

secure code review

AI-generated code, reviewed like it matters. Threat modeling, dependency and supply-chain checks, and hardening before anything reaches production — from a practice that does this for a living.

// scope

what we review

ai-generated code

Code written with Claude, Copilot, or any other assistant — reviewed for the failure modes AI actually produces.

dependencies & supply chain

What your code pulls in, what that pulls in, and whether any of it should worry you.

architecture & access

Threat modeling, authentication and authorization, secrets handling, and the blast radius when something fails.

// deliverables

what you get

- Findings ranked by severity, each with a concrete fix — not a wall of raw scanner output.
- Fixes applied by us or fix guidance your team executes — your choice.
- A re-review that confirms the fixes actually landed.
- A threat model your team can keep updating after we're gone.
- The same scrutiny verrou applies to client production systems.

// approach

how it runs

phase 1 — scope

agree the target

The codebase, the deployment, and what an attacker would actually want from it.

phase 2 — review

read it like an attacker

Manual review backed by tooling — threat model, code, dependencies, configuration.

phase 3 — harden

fix and confirm

Findings become fixes, fixes get re-reviewed — a report for customers and auditors.

// fit

a good fit if

- AI is writing a meaningful share of your code and review hasn't kept pace with it.
- You're about to ship something customers or auditors will ask hard questions about.
- You inherited a codebase and nobody can say what it actually exposes.
- You want findings with fixes attached — not a wall of raw scanner output.

// honesty

probably not, if

- × You want a rubber stamp — our report says what we found, or it says nothing.
- × Nothing is headed to production and there's nothing to protect yet.
- × You need a compliance certificate — we review code; certification bodies certify.

// engagement

how we engage

fixed scope

A fixed-scope proposal with a fixed price — no hourly meter running.

weekly builds

Real builds you can click through every week — no long silence, then a reveal.

you own it

Code, docs, and tests land in a repository your team controls. No lock-in.

reviewed before ship

Every build passes the same review verrou applies to client systems.