

**NAME**

curl\_easy\_option\_next - iterate over easy\_setopt options

**SYNOPSIS**

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

```
const struct curl_easyoption *
curl_easy_option_next(const struct curl_easyoption *prev);
```

**DESCRIPTION**

This function returns a pointer to the first or the next *curl\_easyoption* struct, providing an ability to iterate over all known options for *curl\_easy\_setopt(3)* in this instance of libcurl.

Pass a **NULL** argument as **prev** to get the first option returned, or pass in the current option to get the next one returned. If there is no more option to return, *curl\_easy\_option\_next(3)* returns NULL.

The options returned by this functions are the ones known to this libcurl and information about what argument type they want.

If the **CURLLOT\_FLAG\_ALIAS** bit is set in the flags field, it means the name is provided for backwards compatibility as an alias.

**struct**

```
typedef enum {
    CURLLOT_LONG, /* long (a range of values) */
    CURLLOT_VALUES, /* (a defined set or bitmask) */
    CURLLOT_OFF_T, /* curl_off_t (a range of values) */
    CURLLOT_OBJECT, /* pointer (void *) */
    CURLLOT_STRING, /* (char * to null-terminated buffer) */
    CURLLOT_SLIST, /* (struct curl_slist *) */
    CURLLOT_CBPTR, /* (void * passed as-is to a callback) */
    CURLLOT_BLOB, /* blob (struct curl_blob *) */
    CURLLOT_FUNCTION /* function pointer */
} curl_easytype;
```

```
/* The CURLOPTTYPE_* id ranges can still be used to figure out what type/size
to use for curl_easy_setopt() for the given id */
```

```
struct curl_easyoption {
    const char *name;
    CURLLOToption id;
```

```
curl_easytype type;
unsigned int flags;
};
```

**EXAMPLE**

```
int main(void)
{
    /* iterate over all available options */
    const struct curl_easyoption *opt;
    opt = curl_easy_option_next(NULL);
    while(opt) {
        printf("Name: %s\n", opt->name);
        opt = curl_easy_option_next(opt);
    }
}
```

**AVAILABILITY**

This function was added in libcurl 7.73.0

**RETURN VALUE**

A pointer to the *curl\_easyoption* struct for the next option or NULL if no more options.

**SEE ALSO**

**curl\_easy\_option\_by\_id(3)**, **curl\_easy\_option\_by\_name(3)**, **curl\_easy\_setopt(3)**