

# Linux Command-curl

---

## Curl: A Powerful Networking Tool

`curl` is a command-line utility in Linux that allows you to transfer data to and from a server using HTTP, HTTPS, FTP, SCP, SFTP, and other protocols. It's an essential tool for web developers, network administrators, and anyone who needs to interact with servers.

### Basic Usage

The basic syntax of `curl` is:

```
curl [options] url
```

Here are some common options:

- `-X`: specifies the request method (e.g., GET, POST, PUT, DELETE)
- `-H`: adds a header to the request
- `-d`: sends data with the request body
- `-o`: saves the output to a file

### Example 1: Retrieving a Web Page

Get the HTML content of the Google homepage:

```
curl https://www.google.com
```

This will display the HTML code of the page in your terminal.

### Example 2: Sending Data with a POST Request

Send a simple form data to a server using the `-X` and `-d` options:

```
curl -X POST -H "Content-Type: application/x-www-form-urlencoded" \  
-d "username=john&password=secret" http://example.com/login
```

This will send a POST request with the specified form data to the server.

### Example 3: Saving Output to a File

Save the HTML content of a web page to a file called `index.html`:

```
curl https://www.example.com > index.html
```

This will save the HTML code of the page to a file named `index.html` in your current directory.

#### Example 4: Using Curl for FTP and SFTP

Use `curl` to upload a file to an FTP server:

```
curl -T localfile.txt ftp://username:password@ftp.example.com/upload/
```

This will upload the file `localfile.txt` to the specified FTP server. For SFTP, use the `-s` option instead of `-T`.

These examples demonstrate just a few of the many uses of `curl`. Its flexibility and power make it an essential tool for anyone working with web servers or networks in Linux.

---

*Curated by Brajesh Kumar*