TR-04

Cybersecurity & Ethical Hacking

COURSE BRIEF

SKILLS & TOOLS YOU WILL MASTER

Kali Linux | Burp Suite | Nmap + Wireshark | Metasploit | nuclei + httpx | OWASP Top 10

10 Reasons This Works.

Why students choose Saarathi - and how each card helps you ship a real career.

01

10 Max / Batch

Not 50. Not 30. Ten. Every student seen, heard, and unblocked - every class.

02

Sunday Open Classroom

Every Sunday the classroom is open. Come in, pair-program, get unstuck, or just focus.

03

Weekly Detailed Syllabus

Full syllabus from day one. Every week: what to read, what to build, what the outcome is.

04

Revision Every 5th Day

Every fifth class is a revision session. Nothing piles up or slips through.

05

Week 1: Mindset

The first week is about getting into the right headspace before touching a line of code.

06

3 Ways of Learning

Self-learning + collaboration + mentorship. All three built into every course.

07

1:1 Mock Interview

Real mock interview with your mentor in the final week. Actual hiring prep, not fluff.

08

CV + LinkedIn

Resume writing and LinkedIn optimisation. Leave with a profile that gets replies.

09

Real Projects

Every course ends with a deployed, publicly accessible project you can show employers.

10

Gate Assessment

Aptitude test before Day 1. Maps your strengths, weaknesses, learning style, and pace - then builds your personal plan.

saarathiacademy.com.np/cybersecurity-course-in-nepal +977-9744442469 · +977-9761095364

Cybersecurity & Ethical Hacking

Why This Course

This course trains the junior web/API VAPT analyst and Tier-1 SOC analyst profile employers actually hire (VAPT means vulnerability assessment and penetration testing, finding and proving security holes with written permission; a SOC is a security operations centre, the team watching a company's security alerts): someone who can scope an engagement, run recon (structured information gathering on a target), find vulnerabilities, validate them, write a CVSS-scored report (CVSS is the industry's 0 to 10 severity score) and hand remediation advice (clear fix instructions) a developer can act on. Real employable security work is quiet and disciplined, understanding scope, observing evidence, validating findings and documenting impact, not dramatic hacking demos. Pay scales with your skills, portfolio, and whether you work locally or for international/remote clients. We do not promise a salary.

This 12-week program is paced the way a beginner course should be: the first 4 weeks (a full third of the course) are genuine groundwork, the security mindset, how networks and TCP/IP work, disciplined recon, and Linux and host fundamentals, before any exploitation. From that base it shifts weight heavily onto VAPT and junior penetration testing, keeping a full 4 weeks of web + API testing, with a certification runway of Security+ → eJPT → OSCP (three industry certifications, from foundation to advanced) and a portfolio-grade VAPT capstone report by Week 11. eJPT prep is baked in as graded checkpoints in Weeks 3, 5, 9 and 12, and a Bash/Python scripting strand runs through Weeks 3, 5 and 8 so you automate the repetitive half of testing. At graduation the honest target roles are Junior Web/API VAPT Analyst and Tier-1 SOC Analyst; full network penetration testing is the later runway, built on eJPT plus lab hours. Depth stays on offensive web and API testing because that is where the junior market is most open. Windows security and **Active Directory** are taught as one consolidated block in Week 10 rather than scattered as footnotes, because enterprise targets are overwhelmingly Windows environments, and OSINT sits early because recon quality decides whether an engagement succeeds.

Three things run the length of the course instead of sitting in one week. From Day 1 every command, screenshot and finding goes into one searchable engagement vault in **Obsidian** or **Notion** (mandatory, not a suggestion), and the capstone report is written straight out of it. In Week 6, the exact midpoint, every student sits a private 1:1 review with their mentor and leaves with a written action list that gets closed out in Week 12. And before you attack a web application in Week 6, you learn how one is actually built, front to back.

AI tools are used the way working pen-test teams use them: ChatGPT for report drafting, recon idea generation and CVSS explanation; AI-assisted synthesis during reconnaissance to speed up attack-surface mapping. The goal is responsible augmentation, not automation of judgment.

Who Is This Course For

- **Absolute beginners** with digital literacy who want a real junior pen-tester path
- **IT, networking or support professionals** specialising into security
- **CS/IT students** targeting junior pen-test or SOC analyst roles
- **Developers** who want to build security-aware applications by learning to break them
- **Career changers** attracted to VAPT consulting, bug bounty and freelance pen-testing
- **Learners who want a safer foundation** before moving into cloud security and DevSecOps

Course Snapshot

PARAMETER	DETAILS
Course Code	TR-04
Duration	12 Weeks (3 Months)
Schedule	Monday to Friday, 2 Hours/Day
Total Hours	120 Hours
Batch Size	Maximum 10 Students
Course Fee	NPR 35,500
Level	Beginner → Junior Web/API VAPT Analyst
Outcome	Junior Web/API VAPT Analyst / Tier-1 SOC Analyst

Prerequisites

- Comfortable using a computer daily, no security background required
- A laptop with 8GB+ RAM and 50GB+ free disk space for the lab VMs
- A free **Obsidian** or **Notion** account for your engagement vault, set up in the Week 1 class
- Week 1 starts at how the internet works, TCP/IP and the security mindset, so nothing is assumed
- Saarathi Gate Assessment before Day 1, diagnostic, no pass or fail

Nothing to install beforehand. VirtualBox and the Kali Linux VM go on your machine in the first class.

Optional, only if you want a head start: the free TryHackMe Pre-Security path at tryhackme.com. Good use of a spare weekend, not expected of you.

Self-study runs on the free tiers of TryHackMe and HackTheBox. Paid subscriptions are optional, are not provided by Saarathi, and are never needed to pass the course or ship the capstone.

Assessments at a glance

Two exams bracket the course. The **Saarathi Gate Assessment** runs before Day 1 as a diagnostic with no pass or fail; it maps your learning style and pace. The **Saarathi Certification Exam** is sat at the end of the course, 2 hours and around 30 questions specific to this course in mixed formats (single choice, multi-select, true/false, numeric, ordering, matching, fill-in-the-blank, short or long answer, code where relevant, and short case studies). Pass mark is 60 percent and a pass is mandatory for the Saarathi Certificate of Completion. The exam is proctored: it runs in fullscreen, switching tab or app and pasting are logged, and three strikes end it. You can answer the questions in any order and come back to any of them, but every question needs an answer before you submit. Recommendation letters are discretionary, mentor-issued only to learners who pass the Certification Exam, and never guaranteed.

Your Learning Week

DAY	ACTIVITY
Mon–Fri	2-hour live class session (hands-on, project-driven) with mentor
Mon–Fri	2-hour self-study at home (assignments, practice, debugging)
Sunday	Open classroom for focused build time or rest
Day 5 (every week)	Within the 2-hour session, the last 1 hour is a built-in revision block plus reporting clinic for consolidation and doubt-clearing

> Real improvement happens when you build, break, debug and re-ship between classes. The classroom gives structure; the hours between classes give depth.

Week-by-Week Curriculum

Phase 1 · Security Mindset, Networking, Recon & Host Foundations (Weeks 1–4, 40 Hours)

WEEK	FOCUS AREA	TOPICS
Week 1	Security Mindset, Authorization & Lab Discipline	- What cybersecurity actually is: the **CIA triad** (confidentiality, integrity, availability: data secret, unaltered, reachable) - How the internet works and **TCP/IP** basics: the browser (client) sends requests to servers and reads responses, over the shared rule-set computers talk on, through common ports (numbered doors, like 80 and 443) and protocols (the language behind each door) - Scope, authorization, ethical testing rules: the written permission that defines exactly what you may test, and the lines you never cross - **Wireshark** intro (records network traffic) and your first packet capture (a saved traffic recording) - Engagement vault and note discipline: one **Obsidian** vault (local markdown files, works offline, yours forever) or **Notion** workspace set up in class on Day 1, with a note template per target, every command, screenshot and finding timestamped and written down as you work; mandatory from Day 1, checked at the Week 6 1:1, and the capstone report is written out of it - Social engineering awareness: attacks that trick people instead of machines: phishing (fake emails), pretexting (cover stories), vishing (scam calls) and physical security tricks (tailgating: slipping through secure doors, badge cloning: copying access cards); why human vectors appear in pen-test scope - AI for pen-testing: **ChatGPT** for recon idea generation and attack-surface brainstorming (every door an attacker could try)
Week 2	Network Services & Traffic Analysis	- How networks move and control traffic: routing (the path packets take), NAT (many devices, one public address), segmentation (isolated network zones) and firewall logic (what gets blocked) - Keeping connections safe: **TLS** (the browser-padlock encryption), certificates (digital ID cards for sites), **SSH** (encrypted remote login for admins) and secure service exposure (only open what must be open) - **Wireshark** deeper: streams (one reassembled conversation),

WEEK	FOCUS AREA	TOPICS
		filters (only the traffic you care about) and protocol analysis - **ARP**/**DNS** manipulation awareness and **MITM** concepts: attackers lie to the address-lookup layers (ARP and DNS) to sit in the middle of traffic (man-in-the-middle) - Defensive baselining and anomaly detection: record normal traffic so anything unusual stands out
Week 3	Nmap, Enumeration & Secure Architecture	- **Nmap**, the standard port scanner: scan types, service detection (what software answers each open port) and safer defaults that avoid knocking over live systems - Enumeration workflow: methodically listing everything a target exposes, banner grabbing (reading announced version text), with **httpx** and **subfinder** from the **ProjectDiscovery** toolkit for probing which hosts are live and expanding in-scope subdomains - Correlating **Nmap** and **Wireshark** findings: cross-check the scanner against the raw traffic - Secure architecture: least privilege (only the access each account needs) and defense-in-depth (layered protections) - Attack-surface mapping and reduction: list every exposed service, recommend what to close - **OSINT** techniques (open-source intelligence, profiling a target from public information) and passive recon discipline: Google dorks (search operators for exposed files and login pages), **Shodan** (internet-exposed devices and services), **theHarvester** (email and subdomain harvesting), **Maltego** awareness (link-analysis graphs), building an **OSINT** target profile before active scanning, and never touching a target until permission is signed - Scripting strand: wrap the **Nmap** workflow in a **Bash** script (a program of terminal commands) that parses output into a service-inventory CSV (spreadsheet-friendly) - Graded **eJPT** prep quiz 1 (Assessment Methodologies, the information-gathering domain)
Week 4	Linux & Host Fundamentals	- Linux file system, users, groups, permissions: how a host is laid out (`/etc` , `/var` , `/home`) and how Linux decides who may read, change or run each file - Processes, services, startup, packages: what is running, what launches automatically, what is installed and where it came from - The terminal and shell basics: navigating, piping and reading

WEEK	FOCUS AREA	TOPICS
		`man` pages in **Bash**, the daily fluency every later lab assumes - **SSH** and firewall basics: lock remote login to keys instead of passwords, and set the host firewall to block everything inbound by default with `ufw` - **Sudo** (the run-as-admin command), shell history (your typed commands) and ops-sec habits (never leak your own secrets) - Guided privilege-escalation lab to root (limited account to full control): see how a SUID binary (runs with its owner's power, abused via **GTFOBins**, the public trick catalogue), a sudo misconfiguration and a cron/PATH hijack each hand over the box, then write the fix, so you understand the host before you attack one - Windows and **Active Directory** get a full block of their own in Week 10, so this week stays entirely on Linux

Phase 2 · Host Review & Web VAPT Core (Weeks 5–7, 30 Hours)

WEEK	FOCUS AREA	TOPICS
Week 5	Logging, Auditing & Host Review	- Reading what a Linux machine remembers: **journald** (the system journal), auth logs (who logged in when) and system messages - **auditd** basics: the Linux watchdog recording who touched sensitive files - Cron and systemd timers: job schedulers attackers abuse to keep access, and the persistence indicators that give them away - File integrity: take a `sha256sum` baseline before, compare after, and catch a swapped system binary (**AIDE** and tripwire are the packaged version of the same idea) - Log-based investigation: reconstruct what happened on a host from logs alone - Scripting strand: **Bash** log-triage pipeline (grep/awk/sort) over auth.log to pull failed logins and sudo events - Remediation notes from host evidence: turn log findings into fixes a sysadmin can apply - **SIEM** and the Tier-1 SOC on-ramp: the same logs centralised across every host (**Splunk**, **Microsoft Sentinel**, **ELK**), one guided search run on a sample dataset, and the triage question a SOC analyst answers all day: is this alert real, and who gets it; taught here, next to the logs it is built

WEEK	FOCUS AREA	TOPICS
		from - Graded **eJPT** prep quiz 2 (Host and Network Auditing)
Week 6	Web App Anatomy, Burp & Injection Fundamentals	- How a web application is built, before you attack one: **HTML**, **CSS** and **JavaScript** in the browser, a backend in **PHP**, **Python**, **Node.js** or **Java** with the frameworks around them (**WordPress**, **Laravel**, **Django**, **Express**, **Spring**), a database behind that (**MySQL**, **PostgreSQL**, **MongoDB**), and **Apache** or **Nginx** plus hosting in front; one request traced through every layer, and the bug class each layer owns - How the web talks: **HTTP** methods (request verbs like GET and POST), headers, cookies and sessions (how a site remembers you), plus reading a target from the client side in browser devtools and fingerprinting its stack - **Burp Suite**, the web tester's workbench: proxy (intercept browser traffic), repeater (resend edited requests), intruder (automated payload testing), decoder - **OWASP** methodology: testing to the industry playbook, mapping findings to **OWASP Top 10:2025** categories (the finalised January 2026 list of common web vulnerability classes) - **SQL injection** (sneaking database commands through inputs), under **A05 Injection**, validated hands-on on **PortSwigger Labs** (free, legal targets) - **XSS** (cross-site scripting, your script in someone else's page): reflected, stored and DOM variants, also under A05 Injection - Evidence capture and findings: exact proof for every issue you find, filed in your vault; **ChatGPT** may draft the first wording and explain a **CVSS** score, you rewrite the impact line by hand - **Mid-course 1:1 mentor review**: a private half hour at the exact midpoint, your vault and lab work read line by line, an honest read on where you stand, and two or three things to fix in the second half, written down and closed out in Week 12; mandatory, not graded
Week 7	Auth, Access Control & Misconfiguration	- Attacking how sites prove who you are: authentication flows, password reset and **MFA** (multi-factor authentication, like an OTP code) - **IDOR** (insecure direct object reference, swapping an ID to reach someone else's data), privilege boundaries and

WEEK	FOCUS AREA	TOPICS
		access control (who is allowed to do what); maps to **OWASP A01 Broken Access Control**, still #1 in 2025 - Session management, **CSRF** (cross-site request forgery, tricking a logged-in browser into acting for the attacker) and token handling - Security headers (protective settings sent to the browser), unsafe uploads and default misconfigs (**OWASP** A02 Security Misconfiguration, #2 in the 2025 Top 10) - New 2025 categories on a target: A03 Software Supply Chain Failures (known-vulnerable components, dependency provenance) and A10 Mishandling of Exceptional Conditions (errors that fail open or leak), identification and reporting only - **XXE** (XML External Entity, an XML parser tricked into reading server files, under A05 Injection) and **insecure deserialization** (untrusted saved objects rebuilt into running code, under A08 Software or Data Integrity Failures): recognise and report each once on a lab endpoint, since junior job posts still name them - **Capstone target selection + scope planning**: pick your final-project target and agree in writing exactly what you will test

Phase 3 · API VAPT & Exploitation Toolchain (Weeks 8–9, 20 Hours)

WEEK	FOCUS AREA	TOPICS
Week 8	API VAPT & Modern Web Testing	- **REST API** testing (the behind-the-scenes data doors apps run on): **JWT** review (signed tokens proving login) and rate limits (caps on endpoint hits) - API enumeration and **OpenAPI**/**Swagger** analysis: reading an API's machine-readable docs to list every endpoint worth testing - **GraphQL** awareness (query-style API standard), **SSRF** (server-side request forgery, making the server fetch internal URLs, now reported under OWASP A01 in the 2025 Top 10) and **WebSocket** basics (always-open two-way connections) - Scripting strand: short **Python** requests script (the standard web-call library) for endpoint probing and **JWT** replay - Assessment methodology and proof capture: reproducer steps (exact bug-triggering commands), impact statement, **CVSS** justification

WEEK	FOCUS AREA	TOPICS
		- Drafting findings with impact and remediation - AI for pen-testing: **ChatGPT** for report drafting, executive summaries (the one-page manager version), plain-language remediation advice developers act on
Week 9	Network Service Attacks & the Exploitation Toolchain	- Network service attacks first, on the services you enumerated back in Week 3: **SMB** null sessions (Windows file sharing), **FTP** anonymous access, **SSH** user enumeration and weak-credential spray, each proven on a scoped lab with the misconfiguration written up - Service playbooks and **HackTricks** (hacktricks.wiki): the same commands come back on every box, so you build your own per-service playbook in your vault, cross-checked against the reference working testers keep open, understanding each command before pasting it - **Metasploit** framework, the standard exploitation toolkit: modules (ready-made attack code), payloads (what runs on the target) and sessions (your live connections), aimed at a service you already proved weak - **SQLMap** automation for confirmed SQLi: extract data once you have proven the injection by hand - **hashcat** for password attacks: recovering passwords from hashes (their scrambled stored form) - Engagement cleanup and hygiene: kill sessions, remove what you uploaded, restore configs, delete accounts you made, and leave a written cleanup log - Wireless awareness, plus scoped exploit validation (proving an exploit works while staying inside the agreed scope) - Graded **eJPT** prep quiz 3 (Host and Network Penetration Testing)

Phase 4 · Scanning, Reporting & Career (Weeks 10–12, 30 Hours)

WEEK	FOCUS AREA	TOPICS
Week 10	Windows, Active Directory & Vulnerability Triage	- The Windows security model, in one block instead of scattered footnotes: local accounts and groups, NTFS permissions, UAC, Windows Firewall and Defender reviewed on a Windows 10/11 VM - The Windows Security log read properly: event IDs 4624 (logon), 4625 (failed logon), 4672 (special-privilege logon) and 4688 (process creation), plus Task Scheduler and run-keys checked for persistence

WEEK	FOCUS AREA	TOPICS
		- **Active Directory** structure: domain, OUs, domain controllers and Group Policy Objects, and why one AD misconfiguration decides how far an attacker gets across thousands of machines - **Active Directory** attack paths: **pass-the-hash** (Windows credential reuse), **Kerberoasting** (service-account hash extraction) and **AS-REP roasting** (crackable hashes from accounts without Kerberos pre-auth), each walked end to end with the misconfiguration and the fix; live AD exploitation is TR-11 territory - Naming and scoring vulnerabilities: **CVE** (public flaw ID), **CWE** (weakness catalogue), **CVSS** scoring, with **ChatGPT** used to cross-check a vector and translate findings for non-technical stakeholders - **nuclei** and **httpx** (**ProjectDiscovery**), the week's primary scanner: httpx probes which hosts are live and what they run, nuclei runs community templates by severity and tag, you read one template line by line and write one simple one yourself - **OpenVAS**/**Greenbone** as the broad infrastructure sweep next to nuclei's targeted templates, its report read critically - False-positive triage and multi-source validation before anything reaches a report; older scanners like **Nikto** are one noisy input, never a verdict - Attack-path reasoning and fix-priority planning: chain small findings into one serious break-in, then say what the client fixes first and why
Week 11	Capstone Assessment & Professional Reporting	- Capstone kickoff: scope review and evidence plan agreed before testing - Execute scoped VAPT on **PortSwigger Labs** / **Juice Shop** / trainer target - Report writing: executive summary, findings, **CVSS**, remediation - Peer review and trainer feedback - The report assembled from the vault you have kept since Week 1: evidence pulled by link, not hunted for in a screenshots folder - Adjacent domains you must scope honestly instead of bluffing: mobile pen-test (static versus dynamic analysis, **OWASP MASVS**, when to hand off), cloud security on **AWS** (**IAM** misconfigs, exposed S3 and EC2, where cloud meets VAPT scope, full depth in TR-03) and incident response (clean evidence handoff so a defender can pick it up)

WEEK	FOCUS AREA	TOPICS
Week 12	Portfolio, Certifications & Career Launch	- Bug bounty workflow: **HackerOne** and **Bugcrowd** programmes, scope rules, severity tiers and payout tables - Responsible disclosure: coordinated timelines, no-programme cases, legal safe harbours - Writing good bounty reports: reproduction steps, impact, **CVSS** justification, proof-of-concept; what gets triaged vs rejected - Public writeups (responsibly redacted) - Resume, **LinkedIn**, **GitHub** for security roles - Mock security analyst + pen-tester interviews, plus a closing 1:1 with your mentor that works through the action list from the Week 6 midpoint review, item by item - Certification roadmap (**Security+** → **eJPT** → **OSCP**; Security+ is currently SY0-701, due to retire around late 2026, so check CompTIA for the current version; OSCP now also issues as OSCP+, which expires after three years); book and sit **eJPT** (from **INE Security**) during the course (USD 200–250, 48-hour live lab exam, 35 practical questions, 70 percent pass; the 2026-refreshed eJPT added more web-app testing, recon and responsible AI use, all already drilled here), the best practical match for VAPT - Graded **eJPT** domain mock across all four exam domains (assessment methodologies, host and network auditing, host and network penetration testing, web-app penetration testing) - Career planning

Skills You Will Gain

CATEGORY	TECHNOLOGIES
Recon & Traffic	Nmap, Wireshark, httpx, subfinder, Google Dorks, Shodan, theHarvester
Web VAPT	Burp Suite, PortSwigger Labs, Juice Shop
Exploitation	Metasploit, SQLMap, hashcat, SMB/FTP/SSH service attacks
Hardening & Audit	Linux hardening, Windows hardening, auditd, file-integrity baselines (sha256sum, AIDE)
Windows & AD	Windows Security log (4624/4625/4672/4688), NTFS, UAC, Defender, Active Directory structure and attack paths

CATEGORY	TECHNOLOGIES
Scanners	nuclei, httpx (ProjectDiscovery), OpenVAS (Greenbone Community Edition), Nikto
Notes & Reporting	Obsidian or Notion engagement vault, HackTricks service playbooks
Scripting	Bash recon + log-triage scripts, Python (requests) for API probing
Labs	Kali Linux, VirtualBox, TryHackMe and HackTheBox free tiers
Frameworks	OWASP Top 10:2025, OWASP API Top 10 (2023), CVSS v3.1, MITRE ATT&CK
Bug Bounty	HackerOne, Bugcrowd platform rules, responsible disclosure
AI Assistants	ChatGPT for report drafting, recon synthesis and CVSS explanation

Topic Depth and Awareness

FOCUS AREA	LEARNING FOCUS
Networking, recon (Nmap, Wireshark, httpx)	Depth
OSINT (Google dorks, Shodan, theHarvester)	Depth
How a web application is built (stack anatomy)	Depth
Web VAPT (Burp, OWASP Top 10:2025)	Depth
XXE and insecure deserialization (recognise + report)	Awareness
API VAPT (JWT, REST, GraphQL awareness)	Depth
Exploitation toolchain (Metasploit, SQLMap, hashcat)	Depth
Network service attacks (SMB, FTP, SSH)	Depth
Templated scanning (nuclei, httpx)	Depth
CVSS scoring and report writing	Depth
Note discipline (Obsidian or Notion vault)	Depth, mandatory

FOCUS AREA	LEARNING FOCUS
Linux hardening and host review	Depth
Bash and Python security scripting (working scripts)	Depth
Bug bounty workflow and responsible disclosure	Depth
Social engineering awareness	Awareness
Windows security model, hardening and event logs	Depth
Active Directory structure and common misconfigs	Depth
Active Directory attack paths (pass-the-hash, Kerberoasting, AS-REP roasting)	Walkthrough, no live AD exploitation
SIEM (Splunk, Sentinel, ELK)	Awareness, one guided search
Wireless and mobile pen-test	Awareness
Cloud security (AWS)	Awareness (covered in depth in TR-03)
Incident response and forensics	Awareness

Project Pool

Each student picks **one** capstone in Week 7. Pool rotates per batch:

#	PROJECT	FOCUS
1	PortSwigger Labs / Juice Shop full VAPT report	Web + API VAPT, OWASP Top 10:2025
2	Network service enumeration + scoped service-misconfig report	Nmap, Metasploit, reporting
3	OSINT target profile + attack-surface report	Google dorks, Shodan, theHarvester, passive recon
4	Bug bounty dry-run on a public programme (read-only scope review)	HackerOne/Bugcrowd scope analysis, finding write-up practice

#	PROJECT	FOCUS
5	CTF-style portfolio with 5–8 solved rooms	HackTheBox and TryHackMe free tiers
6	Trainer-scoped assessment	Whatever the brief requires

Career Paths & Trajectory

ROLE	TIMELINE	FOCUS
Junior Web/API VAPT Analyst / Tier-1 SOC Analyst	0–2 yrs	Web + API VAPT support, alert triage, reporting
Penetration Tester / Security Analyst	2–4 yrs	Own engagements, network pen-testing, red-team sub-tasks
Senior Pen-Tester / AppSec Engineer	4–7 yrs	Architecture review, custom tooling
Security Lead / Consultant	7+ yrs	Engagement leadership, client advisory

FAQ

What does Cybersecurity & Ethical Hacking cost and how do I pay?

NPR 35,500. Pay by cash at our Old Baneshwor campus, bank transfer, or Fonepay. Email info@saarathiacademy.com.np for the payment plan.

Do I need prior experience?

Prerequisites are listed above. The Saarathi Gate Assessment runs before Day 1 to map your starting point, diagnostic, not pass-or-fail. Below prerequisites? We suggest a short foundation track first.

How long is the course?

12 Weeks (3 Months). Every student sits a private 1:1 review with their mentor in Week 6, the exact midpoint, and leaves with a written action list. A capstone project follows the guided phase, with a 1:1 mock interview and the close-out of that action list in the final week.

What tools will I use?

Kali Linux, Burp Suite, Metasploit, Wireshark, Nmap, SQLMap, nuclei and httpx. All of it goes into one engagement vault in **Obsidian** or **Notion**, which is mandatory from Day 1. Every phase ships a portfolio artefact: a recon report, a hardening audit, a web + API VAPT report and the capstone engagement report. No toy labs.

What roles does this prepare me for?

At graduation the realistic targets are Junior Web/API VAPT Analyst and Tier-1 SOC Analyst; full network pen-testing roles come later, built on eJPT plus lab hours. We do not promise jobs. We build your portfolio, run mock interviews, polish your CV and LinkedIn, and refer you into our hiring network.

How does the batch and mentorship work?

Maximum 10 students per batch, every student gets direct mentor time and project review. Hybrid by default, in-person at Old Baneshwor with online recordings and office hours. Sunday Open Classroom is free for enrolled students: pair-program, get unstuck, or just focus.

Enrollment

- **Next Batch:** see the live course page
- **Location:** Old Baneshwor Chowk, Kathmandu, Nepal
- **Mode:** Online + Offline
- **Contact:** 9744442469, 9761095364
- Limited to 10 seats per batch

Cybersecurity & Ethical Hacking — full details, fees and next batch dates: saarathiacademy.com.np/cybersecurity-course-in-nepal

Version 1.1 · Updated August 21, 2026