TLEN 5410: Network Management and Operations

COURSE PURPOSE

Given the ever growing difficulty of managing networks it is important for telecommunications engineers to obtain the ability to develop tools that can assist with common network management tasks, or extend upon existing management solutions to meet organizational needs not being adequately addressed.

The primary purpose of this course is to provide students with fundamental network management principles, and to teach them introductory programming techniques for use with solving common network management problems such as:

- Managing configuration files & storing log information for multiple network devices.
- Identifying patterns in network behavior to prevent problems from occurring.
- Alerting management when faults or unexpected changes occur within the network.
- Generate and dissect packets to test performance or identify undesirable network traffic.

Essentially, this course offers students a hands-on experience programmatically managing network hardware and essential network services through the use of cross-platform scripting and management protocols such as SNMP, Netflow, and Netconf.

Students with little or no programming experience will learn scripting by replicating functionality provided in common management suites such as HP OpenView, Nagios, Zennos, IBM Netview and others.

REQUIRED TEXT

- Think Python by Allen B. Downey, O'Reilly Media, August 2012, ISBN 144933072X. A free manuscript is available at the author's website.

REFERENCE TEXT


EXPECTED COURSE OUTCOMES

- Explain the FCAPS Model of Network Management
- Create programs using Procedural and Object-Oriented Programming concepts
- Apply the fundamentals of Python to solve practical problems in Network Management
- Show how management information is stored, accessed, and modified within a network element
- Demonstrate proficiency with common network management protocols & tools
- Apply effective debugging techniques to resolve syntax, runtime, & semantic errors

TENTATIVE COURSE OUTLINE

Module 1: Introduction to Network Management & Python

Introduction to the course, and an overview of Network Management, programming with Python, and the basics of procedural programming.
Duration: 1 Week

Learning Objectives
- Explain and show examples from 3 categories of errors (syntax, semantic, runtime) which can occur when developing a program.
- Recognize the importance of network management and programming skills in the Telecommunications field.
- Use values, variables, conditionals and basic arithmetic operators in Python.

Module 2: Procedural Programming and Basic Data Structures

Presents common data structures (lists, tuples, dictionaries), guidelines for their usage and application towards various problems.
Duration: 1 - 2 Weeks

Learning Objectives
- Apply effective debugging techniques to identify and resolve issues in a program.
- Use iteration to walk through sequences and mappings.
- Decompose a problem into basic steps or procedures and implement them as functions.
- Identify common components in a management network such as a manager, and agent.
Module 3: Object Oriented Programming with Python
Introduces the basic concepts of Object-Oriented Programming (OOP), such as objects, classes, attributes, methods, and operator overloading.
Duration: 2 - 3 Weeks

Learning Objectives
- Explain the benefits & drawbacks of object-oriented programming.
- Apply object-oriented programming techniques to simulate Ethernet.
- Describe what attributes, methods, and operator overloading are.
- Explain the key difference between objects and classes.
- Identify and apply (3) special methods that can be implemented on objects in Python.

Module 4: Introduction to the Simple Network Management Protocol
Describes a framework for categorizing network management problems, how management information is organized, and provides experience using existing tools or developing new ones to query/receive or modify information from network elements.
Duration: 2 - 3 Weeks

Learning Objectives
- Explain the concept of a management information base, and how managed objects are organized in SMIv1/2.
- Describe the difference between scalars and columnars.
- Demonstrate the usage of the get, get-next, set, and trap SNMP v1 operations by using the Net-SNMP command line tools and libraries within a Python program.
- Contrast asynchronous and synchronous communication with a management agent.
- Use threshold crossing alarms to alert manager of certain events.

Module 5: Network Performance Analysis
Presents techniques to capture and analyze high level details about traffic transmitted on a production network using the Netflow protocol. Introduces students to the basics of graphing with Python.
Duration: 1 Week

Learning Objectives
● Analyze traffic reported by netflow and identify what type of traffic was exchanged and with which other entities on the local network, or Internet.

● Explain the concept of an IP flow and the characteristics that uniquely identify them.

● Create line, pie, bar, and radar charts in Python with matplotlib.

● Contrast Netflow with SNMP and describe which functions they can be used to fulfill in the FCAPS model.

**Module 6: Configuration Management**
Explores the various aspects of accessing, manipulating, and auditing configuration information on network elements through a next generation protocol called Netconf.
Duration: 2 Weeks

**Learning Objectives**
● Demonstrate the use of ElementTree to parse or modify configuration formatted in XML.

● Contrast Netconf with SNMP and describe which functions they can be used to fulfill in the FCAPS model.

● Illustrate how standard in, standard out, and standard error can be used as a primitive communication mechanism between processes.

● Develop a tool using Netconf to audit configuration of network elements, and rectify portions of configuration that are out of compliance from a given policy.

**Module 7: Software Defined Networking**
Covers the basic concepts of software defined networking, and provides students with an exposure to the OpenFlow communication protocol for modification of the forwarding plane on network elements.
Duration: 1 Week

**Learning Objectives**
● Explain the role of a controller, network operating system, and packet forwarding hardware in a software defined network, and be able to diagram the communication process between them.

● Develop code for a Python based controller capable of filtering packets based upon source or destination MAC address
ATTENDANCE POLICY

Attendance is a requirement of all on-campus students. At the discretion of the instructor, your grade may be penalized as much as 5% if there are more than 3 absences. Please note that missing the first week of classes without prior approval from the instructor will result in being administratively dropped from the course.

COURSE REQUIREMENTS

Homework, labs, & project must be completed in a prompt and timely manner. All code must run properly on the class virtual machine, which is based on Python 2.7 running in a Linux environment. Supplemental material will be provided in the course to help you learn how to use Linux.

Completed course work must be submitted through Desire2Learn, and assignments that require you to turn in multiple files should be submitted contained in **zip format ONLY**. ALL submitted filenames must start with your Identkey username, example: jeijf2456_hw4.py. Submissions not meeting these requirements will not be accepted and will require resubmission.

All students must complete the programming honor code quiz before a grade will be issued. The CU honor code is taken very seriously in this class and all violations will be reported.

Grading Criteria

**Homework Assignments (30%)**
Homework assignments will consist of reading from the required texts, watching videos, & completing exercises. Detailed homework assignments will be posted in Desire2Learn. Please complete the homework before the date & time due as this will ensure that you are sufficiently prepared to work on the in-class labs.

**Lab Assignments (30%)**
Lab assignments are challenging applied programming problems that will be completed during class time with a partner. Detailed assignments will be posted in Desire2Learn. These assignments will be based on real-world tasks that you might be asked to do in a work environment as a network administrator.

**Project (20%)**
The course project will be determined by the student, and the scope should be equivalent to that of a lab assignment. A proposal must be turned in to the professor by the date posted in the course outline. The project is an individual assignment and may not be completed in teams. Assistance should not be provided by peers unless otherwise instructed.

**Peer Review (10%)**
At the completion of each lab problem assigned in-class you'll be asked to evaluate your partner's performance. The sum of these evaluations will make up the peer review portion of the
final grade. Peer reviews will cover areas such as level of preparedness, amount of contribution & overall participation. Grades will only be assigned for peer reviews if the lab work was completed with a partner during the class time allocated for the work. Please inform the instructor if you are unable to find a partner.

Note: The peer reviews do not apply for distance students, and as they are not required to work on labs with a partner. The quiz grade is increased to 20% of the final grade for distance students. It is strongly encouraged that distance students work together with a partner when possible.

Quizzes (10%)
In order to properly assess your progress in this course there will be a set of quizzes in this class. It is strongly recommended that you complete the associated homework(s) and/or lab(s) before attempting the assigned quiz.

Grading Scale

<table>
<thead>
<tr>
<th>Numeric Score</th>
<th>Letter Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 - 95</td>
<td>A</td>
</tr>
<tr>
<td>94 - 90</td>
<td>A-</td>
</tr>
<tr>
<td>89 - 87</td>
<td>B+</td>
</tr>
<tr>
<td>86 - 83</td>
<td>B</td>
</tr>
<tr>
<td>82 - 80</td>
<td>B-</td>
</tr>
<tr>
<td>79 - 77</td>
<td>C+</td>
</tr>
<tr>
<td>77 - 73</td>
<td>C</td>
</tr>
<tr>
<td>&lt; 73</td>
<td>C-</td>
</tr>
</tbody>
</table>

LATE ASSIGNMENT POLICY
Points will not be deducted for assignments turned in late, however assignments submitted over 72 hours late will not be graded. The one exception to this policy is the project; late projects will not be accepted without prior approval and may be subject to a penalty of up to 6% of the project grade. This policy is in place to ensure fairness to all students, and to prevent excessive burden on myself and the student assistants.

I understand that there may be extenuating circumstances where you will be unable to attend class, or may be late on an assignment. Please try your best to discuss these circumstances with me at least one week in advance.

If you are unsure of what counts as an extenuating circumstance, here are some examples:
- Serious health illness, or family emergency
- Religious observances
- Extraneous work commitments
- Transportation difficulties
- Jury service
- Having to care for someone
- Hospital appointment that clashes with a deadline or examination
- Legal problems

STANDARD CU POLICY STATEMENTS

Disability Services
If you qualify for accommodations because of a disability, please submit to me a letter from Disability Services in a timely manner so that your needs may be addressed. Disability Services determines accommodations based on documented disabilities. Contact: 303-492-8671, Willard 322, or at their website.

Conflicts with Religious Holidays
Campus policy regarding religious observances requires that faculty make every effort to reasonably deal with all students who, because of religious obligations, have conflicts with scheduled, assignments or required attendance. Please notify me at least two weeks in advance of any conflict between a religious holiday and a class activity.

Classroom Behavior
Students and faculty each have responsibility for maintaining an appropriate learning environment. For that purpose, students are expected to be punctual, polite and prepared to engage in discussion with the instructor and classmates. Please turn off your cell phones and place in your bag, pockets, etc., prior to entering the class. Laptops are permitted in class for the purpose of working on the lab assignments.

Students who fail to adhere to such behavioral standards may be subject to discipline. Faculty have the professional responsibility to treat all students with understanding, dignity and respect, to guide classroom discussion and to set reasonable limits on the manner in which they and their students express opinions.

Professional courtesy and sensitivity are especially important with respect to individuals and topics dealing with differences of race, culture, religion, politics, sexual orientation, gender variance, and nationalities.
I will gladly honor your request to address you by an alternate name or gender pronoun. Please advise me of this preference early in the semester so that I may make appropriate changes to my records.

Please see the related classroom policies, and student conduct code for further information.

**Discrimination and Sexual Harassment**

The University of Colorado at Boulder policy on Discrimination and Sexual Harassment applies to all students, staff and faculty. Any student, staff or faculty member who believes s/he has been the subject of discrimination or harassment based upon race, color, national origin, sex, age, disability, religion, sexual orientation, or veteran status should contact the Office of Discrimination and Harassment (ODH) at 303-492-2127 or the Office of Judicial Affairs at 303-492-5550.

Information about the ODH and the campus resources available to assist individuals regarding discrimination or harassment can be obtained at http://www.colorado.edu/odh

**CU Boulder Honor Code**

All students of CU Boulder are responsible for knowing and adhering to the academic integrity policy of this institution, without exception. Violations of this policy may include: cheating, plagiarism, aid of academic dishonesty, fabrication, lying, bribery, and threatening behavior. All incidents of academic misconduct shall be reported to the Honor Code Council (honor@colorado.edu; 303-725-2273).

Students who are found to be in violation of the academic integrity policy will be subject to both academic sanctions from the faculty member and non-academic sanctions (including but not limited to university probation, suspension, or expulsion).

Other information on the Honor Code can be found in the Official CU Policy, and at the Honor Code Councils website.