KLEAP Institute of Information Security™

KIIS

Ethical Hacking & VAPT

2026

Ethical Hacking & VAPT

3-Month Course Outline

Comprehensive Training Program

Duration: 3 Months | 19 Modules | Theory + Practical Labs

Part 1: Ethical Hacking (Month 1 – 1.5) | Modules 1–11

Part 2: VAPT & Web Application Security (Month 1.5 – 3) | Modules 12–19

PART 1 — ETHICAL HACKING (Months 1 – 1.5)

Module 1 Introduction to Cybersecurity & Ethical Hacking

What is Cybersecurity?

- Definition and importance of cybersecurity
- Types of cyber threats and attacks
- Career paths in cybersecurity

What is Ethical Hacking?

- Definition ofethicalhacking
- Legal and ethical boundaries
- Scope of ethical hacking

Types of Hackers

- White Hat Hackers — authorized security professionals
- Black Hat Hackers — malicious/unauthorized attackers
- Grey Hat Hackers — in-between, no explicit permission
- Script Kiddies — use existing tools without deep knowledge
- Hacktivists — hacking for political or social causes

Key Terminologies

- Vulnerability —weakness in a system
- Threat — potential danger to the system
- Exploit — code/technique to take advantage of a vulnerability
- Payload — malicious code delivered after exploit
- Zero-Day — unknown vulnerability with no patch available

Phases of Ethical Hacking

- Reconnaissance — passive/active information gathering
- Scanning — identify open ports, services, and vulnerabilities
- Enumeration — extract detailed info from target
- Exploitation — gain unauthorized access
- Maintaining Access — persist on the target system
- Covering Tracks — erase logs and evidence

Elements of Information Security (CIA Triad)

- Confidentiality — protect data from unauthorized access
- Integrity — ensure data accuracy and trustworthiness
- Availability — ensure systems are accessible when needed

Cyber Laws & Ethics

- Overviewof IT Act and cybercrime laws
- Importance of written authorization before testing
- Responsible disclosure practices

Google Dorking for Awareness

- Usingadvanced Googlesearch operators
- Finding exposed files, login pages, and sensitive data

Practical Session

- Basic Google Dork queries on demo targets
- Introduction to the virtual security lab environment

Module 2 Basics of Networking

What is Networking?

- How computers communicate over a network
- Types of Networks: LAN, WAN, MAN, PAN

NetworkTopologies

- Star — central hub connects all devices
- Bus — single backbone cable
- Ring — each device connected in a circle
- Mesh — every device interconnected

IPAddressing

- What is an IP address and how it works
- Public vs Private IP addresses
- Static vs Dynamic IP addresses
- IP address classes: A, B, C, D, E
- Localhost (127.0.0.1) and loopback interface

OSIModel—7 Layers

- Layer 1: Physical — cables, signals, bits
- Layer 2: Data Link — MAC addresses, switches
- Layer 3: Network — IP addressing, routers
- Layer 4: Transport — TCP/UDP, segmentation
- Layer 5: Session — session management
- Layer 6: Presentation — encryption, data formatting
- Layer 7: Application — HTTP, FTP, DNS, SMTP

TCP/IPModel&Protocols

- TCP/IP 4-layer model vs OSI
- TCPThree-WayHandshake:SYN→SYN-ACK → ACK
-
-

  TCP vs UDP — reliability vs speed
  Data Encapsulation and Decapsulation

Networking Devices

- Router — connects different networks
- Switch — connects devices in LAN
- Hub — broadcasts to all devices
- Firewall — filters traffic by rules

Services,Protocols &Ports

- HTTP (80), HTTPS (443) — web browsing
- FTP (21) — file transfer
- SSH (22) — secure remote access
- DNS (53) — domain name resolution
- SMTP (25) — email sending

PracticalSession

- IP configuration using ipconfig / ifconfig
- Capturing and analyzing packets using Wireshark

- Identifying TCP handshake in packet capture

Module 3 Setting Up a Hacking Lab

VirtualizationConcepts

- What is virtualization and why it matters for hacking
- Types of hypervisors: Type 1 and Type 2

Installing VirtualBox

- Downloading and installing VirtualBox
- Configuring virtual machine settings (RAM, CPU, storage)
- Network adapter modes: NAT, Host-Only, Bridged

InstallingKaliLinux

- Downloading Kali Linux ISO
- Creating and booting VM from ISO
- Installing Kali Linux step by step
- Post-installation updates and configuration

InstallingMetasploitable 2

- Downloading Metasploitable 2 virtual disk
- Importing into VirtualBox
- Configuring network for lab communication
- Testing connectivity between Kali and Metasploitable

InstallingSecurityTools

- Installing Burp Suite Community Edition
- Installing and updating Wireshark
- Verifying all tools are operational

PracticalSession

- Build a complete isolated virtual hacking lab
- Ping test between Kali Linux and Metasploitable 2

Module 4 Windows & Linux Commands

Windows Commands

- systeminfo — display full system information
- ipconfig / ipconfig /all — network configuration
- ping — test network connectivity
- tracert — trace packet route to destination
- netstat — active connections and ports
- nslookup — query DNS records
- tasklist / taskkill — manage processes

Linux Commands

- ls / ls -la — list directory contents
- pwd — print current working directory
- mkdir / rm / cp / mv — file management
- chmod / chown — file permissions and ownership
- ifconfig / ip a — network interface information
- grep — search text patterns in files

- ps / top — process monitoring
- netdiscover — discover hosts on local network
- cat / nano / vim — view and edit files
- wget / curl — download files from the internet

PracticalSession

- Enumerate system information on Windows and Linux
- Network interface configuration and analysis
- Discover active devices in the local network using netdiscover

Module 5 Bash Scripting

ShellBasics

- What is a Shell? — command interpreter
- Types of shells: bash, sh, zsh
- Hello World script — shebang line, echo command

VariablesandInput

- Declaring and using variables
- Reading user input with read command
- Environment variables and export

Operators andExpressions

- Arithmetic operators: +, -, *, /, %
- Comparison operators: -eq, -ne, -gt, -lt
- String operators: =, !=, -z, -n
- Logical operators: &&, ||, !

ControlStructures

- if / elif / else — conditional branching
- case / esac — switch-style selection
- for loop — iterate over a list or range
- while loop — repeat while condition is true
- until loop — repeat until condition is true

Functions and Arrays

- Defining and calling functions
- Passing arguments to functions
- Declaring and accessing arrays
- Looping through arrays

File Input/Output and Error Handling

- Reading from files line by line
- Writing output to files with >, >>
- Redirecting errors with 2>, 2>&1
- Exit codes and error handling with $?

PracticalSession

- Write a Bash script to ping multiple hosts and report results
- Create an automation script for basic network scanning with Nmap
- Build a simple log parser script

Module 6 Information Gathering & OSINT

What is Footprinting?

- Definition offootprinting and its role in hacking
- Passive Reconnaissance — no direct interaction with target
- Active Reconnaissance — direct interaction with target systems

Google Dorking

- Advanced Google search operators
- site: — limit results to specific domain
- filetype: — find specific file types (pdf, xls, doc)
- intitle: / inurl: — search within page title or URL
- 'index of' — find open directories
- intext: — search within page body content
- Finding exposed admin panels, login pages, cameras

WHOIS Lookup

- WhatWHOIS is and what data it reveals
- Extracting registrant name, email, phone, address
- Finding registration and expiry dates
- Identifying name servers and registrar info
- Tools: whois command, who.is, domaintools.com

DNS Enumeration

- What DNS records reveal about a target
- A Record — maps domain to IPv4 address
- AAAA Record — maps domain to IPv6 address
- MX Record — mail exchange servers
- NS Record — name servers for the domain
- TXT Record — SPF, DKIM, verification records
- CNAME Record — alias for another domain
- Zone Transfer attack — AXFR query
- Tools: nslookup, dig, dnsenum, fierce

Email Footprinting

- Gathering email addresses of target organization
- Email header analysis — tracing origin IP
- Identifying email server and client info from headers
- Tools: theHarvester, hunter.io, email header analyzers

Social Media Footprinting

- Extracting information from LinkedIn, Facebook, Twitter
- Identifying employees, roles, and org structure
- Finding internal tools, technologies, and projects
- Analyzing public posts for sensitive info leaks

Shodan.io

- WhatShodan is — a search engine for internet-connected devices
- Finding exposed webcams, routers, ICS/SCADA systems
- Searching by port, product, country, and organization
- Key filters: port:, org:, product:, os:, country:
- Identifying vulnerable services exposed to the internet

OSINT Framework

- Overview of osintframework.com
- Categories: usernames, email, domain, IP, social networks
- Selecting the right OSINT tool for each category

FootprintingthroughWebsites

- Extracting metadata from web pages
- Checking robots.txt and sitemap.xml for hidden paths
- Source code review for comments and hidden links
- Finding subdomains using tools and brute force
- Web archive analysis using Wayback Machine
- Tools: Wappalyzer (technology fingerprinting), BuiltWith

Target Network Footprinting

- Identifying IP ranges and ASN of target organization
- BGP routing information and network topology
- Traceroute analysis to map network path
- Tools: ARIN, RIPE, BGPView, Hurricane Electric

PracticalSession

- Perform full OSINT on a demo target domain
- WHOIS lookup and DNS enumeration using dig and dnsenum
- Shodan searches for exposed services
- theHarvester — collect emails and subdomains

Module 7 Scanning Phase

What is Scanning?

- Purpose of scanning in the hacking methodology
- Types of scanning: port, network, vulnerability

Nmap—Network Mapper

- nmap [target] — basic single host scan
- nmap 192.168.1.0/24 — entire subnet scan
- nmap -iL targets.txt — scan from a file
- nmap -p 80,443,22 — specific port scan
- nmap -p- — all 65535 ports scan
- nmap -sV — version detection scan
- nmap -O — OS detection
- nmap -A — aggressive scan (OS, version, scripts, traceroute)
- nmap -T4 — timing template (faster scan)
- nmap -F — fast scan (top 100 ports)
- nmap -sS — TCP SYN stealth scan
- nmap -sT — TCP connect scan
- nmap -sU — UDP scan
- nmap -sN — TCP NULL scan
- nmap -sX — Xmas scan

BannerGrabbing

- What banner grabbing reveals (service, version, OS)
- Using Netcat for manual banner grabbing: nc -v [ip] [port]
- Using Nmap banner script: --script=banner
- Using Telnet for quick banner check

Practical Session

- Perform scanning on Metasploitable 2
- Identify running services and their versions
- Run stealth scan and compare results with connect scan
- Use version detection and OS fingerprinting

Module 8 Enumeration

What is Enumeration?

- Differencebetweenscanning and enumeration
- Goal: extract detailed info — usernames, shares, services

DNS Enumeration

- Brute-force subdomains using dnsenum and fierce
- Zone transfer test using dig axfr
- dnsrecon — comprehensive DNS recon tool

SMB Enumeration

- WhatSMB (Server Message Block) is
- Identifying open shares on a Windows/Samba server
- Enumerating users via SMB
- Tools: enum4linux, smbclient, smbmap
- enum4linux -a [target] — full SMB enumeration

SNMP Enumeration

- WhatSNMP is and why it's exploitable
- Default community strings: public, private
- snmpwalk -v1 -c public [target] — extract SNMP data
- Gathering system info, interfaces, and running processes

NetBIOS Enumeration

- What NetBIOSis and its use in Windows networks
- nbstat -A [target] — NetBIOS name table
- Identifying workgroup, domain, and MAC address

NFS Enumeration

- WhatNFS (Network File System) is
- showmount -e [target] — list exported shares
- Mounting NFS shares and accessing files

Email Enumeration

- SMTP VRFYand EXPN commands to verify users
- Using smtp-user-enum tool
- Harvesting valid email addresses

Enumeration using Default Credentials

- Identifying services with known default logins
- Common defaults: admin/admin, root/root, admin/password
- Checking web interfaces, SSH, FTP, databases

Practical Session

- SMBenumeration on Metasploitable using enum4linux
- SNMP walk to extract system data

- DNS subdomain brute force

Module 9 System Hacking & Privilege Escalation

What is System Hacking?

- Goal: gain unauthorized access to target systems
- Stages:gainingaccess→escalatingprivileges→maintaining access → covering tracks

Password Attacks

- Dictionary attack — try words from a wordlist
- Brute force attack — try all possible combinations
- Credential stuffing — reuse leaked credentials
- Rainbow table attack — precomputed hash lookups
- Hydra — online brute force (SSH, FTP, HTTP forms)
  - hydra -l admin -P rockyou.txt ssh://[target]
- John the Ripper — offline hash cracking
  - john --wordlist=rockyou.txt hashes.txt

Windows HackingBasics

- Common Windows vulnerabilities and misconfigurations
- Extracting password hashes using Metasploit
- Pass-the-Hash attack overview
- Scheduled tasks and service vulnerabilities

Linux Hacking Basics

- Common Linux vulnerabilities and weak configs
- Exploiting world-writable files
- Cron job vulnerabilities
- Weak sudo permissions

Privilege Escalation — Windows

- Running WinPEAS to enumerate escalation vectors
- Unquoted service path exploitation
- Weak service permissions exploitation
- DLL hijacking overview
- Always-install-elevated misconfiguration

Privilege Escalation —Linux

- Running LinPEAS for automated enumeration
- SUID bit exploitation — find / -perm -4000 -type f
- Sudo misconfigurations — sudo -l
- Writable /etc/passwd exploitation
- Kernel exploit identification
- GTFOBins — bypassing restrictions with common binaries

Post Exploitation Concepts

- Maintaining access: backdoors, netcat listeners • Pivoting to other systems in the network
- Gathering credentials and sensitive files

PracticalSession

- Linux privilege escalation lab on vulnerable VM
- Password cracking with John the Ripper using rockyou.txt

- Run LinPEAS and analyze findings

Module 10 Malware, Dark Web & Security Awareness

Types of Malware

- Virus — attaches to programs, spreads when executed
- Worm — self-replicating, spreads without user action
- Trojan Horse — disguised as legitimate software
- Adware — displays unwanted advertisements
- Spyware — secretly monitors user activity
- Ransomware — encrypts files and demands ransom
- Rootkits — hides malware at OS/kernel level
- Keyloggers — records all keystrokes

Surface Web,DeepWeb& DarkWeb

- Surface Web — publicly indexed by search engines (~4%)
- Deep Web — not indexed: databases, private portals, email (~90%)
- Dark Web — requires TOR browser, intentionally hidden

HowTORWorks

- Onion routing — traffic encrypted in multiple layers
- TOR nodes: entry, relay, and exit nodes
- Anonymity and its limitations
- TOR browser installation and usage demonstration

Risks of the Dark Web

Illegal marketplaces and criminal activity
Risk of malware and phishing on .onion sites
Law enforcement monitoring

IDS,IPS,Firewalls &Honeypots

- IDS — Intrusion Detection System: alerts on suspicious activity
- IPS — Intrusion Prevention System: actively blocks threats
- Firewall — filters traffic by rules (stateful vs stateless)
- Honeypot — decoy system to detect and study attackers

Malware Analysis

- Static Analysis — examine file without executing (strings, hashes, imports)
- Dynamic Analysis — run in isolated sandbox and observe behavior

PracticalSession

- Malware awareness demonstration using sample analysis
- Analyze suspicious network traffic using Wireshark
- TOR Browser installation and demonstration

Module 11 Cryptography & DoS Attack Awareness

CryptographyFundamentals

- What is Cryptography? — securing data through encoding
- Encryption vs Hashing — key differences

Encryption Types

- Symmetric Encryption — same key for encrypt/decrypt
    - AES (Advanced Encryption Standard) — 128, 192, 256-bit
    - DES and 3DES — legacy algorithms
- Asymmetric Encryption — public and private key pair
    - RSA — widely used for key exchange and digital signatures
    - ECC — Elliptic Curve Cryptography, efficient and strong
- Digital Signatures — authenticate message origin and integrity
- SSL/TLS — secures web traffic, HTTPS handshake process

Hashing Algorithms

- MD5 — 128-bit, fast but broken for security use
- SHA-1 — 160-bit, deprecated
- SHA-256 / SHA-512 — strong, widely used
- bcrypt — password hashing with salt
- How hashes are used in password storage
- Hash cracking with John the Ripper and Hashcat overview

DoSandDDoSAttacks

- DoS — single source overwhelms a target
- DDoS — distributed, using a botnet of thousands of compromised systems
- Volume-based attacks — UDP flood, ICMP flood
- Protocol attacks — SYN flood exploiting TCP handshake
- Application layer attacks — HTTP GET/POST flood targeting web servers

Prevention&Mitigation

- Rate limiting and traffic filtering
- CDN and scrubbing centers
- Firewall rules against spoofed packets
- Anycast network diffusion

PracticalSession

- Generate and verify MD5, SHA-256 hashes using Linux (md5sum, sha256sum)
- Analyze SSL/TLS certificate details of a website
- Identify weak hashing in password files

PART 2 — VAPT & WEB APPLICATION SECURITY (Months 1.5 – 3)

Module 12 Web Application Fundamentals

What is a Web Application?

- Howweb appswork:client→server →database
- Difference between static and dynamic web applications

HTTP & HTTPS

- HTTP request structure: method, URL, headers, body
- HTTP methods: GET, POST, PUT, DELETE, PATCH, OPTIONS
- HTTP response codes: 200, 301, 302, 400, 401, 403, 404, 500
- HTTPS — TLS encryption over HTTP

Cookies & Sessions

- Whatcookiesare: key-value pairs stored in browser
- Cookie attributes: HttpOnly, Secure, SameSite, Path, Domain, Expires
- Session management — how servers track logged-in users
- Session tokens and their security implications

Authentication vs Authorization

- Authentication — verifying WHO you are (login)
- Authorization — verifying WHAT you can access (permissions)
- Common auth mechanisms: basic auth, form-based, token-based

Same Origin Policy (SOP)

- What SOP isand whybrowsers enforce it
- Origin = protocol + domain + port
- How SOP prevents cross-site attacks

APIs & Web Services

- REST API — stateless, uses HTTP methods
- SOAP — XML-based, more rigid structure
- GraphQL — query language for APIs

JWT — JSON Web Tokens

- JWT structure: header.payload.signature
- How JWT authentication works
- Common JWT vulnerabilities overview

Introduction to OWASP

- WhatOWASP is and its mission
- OWASP Top 10 overview — most critical web vulnerabilities
- OWASP resources: cheat sheets, testing guide

Practical Session

- Analyze HTTP requests and responses using Burp Suite
- Inspect and modify session cookies
- Identify authentication mechanism on test apps

Module 13 Burp Suite Professional

What is Burp Suite?

- Industry-standard web security testing tool
- Community vs Professional editions

Configuring Browser Proxy

- Settingbrowser proxyto127.0.0.1:8080
- Installing Burp CA certificate for HTTPS interception

Proxy Tab

- Intercepting HTTP/HTTPS requests and responses
- Forwarding, dropping, and editing requests
- HTTP history — reviewing all captured traffic

Repeater

- Resending individual requests with modifications
- Comparing responses across multiple requests
- Manual testing of parameters and headers

Intruder

- Automated customized attacks on parameters
- Attack types: Sniper, Battering Ram, Pitchfork, Cluster Bomb
- Payload types: simple list, numbers, character fuzz
- Brute force login forms using wordlists

Decoder

- Encode/decode: Base64, URL encoding, HTML entities, hex
- Hashing data: MD5, SHA-1, SHA-256

Comparer

- Side-by-side comparison of two requests or responses
- Identify differences in response for enumeration

Target & Site Map

- Buildinga map of the target application
- Identifying all discovered endpoints and parameters

Burp Extensions

- Installingextensions from the BApp Store
- Useful extensions: Logger++, Active Scan++, JSON Beautifier

Practical Session

- Interceptand modify login requests
- Use Repeater to test SQL injection payloads
- Use Intruder to brute force a login form in DVWA

Module 14 SQL Injection

What is SQL Injection?

- How SQLqueries work in web applications
- What happens when user input is unsanitized
- Impact: data theft, authentication bypass, database destruction

Error-Based SQL Injection

- Extracting database info from error messages
- Using ' (single quote) to trigger errors
- Identifying DBMS type from error output

Union-Based SQL Injection

- UNIONSELECT to retrieve data from other tables
- Determining number of columns: ORDER BY technique
- Identifying string columns for output
- Extracting database name, tables, columns, and data
  - ' UNION SELECT 1,database(),3 --
  - ' UNION SELECT 1,group_concat(table_name),3 FROM information_schema.tables --

Blind SQL Injection

- No visibleoutput — infer data from app behavior
- Boolean-based: TRUE vs FALSE conditions change response
  - ' AND 1=1 -- (TRUE) vs ' AND 1=2 -- (FALSE)
- Inferring data character by character

Time-Based Blind SQL Injection

- Using SLEEP() toconfirm injection without visible output
  - ' AND SLEEP(5) -- (page delays = vulnerable)
- Extracting data based on response time differences

Authentication Bypass

- Bypassing login with classic payload: ' OR '1'='1
- Bypassing with comment injection: admin' --

Database Enumeration

- Extracting: database name, version, user, tables, columns
- Dumping table contents

SQLMap — Automated SQL Injection

- sqlmap -u [URL] — detectand exploit SQLi
- --dbs — list all databases
- -D [db] --tables — list tables in a database
- -D [db] -T [table] --dump — dump table contents
- --forms — automatically test HTML forms

Prevention Techniques

- Preparedstatements / parameterized queries
- Input validation and whitelist filtering
- Web Application Firewalls (WAF)
- Principle of least privilege for DB users

Practical Session

- SQLinjection lab on DVWA (all security levels)
- Union-based extraction on vulnerable web app
- Automated exploitation with SQLMap

Module 15 Cross-Site Scripting (XSS)

What is XSS?

- Injecting malicious JavaScript into web pages
- Executed in victim's browser — not on the server
- Impact: session hijacking, credential theft, defacement

Reflected XSS

- Payload included in URL, reflected immediately in response
- Victim must click a crafted link
- Example: search parameter returned in page content
  - <script>alert('XSS')</script>
- Testing in URL parameters, search boxes, error messages

Stored XSS

- Payload stored in the database, served to all visitors
- Most dangerous type — no user interaction needed
- Common locations: comment fields, usernames, profile bios
- Impact: persistent attack on every visitor

DOM-Based XSS

- Payload manipulates the DOM directly in the browser
- No server-side interaction — entirely client-side
- Vulnerable sinks: innerHTML, document.write, eval
- Identifying via browser developer tools

SessionHijackingvia XSS

- Stealing session cookies using document.cookie
- Sending cookies to attacker's server
- Why HttpOnly flag prevents this

XSS Payload Examples

- <script>alert(1)</script> — basic test
- <img src=x onerror=alert(1)> — event handler bypass
- <svg onload=alert(1)> — SVG-based payload
- Bypassing basic filters with encoding and case variation

PreventionTechniques

- Output encoding — encode special characters before rendering
- Content Security Policy (CSP) headers
- HttpOnly and Secure cookie flags
- Input validation and sanitization

PracticalSession

- Reflected and Stored XSS on DVWA
- Cookie theft demonstration in lab
- Test CSP effectiveness on sample app

Module 16 OWASP Top 10 & Web Vulnerabilities

CSRF —Cross-Site Request Forgery

- Tricking authenticated user into making unintended requests
- Attacker crafts malicious page that submits a form to target site
- Prevention: CSRF tokens, SameSite cookie attribute

SSRF — Server-Side Request Forgery

- Attacker makes server send requests to internal/external resources
- Accessing internal services: http://127.0.0.1, http://169.254.169.254
- AWS metadata endpoint abuse
- Prevention: whitelist allowed destinations, block internal IPs

IDOR —Insecure Direct Object Reference

- Accessing objects by changing parameter values (IDs)
- Example: /user/profile?id=101 → change to id=102
- Horizontal (same role) vs Vertical (higher role) privilege escalation
- Prevention: authorization checks on every request

Clickjacking

- Embedding target site in invisible iframe to trick user clicks
- Prevention: X-Frame-Options header, CSP frame-ancestors

Host HeaderInjection

- Manipulating the HTTP Host header to poison cache or reset password links
- Password reset poisoning attack
- Prevention: whitelist allowed Host values

File Upload Vulnerabilities

- Uploading malicious files (webshell, PHP backdoor)
- Bypass techniques: double extension (shell.php.jpg), MIME spoofing
- Accessing uploaded webshell to execute commands
- Prevention: validate file type server-side, use isolated storage

SecurityMisconfiguration

- Default credentials left unchanged
- Unnecessary features/services enabled
- Verbose error messages exposing stack traces
- Missing security headers

Insecure Deserialization

- What serialization/deserialization is
- Tampering with serialized data to execute code or escalate privilege
- Prevention: avoid deserializing untrusted data

CORSMisconfiguration

- What CORS is and how browsers enforce it
- Wildcard (*) origin allowing any site to read responses
- Reflecting arbitrary origins without validation
- Prevention: whitelist specific trusted origins

Broken Authentication

- Weak passwords and no account lockout
- Predictable session tokens
- Exposed credentials in URLs or logs

Broken Access Control

- Accessing admin features without proper role
- Forced browsing to unauthorized URLs
- Prevention: enforce access control server-side on every request

Practical Session

- OWASP Juice Shop: IDOR, CSRF, and SSRF challenges
- File upload webshell exercise on DVWA
- Test CORS misconfiguration using Burp Suite

Module 17 API Security Testing

What is an API?

- Application Programming Interface — allows apps to communicate
- REST API fundamentals: endpoints, methods, status codes
- JSON request and response structure

REST APIs

- Stateless communication over HTTP
- CRUD operations: GET (read), POST (create), PUT (update), DELETE (delete)
- API versioning and endpoint patterns

GraphQL Basics

- Query language for APIs — request exactly what you need
- Introspection queries to enumerate schema
- GraphQL-specific vulnerabilities: deep queries, introspection enabled

Swagger / OpenAPIDocumentation

- Reading and understanding API documentation
- Finding hidden or undocumented endpoints
- Using swagger-ui.html for API testing

APIAuthenticationMethods

- API keys — sent in headers or query parameters
- Basic Auth — base64 encoded username:password
- Bearer Token — JWT or OAuth token in Authorization header

JWT Vulnerabilities

- Algorithm confusion attack — changing alg to 'none'
- Weak secret key — brute forcing the signing key
- Sensitive data exposed in unencrypted payload
- Tools: jwt.io, hashcat for JWT cracking

Rate Limiting

- What rate limiting is and why it matters
- Testing for missing rate limits: OTP brute force, login flood
- Bypassing rate limits: IP rotation, header manipulation

APIAuthorizationIssues

- BOLA (Broken Object Level Authorization) — same as IDOR for APIs
- BFLA (Broken Function Level Authorization) — accessing admin endpoints
- Mass assignment — sending extra fields to elevate privileges

PracticalSession

- Test REST API endpoints using Postman
- Decode and tamper with JWT tokens
- Test for BOLA and mass assignment on lab API
- Use Burp Suite to intercept and modify API calls

Module 18 VAPT Methodology & Report Writing

What is VAPT?

- Vulnerability Assessment — identify and classify vulnerabilities
- Penetration Testing — actively exploit vulnerabilities to assess risk
- Difference between VA and PT

VAPT Lifecycle

- Phase 1: Scoping & Planning — define targets, rules of engagement
- Phase 2: Reconnaissance — gather information about target
- Phase 3: Vulnerability Assessment — scan and identify weaknesses
- Phase 4: Exploitation — attempt to exploit discovered vulnerabilities
- Phase 5: Post-Exploitation — assess impact and access depth
- Phase 6: Reporting — document findings, impact, and remediation

Scoping & Rules of Engagement

- Defining in-scope and out-of-scope systems
- Getting written authorization before testing
- Testing windows and emergency contacts
- Agreed-upon deliverables and report format

Severity Classification

- Critical — immediate exploitation, severe impact
- High — significant risk, relatively easy to exploit
- Medium — moderate risk with some conditions
- Low — minimal risk, hard to exploit
- Informational — best practice recommendations

CVSS — Common Vulnerability Scoring System

- Base score: Attack Vector,Complexity, Privileges, User Interaction
- Impact metrics: Confidentiality, Integrity, Availability
- Score range: 0.0 – 10.0
- Using NVD CVSS calculator

False Positives

- Whatfalsepositives are and how to confirm findings
- Manual verification of scanner results

Professional Report Writing

- Executive Summary — high-level findings for management
- Technical Summary — detailed findings for IT team
- Vulnerability details: title, description, CVSS score, evidence
- Proof of Concept (PoC) — screenshots, request/response
- Remediation recommendations — specific fix for each finding
- Risk rating table and overall security posture

Practical Session

- Create afullprofessional VAPT report on lab findings
- Write remediation recommendations for identified vulnerabilities

Module 19 Final Capstone Project

Project Overview

- Students perform a complete, end-to-end penetration test
- Apply all skills learned across both parts of the course

Phase1: Reconnaissance

- OSINT on the target using theHarvester, WHOIS, Shodan
- DNS enumeration and subdomain discovery
- Technology fingerprinting

Phase2: Scanning&Enumeration

- Nmap full port and service scan
- SMB, SNMP, and service enumeration
- Identifying potential entry points

Phase3: VulnerabilityIdentification

- Manual and automated vulnerability discovery
- Testing for OWASP Top 10 vulnerabilities
- Documenting findings with screenshots

Phase4: Exploitation

- Exploiting identified vulnerabilities on lab targets
- Web application exploitation: SQLi, XSS, file upload
- System access and privilege escalation

Phase5: APITesting

- Testing API endpoints for authorization issues
- JWT tampering and rate limit testing

Phase6: Reporting

- Compiling all findings into a professional VAPT report
- Executive summary and technical section
- Risk rating, PoC evidence, and remediation for each finding

Labs Used

- DVWA — web vulnerability practice
- OWASP Juice Shop — modern web app challenges • Metasploitable 2 — network and system exploitation • VulnHub Machines — realistic CTF-style targets

FinalAssessment

- Practical Examination on lab environment
- VAPT Report Submission — graded on completeness and quality
- Viva / Q&A session with evaluators

KIIS

15th Floor 1502 Segment Spaces,
Manjeera Trinity Corporate, KPHB,
Kukatpally, Hyderabad- 500072

Email us - **kiis@kleap.in**

Call Us - **+91 9398514034**