INFOSECTRAIN
Educate. Excel. Empower.

MALWARE ANALYSIS & REVERSE ENGINEERING TRAINING

Course Highlights

32-Hour LIVE Instructor-led Training

Real-world Malware Samples

Hands-on Session

Covers multiple languages: native PE, .NET, Golang, shellcode

Integrates Memory Forensics

Certified Microsoft Experts

Dedicated Telegram Support Group

Access to Recorded Sessions

Career Guidance & Mentorship

About Course

InfosecTrain's Malware Analysis & Reverse Engineering Training is a hands-on program designed to transform cybersecurity professionals into expert malware analysts. Throughout the 32-hour live sessions, participants will learn how to deconstruct malicious code, identify Indicators of Compromise (IOCs), and reverse-engineer malware behavior to understand how threats operate in real-world networks.

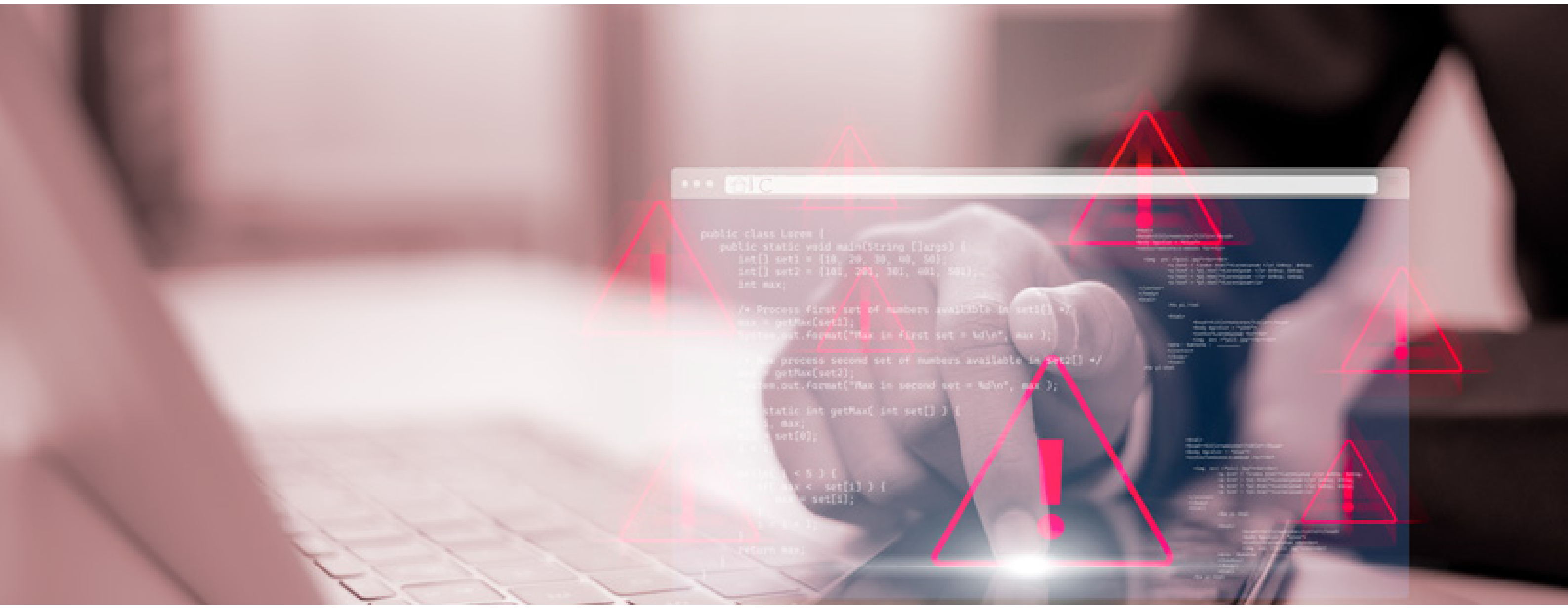
From building an isolated malware lab to mastering disassemblers and memory forensics tools, the course blends theory, practical labs, and mentorship to prepare you for high-demand roles in SOC operations, incident response, and threat research.

Whether you're a beginner or an experienced analyst aiming to enhance your skills, this training provides the clarity, structure, and practical experience to make you confident in analyzing and reporting malware incidents effectively.

Course Objectives

Upon successful completion of the training, participants will be able to:

- Identify and triage suspicious binaries and documents.
- Extract IOCs (file hashes, strings, domains, IPs, mutexes, registry keys) and create actionable YARA rules.
- Perform dynamic sandbox analysis and interpret network/behavioral indicators.
- Reverse engineer functions using disassemblers and debuggers to uncover malicious logic.
- Detect advanced persistence and in-memory techniques such as process injection and shellcode execution.
- Prepare a clear, reproducible malware analysis report for stakeholders.



Target Audience

This training is ideal for:

- SOC Analysts
- Incident Response Teams
- Threat Intelligence Analysts
- Malware Researchers and Reverse Engineers
- Security Engineers and Forensic Investigators
- Penetration Testers interested in binary analysis

Pre-Requisites

- Working knowledge of Windows internals and command line
- Familiarity with networking fundamentals (TCP/IP, ports, common protocols)
- Basic scripting experience (Python, PowerShell)
- Prior exposure to endpoint tools and logs (e.g., EDR, Sysinternals) is advantageous

Course Content

Module 1 Foundations & Static Analysis

- Introduce malware types, lifecycle, threat actor tactics, techniques, and procedures (TTPs)
- Explain static vs dynamic analysis and their applications
- Guide on building a secure, isolated malware analysis lab
- Review Windows internals relevant to malware behavior (processes, services, PE format, registry locations)
- Detail initial triage steps: identifying file hashes, metadata, strings, and suspicious imports

Practical Exercises

- ✓ Analyze a malware sample to extract static features
- ✓ Apply signature-based detections to the sample
- ✓ Create and test YARA rules against the filesystem and running processes

- Discuss dynamic analysis preparation and safe execution practices

Module 2 Dynamic Analysis & Endpoint/Network Forensics

- Explain dynamic analysis workflow and tools (Process Monitor, Process Explorer)

Practical Exercises

- ✓ Execute malware in a controlled environment and monitor behavior
- ✓ Capture endpoint-level activity using EDR and logs
- ✓ Perform network traffic capture and analysis to identify malicious connections

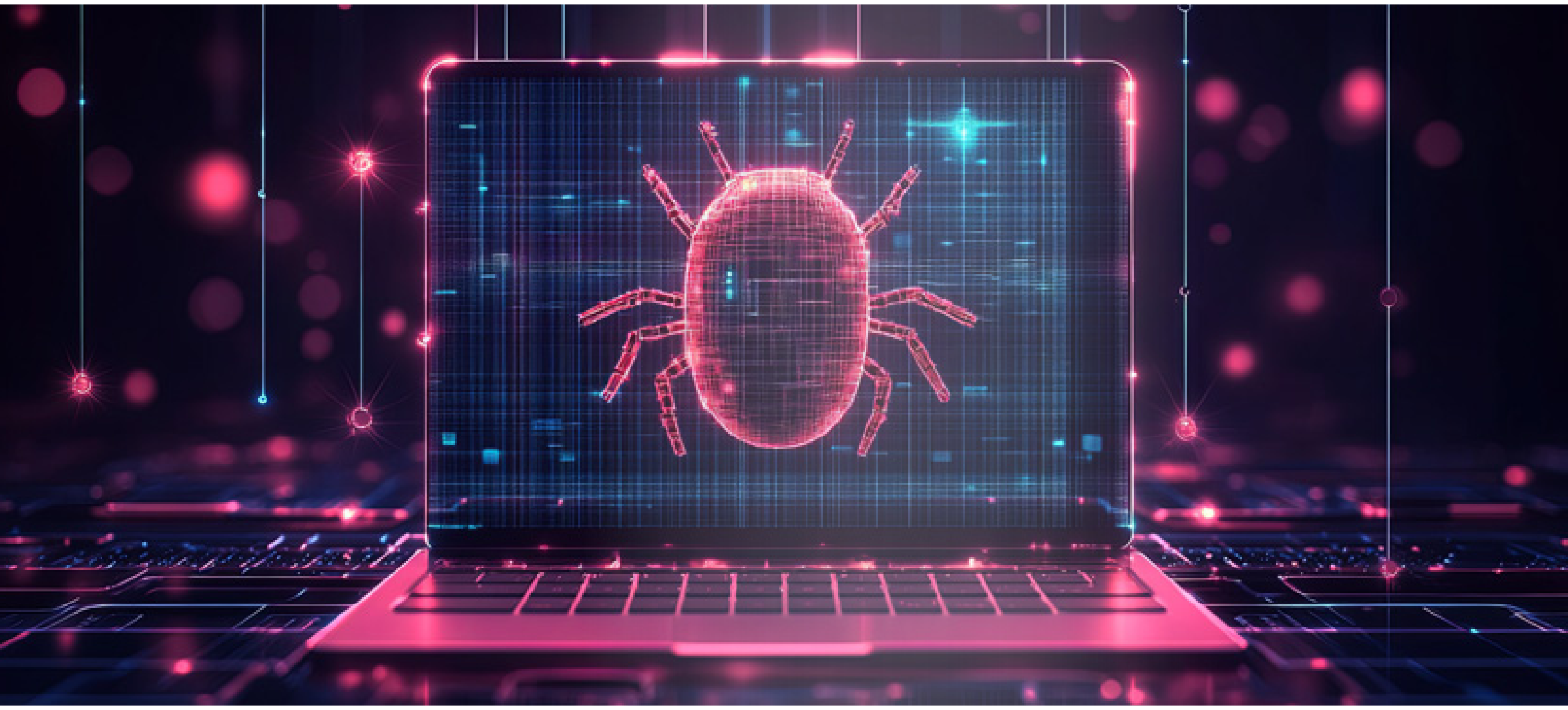
- Demonstrate behavioral monitoring and sandbox analysis interpretation
- Analyze malicious documents (Excel, Word, PDF, RTF, CHM) and identify embedded threats

Module 3 Reverse Engineering & API Analysis

- Present approach and methodology for reverse engineering binaries
- Introduce assembly basics (x86/x64) and instruction patterns
- Overview of reverse engineering tools (IDA, Ghidra, x64dbg, dnSpy) and
- their use cases
- Discuss Windows API functions commonly exploited by malware and sequences for malicious behavior

Practical Exercises

- ✓ Monitor and log API calls to identify malware functionality
- ✓ Analyze process iteration and malicious DLL loading
- ✓ Examine process injection techniques and understand impact
- ✓ Perform detailed code analysis on small binaries



Module 4 Advanced Analysis, Memory Forensics & Reporting

- Explain shellcode analysis techniques and debugging methods

Practical Exercises

- ✓ Identify, dump, and debug shellcode in a safe environment
- ✓ Analyze .NET/C# malware using decompilers to trace logic
- ✓ Unpack and analyze Golang malware samples
- ✓ Acquire memory images and detect in-memory artifacts (shellcode, process hollowing)
- ✓ Analyze stealer malware and ransomware techniques
- ✓ Identify malware persistence mechanisms
- ✓ Compile a professional malware analysis report with IOCs, findings, and remediation recommendations

INFOSECTRAIN
Educate. Excel. Empower.

Contact us

www.infosectrain.com
sales@infosectrain.com

Follow us on