Ozan Malcı

Istanbul, Türkiye | ozanmalci@protonmail.com | 0537 422 96 46 | ozan2003.github.io
linkedin.com/in/ozan-dev

Education

Sakarya University, BSc in Computer Engineering – Sakarya, Türkiye Sept 2021 – July 2026

- Selected Coursework
  - Introduction to Cryptology – classical and modern crypto primitives, symmetric and public-key systems
  - Network Programming – socket APIs, concurrency, client-server systems
  - System Administration – server management, virtualization, network services

Projects

SDN-Based Automated Threat Detection and Mitigation System – Capstone Project

Designed and implemented an SDN-based intrusion response system that integrates **Suricata** IDS with a custom **Ryu** controller to automatically enforce mitigation rules on a simulated enterprise WAN.

- Built a **Python**-based Suricata-to-Ryu alert pipeline that parses EVE JSON events and dynamically installs drop or rate-limit flow rules on an **Open vSwitch** backbone switch in response to detected attacks.
- Designed a multi-segment enterprise WAN topology in **Mininet** with traffic mirroring for live IDS inspection, and automated environment setup and attack simulation workflows with **Bash**.
- Measured end-to-end detection-to-mitigation latency across attack scenarios, quantifying **OpenFlow** rule-installation round-trip overhead.

P2P File Exchange – github.com/ozan2003/p2p_file_exchange

Designed and implemented a secure peer-to-peer file transfer desktop application for local networks using **C#** and **Avalonia UI**.

- Implemented UDP-based peer discovery with Ed25519-signed announcements to mitigate impersonation attacks.
- Built end-to-end encrypted file transfers over TCP using X25519 key exchange and ChaCha20-Poly1305 authenticated encryption.
- Implemented trust-on-first-use identity verification backed by a persistent **SQLite** trust database.

RFC Reader – github.com/ozan2003/rfc_reader

Developed a **Rust** terminal application for browsing IETF RFCs using a TUI built with Ratatui, optimized for keyboard-driven navigation.

- Implemented an HTTP fetch and persistent on-disk caching layer to support offline mode, reduce repeated network calls, and improve document load latency.
- Built full-text search over the document with in-app highlighting/navigation to accelerate lookup of protocol keywords and sections.

Skills

Programming Languages: Python, Rust, C#, Bash

Tools: GNU Make, pytest, Git, uv

Network Simulation: GNS3, Cisco Packet Tracer

RDBMS: SQLite

Languages: Turkish, English (YDS: 87.5)