

톰슨 샘플링 기반 지향성 협력 퍼저

모현민, 김윤희

한양대학교 컴퓨터소프트웨어학부
소프트웨어공학연구실



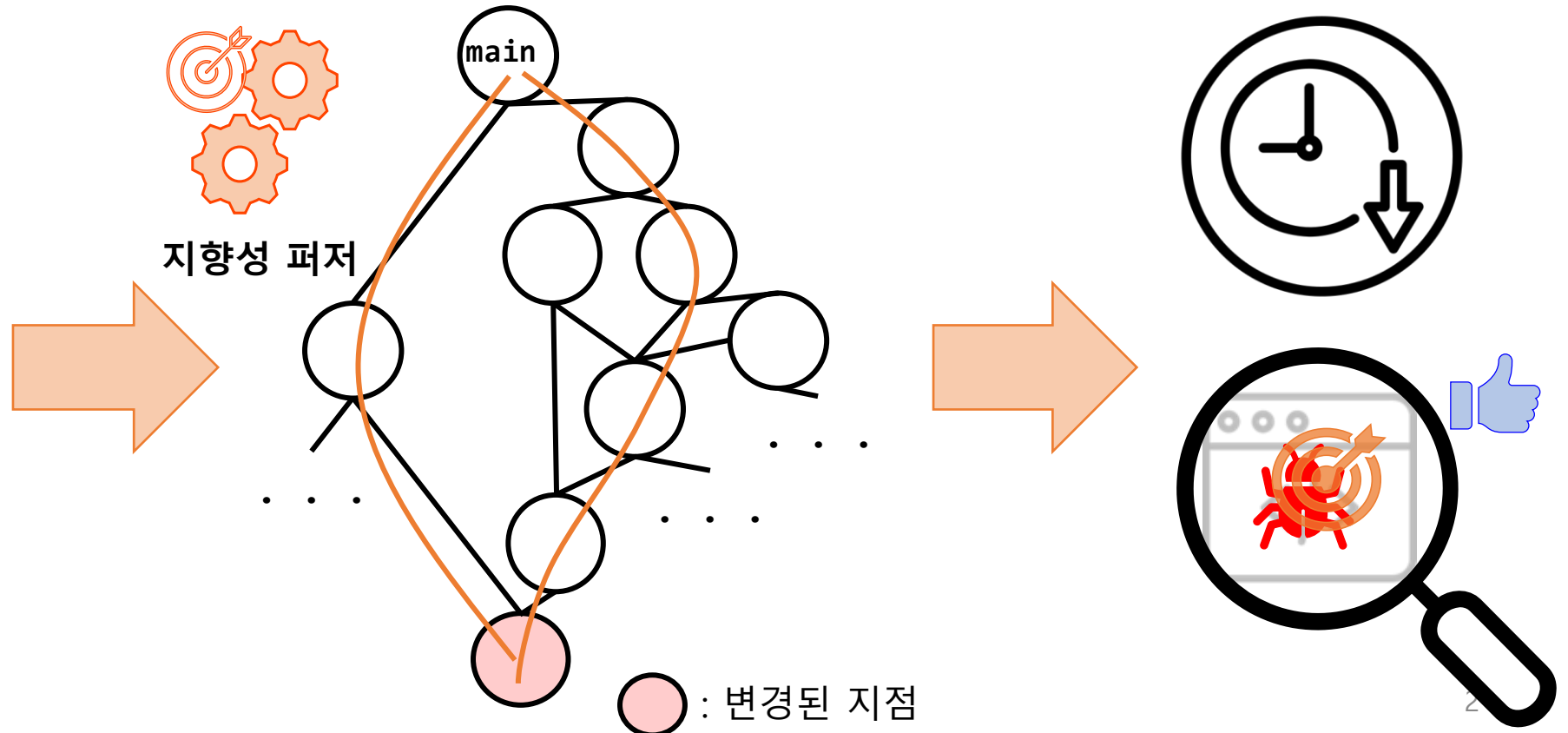
소프트웨어재난연구센터

지향성 퍼징

- 특정 목표 지점에 빠르게 도달하고 취약점을 재현할 수 있도록 탐색을 유도하여 입력을 자동 생성하는 퍼징 기법
 - 패치나 업데이트 등으로 변경된 지점에 빠르게 도달하여 오류 여부 검증하는 데 효과적임.



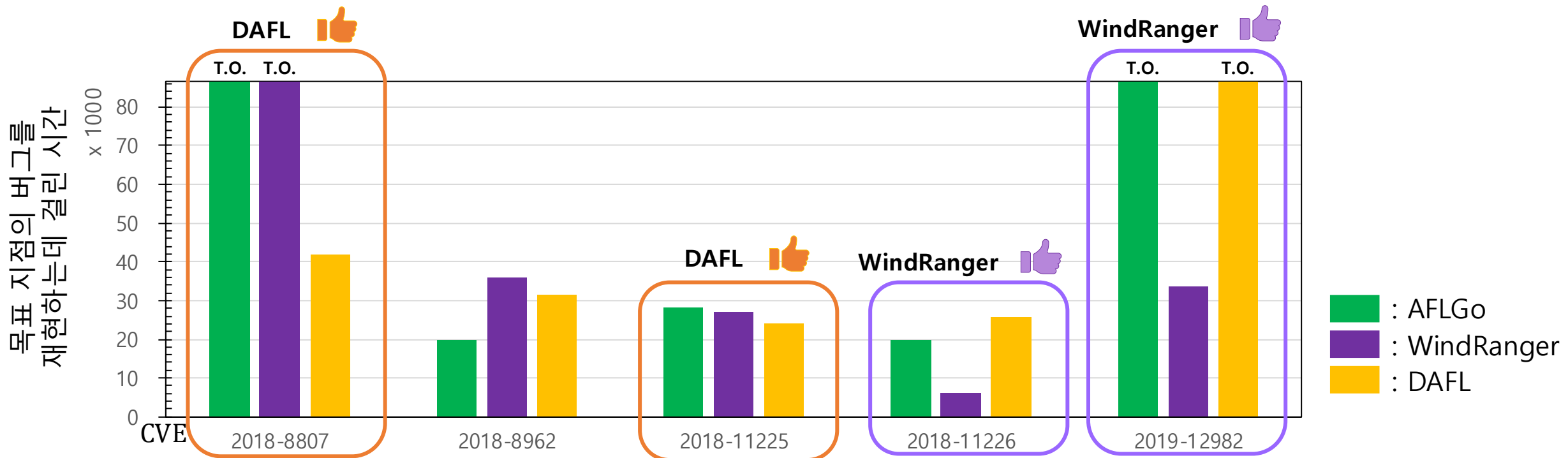
패치나 업데이트된
프로그램



단일 지향성 퍼저의 한계

- 같은 프로그램에서도 목표 지점마다 각 지향성 퍼저의 결과가 다르기 때문에 어떤 지향성 퍼저를 사용하는 것이 좋을 지 알기 어려움.
 - swftophp-2018-8807 과 2018-11225 에서는 **DAFL** 이 가장 성능이 좋음.
 - swftophp-2018-11226 과 2019-12982 에서는 **WindRanger** 가 가장 성능이 좋음.

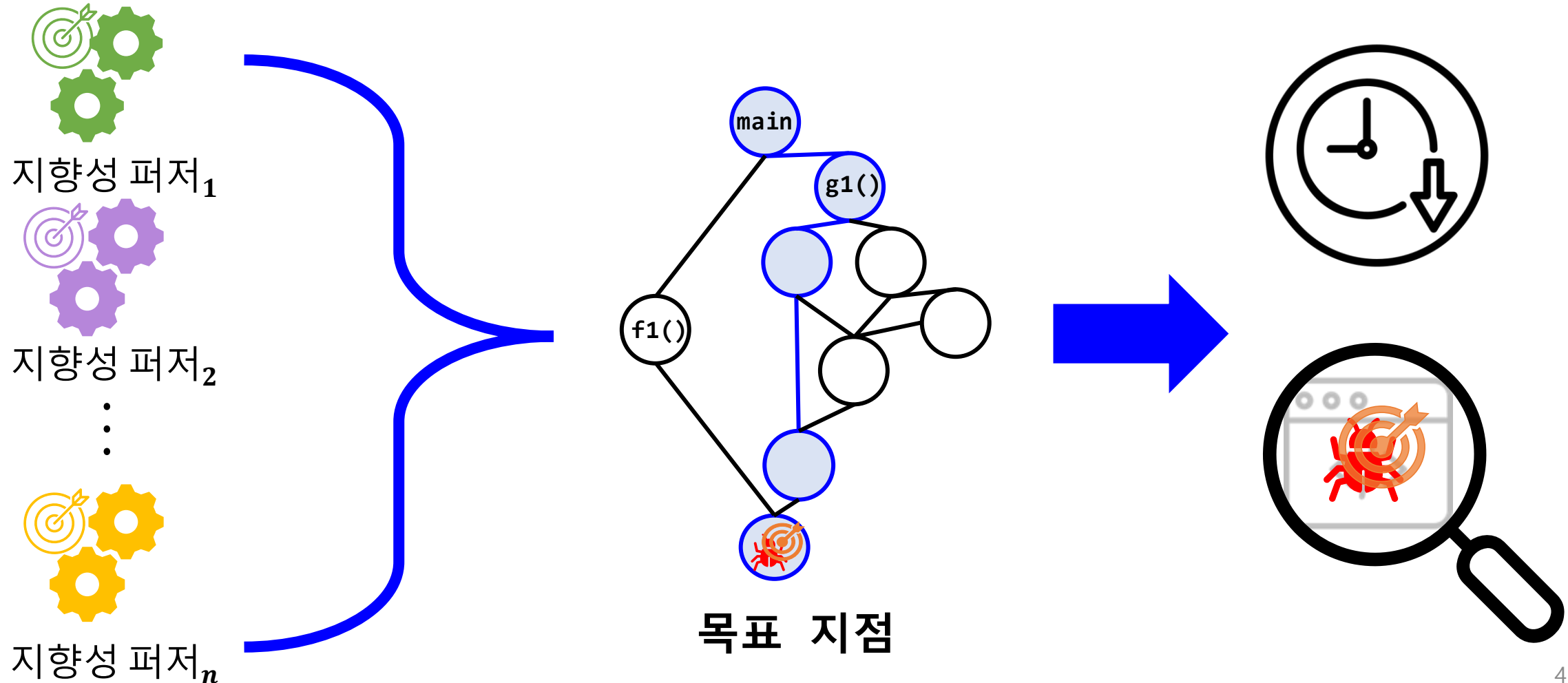
Ex) swftophp 프로젝트의 목표 지점에 대한 각 퍼저의 TTE 결과¹



❖ ¹각 지향성 퍼저를 24시간 씩 10회 실행하여 평균 TTE 측정한 결과

지향성 협력 퍼저

- 서로 다른 **지향성** 퍼저들의 전략 강점을 조합하여 **목표 지점**에 빠르게 도달하고 취약점을 재현하기 위해 협력하는 퍼징 기법



도전 과제

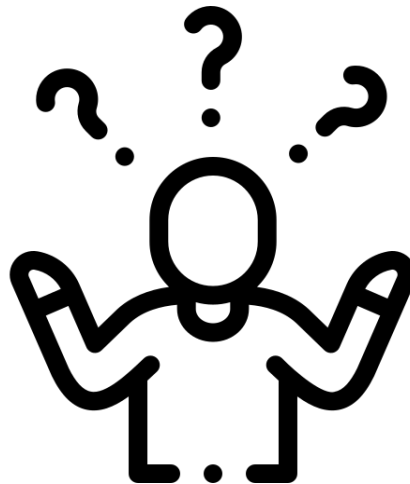
- 공통된 평가 기준 부재

- 목표 지점 도달 이전에 어떤 지향성 퍼저가 목표 지점에 더 가깝게 도달했는지 판단하기 어려움.

WindRanger

AFLGo

DAFL

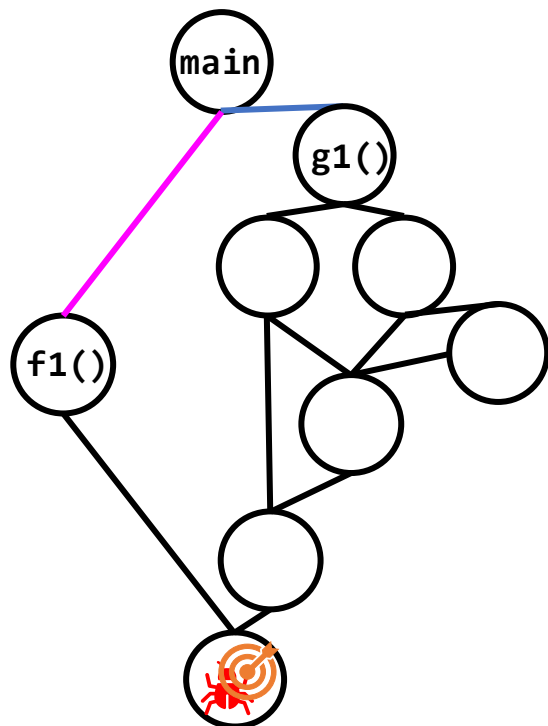


공통된 평가 기준 부재

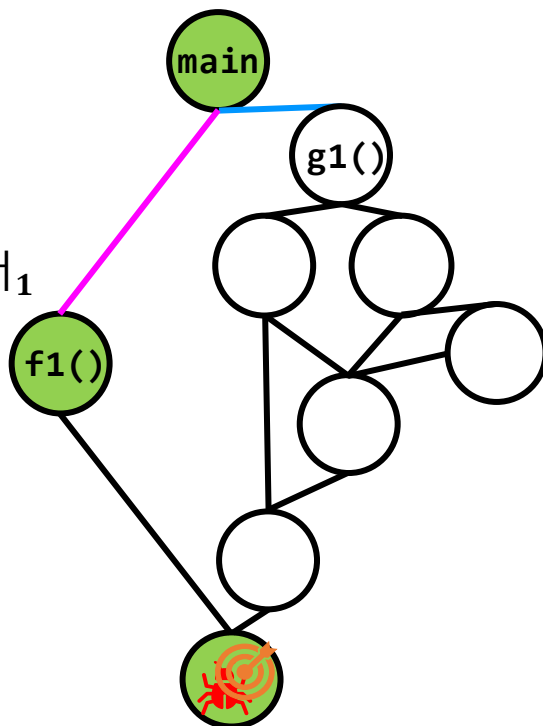
- 각 지향성 퍼저마다의 거리 측정 방식이 달라서 어떤 지향성 퍼저가 목표 지점에 더 가깝게 도달했는지 판단하기 어려움.

❖ 거리(엣지 수)로 평가

➢ main->f1() -> 목표지점 우선 선정



지향성 퍼저₁

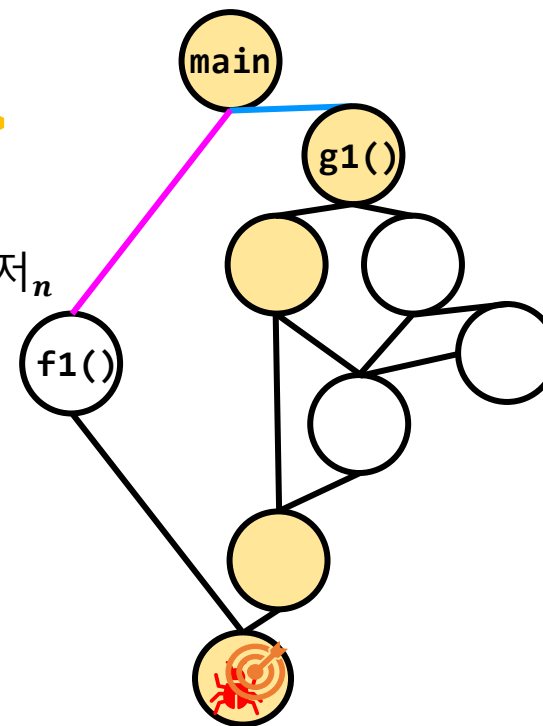


❖ 분기 조건으로 평가

➢ main->g1() -> ... -> 목표지점 우선 선정



지향성 퍼저_n

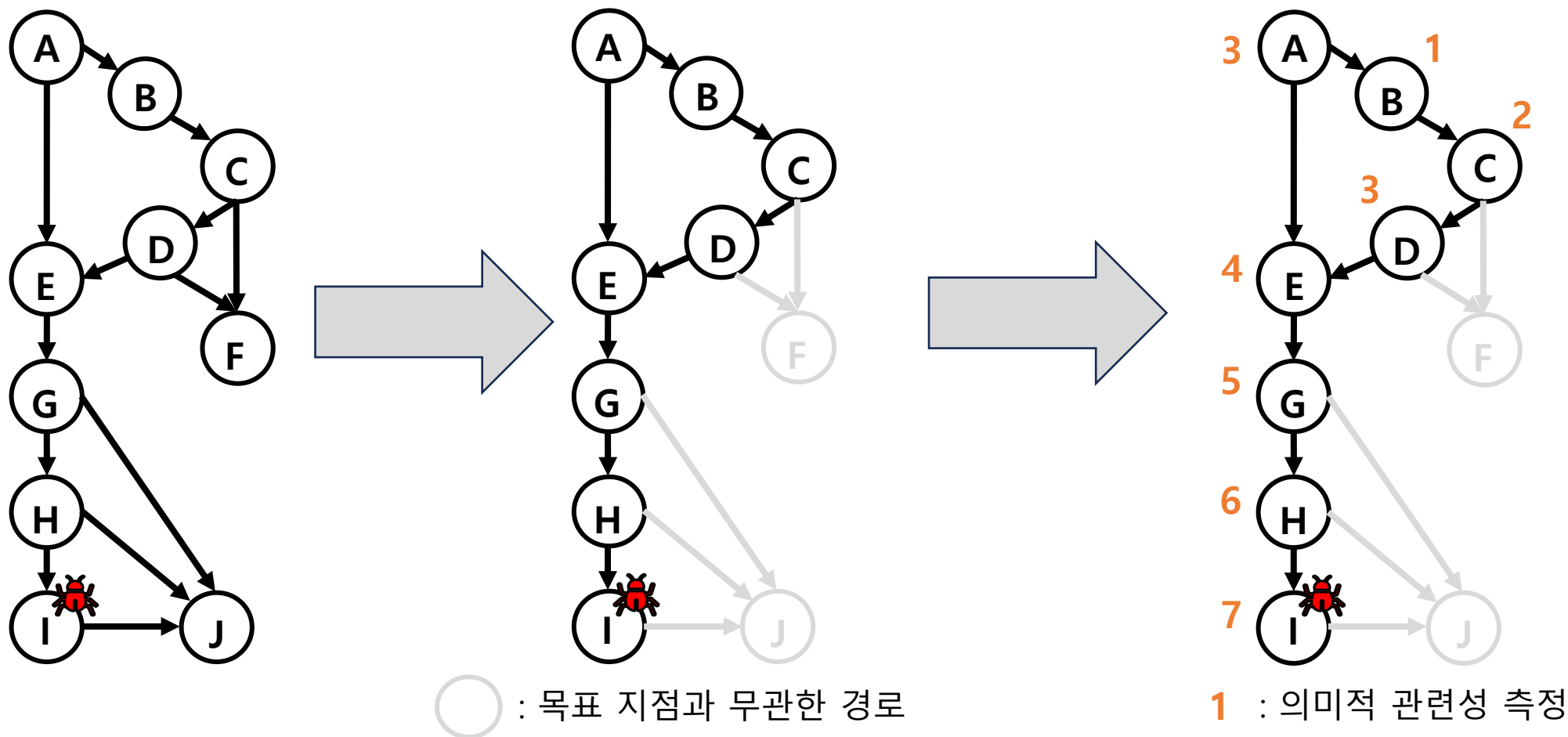


— : 쉬운 조건 ex) if(x>10) 등 ...

— : 어려운 조건 ex) if(hash(input) == 0xDEADBEEF) 특정 해쉬 값 등

공통된 평가 기준 선정

- DAFL 논문의 의미적 관련성 점수 방식으로 선정
 - 목표 지점과 무관한 경로 측정을 제한하여 목표 지점과 관련된 코드 부분만 측정.
 - 목표 지점 거리와 데이터 연관성을 결합한 의미적 관련성 측정을 통해 더 정밀하게 평가



도전 과제

- 공통된 평가 기준 부재

- 목표 지점 도달 이전에 어떤 지향성 퍼저가 목표 지점에 더 가깝게 도달했는지 판단하기 어려움.

- 퍼저 선택 문제

- 성능이 서로 다른 여러 지향성 퍼저 중에서 최적의 지향성 퍼저를 실시간으로 선택하기 어려움.

WindRanger

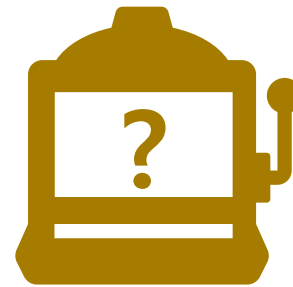
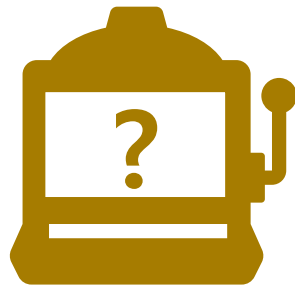
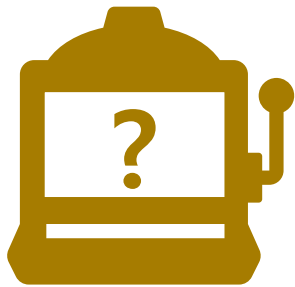
AFLGo



DAFL

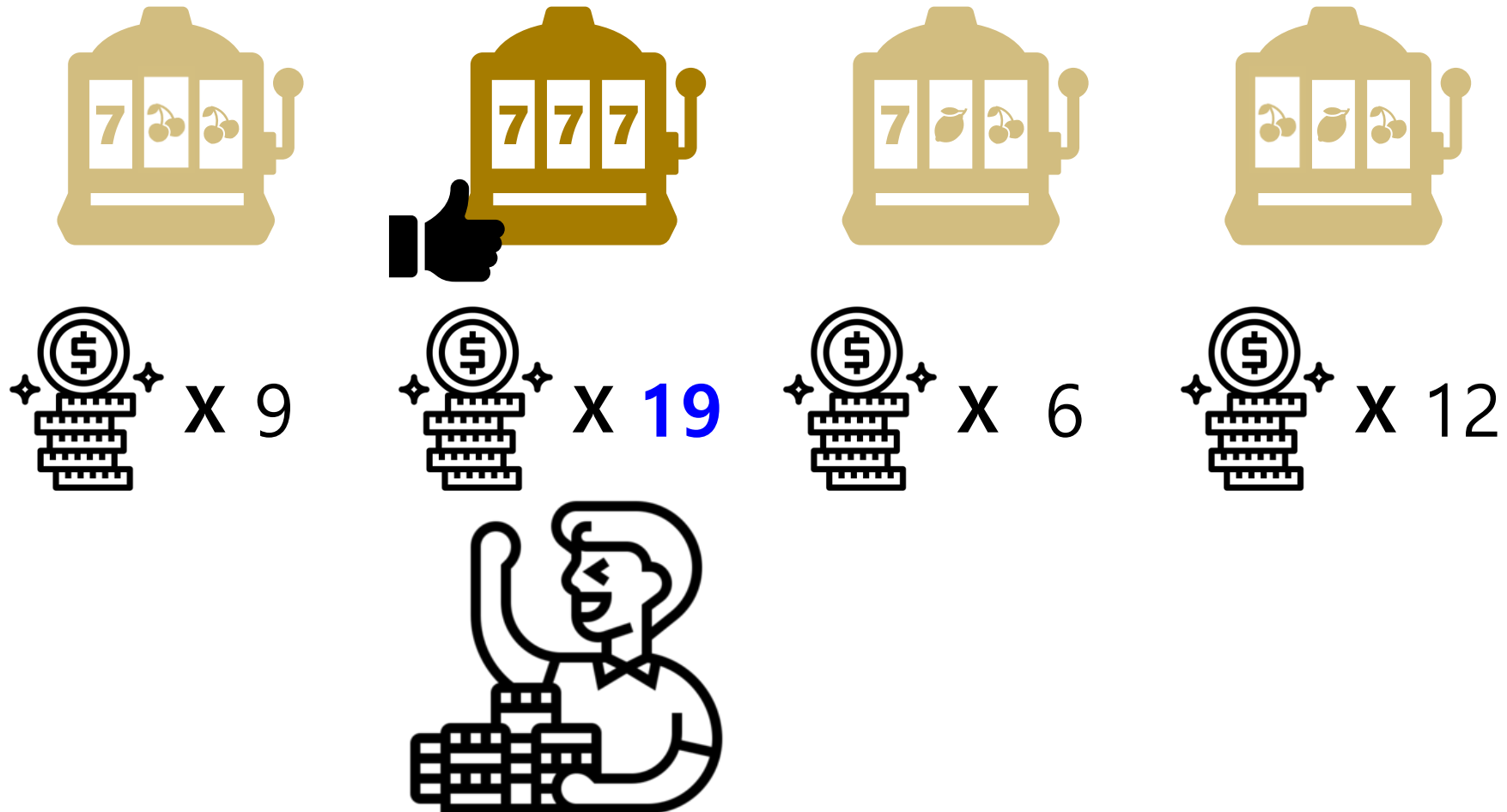
Multi-Armed Bandit(MAB) 문제

- 서로 다른 보상 분포를 가진 여러 개의 슬롯머신 중에서 누적 보상을 최대화하기 위해 어떤 슬롯머신을 선택할 지 결정하는 문제



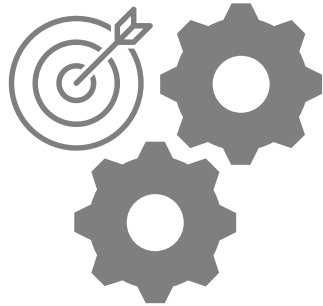
MAB 문제 해결 방안

- 강화학습의 핵심 아이디어 : 탐색과 활용의 균형
 - 탐색 : 보상이 불확실한 슬롯머신을 선택하여, 더 많은 정보를 수집하는 과정
 - 활용 : 현재까지 가장 성능이 좋은 슬롯머신을 선택하여 즉각적으로 보상을 최대화하는 것.



퍼저 선택 문제를 MAB 문제로 모델링

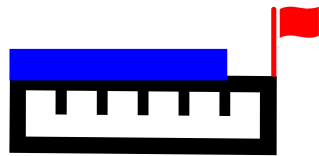
- 실시간으로 성능이 서로 다른 여러 지향성 퍼저들 중에서 **가장 빠르게 목표 지점에 도달할 것 같은 지향성 퍼저를 선택**하는 문제



지향성 퍼저



슬롯 머신



목표 지점 도달

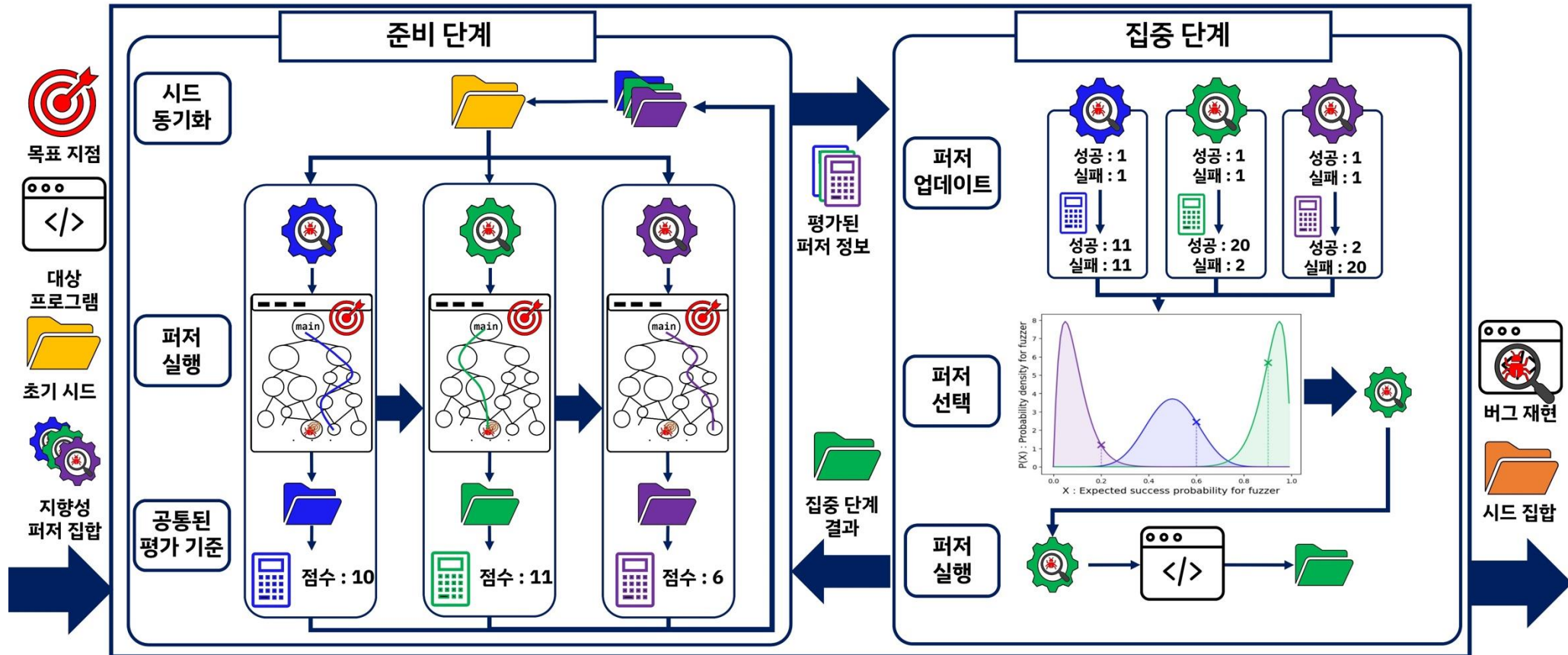


보상

DCFuzz

- DCFuzz(Directed Collaborative Fuzzer)

- **준비 단계:** 각 지향성 퍼저를 실행해서 **공통된 평가 기준**으로 성능 정보를 수집하는 단계
- **집중 단계:** 수집된 성능 정보를 바탕으로 **순위 기반 퍼저 업데이트**와 **톰슨 샘플링 기반 퍼저 선택**으로 목표 지점 도달 및 취약점 재현 효율을 극대화하는 단계



포스터 세션 때 뵙겠습니다.



소프트웨어재난연구센터



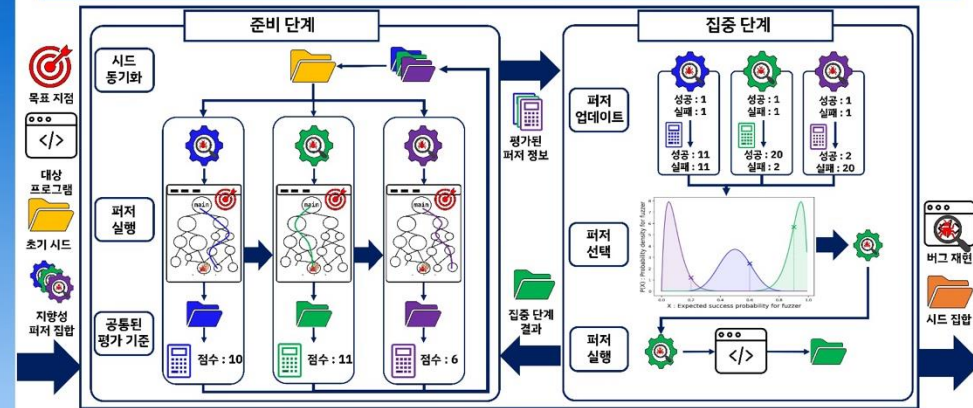
통스 샘플링 기반 지향성 협력 퍼저

모현민, 김윤호
한양대학교 소프트웨어공학연구소

배경 및 동기

- 단일 지향성 퍼저의 한계
 - 같은 프로그램에서도 지향성 퍼저마다 결과가 다르기 때문에 최적의 지향성 퍼저를 선택하는 것이 어려움.
- 지향성 퍼저 성능 비교를 위한 공통된 기준 부재
 - 각 지향성 퍼저마다 거리 계산 방식(예: CFG, DFG, ...) 이 달라 퍼저 간 성능을 일관되게 비교하기 어려움.

통스 샘플링 기반 지향성 협력 퍼저



- 공통된 평가 기준
 - 각 퍼저가 생성한 시드를 DAFL 논문의 Semantic Relevance Score 방식으로 측정해서 가장 높은 점수 시드를 해당 퍼저의 성능으로 선정

$$\text{Score}_{G,t}(s) = \sum_{c \in C_t} (L - |c - t| + 1)$$

- Score_{G,t}(s): 각 시드 경로에서 목표 지점과 의미적으로 관련된 노드들의 점수 합
- L: 목표 지점으로 부터 가장 먼 지점까지의 거리
- C_t: 정의-사용 그래프 상에서 목표 지점 t 까지 도달 가능한 노드 집합

실험 환경

- 벤치마크(총 25개): 기존 지향성 퍼저 연구에서 사용된 CVE로 선정함.
- 비교 대상 퍼저: AFLGo, Windranger, DAFL
- 평가 지표: 최초 목표 지점 도달 시간(TTR)과 최초 목표 지점 취약점 재현 시간(TTE)로 측정함.(단위: 초)

연구 질문 1 - 최초 목표 지점 도달 시간 비교

- DCFuzz는 AFLGo, Windranger, DAFL에 비해 17개, 19개, 10개 목표 지점에서 더 빠르게 목표 지점에 도달함.

Prog	CVE	A	W	D	DC
cxa_mt	2016-4489	543.0	280.4	225.1	250.5
	2016-4490	102.3	118.2	98.4	98.8
	2016-4491	59.2	129.9	92.1	28.5
	2016-4492	1472.3	1768.3	836.7	693.4
	2016-4493	6.9	6.7	3.1	4.6
mm	2017-14949	0.0	0.0	0.0	0.0
	2017-43392	0.0	0.0	0.0	0.0
	2017-43396	0.0	0.0	0.0	0.0
	2017-43397	29.5	51.1	2.6	6.0
	2017-43398	611.3	541.7	745.0	277.0
obj_dump	2016-9827	43.3	64.7	34.6	42.4
	2016-9829	135.2	68.3	35.7	33.7
	2016-9831	0.0	0.0	0.0	0.0
	2017-9988	3033.6	2807.0	2091.5	1838.3
	2017-11728	617.5	930.2	89.1	213.4
swf_to_php	2016-7868	T.O	T.O	31584.4	19477.4
	2016-6131	T.O	T.O	36837.3	29783.2
	2018-9862	63860.0	35715.0	11551.0	32741.3
	2018-11225	27397.0	29750.0	30976.0	14356.4
	2018-11226	17300.0	5538.0	34078.0	24292.3
cxa_mt	2016-4489	543.0	280.4	225.1	250.5
	2016-4490	102.3	118.2	98.4	98.8
	2016-4491	59.2	129.9	92.1	28.5
	2016-4492	1472.3	1768.3	836.7	693.4
	2016-4493	6.9	6.7	3.1	4.6
mm	2017-14949	0.0	0.0	0.0	0.0
	2017-43392	0.0	0.0	0.0	0.0
	2017-43396	0.0	0.0	0.0	0.0
	2017-43397	29.5	51.1	2.6	6.0
	2017-43398	611.3	541.7	745.0	277.0
obj_dump	2016-9827	43.3	64.7	34.6	42.4
	2016-9829	135.2	68.3	35.7	33.7
	2016-9831	0.0	0.0	0.0	0.0
	2017-9988	3033.6	2807.0	2091.5	1838.3
	2017-11728	617.5	930.2	89.1	213.4
swf_to_php	2016-7868	T.O	T.O	31584.4	19477.4
	2016-6131	T.O	T.O	36837.3	29783.2
	2018-9862	63860.0	35715.0	11551.0	32741.3
	2018-11225	27397.0	29750.0	30976.0	14356.4
	2018-11226	17300.0	5538.0	34078.0	24292.3

- 순위 기반 퍼저 업데이트
 - 측정된 퍼저 성능 순위 따라 사전 정의된 성공과 실패 횟수로 업데이트

점수	성공	실패	순위	성공 횟수 증가	실패 횟수 증가
점수: 10 성공: 1 실패: 1	1	+19	1	+19	+1
점수: 11 성공: 1 실패: 1	2	+10	2	+10	+10
점수: 6 성공: 1 실패: 1	3	+1	3	+1	+19

연구 질문

- 연구 질문 1. DCFuzz는 단일 지향성 퍼저에 비해 빠르게 목표 지점에 도달하는가?
- 연구 질문 2. DCFuzz는 단일 지향성 퍼저에 비해 빠르게 목표 지점의 취약점을 재현하는가?

연구 질문 2 - 최초 취약점 재현 시간 비교

- DCFuzz는 AFLGo, Windranger, DAFL에 비해 18개, 16개, 14개 목표 지점에서 더 빠르게 목표 지점 취약점을 재현함.

Prog	CVE	A	W	D	DC
cxa_mt	2016-4489	543.0	280.4	225.1	250.5
	2016-4490	102.3	118.2	98.4	98.8
	2016-4491	59.2	129.9	92.1	28.5
	2016-4492	1472.3	1768.3	836.7	693.4
	2016-4493	6.9	6.7	3.1	4.6
mm	2017-14949	0.0	0.0	0.0	0.0
	2017-43392	0.0	0.0	0.0	0.0
	2017-43396	0.0	0.0	0.0	0.0
	2017-43397	29.5	51.1	2.6	6.0
	2017-43398	611.3	541.7	745.0	277.0
obj_dump	2016-9827	43.3	64.7	34.6	42.4
	2016-9829	135.2	68.3	35.7	33.7
	2016-9831	0.0	0.0	0.0	0.0
	2017-9988	3033.6	2807.0	2091.5	1838.3
	2017-11728	617.5	930.2	89.1	213.4
swf_to_php	2016-7868	T.O	T.O	31584.4	19477.4
	2016-6131	T.O	T.O	36837.3	29783.2
	2018-9862	63860.0	35715.0	11551.0	32741.3
	2018-11225	27397.0	29750.0	30976.0	14356.4
	2018-11226	17300.0	5538.0	34078.0	24292.3

향후 연구

- 목표 지점 도달 이후 목표 지점 취약점 재현을 위한 정밀한 평가 기준 선정
 - 연관성 점수 방식은 목표 지점 도달에 기반한 퍼저 선택에 초점을 두고 있어, 목표 지점 도달 이후 취약점 재현 단계에서 어떤 퍼저가 더 효과적인지는 알기 어려우므로, 여러 기준(크래시 발생, 경로의 다양성 등)을 통합하여 정밀한 평가 기준을 설계할 예정