

Homework

Three tasks to deepen your exploit-dev workflow — all defensive framing.

All homework stays in *defensive validation* mode: detection, patch verification, and disclosure. Never ship a weaponized exploit outside an authorized lab.

Task 1 — Detection Signature

Objective

Write a detection that would catch exploitation on the wire.

Instructions

1. From the CVE and crash dump, identify the exact byte pattern that indicates an exploitation attempt.
2. Write a Suricata or Snort rule that matches it.
3. Write a brief rationale: what is the false-positive rate against benign traffic?

Deliverable

One rule file plus rationale.

Task 2 — Patch Verification Harness

Objective

Extend the --check mode PoC into a fleet scanner.

Instructions

1. Wrap the Exercise 4 PoC so it reads a list of target hosts from a file.
2. Run it in a safe "probe only" mode against each host and output a CSV: host, vulnerable Y/N, version detected.
3. Add rate limiting and logging.

Deliverable

`verify_fleet.py`.

Task 3 — Disclosure Draft

Objective

Practice responsible disclosure writing.

Instructions

1. Write a coordinated-disclosure email to the (fictional) vendor.
2. Include: the finding, impact, reproduction steps, suggested fix, timeline, and your PGP contact.
3. Tone should be professional, not triumphant.

Deliverable

One email draft in markdown.