

Lab 7

Aditya Jain

The screenshot shows the Visual Studio Code editor with the file explorer on the left. The file explorer shows a project structure with folders like 'lab4', 'lab5', 'lab6', 'lab7', and files like 'client.c', 'input.txt', 'output.txt', 'prigame.cpp', 'rabbit.cpp', 'server.c', and 'temp.c'. The main editor window shows the 'client.c' file with the following code:

```
181     exit(1);
182 }
183 int i = 4;
184 while(receive_msg[i-4] != '\r' || receive_msg[i-3] != '\n' || receive_msg[i-2] != '\r' || receive_msg[i-1] != '\n')
185     i++;
186 fwrite(receive_msg, i, count, f);
187 receive_msg[i] = '\0';
188 printf("HTTP response header:\n\n", receive_msg);
189 }
190 while((count = SSL_read(ssl, receive_msg, 100000)) > 0){
191     fwrite(receive_msg, count, 1, f);
192 }
193 printf("reply received.\n");
194 fclose(f);
195 SSL_CTX_free(ctx);
196 }
```

The terminal window at the bottom shows the following output:

```
lab7$ gcc -Wall -o client client.c -l/usr/lib -lssl -lcrypto
lab7$ ./client https://music.youtube.com/img/on_platform_logo_dark.svg
Successfully connected with server
SSL connected.
HTTP response header:

HTTP/1.1 200 OK
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Type: image/svg+xml
Cross-Origin-Resource-Policy: cross-origin
Content-Length: 1644
Date: Mon, 05 Apr 2021 11:42:21 GMT
Expires: Mon, 05 Apr 2021 11:42:21 GMT
Cache-Control: public, max-age=0
Last-Modified: Mon, 31 Aug 2020 17:45:00 GMT
X-Content-Type-Options: nosniff
Server: sffe
X-XSS-Protection: 0
Alt-Svc: h3-29=":443"; ma=2592000,h3-T051=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q046=":443"; ma=2592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=2592000; v="46",
```

The screenshot shows the Visual Studio Code editor with the file explorer on the left. The file explorer shows a project structure with folders like 'lab4', 'lab5', 'lab6', 'lab7', and files like 'client.c', 'input.txt', 'output.txt', 'prigame.cpp', 'rabbit.cpp', 'server.c', and 'temp.c'. The main editor window shows the 'on_platform_logo_dark.svg' file with the following code:

```
1 <svg width="80" height="24" fill="none" xmlns="http://www.w3.org/2000/svg"><ellipse cx="12.18" cy="12" rx="12.18" ry="12" fill="none" /></svg>
```

The terminal window at the bottom shows the following output:

```
lab7$ gcc -Wall -o client client.c -l/usr/lib -lssl -lcrypto
lab7$ ./client https://music.youtube.com/img/on_platform_logo_dark.svg
Successfully connected with server
SSL connected.
HTTP response header:

HTTP/1.1 200 OK
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Type: image/svg+xml
Cross-Origin-Resource-Policy: cross-origin
Content-Length: 1644
Date: Mon, 05 Apr 2021 11:42:21 GMT
Expires: Mon, 05 Apr 2021 11:42:21 GMT
Cache-Control: public, max-age=0
Last-Modified: Mon, 31 Aug 2020 17:45:00 GMT
X-Content-Type-Options: nosniff
Server: sffe
X-XSS-Protection: 0
Alt-Svc: h3-29=":443"; ma=2592000,h3-T051=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q046=":443"; ma=2592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=2592000; v="46",
```

File Edit Selection View Go Run Terminal Help

client.c - Documents | WSL: Ubuntu-20.04 - Visual Studio Code

EXPLORER

- OPEN EDITORS
- DOCUMENTS
- .vscode
- CC
- CN
 - lab4
 - lab5
 - lab6
 - lab7
 - client.c
 - client.c
 - googlelogo_color_272x9...
 - on_platform_logo_dark.s...
 - README
 - midsem
- opti
- client.c
- input.txt
- interview.cpp
- output.txt
- prigame.cpp
- rabbit.cpp
- server.c
- temp.c

client.c

```
CN > lab7 > C: client.c > newHTTPS(char [], char [], int)
181         exit(1);
182     }
183     int i = 4;
184     while(receive_msg[i-4] != '\n' || receive_msg[i-3] != '\n' || receive_msg[i-2] != '\r' || receive_msg[i-1] != '\n')
185         i++;
186
187     fwrite(receive_msg+i, count-i, 1, f);
188     receive_msg[i] = '\0';
189     printf("HTTP response header:\n\n%s", receive_msg);
190
191     while((count = SSL_read(ssl, receive_msg, 100000)) > 0){
192         fwrite(receive_msg, count, 1, f);
193     }
194
195     printf("main loop finished\n\n");
196 }
```

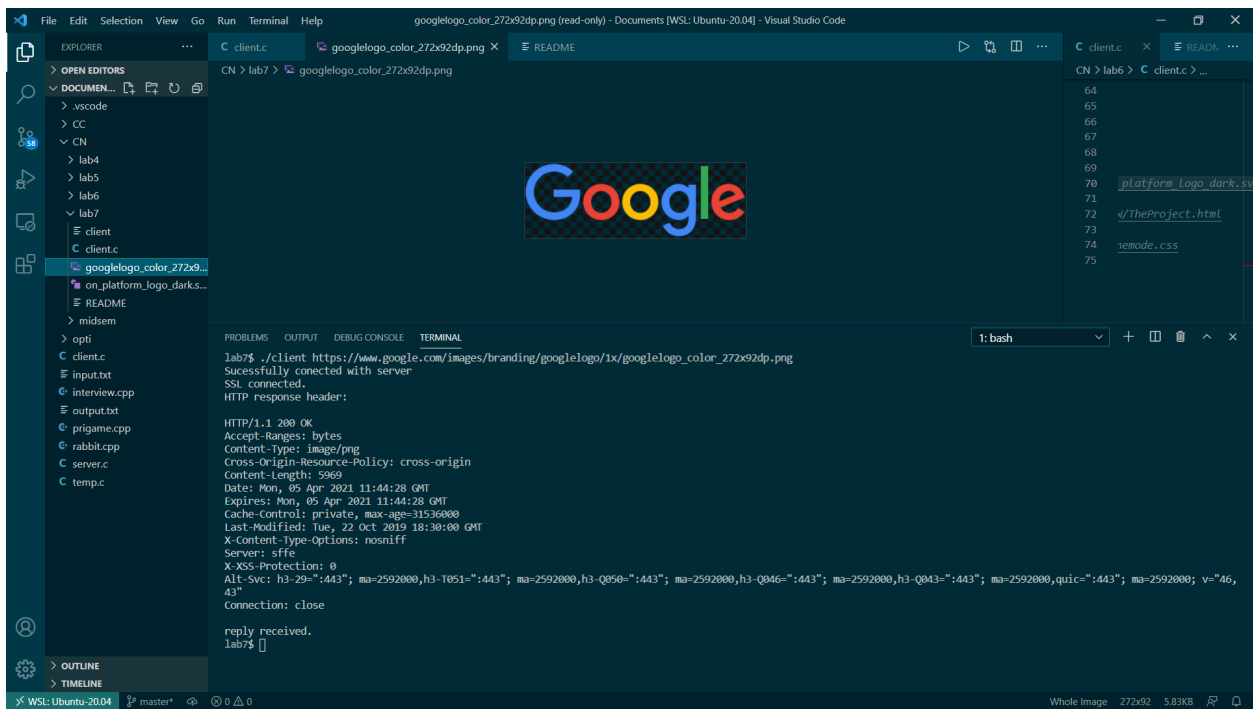
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: bash

```
lab7$ ./client https://www.google.com/images/branding/googlelogo/1x/googlelogo_color_272x92dp.png
Successfully connected with server
SSL connected.
HTTP response header:

HTTP/1.1 200 OK
Accept-Ranges: bytes
Content-Type: image/png
Cross-Origin-Resource-Policy: cross-origin
Content-Length: 5969
Date: Mon, 05 Apr 2021 11:44:28 GMT
Expires: Mon, 05 Apr 2021 11:44:28 GMT
Cache-Control: private, max-age=31536000
Last-Modified: Tue, 22 Oct 2019 18:30:00 GMT
X-Content-Type-Options: nosniff
Server: sffe
X-XSS-Protection: 0
Alt-Svc: h3-29=":443"; ma=2592000,h3-1051=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q046=":443"; ma=2592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=2592000; v=46,
43"
Connection: close

reply received.
lab7$
```



The screenshot shows the Visual Studio Code interface with a C client program open in the editor. The program is located in the file explorer under the 'CN' directory. The code is as follows:

```
181 exit(1);
182 }
183 int i = 4;
184 while(receive_msg[i-4] != '\r' || receive_msg[i-3] != '\n' || receive_msg[i-2] != '\r' || receive_msg[i-1] != '\n')
185 i++;
186
187 fwrite(receive_msg, i, count, f);
188 receive_msg[i] = '\0';
189 printf("HTTP response header:\n\n%s", receive_msg);
190 }
191
192 while((count = SSL_read(ssl, receive_msg, 100000)) > 0){
193 fwrite(receive_msg, count, i, f);
194 }
195 }
```

The terminal output shows the following:

```
lab7$ ./client http://info.cern.ch/hypertext/WWW/TheProject.html
Successfully connected with server
HTTP response header:

HTTP/1.1 200 OK
Date: Mon, 05 Apr 2021 11:46:06 GMT
Server: Apache
Last-Modified: Thu, 03 Dec 1992 08:37:20 GMT
ETag: "8a9-291e721905000"
Accept-Ranges: bytes
Content-Length: 2217
Connection: close
Content-Type: text/html

reply received.
lab7$
```

The screenshot shows the Visual Studio Code interface with the HTML content of 'TheProject.html' open in the editor. The code is as follows:

```
1 <HEADER>
2 <TITLE>The World Wide Web project</TITLE>
3 <H1>The World Wide Web project</H1>
4 </HEADER>
5 <BODY>
6 <H1>World Wide Web</H1>The WorldWideWeb (W3) is a wide-area&ltA
7 NAME=0 HREF="whatIs.html">
8 hypermedia</A> information retrieval
9 initiative aiming to give universal
10 access to a large universe of documents.&ltP>
11 Everything there is online about
12 W3 is linked directly or indirectly
13 to this document, including an &ltA
14 NAME=24 HREF="Summary.html">executive
15 summary</A> of the project, &ltA
16 NAME=29 HREF="Administration/Mailing/Overview.html">Mailing lists</A>
17 , &ltA
18 NAME=30 HREF="Policy.html">Policy</A> , November's &ltA
19 NAME=34 HREF="News/9211.html">W3 news</A> ,
20 &ltA
21 NAME=41 HREF="FAQ/List.html">Frequently Asked Questions</A> .
22 <DL>
23 <DT><A
24 NAME=44 HREF="DataSources/Top.html">What's out there?</A>
```

The terminal output shows the following:

```
lab7$ ./client http://info.cern.ch/hypertext/WWW/TheProject.html
Successfully connected with server
HTTP response header:

HTTP/1.1 200 OK
Date: Mon, 05 Apr 2021 11:46:06 GMT
Server: Apache
Last-Modified: Thu, 03 Dec 1992 08:37:20 GMT
ETag: "8a9-291e721905000"
Accept-Ranges: bytes
Content-Length: 2217
Connection: close
Content-Type: text/html

reply received.
lab7$
```

The screenshot shows the Visual Studio Code interface with a C++ client program in `client.c`. The program is designed to connect to a server via HTTPS and receive a response. The code includes headers for `newHTTPS`, `char`, and `int`. It defines a `newHTTPS` function that takes a character array and an integer, and returns an integer. The main function calls `newHTTPS` with the URL `https://sdjfsjd.sflkasjd.com/jasldfhsjkd/asjdfj.pdf`. The output of the program is displayed in the terminal, showing the error message: `No such url exist. No such host. Exiting, Bye!`

```
181     exit(1);
182 }
183 int i = 4;
184 while(receive_msg[i-4] != '\r' || receive_msg[i-3] != '\n' || receive_msg[i-2] != '\r' || receive_msg[i-1] != '\n')
185     i++;
186
187 fwrite(receive_msg, i, count, f);
188 receive_msg[i] = '\0';
189 printf("HTTP response header:\n%s", receive_msg);
190 }
191
192 while((count = SSL_read(ssl, receive_msg, 100000)) > 0){
193     fwrite(receive_msg, count, 1, f);
194 }
195
196 printf("reply received.\n");
197 fclose(f);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
lab7$ ./client https://sdjfsjd.sflkasjd.com/jasldfhsjkd/asjdfj.pdf
No such url exist. No such host. Exiting, Bye!
lab7$
```

The screenshot shows the Visual Studio Code interface with a C++ client program in `client.c`. The program is designed to connect to a server via HTTPS and receive a response. The code includes headers for `newHTTPS`, `char`, and `int`. It defines a `newHTTPS` function that takes a character array and an integer, and returns an integer. The main function calls `newHTTPS` with the URL `https://github.com/sjdfhas/asfdsah.pdf`. The output of the program is displayed in the terminal, showing the error message: `HTTP/1.1 404 Not Found`. The output also includes the server's response headers and the client's request headers.

```
181     exit(1);
182 }
183 int i = 4;
184 while(receive_msg[i-4] != '\r' || receive_msg[i-3] != '\n' || receive_msg[i-2] != '\r' || receive_msg[i-1] != '\n')
185     i++;
186
187 fwrite(receive_msg, i, count, f);
188 receive_msg[i] = '\0';
189 printf("HTTP response header:\n%s", receive_msg);
190 }
191
192 while((count = SSL_read(ssl, receive_msg, 100000)) > 0){
193     fwrite(receive_msg, count, 1, f);
194 }
195
196 printf("reply received.\n");
197 fclose(f);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
lab7$ ./client https://github.com/sjdfhas/asfdsah.pdf
Successfully connected with server
SSL connected.
Didn't got a 200 OK response. Following is header received. Exiting Bye!!

HTTP/1.1 404 Not Found
Server: GitHub.com
Date: Mon, 05 Apr 2021 11:48:17 GMT
Content-Type: text/html; charset=utf-8
Vary: X-PJAX, Accept-Encoding, Accept, X-Requested-With
Cache-Control: no-cache
Strict-Transport-Security: max-age=31536000; includeSubdomains; preload
X-Frame-Options: deny
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Referrer-Policy: origin-when-cross-origin, strict-origin-when-cross-origin
Expect-CT: max-age=2592000, report-uri="https://api.github.com/_private/browser/errors"
Content-Security-Policy: default-src 'none'; base-uri 'self'; block-all-mixed-content; connect-src 'self' uploads.github.com www.githubstatus.com collector.githubapp.com api.github.com github-cloud.s3.amazonaws.com github-production-repository-file-5c1aeb.s3.amazonaws.com github-production-upload-manifest-file-7dce7.s3.amazonaws.com github-production-user-asset-62104f.s3.amazonaws.com cdn.optimizely.com tags.optimizely.com/v1/events wss://alive.github.com/ font-src github.githubassets.com; form-action 'self' github.com gist.github.com; frame-ancestors 'none'; frame-src render.githubusercontent.com; img-src 'self' data: github.githubassets.com identicons.github.com collector.githubapp.com github-cloud.s3.amazonaws.com secured-user-images.githubusercontent.com/*; githubusercontent.com/*; manifest-src 'self'; media-src 'none'; script-src githublab7$
```