CS 485: Counter-Hacking Techniques

New Jersey Institute of Technology
Ying Wu College of Computing
Department of Computer Science

Instructor Contact Information

Instructor: Haotian Zhang
Email: haotian.zhang@njit.edu
Course website: https://canvas.njit.edu/
Lecture Hours: **Mon, Thu 16:00 – 17:20**
Lecture Room: **Kupfrian Hall 204**
Office Hours: **Mon 13:30 – 15:30**
Office: GITC 4303

Statement of Purpose

This course covers advanced techniques that can be used for offensive or defensive goals in networks, computer systems, and applications. The course follows a "learning by doing" teaching approach through extensive use of virtual machines with vulnerable operating systems and applications. Topics covered include system memory organizations, CPU registers, assembly language fundamentals, GDB debugger, fuzzing-based security testing, development of local and remote Linux and Windows exploits, shellcode development, stealthy attacks, bypassing memory protection techniques, network and wireless hacking techniques, and ethical and legal implications of cyber-attacks.

The general topics covered are:

- Basic Computer Architecture
- Programming in x86 Assembly
- Buffer Overflow Exploits and Protections
- Format String Exploits
- Network-based Attacks
- Code Hardening
- Ethical Hacking
- Cybersecurity Ethics
- Malware Analysis and Reverse Engineering
- Hardware Side Channel Attack (Spectre & Meltdown)

Prerequisites

One of the following courses or approval from the instructor:

- CS 350
- CS 351

Recommended Topics:

- A basic understanding of computer architecture.

• Prior experience working with assembly language will be helpful (x86 specifically) but is not required.

• Some prior experience with the C programming language or the ability to self-learn.

• Some prior experience using a Linux Operating system and with basic Linux commands.

• An understanding of basic cryptographic principles such as hashing and public key infrastructure.

Required Materials

This course **does not require any text**; however, it will reference information and material from the following sources:

• "Hacking: The Art of Exploitation" 2nd Edition, by *Jon Erickson*,
ISBN-13: 978-1593271442 ISBN-10: 1593271441

• "Introduction to Computer Security", by *M. Goodrich* and *R. Tamassia, Addison Wesley*, 2010,
ISBN: 0321512944

• "Computer Systems: A Programmer's Perspective" 3rd Edition, by *Randal Bryant* and *David O'Hallaron*
ISBN-13: 978-0134092669 ISBN-10: 013409266X

• "Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software" 1st Edition, by *Michael Sikorski* and *Andrew Honig*
ISBN-13: 978-1593272906 ISBN-10: 1593272901

To participate in the course, students must either own or be able to use a computer with virtualization capabilities. It is suggested that the computer has at least 50-60GB of available storage space and a minimum of 4GB of memory to allocate to the guest virtual machine. While many operating systems can be used, Windows 10 or 11 is recommended. If a student's computer cannot support the course's virtual machine, they can use Amazon Web Services instead. As long as the student is careful about managing their Amazon EC2 instance uptime, the costs for the semester are expected to be low.

Course Learning Outcomes

After completion of this course, students will be able to:

• Describe cybersecurity and privacy mechanisms, standards, and state-of-the-art capabilities.

• Describe potential cyber-attacks and the actors that might perform them.

• Apply cyber defense methods to prepare a system to repel attacks.

• Perform security review of applications and systems.

• Create exploits to vulnerabilities identified on Linux or Windows systems.

• Design and implement system, network, and infrastructure-level solutions to ensure the security and privacy of communications and data against specific security threats.

• Describe the trade-offs between security, usability, and performance.

• Use standard security terminology to communicate effectively with other cybersecurity professionals.

• Analyze ethical considerations in cybersecurity, such as potential harm, red flags, common

challenges, contributions to the public good, and affected stakeholders.

Course Structure and Components

Announcements

Each week, typically when a new Module is released, a "Housekeeping" announcement that contains crucial information will be posted. This part of the course is by design, and students must read these announcements carefully. **Please be on the lookout for these and other ad hoc announcements and read them carefully; it is essential!**

Lectures

Lecture slides, supplemental materials are available weekly through Canvas modules. The lectures aggregate material from personal experience and training and the text references mentioned in the Required Materials section. This lecture materials are created by **Martin, Michael J**.

Students will find that most lectures include hands-on challenges administered with virtual machines. Attempting the challenges is not optional but a structured part of the course. Many of the topics presented are low-level, semi-advanced, requiring hands-on practice. The lectures follow a learn-by-doing approach where the students are given a concept/technique and then challenged to implement it themselves in a controlled environment.

Weekly Assignments

Students will conduct cyber vulnerability assessments for assigned programs each week. These assessments aim to foster critical thinking and encourage collaboration among students. The assessments require the students to identify vulnerabilities, create exploits, and communicate mitigation strategies to a hypothetical client. While the assignments are closely tied to lecture material, they will also require students to apply learned techniques to novel scenarios. Depending on the student's technical background and experience, they may need additional research. All assessment materials are in a virtual machine designed specifically for this course.

For the assessments, students will work in randomly assigned groups on Canvas, which the instructor may change throughout the semester. After completing the assessment and developing a working exploit, each group member must demonstrate the implementation of that exploit on their instance of the course virtual machine. To prove success, each student on the team must submit their flag on Canvas within one week of the assessment being assigned.

The teams will be given an extra week to work together on a report that conforms to the course's Writing Guide. The Writing Guide, which includes details on writing style and other report necessities, is readily available on Canvas. The reports must be composed on a Google Doc assigned to each time

by the instructor. The instructor will have access to the document and may examine the version history to assess each student's input, if necessary. Students who do not contribute sufficiently to the team may receive lower grades. A rubric is available for the reports to better understand their grades. For both submissions, students must include a textbased version of the challenges' flag files and a screenshot that showcases the output of the flag file printed to the terminal.

Projects

Two assignments are given more time to work on and carry slightly more weight. The first project requires students to showcase their ability to write a functioning program in x86 assembly language. For the second project, **students in CS 485** is **not required** but will reward for 10 bonus point for completion. Students must demonstrate their creativity by crafting an attack that exploits a buffer overflow vulnerability, allowing them to execute arbitrary code on a machine. The exploit must be able to bypass several modern buffer overflow defenses. Additionally, students must write a detailed technical report outlining their methods and procedures.

Exams

The midterm and final exams are closed-book exams. Many problems are open-ended questions that probe the student's knowledge of core concepts taught throughout the course. These exams will be require the student bring their own laptop with Respondus LockDown Browser.

Schedule of Assignments

Important Dates

Fall 2025 Academic Calendar

Sept	1	Labor Day. University Closed
Sept	2	First Day of Classes
Sept	8	Last Day to Add/Drop a Class
Sept	8	Last Day for 100% Refund, Full or Partial Withdrawal
Sept	9	W Grades Posted for Course Withdrawals
Sept	15	Last Day for 90% Refund, Full or Partial Withdrawal - No Refund for Partial Withdrawal after this date

Sept	29	Last Day for 50% Refund, Full Withdrawal
Oct	2	Wellness Day
Oct	20	Last Day for 25% Refund, Full Withdrawal
Nov	10	Last Day to Withdraw from Classes
Nov	25	Thursday Classes Meet
Nov	26	Friday Classes Meet
Nov	27	Thanksgiving Recess Begins. No Classes
Nov	30	Thanksgiving Recess Ends
Dec	11	Last Day of Classes
Dec	12	Reading Day
Dec	13	Saturday Classes Meet
Dec	14	Final Exams Begin
Dec	20	Final Exams End
Dec	22	Final Grades Due

Weekly Outline

Module	Topics and Assignments	Assignments Due At 11:59PM on Sunday at the end of each module.
0x0	**Course Introduction** Assigned: • Pippin Assessment • Syllabus Acknowledgment	• Pippin Assessment Flags • Syllabus Acknowledgment
0x1	**Basic Computer Architecture** • Project 1 Assigned	• Pippin Assessment Report • Pippin Assessment Peer Evaluation
0x2	**x86 Assembly Part 1**	

	x86 Assembly Part 2	
0x3	Work on Project 1	• Project 1 Due • Project 1 Peer Evaluation
0x4	**Exploitation Buffer Overflows Part 1** • Merry Assessment Assigned	• Merry Assessment Flag
0x5	**Buffer Overflows Part 2**	• Merry Assessment Report • Merry Assessment Peer Evaluation
0x6	Midterm Exam **The exact time, date, and location will be announced on Canvas.** • Sam Assessment Assigned	• Sam Assessment Flags
0x7	**Buffer Overflows Part 3** • Project 2 Assigned	• Sam Assessment Report • Sam Assessment Peer Evaluation
0x8	**Buffer Overflows Part 4** **Format String Vulnerabilities** Continue work on Project 2	
0x9	Introduction to Cybersecurity Ethics Continue work on Project 2	• Cybersecurity Ethics Discussion Posts are due by Friday at 23:59 • Responses to Peers are due by Sunday at 23:59
0xa	Network-Based Attacks and Code Hardening Frodo Assessment Assigned	• Project 2 Report • Project 2 Peer Evaluation
0xb	Introduction to Ethical Hacking	• Frodo Assessment Flag
0xc	Malware Analysis and Reverse Engineering	• Frodo Assessment Report • Frodo Assessment Peer Evaluation
0xd	Hardware Side Channel Attack Lab (Spectre & Meltdown)	
0xe	Final Exam **Time and location TBD by Registrar**	

Grading

Letter Grade	Significance	Calculation
A	Excellent	90% and above
B+	Good	85% - 89%
B	Acceptable	80% - 84%
C+	Marginal Performance	75% - 79%
C	Minimum Performance	65% - 74%
F	Failure	Below 65%

Grading based on assignments:

- Vulnerability Assessments (two parts each): 16%

  - Pippin Assessment
  - Merry Assessment
  - Sam Assessment
  - Frodo Assessment
- Cybersecurity Ethics Discussion Post 4%
- Legolas Assessment (Project 2) Bonus 10%
- Programming Project (Project 1) 20%
- Midterm Exam 30%
- Final Exam 30%

To earn full points on the vulnerability assessments, students must:

- Craft a working exploit to obtain the flag file.
- Each student must submit their own flag for both parts of the assignment.
- Collaborate with the team to complete the report that conforms to the course writing guide on Google Docs and contribute to the overall effort.
- Upon the instructor's request, demonstrate the exploit

Some assignments allow for late submission up to 8 hours after the official due date, with a 50% penalty for work submitted during this late window. Assignments that offer this option will have their late submission period clearly specified by their assignment due dates on Canvas. Other assignments must be turned in by the posted deadline and will not be accepted late under any circumstances. Submissions for assignments will not be accepted after the assignment has closed. Please consult Canvas for details on each assignment, and plan accordingly to avoid missing deadlines.

Peer Evaluations

This course's peer evaluations are pivotal in assessing teamwork skills during vulnerability assessments and projects. You will self-reflect and evaluate your peers by considering listening, communication, collaboration, idea development, and overall contribution. Your thoughtful feedback will contribute to a collaborative learning environment and team accountability.

While the peer evaluations will not count toward your final grade, they are mandatory for receiving credit on your vulnerability report and projects. Please ensure timely submission to facilitate the assessment process.

All assignments must be submitted on Canvas for accountability. Submissions will not be accepted through email or any other form.

Responsiveness and Availability

Under normal conditions, emails will be responded to within 24 hours. Students should send another email if there is no response within 48 hours. Assignments will be graded within two weeks of their due date.

IMPORTANT: Start the subject line of all emails with CS647: This will allow course-related emails to be filtered for prompt responses.

Students are encouraged to ask for help and attend office hours. Every student on the course comes from various backgrounds and experiences. Asking questions is excellent and can benefit the entire class. Students should ask questions via email and pay attention to announcements where common questions will be posted.

Students are encouraged to seek help whenever they need it. Every student comes from a different background and has unique experiences. Asking questions is an excellent way to enhance the learning experience. Those needing extra help must contact the instructor early in the modules. They should do so via email. Additionally, responses to common questions may be posted on the announcements page, so please watch for any updates.

Feedback

Feedback for the course assignments will be delivered using the comments feature in Canvas. Any details that require further discussion will take place over email, in an online chat, or over the phone.

Academic Integrity

The below statement also goes for copying code. Students will have individual assignments and group assignments. Source code and exploit information must not be shared outside assigned groups. Information on vulnerability assessments is not to be shared. Students should make sure to read NJIT's academic integrity policy in full here.

Academic Integrity is the cornerstone of higher education and is central to the ideals of this course and the university. Cheating is strictly prohibited and devalues the degree to which students are working. As a member of the NJIT community, it is every student's responsibility to protect their educational investment by knowing and following the academic code of integrity policy, which is found here.

Please note that my professional obligation and responsibility is to report any academic misconduct to the Dean of Students Office. Any student found in violation of the code by cheating, plagiarizing, or using any online software inappropriately will result in disciplinary action. This may include a failing grade of F, and/or suspension or dismissal from the university.

There will be no warnings or second chances about cheating. It is every student's responsibility to understand specifically what constitutes academic dishonesty. Ignorance is not an excuse or a defense. It is also each student's responsibility to understand the rules for properly citing the work of others in the submission of classwork. Improper citation with a simple "copy/paste" from online sources may be grounds for failure of the assignment and/or the course. If students have any questions about the Code of Academic Integrity, they should contact the Dean of Students Office at dos@njit.edu.

Use of Artificial Intelligence Tools

While students are permitted to use artificial intelligence (AI) tools to support their learning and to help them better understand course concepts, the use of AI for completing course work is strictly prohibited. This includes, but is not limited to, using AI to write reports, generate code, or assist in taking quizzes or exams. The primary goal is for students to develop their own skills and demonstrate independent mastery of the material. Any misuse of AI in assessed assignments will be considered academic misconduct.

Student Support

Canvas Accessibility Statement

Please note: For an accessible version of the course PowerPoint slides, open the files with Microsoft Office 365.

Canvas Orientation

For students who are unfamiliar with Canvas, this is a link to self-paced student orientation.

Office of Accessibility Resources and Services

The OARS office goal is to enhance the educational experience for students with disabilities at NJIT. OARS is committed to promoting accessibility, inclusivity and awareness as a resource to all members of NJIT. If students need accommodation due to a disability, they should contact the Office of Accessibility Resources and Services at OARS@NJIT.EDU or visit Kupfrian Hall 201 to discuss their specific needs. A Letter of Accommodation Eligibility from the office authorizing student accommodations is required.