CMPE487: Applied Computer Networks
Course Overview

Objectives and Description

This is a senior level course on real world Computer Networks, providing an applicational overview with an emphasis on Local Area Networks (LANs). The theme of the course will be a series of challenges to as a group design the protocol stack necessary for a simple modern application (peer-to-peer chatting). The intention is to facilitate a robust enough understanding of modern LANs to enable students to go on to create their own applications. The target audience for this course are students with professional interest in developing network oriented software, those who are interested in practical applications of data networks, or of course the generally curious.

Workshops and Quizzes: This course is about building software – each lecture will feature a "what are we building today" section. In-class workshops will form a significant portion of the course – attendance is therefore mandatory. Short quizzes will be given at the beginning of each lecture covering material from the previous lecture.

Midterm: Students will give a short presentation of an academic paper of interest to the course as a midterm assessment.

Final: The final will be a group coding project implementing a network application. Each student will pitch a project idea in class, and the class will choose by popular vote which projects will be implemented. Proponents of the chosen projects will form 2-person teams to run a monthlong implementation project. The final deliverable will be a presentation and code for a working demo.

Friendly warning: This course requires some coding experience. If you haven't built a working application in Python before, the assignments will be very challenging.

The best work may often go unnoticed if not presented well – we will emphasize presenting in this course. Students will be expected to give three presentations: the paper presentation, the 2-minute pitch for the final project, and the final project report itself. Optional weekly reading will be assigned in the form of papers and websites. A laptop booting MacOS or Linux (if desired from a USB drive) is required for this course. Advanced students can complete the course with a laptop running Windows, but should note that no instructor support will be available for these systems.

Course Content

The course will cover the topics listed below from an implementational, hands-on perspective.

- The OSI Network model
- Designing network protocols, IETF RFCs
- MAC addressing, Layer 2 discovery,
- Ethernet packet structure
- ARP, IP discovery, DHCP
- UDP/TCP sockets, TCP flow control
- Python socket programming
- WAN service discovery, authentication
- WiFi

Course Outline

1. What happens when you type in `www.google.com`?
- Course overview
- An overview of the OSI model

2. Layer 2/3
- Ping, ARP
- A home-brew IP client discovery protocol
- `Workshop 1:` Proto P2P chat app using NetCat.

3. Layer 4
- Sockets, UDP/TCP
- A home-brew chat session protocol
- `Workshop 2:` P2P chat app using Python sockets.

4. Back to Layer 3
- DHCP, service discovery
- A home-brew DHCP and address book system
- `Workshop 3:` P2P chat app using broadcast discovery and address books

5. Back to Layer 2
- Layer 2 packet exchanges, ARP-lessness
- A hypothetical home-brew L2 chat session protocol

6. Layer 4
- A home-brew TCP flow control system
- `Workshop 4:` P2P file sharing app using homebrew TCP flow control

7. WiFi
- Intro to the 802.11 and WiFi stacks
- `Workshop 5:` WiFi rate selection using P2P Chat app

8. Authentication
- Layer 6, security, SSL
- Diffie-Hellman Key Exchange
- A home-brew Internet-based key exchange protocol
- `Workshop 6:` Secure P2P chat.

9. WAN
- Layer 7
- WAN protocols
- Modern Internet applications for browsing, searching, chatting and streaming

Grading

- Short Weekly Quizzes 20%
- Presentation 20%
- Workshops 30%
- Final project 30%

`Workshop` assignments are due by the beginning of the following lecture. **Late workshops will receive no credit.** After each workshop deadline, submissions may be open-sourced, and can be licensed by others in the class to build on for future assignments. Only the top 5 (out of a total of 6) workshop grades will be counted for the course.

Prerequisites

Basic understanding of Python programming, and an available working MacOS or Linux laptop (USB boot acceptable).

Course Reading

Complementary reading for each topic may be assigned from academic publications in the lecture notes.