



A Framework for Dynamic Trustworthiness Assessment in Cooperative and Automated Vehicles

5GAA Automotive Association
White Paper


CONTACT INFORMATION:
Executive Manager – Thomas Linget
Email: liaison@5gaa.org

MAILING ADDRESS:
5GAA c/o MCI Munich
Neumarkter Str. 21
81673 München, Germany
www.5gaa.org

Copyright © 2025 5GAA. All Rights Reserved.

No part may be reproduced except as authorised by written permission. The copyright and the foregoing restriction extend to reproduction in all media.

VERSION:	1.0
DATE OF PUBLICATION:	1 September 2025
DOCUMENT TYPE:	White Paper
EXTERNAL PUBLICATION:	Yes
DATE OF APPROVAL BY 5GAA BOARD:	29 May 2025



Connected and Automated Vehicles are expected to make critical decisions based on data from external sources, yet current approaches to security and safety do not address a fundamental question: How much can the content of this data be trusted? This white paper tackles this largely unsolved problem by introducing a dynamic trust assessment framework centred on Actual Trustworthiness Level and Required Trustworthiness Level – metrics that quantify the residual uncertainty about whether external data or system behavior can be trusted based on available evidence and how much trust is needed to meet the operational and safety requirements of the current context. This methodology is intentionally designed to be generic, allowing for different implementations that adapt to evolving operational and organizational contexts. A detailed Automated Emergency Braking use case illustrates how the methodology can be instantiated in a real-world scenario. The paper concludes by identifying open standardization gaps and proposing directions for future harmonization of trust assessment in cooperative and connected mobility ecosystems.

Contents

	Executive Summary	6
	Key Concepts and Methodology	7
	Application Example: AEB Use Case	7
	Open Gaps and Standardization Outlook	8
1	Introduction	9
2	Automated Emergency Brake Use Case	11
	Item Definition	11
	In-vehicle Components	14
	Sequence Diagram	16
	Trustworthiness Properties Evaluated	16
	Trust Sources	17
3	Methodology for ATL Calculation	19
	Preliminaries	19
	Overview of ATL Methodology	23
	Expression Generation	24
	Atomic Propositions	24
	Complex Propositions	24
	Trust Expressions	24
	Expression Evaluation	29
	Evidence Collection	29
	Atomic Opinion Calculation	30
	Evaluate the Expression	31
	Trust Evolution and Feedback	31
	Re-evaluation of Trust Opinions	32
	Re-evaluation of Trust Expressions	32
4	Methodology for RTL Calculation	33
	Background	33
	Threat Analysis and Risk Assessment (TARA)	33
	Hazard Analysis and Risk Assessment (HARA)	34
	Required Trustworthiness Level	35
	Belief Threshold Calculation	36
	Disbelief threshold calculation	37
	Uncertainty Threshold Calculation	38
5	Example Application on Use Case	40
	ATL in Automated Emergency Braking	40
	Trust Expression for the AEB Use Case	40
	Collecting Evidence from Trust Sources and Computing Atomic Trust Opinions	41
	Evaluate the Expression	43
	RTL in Automated Braking Event	44
	Belief Threshold Calculation Method	46
	Disbelief Threshold Calculation Method	50
	Uncertainty Threshold Calculation Method	54
	Possible Behaviors of RxV with ATL and RTL	58
6	Open Topics and Gap Analysis	61
	Open Technical Questions	61
	Specifying the ATL Expression	61

Quantifying Uncertainty in ATL.....	61
Comparison of RTL and ATL.....	62
Federation of Trust Assessment.....	63
Standardization Gap Analysis	65
ISO	65
ETSI	65
3GPP.....	66
Identified Gaps	67
Standardized Procedures	67
Standardized Profiles	68
References.....	70
Annex A: TARA Report of the Automated Emergency Brake Use Case	72
System Diagram of SYS: System	72
Assets and Damage Scenarios	73
Data.....	73
Components	73
Damage Scenarios Overview.....	75
Damage- and Threat Scenarios Table	76
Threat Scenarios and Attack Paths	77
Assumptions Table	78
Attack Steps Tables (Accumulated).....	78
Controls Table (Accumulated)	79
Risks Table	79
Control Scenarios per Risk.....	80
Data Table.....	80
Components Table.....	81
Channels Table.....	88
Data Flows Table.....	88



Executive Summary

Connected and Automated Vehicles (CAVs) operate in increasingly complex, multi-agent environments where decision-making depends on data from external entities including other vehicles, infrastructure nodes, and edge computing services. This evolving landscape of connected mobility presents fundamental challenges for ensuring safe and reliable cooperation among autonomous agents.

A critical gap exists in current approaches; while cryptographic mechanisms ensure secure communication through authentication and integrity verification, they cannot guarantee the trustworthiness of message content. A cryptographically signed message may still contain incorrect, inaccurate, or adversarial information, creating potential safety risks when vehicles make automated decisions based on external data. Therefore, vehicles must not only verify data authenticity but also systematically evaluate whether the content can be relied upon for safety-critical functions.

This white paper presents a comprehensive, modular methodology for dynamic trustworthiness assessment in cooperative systems, centered on two complementary metrics: Actual Trustworthiness Level (ATL) and Required Trustworthiness Level (RTL). The framework provides a structured approach to quantify trustworthiness in relation to specific properties, enabling systems to make informed decisions based on available confidence levels.

Building upon previous work in 5GAA¹ and leveraging insights from the Horizon Europe project CONNECT², this methodology recognizes that in complex, multi-agent

¹ <https://5gaa.org/creating-trust-in-connected-and-automated-vehicles/>

² <https://horizon-connect.eu/>

environments, trust cannot be assumed, but it must be continuously re-evaluated and verified. Rather than relying solely on secure Vehicle-to-Everything (V2X) communication for trust, the framework introduces a structured approach to assess the residual uncertainty surrounding data and entities in real time. This dynamic approach supports decision-making in scenarios where autonomous systems collaborate despite limited prior knowledge or direct control. Crucially, the methodology identifies critical needs for standardized procedures and profiles to ensure interoperability and consistent application of the methodology across the industry.

Key Concepts and Methodology

The framework introduces several foundational concepts that enable systematic trustworthiness evaluation. Trustworthiness properties define multi-dimensional characteristics such as integrity, availability, and accuracy that determine whether data and system behavior can be confidently relied upon in specific contexts. Trust objects represent the entities, data sources, or components being evaluated, while trust sources provide evidence for assessment.

It also introduces two key constructs and the way to calculate them: The ATL, which quantifies the degree of uncertainty around the trustworthiness of incoming data based on current evidence; and the RTL, which represents the acceptable level of uncertainty for a given task or operational scenario. By comparing these levels, the methodology enables dynamic, real-time assessments of whether data meet the security and safety needs of automated and connected systems.

The methodology does not prescribe a single, fixed implementation. Instead, it offers a generic approach that can integrate multiple trust sources, diverse trustworthiness properties, and domain-specific requirements. While subjective logic provides one robust method for handling uncertainty within the trustworthiness assessment, the framework's structure is deliberately agnostic to any single trust model. This ensures that the methodology can be instantiated in ways that suit different OEM or supplier strategies, while still supporting future harmonization efforts and standardization activities within 5GAA and other relevant bodies.

Application Example: AEB Use Case

The methodology is demonstrated through a comprehensive Automated Emergency Braking (AEB) use case, where a receiving vehicle must decide whether to execute emergency braking based on V2X messages from another vehicle. This application illustrates the transition from driver warnings to automated safety responses, highlighting the critical importance of trustworthiness assessment in safety-critical cooperative functions. The use case showcases how the framework integrates multiple evidence sources to support real-time decision-making. Through practical examples, the application demonstrates how ATL calculations and RTL thresholds work together to enable context-aware trust decisions that can adapt vehicle behavior based on trustworthiness levels.

Open Gaps and Standardization Outlook

The white paper identifies significant technical and standardization challenges that require industry collaboration. Key open technical questions include systematic approaches for specifying ATL expressions, quantifying uncertainty across multiple sources, and enabling federated trust assessment across organizational domains.

Critical standardization gaps require coordinated efforts to develop standardized procedures for evidence quantification, trust expression evaluation, and trust model templates for common CAV use cases. The framework identifies immediate priorities including trust model profiles for specific automotive applications and long-term requirements for mechanisms governing dynamic trust evolution.

This work aligns with 5GAA's mission to foster interoperable and safe cooperative systems by providing a foundation for evidence-based trust reasoning that can inform future standardization efforts. The methodology represents a critical building block for enabling more sophisticated cooperative driving functions while maintaining safety and security standards in the evolving landscape of connected and automated mobility.

1 Introduction

Modern Connected and Automated Vehicles (CAVs) operate in open, dynamic, and highly distributed environments, where decision-making increasingly depends on data from entities outside their direct control, namely other vehicles, infrastructure nodes, or edge computing services. In such settings, trust cannot be assumed; it must be continuously assessed. A common misconception is that once communication is secured, e.g., through cryptographic mechanisms, such as Public Key Infrastructure (PKI), trust is guaranteed. But while secure communication ensures authenticity and integrity, it does not answer the more fundamental question: Can the content of the message be relied upon? A message may be validly signed, yet contain incorrect, inaccurate, or adversarial information. Therefore, vehicles must not only verify that data comes from a known source, but also reason about its trustworthiness in context.

Trustworthiness assessment introduces this missing layer: A structured, evidence-based approach to evaluate whether a node or data source is likely to behave as expected under specific conditions. It is not a replacement for safety or security, but a complementary function – essential for resilient cooperation, meaningful assurance, and reliable decision-making in complex, multi-agent ecosystems. This white paper addresses this gap by presenting a modular framework for dynamic trust assessment, rooted in the CONNECT project's work and aligned with emerging approaches for trustworthy Artificial Intelligence (AI) and data-driven mobility.

This brings us to a broader and still largely unresolved challenge: How to assess trustworthiness in cooperative, multi-agent systems. While safety and security have well-established methodologies and standards, trustworthiness remains under-defined and inconsistently addressed. In reality, trustworthiness is a complex, multi-dimensional property that reflects how well a system or data source aligns with expectations over time, across contexts, and under uncertainty. It involves reasoning not just about who produced the information, but also whether it meets the level of confidence required to make a decision or take an action for a specific task in a specific context. What is missing is not just a technical component, but a conceptual and architectural foundation for dynamic, context-aware, and evidence-driven trust assessment.

This white paper contributes to that foundation by proposing a modular framework that treats trustworthiness as an important property of cooperative systems. It provides a structured approach not only to reason about trust, but to quantify it in relation to specific operational goals, enabling systems to take informed actions based on the level of confidence available. While the methodology is applied here to the CAV domain, it is designed to be extensible and adaptable to other contexts where trustworthy cooperation among autonomous agents is required.

This methodology centers around the concepts of Actual Trustworthiness Level (ATL) and Required Trustworthiness Level (RTL), which together enable informed and evidence-based trust decisions in real-time operational settings. ATL is a runtime metric that quantifies the current level of trustworthiness of a given component, system, or data stream based on evidence collected from relevant trust sources. RTL, by contrast, defines the minimum trustworthiness threshold required for safe and

secure system operation. By comparing ATL with RTL, an autonomous vehicle can dynamically determine whether it has sufficient trust to act on a given input or initiate a safety-critical maneuver.

To demonstrate how the methodology can be instantiated, the report applies it to a representative AEB use case, where a Receiving Vehicle must decide whether to perform an emergency braking maneuver upon receiving V2X messages (DENMs³ and CAMs⁴) from a transmitting vehicle. The trustworthiness of these messages is assessed through ATL computation. In parallel, RTL thresholds are derived using established risk analysis methodologies, such as Threat Analysis and Risk Assessment (TARA) and Hazard Analysis and Risk Assessment (HARA), allowing for context-specific trustworthiness requirements.

In addition to its methodological contributions, this work is situated within a wider landscape of ongoing efforts to define, standardize, and operationalize trustworthiness in cooperative, intelligent systems. The approach outlined in this white paper responds to emerging expectations from both research and regulatory communities for structured, interpretable, and evidence-based trust reasoning, offering a foundation that can inform future standardization and cross-domain adoption.

³ DENM: Decentralized Environmental Notification Message

⁴ CAM: Cooperative Awareness Message

2 Automated Emergency Brake Use Case

This section presents a comprehensive examination of the AEB use case, which we will use in this document as a concrete illustration to explore how trustworthiness can be quantified, evaluated, and acted upon in dynamic, safety-critical scenarios. Here we build directly upon the Emergency Brake Warning scenario analyzed in the previous 5GAA STiCAD I white paper [13], which focused on functional and safety aspects of V2V-based emergency braking alerts. While STiCAD I highlighted how cooperative warning messages support driver and system interventions, this document extends the discussion by introducing trustworthiness assessment and shifts the focus from functional performance alone to evaluating whether data from external sources can be trusted as a reliable basis for actually making an automated braking decision. This use case will be demonstrated later in the paper as an example of how the ATL and RTL methodologies can be instantiated.

The use case description in this section focuses on the aspects necessary to develop and showcase a methodology for deriving required levels of trust and to assess the actual trustworthiness at runtime. It is not the purpose of this item description to specify a use case implementation for Automated Emergency Brake. Additionally, this document does not specify an implementation how to facilitate information exchange to perform trustworthiness assessments.

Item Definition

The AEB function is one of the main Advanced Driver Assistance System (ADAS) functionalities designed to detect slow or stopped vehicles and pedestrians ahead, triggering the brakes immediately to prevent accidents or minimize injuries. This system plays a vital role in enhancing road safety by protecting both passengers and other traffic participants. The UN ECE Regulation 131 defines such a system as “a system which can automatically detect an imminent forward collision and activate the vehicle braking system to decelerate the vehicle with the purpose of avoiding or mitigating a collision” [1].

The working process of the AEB system can be divided into the following stages: (1) Early warning stage: Once an impending collision is detected, the AEB system will alert the driver immediately through visual or audial warning sign, or by tightening the safety belt; (2) Braking stage: When the collision becomes imminent, the AEB system uses a single-stage or multistage braking strategy (i.e., directly applies the maximum braking pressure or gradually increases the braking pressure) to avoid the collision. So, AEB differs from forward collision warning in that the latter only alerts/warns the driver but does not by itself brake the vehicle. The AEB system is a relatively new concept that Yang et al. have been able to systematically review from the perspective of impact factors, system structure, and evaluation [2].

The introduction of V2V communications enables use cases that can be applied in

combination with AEB, such as Cooperative Adaptive Cruise Control (CACC) [3] and Emergency Electronic Brake Lights (EEBL) [4]. With EEBL, a vehicle broadcasts a DENM when its deceleration reaches a predefined emergency braking threshold. Receiving these DENMs triggers automatic emergency braking in other vehicles. The effectiveness of EEBL depends on the reliability of the V2V communication channel; higher packet error rates necessitate more DENM repetitions to adequately inform other vehicles of the critical situation [5][6].

The trigger of the DENM is defined by Car2Car safety profile. A DENM message should be followed by an observation of the CAMs over a period of time, which this refines our observation and confirm that there is a deceleration and quantifies it. The DENM is triggered based on a threshold, and it is repeated every 100 milliseconds as long as the trigger holds, but the exact quantification of the deceleration is defined by CAM.

There are different implementation variations on how many CAMs are needed, but in general detecting a DENM alone is not enough to trigger an emergency braking that stops the car, and this decision is taken by considering the combination of DENMs and CAMs, and other factors (e.g., sensor fusion quality, confirmation system, etc.). In this paper, we acknowledge that there is a gap created by different implementation variations, which needs to be addressed in the future, but for the continuation of this work we establish the assumption that the receiver is in possession of a 'modality of messages' (CAMs and DENMs) and within the deceleration scope, and we further assume that this deceleration scope exists for a sequence of messages.

In general, these use cases represent scenarios where information about an emergency braking event is exchanged directly between two vehicles (Transmitting Vehicle – TxV and Receiving Vehicle – RxV) through V2V communication. In this fully automated scenario, the RxV, upon receiving DENMs and CAMs, independently decides whether to apply the brakes without any human intervention. A high-level description of the item is shown in Figure 1. More detailed descriptions will be given in the next sections. Note that it is assumed that the communications between TxV and RxV are direct and use the PC5 interface. For this use case it is assumed that the network is not involved, and that there is no scheduling of access to the PC5 connection by the cellular network.

The item described here is based on the Emergency Brake Warning (EEBL) use case, as defined in 5GAA Use Case Implementation Description (UICD) and STiCAD I [13], with the following modifications to serve the purpose of this work item:

- ▶ Change of the automated decision-making of the RxV from only warning the driver to automated braking in order to showcase the necessity of trustworthiness considerations with direct safety impact.
- ▶ Adding a high-level technical implementation to enable the assessment of cybersecurity risks.
- ▶ Adding trust properties and as input for deriving the RTL.
- ▶ Adding trust sources which provide runtime input for assessing the ATL.

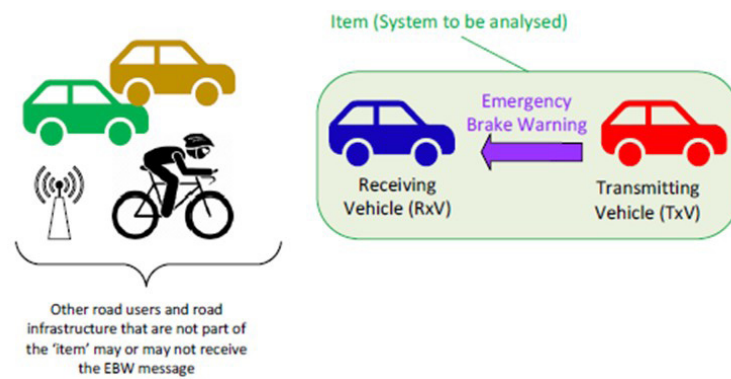


Figure 1 Item definition for Automated Emergency Braking (simple view)

Here, the functional behavior of AEB implies:

1. Detection and transmission by TxV:
 - Event detection: TxV detects an emergency braking event through its onboard sensors, such as rapid deceleration beyond a certain threshold or activation of the vehicle's emergency braking system.
 - Message transmission: TxV transmits an DENMs and CAMs to nearby vehicles using V2V communication protocols.
2. Reception and processing by RxV:
 - Message reception: RxV receives the DENMs and CAMs transmitted by TxV.
 - Message verification: RxV verifies the authenticity and integrity of the DENMs and CAMs using cryptographic checks to ensure it has not been tampered with.
3. Decision-making by RxV:
 - Situation assessment: RxV evaluates the received messages in the context of its current environment and operational state. This includes assessing the proximity, speed, and direction of TxV relative to RxV. It also takes under consideration its own sensor data input (e.g., LIDAR, radar, camera systems). To support the situation assessment, each data item's ATL is calculated and compared with the corresponding RTL.
 - Braking decision: Based on the situation assessment, RxV automatically decides whether to initiate an early warning of the driver or, if necessary, an emergency braking maneuver.
4. Maneuver execution: If the decision to brake is made, RxV automatically applies the brakes with an appropriate force to avoid a collision.

In-vehicle Components

The in-vehicle architecture is based on a so-called Zonal E/E Architecture. In such an architecture, the actual function is assumed to be executed on high-performance vehicle computer Electronic Control Units (ECU), with sensor ECUs providing input and actor ECUs receiving functional commands to control the vehicle's driving behavior. Sensors and actors are connected in regional zones in the vehicle. As shown in Figure 2, each zone is organized by a Zonal Controller ECU, which also functions as a gateway between the sensors/actors and the vehicle computers. Cellular V2X communication is handled via a C-V2X ECU which is directly connected to the ADAS ECU.

The architecture shown in Figure 2 forms the backbone of the AEB use case, where each component plays a vital role in ensuring a safe and effective braking response. Below, we provide a description of the main functional elements and their responsibilities within this zonal architecture.

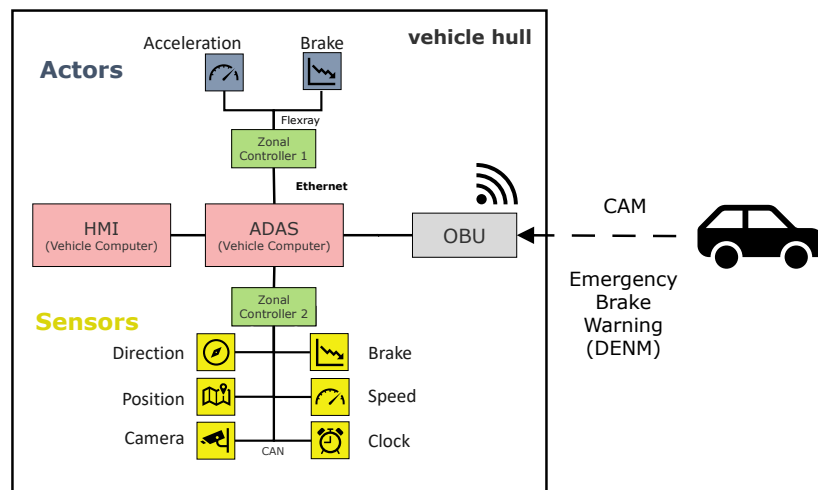


Figure 2 E/E architecture of Automated Emergency Brake (detailed view of technical in-vehicle architecture of the receiving vehicle)

1. ADAS: The core component responsible for processing all incoming data and making decisions. It integrates various sensor inputs and communicates with other vehicle components.
2. Actors: These include:
 - Brake ECU: Executes braking commands.
 - Acceleration ECU: Controls vehicle speed adjustments.
 - Steering ECU: Manages steering actions.
3. Sensors: Provide real-time data about the vehicle's surroundings, including:
 - Direction: Global Positioning System.
 - Brake: Transmission and stability control status sensors.

- Speed: Global Positioning System.
 - Clock: Timestamp.
 - Radar: Provides data on objects' distance and speed.
 - Cameras: Capture visual information.
 - Position: Provides precise location data.
4. OBU: Facilitates communication with other vehicles and infrastructure, acting as an interface for V2X communication.
 5. Zonal controllers: Intermediate controllers that manage communication between sensors, actuators, and the ADAS.

The interplay of these components, ranging from sensors and zonal controllers to the ADAS and C-V2X ECUs, forms a complex, dynamic system where data from various sources is integrated to enable automated emergency braking. To visualize these interactions and illustrate how external data sources like V2X messages are incorporated into the in-vehicle decision-making process, Figure 3 provides a mapping of the Emergency Brake Warning functional model of STiCAD I [13] to the technical in-vehicle architecture used in the Automated Emergency Brake use case. This mapping highlights the key information flows and control paths within the receiving vehicle, emphasizing the role of trustworthiness assessment in evaluating the reliability and relevance of each data source in real time.

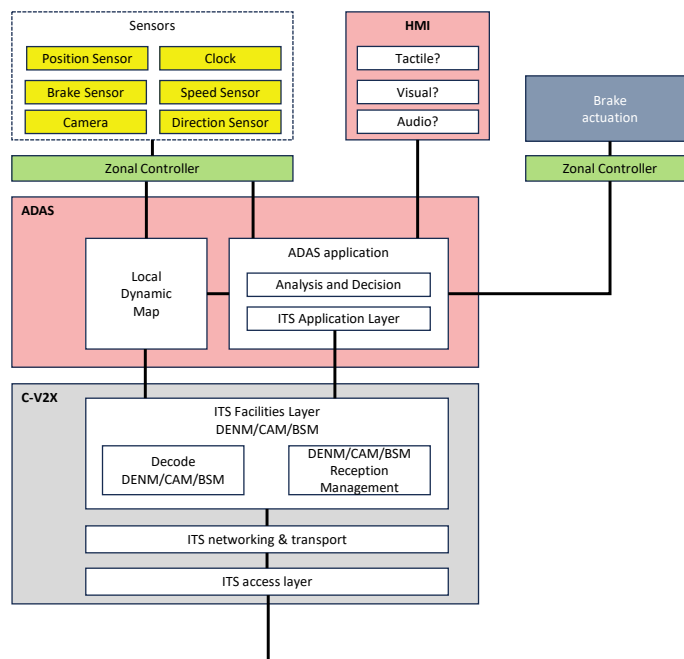


Figure 3 Mapping of Emergency Brake Warning functional model of STiCAD I [13] to the technical in-vehicle architecture of the receiving vehicle of Automated Emergency Brake

Sequence Diagram

In this use case, the following data items are received by the ADAS ECU:

- ▶ In-vehicle sensor data
 - Direction: Vehicle's direction (GPS data).
 - Position: Vehicle's position (GPS data).
 - Speed: GPS, wheel revolution counter, clock.
 - Camera: Video feed showing the road in front of the vehicle.
 - Braking: Transmission and stability control status sensors.
 - Clock: Timestamp.
- ▶ V2X data
 - Emergency Brake Warning message: DENM transmitted from the leading vehicle TxV.
 - CAM messages transmitted from the leading vehicle TxV before and during the emergency brake event.

To visualize how the above data items are used within the broader system, the sequence diagram of Figure 4 provides a step-by-step view of the data flow and decision logic. It illustrates the timing and interactions between the leading vehicle, the receiving vehicle's in-vehicle architecture, and the data sources.

Trustworthiness Properties Evaluated

Trustworthiness properties describe the multi-dimensional characteristics of trust that determine whether data and system behavior can be confidently relied upon in a specific operational context [7][14]. For instance, properties like the integrity of sensor data or messages directly influence the reliability of emergency braking interventions. Others, such as the operational completeness of a dataset or the behavioral consistency of a data source, shape how much a receiving vehicle can rely on external messages in rapidly evolving traffic situations. More specifically, examples of the trustworthiness properties for this use case could be the following.

Security:

- ▶ Integrity of the sensor data (direction, position, speed, camera, braking, clock)
- ▶ Integrity of the DENM

Safety [14]:

- ▶ ODD of TxV sensors
- ▶ Completeness of TxV data

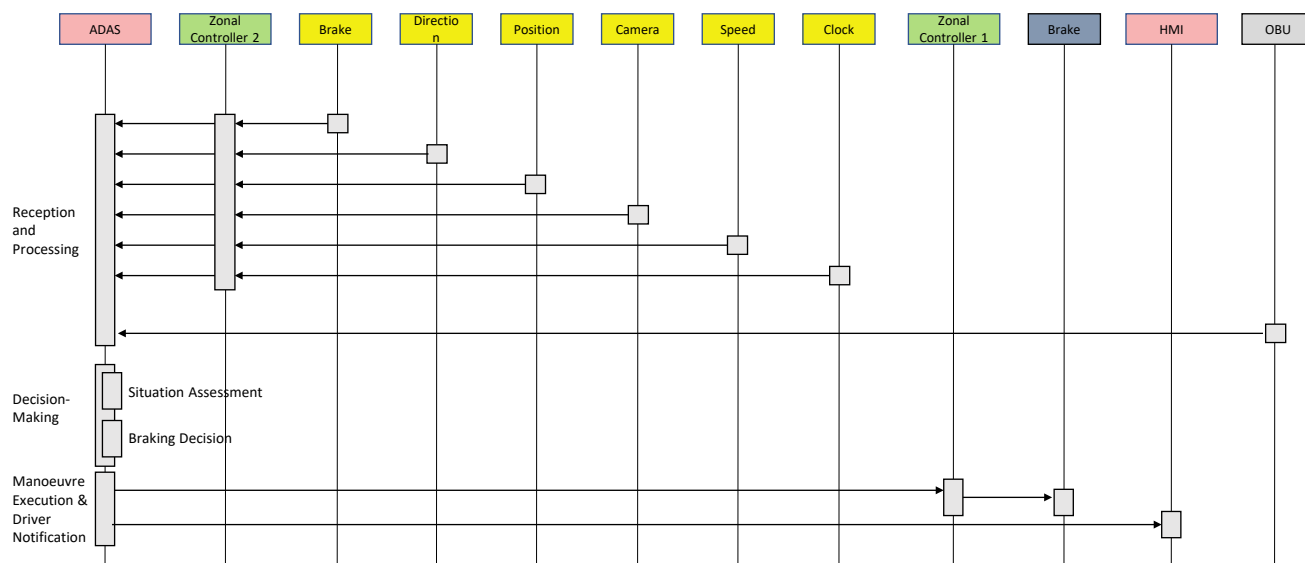


Figure 4 Sequence diagram of messages exchanged in Automated Emergency Brake

Trust Sources

Building on the definitions and categories of trust sources established in the 5GAA Trust4Auto white paper [7], this section identifies the trust sources most relevant to the Automated Emergency Brake use case. Trust sources are essential because they provide evidence that a system can draw upon to evaluate the trustworthiness of data and behaviors. The following list outlines these trust sources within the use case architecture, setting the stage for how they support real-time, context-aware trust reasoning. It is assumed that both vehicles (TxV and RxV) have the same trust sources.

Security-related trust sources may provide evidence on the following security controls across the in-vehicle ecosystem:

- ▶ Vehicle computer
 - Access control
 - Secure communication with zonal controller (e.g., MACsec)
 - Firewall/packet filtering
 - Secure boot
- ▶ Zonal controller
 - Firewall/packet filtering
 - Secure boot
 - Network-based Intrusion Detection (NID) in respective zonal network
- ▶ Sensors (brake, direction, position, camera, speed, clock)
 - Access control
- ▶ Actors (brake ECU)

- Access control
- Secure communication with zonal controller (e.g., SecOC)
- ▶ OBU
 - Firewall/Packet filtering
 - Secure boot

Safety-related trust sources may provide evidence on the following safety controls across the in-vehicle ecosystem:

- ▶ Sensor capabilities of TxV
- ▶ ASIL Level of TxV parts included in generation of DENM

3 Methodology for ATL Calculation

The concept of the Actual Trustworthiness Level emerges from the recognition that trust is not a static property, but rather something that must be continuously assessed against specific tasks and contexts. ATL represents a quantitative measure of how much a data source or system can be considered trustworthy in real time, reflecting the aggregation of evidence and the application of logical reasoning over trust propositions. Unlike traditional security or safety assurances, ATL explicitly incorporates the dynamic and multi-dimensional nature of trustworthiness, accounting for uncertainties and real-world complexity. In this section, we present a step-by-step methodology for calculating the ATL, building upon the principles and structures laid out in the Trust4Auto white paper [7].

The methodology for calculating the ATL, as outlined in this section, is designed to be a generic and extensible framework. Unlike rigid or scenario-specific approaches, this methodology does not prescribe a single way of modeling or quantifying trustworthiness. Instead, it provides a structured, modular approach that can be instantiated according to the unique requirements of each application domain, data source, and operational context. That being said, the framework leverages subjective logic as a mathematical foundation for modeling trust under uncertainty. This section therefore presents the key phases of the ATL calculation process (expression generation, expression evaluation, and trust evolution) demonstrating how evidence from diverse trust sources can be incorporated in a coherent, logically sound manner without limiting the implementation to a specific set of rules or sources.

The methodology presented in this chapter directly draws upon and extends the foundational work described in CONNECT Deliverables D3.1 [16] and D3.2 [17]. Specifically, it leverages the comprehensive architectural and conceptual framework laid out for the Trust Assessment Framework (TAF), which incorporates modular trust model instantiation and the quantification of trust opinions using Subjective Logic. In D3.2, detailed methodologies for calculating ATLs are outlined, including the interplay of trust sources and evidence-based reasoning that underpins trust assessments in dynamic, multi-agent environments such as cooperative and automated mobility scenarios. However, the methodology here is intentionally more generic and modular, designed to support different instantiations of expression generation and trust model representations. This ensures that the framework can adapt to different system architectures, deployment scenarios, but also future developments.

Preliminaries

As defined in [7], a trust proposition is a statement or assertion about something that we need to evaluate for trustworthiness. It represents a specific aspect of an entity, action, or data that we want to either trust or distrust based on available evidence. Trust propositions are the foundation of the ATL methodology because they define what we need to evaluate the trustworthiness of.

Examples of trust propositions:

- ▶ "Vehicle A's position data is accurate."
- ▶ "The received DENM has not been tampered with."
- ▶ "Vehicle B's braking system responds in time."

In complex, real-world systems like V2X communication and CCAM, we often have to make decisions based on incomplete, conflicting, or uncertain information. Traditional probabilistic models assume that we have complete knowledge of all possibilities and their respective probabilities, but in dynamic, distributed environments, this is rarely the case. For instance, a vehicle might receive conflicting data from different sensors or communication channels. Also, not all data sources are equally reliable, and some may be compromised or faulty.

In such environments, using an evidence-based theory becomes essential because it allows us to explicitly represent uncertainty: Rather than forcing a decision based on incomplete data, we can account for the fact that we do not have enough information, thus reflecting uncertainty in our trust assessments. Then, instead of discarding data that does not fully support or contradict a trust proposition, we can blend the evidence – representing both belief and disbelief while leaving room for uncertainty. This also allows us to adjust trust levels dynamically as more data becomes available.

An **opinion** ω [19] in evidence-based theory (e.g., in subjective logic) is a formal way of representing trust in a proposition by combining:

- ▶ Belief (b): The degree to which the evidence supports the proposition.
- ▶ Disbelief (d): The degree to which the evidence contradicts the proposition.
- ▶ Uncertainty (u): The degree to which the evidence is insufficient to make a clear judgment.

This triplet (b, d, u) is especially useful in environments where we often have incomplete or conflicting evidence, such as when data is missing, delayed, or of varying quality. The sum of belief, disbelief, and uncertainty always equals 1: $b + d + u = 1$

In multi-agent systems like CAVs, vehicles and infrastructure do not always gather evidence directly. Often, trustworthiness is assessed indirectly through a chain of agents that pass along their own trust assessments or evidence. That means, agents might have to rely on referrals from other trusted agents to form their opinions. Each agent provides an opinion, which is then discounted based on the trustworthiness of the referring agent. This allows for the propagation of trust assessments through a network of agents [20].

More formally, **trust discounting** is a form of trust transitivity where an agent A takes into account the trust level of a source S in order to deduct (e.g., weight) the trust level reported by S about evidence E. Trust discounting typically reduces the trust level that A derives for E via S proportionally to the trust level of S. The trust-discounted opinion ω typically gets increased uncertainty mass, compared to the original opinion advised by S.

In environments where evidence is uncertain or incomplete, combining opinions from multiple agents helps improve the trust assessment by gathering multiple perspectives. If multiple vehicles or Roadside Units (RSUs) provide trust opinions on the same

proposition (e.g., whether a sensor's data is accurate), the trust evaluation of an agent can use fusion operators to combine these opinions into a single, consolidated trust score [21].

More formally, **trust fusion** is a form of melded belief involving the analyst's trust in the sources where the derived opinions resulting from separate trust paths are fused into one. Trust fusion is the result of calculating the composition of independent opinions that an analyst received from two or more independent sources about the same observation.

Thus, for multi-agent, safety-critical dynamic environments, such as CCAM, we need a formalism that can capture several aspects:

- ▶ Express uncertainty: Due to the lack of sufficient evidence, we are often unable to estimate probabilities with confidence.
- ▶ Express opinions: Whenever the truth of a proposition is assessed, it is always done by an agent, and it cannot be considered to represent a general and objective belief.
- ▶ Transitive trust: Allow for trust evaluation along referral chains, where opinions from other agents can be incorporated into trust assessments by discounting them according to the confidence placed in them.
- ▶ Trust Fusion: Allow for the fusion of evidence from different sources into one.

Subjective logic fits our needs better than any other decision logic, and we thus adopt it for our methodology. Here, we present a brief comparison of decision logic frameworks in order to demonstrate more clearly the pros and cons of each option:

- ▶ Probabilistic logic: Extends binary decision-making by allowing probabilistic truth values. However, it does not explicitly account for missing or conflicting evidence, and it lacks mechanisms for subjective belief modeling or dynamic evidence fusion.
- ▶ Fuzzy logic: Improves on uncertainty modeling by introducing degrees of truth rather than crisp values. Nevertheless, it does not inherently support probabilistic reasoning or fusion of conflicting evidence sources, which are crucial in dynamic, multi-agent systems.
- ▶ Bayesian methods: Model all sources of uncertainty using probability distributions, allowing for coherent updates as new data become available via Bayes' theorem. They can also accommodate both subjective beliefs and objective data-driven inputs, enabling flexible modeling of expert knowledge alongside empirical evidence. However, this type of probabilistic logic does not allow seamless modeling of situations where different agents express their beliefs about the same proposition, so it struggles with subjective belief representation and does not naturally resolve conflicting information from multiple sources.

Dempster-Shafer Theory (DST): Offers more flexibility by allowing explicit modeling of ignorance and combining evidence from different sources. However, some studies have shown that the order in which conflicting sources are aggregated can influence

the outcome, potentially leading to inconsistent results. Furthermore, DST can produce counterintuitive results when merging highly conflicting evidence, sometimes leading to undefined or misleading beliefs. Another limitation is that classic DST does not inherently capture trust transitivity, an essential requirement in systems where trust relationships are often indirect, such as connected and automated vehicles.

In essence, subjective logic extends DST by formally introducing subjective beliefs and providing a rich set of operators for merging conflicting opinions, modeling trust transitivity, and managing uncertainty. Subjective logic is particularly well-suited for trust evaluation scenarios where evidence is partial, uncertain, and possibly conflicting, as is the case in multi-agent, decentralized environments.

The table below provides a comparative overview of these decision logics, highlighting their respective strengths and limitations across several key dimensions. This comparison helps to clarify why subjective logic is particularly well-suited as the basis for our trustworthiness assessment methodology, as it combines support for uncertainty, subjective beliefs, and the ability to reason about trust transitivity in a coherent, mathematically rigorous framework.

Table 1 Comparative overview of different decision logics

Decision logics	Dealing with uncertainties	Probabilistic truth values	Incorporating past evidence	Subjective beliefs	Merging conflicting sources	Trust transitivity
Binary logic	✗	✗	✗	✗	✗	✗
Probabilistic logic	✓	✓	✗	✗	✗	✗
Fuzzy logic	✓	✗	✗	✗	✗	✗
Bayesian probability	✓	✓	✓	✗	✗	✗
Dempster-Shafer Theory	✓	✓	✓	✓	✗	✗
Subjective logic	✓	✓	✓	✓	✓	✓

In the rest of this document, we follow the multi-agent approach, in order to show a more complete methodology of ATL calculation. However, it can also be applied in a single-agent context, such as when a vehicle directly collects evidence from its own sensors. In that case, the trust assessment falls back to direct trust models. The complexity of referral chains and multi-agent fusion is not necessary in this case, but the belief-disbelief-uncertainty model still holds value, especially when evidence is incomplete or contradictory. In single-agent scenarios, opinions are directly formed from the agent's own evidence without needing to discount or fuse information from external sources, making the trust evaluation more straightforward.

Overview of ATL Methodology

The ATL methodology is fundamentally about calculating trust expressions. It provides a systematic process for evaluating the trustworthiness of entities, data, or actions based on trust expressions, which are constructed from one or more trust propositions. Figure 5 illustrates the core components of the ATL methodology, highlighting the sequence of phases that lead to the evaluation and evolution of trust assessments.

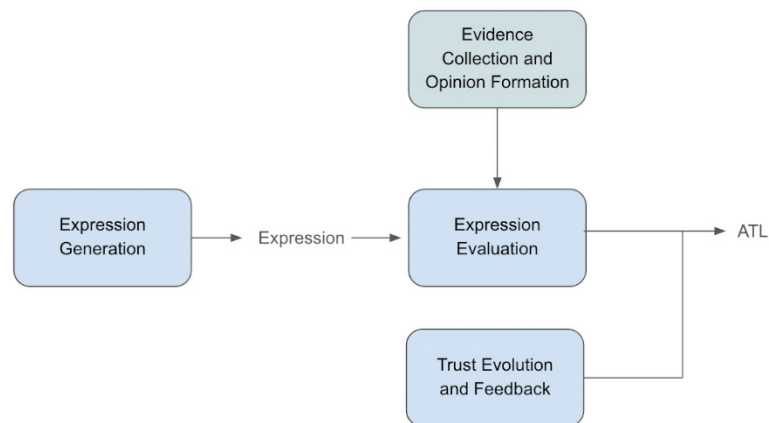


Figure 5 Overview of ATL methodology

The process begins with **expression generation**, where trust expressions are created based on predefined trust propositions (such as “Vehicle A’s position data is accurate”). These expressions represent how trust is structured in the system, and they combine different trust propositions into logical forms that can later be evaluated.

Then the next phase is the **expression evaluation** phase. Here, evidence is collected and used to form opinions about the trust propositions included in the expression. Each proposition is evaluated based on the evidence gathered, and the expression is then assessed using a formal method, resulting in the calculation of the ATL.

Finally, the **trust evolution** and **feedback** phase ensures that trust evaluations are not static but dynamically adjusted as new data or evidence becomes available. The trust model evolves over time, with feedback loops allowing for continuous updates to the trustworthiness levels.

It is important to emphasize that part of the ATL methodology is also choosing a consistent mathematical model for both expression generation and expression evaluation. The grammar used to construct expressions must align with the mathematical framework chosen to evaluate them. The mathematical models have their own sets of rules and computational techniques, and they cannot be used interchangeably. Therefore, part of the methodology involves carefully choosing both the syntax for defining trust expressions and the mathematical model for evaluating them. The choice of model ensures that trust expressions are evaluated in a logically consistent and mathematically sound way, e.g., subjective logic, Bayesian networks, or another formalism.

Expression Generation

When constructing trust expressions, we are essentially building formulas that express how trust in various propositions is calculated. These expressions can vary in complexity, ranging from simple atomic expressions to more complex logical combinations of multiple propositions.

Atomic Propositions

An atomic trust opinion, denoted as ω_X^A represents the trustworthiness of a single proposition (i.e., variable X). This type of expression deals with only one specific aspect of trust, and the opinion about this proposition is formed based on direct evidence observed by agent A. There can be one or multiple direct evidence coming from one or more trust sources. Atomic proposition is a proposition that cannot be broken down to simpler terms and evidence can either support it or contradict it.

When forming atomic propositions in the ATL methodology, it is important to define whether a proposition has only two possible outcomes (binomial) or multiple mutually exclusive outcomes (multinomial). This distinction affects how trust is calculated and combined. A binomial proposition, like "Vehicle A's GPS signal is reliable," has two outcomes: true or false. The trust opinion for such a proposition is represented by $\omega = (b, d, u)$. On the other hand, a multinomial proposition involves multiple possible outcomes. Clearly defining whether a proposition is binomial or multinomial ensures that trust calculations are accurate and that the right mathematical tools are used to combine evidence effectively [19].

Complex Propositions

A complex proposition combines multiple atomic propositions using logical operators [20]. For example, we might combine the following propositions:

- ▶ Proposition 1: "Vehicle A's position data is accurate."
- ▶ Proposition 2: "Vehicle A's speed data is reliable."
- ▶ Proposition 3: "Vehicle A's RSU communication is trustworthy."

A complex proposition can be represented as:

Proposition 1 OR (Proposition 2 AND Proposition 3)

This expression states that either Proposition 1 is trusted, or both Proposition 2 and Proposition 3 are trusted simultaneously.

Trust Expressions

A trust expression is a formalized representation of trust relationships and dependencies between entities in a system. It defines how trust opinions about atomic and complex propositions are combined, evaluated, and propagated using logical and probabilistic operators on atomic or complex trust propositions. A trust expression allows us to evaluate the trustworthiness of a proposition (complex or atomic).

A trust expression allows us to evaluate an atomic proposition that can involve multiple factors or conditions based on evidence stemming from different sources.

Here, evidence can be either directly observable or accessible transitively from remote sources (so not directly observable from the trustor) [22]. For the latter, it allows using trust operations – such as fusion, discounting, and referral – to integrate information from different trust sources. Of course, trust expressions still allow us to build compound expressions combining opinions on different atomic propositions using logical operators such as AND, OR, NOT.

For example, trust expression on an **atomic** proposition could be:

$$\begin{aligned} \triangleright \omega_X^A &= \omega_B^A \oplus \omega_X^B \\ \triangleright \omega_X^A &= (\omega_B^A \oplus \omega_X^B) \otimes (\omega_C^A \oplus \omega_X^C) \end{aligned}$$

where \otimes represents the discount operator and \oplus denotes the fusion operator (see below).

An example of trust expression on a **complex** proposition could be:

$$\triangleright \omega_X^A = (\omega_B^A \oplus \omega_X^B) \wedge (\omega_B^A \oplus \omega_Y^B)$$

In what follows, we define in more detail the different kinds of operators that can be used to build trust expressions and also how trust model representation can help us build trust expressions.

Operators in Trust Expressions

Once we have atomic propositions, we need operators to combine them into compound expressions. Subjective logic provides several key operators to express relationships between opinions about different propositions [19].

1. Addition (AND operator)

The AND operator (also called conjunction) combines two or more propositions, where all must be true (trusted) for the overall expression to be trusted. The AND operator is used when trust in a situation depends on multiple propositions being true simultaneously.

2. Disjunction (OR operator)

The OR operator (also called disjunction) combines propositions where only one needs to be true for the overall expression to be considered true. If any of the propositions are trusted, the overall expression will also be trusted to a certain degree. The OR operator is used when trust in a situation can be satisfied by any one of several propositions being true.

3. NOT operator

The NOT operator is used when we want to express distrust or disbelief in a proposition. It inverts the belief and disbelief components of the opinion. The NOT operator is applied when we want to express that a proposition is not trusted, or we want to reverse the trust assessment.

4. Discounting operator

The discounting operator is used when trust in one proposition is passed through a referral chain, meaning the trust in one proposition is discounted by the trust in the entity providing the opinion. The discounting operator is used when one agent (e.g., Vehicle A) passes on trust to another agent's (e.g., Vehicle B's) opinion. Trust in Vehicle B's data is discounted by how much Vehicle A trusts Vehicle B.

5. Fusion operator

Fusion operators can be used to fuse trust opinions derived using different ways. These fusion operators are essential for merging opinions in various scenarios, such as trust analysis, decision-making, and expert systems. Several fusion operators are used depending on the nature of the information and the relationship between sources.

- ▶ Belief Constraint Fusion (BCF) applies when no compromise is possible between opinions, meaning no conclusion is drawn if there is total disagreement.
- ▶ Cumulative Belief Fusion (CBF), in its aleatory and epistemic forms, assumes that adding more evidence reduces uncertainty, especially in statistical processes (A-CBF) or subjective knowledge (E-CBF).
- ▶ Averaging Belief Fusion (ABF) is used when opinions are dependent but equally valid, averaging them without assuming more evidence increases certainty. Weighted Belief Fusion (WBF) gives more confident opinions greater weight, ideal for expert input where confidence varies.
- ▶ Finally, Consensus and Compromise Fusion (CCF) preserves shared beliefs while turning conflicting opinions into vague beliefs, reflecting uncertainty and fostering consensus.

Choosing the appropriate fusion operator depends on the specific situation. For example, BCF is useful when strict agreement is required, while CCF is suited for cases where compromise is possible. By understanding the nature of the opinions and their relationships, analysts can select the most effective fusion operator to ensure accurate and meaningful results.

Trust Model Representation: A Framework for Generating Trust Expressions

The **trust model** representation serves as a structural framework that defines the relationships between different entities and the variables for which one wants to assess the trust. These models allow us to clearly show how trust is built, which entities are involved, and what aspects of trust need to be assessed. The specific aspect of trust to be assessed is determined within the scope of the trust model representation. From this model, we can derive trust expressions that represent how trustworthiness is calculated based on the trust of each relationship between entities.

Part of defining the trust model can also be specifying the **trust sources**. Namely, depending on the property, appropriate trust sources need to be defined that provide enough evidence for the fulfilment of the corresponding property. Decisions on trust are rarely made on a single parameter, and trust is always contextual. Thus, depending on the trust properties of interest, different sources are selected to do the trustworthiness assessment and quantify the resulting trust opinion and relationship.

Note that generally speaking trust sources might be diverse. However, trust sources for a referral will be based on past experience, since it expresses the trustor's confidence in the trustee's capability and honesty in providing good advice..

The ATL methodology remains agnostic to the specific model used. Below we give a list of some approaches for representing trust models, each suited to different contexts and system requirements. Each of these models offers distinct strengths and trade-offs, making them suitable for different aspects of trust modeling in CCAM systems. By remaining flexible in trust model representation, the ATL methodology can adapt to diverse operational contexts and stakeholder requirements.

1. Trust Matrix

The Trust Matrix is a structured representation of trust values between entities using a matrix-based approach [23]. It primarily captures pairwise trust values, making it a straightforward yet effective model for trust representation. In this approach, each entity corresponds to a row and column in the matrix, and the values within the matrix signify the level of trust between these entities. A zero value, for example, can indicate no trust relationship, while higher values represent stronger trust bonds. This model is commonly employed in scenarios where trust assessments are static and predefined, such as access control systems and simple trust-based recommendation frameworks.

However, the Trust Matrix approach is inherently limited in handling complex relationships, such as transitive trust or probabilistic dependencies. Since it does not explicitly model uncertainty, additional computational mechanisms are required to incorporate probabilistic reasoning. This makes it less flexible in dynamic environments where trust values may evolve over time due to contextual changes or indirect influences.

2. Bayesian Network (BN)

Bayesian Networks offer a probabilistic approach to modeling trust by representing dependencies between variables through conditional probability distributions [24]. In this representation, trust relationships are encoded as Directed Acyclic Graphs (DAGs), where nodes represent entities or propositions, and edges capture the probabilistic dependencies between them. This approach allows for uncertainty modeling, as BNs can quantify the likelihood of trust propagation based on prior knowledge and evidence.

A major advantage of BNs in trust modeling is their ability to handle indirect relationships. By utilizing inference mechanisms, these networks can compute trust levels based on observed data, making them well-suited for applications such as fraud detection and decision-making under uncertainty. However, implementing BNs requires significant computational resources, and defining the conditional probability distributions can be complex, especially in large-scale or dynamic environments.

3. Subjective Trust Network (STN)

Subjective Trust Networks extend traditional trust models by incorporating subjective opinions and explicitly representing uncertainty [20]. These networks leverage subjective logic, which provides a framework for reasoning under uncertainty by supporting logical operators such as AND, OR, and NOT. The edges in a STN denote directed trust relationships, which can exist between entities or between entities and propositions.

The key strength of STNs lies in their ability to handle uncertainty more flexibly than traditional probabilistic models. Since they are built upon subjective logic, they allow for nuanced trust assessments that consider varying degrees of confidence in trust values. However, a drawback of this approach is that it does not inherently support enforcing logical constraints, as these are not encoded within the network structure itself. Despite this, STNs are useful in applications where trust must be dynamically inferred from multiple subjective perspectives, such as reputation systems and decentralized trust management.

4. Subjective Network = STN+BN

Subjective Networks generalize Subjective Trust Networks by integrating elements from both Bayesian Networks and subjective trust modeling [19]. This allows for a more robust representation of trust, where directed edges capture trust relationships and probabilistic dependencies between propositions. Unlike simpler models, Subjective Networks explicitly handle uncertainty and enable flexible reasoning through probabilistic inference mechanisms.

A notable benefit of Subjective Networks is their ability to combine logical reasoning with probabilistic trust assessment. This makes them particularly effective in scenarios requiring adaptive trust evaluation, such as multi-agent systems and collaborative filtering. However, their computational complexity remains a challenge, as reasoning over large networks can require extensive processing power. Additionally, defining the probabilistic dependencies between trust relationships necessitates a thorough understanding of the domain-specific trust dynamics.

5. Semantic Graph (Knowledge Graph)

Semantic Graphs employ a graph-based model to define relationships between concepts using ontologies and semantic rules [25]. Unlike the other trust representations, which primarily focus on trust relationships between entities, Semantic Graphs extend trust modeling to a broader knowledge domain by incorporating objects, actions, and ideas. These relationships are established through predefined rules and structured ontologies, enabling automated reasoning and inference.

One of the key advantages of Semantic Graphs is their ability to encode transitive trust relationships through hierarchical definitions. This makes them particularly effective in applications such as knowledge representation, semantic web technologies, and AI systems. However, while Semantic Graphs can express confidence levels through semantic rules, they do not inherently incorporate probabilistic reasoning. This limits their ability to dynamically adjust trust assessments based on new evidence, necessitating additional mechanisms for probabilistic trust inference.

Expression Evaluation

The expression evaluation phase is the most critical step in the ATL methodology, where trust expressions, constructed from atomic and complex propositions, are evaluated to compute an overall trustworthiness level. This process involves applying logical operators, aggregating evidence, and handling uncertainty to derive a meaningful trust opinion for the evaluated expressions. The output of this phase is a trust opinion in the form of a triplet: belief (b), disbelief (d), and uncertainty (u).

We break down this phase in the following steps: First, evidence collection from the trust sources, second, atomic opinion calculation, and third the expression evaluation itself.

Evidence Collection

Before evaluating a trust expression, the system needs to gather relevant evidence that supports or contradicts the trust propositions in the expression. Evidence is the key input for forming opinions about each trust proposition. The preceding 5GAA white paper [7] emphasizes collecting evidence from several trust sources to ensure reliable trust evaluations in Connected and Automated Vehicle environments.

Broadly speaking, we can categorize the different types of evidence into static evidence and runtime evidence, each serving distinct purposes within the lifecycle of vehicle operation.

- ▶ **Static evidence:** Refers to attributes or properties that are evaluated at a specific point in time, often during development, certification, or pre-deployment phases. So, for example, static evidence ensures compliance with design requirements, manufacturing standards, and safety certifications.
- ▶ **Runtime evidence:** Generated during the operational phase of the vehicle, capturing real-time behavior and system states. Unlike static evidence, runtime evidence reflects dynamic aspects like environmental interactions, system performance, and fault tolerance. Importantly, runtime evidence often produces probabilistic outputs or confidence levels rather than binary evaluations, enabling nuanced trust assessments.

Static evidence provides foundational trust but varies in confidence based on the strength or quality of the mechanism. For example, when it comes to cryptographic protection, long key lengths and robust algorithms can provide higher confidence than shorter key lengths or deprecated algorithms. Or when it comes to compliance certifications, adherence to stringent standards like ISO 26262 (ASIL D for functional safety) can provide higher confidence than compliance with basic regulatory requirements.

Unlike static evidence, which provides foundational trust based on pre-deployment tests and certifications, runtime evidence reflects the system's real-time performance and behavior. For example, runtime attestation mechanisms can validate the integrity of a vehicle's ECUs during operation, ensuring that no unauthorized modifications have occurred. Similarly, runtime evidence for correct sensor operation in real time relates to evidence that sensors remain functional and reliable under the specific environmental

and operational conditions as defined in the Operational Design Domain (ODD).

The evidence must be targeted and context-specific to the propositions being evaluated. For example, if we are assessing the trustworthiness of a vehicle's braking system, the system needs to gather evidence specifically related to the braking performance.

Atomic Opinion Calculation

Once evidence is collected, the next step is to form opinions about each proposition. As explained earlier, trust opinions in subjective logic are represented by a triplet: belief (b), disbelief (d), and uncertainty (u). To calculate the binomial opinion of random variable X from directly observed evidence, we can use one of the following equations depending on the type of evidence we have. In practice, we can categorize them into three groups [18][19]:

- Baseline-Prior Quantification: A prior weight is set so the uncertainty decreases when the total number of evidence (positive and negative) decreases,

$$\begin{cases} b_x = \frac{r_x}{W + r_x + s_x} \\ d_x = \frac{s_x}{W + r_x + s_x} \\ u_x = \frac{W}{W + r_x + s_x} \end{cases} \quad (\text{Eq. 1})$$

where r_x and s_x represent the positive evidence and negative evidence of X taking value x, respectively. W is a non-informative prior weight, which has a default value of 2 to ensure that the prior probability distribution function (PDF) is the uniform PDF when $r_x = s_x = 0$ and $a_x = 0.5$ [19].

- Constant-Uncertainty Quantification: Where the uncertainty is fixed, for example because we know the number of evidence in advance or we know that the quantification itself has uncertainty fixed,

$$\begin{cases} u_x = U, \\ \gamma = \frac{1-U}{r_x + s_x} \\ b_x = \gamma \times r_x \\ d_x = \gamma \times s_x \end{cases} \quad (\text{Eq. 2})$$

- Evidence-Weighted Quantification: In this case, we do not only have positive and negative evidence, but we also have evidence for uncertainty. This type of quantification might be suitable when evidence is not countable (or binary by nature) or when evidence itself can come with uncertainty,

$$\begin{cases} b_x = \frac{r_x}{w_x + r_x + s_x} \\ d_x = \frac{s_x}{w_x + r_x + s_x} \\ u_x = \frac{w_x}{w_x + r_x + s_x} \end{cases} \quad (\text{Eq. 3})$$

The above approach is the most simplistic one, where we assume that each evidence contributes equally to the belief or disbelief. One could incorporate weights for evidence, where different pieces of evidence are assigned varying levels of importance or reliability. This would allow for more nuanced opinion formation based on the quality or credibility of evidence. Also, an additional approach is to define an activation function, in order to describe how much belief or disbelief should be increased when specific evidence for a protection mechanism is in place (or decreased when it is not). This function could be linear, exponential, etc.

Evaluate the Expression

Expression evaluation is a critical phase in the ATL methodology, where trust expressions (comprising atomic and complex propositions) are evaluated to compute trustworthiness. This process integrates the trust opinions derived during the evidence collection phase and applies logical operators and subjective logic mechanisms to combine and propagate trust. Logical operators such as AND, OR, and NOT define how trustworthiness is evaluated across multiple propositions, while trust-specific operators like discounting and fusion handle scenarios involving referrals and multiple independent sources of evidence. The resulting trust opinion, expressed as a triplet of belief (b), disbelief (d), and uncertainty (u), provides a formalized representation of the overall trustworthiness of the evaluated expression.

The evaluation process dynamically accounts for factors such as uncertainty and conflicting evidence. For instance, when two trust sources provide inconsistent opinions about a proposition, the evaluation incorporates this conflict by increasing uncertainty in the resulting trust opinion. Specialized operators, such as discounting, adjust trust based on the reliability of intermediary agents, while fusion combines multiple opinions with weighted confidence levels to ensure fairness and accuracy. This adaptability is especially crucial in dynamic environments, where real-time updates to evidence and contextual changes, such as shifts in ODD, require continuous re-evaluation of trust expressions. By ensuring consistency, robustness, and adaptability, the expression evaluation phase supports accurate trust assessments in complex, multi-agent systems.

Trust Evolution and Feedback

The goal of the continuous updates and feedback loops phase is to ensure that the trustworthiness evaluations are dynamically updated based on new evidence and changes in the environment. This allows the system to adapt to real-time inputs, recalculating trust levels as needed, and maintaining a high level of accuracy in its assessments.

The process operates in an event-driven manner, where external events, such as new sensor data, updated V2X messages, or changes in the trust environment, automatically trigger updates within the trust model. Each event triggers either

- ▶ the gathering of new evidence and updating the existing trust opinions (expression remains the same), or

- ▶ the updating of the trust model itself and therefore having a new trust expression, which ensures that the trust model reflects the current state of entities and relationships.

Re-evaluation of Trust Opinions

Re-evaluation of the trust opinion(s) occurs under one of the following conditions:

- ▶ New or lost trust sources: When a new trust source is added, or an existing trust source is lost, the trust opinion for which that trust source contributes must be updated.
- ▶ New evidence: If new evidence becomes available for a given trust source, or if existing evidence changes (e.g., from negative to positive or vice versa), the trust opinion for that source needs to be revised.

In either case, the updated trust opinion may influence the resulting ATL, prompting the need for a re-evaluation of the trust expression. The ATL therefore needs to be recalculated based on the updated trust opinions.

Re-evaluation of Trust Expressions

In the case when a new entity enters the environment, or when an entity leaves, the trust model needs to be updated accordingly. This ensures that the trust model reflects the most current and accurate information available. In consequence, this needs to trigger the re-synthesis of the trust expression, to account for the changes in the trust model.

4 Methodology for RTL Calculation

This section introduces Required Trustworthiness Level [12], as well as a high-level methodology for defining its thresholds. Since risk assessment is the basis of the method, we introduce important standards that can be used for risk assessment in the context of automotive cybersecurity and safety. RTL is used for trust decisions from the perspective of the receiver vehicle (as a reminder, RxV), all assessments are done based on RxV use and needs.

Background

The method introduced in the following passages may be used to create both safety and security required trustworthiness levels. For this reason, the section briefly introduces the most common risk assessment in the automotive field: Threat Analysis and Risk Assessment, used for cybersecurity engineering for road vehicle, and Hazard Analysis and Risk Assessment, used to assess safety risks associated with E/E systems. It is important to note that while the application of this methodology on security is thoroughly validated in the R&D project CONNECT [15], the application on safety needs further detailed study and validation. The concepts of this white paper can serve as basis for future research.

Threat Analysis and Risk Assessment (TARA)

The ISO/SAE 21434:2021 [10] defines an international standard for cybersecurity engineering for road vehicles. It includes cybersecurity processes, and risk management, and promotes a cybersecurity culture for road vehicles. One of the main contributions of this standard is a framework that includes requirements for cybersecurity processes and risk management. TARA is performed on an item, defined as a component, or set of components that implement a function at the vehicle level. The item definition shall include at least information regarding its function, attack surface, boundaries, and operational environment. The TARA activities are:

- ▶ Item definition: Includes item boundary, functions, and preliminary architecture.
- ▶ Asset identification: Identification of assets that, if any of the cybersecurity properties are compromised, might result in an adverse scenario. This process also identifies potential damage scenarios.
- ▶ Threat scenario identification: Determine attack scenarios that target one or more assets while threatening one or more cybersecurity properties.
- ▶ Impact rating: The impact of a damage scenario is measured and assigned to four categories: safety (S), financial (F), operational (O), and privacy (P), with ratings of severe, major, moderate, and negligible.
- ▶ Attack path analysis: Determining the intentional steps required to execute a threat scenario and thereby initiate a damage scenario.

- ▶ Attack feasibility rating: Estimates how difficult it is to carry out the considered attack. Very low, low, medium, or high are the possible classifications.
- ▶ Risk level determination: It is a value calculated considering the impact rating and attack feasibility rating. The value ranges from 1 to 5, with 5 representing the highest risk. The risk equation can vary between manufacturers.
- ▶ Risk treatment decision: Describes the chosen treatment strategy for an identified risk. It can be done to prevent, mitigate, share, or retain the risk. A TARA can be repeated to calculate the residual risks after risk treatment decisions have been applied.

Hazard Analysis and Risk Assessment (HARA)

ISO 26262 [11] is an international standard for the design and development of automotive E/E systems and makes functional safety a part of the automotive product development, helping to eliminate any unacceptable risk to human life. HARA is defined in this standard and its purpose is to identify and classify malfunctions that could possibly lead to E/E system hazards and assess the risk associated with them. The output of HARA is used to formulate safety goals with their corresponding Automotive Safety Integrity Level (ASIL) related to the prevention or mitigation of unreasonable risk. The ASIL is determined by considering severity, probability of exposure, and controllability factors.

The HARA activities are:

- ▶ Item definition: HARA is based on the item definition without internal safety mechanisms.
- ▶ Situation analysis and hazard identification: Identification of operational situations and modes where an item's malfunctioning behavior can result in a hazardous event. Both correct and incorrect use shall be described in a reasonably foreseeable way, and the possible hazards shall be determined. Failure Mode and Effects Analysis (FMEA) and Hazard and Operability Analysis (HAZOP) are suitable to support hazard identification.
- ▶ Classification of hazardous events: All identified hazardous events shall be classified with respect to severity (S), probability of exposure (E), and controllability (C). Severity can be classified as no injuries (S0), light and moderate injuries (S1), severe and life-threatening injuries with survival probable (S2), life-threatening injuries with survival uncertain or fatal injuries (S3). Probability of exposure can, in turn, be classified as incredible (E0), very low probability (E1), low probability (E2), medium probability (E3), or high probability (E4). Controllability can be classified as controllable in general (C0), simply controllable (C1), normally controllable (C2), or difficult to control or uncontrollable (C3).
- ▶ ASIL determination: ASIL shall be determined for each hazardous event based on the classification of S, C and E. They can be classified as ASIL A, ASIL B, ASIL C or ASIL D, where ASIL A is the lowest safety integrity level and ASIL D the highest one. Another level, known as QM (Quality Management),

covers hazards that do not require any safety measures.

- ▶ Determination of safety goals: For each hazard, a safety goal shall be defined to mitigate the associated risk. They will be used for further verification.
- ▶ Management of variances: The differences in type of vehicle shall be taken into consideration, such as differences in base vehicle configuration or loading conditions for trucks.
- ▶ Verification: The hazard analysis and risk assessment, including safety goals, shall be verified in relation to compliance with the item definition, consistency with related hazards associated with other items, completeness of coverage, and consistency of the safety goals with the assigned ASILs.

Required Trustworthiness Level

Required Trustworthiness Level defines the minimum level of trustworthiness needed for a trustee, such as a vehicle function or data, to be considered reliable. Its definition might be based on risk analysis criteria, technical requirements, or regulations for technical approval. RTL is established during the design phase, and our research presents a method for calculating RTL using risk assessment as the basis. RTL serves as a numerical trustworthiness threshold for trust decision-making [12].

RTL sets thresholds for minimum acceptable belief (b_t), maximum permitted disbelief (d_t), and uncertainty (u_t) values of subjective trust opinions. These three thresholds can range from 0 to 1 and are used independently. Although they might share a common foundation, like inputs from the same risk assessment, they can consider different aspects of the situation. For b_t , 0 means no belief level is required, and 1 means full belief is required. While b_t can be zero, having it at zero is not recommended, as it may indicate a breach in the zero-trust model. The range for d_t and u_t is also from 0 to 1, but they reflect their maximum accepted level. High tolerance of disbelief and uncertainty is not recommended.

Determined during the vehicle design phase, an RTL cannot rely on runtime evidence since the vehicle is still in engineering development. However, it is possible to use risk assessment tools, assumptions and considerations made and observe predicted cybersecurity scenarios.

We develop a risk-based approach to calculate RTL, which can utilize either TARA or HARA output, since they are both already standardized and implemented by automakers as part of their cybersecurity and safety risk assessment process. The terms used by the RTL methodology are generic, as they are intended to be used by safety and cybersecurity analysts, and equivalent terms in both TARA and HARA can be found in Table 2.

Table 2 Mapping RTL methodology terms to TARA and HARA equivalents

Term in RTL methodology	Likelihood	Impact	Risk level
Term in TARA	Attack feasibility	Impact rating	Risk rating
Term in HARA	Exposure	Severity x controllability	ASIL

Companies can enhance the rigor of threshold calculation methods by including additional demands that are not expressly covered in existing standards or the calculation process. This might include incorporating company-specific regulations, processes, or special operational requirements into the RTL calculation methodology. By adjusting the process in this way, companies may establish more accurate and relevant baselines or thresholds for making trust decisions. This variation helps that the RTL is consistent with the company's unique context and objectives, resulting in more effective RTLs.

Belief Threshold Calculation

In terms of belief, positive evidence is used to increase confidence in the trustee. Secure connections, protocols, authentication techniques, physical security, and various other security elements or strategies can be used to build a trustworthy system.

For example, consider two comparable systems that provide the same type and number of trust sources. Based on an assessment, risk levels can vary from low to critical. Consider that system X and Y have the same capabilities and identified risks, but while system X lacks security and safety features, system Y was designed with features to mitigate some critical risks, resulting in lower risk levels when compared to system X. In this example, system X would require more evidence to attest to its trustworthiness since its risk levels are higher. On the other hand, system Y would not require the same amount of evidence as X, considering its lower risk levels. The risk-based approach considers the risk levels of the item to determine the required level of trustworthiness and make the final trust decision.

In the risk-based methodology described in [12], the riskier the system, the higher the required belief value. The risk-based scheme for calculating the b_t component of the RTL is shown in Figure 6. This approach maps risks level through to risk assessment, considering risk likelihood and impact rating, into b_t .



Figure 6 Risk-based belief threshold calculation. Dark-blue box: standardized risk assessment. Grey arrows: inputs/outputs. Light-blue box and arrow: methodology calculation and output

To calculate b_t , we leverage risk assessment, which enables engineers to anticipate risks that the system is likely to face and assists in making risk mitigation decisions. The risk assessment results in a list of risks associated with their likely impact and a specific technical system model.

As shown in the Figure 6, the key is to identify from a wide range of risks which are relevant or in scope and which are irrelevant. This step is crucial to ensure that the b_t calculation is focused on the most critical risks. A suggested approach is to consider the worst-case scenario, which guarantees that the RTL covers all potential risk scenarios, but that approach may make the system more stringent/onerous by demanding stronger trust evidence.

Disbelief threshold calculation

Disbelief refers to the characteristics of a system or the interaction between the trustee and the trustor that indicate the system is not trustworthy, often known as negative evidence. The disbelief threshold (d_t) calculation may use an impact-based method that considers residual risk. In other words, if risks are foreseen and decisions were made to keep them, or if any residual risk exists after implementing mitigation techniques, this may be considered as negative evidence. In this case, the impact of accepting them might rule the level of disbelief as 'accepted' by default. Possible impacts on, for example, safety, economy, privacy, operation/function, or any other relevant aspects, play an important role in the d_t definition, assuming that d_t can be interpreted as the amount of risk the system is allowed to take on.

Figure 7 outlines an impact-based approach for calculating d_t . This method involves a deeper evaluation of the scope- and impact-relevant risks identified during the assessment phase. By examining the potential impact of these risks on several aspects, we can determine d_t .



Figure 7 Risk-based disbelief threshold calculation. Dark-blue boxes: standardized risk assessment or impact analysis methodologies. Grey arrows: inputs/outputs. Light-blue box and arrow: methodology calculation and output

As shown in the figure, once the technical system model is defined, the first step is to gather the risk analysis output, which includes a list of expected cybersecurity and safety risks and the impact of these risks if they occur. For example, we may consider these impacts, whenever possible:

- ▶ **Safety impact:** The possible implications of a system failure or compromise on human safety play an important role in evaluating disbelief. Risks that directly endanger human life or well-being will significantly increase disbelief, decreasing d_t .
- ▶ **Privacy impact:** Another key aspect is the system's vulnerability to unauthorized access leading to the disclosure of sensitive information will raise disbelief and decrease the d_t .
- ▶ **Economic impact:** The possible financial ramifications of a system failure or compromise, such as lost income, operational interruptions, or legal obligations, can all add to disbelief and decreased d_t .
- ▶ **Operational/functional impact:** Another element to examine is how a system failure or compromise affects the system's capacity to operate effectively and efficiently. Risks that threaten the system's operating capability will increase disbelief and decrease d_t .

Uncertainty Threshold Calculation

Uncertainty, in the context of RTL, refers to a lack of knowledge or information resulting in insufficient evidence to establish confidence. During the design phase, factors such as required assurance levels (i.e., uncertainty thresholds and criticality) as well as the ability of the system to detect an incident can be used as parameters to set the maximum acceptable level of uncertainty (u_t) in a given scope, as illustrated in Figure 8.

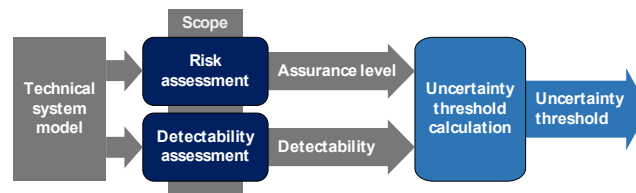


Figure 8 Risk-based uncertainty threshold calculation. Dark-blue boxes: standardized risk assessment or impact analysis methodologies. Grey arrows: inputs/outputs. Light-blue box and arrow: methodology calculation and output

The figure summarizes the key factors that can be considered to calculate u_t :

- ▶ **Detectability of an incident:** The ability to detect system misbehavior that triggers a risk is an important component in assessing u_t . A system with strong detectability mechanisms may accept more uncertainty (u_t) because it can identify, and possibly mitigate, certain risks before they may exploit weaknesses. Detectability can be covered by both safety (e.g., detectability of a failure mode) and security (e.g., detectability of a cybersecurity incident) perspectives. Detectability is concerning coverage and accuracy of the

incident detectors.

- ▶ Assurance level: Uncertainty is influenced by the system's overall security and safety posture, as well as its maturity. A higher minimum assurance level requires a system to be developed with more depth and rigor (e.g., testing), allowing less uncertainty by design. The higher the required assurance, the lower u_t .

To calculate u_t , we consider the interactions of these factors. The method assumes that the higher the assigned assurance level to the system, the lower its uncertainty acceptance. When it comes to the detectability of incidents, a system with few or inappropriate detectors is not expected to deal with uncertainty as well as one with satisfactory detectability features. All these discussed capabilities and expectations need to be reflected in u_t . For all parameters, the scope is applied, and only relevant aspects are considered during analyses.

5 Example Application on Use Case

ATL in Automated Emergency Braking

This section applies the Actual Trustworthiness Level methodology to the Automated Emergency Braking use case. The goal is to systematically assess the trustworthiness of data exchanged between vehicles during such a braking event using the ATL framework.

Trust Expression for the AEB Use Case

To construct the trust expression, we begin by defining the trust proposition under evaluation, selecting an appropriate trust modeling approach, and developing a corresponding model. Consider a scenario from our use case, where Vehicle A relies on sensor data from Vehicle B to make critical driving decisions (in this case speed adjustments). In this setting, Vehicle A must assess whether the position data received from Vehicle B is trustworthy, meaning it is both accurate and maintains data integrity.

The trust proposition can be formally expressed as

$V = \text{"Sensor measurement is accurate and provides the data with integrity"}$

This proposition can be decomposed into two components

$$V = X \vee Y$$

Where:

- ▶ X represents "The sensor provides the data with integrity"
- ▶ Y represents "The sensor makes accurate measurements"

For the next steps, our focus will be exclusively on variable X, analyzing how the trust model evaluates data integrity.

For the trust model representation, we adopt a trust network based on subjective logic, which provides a robust framework for trust assessment. This approach is particularly useful because it integrates uncertainty and enables the fusion of conflicting opinions, making it well-suited for real-world trust evaluations. The trust relationships and trust propositions between two vehicles in a network are illustrated in Figure 9, but the model is scalable, meaning that additional vehicles can be seamlessly integrated by establishing new trust relationships as they join the network.

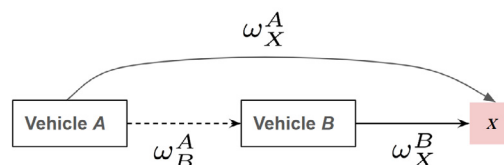


Figure 9 Trust model for the running example

In this subjective trust network, Vehicle A wants to determine how much it can trust sensor x , which is responsible for measuring Vehicle B's position. To form this trust opinion, Vehicle A has two primary approaches:

- ▶ Direct trust assessment – Vehicle A can directly evaluate Sensor x based on its own observations and collected data.
- ▶ Indirect trust assessment – Vehicle A can rely on Vehicle B's opinion about its own sensor, given that Vehicle B has firsthand experience with sensor x . Since Vehicle A already has an established trust relationship with Vehicle B, it can incorporate Vehicle B's assessment into its own trust evaluation of Sensor x .

This indirect approach means that Vehicle A does not have to solely rely on its own observations but can also factor in the trustworthiness of Vehicle B when forming an opinion about Sensor x . By combining both direct and indirect sources, Vehicle A achieves a more comprehensive and nuanced trust evaluation.

Based on the trust model definition and the available subjective logic operations, the Actual Trust Level or ATL of Vehicle A for the defined trust proposition related to Sensor x can be determined by evaluating the following expression:

$$ATL = \omega_X^A \oplus (\omega_B^A \otimes \omega_X^B)$$

where \otimes represents the discount operator and \oplus denotes the fusion operator, both of which are used to integrate trust evidence from multiple sources.

Collecting Evidence from Trust Sources and Computing Atomic Trust Opinions

According to the ATL methodology, the second step involves collecting relevant evidence claims and forming trust opinions based on available data. The set of evidence claims considered as an example in this scenario is summarized in Table 3.

Table 3: Trust model information for Vehicle A, summarizing trust evidence

Opinion	Evidence	Trust property
ω_B^A	Binary integrity measurement	Software stack
	List	Configuration integrity
ω_X^A	CRC checks/sequence numbers	Communication integrity
	Netflow header	Communication resilience
	Attested Execution Isolation	Source integrity

Let us clarify that the trust opinion ω_x^B of Vehicle B on its own Sensor x is assumed to be fully trusted with a belief value of (1,0,0). This assumption is based on the premise that the sensor is highly reliable from Vehicle B's point of view and directly controlled by the vehicle.

The trust opinion ω_B^A between Vehicle A on Vehicle B depends on the collected evidence. As we can see in the table, for this opinion we have a single piece of evidence. If the evidence is positive, we define $\omega_B^A = (0.33, 0, 0.67)$, whereas if the evidence is negative, it is set to $\omega_B^A = (0.33, 0, 0.67)$. Since the binary integrity measurement list consistently returns positive evidence in our scenario, we assume $\omega_B^A = (0.33, 0, 0.67)$ in all cases.

About ω_x^A , again as we can see in the table there are three separate pieces of evidence that Vehicle A uses to assess the trustworthiness of Vehicle B's system configuration:

- ▶ Data trace analysis, which confirms that the sensor data is transmitted successfully without Cyclic Redundancy Check (CRC) errors.
- ▶ Communication resilience, evaluated through the tracking of network robustness and potential disruptions.
- ▶ Attested execution integrity, which incorporates evidence obtained through remote attestation mechanisms about the secure and isolated execution of vehicle B's applications.

Based on the available evidence, we apply Equation 1 of Section 3 (with $W = 2$) to quantify the resulting subjective logic opinions.

We decided to assume two scenarios; for the first there is one piece of negative evidence (among the three possible sources) and for the second there is zero negative evidence, as shown in Table 4.

Table 4 ω_x^A opinion quantification in the two scenarios

	Scenario 1	Scenario 2
Positive evidence	$r_x = 2$	$r_x = 3$
Negative evidence	$s_x = 1$	$s_x = 0$
Final opinion	(0.4, 0.2, 0.4)	(0.6, 0, 0.4)

In scenario 1 (one negative piece of evidence), the belief value for trustworthiness is lower (0.4), while the distrust value is higher (0.2). In scenario 2 (only positive evidence), the belief value increases to 0.6, and distrust is eliminated.

From the results, we also observe that uncertainty remains constant at 0.4 across both scenarios. This is because the total amount of evidence (i.e., the sum of positive and negative evidence) is fixed, and we have applied a default prior information weight of 2 in the subjective logic model.

Evaluate the Expression

The final step involves computing the Actual Trust Level (ATL) by evaluating the trust expression we derived in the previous section:

$$ATL = \omega_X^A \oplus (\omega_B^A \otimes \omega_X^B)$$

So now we can integrate the trust opinions we calculated above and apply the subjective logic operators to obtain the final ATL values.

From our previous steps, we have the following trust opinions:

- ▶ Trust opinion of Vehicle B on its Own Sensor: $\omega_X^B = (1, 0, 0)$
- ▶ Trust opinion of Vehicle A on Vehicle B: $\omega_B^A = (0.33, 0, 0.67)$
- ▶ Trust opinion of Vehicle A on Sensor x: ω_X^A depends on the specific scenario as outlined in the above table.

Using these inputs, we apply the discount operator \otimes and fusion operator \oplus from the subjective logic approach to derive the ATL. The computed values for each scenario are summarized in Table 5.

Table 5 ATL quantification in the two scenarios

	Scenario 1	Scenario 2
ω_X^A	(0.4, 0.2, 0.4)	(0.6, 0, 0.4)
ω_B^A	(0.33, 0, 0.67)	(0.33, 0, 0.67)
ω_X^B	(1, 0, 0)	(1, 0, 0)
$\omega_B^A \otimes \omega_X^B$	(0.66, 0, 0.34)	(0.66, 0, 0.34)
ATL	(0.66, 0.11, 0.23)	(0.77, 0, 0.22)
Projected probability	0.775	0.88

One way to interpret the ATL is to calculate the corresponding projected probability, defined as:

$P = b + aXu$ where b is the belief, u the uncertainty, and a the base rate (usually set to 0.5).

So, what we observe is that Scenario 2, where only positive evidence is present, results in a higher projected probability of 0.88, compared to 0.775 in Scenario 1, where one negative evidence claim is present. The presence of even single negative evidence claim reduces both the belief value and the projected probability, reinforcing the need for comprehensive trust evaluation mechanisms.

This final evaluation step completes the ATL assessment, demonstrating how trust relationships and subjective logic operations influence the final trustworthiness estimation. In the next section we apply the methodology of RTL calculation in our use case.

RTL in Automated Braking Event

Earlier in this paper, we provided the theoretical framework for deriving Required Trustworthiness Level or RTL from risk assessments. In this section, we will bridge the gap between theory and practice by calculating RTL thresholds. Through this exercise, we will gain a deeper understanding of how to quantify these subjective factors.

To determine the RTL, we utilize subjective logic, a formal framework for reasoning with uncertainty and trust. The RTL serves as a threshold for the ATL and, as illustrated in Figure 10, the expected RTL (green diamond) would ideally be positioned within a region of high trust and low uncertainty in the subjective logic triangle, ensuring a balance between confidence and risk.

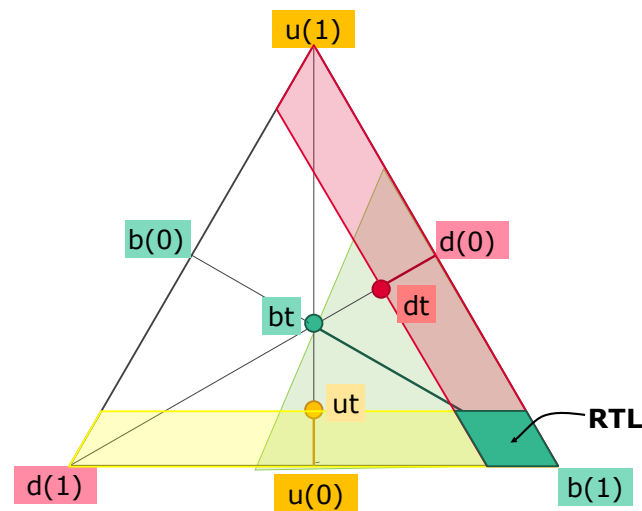


Figure 10 Graphical representation of RTL within subjective logic triangle

To exemplify the method, the use case introduced above is used as the object of analysis. For the trustworthiness evaluation, we define the following scopes:

- ▶ Scope 1 security: Integrity of the in-vehicle sensor data (direction, position, speed, camera, braking, clock).
- ▶ Scope 2 security: Integrity of the DENM and CAM received by ADAS.
- ▶ Scope 3 safety: Availability of cooperative safety messages (DENM and CAM) to be received by ADAS.

Scopes 1 and 2 address the cybersecurity perspective in assessing in-vehicle and V2X trustworthiness, respectively. Scope 3 investigates the safety perspective of the V2X trustworthiness assessment. The architecture for Scope 1, as presented in 11, shows the architecture for Scope 1, while Figure 12 illustrates the considered architecture for Scopes 2 and 3. Note that, as mentioned above, the application of the RTL methodology on safety is experimental. The exemplary application on Scope 3 serves the purpose to further explain how this methodology **may** be implemented for safety.

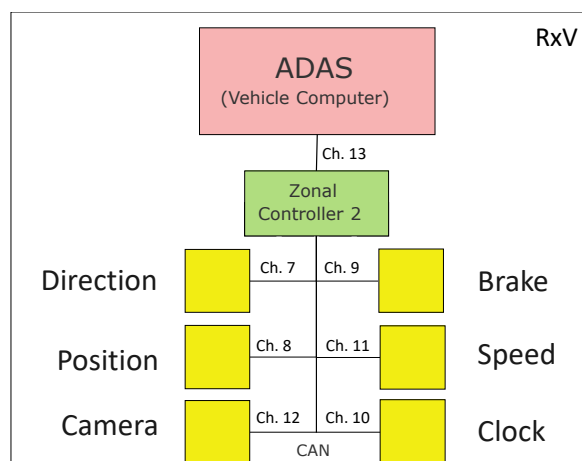


Figure 11 E/E architecture for Scope 1

The architecture represented by Figure 11 concerns the communication between the ADAS and the sensors. It is composed of six sensors and their data, a zonal controller and the ADAS all communicating through CAN bus. Furthermore, it perceives the environment (by camera, position sensors), the vehicle behavior (by direction, brake, speed sensors) and time (clock). Zonal controller 2 forwards the gathered sensors' data.

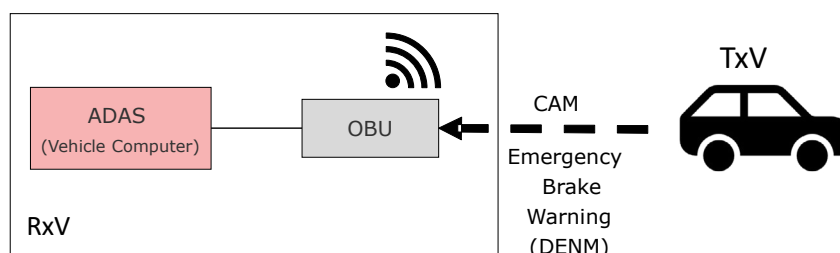


Figure 12 E/E architecture for Scope 2 and Scope 3

Figure 12 illustrates the communication between the RxV and TxV to receive CAM and DENM. It comprises the vehicle computer (ADAS) and OBU, as part of RxV, and CAM and DENM data being sent by TxV and received by RxV using wireless communication. It shows the process responsible for receiving V2V messages.

While each OEM may define their own specific methods for calculating RTL thresholds, the core principles remain consistent. In the following pages, we demonstrate our approach using illustrative formulas to highlight the key steps involved. As the scopes are defined by cybersecurity concerns, TARA will be used for the use case demonstration. The TARA report, detailed in Annex A, provides a summary of the risk assessment. For more comprehensive information and additional details, please refer to the specific example tables.

Belief Threshold Calculation Method

To map the risk level into a belief threshold (b_t), we apply the mathematical approach as described in [12]. First, we define the interval for acceptable belief values using Equation 4:

$$\Delta = \frac{1 - b_{bt}}{5}, 0 \leq b_{bt} \leq 1 \quad (\text{Eq. 4})$$

where

- ▶ Δ represents the interval between risk levels
- ▶ b_{bt} represents the defined baseline, which ranges from 0 to 1
- ▶ 5 is the standardized number of risk levels
- ▶ b_t ranges from b_{bt} to 1, so $(1 - b_{bt})$ calculates the actual range b_t

Next, we calculate the required b_t threshold using Equation 5:

$$b_t = b_{bt} + ((R_{max}(F, I) - 1) \times \Delta) \quad (\text{Eq. 5})$$

where

- ▶ b_t is the belief threshold
- ▶ $R_{max}(F, I)$ is the maximum risk level, ranging from 1 to 5; risk calculation considers feasibility (F) and impact (I)

The risk function $R(F, I)$ can vary between manufacturers. To simplify the calculation and ensure b_t ranges from 0 to 1, we normalize $R_{max}(F, I)$ to range from 0 to 4.

Equation 5 effectively converts the five-level risk scale into a b_t value between 0 and 1. The equation's key variables are $R_{max}(F, I)$ and b_{bt} . Vehicle manufacturers can determine $R_{max}(F, I)$ based on their specific needs. b_{bt} allows manufacturers to incorporate subjective factors and edge-case considerations. By setting b_{bt} to 0, Equation 5 enables b_t to be 0. This might be undesirable in certain scenarios. b_{bt} empowers project owners to establish a baseline for b_t based on their expectations and safety priorities, extending beyond the limitations of the considered risk assessment. b_{bt} is optional and defaults to 0. While determining b_{bt} is beyond the scope of this work, it is considered a manufacturer's assumption, accommodating specific safety, regulatory, or other relevant factors.

Considering HARA, we consider the ASIL level, whose values are QM, ASIL A, ASIL B, ASIL C, ASIL D, and also can be translated to range from 1 to 5 (refer to Table 6), respectively, and $R_{max}(F, I)$ is expressed as $R_{max}(E, S, C)$.

Table 6 Translation of ASIL to a numerical value

ASIL	Numerical value for $R_{max}(E, S, C)$.
QM	1
A	2
B	3
C	4
D	5

Belief Threshold Calculation for the Use Case

To calculate the minimum accepted belief expressed by b_t , we observe from the performed TARA (Annex A) the relevant risks for the scope and evaluate their risk level to select the worst-case scenario. After selecting it, we can calculate d_t using Equation 1 and Equation 2. Following, we perform the b_t calculation for each scope.

- Scope 1: Integrity of the in-vehicle sensor data (direction, position, speed, camera, braking, clock).

Table 7 lists the relevant risks from Scope 1 extracted from Annex A. We can observe risks related to the sensors themselves, communication channels and ECUs that may be able to change the integrity of the transmitted data.

Table 7 List of relevant risks for Scope 1

Name	Title	Caused by	Risk Level
R.2	Tampering on internal perception data and DENM in the channel.	TS.2: Tampering on internal perception data and DENM in the channel.	2
R.4	Exploitation of software weaknesses on the ECUs	TS.4: Exploitation of software weaknesses on the ECUs	2
R.7	Tampering through communication channels	TS.7: Tampering through communication channels	2
R.9	Spoofing on perception components	TS.9: Spoofing on perception components	3

From the list of risks, R.9 will be expanded on due to its higher risk level using Equation 4 and Equation 5 assuming b_{bt} as 0.2.

From Equation 4 we have:

$$\Delta = \frac{1 - b_{bt}}{5} = \frac{1 - 0.2}{5} = 0.16$$

with the interval between levels of 0.16, we can calculate b_t using Equation 5:

$$\begin{aligned}
 b_t &= b_{bt} + ((R_{max}(F, I) - 1) \times \Delta) \\
 b_t &= 0.2 + ((3 - 1) \times 0.16) \\
 b_t &= 0.52
 \end{aligned}$$

which means that for the integrity of the sensor data to be considered trustworthy regarding belief, the actual belief must be at least 0.52.

► Scope 2: Integrity of the DENM and CAM received by ADAS.

Similarly, Table 8 shows the list of relevant risks to be considered for Scope 2, including those against the integrity of the V2X messages.

Table 8 List of relevant risks for Scope 2

Name	Title	Caused by	Risk level
R.1	Spoofing of ExtVehicle-OBU channel. and data flow.	TS.1: Spoofing of ExtVehicle-OBU channel and data flow.	2
R.2	Tampering on internal perception data and DENM in the channel.	TS.2: Tampering on internal perception data and DENM in the channel.	2
R.3	Man-in-the-Middle Attack on ExtVehicle-OBU channel.	TS.3: Man-in-the-Middle Attack on ExtVehicle-OBU channel.	1
R.4	Exploitation of software weaknesses on the ECUs	TS.4: Exploitation of software weaknesses on the ECUs	2
R.5	Tampering on ExternalVehicle-OBU channel.	TS.5: Tampering on ExternalVehicle-OBU channel.	2
R.7	Tampering through communication channels	TS.7: Tampering through communication channels	2
R.13	Tampering of TxV	TS.13: Tampering of TxV	3

From the list of risks, we can observe that R.13 is the worst case, with risk level at 3. Since b_{bt} is not mandatory and the system's dependence on V2X data is limited, we can set the baseline to 0.

From Equation 4 we have:

$$\Delta = \frac{1 - b_{bt}}{5} = \frac{1 - 0.0}{5} = 0.2$$

with the interval between levels is 0.2, we can calculate b_t using Equation 5:

$$\begin{aligned}
 b_t &= b_{bt} + ((R_{max}(F, I) - 1) \times \Delta) \\
 b_t &= 0.0 + ((3 - 1) \times 0.2) \\
 b_t &= 0.4
 \end{aligned}$$

which means that for the integrity of the DENM and CAM received by ADAS to be considered trustworthy regarding belief, the actual belief must be at least 0.4.

- Scope 3: Availability of cooperative safety messages (DENM and CAM) to be received by ADAS.

Table 9, reproduced from the 5G Automotive Association's technical report on safety treatment in connected and automated driving (2021) [13], details the HARA outcomes for the scenario where a cooperative safety message transmission fails.

Table 9 HARA for a message not sent when needed [13]

Hazard Category	Exposure	Severity	Controllability	ASIL rating (possible range)
Message not sent when should be sent	<ul style="list-style-type: none"> ► Highway driving at relatively high speed in busy traffic occurs > 10% of time. ► Cars following relatively closely behind occurs > 10% of time. ► But emergency brake events are rare (assume less than once a year, or a few times a year) <p>Classification: Exposure is low: E1-E2</p>	<ul style="list-style-type: none"> ► Rear-ending on a highway could cause life-threatening injuries, or worse. <p>Classification: S2-S3 (depends on speed of impact)</p>	<p>Human drivers have to rely on their own senses, which means the emergency braking of vehicles in front must be visible. Given that the operational scenario is one where the highway is assumed to be busy, this means controllability will be limited.</p> <p>Classification: C3</p>	<p>QM→B</p> <p>E1-S2-C3=QM</p> <p>E2-S3-C3=B</p> <p>(Indicative values)</p>

From Equation 4, considering b_{bt} as 0, we have

$$\Delta = \frac{1 - b_{bt}}{5} = \frac{1 - 0.0}{5} = 0.2$$

considering the scenario where exposure is E1, severity is S2 and controllability is C3, resulting in ASIL QM

and with $R_{max}(E, S, C) = QM = 1$, we can calculate b_t using Equation 5:

$$b_t = b_{bt} + ((R_{max}(E, S, C) - 1) \times \Delta)$$

$$b_t = 0.0 + ((1 - 1) \times 0.2)$$

$$b_t = 0.0$$

On the other hand, if we consider the scenario where exposure is E2, severity is S3 and controllability is C3, resulting in ASIL B.

With $R_{max}(E, S, C) = B = 3$, we can calculate b_t using Equation 5:

$$\begin{aligned} b_t &= b_{bt} + ((R_{max}(E, S, C) - 1) \times \Delta) \\ b_t &= 0.0 + ((3 - 1) \times 0.2) \\ b_t &= 0.4 \end{aligned}$$

For the first case, as QM level does not dictate safety measures, the related b_t reflects it by not requiring a belief level for safety in this scope. On the other hand, the second case, where we have an ASIL B situation, it requires b_t of at least 0.4 to be considered trustworthy.

Disbelief Threshold Calculation Method

As previously introduced, the method to calculate the disbelief threshold (d_t) uses an impact rating. The impact rating can be evaluated using external frameworks such as Failure Mode and Effects Analysis (FMEA) for evaluating operational impacts, or the gathered impact rating within the used risk assessment.

Based on the scope, each impact category (safety, economic, operation/function, and privacy) needs to be weighted by relevance, we suggest the sum of the weights be 1. Next, we calculate the weighted impact rating (I_w), considering the weights for each category applying Equation 6:

$$I_w = \sum_{n=1}^4 I_n W_n \quad (\text{Eq. 6})$$

where

- ▶ I_w represents the weighted impact rating
- ▶ I_n represents the impact ratings (I_1 = safety, I_2 = financial, I_3 = operational and I_4 = privacy)
- ▶ W_n represents the given weights for each impact category (W_1 = safety, W_2 = financial, W_3 = operational and W_4 = privacy). And $\sum_{n=1}^4 W_n = 1$

The higher the potential impact, the lower the accepted disbelief should be accepted, so d_t is inversely proportional to the impact level and it is expressed by Equation 7:

$$d_t = 1 - I_w \quad (\text{Eq. 7})$$

In the event of several/multiple damage scenarios implicated in deriving impact ratings, we suggest using a worst-case scenario, selecting the damage scenario with the highest impact rating to be the basis of this analysis.

When it concerns impact related to security, we can translate the impact rating to a numerical value as suggested in Table 10. We adapted the values suggested by the example in ISO/SAE 21434 [10], to range between 0 and 1. From the safety perspective, the table also shows the suggested equivalence for severity and controllability ratings.

Table 10 Translation of impact rating, severity level and controllability level to a numerical value

Impact rating (I)	Severity (S)	Controllability (C)	Numerical value for impact rating
Negligible	S0	C0	0
Moderate	S1	C1	0.5
Major	S2	C2	0.75
Severe	S3	C3	1

In the event of a different number of impact ratings defined by the used impact assessment framework, this translation needs to be adapted accordingly.

When using TARA, Equation 6 and Equation 7 can be used directly, as suggested. For safety, using HARA, we first need to calculate the impact based on the severity and controllability levels, as shown in Table 11.

Table 11 Severity-controllability matrix

			Severity level			
			S0	S1	S2	S3
			0	0.5	0.75	1
Controllability level	C3	1	0	0.5	0.75	1
	C2	0.75	0	0.37	0.56	0.75
	C1	0.5	0	0.25	0.37	0.5
	C0	0	0	0	0	0

Another difference is that for safety we consider only safety impact, and, with that, Equation 6 is replaced by the corresponding value in the Table 11, which represents I_w for safety.

Disbelief Threshold Calculation for the Use Case

To calculate the maximum accepted disbelief expressed by d_t , we observe from the performed TARA (Annex A) the relevant damage scenarios for the scope, as the impact ratings are defined there. In addition, we weight these impact ratings by relevance for each scope. After filtering and weighting the impact categories, we can calculate d_t using Equation 6 and Equation 7. Following that, we perform the d_t calculation for each scope.

- Scope 1: Integrity of the in-vehicle sensor data (direction, position, speed, camera, braking, clock).

Table 12 List of relevant damage scenarios and their respective impact ratings for Scope 1

Name	Title	Concerns	Impact safety	Impact economic	Impact operation/function	Impact privacy
DS.1	Perception data is manipulated and cause malfunctioning	I: Dir_S I: Pos_S I: Cm_S I: Speed_S I: Clock_S I: Brake_S	Severe (1)	Moderate (0.5)	Major (0.75)	Negligible (0)
DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system	I: DF.7 I: DF.8 I: DF.10 I: DF.11 I: DF.13 I: DF.12 I: DF.9	Major (0.75)	Moderate (0.5)	Major (0.75)	Negligible (0)
		Weight	0.5	0.2	0.3	0.0

As DS.1 has a higher impact rating, we select it to be the basis of the calculation. Using Equation 6:

$$I_w = \sum_{n=1}^4 I_n W_n = 1 \times 0.5 + 0.5 \times 0.2 + 0.75 \times 0.3 + 0.0 \times 0.0$$

$$I_w = \sum_{n=1}^4 I_n W_n = 0.825$$

then, using Equation 7:

$$d_t = 1 - I_w = 1 - 0.825$$

$$d_t = 0.175$$

which means that for the integrity of the sensor data to be considered trustworthy regarding disbelief, the actual disbelief must be at maximum 0.175.

► Scope 2: Integrity of the DENM and CAM received by ADAS.

Table 13 List of relevant damage scenarios and their respective impact ratings for Scope 1

Name	Title	Concerns	Impact safety	Impact economic	Impact operation/function	Impact privacy
DS.5	RxV receives V2X messages with fake location	I: D.DENM I: D.CAM	Negligible (0)	Negligible (0)	Moderate (0.5)	Negligible (0)
		Weight	0.4	0.0	0.6	0.0

Using Equation 6.

$$I_w = \sum_{n=1}^4 I_n W_n = 0 \times 0.4 + 0.0 \times 0.0 + 0.5 \times 0.6 + 0.0 \times 0.0$$

$$I_w = \sum_{n=1}^4 I_n W_n = 0.3$$

then, using Equation 7.

$$d_t = 1 - I_w = 1 - 0.3$$

$$d_t = 0.70$$

which means that for the DENM and CAM received by ADAS to be considered trustworthy when disbelief is considered, the system must maintain a maximum belief level of 0.70. This less stringent requirement is justified by the lower potential impact of compromised messages in this scope. ADAS primarily relies on own sensors for operation, reducing the dependency on V2X messages.

- Scope 3: Availability of cooperative safety messages (DENM and CAM) to be received by ADAS.

As we are analyzing a safety disbelief threshold, the weights for economic, privacy and operation/function impacts are 0, as we just consider the safety impact. Considering HARA, Table 11 find I_w , we consider severity and controllability: As shown in Table 9, we have two cases:

Considering Table 11, for S2 and C3, we have:

$$I_w = 0.75$$

then, using Equation 7.

$$d_t = 1 - I_w = 1 - 0.75$$

$$d_t = 0.25$$

For that case, the disbelief threshold is set as 0.25.

Considering Table 11, for S3 and C3, we have:

$$I_w = 1$$

then, using Equation 7.

$$d_t = 1 - I_w = 1 - 1$$

$$d_t = 0$$

For that case, the disbelief threshold is set as 0, which is the strictest value and does not allow any negative evidence.

Uncertainty Threshold Calculation Method

As previously introduced, the suggested factors to be considered for uncertainty threshold (u_t) are detectability and required assurance level. Table 14 exemplifies how these two factors might be combined to define the Uncertainty Acceptance Level (UAL), in a generic way.

Table 14 Example of how to derive uncertainty acceptance from required assurance level and detectability

		Required assurance level			
		Low	Moderate	High	Very high
Detect-ability	High	Very high	High	High	Moderate
	Moderate	High	Moderate	Moderate	Low
	Low	Moderate	Low	Low	Very low

From this table, we can observe that a system with high detectability and low required assurance level is expected to accept more uncertainty, while one with very high required assurance level and low detectability has a very low UAL. This table should always be created in accordance with each system and scope, and the criteria may differ based on the impact of the system under analysis and how flexible in terms of uncertain information it can be. The ratings for detectability or assurance level can also include more levels for greater granularity.

To establish an uncertainty threshold, we map the uncertainty acceptance level – ranging from very low to very high, depending on detectability and assurance – to a numerical scale. This mapping utilizes a baseline uncertainty (u_{bt}), which ranges from 0 to 1 and represents the upper limit of accepted uncertainty considering design and engineering decisions.

The UAL assumes values as follows: Very low: 1, Low: 2, Moderate: 3, High: 4, Very high: 5. To define u_t , it is necessary to translate the UAL into a threshold value, whose range is from 0 to 1. The simplest way is to map UAL proportionally to this interval, what we call UAL_{map} , by using Equation 8.

$$UAL_{map} = \frac{UAL}{5} \quad (\text{Eq. 8})$$

u_t assumes the value according to the following condition in Equation 9.

$$u_t = \begin{cases} UAL_{map}, & \text{if } UAL_{map} \leq u_{bt} \\ u_{bt}, & \text{if } UAL_{map} > u_{bt} \end{cases} \quad (\text{Eq. 9})$$

From a cybersecurity perspective, the possible sources to derive detectability can be the effectiveness of the cybersecurity controls or incident detectors– i.e. intrusion detectors – implemented to detect such misbehavior, their system coverage and their accuracy. Required assurance level for automotive functions, on the other hand, can consider Cyber Security Assurance Level (CAL) as the basis. Currently, ISO/SAE 21434:2021 [10] defines CAL ranging from CAL1, where basic assurance and minimal tests are required, to CAL4, implying very high assurance and rigorous tests.

From a safety perspective, detectability can be derived from FMEA, for example, as it already includes detectability analysis, or from the vehicle's own diagnostic tools. The required assurance level, from a safety perspective, can be considered ASIL, which ranges from ASIL A (the lowest safety integrity level) to ASIL D (the highest one), in addition to QM where no special safety requirements are required. The higher the ASIL, the higher the assurance requirements.

Uncertainty Threshold Calculation for the Use Case

To calculate the maximum accepted uncertainty threshold expressed by u_t , we consider the assigned assurance level in the risk assessment and the ability to detect incidents within the system being analyzed. To illustrate this, we can take a look at the application of the method in the use case for each scope.

As a method for calculating detectability is not standardized, for this example we consider that the detectability rating is ranging from 0 to 1, where 0 is the lowest level of detectability and 1 the highest; in other words, if the system is assigned with 0 in detectability, it means that there are no detectors in the system and, in the event of an incident, the system is not aware of it and may be working with a compromised status. On the other hand, if a system is assigned a detectability of 1, it means that the system has incident detectors for all threat cases, and thus is considered to have fully efficient coverage.

In this example, the Cybersecurity Assurance Level, as defined by ISO/SAE 21434 [10], serves as the assurance level indicator, where CAL1, CAL2, CAL3 and CAL4 represent, respectively, low, moderate, high, and very high levels of cybersecurity assurance within the presented methodology. Table 15 outlines the Uncertainty Acceptance Levels, correlating them with assigned assurance levels and corresponding detectability. It also provides the rationale behind each detectability level's definition.

Table 15 UAL in the system based on assurance level and level of detectability of incidents from the automotive cybersecurity perspective

			Required assurance level			
			Low	Moderate	High	Very high
			CAL1	CAL2	CAL3	CAL4
Detectability	Detectors are present and provide the system full or high coverage and accuracy. They are able to identify the root cause of the security violation.	High	Very high	High	High	Moderate
	Detectors are present within the core system, capable of efficiently detecting cybersecurity incidents; however, their coverage is limited, they lack the capacity for root cause identification or exhibit moderate accuracy.	Moderate	High	Moderate	Moderate	Low
	Incident detectors are either absent or do not fully extend to the core system.	Low	Moderate	Low	Low	Very low

For issues related to safety, the detectability reflects the ability of the system to identify incidents that can lead to a safety risk, e.g., a sensor misbehaving. As an alternative, the required assurance level for safety can be extracted from ASIL, with the system's safety requirement level reflecting the assurance level. For that, Table 15 can be adapted as follows in Table 16.

Table 16 UAL in the system based on assurance level and level of detectability of incidents in the automotive safety domain

			Required assurance level				
			Very low	Low	Moderate	High	Very high
			QM	ASIL A	ASIL B	ASIL C	ASIL D
Detectability	Detectors are present and provide the system full or high coverage and accuracy. They are able to identify the root cause of the safety violation.	High	Very high	High	High	Moderate	Moderate
	Detectors are present within the core system, capable of efficiently detecting potential safety incidents caused by unreliable data; however, their coverage is limited, they lack the capacity for root cause identification or exhibit moderate accuracy.	Moderate	High	Moderate	Moderate	Moderate	Low
	Incident detectors are either absent or do not fully extend to the core system.	Low	Moderate	Low	Low	Very low	Very low

For each defined scope, we perform the method as follows.

- Scope 1: Integrity of the in-vehicle sensor data (direction, position, speed, camera, braking, clock).

For this example, the engineering team is considered to have assigned the cybersecurity assurance level as CAL3, which means that a high cybersecurity assurance is required and activities, such as analysis, testing and searching for vulnerabilities, are based on

explorative methods, and the cybersecurity assessment is performed by a different team.

In addition, it considers that the system has no ability to identify whether an attack is occurring, or a vulnerability is being exploited, resulting in a low detectability rating.

In view of these two conditions – high assurance level and low detectability – the system has a low acceptance of uncertainty. Using Equation 8 to calculate the uncertainty threshold, we have:

The uncertainty threshold is 0.4.

$$\text{Low UAL: } UAL_{map} = \frac{UAL}{5} = \frac{2}{5} = 0.4$$

- Scope 2: Integrity of the DENM and CAM received by ADAS.

For this example, the engineering team is considered to have assigned the cybersecurity assurance level as CAL2, which means that a moderate cybersecurity assurance is required and activities, such as analysis, testing and searching for vulnerabilities, are based on already-known information and the cybersecurity assessment is performed by a different person than the originator, but within the same team.

In addition, the system detects misbehavior and checks data plausibility using its sensors but cannot identify the root cause.

Considering these two conditions – moderate assurance level and moderate detectability – the system has moderate acceptance of uncertainty. Using Equation 8 to calculate the uncertainty threshold, we have:

$$\text{Moderate UAL: } UAL_{map} = \frac{UAL}{5} = \frac{3}{5} = 0.6$$

The uncertainty threshold is 0.6.

- Scope 3: Availability of cooperative safety messages (DENM and CAM) to be received by ADAS

As discussed before and shown in Table 9, the assigned ASIL for this scope is QM or ASIL B. As with the previous scope, for this example we need to make an assumption that the system is equipped with misbehavior detectors that can evaluate the cooperative safety messages and compare their claims with the in-vehicle sensors, to check their plausibility. From this, this level of detectability can be considered as moderate because the detectors are not able to check the cause of the problem.

Using Table 16, we can observe that for assurance as QM and moderate detectability, UAL is high (4). Where the assurance level is ASIL B, UAL is moderate (3). In that situation the respective uncertainty thresholds are calculate using Equation 8:

$$\begin{aligned} \text{High UAL : } UAL_{map} &= \frac{UAL}{5} = \frac{4}{5} = 0.8 \\ \text{Moderate UAL: } UAL_{map} &= \frac{UAL}{5} = \frac{3}{5} = 0.6 \end{aligned}$$

In situations where ASIL is QM, the uncertainty threshold is 0.8, and when it is moderate, the uncertainty threshold is 0.6.

Possible Behaviors of RxV with ATL and RTL

When the TxV with activated EEBL transmits CAM and DENM, the receiving RxV verifies the messages. Then, the RxV assesses the situation for decision-making to utilize AEB. To support the situation assessment, ATLs for data in the messages are calculated and compared with the corresponding RTLs defined by the manufacturer. At this point, the following passages show the possible RxV behaviors with the ATL and RTL calculated in the previous section. For example, if the RxV assesses the integrity of the TxV's position data, we use RTL (0.52, 0.175, 0.4) for Scope 1.

The calculated ATL refers to whether the position data received from TxV is trustworthy, particularly from the perspective of data integrity. The following table presents two scenarios where the ATL of Scenario 1, which includes two pieces of positive evidence and one for negative evidence, is (0.66, 0.11, 0.23), while the ATL of Scenario 2, which includes three pieces of positive evidence, is (0.77, 0, 0.22). To explain the possible behaviors of RxV in various cases, additional scenarios are presented here:

- Scenario 3: If there is one piece of positive evidence and two negative incidences for ω_X^A ($r_x = 1, s_x = 2$), ω_X^A is (0.2, 0.4, 0.4).
- Scenario 4: If there are three pieces of negative evidence for ω_X^A ($r_x = 0, s_x = 3$), ω_X^A is (0, 0.6, 0.4).
- Scenario 5: If the evidence is negative for ω_B^A , ω_B^A is (0, 0.33, 0.67).

Table 17 Summary of all ATLs in the described scenarios

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
ω_X^A	(0.4, 0.2, 0.4)	(0.6, 0, 0.4)	(0.2, 0.4, 0.4)	(0, 0.6, 0.4)	(0.4, 0.2, 0.4)
ω_B^A	(0.33, 0, 0.67)	(0.33, 0, 0.67)	(0.33, 0, 0.67)	(0.33, 0, 0.67)	(0, 0.33, 0.67)
ω_X^B	(1, 0, 0)	(1, 0, 0)	(1, 0, 0)	(1, 0, 0)	(1, 0, 0)
$\omega_B^A \otimes \omega_X^B$	(0.66, 0, 0.34)	(0.66, 0, 0.34)	(0.66, 0, 0.34)	(0.66, 0, 0.34)	(0, 0.33, 0.67)
ATL= $\omega_X^A \oplus (\omega_B^A \otimes \omega_X^B)$	(0.66, 0.11, 0.23)	(0.77, 0, 0.22)	(0.43, 0.2, 0.37)	(0.33, 0.3, 0.37)	(0.2, 0.265, 0.535)

The follow graphs in Figure 11 show the ATLs on the subjective logic triangle with the RTL.

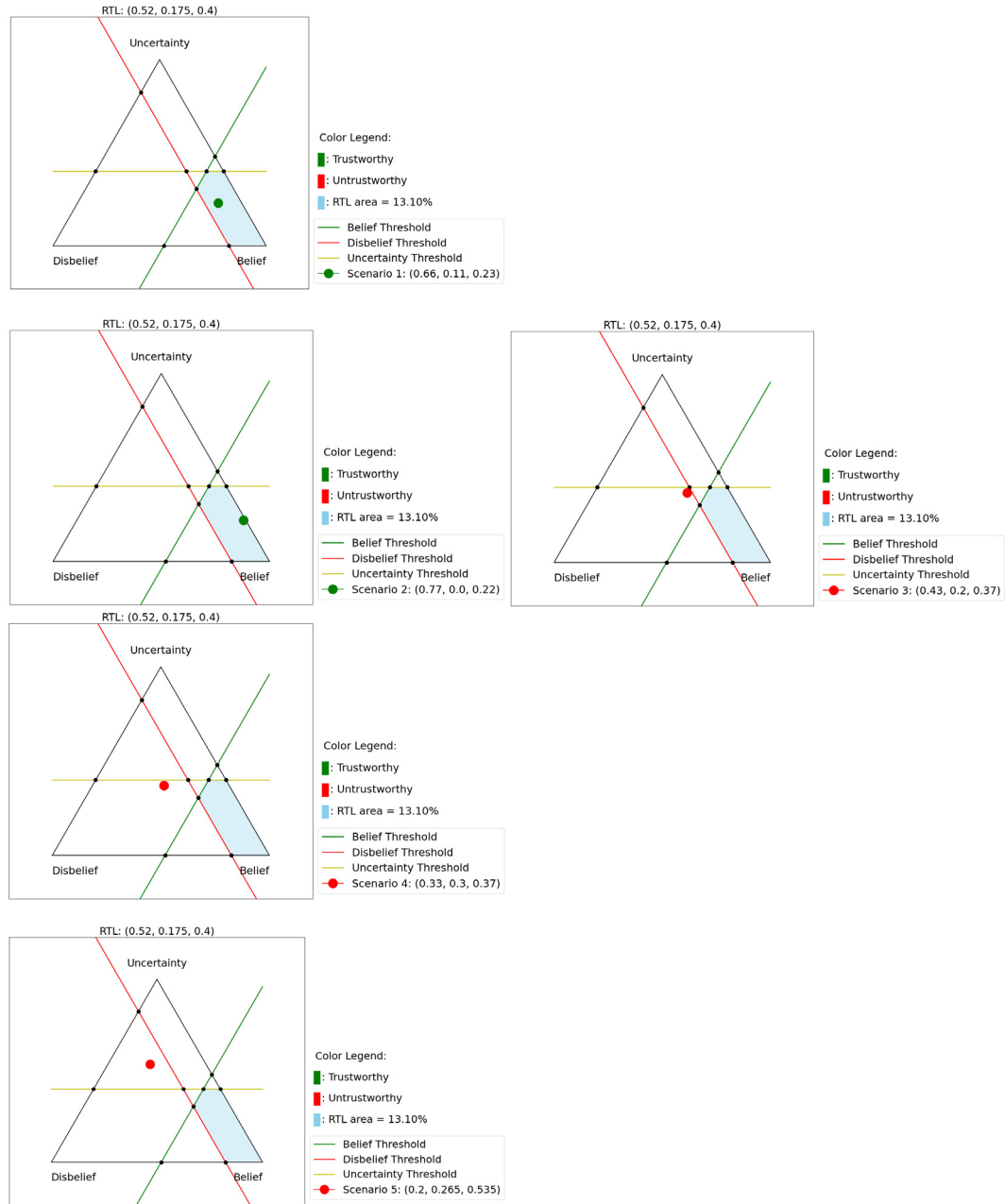


Figure 11 Visualization of each scenario within the subject logic triangle

After comparing the ATL and the RTL, the RxV can decide how to use the received position information from the TxV to activate AEB. In the AEB use case described earlier, single-stage and multi-stage braking strategies are considered. For AEB with a single-stage braking strategy, the possible behavior of RxV could be simple. For example, if RxV obtains ATL of Scenario 1 or Scenario 2, AEB can be activated because both ATLs satisfy RTL. On the other hand, if RxV obtains or responds to the ATL of Scenario 3, 4, or 5 that do not satisfy RTL, RxV can ignore this information or use it only as a display notice to the driver.

For AEB with a multi-stage braking strategy, the AEB system could adapt differently

depending on the ATL value, because ATLs are positioned differently and have different distances from the RTL boundary, as shown in the described subjective logic triangles. For example, if RxV receives the most trustable information and calculates the corresponding ATL value, as in Scenario 1, it could use the information to actuate the braking function within a large range (e.g., maximum full braking pressure or deceleration of 8m/s^2). If RxV receives less trustable information and calculates the corresponding ATL value – even though the ATL satisfies the RTL as in Scenario 2 – RxV could activate AEB within a smaller actuating range (e.g., maximum half braking pressure or deceleration of 4m/s^2). For example, in Scenario 1, the RxV could activate AEB without additionally verifying the position information received by its own sensors, whereas in Scenario 2 the RxV could slow down until it verifies the information from its own sensors. In cases where ATL does not satisfy the RTL, such as Scenario 3, 4 and 5, RxV could also use the information differently. For Scenario 3, RxV could activate AEB within the minimum actuating range (e.g., up to quarter braking pressure or deceleration of 2m/s^2), because the ATL is located close to the RTL boundary. For Scenario 4, RxV could opt to only notify the driver (display the information) without activating AEB. For Scenario 5, RxV does not have to use the information.

6 Open Topics and Gap Analysis

Open Technical Questions

Specifying the ATL Expression

The starting point of the ATL methodology is to identify the optimal set of accumulators and aggregate all relevant trust opinions related to a target proposition. At the same time, in a dynamic environment, such as the one examined in the running example, it is essential to update the overall ATL expression when a new behavior is observed in the system. Thus, the first question to be answered when instantiating the ATL methodology is: How do we model trust relationships of a target system in a systematic way so as to construct the equations for the ATL?

As previously mentioned, there are various strategies to achieve this. Regardless of the strategy employed, it is clear that identifying the target trust proposition heavily affects the incidences and types of evidence to be collected which, when calculated, results in the trust expression and final ATL. In fact, as demonstrated in the evaluation, the accuracy of the ATL is directly dependent on the granularity level of trust propositions or their atomic make-up. In an ideal ‘atomic trust’ proposition, quantification should not require evidence aggregation, as each trust opinion is based on a single evidence type. An approach to achieve this low-level breakdown would be to define one atomic trust proposition per attack vector mapped to the enforcement of a specific security control. Depending on the threat model and the set of attacks under consideration, the atomic trust propositions could be combined into composite trust propositions. Consequently, a key challenge when designing a trust model can be summarized in the following hypothesis: What is the proper level of ‘atomicity’ for the target trust proposition on a trust property?

Finally, there are different strategies in the literature to systematically model trust relationships, one of which is the trust model representation. Each strategy comes with its own inherent complexities, though this work focusses on those related to the accurate construction of the subjective logic trust network of the target system.

Quantifying Uncertainty in ATL

Another core challenge in the systematic modeling of the ATL methodology, is how to best manifest the quantification of uncertainty in the subjective trust modeling. As previously demonstrated, this constitutes the second core pillar dictating ATL accuracy. Uncertainty in this context primarily arises from two sources: the atomic trust opinions and expressions used to derive the ATL. In this context, the interpretation of uncertainty hinges on multiple factors, including the relevance of trust sources, suitability of the selected method, and nature of the input data used by these methods. We can categorize the sources of uncertainty into three main types:

1. Uncertainty due to limited evidence: This type of uncertainty reflects the fact that we have insufficient or incomplete information. It is typically expressed in equations by incorporating prior weight information (W) in Equation 1.

2. Uncertainty in the quantification process: This arises from the inherent limitations or imprecision in the process used to quantify atomic trust opinions. This uncertainty can be caused by the methods or algorithms used in the evaluation process.
3. Uncertainty associated with each piece of evidence: In the case of atomic trust opinions, evidence is generally binary – either positive or negative. However, in practice, the confidence in each piece of evidence can vary. A more confident piece might have a greater influence on the trust level than one with lower confidence. Addressing this variability maintains the binary nature of evidence while accounting for differing levels of certainty.

While these three sources of uncertainty have been identified, challenges remain in their precise quantification:

- (i) Prior weight: The non-informative prior weight (W) can be assigned heuristically. However, determining an exact quantification for (W) remains an open challenge.
- (ii) Uncertainty in quantification: Different methods (e.g., Bayesian, fuzzy logic) and various trust sources can be used to evaluate the atomic trust opinion. Since uncertainties arise from multiple factors, how can it be systematically quantified? This remains an open question.
- (iii) Uncertainty in evidence confidence: While trust evidence is binary, confidence in each piece of evidence varies. Quantifying this confidence as uncertainty is non-trivial and remains an unresolved issue.

Comparison of RTL and ATL

Moving from the ATL methodology towards a complete Trust Assessment Framework, it is important to express the trust requirements in a way that allows comparison of a computed ATL opinion with defined RTL constraints. From the running example, it is clear that the same pieces of evidence have different impact in organizations with different trust requirements, namely different RTL belief threshold and uncertainty modeling. In principle, the derivation/deviation of a trust decision is intrinsically linked to the accepted risk that the trustor is willing to take, with respect to the fact that a trustee behaves as expected in a given context. As part of a holistic risk assessment framework, the implementation of all security controls aims to reduce the overall risk to an accepted level. This could be the common denominator for deriving, on one hand, the RTL (e.g., maximum level of risk that can be considered as accepted) and, on the other hand, the ATL value based on evidence indicating the enforcement of the specified security controls and/or the residual risk remaining at accepted levels. Therefore, to establish a framework for assessing trust, the following question needs to be addressed: How can both ATL value and RTL constraints be expressed in a way that reflects common knowledge, enabling comparison and the derivation of trust decisions?

This is particularly relevant from a 'decision perspective' where trust-aware decision-making is based on the trustor's own verification policies of certain metrics and/or attributes that the potential trustee presents. As mentioned above, such evidence is essentially a set of verifiable claims, self-issued by the trustee system, that informs whether the system is reliable or operating 'as expected', according to design and policy

– doing what is required despite disruptions, errors, and attacks. However, manifesting the trust decision (through the trust opinions) on security policy conformance, comprising an optimal set of security controls and mechanisms that can minimize the impact of identified critical threats, hinges on the completeness of the evidence. Whether one views them as mere standalone tools sufficient to determine the correct (or not) enforcement of a security control – e.g., positive or negative evidence/input in software stack configuration and integrity – has a direct impact on overall system security.

However, such matters do not follow an either/or model, which is why a more nuanced view of the dependencies between composable security mechanisms is needed. For instance, while providing strong integrity guarantees, the verification and validation of runtime operational safeguards, such as Control-Flow Integrity and/or Control-Flow Attestation, can negate (or significantly impact) system dependability by violating safety and availability requirements. This is because they introduce additional attack vectors, such as the exploitation of the dynamically defined control transfers occurring during the tracing of an operation of interest leading to control-flow violations. Thus, the question arises: How can these dependencies be captured and transferred into the subjective logic model in order to correctly calculate the ATL?

Direct and transitive security dependencies can be modeled as (dis-)beliefs on the impact of each security control, and its associated evidence, in the overall trust expression or as opinions affecting the confidence level of employed positive security controls. Whatever approach is adopted, this needs to be mirrored in both ATL and RTL definitions so as to enable their verification, validation, and evaluation.

Federation of Trust Assessment

Let us use the term Trust Assessment Framework (TAF) to refer to a modular framework that operationalize our methodology of trust computation by connecting evidence to trust properties, organizing this information into structured trust expressions, and evaluating these expressions to derive the ATL. That is, the TAF formalizes and abstracts the process already described in the previous sections, providing a clear interface for instantiating different trust models and decision rules across use cases.

In distributed systems, each agent may host its own TAF instance, performing localized trust assessment based on available evidence and context. However, no single agent typically has full visibility over all trust-relevant information. To improve coverage and robustness, TAFs can engage in federated trust assessment, which can occur at various stages. One key stage is the creation of the trust model. Since an individual TAF may have an incomplete or limited perspective, collaborating with other TAFs enables a more comprehensive and accurate trust model. Another crucial aspect is evidence or opinion-sharing. After constructing the trust model, TAFs can exchange trust-related evidence, observations, or subjective opinions to refine their assessments.

In most multi-agent systems, the trustor and the trustee are distinct agents. Now, consider a scenario where two TAFs operate within Agent 1 and Agent 2 respectively. It is important to note that not every node in a Trust Model necessarily hosts a TAF, as resource constraints may limit their deployment. However, the described scenario can also be extended to cases where local TAFs operate within sub-networks, assessing trust at a more granular level. In our scenario of Agent 1 and Agent 2, the challenge is

that the evidence required for Agent 2's evaluation is generated on/within Agent 1 itself (e.g., attestation evidence about the sensor), meaning the TAF in Agent 2 does not have direct access to that evidence. The question and challenge then is how to handle and exchange trust evidence between federated TAFs.

The most straightforward approach is to transmit the raw trust evidence from Agent 1 to Agent 2. However, this evidence often contains sensitive information, making direct sharing a risky proposition due to data leaks or unauthorized access. In addition, the transmission of evidence between agents leads to high communication overhead, increasing bandwidth usage, which may not be available.

A better approach is to enable TAFs to exchange trust opinions instead of raw evidence. In that case, a key challenge is to ensure trust consistency and interpretability across federated entities. Since each TAF generates its own trust opinion based on locally available evidence and specific trust models, differences in opinion computation, trust model(s), or contextual assumptions can lead to inconsistencies in trust assessments. Beyond simply sharing trust opinions, a TAF can also convey the underlying trust expression used in its evaluation. This not only facilitates better interoperability but also enhances the explainability of trust assessments, allowing other TAFs to understand the logic behind the computed trust values.

Despite these potential solutions, challenges remain in ensuring reliable and efficient trust assessment across federated entities:

- (i) Sharing of evidence: One of the primary concerns in a federated system is how evidence is shared between TAFs. Sharing more evidence can enhance the reliability of the trust assessment by increasing the number of data sources. However, this raises important privacy concerns. Some types of evidence may contain sensitive information that could potentially be exposed during the sharing process. It is critical to establish mechanisms to ensure that privacy is maintained, and that only non-sensitive or anonymized evidence is shared.
- (ii) Latency introduced by sharing evidence: While federating evidence can improve trust assessment, it may also introduce additional latency. The process of gathering, verifying, and sharing evidence between different TAFs can delay the overall assessment. Therefore, it is crucial to balance the desire for more evidence with the need for real-time or near-real-time trust evaluations.
- (iii) Resource-sharing and delegation: Federation can also involve the sharing of resources between TAFs. For example, one TAF (TAF1) might delegate the computation of certain trust evaluations to another TAF in order to enhance system efficiency. This delegation can help distribute computational workloads and ensure that trust assessments are performed faster and more effectively, especially when dealing with complex or large-scale systems. However, this raises the challenge of deciding which tasks to delegate, as well as how to coordinate resource-sharing without introducing inefficiencies or excessive interdependencies between TAFs.

Standardization Gap Analysis

The following passages explain the approaches of standardization bodies in addressing trust and trustworthiness.

ISO

ISO has established a robust framework for addressing trust and trustworthiness across various domains, integrating these principles into information security, interoperability, and emerging technologies. Key committees such as ISO/IEC JTC 1/SC 27 (Information Security, Cybersecurity, and Privacy Protection) and ISO/IEC JTC 1/SC 41 (Internet of Things and Digital Twin) serve as focal points for these efforts, harmonizing work across horizontal (cross-cutting) and vertical domains.

Trustworthiness in ISO is formalized through standards like ISO/IEC TS 5723:2022 which provide a common vocabulary for trust-related characteristics, ensuring consistency across sectors. Trust is often tied to system properties like resilience, transparency, reliability, and security. For instance, in the Internet of Things (IoT) and digital twin systems, trustworthiness is embedded via frameworks, such as ISO/IEC 30147 and ISO/IEC TS 30149, which guide the integration of trustworthiness activities within system lifecycles.

At a practical level, ISO emphasizes levels of assurance for systems, devices, and processes; a concept that aligns trust metrics with lifecycle stages. For example, in the context of cybersecurity, ISO/IEC JTC 1/SC 27's standards, such as ISO/IEC 27001 (Information Security Management Systems) and ISO/IEC 15408 (Common Criteria for IT Security Evaluation), provide methodologies for assessing and certifying trust at different assurance levels.

More recently, Draft AWI 11034 from ISO/IEC JTC 1/SC 38/WG 3 extends the above approach to cloud computing, providing a framework for trustworthiness in cloud environments. This draft highlights the importance of quantifiable metrics and evidence to evaluate trustworthiness, tailored to stakeholders like Cloud Service Providers (CSPs) and Cloud Service Consumers (CSCs). The framework prioritizes stakeholder-specific trust metrics, emphasizing compliance, operational transparency, and risk management through measurable indices and structured processes. It introduces concepts such as levels of trustworthiness, aligning expected and actual performance through evidence-based approach, such as cryptographic proofs, service-level agreements, and attestation mechanisms.

ETSI

The ETSI ITS WG1 work item DTR/ITS-001964 (TR 103 917) is working on the pre-standardization study, 'Functional Safety Analysis'. A stable draft is planned for January 2026. The main goal is to describe the challenges for using V2X messages in safety-critical driving functions concerning not only Functional Safety (FuSa, ISO 26262) but also Safety of the Intended Functionality (SOTIF, ISO 21448). In that context, safety-critical means that the V2X driving function may lead to hazards, when automatically triggering actions based on received V2X messages. To reach this goal, different partners contribute safety-critical use cases with or without infrastructure ITS-Stations,

their requirements and the gaps identified in current V2X standards. ETSI's technical report seeks to facilitate a common understanding of the challenges and gaps, leading to follow-up (pre-)standardization activities that help define future solutions.

ETSI TC ITS WG5 has published several technical specifications that collectively establish the current framework for secure and trustworthy communication among vehicles and infrastructure components. The technical specification ETSI TS 102 940 [8], in particular, specifies the ITS communications security architecture and management. In this context, Misbehavior Detection is introduced as the functionality that performs checks on the incoming V2X messages; the Misbehavior Authority is a remote entity able to process Misbehavior Reports sent by the stations, with the aim of identifying stations that are sending incorrect data.

The technical specification ETSI TS 103 759 [9] introduces the Misbehavior Reporting Service, which allows a station to produce and send Misbehavior Reports to the Misbehavior Authority. The scope of this document is the specification of the format of the Misbehavior Report and of the dissemination protocol. Moreover, it contains the specification of some misbehavior detectors. Broadly speaking, Misbehavior Reports can be used as a trust source by the TAF.

ETSI NFV and ETSI SEC have been focusing on the topic of Trust in the context of Network Function Visualization (NFV). Through a series of reports, especially NFV-SEC 003 (2016), NFV-SEC 007 (2017) and, more recently, NFV-SEC 018 (2019), ETSI provides a foundation for ensuring trust across NFV components and operations, complementing 3GPP's functional-level attestation mechanisms.

ETSI introduces the Levels of Assurance (LoA) in order to provide a graded metric for evaluating the trustworthiness of NFV components. LoA quantifies confidence across operational phases: during boot (via Measured Boot), at runtime (through integrity checks and attestation), and at decommissioning (ensuring secure retirement). This dynamic approach surpasses binary trust decisions, aligning with the complexity of NFV ecosystems.

At the heart of ETSI's approach is the concept of Roots of Trust (RoT) and Chain of Trust (CoT), which underpin attestation mechanisms for verifying the integrity of NFVs and Virtual Network Functions (VNF). Remote attestation leverages RoTs and CoTs to verify the integrity of both physical and virtual components in an NFV deployment. ETSI integrates attestation workflows into NFV MANO frameworks, ensuring continuous trust validation during lifecycle operations.

By extending trust beyond functional Network Function (NF) attestation, ETSI complements 3GPP's focus on onboarding and registration. While 3GPP emphasizes functional trust mechanisms, ETSI's infrastructure-centric LoA model ensures continuous trust across administrative domains, lifecycle phases, and operational states. So, in that way, ETSI emphasizes the dynamic nature of trust management by embedding lifecycle-aware trust mechanisms into NFV operations, supported by graded assurance levels, ensuring both dynamic scalability and security.

3GPP

In the 3GPP trust model, all the network entities are assumed to be trusted, although there are two different levels of trust between the Random Access Network (RAN)

entities and the Core NFs. Due to the fact that RAN equipment is more exposed, Core NFs are assumed to be ‘more’ trusted compared to RAN nodes. The separation between these two levels is realized in the standards by independent security mechanisms for establishing secure communication between the device and the Core network versus between the device and the RAN node.

3GPP did not develop any standards for dynamic trust evaluation or trust establishment for NFs in the sense of the present document. 3GPP typically does not consider implementation aspects not relevant to interoperability or the communication protocols 3GPP develops. Nevertheless, 3GPP conducted two internal studies in relation to zero trust that did not produce any standards or normative requirements.

3GPP does provide mechanisms for security monitoring purposes in clause 7 of TS 33.501. Furthermore, 3GPP is planning in its next releases to specify data collection requirements that can be used for security and privacy purposes. Such mechanisms are enablers for security monitoring and, hence, any framework for dynamic trust evaluation. Though the latter has been considered out of 3GPP scope so far.

Identified Gaps

In order to promote clarity, interoperability, and industry-wide adoption of trustworthiness assessments for Connected and Automated Vehicles, we recommend the following standardization activities related to the Actual Trustworthiness Level methodology. These recommendations are structured into two main categories: Standardized procedures and standardized profiles.

Standardized Procedures

Standardized procedures are necessary to ensure consistent generation, evaluation, and evolution of trustworthiness assessments across diverse systems and contexts.

1. **Definition of Trust Model Templates (TMTs)** that capture the relevant Trust Objects, Trust Relationships for specific use cases. Each TMT should provide a structured design-time specification that can be instantiated into one or more Trust Model Instances at runtime, enabling consistent and interoperable trustworthiness assessment across different implementations and contexts.
 - a. **Introduce a versioning and metadata standard** for TMTs and TMIs in order to track updates to trust models over time, maintain backward compatibility.
2. **Representation of trust propositions**, including both atomic and composite forms. This means standardizing how trust propositions are formally defined, structured, and expressed within a trust model or trust assessment system. For that, we need to establish a common approach to defining trust propositions by associating them with well-defined trust properties (e.g., integrity, availability), and define guidelines for when to deconstruct complex propositions into atomic ones.

3. **For each trust source, standardize the quantification function** (e.g., Equation 1) that maps evidence into subjective logic opinions, including how belief, disbelief, and uncertainty are derived. This should include:
 - a. Agreed rules for assigning values to the non-informative prior weight W (baseline-prior quantification),
 - b. Consideration of confidence levels in evidence
4. **Dynamic trust evolution mechanisms:**
 - a. Standardize procedures for runtime ATL updates, including update triggers (e.g., new evidence, loss of communication), evidence sources used, and update algorithms. This would be crucial for ensuring consistent and reliable trust management in dynamic CAV environments.
 - b. Standardize procedures for the Trust Model evolution (e.g., onboarding/removal of nodes).
5. **Impact weighting:** Standardize guidance on how to assign weights based on their relevance to the scope for RTL definition. This procedure will detail the approach for determining and applying weights to different impact ratings (e.g., safety, economic, privacy and operation/function) to ensure consistency in the weighting process.
6. **Baseline threshold definition:** Standardize guidance for defining baseline thresholds enforced by external factors like type approval or company regulations, which are not covered by risk assessment. This procedure will outline the method for identifying, documenting, and incorporating these external constraints to establish baseline requirements for trustworthiness thresholds.

Standardized Profiles

Standardized profiles are needed to ensure that systems apply ATL methodology consistently across different environments, domains, and vehicle architectures.

1. Define a standardized and extensible catalogue of trust sources for each atomic proposition. This ensures consistency in evidence collection, improves traceability, and simplifies comparison across systems or domains. This could include a standardized taxonomy of trust sources and a common format for representing evidence claims.
2. **TMPs for specific use cases:** For common CAV use cases (e.g., Automated Emergency Braking, Cooperative Adaptive Cruise Control), define reference TMPs including mandatory and optional trust propositions and trust sources.
 - a. Profiles should allow instantiation flexibility but must maintain core proposition structures to ensure consistent risk evaluation.
3. **Set of standard subjective logic operators** to be used in expression evaluation. This includes:
 - a. Discounting operator.

- b. Fusion operators (cumulative, averaging, weighted) for combining opinions.
- c. Logical operators (AND, OR, Multiplication) for composing complex propositions.

The standard should also include profiles of operator sets that can be used based on use case context (e.g., independence of opinions, trust propagation scenarios).

4. **Profiles for evidence confidence weighting:** Define standard profiles for how evidence confidence should be weighted based on source type (e.g., sensor reliability classes, communication channel grades).

We outline the following key recommendations in relation profiles for RTL, but the metrics and scores can be specified as profiles specific to use cases or groups of use cases. Guidance is needed on how to use them. These profiles thus need to specify:

5. **Risk level translation and expected levels:** To establish measurable trustworthiness thresholds, standardize the translation of risk levels into numerical values and define consistent expected risk categories. This profile will define the mapping between qualitative risk levels (e.g., High, Medium, Low) and their corresponding numerical representations, as well as the expected categorization of these numerical values. Table 6 and Table 11 provide use case-specific examples which can be discussed in standardization. Standardization needs to provide guidance on how to identify use cases and suitable profiles.
6. **Evaluation matrices and rating:** Standardize the assessment matrices used to assess risks and acceptance levels. This includes defining consistent expected scores and ratings, as exemplified in Table 11, Table 14, Table 15, and Table 16, to support common understanding and interpretation of the evaluation outcomes. Each rating matrix will specify the criteria being assessed, the scales, and the resulting ratings with their clear interpretations.
7. **Rating dimension classification guidance:** Develop guidance for classifying core ratings, such as detectability and assurance level, and their corresponding rating criteria. This profile will provide criteria and descriptions for scoring levels (e.g., high, moderate, low for detectability and defined levels for assurance), supporting consistent understanding and definition across different companies or departments.

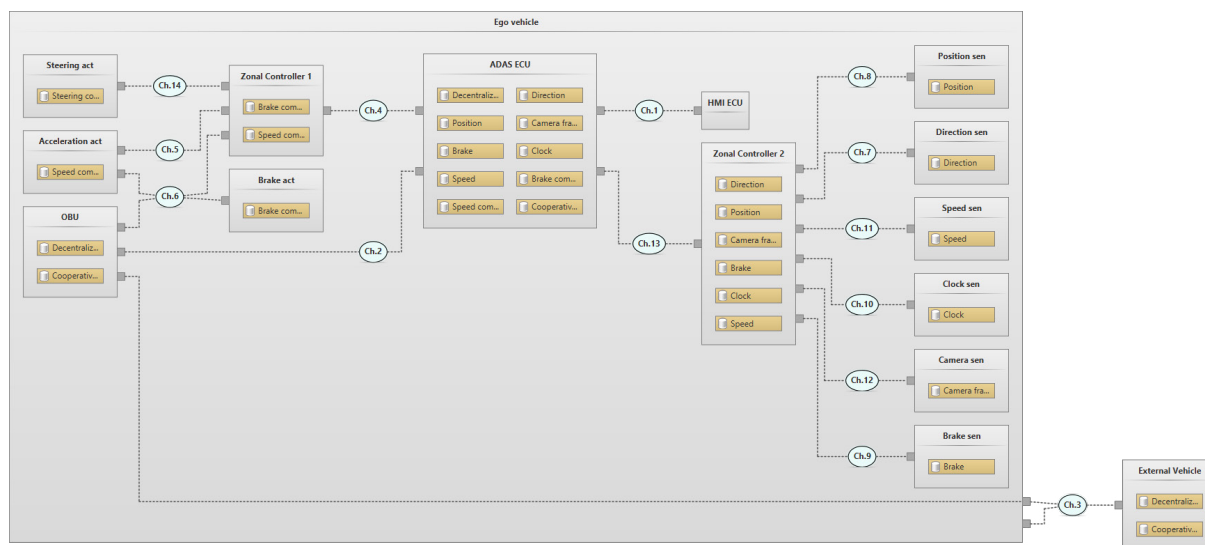
References

- [1] "Uniform provisions concerning the approval of motor vehicles with regard to the Advanced Emergency Braking Systems (AEBS) - Addendum: 130 - Regulation: 131", United Nations. 27 February 2014.
- [2] Yang L, Yang Y, Wu G, Zhao X, Fang S, Liao X, Wang R, Zhang M. (2022). A systematic review of autonomous emergency braking system: impact factor, technology, and performance evaluation. *Journal of advanced transportation*, 2022(1).
- [3] Naus, G. J., Vugts, R. P., Ploeg, J., van De Molengraft, M. J., & Steinbuch, M. (2010). String-stable CACC design and experimental validation: A frequency-domain approach. *IEEE Transactions on vehicular technology*, 59(9), 4268-4279.
- [4] Vinel, A., Lyamin, N., & Isachenkov, P. (2018). Modeling of V2V communications for C-ITS safety applications: A CPS perspective. *IEEE Communications Letters*, 22(8), 1600-1603.
- [5] Sidorenko, G., Thunberg, J., Sjöberg, K., Fedorov, A., & Vinel, A. (2021). Safety of automatic emergency braking in platooning. *IEEE Transactions on Vehicular Technology*, 71(3), 2319-2332.
- [6] Sidorenko, G., Plöger, D., Thunberg, J., & Vinel, A. (2022). Emergency braking with ACC: How much does V2V communication help?. *IEEE networking letters*, 4(3), 157-161.
- [7] 5GAA, "Creating Trust in Connected and Automated Vehicles", 2023.
- [8] ETSI, TS 102 940, "Intelligent Transport Systems (ITS); Security; ITS communications security architecture and security management; Release 2", 2021.
- [9] ETSI, TS 103 759, "Intelligent Transport Systems (ITS); Security; Misbehavior Reporting service; Release 2", 2023.
- [10] ISO/SAE. 21434 – road vehicles — cybersecurity engineering, 8 2021.
- [11] ISO 26262 – Road vehicles – Functional safety – Part 3: Concept phase, 12 2018.
- [12] de Lucena, A. R. F., Hermann, A., Trkulja, N., Kiening, A., Petrovska, A., & Kargl, F. (2024, October). Required Trustworthiness Level based on Threat Analysis and Risk Assessment (TARA). In 2024 IEEE Future Networks World Forum (FNWF) (pp. 519-526). IEEE.
- [13] 5G Automotive Association. (2021). "Safety Treatment in Connected and Automated Driving Functions Report", (Technical Report v1.0). Retrieved from https://5gaa.org/content/uploads/2021/07/5GAA_T-210009_STiCAD-TRv1.0_Final.pdf
- [14] 5G Automotive Association. (2024). "Safety Treatment in Connected and Automated Driving Functions – Phase 2" (Technical Report). Retrieved from <https://5gaa.org/content/uploads/2025/02/5gaa-wi-sticad-technical-report.pdf/>
- [15] CONNECT: Continuous and Efficient Cooperative Trust Management for Resilient CCAM (2025) [Funded project under the Grant Agreement No. 101069688]. European Commission, <https://horizon-connect.eu/>
- [16] CONNECT public deliverable D3.1, "Architectural Specification of CONNECT Trust Assessment Framework, Operation and Interaction", 2024.
- [17] CONNECT public deliverable D3.2, "CONNECT Trust & Risk Assessment and CAD Twinning Framework (Initial Version)", 2024.
- [18] Cho, J. H., & Adali, S. (2018). Is Uncertainty Always Bad?: Effect of Topic Competence on Uncertain Opinions. In 2018 IEEE International Conference on Communications (ICC) (pp. 1-7). IEEE.
- [19] Jøsang, A. (2016). Multinomial Multiplication and Division. In *Subjective Logic: A Formalism for Reasoning Under Uncertainty* (pp. 115-132). Cham: Springer.
- [20] Jøsang, A., Hayward, R., & Pope, S. (2006). Trust network analysis with subjective logic. In *Conference Proceedings of the Twenty-Ninth Australasian Computer Science Conference (ACSW 2006)* (pp. 85-94). Australian Computer Society.

- [21] Cerutti, F., Kaplan, L. M., Norman, T. J., Oren, N., & Toniolo, A. (2015). Subjective logic operators in trust assessment: an empirical study. *Information Systems Frontiers*, 17, 743-762.
- [22] Jøsang, A., Ažderska, T., & Marsh, S. (2012). Trust transitivity and conditional belief reasoning. In *Trust Management VI: 6th IFIP WG 11.11 International Conference, IFIPTM 2012, Surat, India, May 21-25, 2012. Proceedings 6* (pp. 68-83). Springer Berlin Heidelberg.
- [23] Agudo, I., Fernandez-Gago, C., & Lopez, J. (2008). A model for trust metrics analysis. In *International Conference on Trust, Privacy and Security in Digital Business* (pp. 28-37). Berlin, Heidelberg: Springer Berlin Heidelberg.
- [24] Wang, Y., & Vassileva, J. (2003). Bayesian network-based trust model. In *Proceedings IEEE/WIC international conference on web intelligence (WI 2003)* (pp. 372-378). IEEE.
- [25] Hryniowski, A., Wang, X. Y., & Wong, A. (2020). Where does trust break down? A quantitative trust analysis of deep neural networks via trust matrix and conditional trust densities. *arXiv preprint arXiv:2009.14701*.

Annex A: TARA Report of the Automated Emergency Brake Use Case

System Diagram of SYS: System



All system elements are listed at the end of this annex.

Assets and Damage Scenarios

Data

Data (Asset)		Security properties			Damage scenarios	
Name	Title	C	I	A	Name	Title
D.BrakeNotif	Braking notification	-	X	-	DS.9	Driver is confused by HMI output
D.Brake_S	Brake	X	-	-	DS.8	Sensor data can be accessed by an unauthorized party
D.CAM	Cooperative Awareness Messages	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	X	-	DS.5	RxV receives V2X messages with fake location
D.Camera_S	Camera frames	X	-	-	DS.8	Sensor data can be accessed by an unauthorized party
D.Clock_S	Clock	X	-	-	DS.8	Sensor data can be accessed by an unauthorized party
D.DENM	Decentralized Environmental Notification Message	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	X	-	DS.5	RxV receives V2X messages with fake location
D.Direc_S	Direction	X	-	-	DS.8	Sensor data can be accessed by an unauthorized party
D.Pos_S.	Position	X	-	-	DS.8	Sensor data can be accessed by an unauthorized party
D.Speed_S	Speed	X	-	-	DS.8	Sensor data can be accessed by an unauthorized party

Components

Component (Asset)		Security properties			Damage scenarios	
Name	Title	C	I	A	Name	Title
ADAS	ADAS ECU	-	-	X	DS.3	Critical control units stop responding and cause an accident
		-	X	-	DS.7	Firmware has been manipulated and cause function failure
Acc_A	Acceleration act	-	X	-	DS.2	Actuators do not perform the intended action (Acceleration or braking)
Brake_A	Brake act	-	X	-	DS.2	Actuators do not perform the intended action (Acceleration or braking)
		-	-	X	DS.3	Critical control units stop responding and cause an accident
		-	-	X	DS.4	RxV takes too long to stop and hits TxV
Brake_S	Brake sen	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	-	X	DS.10	Perception control units stop responding
Clock_S	Clock sen	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	-	X	DS.10	Perception control units stop responding
Cm_S	Camera sen	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	-	X	DS.10	Perception control units stop responding
Dir_S	Direction sen	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	-	X	DS.10	Perception control units stop responding
HMI	HMI ECU	-	X	-	DS.9	Driver is confused by HMI output
		-	-	X	DS.11	HMI stop responding and driver gets confused
OBU	OBU	-	-	X	DS.10	Perception control units stop responding

Component (Asset)		Security properties			Damage scenarios	
Name	Title	C	I	A	Name	Title
Pos_S	Position sen	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	-	X	DS.10	Perception control units stop responding
Speed_S	Speed sen	-	X	-	DS.1	Perception data is manipulated and cause malfunctioning
		-	-	X	DS.10	Perception control units stop responding
ZC1	Zonal Controller 1	-	-	X	DS.3	Critical control units stop responding and cause an accident
		-	X	-	DS.7	Firmware has been manipulated and cause function failure
ZC2	Zonal Controller 2	-	-	X	DS.10	Perception control units stop responding

Data Flows

Data Flow (Asset)		Security properties			Damage scenarios	
Name	Title	C	I	A	Name	Title
DF.1	D.1: OBU -> ADAS [Ethernet]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.2	D.1: ExternalVehicle -> OBU [mobile]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.3	D.7, D.8: ADAS -> ZC1 [Ethernet]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.4	D.9: ZC1 -> Acceleration act [FlexRay]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.5	D.8: ZC1 -> Brake act [FlexRay]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.6	D.2: Direction sen -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.7	D.3: Position sen -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.8	D.5: Brake sen -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.9	D.Speed_S: Brake_S -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.10	D.6: Clock sen -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.11	D.7: Speed sen -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.12	D.Brake_S: Speed_S -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.13	D.4: Camera sen -> ZC2 [CAN]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system
DF.14	D.2, D.3, D.4, D.5, D.6, D.7: ZC2 -> ADAS ECU [Ethernet]	-	X	-	DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system

Damage Scenarios Overview

Damage scenarios					Impact
Name	Title	Description	Concerns	IS	IL
DS.1	Perception data is manipulated and cause malfunctioning		I: Dir_S I: Pos_S I: Cm_S I: Speed_S I: Clock_S I: Brake_S I: D.DENM I: D.CAM	-	Severe
DS.2	Actuators do not perform the intended action (Acceleration or braking)		I: Acc_A I: Brake_A	-	Severe
DS.3	Critical control units stop responding and cause an accident		A: ADAS A: ZC1 A: Brake_A	-	Severe
DS.4	RxV takes too long to stop and hits TxV		A: Brake_A	-	Severe
DS.5	RxV receives V2X messages with fake location		I: D.DENM I: D.CAM	-	Major
DS.6	Perception data is manipulated in the channels and cause malfunctioning of the emergency braking system		I: DF.2 I: DF.1 I: DF.3 I: DF.4 I: DF.5 I: DF.6 I: DF.7 I: DF.8 I: DF.10 I: DF.11 I: DF.13 I: DF.14 I: DF.12 I: DF.9	-	Major
DS.7	Firmware has been manipulated and causes function failure		I: ADAS I: ZC1	-	Severe
DS.8	Sensor data can be accessed by an unauthorized party		C: D.Direc_S C: D.Pos_S. C: D.Camera_S C: D.Brake_S C: D.Clock_S C: D.Speed_S	-	Moderate
DS.9	Driver is confused by HMI output		I: HMI I: D.BrakeNotif	-	Moderate
DS.10	Perception control units stop responding		A: OBU A: ZC2 A: Dir_S A: Pos_S A: Cm_S A: Speed_S A: Clock_S A: Brake_S	-	Moderate
DS.11	HMI stop responding and driver gets confused		A: HMI	-	Moderate

Damage- and Threat Scenarios Table

Damage Scenario		Threat Scenarios	
Name	Title	Name	Title
DS.1	Perception data is manipulated and cause malfunctioning	TS.2	Tampering on internal perception data and DENM in the channel.
		TS.7	Tampering through communication channels
		TS.9	Spoofing on perception components
		TS.13	Tampering of TxV
DS.2	Actuators do not perform the intended action (acceleration or braking)	TS.7	Tampering through communication channels
		TS.8	Tampering on actuators (component and channel.)
DS.3	Critical control units stop responding and cause an accident	TS.10	Spoofing on actuators
		TS.2	Tampering on internal perception data and DENM in the channel.
		TS.4	Exploitation of software weaknesses on the ECUs
DS.4	RxV takes too long to stop and hits TxV	TS.2	Tampering on internal perception data and DENM in the channel.
		TS.4	Exploitation of software weaknesses on the ECUs
		TS.8	Tampering on actuators (component and channel)
		TS.9	Spoofing on perception components
		TS.10	Spoofing on actuators
		TS.12	Denial of Service on actuators
		TS.1	Spoofing of ExtVehicle-OBU channel. and data flow.
		TS.3	Man-in-the-Middle Attack on ExtVehicle-OBU channel.
DS.5	RxV receives V2X messages with fake location	TS.5	Tampering on ExternalVehicle-OBU channel.
		TS.13	Tampering of TxV
		TS.7	Tampering through communication channels
DS.6	Perception data is manipulated in the channels and causes malfunctioning of the emergency braking system		
DS.7	Firmware has been manipulated and causes function failure	TS.4	Exploitation of software weaknesses on the ECUs
		TS.6	Exploitation of software weaknesses on HMI ECU
DS.8	Sensor data can be accessed by an unauthorized party	TS.9	Spoofing on perception components
		TS.11	Information Disclosure on perception data through the channels
DS.9	Driver is confused by HMI output		
DS.10	Perception control units stop responding		
DS.11	HMI stop responding and driver gets confused		

Threat Scenarios and Attack Paths

The Feasibility Model is given in **gray** within the column 'Path'.

Name	Title	Path	Steps	AFL
TS.1	Spoofing of ExtVehicle-OBU ch. and data flow	AP1 Feasibility Model	AS.4: Spoofing - ExternalVehicle - OBU channel. AS.3: Man-in-the-middle attack to modify intercepted V2X messages AS.1: Interception of v2x messages	Very low
		AP2 Feasibility Model	AS.5: Perception component spoofing	Low
TS.2	Tampering on internal perception data and DENM in the Channel.	AP1 Feasibility Model	AS.7: Injection of malicious sensor data AS.6: Monitor sensors' output to understand normal behavior	Very low
TS.3	Man-in-the-Middle Attack on ExtVehicle-OBU Channel.	AP1 Feasibility Model	AS.3: Man-in-the-middle attack to modify intercepted V2X messages AS.1: Interception of v2x messages	Very low
TS.4	Exploitation of software weaknesses on the ECUs	AP1 Feasibility Model	AS.9: Exploitation of ECUs software weaknesses AS.8: Information gathering of the target ECU	Very low
TS.5	Tampering on ExternalVehicle-OBU Channel.	AP1 Feasibility Model	AS.10: Tampering on ExternalVehicle - OBU channel AS.2: Forge and send a V2X message that mimics intercepted messages	Low
			AS.1: Interception of v2x messages	
TS.6	Exploitation of software weaknesses on HMI ECU	AP1 Feasibility Model	AS.9: Exploitation of ECUs software weaknesses AS.8: Information gathering of the target ECU	Very low
TS.7	Tampering through communication channels	AP1 Feasibility Model	AS.12: Tampering through communication channels AS.11: Monitor exchanged message to understand its structures and patterns	Very low
TS.8	Tampering on actuators (Cmp and Channel.)	AP1 Feasibility Model	AS.14: Tampering on actuators AS.13: Interception of actuators commands	Very low
TS.9	Spoofing on perception components	AP1 Feasibility Model	AS.5: Perception component spoofing	Low
TS.10	Spoofing on actuators	AP1 Feasibility Model	AS.15: Spoofing actuators AS.13: Interception of actuators commands	Very Low
TS.11	Information Disclosure on perception data through the channels	AP1 Feasibility Model	AS.11: Monitor exchanged message to understand its structures and patterns	Very Low
		AP2 Feasibility Model	AS.6: Monitor sensors' output to understand normal behavior	Low
TS.12	Denial of Service on actuators	AP1 Feasibility Model	AS.17: Replay actuators commands out of context AS.13: Interception of actuators commands	Very Low
		AP2 Feasibility Model	AS.16: Bus flooding	Low

Name	Title	Path	Steps	AFL
TS.13	Tampering of TxV	AP1 Feasibility Model	AS.10: Tampering on ExternalVehicle - OBU Channel. AS.2: Forge and send a V2X message that mimics intercepted messages AS.1: Interception of v2x messages	Low

Assumptions Table

Name	Title	Description	Effect
------	-------	-------------	--------

Attack Steps Tables (Accumulated)

Tables Legend

- Black** rating means locally overridden.
Gray rating means derived from catalog class or attack tree children.

Name	Title	Description	ET	SE	KoIC	WoO	Eq	AFL
AS.1	Interception of v2x messages		ET1	SE1	KoIC0	WoO1	Eq0	High
AS.2	Forge and send a V2X message that mimics intercepted messages		ET2	SE2	KoIC1	WoO2	Eq1	Low
AS.3	Man-in-the-middle attack to modify intercepted V2X messages		ET3	SE2	KoIC2	WoO2	Eq2	Very low
AS.4	Spoofing - ExternalVehicle - OBU Channel.		ET3	SE2	KoIC2	WoO2	Eq2	Very low
AS.5	Perception component spoofing		ET2	SE1	KoIC1	WoO3	Eq1	Low
AS.6	Monitor sensors' output to understand normal behavior		ET2	SE1	KoIC1	WoO3	Eq1	Low
AS.7	Injection of malicious sensor data		ET3	SE2	KoIC2	WoO3	Eq2	Very low
AS.8	Information gathering of the target ECU		ET4	SE2	KoIC2	WoO3	Eq2	Very low
AS.9	Exploitation of ECUs software weaknesses		ET4	SE2	KoIC2	WoO3	Eq2	Very low
AS.10	Tampering on ExternalVehicle - OBU Channel.		ET2	SE2	KoIC1	WoO2	Eq1	Low
AS.11	Monitor exchanged message to understand its structures and patterns		ET2	SE2	KoIC2	WoO2	Eq2	Very low
AS.12	Tampering through communication channels		ET3	SE2	KoIC2	WoO2	Eq2	Very low
AS.13	Interception of actuators commands		ET3	SE2	KoIC2	WoO3	Eq2	Very low
AS.14	Tampering on actuators		ET3	SE2	KoIC2	WoO3	Eq2	Very low
AS.15	Spoofing actuators		ET3	SE2	KoIC2	WoO3	Eq2	Very low
AS.16	Bus flooding		ET1	SE1	KoIC1	WoO3	Eq1	Low
AS.17	Replay actuators commands out of context		ET3	SE2	KoIC2	WoO3	Eq2	Very low

Controls Table (Accumulated)

Tables Legend

Black rating means locally overridden.
Gray rating means derived from catalog class or attack tree children.

Name	Title	Description	ET	SE	KoIC	WoO	Eq	AFL
AccessControl	Access control - HMI ECU, ...		ET2	SE1	KoIC2	WoO2	Eq2	Very low
Firewall	Firewall - Zonal Controller 1, ...		ET2	SE1	KoIC2	WoO2	Eq1	Low
MACSec	MACSec - ZC2 - ADAS ECU [Ethernet], ...		ET3	SE2	KoIC2	WoO3	Eq1	Very low
NIDS	NIDS - Clock sen - ZC2 [CAN], ...		ET1	SE2	KoIC1	WoO2	Eq1	Medium
SecBoot	Secure Boot - Zonal Controller 1, ...		ET4	SE3	KoIC3	WoO3	Eq3	Very low
SecOC	SecOC - ZC1 - Acceleration act [FlexRay], ...		ET3	SE2	KoIC2	WoO3	Eq2	Very low

Risks Table

Risk				Risk Level		
Name	Title	Description	Caused by	RL	RU	OEM
R.1	Spoofing of ExtVehicle-OBU Channel. and data flow		TS.1: Spoofing of ExtVehicle-OBU Channel. and data flow	2	2	
R.2	Tampering on internal perception data and DENM in the Channel.		TS.2: Tampering on internal perception data and DENM in the Channel.	2	2	
R.3	Man-in-the-Middle Attack on ExtVehicle-OBU Channel.		TS.3: Man-in-the-Middle Attack on ExtVehicle-OBU Channel.	1	1	
R.4	Exploitation of software weaknesses on the ECUs		TS.4: Exploitation of software weaknesses on the ECUs	2	2	
R.5	Tampering on ExternalVehicle-OBU Channel.		TS.5: Tampering on ExternalVehicle-OBU Channel.	2	2	
R.6	Exploitation of software weaknesses on HMI ECU		TS.6: Exploitation of software weaknesses on HMI ECU	2	2	
R.7	Tampering through communication channels		TS.7: Tampering through communication channels	2	2	
R.8	Tampering on actuators (component and Channel.)		TS.8: Tampering on actuators (component and channel.)	2	2	
R.9	Spoofing on perception components		TS.9: Spoofing on perception components	3	3	
R.10	Spoofing on actuators		TS.10: Spoofing on actuators	2	2	
R.11	Information disclosure on perception data through the channels		TS.11: Information Disclosure on perception data through the channels	2	2	
R.12	Denial of service on actuators		TS.12: Denial of Service on actuators	3	3	
R.13	Tampering of TxV		TS.13: Tampering of TxV	3	3	

Control Scenarios per Risk

Name	Sc.2 No controls	Sc.1 All controls
R.1	2	1
R.2	2	2
R.3	1	1
R.4	2	2
R.5	2	1
R.6	2	2
R.7	2	2
R.8	2	2
R.9	3	2
R.10	2	2
R.11	2	1
R.12	3	2
R.13	3	2

Data Table

Name	Title	Description	Contained data	CAL
D.BrakeNotif	Braking notification			
D.Brake_C	Brake command			
D.Brake_S	Brake			
D.CAM	Cooperative Awareness Messages			
D.Camera_S	Camera frames			
D.Clock_S	Clock			
D.DENM	Decentralized Environmental Notification Message	Emergency Brake Warning		
D.Direc_S	Direction	Data from direction sensor		
D.Pos_S.	Position			
D.Speed_S	Speed			
D.Speed_C	Speed command			
D.Steering_C	Steering command			

Components Table

Name	Title	Description	Stored data	Technology	CAL
ADAS	ADAS ECU		D.DENM: Decentralized Environmental Notification Message D.Direc_S: Direction D.Pos_S.: Position D.Camera_S: Camera frames D.Brake_S: Brake D.Clock_S: Clock D.Speed_S: Speed D.Brake_C: Brake command D.Speed_C: Speed command D.CAM: Cooperative Awareness Messages		
Acc_A	Acceleration actor		D.Speed_C: Speed command		
Brake_A	Brake actor		D.Brake_C: Brake command		
Brake_S	Brake sensor		D.Brake_S: Brake		
Clock_S	Clock sensor		D.Clock_S: Clock		
Cm_S	Camera sensor		D.Camera_S: Camera frames		
Dir_S	Direction sensor		D.Direc_S: Direction		
HMI	HMI ECU				
OBU	OBU		D.DENM: Decentralized Environmental Notification Message D.CAM: Cooperative Awareness Messages		
Pos_S	Position sensor		D.Pos_S.: Position		
RxV	Ego vehicle				
SYS	System	System component			
Speed_S	Speed sensor		D.Speed_S: Speed		
Steer_A	Steering actor		D.Steering_C: Steering command		
TxV	External Vehicle		D.DENM: Decentralized Environmental Notification Message D.CAM: Cooperative Awareness Messages		
ZC1	Zonal Controller 1		D.Brake_C: Brake command D.Speed_C: Speed command		
ZC2	Zonal Controller 2		D.Direc_S: Direction D.Pos_S.: Position D.Camera_S: Camera frames D.Brake_S: Brake D.Clock_S: Clock D.Speed_S: Speed		

Channels Table

Name	Title	Description	Technology	CAL
Ch.1	HMI - ADAS [Eth]		Ethernet: Ethernet	
Ch.2	ADAS - OBU [Ethernet]		Ethernet: Ethernet	
Ch.3	ExternalVehicle - OBU [mobile]		mobile: Wireless Mobile Communication	
Ch.4	ADAS - ZC1 [Ethernet]		Ethernet: Ethernet	
Ch.5	ZC1 - Acceleration act [FlexRay]		FlexRay: Flexray	
Ch.6	ZC1 - Brake act [FlexRay]		FlexRay: Flexray	
Ch.7	Direction sen - ZC2 [CAN]		CAN: Controller Area Network	
Ch.8	Position sen - ZC2 [CAN]		CAN: Controller Area Network	
Ch.9	Brake sen - ZC2 [CAN]		CAN: Controller Area Network	
Ch.10	Clock sen - ZC2 [CAN]		CAN: Controller Area Network	
Ch.11	Speed sen - ZC2 [CAN]		CAN: Controller Area Network	
Ch.12	Camera sen - ZC2 [CAN]		CAN: Controller Area Network	
Ch.13	ZC2 - ADAS ECU [Ethernet]		Ethernet: Ethernet	
Ch.14	ZC1 - Steering Act			

Data Flows Table

Name	Title	Description	Transferred Data	Technology	CAL
DF.1	D.1: OBU -> ADAS [Ethernet]		D.DENM: Decentralized Environmental Notification Message D.CAM: Cooperative Awareness Messages		
DF.2	D.1: ExternalVehicle -> OBU [mobile]		D.DENM: Decentralized Environmental Notification Message D.CAM: Cooperative Awareness Messages		
DF.3	D.7, D.8: ADAS -> ZC1 [Ethernet]		D.Speed_C: Speed command D.Brake_C: Brake command		
DF.4	D.9: ZC1 -> Acceleration act [FlexRay]		D.Speed_C: Speed command		
DF.5	D.8: ZC1 -> Brake act [FlexRay]		D.Brake_C: Brake command		
DF.6	D.2: Direction sen -> ZC2 [CAN]		D.Direc_S: Direction		
DF.7	D.3: Position sen -> ZC2 [CAN]		D.Pos_S: Position		
DF.8	D.5: Brake sen -> ZC2 [CAN]		D.Brake_S: Brake		
DF.9	D.Speed_S: Brake_S -> ZC2 [CAN]		D.Speed_S: Speed		
DF.10	D.6: Clock sen -> ZC2 [CAN]		D.Clock_S: Clock		
DF.11	D.7: Speed sen -> ZC2 [CAN]		D.Speed_S: Speed		
DF.12	D.Brake_S: Speed_S -> ZC2 [CAN]		D.Brake_S: Brake		
DF.13	D.4: Camera sen -> ZC2 [CAN]		D.Camera_S: Camera frames		

The 5G Automotive Association (5GAA) is a global, cross-industry organisation of over 115 members, including leading global automakers, Tier-1 suppliers, mobile operators, semiconductor companies, and test equipment vendors. 5GAA members work together to develop end-to-end solutions for future mobility and transport services. 5GAA is committed to helping define and develop the next generation of connected mobility, automated vehicles, and intelligent transport solutions based on C-V2X. For more information, please visit <https://5gaa.org>

