Syllabus - ECE 48xx – Introduction to Malware Reverse Engineering

Course Summary:
Malware reverse engineering involves deep analysis of the code, structure, and functionality of malicious software. The goal of this course is to provide a solid foundation in reverse engineering, which is crucial in understanding modern malware and crafting solutions for the remediation and prevention of cyber attacks. The course exposes students to an immersive, hands-on experience in malware analysis and examines a wide range of software security topics relating to operating systems, debugging, and software protection.

Grading:

Labs	50%
Midterm	20%
Final Exam	30%
Total	100%

Grading Scale:

90% - 100%	A
80% - 89%	B
70% - 79%	C
60% - 69%	D
< 60%	F

Topics Outline:

1. Intro to the course
   - 1.1. Malware, Assembly Language, Reverse Engineering
   - 1.2. Under what circumstances is reverse engineering useful or breaking contracts?
   - 1.3. Why is reverse engineering necessary?
     - 1.3.1. Interoperability / Competition
     - 1.3.2. Auditing
     - 1.3.3. DRM
     - 1.3.4. Analysis of Malware
   - 1.4. Guest Speaker(s): Career Opportunities in Malware Reverse Engineering
2. Background on Malware
   - 2.1. Current and Next-Generation Malicious Software
     - 2.1.1. Viruses
     - 2.1.2. Worms
     - 2.1.3. Trojans
     - 2.1.4. Botnets
     - 2.1.5. Polymorphic and Metamorphic Malware
     - 2.1.6. Advanced Persistent Threats
   - 2.2. Intro to Defensive Strategies Against Malware
     - 2.2.1. Worm Fingerprinting / Signature Generation
     - 2.2.2. Behavioral Approaches to Detection of Malware
     - 2.2.3. Hardware Agents for System Integrity Checking

3. Low level Software
    3.1. Overview of Intel Assembly Language
    3.2. Virtual Machines for Interpreted High-Level Languages
    3.3. Representation of Compiled High Level Language Structures in Assembly
    3.4. Operating Systems Background
        3.4.1. MS-DOS Internals Related to Malware Case Studies
        3.4.2. Modern Windows Execution Environment
    3.5. Executable File Formats
        3.5.1. PE Files
            3.5.1.1. Import Address Table
4. Static Analysis of Software
    4.1. System Monitoring Tools
    4.2. Dynamic Tracing: System Calls, Filesystem, and Registry
    4.3. Compiler Issues
    4.4. Debuggers
        4.4.1. OllyDbg
        4.4.2. WinDbg
    4.5. Disassemblers
        4.5.1. IDA Pro
        4.5.2. Decompilers
    4.6. Memory Analysis to Support Reverse Engineering
        4.6.1. DRAM Acquisition
        4.6.2. Extraction of Malware
5. Advanced Reverse Engineering Techniques
    5.1. Encrypted/Packed Executables
        5.1.1. Unpacking Case Study
    5.2. Anti-Debugging Techniques
    5.3. Anti-VM Techniques
    5.4. Code Obfuscation
6. Remediation of Advanced Persistent Threats
    6.1. Determination of Malicious Behaviors
    6.2. Analysis of Decompiled Source Code
    6.3. Revelation of Command & Control Functionalities

The laboratory assignments will be on the following topics:

1. Software Disassembly
2. Identification of High Level Language Structures in Assembly
3. Malware Case Study (1)
4. Malware Case Study (2)
5. Malware Case Study (3)
6. Encrypted/Packed Malware
7. Memory Analysis of Ransomware Infection (Extra Credit: Recover Locked Files)
8. Anti-Debugging/Polymorphic Techniques
9. Malware in Embedded Devices
10. Advanced Persistent Threats (1)
11. Advanced Persistent Threats (2)