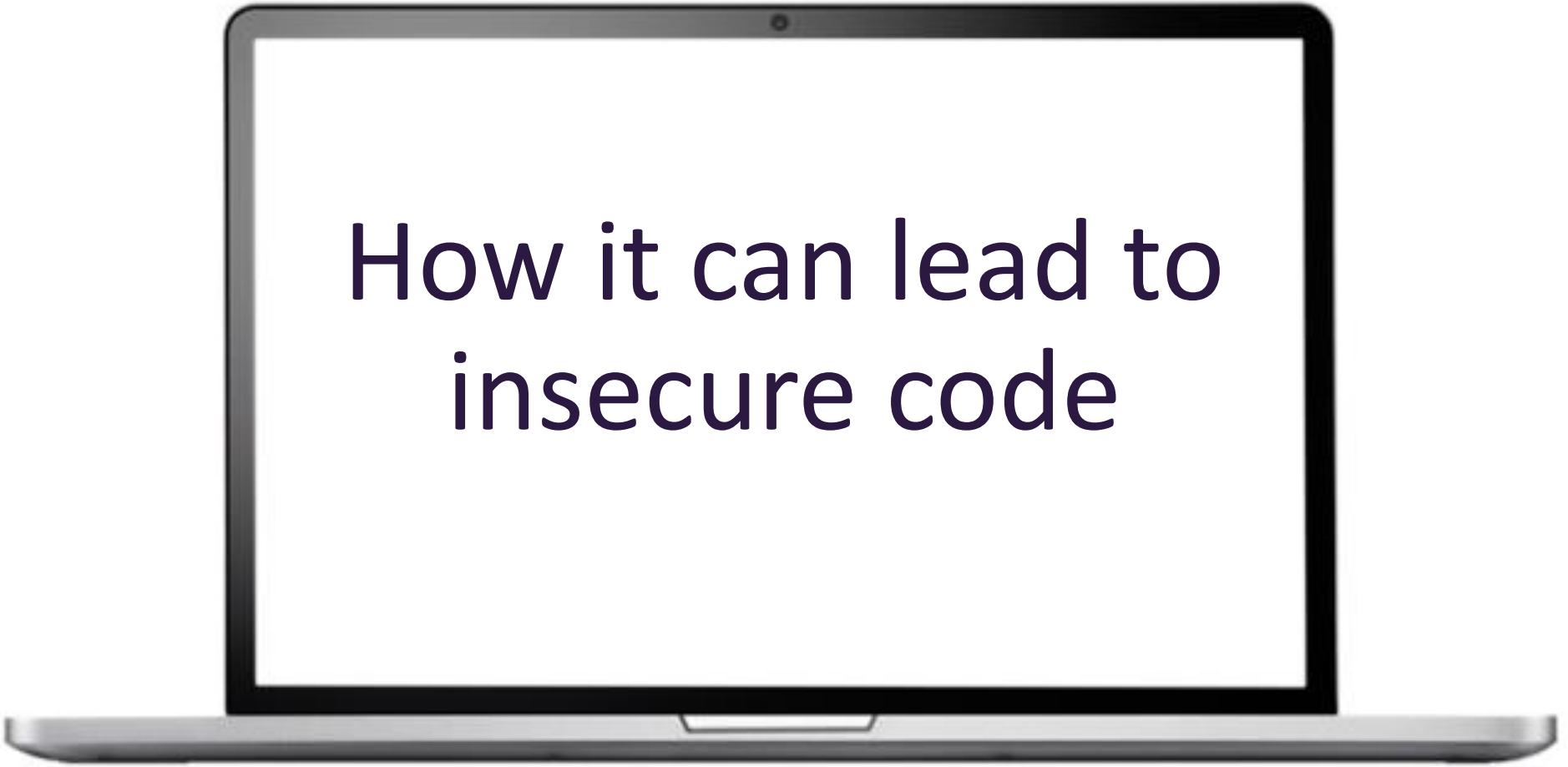
Insecure Vibes: Avoiding the Risks of AI-Assisted Coding

Tanya Janca
SheHacksPurple.ca

What are we going to talk about today?

What are we going to talk about today?



What are we going to talk about today?

The mandatory ‘about me’ slide.

Tanya Janca

- Secure Coding Trainer at SheHacksPurple Consulting
- Author: Alice and Bob Learn Secure Coding & Alice and Bob Learn Application Security
- 28+ years in tech, Sec + Dev
- Founder: We Hack Purple, OWASP DevSlop, #CyberMentoringMonday, WoSEC, DevSec Station
- Advisor: Smithy, Katilyst
- Contributor: OWASP Top Ten, StackOverflow
- Board Member: Forte Group

Agenda

1. What is Vibe Coding?
2. Why is AppSec Struggling?
3. What and How LLMs Learn
4. Real-World Threats from AI Coding
5. What We Can Do About It
6. Call to Action
7. Conclusion & Resources

What is vibe coding? Where did it come from?

Defining Vibe Coding

Writing code based on emotions, without traditional logical structure.

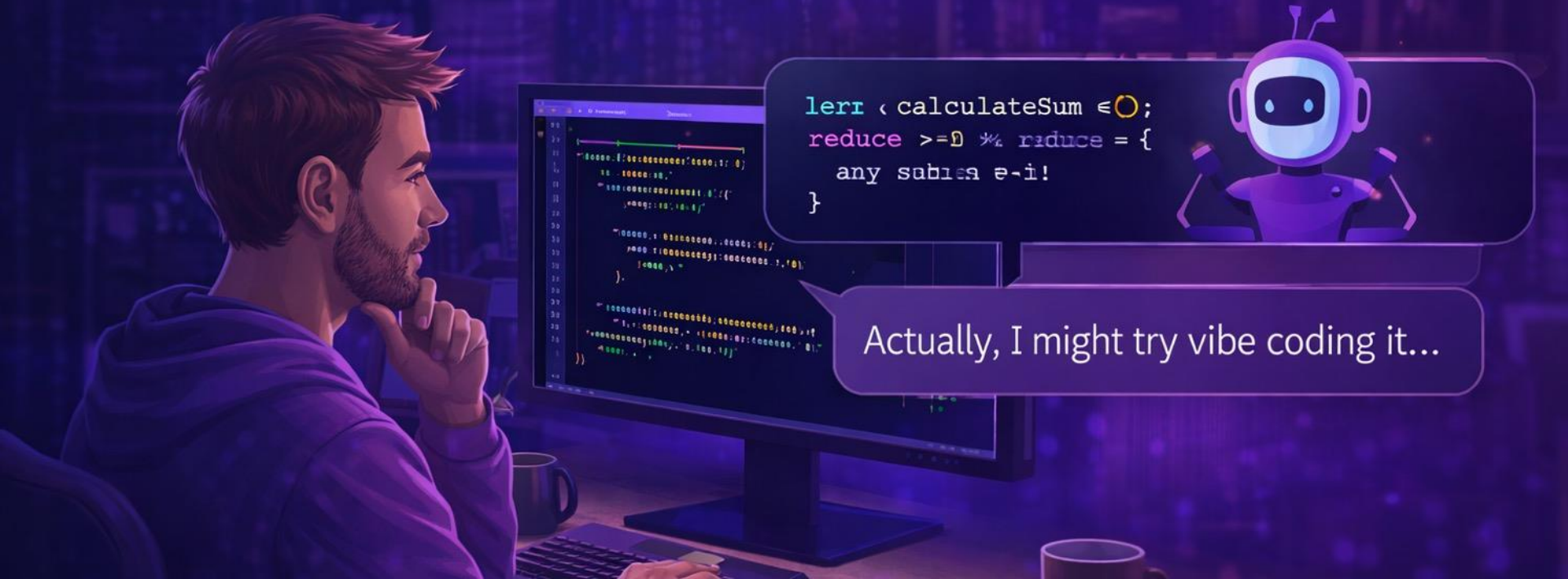
Origins of the Term

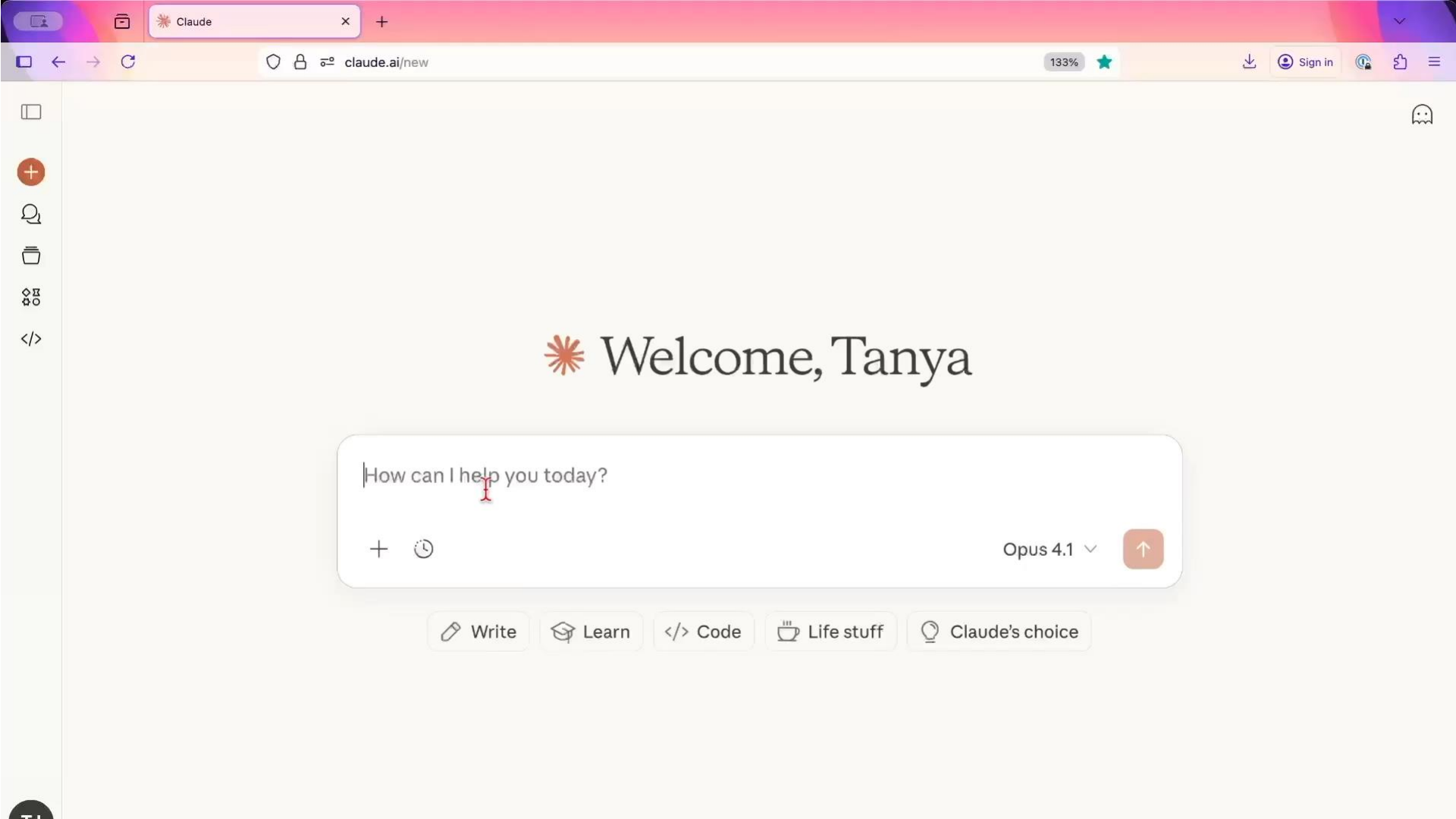
First appeared on Tiktok in 2023.

Influencers covered coding "by vibe" in videos.

Developer behavior is changing

AI assistants and vibe coding influence coding habits.



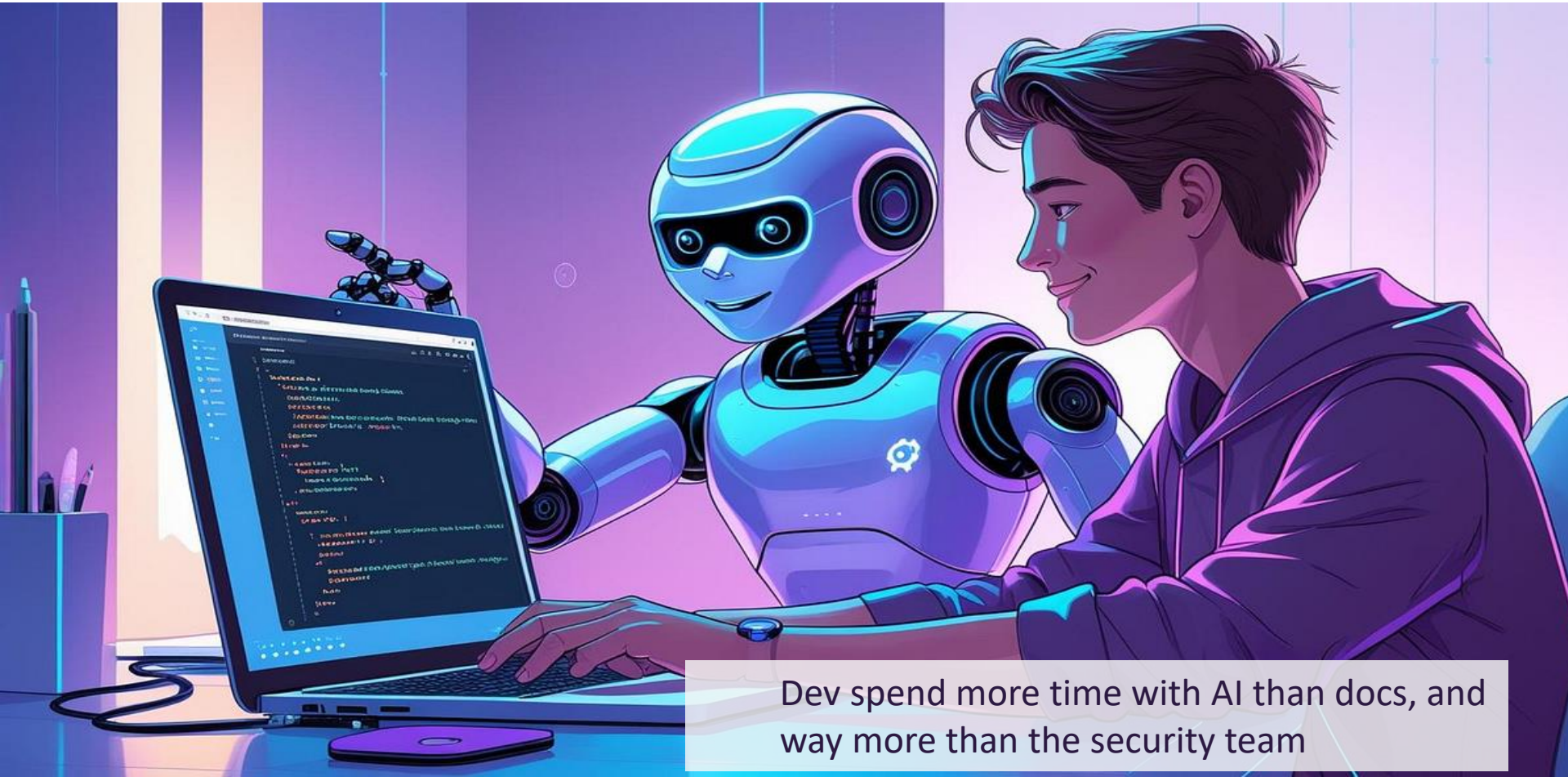

Claude
claude.ai/new
133%
Sign in
Welcome, Tanya
How can I help you today?
Opus 4.1
Write
Learn
Code
Life stuff
Claude's choice

Agenda

1. What is Vibe Coding?
2. Why is AppSec Struggling?
3. What and How LLMs Learn
4. Real-World Threats from AI Coding
5. What We Can Do About It
6. Call to Action
7. Conclusion & Resources

Demo Link ->

Why is AppSec Struggling to Keep Up?



Dev spend more time with AI than docs, and way more than the security team

Devs trust AI too much.

https://stackoverflow.co/company/press/archive/stack-overflow-2024-developer-survey-gap-between-ai-use-trust

Fast shipping incentivizes insecurity

Agenda

1. What is Vibe Coding?
2. Why is AppSec Struggling?
3. What and How LLMs Learn
4. Real-World Threats from AI Coding
5. What We Can Do About It
6. Call to Action
7. Conclusion & Resources

What LLMs Actually Learn

(and Why That's a Problem)

Models like ChatGPT and Claude are **large language models** trained to predict the next token in a sequence.

Code is treated as a language, just like English.

How AI Learns to Write Code

- Large language models predict the next token
- Code is treated like language
- Trained on massive text + code corpora
- Teaches syntax, structure, and common patterns

What that Training Actually Teaches

Patterns, not intent

Fluency, not correctness

Confidence without understanding

Why This Matters for Security

Mostly public code

Mixed quality

Security is optional in the data

The AI is working *as designed*

AI doesn't generate insecure code because developers don't care.

It generates insecure code because our ecosystem taught it that security is optional.

Agenda

1. What is Vibe Coding?
2. Why is AppSec Struggling?
3. What and How LLMs Learn
4. Real-World Threats from AI Coding
5. What We Can Do About It
6. Call to Action
7. Conclusion & Resources

Researchers Find Serious AI Bugs Exposing Meta, Nvidia, and Microsoft Inference Frameworks

Nov 14, 2025 Ravie Lakshmanan Artificial Intelligence / Vulnerability

Cybersecurity researchers have uncovered critical remote code execution vulnerabilities impacting major artificial intelligence (AI) inference engines, including those from Meta, Nvidia, Microsoft, and open-source PyTorch projects such as vLLM and SGLang.

"These vulnerabilities all traced back to the same root cause: the overlooked unsafe use of ZeroMQ (ZMQ) and Python's pickle deserialization," Oligo Security researcher Avi Lumelsky said in a report published Thursday.

— Trending News

Real-World Threats from AI Coding

I’m not even sure what to say anymore.

When I wrote the first version of this talk, I had a couple of examples, but now that the time has come there are too many disasters to count.

Let’s spend the time on fixing the problem instead of wallowing in misery, okay?

Agenda

1. What is Vibe Coding?
2. Why is AppSec Struggling?
3. What and How LLMs Learn
4. Real-World Threats from AI Coding
5. What We Can Do About It
6. Call to Action
7. Conclusion & Resources

What We Can Do About It

This is a system design problem, so the solutions must be **systemic**.

Secure guardrails for AI-assisted development

- Secure coding and privacy guardrails
- Enforced automatically
- Invisible when things go right

Prompts that **enforce** your secure coding standards

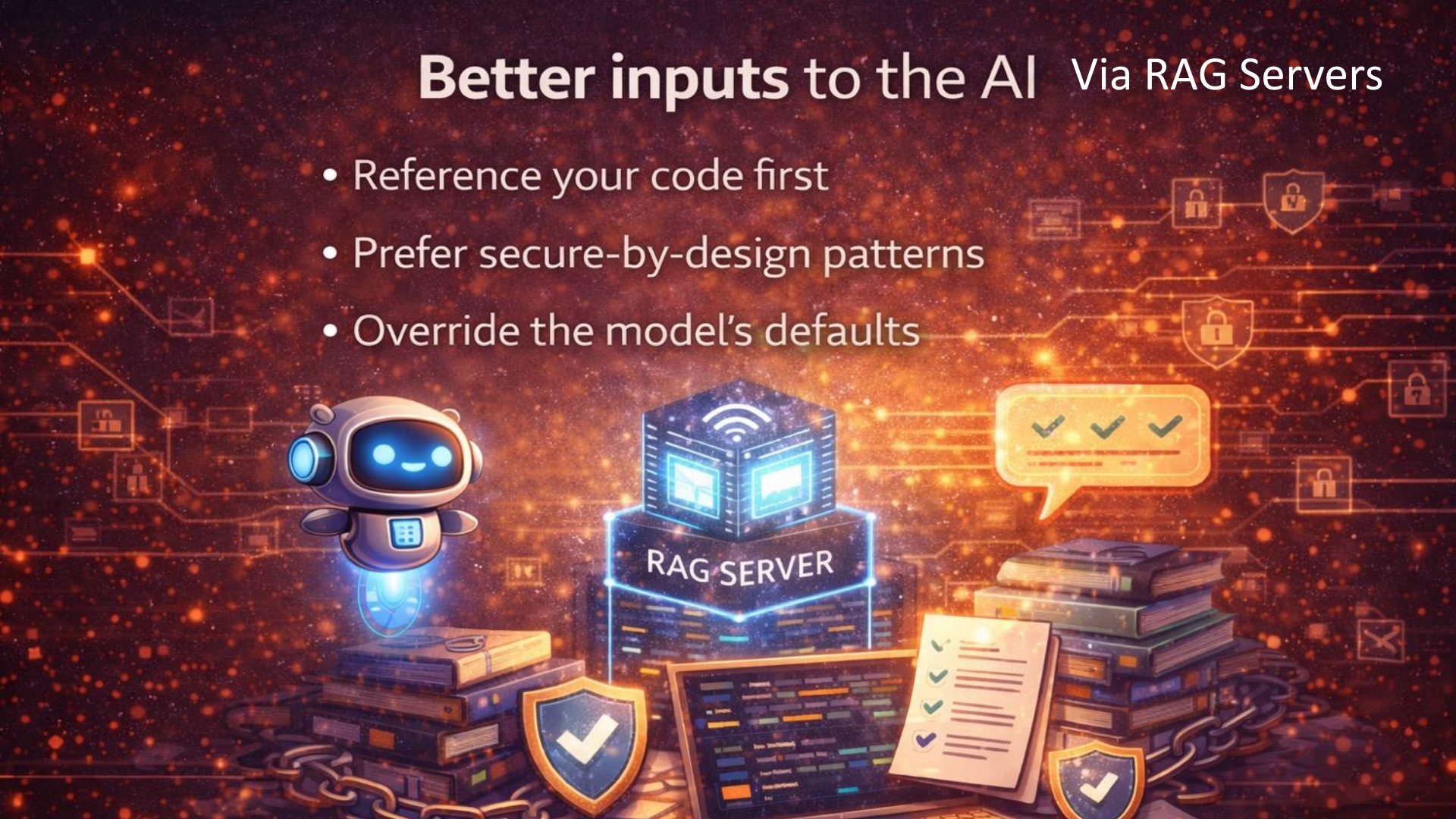
SecureMyVibe.ca

- Apply your secure coding policy
- Apply your privacy requirements
- Every time code is generated

SecureCodingGuideline.com

Better inputs to the AI Via RAG Servers

- Reference your code first
- Prefer secure-by-design patterns
- Override the model's defaults



Automated verification and feedback

MCP

- SAST, DAST, SCA, secrets, IaC
- Triggered from the IDE
- Applied consistently

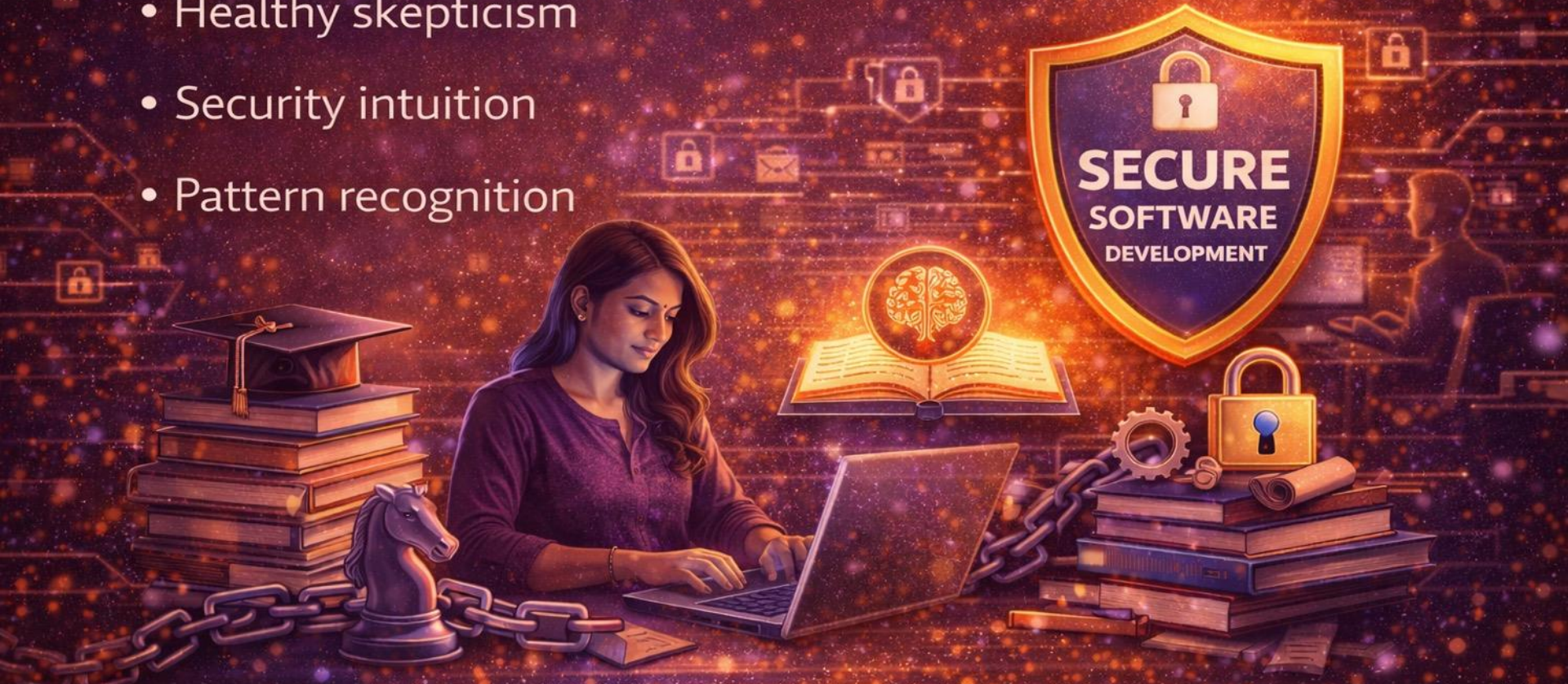
Use AI to fight AI

- Review assistance
- Anomaly detection
- Just-in-time learning

Training

Human skills and foundations

- Healthy skepticism
- Security intuition
- Pattern recognition



Don’t forget the fundamentals!

- Threat modeling
- Security requirements
- Secure SDLC
- Secure coding training

What We Can Do About It - Summary

- Secure coding and privacy guardrails for AI-assisted devs
- RAG servers with secure coding examples to reference first, above what it learned previously
- Prompts that apply your secure coding policy or standard to code generated by the AI.
- MPC servers to call SAST/DAST/Secret/IaC/SCA/etc tools from the IDE. It can also be the final application of your secure coding policy.
- Training developers to critically to review and evaluate AI code
- Use AI to fight AI: anomaly detection, review assistance, mini ‘just in time’ lessons on secure coding
- All the regular AppSec activities: threat modelling, security requirements, a secure SDLC, secure coding training, etc.

Agenda

1. What is Vibe Coding?
2. Why is AppSec Struggling?
3. What and How LLMs Learn
4. Real-World Threats from AI Coding
5. What We Can Do About It
6. Call to Action
7. Conclusion & Resources

Call to Action

What we do next matters

Adjust your SDLC for **AI-Assisted Development**

- Update your SDLC for AI realities
  - Threat modeling
  - Tooling
  - Policies

Train Developers to Evaluate AI Output

- Train developers to evaluate AI-generated code
  - Security intuition
  - Healthy skepticism
  - Pattern recognition

Provide Safe AI Options

- Give developers safe AI to use
  - Approved tools
  - Guardrails built in
  - Secure defaults

Agenda

1. What is Vibe Coding?
2. Why is AppSec Struggling?
3. What and How LLMs Learn
4. Real-World Threats from AI Coding
5. What We Can Do About It
6. Call to Action
7. Conclusion & Resources

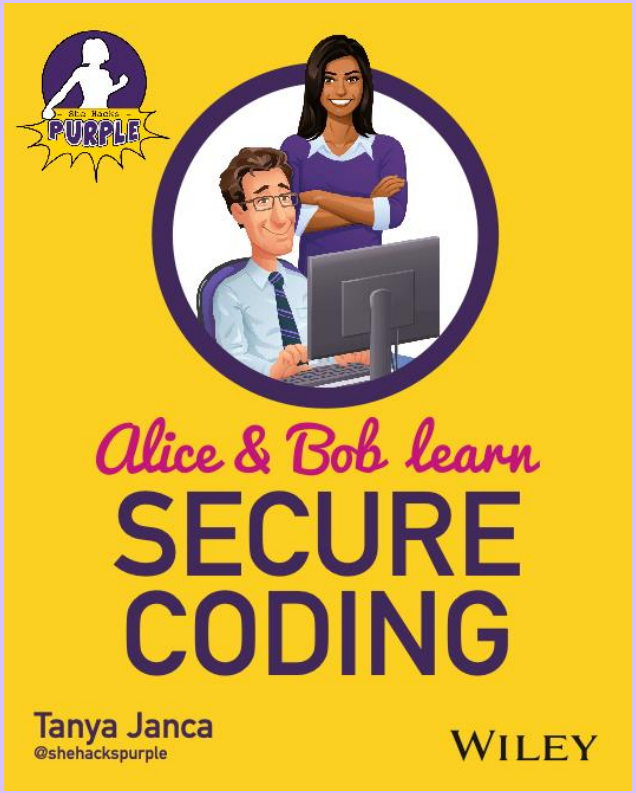
Developers aren’t the problem.

Our systems are perfectly designed to produce insecure behavior.

Let’s change the systems to set them up for success.

Resources

My books!



https://aliceandboblearn.com

JOIN THE
SHE HACKS PURPLE

Free Monthly Secure Coding Newsletter

Sign up now!

Newsletter.SheHacksPurple.ca

Sign my Petition to Secure Canada's Code!

https://links.shehackspurple.ca/e-7115

SecureMyVibe.ca

Secure Coding AI Prompts for Developers Who Refuse to Ship Insecure Code

Resources: Meeeeeeeeee!

@SheHacksPurple

Twitter/TikTok/Mastodon/GitHub/Instagram/etc.

YouTube.com/SheHacksPurple

https://SheHacksPurple.ca

https:// SheHacksPurple.ca/blog

https://Newsletter.SheHacksPurple.ca

Presentation Slides ->

https://links.shehackspurple.ca/insecurevibes

SecureMyVibe.ca

Tanya Janca

SheHacksPurple.ca

Follow me

Learn From Me

THANK YOU!

Tanya Janca

SheHacksPurple