EXP-312: Advanced macOS Control Bypasses (OSMR)

Course Overview

Advanced macOS Control Bypasses (EXP-312) is our first macOS security course. It's an offensive logical exploit development course for macOS, focusing on local privilege escalation and bypassing the operating system's defenses. EXP-312 is an advanced course that teaches the skills necessary to bypass security controls implemented by macOS, and exploit logic vulnerabilities to perform privilege escalation on macOS systems. Learners who complete the course and pass the exam earn the OffSec macOS Researcher (OSMR) certification.

Prerequisites

- C programming knowledge
- Normal user experience with macOS
- Basic familiarity with 64-bit assembly and debugging
- Understanding of basic exploitation concepts

Target Audience

- Penetration Testers
- Exploit Developers
- Security Researchers
- Macos Defenders
- Macos Application Developers

Course Objectives

- Obtain a strong understanding of macOS internals
- Learn the basics of Mach messaging
- Learn how to bypass Transparency, Content and Control (TCC) protections
- Learn how to escape the Sandbox
- Perform symbolic link attacks
- Leverage process injection techniques
- Exploit XPC for privilege escalation
- Perform hooking based attacks
- Write Shellcode for macOS
- Bypass kernel code-signing protection

Duration

5 Days

Certifications

OSMR

Contact Us

800.674.3550

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

Connect with us

APPLIED TECHNOLOGY ACADEMY

Sign Up Today!

EXP-312: Advanced macOS Control Bypasses (OSMR)

APPLIED TECHNOLOGY ACADEMY

Course Outline

macOS Control Bypasses: General Course Information

- About The EXP-312 Course
- Provided Materials
- Overall Strategies for Approaching the Course
- About the EXP-312 VPN Labs
- About the OSMR Exam
- Wrapping Up

Virtual Machine Setup Guide

- Creating VMs on Apple Silicon
- Installing Xcode
- Homebrew
- Old and Other Software
- Third Party Software
- General System Settings
- Specific VM Instructions

Introduction to macOS

- macOS System Overview
- High-Level OS Architecture
- The Mach-O File Format
- Objective-C Primer Wrapping Up

macOS Binary Analysis Tools

- Command Line Static Analysis Tools
- Static Analysis with Hopper
- Dynamic Analysis
- The LLDB Debugger
- Debugging with Hopper
- Tracing Applications with DTrace
- Wrapping Up

The Art of Crafting Shellcodes

- Writing Shellcode in ASM
- Custom Shell Command Execution in Assembly
- Making a Bind Shell in Assembly
- Writing Shellcode in C
- Wrapping Up

Dylib Injection Egghunters

- DYLD_INSERT_LIBRARIES Injection in macOS
- DYLIB Hijacking
- Wrapping Up

The Mach Microkernel

- Mach Inter Process Communication (IPC) Concepts
- Mach Special Ports
- Injection via Mach Task Ports
- BlockBlock Case Study - Injecting execv Shellcode
- Injecting a Dylib
- Wrapping Up

Function Hooking on macOS

- Function Interposing
- Objective-C Method Swizzling
- Wrapping Up

XPC Attacks

- About XPC
- The Low Level C API: XPC Services
- The Foundation Framework API
- Attacking XPC Services
- Apple's EvenBetterAuthorizationSample
- CVE-2019-20057 - Proxyman Change Proxy Privileged Action Vulnerability
- CVE-2020-0984 - Microsoft Auto Update Privilege Escalation Vulnerability
- CVE-2019-8805 - Apple EndpointSecurity Framework Local Privilege Escalation
- CVE-2020-9714 - Adobe Reader Update Local Privilege Escalation
- Wrapping Up

The macOS Sandbox

- Sandbox Internals
- The Sandbox Profile Language (SBPL)
- Sandbox Escapes
- Case Study: QuickLook Plugin SB Escape
- Case Study: Microsoft Word Sandbox Escape
- Wrapping Up

Bypassing Transparency, Consent, and Control (Privacy)

- TCC Internals
- CVE-2020-29621 - Full TCC Bypass via coreaudiod
- Bypass TCC via Spotlight Importer Plugins
- CVE-2020-24259 - Bypass TCC with Signal to Access Microphone
- Gain Full Disk Access via Terminal
- Wrapping Up

GateKeeper Internals

- File Quarantine
- XProtect
- GateKeeper
- Wrapping Up

Sign Up Today!

EXP-312: Advanced macOS Control Bypasses (OSMR)

APPLIED TECHNOLOGY ACADEMY

Course Outline

Bypassing GateKeeper

- CVE-2022-42821 GateKeeper Bypass Using AppleDouble Files
- CVE-2021-30990 GateKeeper Bypass using Symbolic Links
- Wrapping Up

Symlink and Hardlink Attacks

- The Filesystem Permission Model
- Finding Bugs
- CVE-2020-3855 - macOS DiagnosticMessages File Overwrite Vulnerability
- CVE-2020-3762 - Adobe Reader macOS Installer Local Privilege Escalation
- CVE-2019-8802 - macOS Manpages Local Privilege Escalation
- Wrapping Up

Getting Kernel Code Execution

- KEXT Loading Restrictions
- Sample KEXT
- The KEXT Loading Process
- CVE-2020-9939 - Unsigned KEXT Load Vulnerability
- CVE-2021-1779 - Unsigned KEXT Load Vulnerability
- Changes in Big Sur
- Wrapping Up

Injecting Code into Electron Applications

- Setting up an Electron Development Environment
- Creating a Simple Electron App
- The Application
- Environment Variable Injection
- Debug Port Injection
- Source Code Modification
- Protecting Electron Applications
- Wrapping Up

Mount(ain) of Bugs (Archived)

- The MAC Framework
- The mount System Call
- Disk Arbitration Service
- CVE-2021-1784 - TCC Bypass Via Mounting Over com.apple.TCC
- CVE-2021-30782 - TCC Bypass Via AppTranslocation Service
- 16.6. CVE-2021-26089 - Fortinet FortiClient Installer Local Privilege Escalation
- CVE-2021-26089 - Exploitation
- Wrapping Up

The Art of Crafting Shellcodes (Apple Silicon Edition)

- Writing Shellcode in ASM
- Executing Custom Shell Commands in Assembly
- Making a Bind Shell in Assembly
- Writing Shellcode in C
- Wrapping Up

Mach IPC Exploitation

- The Mach Interface Generator (MIG)
- CVE-2022-22639 Exploitation Case Study
- Wrapping Up

Chaining Exploits on macOS Ventura

- macOS Ventura Mitigations
- Exploit Chain on macOS Ventura
- Wrapping Up

macOS Penetration Testing

- Small Step For Man
- The Jail
- I am (g)root
- CVE-2020-26893 - I Like To Move It, Move It
- Private Documents - We Wants It, We Needs It
- The Core
- Wrapping Up

Sign Up Today!