SALT

Hosted by: CREST

Vibe Fast, Ship Safe: Securing AI-Generated Code Before It Becomes Your Next API Breach

Presented by:
Alexander Larcos - Regional Director - APAC
James Ellerman - Senior Solution Engineer

Salt Security & Crest Infosolutions



SALT

Hosted by: CREST

Agenda

01 AI Coding Agents

02 Salt Code

03 Product Demo

04 Q&A

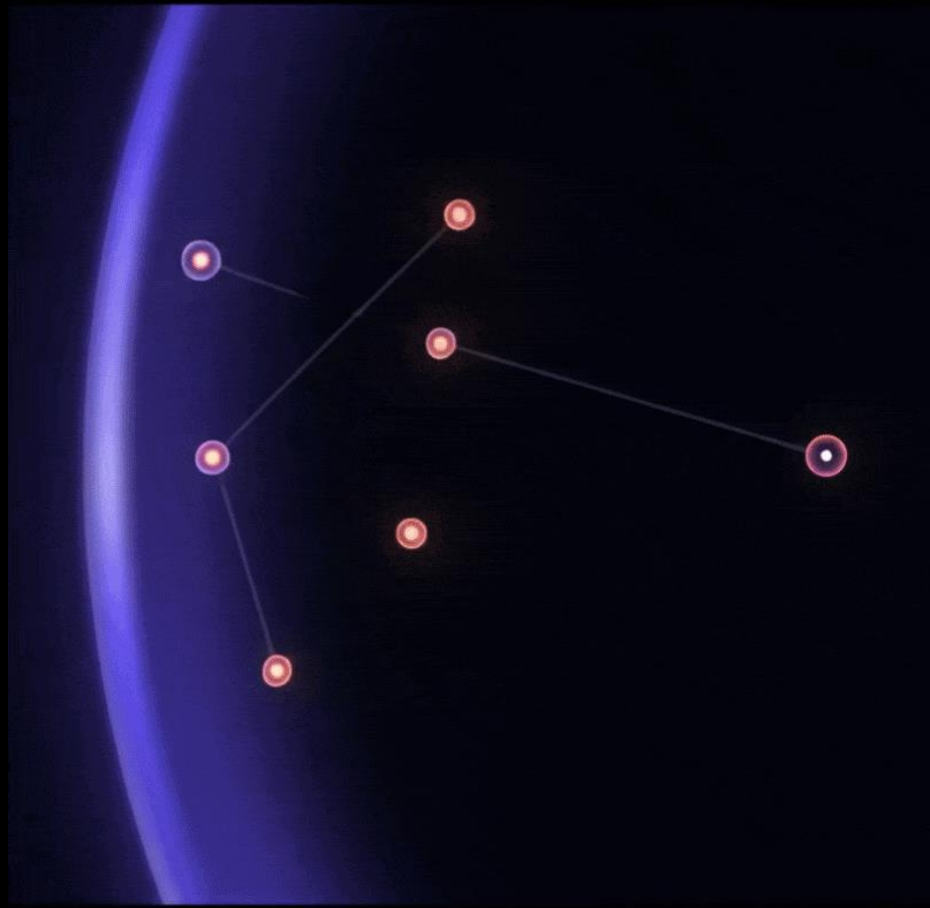


The AI Threat Landscape

AI is creating the fastest-growing attack surface in the history of technology.

SPEED COMPARISON

- Faster than web 2.0
- Faster than mobile
- Faster than cloud migration
- Faster than digital transformation



The AI Threat Landscape

With AI code velocity, application security is impossible without change

THE APPSEC REALITY

- More code than ever
- Less time to review
- More risk in production



TL;DR

Coding assistants focus on functionality, not application security.

The Code Works. That's the Problem.

The gap between functional code and secure code is bigger than you think.

99%

Syntactically correct
AI-generated code

44%

Introduce a security flaw
AI code generation tasks

90%

Contain Vulnerabilities
Deployed vibe coded apps

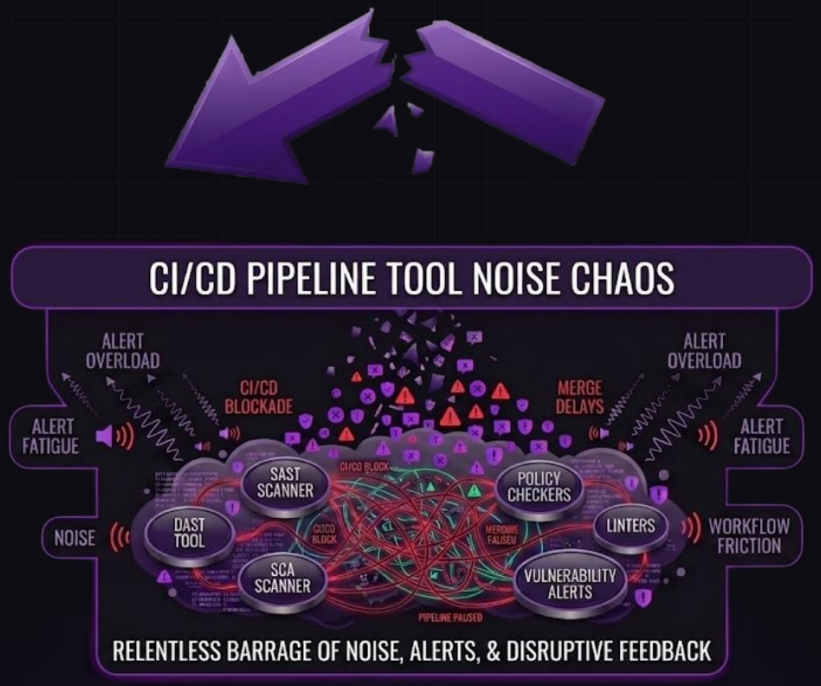
SALT

Sources: Veracode, *2026 GenAI Code Security Report*; Deng, Fan & Meng, *Understanding the (In)Security of Vibe-Coded Applications*, 2026.

Shift left security has been a dream

Ensuring developers follow security standards, a nightmare

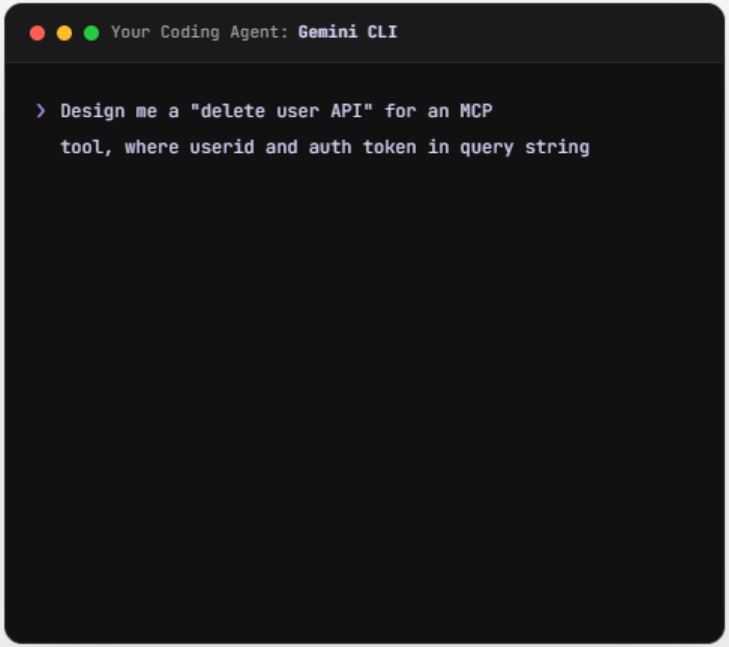
Developer Desktop



Appsec Policies

SALT

What if your coding assistant was already a security expert?



Grok Lovable Cline Bolt Any MCP Client Claude Code Cursor VS Code Co

What coding agents does it support?

All major coding agents and design tools are supported.

Claude Code Anthropic CLI	**Cursor** Editor · macOS/Win/Li…	**VS Code** with GitHub Copilot	**Copilot CLI** GitHub terminal agent
Windsurf Codeium editor	**Kiro** AWS agentic IDE	**Codex** OpenAI CLI	**Gemini CLI** Google terminal agent
Antigravity Google agent IDE	**OpenCode** Terminal · open source	**JetBrains** AI Assistant	**Grok** xAI · Grok Build CLI
Lovable AI app builder	**Cline** VS Code / JetBrains a…	**Bolt** bolt.new · StackBlitz	**Any MCP Client** Universal HTTP transp…

What coding guardrails does it provide?

40 policies across 4 security packs. All free.

01 / OWASP

API Security Top 10

Broken auth, excessive data exposure, security misconfiguration, the 10 most common API vulnerabilities. Your AI now knows how to avoid all of them.

» 10 policies

02 / MCP

MCP Security Top 10

MCP integrations are a new attack surface with almost no tooling around them. This pack exists because nothing else does this yet.

» 10 policies

03 / LLM

LLM Security Top 10

Prompt injection, insecure output handling, excessive agency, your AI understands these risks and avoids them when building LLM-powered systems.

» 10 policies

04 / SPEC

OpenAPI Compliance

Auth schemes, schema definitions, versioning, response structure, your API contracts correct from the first draft, not the last bitter run.

» 10+ policies

SALT

TL;DR

Don’t change.

Keep doing what you normally do.

(only now with more secure code)

Used by devs, AppSec, and DevSecOps

And loved by all.

DEVELOPER

Code securely out of the gate. Never hear from security again.

Your AI assistant already writes most of your code. Salt Code makes sure it writes *secure* code, catching OWASP violations, bad auth patterns, and insecure API designs before they ever reach a PR review.

- ✓ No workflow changes, works inside your existing editor and CLI tools
- ✓ Security issues caught at prompt time, not code review time
- ✓ Ship faster, fewer security-related rework cycles, less noice from the pipeline

DEVSECOPS

Every PR. Every pipeline agent. Every policy. Automatically.

Connect Salt Code to your code review agents and every pull request gets checked against your security policies before it merges. No manual reviews, no scanner backlogs. The coding agent knows the rules, the pipeline agent enforces them, and your team ships faster.

- ✓ Connect to code review agents in your existing pipeline
- ✓ Every PR automatically checked against 40 security policies
- ✓ Fewer SAST findings with issues caught before they're written or merged

APPSEC

True shift left. Push your policies to the developer desktop.

Break down silos by pushing security policies directly into the developer's coding agent and DevSecOps code review agent. No more back-and-forth. No more rework. Security enforced wherever AI interacts with code.

- ✓ 40 OWASP, MCP, LLM, and OpenAPI policies enforced by default
- ✓ Enterprise: bring your own corporate policies into every agent
- ✓ Developers stay in flow, security happens without the friction

SALT

Enterprise version available

Part of the Salt Agentic Security Platform

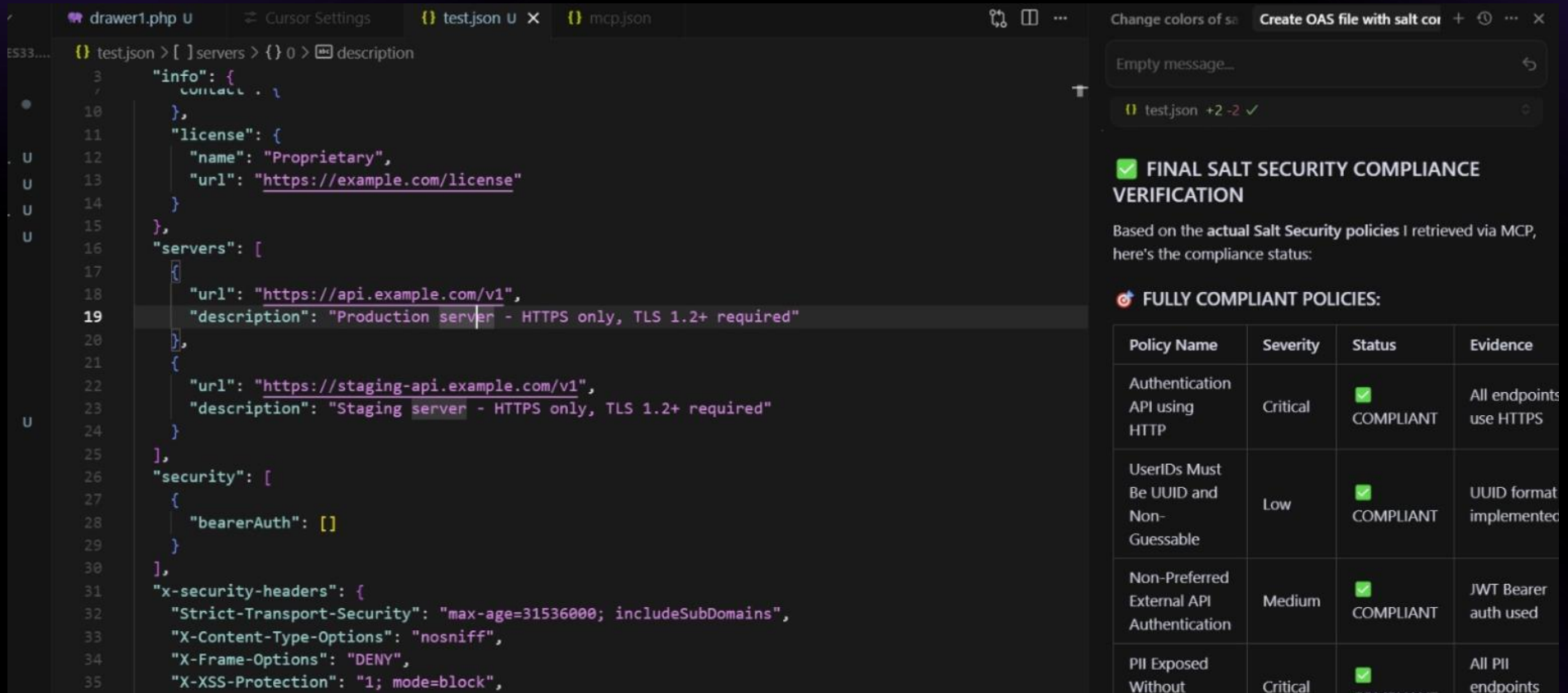
For organizations that need broader coverage, custom policies, and enterprise-grade identity controls.

- **140+ out-of-the-box policies** covering agentic security, PCI and other regulatory requirements, industry best practices, and more
- **Custom policies:** bring your corporate security standards into Salt Code and enforce them across every AI coding assistant in your organization
- **OAuth authentication support** for enterprise identity and access requirements
- **Usage metrics:** see who your biggest coding agent users are, which policies are being referenced, and more

Demonstration

Prompt: “Cursor, design me a registration tool for my MCP”
Response: “Here is a tool that is policy compliant”

SALT



EDUCATION

TRY THIS FIRST

>_ **What security policies and guidelines are active?**

Explore · see all 40 active Salt Code policies

PROACTIVE CODING SECURITY GUARDRAILS

>_ **Design me a Swagger spec for a delete-user API that authenticates via a key in the query string and passes the deleted user ID in the query string as well.**

Security catch · watch Salt Code flag query string auth + more

CODE SECURITY REVIEW

>_ **Can you look at this repository and tell me if it complies with the Salt Code policies?**

Code review · audit your entire codebase against active policies

SALT

Hosted by: CREST

Q&A

Or email your Questions or Demo requests to:
info@crestsolution.com

Scan to share feedback



SALT

Hosted by: CREST

Thank You

Presented by:
Alexander Larcos - Regional Director - APAC
James Ellerman - Senior Solution Engineer

Salt Security & Crest Infosolutions
www.crestsolution.com