CURLOPT_SSLENGINE
>> Use identifier with SSL engine. See *CURLOPT_SSLENGINE(3)*
>
> CURLOPT_SSLENGINE_DEFAULT
>> Default SSL engine. See *CURLOPT_SSLENGINE_DEFAULT(3)*
>
> CURLOPT_SSL_FALSESTART
>> Enable TLS False Start. See *CURLOPT_SSL_FALSESTART(3)*
>
> CURLOPT_SSLVERSION
>> SSL version to use. See *CURLOPT_SSLVERSION(3)*
>
> CURLOPT_SSL_VERIFYHOST
>> Verify the host name in the SSL certificate. See *CURLOPT_SSL_VERIFYHOST(3)*
>
> CURLOPT_SSL_VERIFYPEER
>> Verify the SSL certificate. See *CURLOPT_SSL_VERIFYPEER(3)*
>
> CURLOPT_SSL_VERIFYSTATUS
>> Verify the SSL certificate's status. See *CURLOPT_SSL_VERIFYSTATUS(3)*
>
> CURLOPT_CAINFO
>> CA cert bundle. See *CURLOPT_CAINFO(3)*
>
> CURLOPT_ISSUERCERT
>> Issuer certificate. See *CURLOPT_ISSUERCERT(3)*
>
> CURLOPT_CAPATH
>> Path to CA cert bundle. See *CURLOPT_CAPATH(3)*
>
> CURLOPT_CRLFILE
>> Certificate Revocation List. See *CURLOPT_CRLFILE(3)*
>
> CURLOPT_CERTINFO
>> Extract certificate info. See *CURLOPT_CERTINFO(3)*
>
> CURLOPT_PINNEDPUBLICKEY
>> Set pinned SSL public key . See *CURLOPT_PINNEDPUBLICKEY(3)*
>
> CURLOPT_RANDOM_FILE
>> Provide source for entropy random data. See *CURLOPT_RANDOM_FILE(3)*
>
> CURLOPT_EGDSOCKET
>> Identify EGD socket for entropy. See *CURLOPT_EGDSOCKET(3)*

CURLOPT_SSL_CIPHER_LIST
        Ciphers to use. See *CURLOPT_SSL_CIPHER_LIST(3)*

CURLOPT_SSL_SESSIONID_CACHE
        Disable SSL session-id cache. See *CURLOPT_SSL_SESSIONID_CACHE(3)*

CURLOPT_SSL_OPTIONS
        Control SSL behavior. See *CURLOPT_SSL_OPTIONS(3)*

CURLOPT_KRBLEVEL
        Kerberos security level. See *CURLOPT_KRBLEVEL(3)*

CURLOPT_GSSAPI_DELEGATION
        Disable GSS-API delegation. See *CURLOPT_GSSAPI_DELEGATION(3)*

SSH OPTIONS

CURLOPT_SSH_AUTH_TYPES
        SSH authentication types. See *CURLOPT_SSH_AUTH_TYPES(3)*

CURLOPT_SSH_HOST_PUBLIC_KEY_MD5
        MD5 of host's public key. See *CURLOPT_SSH_HOST_PUBLIC_KEY_MD5(3)*

CURLOPT_SSH_PUBLIC_KEYFILE
        File name of public key. See *CURLOPT_SSH_PUBLIC_KEYFILE(3)*

CURLOPT_SSH_PRIVATE_KEYFILE
        File name of private key. See *CURLOPT_SSH_PRIVATE_KEYFILE(3)*

CURLOPT_SSH_KNOWNHOSTS
        File name with known hosts. See *CURLOPT_SSH_KNOWNHOSTS(3)*

CURLOPT_SSH_KEYFUNCTION
        Callback for known hosts handling. See *CURLOPT_SSH_KEYFUNCTION(3)*

CURLOPT_SSH_KEYDATA
        Custom pointer to pass to ssh key callback. See *CURLOPT_SSH_KEYDATA(3)*

OTHER OPTIONS

CURLOPT_PRIVATE
        Private pointer to store. See *CURLOPT_PRIVATE(3)*

CURLOPT_SHARE
        Share object to use. See *CURLOPT_SHARE(3)*

CURLOPT_NEW_FILE_PERMS
        Mode for creating new remote files. See *CURLOPT_NEW_FILE_PERMS(3)*

CURLOPT_NEW_DIRECTORY_PERMS
        Mode for creating new remote directories. See *CURLOPT_NEW_DIRECTORY_PERMS(3)*

TELNET OPTIONS

CURLOPT_TELNETOPTIONS
        TELNET options. See *CURLOPT_TELNETOPTIONS(3)*

RETURN VALUE

CURLE_OK (zero) means that the option was set properly, non-zero means an error occurred as *<curl/curl.h>* defines. See the *libcurl-errors(3)* man page for the full list with descriptions.

If you try to set an option that libcurl doesn't know about, perhaps because the library is too old to support it or the option was removed in a recent version, this function will return *CURLE_UNKNOWN_OPTION*. If support for the option was disabled at compile-time, it will return *CURLE_NOT_BUILT_IN*.

EXAMPLE

```
CURL *curl = curl_easy_init();
if(curl) {
```

```
    CURLcode res;
    curl_easy_setopt(curl, CURLOPT_URL, "http://example.com");
    res = curl_easy_perform(curl);
    curl_easy_cleanup(curl);
  }
```

SEE ALSO
> **curl_easy_init**(3),   **curl_easy_cleanup**(3),   **curl_easy_reset**(3),   **curl_easy_getinfo**(3),
> **curl_multi_setopt**(3),