Identification of Authenticity Requirements in Systems of Systems by Functional Security Analysis

Andreas Fuchs and Roland Rieke
{andreas.fuchs,roland.rieke}@sit.fraunhofer.de

Fraunhofer Institute for Secure Information Technology SIT, Darmstadt, Germany

Jun 2009

Overview

1. Motivation
   - Scenario - cooperative reasoning in vehicular ad hoc communication
   - Dependence of safety critical decisions raises security concerns
2. Objectives
   - Systematic security requirements elicitation for novel architectures
   - Avoid premature architecture constraints
3. Functional Security Analysis
4. Results and Outlook

Why think about new vehicular Architecture using SoS reasoning

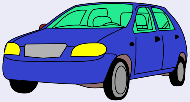
overall goal

reduce number and impact of accidents in Europe

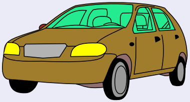
difficulties

to improve safety measures in vehicles $\rightsquigarrow$ improve infrastructure

cooperative approach



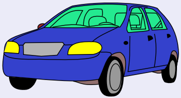
$\Rightarrow$ warning $\Rightarrow$



vehicular communication systems can be more effective in avoiding accidents and traffic congestion than current technologies where each vehicle tries to solve these problems individually

Use case: send danger warning

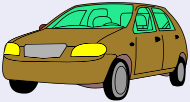
sense(ESP,SlipperyWheels)
positioning(GPS,position)



send(CU,danger(position,type))

⟶

receive(CU,danger(position,type))
positioning(GPS,position)



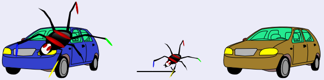
show(HMI,D,warn(relative-position))

ESP - Electronic Stability Protection
GPS - Global Positioning System
CU - CommunicationUnit
HMI - Human Machine Interface
D - Driver

Security is an enabling Technology for novel SoS Applications

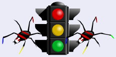
Exposing vehicles to the Internet makes them vulnerable

- Attacks on safety
  - Unauthorized brake
  - Attack active brake function
  - Tamper with warning message



  - Attacking E-Call
  - On-Board Diagnostics (OBD) flashing attack

- Attacks on privacy
  - Trace vehicle movement
  - Compromise driver privacy
- Manipulate traffic flow
  - Simulate traffic jam for target vehicle
  - Force green lights ahead of attacker



  - Manipulate speed limits
  - Prevent driver from passing toll gate
  - Engine refuses to start
- Increase/Reduce driver's toll bill

Security Requirements Engineering Process

- the identification of the target of evaluation and
  the principal security goals and
  the elicitation of artifacts (e.g. use case and threat scenarios)
  as well as risk assessment
- the actual security requirements elicitation process
- a requirements categorisation and prioritisation,
  followed by requirements inspection

Further steps in Security Engineering

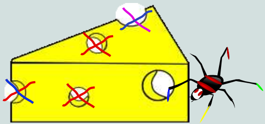
- security requirements (structural) refinement
- mapping of security requirements to security mechanisms

Methods to elicit security requirements

- misuse cases (attack analysis),
- anti-goals derived from negated security goals,
- use Jackson‘s problem diagrams,
- actor dependency analysis (i^* approach)

Why yet another approach ?

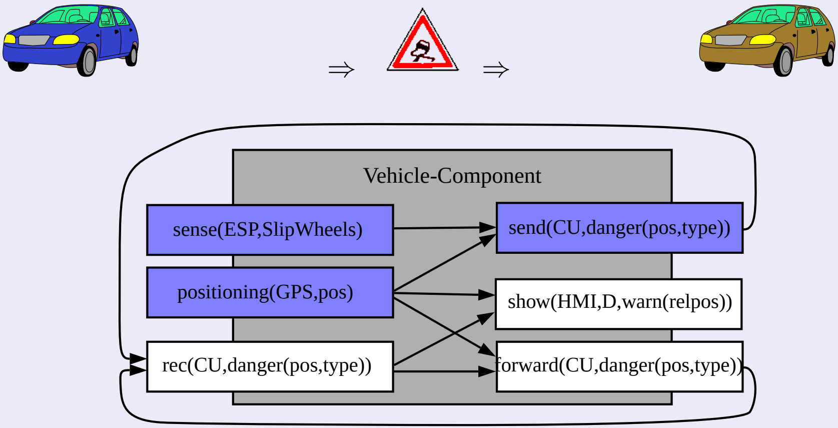
Completeness



Avoid premature architecture constraints

- protocols SSL/TLS/VPN/IPv6
- trust anchor TPM
- infrastructure PKI, PDP/PEP
- end-to-end/hop-by-hop

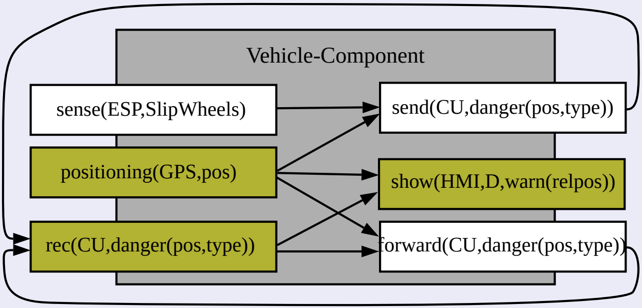
Functional Component Model



Security goal of the system at stake:

Whenever a certain output action happens, the input action that presumably led to it must actually have happened.

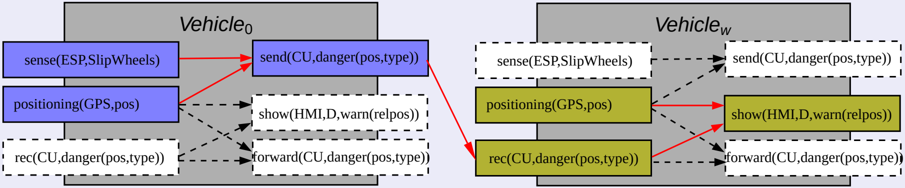
Functional Component Model



Security goal of the system at stake:

Whenever a certain output action happens, the input action that presumably led to it must actually have happened.

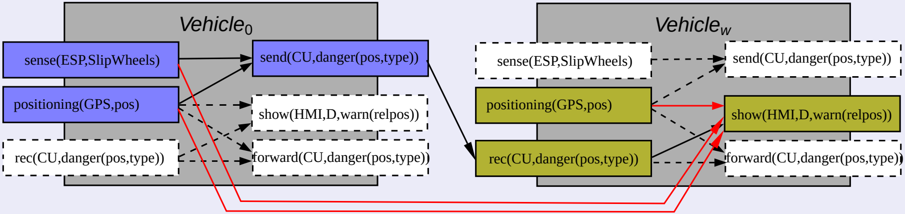
Functional security requirement identification



Formally, the functional flow among actions can be interpreted as an ordering relation ζ_i on the set of actions Σ_i in a certain system instance i.

$$\begin{aligned}\zeta_1 = \{ \; & (positioning(GPS_w, pos), show(HMI_w, D_w, warn(relpos))), \\ & (rec(CU_w, danger(pos, type)), show(HMI_w, D_w, warn(relpos))), \\ & (send(CU_0, danger(pos, type)), rec(CU_w, danger(pos, type))), \\ & (sense(ESP_0, SlipWheels), send(CU_0, danger(pos, type))), \\ & (positioning(GPS_0, pos), send(CU_0, danger(pos, type)))\}\end{aligned}$$

Functional security requirement identification



Restrict ζ_i^* to outgoing (max_i) and incoming boundary actions (min_i).

$$\chi_i = \{(x,y) \in \Sigma_i \times \Sigma_i \mid (x,y) \in \zeta_i^* \wedge x \in min_i \wedge y \in max_i\}$$

$$\begin{aligned}\chi_1 = \{ \; & (sense(ESP_0, SlipWheels), show(HMI_w, D_w, warn(relpos))), \\ & (positioning(GPS_0, pos), show(HMI_w, D_w, warn(relpos))), \\ & (positioning(GPS_w, pos), show(HMI_w, D_w, warn(relpos)))\}\end{aligned}$$

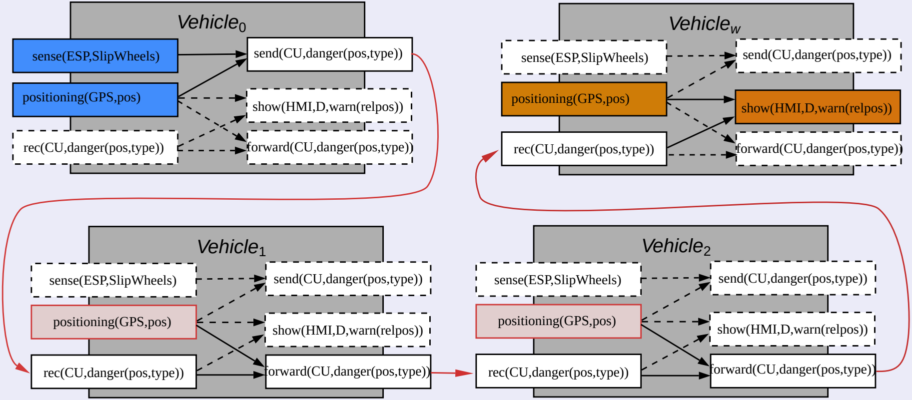
For all $x, y \in \Sigma_i$ *with* $(x,y) \in \chi_i$: $auth(x, y, stakeholder(y))$ is a requirement.

Resulting Authenticity Requirements

For all possible SoS instances for the action $show(HMI_w, D_w, warn(relpos))$ it must be authentic for the driver that:

1. $auth(positioning(GPS_w, pos), show(HMI_w, D_w, warn(relpos)), D_w)$ the relative position of the danger she is warned about is based on correct position information of her vehicle
2. $auth(positioning(GPS_0, pos), show(HMI_w, D_w, warn(relpos)), D_w)$ the position of the danger she is warned about is based on correct position information of the vehicle issuing the warning
3. $auth(sense(ESP_0, SlipWheels), show(HMI_w, D_w, warn(relpos)), D_w)$ the danger she is warned about is based on correct sensor data

System of Systems Instances



An analysis for the second instance will result in:

$$\chi_2 = \chi_1 \cup \{(positioning(GPS_1, pos), show(HMI_w, D_w, warn(relpos)))\}$$

And the third system of systems instance will result in:

$$\chi_3 = \chi_2 \cup \{(positioning(GPS_2, pos), show(HMI_w, D_w, warn(relpos)))\}$$

$$\chi_i = \chi_{i-1} \cup \{(positioning(GPS_{i-1}, pos), show(HMI_w, D_w, warn(relpos)))\}$$

Resulting Authenticity Requirements

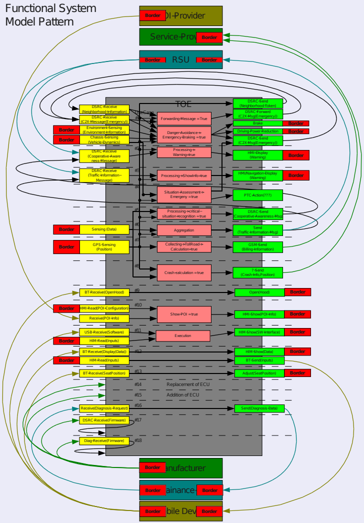
For all possible SoS instances for the action $show(HMI_w, D_w, warn(relpos))$ it must be authentic for the driver that:

1. $auth(positioning(GPS_w, pos), show(HMI_w, D_w, warn(relpos)), D_w)$
   the relative position of the danger she is warned about is based on correct position information of her vehicle
2. $auth(positioning(GPS_0, pos), show(HMI_w, D_w, warn(relpos)), D_w)$
   the position of the danger she is warned about is based on correct position information of the vehicle issuing the warning
3. $auth(sense(ESP_0, SlipWheels), show(HMI_w, D_w, warn(relpos)), D_w)$
   the danger she is warned about is based on correct sensor data
4. $\forall V_x \in V_{forward}$:
   $auth(positioning(GPS_x, pos), show(HMI_w, D_w, warn(relpos)), D_w)$
   position of forwarding vehicles is authentic
   - Breaking (4) would result in a smaller or larger broadcasting area.
   - This cannot cause the warning of a driver that should not be warned.
   - So it is NOT a safety related authenticity requirement.

EVITA (E-Safety Vehicle Intrusion Protected Applications)

In practice, the method has been applied in EVITA [a] to derive authenticity requirements for a new automotive on-board architecture

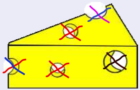
- 17 additional use cases, e.g.
  - safety reaction: active brake
  - traffic information
  - e-Tolling
  - eCall
  - remote car control
  - remote diagnosis/flashing
- 29 authenticity requirements elicited
- system model comprising 38 component boundary actions
- 16 system boundary actions (9 max, 7 min elements)



[a] http://www.evita-project.org/Deliverables/EVITAD2.3.pdf

Contribution of proposed approach

Identification of a consistent and complete set of authenticity requirements



For every safety critical action in a system of systems all information that is used in the reasonig process that leads to this action has to be authentic

Security mechanism independence

avoid to break down the overall security requirements to requirements for specific components or communication channels prematurely
$\leadsto$ requirements are independent of decisions on concrete security enforcement mechanisms and structure (e.g. hop-by-hop, end-to-end)

Formal base

approach fits to formal definition of security requirements

- Authenticity: A set of actions $\Gamma \subseteq \Sigma$ is authentic for $P \in \mathbf{P}$ after a sequence of actions $\omega \in S$ with respect to W_P if $alph(x) \cap \Gamma \neq \emptyset$ for all $x \in \lambda_P^{-1}(\lambda_P(\omega)) \cap W_P$.

Future work

- derivation of confidentiality requirements in a similar way (privacy)
- non-repudiation (relevant security goals from law)
- refinement throughout the design process (paper submitted to STM'09)
- mapping to adequate architectural structure and mechanisms to implement security measures (within EVITA context)

Thank you

evita Part of the work presented in this paper was developed within the project EVITA (`http://www.evita-project.org`) being co-funded by the European Commission within the Seventh Framework Programme.