

CSCI 553 Networking III (Unix Software Development and Network Programming)

INSTRUCTOR:

Derek Harter, Ph.D., Associate Professor, Department of Computer Science
Texas A&M University – Commerce
Office: Jour 208
E-mail: Derek.Harter@tamuc.edu
Office Hours: T-Th 1:30-5:00, or by appointment

CLASS MEETINGS:

Online lectures and materials through eCollege.

DESCRIPTION:

This course is designed to introduce advanced concepts of programming and software development in UNIX-based computing environments. The UNIX model of networking, inter-process communication (IPC), and TCP/IP sockets will be a secondary focus, as an example of applying software development tools and techniques to developing software in a UNIX environment. The class will include programming projects involving intensive coding of applications to demonstrate the development of software in a Unix software development environment.

REQUIREMENTS AND OBJECTIVES:

The course is one of the three required courses for completion of the networking track of the Masters program in Computer Science at TAMU-C. The topic of focus in this course is UNIX network programming. This course also includes an intensive introduction to basic software development tools and practices for programmers, scientists and engineers, in the UNIX programming environment.

In the first half of the course we will introduce students to standard programming and software development tools in the UNIX environment. This includes a detailed look at using the UNIX shell command line. We will also look at development tools that help to streamline and automate the development process, including revision control systems (subversion), automated build tools (make and ant), and debuggers, linkers, system libraries and other standard GNU gcc toolchain functions.

In the second half of the course the student will apply these tools and techniques to developing simple client/server applications using the standard UNIX network programming tools and protocols. We will be focusing on the standard Posix C socket library for doing client/server development. All programs and code in this course will be developed in plain C, using standard Posix system libraries and functions. As part of this course, you will complete labs and programming assignments through online submissions on eCollege and through a source code repository.

TEXTBOOKS/READINGS:

- [SWC] *Software Carpentry* by Greg Wilson
Web Course: <http://software-carpentry.org/>
- [UTB] *UNIX Tutorial for Beginners* by Surry University
Web Course: <http://www.ee.surrey.ac.uk/Teaching/Unix>
- [AUP] *The Art of Unix Programming* by Eric S. Raymond
Addison Wesley Professional, 2003, ISBN: 0-13-142901-9

- [Free CC license version: http://www.fags.org/hdocs/artu](http://www.fags.org/hdocs/artu)
[SVN] *Version Control with Subversion* by Collins-Sussman, Fitzpatrick & Pilato
O'Reilly Media, 2002
(e.g. The Subversion Red Turtle Book)
[Free CC license version: http://svnbook.red-bean.com](http://svnbook.red-bean.com)
[LLD] *Linkers and Loaders* by Sandeep Grover
Linux Journal
[Web Journal article: http://www.linuxjournal.com/article/6463](http://www.linuxjournal.com/article/6463)
[GDB] *Linux software debugging with GDB* by David Seager
IBM.com Articles
[Web Journal article: http://www.ibm.com/developerworks/library/l-gdb](http://www.ibm.com/developerworks/library/l-gdb)
[UNP] *UNIX Network Programming, Vol. 1: The Sockets Networking API, 3/E* by Richard Stevens, Bill Fenner, and Andrew M. Rudoff
Addison-Wesley Professional, 2003, ISBN: 0-13-141155-1
[BEJ] *Beej's Guide to Network Programming* by Brian "Beej Jorgensen" Hall
Web CC 3.0 License Book:
<http://www.beej.us/guide/bgnet/output/html/multipage/index.html>

PREREQUISITES:

CSCI-525, CSCI-530

EVALUATION:

Your grade for the course will be based on the following (approximate) percentages:

Two Exams: 50%

Labs / Programming Assignments (appx 8): 40%

Quizzes and eCollege participation: 10%

STUDENTS WITH DISABILITIES ACT COMPLIANCE:

Students requesting accommodations for disabilities must go through the Academic Support Committee. For more information, please contact the Director of Disability Resources and Services, Hallady Student Services Bldg., Room 303D, (903) 886-5835.

ACADEMIC ETHICS:

"All students enrolled at the University shall follow the tenets of common decency and acceptable behavior conducive to a positive learning environment." (See Student's Guide Handbook, Policies and Procedures, Conduct).

Ethics also includes the issue of plagiarism, and copying code for programming assignments is just as serious as any other type of plagiarism. If you are caught sharing or using other people's work in this class, you will receive a 0 grade and a warning on the first instance. A subsequent instance will result in receiving an F grade for the course, and possible disciplinary proceedings.

ATTENDANCE POLICY:

Students are expected to follow all instructions and visit eCollege regularly many times weekly to complete the materials for this online course. If a student is unable to submit assignments by the due date for the assignment, they are expected to make alternative arrangements to assure that the assignment is turned in ON TIME, before the assignment is actually due. Any student wishing to withdraw from the

course must do so officially as outlined in the class schedule. THE INSTRUCTOR CANNOT DROP OR WITHDRAW ANY STUDENT.

COURSE REQUIREMENT DEADLINES:

Credit will be given for ONLY those exams, programs, and/or projects turned in no later than the deadline as announced by the instructor of this class, unless prior arrangement has been made with the instructor.

TENTATIVE SCHEDULE

Wk	Date	Topic / Activity	Reading	Lab/Assg
1	Jan 13	Intro, Syllabus, Linux Accounts		
2	Jan 20	Linux Overview & Distros	AUP 1 AUP 3	
		Installing and Administering		
3	Jan 27	Shell Basics	UTB (Intro, Lessons 1-6 & 8) SWC (The Shell)	1: Unix Shell and tools
4	Feb 3	More Advanced Shell	AUP 11	
5	Feb 10	Version Control w/ Subversion	SWC (Version Control) SVN 1,2	2: Subversion
6	Feb 17	GCC Compiler / Linker Toolchain	UTB (Lesson 7) LLD	3: Linkers, compilers, loaders
		Unix systems programming		
7	Feb 24			
8	Mar 3	Exam 1		
	Mar 10	Spring Break		
		Spring Break		
9	Mar 17	Automated Builds w/ Make	SWC (Make)	4: Make auto builds
10	Mar 24	Debuggers	GDB	5: Unix command tools & Debugging
		Eclipse IDE in Unix	BEJ 2,3	
11	Apr 1	Introduction to Networking	UNP 1,2 BEJ 1	6: System calls, libraries
12	Apr 7	C socket library API	UNP 3 BEJ 2,3	
		C structures, functions & pointers review		
13	Apr 14	Elementary TCP Sockets	UNP 4 BEJ 5	7: Simple TCP Client/Server
		TCP Client/Server example	UNP 5 BEJ 6	
14	Apr 21	I/O Multiplexing	UNP 6	8: I/O system calls, wait and signals
15	Apr 28	Socket Options	UNP 7	
	May 5	Exam 2		

STUDENT LEARNING OUTCOMES

1. Learn about Unix software development tools.
2. Learn core Unix shell commands.
3. Learn Unix systems programming, signals, forking, stdio libraries, etc.
4. Learn about concurrent and distributed computing.
5. Become familiar with TCP and UDP sockets.
6. Be able to create simple TCP Client/Server applications using Posix C sockets library.

Learning outcomes will be measured through mapping assignment and test questions to specific outcome items, as well as through exit surveys of student experiences with the outcome familiarity.