EDR Internals: Research & Development

About TrainSec Academy

TrainSec is an online learning hub for current and aspiring cybersecurity pros who understand the need for excellence. Level up or change career using skill-expanding modular learning. We turn beginners into experts and experts into masters. TrainSec is designed for those who are serious about being at the vanguard of the latest knowledge, skills, and trends. We’re already working with the best.

Course Overview

This hands-on workshop is designed to give cybersecurity professionals, malware researchers, and detection engineers a rare opportunity to explore how modern Endpoint Detection and Response (EDR) solutions truly work - and how to both research and build them from the ground up.

Over the course, students will gain practical skills and a deep understanding of EDR internals, common detection methodologies, and real-world evasion techniques. Instructors Pavel Yosifovich and Uriel Kosayev will each bring their unique expertise - from low-level Windows internals and kernel development to advanced EDR evasion and reverse engineering.

The purpose of this training is not just to expose students to EDR theory but to empower them with the ability to think like an EDR developer and attacker. You’ll learn how static, dynamic, and heuristic engines operate, and then reverse engineer actual EDR components to analyze their logic and protection mechanisms. You’ll also learn how attackers craft evasive techniques to bypass such detection, and then build the components needed to detect or prevent these techniques yourself.

Whether you start by diving into how EDR drivers hook Syscalls or by exploring process injection and memory bombing, each section includes demos, guided exercises, and lab environments to reinforce the concepts in real-time. The course also provides an OVA-based research lab you can use to safely test EDR behavior and bypass strategies, even after the course ends.

Course Objectives

By completing this course, you will be able to:

- Understand the internal architecture and core components of modern EDR systems.
- Analyze how EDRs detect threats using static, dynamic, and heuristic techniques.
- Set up and operate a dedicated EDR research lab using a preconfigured virtual environment.
- Reverse engineer real EDR components to identify their logic and protection mechanisms.
- Evaluate and bypass common EDR self-protection strategies (file locks, token control, process monitoring).
- Implement advanced evasion techniques such as process injection, obfuscation, and more.
- Develop custom EDR capabilities, including user-mode and kernel-mode detection logic.
- Utilize Windows internals such as ETW, callback registration, and syscall hooking.
- Create and test kernel drivers for monitoring and detecting malicious behavior.
- Detect real-world malware behaviors such as remote thread injection in a live environment.

Course Prerequisites

- Basic understanding of Windows OS internals (processes, threads, memory, and DLLs).
- Familiarity with x86/x64 assembly language.
- Basic experience with reverse engineering (e.g., using tools like IDA, x64dbg, or Ghidra) – Recommended.
- Basic familiarity with malware analysis techniques.
- Comfortable working with C and scripting (e.g., PowerShell, Python).
- Prior exposure to kernel mode dev and familiarity with x86/X64 assembly - Recommended.
- A strong desire to learn offensive and defensive security concepts at a low level.

Course Format

- **URL**: https://trainsec.net/courses/edr-internals-research-development/
- **Scope and Mode**: Recorded sessions with hands-on labs.
- **Pace**: Self-paced with Lifetime access to materials and community.
- **Platform**: TrainSec.net Learning Platform, Discord server.
- **Certification**: Issued to all participants on “Creedly” platform.

Course Content

Endpoint Detection and Response (EDR) solutions are a core layer of defense in modern enterprise environments. As cyber threats continue to evolve, understanding how EDR systems operate - and how adversaries bypass them - has become essential for both red and blue team professionals. This course offers a comprehensive path into the world of EDR internals, providing the technical depth required to dissect, analyze, and build detection capabilities at both user-mode and kernel-mode levels.

From unpacking the inner workings of commercial EDR engines to building your own detection logic against advanced threats, this training empowers students to think critically and creatively about endpoint defense. The content bridges the gap between malware reverse engineering, low-level Windows internals, and kernel driver development - all delivered through practical, real-world labs and research-focused exercises.

Whether you're defending infrastructure, researching detection bypasses, or engineering the next generation of security tools, this course provides the essential skills to navigate and master the world of EDR.

EDR Foundations & Architectural Principles

- EDR vs. EPP / Antivirus
- EDR Architecture
- EDR Detection Techniques
- Static Engine
- Dynamic Engine
- Heuristic Engine

Reverse-Engineering the Defenses: Hands-On EDR Research Lab

- Research Methodology & Tips
- Deploying an EDR Research Lab (Provided OVA)
- Lead Gathering
- Checking the Exclusions List

- Testing EDR Self-Protection Mechanisms
- Process Tokens & File Permissions
- EDR Persistence Mechanisms
- Reverse Engineering an EDR Component

Offensive Tradecraft: Modern EDR Bypass & Evasion Techniques

- FUD Malware vs. Targeted EDR Bypass
- Rename Obfuscation
- Control-Flow Obfuscation
- IAT & API Hashing
- Strings Encryption
- Process Injection
- Timestomping
- Memory Bombing

Designing the Core: Building User- and Kernel-Mode EDR Components

- EDR Component Design
- API and Syscall Hooking
- Leveraging ETW (Event Tracing for Windows)
- User & Kernel Mode Components

Kernel Driver Engineering for Endpoint Visibility & Control

- Process and Thread Callbacks
- Registry Callbacks
- Other Callback Mechanisms
- File System Minifilters
- Additional Techniques
- User-Kernel Communication

Detection Engineering: Implementing High-Fidelity Threat Analytics

- Remote Thread Injection Detection
- Ransomware Detection
- Real-World Detection Implementation Lab

Required Materials

- **Software/Tools**: Visual Studio, a Hypervisor like VMware Workstation, IDA PRO or Ghidra (will be covered during the course), the Provided Virtual Machine based on our OVA.
- **Discord Community**: https://discord.com/invite/qugcNyWdaU
- **VM Tests Labs**: OVA files provided

Instructor Information

- **Name**: Uriel Kosayev, Pavel Yosifovich
- **Email**: uriel@TrainSec.net, pavel@TrainSec.net
- **Trainers Biography**:
  Uriel Kosayev: Cybersecurity researcher and red teamer who lives on both the offensive and defensive fronts. The author of the “Antivirus Bypass Techniques” and “MAoS - Malware Analysis On Steroids” books, an expert in malware research, reverse engineering, penetration testing, digital forensics, and incident response
  Pavel Yosifovich: Software developer, trainer, consultant, author, and speaker. Co-author of “*Windows Internals*” 7th edition (2017). Author of “*Windows Kernel Programming, 2nd ed*” (2023), “*Windows 10 System Programming Part 1*” (2020), and Part 2 (2021).

Contact & Support

For any course-related or billing-related inquiries, please contact info@TrainSec.net