Shipping 10x with AI agents.

Without becoming the next supply-chain headline.

Mark Lechner

CISO · Docker · LeadDev London 2026

Front row to the agent era.

Most of the world's developers, and increasingly their AI agents, build through Docker.

DEVELOPERS

20M+ daily

Build with Docker every day.

AI AGENTS

Growing weekly

Cursor, Claude, Copilot. All on the substrate.

DOCKER

The substrate.

Our vantage is six to twelve months ahead of what hits your environment.

Mark Lechner
CISO · Docker · Berlin

The new gravity.

Code velocity is compounding. Attack velocity is compounding faster.

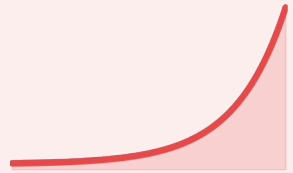
CODE VELOCITY

Compounding.

- Agents produce code humans can't review.
- Volume of vuln code compounds month over month.
- Mandate from leadership: accelerate.

ATTACK VELOCITY

Compounding faster.



- CVSS 10s used to be annual events max. Now weekly.
- Vendors, registries, deploy platforms breached weekly
- Audit cycles can't keep up. Compliance erodes.

THE TRAP

Agents have human-class access at machine-class throughput. The blast radius now extends beyond your controlled environment

Opportunism is the symptom. The ecosystem is the problem.

When attackers breach Tier-1 targets on a weekly basis they steal credentials and exit. Not because credentials are the prize - they have more access they know what to do with.

01

Attackers' problem is now ours.

More access than plans. Credentials first, plans later.

02

The trust model itself is failing.

Checkmarx, Aqua, Vercel, GitHub. Nothing in common except value.

03

You can't out-defend a broken model.

Every pull, image, CI action assumes trust the upstream can't carry.

WHAT'S NEXT

The next 6 to 12 months will be uncomfortable. The teams that come out ahead build trust at their own boundary.

Friday, 4:47pm.

A developer's agent runs npm install to add a dependency. A transitive package was poisoned overnight; its install hook brings in the payload. The dev's reviewing a PR in another session. Nobody notices.

> ***Trust boundary wraps around components, execution, credentials.***

WITHOUT THE TRUST BOUNDARY

- Install hook executes as the developer. Full session, full privileges.
- Reads AWS SSO, SSH keys, gh tokens, registry creds, zsh history.
- VPN alive. Creds touch prod, or exfil to c2.
- You find out Monday. Maybe Tuesday.

WITH THE TRUST BOUNDARY

- Install hook ran inside an isolated environment. Host inaccessible.
- No secrets to take. Secrets proxied, short-lived.
- Egress default-deny. No path to prod, no exfil to c2.
- Sandbox logs, SBOM, canaries all wired into SIEM.

THE POINT

Same developer. Same workflow. On Friday at 4:47pm, nothing happens.

Good old SSDLC is the AI moat.

Six principles. None of them are new. All of them are what stops Friday at 4:47pm.

01

Know your stack

Tech radar humans and agents can read.

02

Control your inputs

Locked-down registry. 7-day cooldown. Hardened baselines.

03

Isolate execution

Sandboxed dev environments. Credentials proxied, never stored.

04

Authenticate everything

Pinned deps. Signed commits. Short-lived tokens.

05

Detect, monitor, comply

Canaries everywhere. Audit posture stays current.

06

Gate at deploy

Admission control. AI review agent in PR.

THE INSIGHT

None of it is new. At agent speed, the cost of getting it wrong just went exponential

What happens if you do all of these.

Five outcomes the SSDLC discipline buys you.

1. **Malicious payloads have nothing to grab.**
   Nothing to steal, nothing to execute on, nothing to backdoor into.

2. **YOLO mode for all agents, safely.**
   You stop being the AI slowdown. You become the champion of AI adoption.

3. **A trust boundary that travels.**
   Laptop, CI, cloud. Same shape everywhere. Same surface for agents and MCP traffic.

4. **Every new app starts from a secure baseline.**
   Hardened, near-zero CVE foundation. No special work required.

5. **Real-time software composition.**
   Verifiable provenance, end to end. See your entire stack at all times.

Plug in agents in phases.

Once the trust boundary holds, you can climb. Without it, you can't get past L2 safely.

L1

Assistive

Responds on demand; you drive every step

GUARDRAIL

Human-in-the-loop throughout.

L2

Delegated

Takes a scoped task, runs multiple steps, returns work for review

GUARDRAIL

Trust boundary + SSDLC

L3

Autonomous

Business function assumed, goal setting, orchestration and execution autonomous

GUARDRAIL

Trust Boundary + SSDLC + AI Governance

WHERE THE PAIN IS

Most teams reach for L2 before the foundation can carry it.

What you ship home with.

Do the SDLC right. Nail the isolation aspect. Climb the ladder. Here's what you get.

Malicious payloads have nothing to steal or execute on.

YOLO mode for all agents, safely.

A trust boundary that travels. Laptop, CI, cloud.

Every new app starts from a secure baseline and uses vetted tech stack

Real-time software composition with verifiable provenance.

Compliant by design. Audit posture stays current.

Control the input and isolate the execution

Build it once. Ship 10x with AI agents.