INDIAN INSTITUTE OF INFORMATION TECHNOLOGY
DESIGN AND MANUFACTURING (IIITDM) KANCHEEPURAM

<table>
<tr><td>Course Code</td><td></td><td>Course Title</td><td colspan="4">Malware Analysis</td></tr>
<tr><td rowspan="2">Dept./Faculty proposing the course</td><td rowspan="2">CSE / Dr.Bhale Pradeepkumar Gajendra</td><td rowspan="2">Structure (LTPC)</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td>3</td><td>0</td><td>2</td><td>4</td></tr>
<tr><td rowspan="2">To be offered for</td><td rowspan="2">UG/PG</td><td>Type</td><td colspan="2">Core ☐</td><td colspan="2">Elective ▬</td></tr>
<tr><td>Status</td><td colspan="2">New ▬</td><td colspan="2">Modification ☐</td></tr>
<tr><td>Pre-requisite</td><td>Familiarity with programming languages</td><td colspan="3">Submitted for approval</td><td colspan="2">Senate # 63</td></tr>
<tr><td>Learning Objectives</td><td colspan="6">• Introduce the fundamentals and impact of malware on computing systems.
• Explain malware behavior, types, and basic classification methods.
• Demonstrate safe handling and containment of malicious code.
• Train students in core static, dynamic, and basic reverse-engineering techniques.
• Emphasize ethical and responsible practices in malware analysis.</td></tr>
<tr><td>Learning Outcomes</td><td colspan="6">Students completing this course will be able to:
• Understand malware as a key cyber-security threat and recognize its impact on systems.
• Observe and interpret basic malicious behaviors during analysis.
• Extract useful indicators and forensic artifacts from malware samples.
• Apply introductory static and dynamic analysis techniques using safe environments and standard tools.
• Demonstrate awareness of defensive strategies and ethical responsibilities in malware analysis</td></tr>
<tr><td>Contents of the course (With approximate break-up of hours for L/T/P)</td><td colspan="6">Foundations of Malware - Introduction to Malware, Evolution and Motivation, Malware Terminology and Components, Basics of OS and Windows Internals for Malware Analysis, Threat Landscape and Real-World Case Studies, Virtual Machines and Safe Lab Setup Basics. (6L)

Malware Classification and Behavior Analysis - Malware Types and Characteristics, Infection Vectors and Payloads, Persistence Mechanisms, Process and Service Manipulation, Anti-Analysis Techniques (Anti-VM, Anti-Debug), Behavioral Indicators and IOC Identification, Sandbox-Based Observation of Malware. (8L)

Static and Dynamic Malware Analysis Techniques - PE File Structure and Metadata Analysis, Hashing and Triage Techniques, String Extraction and Import Table Interpretation, Basic Code Reading and Disassembly Tools, Monitoring System Changes during Execution, API Call Tracing, Handling Packers and Basic Unpacking Techniques. (10L)

Malware Reverse Engineering and Debugging - Assembly basics, debugging workflows, shellcode analysis, payload recovery, Introduction to Kernel-Level Malware and Rootkit Detection Methods. (8L)

Forensics and Defense Strategies - Signature vs. Behavior-Based Detection Techniques, Malware Detection Rules using YARA, AI-driven Malware Classification and Detection, Network Traffic Analysis for Threat Indicators, Memory Forensics with</td></tr>
</table>

	Volatility, Anti-Evasion and Counter-Defense Techniques, Incident Response and SOC Workflows, Ethical and Legal Aspects of Malware Handling. (10L) Practice Component [28 Hours]: Establishing a Safe Malware Analysis Lab - Setup of isolated virtual environments, system snapshots, and secure handling tools for controlled malware experimentation. (2P) Dissecting Malware through Static Analysis - Examination of PE file structures, metadata, hashes, and string indicators to infer functionality without execution. (2P) Exploring Malware Behavior Dynamically - Observation of malware activity within a contained sandbox—tracking processes, registry changes, and file system alterations. (2P) Unpacking Code Logic via Ghidra and Debugging Tools - Performing reverse engineering, control-flow tracing, and payload examination using Ghidra and lightweight debuggers. (3P) Investigating Network and Memory Artifacts - Capturing and analyzing malicious network traffic using Wireshark and performing memory forensics with Volatility for rootkit detection. (2P) Crafting YARA Rules and Linking with IDS - Developing and testing YARA signatures, mapping behavioral indicators to IDS alerts for automated malware detection. (3P)
Text Books	1. Cucci. K, Evasive Malware: A Field Guide to Detecting, Analysing, and Defeating Advanced Threats,1st Ed.,No Starch Press, ISBN-13: 978-1718503267, 2024.
Reference Books	1. Kleymenov, A. and Thabet, A., Mastering Malware Analysis: The Complete Malware Analyst's Guide to Combating Malicious Software, APT, Cybercrime, and IoT Attacks, 1st Ed., Packt Publishing, ISBN-13: 978-1789610789, 2019. 2. Matrosov, A., Rodionov, E and Bratus, S., Rootkits and Bootkits: Reversing Modern Malware and Next Generation Threats, 1st Ed.,No Starch Press, ISBN-13: 978-1593277161, 2019. 3. Saxe, J., & Sanders, H., Malware Data Science: Attack Detection and Attribution, 1st Ed., No Starch Press, ISBN-13: 978-1593278595, 2018. 4. Dang, B., Gazet, A., Bachaalany, E. and Josse, S., Practical Reverse Engineering: x86, x64, ARM, Windows Kernel, Reversing Tools, and Obfuscation, 1st Ed.,John Wiley & Sons, ISBN-13: 978-1118787311, 2014. 5. Sikorski, M. and Honig, A., Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software, 2nd Ed., No Starch Press, ISBN-13. 978-1593274306, 2012.