

Term Project: Linux System(Socket programming)

May. 06, 2020.

Version. 1.1

Contributor : Jong-Beom Jeong, Eun-Seok Ryu

Socket은 네트워크에 연결된 두 node를 연결하기 위해 사용될 수 있다. 그림 1은 socket을 이용한 데이터 통신의 실행 흐름을 나타낸다.

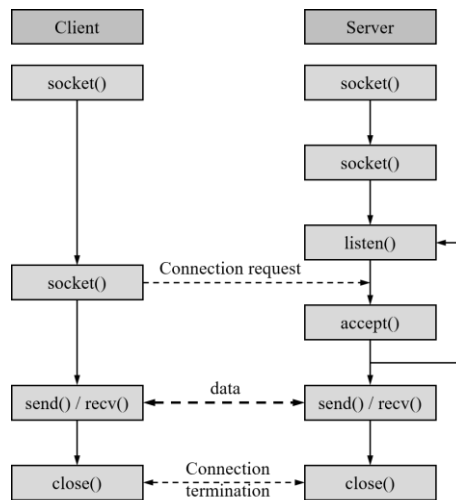


그림 1. Socket communication workflow

Mission1. TCP를 이용한 서버와 클라이언트를 구현하시오. (연습)

C/C++를 사용하되, 리눅스에서 빌드가 가능하고 동작 가능해야 하며, 다음과 같이 동작한다. (1) Client가 알려진 서버 주소로 접속하여 Hello Server 메시지 전달, (2) 서버는 받아서 이를 화면에 printf하고 클라이언트에게 역시 Hello Client 메시지 전달, (3) Client는 이를 받아서 화면에 출력.

인터넷의 Open source를 이용해도 좋으나 본인이 최대한 습득해야 함.

Mission 2. 위의 결과물을 변형하여 지정된 영화 파일을 서버에서 클라이언트로 전송하도록 구현하시오. 다음과 같이 동작한다. (1) 클라이언트는 서버에게 접속하여 Hello Server 메시지를 전송, (2) 서버는 미리 지정된 영화 파일 (예로 1GB의 test.mp4 파일)을 접속한 Client로 발송하되, 1460 바이트의 패킷으로 나누어 계속 전달, (3) 클라이언트는 서버로부터 받아들인 패킷들을 메모리 버퍼 또는 파일버퍼에 모았다가 최종 file write를 하여 저장함. (4) 그 후, debug를 위해 원본 파일과 받은 파일의 내용 및 사이즈가 같은지 비교 (예: cmp나 diff 명령어 사용).

본 내용을 리눅스 상에서 구현하여 (코드 자체는 윈도우에서 편집하여도 무방), 리눅스 환경에서 compile 및 link를 하고, 이를 하나의 머신 (본인의 laptop에서 서버 클라이언트 모두 띄우고 IP를 127.0.0.1로 세팅해도 무방)에서 테스트 하거나, 하나 이상의 머신 (예로 친구 리눅스; IP는 미리 알아야 함. 예로 친구의 DOS CMD시스템에서 ipconfig 명령어로 알 수 있음)에서 테스트한 내용을 보고서로 제출함.

보고서 (MS Word나 PDF로만 받음) 제출내용:

1. Mission2 에 대한 소스코드 (1 페이지를 ½로 나누어 두페이지의 내용을 1 페이지에 담음) – 소스코드는 별도의 파일로 zip 압축하여 보고서 파일과 함께 시스템에 올림
2. 실행화면 캡처
3. 구현/개발 내용: 본인이 학습한 내용 위주로 본인 작품의 기능 등을 설명. 특히 추가적 기능이 있을 경우 이를 강조하여 설명.

기능적 추가의 적절한 예:

1. 만일, 1GB 의 파일을 서버에서 클라이언트로 옮길 경우, 파일을 여러 개로 분할하여 sub-file 들을 만든 후 소켓을 하나 이상 여러 개 열고, 멀티쓰레딩(Multi-threading) 프로그래밍을 통해 동시에 전달함으로써 속도를 올려보는 예 등

기본 1인 1과제이나, 본인이 현저히 코딩 실력이 부족할 경우 (1) 교수자에게 알리고, (2) 보고서 맨 앞에 도움 받았음을 눈에 띄게 (noticeable하게) 표시하는 조건으로 친구의 도움을 받는 것을 허용 (단, 도움을 준 학생이 같은 수업을 듣는 학생이라면, 이름 반드시 보고서에 표기.)

그렇지 않고 Googling한 소스코드를 변형 없이 사용하거나, 친구의 코드를 파라미터 순서 바꾸고, 함수 이름 바꾸고, 코드 상의 위치를 좀 바꾸는 단순 변형 후 거의 그대로 사용한 경우 부정행위 (cheat)로 판단하여 honor code를 위반한 것으로 처리함.

기타 질문은 TA에게 메일을 보내거나 MCSL연구실(중앙학술관 5층)을 방문하여 질문하는 것을 권장함.

마감일: 2020년 6월 6일 (hard deadline)