Reassembleable Disassembling

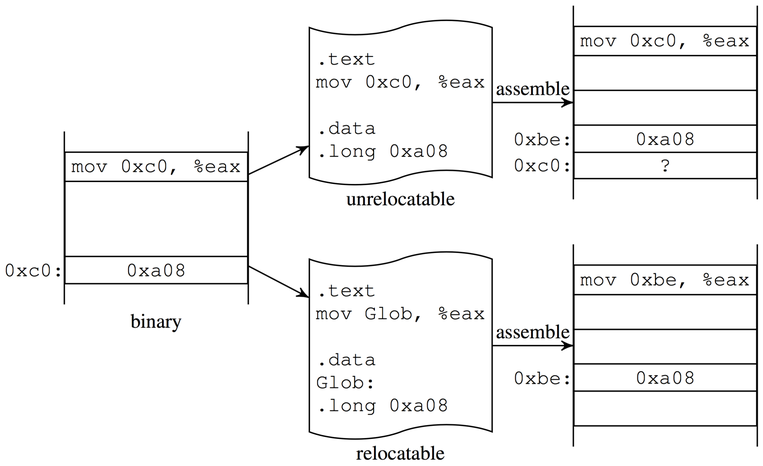
Shuai Wang, Pei Wang, Dinghao Wu
The Pennsylvania State University

Research Problem

- **Binary disassembling** is a foundation of many binary security and instrumentation tasks.
- Output of existing disassemblers cannot be automatically reassembled back with equivalent semantics! As a result, binary instrumentation and retrofitting is tedious and cumbersome.
- **Our goal: design a disassembler to generate reassembleable code!**

Key Challenge

Without relocation info, references can be malfunctioned in the reassembled binary

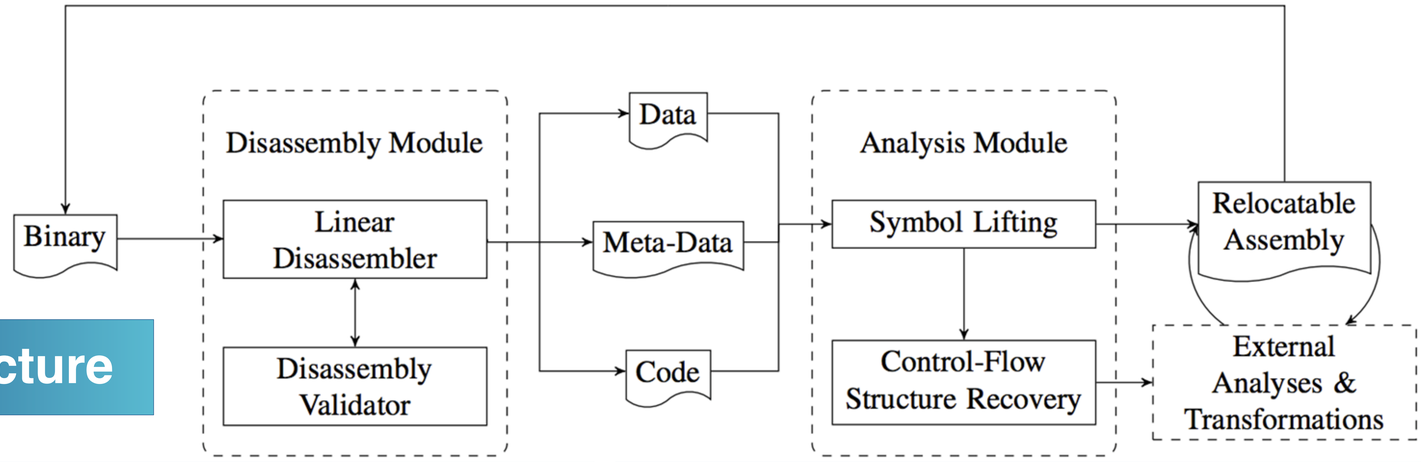


- Disassembled code are untyped, lack of procedure boundaries and control flow logic.
- Most proposed program analysis techniques are code-oriented, lacking the capability of analyzing data sections where references can live.
- Reassembly has almost zero tolerance for type inference errors.

Funding

Supported by NSF Grant No. CNS-1223710 and CCF-1320605.

Architecture



Symbol Lifting

Basic criteria

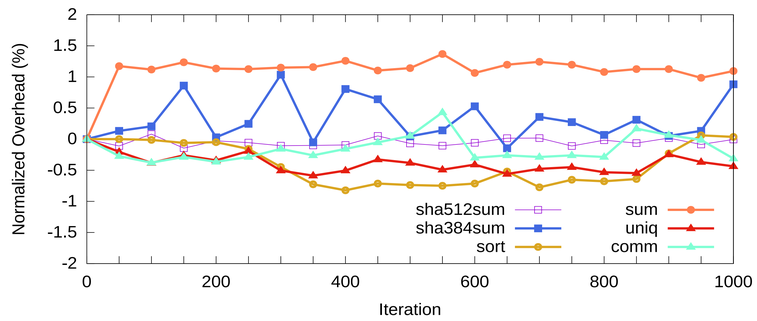
- References must fall in the address space allocated for the binary.
- References on code sections must point to the starting addresses of some instructions.

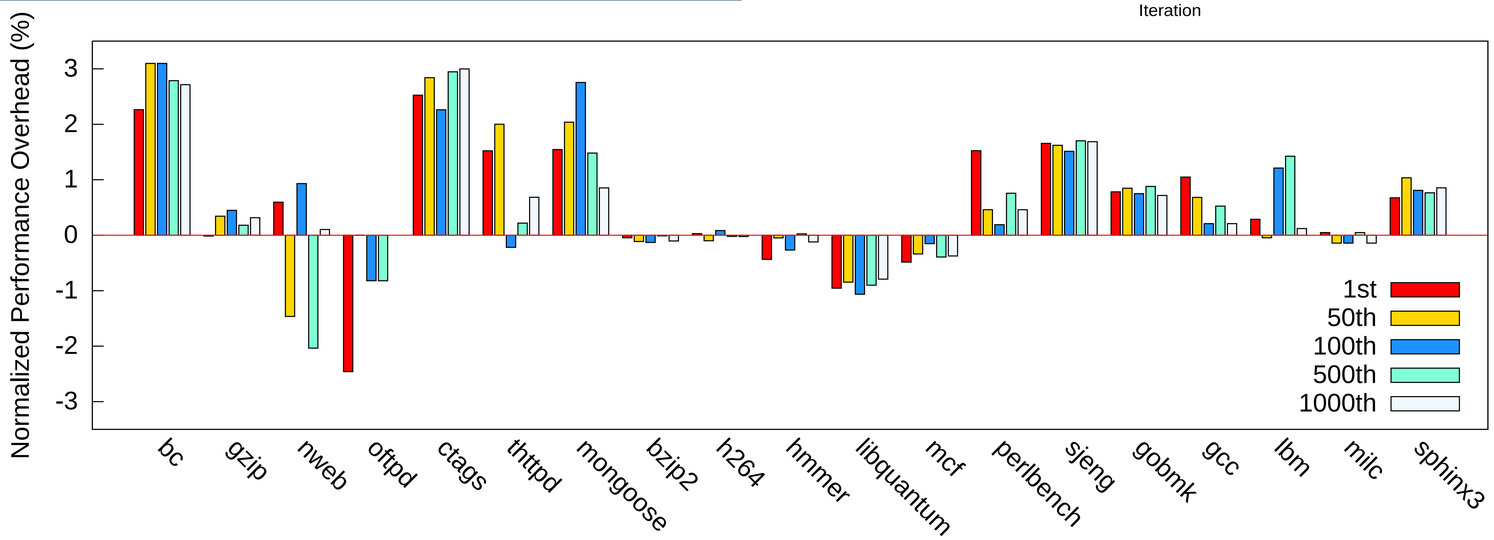
Three more criteria to identify references in data sections

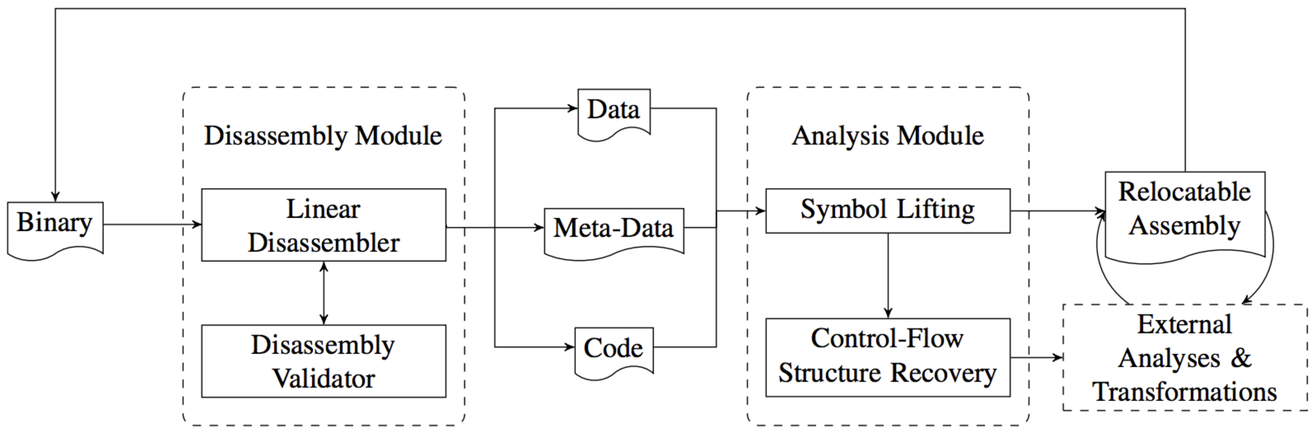
- ***Alignment***: references in data sections are n-byte aligned. (n=32 or 64).
- ***Code pointer layout***: references point to code sections would only be used as function pointers or jump table entries.
- ***Fix data section address***: the starting addresses of data sections would not change after reassemabling (no need to identify references point to data sections).

Key Results

- **Only one** error regarding 244 binaries from GNU Coreutils, SPEC2006 and several server programs.
- Can work with programs as big as **GCC**.
- Can repeatedly disassemble and reassemble for **thousands of times**.
- Negligible performance and size change (< 1%).
- Release: https://github.com/s3team/uroboros.
- See details in Wang et al. (USENIX Security 2015, SANER 2016).






Binary
Disassembly Module
Linear Disassembler
Disassembly Validator
Data
Meta-Data
Code
Analysis Module
Symbol Lifting
Control-Flow Structure Recovery
Relocatable Assembly
External Analyses & Transformations