

New York University Tandon School of Engineering
Department of Computer Science & Engineering
CS-GY 9223 Special Topics: Introduction to Offensive Security
Fall 2026
Professor Mitch Haszard

Professor contact info: mitch.haszard@nyu.edu

Course Assistant contact info: TBD

Course Logistics

Weekly Lecture: Thursday @ 17:00 (5:00 PM) Eastern Time

Weekly Review Session: TBD @ TBD Zoom (TA)

Week Office Hours: Thursday @ 16:00 (4:00 PM) Eastern Time, 370 Jay Street, 10th Floor, Room 1013

Course Assistant

Girik Verma

Course Prerequisites

Introduction to Operating Systems (CS-UY 3224/CS-GY 6233)

Application Security (CS-UY 3923/CS-GY 9163)

Experience with C and Python

Course Description

This course teaches Offensive Security through Capture the Flag (CTF) challenges. It covers techniques and methods to reverse x86_64 assembly, exploitation strategies for compiled binaries, and common vulnerabilities in web applications and cryptographic implementations.

Course Objectives

Upon completion of this course, students should be able to:

- Assess the logic of compiled code via binary reverse engineering and solve programming challenges based on the functionality of the binary
- Employ memory corruption tactics and strategies to exploit compiled binaries
- Recognize web-based vulnerabilities and develop relevant attacks
- Assess cryptographic implementations to identify vulnerabilities and write related exploits to demonstrate understanding
- Apply course-related CTF challenges to real-world software

- Compete in future CTF competitions

Course Structure

Throughout the course, we run a CTF which is available any time at <https://class.osiris.cyber.nyu.edu>

Each week, a set of challenges related to that week's material will be released and marked as "hot," indicating they count towards that week's homework. In addition to solving the challenges by submitting the correct flag, students must submit a write-up for all solved challenges that includes relevant snippets of the solver script(s) (if applicable). *A write-up is more than a solver script; it must include an explanation of the solution methodology, supported by code snippets for brevity and clarity.*

Additionally, students are **required** to compete in at least one [CTFTime](#)-ranked CTF and provide a write-up about at least one non-trivial problem (~300 pts). You are encouraged to form teams with classmates and submit a group write-up. The maximum group size is four students. **Please send an email to the professor and course assistants with your team name and write-up within one week of completing, and no later than the final lecture.**

Learning content will be presented in various formats within each weekly module. Content will include introductory videos, interactive learning modules, and live in-person lectures with your professor. Slides will be included with each weekly Brightspace module. Weekly course assistant office hours cover the just-completed homework assignment (including solving it live, if time permits) and the content for the upcoming challenges.

The final will be similar to the weekly homework but more complex. It will be a single, very large challenge that requires multiple exploits to complete. The final will be considered optional and will replace one other week's homework if accomplished.

Additional extra credit opportunities (to replace up to one week's grade) may occur at the discretion of the professor. In past classes, this was typically limited to the final and one other opportunity, both of which were quite complex (read: it makes more sense not to need the extra credit and just complete each homework assignment as it comes).

Grade Calculation

90% of the course grade is based on weekly assignments, **which must be completed individually**. Weekly assignment grades are based on the number of challenges a student solves and writes up for the weekly CTF. **Full credit for that week is earned by solving 300 (or more) challenge points and submitting write-ups for those challenges.** For instance, a student who solves a 100-point and 200-point challenge and submits write-ups for both will receive full credit of 300/300 points for challenges and 300/300 points for the associated write-ups. Failing to submit write-ups for these challenges will result in the loss of the associate's write-up points, up to a 50% loss (300/300pts for challenges and 0/300pts for write-ups). See below for additional example scenarios if more clarity is required. Points will be tallied for hot challenges at the beginning of class, one week after it is assigned. Past challenges will remain available for the entire semester, and it is recommended to solve as many as possible, as weekly content typically builds on prior weeks.

10% of the course grade is based on the mandatory external CTF, **which can be completed with teammates**, as mentioned above.

Weekly Point Scenarios

Full Credit

Solving a 100pt and 200pt challenge, submitting write-ups for both

- 300/300pts for challenges and 300/300pts for write-ups
-

Solving two 200-point challenges and submitting write-ups for both

- 300/300pts for challenges and 300/300pts for write-ups
- There is no “extra credit” for surpassing the 300 points per week
- The maximum credit a student can receive is 300 points/week for the challenges and write-ups
-

Solving a 100-point challenge and two 200-point challenges, and submitting a write-up for the 100-point challenge and one 200-point challenge

- 300/300pts for challenges and 300/300pts for write-ups
- No deduction for not submitting the second 200-point write-up, as 300 points of challenges were written up and submitted

Partial Credit

Solving a 100-point and 200-point challenge, submitting a write-up for the 100-point challenge

- 300/300pts for challenges and 100/300pts for write-ups => 66% of total weekly grade

Solving two 200-point challenges and not submitting any write-ups

- 300/300pts for challenges and 0/300pts for write-ups => 50% of total weekly grade
- There is no “extra credit” for surpassing the 300 points per week in either category

Solving a 100-point challenge and submitting a write-up

- 100/300pts for challenges and 100/300pts for write-ups => 33% of total weekly grade

Solving a 100-point challenge and not submitting a write-up

- 100/300pts for challenges and 0/300pts for write-ups => 17% of total weekly grade

Weekly assignments are due at 17:00 (5:00 PM) Eastern Time, prior to lecture, two weeks after they are assigned. For example, Week 0 assignments, issued September 3, 2026, are due prior to the lecture on September 17, 2026.

The final grade will be calculated as follows:

- Homework and write-ups will be worth 90% of your final grade
- Live CTF participation & write-up will be worth 10% of your final grade; however, this is **required. You will not pass the course if you do not compete in a CTFTime CTF and submit a write-up.**
- Extra credit may be available and is currently under consideration. Extra credit opportunities will replace no more than a single week's assignment value (300 CTF challenge pts, 300 write-up points). Further details will be provided during the semester.

The mapping between final grade percentages and letter grades submitted to Albert is as follows:

- A: 93+
- A-: 90+
- B+: 86+
- B: 82+
- B-: 78+
- C+: 74+
- C: 70+

Course Requirements

You will need a reverse-engineering toolkit during the Reverse Engineering and Binary Exploitation units of the class. We recommend [Ghidra](#) or [Binary Ninja Free](#), both of which are free, cross-platform, and very effective. All other tools (IDA Free/Pro, Binary Ninja Personal, radare) are also allowed, and the professor will do their best to assist any students who might choose a different reverse-engineering tool.

While CTF is largely a team sport, vulnerability research and exploit development can be a largely individual task. Therefore, we believe that all team members should be able to solve problems independently. As a result, direct collaboration on homework problems is not permitted until the due date has passed.

Course Outline

Part I: Operating Systems Refresher and Introduction to Tooling

Class 0 – September 3, 2026: Operating Systems Refresher & Pwntools

- Binary memory layout
- ASLR & PIE
- Symbols
- Pwntools
- Week 0 CTF Challenges assigned

Week 1 – September 10, 2026: Reverse Engineering Tools

- Static tooling: cmdline, Ghidra, Binary Ninja
- Dynamic tooling: gdb
- New challenges assigned

Part II: Reverse Engineering (RE)

Week 2 - September 17, 2026: Reverse Engineering Basics

- What is a byte? What is a pointer?
- Decompiling a program
- Disassembling a program
- Control Flow
- Class 0 CTF challenges due; new challenges assigned

Week 3 – September 24, 2026: Structs and SAT solvers

- Recognizing structures
- Defining structures
- Z3 SAT solving
- Week 1 CTF challenges due; new challenges assigned

Week 4 – October 1, 2026: Protocols and Stripped Binaries

- Stripped binaries and strategies
- Protocols
- Week 2 challenges due; new challenges assigned

Part III: Binary Exploitation (Pwn)

Week 5 – October 8, 2026: Exploitation Primitives and Stack Buffer Overflows

- Read primitives

- Write primitives
- Stack buffer overflow practice
- Week 3 challenges due; new challenges assigned

Week 6 – October 15, 2026: Controlling Execution and Bypassing Modern Mitigations

- Writing `execve` shellcode
- Position-independent shellcode
- GOT & PLT
- GOT & RELRO
- Week 4 challenges due; new challenges assigned

Week 7 – October 22, 2026: Return-Oriented Programming (ROP)

- Visualizing ROP
- Finding ROP gadgets
- Building ROP chains
- Week 5 challenges due; new challenges assigned

Week 8 – October 29, 2026: Glibc Heap Exploitation <v2.32

- Allocating and freeing data
- Caches & linked lists
- UAFs & buffer overflows
- Tcache poisoning
- Malloc & free hooks
- Week 6 challenges due; new challenges assigned

Week 9 – November 5, 2026: Bypassing Safe-linking in Modern Glibc >=v2.32

- Safe-linking
- Bypassing safe-linking
- Goodbye `\_\_free\_hook`, hello `environ`
- Week 7 challenges due; new challenges assigned

Part IV: Web Hacking (Web)

Week 10 – November 12, 2026: Exploiting Common PHP Misconfigurations

- File inclusion
- Command injection
- File upload bypasses
- Week 8 challenges due; new challenges assigned

Week 11 – November 19, 2026: SQL Injection

- Writing queries
- Injecting arbitrary SQL commands
- Leaking data via Blind SQLi
- Week 9 challenges due; new challenges assigned

November 26, 2026 – Thanksgiving (No Class)

Week 12 – December 3, 2026: NOSQL Injection & XSS

- NOSQL attacks
- XSS (Cross-site scripting) basics
- Advanced XSS
- Week 10 and 11 challenges due; new challenges assigned

Part V: Cryptographic Vulnerabilities (Crypto)

Week 13 – December 10, 2026: Cryptographic Vulnerabilities

- Frequency Analysis
- Bad randomization
- Common RSA & AES attacks

December 17, 2026: Final Class – Recap & Next Steps / External CTF, Week 12/13 challenges, and Final due

Late Policy

Late homework will not be accepted under ANY circumstances; any homework not turned in on time will receive a grade of zero. Make-up examinations do not apply and will not be provided.

Moses Center Statement of Disability

If you are a student with a disability who is requesting accommodations, please contact New York University's Moses Center for Students with Disabilities (CSD) at 212-998-4980 or mosescsd@nyu.edu. You must be registered with CSD to receive accommodations. Information about the Moses Center can be found at www.nyu.edu/csd. The Moses Center is located at 726 Broadway on the 3rd floor.

NYU School of Engineering Policies and Procedures on Academic Misconduct – complete Student Code of Conduct [here](#)

- A. Introduction: The School of Engineering encourages academic excellence in an environment that promotes honesty, integrity, and fairness, and students at the School of Engineering are expected to exhibit those qualities in their academic work. It is through the process of submitting their own work and receiving honest feedback on that work that students may progress academically. Any act of academic dishonesty is seen as an attack upon the School and will not be tolerated. Furthermore, those who breach the School's rules on academic integrity will be sanctioned under this Policy. Students are responsible for familiarizing themselves with the School's Policy on Academic Misconduct.
- B. Definition: Academic dishonesty may include misrepresentation, deception, dishonesty, or any act of falsification committed by a student to influence a grade or other academic evaluation. Academic dishonesty also includes intentionally damaging the academic work of others or assisting other students in acts of dishonesty. Common examples of academically dishonest behavior include, but are not limited to, the following:
 - 1. Cheating: intentionally using or attempting to use unauthorized notes, books, electronic media, or electronic communications in an exam; talking with fellow students or looking at another person's work during an exam; submitting work prepared in advance for an in-class examination; having someone take an exam for you or taking an exam for someone else; violating other rules governing the

- administration of examinations.
2. Fabrication: including but not limited to, falsifying experimental data and/or citations.
 3. Plagiarism: intentionally or knowingly representing the words or ideas of another as one's own in any academic exercise; failure to attribute direct quotations, paraphrases, or borrowed facts or information.
 4. Unauthorized collaboration: working together on work meant to be done individually.
 5. Duplicating work: presenting for grading the same work for more than one project or in more than one class, unless express and prior permission has been received from the course instructor(s) or research adviser involved.
 6. Forgery: altering any academic document, including, but not limited to, academic records, admissions materials, or medical excuses.

NYU Tandon School of Engineering Policies and Procedures on Excused Absences (complete policy [here](#))

- A. Introduction: An absence can be excused if you have missed no more than **10 days of school**. If an illness or special circumstance has caused you to miss more than two weeks of school, please refer to the section labeled Medical Leave of Absence.
- B. Students may request special accommodations for an absence to be excused in the following cases:
 1. Medical reasons
 2. Death in immediate family
 3. Personal qualified emergencies (documentation must be provided)
 4. Religious Expression or Practice
- C. If illness or an accident causes you to miss a class (or classes) or an exam, you should do the following:
 1. Notify the Office of Student Advocacy by [filling out this Google form in its entirety](#).
 2. Students should not provide anyone except the Office of

Student Advocacy with a copy of medical documentation. If a professor requests a copy, refer them to the Office of Student Advocacy. It is important for instructors to know when you are experiencing an issue that might interfere with your studies. However, it is also important that your personal matters be kept confidential.

NYU Academic Calendar (available [here](#))

Please pay attention to notable dates such as Add/Drop, Withdrawal, etc. For confirmation of dates or further information, please contact Susana Garcia Henriquez, Manager of Academics for the Department of Computer Science & Engineering: sgarcia@nyu.edu

AI Policy

Students are free to use LLMs or similar tools to assist with writing code snippets and understanding homework problems. That said, these tools are good for routine questions and answers but struggle to reason about nuance and novel research. This class deals almost exclusively in nuance and granular reasoning, and thus, the benefit of LLMs wanes quickly. In fact, they frequently make errors that are impossible to detect without an already intimate understanding of the prompt's context and expected answer. You may find these tools useful for high-level reasoning about systems, but trusting their output for harder challenges may lead to costly errors or incorrect proposed solutions, resulting in more effort overall. It is important that the written work required for the course is your own.

The Office of Student Advocacy

At NYU Tandon, we recognize that medical, mental health, or personal issues can sometimes affect your attendance and academic performance. If you are experiencing challenges that you feel are impacting your ability to attend class or succeed in your coursework, we encourage you to reach out to the NYU Tandon Office of Student Advocacy. The Student Advocacy team is here to connect you with the resources and support you need to address these challenges and ensure you can continue to thrive academically. If you begin experiencing any issues that may affect your attendance or academic performance, please reach out to the NYU Tandon Office of Student Advocacy as soon as possible as we will be unable to appropriately address these concerns as the semester progresses, so early communication is essential to ensure the support and resources you need.

The NYU Tandon Office of Student Advocacy can be reached at advocacy.tandonstudentlife@nyu.edu and is located in 5 MetroTech Center LC240.

Inclusion

NYU is committed to creating an inclusive and equitable environment for all students. In this class, I aim to cultivate a strong sense of community, emphasizing that it is a space where individuals of diverse backgrounds, spanning beliefs, ethnicities, national origins, gender identities, sexual orientations, religious and political affiliations, and abilities, are treated with respect. All members of the class are expected to contribute to an inclusive atmosphere for each other. While disagreement and differing ideas are welcome, the expectation is that we respect perspectives beyond our own, even when they diverge from our personal beliefs or experiences. If you encounter disrespect or discrimination in the class or feel that the outlined standards are not being upheld, please report your experiences to me.