

# CURL THAM KHẢO NHANH

HTTP request, header, xác thực, form, debug

## Sử Dụng Cơ Bản

### Request Đơn Giản

```
curl https://example.com # GET request
curl -o file.html https://url # Lưu vào file
curl -O https://url/file.tar.gz # Lưu theo tên từ server
curl -L https://url # theo redirect
```

### Flag Thường Dùng

- s** Chế độ im lặng (không hiện tiến trình)
- S** Hiện thị lỗi trong chế độ im lặng
- f** Thất bại im lặng khi lỗi HTTP
- L** Theo redirect
- o file** Ghi đầu ra vào file
- O** Lưu theo tên file từ xa
- C -** Tiếp tục tải bị gián đoạn
- max-time 30** Timeout sau 30 giây

## HTTP Methods

### GET & HEAD

```
curl https://api.example.com/users
curl -I https://example.com # HEAD (chỉ header)
curl -i https://example.com # bao gồm response header
```

### POST

```
curl -X POST https://api.example.com/users \
-H "Content-Type: application/json" \
-d '{"name": "Jo", "email": "jo@ex.com"}'
```

### PUT & PATCH & DELETE

```
curl -X PUT https://api.example.com/users/1 \
-d '{"name": "Updated"}'
curl -X PATCH https://api.example.com/users/1 \
-d '{"email": "new@ex.com"}'
curl -X DELETE https://api.example.com/users/1
```

## Headers

### Đặt Headers

```
curl -H "Content-Type: application/json" URL
curl -H "Accept: text/html" URL
curl -H "X-Custom: value" URL
curl -H "Header1: v1" -H "Header2: v2" URL
```

### Response Headers

- i** Bao gồm response header trong đầu ra
- I** Chỉ lấy header (HEAD)
- D file** Dump response header vào file
- w '%{http\_code}'** In HTTP status code

## Xác Thực

### Basic & Token Auth

```
curl -u user:pass https://api.example.com
curl -H "Authorization: Bearer TOKEN" URL
curl -u user:pass --digest URL
curl --negotiate -u :URL # Kerberos/SPNEGO
```

### Các Phương Thức Auth

- u user:pass** Xác thực Basic
- digest** HTTP Digest auth
- negotiate** Kerberos/SPNEGO auth
- ntlm** NTLM authentication
- n** Dùng thông tin xác thực từ ~/.netrc

## Dữ Liệu & Form

### Gửi Dữ Liệu

```
curl -d "key=val&key2=val2" URL # form urlencoded
curl -d @data.json URL # dữ liệu từ file
curl --data-raw '{"raw": "json"}' URL
curl --data-urlencode "q=hello world" URL
```

### Tải File Lên

```
curl -F "file=@photo.jpg" URL
curl -F "file=@doc.pdf;type=application/pdf" URL
curl -F "field=value" -F "file=@img.png" URL
```

### Multipart vs URL-Encoded

- d** application/x-www-form-urlencoded
- F** multipart/form-data
- json** Rút gọn: đặt Content-Type + Accept thành JSON
- T file** Tải file lên qua PUT

## SSL/TLS

### Tùy Chọn Certificate

```
curl --cacert ca.pem URL # CA bundle tùy chỉnh
curl --cert client.pem URL # client certificate
curl --cert client.pem --key key.pem URL
curl -k URL # bỏ qua xác minh TLS (chỉ dev)
```

### TLS Flags

- k / --insecure** Bỏ qua xác minh certificate TLS
- cacert file** Dùng CA certificate tùy chỉnh
- cert file** Client certificate
- key file** Private key của client
- tlsv1.2** Bắt buộc TLS tối thiểu 1.2
- tlsv1.3** Bắt buộc TLS tối thiểu 1.3

## Đầu Ra & Debug

### Verbose & Trace

```
curl -v URL # đầu ra verbose
curl --trace dump.txt URL # toàn bộ trace vào file
curl --trace-ascii - URL # trace ra stdout
curl -w "\n%{http_code}\n" URL # định dạng đầu ra tùy chỉnh
```

### Biến Write-Out

- %{http\_code}** HTTP response status code
- %{time\_total}** Tổng thời gian tính bằng giây
- %{time\_connect}** Thời gian thiết lập kết nối
- %{size\_download}** Byte đã tải
- %{speed\_download}** Tốc độ tải trung bình
- %{redirect\_url}** URL redirect (nếu có)

**%{ssl\_verify\_result}** Kết quả xác minh SSL (0 = OK)

### Ví Dụ Write-Out

```
curl -s -o /dev/null -w \
"code: %{http_code}\ntime: %{time_total}s\n" \
https://example.com
```

## Các Pattern Thường Gặp

### Workflow API

```
# GET JSON và pipe vào jq
curl -s https://api.example.com/data | jq '.items[]'
# POST JSON có xác thực
curl -s -H "Authorization: Bearer $TOKEN" \
--json '{"key": "val"}' https://api.example.com
```

### Các Pattern Tải

```
# Tải với progress bar
curl -# -O https://releases.example.com/v2.tar.gz
# Tiếp tục tải bị gián đoạn
curl -C - -O https://releases.example.com/v2.tar.gz
# Tải nhiều file
curl -O https://url/file1 -O https://url/file2
```

### Script Helper

```
# Kiểm tra URL có truy cập được không
curl -sf -o /dev/null https://example.com && echo OK
# Lưu cookie và tải sử dụng
curl -c cookies.txt -b cookies.txt URL
# Giới hạn tốc độ request
curl --limit-rate 100k URL
```

## Proxy & Mạng

### Cài Đặt Proxy

```
curl -x http://proxy:8080 URL
curl -x socks5://proxy:1080 URL
curl --proxy-user user:pass -x http://proxy:8080 URL
curl --noproxy "*" local,localhost URL
```

### DNS & Resolve

- resolve host:port:addr** Ép phân giải DNS thành addr
- dns-servers 8.8.8.8** Dùng DNS server tùy chỉnh
- interface eth0** Dùng network interface cụ thể
- 4 / -6** Bắt buộc IPv4 / IPv6

## Config & Nâng Cao

### File Config

```
# ~/.curlrc - tùy chọn mặc định
--silent
--location
--max-time 30
```

```
# Dùng file config tương minh
curl -K myconfig.txt URL
```

### Flag Hữu Ích

- retry 3** Thử lại khi gặp lỗi tạm thời
- retry-delay 2** Trễ giữa các lần thử lại (giây)
- compressed** Yêu cầu và giải nén gzip/br
- limit-rate 100k** Giới hạn tốc độ truyền
- Z** Truyền song song (curl 7.66+)
- create-dirs** Tạo thư mục đường dẫn cho -o