AUTOMOTIVE SECURITY

PRODUCT FEATURES

JOHN COTNER

AMF-AUT-T2696 | JUNE 2017
THE CONNECTED CAR …

A cloud-connected computer network on wheels

• A networked computer
  – up to 100 ECUs per car
  – and many sensors
  – inter-connected by wires
  – more and more software

• Increasingly connected to its environment
  – to vehicles & infrastructure
  – to user devices
  – to cloud services
... IS AN ATTRACTIVE TARGET FOR HACKERS!

Valuable Data
- Collection of data/info
- Storage of data
- Diagnostic functions

High Vulnerability
- Increasing number of nodes
- More advanced features
- X-by-Wire

Easy (Remote) Access
- Fully Connected Car
- External & internal interfaces
- Wired & wireless interfaces

Increase Safety
- Prevent Unauthorized Access

Protect Privacy

Cloud Connection

Consumer Device Integration

In-Vehicle Network

Car2X
CAR HACKING IS ‘HOT’ …

Dieter Spaar, Translation: Fabian A. Scherschel

Beemer, Open Thyself! – Security vulnerabilities in BMW’s ConnectedDrive

How A 14-Year-Old Hacked A Car With $15 Worth Of Radio Shack Parts

Tesla Model S Digital Weaknesses To Be Exposed By Hackers Next Month
... AND IT’S REAL!

- Hackers took over the control of a Jeep, that was driving on the highway, from their basement
- Did it come as a surprise? Not really…


“2014 Jeep Cherokee, 2015 Cadillac Escalade and 2014 Toyota Prius were the most hackable.”

"The most hackable cars had the most [computerized] features and were all on the same network and could all talk to each other."

"The least hackable ones had [fewer] features, and [the features] were segmented, so the radio couldn't talk to the brakes."

Charlie Miller, security engineer

http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

01. WHAT IS SECURITY?
WHAT IS SECURITY?

• Security is a quality aspect…
  – Attackers should not be able to subvert the proper operation of a system
• …in an uncontrolled and evolving environment
  – Attackers do not obey to “the rules”
  – Attack(er)s only get better over time
• Security must be an integral part of the system design
  – Security is as strong as the weakest link → point solutions usually don’t work
  – Secure by design vs. security as an afterthought
• System security solutions are (usually) custom-made
  – Different use cases & architectures may (will) require different security solutions
  – But they often use generic building blocks
• 100% secure (or safe) does not exist in the real world
  – The challenge is to find the right balance between risk and protection (cost)
02.
OUR APPROACH TO SECURITY
Core Cyber Security Principles

• These are the core security principles:
  - Secure external interfaces
  - Secure domain isolation
  - Secure internal communication
  - Secure software execution

• They need to be in place in any E&E network
  - Regardless of the actual architecture and implementation
### Explanation of the Core Cyber Security Principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Concerns</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>secure external interfaces</strong></td>
<td>• Who is trying to access my network, and for what purpose?</td>
<td>• Only the OEM server can send firmware updates</td>
</tr>
<tr>
<td></td>
<td>• How to prevent spoofing, data manipulation and/or theft?</td>
<td>• Only real vehicles &amp; infra can send V2X messages</td>
</tr>
<tr>
<td><strong>secure domain isolation</strong></td>
<td>• How to separate domains (with different criticalities)?</td>
<td>• The infotainment system can receive position &amp; speed</td>
</tr>
<tr>
<td></td>
<td>• How should they (in a controlled manner) be able to interact?</td>
<td>• But it cannot send control messages to the brakes</td>
</tr>
<tr>
<td><strong>secure internal communication</strong></td>
<td>• Who is on my network? And who is sending messages?</td>
<td>• Only genuine ECUs can be installed in the network</td>
</tr>
<tr>
<td></td>
<td>• How to prevent data manipulation (or theft)?</td>
<td>• Messages cannot be replayed (repeated)</td>
</tr>
<tr>
<td><strong>secure software execution</strong></td>
<td>• How do I ensure that software cannot be modified (hacked)?</td>
<td>• Monitoring execution (run-time integrity checks)</td>
</tr>
<tr>
<td></td>
<td>• How do I enable secure updating (/ fixing) of the software?</td>
<td>• Secure boot / firmware image verification</td>
</tr>
<tr>
<td><strong>perimeter security</strong></td>
<td>• How to prevent people from getting close to the electronics?</td>
<td>• Install the electronics systems behind steel &amp; glass</td>
</tr>
<tr>
<td>*</td>
<td>• How to prevent unauthorized access to (/use of) a vehicle?</td>
<td>• (Electronic) door locks</td>
</tr>
</tbody>
</table>

* Perimeter security forms an important aspect of physical security for vehicles, but it is not a cyber security principle. As such, perimeter security is only listed here for completeness. In other words: one must not rely on perimeter security for the protection of the electronic systems against cyber attacks. For example, electronic car access systems must be protected against cyber attacks using the 4 principles listed above.
4 LAYERS TO SECURING A CAR

Layer 1: Secure Interface
Secure M2M authentication, secure key storage

Layer 2: Secure Gateway
Domain isolation, firewall/filter, centralized intrusion detection (IDS)

Layer 3: Secure Network
Message authentication, CAN ID killer, distributed intrusion detection (IDS)

Layer 4: Secure Processing
Secure boot, run time integrity, OTA updates
03.
Secure Element
SECURE ELEMENT – OVERVIEW

- Secure authentication of users, devices and messages
  - Automotive (M2M): V2X communication, Telematics (online services), …
  - Traditionally (user): banking cards, electronic passports, …

- Implemented on a tamper-resistant platform…
  - Securely hosts security applications and their confidential data
  - Protected against physical attacks (SCA, reverse engineering, …)
  - Proven security, via 3rd party evaluation and certification
    (Common Criteria, EMVCo, FIPS 140, …)

- …that provides:
  - secure crypto processing (AES, RSA, ECC, TRNG, …)
  - secure (crypto) key generation & storage
  - secure certificate handling (validate & store)
SECURE ELEMENT ≠ SECURE MCU

(Secure) MCUs are optimized for performance & flexibility:

- General purpose applications vs. crypto-only
- High performance, large memories, multiple interfaces
- Protection against logical attacks only (typically)

⇒ Increased robustness for in-car systems

Secure Elements are optimized for security:

- Crypto-only applications (payment, M2M authentication, …)
- Protection against “million dollar attack” (vs. 10 dollar attacks)
- Highly-secure architecture protects against physical attacks

⇒ Physical protection for external interfaces
LOGICAL SECURITY VS. PHYSICAL SECURITY

REMOTE / ONLINE / LOGICAL ATTACKS
- Targeting devices that are remotely accessible
- Attack Potential: (enhanced) basic

LOCAL / OFFLINE / PHYSICAL ATTACKS
- Targeting devices that live in a hostile environment
- Attack Potential: moderate to high
LOGICAL SECURITY VS. PHYSICAL SECURITY

REMOTE / ONLINE / LOGICAL ATTACKS
• Targeting devices that are remotely accessible
• Attack Potential: (enhanced) basic

LOCAL / OFFLINE / PHYSICAL ATTACKS
• Targeting devices that live in a hostile environment
• Attack Potential: moderate to high

FUNCTIONAL SECURITY
• “Internet security” with strong crypto, secure protocols, secure boot, e2e security, authentication
• Supported by hardware (for isolation, acceleration)
• Implementation is not important: a skilled attacker in possession of a device will hack it

PHYSICAL SECURITY
• Functional security, plus protection against physical attacks such as side-channel analysis, fault injection, reverse engineering, etc.
• Supported by dedicated, hardened hardware (providing a high level of tamper-resistance)
• Implementation of HW & SW matters: high resistance against a skilled attacker in possession of the device
LOGICAL SECURITY VS. PHYSICAL SECURITY

REMOTE / ONLINE / LOGICAL ATTACKS
- Targeting devices that are remotely accessible
- Attack Potential: (enhanced) basic

LOCAL / OFFLINE / PHYSICAL ATTACKS
- Targeting devices that live in a hostile environment
- Attack Potential: moderate to high

FUNCTIONAL SECURITY
- MCU / MPU / TRX

PHYSICAL SECURITY
- SMARTCARD
- smart card
- embedded Secure Element (eSE)
### Classification of Physical IC Attacks

**Countermeasures (“tamper-resistance”) require very specific IC designs...**

<table>
<thead>
<tr>
<th>Invasive Attacks</th>
<th>Semi-Invasive (Fault) Attacks</th>
<th>Non-Invasive Attacks</th>
</tr>
</thead>
<tbody>
<tr>
<td>“By permanent modification”</td>
<td>“By temporarily changing the state”</td>
<td>“By observing”</td>
</tr>
<tr>
<td>Reverse Engineering</td>
<td>Global And Local Light Attacks</td>
<td>Photo emission Analysis</td>
</tr>
<tr>
<td>Delayering</td>
<td>Alpha Particle Penetration</td>
<td>EMA Analysis</td>
</tr>
<tr>
<td>Micro-probing</td>
<td>Spike/Glitch injection</td>
<td>Timing Analysis</td>
</tr>
<tr>
<td>Forcing</td>
<td></td>
<td>SPA/DPA Analysis</td>
</tr>
<tr>
<td>Manipulation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Electron Microscopy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Atomic Force Microscopy (AFM)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contrast Etching</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Decoration</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Emerging attacks (in IoT/Automotive)**

**ChipWhisperer:** $130 kit (open source software, hardware and training). 
Objective: enable engineers/hobbyists to perform side-channel attacks.
SECURE ELEMENT – HARDWARE PLATFORM

Highest level of protection against the most advanced physical attacks

Features:

- A tamper-resistant platform, protecting against:
  - Side channel attacks, fault injection, physical reverse-engineering, …

- Securely hosts security applications and their confidential data

- Programmable platform, with Java Card OS
  - Interfaces: ISO 7816, I2C, SPI, NFC

- Cryptolib supported by accelerators:
  - (3)DES, AES, RSA, ECC, SHA(2), TRNG

- Proven security via 3rd party assessments
  - Common Criteria EAL6+, EMVCo, …
SECURE JAVA CARD OS (JCOP)
Rich platform, enabling fast software development
Location in the Vehicle E&E architecture

Secure TCU (external interface oriented)
Secure M2M authentication, secure key storage

Secure connected Gateway
Central node managing data streams and access rights (M2M, secure key storage), Domain isolation, firewall/filter, centralized intrusion detection (IDS).
Use Case Overview embedded Secure Element in Telematics Units, connected Gateways or V2X modules

Securing external interfaces
- Connection-oriented communication
  - Typical Use Cases: Cloud Services:
    - OTA FW Update,
    - Fleet management,
    - Connected Drive Services & Remote Diagnostics/Servicing,
    - OEM proprietary web services,
    - Payment Services

Securing external interfaces
- Ad-hoc communication
  - Typical Use Cases: V2X communication

Device/Feature authentication
- Centralized Key Mgmt
  - Typical Use Cases:
    - M2M authentication / feature unlocking
    - Key Management within IVN
    - Key Management of Virtual Car Keys
      (pairing of smartphones used as car key)

Specific Applications
- Potential Use Cases:
  - Secure storage of IDS logs
  - Secure storage of Vehicle Life Cycle Data
    (Mileage, Maintenance Status, …)
04. MCU / MPU
SECURE PROCESSING (MCU/MPU)

• Defined by hardware accelerated Crypto capability
• IP can be applied to any MCU/Processor
  - HW: MCU, AES accelerator, RNG, secure memory
  - SW: APIs incl. public key crypto, Hash
• Use cases:
  - CAN Message authentication
  - Secure boot – FW auth.
  - Key storage
  - Encryption
CURRENT SECURITY MODULES

Security Features

- CSE SHE compliant security module
- CSE enhanced by additional security features
- Programmable by customer (EVITA-Medium)
- Add flash-less device support
- SHE compliant cost optimised solution

Programmable Devices:
- MPC5746C
- MPC5748G
- MPC5777M
- MPC564xB/C
- MPC5777C
- MAC57D54H
- S32R247
- S32V
- S32K1
- CSE
- CSE2
- CSE3
- CSEc

Timeline:
- 2010
- 2011
- 2012
- 2013
- 2014
- 2015
- 2016
- 2017
- 2018
04.
i.MX 6 Security Features
i.MX Trust Architecture Features

**Trusted Execution**
- Isolates execution of critical SW from possible malware
- TrustZone Secure & Normal Worlds (processor modes)
- Hardware firewalls between CPU & DMA masters and memory & peripherals

**High Assurance Boot**
- Authenticated boot: prevents unauthorized SW execution
- Encrypted boot: protects SW confidentiality
- Digital signature checks embedded in on-chip boot ROM
- Run every time processor is reset

**HW Cryptographic Accelerators**
- i.MX family dependent
- Symmetric: AES-128, AES-256, 3DES, ARC4
- Asymmetric: RSA and ECC on some variants
- Message Digest & HMAC: SHA-1, SHA-256, MD-5
i.MX Trust Architecture Features (continued)

Secure Storage
- Protects data confidentiality and integrity
- Off-chip: cryptographic protection including device binding
- On-chip: self-clearing Secure RAM
- HW-only keys: no SW access

HW Random Number Generation
- Ensures strong keys and protects against protocol replay
- On-chip entropy generation
- Cryptographically secure deterministic RNG

Secure Clock
- Provides reliable time source
- On-chip, separately-powered real-time clock
- Protection from SW tampering
i.MX Trust Architecture Features (continued)

Secure Debug:
- Protects against HW debug (JTAG) exploitation for:
  - Security circumvention
  - Reverse engineering
  - Three security levels + complete JTAG disable

Tamper Detection
- Protects against run-time tampering
- Monitoring of various alarm sources
  - Debug activation
  - External alarm (e.g. cover seal)
  - SW integrity checks
  - SW alarm flags
- HW and SW tamper response
- Support varies by i.MX family
i.MX Trust Architecture – Overview

- **ARM CPU**
- **Secure Clock**
- **Peripheral Slave**
- **Secure RAM**
- **Accelerator (Cipher, Hash, RNG)**
- **DMA Master**
- **ROM (High Assurance Boot)**
- **Electrical Fuses (keys, security levels)**
- **External Memory**

- **Tamper Detect**
- **Secure Debug**
- **JTAG**
- **SW alarm**
- **GPIO alarm**
- **Erase**
- **HW Firewall**

- **Electrical Fuses** (keys, security levels)
Resource Domain Partitioning Concept
05. Memory and Peripheral Isolation for TrustZone Support
TEE Software Stack

Normal World
- Rich OS Applications
- Rich OS Libraries
- Rich OS
- BSP support
- GlobalPlatform Client API
- Trustzone Library
- Trustzone Driver

Secure World
- Trusted Applications
- Crypto service
- DRM service
- GlobalPlatform Internal API
- Crypto operation
- Data storage
- Task dispatcher
- Trusted OS
- BSP support
- Secure Timer
- Secure RAM
- OTP
- Crypto Engines
- Drivers

TEE Stack
- Trusted OS BSP
- Secure Applications
- Rich OS

TEE stack add-ons
06.

i.MX Security Configuration and Lifecycle
Security configurations

• Fab
  - FSL-internal configuration
    - Security state: all reachable
    - Debug: Full debug possible, security blocks unprotected
    - OTP key: available
    - HAB: failures non-fatal

• Open
  - Shipping configuration (FSL → OEM)
    - Non-secure end-product configuration (OEM → end user)
    - Security state: Non-Secure
    - Debug: configurable, security blocks scan-protected
    - OTP key: unavailable (test key)
    - HAB: failures non-fatal

• Closed
  - Secure end-product configuration (OEM → end user)
  - Security state: Trusted/Secure
  - Debug: configurable, security blocks protected
  - OTP key: available
  - HAB: failsafe policy

• Field Return
  - Field Return configuration (OEM → FSL)
  - Rest as for Open configuration
  - FSL Test modes re-enabled
07. NXP MCU Security Features
HSM Security Architecture

Features:

• Device life cycle scheme
• Unique ID for each device
• Debugger restrictions
• Flash Protection (TDM & PASS)
  - OTP
  - read / write & erase
  - diary to log erasing-steps
Flash Memory Protection

Non volatile flash memory consists of multiple blocks with different purpose and access possibilities:
- Read (location, master, lifecycle)
- Erase (location, master, lifecycle, OTP)
- Write (location, master, lifecycle, OTP)

The Password And Device Security Module (PASS) and the TamperDetectionModule (TDM) handle the access.
Hardware Security Module (HSM)
v1: MPC5746M / MPC5777M & v2: MPC5748G / MPC5746C

HSM is free programmable by the customer, additional security algorithm could implemented in software

Features:

- e200z0h core (v1: 100MHz / v2: 80 MHz)
- 4Kbytes Instruction cache
- Secure Debugger Interface
- Cryptographic Modules with AES-128, Random Number Generator, DMA
- Sensor Interface – monitor for voltage, temperature and clock (v1)
- Memory
  - SRAM (v1: 40 Kbytes / v2: 32 Kbytes)
  - Flash
    code: 2 x 64 Kbytes + 1 x 16KBytes
data : 2 x 16 Kbytes
SHE Firmware

- Release 1.0 is available for MPC574xG (3M & 6M)
- Firmware implements the CSE2 feature set (SHE firmware + Global-B requirements) on the HSM
- Firmware „emulates“ the CSE register interface, to simplify porting of existing SW stacks (e.g. Elektrobit)
- Firmware is delivered pre-programmed in the device
  - No SHE firmware programming and DCF configuration required by customer
08. Gateway PLATFORM
WHAT IS A GATEWAY?

- **Gateway is THE central node in the vehicle architecture**
  - Connects all the vehicle domains across all the interfaces (Ethernet, CAN FD, LIN)
  - Provides network isolation and security between functional domains and networks
  - Includes hardware accelerated crypto capability (HSM/CSE)
  - Transmits message to ECU on destination domain (adding secure signature to message)

- **~20% adoption in vehicle architecture today, moving to ~50% by 2020**
  - NXP will be #1 in this market by 2018
"We believe that the Tesla Model S is an archetype for what all cars will look like in the future – others will follow”

Marc Rogers

What every car company should do:

1. OTA Update process
   - Without customers having to subscribe to separate data service

2. Isolation of vehicle and infotainment systems
   - Have a “gateway”
   - Spend a lot of effort securing the “gateway”

3. Harden each component individually
   - Assume infotainment is compromised

M. Rogers
SECURE CONNECTIONS FOR A SMARTER WORLD