Journal of Universal Computer Science, vol. 27, no. 8 (2021), 850-867
submitted: 8/4/2021, accepted: 2/5/2021, appeared: 28/8/2021 CC BY-ND 4.0

Cybersecurity Verification and Validation Testing in Automotive

Damjan Ekert
(ISCN GesmbH, Graz, Austria,
https://orcid.org/0000-0001-9301-242X, dekert@iscn.com)

Jürgen Dobaj
(TU Graz, Graz, Austria,
https://orcid.org/0000-0001-6460-8080, juergen.dobaj@tugraz.at)

Alen Salamun
(Real Security, Maribor, Slovenia,
alen.salamun@real-sec.com)

Abstract: The new generations of cars have a number of ECUs (Electronic Control Units) which are connected to a central gateway and need to pass cybersecurity integration tests to fulfil the homologation requirements of cars. Cars usually have a gateway server (few have additional domain servers) with Linux and a large number of ECUs which are real time control of actuators (ESP, EPS, ABS, etc. – usually they are multicore embedded controllers) connected by a real time automotive specific bus (CAN-FD) to the domain controller or gateway server. The norms (SAE J3061, ISO 21434) require cybersecurity related verification and validation. Fir the verification car manufacturers use a network test suite which runs > 2000 test cases and which have to be passed for homologation. These norms have impact on the way how car communication infrastructure is tested, and which cybersecurity attack patterns are checked before a road release of an ECU/car. This paper describes typical verification and validation approaches in modern vehicles and how such test cases are derived and developed.

Keywords: Automotive Cybersecurity, Verification, Validation, Best Practice Design Patterns
Categories: D.2.1, D.2.2, D.2.5, D.4.6
DOI: 10.3897/jucs.71833

1 Introduction

In Feb. 2021 VDA (German Automotive Association) AK 13 [ASPICE 2020] published the Automotive SPICE for cybersecurity assessment model (Figure 1). This will be used for cybersecurity homologation assessments which are mandatory for car makers from July 2022.

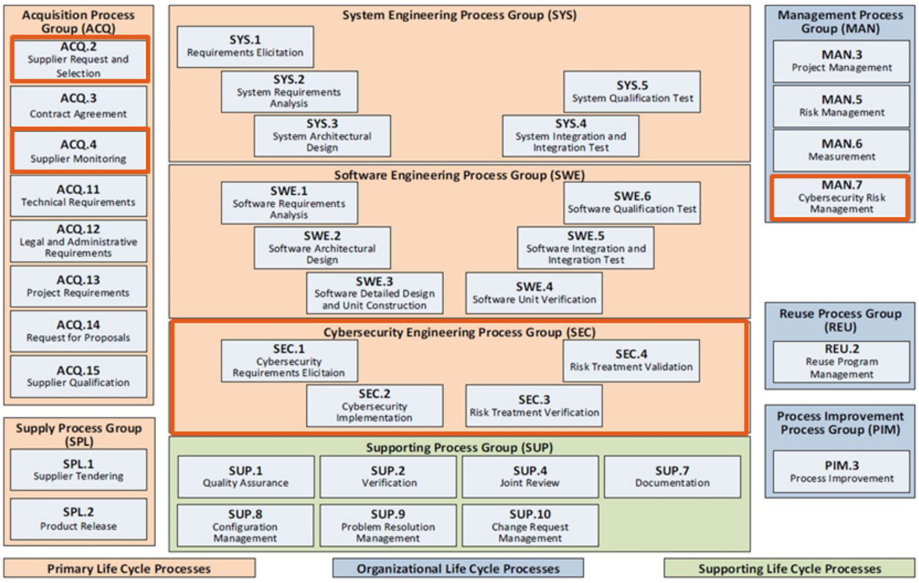

Acquisition Process Group (ACQ)
ACQ.2 Supplier Request and Selection
ACQ.3 Contract Agreement
ACQ.4 Supplier Monitoring
ACQ.11 Technical Requirements
ACQ.12 Legal and Administrative Requirements
ACQ.13 Project Requirements
ACQ.14 Request for Proposals
ACQ.15 Supplier Qualification
Supply Process Group (SPL)
SPL.1 Supplier Tendering
SPL.2 Product Release
System Engineering Process Group (SYS)
SYS.1 Requirements Elicitation
SYS.2 System Requirements Analysis
SYS.3 System Architectural Design
SYS.4 System Integration and Integration Test
SYS.5 System Qualification Test
Software Engineering Process Group (SWE)
SWE.1 Software Requirements Analysis
SWE.2 Software Architectural Design
SWE.3 Software Detailed Design and Unit Construction
SWE.4 Software Unit Verification
SWE.5 Software Integration and Integration Test
SWE.6 Software Qualification Test
Cybersecurity Engineering Process Group (SEC)
SEC.1 Cybersecurity Requirements Elicitaion
SEC.2 Cybersecurity Implementation
SEC.3 Risk Treatment Verification
SEC.4 Risk Treatment Validation
Supporting Process Group (SUP)
SUP.1 Quality Assurance
SUP.2 Verification
SUP.4 Joint Review
SUP.7 Documentation
SUP.8 Configuration Management
SUP.9 Problem Resolution Management
SUP.10 Change Request Management
Management Process Group (MAN)
MAN.3 Project Management
MAN.5 Risk Management
MAN.6 Measurement
MAN.7 Cybersecurity Risk Management
Reuse Process Group (REU)
REU.2 Reuse Program Management
Process Improvement Process Group (PIM)
PIM.3 Process Improvement
Primary Life Cycle Processes
Organizational Life Cycle Processes
Supporting Life Cycle Processes

Figure 1: Automotive SPICE Cybersecurity Assessment Model Processes

The model includes new processes

- MAN.7 Cybersecurity Risk Management (Management, performing and tracking a TARA – Cybersecurity Threat Analysis and Risk Analysis, and deriving cybersecurity goals)
- SEC.1 Cybersecurity Requirements Elicitation (Requirements derived from a TARA)
- SEC.2 Cybersecurity Implementation (Designing counter measures against the risks and threats and to achieve the cybersecurity goals)
- SEC.3 Risk Treatment Verification (verification against the cybersecurity requirements)
- SEC.4 Risk Treatment Validation (validation against the cybersecurity goals)

And extended 2 existing processes

- ACQ.2 Supplier Request and Selection (including cybersecurity in the tender)
- ACQ.4 Supplier Monitoring (monitoring the fulfilment of cybersecurity goals work products, requirements)

This paper addresses how the test concepts in cybersecure vehicle architectures are implemented in the SEC.3 and SEC.4 processes.

2 Cybersecurity Testing Related Work Products

In a working party SOQRATES a group of leading automotive suppliers, universities and training bodies and the EU Blueprint project DRIVES [Messnarz 2020] for automotive developed a concept of what typical work products are expected. Figure 2 illustrates the typical work products in listings for 04 to 06.

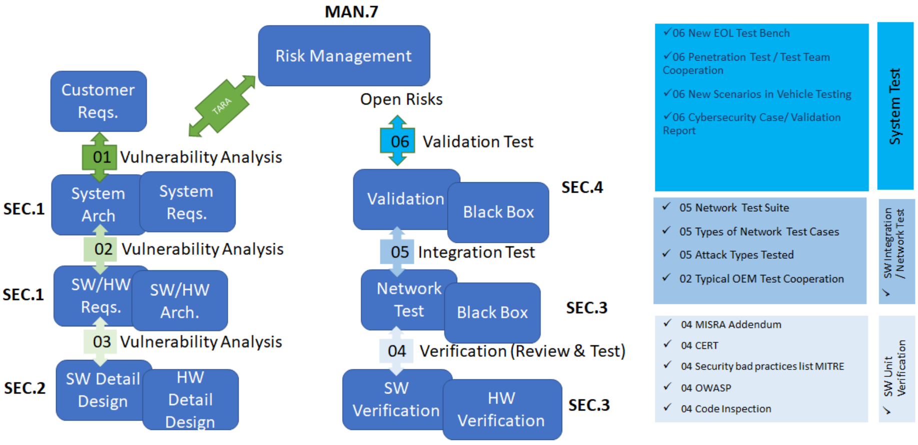


Figure 2: Cybersecurity related work products 04 – 06 for cybersecurity testing

SEC.3 Risk Treatment Verification: SW Unit Verification (see Figure 2 above)

04 MISRA and CERT

- Cybersecurity coding rules MISRA C:2012 Amendment 1 from 2016, and ISO/IEC TS 17961:2013 C-Secure

04 MITRE guidance

- Cybersecurity patterns (list of attacks and recommended actions) https://attack.mitre.org
- 04 OWASP
- The Open Web Application Security Project (owasp.org) with good and bad practices for programming in web based secure environments

04 Code Inspection

- Review checklists are extended to include cybersecurity relevant aspects. Typically coding rules are checked automatically. However, cybersecurity relevant SW parts need to be marked, need to be independent from the normal code (freedom from interference) applying an extended cybersecurity related checklist and rule set

SEC.3 Risk Treatment Verification: SW Integration / System Integration / Network Test (see Figure 2 above)

05 NTS Network-Test-Suite

- This is a test suite which is connected by a bus (LIN, CAN, CAN-FD, Ethernet) to the ECU (Electronic Control Unit) and has a programmable interface to write test cases (e.g. a replay attack). automatically performs tests and reports cybersecurity coverage.
- Advanced customers who have already an advanced cybersecurity architecture in place usually require to use a specific test suite and run given customer diagnostic tests on the ECU.

05 Types of Network Test Cases

- Test case types are usually structured by cybersecurity objectives and diagnostic chapters
- e.g. tests for authentication
  - e.g. tests for secure diagnostics and flashing
  - e.g. test for secure boot
  - e.g. tests for replay attacks
  - e.g. tests for UDS (Unified Diagnose Services) diagnostic functions and trying to access secure parts (normally UDS $27) by non-authenticated diagnostic protocol functions
  - e.g. denial of service attacks
  - e.g. behaviour in case messages get lost, come too often in incorrect time frame etc.

05 Attack Types Tested

- All attack types (usually the MITRE catalogue is checked and those parts which are applicable the product being developed are selected) must be covered.

02 Typical OEM Test Cooperation

- Advanced car manufacturers such as VW, Daimler, BMW, etc. provide requirements for the test tool and test procedure set up: e.g. NTS Network Test Suite for Daimler AG, FAT Tool Suite for BMW, CANoe with CAPL programmed test cases in a combination with a HIL platform (HIL – Hardware in the Loop) in case of VW, etc.

SEC.4 Risk Treatment Validation: Software & System Test (see Figure 2 above)

06 Test Bench for Cybersecurity

- Cars in production at the end of line test receive an IP address and all certificates and keys configured. In a test bench these key updates are tested.
- 06 Penetration Test / Test Team Cooperation
- Penetration testing is done an external cybersecurity teste team. This team does not receive all internal designs but the cybersecurity goals and technical data about the ECU.

06 Vehicle Integration Testing

- Vehicle integration requires now e.g. that the test driver flashes new keys in the vehicle set up.

- 06 Cybersecurity Case/ Validation Report
- A cybersecurity manager produces reports showing that 100% of cybersecure relevant requirements have been tested and passed.

3 Cybersecurity Requirements for Verification

The cybersecurity norms require to perform a TARA (Threat Analysis and Risk Analysis). The TARA attributes are described in the ISO 21434 norm [SAE 2016][Schmittner 2019][Schmittner 2019 2][ISO 21434 2020][VDA 2021] and a TARA delivers an impact level, a threat level and a security level (combination of impact and threat level by a risk table). [ISO 21434 2020][Macher 2016][Macher 2017][Macher 2017 2] Also, each line in the TARA includes a cybersecurity goals, if the security level shows a security rating (se Figure 4 example extract from a TARA).

Before a TARA is performed a cybersecurity item analysis or threat model is drawn which shows the system, all critical interfaces and critical data and groups the assets which can be attacked [Dobaj 2018][Dobaj 2019][Macher 2020].

See the example in Figure 3 which describes a 6 phase e-motor EPS (Electric Power Steering). One critical interface is e.g. the steering angle request which can lead to unwanted steering of the vehicle [Macher 2019][Macher 2019 2][Macher 2019 3][Messnarz 2016][Messnarz 2016 2][Messnarz 2017][Armengaud 2019][Messnarz 2019 2][Veledar 2019].

Figure 4 shows example lines and selected attributes of the related TARA where the ADAS steering angle request as a critical signal is rated with a threat level medium, and because (not visible) the impact level was critical (since sudden steering leads almost always to an accident), the security level is high.

The assigned security goal in Figure 4 e.g. is “Prevent unwanted steering due to unauthorized commands and assure secure logging of received commands over a period of 800 ms.

The cybersecurity goal is then broken down to system requirements (authorize and authenticate commands) and cybersecure critical SW function and cybersecure critical software data requirements. For the critical function requirements related critical data are referenced, the SW states in which the function is active (white listing) is defined, and security objectives are assigned (see Figure 5).

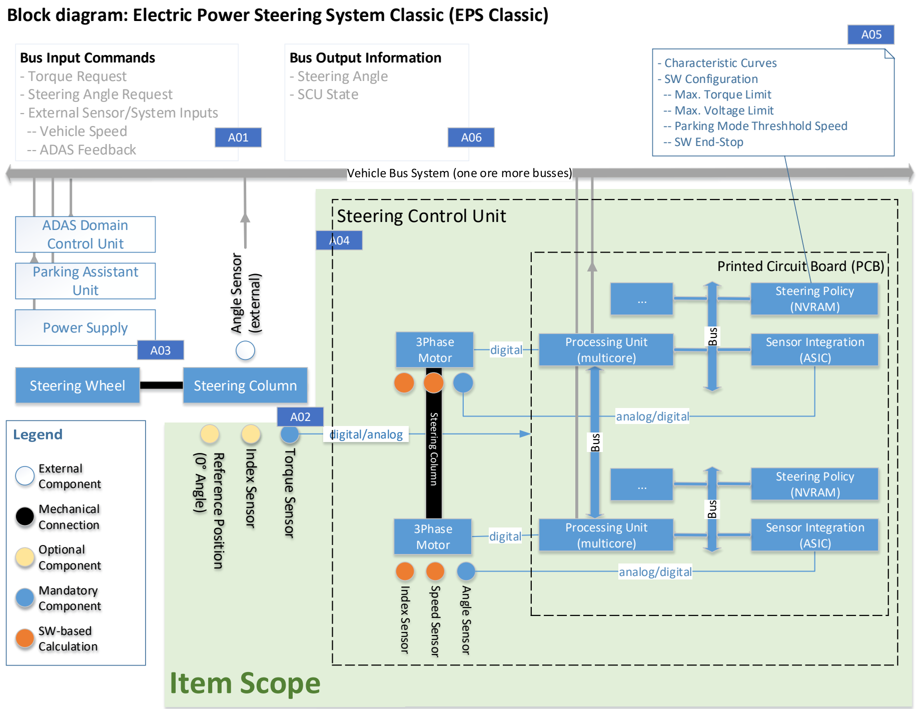
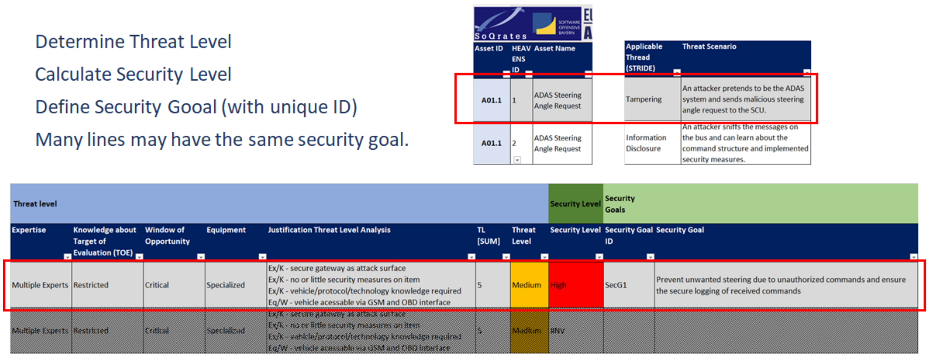


Figure 3: Cybersecurity item analysis – Critical assets and interfaces



Determine Threat Level

Calculate Security Level

Define Security Gooal (with unique ID)

Many lines may have the same security goal.

Asset ID	HEAVENS ID	Asset Name
A01.1	1	ADAS Steering Angle Request
A01.1	2	ADAS Steering Angle Request

Applicable Thread (STRIDE)	Threat Scenario
Tampering	An attacker pretends to be the ADAS system and sends malicious steering angle request to the SCU.
Information Disclosure	An attacker sniffs the messages on the bus and can learn about the command structure and implemented security measures.

Threat level							Security Level	Security Goals	
Expertise	Knowledge about Target of Evaluation (TOE)	Window of Opportunity	Equipment	Justification Threat Level Analysis	TL [SUM]	Threat Level	Security Level	Security Goal ID	Security Goal
Multiple Experts	Restricted	Critical	Specialized	Ex/K - secure gateway as attack surface Ex/K - no or little security measures on item Ex/K - vehicle/protocol/technology knowledge required Eq/W - vehicle acessable via GSM and OBD interface	5	Medium	High	SecG1	Prevent unwanted steering due to unauthorized commands and ensure the secure logging of received commands
Multiple Experts	Restricted	Critical	Specialized	Ex/K - secure gateway as attack surface Ex/K - no or little security measures on item Ex/K - vehicle/protocol/technology knowledge required Eq/W - vehicle acessable via GSM and OBD interface	5	Medium	#NV		

Figure 4: Cybersecurity goals – results from a TARA

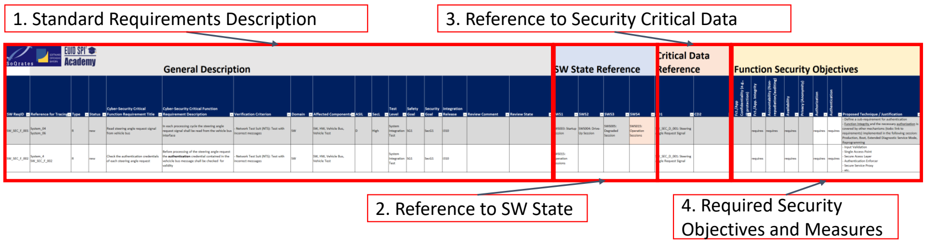
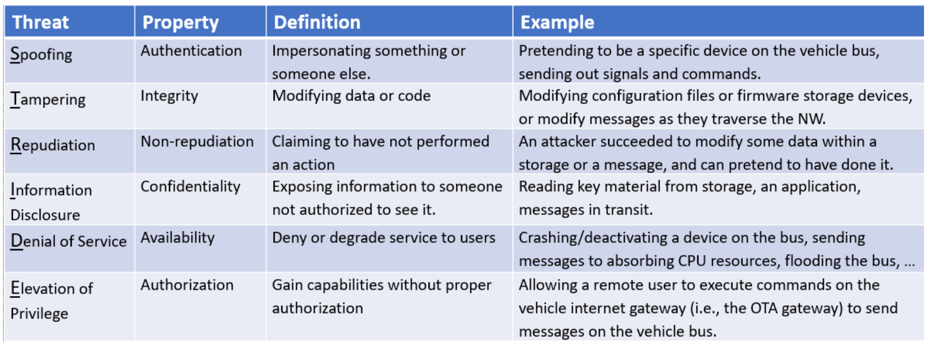

1. Standard Requirements Description
3. Reference to Security Critical Data
General Description
SW State Reference
Critical Data Reference
Function Security Objectives
2. Reference to SW State
4. Required Security Objectives and Measures

Figure 5: Cybersecurity critical function requirements

For each cybersecurity objective corresponding counter measures are programmed and need to be tested.



Threat	Property	Definition	Example
Spoofing	Authentication	Impersonating something or someone else.	Pretending to be a specific device on the vehicle bus, sending out signals and commands.
Tampering	Integrity	Modifying data or code	Modifying configuration files or firmware storage devices, or modify messages as they traverse the NW.
Repudiation	Non-repudiation	Claiming to have not performed an action	An attacker succeeded to modify some data within a storage or a message, and can pretend to have done it.
Information Disclosure	Confidentiality	Exposing information to someone not authorized to see it.	Reading key material from storage, an application, messages in transit.
Denial of Service	Availability	Deny or degrade service to users	Crashing/deactivating a device on the bus, sending messages to absorbing CPU resources, flooding the bus, …
Elevation of Privilege	Authorization	Gain capabilities without proper authorization	Allowing a remote user to execute commands on the vehicle internet gateway (i.e., the OTA gateway) to send messages on the vehicle bus.

Figure 6: Cybersecurity objectives / Required Property

Continuing with the example from steering a MAC will be used for only accepting authenticated and authorized messages:

The MAC (Message Authentication Codes) shall provide the following security properties/objectives:
- freshness (Sec. Objective)
- authentication (Sec. Objective)
- information/payload: steering angle request
- integrity (Sec. Objective)

Algorithmic Details we may assign to the SW Level Analysis:
- freshness: a new random number is added to each message
- authentication: via shared secret
- information/payload: steering angle request
- integrity: via HASH plus CRC for error correction from safety
- CAN-message = [h1|steering angle request|CRC]
- p1 = [shared-secret | increase-of-random-number | steering angle request]
- h1 = HASH[shared-secret | HASH[p1]]

4 Integration Testing Verification Strategy - Network Test Suites Approach

Figure 7 illustrates a typical general test framework [Akka 2021][Vector 2021] applied and Figure 7 shows the specific test environment which typically is used, for instance, in Daimler projects.

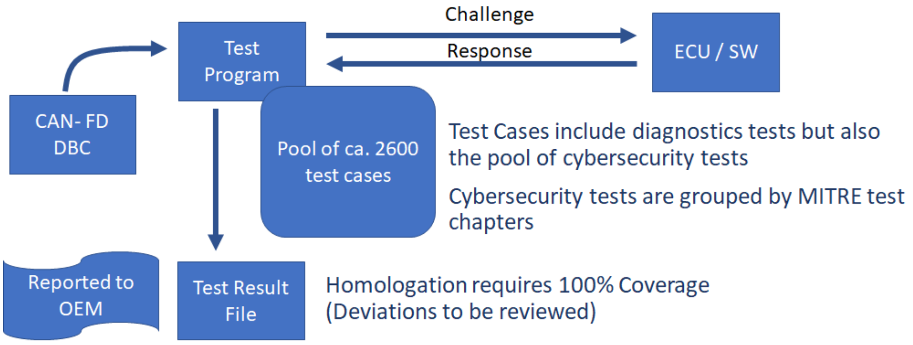

Challenge
Response
Test Program
ECU / SW
CAN- FD DBC
Pool of ca. 2600 test cases
Test Cases include diagnostics tests but also the pool of cybersecurity tests
Cybersecurity tests are grouped by MITRE test chapters
Reported to OEM
Test Result File
Homologation requires 100% Coverage (Deviations to be reviewed)

Figure 7: Typical set up of network test suites

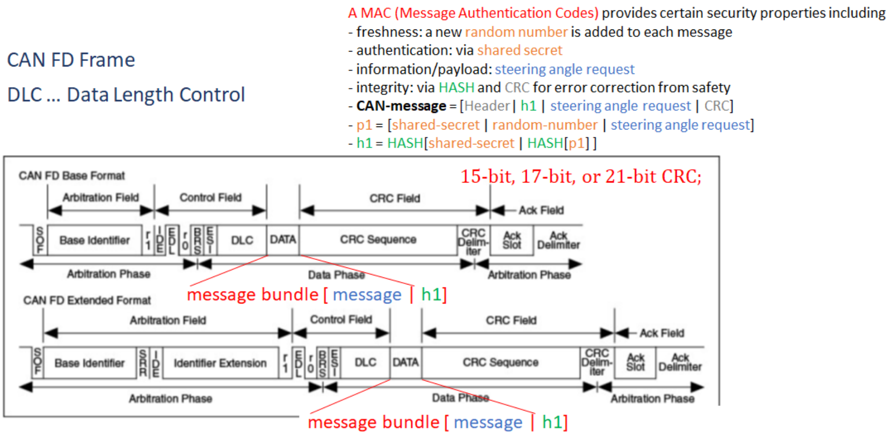
A tool affordable for research teams, and small companies is PCAN where you can program test cases in C/C++ and even in Visual Basic. There is also a whole set functions to import DBC files, sniff the bus, and change, adapt and replay messages.

In vehicles a CAN bus is based on a norm J1939 (https://wiki2.org/en/SAE_J1939) in which the message format is standardized (see Figure 9). Research is dine to create more secure protocols based on Ethernet [Dobaj 2020].

A CAN FD [ISO 11898] is structured like a CAN and allows Flexible Data (FD) rates. To add to a CAN a cybersecurity (which was not meant at the beginning) a cybersecurity related message bundle is created which contains both, the message and the cybersecurity related message part to authenticate, authorize, check the command/message (see Figure 8 to Figure 10).

It is important to not confuse security with the built in CRC checksum algorithms (see Figure 8) which support a 15-bit, 17-bit, or 21-bit CRC. This CRC is used for an E2E (End to End) validation of messages, that messages have not been altered in general.

The added SecOC bundle is the protection part against the attackers (see h1 in Figure 8).



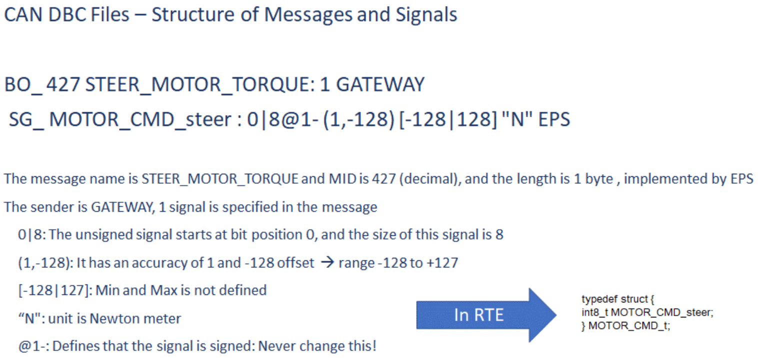
CAN FD Frame

DLC ... Data Length Control

A MAC (Message Authentication Codes) provides certain security properties including
- freshness: a new random number is added to each message
- authentication: via shared secret
- information/payload: steering angle request
- integrity: via HASH and CRC for error correction from safety
- **CAN-message** = [Header | h1 | steering angle request | CRC]
- p1 = [shared-secret | random-number | steering angle request]
- h1 = HASH[shared-secret | HASH[p1]]

Figure 8: CAN FD Structure and Extension for Cybersecurity

For the design of a vehicle (a vehicle contains many ECUs Electronic Control Units connected by a bus) the CAN message catalogue is specified according to a nom (J1939) and imported by the ECUs. ECU software which is developed for cybersecurity has usually a standard service architecture supported by Autosar >= 4.3 (see Figure 10) which offers services to manage secure communication (SEcOC component), connects to a hardware security module (vHSM and HSM), and allows functional programs to use encryption, decryption, signing, authentication, hash algorithms, etc. The decrypted data are made available via the RTE (Run Time Environment).



CAN DBC Files – Structure of Messages and Signals

BO_ 427 STEER_MOTOR_TORQUE: 1 GATEWAY

SG_ MOTOR_CMD_steer : 0|8@1- (1,-128) [-128|128] "N" EPS

The message name is STEER_MOTOR_TORQUE and MID is 427 (decimal), and the length is 1 byte , implemented by EPS

The sender is GATEWAY, 1 signal is specified in the message

0|8: The unsigned signal starts at bit position 0, and the size of this signal is 8

(1,-128): It has an accuracy of 1 and -128 offset → range -128 to +127

[-128|127]: Min and Max is not defined

"N": unit is Newton meter

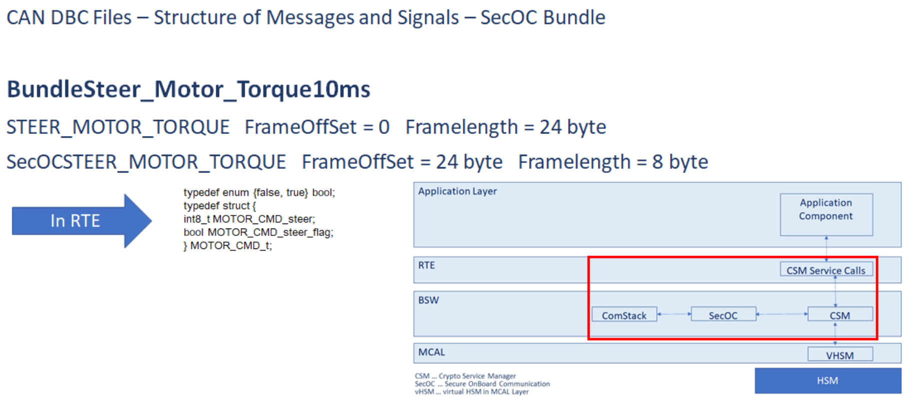
@1-: Defines that the signal is signed: Never change this!

In RTE

```
typedef struct {
int8_t MOTOR_CMD_steer;
} MOTOR_CMD_t;
```

Figure 9: Message Structure without Security Bundle

Figures 9 and 10 show a typical outcome in the RTE, where you can read the requested motor torque and also you can read by a flag if the authentication was ok.



CAN DBC Files – Structure of Messages and Signals – SecOC Bundle

BundleSteer_Motor_Torque10ms

STEER_MOTOR_TORQUE FrameOffSet = 0 Framelength = 24 byte

SecOCSTEER_MOTOR_TORQUE FrameOffSet = 24 byte Framelength = 8 byte

In RTE

```
typedef enum {false, true} bool;
typedef struct {
int8_t MOTOR_CMD_steer;
bool MOTOR_CMD_steer_flag;
} MOTOR_CMD_t;
```

CSM ... Crypto Service Manager
SecOC ... Secure OnBoard Communication
vHSM ... virtual HSM in MCAL Layer

Figure 10: Message Bundles integrating a message and a security part

In testing Automotive uses an integrated test equipment (Figures 7 and 8) which allow to change, replay, etc. messages and observe the outcome and read by a protocol the values from the RTE.

5 Test Case Design for Verification

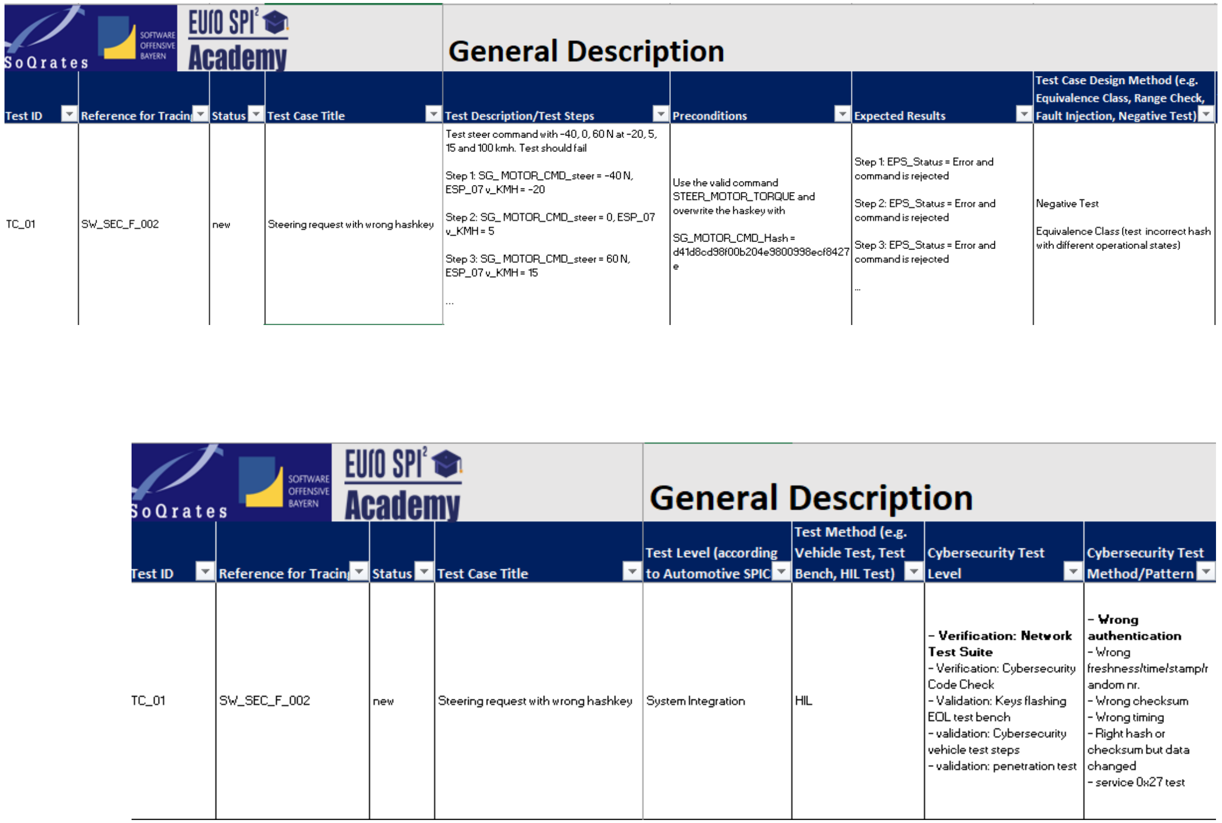
Test cases in Automotive need to be linked to the corresponding requirements. This traceability of requirements to test cases and test results is a must in Automotive to prove that features are complete when homologating a vehicle. Different norms check this traceability such as Automotive SPICE [ASPICE 2017] ASPICE 2017 2] and Automotive SPICE extension for Cybersecurity [ASPICE 2021], Functional Safety, and cybersecurity norms like ISO 21434 for proof of coverage of a cybersecurity case. Assessments are made to check this coverage and if the development process supports this [Ekert 2020][Höhn 2015][Messnarz 2007][Messnarz 2009][Messnarz 2012][Messnarz 2019][Schlager 2018][Wegner 2020].

Since the tool set up is supported by a test framework with scripting the test cases usually are designed in a database, linked to requirements and scripts write test results logs which can be imported back to the database to provide a test status.

In SOQRATES best practice attributes for describing such test cases are discussed and in Figure 11 below you find such an example specification using attributes like:

- Text case ID
- Related SW Security Requirement(s)
- Test Case Title
- Test Case Description / Test Steps
- Preconditions, Expected Results
- Test Case Design method used
- Test Level
- Test (Tool) Environment
- Cybersecurity Test level

- Cybersecurity test method / Verification Criteria / Test case design patterns



General Description

Test ID	Reference for Tracin	Status	Test Case Title	Test Description/Test Steps	Preconditions	Expected Results	Test Case Design Method (e.g. Equivalence Class, Range Check, Fault Injection, Negative Test)
TC_01	SW_SEC_F_002	new	Steering request with wrong hashkey	Test steer command with -40, 0, 60 N at -20, 5, 15 and 100 kmh. Test should fail Step 1: SG_ MOTOR_CMD_steer = -40 N, ESP_07 v_KMH = -20 Step 2: SG_ MOTOR_CMD_steer = 0, ESP_07 v_KMH = 5 Step 3: SG_ MOTOR_CMD_steer = 60 N, ESP_07 v_KMH = 15 ...	Use the valid command STEER_MOTOR_TORQUE and overwrite the haskey with SG_MOTOR_CMD_Hash = d41d8cd98f00b204e9800998ecf8427e	Step 1: EPS_Status = Error and command is rejected Step 2: EPS_Status = Error and command is rejected Step 3: EPS_Status = Error and command is rejected ...	Negative Test Equivalence Class (test incorrect hash with different operational states)

General Description

Test ID	Reference for Tracin	Status	Test Case Title	Test Level (according to Automotive SPIC	Test Method (e.g. Vehicle Test, Test Bench, HIL Test)	Cybersecurity Test Level	Cybersecurity Test Method/Pattern
TC_01	SW_SEC_F_002	new	Steering request with wrong hashkey	System Integration	HIL	**- Verification: Network Test Suite** - Verification: Cybersecurity Code Check - Validation: Keys flashing EOL test bench - validation: Cybersecurity vehicle test steps - validation: penetration test	**- Wrong authentication** - Wrong freshness/time/stamp/random nr. - Wrong checksum - Wrong timing - Right hash or checksum but data changed - service 0x27 test

Figure 11: Cybersecurity verification test case example

6 Test Case Approach in Case of Validation

While the cybersecurity verification gives the test team a white box view with a number of cybersecurity related function and data requirements, the validation is based on a black box view. The testers know the functions, the technical description and the cybersecurity goals, but they do not receive in e.g. penetration testing the details. This is used to simulate a real case attacker from outside.

Also, attacker teams for penetration test have a structured test strategy which is called “Creating Adversary Emulation Plans based on attack patterns derived from MITRE” (Figure 12) [MITRE 2021][MITRE 2021 2].

Using the test specification, the editors and scripts in the tools framework are used to e.g. resend the message with wrong message counter (which e.g. would happen in a replay attack).

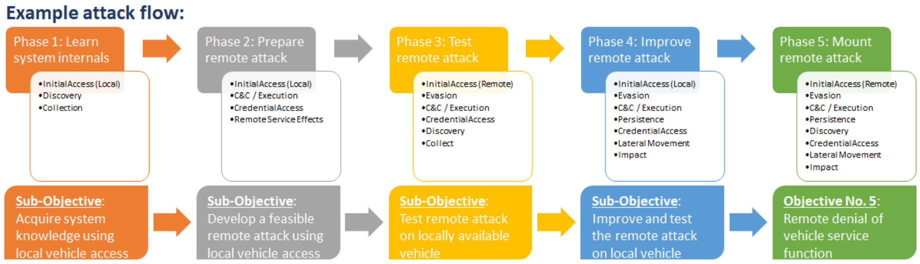
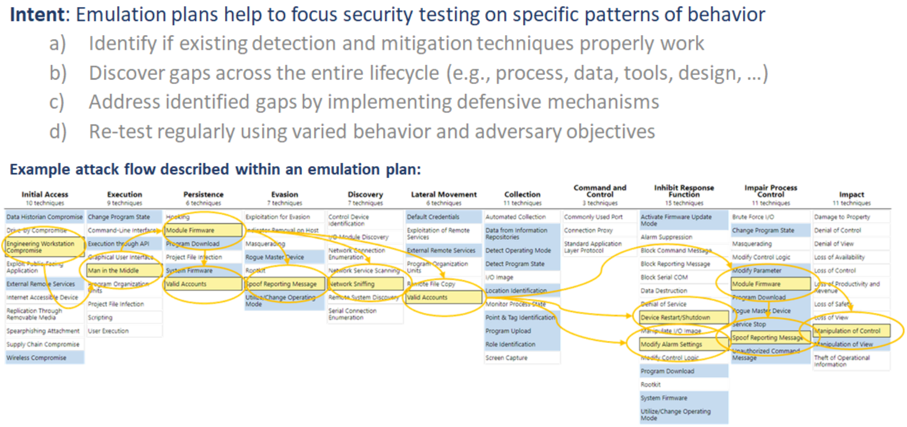

Example attack flow:
Phase 1: Learn system internals
•InitialAccess (Local)
•Discovery
•Collection
Phase 2: Prepare remote attack
•InitialAccess (Local)
•C&C / Execution
•CredentialAccess
•RemoteService Effects
Phase 3: Test remote attack
•InitialAccess (Remote)
•Evasion
•C&C / Execution
•CredentialAccess
•Discovery
•Collect
Phase 4: Improve remote attack
•InitialAccess (Local)
•Evasion
•C&C / Execution
•Persistence
•CredentialAccess
•Lateral Movement
•Impact
Phase 5: Mount remote attack
•InitialAccess (Remote)
•Evasion
•C&C / Execution
•Persistence
•Discovery
•CredentialAccess
•Lateral Movement
•Impact
Sub-Objective: Acquire system knowledge using local vehicle access
Sub-Objective: Develop a feasible remote attack using local vehicle access
Sub-Objective: Test remote attack on locally available vehicle
Sub-Objective: Improve and test the remote attack on local vehicle
Objective No. 5: Remote denial of vehicle service function

Figure 12: Cybersecurity attack flow analysis

For the preparation of such plans the MITRE Also, attacker teams for penetration test have a structured test strategy which is called “Creating Adversary Emulation Plans based on attack patterns derived from MITRE” (Figure 12).

Also, attacker teams for penetration test have a structured test strategy which is called “Creating Adversary Emulation Plans based on attack patterns derived from MITRE” (Figure 12).

In Mitre.org you can select the attack navigator functionality, create a new layer model for you, and start planning. This might result in a strategy described in Figure 13.



Intent: Emulation plans help to focus security testing on specific patterns of behavior

a) Identify if existing detection and mitigation techniques properly work
b) Discover gaps across the entire lifecycle (e.g., process, data, tools, design, …)
c) Address identified gaps by implementing defensive mechanisms
d) Re-test regularly using varied behavior and adversary objectives

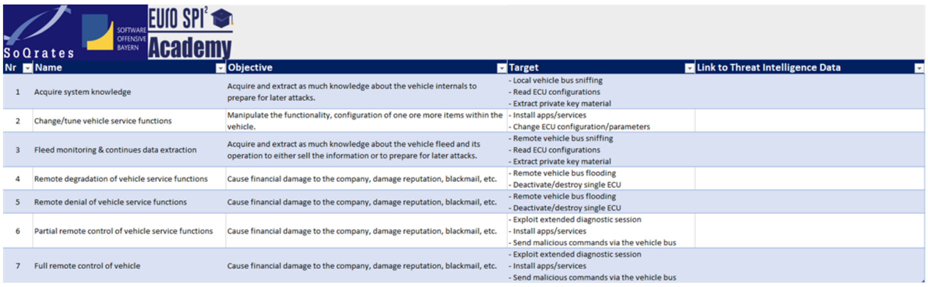
Example attack flow described within an emulation plan:

Initial Access 10 techniques	Execution 9 techniques	Persistence 6 techniques	Evasion 7 techniques	Discovery 7 techniques	Lateral Movement 6 techniques	Collection 11 techniques	Command and Control 3 techniques	Inhibit Response Function 15 techniques	Impair Process Control 11 techniques	Impact 11 techniques
Data Historian Compromise	Change Program State	Hooking	Exploitation for Evasion	Control Device Identification	Default Credentials	Automated Collection	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Drive-by Compromise	Command-Line Interface	Module Firmware	Indicator Removal on Host	I/O Module Discovery	Exploitation of Remote Services	Data from Information Repositories	Connection Proxy	Alarm Suppression	Change Program State	Denial of Control
Engineering Workstation Compromise	Execution through API	Program Download	Masquerading	Network Connection Enumeration	External Remote Services	Detect Operating Mode	Standard Application Layer Protocol	Block Command Message	Masquerading	Denial of View
Exploit Public-Facing Application	Graphical User Interface	Project File Infection	Rogue Master Device	Network Service Scanning	Program Organization Units	Detect Program State		Block Reporting Message	Modify Control Logic	Loss of Availability
External Remote Services	Man in the Middle	System Firmware	Rootkit	Network Sniffing	Remote File Copy	I/O Image		Block Serial COM	Modify Parameter	Loss of Control
Internet Accessible Device	Program Organization Units	Valid Accounts	Spoof Reporting Message	Remote System Discovery	Valid Accounts	Location Identification		Data Destruction	Module Firmware	Loss of Productivity and Revenue
Replication Through Removable Media	Project File Infection		Utilize/Change Operating Mode	Serial Connection Enumeration		Monitor Process State		Denial of Service	Program Download	Loss of Safety
Spearphishing Attachment	Scripting					Point & Tag Identification		Device Restart/Shutdown	Rogue Master Device	Loss of View
Supply Chain Compromise	User Execution					Program Upload		Manipulate I/O Image	Service Stop	Manipulation of Control
Wireless Compromise						Role Identification		Modify Alarm Settings	Spoof Reporting Message	Manipulation of View
						Screen Capture		Modify Control Logic	Unauthorized Command Message	Theft of Operational Information
								Program Download		
								Rootkit		
								System Firmware		
								Utilize/Change Operating Mode		

Figure 13: Cybersecurity attack flow analysis using MITRE Attack Navigator

MITRE is based on a collection of attack patterns observed in ICT industry, and thus the vehicle testers need to adapt the proposed ICT strategy to a vehicle based strategy. This results in an attack strategy as described in Figure 14.

This way pen Testing structured approach based on the assumed objectives of an adversary pan that was derived from MITRE and adapted for automotive.



Nr	Name	Objective	Target	Link to Threat Intelligence Data
1	Acquire system knowledge	Acquire and extract as much knowledge about the vehicle internals to prepare for later attacks.	- Local vehicle bus sniffing - Read ECU configurations - Extract private key material	
2	Change/tune vehicle service functions	Manipulate the functionality, configuration of one ore more items within the vehicle.	- Install apps/services - Change ECU configuration/parameters	
3	Fleed monitoring & continues data extraction	Acquire and extract as much knowledge about the vehicle fleed and its operation to either sell the information or to prepare for later attacks.	- Remote vehicle bus sniffing - Read ECU configurations - Extract private key material	
4	Remote degradation of vehicle service functions	Cause financial damage to the company, damage reputation, blackmail, etc.	- Remote vehicle bus flooding - Deactivate/destroy single ECU	
5	Remote denial of vehicle service functions	Cause financial damage to the company, damage reputation, blackmail, etc.	- Remote vehicle bus flooding - Deactivate/destroy single ECU	
6	Partial remote control of vehicle service functions	Cause financial damage to the company, damage reputation, blackmail, etc.	- Exploit extended diagnostic session - Install apps/services - Send malicious commands via the vehicle bus	
7	Full remote control of vehicle	Cause financial damage to the company, damage reputation, blackmail, etc.	- Exploit extended diagnostic session - Install apps/services - Send malicious commands via the vehicle bus	

Figure 14: Example Attack Strategy Sheet

The penetration test cases are then grouped by these attack flows / scenarios

7 Conclusions

Modern cars have a gateway server with an IP address and are connected to the network and the server / gateway can be attacked [Macher 2019] [Messnarz 2020]. Cars will in future expand into e-city, e-environment modes building a connected society [Feuer 2002] [Messnarz 2020].

Cybersecurity norms like ISO 21434 and legal regulations for cars in UNECE law require the coverage of cybersecurity norms and requirements and to prove that by cybersecurity verification and validation.

Automotive SPICE for cybersecurity has been recently published in Feb. 2021 and has processes included for cybersecurity verification and cybersecurity validation.

This paper described how test strategies in cybersecure vehicles are implemented and how this mas onto a cybersecure vehicle architecture.

Acknowledgements

We are grateful to the guest editors Dr Richard Messnarz from ISCN GesmbH and Dr Georg Macher from TU Graz who are experts of the cybersecurity task force in the German and Austrian working group SOQRATES (soqrates.eurospi.net) and who contributed state of the art experiences in the field.

We are grateful to the European Commission which has funded the BLUEPRINT project DRIVES (2018 – 2021) [Messnarz 2020] [DRIVES 2021] [Stolfa 2020],[Stolfa 2020 2]. In this case the publications reflect the views only of the author(s), and the Commission cannot be held responsible for any use, which may be made of the information contained therein.

We are grateful to the EuroSPI community and conference series (www.eurospi.net) in which experts share experiences since 1994 and cybersecurity and JUCS has been promoted and experts from that community submitted papers to this journal.

We are grateful to the EU Project ECQA Certified Cybersecurity Engineer and Manager – Automotive Sector, Erasmus+ Programme, Grant Agreement No. 2020-1-CZ01-KA203-078494.

We are grateful to a working party of Automotive suppliers SOQRATES [SOQRATES 2021] (https://soqrates.eurospi.net) who exchange knowledge about such assessment strategies. This includes:

Aschenberger Werner (KTM AG), Bauer-Wukitsevits Robert (msg Plaut), Böhner Martin (Elektrobit), Brasse Michael (HELLA), Bressau Ernst (BBraun), Dallinger Martin (ZF), Dorociak Rafal (HELLA), Dreves Rainer (Continental), Ekert Damjan (ISCN), Forster Martin (ZKW), Geipel Thomas (BOSCH Engineering), Grave Rudolf (Elektrobit), Griessnig Gerhard (AVL), Gruber Andreas (Kreisel Electric), Habel Stephan (Continental), Hällmayer Frank (SF), Haunert Lutz, Karner Christoph (KTM AG), Kinalzyk Dietmar (AVL), König Frank (ZF), Lichtenberger Christoph (MAGNA), Lindermuth Peter (MAGNA), Macher Georg (TU Graz), Maier Bjoern (AVL), Mandic Irenka 8MAGNA), Maric Dijaz (LORIT Consultancy), Mayer Ralf (BOSCH Engineering), Mergen Silvana (TDK EPCOS), Messnarz Richard (ISCN), Much Alexander (Elektrobit), Nikolov Borislav (msg Plaut), Oehler Couso Daniel (MAGNA), Riel Andreas (ISCN Group), Rieß Armin (Braun), Santer Christian (AVL), Schlager Christian (MAGNA), Schmittner Christoph (AIT),

Schubert Marion (ZKW), Sechser Bernhard (Process Fellows), Sokic Ivan (Continental), Sporer Harald (Infineon), Stahl Florian (msg Plaut), Wachter Stefan (msg Plaut), Walker Alastair (LORIT Consultancy), Wegner Thomas (ZF), Geyer Dirk (AVL), Dobaj Jürgen (TU Graz), Wagner Hans (msg Group), Aust Detlev (AustCON).

References

[Akka 2021] Akka Technologies, NETWORKING TEST SUITE (GT-NTS), Network Test Suite Description used for Daimler, https://www.akka-technologies.com/app/uploads/digital-solution-nts-1.pdf, last visited 11.4.2021

[Armengaud et al., 2019] Armengaud, E.; Frager, S.; Jones, S.; Massoner, A.; Parrilla, A. F.; Wikström, N. & Macher, G., Development Framework for Longitudinal Automated Driving Functions with Off-board Information Integration arXiv preprint arXiv:1906.10009, 2019

[ASPICE 2017 2] Automotive SPICE © Guidelines, 2nd Edition Nov 2017, VDA QMC Working Group 13, Nov. 2017

[ASPICE 2017] Automotive SPICE © 3.1, Process Assessment Model, VDA QMC Working Group 13 / Automotive SIG, Nov. 2017

[ASPICE 2021] Automotive SPICE for Cybersecurity, 1st Edition, Feb. 2021, VDA QMC Working Group 13, Feb. 2021

[Dobaj 2018] Dobaj J., Iber J., Krisper M., Kreiner C. (2018) Towards Executable Dependability Properties. In: Larrucea X., Santamaria I., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2018. Communications in Computer and Information Science, vol 896. Springer, Cham. https://doi.org/10.1007/978-3-319-97925-0_28

[Dobaj 2019] Dobaj J., Krisper M., Macher G. (2019) Towards Cyber-Physical Infrastructure as-a-Service (CPIaaS) in the Era of Industry 4.0. In: Walker A., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2019. Communications in Computer and Information Science, vol 1060. Springer, Cham. https://doi.org/10.1007/978-3-030-28005-5_24

[Dobaj 2020] Dobaj J., Seidl M., Krug T., Krisper M., Macher G. (2020) A Seamless Self-configuring EtherCAT Master Redundancy Protocol. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham. https://doi.org/10.1007/978-3-030-56441-4_28

[DRIVES 2021] EU Blueprint Project DRIVES, https://www.project-drives.eu/, last access date: April 6, 2021

[Ekert et al., 2020] Ekert D., Messnarz R., Norimatsu S., Zehetner T., Aschbacher L. (2020) Experience with the Performance of Online Distributed Assessments – Using Advanced Infrastructure. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham. https://doi.org/10.1007/978-3-030-56441-4_47

[Feuer et al., 2002] Eva Feuer, Richard Messnarz, Nacho Sanchez, Best practices in e-commerce: strategies, skills, and processes, Proceedings of the E2002 Conference, E-Business and E-Work, Novel solutions for a global networked economy, eds. Brian Stanford Smith, Enrica Chiozza, IOS Press, Amsterdam, Berlin, Oxford, Tokyo, Washington, 2002

[Höhn et al., 2015] Holger Höhn, Bernhard Sechser, Klaudia Dussa-Zieger, Richard Messnarz, Bernd Hindel, Software Engineering nach Automotive SPICE: Entwicklungsprozesse in der Praxis-Ein Continental-Projekt auf dem Weg zu Level 3, Kapitel: Systemdesign, dpunkt. Verlag, 2015

[Innerwinkler et al., 2018] Innerwinkler, P.; Karci, A. E. H.; Tarkiainen, M.; Troglia, M.; Kinav, E.; Ozan, B.; Aydemir, E.; Derse, C.; Stettinger, G.; Watzenig, D. & others, TrustVehicle--Improved Trustworthiness and Weather-Independence of Conditionally Automated Vehicles in Mixed Traffic Scenarios International Forum on Advanced Microsystems for Automotive Applications, 2018, 75-89

[ISO 11898] CAN FD, ISO 11898-1 Road vehicles — Controller area network (CAN) — Data link layer and physical signalling

[ISO 21434 2020] ISO 21434ISO/SAE 21434 DIS, Road vehicles – Cybersecurity engineering, DIS version, Feb 2020

[ISO 26262 2011] ISO - International Organization for Standardization. "ISO 26262 Road vehicles Functional Safety Part 1-10", 2011.

[ISO 26262 2018] ISO – International Organization for Standardization. "ISO CD 26262-2018 2nd Edition Road vehicles Functional Safety", 2018

[Krisper 2019] Krisper M., Dobaj J., Macher G., Schmittner C. (2019) RISKEE: A Risk-Tree Based Method for Assessing Risk in Cyber Security. In: Walker A., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2019. Communications in Computer and Information Science, vol 1060. Springer, Cham. https://doi.org/10.1007/978-3-030-28005-5_4

[Krisper 2020] Krisper M., Dobaj J., Macher G. (2020) Assessing Risk Estimations for Cyber-Security Using Expert Judgment. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham. https://doi.org/10.1007/978-3-030-56441-4_9

[Macher et al., 2016] Macher, G.; Sporer, H.; Brenner, E. & Kreiner, C. "Supporting Cyber-security based on Hardware-Software Interface Definition Systems", Software and Services Process Improvement - 23nd European Conference, EuroSPI 2016 Proceedings, Springer, 2016.

[Macher et al., 2017 2] Macher G., Much A., Riel A., Messnarz R., Kreiner C. (2017) Automotive SPICE, Safety and Cybersecurity Integration. In: Tonetta S., Schoitsch E., Bitsch F. (eds) Computer Safety, Reliability, and Security. SAFECOMP 2017. Lecture Notes in Computer Science, vol 10489. Springer, Cham

[Macher et al., 2017] G. Macher, R. Messnarz, C. Kreiner, et. al, Integrated Safety and Security Development in the Automotive Domain, Working Group 17AE-0252/2017-01-1661, SAE International, June 2017

[Macher et al., 2019 2] Macher G., Schmittner C., Dobaj J., Armengaud E., Messnarz R. (2020). An integrated view on automotive spice, functional safety and cyber-security. In 2020 SAE Technical Papers 2020-01-0145, SAE International.

[Macher et al., 2019 3] Macher, G.; Druml, N.; Veledar, O. & Reckenzaun, J., Safety and Security Aspects of Fail-Operational Urban Surround perceptION (FUSION), International Symposium on Model-Based Safety and Assessment, 2019, 286-300

[Macher et al., 2019] Macher, G.; Diwold, K.; Veledar, O.; Armengaud, E. & Römer, K. The Quest for Infrastructures and Engineering Methods Enabling Highly Dynamic, Autonomous Systems European Conference on Software Process Improvement, 2019, 15-27

[Macher et al., 2020] Macher G., Schmittner C., Veledar O., Brenner E. (2020). ISO/SAE DIS 21434 Automotive Cybersecurity Standard-In a Nutshell. In 2020 International Conference on Computer Safety, Reliability, and Security, pages 123-135, Springer International.

[Messnarz et al. 2007] Richard Messnarz, Damjan Ekert, Assessment-based learning systems - learning from best projects, in Wiley Inerscience, Software Process Improvement in Practice, https://doi.org/10.1002/spip.347 , Volume12, Issue6, Special Issue: Special Issue on Industrial Experiences in SPI, November/December 2007, Pages 569-577

[Messnarz et al. 2009] Richard Messnarz, Hans-Leo Ross, Stephan Habel, Frank König, Abdelhadi Koundoussi, Jürgen Unterrreitmayer, Damjan Ekert, Integrated Automotive SPICE and safety assessments, Volume14, Issue5, Special Issue: Part 1: Special Issue on SPI Experiences and Innovation for Global Software Development, WILEY, September/October 2009, Pages 279-288, https://doi.org/10.1002/spip.429

[Messnarz et al. 2012] Messnarz R., König F., Bachmann V.O. (2012) Experiences with Trial Assessments Combining Automotive SPICE and Functional Safety Standards. In: Winkler D., O'Connor R.V., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2012. Communications in Computer and Information Science, vol 301. Springer, Berlin, Heidelberg

[Messnarz et al. 2016 2] Messnarz R., Kreiner C., Riel A., Integrating Automotive SPICE, Functional Safety, and Cybersecurity Concepts: A Cybersecurity Layer Model, Software Quality Professional . Sep2016, Vol. 18 Issue 4, p13-23. 11p., 2016

[Messnarz et al. 2016] Messnarz, R.; Kreiner, C. & Riel, A. "Integrating Automotive SPICE, Functional Safety, and Cybersecurity Concepts: A Cybersecurity Layer Model", Software Quality Professional, 2016.

[Messnarz et al. 2017] Messnarz R., Much A., Kreiner C., Biro M., Gorner J. (2017) Need for the Continuous Evolution of Systems Engineering Practices for Modern Vehicle Engineering. In: Stolfa J., Stolfa S., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2017. Communications in Computer and Information Science, vol 748. Springer, Cham. https://doi.org/10.1007/978-3-319-64218-5_36

[Messnarz et al. 2019 2] Messnarz R., Macher G., Stolfa J., Stolfa S. (2019) Highly Autonomous Vehicle (System) Design Patterns – Achieving Fail Operational and High Level of Safety and Security. In: Walker A., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2019. Communications in Computer and Information Science, vol 1060. Springer, Cham. https://doi.org/10.1007/978-3-030-28005-5_36

[Messnarz et al. 2019] Messnarz R., Ekert D., Zehetner T., Aschbacher L. (2019) Experiences with ASPICE 3.1 and the VDA Automotive SPICE Guidelines – Using Advanced Assessment Systems. In: Walker A., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2019. Communications in Computer and Information Science, vol 1060. Springer, Cham

[Messnarz et al. 2020] Messnarz R. et al. (2020) Automotive Cybersecurity Engineering Job Roles and Best Practices – Developed for the EU Blueprint Project DRIVES. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham. https://doi.org/10.1007/978-3-030-56441-4_37

[Mitre 2021 2] MITRE Attack Flow Navigator, https://attack.mitre.org/matrices/enterprise/, last visited 11.4.2021

[Mitre 2021] MITRE Corporation with associated federal funded research centers, https://www.mitre.org/, last visited 11.4.2021

[SAE 2016] SAE J3061, Cybersecurity Guidebook for Cyber-Physical Vehicle Systems, SAE - Society of Automotive Engineers, USA, Jan. 2016

[Schlager et al. 2018] Schlager C., Messnarz R., Sporer H., Riess A., Mayer R., Bernhardt S. (2018) Hardware SPICE Extension for Automotive SPICE 3.1. In: Larrucea X., Santamaria I., O'Connor R., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2018. Communications in Computer and Information Science, vol 896. Springer, Cham. https://doi.org/10.1007/978-3-319-97925-0_41

[Schmittner et al. 2019 2] Schmittner, C. & Macher, G. Automotive Cybersecurity Standards-Relation and Overview International Conference on Computer Safety, Reliability, and Security, 2019, 153-165

[Schmittner et al. 2019] Schmittner, C.; Christl, K.; Macher, G.; Knapitsch, J.; Parapatits, M.; Tauber, M.; Pichler, H. & Gnauer, C., Smart industrial indoor farming-Technical and societal challenges, IDIMT 2019: Innovation and Transformation in a Digital World-27th Interdisciplinary Information Management Talks, Trauner Verlag Universitat, 2019, 401-409

[Schmittner et al., 2020] Schmittner C., Chlup S., Fellner A., Macher G., Brenner E. (2019). ThreatGet: Threat modeling based approach for automated and connected vehicle systems. In 11th GMM-Symposium AmE 2020-Automotive meets Electronics, VDE.

[SOQRATES 2021] SOQRATES, Task Forces Developing Integration of Automotive SPICE, ISO 26262 and SAE J3061, http://soqrates.eurospi.net/, last visited 10 April 2021

[Stolfa et al. 2020 2] Jakub Stolfa, Svatopluk Stolfa, Christian Baio, Utimia Madaleno, Petr Dolejsi, Federico Brugnoli, Richard Messnarz, DRIVES—EU blueprint project for the automotive sector—A literature review of drivers of change in automotive industry, in: Journal of Software: Evolution and Process, Volume32, Issue3, Special Issue: Addressing Evolving Requirements Faced by the Software Industry, March 2020

[Stolfa et al. 2020] Jakub Stolfa, Svatopluk Stolfa, Richard Messnarz, Omar Veledar, Damjan Ekert, Georg Macher, Utimia Madaleno (2020) Automotive Engineering Skills and Job Roles of the Future?. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham. https://doi.org/10.1007/978-3-030-56441-4_26

[VDA 2020] Automotive Cybersecurity Management System Audit Guideline, 1st Edition, VDA-QMC, December 2020

[Vector 2021] Vector, NTS - Network Test Suites for Vehicles, https://www.vector.com/bo/en/news/news/technical-article-with-daimler-network-tests-for-everyone/

[Veledar et al. 2019] Veledar, O.; Damjanovic-Behrendt, V. & Macher, G., Digital Twins for Dependability Improvement of Autonomous Driving, European Conference on Software Process Improvement, 2019, 415-426

[Wegner et al. 2020] Thomas Wegner, Richard Messnarz, Damjan Ekert, Bernhardt Steger, Ralf Mayer, Rainer Dreves, Bernhard Sechser, Christian Schlager, Christoph Karner (2020) Enough Assessment Guidance, It's Time for Improvement – A Proposal for Extending the VDA Guidelines. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham. https://doi.org/10.1007/978-3-030-56441-4_34