**Code Obfuscation**

**Why is This Still a Thing?**

Christian Collberg  
Department of Computer Science  
University of Arizona  
collberg@gmail.com  
http://collberg.cs.arizona.edu

**ABSTRACT**

Early developments in code obfuscation were chiefly motivated by the needs of Digital Rights Management (DRM) [7]. Other suggested applications included intellectual property protection of software [4] and code diversification to combat the monoculture problem of operating systems [2].

Code obfuscation is typically employed in security scenarios where an adversary is in complete control over a device and the software it contains and can tamper with it at will. We call such situations the Man-At-The-End (MATE) [3] scenario. MATE scenarios are the best of all worlds for attackers and, consequently, the worst of all worlds for defenders: Not only do attackers have physical access to a device and can reverse engineer and tamper with it at their leisure, they often have unhindered resources (time, computational power, etc.) to do so. Defenders, on the other hand, are often severely constrained in the types of protective techniques available to them and the amount of overhead they can tolerate.

In other words, there is an asymmetry between the constraints of attackers and defenders. Moreover,

- DRM is becoming less prevalent (songs for sale on the Apple iTunes Store are no longer protected by DRM, for example);
- there are new cryptographically-based obfuscation techniques [1] that promise provably secure obfuscation;
- secure enclaves [5] are making it into commodity hardware, providing a safe haven for security sensitive code; and
- recent advances in program analysis [12] and generic de-obfuscation [13] provide algorithms that render current code obfuscation techniques impotent.

Thus, one may reasonably ask the question: “Is Code Obfuscation Still a Thing?”

Somewhat surprisingly, it appears that the answer is yes. In a recent report, Gartner [14] lists 19 companies active in this space (8 of which were founded since 2010) and there are still (in 2017) many papers published on code obfuscation, code de-obfuscation, anti-tamper protection, reverse engineering, and related technologies.

One of the reasons for this resurgence of code obfuscation as a protective technology is that, more and more, we are faced with applications where security-sensitive code needs to run on unsecured endpoints. In this talk we will show MATE attacks that appear in many novel and unlikely scenarios, including smart cars [6], smart meters [9], mobile applications such as Snapchat and smartphone games, Internet of Things applications [8], and ad blockers in web browsers [11]. We will furthermore show novel code obfuscation techniques that increase the workload of attackers [10] and which, at least for a time, purport to restore the symmetry between attackers and defenders.

**CCS CONCEPTS**

- Security and privacy → Digital rights management; Software reverse engineering; Malware and its mitigation; Hardware attacks and countermeasures;

**KEYWORDS**

Software protection, code obfuscation, reverse engineering, Man-At-The-End, diversification, digital rights management, software watermarking, anti-tamper

**BIOGRAPHY**

Christian Collberg is a Professor in the Department of Computer Science at the University of Arizona. Prior to arriving in Tucson he worked at the University of Auckland, New Zealand, and before that got his Ph.D. from Lund University, Sweden. He has also held a visiting position at the Chinese Academy of Sciences in Beijing, China, and taught at universities in Russia and Belarus.

Dr. Collberg’s main research interest is Software Protection, the use of code obfuscation, software watermarking, anti-tamper, and related techniques to protect software that runs on endpoints under the control of an adversary. He is the co-author of the first book on software protection, *Surreptitious Software: Obfuscation, Watermarking, and Tamperproofing* for Software Protection, published in Addison-Wesley’s computer security series. It has also been translated into Portuguese and Chinese.

In addition to his security research, Dr. Collberg is an advocate for Reproducibility, Repeatability, and Sharing in Computer Science. He maintains the site FindResearch.org which aims to be the most authoritative and complete catalog of research artifacts (i.e., code and data) related to Computer Science publications.

**ACKNOWLEDGMENTS**

This work was funded in part by the NSF under grants CNF-1145913 and CNF-1145913.
REFERENCES