Basic client-server model with network sockets

Jaakko Koivuniemi

CERN/PH

April 11, 2014

Abstract

client-server, examples, sockets, security

Raspberry Pi at CERN

Client-server model

- networked distributed application structure

- server: provider of a resource or service

- client: service requester

- typical applications: database server (mysqld), web server (httpd), file server (ftpd/scp)

Example: music player daemon and client

```
1 pi@raspberrypi ~ $ service mpd status
  [ ok ] mpd is running.
3 pi@raspberrypi ~ $ mpc play 1
  Dire Straits — Telegraph Road
5 [playing] #1/25    0:00/14:21 (0%)
  volume: n/a    repeat: off    random: off    single: off    consume: off
7 pi@raspberrypi ~ $ mpc stop
```

Second example: low earth orbit satellites

```
1 pi@raspberrypi ~ $ predict −s
  pi@raspberrypi ~ $ psatcal −e 20 −s VO−52
3 F5VGL lat 46.2292 deg long −5.9584 deg at 450 m alt
  satellite passes TCA elevations >= 20 deg for 1 days
5 predict utc time now Sun Apr 06 20:27:25 2014
  Unix time from predict: 1396816045

7

  ─────────────── 07Apr14 Mon ───────────────
9 aos[utc] los[utc] sat      el range compass  sun up[kHz]down up [dB] down
  05:33:22 05:46:10 VO−52    43   872 NN−EE−SS *   −9.7  +3.2  144/154
     134/145
11 07:09:42 07:21:17 VO−52   20  1419 NN−NW−SW *   −9.4  +3.1  148/154
     139/145
  16:29:30 16:41:57 VO−52    42   865 SE−NE−NN *   −9.9  +3.3  144/154
     134/145
13 pi@raspberrypi ~ $ dpl847 −s VO−52 −d /dev/ttyUSB0 −o
```

Sockets

- network socket: end-point in a communication across a network or internet

- unix domain socket: end-point in local inter-process communication

- use **netstat** to list open internet and unix sockets

- **/etc/services** port numbers for different services

- Berkeley sockets application programming interface (API)

- provides functions: socket(), bind(), listen(), connect(), accept() etc

References

- http://en.wikipedia.org/wiki/Client-server_model

- http://en.wikipedia.org/wiki/Berkeley_sockets

- http://en.wikipedia.org/wiki/Iptables

- http://en.wikipedia.org/wiki/Secure_Sockets_Layer

- https://github.com/oh7bf/PiPIC/tree/master/pipicswitch