Project Proposal
Open-Source Offensive and Security Testing Tools

November 6, 2025

Group Members

- Xin Huang (Leader) — Student IDs: 202283910036 / 20109967
- Yi Gu — Student IDs: 202283910033 / 20109964
- ShunYi Yang — Student IDs: 202283910016 / 20109947

Topic

Discuss, evaluate and demonstrate 5 tools that are used by hackers and security testers (open source).

Rationale for Topic Selection

Open-source offensive and defensive tooling offers a realistic and ethical way to understand how attacks occur and how they are detected or prevented. The topic directly supports course objectives by developing both the attacker mindset (how weaknesses are discovered and exploited) and the defender mindset (how telemetry and controls are applied). Using widely adopted tools ensures reproducibility, low cost, and strong documentation communities. The selected tools span key phases of the kill chain: reconnaissance and service discovery (e.g., Nmap), credential attacks (e.g., John the Ripper), wireless assessment (e.g., Aircrack-ng), exploitation frameworks (e.g., Metasploit Framework), and packet capture/analysis (e.g., Wireshark or tcpdump). Comparing capabilities, usability, and risk profiles across tools highlights practical trade-offs and responsible use. The project will emphasize safe lab environments, permission, and compliance, translating technical results into actionable security recommendations and showcasing how defenders can harden configurations, monitor signals, and reduce attack surface.

Feasibility Assessment

The scope is well-bounded: five mature open-source tools with clear documentation and active communities. A feasible lab can be built using Linux VMs and intentionally vulnerable targets (e.g., DVWA, Metasploitable2, or local test services) isolated from production

networks. Required resources are modest: one host for tools, one host for targets, standard networking, and non-privileged accounts with controlled escalation where necessary. Evaluation criteria will include installation complexity, feature breadth, learning curve, output quality, and defensive countermeasures. Deliverables will comprise a comparative matrix, repeatable demonstration scripts, evidence (screenshots, logs), and security recommendations. Risks (service disruption, data exposure) are mitigated by strict scoping, network isolation, and written authorization. The team can execute within typical course timelines: setup (1–2 days), tool exploration (1 –2 days), demonstrations and documentation (1–2 days). Overall, the plan is achievable, safe, and aligned with learning outcomes.