

CSCI 491-01

Topics: Internet Programming

Fall 2008

Preliminaries

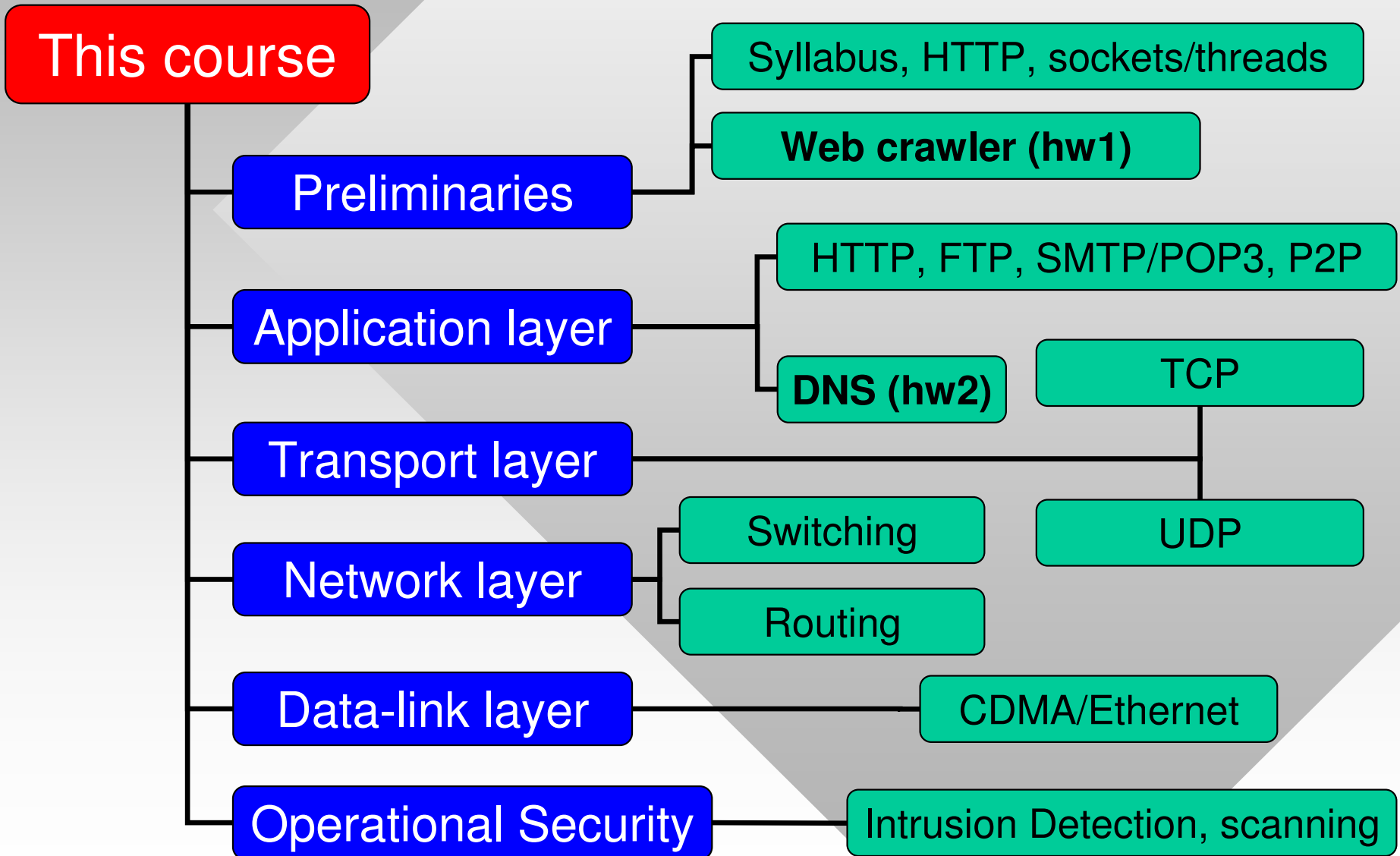
Derek Leonard
Hendrix College

August 27, 2008

Preliminaries: Agenda

- Roadmap
- Syllabus
- Academic integrity
- Homework expectations
- Web Crawler
- Questions

Preliminaries: Roadmap



Preliminaries: Agenda

- Roadmap
- **Syllabus**
- Academic integrity
- Homework expectations
- Web crawler
- Questions

Preliminaries: Syllabus

- Instructor: Derek Leonard
 - Office hours: MWF 2:00-3:00 pm in MCREY 320
 - Office Phone: 505-2933
 - Cell Phone: 979-571-5310
- Text:
 - J.F. Kurose and K.W. Ross, “Computer Networking: A Top-Down Approach,” Addison-Wesley, 4rd edition, 2007
- Site: <http://ozark.hendrix.edu/~leonard/491-01/>

Preliminaries: Syllabus 2

- Homework (40% of final grade):
 - 3 programming assignments
 - Each explores a different aspect of computer networks as they pertain to the Internet
- Exams (60% of final grade):
 - **Closed-book, no cheat-sheets**
 - 3 quizzes (15% of final grade):
 - Problems from the back of each chapter
 - 3 midterms (45% of final grade):
 - Cover topics from class and homework
 - **No final**

Preliminaries: Syllabus 3

- Final grade distribution
 - 90-100% (A)
 - 80-89% (B)
 - 70-79% (C)
 - 60-69% (D)
 - 0-59% (F)
- Do not hesitate to ask for help with the homework
 - Homework is time-consuming
 - Multithreaded programming may be hard to debug

Preliminaries: Syllabus 4

- Ask questions!
 - Office hours right after this class (MWF 2:00-3:00 pm)
 - During class
 - Through email
 - leonardd@hendrix.edu
 - Make an appointment
 - Stop by if my door is open
- You can even send me or your code with a specific question
 - Do not expect to write the whole thing for you though

Preliminaries: Agenda

- Roadmap
- Syllabus
- **Academic integrity**
- Homework expectations
- Web crawler
- Questions

Preliminaries: Academic Integrity

- No teamwork is allowed
 - General discussion is acceptable, but **no part** of an assignment may be copied
- Academic Integrity (pg. 39 in the '08-'09 catalog)
 - All sources must be properly acknowledged (including code!)
 - No information may be copied from the Internet or books (exception: `man` page sample code is OK to use)
 - Do not submit someone else's work
- **All** parties involved in cheating will be punished equally
- Cheating:
 - Any occurrence: F in class and potential suspension/expulsion from the college

Preliminaries: Agenda

- Roadmap
- Syllabus
- Academic integrity
- **Homework expectations**
- Web crawler
- Questions

Preliminaries: HW Expectations

- Homework:
 - Due at the **beginning** of class, no exceptions
 - Delays for personal reasons must be requested **well in advance**
 - If late, 20% penalty per day (no points after 5 days)
- Conform to the statement of the problem

Preliminaries: HW Expectations 2

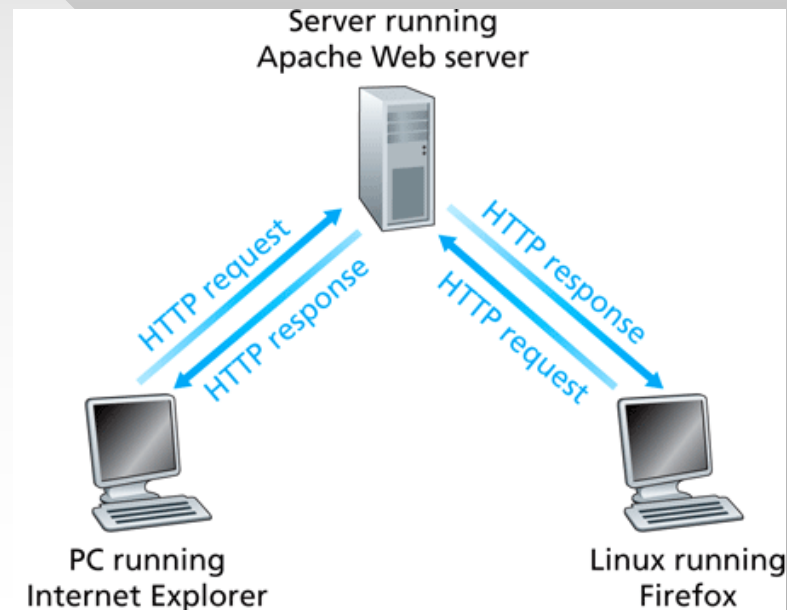
- Provide a detailed written report
 - Explain what your code does and how it accomplishes the required functionality
 - Describe tricky or interesting parts of your implementation
 - Provide analysis of your results
 - Answer questions posed in the problem statement
- Sample runs
 - Capture screenshots or print into a file details of what your code does on test input data; put this into your report
- Goal: demonstrate in your report that you really **understood** the material

Preliminaries: Agenda

- Roadmap
- Syllabus
- Academic integrity
- Homework expectations
- **Web Crawler**
- Questions

Preliminaries: Web Crawler

- Implement a crawler to examine the link structure of the World Wide Web.
 - Example of an **application-layer** (ch 2 of the book) program
- HTTP (Hypertext Transfer Protocol)
 - Application layer protocol for the web
 - Allows web browsers (clients) to communicate with web servers



Preliminaries: Web Crawler 2

- Web pages have links to other pages and content.
- Crawling using BFS:
 - Extract a URL from the queue Q
 - Download the page using an HTTP GET request
 - Parse the result, extract URLs
 - For each URL x, check if it has been inserted into Q
 - If not, mark x as visited, add to the queue Q of pending pages
 - Add the URL and other information to set S for processing
- The crawl starts from a seed website
 - Given at the command line
- The crawl ends when the BFS queue is empty or termination parameters are reached

Preliminaries: Web Crawler 3

- Goals of the homework
 - Record the set S of all found URLs and their relevant statistics into a file
 - Build a distribution of URL in-degree to discover the most popular web pages
 - Plot the growth of newly discovered links vs. depth of the crawl
 - Build a distribution of web servers
- This homework is due in 3 parts
 - Part 1: connect and obtain URLs from a webpage (25%)
 - Part 2: single-threaded crawler (25%)
 - Part 3: full multi-threaded version (50%)

Preliminaries: Web crawler 4

- HTTP (HyperText Transfer Protocol)
 - Server takes requests, client retrieves webpages
 - Server usually listens on port 80, but this may be coded inside the URL following the colon
- Steps of an HTTP client
 - Open a TCP connection to the server
 - Send a GET request (e.g., `GET index.html HTTP/1.0`)
 - Wait for server to return the requested webpage
- Once you have web page, add links to the BFS queue
- See <http://www.w3.org/TR/html401/struct/links.html> for information on parsing links in HTML

Preliminaries: Agenda

- Roadmap
- Syllabus
- Academic integrity
- Homework expectations
- Web Crawler
- Questions

Next Time

- More about sockets
- More on multi-threaded applications
- Suggestions before next class:
 - Read and play with sockets tutorial on course webpage
 - Attempt connecting to a web server and issuing a GET request as shown in section 3.6 of the handout
- Read Section 2.2.3 of the book
 - If you have a question, bring it up!

Useful tools

`telnet, nslookup, ping`