



# Daniel Sternberg on 25 years of cURL

## 1. Introduction

cURL is a command line tool (curl) and library (libcurl) used to transfer data using various network protocols like DICT, FILE, FTP, FTPS, GOPHER, GOPHERS, HTTP, HTTPS or IMAP. It was developed by Daniel Stenberg<sup>1</sup> in 1997 using the programming language C. It is a free and open source<sup>2</sup> software, having more than 2000 committers that provided code, feedback, advice, etc. It is used daily in cars, television sets, routers, printers, etc. The most recent stable version is 7.87.0, released on 2022-12-21.

## 2. History

The cURL project started in the mid-90s. At that moment, Stenberg was working on an open Source project to provide an exchange service and currency rates had to be reasonably updated. For this reason, he needed a tool to download them using HTTP.

He found the tool HTTP GET to do the job, so he started working with it. However, he found some issues with the tool so he got in touch with its author and after some time Stenberg ended up being the maintainer of the tool.

Daniel Stenberg kept working on HTTP GET until he realized he wanted to extend the currency exchange service with more rates. He found other sites that hosted currency rates on Gopher and FTP, so he made the tool support both technologies. This led him to change the name of the tool to URL GET, and then to cURL, which comes from “clients for URLs” and because it was easy to pronounce.

In 2000 he remodeled its internals and provided a library, libcurl. Around that time, PHP adopted cURL as a library and, since the programming language had lots of users, they submitted lots of bugs. With this, more people started to use it and he kept on fixing bugs and adding support for more protocols (TLS, HTTPS).

Finally, cURL started to support multiple implementations for different protocols such as SSL, GNU TLS or NSS for TLS and multiple libraries like SSH, IDN, as the author wanted to let the users building cURL decide what kind of third-party libraries they wanted to use when building cURL.

## 3. LibcURL

Libcurl is a free, portable, thread-safe, client-side URL transfer library (or API) that offers support for a wide range of protocols, including FTP, HTTP, HTTPS, IMAP, and SMTP, among others. It also offers support for various features, such as HTTP form-based upload, and user-plus-password authentication, and it is compatible with IPv6 and several security protocols

It can be incorporated by users into their programs where cURL acts as an isolated wrapper for the LibcURL library.

## 4. How to contribute?

---

<sup>1</sup> Polhem Award prize in Sweden in 2017.

<sup>2</sup> Latest source code in GitHub: <https://github.com/curl/curl>.



If you want to share an idea or improvement to cURL, use the mailing lists and if you already started to write some code, you can submit a pull request. Once you do this, you should be available for the follow-up discussion; that is, if you want to add something to cURL, you must be ready to work with the team

## 5. Release cycles

Releases are done every 8 weeks. The first half of the release cycle is open for merging features and changing things, while the second half is devoted to bug fixing.

Releases are always done on Wednesdays. The feature window is open next Monday, so there are a few days for anyone to report alarming bugs. In this case, the team doesn't open the feature window and works on emergency fixes.

## 6. Benefits of using cURL

- It is highly versatile and can be used in many operating systems and devices.
- It is beneficial for evaluating endpoint functionality and determining if they are functioning properly.
- It provides detailed, verbose output of the data sent and received, making it useful for debugging purposes.
- It has strong error logging capabilities.
- It can be rate limited.

## 7. Problems: security and layers.

The developers of cURL keep a close track <sup>3</sup>of all security problems that have been reported: they document when the problem was inserted, when it was fixed and how it happened.

About half of the problems in cURL were C-related problems (eg. problems with reallocs or mallocs). Therefore, the core HTTP functionality of cURL was also implemented in Rust. It is at build time where you can select the C or Rust version.

Other important issue is that in cURL there's a lot of layers, applications, and languages. This is the reason why they spend a very long time searching for errors as there is a lot of code to figure out where the bug is.

## 8. Further information

- HTTP has been the protocol which has taken them the most time to implement.
- Reasons on why they developed cURL in C:
  - In the nineties there was no other choice than C to make a portable library or portable tool.
  - C has made it possible to really make cURL and Libcurl deportable available everywhere.
- Apple included it in macOS in 2001 becoming one of the first non-Linux operating systems that actually adopted cURL as a standard tool in their operating systems.
- It was confirmed to be used in the Mars helicopter mission in 2021.

---

<sup>3</sup> They give rewards to researchers who find security problems