EVITA-Project.org:
E-Safety Vehicle Intrusion Protected Applications

7th escar Embedded Security in Cars Conference
November 24–25, 2009, Düsseldorf

Dr.-Ing. Olaf Henniger, Fraunhofer SIT Darmstadt

Hervé Seudié, Robert Bosch GmbH

Presentation outline

- Project overview
- Overview of technical work packages
  - Security requirements engineering
  - Secure on-board architecture design
  - Security architecture implementation
  - Prototype-based demonstration
- Summary and outlook

Administrative project details

- **Programme**
  - FP7-ICT-2007 of the European Community
- **Research Area**
  - ICT-2007.6.2 ICT for Cooperative Systems
- **Funding scheme**
  - Collaborative project
- **Budget / Funding from European Community**
  - € 6,022,807 / € 3,825,993
- **Start date / End date / Duration**
  - 1 July 2008 / 30 June 2011 / 36 months
- **Coordinator**
  - Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V.
- **Project Website**
  - http://www.evita-project.org

EVITA project objectives

- **Objectives**
  - To design, verify, and prototype a secure architecture for automotive on-board electronics networks.
- **Motivation**
  - *In-vehicle* IT security (trust anchor, secure storage of secret keys etc.) is required as a basis for secure *inter-vehicular* communication.
- **Approach**
  - Hardware security modules at root of trust.
  - Open specifications

EVITA project partners

BMW Group Research and Technology
BOSCH
Continental
escrypt Embedded Security
EURECOM Sophia Antipolis
Fraunhofer ISI
Fraunhofer SIT
FUJITSU
infineon
KATHOLIEKE UNIVERSITEIT LEUVEN
MIRA
TELECOM ParisTech
TRIALOG

EVITA project structure

Work packages

Dissemination and external interfaces
- Open specifications
- Liaison with related initiatives in the field of eSafety

Security Requirements Analysis	Secure On-board Architecture Design	Security Architecture Prototyping	Validation & Demonstration
• Use cases • Threat & risk Assessment • Legal aspects	• Trust model • Software/ hardware partitioning • HW security module design • SW security design & protocols • Model-based verification	• Based on FPGAs and software • Partial model-based code generation • Code validation & test	• Prototypical integration in vehicular environment • Showing safety applications based on V2X communication

Milestones

- M1 (Q1 2009): Risks & Requirements available
- M2 (Q4 2009): Security & Trust Model available
- M3 (Q2 2010): Secure On-*board* Architecture & Protocol Specification available
- M4 (Q4 2010): Model based Verification available FPGA Prototype & SW framework available
- M5 (Q2 2011): Final validation & Demonstrator available

Presentation outline

- Project overview
- Overview of technical work packages
  - Security requirements engineering
  - Secure on-board architecture design
  - Security architecture implementation
  - Prototype-based demonstration
- Summary and outlook

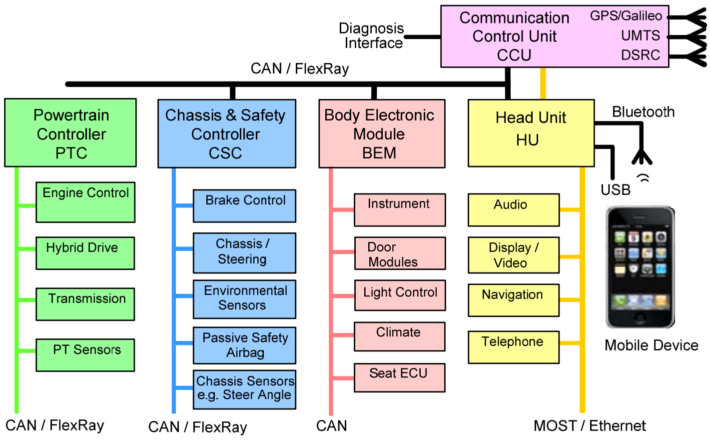
Security requirements engineering – Overview

- Description of system under investigation and use cases
- Identification of IT security threats
- Identification of IT security requirements to counter the threats
- Assessment of the risks associated with the threats and prioritization of the IT security requirements based on the risks addressed
- Analysis of legal requirements

Assumed automotive on-board network architecture

Assets

- On-board electronic components such as ECUs, sensors, and actuators
- Links between components and within ECUs
- Software on the ECUs



Use case categories

- Vehicle-to-vehicle and vehicle-to-infrastructure communication
- Use of nomadic devices, USB sticks, or MP3 devices
- Aftermarket and workshop/diagnosis

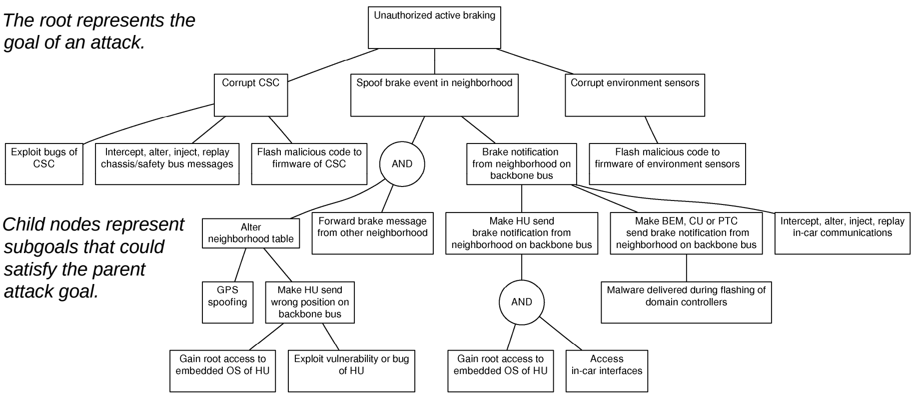
Possible attack goals

- To gain advantages or just to harm others e.g. by
  - enhancing traffic privileges (like forcing green lights ahead),
  - fraudulent commercial transactions (like manipulating toll bills),
  - hoaxes (like unauthorized active braking),
  - avoiding liability for accidents,
  - information theft,
  - identity theft

Example attack tree

The root represents the goal of an attack.

Child nodes represent subgoals that could satisfy the parent attack goal.



IT security requirements

- Say what needs to be protected, but not how
- Based on compact functional models derived from use case descriptions, independent from implementation
- Main approach
  - Incoming data and their origins shall be authentic.
  - Outgoing data shall be confidential to an appropriate level.

Summary of security requirements

- **Integrity of hardware security module**
  - Prevention/detection of tampering with hardware security modules
- **Integrity and authenticity of in-vehicle software and data**
  - Unauthorized alteration of any in-vehicle software must be infeasible / detectable
- **Integrity and authenticity of in-vehicular communication**
  - Unauthorized modification of data must be detectable by the receiver
- **Confidentiality of in-vehicular communication and data**
  - Unauthorized disclosure of confidential data sent or stored must be infeasible.
- **Proof of platform integrity and authenticity to other (remote) entities**
  - Capability to prove the integrity and authenticity of its platform configuration
- **Access Control to in-vehicle data and resources**
  - Enabling availability and well-defined access to all data and resources

Risk analysis

- **Risk** associated with an attack is a function of:
  - **Severity** of impact (i.e. harm to stakeholders)
  - **Probability** of successful attack
- Not possible to quantify severity and probability in many applications
  - but qualitative rankings allow relative severity, probability and risk to be identified

Security threat severity classification

Class	Safety	Privacy	Financial	Operational
S0	No injuries.	No data access.	No financial loss.	No impact on operation.
S1	Light/moderate injuries.	Anonymous data only (no specific user or vehicle).	Low level loss (~€10).	Impact not discernible to driver.
S2	Severe injuries (survival probable). Moderate injuries for multiple units.	Vehicle specific data (vehicle or model). Anonymous data for multiple units.	Moderate loss (~€100). Low losses for multiple units.	Driver aware. Not discernible in multiple units.
S3	Life threatening or fatal injuries. Severe injuries for multiple units.	Driver identity compromised. Vehicle data for multiple units.	Heavy loss (~€1000). Multiple moderate losses.	Significant impact. Multiple units with driver aware.
S4	Fatal for multiple vehicles.	Driver identity access for multiple units.	Multiple heavy losses.	Significant impact for multiple units.

Attack potential and probability of success

- **Attack potential**
  - corresponds to the minimum effort required to create and carry out an attack
  - evaluation using established structured approach from "Common Criteria" taking into account the required
    - time, expertise, knowledge of system, window of opportunity, and equipment
- Indicative of **probability of success**
  - Inverse relationship: Easy attacks more likely to be successful than difficult ones.
  - Numerical scale used to represent relative ranking of probability of success

Attack potential		Probability of success	
Rating	**Descrip-tion**	**Likeli-hood**	**Ranking**
0–9	Basic	Highly likely	5
10–13	Enhanced basic	Likely	4
14–19	Moderate	Possible	3
20–24	High	Unlikely	2
≥25	Beyond high	Remote	1

Sample asset attack ratings

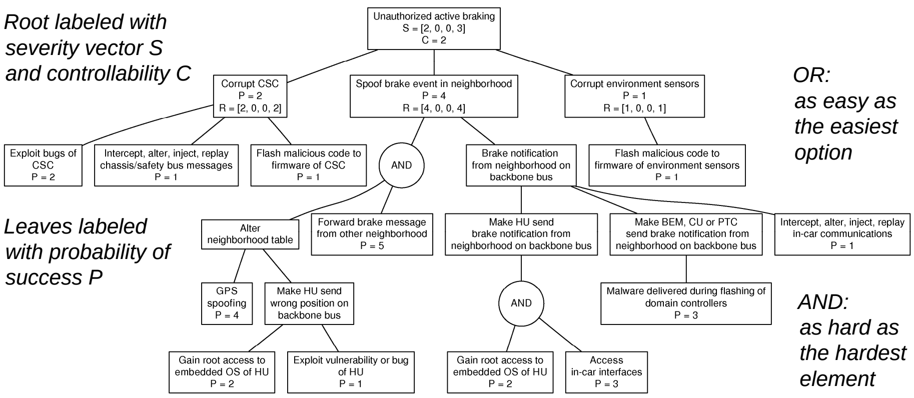
Attack	Required attack potential		Probability of success
	Value	Rating	
Forward brake message from other neighbourhood	8	Basic	5
GPS spoofing	11	Enhanced-Basic	4
Access in-car interfaces	14	Moderate	3
Gain root access to embedded OS of HU	21	High	2
Flash malicious code to firmware of environment sensors	41	Beyond High	1

Risk mapping table (for situations controllable by driver)

Risk level R		Probability of success P				
		P=1	P=2	P=3	P=4	P=5
Severity S_i	S_i=1	0	0	1	2	3
	S_i=2	0	1	2	3	4
	S_i=3	1	2	3	4	5
	S_i=4	2	3	4	5	6

The less controllable the situation by the driver, the higher the safety-related risk.

Sample risk analysis



Prioritising security requirements

- Security requirements mapped to attacks
- Summary of risk analysis
  - collates results from risk assessment of all attack trees
  - identifies risk levels found from attack trees and the number of their occurrences
- Interpretation
  - few instances and/or low risk suggest low priority for protection
  - high risk and/or many instances suggest higher priority for protection

Presentation outline

- Project overview
- Overview of technical work packages
  - Security requirements engineering
  - Secure on-board architecture design
  - Security architecture implementation
  - Prototype-based demonstration
- Summary and outlook

Secure on-board architecture design – Overview

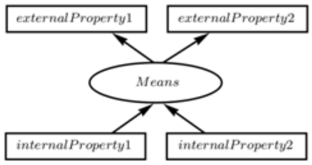
- Design a toolkit of security measures (software, hardware, and architectural) that can be selected for implementation in future automotive on-board systems
  - Model Driven Engineering (MDE) approach under development
- Formal verification of security properties of Security Building Blocks"

Fraunhofer SIT Security Modeling Framework

- Describes system behaviors as (sets of) sequences (traces) of actions
- Actions associated with agents (entities) in the system
- Satisfaction of security properties depends on the agents' view of the system
  - Authenticity = agent is certain of occurrence of an action
  - Confidentiality = action parameter (e.g. sender or message contents) is indistinguishable for all other agents

Security engineering with formal model approach

- Describe protocols/mechanisms as Security Building Blocks (SeBB)
- Refine security requirements (*external properties*) through *means* to hardware/contractual roots (*internal properties*)



Hardware Security Module as security anchor

- **Main goal**
  - Providing secure platform for cryptographic functionalities that support use cases
- **Features**
  - Secure Storage
  - Hardware Cryptographic Engines
  - Secure CPU Core
  - Scalable Security Architecture
- **Advantages**
  - Flexibility
  - Extendability
  - Migration Path from existing SW solutions

Options of general structure of hardware security modules

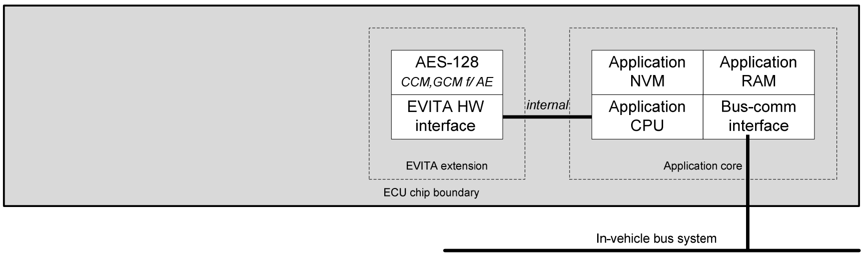
- **HSM physically separated from CPU**
  - Less secure than a single chip: connection between CPU and HSM not secure.
  - Suitable for short-term designs or low-security applications with very small production runs
  - Expensive: extra chip costs more due to the extra pins,
- **HSM in the same chip as the CPU but with a state machine**
  - More secure than external chip and more cost-effective
  - Not flexible: Hardware not modifiable, but automotive µC life cycle is more than 20 years
  - Suitable for very high security applications with very short lifetimes
  - Cryptographic applications will need to be implemented at the application CPU level: possible performance issues.
  - Changing a state machine requires hardware redesign and is very expensive
- **HSM in the same chip as the CPU but with a programmable secure core**
  - proposed solution
  - Secure and cost-effective
  - Flexible because of programmable core
  - Usable for other industries

Classes of Hardware Security Modules

- Light HSM
  - Security module applicable e.g. for sensors
- Medium HSM
  - Selected security functions e.g. required for a gateway or router
- Full HSM
  - Provides security for very critical application requiring powerful security
  - Enabled by enough resources of the ECU

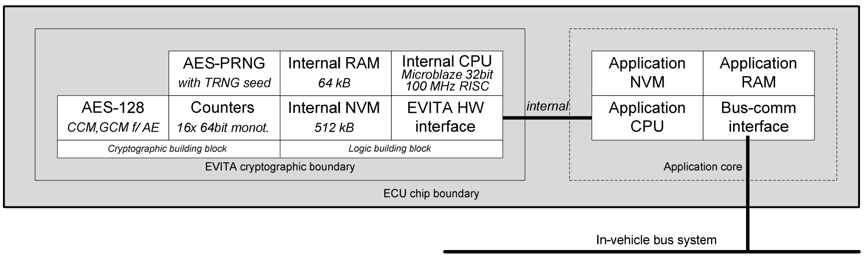
Topology of EVITA light version HSM

- sensor/actuator level



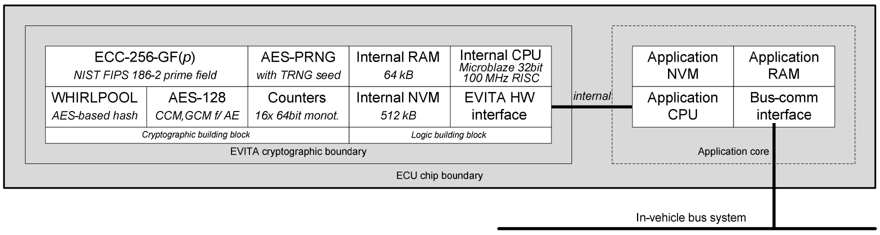
Topology of EVITA medium version HSM

- ECU Level



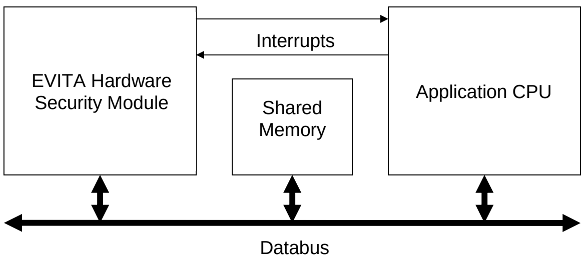
Topology of EVITA full version HSM

- ECU Level – V2X



Hardware interface between HSM and application CPU

- HSM and application CPU have write/read rights for the Shared Memory
- Trigger through interrupts
- Optional polling: periodic check of the result buffer



Presentation outline

- Project overview
- Overview of technical work packages
  - Security requirements engineering
  - Secure on-board architecture design
  - Security architecture implementation
  - Prototype-based demonstration
- Summary and outlook

Security architecture implementation – Overview

- Prototype a secure on-board **hardware architecture** using a standard automotive controller with an FPGA acting as Hardware Security Module (secure crypto-coprocessor)
- Prototype a secure on-board **software architecture**, i.e. hardware drivers, basic software extensions (e.g., crypto library), and necessary security protocols
- **Validate** functional compliance, security **compliance**, partitioning (i.e. SW/HW, light/medium/full), performance, and costs of hardware and software implementation

Presentation outline

- Project overview
- **Overview of technical work packages**
  - Security requirements engineering
  - Secure on-board architecture design
  - Security architecture implementation
  - **Prototype-based demonstration**
- Summary and outlook

Prototype-based demonstration

- inside a lab car demonstrating e-safety applications based on vehicle-to-X communication
- to come

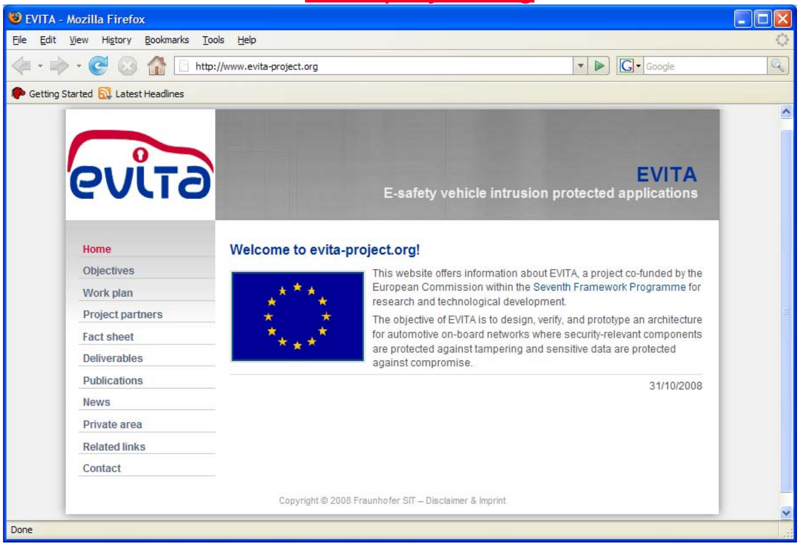
Presentation outline

- Project overview
- Overview of technical work packages
  - Security requirements engineering
  - Secure on-board architecture design
  - Security architecture implementation
  - Prototype-based demonstration
- Summary and outlook

Summary and outlook

- **Summary**
  - Goal: Securing in-vehicular applications and components
  - Achievements so far
    - Security requirements analysis based on threat analysis
    - Design of three classes of HSMs
    - Design of a security software architecture based on AUTOSAR
- **Next Steps**
  - Open specification of soft- and hardware design and protocols: Input for standardization
  - Proof-of-concept by formal verification
  - Prototypical implementation using the AUTOSAR stack CUBAS from Bosch
  - Integration into a demonstrator

More information: Visit evita-project.org



Thank you for your attention.

Dr.-Ing. Olaf Henniger
Fraunhofer Institute SIT
olaf.henniger@sit.fraunhofer.de

Dipl.-Inf. Hervé Seudié
Robert Bosch GmbH
herve.seudie@de.bosch.com