Introduction to Linux Systems

Inter-Process Communication (IPC)

Chia-Heng Tu
Dept. of Computer Science and Information Engineering
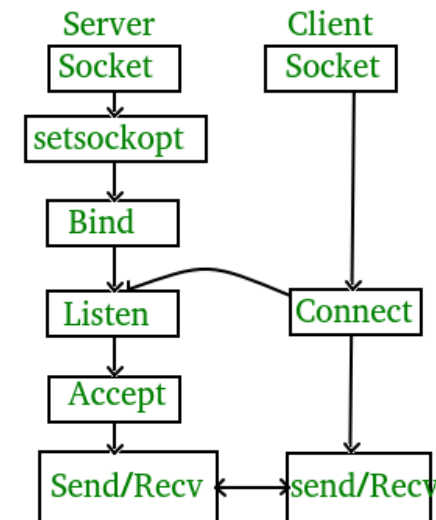National Cheng Kung University
Fall 2022

Courtesy of

Mini project: Socket Programming in C

- https://www.geeksforgeeks.org/socket-programming-cc/

```
Client:
Hello message sent
Hello from server

Server:
Hello from client
Hello message sent
```
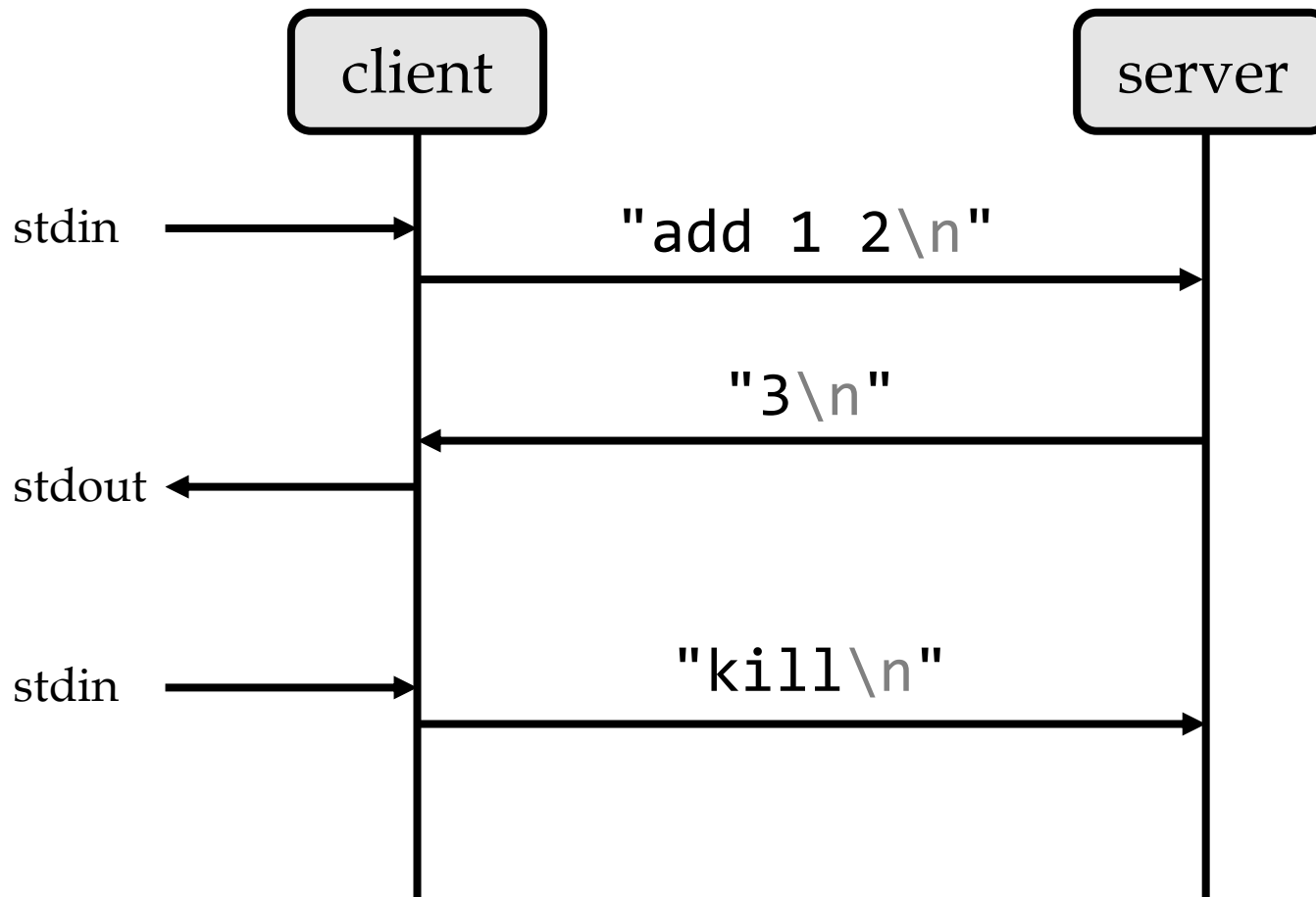
IPC using simple socket

- server

  - supports three commands   -   supports command kill
    1. "add a b":   return $a + b$
    2. "abs a":   retuen $|a|$
    3. "mul a b":   return $a \times b$

  - When getting unsupported command: return Hello

- client

  - reads the commands until EOF
  - sends the commands to server

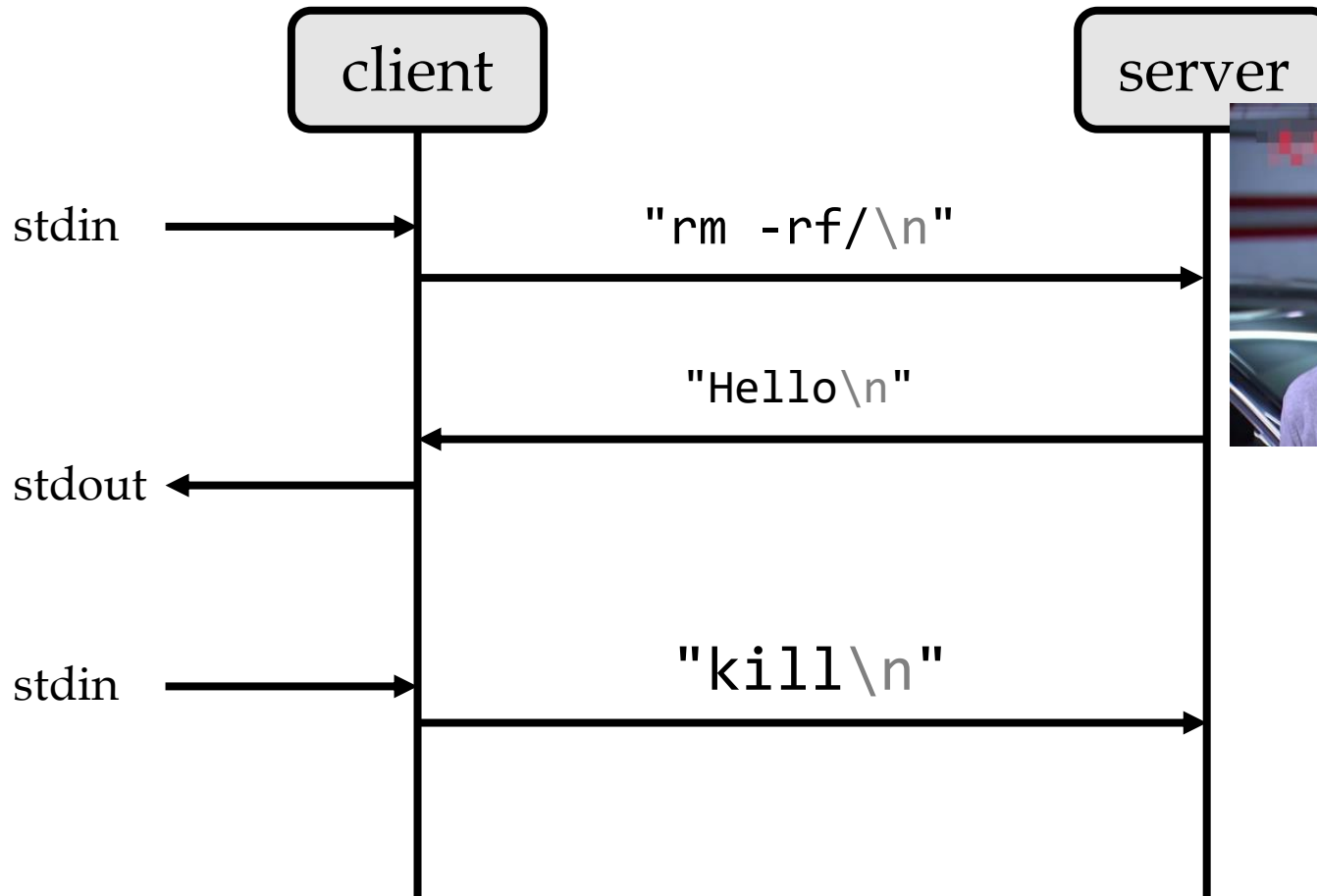# Execution flow (supported)

stdin → client ──── "add 1 2\n" ──────→ server

server ──── "3\n" ──────→ client → stdout

stdin → client ──── "kill\n" ──────→ server

Execution flow (unsupported)


```
stdin ──→ client                    "rm -rf/\n"                    server
              ├──────────────────────────────────────────────────→│
              │                      "Hello\n"                      │
stdout ←──────┤←──────────────────────────────────────────────────┤
              │                      "kill\n"                       │
stdin ──→─────┤──────────────────────────────────────────────────→│
```

You should pass **4** tasks (60pt)

```
1 add 1 26
2 abs -100
3 mul 33 55
4 unsupport!!
5 kill
6
```

```
$ cat out1
27
100
1815
Hello
```

Input 檔案最後一行是空的

拿回來的答案包含換行

Test by yourself

This also how we test your program

$ make

$ ./server & ./client < in > out

$ diff -s out ans

if your output is correct, you will get

Files out and ans are identical

Hint: you can add some delay in client to avoid random connection error

Operation process and comments (40pt)

- Clear, Simple, and Useful

- Comments in source code

- Bonus: consider more complex case (e.g. avoid overflow) and mention in demo video

This will be scored by demo video (video should not exceed 3 min)

Carter Wickstrom @carter... · 1d ⌄
90% of all code comments:

Submission

- Upload **F12345678.zip** to moodle

- **F12345678** is your Student ID

- Due date : 1/4 12:00

```
F12345678.zip
└── F12345678/
        ├── in
        ├── ans
        ├── client.c
        ├── server.c
        ├── Makefile
        └── Demo.mp4
```

繳交不符合規定扣 10 分