Model-based Cybersecurity Engineering for Connected and Automated Vehicles

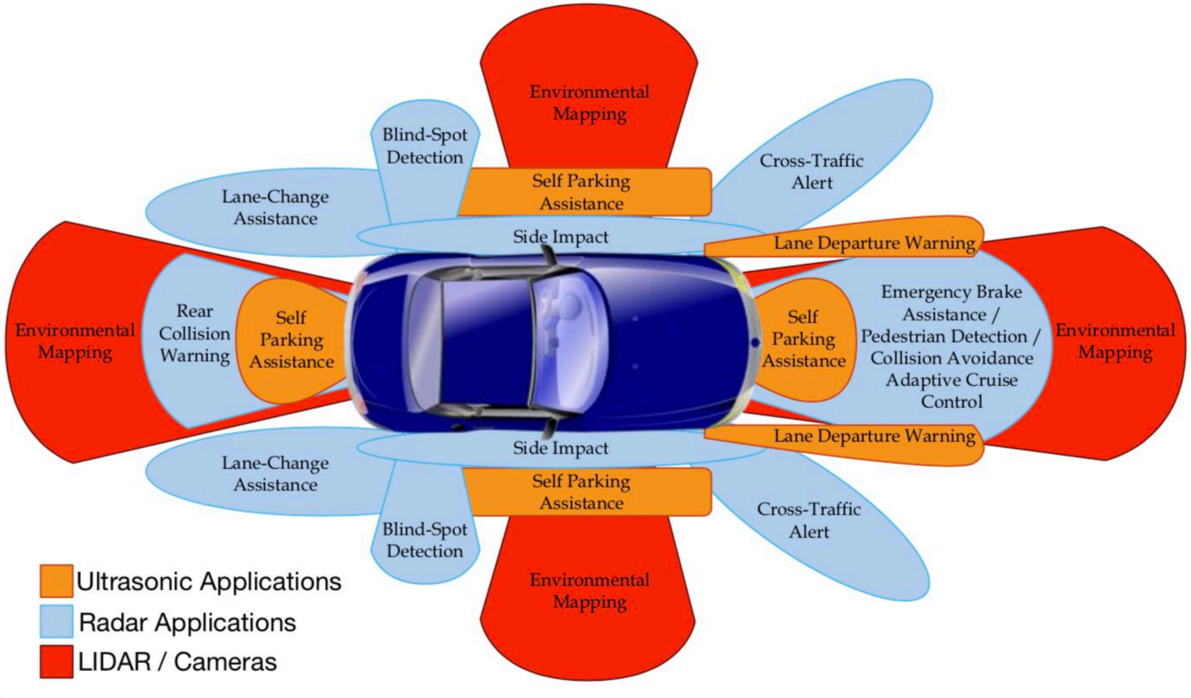
The FLOURISH Project

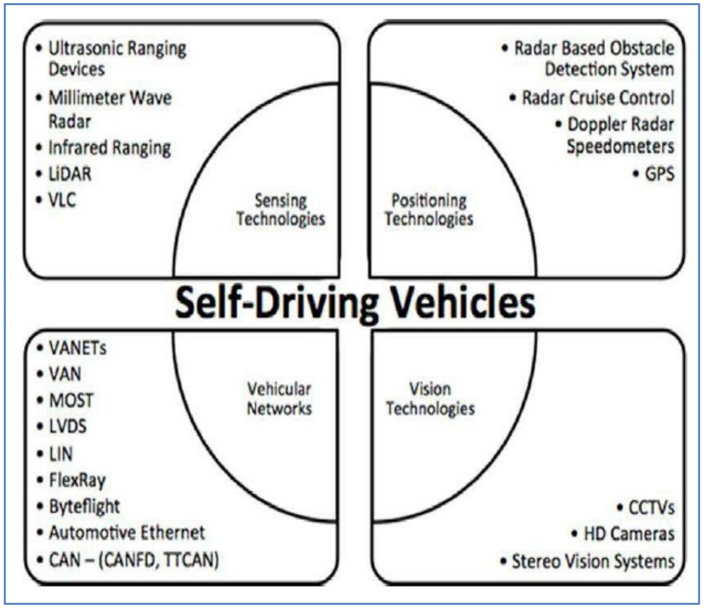
David Robles-Ramirez, **Theo Tryfonas,** Ponciano J. Escamilla-Ambrosio, Tesleem Fagade, Kalliopi Anastasopoulou, Andrea Tassi, Robert Piechocki


Systems
Engineering
information
industry
cities
smart
work
development
control
Computer
engineering
mobile
future
etc
wireless
technologies
sensor
computer
worked
security
efficiency
diverse
networks
includes
systems
assurance
also
Science
across
introduction
IoT
well
time techniques
cybersecurity
range
applications
Chartered
Information
evidence
digital
looking
Civil member
architectures
system
doctoral
postgraduate
deliver
EngD
management
consultant
improvement
research
providing
Faculty
talking
spending
Young
Things
large-scale
Catherine
University
mainly
Chrissa-Sophia
Council
planetary
professional
Institute
Group
utilities
complex
subjects
sectors
capabilities
software
graduated
analysis
anything
past
aspects
year
cases
moved
SCADA
MBCS
fields

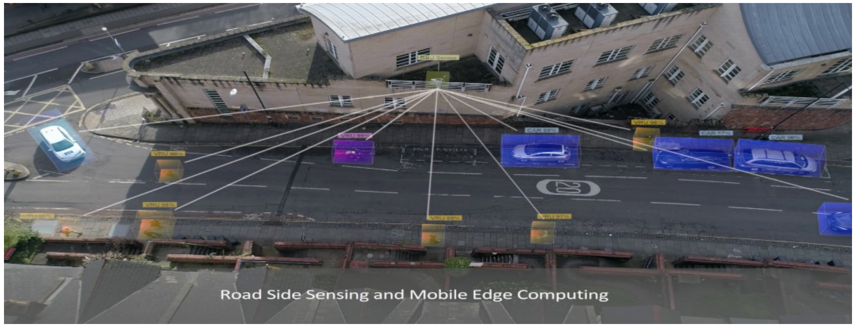

Irish Sea
Dublin
Liverpool
England
United Kingdom
Birmingham
Bristol
London
M20 Motorway
Bristol Channel
English Channel
Map
Photos
Data SIO, NOAA, U.S. Navy, NGA, GEBCO

Connected and Automated Vehicles?





- FLOURISH is a multi-sector collaboration aiming to contribute to the delivery of advanced technologies for Connected Autonomous Vehicles (CAVs) in the UK. The Flourish project addresses two different but critical issues for CAVs:
  - Ensuring that wireless connectivity and cybersecurity of CAVs are considered by design; and
  - Optimising individuals' experience when using the technology, with a particular focus on the needs of an ageing population.



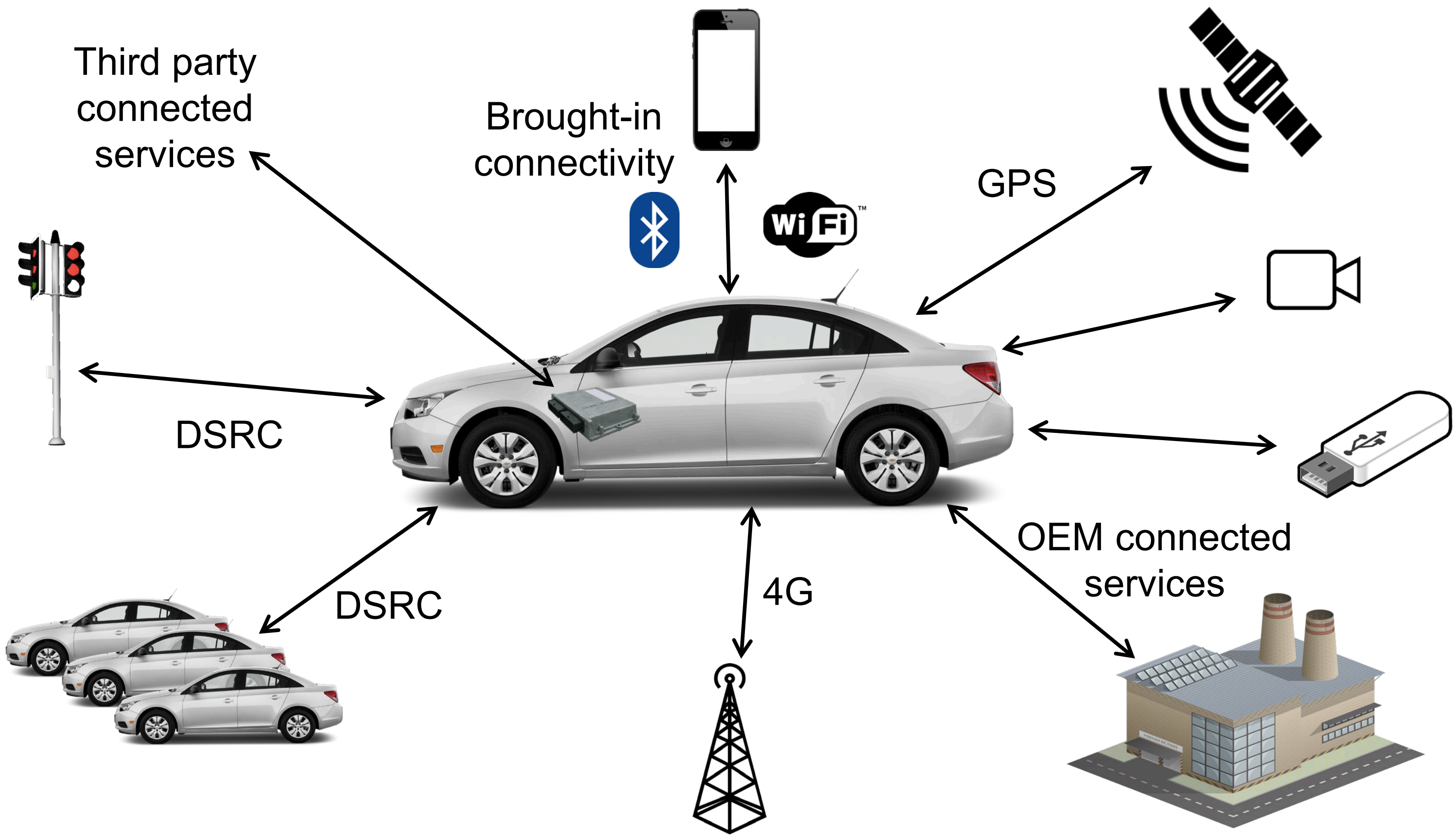
flourish

EMPOWERMENT THROUGH **TRUSTED** SECURE MOBILITY

Open Challenges

- Geographical addressing
- Risk analysis and management
- Data-centric Trust and Verification
- Anonymity, Privacy and Liability
- Secure localization
- Forwarding algorithms
- Delay and Reliability constraints
- Prioritization of data packets and congestion control

Attack Surfaces



Solutions for Security, Privacy and Trust (SPT)

- **Proactive security**: digitally signed messages with certificates (PKI), proprietary system design (protocols and HW), tamper resistant HW
- **Reactive security**: signature-based and anomaly-based intrusion detection
- **Anonymity and privacy**: linkability between *pseudonyms*
- **Secure localization**: tamper-proof GPS, etc

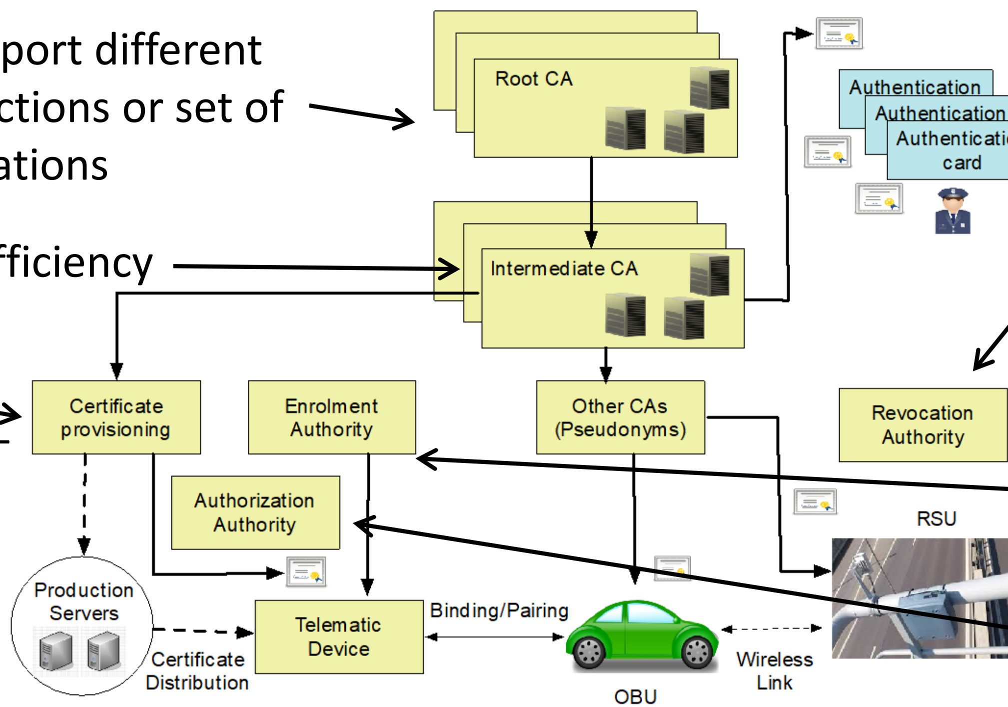
Public Key Infrastructure for C-ITS: an example

Generic C-ITS deployment model for generation and distribution of cryptographic material (EU C-ITS WG5-Annex 1)

To support different jurisdictions or set of applications

To increase efficiency

To distribute certificates to C-ITS device



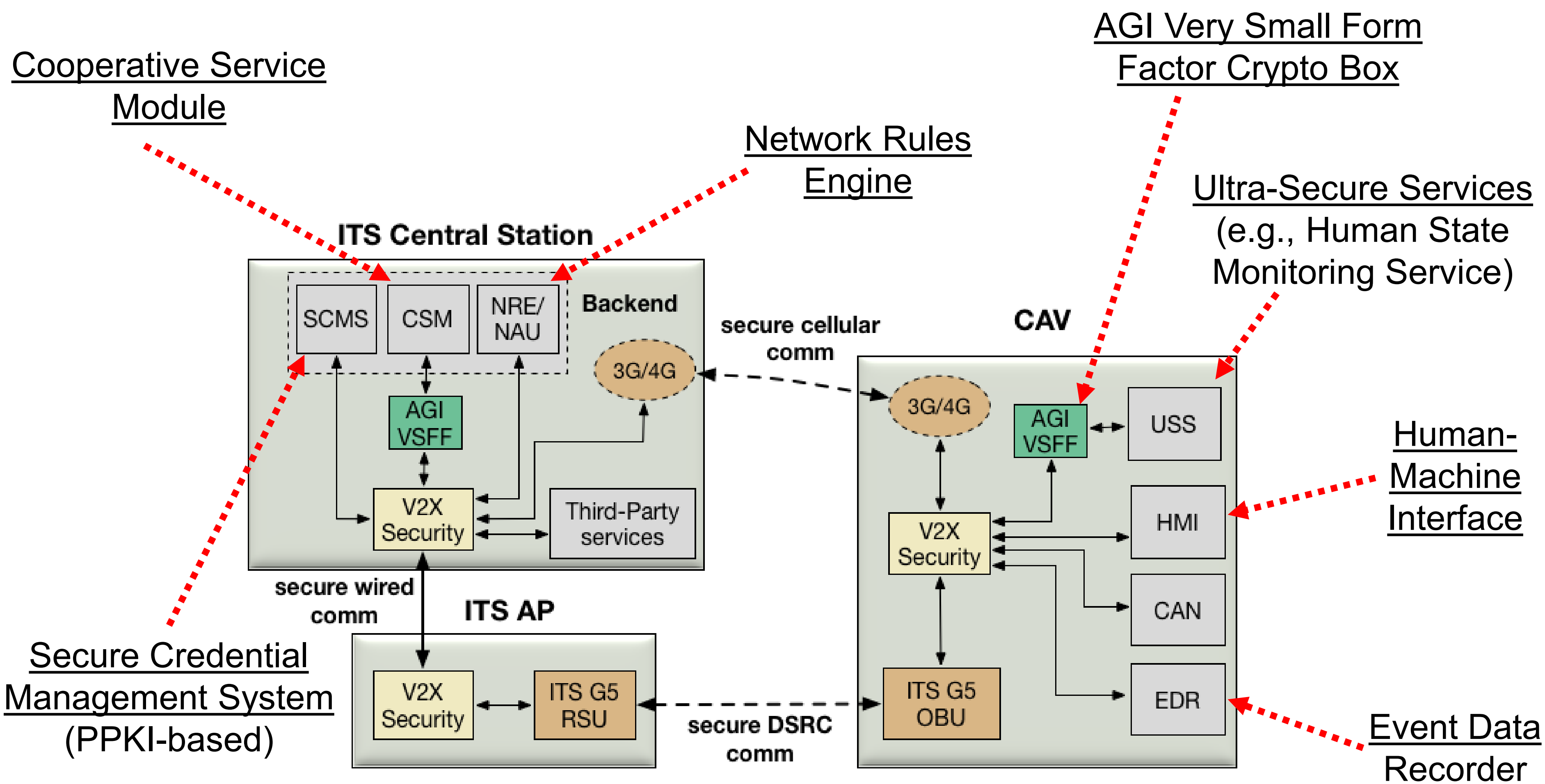
To revocate certificates associated with specific telematics devices

To ensure the registration of the telematics device

To grant the permission to operate

Building blocks

23 March 2020



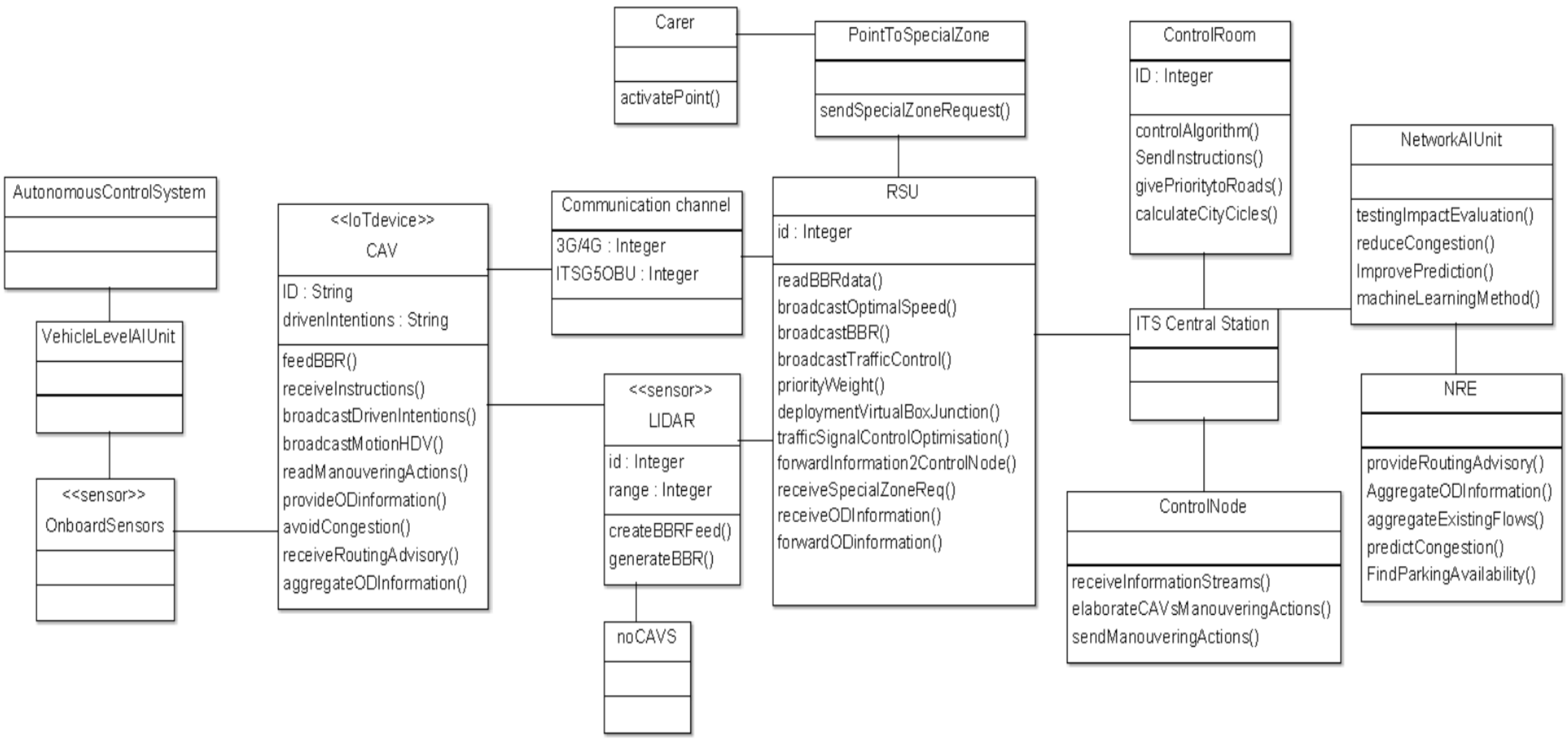
Model-Based Systems Engineering

- Define CAV development lifecycle and IV&V strategy
  - Considering alignment with ISO/IEC 15288 “Systems Engineering Life Cycle Processes” and/or IEEE 1220 “Application and Management of the Systems Engineering Process”
- Develop generic CAV system architecture
  - Using MODAF/SysML embedded systems modelling
  - In line with MoD Standard 23-09 (GVA)

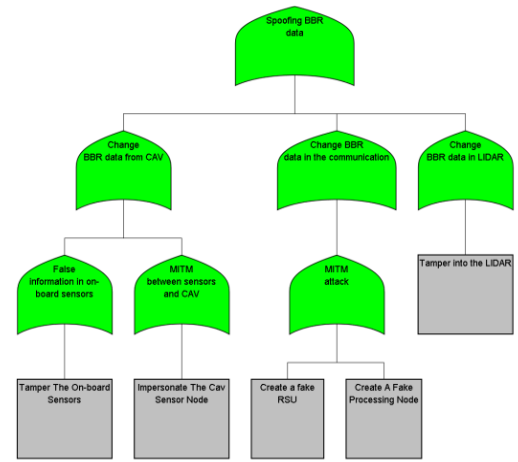
Model-Based Systems Engineering (cont'd)

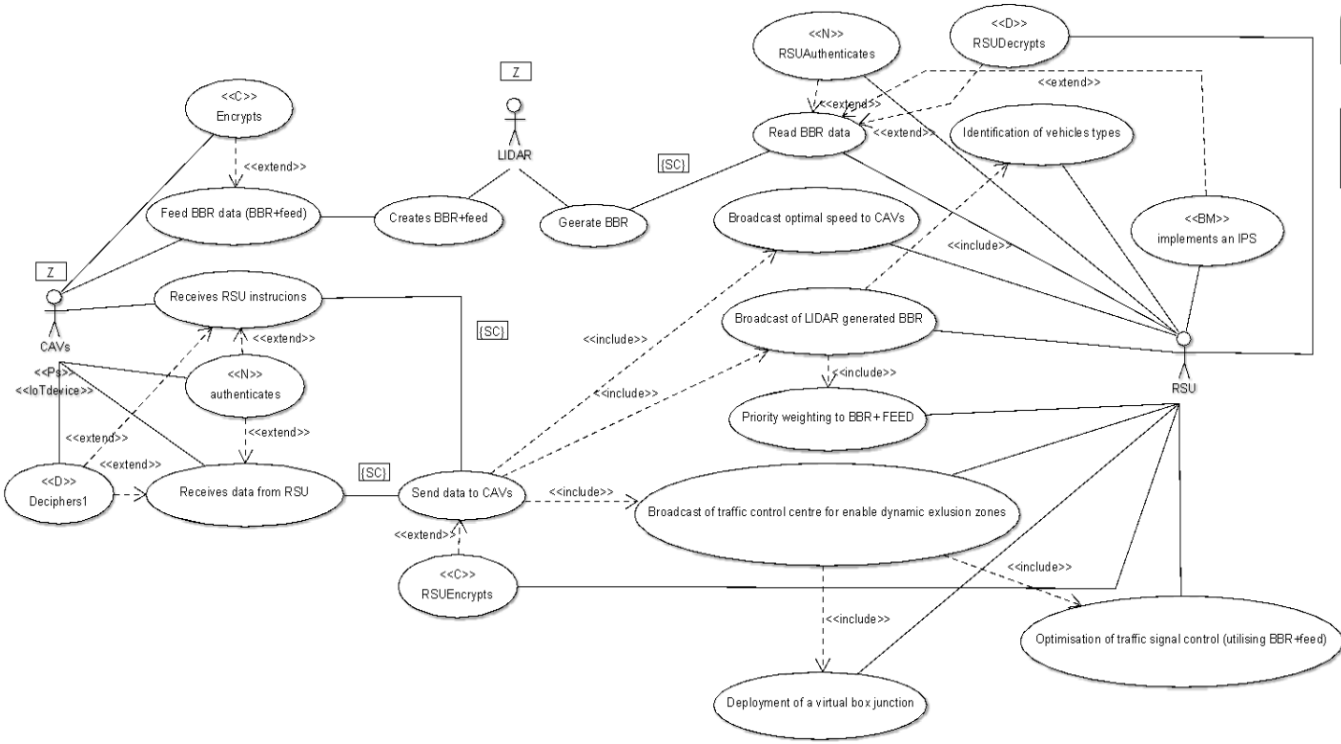
- Assess CAV GVA against C-ITS platform 2016 report key requirements
  - Consider tailoring IV&V processes towards generic certification
- Develop an SoS view of CAV operation
  - System-of-systems (or network-of-networks) model
  - Suitable simulation (e.g. system dynamics, cybernetics/viable system, game theory etc. - tbc) to identify emergent behaviours, unintended consequences and potential risks

Original Architecture



LIDAR scenario diagram

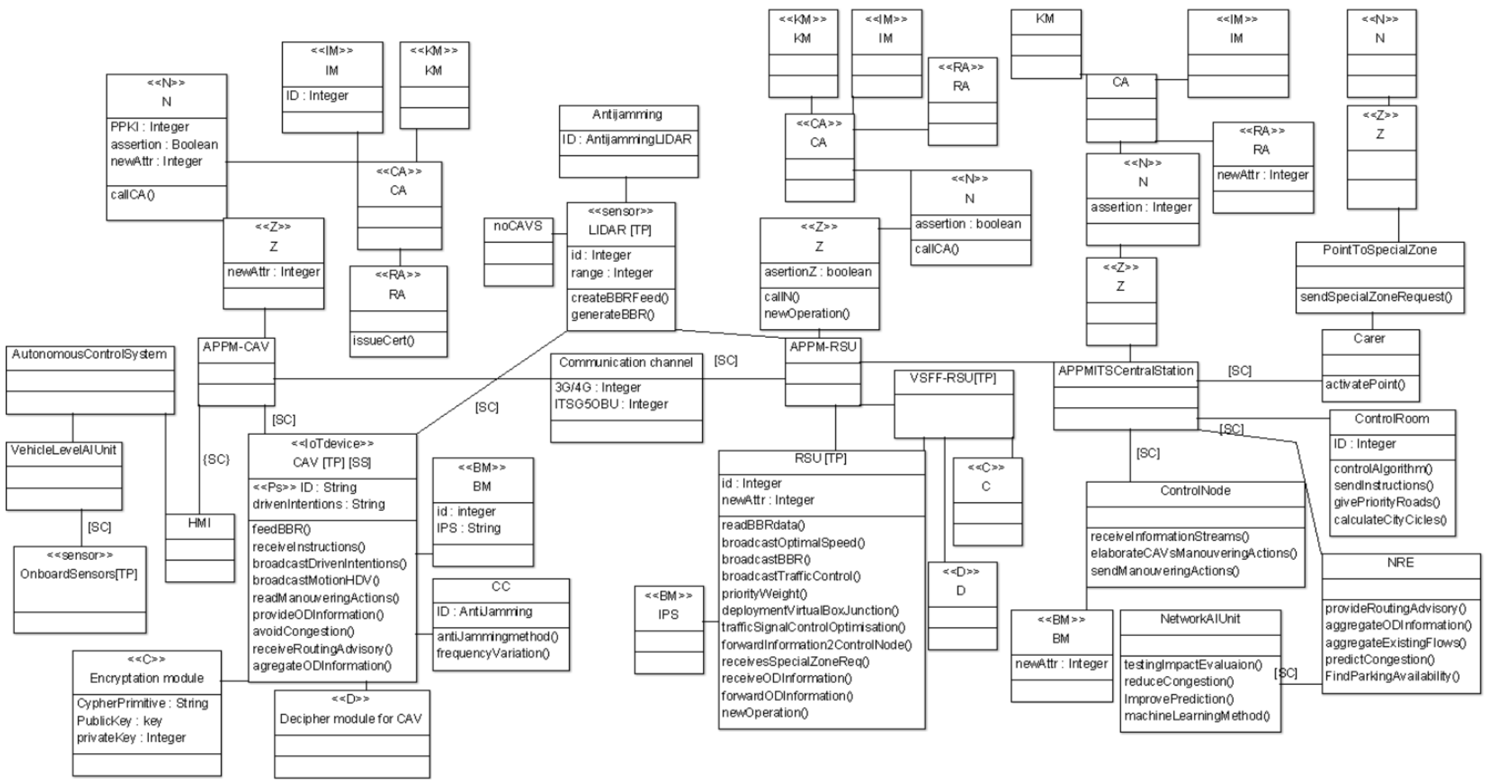




Use case name: <<N>> authenticates
Participating actor: CAV
Entry condition: An entry package is sent from RSU
Events flow: The package is received. The CAVs actor runs the authentication element. The <<N>> element obtains the RSU credentials from the package. The <<N>> stereotype instance creates complementary information from de credentials. The <<N>> stereotype instance runs the authentication function. The <<N>> creates the assertion {True, False}. This use case extends the Receives RSU instructions use case and Receives data from RSU use case
Exit condition: The CAVs authenticate the package received

Use case name: <<N>> RSU Authenticates
Participating actor: RSU
Entry condition: Receive data from the LIDAR
Events flow: The package is received. The RSU actor runs the authentication element. The <<N>> element obtains the LIDAR credentials from the package. The <<N>> stereotype instance create complementary information from de credentials The <<N>> stereotype instance runs the authentication function. The <<N>> creates the assertion {True, False}. This use case extends the Read BBR data instructions use case.
Exit condition: The RSU authenticates the package received.

Re-engineered architecture using IoTSecM



Model-based security

- Systematic documentation of security requirements
- Controls not just baselined, but selected based on need
  - which became clear to all
- Ability to simulate the logic of external security analysis techniques
  - here fault trees, could be others (or combination)
- Compatible with use cases implemented at other WPs
- Potentially standardizing the UK GVA for CAVs

flourish

EMPOWERMENT THROUGH **TRUSTED** SECURE MOBILITY

- Thank you