CERIAS

The Center for Education and Research in Information Assurance and Security

Fafnir: Type Inference Based Static Rewriting

Sushant Dinesh, Lovepreet Singh, Mathias Payer

Purdue University

Motivation

- Binary Patching
- Binary Hardening
- Software Customization

Challenges: Lack of Abstractions

```
0x4007eb: 0x48,0x65,0x6c,0x6c,0x6f
...
mov edi,0x4007eb   <.string:"Hello">
mov eax,0x0
call 4004a0        <printf@plt(<ref>, ... )>
```

Fafnir Overview

1. Lift to MIR

- IR geared towards abstraction recovery.

2. Type Inference Analysis

- Assign coarse grained types: *Scalar* or *Reference* to every expression.

3. Symbolization

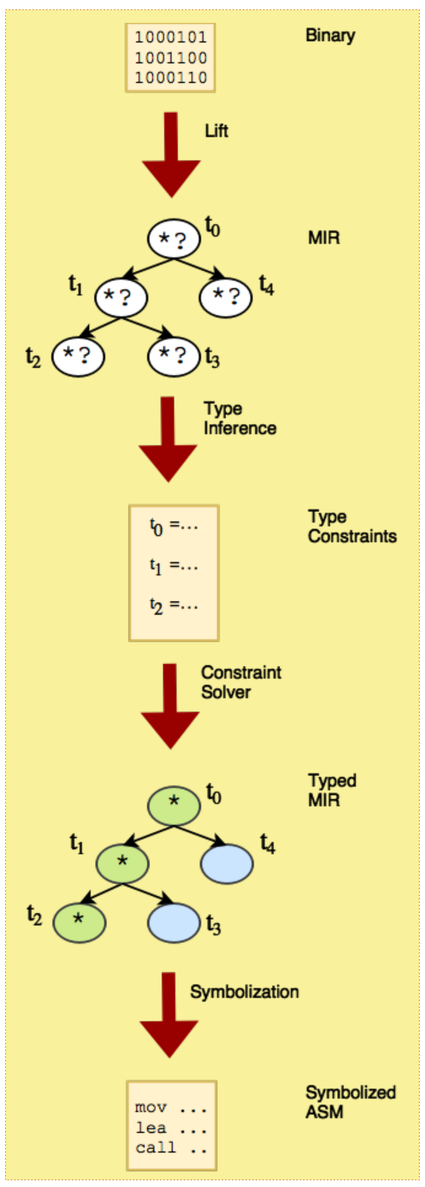
- Replace constants typed to be references by assembler labels.

4. Modification

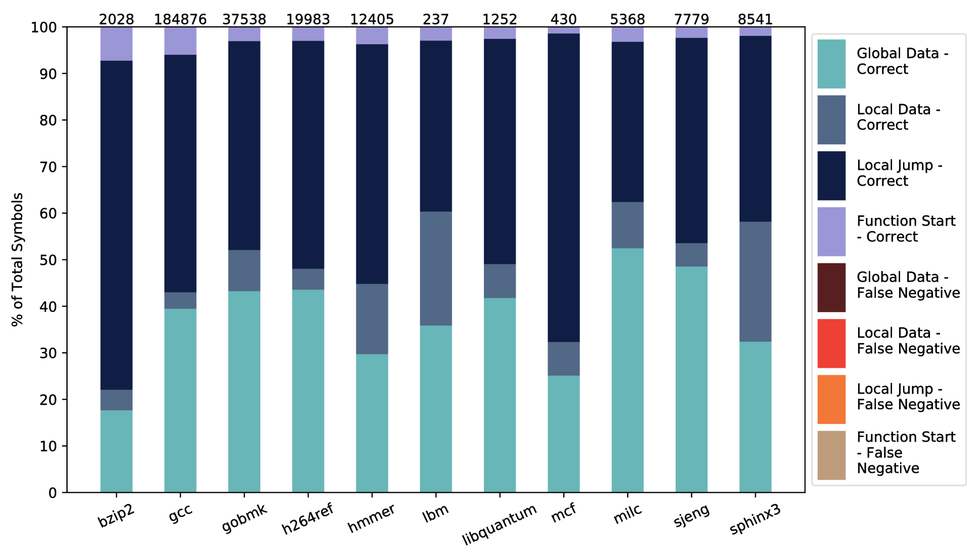
- Edit the generated assembly file freely.

5. Reassemble

- Assemble using any COTS assembler to a working binary!



100% Symbolization Accuracy



SPEC CPU2006 C: Performance

Benchmark	Base	Reasm	Ovhd. %
401.bzip2	330	329	-0.30
403.gcc	191	191	0.00
429.mcf	246	255	3.65
433.milc	531	531	0.00
445.gobmk	349	354	1.43
456.hmmer	272	271	-0.36
458.sjeng	383	382	-0.26
462.libquantum	249	244	-2.00
464.h264ref	370	370	0.00
470.lbm	460	459	-0.21
482.sphinx3	357	359	0.56

Nginx Performance

Statistic	Time	Req/s
Base	1.93s	51724.90
Reasm	1.94s	51332.86
%Ovhd.	0.78	-0.75%

Case Studies

CVE	Program	Vulnerability
CVE-2013-2028	nginx	Buf. Overflow
CVE-2012-0809	sudo	FMT String
CVE-2014-2299	wireshark	Buf. Overflow
CVE 2014-0195	openssl	Buf. Overflow

PURDUE UNIVERSITY Discovery Park

CERIAS