boost®
security.io

PRODUCT BRIEF

Boost Security Developer Endpoint Security

THE PROBLEM

We have moved past "vibe coding." We are now in the era of industrial AI generation.

The early days of AI were about prototypes and single-file scripts. Today, autonomous agents are modifying complex codebases, managing dependencies, and executing operational tasks at machine speed. The volume of code entering your repositories is exploding, and the "human review" layer effectively no longer exists for the majority of generated lines. While security teams focus on the pipeline, risk has moved upstream to the developer's workstation. This is where the agents run, where the context is gathered, and where the decisions are made.

Attackers have noticed. They are targeting the components that feed these agents:

The Unmanaged Toolchain

Developers connect agents to unverified MCP servers and install untrusted IDE extensions to optimize workflows.

The Context Layer

Agents ingest everything in their path (dotfiles, environment variables, local configs), often leaking credentials into the context window or out to third-party models.

The Supply Chain

Agents hallucinate dependencies and ingest typosquatted packages instantly. Malware enters the environment before a human ever reviews the commit.

Security teams operating only at the CI/CD level are too late. You cannot protect a factory running at machine speed if you have no visibility into the machines doing the work.

THE BOOST SOLUTION

Developer Endpoint Security

Boost Security Developer Endpoint Security extends your control plane to the point of creation. It secures the developer machine, governs the AI agents running on it, and enforces policy before code is ever committed.

With Boost Security, you can:

Govern the Toolchain

Inventory and control which agents, MCP servers, and extensions are allowed to run.

Enforce at Generation

Inject security standards directly into the agent's workflow, ensuring code is generated securely from the first token.

Secure the Context

Sanitize prompts and harden the environment to prevent secrets from leaking into models.

Automate the Fix

Give developers the ability to fix vulnerabilities instantly using organizational context, matching the speed of generation.

How It Works

Boost Security Developer Endpoint Security delivers eight integrated capabilities that secure the developer machine, the coding agent, and the code itself — from the moment a prompt is sent to the moment code is committed.

1 Unified Endpoint Inventory

Stop flying blind. Boost continuously maps the "AI-BOM" of your developer fleet. We detect which coding agents (Cursor, Windsurf, Claude Code), MCP servers, local models, and IDE extensions are active.

You get a real-time view of the tools shaping your software.

2 Coding Agent Governance

An agent is only as safe as its connections. We validate that coding assistants are connecting only to approved MCP servers and plugins. We block unvetted connections and prevent "drift" from your security baseline, ensuring agents operate within defined boundaries.

3 Workstation Hardening

Attackers know that developer laptops are the path of least resistance. Boost scans the local environment for exposed credentials in config files, shell history, and environment variables. We flag misconfigurations that widen the blast radius, locking down the machine before an agent (or an attacker) can exploit it.

4 Secure Agentic Code Generation

Prevention beats remediation. Boost injects your secure coding standards and architecture patterns directly into the agent's context window. The agent knows what is allowed before it starts typing, and generated code is analyzed in-line. Issues are caught and fixed in the IDE, not the CI pipeline.

5 Supply Chain Defense

Block the ingestion of malware. Whether it's a developer running npm install or an agent hallucinating a package, Boost checks every artifact against a comprehensive threat model. We block typosquatting, dependency confusion, and known malware at the moment of ingestion.

6 Data Leakage Prevention

Your API keys do not belong in a model's training data. Boost scans outbound prompts and context files, masking credentials and sensitive data before they leave the local environment. Developers stay fast while data stays private.

7 Machine-Speed Remediation

Fix vulnerabilities as fast as they are generated. Boost provides context-aware remediation that understands your repository's architecture. We generate fixes that match your coding standards, allowing developers to resolve security findings instantly without breaking flow.

8 Distributed Policy Enforcement

Define locally, enforce globally. Security teams set the standard once, and Boost projects that policy to every developer machine and AI agent. Enforcement happens at the edge, ensuring consistency across the entire software factory without manual intervention.

info@boostsecurity.io
55 Rue Peel Suites 203-208, Montréal
H3C-2G9, QC, Canada
boostsecurity.io

boost security.io