Model-Driven Engineering for Safety, Security and Performance:
SysML-Sec

Ludovic APVRILLE
ludovic.apvrille@telecom-paristech.fr

Invited talk - ISAE Supaero
Telecom ParisTech
Research Work: Overview

Modeling and Verification for Safe and Secure Embedded Systems
Security Analysis of Embedded Systems

Application domains

IoT, Transports (automotive, drones), telecommunication systems (5G, Software-Defined Radio)

Contracts&projects

Local (Labex, FUI, ...), national (Nokia lab, ...), international (H2020 "AQUAS", ...)
Research Work: Ph.D. Positions

- Definition of safe and secure functional strategies for autonomous vehicles using critical scenarios
- Model-based Joint analysis of Safety and Security
Outline

Context: Security for Embedded Systems
Embedded systems

SysML-Sec
Method
SysML-Sec

Case study
Case Study

Demo
Demo

Conclusion
Conclusion, future work and references
Examples of Threats

Transport systems

- Use of exploits in Flight Management System (FMS) to control ADS-B/ACARS [Teso 2013]
- Remote control of a car through Wifi [Miller 2015] [Tecent 2017]

Medical appliances

- Infusion pump vulnerability, April 2015.
  http://www.scip.ch/en/?vuldb.75158
Examples of Threats (Cont.)

Internet of Things

- Proof of concept of attack on IZON camera [Stanislav 2013]

- Vulnerability on fitbit [Apvrille 2015]

- Hacking a professional drone [Rodday 2016]
Finding Vulnerabilities on IoTs

What’s inside? Let’s look together!
Inside a Fitbit

Don’t try this at home!
Inside a Fitbit (Cont.)

Again: don’t try this at home!
Inside a Fitbit (Cont.)
Fitbit: Hardware Components

- LIPo battery
- Vibrator
- STM LIS2DH - Triaxial MEMS
- STM 32L151C6
- NRF 8001
  1386KV
- TI Charger
  BQ24040
- LEDs
Firmware Dumping
Then, How to Identify Vulnerabilities?

**Investigations**

- Testing ports (JTAG interface, UART, . . . )
- Firmware analysis
- Memory dump
- Side-channel analysis (e.g. power consumption, electromagnetic waves)
- Fault injection
- . . .

**Secure your systems!**

Develop your system with security in mind from the very beginning

Our solution: SysML-Sec, supported by TTool
Goal: Designing Safe and Secure Embedded Systems
TTool: Key Features

- Model-Driven Engineering tool
- Free and Open-Source
  - Plug-in can be used to insert private/commercial features
- Easy to use
- Focus on safety, security and performance
- Formal verification at the push of a button
Common issues (addressed by SysML-Sec):

- Adverse effects of security over safety/real-time/performance properties
  - Commonly: only the design of security mechanisms
- Hardware/Software partitioning
  - Commonly: no support for this in tools/approaches in MDE and security approaches
Partitioning

Before mapping

- Security mechanisms can be captured but not verified

After mapping

- Verify security (confidentiality, authenticity) according to attacker capabilities
  - Whether different HW elements are or not on the same die
  - Where are stored the cryptographic materials (keys)
  - Where are performed encrypt/decrypt operations
- Impact of security mechanisms on performance and safety
  - e.g. increased latency when inserting security mechanisms
Partitioning Verification

Modeling

Automatic Verification

Safety

Performance

Security
Safety and Security Mechanisms

Data Encryption/ Authentication

Safety
Security
Performance
Safety and Security Mechanisms (Cont.)

Data Security with Hardware Security Module

Safety

Security

Performance
Safety and Security Mechanisms (Cont.)

Redundancy/Coherence Check

- AutonomousSystem
  - Perception1
  - Supervisor
    - coherenceCheck
    - data1: data2 >= threshold
    - dataValid
    - error

- <<CPU-RR>> CPUPer1
- Perception1
- <<BUS-CAN>> Bus1

- <<CPU-RR>> CPUPer2
- Perception2
- <<BUS-CAN>> Bus2

Add security

Safety

Security

Performance
Safety and Security Mechanisms

Failsafe mode

- systemCheck
  - [systemOk]
  - [else]
  - defaultMode
  - failsafeMode

Safety
- Green arrow

Security
- Red cross
- Green arrow

Performance
- Red cross
- Green arrow
Safety/Security/Performance

Requirements

Security
Automated generation

Safety

Performance

System design

Verification of design w.r.t. requirements

Security
Fails
Reconsider security req.
Add/modify security mechanisms
Modify architecture (private bus, etc.)
Modify mapping
Succeeds :-) Security leads to unsafe behaviour
Succeeds :-) Security leads to degraded perf. (e.g., increased mean latency)

Safety
Fails
Reconsider safety req.
Add/modify safety mech. (e.g. safe modes)
Modify architecture (e.g. redundancy)
Modify mapping
Succeeds :-) Safety leads to unsecure behaviour
Succeeds :-) Safety leads to degraded performance

Performance
Succeeds :-) Performance issue due to safety mechanisms
Succeeds :-)
Reconsider performance req.
Reconsider algorithms
Modify architecture (Nb of cores, etc.)
Modify mapping
Fails
Performance issue due to security mechanisms
- Precise model of security mechanisms (security protocols)
- Proof of security properties: confidentiality, authenticity
- Channels between software blocks can be defined as private or public
  - This should be defined according to the hardware support defined during the partitioning phase
Case Studies

Cyber security of connected vehicles
- Safety/Security/Performance
- EVITA FP7 Partners: Continental, BMW, Bosch, ...
- VEDECOM

H2020 AQUAS
- Automated train sub-systems (ClearSy): Safety/Security/Performance
- Industrial Drives (Siemens): Safety/Security/Performance

Nokia
- Digital architectures for 5G networks (Safety/Performance)
Case Study: VEDECOM Autonomous Vehicle

Model

Verification

Tests

Institut Mines-Telecom
Constraints

- Standard: ISO26262
  - SOTIF: Safety Of The Intended Function
- Security: impact of potential attacks on safety
### Requirements

**SecurityMain**

- **ID=0**
  - Text: The autonomous system will be secure
  - Kind: Functional
  - Risk: Low
  - Reference elements:

- **ID=8**
  - Text: The system will not broadcast previous GPS locations
  - Kind: Privacy
  - Risk: Low
  - Reference elements:

- **ID=7**
  - Text: The system will ensure Confidentiality of Keys
  - Kind: Confidentiality
  - Risk: Low
  - Reference elements:

- **ID=11**
  - Text: The system will verify sensor data
  - Kind: Integrity
  - Risk: Low
  - Reference elements:

- **ID=10**
  - Text: The system will protect against replay attacks
  - Kind: Freshness
  - Risk: Low
  - Reference elements:
Attacks

- <<root attack>> attackBraking
  - <<block>> Vehicle
    - <<OR>>
      - <<attack>> preventObstacleDetection
      - <<OR>>
        - <<attack>> preventBrakingFunction
        - <<OR>>
          - <<attack>> preventBrakingCommandIssue
          - <<OR>>
            - <<attack>> preventDataComputation
            - <<OR>>
              - <<attack>> disableSensors
              - <<OR>>
                - <<attack>> preventBrakingCommandIssue
                - <<OR>>
                  - <<attack>> jamPerceptionCommunications
                  - <<OR>>
                    - <<attack>> forgeECUCommands
                    - <<OR>>
                      - <<attack>> jamECUCommunications
                      - <<OR>>
                        - <<attack>> forgePerceptionData
                        - <<OR>>
                          - <<attack>> authenticatePerceptionData
                          - <<OR>>
                            - <<attack>> authenticateECUCommands
                            - <<OR>>
                              - <<attack>> filterCommunications
                              - <<OR>>
                                - <<attack>> manipulateCamera
                                - <<OR>>
                                  - <<attack>> manipulateLIDAR
                                  - <<AND>>
                                    - <<attack>> checkComponentStatus
                                    - <<OR>>
                                      - <<attack>> authenticatePerceptionData
                                      - <<OR>>
                                        - <<attack>> filterCommunications
                                        - <<OR>>
                                          - <<attack>> checkComponentStatus

<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<<countermeasure>> filterCommunications
<<countermeasure>> authenticatePerceptionData
<<countermeasure>> authenticateECUCommands
<...>
Safety Verification (Before Mapping)

Reachability/Liveness

Queries

Safety Pragma
A[] Supervisor.running
Perception.distance< threshold -->
Supervisor.brakingOrder
Architecture and Mapping Views
Safety Verification (After Mapping)

Reachability Graph

Minimized RG
Security Verification

Dialog window

Backtracing
Performance Verification

Latency

Bus/CPU Load

[Diagram showingLatency and Bus/CPU Load with various components and metrics such as delay, chl, LidarData(1), MABXcommand(1), processCommand, commandValid, and ECUcommand(1).]
SW Design, Code generation, Test

- First SW model from mapping models
- SW model refinement
- SW model verification (safety, security)
- Code generation
  - (Virtual) Prototyping, test
Demo: SmartCard

- Main functions of the system
- Safety of the system (before mapping, after mapping)
- Performance
- Model enhanced with Security
- Impact on performance
Achievements: SysML-Sec

- Methodology for designing safe and secure embedded systems
- Fully supported by TTool
- Applied to different domains, e.g., automotive systems, IoTs, malware

Future work

- Security risk assistance and backtracing
- Assistance to handle conflicts between security/safety/performance
  - Design space exploration
To Go Further ...

Web sites


References
