NAME

CURLOPT_SSH_PUBLIC_KEYFILE − public key file for SSH auth

SYNOPSIS

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

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_SSH_PUBLIC_KEYFILE,
                          char *filename);
```

DESCRIPTION

Pass a char pointer pointing to a *filename* for your public key. If not used, libcurl defaults to **$HOME/.ssh/id_dsa.pub** if the HOME environment variable is set, and "id_dsa.pub" in the current directory if HOME is not set.

If NULL (or an empty string) is passed to this option, libcurl passes no public key to the SSH library, which then rather derives it from the private key. If the SSH library cannot derive the public key from the private one and no public one is provided, the transfer fails.

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

DEFAULT

NULL

PROTOCOLS

This functionality affects scp and sftp

EXAMPLE

```
int main(void)
{
  CURL *curl = curl_easy_init();
  if(curl) {
    CURLcode result;
    curl_easy_setopt(curl, CURLOPT_URL, "sftp://example.com/file");
    curl_easy_setopt(curl, CURLOPT_SSH_PUBLIC_KEYFILE,
                     "/home/clarkkent/.ssh/id_rsa.pub");
    result = curl_easy_perform(curl);
    curl_easy_cleanup(curl);
  }
}
```

HISTORY

The "" trick was added in 7.26.0

AVAILABILITY

Added in curl 7.16.1

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_AUTH_TYPES(3), **CURLOPT_SSH_PRIVATE_KEYFILE**(3)