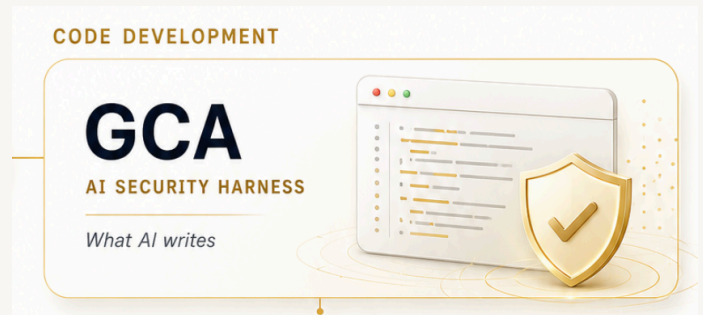
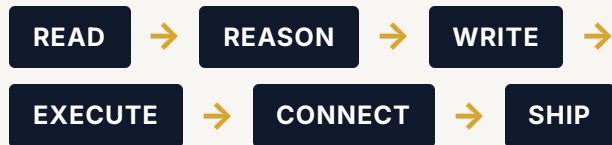


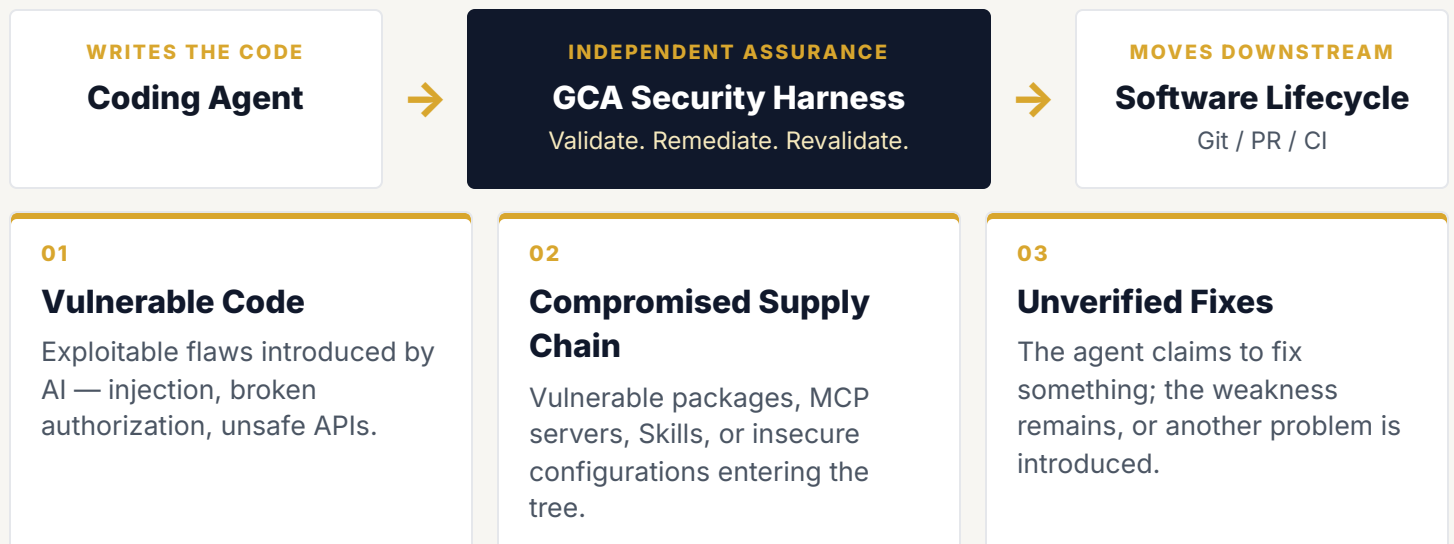


Why GCA Exists

AI coding agents can read, reason, write, execute, connect and ship. Their output needs independent assurance.



AI GENERATED IT. Who verified it?



SECURITY SHOULDN'T BE A PREMIUM FEATURE

GCA gives every developer a local security baseline across code, dependencies, secrets, AI/MCP, APIs and the software supply chain.

Start free. Upgrade when you need team governance.



WORKS ACROSS THE AI CODING STACK

Claude Code

Cursor

Windsurf

Copilot / VS Code

OpenAI Codex

JetBrains

Gadriel.ai

GCA: AI Development & Security Harness

Enterprise knowledge and security controls surrounding every developer and AI coding agent.

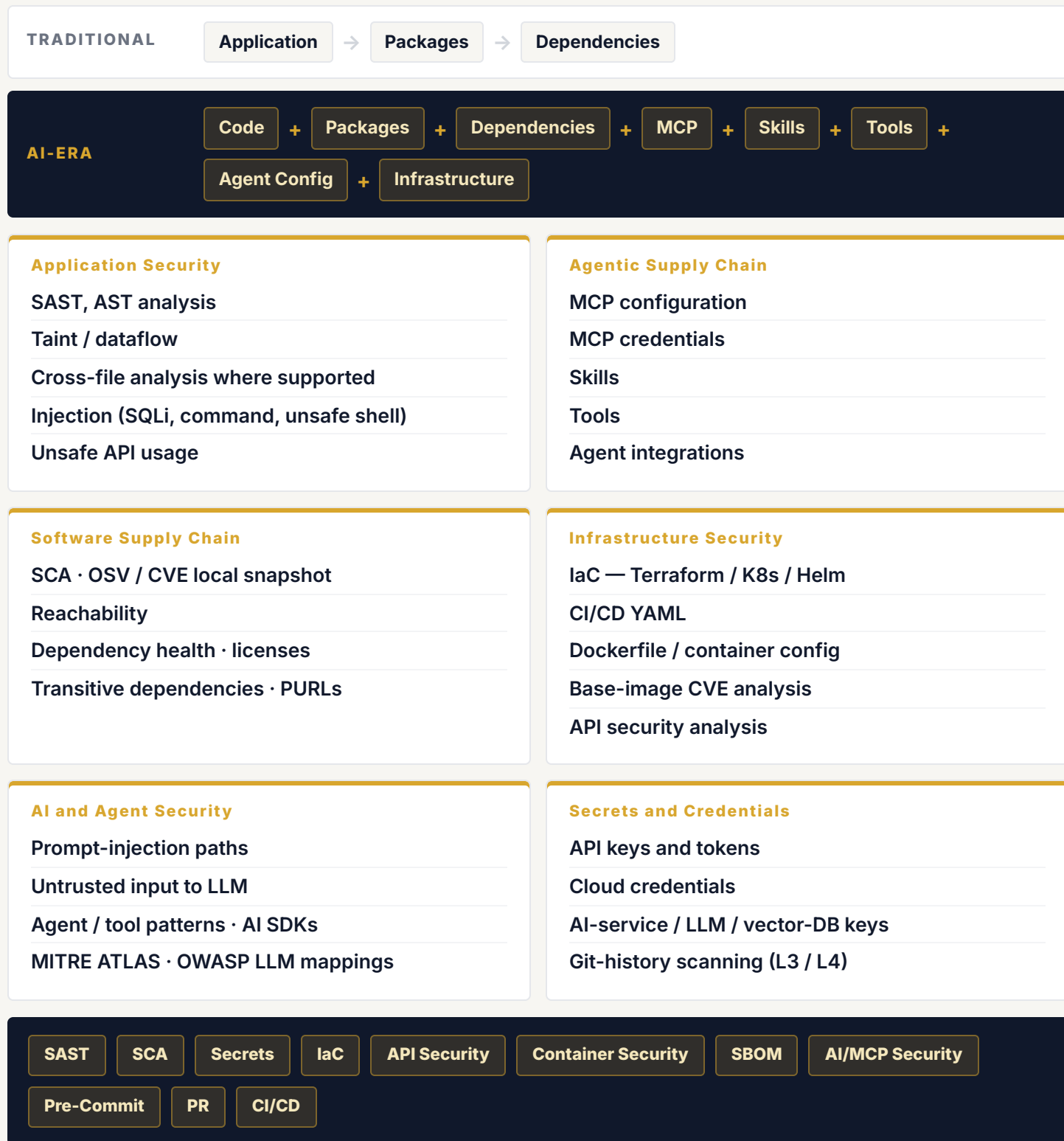


THE CODING AGENT CAN PROPOSE THE FIX.

The deterministic GCA security gate remains the source of truth.



Comprehensive Code Security. Built for the AI Development Era.



Language depth and scan options: gadriel.ai/docs/gca/scan-coverage



From finding to verified fix

Example remediation workflow. A deterministic finding goes back to the coding agent, a fix is proposed, GCA revalidates independently, and the result is recorded locally.

AI-GENERATED CODE STILL IN CONTEXT

```
def get_user(uid):  
    q = "SELECT * FROM users WHERE id=" + uid  
    return cursor.execute(q)
```

HIGH SQL injection path

Untrusted request input reaches a SQL execution sink without a recognized sanitizer or parameter binding.

uid → query string → cursor.execute()

AGENT-ASSISTED FIX

```
cursor.execute("SELECT * FROM users WHERE id = %s", (uid,))
```

01

Detect

validate_buffer /
validate_file returns
deterministic
findings.

02

Return

Finding enters the
coding agent's live
context through
MCP.

03

Fix

The agent proposes
the change. GCA
never silently
commits.

04

Revalidate

GCA rescans and
clears or reopens
the finding.

05

Record

Decision and
remediation state go
to the local audit
trail.

WHY THIS MATTERS

The security result does not depend on the model admitting a mistake. The developer is not the relay. The fix is verified, not assumed. Every step can remain local and auditable.

KEY DIFFERENTIATOR

AI can generate the code and propose the remediation. GCA provides the independent assurance decision.



Enterprise Assurance

Know exactly what is in the software you ship.

What ships

SPDX 2.3

CycloneDX 1.5

Direct and transitive components

PURLs

Vulnerability metadata

Change and evidence

License where available

Per-commit snapshots

SPDX diff — added / removed / version / CVE-changed

Local JSON

EU CYBER RESILIENCE ACT READINESS

GCA supports CRA readiness. It does not make an organization CRA-compliant.

Helps teams produce inventory, connect it to vulnerabilities, and keep local evidence as software evolves.

11 September 2026 — Article 14 reporting begins. 24-hour early warning for an actively exploited vulnerability or a qualifying severe security incident — not every vulnerability.

11 December 2027 — CRA generally applicable.

Fines: up to €15 million or 2.5% of total worldwide annual turnover, whichever is higher.

QUESTIONS GCA ANSWERS

Did the coding agent introduce a vulnerability?

Did it add a vulnerable dependency?

Is that dependency actually reachable?

Did it expose a credential?

Did it create an unsafe API?

Did it add or modify an MCP/Skill integration?

Did the generated fix actually work?

Does the change satisfy our security policy?

What changed in our software supply chain?

Can we produce an SBOM and audit evidence for this release?

Enterprise

Watch

Pre-commit

PR

CI/CD

MCP

Agentic remediation

GVL custom policies

PASS / PARTIAL / FAIL

Audit trail

Air-gap

Privacy by architecture

Source, findings, graphs, SBOM, and audit evidence remain local by default. Closed outbound list. Agent LLM independent of Gadriel.



Why GCA. Start free.

Code security should be the default, not a premium upgrade. You do not need to rip out what you have. Run GCA and compare.

01

This is the capability you are paying for

SAST, SCA, secrets, IaC, API, containers, SBOM, pre-commit, PR, and CI/CD in one local harness, plus AI/MCP surfaces those tools were not designed around.

02

Free for individual developers

The core deterministic engine is available to individual developers at no cost. Enterprises pay for governance, policy and scale, not for the right to discover insecure code.

03

You do not have to believe us

Do not uninstall your current scanner. Run GCA beside it on the same repository. Compare the findings before the next renewal.

04

No procurement to try it

Install it locally. Point it at your repo. Your repository does not have to be onboarded into another SaaS security platform.

Does GCA have the capabilities you are paying for today?

SAST	YES	SCA / dependency security	YES
Secrets detection	YES	Reachability	YES
IaC / CI security	YES	API security analysis	YES
Container / Dockerfile	YES	SBOM	YES
Pre-commit security	YES	PR / CI/CD security	YES
AI/MCP-specific security	YES	Agent-assisted remediation	YES
Independent re-scan	YES	Local-first / air-gap	YES

Run GCA beside your current scanner on the same repository and compare the findings.

Individual — Free

Local scans, CLI, coding-agent integration. The deterministic engine, at no cost.

Team

Governance, shared policy, team scale.

Enterprise

Air-gap, SSO, custom rules, dedicated success.

SEE WHAT GCA FINDS IN YOUR REPOSITORY

```
npm install -g gadriel
gadriel auth login
gadriel code init
gadriel code scan .
```

Scanning and security evidence stay local by default. Optional AI-assisted reasoning requires explicit source consent.

FREE FOR INDIVIDUAL DEVELOPERS



gadriel.ai
START FREE

Already using GitHub Code Security, Snyk, Sonar or another scanner? Don't uninstall it. Run GCA beside it on the same repository. Compare the findings.