CS 501
Introduction to Malware, Threat Hunting and Offensive Capabilities Development

Course Developers
Kai Bernardini
Winnona DeSombre

Instructor
Kai Bernardini
Email: (See Piazza)

Course Description
The class introduces students to the wild world of offensive capabilities development and cyber threat hunting by taking on the role of both attacker and defender to better understand various stages of cyber attacks. Focusing on the Windows operating system, students will create and analyze malware in the context of combating a simulated threat actor APT-ChonkyBear in addition to creating their own tools to emulate their capabilities.

PreReqs:
tools:
- GDB/experience debugging native binaries
- Experience with compiler toolchains (compilers, linkers)

Programing/scripting:
Python3: you need to be **very** proficient with python.
c/c++/some other systems programming language: you should be able to write simple C programs, know the difference between Stack and Heap memory, and how to manage memory
intel x86 assembly: you should be able to read x86 assembly. Being able to write it is a plus!

Protocols:
- IP : you should know the difference between a private and public IP, the basics of addressing, and know the relationship between a domain name, DNS and an IP address
- TCP: More so the basics of socket programming, less so the internals of how reliability is achieved or what the handshake process looks like.
- DNS: You don't need to understand **how** DNS works, but you need to know what it does.
- TLS: You need to understand what a Certificate Authority is, a TLS cert is, and the basics of establishing secure tunnels
- HTTP(s): HTTP verbs, request structure, and error codes

Courses/topics:
CS-131/Combinatorics.

CS-237/Probability

CS-210/Computing systems : familiarity with memory management, c programming, reading assembly, using debuggers and compiler toolchains. **You should know the difference between stack and heap memory.**

CS-357/Basic cryptography: familiarity with Ciphers, MACs and Hash functions. Ideally also the idea behind asymmetric cryptography.
CS460/ TCP/IP, TLS, UDP HTTP(s), DNS

Grading:
Subject to change:
Writeup: 10%
Exercises: 10%
Capture the Flag (CTF): 10%
Reverse Engineering Assignments: 20%
Coding Assignments: 20%
Capstone Project: 30*-50% (if you do really well on the capstone project, I will weight it more)
Bonus Points:
Class Participation: 2.5%

Assignments
Assignments in this class are divided into one of three primary categories:

Exercises
Exercises are weekly, quick turn around homework, and in class assignments .They are designed to be sanity checks for topics as the course progresses.They are questions that should take between 5 and 10 minutes. Questions will always be based on recent material, but topics build on each other. **You should bring your laptop to class each Tuesday.**

Malware Reverse Engineering
Each assignment consists of a new epoch of malware sent to students highlighting a new technique to analyze. It will be their job to reverse engineer the new implant, detail its functionality, and identify attacker infrastructure.

Coding/Malware Development
On the offensive side, students will work to recreate capabilities observed within new epochs of malware that can be integrated into their own Command and Control (C2) framework. For more on this, see Capstone

Capture the Flag (CTF)
The course will have a semester-long capture the flag competition where students will compete to solve challenges.

Capstone

This class will guide students through the creation of their own production ready Remote Administration Tool (RAT) and Command and Control (C2) team server. Each component of the capstone will be introduced as an offensive homework assignment, and will be integrated into the framework over the course of the semester. Ambitious students are encouraged to get involved with Porchetta Industries to release their tool as sponsorware.

References:

There are no mandatory textbooks for this course. Required readings will consist of blog posts, or source code. Below are several references that will be relevant to various homework assignments

- Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software 1st Edition
- Practical Reverse Engineering: x86, x64, ARM, Windows Kernel, Reversing Tools, and Obfuscation 1st Edition
- Operator Handbook: Red Team + OSINT + Blue Team Reference
- Rtfm: Red Team Field Manual 1.0 Edition
- Blue Team Field Manual (BTFM)
- http://www.harmj0y.net/blog/empyre/building-an-empyre-with-python/
- https://vx-underground.org
- https://github.com/yeyintminthuhtut/Awesome-Red-Teaming