Hella Security testing framework
Integration of Security test methods on different test levels throughout the validation / verification process

Katharina Lohmann, M.Sc.

Vector Cyber Security Symposium
Stuttgart, 23.06.16
HELLA Group Overview

- Partner of the automotive industry and the aftermarket for over 100 years
- Global family-owned company with more than 100 locations in over 35 countries
- Sales of €5.8 billion FY 2014/2015
- Approx. 32,000 employees worldwide, thereof over 6,000 in Research & Development
- One of the top 40 automotive suppliers in the world and one of the 100 largest German industrial companies

Business Segments

<table>
<thead>
<tr>
<th>Automotive</th>
<th>Aftermarket</th>
<th>Special Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lighting</td>
<td>Electronics</td>
<td></td>
</tr>
<tr>
<td>• Headlamps</td>
<td>• Body Electronics</td>
<td></td>
</tr>
<tr>
<td>• Rear Lamps</td>
<td>• Energy Management</td>
<td></td>
</tr>
<tr>
<td>• Small Lamps</td>
<td>• Driver Assistance Systems</td>
<td></td>
</tr>
<tr>
<td>• Interior Lighting</td>
<td>• Sensors</td>
<td></td>
</tr>
<tr>
<td>• Lighting Electronics</td>
<td>• Actuators</td>
<td></td>
</tr>
<tr>
<td>• Electric Power Steering</td>
<td>• Electric Power Steering</td>
<td></td>
</tr>
<tr>
<td>Independent Aftermarket</td>
<td>Workshop Equipment</td>
<td>Special Original Equipment</td>
</tr>
<tr>
<td>• Parts</td>
<td>• Full Range</td>
<td>• Original equipment for commercial vehicles, e.g. buses, caravans, heavy duty machinery with lighting and electronics</td>
</tr>
<tr>
<td>- Wear Parts</td>
<td>- Parts</td>
<td>- Street Lighting</td>
</tr>
<tr>
<td>- Spare Parts</td>
<td>- Tools</td>
<td>- Interior Lighting</td>
</tr>
<tr>
<td>- Accessories</td>
<td>- Workshop Concepts</td>
<td></td>
</tr>
<tr>
<td>- Tools</td>
<td>- Services</td>
<td></td>
</tr>
<tr>
<td>- Services</td>
<td>- Information</td>
<td></td>
</tr>
<tr>
<td>- Technical Service</td>
<td>- Local Branch Network</td>
<td></td>
</tr>
<tr>
<td>- Sales Support</td>
<td>- Logistics</td>
<td></td>
</tr>
<tr>
<td>Wholesale</td>
<td>Workshop Equipment</td>
<td>Special Original Equipment</td>
</tr>
<tr>
<td>• Vehicle Diagnostic &amp; Vehicle Data</td>
<td>- Air Conditioning Service</td>
<td></td>
</tr>
<tr>
<td>• Tools</td>
<td>• Lighting Service</td>
<td></td>
</tr>
<tr>
<td>• Workshop Concepts</td>
<td>• Battery Service</td>
<td></td>
</tr>
<tr>
<td>• Services</td>
<td>• Tools</td>
<td></td>
</tr>
<tr>
<td>• Information</td>
<td>• Local Branch Network</td>
<td></td>
</tr>
<tr>
<td>• Local Branch Network</td>
<td>- Logistics</td>
<td></td>
</tr>
<tr>
<td>• Logistics</td>
<td>• Accessories</td>
<td></td>
</tr>
<tr>
<td>• Accessories</td>
<td>• Tools</td>
<td></td>
</tr>
</tbody>
</table>

Industries

- Street Lighting
- Interior Lighting
- Industry Lighting
HELLA Group
Regional market coverage by location of end customers FY 2014/2015

Sales Automotive by regions FY 2014/2015

13% Germany
33% Europe without Germany
25% NAFTA/South America
29% Asia-Pacific/Rest of World

Sales* of € 4.4 billion

Presence boosted in global growth markets

Sales Aftermarket and Special Applications by regions FY 2014/2015

15% Germany
16% Europe without Germany
8% NAFTA/South America
61% Asia-Pacific/Rest of World

Sales* of € 1.4 billion

Business focus Europe ("regional business")

* with third-party companies, without intersegment sales
31,864 employees* are working for HELLA at more than 100 locations in over 35 countries

* As of balance sheet date (May 31, 2015)
Agenda

1. Introduction & Motivation
2. Security Testing Methods
3. Summary & Outlook
Agenda

1. Introduction & Motivation

2. Security Testing Methods

3. Summary & Outlook
Introduction
Combined Approach Functional Safety & Automotive Security

- Safety (accident prevention) & Security (attack prevention) = Freedom from intolerable risk
- Quality and Reliability require Safety & Security
- Goal: reduce risk to a tolerable level by combining all aspects
Introduction
Combined Approach Functional Safety & Automotive Security

- Safety (accident prevention) & Security (attack prevention) = Freedom from intolerable risk
- Quality and Reliability require Safety & Security
- Goal: reduce risk to a tolerable level by combining all aspects

Security test methods shall be integrated into Quality + Safety compliant process
Motivation

Security test methods shall be integrated into Quality + Safety compliant process by using a combination of known and implemented methods – why?

- To use synergies from already known methods
- To reduce additional effort
- Security test methods mostly known from IT-Security -> usability and benefit for automotive not yet clearly known

Under consideration of:

- ISO 26262
- SAE J 3061
- A-SPICE
Overview Security Testing Methods
IT-Security Test Methods

Secure embedded systems / device architecture

Shall be covered by already known and implemented methods as far as possible
Functional Testing

Correctness
- Algorithms (AES, RSA), protocols, Security functions (e.g. secure diagnostics)
  - positive tests

Robustness
- Wrong inputs, error handling
  - negative tests

Performance
- Execution time, memory usage
  - positive tests; negative tests

Security Testing Methods

Security related instances can be tested by official test vectors
## Security Testing Methods

### Functional Testing

#### Current status & benefits
- Functional requirements based testing done anyway (easy and useful to integrate Security here)
- Implemented Security mechanisms are verified according their expected and defined behavior
- Performance of the mechanism is verified

#### Open points and challenges
- Covers only functionality and no real attacks

---

#### Benefits and integration of Functional Testing in Hella Process
- ENG. 6, ENG. 8, ENG. 10: Requirements based tests (correctness, robustness and performance)
- ENG. 7, ENG. 9: Interface based tests (robustness and performance)

→ Functional Testing is already included in Hella process and master test strategy

---

#### Combination and integration at Hella
Security Testing Methods
Vulnerability Scanning

**Type 1**
Software/firmware is scanned statically and dynamically

Security design guidelines like CERT C Secure Coding Standard are quite new for automotive industry

**Type 2**
Whole system is scanned

System scans regarding open ports and interfaces (JTAG/XCP) and also for available services running on the interfaces (diagnosis) to access critical functions without authentication

Examines in relevant applications, source codes, networks and backend infrastructure all known automotive Security vulnerabilities
Security Testing Methods
Vulnerability Scanning

Current status & benefits

• “Easily” done by a scanning tool if database with vulnerabilities is available
• Classic bus systems are already tested by Hella regarding standards and known problems (from the OEMs and Hella itself)
• Code can be analyzed dynamically & statically

Open points and challenges

• Currently only official database for scanning Ethernet vulnerabilities available (no CAN, LIN, …)
• Static code analysis should be enhanced regarding Security ruleset
• CAN Conformance Test should be enhanced regarding Security requirements

Benefits and integration of Vulnerability Scanning in Hella Process

• ENG. 6: Static code analysis (CERT/MISRA) & Dynamic code analysis
• ENG. 7: Static code analysis (CERT/MISRA) & Dynamic code analysis & interface test
• ENG. 8: Interface test (entry point, stimulation for functional testing)
• ENG. 9: Interface test
• ENG. 10: Interface test (entry point, stimulation for functional testing) & CAN Conformance Test

→ All test methods are basically already included in Hella process and master test strategy

Combination and integration at Hella
Security Testing Methods - Example

Vulnerability Scanning

QA-C (Vulnerability Scanning)

- Existing tools (QA-C with MISRA) were enlarged regarding Security (CERT Secure Coding)
- Investigations on a pilot-project result in many rule violations regarding Security
- Typical rule violations: undefined function, undefined behavior, buffer overflows,…
- Analyzed and evaluated in system context and rated as uncritical
- Focus for future projects should be on manually programmed code
- Theoretically possible to use these rule violations as entry points
- Useful to apply QA-C; analyze and evaluate results by developer
- Define measures like watchdog, MPU,…
# Security Testing Methods

## Systematic Fuzzing

<table>
<thead>
<tr>
<th>Fuzzing</th>
<th>White-Box</th>
<th>Black-Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamical test</td>
<td>(Code coverage and internal values are observed by debugger)</td>
<td>(System return values are monitored)</td>
</tr>
<tr>
<td>method</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Plain Fuzzing</strong></td>
<td>1. Generating random input values</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Insert input values into target system</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Monitor and evaluate behavior of system in order to detect errors in program flow</td>
<td></td>
</tr>
<tr>
<td><strong>Generation-based Fuzzing</strong></td>
<td>1. Generating random input values</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Insert input values into target system</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Monitor and evaluate behavior of system in order to detect errors in program flow</td>
<td></td>
</tr>
</tbody>
</table>

Induce program failures to detect unknown Security issues

Exposing the interfaces to unexpected, invalid and random input and analyze reaction of system
Security Testing Methods
Systematic Fuzzing

Current status & benefits
- Combines requirements based negative testing and interface testing
- Can be largely automated
- Can find Security errors, which likely won’t be triggered during normal use of testing
- Little or no information are needed about test target

Open points and challenges
- Often used with UNIX systems and ASCII inputs
- Detects just simple faults and vulnerabilities
- CAN Conformance Test should be enhanced regarding Security requirements

Benefits and integration of Fuzzing in Hella Process
- ENG. 6: Interface test
- ENG. 7, 8: Interface test & equivalence class & boundary value analysis
- ENG. 9: Interface test & equivalence class & boundary value analysis & stress test
- ENG. 10: Interface test & equivalence class & boundary value analysis & stress test & CAN conformance test

Combination and integration at Hella

→ All test methods are basically already included in Hella process and master test strategy
### Security Testing Methods - Example

#### Systematic Fuzzing

<table>
<thead>
<tr>
<th>Conformance Test (Fuzz Testing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Checks, if protocol standards are maintained -&gt; out of intended range failures can occur</td>
</tr>
<tr>
<td>• Existing CAN Conformance Test incl. strategy enlarged regarding Security</td>
</tr>
<tr>
<td>• Generate automatic input data</td>
</tr>
<tr>
<td>• Keep ECU active and insert data</td>
</tr>
<tr>
<td>• Observe behavior of ECU while running for a while</td>
</tr>
<tr>
<td>• „Easily integrated“</td>
</tr>
<tr>
<td>• Enlarging of test specification and its implementation and execution on-going</td>
</tr>
<tr>
<td>• Results outstanding</td>
</tr>
</tbody>
</table>
Security Testing Methods
Penetration Testing

1. Planning
   - Limiting system scope and boundaries
   - Set up focus and intensity of attack

2. Discovering
   - Interfaces, components and their connections are listed + observed

3. Attack
   - Execute attack
   - Collect information and re-consider for further attacks
   - Discovered vulnerabilities are noticed + evaluated regarding its criticality

Common method to find unknown Security vulnerabilities e.g. undocumented debug access interfaces

Penetration Testing
Simulates realistic and physical attack

- White-Box Test (Full spec + documentation)
- Grey-Box Test (Partial information)
- Black-Box Test (No information)
## Security Testing Methods
### Penetration Testing

<table>
<thead>
<tr>
<th>Current status &amp; benefits</th>
<th>Open points and challenges</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Simulates a realistic physical attack on the embedded system</td>
<td>• Requires reverse engineering (for black box)</td>
</tr>
<tr>
<td></td>
<td>• In short time only „easy“ vulnerabilities will be discovered</td>
</tr>
<tr>
<td></td>
<td>• Should be done externally (independency given)</td>
</tr>
<tr>
<td></td>
<td>• Requires very experienced knowledge</td>
</tr>
<tr>
<td></td>
<td>• Time and cost consuming</td>
</tr>
</tbody>
</table>

**Benefits and integration of PenTesting in Hella Process**

- ENG. 8 / 9 : Fault injection (incl. error guessing)

→ Fault injection partly included in Hella process and master test strategy

**Combination and integration at Hella**
Agenda

1. Introduction
2. Security Testing Methods
3. Summary & Outlook
Summary & Outlook

Summary

Security Testing

- Functional Testing
- Penetration Testing
- Vulnerability Testing
- Fuzz Testing
Summary & Outlook

Summary

Security Testing

Functional Testing
Requirements based
Positive
Negative

Penetration Testing
Fault Injection
(Error Guessing)

Fuzz Testing
Interface Test
Stress Test
Equivalence Class
Boundary Value
Conformance Test

Vulnerability Testing
Static Code Analysis
Dynamic Code
Analysis
Interface Test
Conformance Test

Possibility and easiness to integrate methods into process
Summary & Outlook

Outlook

• Continue with validations in several projects (e.g. static code analysis with CERT, CAN conformance)
  • Develop recommendations and add ons regarding Security
  • Deeper investigations regarding Penetration Testing
  • Establish recommendation on test methods regarding the Security criticality
Thank you for your attention

Katharina Lohmann, M.Sc.
Functional Safety & Security Expert
Hella KGaA Hueck & Co
Center of Competence Safety
Beckumer Straße 130
59552 Lippstadt, Germany
Web  www.hella.com
Phone  +49/ (0)2941/ 38-32880
Email  Katharina.Lohmann@hella.com
## Security Testing Methods

**Hella Approach as an example for ENG.7 + ENG.10**

<table>
<thead>
<tr>
<th>Functional Testing</th>
<th>Vulnerability Scanning</th>
<th>Systematic Fuzzing</th>
<th>Penetration Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ENG.7 Software Integration Test</strong></td>
<td>• Code of software is scanned statically and dynamically</td>
<td>• Software architecture interfaces are tested regarding unexpected/random values</td>
<td>• Penetration Testing should be done in an advanced development level</td>
</tr>
<tr>
<td>• Focus on testing robustness and performance of security implementations between modules</td>
<td>• Software system can be scanned for open ports and interfaces and also for available services running on the interfaces</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Combinable with existing Hella test method</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ENG.10 System Test</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Testing of system security protocols and functions</td>
<td>• System is scanned for open ports and interfaces</td>
<td>• System interfaces are tested regarding unexpected values</td>
<td>• No access to system components</td>
</tr>
<tr>
<td>• Requirements based tests</td>
<td>• Whole system configurations are checked</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Combinable with existing Hella test method</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Requirements based tests</td>
<td>• Interface tests</td>
<td>• Interface tests</td>
<td>• ---</td>
</tr>
<tr>
<td>• positive: functionality of protocols</td>
<td>• CAN / LIN Conformance Test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• negative: error handling</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Related Links