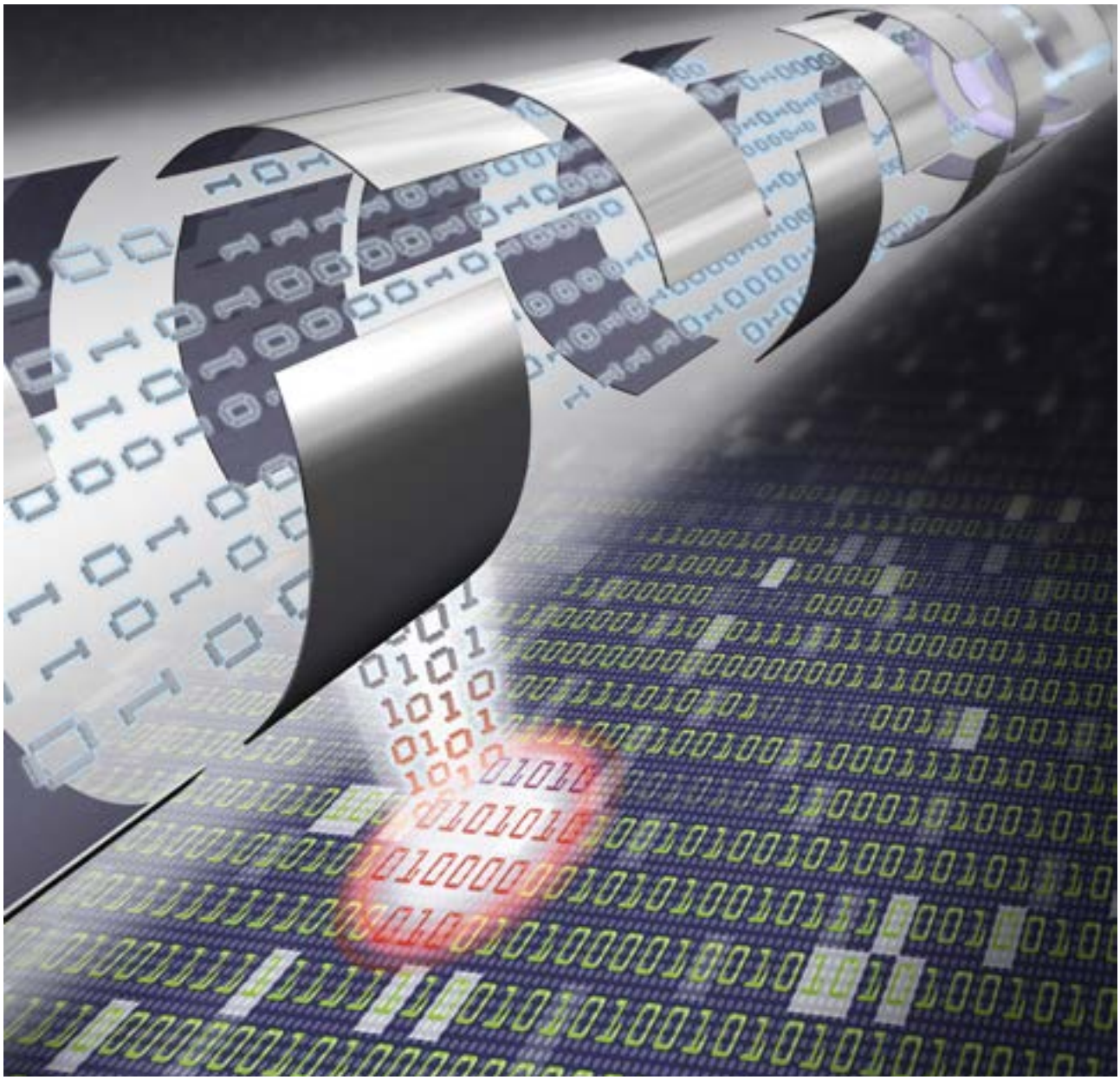
Translated article "Effektive Security-Tests am HiL-System," Hanser Automotive 11-12/2016

Successful security tests using fuzzing and HiL test systems

In automotive development, it is a given that hardware-in-the-loop (HiL) systems undergo systematic testing to ensure functional safety. These tests verify whether the tested system behaves correctly and reliably as per functional requirements, and whether it goes into safe mode in case of a malfunction. Safeguards against unauthorized access require similar testing. This article shows how gaps in safety can be found using effective security tests on the HiL system.



There is increasing discussion of systematic security tests to check for safety gaps in the context of unauthorized access to automotive ECUs. Until now, ECUs have largely been isolated from external networks. This is changing as cars open up to the Internet of Things (IoT), making it all the more important to systematically and efficiently safeguard electronic systems. This is considered security by design.

That is why developers have for some time been devising appropriate security tests for automotive ECUs and incorporating them into the development process. This means security gaps can be identified in the development process and closed using suitable measures.

For decades, security tests have been an integral part of the development process for critical software functions in "traditional" IT. These tests fall into several classes. Functional security tests are used primarily to verify whether the security concept has been implemented. Vulnerability scanning, which is usually automated, focuses on already known gateways for cyber attacks. Fuzz testing, or fuzzing, tests the input processing of the software with a large number of random input values. Finally, penetration tests attempt to actively exploit any security gaps.

For a few years now, developers have been tailoring these test methods and their specific requirements for use in automotive development. This is particularly challenging given the primacy of prioritizing cost efficiency and the complexity of software functions. A further important distinction from conventional security tests lies in the functional dependency of embedded systems and their interaction with each other.

In traditional IT, security tests for individual (software) components typically consider the component in isolation and ignore the surrounding components, such as the operating system or running processes. This simplification doesn't make sense for automotive systems. Security tests on isolated ECUs deliver only partially reliable results and insights into the ECU's behavior in the productive system. This is where functional test environments for ECU security testing come in.

Image 1: Overview of automotive fuzzing with limited monitoring

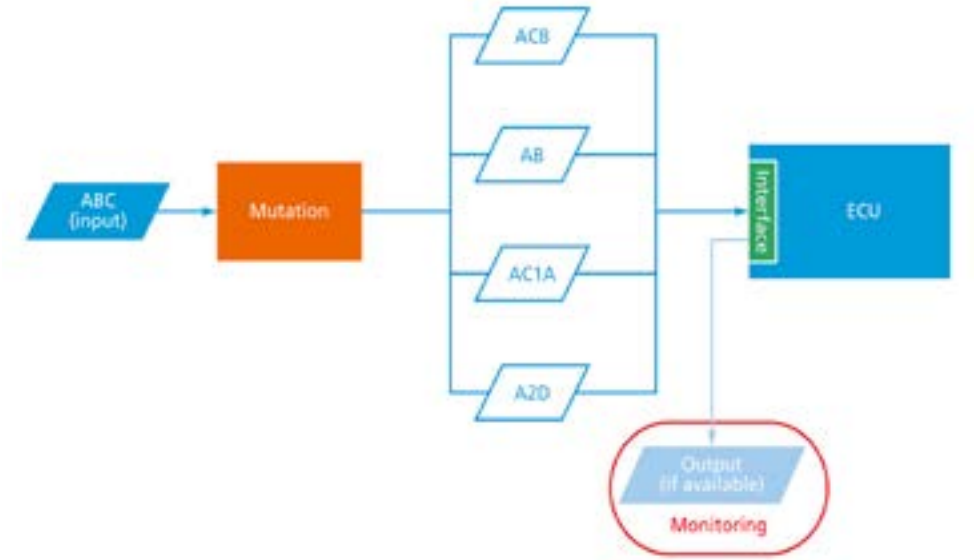
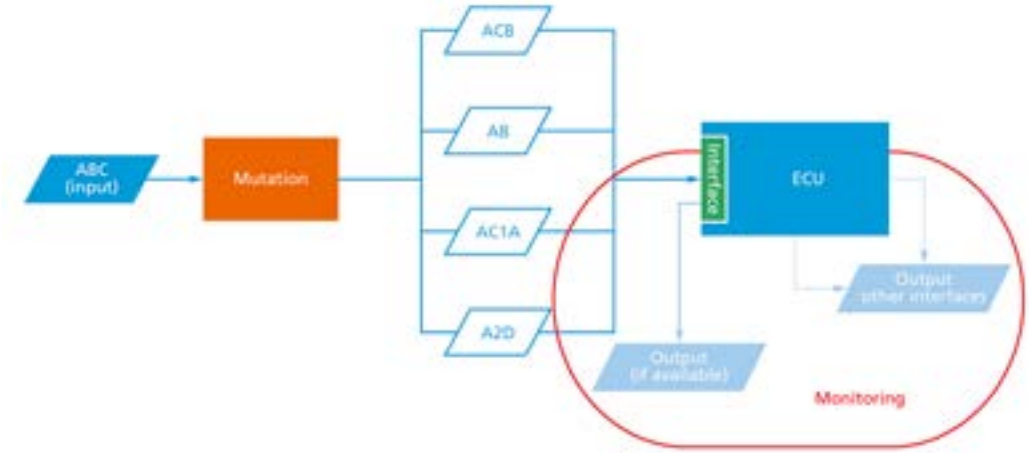


Image 2: Overview of automotive fuzzing with extended monitoring



Challenges in automotive fuzzing

Fuzzing is the name given to a relatively new method for the automated testing of interfaces, which was developed at the University of Wisconsin-Madison in 1989. In this method, the test software – the fuzzer – sends a large number of automatically generated messages to the system being tested and observes its reaction to those messages. The messages are generated either by random permutations from valid initial messages or from a prescribed set of messages. The goal is to expose faulty system behavior in the face of potentially invalid or implausible messages. The fuzzer is specifically adapted to the target being tested.

There are several solutions for different protocols and systems today, including for the automotive industry. The existing fuzzers for CAN and UDS can already generate a large number of such messages and send them via suitable interfaces to the system being tested, for example an ECU. However, these solutions only partly meet the requirements for the testing of automotive ECUs, since the monitoring of the system reaction takes place on the same interface as the message transmission (see picture: overview of automotive fuzzing with limited monitoring). Device-internal effects are not necessarily visible on this interface because, unlike HTTP or FTP on a web server, the protocols used do not always provide a response to an inquiry message. The properties of the CAN standards allow only limited monitoring of system reactions. In addition, some ECUs require specific electrical signals or even a simulation of the entire vehicle network in order to be tested correctly.

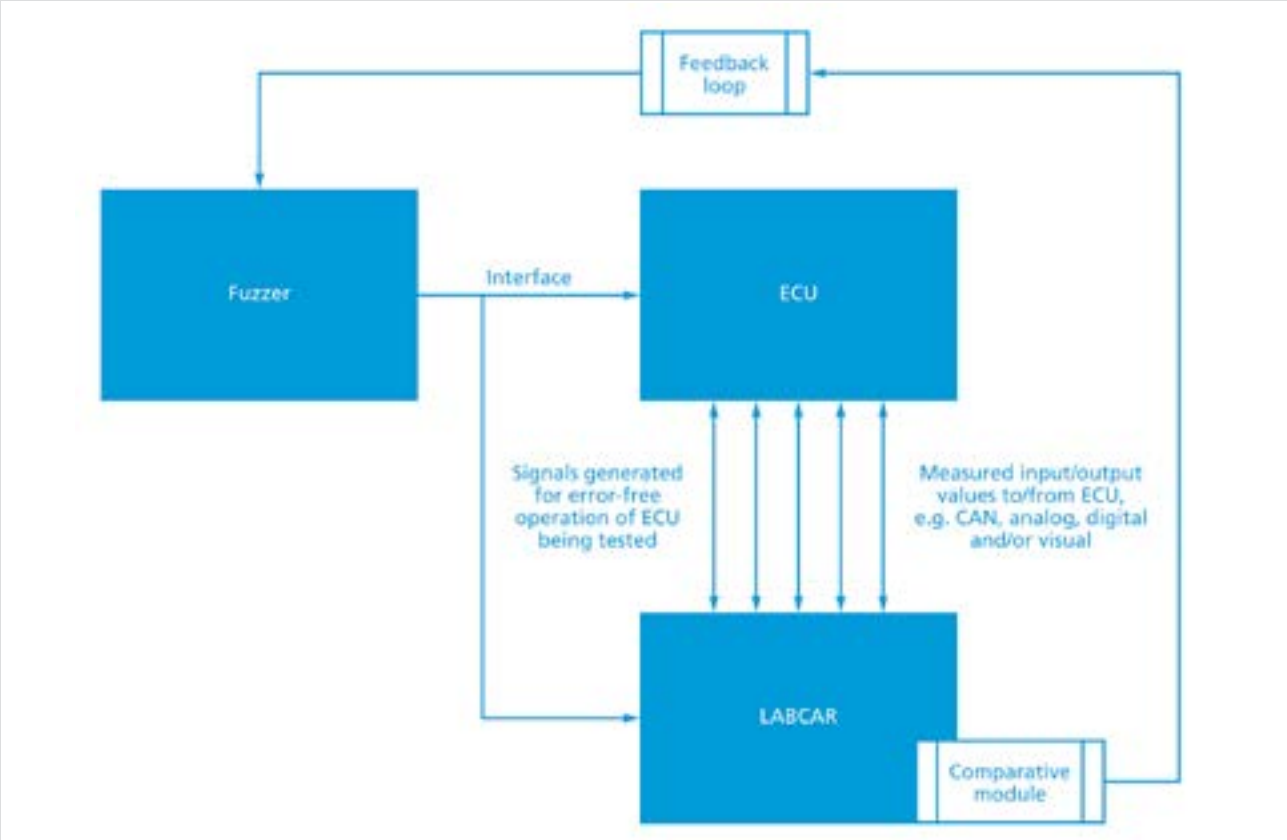
Automotive fuzzing in a functional test environment

This is where virtual test environments come into play. Hardware-in-the-loop (HiL) test systems such as ETAS LABCAR enable real-time testing of a real ECU against a simulated control path. These could be either simple messages to vehicle buses or a simulation of the behavior of the whole vehicle. These virtualization solutions are increasingly used for functional testing of ECUs. They allow for affordable, early, automated, and reproducible testing for a wider variety of more complex software functions.

In addition to real-time simulation of the environment, the signals returned from the ECU being tested are processed and, if necessary, further messages are sent to the ECU based on the ECU signals. The goal of the LABCAR environment is to operate the ECU as it would be in a real vehicle, and consequently facilitate functional tests in the laboratory. LABCAR enables the monitoring of the ECU's internal conditions via full access to the memory and to the tested ECU's program sequences through an ETK access (white-box testing).

The functionality LABCAR provides can also support security tests, as the processing of all output values makes better monitoring possible. (see picture: "Overview of automotive fuzzing with extended monitoring"). The fuzzer receives feedback that can be taken into consideration when generating further test signals. This has remedied one of the drawbacks of existing fuzzers in an embedded context and has increased the effectiveness of fuzz testing of ECUs.

Image 3: Overview of the test setup



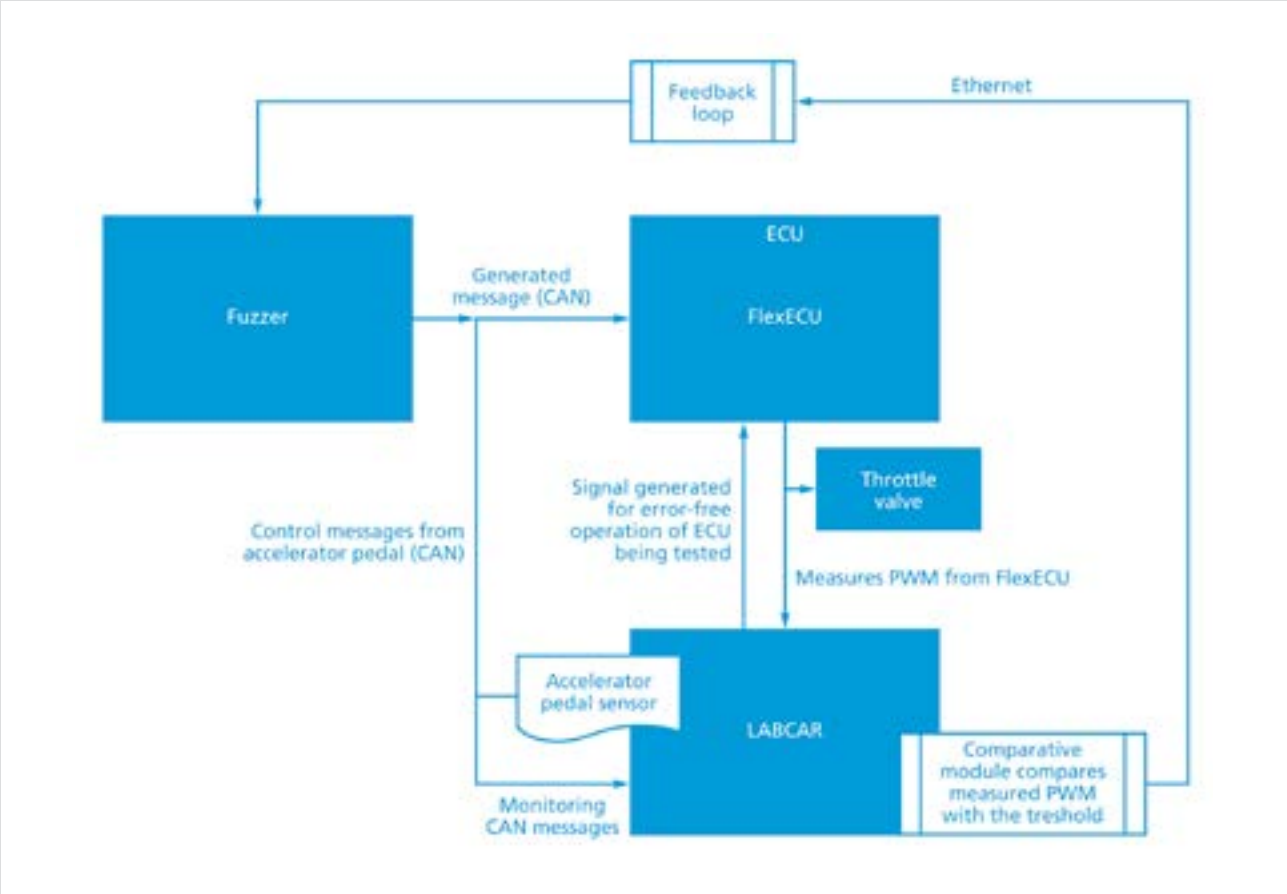


Image 4: Test setup with LABCAR and FlexECU

An overview of the major components and user interfaces of the proposed setup of a CAN-controlled ECU is illustrated in the picture "Overview of the test setup." The comparative module depicted there detects any inadmissible conditions and outputs of the ECU.

Example of a successful implementation

The described solution has already been implemented in a test setup and used to test an ETAS FlexECU electronic control unit. The FlexECU can be used as an engine control unit; in this example scenario, it is connected to an accelerator-pedal sensor via CAN and to a throttle valve via analog and digital connections. The sensor measures the angle of the accelerator pedal, which is valid only in a certain range (0°-120°), and sends the value via CAN to the FlexECU. The ECU uses this signal to control the throttle valve's actuator.

A functional test was done in advance on a vehicle in normal operating mode to ensure that overriding the throttle valve would pose no problems. However, should a hacker succeed in sending arbitrary values to the ECU, especially values outside of the defined range, and thus control the behavior of the throttle valve and the engine, considerable problems could arise. Fuzzing provides a remedy here: the fuzzer is sending faulty inputs to the FlexECU so that the ECU's reaction can be checked and adjusted, if necessary. This ensures safety.

The FlexECU processes the input values from the accelerator pedal (simulated through LABCAR) and controls the throttle valve. For this the ECU calculates a pulse width modulation frequency and sends this to the throttle valve. The red line in the graph corresponds to this frequency's maximum value of 1,200 Hz, which was determined in the comparative module. The black curve shows the actual course of the frequency controlled by the FlexECU.

The first graph shows normal operating conditions, where the frequency value does not exceed the defined maximum value of 1,200 Hz. The second graph shows the value exceeding the defined maximum value, when data generated by a fuzzer is not adequately verified by the FlexECU. This indicates a weak point. The initial proof-of-concept implementation of the presented fuzzing solution described here shows that the approach leads to more efficient and feasible security tests in practice. An additional test setup for a gateway control unit is described in the publication of Oka et al. [3].

Image 5: Normal behavior of PWM over time. The red line represents the threshold (max 1,200Hz).

Image 6: Behavior of the PMW frequency under fuzzing. PMW exceeds the max. 1,200 Hz (red line represents the threshold)

Integrated safety and security test methods

Potential security gaps must be found and closed during the development process for automotive electronics. As security tests on an isolated ECU seldom deliver reliable results and do not allow statements about the ECU's behavior in the productive system, ETAS and ESCRYPT have developed an integrated test method that combines the security test with a functional test environment simulation. It is an innovative response to the special challenges of security tests in automotive industry.

Authors

Dr. Stephanie Bayer, ESCRYPT GmbH
Dr. Tobias Kreuzinger, ETAS GmbH
Dr. Dennis Kengo Oka, ETAS GmbH
Dr. Marko Wolf, ESCRYPT GmbH

[1] C. Miller and C. Valasek, Remote Exploitation of an Unaltered Passenger Vehicle, 2015.
[2] F. Maier, „Computerwoche", 21 12 2015. [Online]. Available: http://www.computerwoche.de/a/auto-hacks-2015-remote-gefahr,3215128.
[3] D. K. Oka, A. Yvard, S. Bayer and T. Kreuzinger, „Enabling Cyber Security Testing of Automotive ECU's by Adding Monitoring Capabilities", in escar Europe, Munich, 2016.