

RPC Worksheet

Operating Systems for Distributed Systems

Michael Reißner

Task 1

Implement two programs (using *Python* or *C*) which use sockets to do the following tasks:

- A server should expect text in an endless loop via a network interface as input.
- The server adds its hostname followed by a colon and a space at the beginning of this text and sends it back to the client.
- The client gets via shell input an IP-adress or DNS-name, a port-number and a message to be sent to the server.
- The client waits for a response from the server after sending its message.
- Client and server print the messages they receive locally.
- You can expect that no more than 1024 Bytes have to be transmitted.
- The server gets via shell input a port-number on which it expects messages.

The following sys-calls might be of interest:

```
socket(), bind(), listen(), accept(), send(), recv(), recvfrom(),  
sendto(), gethostname(), gethostbyname()
```

- a) Solve the task using UDP.
- b) Solve the task using TCP.
- c) Is there a difference from a functional point of view between both solutions?

Task 2

Solve task 1 using *Remote Procedure Calls (RPC)*. Use `XMLRPC` if you used with *Python* or `rpcgen` if you used *C*.