

NAME

CURLOPT_SSH_HOSTKEYFUNCTION – callback to check host key

SYNOPSIS

#include <curl/curl.h>

```
int keycallback(void *clientp,
               int keytype,
               const char *key,
               size_t keylen);
```

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_SSH_HOSTKEYFUNCTION,
                          keycallback);
```

DESCRIPTION

Pass a pointer to your callback function, which should match the prototype shown above. It overrides *CURLOPT_SSH_KNOWNHOSTS(3)*.

This callback gets called when the verification of the SSH host key is needed.

key is **keylen** bytes long and is the key to check. **keytype** says what type it is, from the **CURLKHTYPE_*** series in the **curl_khtype** enum.

clientp is a custom pointer set with *CURLOPT_SSH_HOSTKEYDATA(3)*.

The callback must return one of the following return codes to tell libcurl how to act:

CURLKHMATCH_OK

The host key is accepted, the connection should continue.

CURLKHMATCH_MISMATCH

the host key is rejected, the connection is canceled.

DEFAULT

NULL

PROTOCOLS

This functionality affects scp and sftp

EXAMPLE

```
struct mine {
    void *custom;
};

int hostkeycb(void *clientp, /* passed with CURLOPT_SSH_HOSTKEYDATA */
             int keytype, /* CURLKHTYPE */
             const char *key, /* host key to check */
             size_t keylen) /* length of the key */
{
    /* 'clientp' points to the callback_data struct */
    /* investigate the situation and return the correct value */
    return CURLKHMATCH_OK;
}

int main(void)
{
    struct mine callback_data;
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
```

```

curl_easy_setopt(curl, CURLOPT_URL, "sftp://example.com/thisfile.txt");
curl_easy_setopt(curl, CURLOPT_SSH_HOSTKEYFUNCTION, hostkeycb);
curl_easy_setopt(curl, CURLOPT_SSH_HOSTKEYDATA, &callback_data);

result = curl_easy_perform(curl);
curl_easy_cleanup(curl);
}
}

```

NOTES

Work only with the libssh2 backend.

AVAILABILITY

Added in curl 7.84.0

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_SSH_HOSTKEYDATA(3), CURLOPT_SSH_KNOWNHOSTS(3)