Chalmers university
Computer science department

Lab 2 - Simple, flawed Server

Homework from computer networks

Michal Šustr, Arnaud D'Artigues
michal.sustr@gmail.com
arnaud.dartigues@gmail.com

8. 9. 2014

1 Answers to lab questions

Question I.a.1

The symbolic name of the address in the code is `INADDR_ANY` and it is `0.0.0.0`.

Question I.a.2

From the manual of `recv`:

> Return the number of bytes received, or -1 if an error occurred. The return value will be 0 when the peer has performed an orderly shutdown.

This answers the cases a), b) and c). The last case d) `ret=kTransferBufferSize` means that the buffer is completely filled.

`cd.buffer` is defined to be of size `kTransferBufferSize+1` because in C we need to include the `NULL` terminating character.

Question I.a.3

The connection is closed/reset if the return value is -1 and error code of value `ECONNRESET`.

`MSG_NOSIGNAL` flag stops generating `SIGPIPE` signals on stream oriented sockets when the other end breaks the connection, the `EPIPE` signal is still generated indicating if the local end has been shut down on a connection oriented socket.

Question I.c.1

When we connect with the first client, it can send and receive messages. Once a second connection with another client is established, it can send a message but gets blocked until the first client disconnects. When we do a netstat, we can see that for both clients, the connection with the server is `ESTABLISHED`, which means their sockets are truly connected.

Question I.c.2

Once the first client gets disconnected, it receives response from the server. The server is iterative, so it can handle requests only from one client at a time until it disconnects. Then the next client can be served.

Question I.c.3

The round trip time on the same machine depends on the length of the message, it varies from $0.1ms$ to $0.0001ms$. If the message length is the same as `kTransferBufferSize`, the 5 measured times are: $0.000212ms$, $0.000189ms$, $0.000205ms$, $0.000172ms$, $0.000222ms$ with average $\overline{T_{same}} = 0.0002ms$.

We connected to different machine, and the measured round trip times are $185.15ms$, $236.51ms$, $231.71ms$, $544.55ms$ and $269.42ms$, with average $\overline{T_{different}} = 293.46ms$.

Question I.d.1

We ran the command `./client-multi localhost 31336 7 255` with following measurements:

```
Establishing 7 connections...
  successfully initiated 7 connection attempts!
Connect timing results for 7 successful connections
  - min time: 0.366060 ms
  - max time: 0.549235 ms
  - average time: 0.461344 ms
 (0 connections failed!)
Roundtrip timing results for 7 connections for 255 round trips
  - min time: 10159.585865 ms
  - max time: 71119.216374 ms
  - average time: 40639.376403 ms
```

Some other records:

```
Establishing 10 connections...
  successfully initiated 10 connection attempts!
Connect timing results for 10 successful connections
  - min time: 0.103116 ms
  - max time: 0.218299 ms
  - average time: 0.138663 ms
 (0 connections failed!)
Roundtrip timing results for 10 connections for 255 round trips
  - min time: 10160.063502 ms
  - max time: 101603.979012 ms
  - average time: 55882.432563 ms


Establishing 15 connections...
  successfully initiated 15 connection attempts!
Connect timing results for 15 successful connections
```

```
  - min time: 0.382923 ms
  - max time: 3001.949394 ms
  - average time: 333.608482 ms
 (0 connections failed!)
Roundtrip timing results for 15 connections for 255 round trips
  - min time: 10158.026511 ms
  - max time: 151399.968558 ms
  - average time: 80944.698838 ms
```

Question I.d.3

We measured the time from beginning of the connection until error in *recv* was obtained as a difference of timestamps from `get_time_stamp()`. On average, the time is $\overline{T_{timeout}} = 34371.25ms$.