

System Programming

(Assignment3-2)

과 목	시스템프로그래밍실습
담당교수	이기훈 교수님
학 과	컴퓨터공학과
학 번	2010720149
성 명	이동현
날 짜	2016. 05. 13 (금)

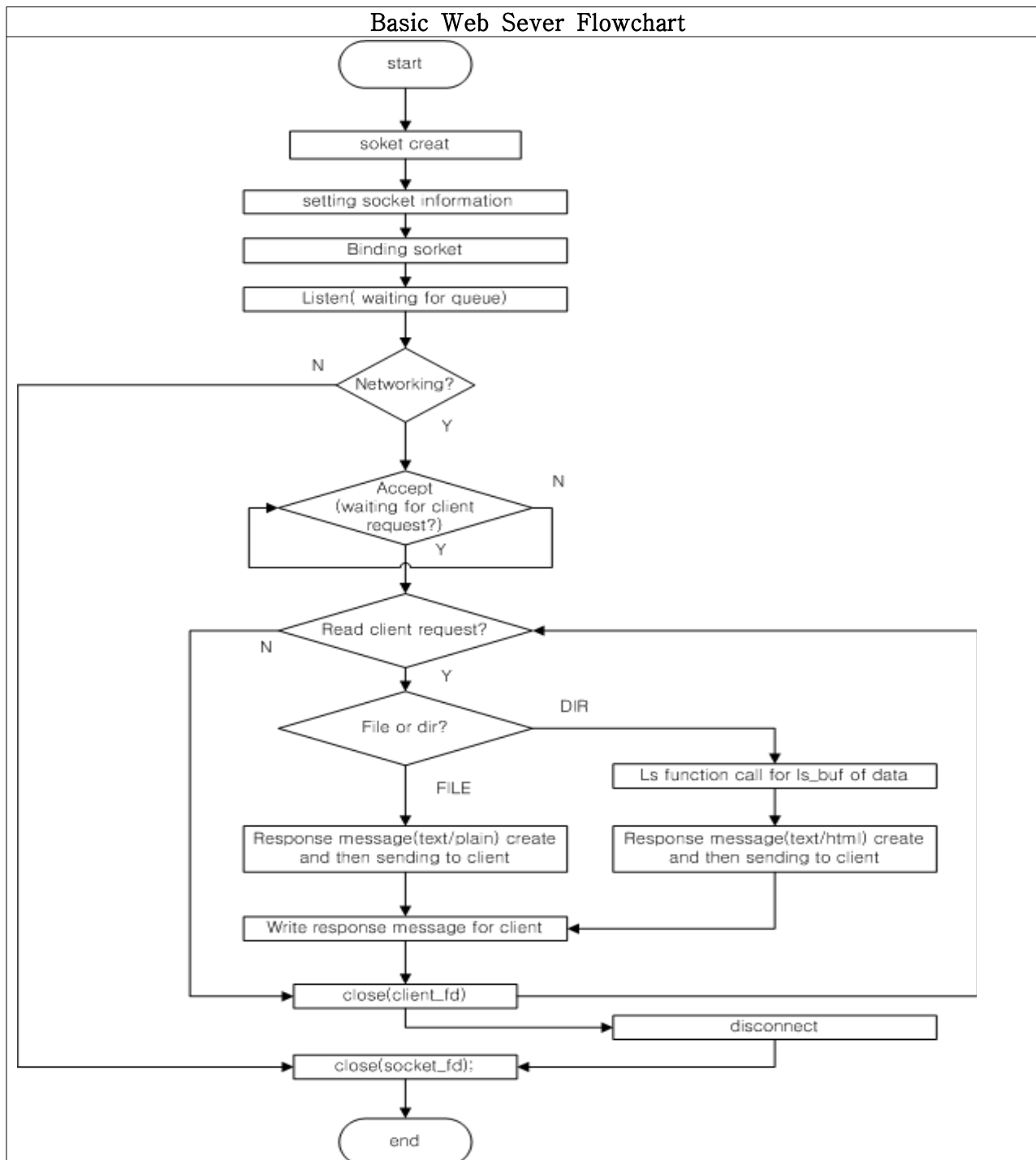


A. Introduction

♣ Basic Web Sever ♣

이번 과제는 소켓 프로그래밍을 통해 네트워크 상에서 데이터 교환을 가능하게 해주는 소켓 통신을 구현하는 것으로, 이전 과제에 구현했던 html_ls를 이어서 소켓 서버를 구축하여 사용자가 웹브라우저에 접속하여 서버로부터 받은 데이터를 웹브라우저에서 실행 결과를 확인하는 과제이다.

B. Flowchart



C. Pseudo code

Basic Web Sever Pseudo code

```

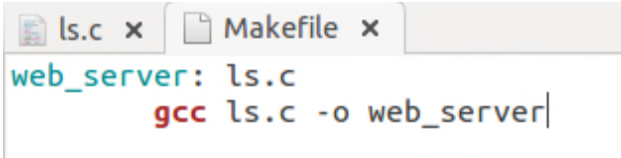
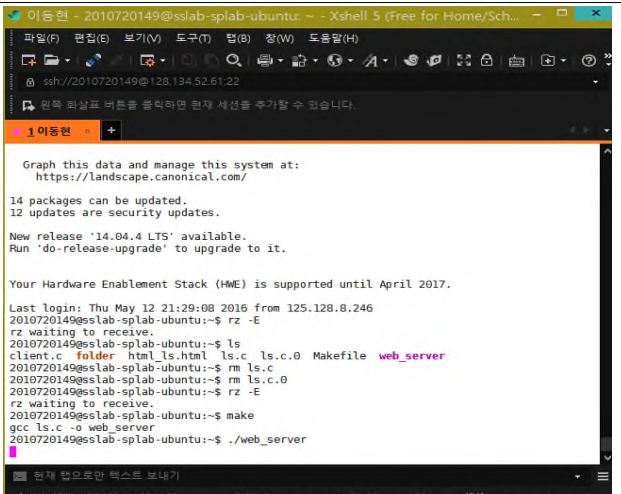
int main(){
    initializing buf, arv
    socket_fd <- socket(PF_INET, SOCK_STREAM, 0)
    opt<-1;
    bzero((char*)&server_addr, sizeof(server_addr));
    server_addr.sin_family <- AF_INET;
    server_addr.sin_addr.s_addr <- htonl(INADDR_ANY);
    server_addr.sin_port <- htons(PORTNO);
    setsockopt(socket_fd, SOL_SOCKET, SO_REUSEADDR, &opt, sizeof(opt));
    bind(socket_fd, (struct sockaddr*)&server_addr, sizeof(server_addr));
    listen(socket_fd, 10);
    while(1){
        initializing buf, response, argv
        len <- sizeof(client_addr)
        client_fd <- accept(socket_fd, (struct sockaddr*)&client_addr, &len);
        while(len_out <- read(client_fd, buf, BUFFSIZE)>0){
            argc <-2;
            Dflag, Fflag <-0;
            if(Dflag==1)
                ls(argc, argv, ls_buf);
            strcpy(response, "HTTP/1.1 200 OK\r\nAccept-Ranges: bytes\r\nConnection:
close\r\n");
            strcat(response, "Content-Length: 100000\r\n");
            strcat(response, "Content-Type: text/html\r\n");
            strcat(response, "\r\n");
            strcat(response, ls_buf);
        else if(Fflag==1)
            strcpy(response, "HTTP/1.1 200 OK\r\nAccept-Ranges: bytes\r\nConnection:
close\r\n");
            strcat(response, "Content-Length: 100000\r\n");
            strcat(response, "Content-Type: text/plain\r\n");
            strcat(response, "\r\n");
            strcat(response, ls_buf);
            write(client_fd, response, 100000);
            initializing buf, response
        }
        close(client_fd);
    }
    allocation delete for argv
    close(socket_fd);
}

```

D. Reference

- 강의자료 ' 2016-1_SPLab_09_Basic+Server_Is
- 강의자료 ' 2016-1_SPLab_09_Basic+Server_Is_v3

E. Conclusion

♣ 조건 ♣	Makefile
<p>a, l 옵션을 모두 적용한 결과를 출력 default directory: web_server가 실행된 폴더 // 실행 위치가 /home/user/sp인 경우 // http://128.134.52.61:1111로 접속 시, /home/user/sp의 ls -al의 결과 출력 // http://128.134.52.61:1111로 접속 시, /home/user/sp/temp의 ls -al의 결과 출력</p>	 <pre>web_server: ls.c gcc ls.c -o web_server</pre>
My port number	My shell
40051	 <pre>Graph this data and manage this system at: https://landscape.canonical.com/ 14 packages can be updated. 12 updates are security updates. New release '14.04.4 LTS' available. Run 'do-release-upgrade' to upgrade to it. Your Hardware Enablement Stack (HWE) is supported until April 2017. Last login: Thu May 12 21:29:08 2016 from 125.128.8.246 2010720149@sslab-splab-ubuntu:~\$ rz -E rz waiting to receive. 2010720149@sslab-splab-ubuntu:~\$ ls client.c folder html_ls.html ls.c ls.c.o Makefile web_server 2010720149@sslab-splab-ubuntu:~\$ rm ls.c 2010720149@sslab-splab-ubuntu:~\$ rm ls.c.o 2010720149@sslab-splab-ubuntu:~\$ rz -E rz waiting to receive. 2010720149@sslab-splab-splab-ubuntu:~\$ make gcc ls.c -o web_server 2010720149@sslab-splab-ubuntu:~\$./web_server</pre>

♣ 결과 화면 1 ♣

← → ✕ 128.134.52.61:40051

ls -al /home/2010720149

Directory path: /home/2010720149

total: 136

Name	Permission	Link	Owner	Group	Size	Last Modified
.	drwx-----	4	2010720149	splab_students	4096	5 13 00:36
..	drwxr-xr-x	93	2010720149	splab_students	4096	4 23 03:50
.bash_history	-rw-----	1	2010720149	splab_students	7804	5 13 00:36
.bash_logout	-rw-r--r--	1	2010720149	splab_students	220	3 29 03:02
.bashrc	-rw-r--r--	1	2010720149	splab_students	3486	3 29 03:02
.cache	drwx-----	2	2010720149	splab_students	4096	5 10 19:42
client.c	-rw-r--r--	1	2010720149	splab_students	997	5 8 19:56
folder	drwx-----	2	2010720149	splab_students	4096	5 11 04:20
html_ls.html	-rw-----	1	2010720149	splab_students	0	5 13 00:39
ls.c	-rw-r--r--	1	2010720149	splab_students	45686	5 13 00:35
Makefile	-rw-r--r--	1	2010720149	splab_students	41	5 10 21:41
.profile	-rw-r--r--	1	2010720149	splab_students	675	3 29 03:02
web_server	-rwx-----	1	2010720149	splab_students	43922	5 13 00:36

원격서버 주소의 나인 port(40051)을 주소창에 클라이언트가 요청했을 경우 서버에서 응답하는 ls -al의 화면 결과이다. default directory가 출력이 되었음을 확인 할 수 있다.

♣ 결과 화면 2 ♣

← → ✕ 128.134.52.61:40051/folder

ls -al /home/2010720149/folder

Directory path: /home/2010720149/folder

total: 8

Name	Permission	Link	Owner	Group	Size	Last Modified
.	drwx-----	2	2010720149	splab_students	4096	5 11 04:20
..	drwx-----	4	2010720149	splab_students	4096	5 13 00:36
hello.c	-rw-----	1	2010720149	splab_students	0	5 11 03:00

위 조건처럼 default directory에서 /folder를 주소창에 추가 입력하여 클라이언트가 요청을 할 경우 default directory내의 folder의 ls -al 결과를 보여주는 것이다.

| ⓘ | 128.134.52.61:40051/home |

ls -al /home

Directory path: /home

total: 372

Name	Permission	Link	Owner	Group	Size	Last Modified
.	drwxr-xr-x	93	2011722012	splab_students	4096	4 23 03:50
..	drwxr-xr-x	23	2011722012	splab_students	4096	4 22 22:06
2005720170	drwx-----	2	2011722012	splab_students	4096	4 23 03:58
2006720002	drwx-----	4	2011722012	splab_students	4096	5 13 00:54
2006720015	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2007720033	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2007720044	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2009720006	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720047	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720049	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720086	drwx-----	4	2011722012	splab_students	4096	5 12 12:10
2010720124	drwx-----	2	2011722012	splab_students	4096	4 23 03:44

Transferring data from 128.134.52.61...

이 주소창 입력은 상대패스를 통한 접근을 한 것으로 /home으로 출력이 됨을 확인 할 수 있다.

♣ 결과 화면 3 ♣

128.134.52.61:40051/home

ls -al /home

Directory path: /home

total: 372

Name	Permission	Link	Owner
.	drwxr-xr-x	93	2011722012
..	drwxr-xr-x	23	2011722012

home directory에서 ..의 하이퍼링크를 클릭을 하였을 경우!!

```

Directory path: /home/2010720149
total: 136
drwx----- 4 2010720149splab_students 4096 5 13 01:00 .
drwxr-xr-x 93 2010720149splab_students 4096 4 23 03:50 ..
-rw----- 1 2010720149splab_students 7871 5 13 00:53 .bash_history
-rw-r--r-- 1 2010720149splab_students 220 3 29 03:02 .bash_logout
-rw-r--r-- 1 2010720149splab_students 3486 3 29 03:02 .bashrc
drwx----- 2 2010720149splab_students 4096 5 10 19:42 .cache
-rw-r--r-- 1 2010720149splab_students 997 5 8 19:56 client.c
drwx----- 2 2010720149splab_students 4096 5 11 04:20 folder
-rw----- 1 2010720149splab_students 0 5 13 01:05 html_ls.html
-rw-r--r-- 1 2010720149splab_students 45705 5 13 00:59 ls.c
-rw-r--r-- 1 2010720149splab_students 41 5 10 21:41 Makefile
-rw-r--r-- 1 2010720149splab_students 675 3 29 03:02 .profile
-rwx----- 1 2010720149splab_students 43922 5 13 01:00 web_server
[-167214979:10713] client was disconnected.
[-167214979:10969] client was connected.

```

total: 136

Name
.
..
.bash_history
.bash_logout
.bashrc
.cache
client.c
folder

/home에서 '..'을 입력을 하였을 경우 디렉토리 이동 불가하다.

옆 캡처는 위에 /home을 터미널 출력한 결과인데, '..'을 클릭하였을 경우, 클라이언트 요청에 따른 서버가 response를 해주지 않는 것을 확인 할 수 있다. 상위 디렉토리의 내용물을 알 수 없으므로 서버에서 응답 자체가 불가능함을 알 수 있다.

♣ 결과 화면 4 ♣

하이퍼링크클릭으로 디렉토리 이동

```

Directory path: /home
total: 372

```

Name	Permission	Link	Owner	Group	Size	Last Modified
.	drwxr-xr-x	93	2011722012	splab_students	4096	4 23 03:50
..	drwxr-xr-x	23	2011722012	splab_students	4096	4 22 22:06
2005720170	drwx-----	4	2011722012	splab_students	4096	4 23 03:58
2006720002	drwx-----	2	2011722012	splab_students	4096	5 13 01:05
2006720015	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2007720033	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2007720044	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2009720006	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720047	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720049	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720086	drwx-----	4	2011722012	splab_students	4096	5 12 12:10
2010720124	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720138	drwx-----	2	2011722012	splab_students	4096	4 23 03:44
2010720149	drwx-----	4	2011722012	splab_students	4096	5 13 01:00
2010720128	drwx-----	2	2011722012	splab_students	4096	4 23 03:44

```

Directory path: /home/2010720149
total: 136

```

Name	Permission	Link	Owner	Group	Size	Last Modified
.	drwx-----	4	2010720149	splab_students	4096	5 13 01:00
..	drwxr-xr-x	93	2010720149	splab_students	4096	4 23 03:50
.bash_history	-rw-r--r--	1	2010720149	splab_students	7871	5 13 00:53
.bash_logout	-rw-r--r--	1	2010720149	splab_students	220	3 29 03:02
.bashrc	-rw-r--r--	1	2010720149	splab_students	3486	3 29 03:02
.cache	drwx-----	2	2010720149	splab_students	4096	5 10 19:42
client.c	-rw-r--r--	1	2010720149	splab_students	997	5 8 19:56
folder	drwx-----	2	2010720149	splab_students	4096	5 11 04:20
html_ls.html	-rw-----	1	2010720149	splab_students	0	5 13 01:15

```

Directory path: /home/2010720149/folder
total: 8

```

Name	Permission	Link	Owner	Group	Size	Last Modified
.	drwx-----	2	2010720149	splab_students	4096	5 11 04:20
..	drwx-----	4	2010720149	splab_students	4096	5 13 01:00
hello.c	-rw-----	1	2010720149	splab_students	0	5 11 03:00

주소창에 /home을 입력을 하여 서버에 요청을 보냈고, 다음은 하이퍼링크 클릭을 통한 directory 이동을 확인을 보여주는 결과 창이다. 우선 나의 학번 2010720149를 클릭하면 서버에게 요청을 하여 결과창이 바뀌고, 마찬가지로 folder 하이퍼링크를 클릭 할 경우 이동하는 것을 확인을 할 수 있다.

♣ 결과 화면 5 ♣

파일 및 실행파일을 클릭을 하였을 경우

Top Screenshot: Browser address bar shows 128.134.52.61:40051/home/2010720149/folder. Command: `ls -al /home/2010720149/folder`. Directory path: /home/2010720149/folder. total: 8.

Name	Permission	Link
.	drwx-----	
..	drwx-----	
hello.c	-rw-----	

Bottom Screenshot: Browser address bar shows 128.134.52.61:40051. Command: `ls -al /home/2010720149`. Directory path: /home/2010720149. total: 136.

Name	Permission	Link	Owner	Group	Size	Last Modified
.	drwx-----					
..	drwxr-xr-x					
.bash_history	-rw-----					
.bash_logout	-rw-r--r--					
.bashrc	-rw-r--r--					
.cache	drwx-----					
.client.c	-rw-r--r--					
.folder	drwx-----					
.html ls.html	-rw-----					

클라이언트가 하이퍼링크를 클릭을 할 때, 그 파일이 디렉토리가 아닌 일반 파일이거나 실행파일일 경우 파일을 open 혹은 save 시킬지 인터페이스 창을 제공한다.

♣ 고찰 ♣

이번 과제는 이전 과제 `html_ls`를 구현한 것을 소켓 프로그래밍을 이용하여 소켓 통신 객체로 네트워크 상에서 데이터를 교환을 가능하게 하는 프로그램을 구현을 하는 것이다.

서버와 클라이언트간의 통신 알고리즘을 이해를 하여, 프로그램에 구현이 큰 도움이 되었고, 소켓 생성 및 연결 등 시스템 콜 함수와 기본 설정에 대해서는 강의자료에 자세히 설명이 되었기 때문에 생소하지만, 구현을 할 수 있었다. shell 원격 서버를 이용하여, 나의 포트 40051에 접속하여 내가 서버를 구현을 하여 나의 웹상에서 요청에 따른 `ls` 결과물이 출력되는 것이 다소 신기하기도 하였다. 살아오면서 자주 인터넷을 사용하였지만, 이전에 데이터통신시간에 배운 서버와 클라이언트 사이에 연결을 직접 구현을 하게 되니, 좀 더 폭 넓은 서버 통신을 이해하기 좋았다. 아무래도 서버와 클라이언트 통신이기 때문에 request message와 response message가 매우 중요했다. 그 중에서 서버를 구현을 했기 때문에 후자의 message의 구현을 매우 주의를 해야했는데, 반드시 HTTP version, Server response code와 Content-Type field는 포함을 시켜주지 않으면 올바른 통신을 할 수 가 없었다. response message를 write를 할 때 이어서 개행과 함께 데이터를 함께 전송하는 방식이기 때문이다. message가 별거 아니라고 생각하면 제대로된 통신을 확인을 할 수 없기 때문에 http response format을 확인 할 필요가 있었다. 즉, 서버와 클라이언트 통신에 있어서 status line과 header line은 꼭 필요하다고 생각을 했다. 그리고 프로그램 구현 과정 중에 파일과 디렉토리 하이퍼링크 클릭에 따른 서로 다른 결과가 생성이 되어야하는데, 파일은 그 파일을 열어 저장할 것인지 오픈할 것인지 인터페이스 요구이거나 디렉토리는 path를 이동하여 해당 디렉토리의 `ls`결과를 확인 할 수 있도록하여야 한다. 여기서 중요한 것은 위에 언급 했던 content-type field(data format)에 따라 다르게 취해주어야 한다. 즉, text/html일 경우는 해당 html 코드를 뜻을 하며, text/plain일 경우 그 이외의 일반 파일임을 가리킨다. 따라서 이 경우 두 가지의 경우의 수에 맞게 처리를 해주어야 프로그램 구현을 할 수 있었다. 그리고 리눅스 가상머신 내의 파이어폭스 말고 익스플로어나 구글 크롬에 서버를 요청시 클라이언트 요청에 따른 서버 응답이 느려 확인 하기가 어려웠다. 나의 컴퓨터가 문제인지, 어떤지는 정확히는 알지 못하지만 리눅스 내에서는 잘 되었는데, 이 부분을 한번 확인 해봐야 할 것 같다.

이번 과제를 수행함으로써 여태 배웠던 데이터 통신에 대해서 폭 넓은 개념 이해를 할 수 있었고, 다음 과제 여러 client의 요청에 따른 서버의 응답을 구현을 하게 될 것인데, 큰 흥미가 생겼다.