05 / DEFENSIVE

Tornike Kalandadze

Security-Minded Software Engineer

Application security · Defensive tooling · Threat modeling

tornikekalandadze.work@gmail.com
+995 599 258 808
Tbilisi, Georgia · GMT+4
github.com/stariik
linkedin.com/in/tornike-kalandadze-997701365

Top 4%	91	98
TRYHACKME	ROOMS COMPLETED	NETLENS CORE TESTS

01 PROFILE

Security-minded software engineer with hands-on application-security training and a portfolio of deliberately defensive systems. I build around distrust: bounded parsing, sandboxed execution, default-deny authorization, secure evidence handling, audit logs, explainable findings, and documented limitations. TryHackMe: top 4%, 91 rooms, 15 badges, four completed learning paths.

02 EXPERIENCE

Independent Security Engineer

Defensive engineering & authorized labs

2022–Present
Application security, forensics, detection engineering

- Build defensive systems around hostile-input handling, sandboxing, authentication, authorization, auditability, and explainable findings.
- Completed 91 TryHackMe rooms, earned 15 badges, reached the top 4%, and completed four security learning paths in 2023.
- Keep offensive practice inside authorized labs; public projects are deliberately defensive and document their limitations.

Winner · Solo Builder

ShieldMesh · VibeCoding From 0

2026
Superteam Georgia × CyreneAI × Solana · $800 prize

- Built an offline-first mobile threat-intelligence system end to end: Android app, local threat detection, device-to-device alert relay, Solana devnet program, and public dashboard.
- Shipped a downloadable APK and working demo as the hackathon's championship winner.

04 OPERATING STRENGTHS

- Treat uploads, parser output, paths, and generated content as attacker-influenced.
- Keep security findings explainable and human-reviewed—not automated verdicts.
- Document incomplete controls and scope limits instead of hiding them.

05 TECHNICAL SKILLS

APPLICATION SECURITY
Threat modeling · OWASP · AuthN/AuthZ · RLS/RBAC · Secure uploads · Audit logging

DEFENSIVE TOOLING
Zeek · Suricata · Sigma · MITRE ATT&CK · CVSS v3.1 · PCAP analysis

ENGINEERING
Python · FastAPI · PostgreSQL · Docker sandboxing · JWT · Argon2/bcrypt

PRACTICE
TryHackMe · Web assessment labs · Linux · Nmap · Burp Suite · Secure reporting

06 EDUCATION & CREDENTIALS

TRYHACKME · TOP 4%
91 rooms · 15 badges · profile: tryhackme.com/p/z1x0

FOUR LEARNING PATHS · 2023
Web Fundamentals, Complete Beginner, Pre Security, and Introduction to Cyber Security.

IT STEP ACADEMY · 2016–2018
Software-development studies with course certificates in PHP, HTML/CSS, JavaScript, databases, and C#.

07 LANGUAGES

Georgian — native · English — C1 · Russian — B1/B2

03 SELECTED ENGINEERING WORK

NetLens

Defensive prototype

github.com/stariik/NetLens

Passive, offline PCAP-forensics backend that validates hostile captures and normalizes Zeek/Suricata evidence.

- Docker analysis jail: no network, dropped capabilities, read-only root, non-root user, no-new-privileges, and CPU/memory/PID/time ceilings.
- 98 core tests; ten explainable heuristics present rationale, calculations, benign explanations, and validation steps.

Python · FastAPI · Zeek · Suricata · Docker · PostgreSQL

SentinelForge

Defensive prototype

github.com/stariik/SentinelForge

Versioned Sigma-rule workspace with safe import, explainable scoring, ATT&CK mapping, RBAC, and audit history.

- Bounds YAML and ZIP inputs against traversal, symlinks, nesting, expanded size, entry count, and compression-ratio abuse.
- Immutable versions and diffs make restoration non-destructive; production refuses to start with the example secret.

FastAPI · pySigma · SQLAlchemy · MITRE ATT&CK · JWT

PentestFlow

Authorized-assessment prototype

github.com/stariik/pentestflow

Engagement workspace that converts imported scanner output into triage, evidence, attack paths, findings, and redacted reports.

- Parses Nmap, Nuclei, Burp, ZAP, and Nikto output strictly as data; it contains no scanner, payload delivery, or command execution.
- Central engagement authorization, audit logging, sanitized Markdown, computed CVSS v3.1, and automatic secret redaction.

Next.js 15 · PostgreSQL · Prisma · Auth.js · Playwright

ShieldMesh

Solo hackathon winner

github.com/stariik/shieldmesh

Offline-first threat-intelligence system spanning Android, local detection, device-to-device relay, and Solana verification.

- Built solo for VibeCoding From 0 and won the championship plus an $800 prize.
- Delivered a downloadable APK, Room-backed offline state, devnet Anchor program, and public Next.js dashboard.

Kotlin · Jetpack Compose · Room · Rust · Anchor · Solana

Worldwide remote · employment, contract, or freelance

Portfolio evidence and source code available through the links above.