

HTTP client requests

HTTP
request

<code>method <sp> URL <sp> version <cr><lf></code>
<code>header field name: value <cr><lf></code>
<code>...</code>
<code>header field name: value <cr><lf></code>
<code><cr><lf></code>
body

HTTP client requests

method	GET	return resource
	HEAD	return headers only
	POST	send data to server (forms)
URL		relative to server (<i>e.g.</i> , /index.html)
version		1.0, 1.1, 2.0

HTTP client requests

HTTP
request

method <sp> URL <sp> version	<cr><lf>
header field name: value	<cr><lf>
...	
header field name: value	<cr><lf>
<cr><lf>	
body	

Request headers are variable length but still human readable

Uses

Authorization info

Acceptable document types/encoding

From (user email)

Host (identify the server to which the request is sent)

If-Modified-Since

Referrer (cause of the request)

User Agent (client software)

HTTP server responses

HTTP
response

<code>version</code> <sp> <code>status</code> <sp> <code>phrase</code> <cr><lf>
header field name: value <cr><lf>
...
header field name: value <cr><lf>
<cr><lf>
body

HTTP server responses

	3 digit response code		reason phrase	
Status	1XX	informational		
	2XX	success	200	OK
	3XX	redirection	301	Moved Permanently
			303	Moved Temporarily
			304	Not Modified
	4XX	client error	404	Not Found
	5XX	server error	505	Not Supported

HTTP server responses

HTTP
response

version <sp> status <sp> phrase	<cr><lf>
header field name: value	<cr><lf>
...	
header field name: value	<cr><lf>
<cr><lf>	
body	

Like request headers, response headers are of variable lengths and human-readable

Uses

Location (for redirection)

Allow (list of methods supported)

Content encoding (*e.g.*, gzip)

Content-Length

Content-Type

Expires (caching)

Last-Modified (caching)

Exercise using chrome

- Open Chrome Developer Tools window
 - Network tab
- Browse to [nytimes.com](https://www.nytimes.com)
 - How many objects were downloaded?
 - Click on an object in the list
 - What's the status code?
 - What was the request method?
 - How large was the object?
 - Request headers?
 - Response headers?

Exercise using netcat

Netcat allows us to read/write directly to TCP / HTTP tunnels

- Open a connection with `nc [host] [port] -vvv` (for verbose output)
- Issue raw HTTP requests
 - Example: `GET /index.html HTTP/1.1`
`Host: www.somesite.com`

Tasks:

- Connect to `google.com` on port 80 and request just the headers (HEAD)
- Connect to `google.com` on port 80 and request the full page (GET)
 - What headers are there?
 - Content type?
- Connect to `salmo-trutta.pschmitt.net` on port **8080** and (PUT) a file (`yourname.txt`) with your name in it

<code>method <sp> URL <sp> version</code>	<code><cr><lf></code>
<code>header field name: value</code>	<code><cr><lf></code>
...	
<code>header field name: value</code>	<code><cr><lf></code>
<code><cr><lf></code>	
body	

HTTP is a stateless protocol, meaning each request is treated independently

advantages

server-side scalability

failure handling is trivial

disadvantages

some applications **need** state!
(shopping cart, user profiles, tracking)

How can you maintain state in a stateless protocol?

HTTP makes the client maintain the state. This is what **cookies** are for



client stores small state
on behalf of the server X

client sends state
in all future requests to X

can provide authentication

Demo

nc google.com 80 -vww

Request GET / HTTP/1.1

Host: www.google.com

```
HTTP/1.1 200 OK
Date: Sat, 22 Apr 2023 19:32:03 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=ISO-8859-1
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-t5Ensfszo5YklzA9MUbD3Q' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http:;report-uri https://csp.withgoogle.com/csp/gws/other-hp
P3P: CP="This is not a P3P policy! See g.co/p3phelp for more info."
Server: gws
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Set-Cookie: 1P_JAR=2023-04-22-19; expires=Mon, 22-May-2023 19:32:03 GMT; path=/; domain=.google.com; Secure
Set-Cookie: AEC=AUEFqZeIq0yVN3iWoiTycalgcIUI5PeiKcoELP1P5xF7_x7Q0nJ2J0V; expires=Thu, 19-Oct-2023 19:32:03 GMT; path=/; domain=.google.com; Secure; HttpOnly; SameSite=lax
Set-Cookie: NID=511=IABJQPAY9XTAFpI0pu0LY7rmzd_DxEUou7p7vy8Wrb9T8EQcBSC1qKfszJdVq1k0b8mHNVoxmeG9KHVKH1kNCm3JFXim5yUnbeRvy93rMVSrspbLwlpamaceGZ_GPItqhXhkzc0jZXFcfG-cYlt-RFTMPo4iL3gG0x_mi_D6g; expires=Sun, 22-Oct-2023 19:32:03 GMT; path=/; domain=.google.com; HttpOnly
Accept-Ranges: none
Vary: Accept-Encoding
Transfer-Encoding: chunked
```

Demo

nc google.com 80 -vww

Request GET / HTTP/1.1

Host:

Browser will relay this value in subsequent requests

```
HTTP/1.1 200 OK
Date: Sat, 22 Apr 2023 19:32:03 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=ISO-8859-1
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-t5Ensfszo5YkIzA9MUbD3Q' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http:;report-uri https://csp.withgoogle.com/csp/gws/other-hp
P3P: CP="This is not a P3P policy! See g.co/p3phelp for more info."
Server: gws
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Set-Cookie: IP_JAR=2023-04-22-19; expires=Mon, 22-May-2023 19:32:03 GMT; path=/; domain=.google.com; Secure
Set-Cookie: AEC=AUEFqZeIq0yVN3iWoiTycalgcIUI5PeiKcoELP1P5xF7_x7Q0nJ2J0V; expires=Thu, 19-Oct-2023 19:32:03 GMT; path=/; domain=.google.com; Secure; HttpOnly; SameSite=lax
Set-Cookie: NID=511=IABJQPay9XTAFpI0pu0LY7rmzd_DxEUou7p7vy8Wrb9T8EQcBSC1qKfszJdVq1k0b8mHNVoxmeG9KHVKH1kNCm3JFXim5yUnbeRvy93rMVSrspbLwlpamaceGZ_GPItqhxhkzc0jZXFcfg-cYlt-RFTMPo4iL3gG0x_mi_D6g; expires=Sun, 22-Oct-2023 19:32:03 GMT; path=/; domain=.google.com; HttpOnly
Accept-Ranges: none
Vary: Accept-Encoding
Transfer-Encoding: chunked
```