CIS 518: Secure Software Engineering, Sections 1 & 2

College of Computing, Grand Valley State University

Fall 2025

Contact Information

Instructor:	Dr. Jag Nandigam
Phone:	616-331-3639
Email:	nandigaj@gvsu.edu (*preferred*)
Office Hours:	Tuesday, 5:15PM – 6:00PM, DCIH 507
	Monday & Wednesday, 5:30PM – 6:30PM (on Zoom)
Zoom Link:	See Blackboard course site

Course Description

This course explores characteristics that make software secure and less vulnerable to attacks. Basic techniques for securing applications such as input validation, output encoding, memory management, race conditions, vulnerability analysis and testing, authentication, access control and secure database management will be covered in detail.

Multiple Delivery (MD) Course Information for International Students

F-1 and J-1 International Students who are enrolled in one or more Multiple Delivery (MD) courses or one MD course along with an online course in a standard semester must submit the Multiple Delivery Attestation form to confirm that they will attend the necessary MD course at least 25% in-person for the duration of the semester. 25% attendance in-person is approximately 4 weeks of in-person meetings in a standard Fall/Winter semester. When attending this way, the course is considered in-person for immigration purposes. Fully online participation in MD courses does not meet immigration requirements when a student is taking the MD course in combination with other MD courses or online courses.

Course Objectives

After completing this course, students should be able to:

- Describe characteristics of secure software.
- Apply principles of secure software development lifecycle.
- Describe software vulnerabilities such as buffer overflow, format string vulnerability, race condition vulnerability, SQL injection vulnerability, cross-site scripting vulnerability and defense mechanisms.
- Build input validation and output encoding into software.
- Design shellcode to test the presence of vulnerabilities and build countermeasures.

Course Prerequisites

- Admission to a College of Computing graduate program

Required Course Material

- Wenliang Du, ***Computer Security – A Hands-on Approach***, **Third Edition** (on Amazon)
- Lecture slides, articles, and handouts (available on Blackboard)

Additional References (not required)

- James N. Helfrich, Security for Software Engineers, Chapman and Hall/CRC.

- Gary McGraw, *Software Security – Building Security In*, Addison-Wesley.
- Brian Chess and Jacob West, *Secure Programming with Static Analysis*, Addison-Wesley.
- Michael Howard and David LeBlanc, *Writing Secure Code*, 2nd ed., Microsoft Press.
- Patrick Engebretson, *The Basics of Hacking and Penetration Testing*, Syngress.

Grading

Graded Activity	Weight
SEED Labs (10)	40%
Midterm Exam	25%
Final Exam	25%
Term Paper	10%
Total	**100%**

Grading Scale

A: >= 93%	**A-: >= 90%**	**B+: >= 87%**	**B: >= 83%**
B-: >= 80%	**C+: >= 77%**	**C: >= 73%**	**C-: >= 70%**
D+: >= 67%	**D: >= 60%**	**F: < 60%**	

Academic and Technology Support

- GV Technology Help Desk: helpdesk@gvsu.edu; 616-331-2101
- GV IT Support Pages: www.gvsu.edu/it

Additional Information

1. **Withdraw with a "W" Grade Deadline: Friday, November 7, 2025, 5:00pm**.

2. Click this link for **Fall 2025 Academic Calendar**

3. **Academic Honesty**: All students are expected to adhere to the academic honesty standards set forth by Grand Valley State University. In addition, students in this course are expected to adhere to the academic honesty guidelines as set forth by the School of Computing, the details of which can be found at https://www.gvsu.edu/computing/academic-honesty-30.htm.

4. **Special Needs:** If there is any student in this class who has special needs because of a disability, please let me know and also contact Student Accessibility Resources at https://www.gvsu.edu/accessibility/ at 616-331-2490.

5. **GVSU Course Policies**: This course is subject to the GVSU policies listed at http://www.gvsu.edu/coursepolicies/.
6. **Graduate Academic Policies and Regulations (IMPORTANT)**
   https://www.gvsu.edu/catalog/navigation/academic-policies-and-regulations.htm#anchor-63

7. **In Case of Emergency**: Fire: Immediately proceed to the nearest exit during a fire alarm. Do not use elevators. More information is available on the University's Emergency website located at http://www.gvsu.edu/emergency.

Course Schedule/Calendar (TENTATIVE)

Week	Week Of	Lecture/Discussion/Lab Topic(s)
1	08/25	Course Introduction; Software Security Fundamentals
2	09/01	Software Security Development Lifecycle Processes and Activities Requirements Engineering for Secure Software
3	09/08	Secure Design Principles and Threat Modeling Ubuntu 20.04 VM Setup for SEED Labs (https://www.seedsecuritylabs.org)
4	09/15	Linux Security Basics (Ch. 1); Set-UID Programs (Ch. 2)
5	09/22	Environment Variables and Attacks (Ch. 3) SEED Lab 1: Set-UID Programs and Environment Variables
6	09/29	Buffer Overflow Attack (Ch. 4) SEED Lab 2: Buffer Overflow Attack
7	10/06	Secure Coding with Static Analysis Return-to-libc Attack (Ch. 4) SEED Lab 3: Return-to-libc Attack
8	10/13	Format String Vulnerability (Ch. 6) SEED Lab 4: Format String Vulnerability **Midterm Exam (details to be discussed)**
9	10/20	**Fall Break**
10	10/27	Handling Input, Errors, and Exceptions Race Condition Vulnerability (Ch. 7) SEED Lab 5: Race Condition Vulnerability
11	11/03	Shellcode (Ch. 9) SEED Lab 6: Shellcode Development
12	11/10	Reverse Shell (Ch. 10) Software Security Testing SEED Lab 7: Reverse Shell
13	11/17	Web Security Basics (Ch. 11) Cross-Site Request Forgery (Ch. 12) SEED Lab 8: Cross-Site Request Forgery Attack
14	11/24	Cross-Site Scripting Attack (Ch. 13) SEED Lab 9: Cross-Site Scripting Attack
15	12/01	SQL Injection Attack (Ch. 14) SEED Lab 10: SQL Injection Attack
16	12/08	**FINAL EXAM: Tuesday, December 9th (Sections 1 & 2)**

Sample List of Topics for Term Paper (TENTATIVE)

You **are not** restricted to the following list of topics for the term paper. If another topic related to software security interests you, you are free to choose that topic, but please check with me first before proceeding with that topic.

- Microsoft SDL Threat Modeling Tool: https://www.microsoft.com/en-us/securityengineering/sdl/threatmodeling
- Attack Patterns
- Cryptography topics (see section IV in the textbook)
- Penetration Testing
- WebGoat: https://www.owasp.org/index.php/Category:OWASP_WebGoat_Project
- Google's Gruyere: https://google-gruyere.appspot.com/
- OWASP Zed Attack Proxy (ZAP) Tool https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project
- Static Application Security Testing (SAST) Tools: https://www.owasp.org/index.php/Source_Code_Analysis_Tools
  - OWASP WAP (Web Application Protection) Tool https://wiki.owasp.org/index.php/OWASP_WAP-Web_Application_Protection
  - FindSecBugs: https://find-sec-bugs.github.io/
  - FlawFinder: https://dwheeler.com/flawfinder/
  - SonarQube: https://www.sonarqube.org/
  - SonarLint: http://www.sonarlint.org/