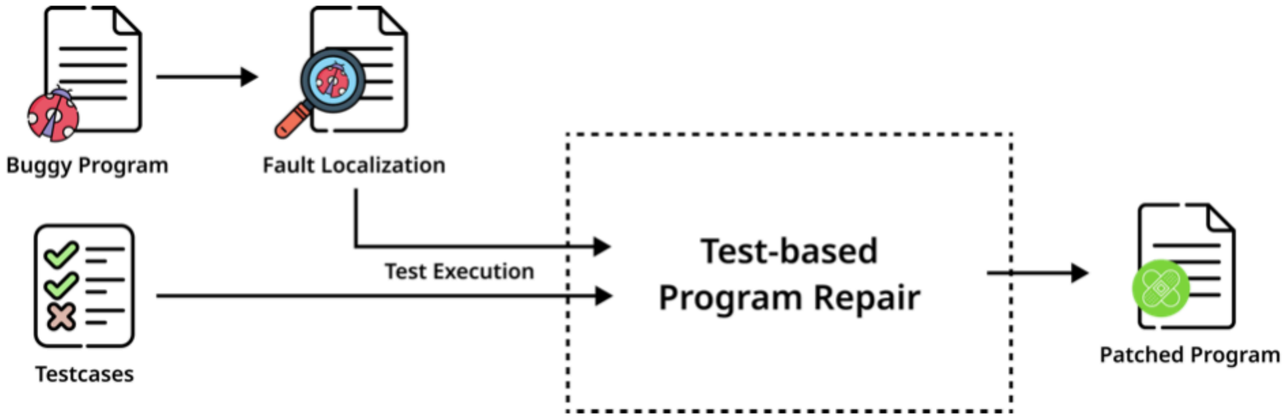
일반화를 보장하는 PBE 기반 프로그램 수정

한양대학교 프로그래밍시스템연구실 이제형
ERC 여름 워크샵
2023년 7월 3일

프로그램 오류 자동 수정 문제

- 오류가 포함된 프로그램을 **자동으로 수정**하는 기술
- 테스트 기반 오류 수정
  - **1개 이상의 실패하는 테스트**가 포함된 테스트케이스와 **버그가 있는 코드** 받아 **모든 테스트가 통과하는 코드**로 수정
  - 기존 연구 : FAngelix (IST'22), VulnFix (ISSTA'22), CPR (PLDI'21), ...



그 패치가 올바른 거 맞아?

- 모든 테스트를 통과하는 패치 코드는 여러개 존재할 수 있음
  - **여러개의 그럴듯한 패치 중에서 어떻게 올바른 패치를 고를까?**
  - 기존의 프로그램 수정 기법들 : **다양한 휴리스틱 사용** (원본 식과 비슷하게, 신경망 기반 등..)
- 프로그램 수정 도구들은 결과로 패치 코드를 제공하지만,

  제공된 패치가 올바른 지에 대한 확신을 줄 수는 없음

프로그램 수정 도구

테스트 통과하는 패치들이에요!

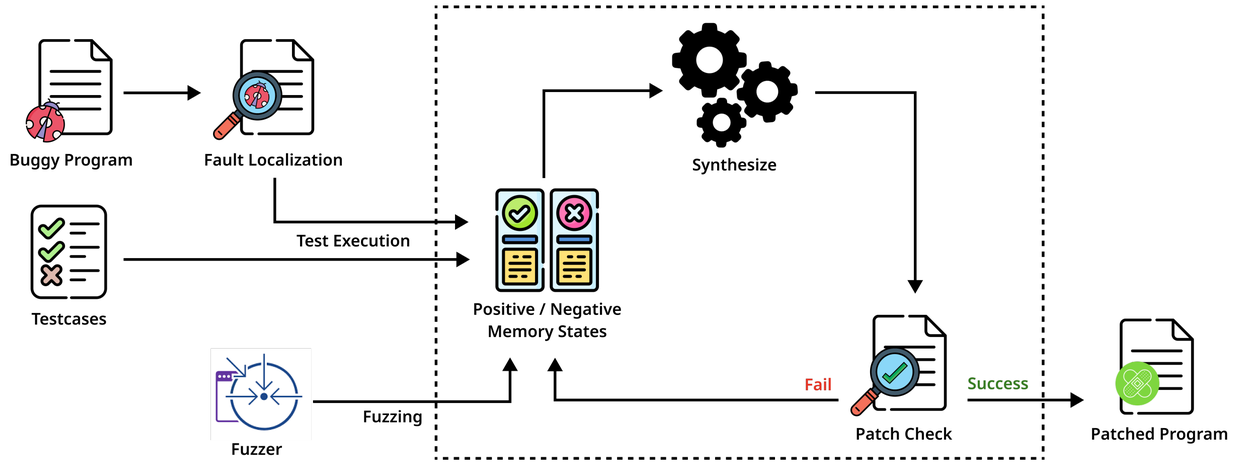
`k = a*b, k = a*2, k = a*b, ...`

사용자

패치가 너어어어무 많아..
어느게 맞는거야?

Crash Fix에서 과적합 : VulnFix 소개

- VulnFix : **Crash가 나는 테스트를 수정**해주는 프로그램 수정 도구
  - Crash가 나지 않는 바람직한 메모리 상태와 Crash가 발생하는 좋지 않은 **메모리 상태를 구분하는 Invariant를 합성하는 방식**으로 프로그램 수정
  - 주어진 테스트 케이스 외에 Fuzzer를 이용하여 Crash가 발생하는 입력을 찾아서 합성에 사용



Crash Fix에서 과적합 : VulnFix에서의 예

- libtiff-CVE-2016-10094
  - 매우 간단한 연산자 패치
  - 1시간의 Fuzzing으로 얻은 입력으로도 **36개의 후보 패치를 얻어 이들 중에 올바른 패치가 있는지 알 수 없음!**
- **후보 패치가 여러 개일 경우**, VulnFix는 올바른 패치를 고를 능력이 없어 **패치 생성 실패로 처리**
  (실험 결과 중 약 16%의 패치가 여러 개의 후보 패치로 인해 실패)

```
...
   if(TIFFGetField(input, ..., &count, &jpt) != 0) {
-      if (count >= 4) {
...
```

```
...
   if(TIFFGetField(input, ..., &count, &jpt) != 0) {
+      if (count > 4) {
...
```

libtiff-CVE-2016-10094

VulnFix

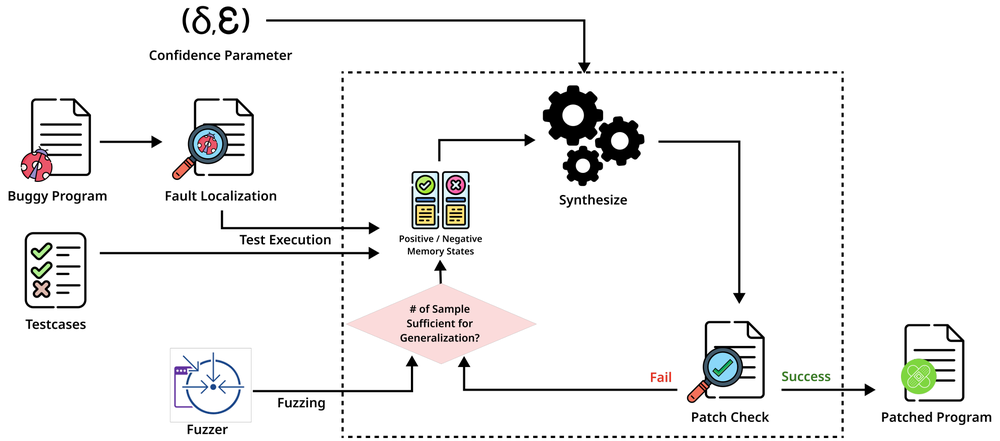
```
'_GSize_input->tif_name == 1164',
't2p->tiff_datasize > t2p->tiff_length',
'*input->tif_dir.td_stripbytecount >= 3',
...
```

36개의 후보 패치!

우리의 기법

- PAC Learning을 이용하여 **일반화된 패치를 합성하는 데 필요한 메모리 상태의 수를 계산**
- Fuzzer를 이용해서 **필요한 만큼의 메모리 상태를 확보하여 합성 → 일반화를 보장하는 패치 합성**
- PAC Learning
  - 어떤 가설 공간 (정답 함수가 될 수 있는 함수들의 공간) H에서 **에러 확률이 ϵ보다 낮을 확률이 δ보다 낮음을 보장**하기 위해서 얼마나 많은 샘플이 필요한지 계산 하는 이론
  - 패치를 좋은 메모리 상태와 나쁜 메모리 상태를 구분하는 일종의 함수로 보면,

    사용자가 제공한 ϵ,δ에 대해서 필요한 샘플 (메모리 상태의 수)를 계산 가능



요약

- 기존의 프로그램 수정 도구들은 **패치의 올바름을 보장해주는 기법이 없었음**
  - 결국 패치가 자동 생성되어도 개발자는 수동으로 패치를 검토해야 함
- PAC Learning 이라는 이론적 도구를 활용하여 패치의 올바름에 대한 **확신을 주기 위해 필요한 메모리 상태의 수를 계산** 가능
- 이를 이용해 Crash를 자동으로 수정하는 **올바른 패치를 생성하는 실험을 진행할 예정**

감사합니다