Indian Institute of Technology Kanpur

Department of Computer Science and Engineering

Proposal for a New Course

1. Course Title: Binary Instrumentation for Systems Security

2. Course No.:

3. Credits: 3-0-0-0 [9]

4. Duration of Course: Full Semester

5. Who can take this course: UG/PG students with a background in operating systems (CS330) or equivalent.

6. Proposing Department: CSE

7. Other Departments/IDPs which may be interested in the proposed course: Electrical/Electronics Engineering.

8. Other faculty members interested in teaching the proposed course: Any faculty member working in systems or security area.

9. Proposing Instructor: Soumyakant Priyadarshan, Department of CSE.

10. Objectives:

The objective of this course is to provide students with a practical understanding of **systems security, with an emphasis on binary analysis and instrumentation**. Modern security analysis often requires understanding, monitoring, and securing software at the binary level, particularly when source code is unavailable, untrusted, or insufficient to reveal runtime behavior. Binary analysis, reverse engineering, and instrumentation therefore play a key role in **vulnerability discovery, malware analysis, exploit analysis, and enforcement of security mechanisms**. The course will cover memory-safety vulnerabilities and defenses, static and dynamic binary analysis and instrumentation, vulnerability discovery, and reverse engineering. A semester-long hands-on project will allow students to apply these concepts to analyze a vulnerable program and progressively develop a working exploit.

11. Contents:

S.No.	Broad Title	Topics	No. of Lectures
1	**Memory Safety and Software Exploitation**	• Program memory layout and memory-safety threat models • Stack-based buffer overflows and control-flow hijacking • Return-to-libc and return-oriented programming (ROP)	4
2	**Memory-Safety and Control-Flow Defenses**	• Stack canaries and shadow stacks • DEP/NX and ASLR • Control-flow integrity (CFI) and related defenses • Limitations and bypasses of security mechanisms	4
3	**Binary Analysis and Instrumentation Fundamentals**	• Need for binary analysis and instrumentation in systems security • Binary fundamentals: compilation, linking and loading, ELF, sections/segments, symbols, relocations, PLT/GOT, assembly and calling conventions • Basic reverse engineering and disassembly • Challenges in static binary analysis • Static vs. dynamic binary analysis	4
4	**Dynamic Binary Analysis and Instrumentation**	• Runtime program analysis and execution tracing • Instruction-, basic-block- and function-level instrumentation • Monitoring memory accesses and control transfers • limitations of dynamic analysis	5

5	**Static Binary Analysis and Instrumentation**	• Disassembly, basic-block and function identification • Control-flow graph and call-graph recovery • Indirect control-transfer resolution • Static binary instrumentation and rewriting • Security applications and limitations of static analysis	5
6	**Advanced Program Analysis for Security**	• Taint analysis and information-flow tracking • Program slicing • Symbolic and concolic execution • Path constraints and constraint solving • Applications to vulnerability analysis and input generation	4
7	**Fuzzing and Automated Vulnerability Discovery**	• Fuzzing fundamentals and input generation/mutation • Coverage-guided fuzzing and coverage instrumentation • Sanitizers, crash detection, minimization and triage • Hybrid fuzzing and integration with program analysis	4
8	**Reverse Engineering and Malware Analysis**	• Reverse-engineering workflows and analysis of stripped binaries • Static and dynamic malware analysis • Obfuscation and packing • Anti-analysis and anti-debugging techniques	3
9	**Security of DNN Systems**	• Security threat models for deployed DNNs • Adversarial examples and backdoor attacks • Model supply-chain threats and model tampering • Attacks and defenses for DNN systems	2
10	**Binary Analysis and Instrumentation for DNN Security**	• Deployment of DNNs in compiled/binary form • Instrumentation of DNN inference • Recovery of runtime and model information from binaries • Analysis of intermediate activations and execution behavior • Binary-level backdoor detection and sanitization	4
Total			**39**

Reference books: Relevant research papers, tool documentation, and selected readings will be provided.

Text books: None.

Dated 14/09/2026 Proposer: Soumyakant Priyadarshan

Dated _____ DPGC Convener: _____

The course is approved/not-approved

SPGC/SUGC Chairperson

Dated: _____