

NAME

CURLOPT_ECH – configuration for Encrypted Client Hello

SYNOPSIS

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_ECH, char *config);
```

DESCRIPTION

ECH is only compatible with TLSv1.3.

This experimental feature requires a special build of OpenSSL, as ECH is not yet supported in OpenSSL releases. In contrast ECH is supported by the latest BoringSSL, wolfSSL and Rustls-ffi releases.

There is also a known issue with using wolfSSL which does not support ECH when the HelloRetryRequest mechanism is used.

Pass a string that specifies configuration details for ECH. In all cases, if ECH is attempted, it may fail for various reasons. The keywords supported are:

false Turns off ECH.

grease Instructs client to emit a GREASE ECH extension. (The connection fails if ECH is attempted but fails.)

true Instructs client to attempt ECH, if possible, but to not fail if attempting ECH is not possible.

hard Instructs client to attempt ECH and fail if attempting ECH is not possible.

ecl:<base64-value>

If the string starts with *ecl*: then the remainder of the string should be a base64-encoded ECH-ConfigList that is used for ECH rather than attempting to download such a value from the DNS.

pn:<name>

If the string starts with *pn*: then the remainder of the string should be a DNS/hostname that is used to over-ride the *public_name* field of the ECHConfigList that is used for ECH.

The application does not have to keep the string around after setting this option.

Using this option multiple times makes the last set string override the previous ones. Set it to NULL or "false" to disable its use again.

DEFAULT

NULL, meaning ECH is disabled.

PROTOCOLS

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

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

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();

    const char *config = \
        "ecl:AED+DQA87wAgACB/RuzUCsW3uBbSFI7mzD63TUXpI8sGDTnFTbFCDpa+" \
        "CAAEAAEAAQANY292ZXIuZGVmby5pZQAA";
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_ECH, config);
        result = curl_easy_perform(curl);
    }
```

```
    curl_easy_cleanup(curl);  
  }  
}
```

AVAILABILITY

Added in curl 8.8.0

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_DOH_URL(3)