버그 없는 Rust 컴파일러를 위하여

변지석
고려대학교 소프트웨어분석연구실

2026. 08. 26
SIGPL @ 한국문화연수원

Rust 프로그래밍 언어

- **메모리 오류 / 순서 충돌(data race) 오류를 타입으로 잡아내는 프로그래밍 언어**
  - Rust 컴파일러는 세상에서 가장 유명한 메모리 오류 / 순서충돌 분석도구

주요 소프트웨어 기업들이 Rust 사용

Microsoft

Dropbox

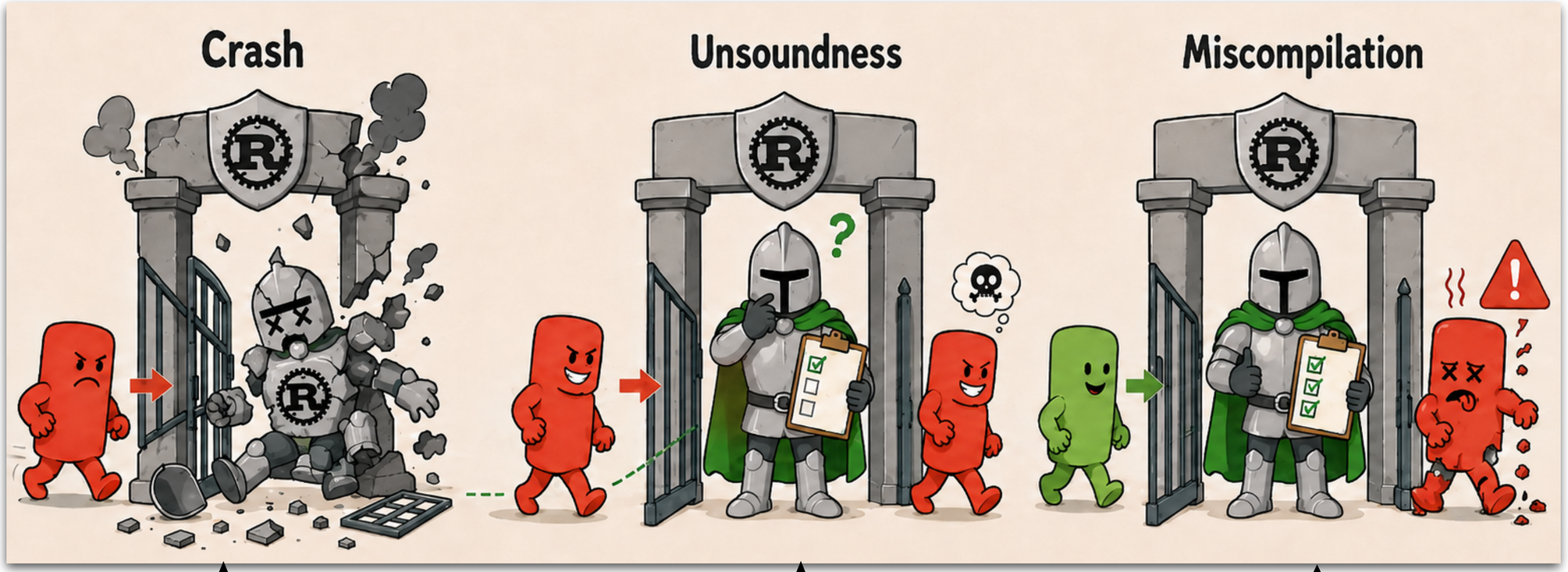
CLOUDFLARE

Discord

기존 C 코드를 Rust 로 옮기려는 연구 진행됨



Rust 컴파일러에는 버그가 많다



고장 이슈 **604**개

안전성 이슈 **75**개

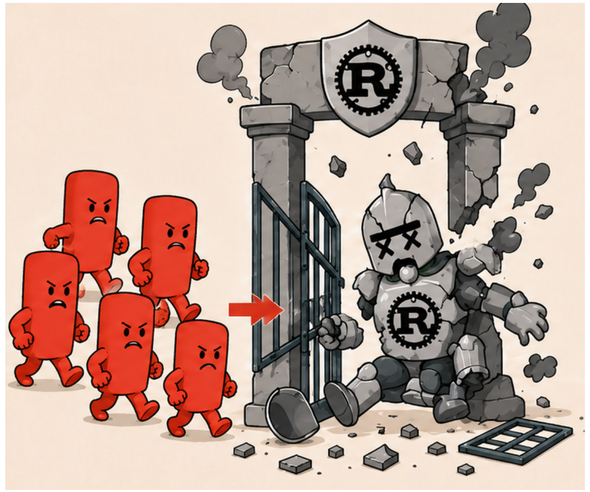
컴파일 실수 **30**개

* 2026년에 생성된 이슈 수

이전 연구: 크래시 버그 잘 찾는 마구실행기(fuzzer)

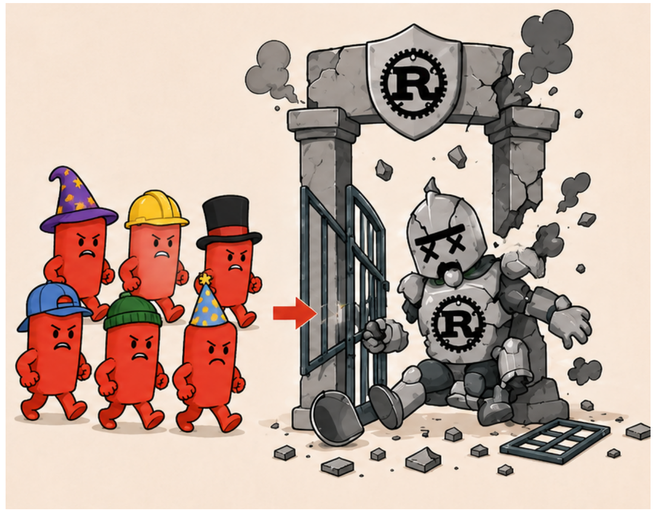
기존의 마구실행기

마구생성할 때
재료코드의 컴파일 맥락
무시

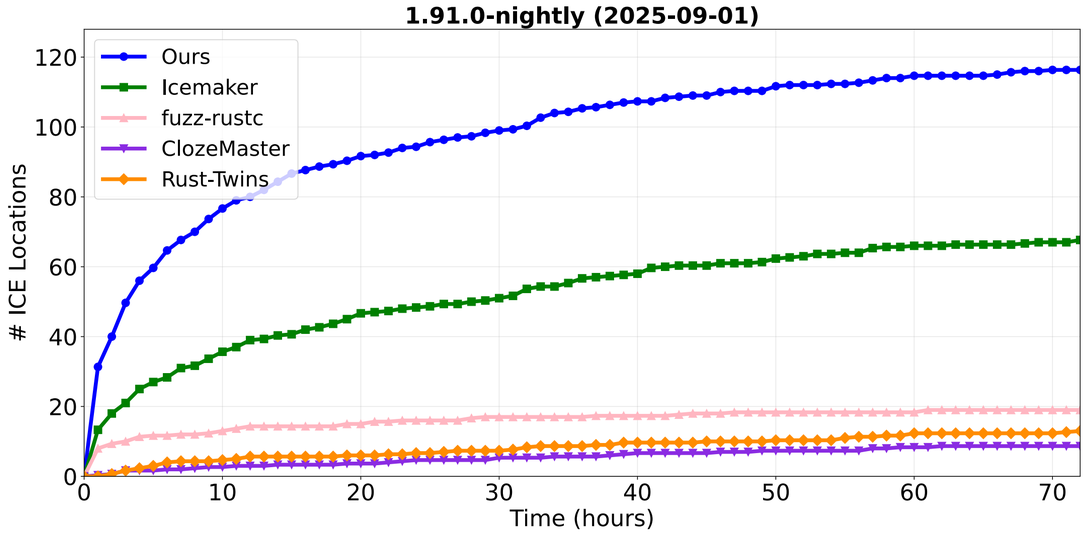


우리의 새 마구실행기

마구생성할 때
재료코드의 컴파일 맥락
활용



버그를 더 빠르고 다양하게 찾음

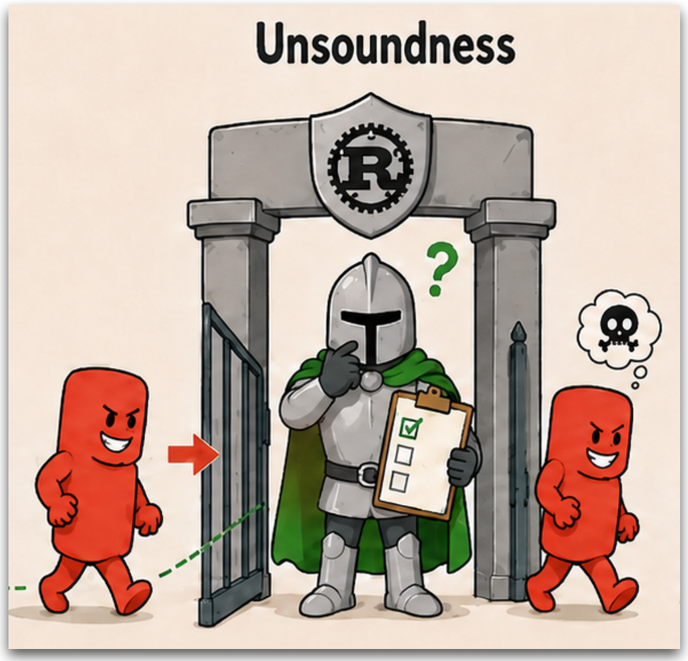


새 버그 109개 찾아 제보했고,

그 중 61개는 Rust 컴파일러
공식 테스트케이스에 추가됨

진행중인 연구: 불안전한 버그 잘 찾기

불안전 버그가 있으면
모르는 동작 있는 프로그램이
컴파일에 성공



최신 버그 코드*

```
1.  #![forbid(unsafe_code)]
2.  fn main() {
3.      let mut a = (Some(&42u64), 0u8);
4.      let mut b = (None::<&u64>, 0u8);
5.      let mut p = &mut a;
6.      match p.0 {
7.          Some(_) if { p.1 = 1; p = &mut b; false }
                                        => unreachable!(),
8.          Some(r) => println!("{:?}", p.0),
9.          None => unreachable!(),
10.     }
11. }
```

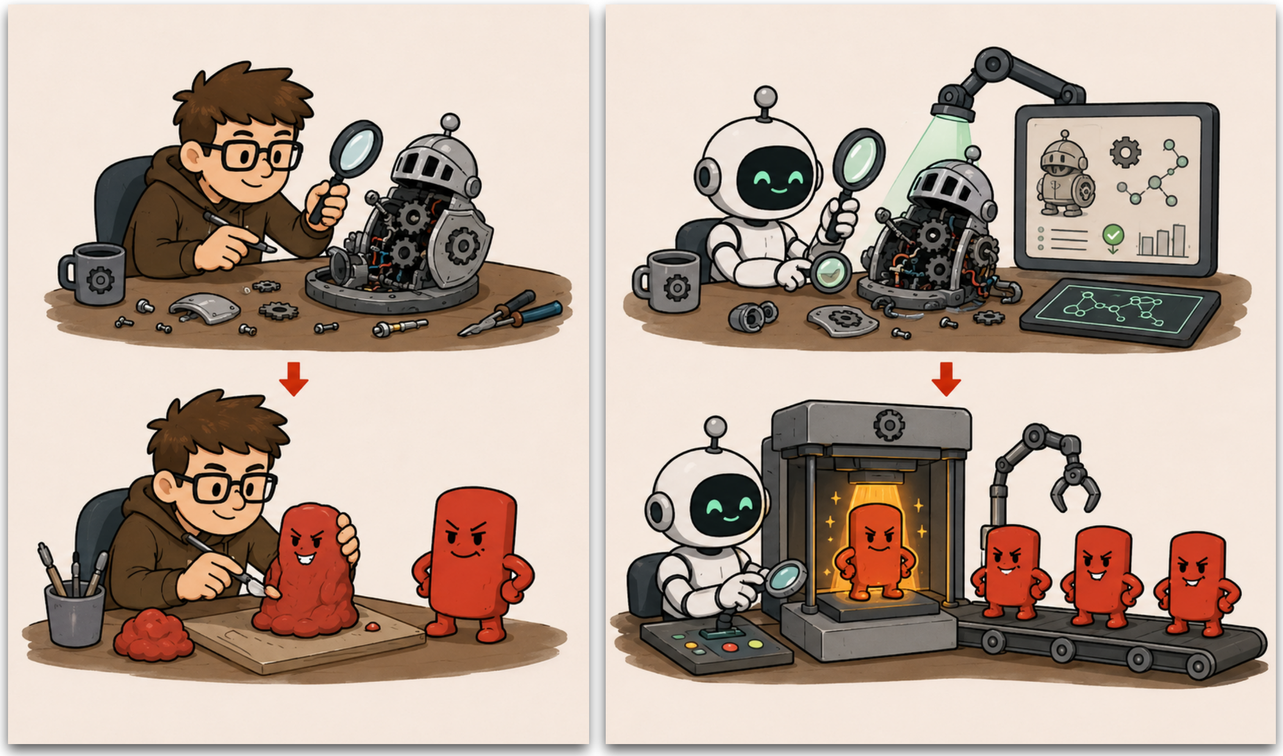
p.0 값 패턴매칭 결과: "**Some(r)**"
p.0 값 출력 결과: "**None**"

* #160599

진행중인 연구: 불안전한 버그 잘 찾기

아직 CPU-자동화된 테스트 방법은 없음

사람이 직접 설계하거나 AI 전문요원 활용

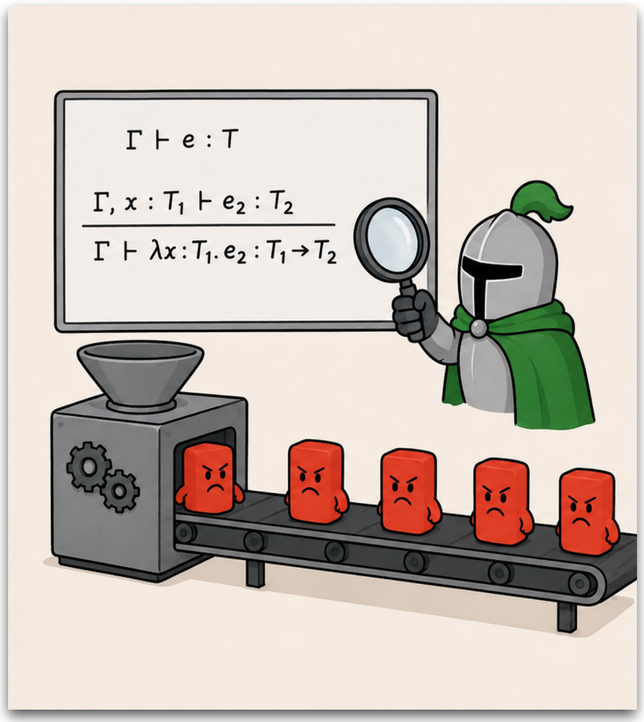


Antropic 수석 보안 연구원 Nicholas Carlini 가

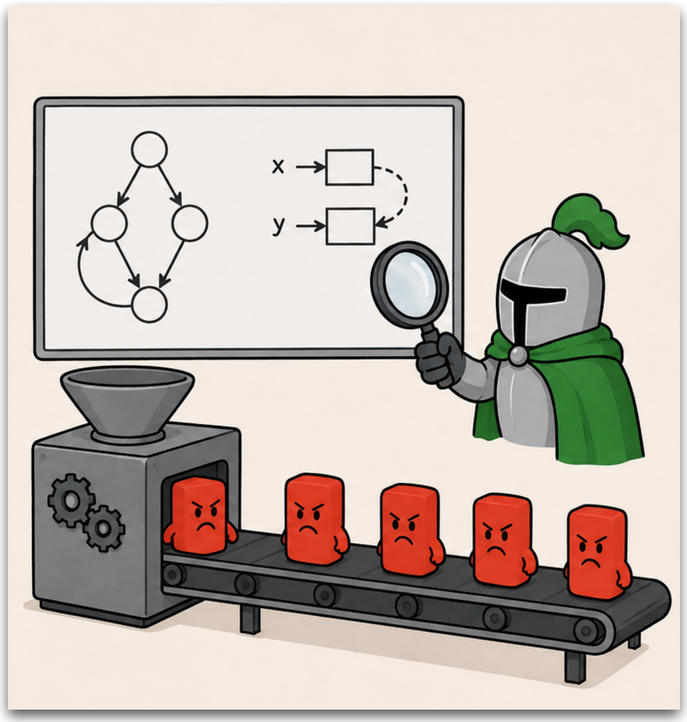
최근 한 달 동안 10개의 안전성 버그 보고

진행중인 연구: 불안전한 버그 잘 찾기

타입 시스템 구현 테스트하기



Rust 의 복잡한 실행흐름과 메모리 사용패턴들을 테스트



AI-전문요원 에게 테스트 맡겨놓기

