



code intelligence

Automated Application Testing for **Embedded Systems**

CI Fuzz is the state-of-the-art solution for automated and highly scalable software testing that can be used by everyone. It offers a clear business value compared to traditional testing methods due to significant cost savings, greatly reduced time-to-market and improved software quality. You can focus on building the best product while Code Intelligence continuously takes care of finding bugs and vulnerabilities.

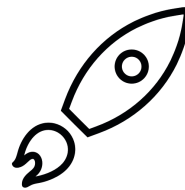
Fuzzing gained enormous popularity both in the industry and scientific community in the last years. For instance, 16.000 bugs have been automatically discovered in Google Chrome in 2019. While proven to be a very effective technology to find vulnerabilities and bugs, modern fuzzing is still too complex to integrate and maintain correctly for most development projects. Code Intelligence has the goal to make modern fuzzing more usable and software more secure.

Why Code Intelligence?



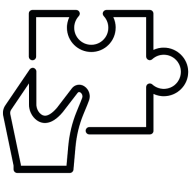
Reduce costs

Save over 60% of developers time due to our fully automated solution and our easy-to-use IDE plugin



Usable modern fuzzing

Modern software tests without expert knowledge supported by our preloaded settings and intelligent execution engine



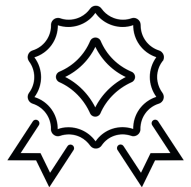
No prerequisites needed

Our agnostic approach ensures seamless integration into your existing process landscape



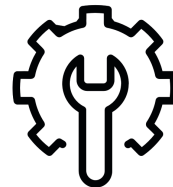
Increase productivity

Maximize the productivity of your costly developers on programming code instead of hunting bugs and security issues



State-of-the-art technology

Unparalleled combination of latest breakthrough technologies under one roof to deliver added business values



Easy setup

Little effort to set up fuzzing for embedded devices



code intelligence

Technical Features

- **Feedback-based fuzzing** based on dedicated instrumentations for common processor architectures
- **Support of the most important architectures** like ARM, AARCH64, MIPS or MIPS64
- **Scalability** of running tests in virtual environment
- **Virtualization and platform agnostic** due to the usage of QEMU (customizable virtualization software)
- **Support of many platforms** which makes it possible to test end-to-end embedded systems
- **Simulation of widely used peripherals** such as WIFI, Bluetooth, CANBus or Serial interfaces
- **Fuzzing via different interfaces** like networks, files and devices
- **Low false-positive rate** due to actual execution of the code rather than static analysis based on patterns and data flows
- **Structure-aware fuzzing** approach for maximization of code coverage with support for custom data types and protocols defined in your own codebase with low integration effort and are pre- & self-defined (JSON/XML/JAML/CSV/...)
- **Easy to integrate into Devops:** Support for common build systems (e.g. Autotools, CMake, Scons) and Continuous Integration Pipelines (e.g. Jenkins, Gitlab Pipelines)
- **Easy to use with our IDE plugin** for VSCode, IntelliJ or Eclipse

Detected Bugs & Vulnerabilities

Buffer overflows

Use after free

Memory leaks

Data races

Software crashes

Hangs/freezes

Call stack overflows

Uncaught exceptions

Undefined behavior

SQL injection

Path traversal

And more ...

Code Intelligence GmbH

Rheinwerkallee 6

53227 Bonn


www.code-intelligence.com


Contact

+49 228 2869 5830

sales@code-intelligence.com

Follow us

 @CI_GmbH

 [company/codeintelligence/](https://www.linkedin.com/company/codeintelligence/)