Syllabus
Modern Binary Exploitation

Course Information
Course Title: Modern Binary Exploitation
Course Number: CSCI 4967/ITWS 4967
Credit Hours: 4
Semester / Year: Fall 2021
Lectures: TF 12pm-1:50pm in SAGE 3713
Website: www.cs.rpi.edu/~milanova/csci4967
Prerequisite: CSCI 2500 Computer Organization
Instructors: Ana Milanova (Lectures adapted from RPISEC notes.)

Course Description
Cybersecurity is one of the fastest growing fields in computer science. Modern Binary Exploitation will focus on teaching practical offensive security skills in binary exploitation and reverse engineering. Through a combination of interactive lectures, hands-on labs, and guest speakers from industry, the course will offer students a rare opportunity to explore involved subjects in a rapidly evolving field.

The course will start off by covering basic x86 reverse engineering, vulnerability analysis, and classical forms of Linux-based userland binary exploitation. It will then transition into defenses found on modern systems (Cookies, DEP, ASLR, etc) and the techniques used to defeat them.

Course Material
Mandatory Resources:
- WarGames platform. Register through this link (Ret2 charges $59 this term)
  - There is a trial period. If you email support or let me know within 2-3 weeks (e.g., if you drop the class) your account will be discontinued and you will be reimbursed.
Suggested Textbooks:
- Hacking: The Art of Exploitation, 2nd Edition by Jon Erickson
  - ISBN 978-1593271442
  - ISBN 978-0470080238
- Practical Malware Analysis by Michael Sikorski and Andrew Honig
  - ISBN: 1593272901

Student Learning Outcomes
Upon successful completion of this course, students will:
1. Possess the skills necessary to carry out vulnerability analysis of binary applications.
2. Understand executable formats, program control flow at the assembly level, and other low level concepts.
3. Understand classic and contemporary vulnerabilities and exploitation techniques.
4. Apply both source code auditing and binary reverse engineering to the vulnerability discovery process.
5. Be capable of exploiting vulnerabilities found in real world software as defined by MITRE’s
Critical Vulnerabilities and Exposures (CVE) system.

Course Assessment Measures

- **Labs:** Typically, each lecture will be accompanied by a lab that will solidify understanding of the topic of the lecture. Each lab will contain at least 3 problems, a C-level, a B-level, and an A-level problem.

- **Projects:** There will be two projects over the course of the semester. In these projects students will be exploiting known vulnerabilities in a realistic piece of software. The projects resemble lab problems, however, they are lengthier and combine multiple topics.

Grading Criteria

- **Labs:** 60%, 9 chapters and labs equally weighted at 6.67% of the final grade
  - Labs will typically consist of 3 problems, a C-, B-, and A-level problems with each problem harder than the last. Lab grading works as follows.
    - Students who complete only the C problem will receive 75% (C)
    - Students who complete both the C and B problems will receive 85% (B)
    - Students who complete all three lab problems will receive 100% (A)
  - If a student does not complete the C-level problem before the end of the associated lab period, they will receive 5% reduction. For example, if a student fails to complete the C-level problem during the lab period, but then goes on to successfully complete the C- and B-level problems for the week, the grade received for that week’s lab will be 80%.
  - Problems have several checkpoints and credit will be awarded per checkpoint, however, checkpoints may be unequally weighted. For example, if a B-level problem has 3 checkpoints where points 1 and 2 are worth 25% each and a student completes only 1 and 2 then they will receive a grade of 80%. Note that one cannot access a B-level problem in WarGames before completing the C-level problem first. Similarly, one cannot access an A-level problem before completing the corresponding B-level one.
  - Lab problems will be introduced by the end of lecture typically once a week. The C-level problem will be due in person by the end of the associated lab period, typically Friday.
    - All other problems become homework, and are due at the start of class on Tuesday the following week. (See calendar.)
    - Lab problems submitted late will receive a letter grade reduction (-10%). The calendar specifies firm deadlines for late lab submission. Late labs won’t be accepted past those deadlines.

- **Term Projects:** 40%, two projects equally weighted at 20%
  - Project specific grading breakdowns will be given when they are assigned.
  - Project checkpoints exist to keep you on pace with projects.
  - Projects submitted late will receive a -10% reduction. Projects may be submitted up to 5 days late.

- **Letter Grade Assignment (subject to change):**
  - [92 – 100]: A
  - [90 – 92): A-
  - [88 – 90): B+
  - [82 – 88): B
There will be servers dedicated to hosting the problems to be completed by students for both the lab and projects.

Grades and course progress will be made available to students throughout the semester. Grades will be made available via Submitty’s Rainbow grades.

**Attendance Policy**

Although lecture and lab attendance is not mandatory, it will be **exceptionally** difficult to keep up without attending. Problems are challenging and attending lecture and lab is your best chance to have questions answered, clear up confusion, get hints and get debugging help. **Note that the C-level (Level 1) problem must be turned in by the end of the lab period. See Grading Criteria for more details.**

**Academic Integrity**

Student-teacher relationships are built on trust. For example, students must trust that teachers have made appropriate decisions about the structure and content of the courses they teach, and teachers must trust that the assignments that students turn in are their own. Acts that violate this trust undermine the educational process. The Rensselaer Handbook of Student Rights and Responsibilities defines various forms of Academic Dishonesty and you should make yourself familiar with these.

As always, discussion is allowed and even encouraged, however taking written notes out of discussion is not allowed. All code that is turned in for a grade must represent the student’s own work.

Submission of any assignment that is in violation of this policy will result in a penalty of a **zero for the assignment** for all parties involved. **Repeated offenses will result in a failing grade for the course.**

If you have questions concerning this policy before submitting an assignment, please ask for clarification.

**Other Course-Specific Information**

The labs and interactive exercises used in lecture will be hosted in a variety of different series on an internal ‘Wargame’ server meant purely for teaching security concepts in a safe and educational environment. Students enrolled in the course will have SSH access to the server and are expected to use it as an aide to the learning process. Attacking parts of the server infrastructure beyond the scope of what is assigned or attempts to disrupt services for the course or other students is **not allowed** and will be considered a violation of academic integrity.

Due to the experimental nature of the course and assignments being offered, the schedule may change.

Note: This syllabus was modified by Prof. Milanova from previous editions of Modern Binary Exploitation.