

# COMP-4384 Course Syllabus

March 26, 2026

## 1 Course Information

Course Name	COMP 4384: Secure Software Engineering
-------------	--

## 2 Course Description

Secure software engineering provides students with the background of software security, why security matters, and how they can take security into account when writing new software programs.

## 3 Textbook and Materials

You will need a good internet connection and a laptop that meets DU specifications. (See <https://www.du.edu/it/support/how-to/students/laptops>). For technical support in using Canvas, please go to <http://otl.du.edu/knowledgebase/canvas>

## 4 Course Learning Objectives

By completing this course, students will be familiar with the following specific topics:

- Common vulnerabilities in software
- Security by design, domain driven software security, defense in depth
- Using domain models
- Creating models in domain driven design for security
- Programming practices that promote security (immutability, non-blocking processes, etc.)
- Contracts, context, validation of states and user input
- The builder pattern for state verification
- Software testing for security (unit testing, valid/invalid inputs, fuzzing)

## 5 Course Outcomes

Upon successful completion of this course, students will have learned the following overall information:

1. What kind of vulnerabilities are common in software, and how can they be prevented. A theoretical understanding of those vulnerabilities will be gained as well as hands-on experience with one or more instances of those vulnerabilities.
2. How to assess whether software contains particular vulnerabilities and the methods used to do so (e.g. dynamic and static analysis, fuzzing).
3. Students will also have gained the ability to converse about software and software vulnerabilities including common terms and definitions.

## 6 Program Level Goals

Courses in the cyber security MS program, including this one, should contribute to overall program level outcomes for students. This course contributes to the following program level goals:

1. Apply programming and core CS topics; including network programming, algorithms and data structures, and computer organization.
2. Utilize and comprehend at least one compiled programming language (C/C++) and two or more interpreted/scripting languages (Bash, Python).
3. Understand how modern operating systems work (with a leaning towards Unix style OSes). Be familiar with their security components (identification, authentication, access controls, auditing) and how to configure them.
4. Take on the role of an attacker (with respect to penetration testing/red teaming), and be familiar with the tools used by ethical hackers in ascertaining the security of a system or network of systems.
5. Employ common tools and techniques used in the cyber security field. For example, configuring firewalls, IDSes, on-host security software and vulnerability scanners.
6. Understand common cyber security vocabulary, and understand and intelligently discuss cyber security topics (incidents, advisories, etc.)

## 7 Knowledge Units

This course covers the following NSA cybersecurity Knowledge Units:

1. Operating Systems Concepts (OSC, Core Technical CDE)
2. Low-level Programming (LSP, Optional KU)
3. Systems Programming (SPG, Optional KU)

## **8 Attendance Policy**

Regular attendance is expected. Students are responsible for all material covered in class, including announcements and handouts.

## **9 Academic Integrity**

All work submitted for this course must be the student's own original work. Any instance of plagiarism or cheating will be dealt with according to the university's academic integrity policy. Refer to the Student Rights and Responsibilities, as well as the University of Denver Student Honor code, here: <https://studentaffairs.du.edu/student-rights-responsibilities>

## **10 Disability Services**

If you have a disability that may affect your ability to complete the work for this course, please contact the Disability Services office at: <https://studentaffairs.du.edu/disability-services-progr>