Security Architectures for Software Updates and Content Protection in Vehicles

Ulrich Huber, Ahmad-Reza Sadeghi and Marko Wolf

Horst-Görtz Institut für IT Sicherheit

Horst-Goertz-Institute for IT-Security
Ruhr-University Bochum, Germany

October 12, Automotive 2006

Outline

Introduction

Flashable ECUs allow SW updates after delivery of the vehicle

- warranty-based updates
- correcting defective SW
- SW-based features sold in the after-market
- digital content (routing, LBS, multimedia)

SW protection required to prevent misuse

- IP theft, counterfeits
- unauthorized modifications
- unauthorized feature activation

Current ECUs

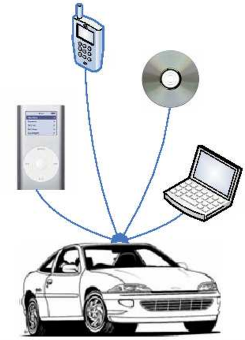
- provide low resistance to attacks of skilled adversaries
- SW protection based on cryptography and tamper-resistant HW still uncommon
- awareness of the need for SW protection exists (e.g, implementations using digitally signed SW updates)
- SW protection alone doomed to fail in hostile environment

Our Contribution

- model for security requirements of all relevant roles
- security analysis of all HW components and comm. channels
- breaking down overall requirements to component level
- enhance standard architecture with a secure HW component

Application Background

(a) Delivery and installation of SW components



(b) Delivery and usage of digital content

Roles

U User/owner, i.e., person that currently uses the vehicle

OEM Original Equipment Manufacturer assembles, sells, and delivers the vehicle

MSP Maintenance Service Provider maintains the user platform, i.e., repairs hardware components and/or updates software components with specific equipment

CP Content provider develops and distributes the content employed by User

Potential Adversaries

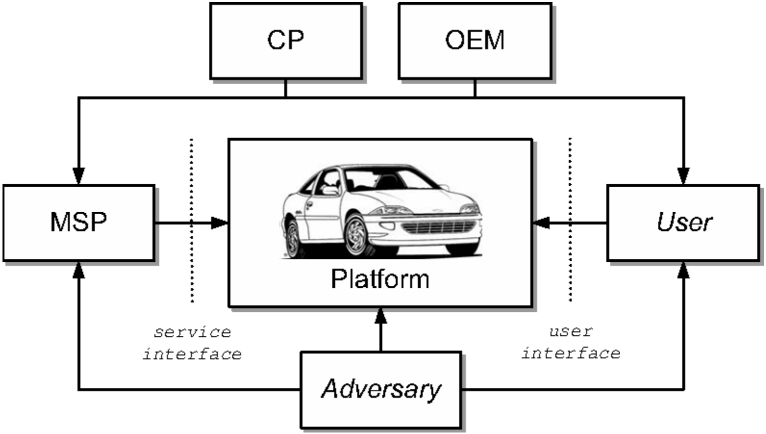
Today targets Theft (vehicle, components); Manipulation of mileage (resale, tax return) motor control unit (chip tuning), tachograph (legal driving restrictions, toll) . . .

Future targets Electronic license plate, Event data recorder, C2C & C2I communication, Infotainment . . .

Adversary	Capabilities	Access (additional to A_{i-1})
A_1 (usual user/owner)	Low	Communication channels
A_2 (sophist. owner, MSP)	Medium	Memory components
A_3 (competitor, org. crime)	High	CPU

Adversaries in the automotive scenario based on [Paar2003].

Roles and Relations



Security Requirements

☐	Straightforward
■	Advanced, Organizational

OEM
- Correctness
- Content integrity
- Content pre-selection (advanced)
- Non-repudiation (advanced)

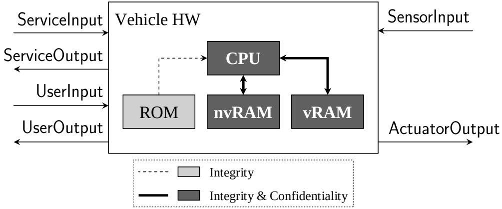
Content Provider (CP)
- Correctness
- Content integrity
- Access rights enforcement (advanced)
- Non-discrimination (advanced)
- Non-repudiation (advanced)

Maintenance service provider (MSP)
- Correctness
- Content integrity
- Non-discrimination (advanced)
- Non-repudiation (advanced)

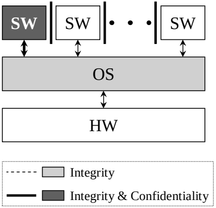
Vehicle Owner/User (U)
- Correctness
- Content integrity
- Content authenticity
- Non-repudiation (advanced)
- Privacy (advanced)

Breakdown of Req. for Standard Architecture

At first glance it seems that *all* HW and SW components as well as channels must fulfill the security requirements in order to defeat at least adversary A_2. However, this is too restrictive and increases the implementation cost due to the high cost of the secure HW and SW components that become necessary. An analysis on the component level allows to liberate some components from these two requirements.

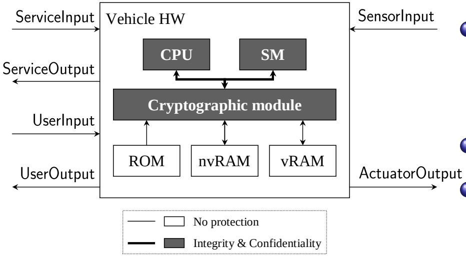


(a) HW architecture



(b) SW architecture

An Enhanced HW Architecture



- Crypto module for (external) confidentiality enforcement and integrity verification
- Small secure memory (SM)
- Defeats A_1 (comm) and A_2 (memory) w/o assumptions on security of ROM, nvRAM, and vRAM and corr. comm. channels
- Security assumptions only for CPU, SM & corr. channels (thus, both architectures fail to defeat adversary A_3)

Cryptographic Module

Capable embedded microprocessors, e.g. ARM or Atmel

- integrated cryptographic HW & secure memory
- proprietary (little flexibility)
- prob. oversized (resources, costs, . . .)

Customized controller

- fast & free cryptographic cores available (cf. paper)
- adaptable in performance & secure memory size
- maximum flexibility
- custom controller design (high volumes req. to bring costs down)

Conclusions

Implementation	Feasible Attacks	Add. Costs
Standard HW Architecture	A_1, A_2, A_3	None
Secure (Std) HW Architecture	A_3	High
Enhanced (Std) HW Architecture	A_3	Moderate

Enhancing standard architecture with a cryptographic module . . .

- significantly **reduces trust assumptions on memory** and all **corresponding communication channels**
- **defeats** even **sophisticated adversaries** (A_2)
- can be **implemented efficiently** w/ manageable changes
- causes only **moderate additional costs**

Thank you for your attention!

Security Architectures for
Software Updates and Content Protection in Vehicles

Ulrich Huber, Ahmad-Reza Sadeghi and Marko Wolf

Horst-Görtz Institut für IT Sicherheit

Horst-Goertz-Institute for IT-Security
Ruhr-University Bochum, Germany