



EXP-301: Windows User Mode Exploit Development (OSED)

Course Overview

Windows User Mode Exploit Development (EXP-301) is a course that teaches learners the basics of modern exploit development. Despite being a fundamental course, it is at the 300 level because it relies on substantial knowledge of assembly and low level programming. It begins with basic buffer overflow attacks and builds into learning the skills needed to crack the critical security mitigations protecting enterprises. Learners who complete the course and pass the exam earn the OffSec Exploit Developer (OSED) certification. The OSED is one of three certifications making up the OSCE³ certification along with the OSEP for advanced penetration testing and OSWE for web application security.

Prerequisites

- Familiarity with debuggers (ImmunityDBG, OllyDBG)
- Familiarity with basic exploitation concepts on 32-bit
- Familiarity with writing Python 3 code
- Ability to read and understand C code at a basic level
- Ability to read and understand 32-bit Assembly code at a basic level

Target Audience

- Penetration Testers
- Exploit Developers
- Security Researchers
- Malware Analysts
- Software Developers Working On Security Products

Course Objectives

- Learn the fundamentals of reverse engineering
- Create custom exploits
- Develop the skills to bypass security mitigations
- Write handmade Windows shellcode
- Adapt older techniques to more modern versions of Windows

Duration

5 Days

Certifications

EXP-301

Contact Us

800.674.3550

2151 W. Hillsboro Blvd.
Suite 210
Deerfield Beach, FL 33442

Connect with us



Sign Up Today!





EXP-301: Windows User Mode Exploit Development (OSED)



Course Outline

Windows User Mode Exploit Development: General Course Information

- About the EXP301 Course
- Provided Materials
- Overall Strategies for Approaching the Course
- About the EXP301 VPN Labs
- About the OSED Exam
- Wrapping Up

WinDbg and x86 Architecture

- Introduction to x86 Architecture
- Introduction to Windows Debugger
- Accessing and Manipulating Memory from WinDbg
- Controlling the Program Execution in WinDbg
- Additional WinDbg Features
- Wrapping Up

Exploiting Stack Overflows

- Stack Overflows Introduction
- Installing the Sync Breeze Application
- Crashing the Sync Breeze Application
- Win32 Buffer Overflow Exploitation
- Wrapping Up

Exploiting SEH Overflows

- Installing the Sync Breeze Application
- Crashing Sync Breeze
- Analyzing the Crash in WinDbg
- Introduction to Structured Exception Handling
- Structured Exception Handler Overflows
- Wrapping Up

Introduction to IDA Pro

- IDA Pro 101
- Working with IDA Pro
- Wrapping Up

Overcoming Space Restrictions: Egghunters

- Crashing the Savant Web Server
- Analyzing the Crash in WinDbg
- Detecting Bad Characters
- Gaining Code Execution
- Finding Alternative Places to Store Large Buffers
- Finding our Buffer - The Egghunter Approach
- Improving the Egghunter Portability Using SEH
- Wrapping Up

Creating Custom Shellcode

- Calling Conventions on x86
- The System Call Problem
- Finding kernel32.dll
- Resolving Symbols
- NULLFree Position-Independent Shellcode PIC
- Reverse Shell
- Wrapping Up

Reverse Engineering for Bugs

- Installation and Enumeration
- Interacting with Tivoli Storage Manager
- Reverse Engineering the Protocol
- Digging Deeper to Find More Bugs
- Wrapping Up

Stack Overflows and DEP Bypass

- Data Execution Prevention
- Return Oriented Programming
- Gadget Selection
- Bypassing DEP
- Wrapping Up

Stack Overflows and ASLR Bypass

- ASLR Introduction
- Finding Hidden Gems
- Expanding our Exploit ASLR Bypass)
- Bypassing DEP with WriteProcessMemory
- Wrapping Up

Format String Specifier Attack Part I

- Format String Attacks
- Attacking IBM Tivoli FastBackServer
- Reading the Event Log
- Bypassing ASLR with Format Strings

Format String Specifier Attack Part II

- Write Primitive with Format Strings
- Overwriting EIP with Format Strings
- Locating Storage Space
- Getting Code Execution
- Wrapping Up

Trying Harder: The Labs

- Challenge 1
- Challenge 2
- Challenge 3
- Wrapping Up

Sign Up Today!

