A Qualitative Study of Indistinguishability Obfuscation
Bachelor thesis

Supervisors: Prof. Dr. Alexander Pretschner, Dr. Martin Ochoa, Sebastian Banescu
Email: {alexander.pretschner, martin.ochoa, sebastian.banescu} @ in.tum.de
Phone: +49 89 289 – 17, 314
Starting date: immediately

Context
Software obfuscation is a code transformation which aims to make a computer program “unintelligible” while preserving its functionality. A recent breakthrough in the formal study of obfuscation is presented by Garg et al. [1]. They propose a candidate indistinguishability obfuscation approach, which is applicable to all polynomial-size log-depth circuits and is based on multi-linear jigsaw puzzles (a simplified variant of multi-linear maps). This work promises to be ground-breaking since Goldwasser and Rothblum have proven that indistinguishability obfuscators achieve the notion of best-possible obfuscation [2].

Indistinguishability obfuscation has several interesting applications, e.g. functional encryption [1] and deniable encryption [3]. However, as to our knowledge, no studies have been published as to the effectiveness of this approach against reverse engineering attacks of simple algorithms implemented in software or hardware programs.

Goal
One goal of this thesis is to investigate the applicability of the approach proposed by Garg et al. [1] on modern software and/or hardware programs. In this context, applicability refers to possibility of implementing the approach using popular programming languages (e.g. C, Java) and its impact on performance and other resources. Another goal of this thesis is to compare indistinguishability obfuscators with other practical obfuscation techniques, which are generally used to protect common software [4].

Ideally, the outcome of this thesis will help software and security engineers understand what are the practical and theoretical security guarantees offered by indistinguishability obfuscation as well as its implied overhead.

Work-plan
1. Develop understanding of indistinguishability obfuscation and other formal obfuscation notions from the provided references and relevant literature cited by these.
2. Evaluation of an indistinguishability obfuscator for Boolean circuits
   a. Implement a proof-of-concept indistinguishability obfuscator as described by [1].
   b. Evaluate the run-time performance impact of the obfuscated circuits as a function of their size.
   c. Discuss the practical value of the observed obfuscation. This discussion should answer the questions: “How difficult is it to reverse-engineer the function of the original circuit from the obfuscated circuit?”, “Can this be done via static analysis only?”
   d. Discuss the applicability of the approach in [1] to software programs written in popular programming languages, e.g. C, Java, Assembler, etc.
3. Comparison of formal obfuscation methods with practical software obfuscation methods described in scientific and non-scientific literature [4].

4. The final thesis document must contain:
   a. Description of the problem and motivation for the chosen approach
   b. Description of the theoretical background
   c. Implementation description
   d. Performance evaluation of implementation
   e. Comparison of formal and practical obfuscation
   f. Discussion on potential security and performance threats
   g. Conclusions and future work.

**Deliverables**
- Virtual machine able to run a demo of the implementation, including instructions on how to run the demo.
- The VM should also include the source code of the implementation.
- Technical report with comprehensive documentation of the implementation, i.e. design decision, architecture description, API description and usage instructions.
- Final thesis report written in conformance with TUM guidelines.

**References**