Vibecodované aplikace báječně fungují.
Dokud nepřijde security review.

Aneb jak pomocí AI hledat zranitelnosti v kódu



Zdeněk Plachý
plachy.ai · zdenek@plachy.ai

OTÁZKA

Je kód od agenta bezpečný?

> Registroval jsem si doménu na cloudflare a prostě jsem dal Astře počítač k dispozici a řekl jsem: tady je appka na heroku, tady je koupená doména, nastav to, ať to funguje. A **za 4 minuty bylo vymalováno**. Dřív bych musel studovat, jak přesně nastavovat CNAME apod.
>
> Je to 100% vibecoding, **na kód jsem se ani nesnažil dívat nebo ho revidovat** (a to se živím jako software engineer).

Příspěvek z Facebooku 4. 11. 2026

Co by mohlo stát, že

100 % vibecoding

Kód nikdo nečetl

REÁLNÝ PŘÍBĚH #1

Faktura cizí firmy na /api/invoice/55

```
GET /api/invoice/55
Authorization: Bearer <Firma B>

HTTP/1.1 200 OK
{
  "owner": "Firma A",
  "total": 48 200,
  "items": [ ... ]
}
```

- [Authorize] na endpointu
- CORS a throttling
- 400 / 403 / 429, vše otestováno
- JWT s expirací, HTTPS only

Co agentovi neřeknete, to neudělá. Multi-tenant patří do zadání.

REÁLNÝ PŘÍBĚH #2

Admin pro každého

Nepřihlášený viděl celou strukturu administrace.

Viděl jsem stejný případ dvakrát: v Blazoru i v Next.js.



Kde komponenta poběží, agent neví. Řekněte mu to.

PROČ DÍRY VZNIKAJÍ

Může za to člověk: prostě to neřekl



Žádný test, prostě „funguje to“



AI jde nejkratší cestou



AI nezná váš funkční, ani threat model



Znejte svou technologii

ÚROVEŇ 1 · VÍKENDOVÝ VIBEKODÉR

Alespoň mu řekněte: /security-review

```
> /security-review

> Kontext: aplikace je multi-tenant,
  každý dotaz filtruj podle TenantId.
  Admin je pod /admin, layout i stránky.

> Ke každému nálezu napiš test,
  který útok předvede.
```



Před každým nasazením

minuta práce, nic navíc



Řekněte mu, co je appka zač

multi-tenant, API, granularita



Nález = test, ověření, regrese

AI nemusí pochopit kontext

ÚROVEŇ 2 · POKROČILÝ

Virtuální testovací tým z agentů

Analytik

API endpointy, oprávnění, přihlášení…

Auditor

jedna třída chyb na jeden běh

Ověřovatel

test, který selže, dokud díra existuje

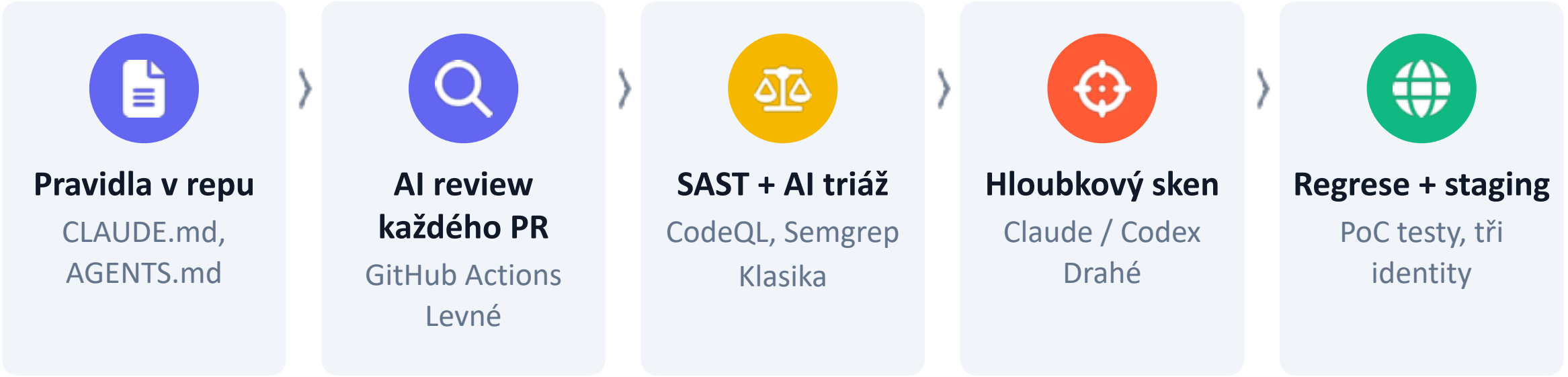
Inženýr

navrhuje a aplikuje fix

Jiný výrobce než autor kódu: Claude psal, Codex audituje a vice versa.

ÚROVEŇ 3 · FIREMNÍ TÝM

Klasický security audit, jen s LLM



Nálezy triáduje člověk. AI je výborný lovec, ale občas špatný soudce.

Co si odnést

Zdeněk Plachý
plachy.ai · zdenek@plachy.ai



1 **Testujte s LLM: pečlivé, neúnavné, levné, komplexní.**

2 **Testujte opakovaně, ne jednou před releasem.**

3 **Při vývoji říkejte agentovi, co je vaše aplikace zač.**

Zítra: /security-review a jedna věta: „aplikace je multi-tenant“.

Díky.
Otázky?