HTTP

Getting What We Ask For

- HTTP: How Browsers & Servers Communicate

  - HTTP 1.1 - http://www.w3.org/Protocols/

- TCP Connection, usually over port 80

- Text Based Instructions

- Simple Verbs

  - GET, POST, PUT, DELETE, HEAD, CONNECT, OPTIONS, TRACE

- Optional Headers

- Basic GET Example

```
GET / HTTP/1.1
Host: www.example.com
```

- HOST header is required for HTTP/1.1

- Two CRLF to indicate the request has finished

  - CRLF = \r\n   Although most Web Servers will accept \n

"Although the line terminator for the start-line and header fields is
the sequence CRLF, a recipient MAY recognize a single LF as a line
terminator and ignore any preceding CR."

http://tools.ietf.org/html/rfc7230#section-3.5

- Verbs and HTTP versions **are** Case Sensitive

```
get / HTTP/1.1
host: example.com

HTTP/1.1 501 Not Implemented
```
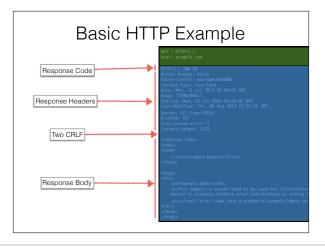
```
get / http/1.1
host: example.com

HTTP/1.0 505 HTTP Version Not Supported
```

- Headers **are not** Case Sensitive

```
GET / HTTP/1.1
hoSt: exAMPle.cOm

HTTP/1.1 200 OK
```

Basic HTTP Example

```
GET / HTTP/1.1
host: example.com                                              Request

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: max-age=604800
Content-Type: text/html
Date: Mon, 21 Jul 2014 05:04:02 GMT
Etag: "359670651"
Expires: Mon, 28 Jul 2014 05:04:02 GMT
Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT
Server: ECS (cpm/F858)
X-Cache: HIT
x-ec-custom-error: 1
Content-Length: 1270
                                                               Response
<!doctype html>
<html>
<head>
    <title>Example Domain</title>
</head>

<body>
<div>
    <h1>Example Domain</h1>
    <p>This domain is established to be used for illustrative examples in documents. You may use this
    domain in examples without prior coordination or asking for permission.</p>
    <p><a href="http://www.iana.org/domains/example">More information...</a></p>
</div>
</body>
</html>
```

Basic HTTP Example

- Response Code
- Response Headers
- Two CRLF
- Response Body

```
GET / HTTP/1.1
host: example.com

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: max-age=604800
Content-Type: text/html
Date: Mon, 21 Jul 2014 05:04:02 GMT
Etag: "359670651"
Expires: Mon, 28 Jul 2014 05:04:02 GMT
Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT
Server: ECS (cpm/F858)
X-Cache: HIT
x-ec-custom-error: 1
Content-Length: 1270

<!doctype html>
<html>
<head>
    <title>Example Domain</title>
</head>

<body>
<div>
    <h1>Example Domain</h1>
    <p>This domain is established to be used for illustrative
    domain in examples without prior coordination or asking f
    <p><a href="http://www.iana.org/domains/example">More inf
</div>
</body>
</html>
```

Command Line Basics

Program Name   **Program Arguments**

> telnet example.com 80

Command Prompt
(don't type this part)

Host   Port

Press the **return** key
at the end to run the program
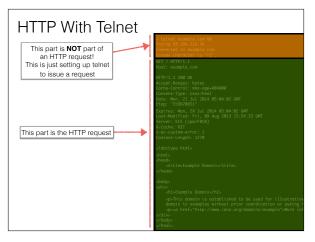
HTTP With Telnet

- **telnet** is a very simple program that basically opens a TCP connection to a host

- Key parts: **host** and **port**

```
                markfischer — telnet — 67×18
~ ⚡telnet example.com 80
Trying 93.184.216.34...
Connected to example.com.
Escape character is '^]'.
```

HTTP With Telnet

We typed in this stuff

Local **telnet** program prints this

Remote server sends this back

```
> telnet example.com 80
Trying 93.184.216.34...
Connected to example.com.
Escape character is '^]'.
GET / HTTP/1.1
host: example.com

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: max-age=604800
Content-Type: text/html
Date: Mon, 21 Jul 2014 05:04:02 GMT
Etag: "359670651"
Expires: Mon, 28 Jul 2014 05:04:02 GMT
Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT
Server: ECS (cpm/F858)
X-Cache: HIT
x-ec-custom-error: 1
Content-Length: 1270

<!doctype html>
<html>
<head>
    <title>Example Domain</title>
</head>

<body>
<div>
    <h1>Example Domain</h1>
    <p>This domain is established to be used for illustrative
    domain in examples without prior coordination or asking
    <p><a href="http://www.iana.org/domains/example">More inf
</div>
</body>
</html>
```

HTTP With Telnet

This part is **NOT** part of an HTTP request! This is just setting up telnet to issue a request

This part is the HTTP request

```
> telnet example.com 80
Trying 93.184.216.34...
Connected to example.com.
Escape character is '^]'.
GET / HTTP/1.1
host: example.com

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: max-age=604800
Content-Type: text/html
Date: Mon, 21 Jul 2014 05:04:02 GMT
Etag: "359670651"
Expires: Mon, 28 Jul 2014 05:04:02 GMT
Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT
Server: ECS (cpm/F858)
X-Cache: HIT
x-ec-custom-error: 1
Content-Length: 1270

<!doctype html>
<html>
<head>
    <title>Example Domain</title>
</head>

<body>
<div>
    <h1>Example Domain</h1>
    <p>This domain is established to be used for illustrative
    domain in examples without prior coordination or asking
    <p><a href="http://www.iana.org/domains/example">More inf
</div>
</body>
</html>
```

Telnet Example

Telnet On a Mac

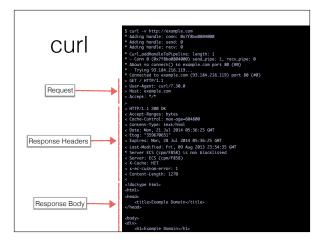- Applications folder → Utilities Folder

  - Terminal

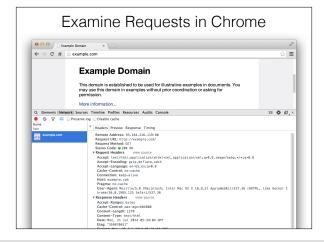- Just type in **telnet example.com 80** and hit return

Telnet On a Windows

- Windows ships with telnet, but its disabled.

- Search: "`enable telnet on windows X`"

  - where `X` is your version of Windows.

- Here's a great explanation for Windows 10

  - https://www.rootusers.com/how-to-enable-the-telnet-client-in-windows-10/

Telnet On Linux

- If you're running linux on your desktop, you already know how to use telnet. 😎

curl

```
$ curl -v http://example.com
* Adding handle: conn: 0x7f8ba0804000
* Adding handle: send: 0
* Adding handle: recv: 0
* Curl_addHandleToPipeline: length: 1
* - Conn 0 (0x7f8ba0804000) send_pipe: 1, recv_pipe: 0
* About to connect() to example.com port 80 (#0)
*   Trying 93.184.216.119...
* Connected to example.com (93.184.216.119) port 80 (#0)
> GET / HTTP/1.1
> User-Agent: curl/7.30.0
> Host: example.com
> Accept: */*
>
< HTTP/1.1 200 OK
< Accept-Ranges: bytes
< Cache-Control: max-age=604800
< Content-Type: text/html
< Date: Mon, 21 Jul 2014 05:36:25 GMT
< Etag: "359670651"
< Expires: Mon, 28 Jul 2014 05:36:25 GMT
< Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT
* Server ECS (cpm/F858) is not blacklisted
< Server: ECS (cpm/F858)
< X-Cache: HIT
< x-ec-custom-error: 1
< Content-Length: 1270
<
<!doctype html>
<html>
<head>
    <title>Example Domain</title>
</head>

<body>
<div>
    <h1>Example Domain</h1>
```

Request

Response Headers

Response Body

curl Example

Examine Requests in Chrome

Response Codes

- Informational: 1xx

- Successful: 2xx

  - 200 OK

- Redirection: 3xx

  - 301 Moved

- Client Error: 4xx

  - 404 Not Found

- Server Error: 5xx

  - 500 Internal Server Error

  http://tools.ietf.org/html/rfc7231#page-4

HTTP/2.0

- New binary method of allowing multiple requests through a single TCP socket

- More of a change to how the protocol is implemented on the wire than in the concepts of how the protocol works

- Advanced topic, if you're interested in more details:

  - http://http2-explained.readthedocs.org/en/latest/src/http2protocol.html

- Otherwise, just know its a thing