Offensive Network Security

CIS 5930, Spring 2017
Department of Computer Science, Florida State University

Class time and location

Tuesday and Thursday, 9:30am – 10:45am, Room 016, Love Building

Instructor

- Instructor: Xiuwen Liu (pronounced as Shu-wen Lea-l)
- Email: liux@cs.fsu.edu (most effective way to contact me)
- Home page: http://www.cs.fsu.edu/~liux
- Office: 166 Love Building (LOV); Phone: (850) 644-0050
- Monday and Wednesday from 1:30PM – 3:00PM and by appointments.

Class Home Page

http://www.cs.fsu.edu/~liux/courses/offensivenetsec/index.html.
This web site contains the up-to-date information related to this class such as news, announcements, assignments, lecture notes, and useful links to resources that are helpful to this class. Besides the web pages, Blackboard will be used to communicate changes and updates and post grades for this class; in particular, I will send emails using email addresses in the Blackboard system and please make sure that your email address on record is current.

Rationale

With affordable desktop and laptop computers, large storage devices (e.g., hard drives), hardware, wide availability of the high speed internet connections, and more recently Internet-capable 3G and 4G smartphone and similar devices, the earth becomes highly connected that almost everyone can reach any other one on the planet as long as they are connected to the Internet. The unprecedented connectivity has unleaded unique potentials of computer technology (e.g., huge storage spaces and fast computing), leading to new services that were not imagined ten years ago. Not only our daily life activities heavily rely on the Internet, and government and the critical infrastructures we take for granted rely on the intended behaviors of computers and the underlying network. Unfortunately, the high connectivity has also created new problems, from spyware to steal data, computer viruses and worms to destroy data, to network-enabled weapons, to cyber wars that can disable companies and even countries (such as Stuxnet). All these problems are related to computer security. Due to its paramount importance, computer security is not just one academic research area. Many security products are installed on typical computers; in the United States, there are multiple federal agencies dedicated to computer security; the computer security is a multi-billion industry that is estimated to grow steadily (just click https://www.google.com/#q=computer+security+industry+growth and see). Computer security related issues have been widely recognized in software development companies. As computer security techniques evolve continuously along with product improvements and new service opportunities, computer security is and will remain to be an important and valuable area in the perceivable future with new career opportunities; in recent years, computer security has enjoyed a zero and low unemployment rate (see http://www.techjournal.org/2011/07/information-security-analysts-unemployment-rate-zero/ and

http://www.bls.gov/oes/current/oes151122.htm). Due to the complexity of networked systems at typical organizations, securing such systems is much more than installing antivirus products, vulnerability scanning, and firewall and network parameter configurations; zero-day vulnerabilities and their exploitations render all existing antivirus products ineffective and active development of malicious cyber activities posts new threats constantly to the Internet and cyber space. To better secure systems against such attacks, offensive computer security, also called penetration testing, has evolved to be an effective way to evaluate and enhance the security of computer systems. Developing effective penetration testing is an exciting and challenging endeavor and can have a significant impact on the operations of companies by avoiding potentially costs due to cyber incidences. As organizations and their employees rely heavily on networking to access information, networking can be used by malicious users to gain unauthorized access and then perform unauthorized actions. Due to the open nature of the Internet protocols, such vulnerabilities are inherent and proper prevention mechanisms must be in place to be effective.

Course Description

This course provides introductory but comprehensive coverage of fundamental problems, principles, and techniques in offensive network security, including basic networking techniques, networking sniff techniques, denial of service techniques, TCP/IP hijacking, port scanning, remote shellcode development (including port-binding shellcode, and connect-back shellcode), network packet manipulation techniques (sending, dissecting, and forging network packets), network protocol analysis (including botnet protocol analysis), and then commonly used tools for offensive network security with an emphasis on their principles and fundamental techniques. Additionally, as offensive network security relies on many interdependent components, this course will also cover real world policy (legal) and implementation issues in penetration testing. It also involves opportunities to develop fully working exploits by implementing and demonstrating such exploitations on virtual machines and tools to assist to analyze network packets, protocols, and vulnerabilities and develop exploits.

Prerequisites

CDA 3100 – Computer Organization I; have a good understanding of operating systems and computer networking models; have a good understanding of basic data types, data structures, function calls, and memory layout of programs; be able to read and understand C programs; be able to understand x86 assembly and write short x86 assembly programs; have a general understanding of computer security; have hands-on working knowledge of common vulnerabities and exploits (such as buffer overflow and string format vulnerabilities) and various shellcode development.

Course Objectives

Upon successful completion of this course of study, the student will:

- Know how to identify common buffer overflow vulnerabilities in networking applications
- Know how to exploit buffer overflow vulnerabilities in networking applications
- Know how to identify common format string vulnerabilities in networking applications
- Know how to exploit common format string vulnerabilities in networking applications
- Know how to write network sniffers on unswitched networks at layer 2
- Know how to construct network packets to perform ARP (Address Resolution Protocol) cache poisoning
- Know how to write network sniffers on switched networks at layer 2
- Know how to implement SYN flooding to perform a DoS (Denial of Service) attack
- Know how to implement the ping of death to perform a DoS attack

- Know how to implement ping flooding to perform a DoS attack
- Know how to implement amplification techniques of DoS attacks
- Know how to implement distributed DoS attacks
- Know how to perform standard port scanning
- Know how to perform stealth SYN port scanning
- Know how to perform FIN port scanning
- Know how to perform X-mas port scanning
- Know how to perform Null port scanning
- Know how to use decoy IP addresses to avoid detection
- Know how to perform idle port scanning
- Know how to implement proactive defense against port scanning
- Know how to implement backdoors in Linux and Windows machines
- Know how to implement packet analyzer using Scapy
- Know how to implement packet manipulator using Scapy
- Know how to implement port scanning techniques in MetaSploit
- Know how to implement application vulnerability analysis in MetaSploit
- Know how to effectively report and communicate the flaws

Textbook and Course Materials

Required textbook: "Hacking: The Art of Exploitation, 2nd Edition" by Jon Erickson (published by No Starch Press, 2008)**;** mainly chapters 0x400 and 0x600 from the book.

Recommended textbooks: "**Metasploit: The Penetration Tester's Guide 1st Edition**" by David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni (published by No Starch Press; 1st edition, July 25, 2011); "**The IDA PRO Book: The Unofficial Guide to the World's Most Popular Disassembler, 2nd Edition**" by Chris Eagle (published by No Starch Press, 2011); "**Practical Malware Analysis: The Hands-on Guide to Dissecting Malicious Software**" by Michael Sikorski and Andrew Honig (published by No Starch Press, 2012); "**Black Hat Python: Python Programming for Hackers and Pentesters, 1st Edition**" by Justin Seitz (published by No Starch Press, 2014).

In addition to the textbooks, papers, documents, and notes from the literature will be distributed along the lectures.

Student Responsibilities

Attendance is required for this class. Unless you obtain prior consent of the instructor, missing classes will be used as bases for attendance grading. In case that it is necessary to skip a class, students are responsible to make up missed materials. Participation in in-class discussions and activities is also required. All submitted assignments and projects must be done by the author(s). It is a violation of the Academic Honor Code to submit other's work and the instructor of this course takes the violations very seriously.

As this course will cover certain techniques to break down known systems in order to demonstrate their vulnerabilities, it is **illegal**, however, to practice these techniques on others' systems. The students will be liable for their behaviors and therefore consequences.

Assignments and Projects

About six homework assignments will be given along the lectures and they need to be done individually and turned in. There will be hands-on projects, where the full exploitation cycles must

be implemented and demonstrated in virtual machine environments. There will be a midterm exam and a term project.

Grading Policy

Grades will be determined as follows:

Assignment	Points	Assignment	Points
Class Attendance & Participation	10 %	Midterm Exam	25 %
Homework Assignments	30 %	Term Project	15 %
Hands-on projects	20 %		

Grading will be based on the weighted average as specified above and the following scale will be used (for graduate students, S is the weighted average on a 100-point scale; for undergraduate students, S will be the sum of ten and the weighted average on a 100-point scale):

Score	Grade	Score	Grade	Score	Grade
$93 \le S$	A	$80 \le S < 83$	B-	$67 \le S < 70$	D+
$90 \le S < 93$	A-	$77 \le S < 80$	C+	$63 \le S < 67$	D
$87 \le S < 90$	B+	$73 \le S < 77$	C	$60 \le S < 63$	D-
$83 \le S < 87$	B	$70 \le S < 73$	C-	$S < 60$	F

Late Penalties

Assignments are due at the beginning of the class on the due date. Assignments turned in late, but before the beginning of the next scheduled class will be penalized by 10 %. Assignments that are more than one class period late will **NOT** be accepted.

Submission and Return Policy

All tests/assignments/projects/homework will be returned as soon as possible after grading but no later than two weeks from the due date.

Tentative Schedule

Here **Art** refers to the "**Hacking: The Art of Exploitation**".

- Week 1: Introduction (**Art**: Chapter 0x400 and other sources)
  - General introduction to offensive networking security, penetration testing, and hacking
  - Steps in penetration testing and offensive network security (see http://www.pentest-standard.org/index.php/Main_Page)
  - Fundamental principles and techniques to network exploitation development
- Weeks 2-3: OSI model, network applications, and common vulnerability exploits (**Art**: 0x410-0x420, 0x620-0x660)
- Weeks 4-5: Networking sniffing (**Art**: 0x430-0x440)
- Weeks 6-7: Port scanning (**Art**: 0x470)
- Week 8: Denial of service attacks (**Art**: 0x450)

- Week 9: TCP/IP hijacking (**Art**: 0x460)
- Week 10: Backdoor development (**Art**: 0x550 and papers)
- Week 11: MetaSploit (On-line documents and papers)
- Week 12: Networking exploitation on Windows (including NDIS (Network Driver Interface Specification from Microsoft) driver development)
- Week 13: Midterm exam (**Thursday, April 6, 2017**)
- Week 14: Network protocol vulnerability analysis and exploitation
- Week 15: Vulnerability analysis and exploitation of software defined networks
- Week 16: Case studies of offensive network exploitation tools (from the literature and other sources)
- Final exam week: Term project due at **5:00pm, May 5, 2017**

Academic Honor Code

The Florida State University Academic Honor Policy outlines the University's expectations for the integrity of students' academic work, the procedures for resolving alleged violations of those expectations, and the rights and responsibilities of students and faculty members throughout the process. Students are responsible for reading the Academic Honor Policy and for living up to their pledge to ". . . be honest and truthful and . . . [to] strive for personal and institutional integrity at Florida State University." (Florida State University Academic Honor Policy, found at http://dof.fsu.edu/honorpolicy.htm)

Assignments/projects/exams are to be done individually, unless specified otherwise. It is a violation of the Academic Honor Code to take credit for the work done by other people. It is also a violation to assist another person in violating the Code (See the FSU Student Handbook for penalties for violations of the Honor Code). The judgment for the violation of the Academic Honor Code will be done by the instructor and a third party member (another faculty member in the Computer Science Department not involved in this course). Once the judgment is made, the case is closed and no arguments from the involved parties will be heard. Examples of cheating behaviors include:

- Discuss the solution for a homework question.
- Copy programs for programming assignments.
- Use and submit existing programs/reports on the world wide web as written assignments.
- Submit programs/reports/assignments done by a third party, including hired and contracted.
- Plagiarize sentences/paragraphs from others without giving the appropriate references. Plagiarism is a serious intellectual crime and the consequences can be very substantial.

Penalty for violating the Academic Honor Code: A 0 grade for the particular assignment /exam and a reduction of one letter grade in the final grade for all parties involved for each occurrence. A report will be sent to the department chairman for further administrative actions.

Accommodation for Disabilities

Students with disabilities needing academic accommodations should: 1) register with and provide documentation to the Student Disability Resource Center (SDRC), and 2) bring a letter to the instructor indicating the need for accommodation and what type. This should be done within the first

week of class. *This syllabus and other class materials are available in alternative format upon request.*

For more information about services available to FSU students with disabilities, contact the Assistant Dean of Students:

Student Disability Resource Center
97 Woodward Avenue, South
108 Student Services Building
Florida State University
Tallahassee, FL 32306-4167
(850) 644-9566 (voice)
(850) 644-8504 (TDD)
sdrc@admin.fsu.edu
http://www.disabilitycenter.fsu.edu/
