

Program Obfuscation

Theory and Practice

Outline

- Introduction
- Theory
- Practice
- Future Work
- Conclusion

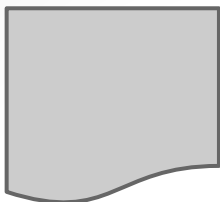
Introduction

- General idea
- Utility of program obfuscation
 - From the developer's perspective
 - From the user's perspective
- Parallels with message encryption
- Modernization of program obfuscation

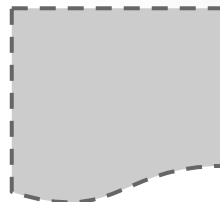
Program obfuscation: Theory

- Definition of obfuscation
 - The syntactic requirements
 - The additional criteria
- Definition of an obfuscator
- Three classes of obfuscation
 - Virtual black box obfuscation
 - Differing-inputs obfuscation
 - Indistinguishability obfuscation

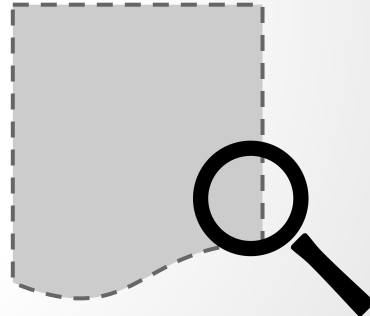
Virtual black box obfuscation



Original



Obfuscated



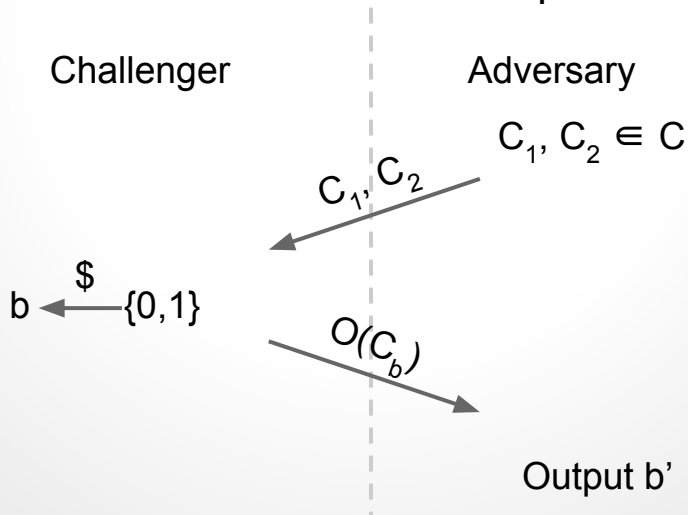
$\Pi(P)$

Differing-inputs obfuscation

- If there is an adversary A that can distinguish $O(P_1)$ from $O(P_2)$,
- then a second adversary A' can find two inputs where the output of $O(P_1')$ and $O(P_2')$ differ.
- Not likely to exist.

Indistinguishability obfuscation

- There is a class of circuits C that computes a function f .
- For any two circuits C_1, C_2 in C , a PPT adversary A cannot distinguish between $O(C_1)$ and $O(C_2)$.

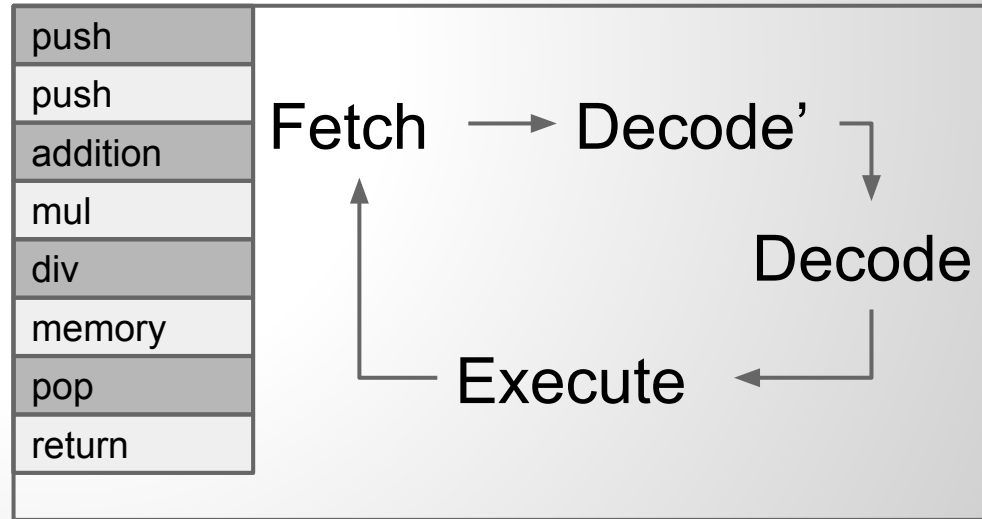
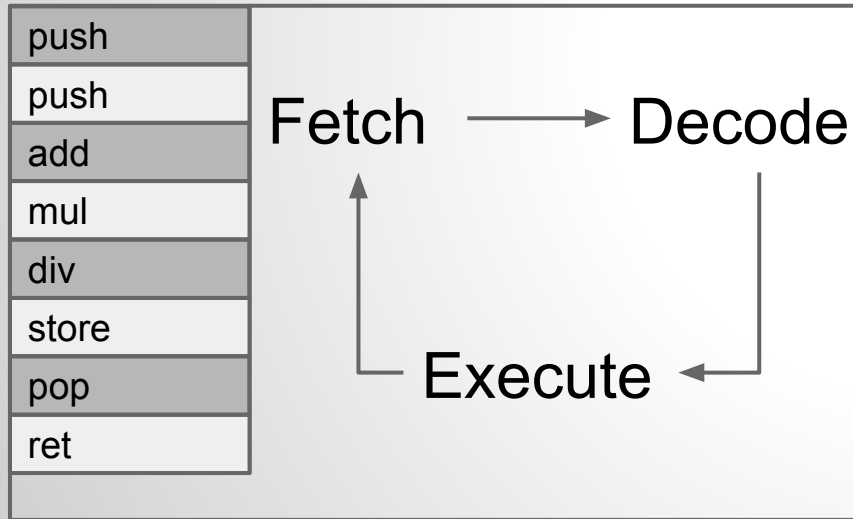


Program obfuscation: Practice

- Uses of randomness
- Formalizations (i.e., the modernity of security)
- Threat model

Software dynamic translation

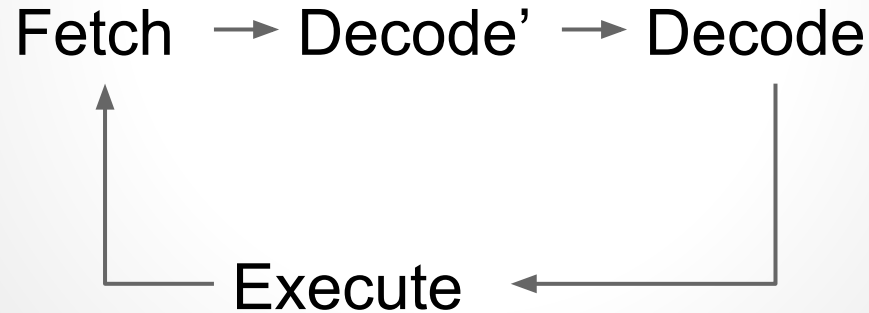
- Interrupt the normal execution process at runtime.



Software dynamic translation

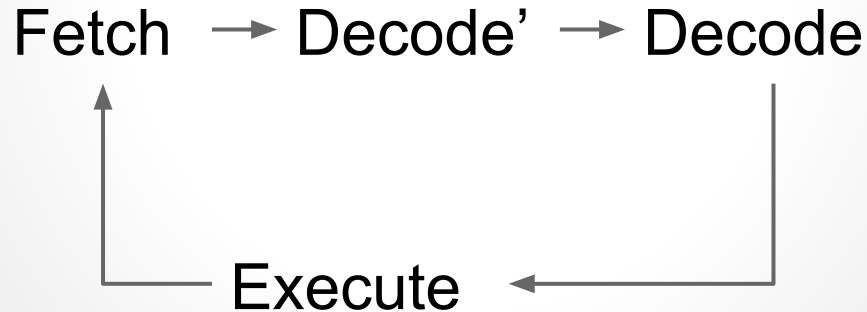
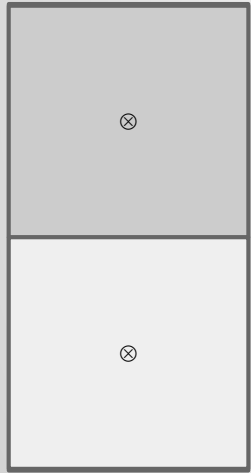
- Randomize the instructions

???
???
???
???
???
???
???
???

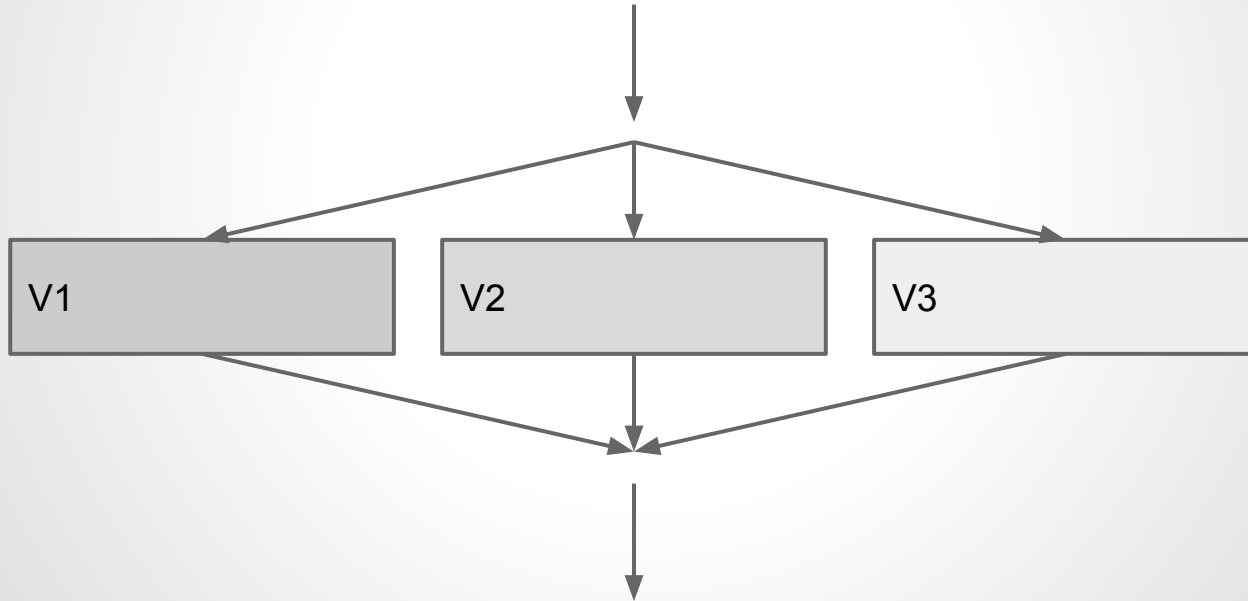


Software dynamic translation

- Encrypt blocks of instructions



N-Variant Systems



Future Work

- Return oriented programming.

Conclusion