Webinar: Automotive Cyber Security

Christof Ebert
Vector Consulting Services
Agenda

- **Welcome**
  - Motivation
  - Safety meets Security
  - Practical Guidelines
  - Case Study
  - Conclusions and Outlook
Christof Ebert

Prof. Dr. Christof Ebert is managing director at Vector Consulting Services.

He supports clients around the world to improve product strategy and product development and to manage organizational changes. Prior to that, he held global management positions for ten years at Alcatel, then ICT world market leader.

A trusted advisor for companies around the world and a member of several of industry boards, he is a professor at the University of Stuttgart and at Sorbonne in Paris. He authored several books including the most recent “Global Software and IT” published by Wiley.

christof.ebert@vector.com  www.vector.com/consulting
Welcome

Vector Consulting Services

- ...supports clients worldwide
- ...improving product development, IT and providing interim management
- ...offers with the Vector Group a portfolio of tools, software components and services
- ...is as Vector Group globally present with 1500 employees and well over 300 Mio. € sales
- ...continuously hiring

www.vector.com/consulting
Vector Offers a Comprehensive Portfolio for Cyber Security

Vector Cyber Security and Safety Solutions

- Security and Safety Consulting
- AUTOSAR Basic Software
- HW based Security
- Tools (PLM, Architecture, Test, Diagnosis etc.)

Engineering Services for Safety and Security
Agenda

Welcome

- **Motivation**
  - Safety meets Security
  - Practical Guidelines
  - Case Study
  - Conclusions and Outlook
Motivation

Software Drives Value – and Risks

- **Smart solutions**
  Products adapt to the changing environment and use

- **Smart networking**
  Software, mechanics and electronics to create new technologies, business models and ways of working

- **Smart environments**
  Cooperating systems to communicate and exchange information

- **Smart production systems**
  Flexible production systems enable resource-efficient production and custom products
Different Topics – Coherent Threats

**Function Safety**
- Health
- Accident
- ISO 26262
- HARA, FTA, FMEA
- ASIL / SIL
- Fail safe / operational
- Redundancy

**Cyber Security**
- Assets
- Attack
- Governance
- TARA
- Authenticity
- Cryptography, IDIP
- Key management

**Privacy**
- Personal data
- Data breach
- Privacy laws
- TARA
- Authorization
- Cryptography
- Explicit consent

**Liability ➔ Risk management ➔ Systematic engineering**
Motivation

Who Doesn’t Learn from History Is Doomed to Repeat It

1980s: IT Systems were
- Complex
- Distributed
- Software Intensive
- Perceived as secure

Then came the Morris worm

2015: Industry Systems are
- Complex
- Distributed
- Software Intensive
- Perceived as secure

Then came ...

Critical systems are insecure. By definition.
A 100% security solution is not possible.
Advanced risk assessment and mitigation is the order of the day.
Risk-Based Security Engineering

Motivation

- Asset
- Attack
- Threat
- Attack Potential
- Security Goal

Stakeholders (e.g., owner, driver, OEM)

- has value for
- is performed against
- requires
- causes
- risk is reduced by
- has
- is achieved by

- Threat Agent (e.g. hacker)
- Security Engineering

- Security Case, Audit, Compliance
- Security Validation
- Test Security Mechanisms
- Technical Security Concept
- Security Implementation
- Security Goals and Requirements
- Assets, Threats and Risk Assessment
**Motivation**

**Terminology**

- **Security**
  The sum of all attributes of a system which contribute towards ensuring that it can neither be accidentally nor deliberately be attacked or manipulated.

- **Safety**
  The sum of all attributes of a system which contribute towards ensuring that it is free from unacceptable risks and dangers.
  All safety-critical electronic systems are security-critical.

- **Cyber physical system**
  A system of collaborating electronic systems controlling physical entities.

- **Cyber security**
  An attribute of a cyber-physical system that relates to avoiding unreasonable risk due to an attack.

- **Attack**
  Exploitation of vulnerabilities to obtain unauthorized access to or control of assets resulting in harm or undesirable behavior.

- **Threat**
  A circumstance or event with potential to cause harm related to financial, operational performance, safety, reputation, privacy and/or sensitive data.
Agenda

Welcome
Motivation

- Safety meets Security
  Practical Guidelines
  Case Study
  Conclusions and Outlook
Innovation...

Unlike the awkward-looking first Google cars, the incumbent automotive companies, are showing concept vehicles that just look like cars.

...Quality

50% of the automotive recalls are due to software problems.

The recall rate of cars is exploding. In 2014 5 times the number of sold cars are recalled in the same timeframe.
Quality Requirements Are Business-Critical

People and environment attack the system thus creating harm or unintended behavior.

People and environment can be harmed by malfunctions.

**Security:** Prevention of harm or unintended behavior

**Safety:** Prevention of harm and injuries

Focus on systematically and reliably achieving quality requirements.
Reduce complexity from endless feature lists.
### Recent Standard Evolution

<table>
<thead>
<tr>
<th>Standard</th>
<th>Scope</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEC 61508</td>
<td>Functional safety for electronic systems</td>
<td>Functional safety guidance for electronic systems; too high-level and generic; today considered inappropriate</td>
</tr>
<tr>
<td>ISO 26262</td>
<td>Automotive functional safety</td>
<td>Clear focus on automotive electronic systems with good coverage of entire life-cycle; revision in 2016</td>
</tr>
<tr>
<td>ISO 15408</td>
<td>Evaluation criteria for IT security</td>
<td>Focus on IT systems; provides 7 evaluation assurance levels (EAL) for security requirements; made common criteria a standardized practice</td>
</tr>
<tr>
<td>ISO 27001</td>
<td>Information security management systems</td>
<td>Governance requirements for security engineering across the entire value chain; widely used in IT</td>
</tr>
<tr>
<td>IEC 62443</td>
<td>Industrial communication network security</td>
<td>Strong view on distributed industry automation systems with technologies and governance</td>
</tr>
<tr>
<td>SAE J3061</td>
<td>Automotive cyber security</td>
<td>Close relationship to ISO 26262; difficult to apply in stand-alone to cyber security</td>
</tr>
</tbody>
</table>
Safety meets Security

Security Directly Impacts Safety

**Functional Safety (IEC 61508, ISO 26262)**

- Hazard and risk analysis
- Functions and risk mitigation
- Safety engineering

Security demands implicitly addressed

+ Security (ISO 15408, J3061)

- Threat and risk / remedy analysis
- Misuse cases and mitigation
- Security engineering

Security and Safety are different!

But for efficiency reasons and fast start security engineering can be connected to safety framework
Safety meets Security

Safety and Security must be addressed in parallel

Innovative functionality...
- Distributed systems
- Complex feature interaction
- High data volume
- External interfaces (V2X; vehicle as IP node)

... Drives new challenges
- Fail-safe and fail-operational behaviors
- High-performance micro-controllers
- Software development for critical systems
- Safety functions must be secured against attacks
- Cost-effective evolution and support over the entire life-cycle

Need to efficiently implement quality requirements
Agenda

Welcome
Motivation
Safety meets Security

» Practical Guidelines
Case Study
Conclusions and Outlook
Focus on rigorous security engineering, because most security attacks are input/output and implementation related, and rarely lie within the cryptographic protocols and algorithms.
Practical Guidelines

Security Engineering

Assets, Threats and Risk Assessment

Security Goals and Requirements

Technical Security Concept

Security Implementation

Security Validation

Test Security Mechanisms

Security Case, Audit, Compliance

Security Verification
First and Pivotal: Determine Assets to be Protected

**Security policies**, must be enforced throughout the life-cycle and all operational scenarios.

Authorized entities (e.g. hardware modules, software processes, users) must have **proper and timely access** to their data and services.

Only **authorized access** to protected resources must be possible.

The **origin** of information (e.g. message, data) or a component (e.g. ECU, firmware) must be verifiable.

Only **authorized modifications** of information or components must be possible and unauthorized modifications must be **detectable**.
## Threat Examples

<table>
<thead>
<tr>
<th>Property</th>
<th>Threat</th>
<th>Definition</th>
<th>Example</th>
</tr>
</thead>
</table>
| Authenticity   | Spoofing        | Gain capabilities without proper authorization     | • Pretending to be an ADAS element, which sends an emergency brake signal.  
• Allow a remote internet user to send signals on the vehicle bus.  
• Activate features without paying for them. |
| Integrity      | Tampering       | Modifying data or code                             | • Modifying a software code executed in an ECU, or a frame transmission as it traverses the bus system.                                         |
| Governance     | Repudiation     | Claiming to have not performed an action.           | • “I did not use the motorway “, “I did not modify the mileage counter”                                                                                                                                   |
| Confidentiality| Information Disc | Exposing information                               | • Allow reengineering of SW IP. Publish payment data on the web.                                                                                                                                         |
| Availability   | Denial of Service| Deny or degrade service to users                   | • Switch car into limp home mode.  
• Delay emergency brake signal.  
• Crash navigation system.  
• Deny access to necessary cloud services. |
Practical Guidelines

Security Engineering

- Assets, Threats and Risk Assessment
- Security Goals and Requirements
- Technical Security Concept
- Security Implementation
- Security Verification
- Test Security Mechanisms
- Security Validation
- Security Case, Audit, Compliance
### Practical Guidelines

#### Determine Necessary Security Level with TARA Results

<table>
<thead>
<tr>
<th>No.</th>
<th>Asset ID</th>
<th>Asset / Vehicle Function</th>
<th>CIAAP</th>
<th>Attack vector</th>
<th>Threat ID</th>
<th>Threat</th>
<th>Expertise</th>
<th>Window of Opportunity</th>
<th>Equipment / Effort</th>
<th>Threat level (high=4; low=1)</th>
<th>Impact Level</th>
<th>Security level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ast 2</td>
<td>Business model</td>
<td>Auth</td>
<td>Exploiting a vulnerability of ECU</td>
<td>Th-1</td>
<td>Unpaid functional upgrades</td>
<td>Expert</td>
<td>Medium</td>
<td>Tailored</td>
<td>2</td>
<td>Mod. Injuries</td>
<td>Medium</td>
</tr>
</tbody>
</table>

![Security Level Diagram](https://via.placeholder.com/150)

**Security Level (SL) vs. Impact Level (IL)**

<table>
<thead>
<tr>
<th>Threat Level (TL)</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>QM</td>
<td>QM</td>
<td>QM</td>
<td>QM</td>
<td>Low</td>
</tr>
<tr>
<td>1</td>
<td>QM</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>2</td>
<td>QM</td>
<td>Low</td>
<td>Medium</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>3</td>
<td>QM</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>4</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
<td>Critical</td>
<td></td>
</tr>
</tbody>
</table>
## Practical Guidelines

### Mapping Security Levels to Security Mechanisms

<table>
<thead>
<tr>
<th>Level</th>
<th>Prevent</th>
<th>Detect</th>
<th>Forensic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>High</td>
<td>+</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Medium</td>
<td>+</td>
<td>+</td>
<td>++</td>
</tr>
<tr>
<td>Low</td>
<td>(+)</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>QM</td>
<td>O</td>
<td>O</td>
<td>+</td>
</tr>
</tbody>
</table>

**Examples**

- Network/process/information separation
- Encryption, digital signatures
- Key management
- Access control
- Firewall
- Intrusion prevention systems (IPS)

- Intrusion detection systems (IDS)
- Monitoring

- Logging
- Security issue knowledge base
- Analysis and investigation of digital evidence

O No recommendation for or against approach

+ Approach is recommended for security level

++ Approach is highly recommended for security level
Practical Guidelines

Security Engineering

- Assets, Threats and Risk Assessment
- Security Goals and Requirements
- Technical Security Concept
- Security Implementation
- Security Verification

- Security Validation
- Test Security Mechanisms
- Security Case, Audit, Compliance
1. Secure System Status
   - Goal: Ensure a secure system status at any time
   - Method:
     > Use secure boot of the OS
     > Apply intrusion detection on network
     > Separate networks (gateways, ID proxies, firewalls)
     > Use highest protections in mixed-criticality systems

2. Secure communication with right partner
   - Goal: Ensure authenticity (“right partner?”)
   - Method:
     > Certificates, asymmetric encryption (private / public key)
     > Whitelists and blacklists for access control

3. Secure communication with trusted partner
   - Goal: Protect communication (modification and eavesdropping)
   - Method:
     > Secure channel (PKI) to distribute symmetric session keys
     > HW-supported symmetric encryption
Security by Design: Separate Concerns and Networks

Practical Guidelines
Goal: Separate security privileged functions from the applications by hardware

Approach: Secure Hardware Extension

- On-chip extension to microcontroller
- Secure boot directly triggered by hardware upon start
- Pre-shared cryptographic key
- Memory for secure storage of (cryptographic) data
- Hardware extension for cryptographic primitives
Security by Design: AUTOSAR Architecture

**Crypto Abstraction Library – CAL**
- BSW, CDD or SWC use CAL by inclusion
- Memory allocated by caller
  - Enables re-entrance

**Crypto Primitive Library – CPL**
- SW implementation of cryptographic primitives

**Crypto Service Manager – CSM**
- SWC use CSM through RTE
- BSW/CDD use CSM by inclusion
  - Asynchronous operation possible
    - Callback indicates application

**Crypto library module – CRY**
- Implementation of cryptographic primitives
  - Usage of SW or crypto HW
Practical Guidelines

Security by Design: Secure Coding

- **Goal**
  - Avoid design and code errors which can lead to security exploits

- **Approach**
  - **Use a hardened OS with secure partitioning**
    Avoid embedded Linux due to its complexity and rapid change and thus many security gaps, (e.g. NULL function pointer dereferences, which allow hackers to inject executable code).
  - **Deploy secure boot strategy**
    Starting with first-stage ROM loader with a pre-burned cryptographic key, the next levels are verified before executing to ensure authenticity of each component of the boot.
  - **Apply rigorous static code analysis**
    Tools such as Coverity and Klocwork offer many security checks, such as NULL pointer dereferences, memory access beyond allocated area, reads of uninitialized objects, buffer and array underflows, resource leaks etc.
  - **Use modified condition/decision coverage (MC/DC)**
    Detect backdoors
Goal
Consistent security evaluation and certification of products and protection profiles

Approach
ISO 15408: 7 Evaluation Assurance Levels (EAL) for security requirements
ISO 27001: techniques for security engineering

Applicability
Operating systems, key management systems, ICs, smart cards, crypto libraries, ...
Adapt and adopt common criteria for critical systems, such as automotive, automation, aerospace, defense,

Benefits
- Clarify security targets and consistency in architecture and PLM
- Reduce risk and cost by balancing and focusing best practices

Tailored protection profile combined with systematic safety/security engineering provide a thorough yet cost-effective solution.
Security Engineering

- Assets, Threats and Risk Assessment
- Security Goals and Requirements
- Technical Security Concept
- Security Implementation
- Security Verification
- Test Security Mechanisms
- Security Validation
- Security Case, Audit, Compliance
Practical Guidelines

Security Test Tools

- Static Code Analyzer
- Dynamic Code Analyzer
- Encryption Cracker
- Vulnerability Scanner
- Network Traffic Analyzer
- Network Stress Tester
- Hardware Debugger
- Fuzz Tester
- Known Answer Tester
- Application Tester
- Interface Scanner
- Exploit Tester

Apply different tools for cyber security test and hardening
Secure Software Distribution, e.g. OTA

Ensure that deployed software is always known and secure

- Data encryption: Protection of intellectual property by encryption
- Authorization: Protection against unauthorized ECU access
- Validation: Safeguarding of data integrity in the flash memory
- Authentication: Verification of authenticity through signature methods
Integrated Safety and Security Engineering

Security needs to be an integrated part of the life-cycle
Agenda

Welcome
Motivation
Safety meets Security
Practical Guidelines

» Case Study
Conclusions and Outlook
Case Study

Advanced Driver Assistance System – Overview

**ADAS Basic Functions (Use cases)**

- Warns driver when vehicle is getting too close to preceding vehicle
- Warns driver if vehicle is leaving the driving lane
- Performs action such as counter-steering or braking if required
Step 1: Define the assets of the system which shall be protected

- **A1:** Messages received (e.g. steering angle, lane information) or send by the LKA-System (warning message, counter steering request)
- **A2:** Software of the ADAS-System
- **A3:** Security keys
ADAS – Step 2: Threat and Risk Analysis (TARA)

- Identification of attacks should be performed without taking into account potential security mechanisms
- Assess attack potential (e.g. STRIDE, etc.):
  - Expertise required to perform an attack
  - Available knowledge about the system to perform an attack
  - Window of opportunity required to perform an attack
  - Equipment required to perform an attack
- Use expert judgment, based on available rating scales

Attacks:
- A1-AT1: Messages for braking send are manipulated
- A1-AT2: Messages are replayed

Threats:
- A1-AT1-T1: Manipulation of braking message can lead to the vehicle suddenly braking with different damages
- A1-AT2-T1: Replay of warning messages at critical situations can lead to erroneous behavior and massive driver distraction
Case Study

ADAS – Step 3: Security Goals

Security goals are high level security requirements

- **A1-AT1-T1-SG1**: The system shall prevent manipulation of the messages send by the driver assistance system

- The communication between driver assistance and sensors shall prevent manipulations of messages

- The MAC shall be calculated by a SHE-compliant hardware trust anchor using the algorithm RSA2048

- The MAC shall be truncated after x byte
# ADAS – Step 3: Security Goals

## Case Study

### CIAAG

Confidentiality, Integrity, Availability, Authenticity, Governance

### Resulting Security Goals

Maximum (Safety, Financial)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Platform (TBC)</td>
<td>Ast 2</td>
<td>Braking to prevent collision</td>
<td>A</td>
<td>Tht-1</td>
<td>Driver crashes into preceding car. Passengers in both cars are severely wounded or killed.</td>
<td>Expert</td>
<td>Medium</td>
<td>Sensitive</td>
<td>Bespokes</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
<td>No impact</td>
<td>No impact</td>
<td>Critical</td>
<td>Medium</td>
<td>SG1</td>
<td>If requested the brakes shall be activated</td>
</tr>
<tr>
<td>2</td>
<td>Platform (TBC)</td>
<td>Ast 2</td>
<td>Braking to prevent collision</td>
<td>I</td>
<td>Tht-2</td>
<td>Braking although not authorized, e.g. &gt; 10 km/h</td>
<td>Expert</td>
<td>Medium</td>
<td>Sensitive</td>
<td>Bespokes</td>
<td>Low</td>
<td>Life-threatening or fatal injuries</td>
<td>High</td>
<td>No impact</td>
<td>No impact</td>
<td>Critical</td>
<td>High</td>
<td>SG2</td>
<td>Unauthorized braking shall be avoided.</td>
</tr>
<tr>
<td>3</td>
<td>Platform (TBC)</td>
<td>Ast 1</td>
<td>IPR of functions</td>
<td>C</td>
<td>Tht-3</td>
<td>RCTA function becomes public knowledge</td>
<td>Expert</td>
<td>High</td>
<td>Public</td>
<td>Bespokes</td>
<td>Medium</td>
<td>No injuries</td>
<td>High</td>
<td>No impact</td>
<td>No impact</td>
<td>Critical</td>
<td>High</td>
<td>SG3</td>
<td>RCTA function shall remain secret.</td>
</tr>
</tbody>
</table>
ADAS – Step 4: Security Mechanisms (1/3)

- Braking while driving with speed > 10 km/h
  - OR
  - Deliberate Manipulation
    - OR
    - Overtake Brake ECU
    - Manipulation of Radar Object on CAN Bus
      - AND
      - Write message to CAN
      - Create correct message on CAN
  - Systematic / Random HW Fault
    - Plausibility Checks, e.g. Vehicle Speed, Engine_Status
Case Study

ADAS – Step 4: Security Mechanisms (2/3)

- Write message to CAN
- Overtake ECU on same CAN Bus
- Create correct message on CAN
- AND
- Know-How CAN message
- Create authenticated CAN message
- AND
- Secure Communication
- Secure Download
- Enter programming Session (0x27)
- Flash Firmware on ECU
- Connection to ECU
- Know-How Firmware
- Access to Flash
- Secure Diagnostics
Case Study

ADAS – Step 4: Security Mechanisms (3/3)

Secure Diagnostics

• No Keys on Diagnostic Tool
• Secure Access with organizational access management and guidelines

Secure Internal Communication

• Efficient encryption and message authentication (e.g., H-MAC)
• Rationality Checks (e.g., Vehicle speed < 10 km/h)

Secure Download

• PKI with RSA-2048
• Closing Programming Interface

Secure Implementation

(e.g. Standard Architecture, Design Rules, Coding Guidelines, Process Rules, etc)

Reduce likelihood of attack
Agenda

Welcome
Motivation
Safety meets Security
Practical Guidelines
Case Study

- Conclusions and Outlook
Security starts is a management responsibility
- Cyber security is relatively new in many companies
- Cyber security principles, process and terminology are needed that can be commonly understood between key stakeholders
- All functions are exposed to security threats, both engineering and management

Need for risk-based cyber security
- By definition no system can be guaranteed 100% secure
- Similar to the functional safety approach, cyber security must be designed in to the system
- Threat Analysis and Risk Assessment (TARA), vulnerability analysis and penetration testing are some of the key activities to be systematically rolled-out
Conclusions and Outlook

Implement a Thorough Safety and Security Culture

**Standard Software**

**Technical measures** to protect hardware and software security

Examples: **Robustness and Hardening** in AUTOSAR, Security adjusted to safety integrity needs

**Tools**

Consistent approach for all **development activities**.

Examples: **Threat and Hazard analysis** during concept definition, **consistent modeling** in PREEvision

**Consulting**

Support for **methods** and **skills** as well as the necessary **cultural changes**.

Examples: **Vector Security Check**, Security engineering, Incident management

Check our comprehensive cyber security portfolio:  
www.vector.com/security
Vector Cyber Security Symposium 2016

- **Date**
  
  Thu. 23. June 2016

- **Event location**

  **Stuttgart, Germany**
  Kongresszentrum Liederhalle

- **Topics**

  - Experiences with cyber security at OEMs and TIER1s
  - Interaction between functional safety and cyber security
  - Practical experiences: Resilience, test, cryptography and security standards
  - Automotive security trends in North America and Asia
  - Exhibition and discussion opportunity with the Vector product specialists

- **Agenda and Registration**

  www.vector.com/vses2016
Conclusions and Outlook

Cyber Security: Vector Trainings and White Papers

Training “Cyber security”
- Stuttgart
- Tue. 5. Jul. 2016
- In-house at your preferred location feasible
- www.vector.com/vcs-training

Boost your security competence

Free white papers
Thank you for your attention.
For more information please contact us.

Vector Consulting Services
Your Partner in Achieving Engineering Excellence
Phone +49 711 80670-0  www.vector.com/consulting
Fax +49 711 80670-444  consulting-info@vector.com