Network Programming

Client/Server Architecture

Note: **This class lecture will be recorded!**
If you do not consent to this recording, please do not ask questions via your video, audio or public chat; send your question to the instructor using the private chat.

Lisa Frye, Instructor
frye@Kutztown.edu
Kutztown University

CSC328: Network Programming

- ***Network*** Programming
- Expectations
  - Mine
  - Yours
- Semester Plan

Terminology

- Client/Server Application
- Distributed Application

- Protocol

- Concurrent Processing

Standards

- Explain the difference
  - Standard
  - Recommendation
  - De facto standard

Client / Server

- Client/Server Paradigm / architecture / model

- Client
- Server

Rendezvous Problem

- Explain
- How can Client/Server model solve it?
- Daemon program
- Port numbers

Server State

- State Information
  - Stateless
  - Stateful

- Two approaches for Stateful Server
  - Endpoints
  - Handle

Retain State Information

- Maintain indefinitely
  - Resulting problem?
  - Solution?

- Difficulty of maintaining state information?
  - Solution?

More Terms

- Idempotent
- Atomic operation

Discussion

- Exercise 2.7 – Write down the data structures and message formats needed for a stateful file server. Use the operations *open*, *read*, *write*, and *close* to access files. Arrange for *open* to return an integer used to access the file in *read* and *write* operations. How do you distinguish duplicate *open* requests from a client that sends an *open*, crashes, reboots and sends an *open* again? See sections 2.3.7 and 2.3.8 for information on stateless and stateful file server examples.

Discussion 2

- Exercise 2.8 – In the previous example, what happens in your design if two or more clients access the same file? What happens if a client crashes before closing a file?

Signals

- Interrupt
- Signal
  - Process, block, or ignore
- Signal handler
  - Passing data