Corelan Advanced Exploit Development

Abstract

The Corelan "**ADVANCED**" exploit development class is a fast-paced, mind-bending, hands-on course where you will learn advanced exploit development techniques from an experienced exploit developer. During this course, students will get the opportunity to learn how to write exploits that bypass modern memory protections for the Win32 platform, using Windows 7 and Windows 10 as the example platform, but using techniques that can be applied to other operating systems and applications. We will discuss differences between Windows 7 and Windows 10 and explore previously undocumented techniques to achieve important exploitation primitives in Windows 10. The trainer will share his "notes from the field" and various tips & tricks to become more effective at writing exploits.

This is most certainly not an entry level course. In fact, this is a one of the finest and most advanced courses you will find on Win32 exploit development.

This hardcore, practical, hands-on course will provide students with solid understanding of x86 Windows heap exploitation. We make sure the course material is kept updated with current evolutions, includes previously undocumented tricks and techniques, and details about research we performed ourselves, so you can apply the research techniques on other applications and operating system versions. Combined with the way the course is built up, this will turn this class into a truly unique learning experience.

During all of our courses, we don't just focus on techniques and mechanics, but we also want to make sure you understand why a given technique is used, why something works and why something doesn't work. In the advanced course, we also provide you with insights on how to do your own research related with heap exploitation in general (not just Windows 7 or Windows 10)

The new 2019 edition of the course is based on Windows 7 and Windows 10. (As the Windows 10 Heap Manager contains additional mitigations, we use Windows 7 first to teach the basics, and then use Windows 10 later on)

We believe those are just a few arguments that makes this training stand out between other exploit development training offerings. Feel free to check the testimonials page if you want to see real, voluntary, unmodified and uncensored reactions by some of our students.

Finally, we offer you post-training support as well. If you have taken the course and you still have questions afterwards, we will help.

Learning Objectives

- Familiarize yourself with the basics of exploit development. Learn to write exploits for saved return pointer overwrites and abuse SEH records with your eyes closed. Understanding how heap spraying works, and why it works. If exploitation still a black box for you, this is the next step.
- If you've taken the Bootcamp or other commercial courses on exploit development this is the next phase.
- Learn modern techniques to exploit applications on Windows 7/10.
- Learn the fine art of writing browser exploits.
- Learn the skills to investigate heap managers on modern Windows versions (Win10) and find your own exploitation primitives.
- Learn what (generic) questions to ask (rather than being spoonfed exploit-specific solutions & answers).
- Learn to write ROP chains blindfolded. (It is fundamentally important that you have practical experience with constructing/writing your own ROP chain!)
- Be ready to suffer and bleed, absorb new knowledge fast and not intimidated by debuggers and assembly instructions…

Target Audience

Pentesters, auditors, network/system administrators, reverse engineers, malware analysts, developers, members of a security department, security enthusiasts, or anyone that has a solid and practical basic knowledge of exploit development for Windows already.

Course Outline

- **ASLR & DEP Refresher**
  - Bypassing ASLR
  - Bypassing DEP
- **WinDBG**
  - Introduction to WinDBG
- **Windows Heap Management**
  - Terminology & building blocks
  - Windows 7 Heap, Windows 10 Heap ("NT" and "Segment" heap)
  - Front-End-Allocator and Back-End-Allocator
  - Differences between Windows 7 and Windows 10
  - Heap manipulation primitives
- **Heap Spraying**
  - Basic mechanisms
  - Data & object spraying
  - Precise heap spraying
- **Heap Exploitation**
  - Use-After-Free
  - Linear & non-linear overflows / controlled write
  - Double Free
  - Type confusion
  - Use of uninitialized memory
  - Memory leaks / Information Disclosure
  - Heap Manipulations and heap primitives

Intro to x64 exploitation

- x64 processes, memory map, registers
- Functions & calling conventions
- Structured Exception Handling
- ASLR
- Stack Buffer Overflows

- Heap exploitation primitives on x64

What's next

- Overview of memory protection evolutions
- Thoughts & ideas on fuzzing and bug hunting

During the course, students will get the opportunity to work on real vulnerabilities in real applications, use a wide range of heap exploitation techniques and most importantly learn how to do your own research to find exploitation primitives in complex applications and new versions of Windows.

REQUIREMENTS

Student Machine/ Laptop Requirements

- A laptop (no netbook) with vmware workstation/virtualbox and enough processing power and RAM (we recommend 4Gb of RAM) to run up to 2 virtual machines at the same time. The use of a 64bit processor and a 64bit operating system on the laptop will make the exercises more realistic.
- 3 Virtual machines (Windows 10 (no patches), Windows 7 SP1 (no patches), Kali Linux (fully up-to-date))

Note : you will receive the exact installation instructions after registration, so <u>don't start installling the VMs yet</u>.

All required tools and applications will be provided during the training or will be downloaded from the internet during the training.

You must have full administrator access to all machines. You must be able to install and remove software, and you must be able to disable and/or remove firewall/antivirus/… when necessary.

Students Knowledge Pre-Requisites:

Students must:

- be able to read and write simple C/C++ code and simple scripts (python, javascript)
- truly master all basic concepts of exploit development, as listed in our "BOOTCAMP" course. If you have taken the Bootcamp course and done a lot of practice after taking the class, then you're probably ready for this class.
- be familiar with ROP (i.e. understand how it works on Windows, know how to build a ROP chain, know how to use mona.py to generate a chain and how to fix the chain if it doesn't work)

- be familiar with using debuggers (we'll use WinDBG for most part of the course, but we'll spend some time explaining the basics of using WinDBG. It is assumed that you have practical experience with Immunity Debugger and mona.py)
- be ready to dive into a debugger and read asm for hours and hours and hours
- be ready to think out of the box and have a strong desire to learn
- be fluent with managing Windows / Linux operating system and with using vmware workstation/virtualbox
- be familiar with using Metasploit to generate shellcode
- have basic practical knowledge of assembly