

COM SCI 118
Computer Network Fundamentals
Winter 2017

Project 1: Web Server Implementation using BSD Sockets

Shannon Tom
Michelle Tran

Design

The server supports HTML, GIF, and JPEG files and implements persistent HTTP. The server prints each received HTTP request message onto the terminal and responds with an HTTP response message accordingly. The HTTP response message is also printed to the terminal for debugging purposes. The server continues to listen through the TCP socket until manually terminated by `Ctrl-C`.

In a high level overview, the program first establishes a TCP connection via socket and accepts the connection once from the browser. The server then reads the HTTP request from the socket and processes the request to get basic information involving the specified file and HTTP protocol. The server then forms the response message and writes the file data to the socket.

The error response message 404 is returned when the specified file could not be opened or was not found. The error response message 415 is returned when a file that is not HTML, JPEG/JPG, or GIF is requested. The error response message 505 is returned when the requested protocol is not HTTP/1.0 or HTTP/1.1.

Difficulties

The main issue that we ran into while developing the server was parsing the HTTP request message. We saved the three terms in the first line of the request into separate, statically allocated character buffers and appended them to the HTTP response message. The problem that arose was that the unused bytes in the buffers were not all NULL bytes, and caused errors in interpretation of the command, HTTP version, and file name. To fix this, we used `memset()` after declaring the buffers to set all bytes to NULL.

While we were testing our code, an HTTP request message for a file named 'favicon.ico' came after the initial HTTP request message for an HTML file. The HTML file we tested with in this instance was saved off the Internet, and since ICO files were not supported by this server, an 415 error response message was returned after the Web browser (Mozilla Firefox) displayed the HTML file without the icon.

Manual

Compilation:

The server executable can be compiled by running 'make' into the terminal. All files (HTML, GIF, and JPEG) as well as source code and Makefile must be stored in the same directory. The command 'make' runs the recipe `gcc -g -o server server.c`

Running the Executable:

To run the server, type '`./server <portno>`' into the terminal, where portno is an unreserved portnumber. Reserved port numbers are 0-1024. We used 12345, 12346, and 12347 to test our code. To terminate the server, type `Ctrl-C`.

Connecting to the Server:

To connect to the server, open a browser (we used Firefox) on the same machine. Request a file from the server using the URL: `http://localhost:<portNumber>/<filename>`

The filename is the path to the file from the folder the executable is in. Only HTML, JPEG/JPG, and GIF file types are supported for this server.

Sample Outputs

The following are HTTP request messages for HTML, GIF, and JPEG files.

HTML:

```
cs118@ubuntu:~/Downloads$ ./server 12348
GET /index.html HTTP/1.1
Host: localhost:12348
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:7.0.1) Gecko/20100101 Firefox/7.0.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: keep-alive
```

GIF:

```
cs118@ubuntu:~/Downloads$ ./server 12345
GET /cat.gif HTTP/1.1
Host: localhost:12345
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:7.0.1) Gecko/20100101 Firefox/7.0.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: keep-alive
```

JPEG:

```
GET /doggy.jpeg HTTP/1.1
Host: localhost:12345
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:7.0.1) Gecko/20100101 Firefox/7.0.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: keep-alive
```

The following are HTTP response messages and error codes.

200 OK response:

```
HTTP/1.1 200 OK
Date: Sat, 04 Feb 2017 04:17:05 GMT
Server: ubuntu/3.0.0-12-generic (Linux)
Last-Modified: Wed, 01 Feb 2017 08:28:11 GMT
Content-Length: 1011425
Content-Type: image/gif
Connection: keep-alive
```

404 Not Found error:

```
HTTP/1.1 404 Not Found
Date: Sat, 04 Feb 2017 02:18:27 GMT
Server: ubuntu/3.0.0-12-generic (Linux)
Content-Length: 13
Connection: keep-alive
```

415 Unsupported Media Type error:

```
HTTP/1.1 415 Unsupported Media Type
Date: Sat, 04 Feb 2017 04:29:49 GMT
Server: ubuntu/3.0.0-12-generic (Linux)
Content-Length: 26
Connection: keep-alive
```