Agentic AppSec Unleashed '26

by Checkmarx

Agentic AppSec Unleashed '26 by Checkmarx

Trust the Agent. Verify the Code.

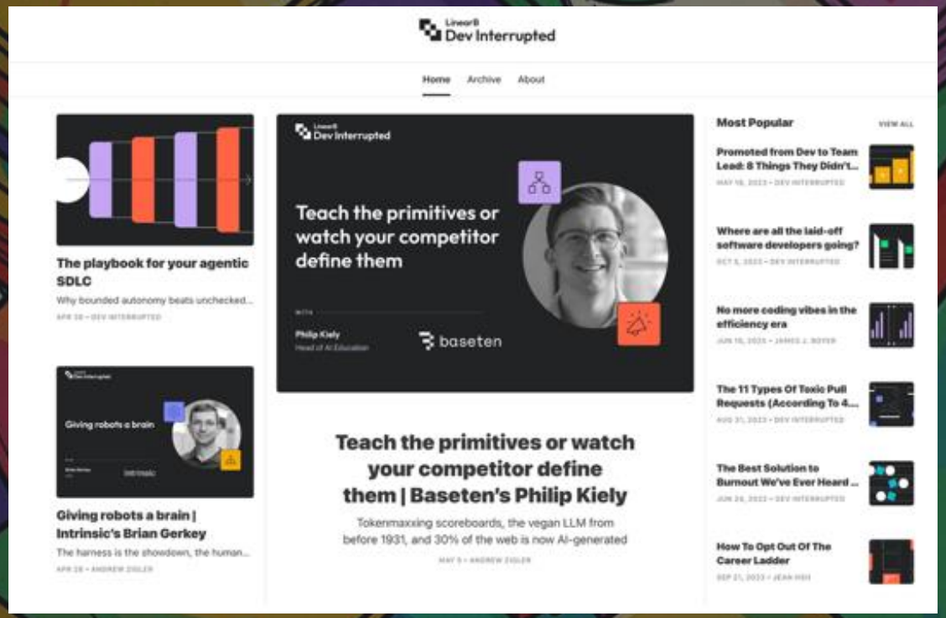
Andrew Zigler
GTM Engineer

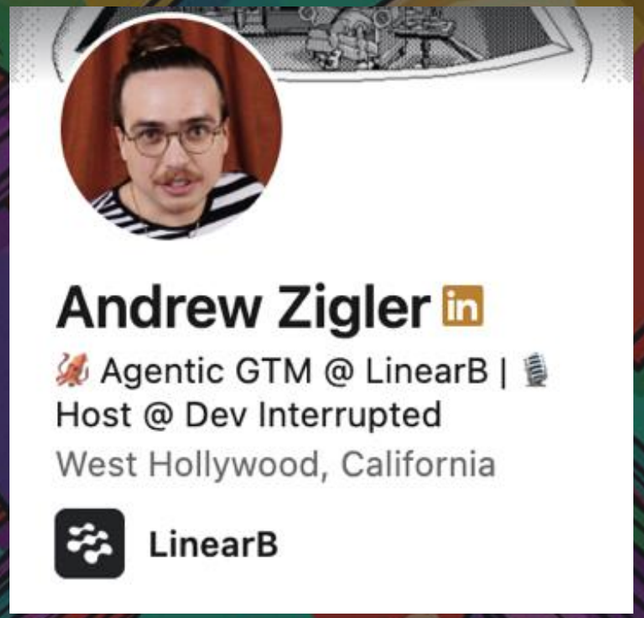
LinearB
Dev Interrupted

Andrew Zigler

Host @
Dev Interrupted

GTM Engineer @
LinearB





Before I was an engineer…


SCIENCE!
ART

Trust before.
Verify after.

For 7 control points — 4 things.

1

Surface

What the control is about.

2

Recommendation

The soft, recommendation-based way to do the control.

AKA the policy that gets ignored

3

Mechanism

The deterministic, mechanical way to do the same control.

AKA the hook that can't be bypassed

4

First Question

What to ask to investigate whether your team has it in place.

For 7 control points — 4 things.

1

Surface

What the control is about.

2

Recommendation

The soft, recommendation-based way to do the control.

AKA the policy that gets ignored

3

Mechanism

The deterministic, mechanical way to do the same control.

AKA the hook that can't be bypassed

4

First Question

What to ask to investigate whether your team has it in place.

What the agent is told

Surface

System prompt, CLAUDE.md, loaded skills, MCP descriptors, the content the agent reads while acting (the prompt-injection surface).

Recommendation

“Follow the AI usage guidelines in Confluence.”

Mechanism

Instructions are git-tracked, diff-able per session, knowable per run; ingested content treated like supply-chain code at the boundary.

First Question

Where does the agent's instruction set live, and can you diff it from a month ago?

A wiki page isn't a policy.

What the agent can do

Surface

Tools, MCP scopes, filesystem and network permissions, credentials.

Recommendation

"Don't put secrets in prompts."

Mechanism

Deny-by-default permissions; hooks that physically block disallowed tools; secrets stored where the agent's tool surface can’t read them.

First Question

If the agent tried to read your .env, what would stop it?

Capabilities are the real wall.

What the agent can write

Surface

The filesystem boundary of the agent's writes.

The blast radius.

Recommendation

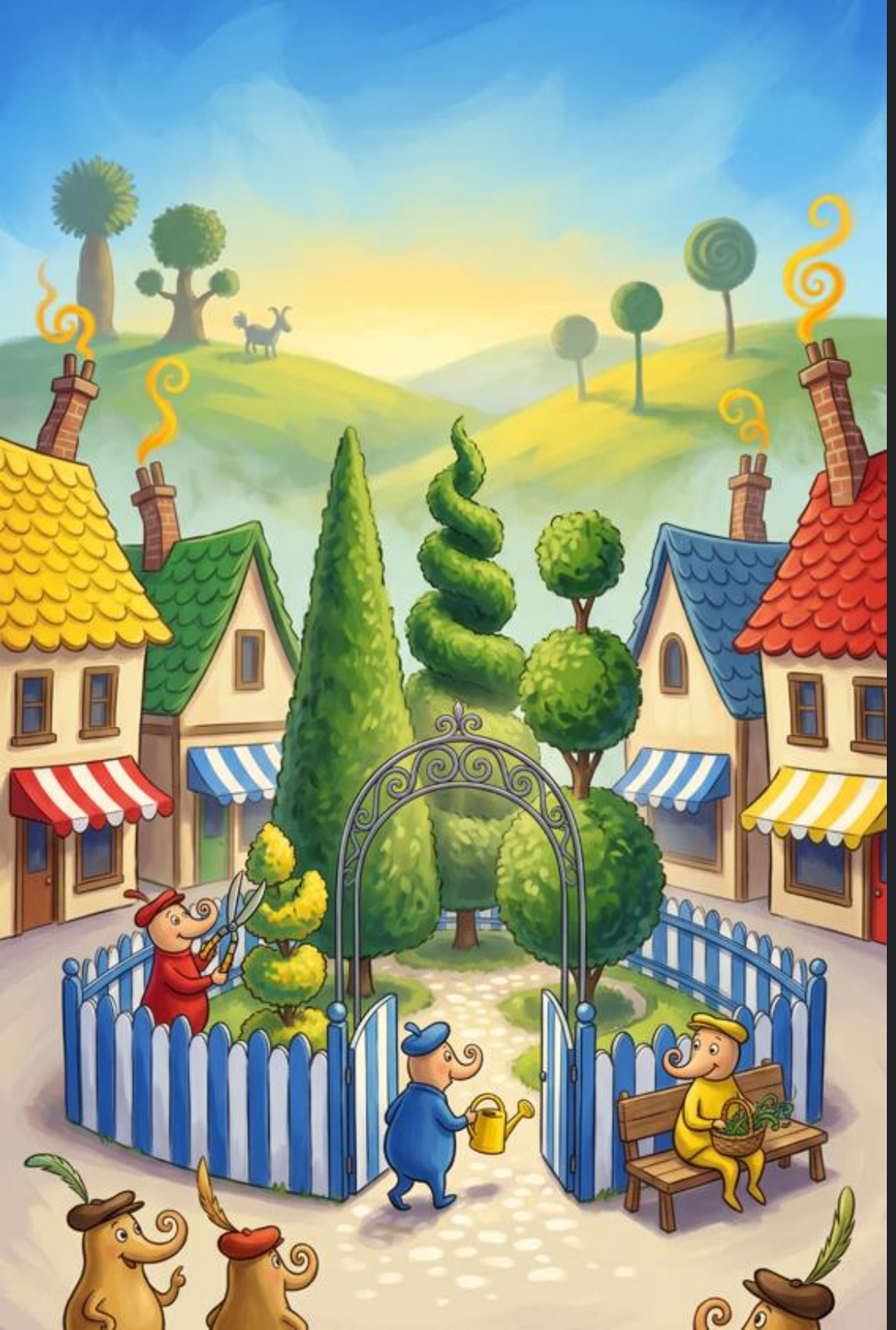
"The agent is told to write only in / repos/foo."

Mechanism

Per-agent worktrees; PreToolUse hooks reject file paths outside the worktree; the merge into trunk is gated.

First Question

If your AI tool wrote outside the project, what would stop it — policy or mechanism?



Saying so doesn't make it so.

What the agent generates

Surface

Code, configs, data files, and any new dependencies the agent introduces.

Recommendation

"Our pre-commit hook runs locally."

Mechanism

Lint and secret-detection run at the moment of write, not days later in CI; new dependencies clear an allowlist or signature check before commit.

First Question

When an agent generates a file, when does linting and secret detection happen?

If lint runs in CI, lint is too late.

What the agent must satisfy

Surface

The contract the output is held to: tests, schemas, acceptance criteria.

Recommendation

"We let AI generate the tests too."

Mechanism

Tests written and merged by a separate role BEFORE implementation; the impl agent cannot modify the tests; tests are the frozen contract.

First Question

When AI writes the implementation, who wrote the test it has to pass?

You can’t grade your own homework.

What the agent actually shipped

Surface

The function bodies committed, distinct from "tests pass."

Recommendation

"Tests pass, ship it."

Mechanism

Read the bodies, not the test-runner output; stub-body audits; delegation-assertion tests that spy on the real dependency.

First Question

When tests pass on AI-generated code, who verifies the body actually does work?

Passing tests only prove the tests passed.

What crosses the threshold

Surface

The moment code moves from agent worktree to shared trunk to production.

Recommendation

"PR review catches it."

Mechanism

Pre-commit hooks on staged files; commit messages carry an audit trail; an explicit handoff checklist refuses to sign off without verifications.

First Question

What stops an agent's commit from landing in main without verification, and who is that "what"?

You can't review at agent speed.

Trust isn't a feeling.

Verify isn't a wish.

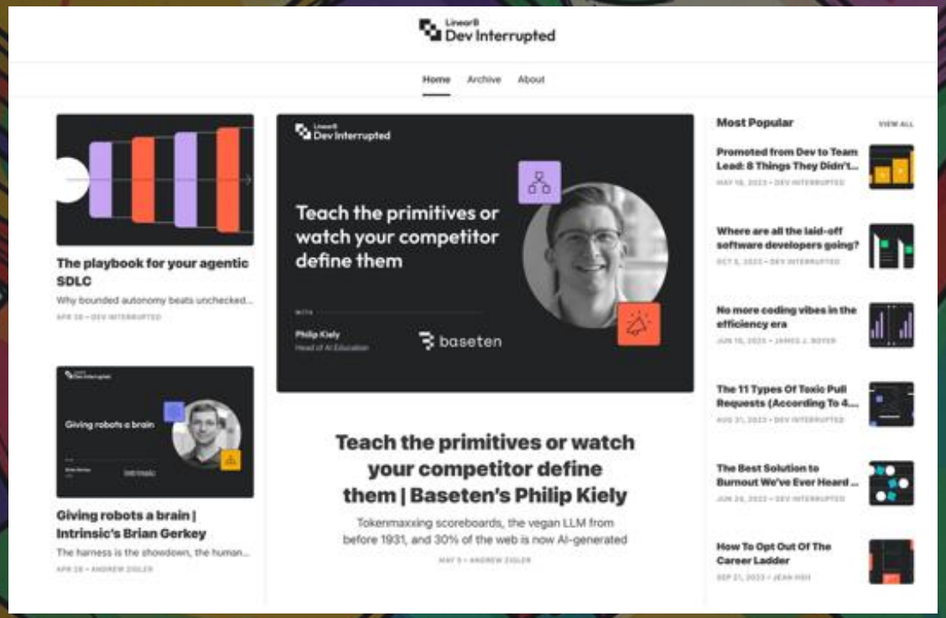
Both are surfaces.

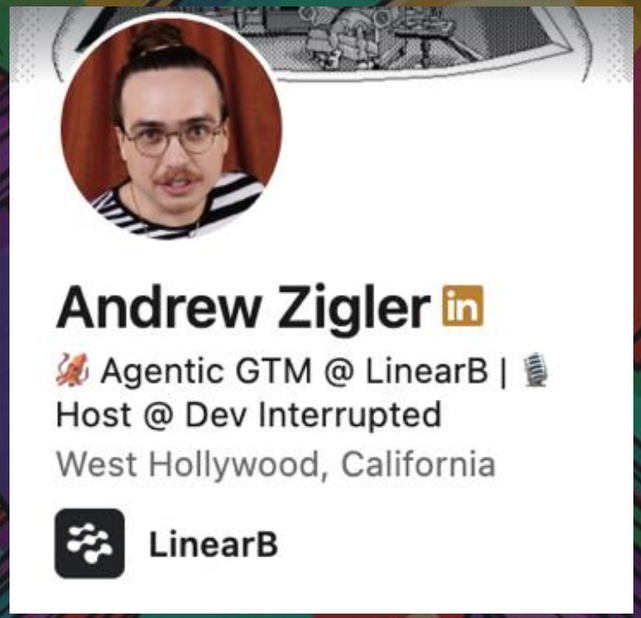
Build like it.

Andrew Zigler

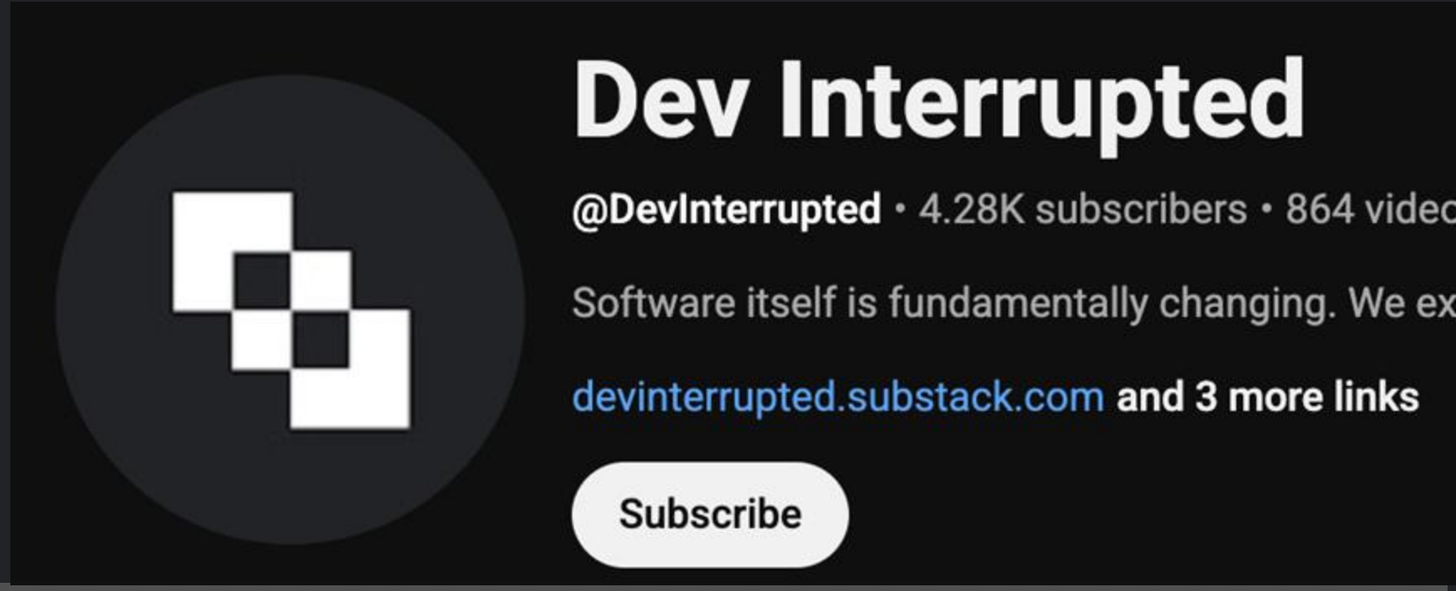
Host @
Dev Interrupted

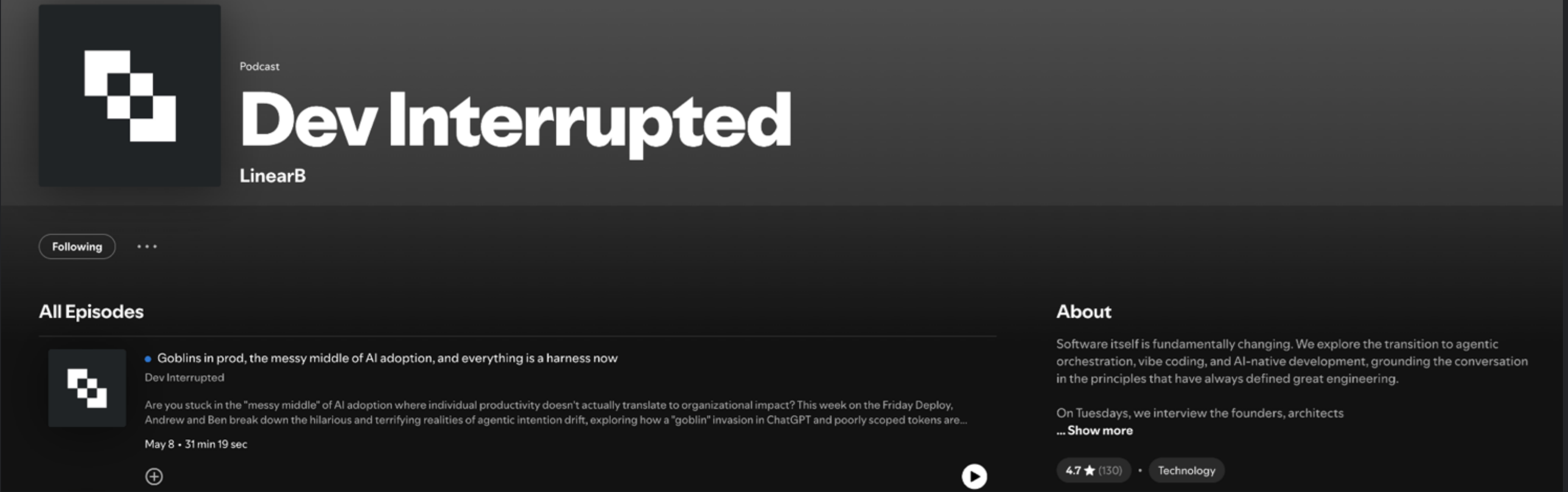
GTM Engineer @
LinearB





Andrew Zigler





Agentic AppSec Unleashed '26

by Checkmarx