CS 763 Secure Software Development

Department of Computer Science

Metropolitan College

Boston University

Spring 2024 Syllabus

Instructor Information

Name: Yuting Zhang
Office: 1010 Commonwealth ave, Rm 322
Email: danazh at bu dot edu
URL: http://people.bu.edu/danazh

Course Information

Lecture time and location

Tuesday 6:00-8:45, CAS 223

Prerequisites

At least two 500- level (or above) computer programming-intensive science courses or instructor's consent. As this is an advanced 700 level course, you should be familiar with programming and software development.

Reference Books:

Wenliang Du, Computer & Internet Security: A Hands-on Approach 2nd Edition. May 1, 2019.

Gary McGraw. Software Security: Building Security In. Addison-Wesley Professional; 1 edition (February 2, 2006)

Michael Howard, David LeBlanc & John Viega . 24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them (Networking & Comm - OMG). McGraw-Hill Education; 1 edition (September 24, 2009)

Additional Books:

Ross Anderson. Security Engineering. Wiley. 2nd Edition. (https://www.cl.cam.ac.uk/~rja14/book.html)

Mathias Paye. Software Security Principles, Policies, and Protection. (January 2019, v0.33) (https://nebelwelt.net/SS3P/softsec.pdf)

Theodor Richardson & Charles Thies. Secure Software Design. Jones & Bartlett Learning. 2013

Dafydd Stuttard & Marcus Pinto. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, 2nd Edition. Wiley.

Other Reading Materials

- Microsoft Secure Development Life Cycle: https://www.microsoft.com/en-us/sdl/
- OWASP SAMM Project: https://www.owasp.org/index.php/OWASP_SAMM_Project
- OWASP TOP 10: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
- Developer Guide: https://www.owasp.org/index.php/Category:OWASP_Guide_Project
- Testing Guide: https://www.owasp.org/index.php/Category:OWASP_Testing_Project
- Secure Coding Practice Guideline: https://security.berkeley.edu/secure-coding-practice-guidelines
- Seed Labs: https://seedsecuritylabs.org/

Please find more reference materials on the course blackboard website (https://onlinecampus.bu.edu**) (under the Content/References folder)**

Description

Overview of techniques and tools to develop secure software. Focus on application security. Topics include secure software development processes, threat modeling, secure requirements and architectures, vulnerability and malware analysis using static code analysis and dynamic analysis tools, vulnerabilities in C/C++ and Java programs, Crypto and secure APIs, vulnerabilities in web applications and mobile applications and security testing. Hands-on lab and programming exercises using current tools are provided and required. 4 credits.

Objectives

At the end of the semester, students are expected to

- Explain secure software development process and activities in the process.
- Explain security principles and how to apply them in software development.
- Explain risk management and threat modeling and identify security requirements and threats in real world projects.
- Explain, identify and apply DevSecOps best practices in real world applications
- Explain, identify common vulnerabilities and corresponding mitigations in programs written in high-level programming languages such as C/C++. Java, Python.
- Explain basic cryptographic algorithms and use right crypto APIs properly in the application.
- Explain the security mechanisms and identify common vulnerabilities and corresponding mitigations in web applications and mobile applications.
- Design and conduct security testing for real world applications.

Course Requirements

- Class participation
- Reading and study
- Assignments
    - Discussion
    - Labs
    - Project
- Quizzes and Exams

Class Schedule

(This is a tentative class schedule. It is subject to change according to the progress of the class and the feedback of the student.)

Module & Date	Topics	Discussion	Project Assignments	Labs	Quizzes and Exams
1 01/23 01/30 02/06	Intro to Secure Software Development Process and DevOps	1. Cyber attack Due: 02/13	1: Project proposal and DevSecOps pipeline setup Due: 02/13	1: explore SAST Due: 02/27	
2 02/20 02/27	Security Principles and Design	2. Project Proposal Due: 02/27	2: Security requirements and threat modeling, SAST tools Due: 03/19		Quiz1 Due: 03/05
3 03/05 03/19	Secure coding and Language level vulnerabilities	3. MITRE ATT&CK Due: 03/12		2: buffer overflow attack (seed lab) Due: 04/02	
4 03/26 04/02	Crypto Basics and Sins	4. Project Progress Due: 04/02	3: Implementing security requirements Using Crypto APIs. Due: 04/16		Quiz2 Due: 04/02
5 04/09 04/16	Web Application Security	5. OWASP Top 10 Due: 04/16		3: XSS attack (seed lab) Due: 04/16	
6 04/23	Security Testing Reverse engineering Review	6. Project Final Submission Due: 04/30	4: SAST and/or DAST tools to analyze the project Due: 04/30		Quiz3 Due: 04/30

04/30	Final Presentation				
05/07	Final Exam				

Course Policies

Grading Policy

The grade that a student receives in this class will be based on class participation, in-class exercises, assignments, quizzes, the final project and the final exam. The grade is broken down as shown below. All percentages are approximate and the instructor reserves the right to make necessary changes.

- 5% on the class participation
- 9% on the discussion forum
  (6% on general discussion, 3% on project related discussion)
- 24% on lab assignments
- 28% on the project
- 9% on quizzes
- 25% on the final exam

Letter grade/numerical grade conversion is shown below:

A (95-100)	A- (90-94)	
B+ (85-89)	B (80-84)	B- (79-77)
C+ (74-76)	C (70-73)	C- (65-70)
D (60-65)	F (0 – 59)	

Attendance Policy

Attendance is expected at all class meetings. You are responsible for all materials discussed in class. In general, no makeup quizzes and exams will be given unless an extremely good, verifiable reason is given in advance.

Assignment Late Policy

Every assignment has a due date. The late assignments will be penalized within a week with **3 points per day**. No assignments will be accepted one week after the deadline. It is the students' responsibility to keep secure backups of all assignments.

Academic Integrity

Academic conduct in general and MET College rule in particular require that all references and uses of the work of others must be clearly cited. All instances of plagiarism must be reported to the College for action. *For the full text of the academic conduct code, please check* https://www.bu.edu/academics/policies/academic-conduct-code/.