HOW TO CONNECT VEHICLE IN SAFE AND SECURE WAY

MIKKO HURSKAINEN
TECHNOLOGIST
17+ YEARS IN EMBEDDED SOFTWARE BUSINESS
200+ AUTOMOTIVE SOFTWARE PROJECTS DELIVERED
70+ TOP NOTCH PROFESSIONALS BUILDING THE PRODUCTS
5 LOCATIONS AROUND THE GLOBE

SHANGHAI OFFICE IN 2017 H2
SHENZHEN OFFICE IN 2017 H2
CONTENTS

• Connected vehicles
• What is security?
• Security solutions
• What’s next?
CONNECTED VEHICLES
CONNECTED VEHICLES

- Connected car market is experiencing rapid growth
- There's a need for secure and safe solutions

Source: Gartner
CONNECTED VEHICLE DEVELOPMENT MODEL

Vehicle Computer and Platform + SDK → Connected Vehicle
APPSTACLE PLATFORM

- European collaboration project for open connected car architecture
- Link Motion is promoting AGL
APPSTACLE ARCHITECTURE

Permission Control

Application runtime

APPSTACLE API
communication services

network IDS
in-vehicle connectivity

ex-vehicle connectivity

OTA manager
in-vehicle APPSTACLE platform

boot loader

Source: APPSTACLE ITEA program
WHAT IS SECURITY?
PROTECTION OF ASSETS

Assets

Vehicle theft

Threat
Exposure
Severity
Controllability

Distraction
Loss of control

SAE J3061
ISO 15288

Security technologies
Assets in connected vehicle

- **Data.** If data has been compromised, it can lead to hijacking of vehicle, lost property or manipulation of operation. Examples of data include remote control keys, maintenance data, routing information.

- **Privacy.** Lack of privacy can lead to uncomfortable situation or expose user to greater security risks. Examples of privacy assets include location information, route history and consumer habits.

- **Control.** Loss of control can lead to unwanted behaviour of vehicle during driving or even hijacking of passengers inside the vehicle. Loss of control also compromises owner’s ability to use car.

Tangible and intangible
THREATS

- Ransomware
- Publicized vulnerability
- Leakage of privacy data
- Blocking use of system

=> Remotely attack fleet
SAFETY AND SECURITY

Source: SAE J3061
SECURITY SOLUTIONS
SECURITY FEATURES

- Modularity and layering
- Hierarchical protection
- Attack surface minimization
- Least privilege principle
- Predicate permission
- Defense-in-depth

[Diagram showing the connection between ECU, Vehicle Access Controller, CAN gateway, Secure Container, Connected Application, and Internet.]
SANDBOXING OF THE SYSTEM

- Vehicle Access Controller
- Microcontroller
- Auto OS
  - Secure Container
  - Unprivileged container
- IVI OS
  - Unprivileged container
- Secure RTOS
- i.MX6Q+ Main Processor
DEFENSE IN DEPTH

- Minimizes impact of successful attacks
- Allows protection according to needs
- Innermost layer (TCB) is compact and most secure
VEHICLE NETWORK DATAFLOWS

Vehicle Network Gateway / Firewall

Abstract Interface

Auto OS

Secure Container

IVI OS

Secure RTOS

Configurable access

i.MX6Q+ Main Processor

Unprivileged container

Unprivileged container

Vehicle Access Controller

Microcontroller

CAN Bus

Internet

Very limited access

Wide access

Read access

Unprivileged container

Very limited access

Wide access

Read access
Vehicle Network Controlled Access

- Vehicle Access Controller
- Auto OS
- Secure Container
- Unprivileged container
- IVI OS
- Secure RTOS
- i.MX6Q+ Main Processor
- Microcontroller
- Vehicle Network

Vehicle Network Controller
SECURITY MINDED DESIGN PATTERN

- Follows automotive design patterns
- Separation of control, critical control and rich control
- Example: Diagnostics vECU
HARDWARE SECURITY TECHNOLOGIES

Secure Key Storage

High Assurance Boot and Chain of Trust

ARM TrustZone

i.MX6Q+ Main Processor

ARM Cortex-A9 Quad

CAAM

RAM

ROM Code

Load

Decrypted & Signed Image #1

Image data

Signature

Public Key

Decrypted & Signed Image #2

Image data

Signature

Public Key

Auto OS

Secure Container

IVI OS

Unprivileged container

Unprivileged container

Secure RTOS

ARM TrustZone

i.MX6Q+ Main Processor
MORE SECURITY SOLUTIONS

- Vehicle network protection
- Cryptography
- Intrusion detection system
- Open source development model
- External partners
- Research
- Training
WHAT’S NEXT
SECURITY FORMALIZATION

- Broader analysis
- NIST SP-800, SAE J3061, ISO 15288
- Privacy standards
- Integration to processes
- Secure System State
- Security Taxonomy
- Mathematical proofs
# SECURITY TAXONOMY

## SECURITY DESIGN PRINCIPLES

### Security Architecture and Design

<table>
<thead>
<tr>
<th>Principle</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear Abstraction</td>
<td>Hierarchical Trust</td>
</tr>
<tr>
<td>Least Common Mechanism</td>
<td>Inverse Modification Threshold</td>
</tr>
<tr>
<td>Modularity and Layering</td>
<td>Hierarchical Protection</td>
</tr>
<tr>
<td>Partially Ordered Dependencies</td>
<td>Minimized Security Elements</td>
</tr>
<tr>
<td>Efficiently Mediated Access</td>
<td>Least Privilege</td>
</tr>
<tr>
<td>Minimized Sharing</td>
<td>Predicate Permission</td>
</tr>
<tr>
<td>Reduced Complexity</td>
<td>Self-Reliant Trustworthiness</td>
</tr>
<tr>
<td>Secure Evolvability</td>
<td>Secure Distributed Composition</td>
</tr>
<tr>
<td>Trusted Components</td>
<td>Trusted Communication Channels</td>
</tr>
</tbody>
</table>

### Security Capability and Intrinsic Behaviors

<table>
<thead>
<tr>
<th>Capability</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous Protection</td>
<td>Secure Failure and Recovery</td>
</tr>
<tr>
<td>Secure Metadada Management</td>
<td>Economic Security</td>
</tr>
<tr>
<td>Self-Analysis</td>
<td>Performance Security</td>
</tr>
<tr>
<td>Accountability and Traceability</td>
<td>Human Factored Security</td>
</tr>
<tr>
<td>Secure Defaults</td>
<td>Acceptable Security</td>
</tr>
</tbody>
</table>

### Life Cycle Security

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeatable and Documented Procedures</td>
<td>Secure System Modification</td>
</tr>
<tr>
<td>Procedural Rigor</td>
<td>Sufficient Documentation</td>
</tr>
</tbody>
</table>

Source: NIST SP 800-160
SECURE SYSTEM STATE

- Design with safe state (ISO 26262)
- Example implementation:
  - Reference monitor (IDS)
  - Re-flash from ROM

Source: NIST SP 800-160
**INTEGRATION TO PROCESSES**

### System Life Cycle Processes

*Recursive, Iterative, Concurrent, Parallel, Sequenced Execution*

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquisition</td>
<td>Life Cycle Model Management</td>
<td>Project Planning</td>
<td>Business or Mission Analysis</td>
</tr>
<tr>
<td>Supply</td>
<td>Infrastructure Management</td>
<td>Project Assessment and Control</td>
<td>Stakeholder Needs and Requirements Definition</td>
</tr>
<tr>
<td></td>
<td>Portfolio Management</td>
<td>Decision Management</td>
<td>System Requirements Definition</td>
</tr>
<tr>
<td></td>
<td>Human Resource Management</td>
<td>Risk Management</td>
<td>Architecture Definition</td>
</tr>
<tr>
<td></td>
<td>Quality Management</td>
<td>Configuration Management</td>
<td>Design Definition</td>
</tr>
<tr>
<td></td>
<td>Knowledge Management</td>
<td>Information Management</td>
<td>System Analysis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Measurement</td>
<td>Implementation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Quality Assurance</td>
<td>Integration</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Verification</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Transition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Validation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Operation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Maintenance</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Disposal</td>
</tr>
</tbody>
</table>

- ISO 15288 good framework
- Code first vs specification
- Not just engineering
- Aims to enable ‘organizational learning’ -> same breach does not happen twice
- Work split between OEM/T1 and AGL?
MORE SECURITY SOLUTIONS

- More cost-efficient solutions enable better security
  - AGL, APPSTACLE, ASSET
- Improve overall level of security
- Implement HW solutions with SW
- Developer training
SUMMARY

- Connected vehicles are happening now
- Need uncompromised solutions
  - Same as safety
- There are plenty of solutions
  - But none solves it alone
- More holistic approach is future
LINK-MOTION.COM
info@link-motion.com
mikko.hurskainen@link-motion.com
kanae.kubota@link-motion.com