Your Time Is HERE

The AppSec Career Nobody Trained For

Bruce Fram | OWASP Bay Area, April 16, 2026

AppSecAI

Hi Mom & Dad – I am an AppSec Engineer!

SUNDAY, APRIL 12, 2026

The New York Times

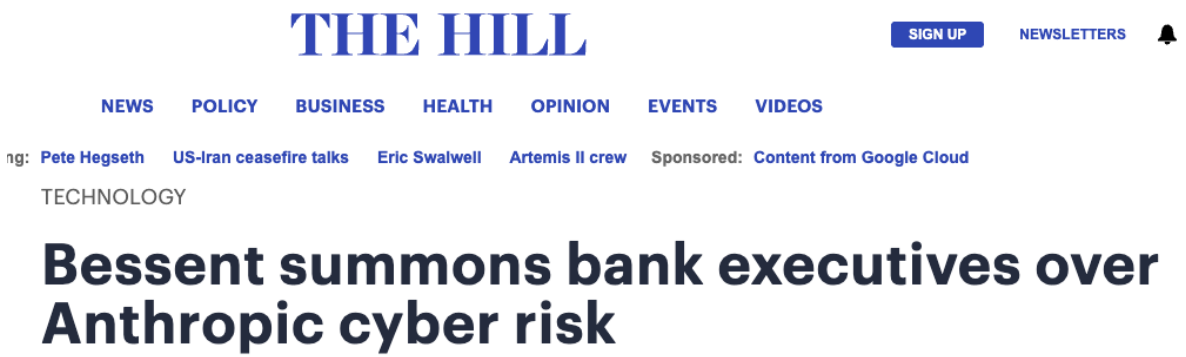
The Big Bang: A.I. Has Created a Code Overload

Companies are scrambling to deal with the glut.

"Companies are struggling to hire enough people to monitor the A.I. code for risks, a role called **application security engineer**. "There are not enough **application security engineers** on the planet to satisfy what just American companies need"

The CEO/Board REALLY Cares...

April 7, 2026 — Treasury Secretary and Fed Chair call an urgent meeting


THE HILL

SIGN UP NEWSLETTERS

NEWS POLICY BUSINESS HEALTH OPINION EVENTS VIDEOS

ng: Pete Hegseth US-Iran ceasefire talks Eric Swalwell Artemis II crew Sponsored: Content from Google Cloud

TECHNOLOGY

Bessent summons bank executives over Anthropic cyber risk

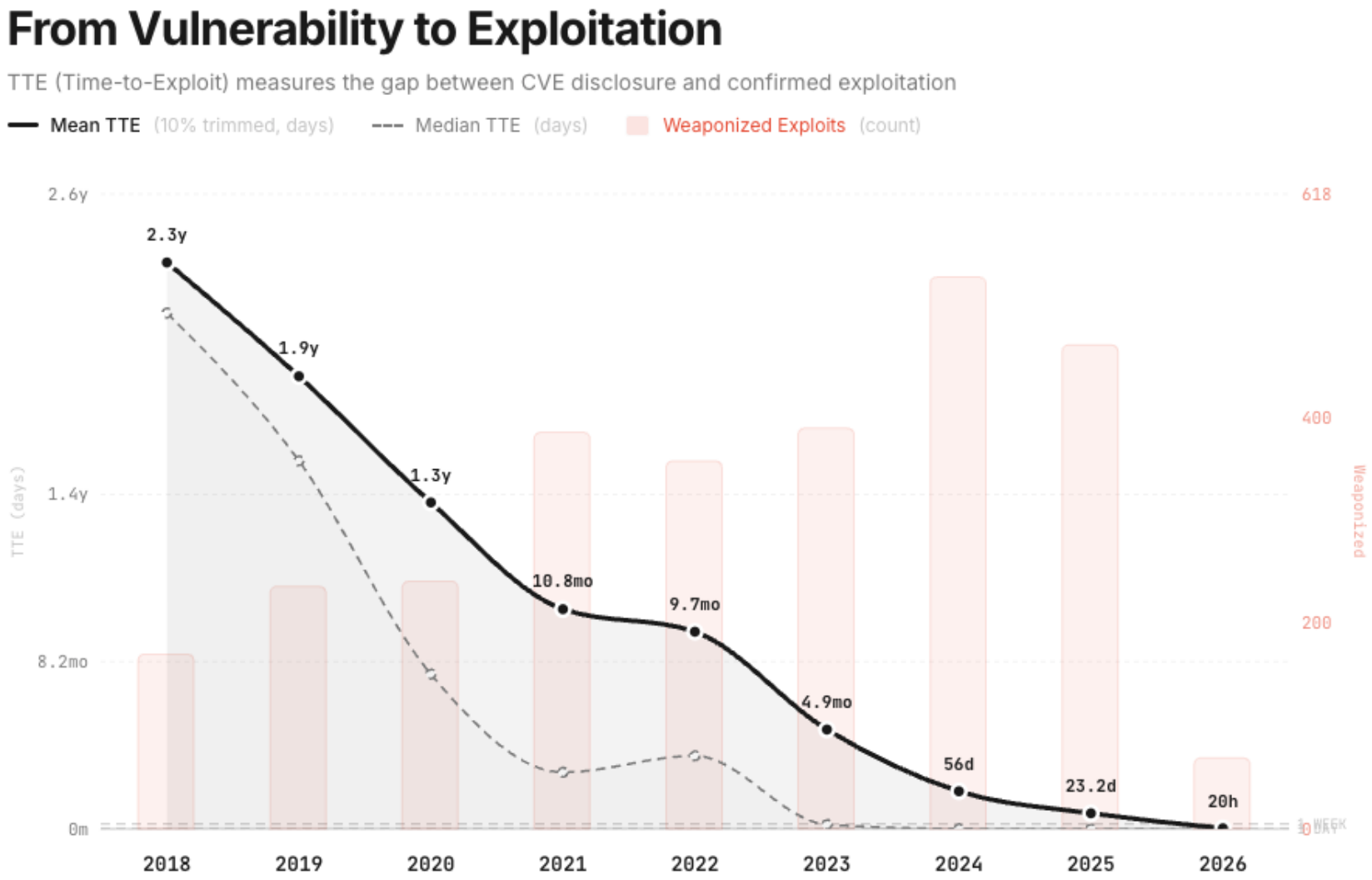
Scott Bessent — Treasury Secretary
Jerome Powell — Federal Reserve Chair
Brian Moynihan — CEO, Bank of America
Jane Fraser — CEO, Citigroup
David Solomon — CEO, Goldman Sachs
Ted Pick — CEO, Morgan Stanley
Charlie Scharf — CEO, Wells Fargo
Jamie Dimon — CEO, JPMorgan Chase


yahoo!finance

Anthropic's Mythos sparks Washington's big bank anxiety

AppSecAI

… Because they Understand This …



AppSecAI

… Can Collapse the Economy

Software runs everything. Software has vulnerabilities. Vulnerabilities are being exploited faster than ever. We are not fixing them fast enough.

$10.5T
Annual global cybercrime cost

20h
Time to exploit (was 2.3 years)

252d
Average fix time (up from 171)

That's why Bessent and Powell called the meeting.

AppSecAI

Great for US. Might Even Be Fun!

→ **The code explosion** created strong demand for what we do

→ **The answers are in the code** — that's where vulnerabilities live and where fixes happen

→ **That's our domain** — and now we have tools that let us work at machine speed

How do we deal with this and win?

1. **This is not the first time there has been a technology transformation**
   The Power Parallel — Engines → AI

2. **The new Dev / Sec / Ops**
   Who owns what — and who takes action

3. **Automation and Adoption**
   Validation, speed, and the 8.2-minute fix

4. **The GRASP Framework**
   Evaluating AI technology — five pillars

5. **Getting started — OASIS**
   Open-source security you can join today

Do you feel confused? Unsure? Behind?

New models every week. New tools every day. Your vendors are pivoting. Your developers are using tools you haven't vetted. The landscape is moving faster than anyone can keep up with.

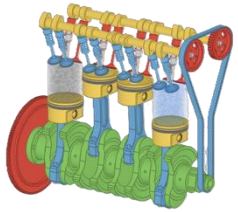
You're not behind. Let's figure this out together. Scan the QR →

AppSecAI

appsecai.io/owasp-april-2026

Good News – We have Seen This Before

The Internal Combustion Engine (1876 → 1908 Model T)



Automobiles

Agriculture

Logistics

Global Trade

Transport

Manufacturing

Personal Mobility

Energy

Commerce

Infrastructure

Same Story – Different Technology – A Little Faster Artificial Intelligence (2023 → ?)

Autonomous Vehicles

Drug Discovery

Supply Chain

Customer Service

Scientific Research

Robotics

Software Engineering

Energy Grids

Commerce

Education

Devs don't code anymore – They ask!

The More they use AI, the more they automate

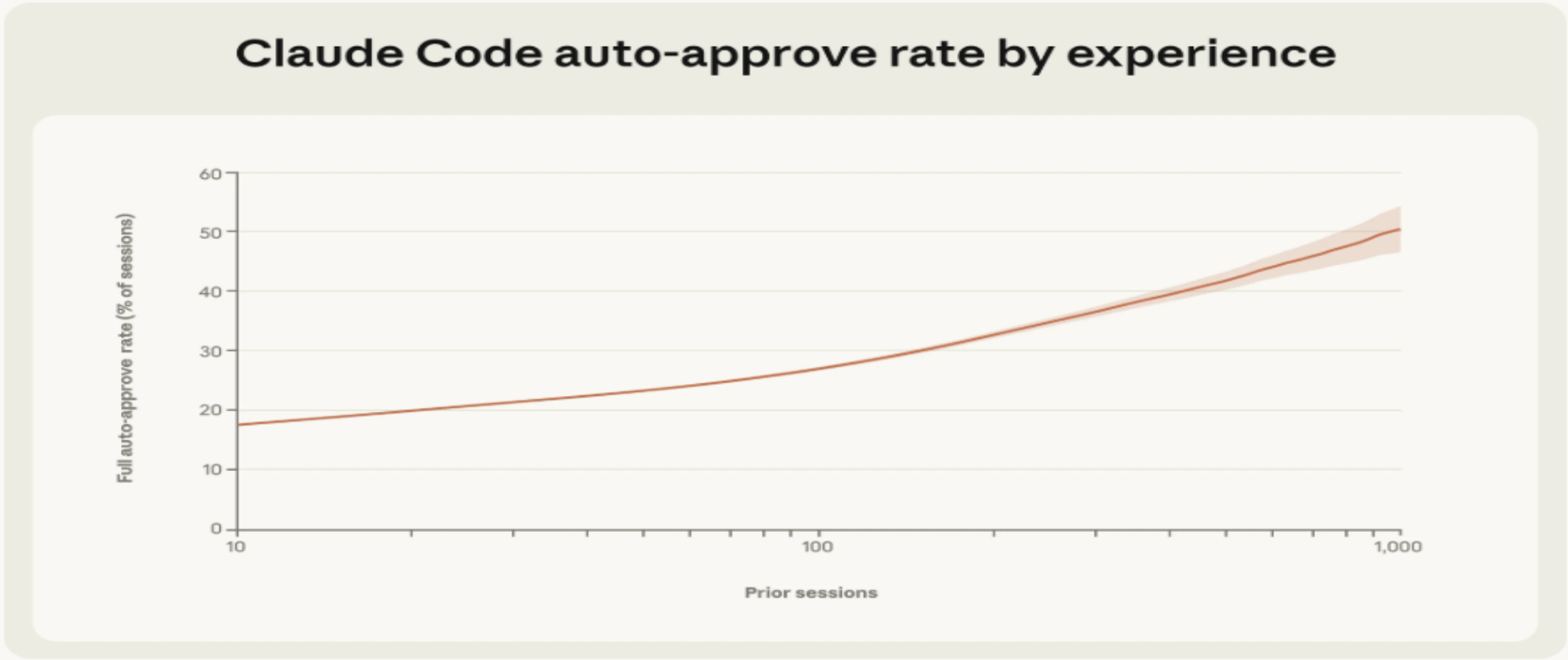
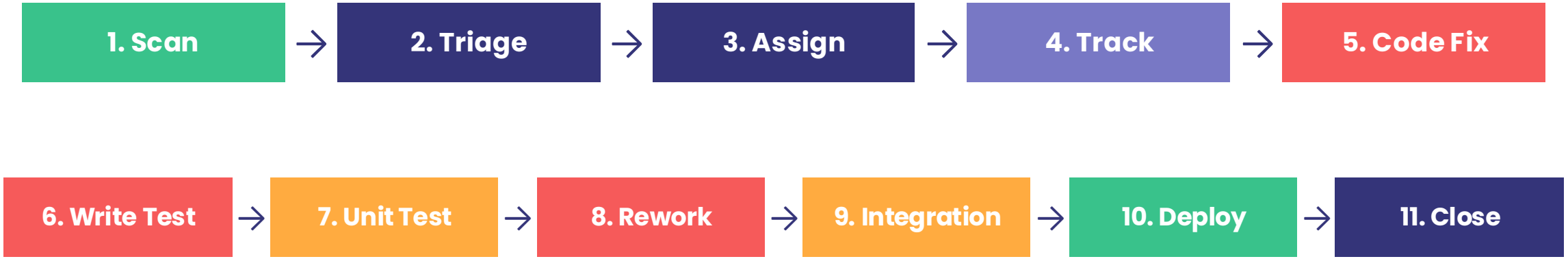


Figure 2. Auto-approve rate by account tenure. Experienced users increasingly let Claude run without any manual approval. Data reflects all interactive Claude Code usage for users who signed up after September 19, 2025. Line and CI bounds are LOWESS-smoothed (0.15 bandwidth). The x-axis is a log scale.

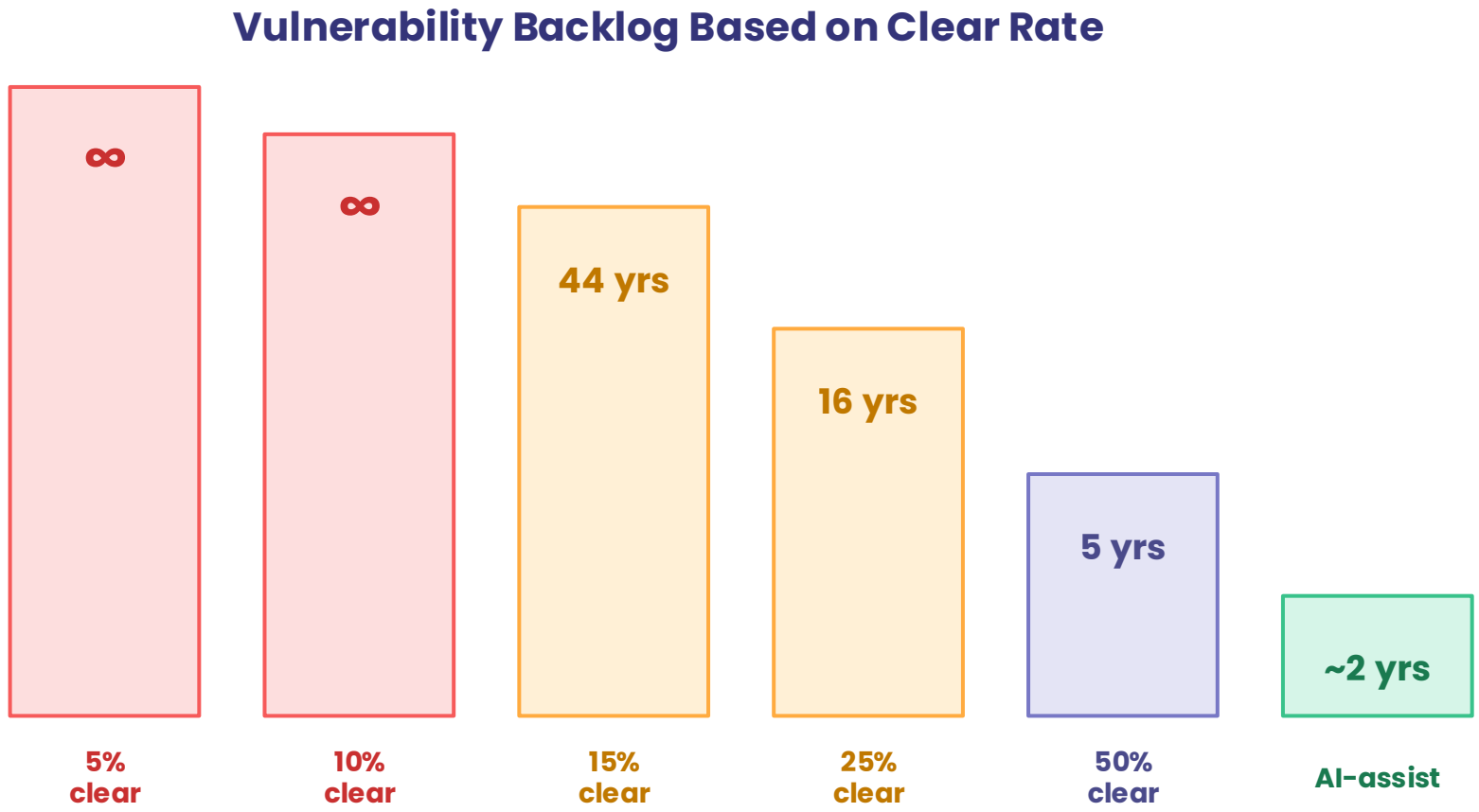
AppSec is manual and slow 11 steps — and none of them are free



Each step has a person. Each person has a delay. Each delay has a cost.

Total AppSec Spend ÷ Vulns Actually Fixed = Your Real Cost Per Fix

We mathematically catch up ... never



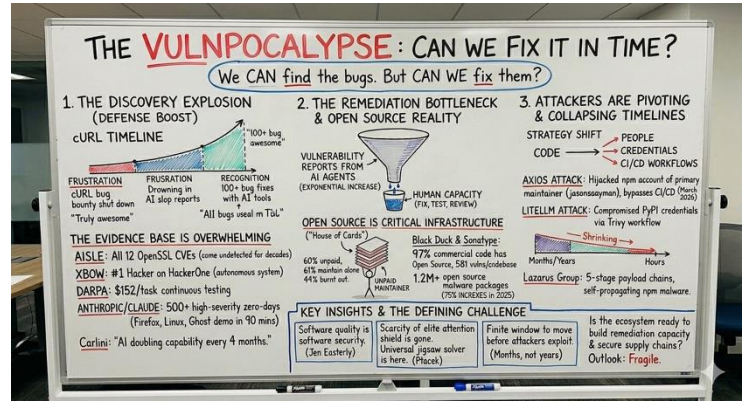
252
days avg to fix (↑ from 171)

180%
↑ exploit attacks (DBIR)

5%
of orgs fix even 15%/year

The Vulnpocalypse is coming – Finding is Easy – Will be free

"The question is no longer whether AI can find the bugs, it can. The question is whether we can fix them before attackers exploit them, and right now, the evidence suggests we're losing that race."



Chris Hughes

CEO & Co-Founder, Aquia · Founder, Resilient Cyber

20+ years cybersecurity · Author, "Effective Vulnerability Management" · Former U.S. Air Force / FedRAMP

April 7, 2026

https://www.resilientcyber.io/p/vulnpocalypse-ai-open-source-and

The industry finds problems. Nobody fixes them.

YESTERDAY	TODAY
$5,000 - 20,000+ per fix	→ 1/100 the cost
200+ days to remediate	→ Minutes to remediate
40%+ false positive rate	→ 90++% accurate triage and remediation
AppSec = "Department of No"	→ AppSec = Acceleration Team

AppSecAI

The new model – you will take direct action

What's Blocking You Today?

Dev owns the code (but not for much longer)
With AI-generated code, there is less code ownership.

Business does not want to invest in security
Every dollar invested costs $ in labor.

Tools undermine you
40% false positives. Negative ROI for years.

You have influence, not agency.
You advise on risk, but can't take direct action.

From Advising to Agency

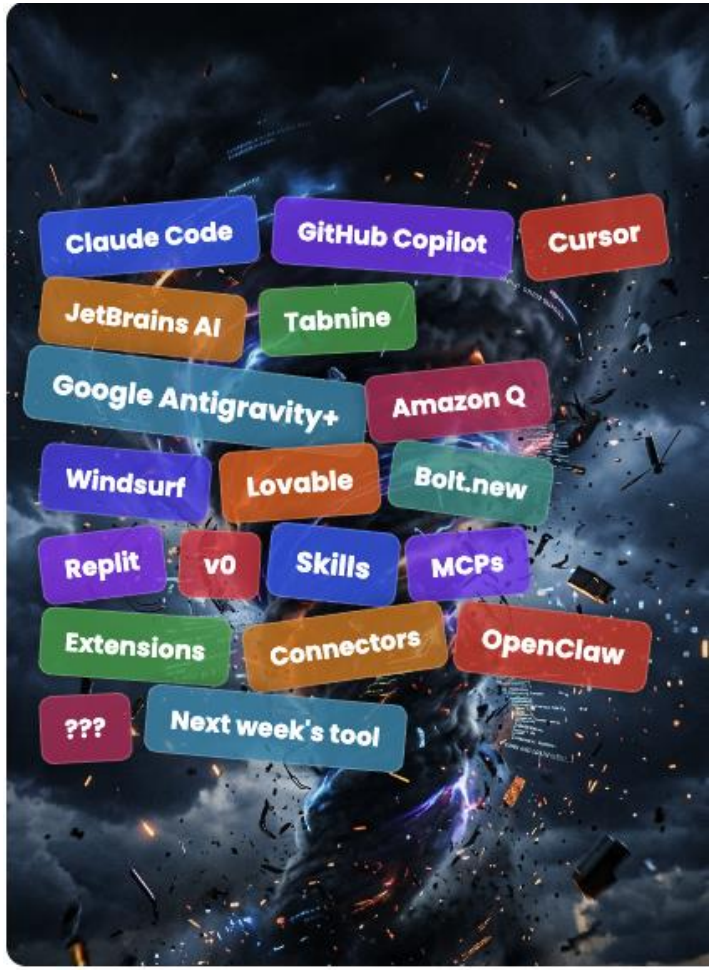
Dev = Reward
Features, functionality — what gets shipped.

Security = Risk
Assess, manage, remediate — own the outcome.

Ops = Scale
Reliability, performance — keep it running.

Equal partners at the table.
Conflict → Collaboration.

Someone Has to Own What Goes Out the Door



AUTOMATED GATEWAY

AUTOMATED ACTION — YOU CONTROL THE RULES

🟢 **Full Auto**
Decision is made automatically. Low risk, high confidence.

🔵 **Exception Auto**
Most decisions auto — exceptions proofed by people.

🟡 **Manual Assist**
AI generates — all approved by people.

⚪ **Manual**
Generated and approved by people. Highest risk, lowest volume.

Pre-Mythos – Is this fast enough? Manual Assist

8.2 Minutes per vuln

Triage → Validate → Comment → Merge

The Setup

- Never seen the code before
- Never seen our product
- Used GitHub only

What They Did

- Triaged the vulns
- Validated the fixes
- Wrote comments in GitHub
- Merged the code

Your Job

The real value isn't the frontier model itself, but **the system you build around it**: orchestration, validation loops, integration with traditional tools (SAST, DAST, SCA) and deep security expertise.

🔧
Build the System
Orchestration, validation, integration — the car around the engine

📋
Set the Policy
What's auto-fixed, what needs review, what's blocked — you decide

✅
QA the Output
Validate AI decisions, catch drift, ensure fixes are correct and relevant

📊
Measure the Outcome
Backlog shrinking? Developers adopting? Posture improving? Prove it

How — The GRASP Framework

G

Governance

Who controls the AI and what policies govern it?

R

Reliability

Does it stay up and are results consistent?

A

Assurance

Are results accurate, relevant, and solving the right problem?

S

Scalability

Can it handle the full portfolio, not just a pilot?

P

Protection

Is your data safe and do you trust the providers?

Draft for Feedback — OWASP Bay Area, April 16, 2026

GRASP Category Definition

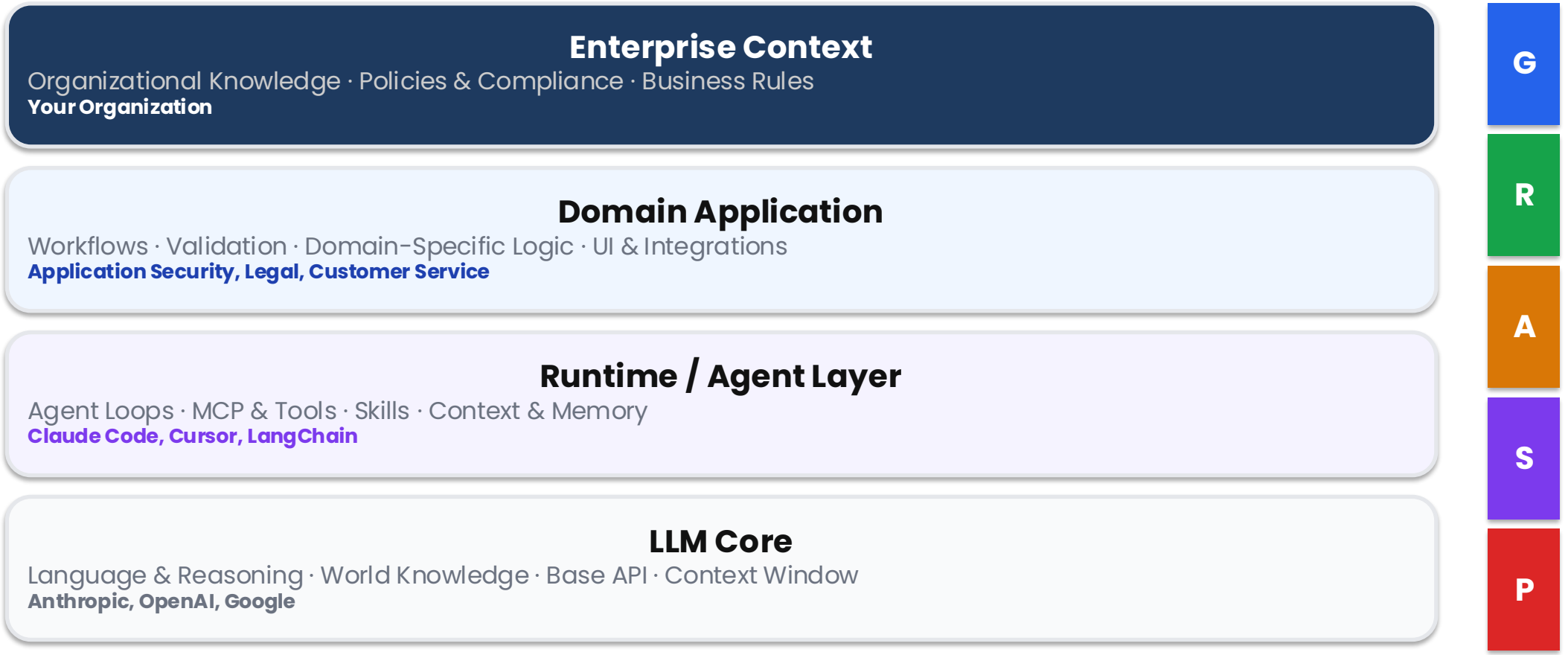
Pillar	The Question	Without It
Governance	Who controls it, and what are the rules?	"Shadow AI" — people using AI tools with no oversight, no policies, no one accountable.
Reliability	Does the system stay up, and are results consistent?	"Works on my machine" — crashes when a service is down, runs out of tokens, model update silently degrades quality.
Assurance	Are results accurate, relevant, and solving the right problem?	"Automated confidence, not competence" — 10,000 outputs, nobody checked if they're correct or useful.
Scalability	Can it handle the full workload, not just a pilot?	"Artisanal AI" — works for one team, one use case. The other 500 problems are still done by hand.
Protection	Is your data safe — and do you trust the providers?	"The locksmith leaves the door open" — your data trains someone else's model, no policy governing any of it.

GRASP Evaluation Matrix

	① None	② Basic	③ Working	④ Strong	⑤ Leading
Governance	Blind No visibility. Every session starts from scratch	Aware Someone knows AI is being used; ad hoc oversight	Visible Dashboards, approval gates, cost visible per team	Policy-Driven Automated rules; budget controls and cost alerts	Autonomous Full policy engine, configurable autonomy, cost governance
Reliability	Fragile No error handling; results vary wildly	Reactive Basic error handling; manual recovery	Robust Fallbacks, retries, consistency checks	Consistent Self-heals; drift tracked; regressions caught	Resilient Full redundancy; drift detection + automated rollback
Assurance	Unproven No validation of outputs	Spot-Checked Manual review of some outputs	Fit for Purpose Systematic validation; solving the right problem	Outcome-Driven Evidence tied to business outcomes; drift corrected	Accountable Continuous measurement; efficacy vs objectives; SLAs
Scalability	Manual One use case, by hand	Team Works for one team	Multi-Use Multiple teams; some automation	Enterprise Org-wide; cost-effective at scale	Industrial Full workload; cost-per-outcome optimized
Protection	Exposed Unvetted providers; data may train models	Basic Trusted models; opt-outs; basic agreements	Policy-Based Providers vetted; data policy enforced	Defended Data scanned before leaving; prevention rules	Self-Hosted All AI internal; no data leaves perimeter

How it fits in your Enterprise

The LLM is the engine. Everything built around it determines whether it's enterprise-ready.



Applied: OpenClaw

Pillar	Score	Level	Evidence	What's Missing
Governance	1	Blind	No policy on merges. No visibility into AI-generated code. No approval gates.	Dashboard, merge policies, configurable review requirements
Reliability	2	Reactive	CI/CD runs, builds pass. No security benchmarks — same vuln patterns reappear.	Result consistency, pattern detection, quality trending
Assurance	1	Unproven	No audit trail. No FP tracking. No validation that findings lead to fixes.	Any evidence that security posture is good or improving
Scalability	2	Team	Small core team handles security manually. Growth outpaced review capacity.	Automated scanning, sustainable reviewer ratio, scaled practices
Protection	1	Exposed	Severe vulns in production. No AI contribution vetting. No supply chain threat model.	Data policy, provider vetting, downstream risk awareness
Total	**7**		out of 25 — needs 21+ for its scale	(750K weekly consumers)

Verdict: Critical Risk — Scale/Score Mismatch

Scores 7/25 but operates at industrial scale. The gap between where it is (7) and where it needs to be (21+) is the largest possible mismatch.

Skills You Need for the New World

⭐ Curiosity — the most important skill. New tools ship every day. Stay on top of what's real, what's noise, and what's coming next.

TRADITIONAL APPSEC SKILLS (STILL NEEDED)

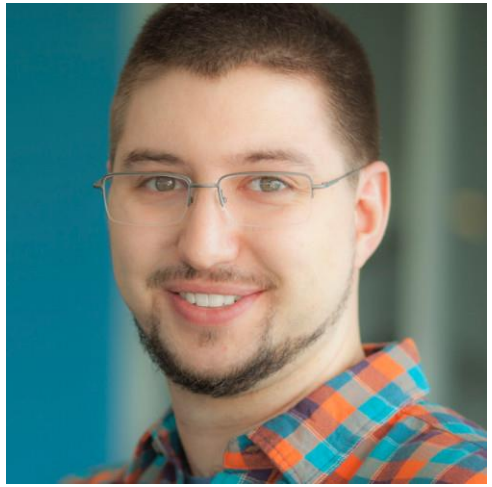
- Vulnerability analysis — vuln classes, severity, exploitability
- Code review — reading code, spotting insecure patterns
- Scanner operation — SAST, DAST, SCA tool usage
- Threat modeling — identifying attack surfaces and risk
- Compliance knowledge — PCI, SOC 2, regulatory frameworks

NEW SKILLS REQUIRED

- AI/ML literacy — how models work and how to leverage them
- Policy design — automation rules at scale
- Building – adding context to AI and building own tools
- Data and metrics — measuring outcomes, dashboards
- Solution & tech evaluation — separating hype from substance
- AI output QA — validating AI fixes, monitoring drift

Get Started – Project OASIS

Open Automated Security Initiative for Software



Chris Holt

Intigriti

15+ years in AppSec and Bug Bounty
Previously ran Intel's Bug Bounty program

🔍 **Review AI-generated security fixes** for real open-source projects

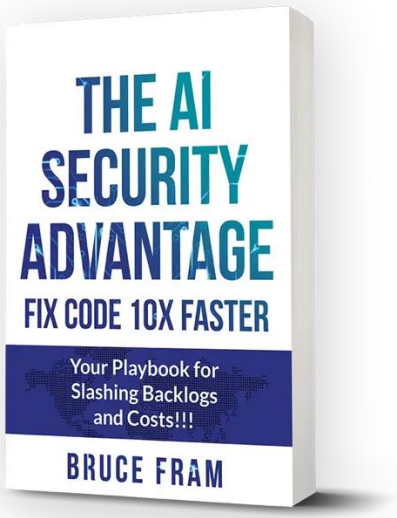
⏱️ **15 min/week** — low commitment, real impact

📤 **Your name on vetted PRs** submitted to project maintainers

🔑 **Just a GitHub account** — scan the QR code to start this week

Build your portfolio · Get published credit · Ship safer code

The Playbook You Need



Applies post-Mythos. Here's what's inside:

→ **Quantify the economics** — build the business case your CISO needs

→ **Fix the manual process** — what's broken and how to automate each step

→ **Manage up with data** — real enterprise cost numbers to justify investment

→ **Get started now** — practical advice you can apply Monday morning

See QR link or grab it before you leave

Friday Morning — Your Three Moves

Move 1 — Sign up for OASIS

Pick a project. Review one AI-generated fix. Your name on the PR.

Move 2 — Try the GRASP skill on a project

Run it against any AI tool your team uses. Scored assessment — what's strong, what's risky.

Move 3 — Download and read the book

The economics, the broken process, and how to manage up — all in one playbook.

Scan to start →

All free. All at the QR link. Start before the weekend.

One Scan — Everything You Need

📊 GRASP Framework

Five pillars for evaluating any AI system: **Governance**, **Reliability**, **Assurance**, **Scalability**, **Protection**

Slides & Deck

Full slide deck

⭐ AI Skills for GRASP

An AI Skill to evaluate projects/Repos in GRASP

appsecai.io/owasp-april-2026

💼 **AppSecAI** appsecai.io

🔗 **Bruce Fram** linkedin.com/in/brucefram

Backup

Pre-Mythos: Automated Validation is Everything

"LLMs can do anything - unreliably"

Security validation
Re-scan confirms the fix resolves the vuln

Functional equivalence
Fix doesn't break existing behavior

Quality/style alignment
Matches your coding standards

And more
Compiles, lints, passes CI gates

AppSec answers:
"Does this remediate the vulnerability safely?"

Engineering answers:
"Does this preserve expected behavior?"

Each team answers questions they can answer.