HoliSec

HoliSec Reference Architecture

Holistic Approach to Improve Data Security

Report type	**Deliverable D4.1.3**
Report name	**HoliSec Reference Architecture**
Dissemination level	**Public**
Status	**Release**
Version number	**1.0**
Date of preparation	**2018-04-02**

Authors

Authors	E-mail
Atul Yadav	atul.yadav@volvo.com
Christian Sandberg	christian.sandberg@volvo.com

Revision chart and history log

Version	Date	Reason
1.0	2018-04-02	Creation of the initial draft

Table of contents

Authors 1

Revision chart and history log 2

Table of contents 3

1 Introduction 5

1.1 Scope and purpose 5

1.2 Related work 7

2 Data classification 8

3 Specification of the in-vehicle architecture 10

3.1 Communication specification 11

3.1.1 Internal communication 11

3.1.2 External communication 12

3.1.3 Communications matrix 13

3.2 Functionality specification 15

3.2.1 Engine brake 15

3.2.2 Gear shift 15

3.2.3 Foundation brake 15

3.2.4 Park brake 16

3.2.5 Cruise control 16

3.2.6 Engine Speed Control 16

3.2.7 Fleet tracking 16

3.2.8 Vehicle information logging 17

3.2.9 Vehicle Speed 17

3.2.10 Light control 18

3.2.11 Immobilizer/Alarm 18

3.2.12 Extended vehicle warning 18

4 ECU Specifications 19

4.1 Edge Node GW 19

4.2 Driver Control 19

4.3 Connectivity GW 20

4.4 Engine ECU 20

4.5 Brake ECU 20

4.6 Driver Display 21

4.7 V2X 21

4.8 Vehicle ECU 21

5 Use cases 22

5.1 Use case 1: GPS position and warning light status sent via V2X 22

5.2 Use case 2: Set target Cruise Control Speed 22

5.3 Use case 3: Fleet management system collects location, speed and status of the vehicle 22

5.4 Use case 4: The immobilizer detects a counterfeit key and blocks start of the engine 23

5.5 Use case 5: Readout of emissions data using Client tester tool 23

6 References 24

1 Introduction

This document is part of to the deliverable “**D4 Secure development techniques for automotive software and systems**” of the HoliSec project. The deliverable describes a vehicle reference architecture that may be used, albeit not limited to, developing/improving methods to analyse threats and assess risks.

1.1 Scope and purpose

Today, in-vehicle architectures consist of a number of electronic control units (ECU) interconnected by different bus communication systems, typically CAN.

Traditional communications buses, such as CAN have shown to be a bottleneck in bandwidth intense applications, thus a trend in the vehicle industry is to introduce different high bandwidth communication protocols for internal vehicle communication, like those based on IP/Ethernet.

There is an increasing need to interact externally due to requirements from advanced telematics services, support for entertainment though nomadic devices, future applications based of vehicle to vehicle and vehicle to infrastructure communication, support for customer tailored add-ons etc.

In HoliSec project activities, involving OEM, suppliers, research institutes and academia, it is important to have a common view and understanding of in-vehicle architecture and its interaction with the external world. This report elaborates around the definition of a reference architecture that can be used as a basis in the HoliSec project when evaluating different security concepts, such as threat analysis, risk assessment, secure diagnostic mechanisms and key management.

HoliSec project goals (Relevant WP)	Deliverable D4.1.3	
	Relevance	Contributions
● Identify needs and requirements of security in the automotive industry. (WP1)	None	
● Study and identify state-of-the-art in security in the automotive industry. (WP1)	None	
● Investigate and describe cryptographic solutions ● Investigate and define a key management lifecycle (WP2)	Low	The reference architecture may be used when investigating cryptographic algorithms as well as approaches to key management in a vehicle.
● Define methods to establish trust between diagnostics tools and ECU ● Investigate ways to secure vehicle communication ● Investigate the interplay between safety and security ● Define solutions for detecting and log intrusions (WP3)	High	The reference architecture may be used when determining the suitability of a mechanism in a vehicle. Furthermore, an instantiation of the reference architecture may be used to verify the mechanism in a lab environment.
● Propose methods for secure software and system design ● Propose methods for secure software verification & validation ● Investigate solutions for security intelligence (WP4)	High	The reference architecture will be used when developing new methods for threat analysis and risk assessment, allowing for a small, non-sensitive from intellectual property point of view, to be used as reference in public papers.
● Demonstrate proof of concepts. (WP5)	Medium	D4.1.3 provides a description of a vehicular reference architecture that may be used to show security mechanisms in a demo setup.

1.2 Related work

The EVITA project was a project running between 2008 and 2011 and was co-funded by the European Union. The objective of the project was to design, verify and prototype an architecture for automotive on-board networks where security-relevant components that are protected against tampering and sensitive data are protected against compromise. As part of the EVITA project, the "EVITA use case reference architecture" was produced. See Figure 1.

The EVITA architecture was found not suitable for this analysis due to the following reasons:

- A trend in the automotive industry is to move towards fewer more powerful nodes in star network topology, or multi-star, which is not reflected in the architecture
- The documentation lacks contextual information about the functionality, such as criticality and information exchange

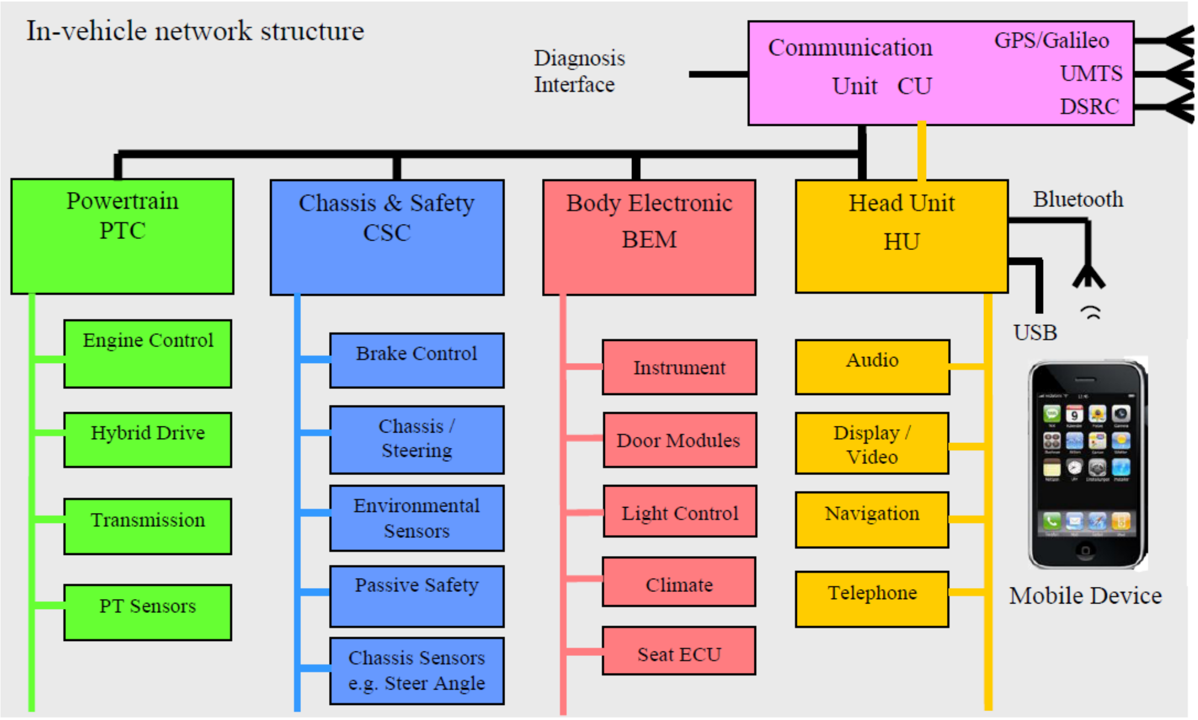

Figure 1 EVITA Use Cases Reference Architecture [1]

2 Data classification

Data at rest, in transit, and in use, may be classified in different ways. Data, such as a signal used to request a certain vehicle speed, may be safety critical in the way that a tampered signal may result in unintended acceleration, and at the same time privacy critical since it shows the speed the driver has requested.

While there are many ways to classify data, for this reference architecture four different classifications attributes are considered: **Safety**, **Privacy**, **Intellectual Property** and **Legal**.

Safety

Safety critical data is data that may affect EE (Electronic & Electrical) systems in the way that it can lead to injury, death, or loss of vital (and expensive) equipment.

This type of safety critical data must often be protected and treated with strict care, obliged to fulfil security requirements that it shall not be tamperable by a malicious user, nor spoofable.

A major direction for enhancing the EE system safety is to ensure that the system functions are safe to use in all situations, even when the system on a whole or some sub-system does not function properly, which is a very important dependability aspect.

Dependability is a term that summarizes several attributes, such as availability, reliability and maintainability. Safety (absence of catastrophic consequences on the user(s) and the environment) is also a dependability attribute.

There may be safety critical systems that put no requirements on confidentiality of data.

Privacy

Privacy critical data may store information about one or more persons, typically the driver, such as driving behavior, speeds at which the vehicle has been driven, where it has been driven and when. Privacy critical data shall not be disclosed to a party to which consent of sharing has not been given.

Intellectual Property

The vehicle contains intellectual data in the form of software and data stored in the ECUs. Data is constantly gathered, processed and this knowledge may add to the OEM intellectual property pool of data. Intellectual property may be used by other companies to gain competitive advantage or by hackers to understand and misuse a system. Data classified as intellectual property may have similar security requirement with data that is classified as Privacy critical data. An example is data that shall not be disclosed to an unauthorized third party.

Legal

Legislation may put different kinds of security requirements on data, for instance the non-disclosure of personal information, or the availability of data when requested, or that the data is authentic and not tampered.

Table 2-1 presents a mapping of the four groups of data classification to both STRIDE and CIA, in order to visualize what security requirements each classification puts on a system.

Table 2-1 Data classification and STRIDE/CIA mapping

Name	S	T	R	I	D	E		C	I	A
Safety	X	X			X	X			X	X
Privacy				X				X		
Intellectual Property				X				X		
Legal	X	X	X	X	X	X		X	X	X

3 Specification of the in-vehicle architecture

This section present the topology (Figure 2), functionality and communication specification of the HoliSec Reference Architecture.

The architecture has been designed with a number of design goals:

- Encompass several types of communication technologies, such as CAN, Ethernet, Wireless and cellular
- Include both vehicle internal and external interfaces
- Include several types of electronic devices, such as ECUs and sensors
- Communication exchange of different types of critical data between ECUs. This means the design of functions has been deliberately distributed, sometimes in non-optimum ways, to expose data of different criticality on the buses, such as safety critical and privacy critical. The main purpose of this is to make the reference architecture more useful when developing threat analysis and risk assessment methodologies.

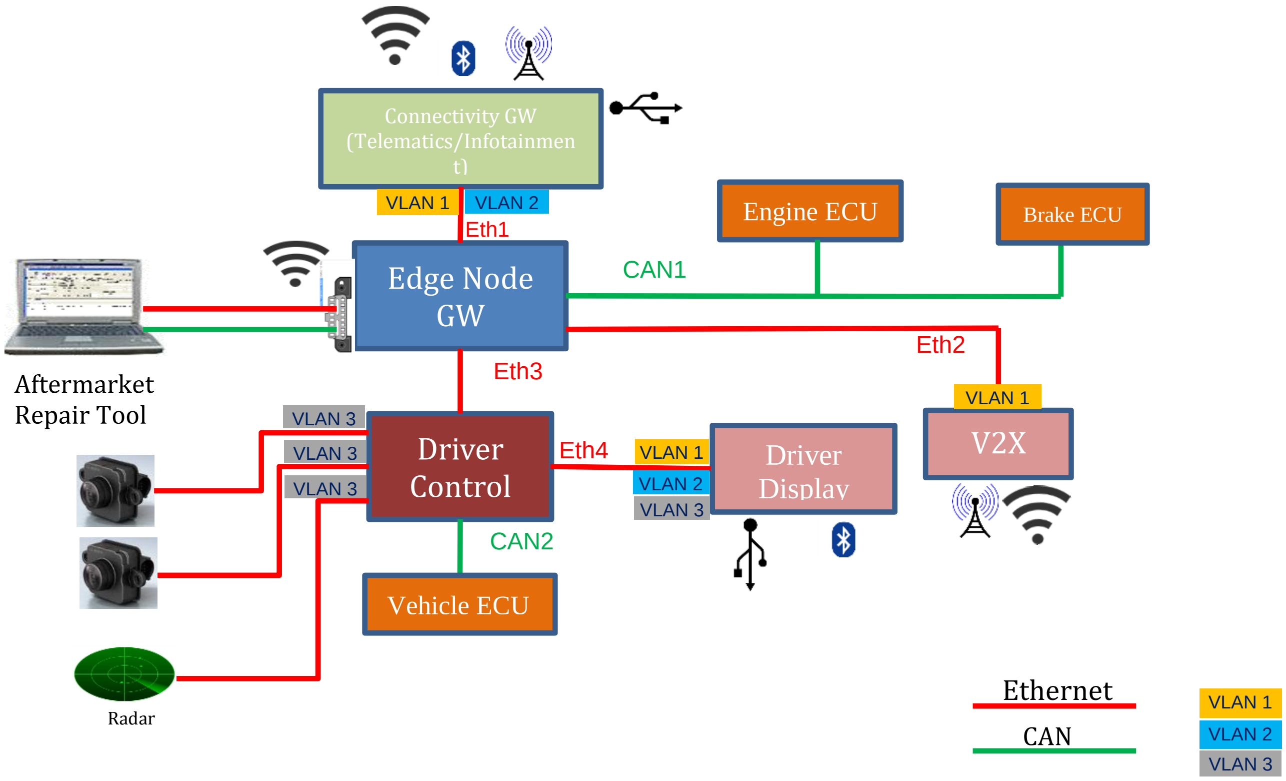

Figure 2 Topology

3.1 Communication specification

Vehicles rely on data exchange both internally between ECU's, sensors and actuators and to an increasing degree to external parties such as back office, cloud servers, other vehicles, nomadic devices and infrastructure.

For this section the internal communication is considered to be communication between entities that have been installed into the vehicle by the OEM either during manufacturing or in aftermarket. External communication is communication to third party devices that are connected either physically or wirelessly to the vehicle.

3.1.1 Internal communication

Contemporary vehicles use different communication technologies to exchange data between ECUs, sensors, and actuators, where the protocols CAN, FlexRay, LIN, MOST and Ethernet are some of the more commonly used physical layers. A trend is to go from low bandwidth communication technologies, such as CAN, towards technologies with higher bandwidth, such as Ethernet, with FlexRay and CAN FD as other alternatives offering higher bandwidth. Depending on the criticality of data transported on the buses there may be requirements that guarantee that data is sent on the bus within a certain time. This is solved in different ways for different buses. For CAN, the id sent with each frame also indicates its priority on the bus. A high priority message may override a low priority message. This feature could lead to denial of service if the bus is drowned with high priority messages. FlexRay guarantees data to be sent on the bus by offering dedicated time slots for transmission.

In general, most communication on the internal vehicle buses are broadcast messages. All nodes on the same subnet as the sender (original sender or on a subnet where a router has rebroadcasted the message) may read the message. In the case of CAN, 8 bytes of data are transmitted in a frame and either 11 or 29 bits of identifier data may be used to identify the frame. Typical speeds for CAN range from 250kbit/s to 1Mbit/s.

Most signals exchanged between ECUs are transmitted periodically, ranging from 100 times a second to a message every few seconds in between.

Ethernet provides far greater bandwidth than e.g. CAN, reaching 100 Mbit/s, moving towards 1 Gbit/s. With its greater bandwidth, Ethernet is suitable for camera feeds (e.g. as sensor input for autonomous drive) and media streams. Virtual LAN, VLAN, may be used to partition the physical Ethernet layer into virtual networks, where e.g. prioritization of messages may take place, in order to guarantee transmission of critical data.

Virtual LAN	**Description**
VLAN1	High priority communication between applications in Connectivity ECU, V2X and Driver Display
VLAN2	Low priority communication between applications in Connectivity ECU and Driver Display
VLAN3	Medium priority communication dedicated to cameras and radars

3.1.2 External communication

The need for external information exchange has increased over the years, starting with connections to back office for reading diagnostic trouble codes, reporting accidents, play music over media from USB thumb drives and connecting Bluetooth devices to the audio equipment. As more services have been provided by the OEM, more data has been exchanged between the vehicle and back office (3G, 4G networks), such as navigation data, fleet management data, diagnostics data, media streaming via nomadic devices and remote control of vehicular functions, such as remote lock via smartphone (Bluetooth, 3G, 4G).

More and more vehicles will begin to communicate directly with other vehicles, known as V2V/V2X (vehicle-to-vehicle, vehicle-to-infrastructure). Such communication is likely to take place over 802.11p or 5G networks. Bandwidth over cellular networks may vary greatly, or even be interrupted, depending on the cell coverage and signal strength.

3.1.3 Communications matrix

Table 3-1 summarizes all the signals being sent inside the system as depicted in Figure 2. The signals are tagged as broadcast if a node sends the message with no specific receiver node. Furthermore, different classifications are indicated by the labels, **S** (Safety), **P** (Privacy), **IP** (Intellectual property) and **L** (Legal) with reasoning why a signal has a specific label covered in chapter 3.2. There is always a sender of data, although not necessarily a specified receiver. Looking at a functional view, there are receivers in the form of functions using received data, although in the case of broadcast messages, anyone listening on the bus may hear the signal and act based on its value. For instance, the main purpose of the Vehicle Speed function may be to calculate the speed based on data from a speed sensor and then distribute the current speed to the rest of the system. This information could then be used by other functions or e.g. as entry condition to perform diagnostics operations (to ensure the vehicle is at standstill), thus all ECUs are potential users of the VehicleSpeed signal.

Table 3-1 Communications matrix

Name	Associated functions	Source	Receiver(s)	Broadcasted on networks	Security mechanism
BrakeCommand [*broadcast*] [*criticality: S*]	Engine brake, Foundation brake	Engine ECU		CAN1	
BrakeEngagedPercentage [*broadcast*]	Foundation Brake	Brake ECU		CAN1, Eth3, CAN2	
RequestedGear [*broadcast*]	Gear Shift	Vehicle ECU		CAN1, Eth3, CAN2	
CurrentGear [*broadcast*]	Gear Shift	Engine ECU		CAN1, Eth3, CAN2	
ParkBrakeStatus [*broadcast*]	Park Brake	Brake ECU		CAN1, Eth3, CAN2, Eth4	
VehicleSpeed [*broadcast*]	Vehicle Speed, Fleet tracking	Vehicle ECU		CAN1, Eth3, Eth1, CAN2	
VehicleSpeedCommand [*broadcast*] [*criticality: S*]	Vehicle Speed	Vehicle ECU		CAN1, Eth3, CAN2,	
DiagnosticsRequest *[**request**/response]*	Fleet tracking, Vehicle information logging	Connectivity ECU (internal tester)		CAN1, CAN2, Eth1, Eth2, Eth3, Eth4	ISO14229: Security Access (Challenge response)

DiagnosticsResponse *[request/**response**]* [*criticality: IP,P*]	Fleet tracking, Vehicle information logging	All ECU besides Connectivity ECU	Connectivity ECU	CAN1, CAN2, Eth1, Eth2, Eth3, Eth4	ISO14229: Security Access (Challenge response)
EmissionsData [*broadcast*]	Vehicle information logging	Engine ECU		CAN1	
EmissionsDataRequest *[**request**/response]*	Vehicle information logging	External interactor	Edge Node ECU	OBD	
EmissionsDataResponse *[request/**response**]* [*criticality: L*]	Vehicle information logging	Edge Node ECU	External interactor	OBD	
DiagnosticsRequest *[**request**/response]*	Vehicle information logging	External interactor	Edge Node ECU	OBD	ISO14229: Security Access (Challenge response)
DiagnosticsResponse *[request/**response**]* [*criticality: IP,P*]	Vehicle information logging	Edge Node ECU	External interactor	OBD	ISO14229: Security Access (Challenge response)
ImmobilizerStatus [*broadcast*]	Immobilizer/Alarm	Connectivity ECU		CAN2, Eth1, Eth 3	
ImmobilizerStatus [*broadcast*]	Immobilizer/Alarm	Connectivity ECU	Engine ECU	CAN1	Authenticated
V2X_Warning	Extended vehicle warning	V2X ECU	Driver Display	Eth 2, Eth 3, Eth 4	
WarningLight [*broadcast*]	Light Control, Extended vehicle warning	Vehicle ECU		CAN2, Eth 2, Eth 3, Eth 4	
GPS_Position *[criticality: P]*	Extended vehicle warning	Connectivity ECU	V2X ECU	VLAN 1 (Eth1, Eth3)	
ECUFaultStatus	Vehicle information logging	All ECUs	Driver Display	All networks on the route from a sender ECU to the Driver Display ECU	
DiagnosticsRequest: ReadDID: VehicleFaultStatus *[**request**/response]*	Vehicle information logging	Connectivity ECU	Driver Display	Eth1, Eth3, Eth4	
DiagnosticsResponse: ReadDID: VehicleFaultStatus *[request/**response**]*	Vehicle information logging	Driver Display	Connectivity ECU	Eth4, Eth3, Eth1	

3.2 Functionality specification

This section contains brief descriptions of functionality in a vehicle, representing much-simplified versions of functions compared with the complete functionality used in vehicles. The purpose of these descriptions is to give context and aiding analysis of the architecture. Functions have been given different criticality classification, as well as, in some case, indication whether they are a primary or secondary feature in the vehicle.

A primary or a secondary feature may vary with different types of vehicles, for passenger cars and buses; the primary feature may be to transport people safely from one point to another, while the comfort of the drivers and passengers may be secondary feature as long as the vehicle can be moved safely from point A to point B.

For commercial vehicles, such as trucks, a primary feature may be the timely delivery of goods, as the OEM customers are transport companies.

3.2.1 Engine brake

The engine brake is a kind of retarder used to complement the functionality of the friction based braking system (foundation brakes). Retarders are important braking mechanisms in that they significantly reduce the wear of the friction based brakes, making them last longer, provide additional deceleration to reduce stopping distance and reduce the chance of the foundation brakes overheat e.g. in the case of a long decline. The vehicle automatically controls the different brake systems at its disposal, depending on the situation and the functionality is blended as one brake request signal, either by the brake pedal connected to the Brake ECU or as a request from the Cruise Control.

- Criticality
    - Safety
        - Critical to be able to stop at shortest possible distance.
        - Without the engine brakes the foundation brakes may overheat, fade, and malfunction in long declines.
- Primary feature of the vehicle

3.2.2 Gear shift

The vehicle automatically shifts gear to maintain a proper cruise speed/torque ratio. In this reference architecture, the Engine ECU is responsible for the entire power train, including the gearbox. The Gear lever positions are sent from the Vehicle ECU to the Engine ECU (Reverse, Drive, Neutral, Park). Gear shift strategies are apart from the position of the lever, made internally in the Engine ECU. The Engine ECU broadcasts the status of the currently active gear.

- Primary feature of the vehicle

3.2.3 Foundation brake

Foundation brakes are friction based and one of the main responsibilities of the Brake ECU. The brake pedal is directly connected to the Brake ECU. The Brake ECU applies the foundation brakes based on the brake pedal position or brake request signals received on the communications bus (such as from the Cruise Control function). The Brake ECU broadcasts the brake power as a percentage from 0% applied to 100% applied) in a CAN signal.

- Criticality
    - Safety
        - Critical to be able to stop at shortest possible distance.
- Primary feature of the vehicle

3.2.4 Park brake

Park brake is another of the main responsibilities of the Brake ECU. It is used to force the vehicle to a stationary state when parked. The Brake ECU broadcasts the state of the park brake as a binary signal on the communications bus (not applied, applied). ParkBrakeStatus, along with VehicleSpeed are signals frequently used as entry conditions for functions that check if the vehicle is at standstill before proceeding e.g. with allowing a diagnostics connection.

- Criticality
    - Safety
        - Malfunctioning park brake may engage the vehicle brakes while driving
- Primary feature of the vehicle

3.2.5 Cruise control

Cruise control makes it possible for the driver to avoid keeping constant force on the throttle, by either keeping a constant speed, or adapt speed to other vehicles on the road (keeping constant time gap but variable speed. Cruise control is managed by the Vehicle ECU, who monitors the Cruise Control buttons, as well as keeping track of the vehicle speed. The Vehicle ECU requests a vehicle speed from the Engine ECU, who is responsible to meet that request, by either accelerating, maintaining current speed, or decelerate (using Engine brake or request foundation brakes to be applied by the Brake ECU).

- Criticality
    - Safety
        - The function is able to request both acceleration and deceleration
- Secondary feature of the vehicle
    - Seen as a convenience function. The vehicle is operational without it with some more effort from the driver.

3.2.6 Engine Speed Control

The gas pedal is directly connected to the Engine ECU, providing the main source of input to the engine that the engine revs should be increased.

- Criticality
    - Safety
        - Unintended acceleration is a major safety hazard
- Primary feature of the vehicle

3.2.7 Fleet tracking

Vehicles owned by big transport companies, e.g. taxis, goods transport, are frequently connected in a fleet management system. The fleet management system tracks several status parameters from the vehicle and reports to the back office. Data collected are:

- Position using GPS
- Fault codes
- Vehicle Speed
- VIN number (Vehicle Identification Number)
- Driving behavior (Acceleration, Deceleration)

The Connectivity node is responsible for collecting the requested data from the other ECUs. It uses diagnostics logins to all ECUs to check the fault code status. GPS, VIN is stored internally in the Connectivity node. Driving behavior is collected by monitoring signals sent by the Engine ECU and Vehicle Speed is collected by monitoring the Vehicle Speed signal sent by the Vehicle ECU.

- Criticality
    - Privacy
        - Contains PII data
    - Legal
        - Contains PII data
- Primary feature of the vehicle
    - For heavy vehicles, the customers expect efficient transportation and tracking of vehicles, making this a primary feature to be able to plan routes and get the transported goods to the destination in time.

3.2.8 Vehicle information logging

Collects information over time and either provides it for immediate readout or stores it for the future (such as workshop analysis).

Information stored:

- Emissions data collected from the Engine ECU (used by law enforcement for readout via the OBD connector on the Edge Node)
- GPS positions over time/routes (from Connectivity Node)
- Fault code (Driver Display collects and store fault code information from all ECUs). The Connectivity GW uses diagnostics readout of a data identifier to read the data. It sends ReadDataByIdentifier request and gets a response from the Driver Display.

- Criticality
    - Privacy
        - GPS and time data which may be translated to average speed between two locations

3.2.9 Vehicle Speed

The Vehicle ECU is responsible for determining and providing information about vehicle speed to the rest of the system. Correct notion of speed is essential for a number of features in the vehicle, such as cruise control, or the brake system to determine braking strategies. The vehicle speed is calculated using a speed sensor directly connected to the Vehicle ECU and the Vehicle Speed signal is broadcasted by the Vehicle ECU on the immediate CAN bus.

- Privacy critical
    - Logging of vehicle speed can be used to determine if someone has been driving too fast.
- Primary feature of the vehicle

3.2.10 Light control

The Vehicle ECU is responsible for the light control. Light control involves headlights, interior light, dashboard as well as indicator and brake lights.

The Light control may be controlled, apart from using buttons and levers in the cab, by the Immobilizer/Alarm functionality in the Connectivity ECU to flash the lights when it detects an intrusion.

- Primary feature of the vehicle

3.2.11 Immobilizer/Alarm

The Immobilizer/Alarm functionality, located in the Connectivity GW, is a theft prevention system that prevents the engine from starting unless an authenticated key is used that is authorized to start the vehicle. The Immobilizer broadcasts a signal indicating the status of the Immobilizer system. (Authenticated key, Unauthenticated key, Alarm, Internal integrity error). The Engine ECU uses the Immobilizer status information to determine whether to inject fuel into the engine or not and the Light Control functionality uses the information to determine whether to start flashing the lights in case of an alarm.

Information between Immobilizer ECU and Engine ECU is authenticated to prohibit theft by manipulation of signal data.

- Secondary feature of the vehicle

3.2.12 Extended vehicle warning

Vehicle to X (V2X) communication, where X stands for vehicle, infrastructure etc., adds possibility to exchange information between vehicles, thus allowing for early warning notifications using messages passed by other vehicle, such as slippery road conditions, accidents or if a vehicle is using the warning lights.

The V2X ECU passes on warnings to the Driver Display ECU

- Privacy critical
    - GPS position
- Secondary feature of the vehicle
    - The vehicle may operate without this feature

The Vehicle ECU, responsible for the lights of the vehicle as well as interior buttons, broadcasts a status signal indicating that warning lights are active and this is passed on to other vehicles using the V2X ECU together with the GPS position that is sent from the Connectivity ECU to the V2X ECU.

4 ECU Specifications

This section describes ECU-internal architectures and behaviour, such as whether signal translation is being performed, forwarding, isolation of data within the ECU. Function behaviour has only been provided to the extent necessary to provide contextual description to the use cases in chapter 5.

4.1 Edge Node GW

The Edge Node Gateway component forwards and translates signals between the networks External, Eth1, CAN1 and Eth3. The component acts upon a predefined whitelist on signal level and does not forward signals not explicitly listed in the Communications Matrix (Figure 2) as to be forwarded to a certain network.

Software architecture:

AUTOSAR Classic Platform 4.3.1 [3]

Communication interfaces:

- External
  - CAN
  - Ethernet
  - Wi-Fi
- Internal
  - Ethernet
  - CAN

4.2 Driver Control

The Driver Control ECU contains a gateway component that forwards and translates signals between the networks Eth3, CAN2 and Eth4. The component acts upon a predefined whitelist on signal level and does not forward signals not explicitly listed in the Communications Matrix (Figure 2) as to be forwarded to a certain network.

Software architecture:

AUTOSAR Classic Platform 4.3.1 [3]

Hardware architecture:

Communication interfaces:

- External
  - Bluetooth
  - USB
- Internal
  - Ethernet
  - CAN

4.3 Connectivity GW

Software architecture:

Hypervisor (EAL5+ certified)

- Virtual machines:
  - Linux
    - Only has access to all external hardware interfaces
  - AUTOSAR Classic Platform 4.3.1 [3]
    - Only has access to internal hardware interfaces

Data is passed between the virtual machines using a virtual TCP/IP communications bus controlled by the hypervisor.

Hardware architecture:

Communication interfaces:

- External
  - Wi-Fi
  - Bluetooth
  - 4G cellular
  - USB
- Internal
  - Ethernet

4.4 Engine ECU

Software architecture:

AUTOSAR Classic Platform 4.3.1 [3]

Hardware architecture:

Communication interfaces:

- Internal
  - CAN

4.5 Brake ECU

Software architecture:

AUTOSAR Classic Platform 4.3.1 [3]

Hardware architecture:

Communication interfaces:

- Internal
  - CAN

4.6 Driver Display

Software architecture:

Linux

Hardware architecture:

Communication interfaces:

- Internal
  - Ethernet

4.7 V2X

Software architecture:

Linux

Hardware architecture:

Communication interfaces:

- External
  - Wi-Fi
  - 4G cellular
- Internal
  - Ethernet

4.8 Vehicle ECU

Software architecture:

AUTOSAR Classic Platform 4.3.1 [3]

Hardware architecture:

Communication interfaces:

- Internal
  - CAN

5 Use cases

This section contains a series of brief use cases that may be used as input when e.g. performing threat analysis and risk assessment on the reference architecture.

5.1 Use case 1: GPS position and warning light status sent via V2X

The vehicle gathers the warning light status from the Vehicle ECU as well as the GPS position from the Connectivity ECU and broadcasts the information via the V2X ECU to nearby road vehicles in case the warning light is active.

Signals involved:

WarningLight: is broadcasted from the Vehicle ECU, it is forwarded via “Driver Control” ECU to Eth 3. Edge Node GW then forwards the signal to Eth 2, where it is received by the V2x ECU

GPS_Position: is sent over VLAN 1 from the Connectivity ECU to the V2x ECU over the physical connection Eth1 and then Eth2 after being forwarded between the two physical networks by the Edge node.

5.2 Use case 2: Set target Cruise Control Speed

The cruise control function located in the Vehicle ECU is activated and the driver wants to set a target speed. Two signals are used by the Engine ECU in order to know the target speed as well as the current speed.

VehicleSpeedCommand: is broadcasted from the Vehicle ECU onto CAN2, then forwarded to Eth3 via the “Driver Control” ECU and forwarded once again onto CAN1 via the Edge Node GW. This signal contains the target Vehicle Speed in km/h.

Vehiclespeed: is broadcasted from the Vehicle ECU onto CAN2, then forwarded to Eth3 via the “Driver Control” ECU and forwarded once again onto CAN1 via the Edge Node GW This signal contains the current Vehicle Speed in km/h

5.3 Use case 3: Fleet management system collects location, speed and status of the vehicle

The fleet manager wants to visualize the location, speed and status (fault status) of vehicles on a map, thus this information is periodically sent via the Connectivity GW to back-office for visualization.

Vehiclespeed: is broadcasted from the Vehicle ECU onto CAN2, then forwarded to Eth3 via the “Driver Control” ECU and forwarded once again onto Eth1 via the Edge Node GW This signal contains the current Vehicle Speed in km/h

GPS_Position: Internal in the Connectivity ECU. In this case not exposed on internal networks

VehicleFaultStatus: Contains a summary of currently active faults in the vehicle. The Connectivity GW sends ISO14229:ReadDataByIdentifier:Request to the Driver display, which provides the VehicleFaultStatus data in ISO14229:ReadDataByIdentifier:Response.

5.4 Use case 4: The immobilizer detects a counterfeit key and blocks start of the engine

The key is turned in order to crank the engine; however, the immobilizer detects a counterfeit key and does not allow the engine to start cranking.

ImmobilizerStatus: Contains information from the immobilizer component inside the Connectivity GW to the Engine ECU that indicates whether the engine is allowed to start cranking. The signal is authenticated and the Engine ECU is able to determine that the content is genuine and from the Connectivity ECU. If the signal for some reason is not available, the Engine ECU will not start cranking. The signal is exposed on Eth1 and CAN1 networks.

5.5 Use case 5: Readout of emissions data using Client tester tool

Law enforcement connects to the OBD connector and reads emissions data stored in the Edge node. The Engine ECU periodically transmits emissions data that is stored in the Edge Node GW.

EmissionsData: Contains current emissions data. It is sent periodically from the Engine ECU to the Edge Node GW and is exposed on CAN1. The Edge node stores the data for future readout

EmissionsDataRequest: A tester client connected to the OBD connector sends this signal to the Edge Node GW as a request for emissions data and waits for an **EmissionsDataResponse** message.

EmissionsDataResponse: Contains the emissions data for the vehicle and is sent to the tester client via the OBD connector

6 References

[1] Deliverable D2.3, Security requirements for automotive on-board networks based on dark-side scenarios, EVITA (E-Safety Vehicle Intrusion Protected Applications) , Dec 2009, http://www.evita-project.org, Accessed 2018-03-29

[2] Deliverable 1.1, Security Requirements of Vehicle Security Architecture, PRESERVE (Preparing Secure V2X Communication Systems), June 2011, http://www.preserve-project.eu/, Accessed 2018-03-29

[3] AUTOSAR, “AUTOSAR Classic Platform 4.3.1”, https://www.autosar.org/standards/classic-platform/, Accessed 2018-04-02