

Malware Analysis I – Introduction to Malware Analysis

Almost every Incident Response involves some Trojan, back door, virus component or rootkit. Incident Responders must be able to perform rapid analysis on the malware encountered in an effort to determine the purpose of unknown code. Without understanding the function of the malware, remediation efforts usually fail to meet expectations. This course provides an introduction to the tools and methodologies used to perform dynamic and static analysis on portable executable programs found on Windows systems.

Duration	2 days
Who Should Attend	Information technology staff, information security staff, corporate investigators or others requiring an understanding of how malware works and the steps and processes involved in Malware Analysis.
Prerequisites	General knowledge of computer and operating system fundamentals is required. Some exposure to software development. Experience in assembly and C, while not required, would be beneficial.
Students Will Learn	<ul style="list-style-type: none">▪ The primary types of malware – A malware bestiary▪ How to create a safe malware analysis environment▪ Malware analysis shortcuts▪ Legal issues involving malware analysis and reverse engineering▪ Methodologies-differences between static and dynamic analysis▪ How malware discovered on real systems was used as part of an elaborate intrusion▪ Bits, bytes, binary, decimal, hexadecimal and converting values between the various numbering conventions▪ Code, compilers and compilation▪ The tools used to identify obfuscation methods used by malware authors and the tools used by analysts to recover the “hidden” data▪ The fundamentals of assembly language programming▪ How to perform dynamic analysis using virtual machines and a system monitoring utilities to capture the system, registry and network activity generated during malware analysis
Exercises	<ul style="list-style-type: none">▪ Dynamic and static analysis of unknown binary▪ Actual intrusion walkthrough, with live malware embedded inside of legitimate content▪ Using VMWare to create a safe analytical environment▪ Converting numbers between base systems▪ Binary creation and analysis
Course Materials	<ul style="list-style-type: none">▪ Student manual▪ Class handouts▪ MANDIANT gear
Suggested Next Courses	<ul style="list-style-type: none">▪ MANDIANT Intermediate Malware Analysis▪ MANDIANT Advanced Malware Analysis▪ MANDIANT Enterprise Incident Response
Contact	1.800.647.7020 education@mandiant.com www.mandiant.com/education.htm

Introduction to Malware

This section introduces the methodology behind malware analysis and establishes a baseline for advanced discussions. Students will be introduced to some of the tools commonly used for malware analysis. Instructors will walk students through how to establish a “safe” environment to conduct malware analysis and the importance of this process.

Malware Analysis Case Studies

Instructors will share their experiences through the study of four different malware binaries that were part of a real intrusion. Students will learn how the binaries were discovered, how they were analyzed and results of the analysis. Students will also gain an understanding of how complex the use of malware was as part of the larger intrusion. In short, they gain the “big picture” methods required to perform malware analysis.

Simple Malware Analysis Tools

Students will be introduced to a variety of “shortcuts” that can be used to facilitate the analysis of commonly used malware. Students will learn how to generate MD5 checksums to identify known malware, various websites which allow upload of suspicious code and other approaches which can give a malware analyst a head start on the analysis process.

Source Code & Compilers

Students will be introduced to the C programming language and create sample C programs to better understand how binary executable programs are produced by malware authors.

Bits and Bytes

When performing malware analysis, you will need review low-level data in a variety of formats. In this section, students will gain a firm understanding of how to convert numbers to and from binary, decimal and hexadecimal. Instructors will also cover more advanced topics critical to malware analysis such as little and big endian bit ordering, signed and unsigned numbers, floating point and character sets. This section covers the basic information required as the foundation for low-level malware analysis.

Introduction to Assembly Language (x86) & Windows Programming

Assembly is the highest-level language that can be reliably recovered from machine code when source code is not available. As a result, a basic understand of assembly is important to the investigator performing malware analysis. Students will learn how x86 processors handle assembly instructions, how memory is handled, how mathematical functions are performed and basic looping processes. All of this is presented at a level appropriate to someone with little or no programming experience.

Methodology and Review

Prior to starting the final exercise, students are reminded of each of the steps involved in dynamic and static malware analysis. Students are provided with an example checklist of steps to help them stay on target during the exercise. Instructors also provide tips and hints from their personal analysis experience to help students reach their goal successfully.

Debugging Malware Walkthrough

As the final exercise, instructors help students through the analysis of an unknown binary found on a compromised system “in the wild”. During this dynamic analysis process, students use all of the tools and techniques learned during the previous sections in an organized manner to determine the functionality and purpose of the file. This exercise also stresses the importance of careful documentation during the malware analysis process.