

Walter Marshall

Fullstack, mobile and security engineer

elatchash@gmail.com · Jacksonville,
FL · github.com/ElatDev · www.linkedin.com/in/waltermarshalldev · waltermarshall.dev

I build and run Dealer Recon Systems, a dealer management platform on iOS, Android and web. About 50,000 lines of Dart, all of it mine: the app, the backend, the security rules. I do the billing and the support calls too. Before this I worked Android security at Kik, where I killed a photo-spoofing exploit without the people using it ever finding out it was gone. All of this happens around a full-time finance job. That's why the commits run at night.

EXPERIENCE

Dealer Recon Systems

Mar 2026 - present

Founder · Sole engineer · Jacksonville, FL · Remote

- Designed, built and shipped a dealer management platform covering reconditioning, inventory, sales and CRM, service, parts and accounting — roughly 50,000 lines of Dart across 746 files, serving iOS, Android and web from one Flutter codebase. Built in six months alongside full-time employment. Web is live at dealerreconsystems.com; mobile is in App Store and Google Play review.
- Wrote the entire backend: 37 Cloud Functions on Node.js and 900+ lines of Firestore security rules that fail closed, so a dealership without an active account can read nothing. Role-based access over ~40 permissions with per-employee overrides, two-factor login and logout.
- Modeled the shop floor as a seven-stage pipeline with per-stage day targets: overdue vehicles surface on their own, and a unit cannot leave Photos until it has enough shots to list.
- Made it work where the signal does not: local caching, queued photo uploads, VIN decoding, barcode and document scanning, e-signatures, and an audit log that restores any earlier version of a record.
- Ran everything else solo — infrastructure, networking, customer support, legal and payroll.

U.S. Department of Education

Present

Finance and customer support · Remote

- Resolve finance questions on federal student aid accounts end to end — balances, disbursements, payment history and corrections — working from regulation and written policy, not a script.
- Fully remote against handle-time and quality-assurance targets, handling personally identifiable financial information under federal privacy rules.

Kik

Until Jun 2019

Software Engineer · Contract · Remote

- Closed a catfishing exploit in the Android app: modified builds were sending gallery photos flagged as live camera shots, which is the signal recipients trusted to mean a photo had just been taken.
- Designed a per-version integrity hash checked at login and shipped it as a silent failure: modified clients still signed in, but their photos arrived as gallery photos while their own UI showed camera shots, so the exploit looked like it worked and its authors got no signal about what had tripped.

- Designed the UI and UX for Kik X, a full redesign of the app.

PROJECTS

Hindsight

Open source · TypeScript, Electron, Stockfish

- Free, offline chess game review — a local alternative to the review Chess.com charges for. Imports PGN, runs Stockfish in up to four parallel processes, grades every move by centipawn loss, scores accuracy on the same curves Lichess uses, and explains mistakes by detecting the actual pin, fork or hanging piece on the board.
- Released v0.1.0 with installers for Windows, macOS and Linux, built by a GitHub Actions release workflow; CI runs all 581 tests on all three operating systems. 136 commits, MIT licensed, with Stockfish's GPL license and source shipped alongside it.

Mogadishu

Reverse engineering · C, Python, Ghidra

- Rebuilt a working model of a 2003 game engine — movement, collision, blast damage, mission scripting — from compiled x86 with no source, symbols or documentation. 467 commits, ~210,000 lines, 826 tests.
- Every claim is labeled as read from the disassembly, derived, or a guess (444/150/40), and cross-checked against the running game: predicted blast knockback of 7.69 m/s against 7.7 measured. Wrote a Direct3D 8 proxy overlay in C that the game loads itself, with plugin hot reload in about a second.

Parsers and binary tools

Open source · github.com/ElatDev

- paeth (Rust): a PNG decoder written from the spec. Decodes all 162 valid PngSuite images pixel-for-pixel identically to an independent decoder and rejects all 14 corrupt ones. 151 tests.
- loupe (C): a read-only PE and ELF inspector with every read bounds-checked. Parses all 4,264 binaries in System32 under AddressSanitizer, UBSan and LeakSanitizer with no crashes, memory errors or leaks.
- unspool (Python): a pcapng parser and protocol decoder, checked frame by frame against tshark across all 562 public Wireshark sample captures, 1,290,282 packets.
- vinspect (Go): VIN validation and offline decoding, verified against 6,516 real VINs from NHTSA crash-test records.

SKILLS

Languages	Dart, TypeScript, JavaScript, Python, Rust, Go, C, C#, Java, x86/x64 assembly, SQL
Mobile & web	Flutter, iOS, Android, React, Electron, three.js, Vite, HTML, CSS, Responsive design, Accessibility
Backend & cloud	Node.js, Firebase, Firestore, Cloud Functions, Firebase Auth, REST APIs, NoSQL, SQLite, Serverless, Twilio
Security	Application security, Reverse engineering, Ghidra, Capstone, Authentication, Role-based access control, Two-factor authentication, Static analysis
Practices & tools	Git, GitHub Actions, CI/CD, Automated testing, Code review, Technical documentation, Win32