YUMMY — HackTheBox

Author: Ichwanul Ammar Alfajri

Role: Offensive Web Exploitation (HTB)

Target: YUMMY (HackTheBox) — public write-up (sanitized)

Duration: 2 Hari

Status: Public, sanitized (High)

Tags: web-exploitation, jwt, sqli, rce, priv-esc, HTB

Contents

Executive Summary 3

TL;DR Teknikal (Highlights) 3

Tools & Environment 3

Reconnaissance 3

Phase 1 — Path Traversal (Discovery → Proof) 3

Phase 2 — JWT Manipulation (Privilege Escalation concept) 3

Phase 3 — SQL Injection (Search feature) 4

Phase 4 — RCE & Backdoor (chained exploit, sanitized) 4

Privilege Escalation & Root (concept & mitigation) 4

Vulnerability Summary 4

Skills Demonstrated 5

Ethical Note & Disclosure 5

Appendix — Artefak (stored privately) 5

Checklist sebelum publish (Notion) 5

Executive Summary

Singkat: Saya mengidentifikasi beberapa kerentanan tingkat tinggi pada lab YUMMY, termasuk path traversal, JWT manipulation (lab-only), SQL injection pada fitur search, RCE via chained automation, dan privilege escalation melalui cron/writable scripts. Bukti lengkap disimpan privat; di sini hanya ringkasan dan mitigasi.

TL;DR Teknikal (Highlights)

- Recon: nmap, service fingerprint.

- Temuan utama: Path Traversal → JWT forgery (lab) → SQLi → RCE chaining → PrivEsc via cron.

- Outcome (public): Methodology & mitigations documented. (Full artefak = private)

Tools & Environment

Tools used: nmap, curl, Burp Suite, pyjwt/cryptography (analysis), sqlmap (detection), linpeas, netcat.

Note: Jangan jalankan PoC eksploit pada sistem non-lab.

Reconnaissance

Command (sanitized): nmap -sC -sV TARGET_IP

Summary: SSH (22), HTTP (80). Web app mempunyai endpoint download & search.

Placeholder — Evidence: 01-nmap.png — Nmap output (sanitized)

Phase 1 — Path Traversal (Discovery → Proof)

What: File-download endpoint menerima path yang memungkinkan traversal.

Concept (sanitized): Manipulasi parameter `export` mengembalikan file internal (contoh: /etc/passwd, /etc/crontab, /data/scripts/*.sh, /var/www/backupapp.zip). Isi file disimpan privat; di sini hanya ringkasan.

Mitigasi singkat: Whitelist path; normalisasi input; ID-to-path mapping; authorization checks.

Placeholder — Evidence: 02-pathtraversal-request.png; 03-sanitized_file_list.txt

Phase 2 — JWT Manipulation (Privilege Escalation concept)

What: signature.py di konfigurasi aplikasi memungkinkan pembuatan token RS-based (lab artifact). Dalam lab kami mensimulasikan pembuatan token admin untuk akses admin dashboard.

Sanitized explanation: Tidak ada kunci/skrip PoC di publik. Hanya: menemukan script pembuatan token → analisis flow → simulated forgery in lab.

Mitigasi: Key management, rotate keys, enforce `kid` validation, short lived tokens, revocation.

Placeholder — Evidence: 04-admin_access_screenshot.png

Phase 3 — SQL Injection (Search feature)

What: Fitur search rentan; boolean- & error-based techniques mengindikasikan SQLi (MySQL). Auth-token rotation menghambat otomatisasi; konfirmasi manual digunakan.

Mitigasi: Parameterized queries, input validation, rate-limiting, monitoring.

Placeholder — Evidence: 05-sqli_notes.txt

Phase 4 — RCE & Backdoor (chained exploit, sanitized)

What (concept): Monitoring script (dbmonitor.sh) men-trigger fixer scripts jika dbstatus.json bilang 'down'. Dengan memanipulasi kondisi (lab-only) dan file yang dapat diakses, script tersebut dapat dipicu untuk menjalankan perintah yang akhirnya menghasilkan shell (lab environment only).

Mitigasi: No external content execution without integrity checks; principle of least privilege; signed scripts.

Placeholder — Evidence: 06-netcat_revshell.png

Privilege Escalation & Root (concept & mitigation)

What: Writable cron-invoked scripts dan folder VCS (.hg) ditemukan; bila dikelola secara tidak aman, dapat dimanfaatkan untuk eskalasi (lab-only).

Mitigasi: Harden file permissions; protect VCS metadata; audit cron jobs; immutable deployments.

Placeholder — Evidence: 07-crontab_excerpt_sanitized.txt

Vulnerability Summary

Vulnerability | Impact | Severity | Recommendation

Path Traversal | Sensitive file disclosure | High | Whitelist & normalize paths

JWT Misuse | Privilege escalation | High | Secure key storage; token validation

SQL Injection | Data exfiltration | High | Prepared statements

RCE (chained) | Remote command execution | Critical | Least privilege; do not execute external content

Insecure file permissions | Privilege escalation | High | Fix permissions; audit cron jobs

Skills Demonstrated

- Web exploitation (path traversal, SQLi)

- JWT analysis & manipulation (lab-controlled)

- RCE chaining & reverse-shell handling

- Privilege escalation via system configuration & cron analysis

- Tooling: nmap, Burp, sqlmap (detection), linpeas, pyjwt, netcat

Ethical Note & Disclosure

Write-up ini adalah ringkasan lab dan telah disanitasi. Jangan lakukan exploit terhadap sistem non-consenting. Untuk artefak penuh/PoC (private) hubungi penulis.

Appendix — Artefak (stored privately)

private/01_nmap_full.txt

private/raw_downloaded_files.zip

private/backupapp_full.zip

private/signature_py_original.py

private/lab_poc_scripts/

private/linpeas_output.txt

private/netcat_session_log.txt

private/README_PRIVATE.md

Checklist sebelum publish (Notion)

- [ ] Upload images/logs matching the filenames above (to private storage)

- [ ] Replace placeholders in this page with Notion-embedded images/files

- [ ] Add short author bio & contact

- [ ] Link to public GitHub README (this repo)

- [ ] Mark 'Availability: Open to internship'