

# ECE 18-845: Internet Services

David R. O'Hallaron  
Carnegie Mellon University

Spring 2017

## 1 Organization

### Instructors

Prof. David R. O'Hallaron, GHC 7517, droh@cs.cmu.edu  
Office hours: Mon 4-5pm (or by appointment).

TA: Nate Horan, nhoran@andrew.cmu.edu  
Office hours: TBD

### Class

Mon Wed 2:30-3:50pm, DH 2105

### Resources

**Web page:** [www.ece.cmu.edu/~ece845](http://www.ece.cmu.edu/~ece845)

**Class mailing list:** [18-845@cs.cmu.edu](mailto:18-845@cs.cmu.edu)

**Course directory:** [/afs/ece/class/ece845](http://afs/ece/class/ece845)

## 2 Objectives

Not so long ago, a busy web server might be expected to handle thousands of requests. And if it went down for awhile, few people were seriously inconvenienced. Today it's a much different story. Busy servers are expected to handle hundreds of millions of requests a day, 24 hours a day, 7 days a week. Slow response times and downtimes cost money, jobs, and business.

Our aim in 18-845 is to understand the technology and research issues involved in providing scalable Internet services. The general research area we will be studying is system-level software services that run inside large data centers. We will study the techniques for building services, trying to gain a deep understanding

of the advantages and disadvantages of the various approaches. We will place particular emphasis on how to measure and evaluate the effectiveness of various approaches.

### 3 Textbook

There is no required textbook for 18-845.

The following pair of books are the standard references for Linux programming and Unix network programming:

Michael Kerrisk, *The Linux Programming Interface: A Linux and UNIX System Programming Handbook*, No Starch Press, 2010.

W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, *Unix Network Programming: The Sockets Networking API*, Volume 1 (3rd Edition), Prentice Hall, 2003.

The CS:APP3e textbook, which is on permanent reserve in the Engineering and Science Library in Wean Hall, covers system-level programming topics such as dynamic linking, process control, Unix I/O, the sockets interface, writing Web servers, and application level concurrency and synchronization:

R. Bryant and D. O'Hallaron, *Computer Systems: A Programmer's Perspective, 3rd Edition*, Pearson, 2016.

### 4 Course Organization

18-845 is a computer systems research course, and we will approach the material as system researchers. This means diligent investigation of existing work, striving for deep understanding of ideas and their advantages and disadvantages, careful measurement and quantitative evaluation, and sharing our ideas with others in discussions, posters, and oral presentations.

Your participation in the course will involve five forms of activity. The percentage of your final grade is shown in parentheses. Note that there are no exams.

1. Participating in the classroom discussions (15%).
2. A 3-week individual project (15%).
3. Leading classroom discussions (10%).
4. Research paper critiques (20%).
5. A 10-week group project paper and poster presentation (40%).

## **Classroom Participation**

A research course like this is only as good as the intellectual energy and involvement of the students. During each class, we will sit in a circle and talk about the papers. It is not the kind of course where you can show up sporadically, cram for a few exams, and expect to learn much.

*You are required to attend every class and participate actively in the discussions.*

## **Individual Project**

The first part of the course is a 4-week individual project, assigned by your instructors. You will work by yourself. This is a warm-up exercise whose purpose is to give you some practical context when we study the research issues later in the course.

## **Leading Classroom Discussions**

The course primarily involves reading and discussing research papers. Students will take turns leading the discussions on particular papers and topics. As the discussion leader, your job is provide any necessary technical background (your instructor can help you here), summarize the problem being addressed and the key ideas that address the problem, compare and contrast the different approaches, and generally lead the discussion with the rest of the class.

*Since they tend to inhibit participation, Powerpoint slides are banned from classroom discussions.*

## **Paper Critiques**

Students who are *not* leading the discussion that day should prepare a 1-page critique of the paper(s) denoted by a "\*" on the course Web page. The purpose of the critiques is to ensure that you have carefully read and actively thought about the papers being presented that day, so that you are armed with good questions to stimulate the discussion. See the course Web page for instructions on writing your critiques.

### **Handing in your critiques:**

Critiques are due *before* the class in which the papers are discussed. Bring a *hardcopy* of your critique (no email) to class and give it to the TA. The TA will grade it and return it to you the next class. Your two lowest critique scores will be dropped.

**Late policy:** Late or emailed critiques are not accepted.

## **Group Project**

This part of the course will culminate in a group project of your choice. You may work in groups of one or two. There is a lot of leeway with the choice of group projects. Your instructors will work with you to come up with a suitable project.

We will run the group project like a research workshop:

- Ten days before the end of the term, each group will submit a report describing their project. Reports are 10 pages max and must follow the official ACM Proceedings format. Word and Latex templates are available on the course Web page.
- Each paper will then be formally reviewed by four reviewers, your instructor, your TA, and two anonymous classmates chosen by your instructors.
- Each group will then incorporate the insights from the reviews to produce the final version of their report, which is due on the Sunday after the last day of class.
- On the last day of class, we will hold a reception/poster session, where each group will present their work to their instructors and classmates and run demos.

You will be graded on the quality of your report, your poster, your demo, and your reviews of the other students' reports.

## **5 Getting Help**

You are always welcome to visit or phone your instructors if you have questions, problems, comments, or just want to chat.

Nate Horan, the course TA, is your primary point of contact for questions related to specific programming and systems issues.

Prof. O'Hallaron normally works with his office door open, and welcomes visits from students whenever his door is open. However, if his door is closed, then he is busy with a meeting or a phone call and should not be disturbed.

We will use the course Web page the central repository for all information about the class.

## **6 Cheating**

In general, the only thing you are not allowed to do is copy code from another student or group. If you do this, you will automatically fail the course. Otherwise, we encourage to talk to and help each other as much as possible.

## **7 Class Schedule**

The up-to-date course schedule is maintained on the course Web page.