Digital Watermarks for LLVM Intermediate Representation

R. Nagayama, L. Chen, H. Inaba Kyoto Institute of Technology

Background

There is little research on software watermarking.
2 basic approach of software watermarking:

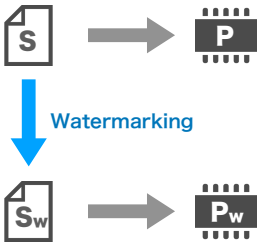
1. Modifying the Program Binary.

Embedding by modifying the executable or the bytecode directly.

- No resistance to overwrite attacks.
- Depends on target platform.

2. Modifying the Source Code.

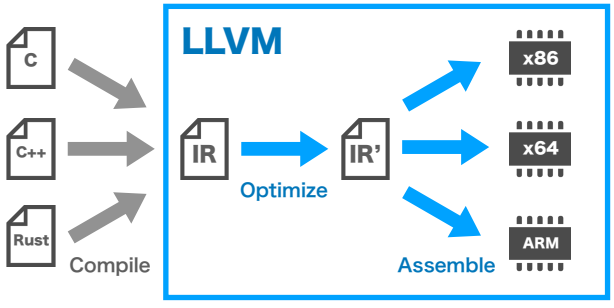
Embedding in the source code and compiling it.



- No resistance to optimization.
- Depends on development language.

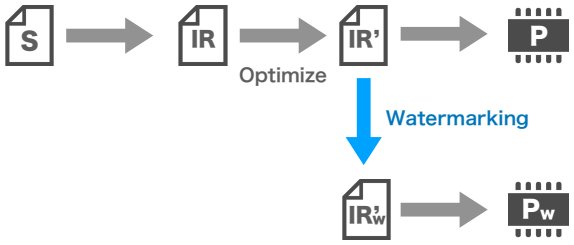
LLVM

LLVM is a compiler infrastructure that supports native code output to various target platforms.
LLVM IR is an intermediate representation (IR) provided by LLVM.
LLVM provides several optimization paths for IRs.



Proposal

Watermarks are embedded in LLVM IR.



We proposed 3 embedding methods.
- Method-1: Changing the order of basic blocks
- Method-2: Swapping instruction operands
- Method-3: Changing the order of functions

Evaluation

LLVM has 3 optimization levels.
- IR level optimization (IRO)
- Machine code optimization (MCO)
- Link time optimization (LTO)

Resistant to optimization

	IRO	MCO	LTO
Method-1	✓		
Method-2	✓		
Method-3	✓	✓	

Conclusion

1. Resistance

- All methods are resistant to overwrite attacks.
- All methods are resistant to IR level optimization.
- Method-3 is resistant to machine code optimization.

2. Independence

- Independent of target platform.
- Independent of development language.