Protocol Software Engineering

Tiina Niklander

Spring 2008

Linux networking

- Key themes?
- Unclear parts?
- What was new to you in the material you read?

Linux networking

- Would you feel confident in modifying/replacing the protocols provided by the kernel? Why / Why not?
- Do you have clear view of the general structure of the Linux networking?

Sockets

- Did you manage to get a socket-based program running?
- What modificationd did you try?
- Think different ways of 'mimicing' the channel errors for testing the reliable bahaviour of your protocol
  - There has to be errors
  - You may want to be able to control the errors (at least to some extent)

Controlling the errors

- Task: *Think different ways of 'mimicing' the channel errors for testing the reliable bahaviour of your protocol*
  - *There has to be errors*
  - *You may want to be able to control the errors (at least to some extent)*
- Solutions:
  - .
  - .
  - .

Project tasks:

Project / laboratory

- Topic: Reliable datagram delivery service on top of UDP
  - UDP is unreliable
  - Need to have acknowledgement and/or flow control (simple stop-and-wait not accepted)
- Three subtasks:
  - Protocol design
  - Protocol specification and verification using SDL
  - Implementation (in C) of a small extension or modification to an existing protocol implementation

Task 1: Design dl 17.3.08

- write a RFC style description of your protocol
  - message types,
  - header structure and
  - the functionality
- Lenght of the document depends on the number of features in your protocol
  - A good estimate: 5 to 10 pages

Task 2: SDL dl 7.4.08

- Protocol specification and verification using SDL
  - Model your protocol (or part of it)
  - Model the unreliable channel (minimum requirement: loosing messages)
  - Verify by simulation that it works
- Submit:
  - your design,
  - some MSC diagrams that show the behaviour of your protocol
  - coverage of your simulations
- Email with a weblink to the actual material is the prefered submission format.
  - (alternatively (*when you do not master web pages*) a pdf document as attachment is accepted)

Task 3: Implementation dl 12.5.08

- Implement (in C)
  - Your protocol (or part of it)
  - Preferably as a small extension or modification to an existing protocol implementation
- Please notice that you must implement your protocol in such a way that it can easily be used by different clients and servers.
- Submit:
  - Implementation document
    - Specify the original protocol code location
  - Testing document
  - Source code
    - Protocol implementation, test clients and test servers

Exam

Friday 29.2. 16.00 in room A111

Exam covers

- Reading material, slides, stored lectures
- Five main themes:
  - Internet architecture,
  - SDL,
  - Testing,
  - Linux kernel,
  - Linux networking
- However: not everything can be asked in the exam due to time limit