INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)
Dundigal - 500 043, Hyderabad, Telangana

COURSE CONTENT

MALWARE ANALYSIS AND REVERSE ENGINEERING								
VI Semester: CSE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
ACSD35	**Elective**	**3**	**-**	**-**	**3**	**40**	**60**	**100**
Contact Classes: 48	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 48**		
Prerequisite: Information Security and Management								

I. COURSE OVERVIEW:

This course provides students a foundational knowledge about reverse engineering and malware analysis, through the study of various cases and hand-on analysis of malware samples. It covers fundamental concepts in malware investigations so as to equip the students with enough background knowledge in handling malicious software attacks. Various malware incidents will be covered, such as cases in Ransomware, banking-trojan, state-sponsored and APT attacks, cases in Stuxnet and malicious software attacks on Industrial Control System and IoT devices. With the experience of studying these cases and analyzing selected samples, the students will be able to understand the global cyber security landscape and its future impact. Hands-on exercises and in-depth discussion will be provided to enable students to acquire the required knowledge and skill set for defending and protecting an enterprise network environment.

II. COURSES OBJECTIVES:

The students will try to learn:

I. The methodology of malware analysis and techniques for analyzing malware.
II. An insight to fundamentals of malware analysis which includes analysis of JIT compilers for malware detection in legitimate code.
III. DNS filtering and reverse engineering and memory forensics and volatility.
IV. Identifying Injected Code with Mal find and YARA.

III. COURSE OUTCOMES:

At the end of the course students should be able to:

CO1 Understand the concept of malware and reverse engineering and implement tools and techniques of malware analysis.
CO2 Analyze the security risks, threats and potential vulnerabilities on enterprise networks environment.
CO3 Carry out independent analysis of modern malware samples using behavioral, code analysis and memory forensic techniques.
CO4 Apply the learned techniques to protect, reduce the security risks and avoid malicious software attacks on computer systems or networks.
CO5 Research independently and use learned skills and tools to investigate malicious software attacks and implement or update a cyber protection plan.
CO6 Evaluate malware behavior in reverse Engineering tools and Techniques.

IV. COURSE CONTENT:

MODULE - I: INTRODUCTION TO MALWARE (10)

Fundamentals of malware analysis (ma), reverse engineering malware (REM) methodology, brief overview of malware analysis lab setup and configuration, introduction to key ma tools and techniques, behavioral analysis vs. code analysis, resources for reverse-engineering malware (REM) understanding malware threats, malware indicators, malware classification, examining clamav signatures, creating custom clamav databases, using YARA to detect malware capabilities, creating a controlled and isolated laboratory, introduction to ma sandboxes, ubuntu, zeltser'sremnux, sans sift, sandbox setup and configuration new course form, routing TCP/IP connections, capturing and Analyzing Network Traffic, Internet simulation using I Net Sim, using deep freeze to preserve physical systems, using FOG for cloning and imaging disks, using MySQL database to automate FOG tasks.

MODULE – II: MALWARE FORENSICS (09)

Using TSK for network and host discoveries, using Microsoft offline API to registry discoveries, identifying packers using PEiD, registry forensics with reg ripper plu-gins: bypassing poison Ivy's locked files, bypassing conficker's file system ACL Restrictions, detecting rogue Pki certificates.

MODULE – III: MALWARE AND KERNEL DEBUGGING (10)

Opening and attaching to processes, configuration of JIT debugger for shellcode analysis, controlling program execution, setting and catching breakpoints, debugging with python scripts and Py commands.

DLL export enumeration, execution, and debugging, debugging a VMware workstation guest (on windows), debugging a parallels guest (on Mac OS X). Introduction to WinDbg commands and controls, detecting rootkits with WinDbg scripts, kernel debugging with IDA Pro.

MODULE – IV: MEMORY FORENSICS AND VOLATILITY (10)

Memory dumping with Moonsols windows memory toolkit, accessing VM memory files overview of volatility, investigating processes in memory dumps, code injection and extraction, detecting and capturing suspicious loaded DLLS, finding artifacts in process memory, identifying injected code with Malfind and YARA**.**

MODULE – V: RESEARCHING AND MAPPING SOURCE DOMAINS/IPS (9)

Using Whois to research domains, DNS hostname resolution, querying passive, DNS checking, DNS records, reverse IP search new course form, creating static maps, creating interactive maps. case study of finding artifacts in process memory, identifying injected code with Malfind and YARA.

IV. TEXT BOOKS:

1. The Hands-On Guide to Dissecting Malicious Software Michael Sikorski and Andrew Honig No Starch Press 1st Edition.
2. Practical Malware Analysis: The Hands-On Guide to Dissection Malicious Software by Michael Sikorski and Andrew Honig, 2012.
3. The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory by Michael Hale Ligh, Andrew Case, Jamie Levy, AAron Walters, 1st Edition, 2014.

V. REFERENCE BOOKS:

1. Dang, Gazet and Bachaalany, "Practical Reverse Engineering", Wiley publications.
2. Jamie Butler and Greg Hoglund, "Rootkits: Subverting the Windows Kernel", Addison-Wesley.

VI. ELECTRONICS RESOURCES:

1. https://www.sans.org/cyber-security-courses/reverse-engineering-malware-malware-analysis-tools-techniques/
2. https://www.udemy.com/course/malware-analysis-and-reverse-engineering/?couponCode=LEADERSALE24B

VII. MATERIALS ONLINE

1. Course outline description
2. Tutorial question bank
3. Definition and terminology
4. Tech-talk topics
5. Open-ended experiments
6. Assignments
7. Model question paper - I
8. Model question paper - II
9. Lecture notes
10. Power Point Presentation
11. Early learning readiness videos (ELRV)