70 CYBERSECURITY PROJECTS

A PRACTICAL ROADMAP
FROM ZERO TO ADVANCED



A HANDS-ON LEARNING WORKBOOK

Preview Contents

1 Welcome 1
1.1 What the Full Workbook Covers 1
2 A Look Inside: Sample Projects 1
2.1 Build a Personal Cyber Lab 1
2.2 Wireshark Protocol Explorer 1
2.3 Secure Authentication Demo 1
2.4 Threat Modeling with STRIDE 2
2.5 Incident Response Playbook 2
2.6 AI-Assisted Log Triage Benchmark 2
3 How Projects Are Structured 2
4 Responsible Practice Comes First 2
4.1 Ready to Start? 2

1 Welcome

Build practical cybersecurity skills, one project at a time.

This preview introduces **70 Cybersecurity Projects: A Practical Roadmap from Zero to Advanced**, a hands-on workbook designed to help learners move beyond passive study. The complete guide contains 70 progressively structured projects. Each project gives you a learning goal, a practical build brief, and a tangible outcome to document.

Use the roadmap at your own pace. Start with core computing and safe lab habits, then progress through network visibility, web security, vulnerability assessment, detection and incident response, cloud security, malware triage, and applied AI.

1.1 What the Full Workbook Covers

- **Stage 1 — Foundations & Security Habits:** build a safe lab, strengthen operating-system and networking basics, and practice documentation.
- **Stage 2 — Network Visibility & Defense:** inspect authorized traffic, develop monitoring skills, and plan segmentation.
- **Stage 3 — Web Security & Secure Development:** learn secure design and testing with local training applications.
- **Stage 4 — Vulnerability Assessment & Ethical Testing:** practice scope, triage, validation, and remediation in sanctioned environments.
- **Stage 5 — Detection Engineering & Incident Response:** work with logs, alerts, timelines, and response playbooks.
- **Stage 6 — Cloud, Identity & DevSecOps:** explore access control, infrastructure checks, container security, and secure delivery.
- **Stage 7 — Malware Analysis, AI & Capstone:** practice safe file triage, detection rules, AI guardrails, and portfolio capstones.

2 A Look Inside: Sample Projects

2.1 Build a Personal Cyber Lab

Goal: Create an isolated environment for safe experimentation.

Build: Set up a Linux virtual machine on a host-only network, take a clean snapshot, and test how to restore it.

Deliverable: A topology sketch, setup checklist, and verified restore procedure.

2.2 Wireshark Protocol Explorer

Goal: Learn to interpret common network conversations.

Build: Review a supplied sample capture or traffic created inside your own lab. Identify DNS lookups, TCP handshakes, and visible TLS metadata.

Deliverable: An annotated packet walkthrough with useful display filters and privacy-safe evidence.

2.3 Secure Authentication Demo

Goal: Understand practical protections in a simple login flow.

Build: Create a local demonstration using a reputable password-hashing library, rate limits, and safe error messages.
Deliverable: A working demo, a brief threat model, and tests for expected behavior.

2.4 Threat Modeling with STRIDE

Goal: Identify design-level risks before implementation.
Build: Map a small fictional service, mark trust boundaries, and consider spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege.
Deliverable: A data-flow diagram and a prioritized list of mitigations.

2.5 Incident Response Playbook

Goal: Prepare a clear response to a fictional account-compromise scenario.
Build: Outline containment, credential reset, evidence preservation, communications, and recovery steps.
Deliverable: A practical checklist with decision points and contact placeholders.

2.6 AI-Assisted Log Triage Benchmark

Goal: Evaluate where AI can help—and where human review remains essential.
Build: Compare an AI-assisted summary of synthetic security logs with a simple rule-based baseline. Remove identifying data and verify every conclusion.
Deliverable: A benchmark with accuracy checks, failure examples, and human-review requirements.

3 How Projects Are Structured

Each project is designed to produce evidence of learning, not just tool output. Define your scope, work in the appropriate lab, test assumptions, record what you observed, and explain what remains uncertain. A strong project write-up describes the environment, the method, the result, and a practical next step.
A suggested learning loop is simple: define a question, set safety limits, complete the exercise, test expected cases, document the outcome, and reflect on one improvement. You can complete the projects in order or select a project that matches your current goals, while building foundational knowledge first.

4 Responsible Practice Comes First

Only test systems you own or have explicit written permission to assess. Keep experiments inside isolated labs, sanctioned training platforms, or a clearly documented scope. Do not scan, exploit, disrupt, intercept, or access data on third-party systems without authorization. Use dummy credentials and synthetic data, protect private logs, and stop if an exercise affects a real service or leaves its approved scope.
This workbook is educational. It is not legal advice, a certification, or a guarantee of employment. Always follow applicable laws, organizational policies, and platform rules.

4.1 Ready to Start?

The complete workbook includes **70 guided projects across seven progressive stages**, with practical deliverables to help you build a repeatable learning process and a portfolio of safe, well-documented work.
Learn by building. Practice responsibly. Explain what you discover.