

# AI가 생성한 코드로 오픈소스 기여해도 되나요?

오픈소스 기여에서 생각해볼 AI 생성 코드

---

이서연  
LINE+



# CONTENTS

- 01 AI 생성물에 대한 오픈소스 커뮤니티의 반응
- 02 주요 쟁점
- 03 그래서, AI 생성물로 오픈소스 기여해도 되나요?





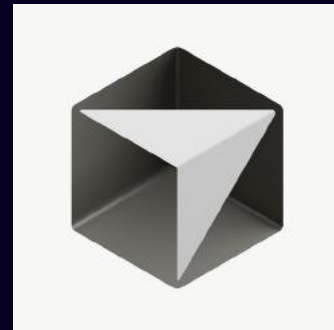
# 01

## AI 생성물에 대한 오픈소스 커뮤니티의 반응



# AI 생성물은 이제 일상적으로 만들게 되었습니다.

- 코드 생성 도구의 급성장으로 개발 속도 증가
- 그대로 오픈소스에 포함시켜도 괜찮을까요?



코드 생성 도구 3대장:  
GitHub Copilot, Claude, Cursor



# DEATH BY A THOUSAND SLOPS

- curl 프로젝트의 “AI slop” 대응
  - 품질 낮은 AI 생성 콘텐츠를 지적
    - 특히 버그 바운티 프로그램을 통해 들어오는 AI 생성 보안 보고서로 인해 메인테이너들이 불필요한 에너지를 쓰고 있음
  - AI로 의심되는 보안취약점 보고서를 즉시 차단 조치를 시행
- AI 사용시 투명성과 철저한 검증을 의무화

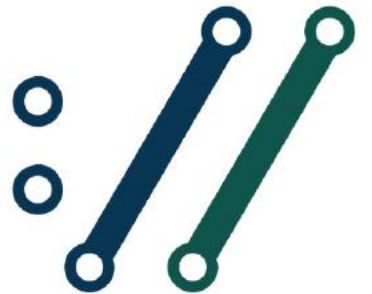
CURL AND LIBCURL

## DEATH BY A THOUSAND SLOPS

🕒 JULY 14, 2025 👤 DANIEL STENBERG 💬 58 COMMENTS

I have previously blogged about the relatively new trend of [AI slop](#) in vulnerability reports submitted to curl and how it hurts and exhausts us.

This trend does not seem to slow down. On the contrary, it seems that we have recently not only received more AI slop but also more *human slop*. The latter differs only in the way that we cannot immediately tell that an AI made it, even though we many times still suspect it. The net effect is the same.



(source: <https://daniel.haxx.se/blog/2025/07/14/death-by-a-thousand-slops/>)



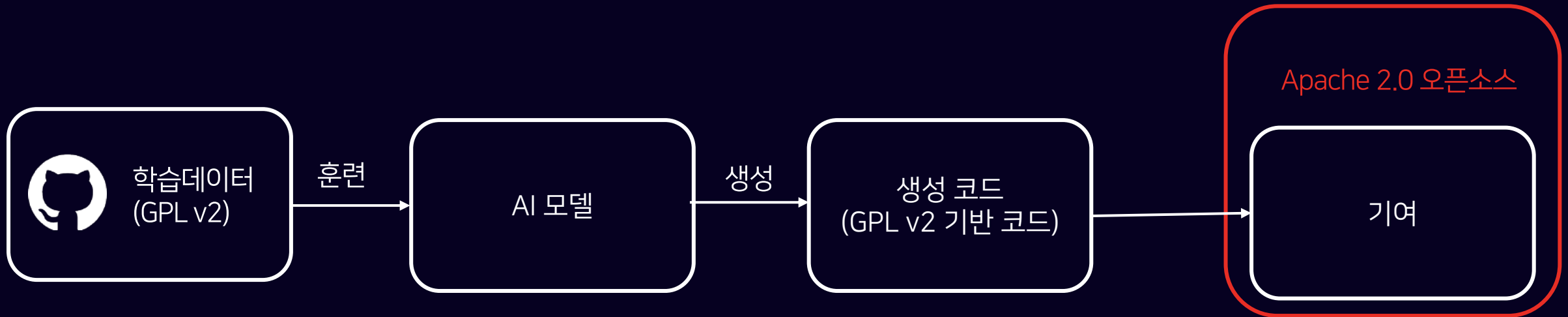
02

주요 쟁점

## 02

# 3가지 주요 쟁점 - 1. 라이선스 호환성

- AI 모델이 학습한 데이터에는 다양한 라이선스의 코드가 포함되어 있습니다.
- 오픈소스 프로젝트의 라이선스와 호환되지 않는 라이선스가 혼입될 가능성이 있습니다.



## 02

# 3가지 주요 쟁점 - 2. 신뢰성 문제

- 기여자가 AI의 사용 사실을 알리지 않으면, 커뮤니티는 코드의 출처를 알 수 없습니다.
  - 만약, 오픈소스 이용자가 AI 생성 코드를 이용하기를 원치 않는다면?
  - 만약, AI 생성 코드에 보안 취약점이 있다면?
  - 만약, AI 생성 코드에 논리적 오류가 있다면?



- 이렇게 계속 서로 의심만 쌓여간다면...?





## 3가지 주요 쟁점 - 3. 기여자 자격 문제

- 저작권이 성립하지 않는 산출물에 대하여, CLA/DCO가 유효할까요?

“You” (or “Your”) shall mean the copyright owner or legal entity



Apache ICLA

The contribution was created in whole or in part by me and I have the right to submit it



Linux Foundation DCO



# 03

**그래서, AI 생성물로 오픈소스 기여해도 되나요?**

# 03

## 커뮤니티의 대응

### The future of AI in OSS projects:

**Human-in-the-loop:** AI proposes; maintainers decide. Require review for AI-generated code.

**Provenance & licensing:** Log AI assistance, check license compatibility, retain attribution.

**Security by design:** Treat AI suggestions like untrusted input—scan, test, and sandbox.

**Project conventions:** Codify style/architecture rules so AI can enforce them consistently.

**Transparency to contributors:** Document how AI is used (and limited) in CONTRIBUTING.md.



49



THE LINUX FOUNDATION

OPEN  
SOURCE  
SUMMIT  
EUROPE

(source: Open Source Summit Europe 2025 - Keynote: State of the Union)



피할 수 없다면 즐겨야겠죠.

언제나 그랬듯이. 투명하게.

투명성 = 신뢰와 협력의 기반



피할 수 없다면 즐겨야겠죠.

언제나 그랬듯이. 투명하게.

투명성 = 신뢰와 협력의 기반

✓ 어떤 AI 모델을 썼는가?

✓ 인간의 개입은 어느정도?

✓ 인간의 확인은 거쳤는가?

# 감사합니다.

AI가 생성한 코드로 오픈소스 기여해도 되나요?

