Indian Institute of Technology Kanpur

Proposal for a New Course

1. Course No: A **300 level** number requested**.**
2. Course Title: **Vulnerability Analysis and Penetration Testing**
3. No. of Lectures per week: 0 (L), Tutorial: 1 (T), Laboratory: 3 (P), Additional Hours: 3 (A),

   Credits (3*L+2*T+P+A): 08 Duration of Course: Full Semester
4. Proposing Department/IDP : WSAIS

   Other Departments/IDPs which may be interested in the proposed course: CSE

   Other faculty members interested in teaching the course: Somitra Sanadhya, Manindra Agrawal
5. Proposing Instructor: DUGC (WSAIS)
6. Course Description:

A) Objectives: The objective of this practical-based course is to impart hands-on training in vulnerability analysis and penetration testing. Students will be trained to identify, exploit, and report security weaknesses in systems, networks, and applications.

B) Contents: **(for 13 weeks of labs, tutorials, and additional work)**

S.No	Type	Title	Topics	Tools	
1	Theory	VAPT Lab Setup	External vs internal testing scope, attack paths, legal boundaries, lab safety.	VirtualBox / VMware Player, Kali Linux, DVWA, Metasploitable2	1 hour
	Practical		**VAPT Lab Setup:** **Lab Objectives:** To understand the difference between external and internal testing environments by implementing a controlled VAPT lab. To safely configure attacker, target, and client systems while respecting legal and ethical boundaries. **Tasks:** Install VirtualBox or VMware Player on the host system. Import and configure the following virtual machines: • Kali Linux (Attacker) • DVWA (Web Application Target) • Metasploitable2 (Vulnerable Server) • Optional internal		3 hours

			client machine Configure network modes: • NAT for external testing simulation • Host-only or Internal Network for internal testing simulation Create a segmented virtual network separating: • Attacker machine • Target servers • Internal client system Verify network connectivity using: • ip a • ping • ifconfig Confirm isolation from the host's real network. Document IP addresses and network topology. Ensure lab safety by disabling internet access where required.		
2	Theory	Passive Reconnaissance and Attack Surface Mapping	Passive reconnaissance principles, data correlation, asset visibility.	theHarvester & Whois	1 hour
	Practical		**Passive Reconnaissance and Attack Surface Mapping:** **Lab Objectives:** • To collect publicly available information about a target without directly interacting with its systems. • To identify exposed digital assets and understand the organization's external attack surface. **Tasks:** Identify the target domain or organization name provided for the lab. Perform passive reconnaissance using **theHarvester** to gather: • Domains and subdomains • Email addresses		3 hours

			• Public IP addresses Perform WHOIS lookup using: • whois <domain> Extract and document: • Domain registration details • Name servers • IP address ranges Correlate data from theHarvester and WHOIS to identify: • Exposed domains and subdomains • Associated IP ranges Identify publicly accessible services linked to discovered IPs. Create an attack surface map showing: • Domains • IP ranges • Publicly exposed services		
3	Theory	Active Reconnaissance and Network Discovery	Safe active scanning, noise control, interpretation of results.	Nmap, Netdiscover	1 hour
	Practical		**Active Reconnaissance and Network Discovery:** **Lab Objectives:** • To discover live hosts and network services using safe and controlled active scanning techniques. • To interpret scan results accurately while minimizing unnecessary network noise. **Tasks:** Identify the target network range from the lab setup. Perform live host discovery using: • nmap -sn <network-range> • netdiscover Compare results from Nmap and Netdiscover. Perform port scanning on discovered live hosts: • TCP SYN scan		3 hours

			using nmap -sS • Full port scan using nmap -p- Identify running services and versions using: • nmap -sV Document open ports, services, and service versions. Analyse scan output to identify potential attack entry points. Save scan results for further enumeration and exploitation stages.		
4	Theory	Deep Service Enumeration	Why enumeration leads to real attacks, common service misconfigurations.	Nmap NSE scripts, Nikto	1 hour
	Practical		**Deep Service Enumeration:** **Lab Objectives:** • To perform detailed enumeration of network services in order to identify misconfigurations and weaknesses. • To gather service-level information that can be leveraged for real-world exploitation. **Tasks:** Select live hosts identified during active reconnaissance. Perform service-specific enumeration using **Nmap NSE scripts**: • DNS enumeration using dns-* scripts • SMB enumeration using smb-* scripts • FTP enumeration using ftp-* scripts • SSH enumeration using ssh-* scripts Identify anonymous or weak access configurations in FTP services. Enumerate SMB shares and user information where available. Perform HTTP service enumeration using:		3 hours

			• Nmap HTTP NSE scripts • **Nikto** for web server misconfiguration detection Identify outdated services, default files, and insecure configurations. Document findings that could lead to further exploitation.		
5	Theory	Automated Vulnerability Assessment	Scanner logic, coverage vs depth, prioritisation.	OpenVAS Community Edition	1 hour
	Practical		**Automated Vulnerability Assessment:** **Lab Objectives:** • To identify known security vulnerabilities using automated vulnerability scanning techniques. • To analyse scan results and prioritize vulnerabilities based on severity and impact **Tasks:** Install and configure **OpenVAS Community Edition**. Update vulnerability feeds and verify scanner readiness. Create a target for the vulnerable system identified in previous labs. Perform an **unauthenticated vulnerability scan**. Perform an **authenticated vulnerability scan** using valid credentials (if provided). Monitor scan progress and ensure complete coverage. Analyse scan results to identify: • CVE IDs • Severity ratings • Affected services Compare authenticated vs unauthenticated scan results.		3 hours

			Prioritize vulnerabilities based on risk and exploitability. Export and save scan reports for further validation.		
6	Theory	Manual Vulnerability Validation and Exploit Selection	False positives, exploit feasibility, impact assessment.	Nmap scripts, Netcat	1 hour
	Practical		**Manual Vulnerability Validation and Exploit Selection:** **Lab Objectives:** • To validate high-risk vulnerabilities manually and eliminate false positives reported by scanners. • To evaluate exploit feasibility and assess the real-world impact of identified vulnerabilities. **Tasks:** Review high-severity vulnerabilities identified from automated scan reports. Select vulnerabilities suitable for manual validation. Use **Nmap NSE scripts** to manually verify service-specific vulnerabilities. Perform manual connectivity and banner checks using **Netcat**. Validate whether the vulnerability is exploitable or a false positive. Research publicly available exploits related to confirmed vulnerabilities. Assess the potential impact of each validated vulnerability. Document validated vulnerabilities with supporting command outputs.		3 hours
7	Theory	Web Application	Server-side trust boundaries, injection flaws.	DVWA, Burp Suite	1 hour

	Practical	Exploitation – Server Side	**Web Application Exploitation – Server Side:** **Lab Objectives:** • To identify and exploit server-side injection vulnerabilities in web applications. • To understand how improper input validation leads to backend system compromise. **Tasks:** Access the vulnerable web application (DVWA) from the attacker machine. Configure **Burp Suite Community Edition** as an intercepting proxy. Identify input fields and parameters processed by the server. Test parameters for **SQL Injection** using manual payloads. Intercept and analyse SQL Injection requests in Burp Suite. Exploit SQL Injection using **SQLmap** to: • Enumerate databases • Enumerate tables • Extract sensitive data Identify command injection points in the application. Test command injection using OS command payloads. Execute system-level commands through the web interface. Capture and document successful exploitation results.	Community Edition, SQLmap	3 hours
8	Theory	Web Application Exploitation – Client Side	Client-side execution context, impact of XSS.	DVWA, Burp Suite Community Edition	1 hour
	Practical		**Web Application Exploitation – Client Side:** **Lab Objectives:** • To identify client-side vulnerabilities that allow execution		3 hours

			of malicious scripts in a user's browser. • To understand the security impact of reflected and stored Cross-Site Scripting (XSS) attacks. **Tasks:** Access the vulnerable web application (DVWA) from the attacker machine. Configure **Burp Suite Community Edition** as an intercepting proxy. Identify user input points reflected in the application response. Test input fields for **reflected XSS** using JavaScript payloads. Intercept and modify requests using Burp Suite to confirm XSS behavior. Identify persistent input fields vulnerable to **stored XSS**. Inject malicious JavaScript payloads into stored input fields. Access affected pages to observe script execution. Analyse the impact of XSS on user sessions and data. Document payloads used and observed outcomes.		
9	Theory	Authentication, Authorization, and Session Weaknesses	Broken authentication flows, access control failures.	Burp Suite Community Edition, OWASP ZAP	1 hour
	Practical		**Testing Login Mechanisms, Session Handling, and Privilege Misuse:** **Lab Objectives:** • To analyse authentication mechanisms for weaknesses. • To test authorization controls and identify access control failures. • To examine session handling issues that can lead to account or privilege		3 hours

			compromise. **Tasks:** Access the login functionality of the vulnerable web application. Configure Burp Suite Community Edition as an intercepting proxy. Intercept login requests and identify authentication parameters. Test login mechanisms for: • Weak credentials • Default credentials • Improper error messages Use OWASP ZAP to perform controlled authentication testing. Capture session cookies after successful login. Analyse session handling by: • Reusing session cookies after logout • Modifying session identifiers Attempt to access restricted pages using a low-privileged account. Test horizontal and vertical privilege escalation by modifying parameters. Identify broken authentication and access control vulnerabilities. Document findings with request/response evidence.		
10	Theory	System Exploitation and Privilege Escalation	Local privilege escalation concepts and attack paths.	Metasploit Framework, Linux privilege escalation scripts	1 hour
	Practical		**System Exploitation and Privilege Escalation:** **Lab Objectives:** • To gain initial shell access on a vulnerable system. • To escalate privileges by exploiting local system misconfigurations. **Tasks:** Identify exploitable services on the target system. Launch Metasploit		3 hours

			Framework from the attacker machine. Select and execute an appropriate exploit to gain initial shell access. Verify shell access and current privilege level. Enumerate the system to identify privilege escalation vectors: • Kernel version • SUID binaries • Misconfigured services Transfer and execute Linux privilege escalation scripts. Analyse script output to identify escalation opportunities. Exploit a valid privilege escalation path. Gain root or administrative access. Capture proof of privilege escalation. Document the complete attack path and commands used.		
11	Theory	Internal Pivoting and Lateral Movement	Internal network attacks, pivoting concepts, lateral movement.	Metasploit Framework	1 hour
	Practical		**Internal Pivoting and Lateral Movement:** **Lab Objectives:** To understand how attackers move inside an internal network after initial compromise. To use a compromised system as a pivot point to access internal-only network resources. **Tasks:** Use an already compromised system obtained from previous labs. Identify network interfaces and routes on the compromised host. Determine internal network ranges inaccessible directly from the attacker machine. Configure Metasploit to		3 hours

			enable pivoting through the compromised system. Add internal routes using Metasploit pivoting features. Scan internal network hosts and services through the pivot. Identify internal systems and exposed services. Attempt access to an internal service using the pivoted connection. Validate successful lateral movement to another internal resource. Document pivoting steps, accessed resources, and observed limitations.		
12	Theory	Log Analysis and Attack Traceability	Importance of logs, attacker traces, basic blue-team visibility.	Linux logs, Windows Event Viewer	1 hour
	Practical		**Log Analysis and Attack Traceability:** **Lab Objectives:** To analyse system and application logs for evidence of attacks. To trace attacker actions using log entries generated during exploitation. **Tasks:** Access the target Linux system used in previous exploitation labs. Review Linux authentication logs: • /var/log/auth.log Identify failed and successful login attempts. Analyse web server logs: • access.log • error.log Identify indicators of: • SQL injection attempts • Command injection payloads Correlate log timestamps with attack activities performed in earlier labs. Access the Windows target system (if available). Open Windows Event Viewer and analyse: • Security logs		3 hours

			• System logs Identify events related to: • User logins • Privilege escalation Map attacker actions to corresponding log entries. Document attack footprints and traceability evidence.		
13	Theory	Full-Scope VAPT with Reporting	End-to-end VAPT lifecycle and reporting expectations.	**Manual** / Automated tools	1 hour
	Practical		**Full-Scope Vulnerability Assessment and Penetration Testing with Reporting:** **Lab Objectives:** • To perform a complete VAPT engagement following industry methodology. • To document findings in a structured, professional VAPT report. **Tasks:** Define assessment scope and rules of engagement. Identify target systems, applications, and network ranges. Perform passive reconnaissance to gather publicly available information. Conduct active reconnaissance and service discovery. Enumerate identified services and applications. Perform automated vulnerability scanning. Manually validate high-risk vulnerabilities. Exploit confirmed vulnerabilities in a controlled manner. Capture evidence including: • Screenshots • Command outputs • Exploitation proof Assess technical and security impact of each finding. Assign severity levels based		3 hours

			on risk. Prepare a structured VAPT report including: • Executive summary • Scope and methodology • Vulnerability findings with proof of concept • Risk impact analysis • Remediation recommendations Review and finalize the report for submission.		

C) Pre-requisites:

- Working knowledge of Windows and Linux OS
- Basic networking concepts
- Basic understanding of web technologies
- Introductory cyber security knowledge

D) Short summary for including in the Courses of Study Booklet: The course provides hands-on training to the students on the fundamentals of vulnerability analysis and penetration testing (VAPT). Students will be trained to Understand VAPT methodologies and lifecycle, Perform vulnerability scanning and assessment, Conduct penetration testing on networks and web applications, Exploit common vulnerabilities safely, and Prepare professional VAPT reports.

7. Recommended books:
   • Tools which will be used are already listed in the weekly description.

Dated: 17 July 2026; Proposer: DUGC, WSAIS

Dated: ; DUGC Convener (WSAIS):_____

The course is approved / not approved

_____

Chairperson, SUGC

Dated:_____