

# Obada Daghlas

## Entry-Level Cybersecurity Engineer

[obada.daghlas.dev@gmail.com](mailto:obada.daghlas.dev@gmail.com) | [GitHub](#) | [LinkedIn](#) | +970594408329

## Summary

---

**Entry-level Cybersecurity Engineer** with a **software engineering background** and hands-on experience in **web/API security testing, Linux administration, and secure-by-design backend engineering**. Practiced vulnerability discovery and exploitation (OWASP Top 10) through labs and CTF-style scenarios, with strong documentation and remediation reporting skills. Experienced building production-grade backend services with TypeScript, Docker, PostgreSQL, Redis, and CI/CD pipelines, and translating complex workflows into technical specs with validation gates, access boundaries, and audit-friendly controls.

## Education

---

**BSc Computer Systems Engineering, Arab American University, Jenin- Palestine**

**10/2020 – 02/2025**

- **GPA:** 3.02.
- **Key modules:** Fundamentals of Object-Oriented Programming, Database Systems, Software Engineering, Web Development, Artificial Intelligence, and Networking Fundamentals.
- **Project:** Holdwise - Posture Detection and Correction – smart mobile app for monitoring and correcting human posture using computer vision and machine learning techniques. Focused on real-time feedback and ergonomic improvements.

## Skills

---

### Penetration Testing & Security Analysis

- **Web & API Testing:** OWASP Top 10, authentication/session handling, access control, input validation
- **Vulnerability Discovery & Exploitation:** XSS, SQLi, SSRF, IDOR, Command Injection (lab/sandbox)
- **Tooling:** Burp Suite, Postman, Nmap, Gobuster, ffuf, sqlmap
- Threat Modeling, Attack Surface Mapping, and Security Checklists
- **Reporting:** clear writeups, impact analysis, reproduction steps, remediation recommendations

### Technical & Development Knowledge

- **Languages:** Python, JavaScript/Node.js, TypeScript, C++, SQL
- **Scripting:** Bash, automation basics
- **Web:** React, Next.js, REST APIs, JWT / Token-based Auth
- **Databases:** PostgreSQL, Oracle, Firebase, Redis
- **DevOps/Systems:** Linux fundamentals, Docker, Docker Compose, Git/GitHub
- **Cloud:** Google Cloud fundamentals (IAM concepts, networking/firewall basics)

## Experience

---

### The Consulting Lab | Software Engineer

**07/2025 – 02/2026**

- Delivered 6 end-to-end workflow definitions for Compass for Law (Eminent Domain), translating domain operations into executable steps (intake → research → negotiation → filing) with clear state transitions and ownership.
- Designed a hybrid orchestration approach: internal service manages workflow state + rules, while external automation executes integrations—enabling retries, idempotency expectations, and failure recovery patterns.
- Contributed to secure-by-design implementation planning by defining access boundaries, validation checkpoints, and human-in-the-loop approvals for sensitive legal workflow stages.
- Standardized requirements into testable acceptance criteria and edge-case lists, improving feature completeness and decreasing “undefined behavior” during implementation.

### Titan Shield | Cybersecurity Specialist Internship, Ramallah – Palestine

**07/2025 – 10/2025**

- Completed hands-on labs & **CTF-style scenarios:** web app vulns (**SQLi, XSS**), network enumeration (nmap), and credential attacks (Hydra/John); documented findings and mitigations.
- Practiced cloud security basics in sandboxes (GCP/AWS): IAM least-privilege roles, S3/Cloud Storage access policies, and security groups/firewall rules.
- Built a foundation in system hardening and secure configuration of Linux environments, focusing on SSH, permissions, and patching workflows.

### ASAL Technologies | React Developer Intern, Jenin – Palestine

**10/2024 – 02/2025**

- Built accessible, responsive UI with Next.js/React; collaborated with backend engineers via Git PRs and code reviews.
- Improved **application performance by 30%** through code optimizations and efficient state management (Redux Toolkit).
- Integrated **TypeScript for scalable and maintainable development**.
- Developed and deployed UI components with **Material-UI (MUI)**, enhancing design consistency.

## Certifications

---

**Titan Shield Certified Web Assessor (TSCWA)** – Issued Oct 2025

**Boot.dev Backend Development Path** — 10 courses completed, including CI/CD with GitHub Actions, Docker, HTTP Servers in TypeScript, SQL, and Linux (*Feb–Apr 2026*)

## Projects

---

### Pipeline Orchestrator | Webhook-Driven Task Processing Platform

- Built a TypeScript/Node.js backend service that receives incoming webhooks, queues them asynchronously via BullMQ + Redis, and fans out processed payloads to subscriber endpoints with exponential backoff and delivery tracking.

- Implemented API key authentication with Argon2id hashing and HMAC-SHA256 webhook signature verification (replay-attack prevention via 5-minute timestamp window), following security patterns used by Stripe and GitHub.
- Designed multi-tenant team workspaces with per-user pipeline ownership, role-based access boundaries, and enumeration-safe 404 responses for cross-user resource requests.
- Integrated PostgreSQL with Drizzle ORM for type-safe, migration-driven schema management and a full audit event log covering auth, key lifecycle, and team operations.
- Containerized the full stack (API, background worker, PostgreSQL, Redis) with Docker Compose and set up a CI/CD pipeline via GitHub Actions with automated testing (Vitest + Playwright e2e).

### Chirpy API | Microblogging REST API

- Built a RESTful HTTP server in TypeScript/Express with full user authentication: JWT access tokens (1-hour expiry), 60-day refresh tokens stored in PostgreSQL, and token revocation endpoints.
- Implemented Argon2id password hashing, ownership-enforced authorization (users can only delete their own content), and webhook security using API key authentication.
- Applied Drizzle ORM for type-safe, migration-driven schema management; structured error handling via custom error classes and a centralized async wrapper middleware.

### Pakistan Tax Calculator | Full-Stack Web Application

- Containerized Node.js/Express backend and React frontend using Docker and Docker Compose for streamlined local development
- Designed **PostgreSQL** database schema for **tax data storage**
- Implemented business logic for tax calculations based on multiple parameters
- Followed modular architecture principles for maintainable code

### BlogAggregator | CLI RSS Feed Aggregator

- Developed a multi-user command-line RSS aggregator in TypeScript backed by PostgreSQL and Drizzle ORM, with a configurable polling loop and graceful SIGINT shutdown.
- Implemented a command registry pattern with auth-gated middleware, many-to-many feed follow relationships, and duplicate-URL deduplication using ON CONFLICT DO NOTHING.