

# Firmware Fuzzing via Rehosting

---

Tobias Scharnowski

## Abstract

Embedded systems are tiny, invisible computers controlled by firmware. Securing firmware is challenging, as traditional automated security tools are ineffective for firmware.

This thesis introduces techniques to make an important security testing technique, fuzz testing, effective for firmware via fully automated firmware rehosting. First, we introduce a technique to model hardware behavior, enabling the automatic creation of an execution environment for firmware on a general-purpose host. Second, we restructure fuzzing inputs to achieve vastly improved fuzzing results, such as 5x increased code coverage and 550x faster bug identification. Finally, we introduce the first technique to rehost complex Direct Memory Access (DMA) transfers generically to find security flaws in firmware that could previously not be analyzed. In combination, these techniques allow us to test firmware for security vulnerabilities and robustness issues in a fully automated, scalable manner.

## Zusammenfassung

Eingebettete Systeme sind nahezu unsichtbare Computer, die von Firmware gesteuert werden. Sicherheitsanalysen von Firmware sind herausfordernd, da bestehende Werkzeuge aus dem Softwarebereich für Firmware unwirksam sind.

Diese Arbeit führt Techniken zur Sicherheitsüberprüfung von Firmware mittels Fuzz Testing ein. Zunächst stellen wir hierzu einen Mechanismus vor, der mittels der vollautomatischen Modellierung von Hardwareverhalten automatisch eine Ausführungsumgebung für Firmware schafft. Daraufhin restrukturieren wir Fuzzingeingaben für Firmware, was die Testabdeckung auf bis zu das Fünffache erhöht. Drittens modellieren wir automatisch komplexe Direct Memory Access (DMA) Transfers, welches die Sicherheitsüberprüfung zusätzlicher Firmwaretypen ermöglicht. Zusammengenommen erlauben diese Techniken die vollautomatisierte und skalierbare Sicherheitsüberprüfung von Firmware, was es großflächiger erlaubt, das Sicherheitslevel eingebetteter Systeme zu erhöhen.