Course Objectives:

This graduate-level course is a part of a book project “Modern Computer Networks, An Open Source Approach”. The book project is motivated by an observation that none of the existing textbooks on computer networks really demonstrate how network protocols and algorithms are implemented in real systems. This course covers why and how various protocols and algorithms are designed (domain knowledge) and implemented (hands-on skills) into Linux kernel 2.4, drivers, and various daemons. In short, physical and data link layers are embedded into network adaptors and their drivers, while IP and TCP/UDP layers are built into kernel; and various application servers stand as daemons.

The loading for students includes (1) 5 homework sets for the first 5 chapters (hand-writing and hands-on), (2) midterm and final exams, and (3) a term project. The per-person term project is usually an extension of a hands-on homework problem (e.g. measurement of table lookup time, driver’s interrupt service time, or server latency; modification of TCP flow control or Web caching; configuration of routers, firewall, or servers; etc.).

Course materials for each chapter (slides and book materials) are downloadable from the course homepage. Feedbacks (error reporting or suggestions) to the book materials are encouraged and will receive extra credits for your grade.


References:
2. 林盈達, 計算機網路實驗, 維科出版社, 1999 年 9 月.

Grading: Homework (x5) 25%, Midterm 30%, Final 25%, Term Project Report 20%

Course Outline: Modern Computer Networks, An Open Source Approach

1. Fundamentals (9/17-9/27)
   Requirements of Networking
   Principles of Networking
   Internet Architecture
   Open Source Implementations
   Who’s Who
   Book Roadmap

2. Data Link Layer (10/1-10/11)
   General Issues
   Point-To-Point Protocol
   Ethernet
   Wireless Protocols
   Device Drivers

3. Internet Protocol Layer (10/15-11/1)
   General Issues
User-Plane Protocols and Algorithms (kernel 2.4)
Control-Plane Protocols and Algorithms (dhcpcd, zebra)

→ **Midterm Exam (11/8)**

### 4. End-To-End Protocols (11/5-11/19)
- **General Issues**
- **UDP (kernel 2.4)**
- **TCP (kernel 2.4)**
- **Socket Interfaces (kernel 2.4)**
- **Real-Time Transport**

### 5. Internet Services (11/26-12/6)
- **General Issues**
- **Domain Name System (bind)**
- **E-Mail (qmail)**
- **World Wide Web (apache)**
- **Web Caching (squid)**
- **File Transfer Protocol (wu-ftp)**
- **Simple Network Management Protocol (net-snmp)**
- **Multimedia (openphone)**

→ **Term Project Proposal (2 pages) Dues (12/6)**

### 6. Internet Quality of Service (12/10-12/17)
- **General Issues**
- **Integrated Services (tc)**
- **Differentiated Services (tc)**

### 7. Internet Security (12/20-12/31)
- **General Issues**
- **Access Control with Firewall and Application Proxy (netfilter, tis)**
- **Data Security with Encryption and Authentication (freeS/wan)**
- **Intrusion Detection (snort)**

→ **Final Exam (1/10)**
→ **Term Project Report (10-15 pages) Dues (1/17)**

**Appendix A. Linux Operating System**
- **Getting Started**
- **Installation Guide**
- **Configuration Guide**
- **The Boot Process**
- **Linux Internals (kernel 2.4)**
- **Networking Kernel (kernel 2.4)**

**Appendix B. Development Tools**
- **Programming Tools (gcc, make, gdb, kgdb, ddd, cvs, rpm)**
- **Benchmarking Tools (ttcp, webench)**

**Appendix C. Experimental Tools**
- **Addressing Tools (host, arp)**
- **Probing Tools (ping, traceroute)**
- **Monitoring Tools (tcpdump, netstat)**
- **Simulation and Emulation Tools (ns2, nist-emulator)**
- **Hacking Tools (nessus, ettercap, tfn2k)**