

NAME

curl_easy_unescape – URL decode a string

SYNOPSIS

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

```
char *curl_easy_unescape(CURL *curl, const char *input,  
                        int inlength, int *outlength);
```

DESCRIPTION

This function converts the URL encoded string **input** to a "plain string" and returns that in an allocated memory area. All input characters that are URL encoded (%XX where XX is a two-digit hexadecimal number) are converted to their binary versions. Although not constrained by its type, the returned data may not be altered.

If the **length** argument is set to 0 (zero), *curl_easy_unescape(3)* uses `strlen()` on **input** to find out the size.

If **outlength** is non-NULL, the function writes the length of the returned string in the integer it points to. This allows proper handling even for strings containing %00. Since this is a pointer to an *int* type, it can only return a value up to *INT_MAX* so no longer string can be returned in this parameter.

Since 7.82.0, the **curl** parameter is ignored. Prior to that there was per-handle character conversion support for some old operating systems such as TPF, but it was otherwise ignored.

You must *curl_free(3)* the returned string when you are done with it.

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        int decodelen;
        char *decoded = curl_easy_unescape(curl, "%63%75%72%6c", 12, &decodelen);
        if(decoded) {
            /* do not assume printf() works on the decoded data */
            printf("Decoded: ");
            /* ... */
            curl_free(decoded);
        }
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.15.4

RETURN VALUE

A pointer to a null-terminated string or NULL if it failed.

SEE ALSO

curl_easy_escape(3), *curl_url_get(3)*