HTTP in Action

Some experiments using HTTP

What is my browser sending?

Start a dumb TCP server listening on port 80

```
    netcat -l 80
```

- Note: you must be *root* (*admin*) to use port 80. Why? For non-root, use a different port (try 8080).

- "netcat" (nc) is a Linux (and Unix) command.
  There is a version for Windows. Or get "ncat" (newer).

- If you want to send request from a *different* host, then you need this host's IP address (`ifconfig`) and make sure there is no firewall blocking tcp port 80.

What is my browser sending?

Now, open a web browser and send any request:

`http://localhost/make-my-day`

or:

`http://localhost:8080/make-my-day`

if using port 8080.

What did the server receive?

The "nc" console should print something like this:

```
GET /make-my-day HTTP/1.1
Host: localhost
Connection: keep-alive
User-Agent: Mozilla/5.0 (X11; Linux x86_64)
  AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/69.0.3497.81 Safari/537.36
Accept:
  text/html,application/xhtml+xml,application/xml;
  q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9,th;q=0.8
```

The Browser is Waiting for a Reply

You can see the browser is waiting for a reply.

We will use "nc" to send a reply.

We are a human web server!

Send HTML Reply from "nc"

In the "nc" window, send a reply to the browser.

We must reply using the HTTP protocol.

The body of the reply is HTML. Type:

```
HTTP/1.1 200 OK              <-- status code
Content-type: text/html
                             <--blank line
<h1>Hello, Nerd</h1>
```

(you can type more stuff and it will be appended to the browser page. Text *really* should start with `<html>`.)

What does a *real* web server send?

Now we know what a request from a web browser looks like.

What does a *real* reply from a *real* web server look like?

View Reply from a Web Server

Send an http request for http://www.yahoo.com/news/

```
    > nc www.yahoo.com 80   (or telnet or ncat)
    GET /news/ HTTP/1.1
    Host: www.yahoo.com
                                (a blank line)
```

- Another way: **curl -v http://www.yahoo.com/news/**, curl can also use https.

Redirect!

The Web is switching to **`https`** everywhere.
Yahoo sends back:

```
HTTP/1.1 301 Moved Permanently
Location: https://www.yahoo.com/news/
Content-Type: text/html
(more header lines)
```

- 301 is the HTTP Status Code.
- 3xx codes mean "moved" or "redirect".
- Most browsers will automatically go to new URL.

Can We Redirect a Browser?

1. First get your host's IP address (ifconfig or ipconfig)
2. Start "nc" and wait for connection:

```
nc -l 80
```

3. Someone else connects to your server using browser:

```
http://158.108.216.99
```

4. On "nc" console, tell the browser to go away:

```
HTTP/1.1 301 Don't Bother Me
Location: https://www.ku.ac.th
 (blank line)
may also need to enter Ctrl-D (EOF)
```

Did the Browser go to www.ku.ac.th?

It should follow 301 Redirect to new Location.

On some servers, you can also send a message in the response body. In case browser doesn't follow redirect.

```
HTTP/1.1 301 Sorry, humans not allowed
Location: https://www.facebook.com
Content-type: text/plain

Only bots allowed.
Try Facebook instead. :-)
```

Redirect inside a Web Page?

In some situations you may want to use redirect inside a web page.

You can also add a delay and display a text message:

```
<html>
<head>
<meta http-equiv="refresh"
        content="5; URL='https://web.facebook.com'" />
</head>
<h2>You will be redirected in 5 seconds.</h2>
</html>
```

Exercise

How many HTTP requests are sent to show this page?

```
<HTML>
<link rel="stylesheet" href="stylesheet.css">
<!-- Bootstrap makes my page look cool. -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/
   css/bootstrap.min.css">
<BODY>
<h1>My vacation</h1>
<p>
For vacation we went to <a
   href=”http://www.unseen.com/bangkok”>Bangkok</a>.
We visited <em>Wat Phra Kaeo</em>, and took this photo:
<br/>
<IMG src="images/watprakaew.jpeg" alt="Wat Phra Keao"/>
```

How Many Requests to Load Yahoo.com?

Use browser developer tools to count requests.

Chrome -> More Tools -> Developer Tools -> Network

Use *Incognito* or *Private Browsing* window to avoid caching. Also disable request filtering (UBlock Origin).

Goto `https://www.yahoo.com`

How many requests?   How many seconds?

For Just one web page!

Why Care About Http?

Its the protocol used by web apps and web services.

We need to understand response codes.

For professional web development, knowing the details matters.

Performance matters, too.