5GAA Automotive Association

Creating Trust in Connected and Automated Vehicles

5GAA Automotive Association
White Paper

Connected and Automated Vehicles (CAVs) are complex and dynamic systems, where the safety and behaviour of one node affects the efficiency and safety of the whole system. Such systems are usually vulnerable to agents that are untrustworthy for various reasons. In these cases, a way to assess and quantify the trustworthiness of the data shared by the nodes is necessary, in order to establish trust between multiple cooperative nodes, i.e., vehicles that work in collaboration.

In order to address this challenge, a necessary step is to define shared vocabulary and definitions. In response to that, this document introduces and defines terms relevant to the definitions of trustworthiness and trust, followed up by a taxonomy of trust relationships.

It also gives a detailed list of trustworthiness properties in cooperative intelligent transport systems (C-ITS) and CAVs, based on which trust can be assessed. It then emphasises the importance of performing this assessment dynamically and in real time, as well as providing evidence for the evaluation of the corresponding property.

Verifying such evidence is a key part of the approach to trust assessment, and it should also consider cases where evidence holds an inherent level of uncertainty.

Contents

1 Introduction 5
1.1 Reference Use Case: Autonomous Intersection Management (AIM) ... 7

2 Threat Landscape 10

3 Definitions of Trustworthiness and Trust 13
3.1 Definitions of Related Terms 13
3.2 Trustworthiness 15
3.3 Trust 17
3.4 Properties of Trustworthiness 17

4 Evidence-based Evaluation of Trustworthiness 22
4.1 Sources of Trust 23
4.1.1 Trust Sources Related to Communication 23
4.1.2 Trust Sources Related to System Integrity 23
4.1.3 Trust Sources Related to Applications 24
4.1.4 Trust Sources Related to Entity Behaviour 25
4.1.5 Sources of Trust from a Safety Point of View 26
4.1.6 Trust Sources Related to Sensor Data Integrity 27
4.2 Verifiability of Evidence for Evaluation of Trustworthiness 29
4.3 Trust Assessment 30

5 Conclusions 32

6 References 33

Annex A Abbreviations 35

1 Introduction

Connected and Automated Vehicles (CAVs) will benefit from increased connectivity with other vehicles, the infrastructure and other road users. This heightened level of connectivity allows them to exchange planned trajectories/routes and coordinate manoeuvres with other traffic participants as well as the infrastructure. Such information sharing paves the way for the implementation of cooperative automated driving scenarios where automated vehicles can collaborate implicitly or explicitly to execute manoeuvres while avoiding conflicts and ensuring overall safety.

In the updated roadmap published by 5GAA [1], a lot of emphasis is placed on sensor sharing use cases with different variations (e.g., data collection and sharing for HD maps, data sharing of dynamic objects, non-analysed sensor signal sharing). Sensor sharing is the cornerstone of Advanced Driver Assistance Systems (ADAS) ranging from Level 2 (AD L2+) to Level 3 (AD L3), as well as connected ADAS assistance, as they are building blocks required for automated driving.

However, the shift towards higher levels of automation poses a significant challenge – the need for external data to facilitate partially automated or fully automated driving functions. In this context, the integrity and trustworthiness of external data sources, such as sensor information, maps, and positioning data, becomes paramount. If the integrity of this data is compromised or not provided with the expected quality, the building blocks of the automated operational functions will use incorrect data to control the vehicle. There is a broad set of security attacks that have consequences on the trustworthiness of the data and data sources. The dependability and resilience of CAVs can be seriously affected by these attacks at run-time. Furthermore, there are many sources and reasons that can negatively impact dependability and safety that are not related to security. Mechanical defects, failure of Electronic Control Unit (ECUs), or decreased sensor accuracy are just some examples of events from this category.

The need to solve this problem becomes increasingly pressing as we move towards more advanced use cases and entities increasingly depend on external information to make safety-critical decisions. Consequently, for all forthcoming use cases of smart mobility in the realm of C-ITS and CAVs to effectively utilise external information, it becomes imperative to explicitly define and quantify the trustworthiness of exchanged data, which is used as evidence. The integrity of any evidence, particularly when it is used in safety-critical decision-making, should be trustworthy hence verifiable.

Even when the security and integrity of C-V2X communication is somehow established, the problem of assessing how much trust to assign to the exchanged information in such a highly dynamic, distributed, and ubiquitous environment, remains open. That is because we lack tools to reason about trust relationships between data sources that were previously unknown to each other. In the emerging scenarios, it might be the case that the sources of evidence offered by others are untrusted, or the evidence is indirect and obtained through a referral chain.

The issue of trust in C-ITS and CAVs extends beyond the realm of data and data sources. Multi-access Edge Computing (MEC) [2] and its application is widely discussed and tested in the automotive industry for use cases requiring low latency, and

it is also considered an important enabler for automated driving functions. This is because MEC can bring processing power near the vehicle, to meet ultra-low-latency requirements and reduce network traffic towards a datacentre. This has two important advantages. Firstly, with the help of MEC, massive computation and storage tasks need not be handled in the vehicle with its limited power and resources. Instead, these functionalities can be offloaded to the MEC, which can handle it in a more cost-effective way in real time. Secondly, MEC can act as a coordinating anchor for various Cellular Vehicle-to-Everything (C-V2X) services and enable access in critical safety and the real-time processing of sensor signals from various vehicles and Roadside Units (RSUs).

However, it is essential to acknowledge that such edge-computing environments possess inherent characteristics of a complex and highly heterogeneous ecosystem due to the involvement of multiple vendors, suppliers, Original Equipment Manufacturers (OEMs), and stakeholders [3]. Additionally, in the context of distributed systems, it is not feasible to presume the presence of a central entity responsible for implementing universal security measures (and updates) across the entirety of the system. Hence, it becomes apparent that in such highly complex environments, trust levels vary. Towards this end, the existence of various MEC hosts, which correspond to different trust domains, and may require seamless information exchange, necessitates the implementation of mechanisms for evaluating the level of trust for each party involved [4]. This evaluation should take into account the dynamic nature of the environment along with its heterogeneity, particularly in relation to activities involving lifecycle management (i.e., secure enrolment or deployment).

Given the above challenges, this document focuses on defining the concept of dynamic trust assessment in the automotive domain and especially CAVs. More specifically, this White Paper lays the groundwork for clearer definitions of fundamental concepts regarding trust and trust assessment of nodes and data. It thus provides answers to the following questions:

- How can we define trust and trust assessment in dynamic multi-agent systems like connected automated vehicles?
- What are the properties for evaluating trustworthiness?
- What are possible sources of trust that can be utilised for generating evidence corresponding to these properties?
- What does it mean to assess and quantify the trustworthiness of nodes and data in a dynamic and ever-changing environment?

This White Paper does not extend to how we can provide dynamic trust assessment solutions, but it rather focuses on defining the concepts. Therefore, it remains agnostic to which specific properties and which corresponding sources of evidence should be chosen for evaluating trust. It also remains agnostic to which methodology is used to quantify trustworthiness, i.e., how the acquired evidence can be leveraged to calculate a specific opinion on the trust level. The human aspects of perceiving trust and how this effects the acceptance of vehicle technologies by users is also out of scope of this document.

1.1 Reference Use Case: Autonomous Intersection Management (AIM)

In order to reflect the dynamic nature and heterogeneity of C-ITS applications and the environments in which the systems operate, no initial trust between nodes should be assumed, but trust needs to be built up from zero based on trust sources, and continuously re-evaluated. The vehicles need to establish a sufficient level of trust before they can extend that to one another and collaboratively execute safety-critical tasks. Take the V2X use case of the Intersection Movement Assist (IMA) as a motivating example, where two or more vehicles drive towards an intersection. The goal of the IMA application is to alert the driver approaching the intersection of a potential collision with other vehicles in. In this use case, we require a trust assessment mechanism that answers the question "How much trust can vehicle V_a put into vehicle V_b to cooperatively execute a specific function (e.g., safely passing the intersection)?".

An example of how the IMA application works is provided in Figure 1 taken from 5GAA's C-V2X Use Cases and Service Level Requirements Volume I [5]. The ego vehicle, in blue, is approaching the intersection. The blue vehicle knows the geometry of the intersection and knows the position and kinematic information of the red vehicle thanks to C-V2X communications. The blue vehicle predicts the possible trajectories of the red vehicle and identifies the possible crash zones. As the C-V2X exchange continues, the blue vehicle learns which trajectory is taken by the red vehicle, and continuously estimates the probability of collision in the identified crash zones. As the collision probability reaches a threshold the application issues a timely warning to the driver.

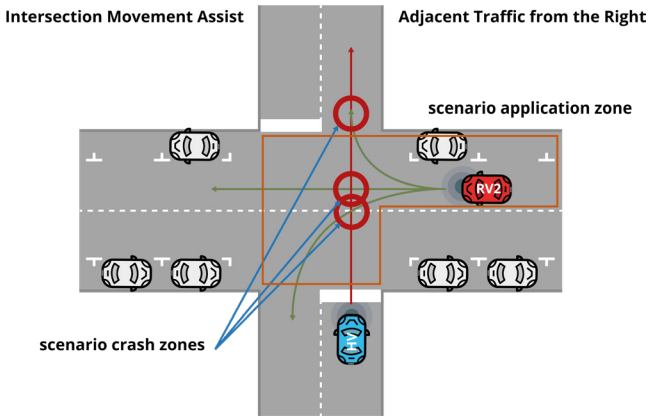

Figure 1 - Example of IMA scenario [5]

A similar application, called Intersection Collision Risk Warning (ICRW), is described in ETSI TS 101 539-2 v1.1.1 [6], and is also referred to as Intersection Collision Warning (ICW) in [7]. All these different versions of the IMA application do not assume any infrastructure equipment at the intersection; they are entirely based on the exchange of V2X messages between the vehicles.

Figure 2(a) describes the radio interfaces involved in a specific scenario. The connected vehicles exchange messages on the V2V interface via direct (short-range)

communication. Cryptographic material is provided via the vehicles' Public Key Infrastructure (PKI), which they use to enable authentication and communication integrity. They periodically broadcast Collective Awareness Messages (CAMs) containing their position and kinematic state.

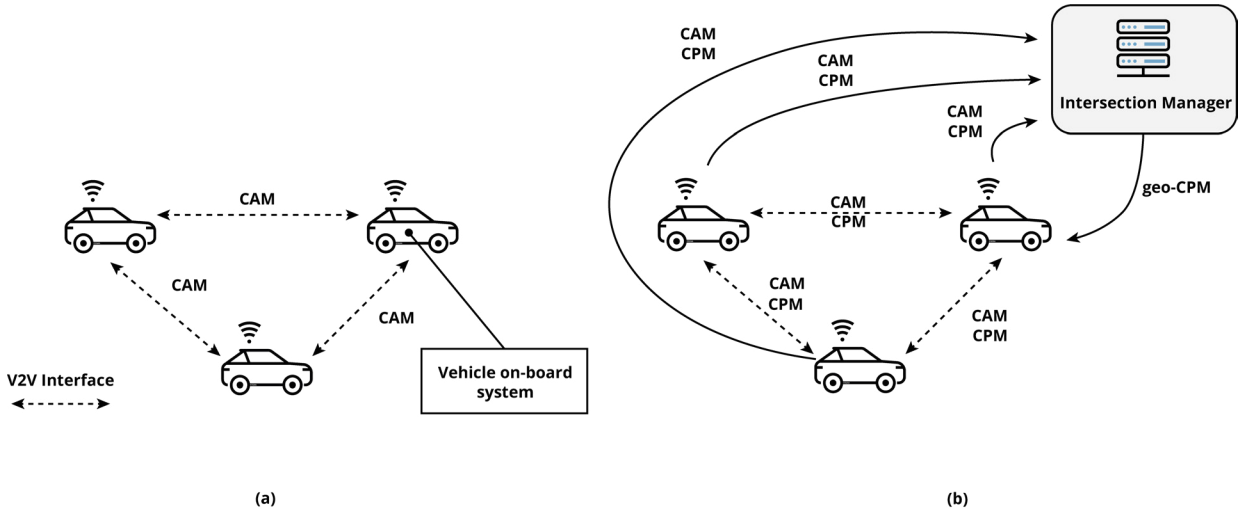

Figure 2 - (a) Radio interfaces in an IMA scenario with three vehicles (b) IMA and Misbehaviour Detection Use Case of CONNECT [8]

An extended version of IMA is presented by the CONNECT project [8], where an Intersection Manager (IM) running as a MEC service is incorporated in the architecture (see Figure 2(b)). With the MEC service being present, V2X nodes share with the MEC their CAMs and Collective Perception Messages (CPM) over the uplink of the Vehicle-to-Network (V2N) radio interface. The MEC is now able to process kinematic data from all vehicles in the intersection, thus significatively improving extended perception compared to what is available at or in the ego vehicles. The MEC hosts the geo-Collective Perception Service (geo-CPS): it encodes geo-CPM messages, which are then disseminated to the V2X nodes over the V2N radio interface downlink. A geo-CPM contains the MEC view of the environment in the form of a collection of observations, as in a standard CPM. The V2X nodes may decide to use geo-CPMs to form the local view of the scene, which is exploited by the IMA application.

Moving towards the connected and automated mobility traffic scenarios, work addresses the complex issue of coordinating connected self-driving cars as they cross an intersection in an autonomous fashion. In this case, the approach is to treat the traffic system as a Multi-Agent System (MAS), where each vehicle is considered a dynamic agent that can autonomously control its behaviour based on both local information and data shared with neighbouring vehicles through a communication network. Zhong et al. [9] surveyed a variety of AIM schemes, where they use centralisation as one of the features to distinguish between them. In a fully distributed AIM, a cooperative plan is negotiated by the vehicles on their own. On the other hand, a fully centralised scheme exhibits a single coordination unit, i.e., IM in charge of planning the traversing of the intersection, acting as the communication partner for all vehicles.

There are various schemes that assign a role to a centralised entity like the IM. In the most general case, the IM is running as a MEC service and gathers further information on relevant road objects and road users, scene prediction, and trajectory planning from the connected automated vehicles and from not automated road users (e.g., VRUs) connected by nomadic smart devices (smart phones or tablets). The edge server processes the information for a dynamic prediction of the overall traffic in the local environment. For example, in the ICT4CART project, the intention is to exploit hybrid connectivity and MEC to create 360° awareness around the vehicle with very low latency, creating a kind of "virtual mirror" to support the automated vehicle while crossing an intersection [10].

In this document we adopt the AIM use case, as defined by Cheng et al. [11], where the IM has a more active role and gives specific orders to the vehicles in order to coordinate their movement though the intersection (I). More specifically, assume a vehicle (X) on the road travelling to, but not yet entering, the intersection area (M). By entering area M, vehicle X communicates with the intersection manager (A) using MEC (see Figure 3) by sending a request Q^X in which they communicate their state, which includes the location and the dynamics (e.g. predicted arrival time, velocity, acceleration, arrival and departure lanes). The IM (A) then calculates the trajectory of X and makes a "grant or reject" decision based also on the intentions of other vehicles in area M. In the event there is a conflict in the simulated trajectories, A rejects Q^X; if not, A approves it and then sends the decision back to X. After that, X is responsible for following the instruction to enter and drive through I. In "reject" case, X has to resend the request and wait for further instructions.

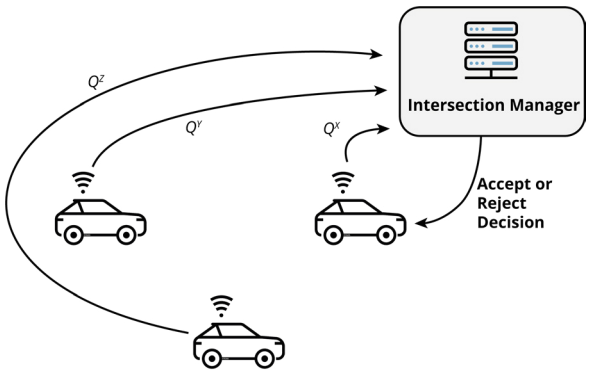

Figure 3 - The Autonomous Intersection Management (AIM) scenario

To be able to implement this use case, all actors need to assess the trustworthiness of the data exchanged at run-time: the IM (A) needs to assess the trustworthiness of requests (Q^X) sent by vehicle X and other vehicles to assess the trustworthiness of the decisions sent by A.

2 Threat Landscape

In order to understand why vehicles cannot implicitly trust the data coming from another system, it is worth having a look at the landscape of potential attacks and malicious incidents that could undermine a V2X system's security. As the level of driving automation in vehicles increase, the amount and sophistication of in-vehicle electronics and networks also rise. Moreover, the introduction of C-ITS connects vehicles with the surrounding environment (e.g., other vehicles, RSUs, and pedestrians) via V2X technologies. As a result, the in-vehicle network components are exposed to the outside world even more. Therefore, we can categorise the security threats in three different domains, namely i) in-vehicle, ii) V2X access, and iii) infrastructure.

In-Vehicle Attacks

Kim et al. [12] conducted an extensive survey on cybersecurity attacks for autonomous vehicles and broke them down into two different categories: attacks on automotive control systems, and attacks on autonomous driving systems components. The first category concerns attacks that mainly target the ECU, the in-vehicle network and the automotive key. Software vulnerabilities in ECUs can be exploited in order to gain unauthorised access and manipulate the ECU functionalities [13]. Firmware exploitation can also lead to unauthorised control of critical vehicle functions [14]. In a parallel context, supply chain vulnerabilities in the context of automotive systems also pose potential risks and might cause malicious software to be embedded in the construction cycle. ECUs and CANs continue to be the targets of attacks. Initially, they were physically connected and attacked, but recently advanced techniques such as side-channel and fuzzing have been used. We should also consider attacks on autonomous driving components and especially sensor attacks. In this context, sensor data integrity is paramount. Attacks involving spoofing or tampering with sensor data can produce untrustworthy information and pose a significant threat [12].

Attacks on V2X Communication Technologies

A wide range of possible attacks can disrupt V2X communication. Attackers can easily disseminate fake or wrong information in order to mislead other vehicles. Attackers can also gain access to the system to delete, or intercept forwarded data. This type of attack is usually launched by "insiders" and can be the result of a Sybil attack or any other attack that leads to identity theft. Another approach is where the attackers seek to prohibit the use of system communications channels (e.g., channel jamming attack), thus undermining the trustworthiness of the system. Another type of attack reuses or replays the old data at a different or later point in time. The effect of this attack is similar to bogus information dissemination. This could also be as a result of identity theft and other approaches such as a Sybil attack. In general, the above attacks could lead to inaccurate traffic messages, forgeries, false warnings, and bogus misconduct reports which could result in node failures, collisions, message tampering, and other risks to safety services.

Infrastructure Attacks

In addition to the aforementioned attacks, there are attacks that could be launched directly on the primary Vehicular Ad-hoc Network (VANET) infrastructure or through

integrated technologies such as cloud computing [15] and Software-Defined Networks (SDN) [16]. Also, it is important to consider security threats in the MEC, as an important enabler of several new use cases and various services in automotive scenarios [4], and that security and compliance is a shared responsibility between several parties; the MNO, MEC tenant application provider, and the application user. In particular, MEC deployments are characterised by the presence of multiple MNOs, and edge computing infrastructures, where systems are virtualised (with different parties potentially providing portions of an overall compute solution). Based on the 5GAA study on Cybersecurity for Edge Computing [4], the main aspects to be considered, when referring security threats in such environments, are:

- Workloads are outside the trusted Public Land Mobile Network (PLMN) domain, but instead run in external Electronic Communication Service Provider (ECSP) domains.
- Mutual trust between MEC applications and MEC platforms, meaning that 1) in principle the edge application from MNO A should be considered as though it would be running in a "hostile" environment (MNO B), and vice-versa, 2) a platform operated by MNO B is hosting "unknown" applications which may endanger the system.
- Security threats are also related to all the communication links (both data plane and control plane), meaning that all relevant communication channels can be untrusted, in principle.

Overall, it is essential to acknowledge that such edge-computing environments possess inherent characteristics of a complex and highly heterogeneous ecosystem due to the involvement of multiple vendors, suppliers, OEMs, and stakeholders. Additionally, in the context of distributed systems, it is not feasible to presume the presence of a central entity responsible for implementing universal security measures (and updates) across the entire system. Hence, it becomes apparent that in such highly complex environments, trust levels vary. Towards this end, the existence of various MEC hosts, which correspond to different trust domains and may require seamless information exchange, necessitates the implementation of mechanisms for evaluating the level of trust for each party involved.

From the above overview of security threats, we can underscore the challenges posed by insider threats, where legitimate users or vehicles in the network may be altered or counterfeited by malicious entities. These threats cannot be mitigated by traditional cryptographic solutions alone, thus requiring a more nuanced approach. Also, we need to highlight that the highly dynamic nature of vehicular networks and the incorporation of new technologies like MEC, makes it impossible to employ traditional network security models that assume a network perimeter or "trust zone" protected against unauthorised access.

In the past, trust models have been based on concepts like PKI solutions used for V2X communications, which rely on central authorities and assume that the main On-Board Unit (OBU) within vehicles cannot be compromised. However, the evolution of Day-2+ operations has complicated the threat landscape, necessitating a paradigm shift in trust assumptions.

This leads us to the concept of continuous evaluation of data sources. Each piece

of data and its source must be continuously verified. This means that every piece of information exchanged between vehicles would undergo verification and assessment of its trustworthiness before acted upon. **Trust can never be assumed; instead, it is continually earned**. This approach is especially critical in dynamic environments like V2X, where the accuracy and integrity of data are crucial for safety and operational decisions.

3 Definitions of Trustworthiness and Trust

As the above discussion demonstrates, there is an emerging need to assess trust in complex and dynamic systems, where the safety and behaviour of one node affects the efficiency and safety of the whole system. Such systems are usually vulnerable to agents that are untrustworthy for various reasons, as described in the previous section. In these cases, we need a better way to measure the trustworthiness of the data shared by the nodes in order to establish trust between multiple cooperative nodes, i.e., vehicles or MEC that work in collaboration. To address this challenge, a necessary step is to define shared vocabulary and definitions. In response to that, in Section 3.1, we first introduce and define terms relevant to the definitions of "trustworthiness" and "trust", backed by a taxonomy of "trust relationships" in Section 3.2. We define trustworthiness and trust in Sections 3.3 and 3.4, respectively. Finally, in Section 3.5, we discuss trustworthiness properties in C-ITS systems and CAVs.

3.1 Definitions of Related Terms

Firstly, we start by defining some fundamental concepts for modelling trust and trust assessment.

Trust objects. Trust objects are entities that assess trust (or for which trust is assessed), and based on this trust relationships are built. Two things are relevant when identifying trust objects: the components and the propositions.

In general, trust objects can represent both **nodes** and **data**. For example, nodes can be vehicle ECUs, Zonal Controllers (ZC), MEC, etc., and data can be geolocation coordinates, camera feed, etc.

A **proposition** is a logical statement about some phenomenon of interest (i.e., a variable) whose level of trustworthiness we are interested in assessing. The proposition describes the fulfilment of a certain property of data or a node. A proposition could be 1) atomic – a proposition whose truth or trustworthiness can be directly assessed or verified through some evidence (from one of several trust sources), or 2) composite – consisting of multiple atomic propositions.

In the AIM example from Section 1.1, the proposition would be, for example, assessing the "integrity of the data" (e.g., geolocation information, vehicle size, predicted arrival time, etc.) sent through the request Q^X from the vehicle X to the intersection manager A. The integrity is the concrete property we want to assess, and the data is the concrete location and dynamics. Another example of a proposition would be assessing the "accuracy of the data".

The trust objects are the main building blocks for trust relationships. Again, in the AIM example, the trust objects are all the entities present at the intersection, e.g., vehicles, intersection manager A, as well as the propositions for which we want to assess trustworthiness. Based on these trust objects we build the trust relationships

explained in the next step.

Trust relationship. Trust relationship is a directional relationship between two (trust) objects that can be called "trustor" and "trustee" (the one who is trusted). The trust relationship is always tied to a concrete property.

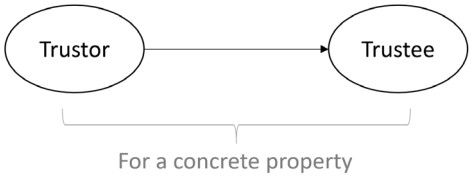

Figure 4 - Trust relationship

As shown in Figure 4, the trustor is the "source" trust object in the trust relationship that is assessed (one who trusts, the "thinking entity", the assessor), and must be a node. The trustee is a "sink" trust object in the trust relationship that is assessed (one who is trusted), and can be either a node or data.

In the AIM example, a trust relationship would exist between the IM (A) and the vehicle X (node-to-node trust relationship). Another example of a node-to-node trust relationship would be one between two vehicles. Additionally, there could be trust relationships between a node and data, in this case, vehicle X and the data inside the request Q^X sent to the IM.

More specifically, we refer to Jøsang et al. who defined the notions of **functional** and **referral** trust [17]. Functional trust represents a type of belief that some data fulfil a certain purpose or possess a certain property; that a vehicle's component (e.g., sensor) has the ability to perform its designated function. Referral trust is observed when a node relies on the recommendation of another node to make a trust assessment for some data or a node. For example, in the AIM scenario, vehicle X has a functional trust in the data Q^X, and the IM has referred trust in vehicle X for the purpose of forwarding this data. Both of these trust relationships are **direct**. Then, through the forwarding of data from vehicle X, the Intersection Manager also trusts data Q^X, however this function trust is indirect, or **derived**, since A has no direct role in producing the data.

Trust network. The trust network combines various trust relationships among different trust objects. Figure 5 shows an example of a trust network, where the same trust object (for example, vehicle X), can be both a trustor (in $X \rightarrow Q^X$ trust relationship), and a trustee (in $A \rightarrow X$ trust relationship). With red boxes we label the (atomic) propositions as trust objects as part of the trust network. The (atomic) propositions are always in the leaves of the trust networks, and are always trustees. There are two types of trust relationships as part of the trust network that we mark with different arrows:

- dashed arrows that represent referral trust relationships (e.g., $X \rightarrow A$, $Y \rightarrow A$); always related to trustworthiness assessment on nodes (from a node to a node), and
- solid arrows that represent functional trust relationships, related to trustworthiness assessment of a proposition (from a node to a proposition), e.g., $A \rightarrow Q^X$, $X \rightarrow Q^X$, $Y \rightarrow Q^Y$; the nodes have a direct observation on a concrete proposition.

Additionally, as previously explained, the propositions are related to the fulfilment of the certain properties of data or nodes. As a result, based on the type of propositions, we differentiate between two types of direct trust relationships: **data-centric** and **node-centric**. As part of the data-centric trust relationship, the trustee is expressed through a proposition on (a piece of) data; whereas, in a node-centric trust relationship, the trustee is expressed through a proposition on a concrete node. Please note that referral trust relationships are always node-centric.

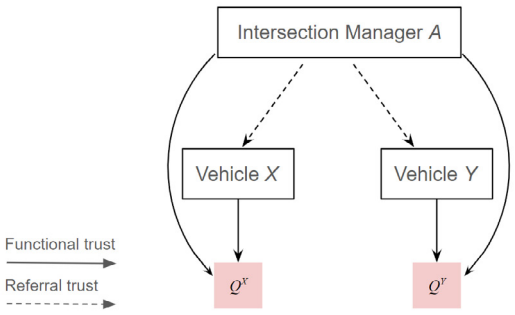

Figure 5 - Trust network corresponding to the dynamic intersection management use case

Figure 5 depicts a simple example of such a trust network built according to our AIM use case. As shown in the trust model, the intersection manager can derive its trust in the request Q^X through the referral trust that the IM has in the vehicle X ($X \rightarrow A$) and the direct trust that vehicle X has in its own data Q^X ($X \rightarrow Q^X$). So, in the end A has a derived functional trust to Q^X ($A \rightarrow Q^X$).

In this case, the Intersection Manager can derive and assess trust on the request Q^X, through the referral trust that the IM has in vehicle X and the direct trust that vehicle X has on its own data Q^X.

3.2 Trustworthiness

There are at least two main aspects associated with the trustworthiness of a given trustee: its ability to deliver the expected performance, and the extent to which it is aligned with the goal of the trustor. For example, a given system in C-ITS needs to have the required technical ability to exhibit the relevant properties of safety, robustness, usability, etc. and this ability needs to be aligned with the expectations of the stakeholders; for example, the users of this system or policymakers regulating their design and use – i.e. what the appropriate level of safety is, what criteria need to be satisfied in order to deem a system trustworthy, etc.

Here, trustworthiness is to be defined within a specific "context", or the restrictions on a set of circumstances under which the trustee is expected to perform or achieve the given tasks. That is, the trustee is not expected to fulfil expected tasks under all circumstances, but under a limited set of defined circumstances. For example, a system in C-ITS may be expected to conform to relevant safety standards under proper conditions of use.

Given this discussion, we define trustworthiness as the *likelihood of the trustee to fulfil*

the trustor's expectations in a given context, where such expectations can be a function of the entrusted task, the process through which it was achieved, and the purpose for which the task was chosen.

Here, expectations could relate to the correctness of data, as well as the assessor's ability to ascertain the correctness of the data. However, expectations can also relate to the behaviour of the trustee (e.g., if the trustee is a node, then in relation to the functionality of that node), the process through which the entrusted task was carried out by the trustee, and the purpose for which the task was chosen. We therefore need to bridge the trustworthiness of data sources with expected behaviour, since there is nothing in the trustworthiness of data sources and data that would entail consistent behaviour. Different assessors (i.e., trustors) might have different rules on how to translate this to expected behaviour. So, we can do this bridging based on decision-making, for example rules or policies that could support calculations on expected behaviour related to the data collected by different data sources. For example, one of the ways in which an autonomous vehicle can be trustworthy to a user or another vehicle is by fulfilling certain expectations regarding safety (by driving safely and not causing accidents) and providing evidence to the user or the other vehicle regarding how such safety expectations will be met.

Formally, given a trustor *A* and a trustee *B*, one can denote the trustworthiness of *B* for *A*'s reasonable expectations regarding *B*'s behaviour *R(x)* in a context *C* as:

$Tw_{B,A}$ – The likelihood that *B* will exhibit behaviour *R(x)* in Context *C*. (1)

Further, trustworthiness can be a matter of degree or levels. That is, a trustee *B* may be more likely to fulfil the trustor's expectations to some degree or level *L* between 0 to 1, where 0 denotes no such likelihood and 1 denotes a maximal likelihood to fulfil trustor expectations.

(1) can then be re-written as:

$Tw_{B,A}$ – The likelihood that *B* will exhibit behaviour *R(x)* in Context *C* to a level *L*. (2)

Finally, trustworthiness needs to be verifiable in the sense that the trustor should have access to evidence regarding *B*'s likelihood to fulfil the relevant expectations. For the ideal/maximal evidence *E*, which would warrant appropriate trust in the trustee, we can write:

$Tw_{B,A}$ – The likelihood that *B* will exhibit behaviour *R(x)* in Context *C* to a level *L* established by evidence *E.* (3)

E can potentially have many sources. Some examples include:

- Evidence (direct or indirect) of *B*'s past behaviour (ideally in context *C*, or a similar contexts) available or applicable to *A*.
- An assessment made by an independent agent *Z* about *B*'s ability (and willingness) to exhibit *R(x)* in context *C* made available to *A* (referral or transitive trust).
- Information about compliance with, for example, legal regulations that incentivise *B* to exhibit *R(x)* or disincentivise/prohibit *B* to deviate from exhibiting *R(x))*.

We should note here that the **evidence is objective, but verification is subjective**,

meaning that the interpretation of the same evidence by different trustors might be based on different procedures, resulting in different verification results for the same evidence. For example, according to the verification procedure of trustor *A*, evidence given in relation to a proposition might not be sufficient to justify the proposition, while in a different verification procedure, for example of trust *B*, it could be sufficient.

3.3 Trust

While trustworthiness is related to the trustee, trust itself is in relation to the trustor. As previously explained, trustworthiness is a measure of a trustee's ability to meet the trustor's expectations. On the other hand, trust is a decision made (or an attitude held) by the trustor to trust or not trust a concrete trustee.

Given two entities *A* and *B*, where *A* is the trustor (one who trusts) and *B* is the trustee (one who is trusted),

***A* Trusts *B* implies that *A* has expectations that *B* will have the property of being trustworthy.**

In other words, when *A* trusts *B*, *A* deems that the likelihood that *B* will meet *A*'s expectation is very high, or higher than what may be required given *A*'s expectations and risks taken by *A*. In trusting *B*, it is critical that *A*'s expectations and evaluation of *B*'s trustworthiness is reasonable, appropriate and calibrated to *B*'s actual trustworthiness.

3.4 Properties of Trustworthiness

As mentioned, trustworthiness can be defined as the measure of the likelihood of the trustee being able to fulfil the expectations of the trustor in a given context. One way to evaluate this likelihood is by assessing whether the trustee exhibits the right and relevant set of properties that enable it to meet the trustor's expectations in a given trust relationship. For example, consider a trust relationship between a zonal controller within a vehicle and a camera ECU during a Cooperative Adaptive Cruise Control (CACC) function, where the zonal controller is a trustor that relies on the camera ECU, the trustee, to deliver non-compromised camera data. Here, the camera ECU needs to exhibit, among others, the property of reliability. So, assessing whether the camera ECU is reliable in passing on its data to the ECU can give positive evidence of its trustworthiness.

Properties to evaluate the trustworthiness of such trustees can in general be categorised into three broad categories:

1. Performance-based – These properties are linked to performance criteria such as reliability, accuracy, and robustness. Such properties are vital in C-ITS to ensure the safe and efficient operation of vehicles, and they are well defined in the corresponding standards. Properties such these are critical in delivering consistent and dependable performance, while a property like resilience is essential for adapting to various real-world scenarios, fostering user trust.

2. Ethical aspects – These properties are clearly linked to the ethical aspects and implications in a given context, such as privacy protection and safety. Ethics-based properties play a crucial role in C-ITS as they define the moral framework governing the behaviour of vehicles and other key components. These properties are paramount when considering trustworthiness due to their direct impact on public perception and societal implications. Properties such as accountability and transparency are essential for holding the system and manufacturers responsible, and for providing insights into decision-making, promoting accountability and regulatory compliance. Explainability ensures that system actions are interpretable to users and regulators, addressing concerns about the "black-box" nature of AI (or components of AI-based technologies). Usability and authenticity reinforce the system's commitment to user objectives and protect against malicious actors, enhancing public trust in C-ITS. These properties are essential to address concerns related to liability, unintended consequences, and the potential for unethical behaviour, which can significantly influence public trust and acceptance of automated vehicles. By upholding strong ethical principles, systems in C-ITS can build a foundation of trust with users and society, promoting widespread adoption and contributing to the safe and responsible advancement of autonomous mobility technologies.
3. User acceptance – These properties are linked to issues of transparency and usability, and they have implications on overall acceptance of the system by users. Such properties are paramount in C-ITS to gain public confidence and ensure further adoption of automated vehicles. Privacy protection safeguards personal data, alleviating privacy concerns and respecting users' rights. Usability addresses how easy it is to interact with and use the system, making the technology accessible and user-friendly for a broader audience. Safety and security instil confidence in passengers by prioritising their well-being and mitigating cybersecurity risks. Relevance and consistency provide accurate and pertinent information, bolstering user confidence in the system's capabilities. Recency and credibility emphasise the importance of up-to-date and trustworthy data, enhancing user trust in the information provided. Equitable access ensures fair market opportunities for various C-ITS providers, fostering a competitive and diverse landscape.

These properties can be overlapping in the sense that certain properties may belong to more than one category, or even all three categories. For example, safety is a property that is linked to the performance of the C-ITS system. To provide safety a level of rigor (high Automatic Safety Integrity Level, ASIL) is required. In that way, safety implies sufficient redundancy, ability to detect and report faults, understanding and mitigating the functional deficiencies in performing the function. But safety is also a required ethical value for the system to exhibit is also a property that potentially leads to higher acceptance of the system by the users. Similarly, integrity is an important property which relates to the communication and data exchanged between different sensors and vehicle software remaining unaltered without proper authorisation. The integrity of such communication also has critical significance both in terms of performance as well as for protecting key ethical values such as safety.

Here, we describe an indicative set of properties relevant when evaluating the

trustworthiness of systems in C-ITS, and their components. This list has been extracted from sources such as documentation on standards[1] (ISO (5723:2022) [18], ISO (22624:2020) [19] and ITU-T (Y.3057) [20]), existing literature on autonomous vehicle systems and trustworthiness (i.e. Fernandez Llorca & Gomez [21]), and existing documentation on Cooperative, Connected and Automated Mobility systems [22].

The descriptions of properties here are meant to indicate how a trustee can exhibit these properties, or how such properties can be verified, in the context of C-ITS. As stated earlier, verifiability itself is an important aspect of establishing trustworthiness in this context. Verifiability also involves the trustee's ability to provide evidence justifying the decision by the trustor to trust the trustee. In future work, more precise conceptions of what properties are applicable and relevant to the evaluation of a particular trustee (for example, a particular zonal controller), how this particular trustee can exhibit these relevant properties, and what criteria it should fulfil for a positive evaluation or verification of trustworthiness will be formulated.

Property	Description
Accountability	Being responsible and answerable for the actions and decisions made by the autonomous vehicle system or its components (see [18] [21]). For example, in order to be trustworthy, in the event of a technical problem with a specific component, the manufacturer who implemented the component must be accountable for the malfunction.
Safety	According to 5GAA [24], trust in the context of received V2X data from the point of view of functional safety implies a) knowledge of the intended function, b) information about the required quality and accuracy, and c) knowledge of how the data-generating subsystem is designed, developed, implemented, maintained, and operated. One can turn to the vehicle-centric safety principles of ISO26262 [23] for functional safety, ISO21448 for SOTIF [25] and the broader, system-level safety considerations of IEC 61508 [26] for the infrastructure components like RSUs.
Privacy protection	Safeguarding personal information and ensuring that it is appropriately collected, used, secured, and removed when not needed, and accessible only to authorised parties. Aspects of such appropriate collection include, for example, proper consent mechanisms or other similar measures which may be enshrined in the local data regulations [21]. In the context of C-ITS, where data is collected through various sensors and cameras, privacy protection may also include data sanitisation procedures, such as removal of personally identifiable information and/or data anonymisation procedures.

[1] The ISO standards use the term "characteristics" which is equivalent to our use of the term "properties". Similarly, the ITU-T document uses the terms "characteristics" and "trust indicators" which also retains the meaning we apply with the term "properties" here.

Security	(Integrity) Regarding data, integrity is a property whereby data have not been altered in an unauthorised manner since they were created, transmitted or stored [27]. Regarding a system, integrity is a property of accuracy and completeness [28]. Integrity measures the confidence an entity can place in the correctness of the information supplied, which can be achieved through evidence provided to the entity.
	(Availability) When the property of a system, service, or data is accessible and operational for authorised users. It ensures that the necessary resources are reliably and consistently available, without interruptions, failures, or deliberate attacks, thereby enabling the execution of intended tasks effectively [18].
	(Authenticity) Ensuring the identity of an entity is as it is claimed by it [18].
	(Confidentiality) Ensuring the protection of sensitive information from unauthorised access or disclosure. Confidentiality could be relevant in the context of C-ITS, for example, when camera pictures are processed in the vehicle or in a MEC server [19].
Accuracy	The ability to provide outputs within the expected range of closeness between the measured or estimated value and the true value (or the value accepted as being true) [18]. Typically, a system, e.g., a positioning system, demonstrates such ability by reporting the distribution of the errors under the form of an error percentile, which represents the accuracy of its output with a certain confidence.
Sensor data integrity	A measure of the trust in the accuracy of the specific data and the ability to provide associated alerts [29].
Reliability	The ability of a system to demonstrate dependable behaviour and performance under varying conditions. As an example, for CACC to run properly, it must receive reliable data from various in vehicle sensors and cameras [18] [20].
Robustness	Demonstrating the ability to operate with a sufficient level of performance (and also a high level of consistency) in a variety of circumstances; i.e., under challenging conditions and scenarios. In the context of C-ITS, the system is robust when it can still provide results even if, for example, available data is not that accurate or the conditions on the road do not allow for high accuracy. Further, in the same service ecosystem, the property of robustness also relates to the ability of the system to react when conditions have reached a critical point of uncertainty such that safe functioning is impossible given the quality and accuracy of the data available [18].
Resilience	Being able to adapt, recover, and continue functioning effectively in the face of disruptions, failures, or unexpected events [18] [21]. Such failures or unexpected events in C-ITS systems could be message loss or invalid input provided by a sub-system and evidence is provided that appropriate (certified) mitigation plans have been deployed that are capable of responding to such situations.
Transparency	Being open, clear, and understandable about the functioning, algorithms, and data usage of the autonomous vehicle system [18] [21].
Stability	Not changing easily or maintaining consistency over time without fluctuations [20]. Further, the system should be able to produce the output of the system should not fluctuate if the input remains the same.
Completeness	Ensuring all necessary and relevant information is available without omission [20].
Relevance	The ability to match the expected extent to which the information and outputs provided by the autonomous vehicle system are applicable and useful to the current context or situation [20].
Consistency	The ability to deliver coherent and reliable performance, outputs, and decision-making over time and across different scenarios. A vehicle would have the property of consistency if, for example, the positions provided by the vehicle are compatible with each other. So, the position provided by the vehicle is compatible with the predicted position of the vehicle, based on the previous “send position” coordinates and kinematic data from the vehicle [20].

Recency	Ensuring that the most up-to-date and recent information is used in decision-making and operation [20].
Explainability	Providing understandable explanations or justifications for the decisions and actions taken by the autonomous vehicle system [21].
Usability	The ease of use or user-friendliness of the autonomous vehicle system, ensuring that it is accessible, understandable, and navigable for users, promoting effective interaction and greater acceptance [18].
User-centric	The ability to match the expected extent to which the autonomous vehicle system's decisions, actions, and behaviours align with and take into account the intended goals and objectives of the users or stakeholders [21].
Equitable access	Ensuring fair and unbiased access to the market for autonomous vehicle systems, without undue advantage or discrimination towards any particular system or provider [22].

4 Evidence-based Evaluation of Trustworthiness

As explained in the previous section, a key part of the assessment of trust and trustworthiness is defining the relevant property for a concrete trust relationship. However, to assess a specific trust relationship, it is important to include the evidence that is necessary to assess the particular property, i.e., the concrete trust relationship. This includes how the evidence is chosen, what is the best evidence or the best set of data to represent and assess a specific property, how evidence is managed, etc. 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.

We divide the trust sources into four categories: (1) trust sources related to communication, (2) trust sources related to system integrity, (3) trust sources related to applications, and (4) trust sources related to entity behaviour. However, depending on the trustee, not all categories may be relevant. The trust sources of the first three categories are predominantly security mechanisms. In addition to security mechanisms, a fourth category was added that considers the behaviour of the node to get further evidence about its trustworthiness. Some trust sources might require regular evaluations, while others only require one-time assessments at system start up.

There are various open questions when it comes to evidence and trust sources. For example, how are different trust sources chosen to calculate the trustworthiness of a trust relationship? Based on which trust sources and evidence can we quantify the fulfilment of a certain property within the trust relationship? How are the trust sources chosen in an automated manner in use cases where the trust relationships within the trust model change dynamically at run-time?

To summarise, evaluating trustworthiness and trust is the exercise of going through the various measures of trust applicable to a trust relationship, evaluating the levels of assurance, and if they meet the criteria set, validating the trust relationship. This is done based on evidence, received from the trustees (as sources of trust), conveying information on the status of those properties of interest that can be used in a verifiable manner to calculate and quantify the trustworthiness. We detail more on verifiability of evidence in the following section.

4.1 Sources of Trust

4.1.1 Trust Sources Related to Communication

Concept	Description
Protection mechanisms of communication	The communication between two nodes of the network can be protected against attack through different mechanisms providing security properties, such as confidentiality, integrity or authenticity. Examples of such attacks could be the modification or retransmission of messages that lead to undesired behaviour exhibited by the application. In addition, depending on the protection mechanisms, the level of assurance against attacks varies. For example, the integrity of transmitted messages is protected to varying degrees depending on the type of signatures and key length used. Since different attacks can be realised through different efforts, depending on the protection mechanisms of the communication, the trustworthiness of the communication should be adjusted according to the protection mechanisms used. For example, the use of pseudonym certificates provided by the PKI system and used by vehicles to secure each massage they broadcast.
Hardware security mechanisms	Hardware secure elements are physical computing chips attached to the host device, and can (among other things) manage and store cryptographic keys and perform encryption and decryption functions for cryptographic functions, such as creating signatures, with only authenticated and certified applications having access to the key. The cryptographic keys are managed by the hardware security mechanisms, so it is more difficult for an attacker to get access to the keys, since the system provides secure storage capabilities with strict trust boundaries. Therefore, when hardware security mechanisms are used, it becomes more difficult to impersonate another node in a network, making these components inherently trustworthy, acting as a "Root of Trust" in a system. An example of hardware security mechanism is a Hardware Security Module.

4.1.2 Trust Sources Related to System Integrity

Concept	Description
Secure boot	Secure boot ensures that devices boot up using only trustworthy software, by verifying integrity and authenticity. Thus, during the boot process, the signature of each software component, such as boot loader and operating system, is analysed. This allows the system to determine if this software has been altered or tampered with by a malicious actor. However, it should be emphasised that secure boot can only verify the integrity of the software components at boot time and not during run-time. Secure boot can be used, for example, in an ECU as part of the in-vehicle architecture. Whether it should be deployed in the corresponding ECU must be considered at design stage. Depending on the existence or non-existence of secure boot, the trustworthiness of the system should be adjusted. If secure boot is implemented, this would allow the ECU to verify the integrity and authenticity of all software components relevant for the boot process. Thus, software failing to pass the integrity/authenticity test would not be used by the ECU, leading to greater trustworthiness.
Run-time integrity check	The integrity and authenticity of the Operating System (OS), or parts of it, and the software stack deployed in the target device are checked during run-time. Attacks on the software stack that occur after the boot process, where integrity and authenticity checks might have already been performed by secure boot, are still detected. In this way, it can be determined if the OS was compromised during run-time. Such run-time integrity checks in the automotive domain are proposed, for example, by the SAE, where the kernel code of the system is periodically checked at run-time.

Known OS-vulnerabilities	Based on the OS version of the node and a vulnerability database, it can be determined whether there are any known vulnerabilities in the corresponding OS. The vulnerabilities usually also contain a risk value, which can be used to determine how critical the vulnerability is for the node. Based on that, the trustworthiness of the node can be adjusted. This trust source should ensure that the OS is up-to-date, so that all vulnerabilities are patched and thus the trust in the system is higher compared to an outdated OS version. There are several vulnerability databases that contain vulnerabilities of nodes in the automotive domain, such as in vehicles. An example is the Common Vulnerabilities and Exposures (CVE) database, which describes vulnerabilities and their risk. Based on this risk value, the trustworthiness of the node can be adjusted.
Up-to-date OS/ firmware	If there is a new OS or firmware version, this could indicate that there are vulnerabilities or errors in the old version, such that the data in the corresponding node is not processed correctly. Therefore, out-of-date OS or firmware can reduce the trustworthiness of the entity.
Authentic OS update	Especially in the context of vehicles, ECU update mechanisms are necessary so that a new version of the OS can be installed when vulnerabilities or bugs are discovered. For vehicles, Over-the-Air (OTA) updates are becoming more and more common. To prevent a compromised OS version being installed in a vehicle, a secure update mechanism could be used to verify that the OS update was really provided by an OEM. This mechanism ensures that no compromised OS version is installed. Such a secure update mechanism protects the system from various attack vectors, such that this mechanism should increase the trustworthiness of the node. In the context of secure OTA updates, several mechanisms can be used to secure the update mechanism, such as cryptographic signatures created by the OEM to verify that the OS updates really comes from them.

4.1.3 Trust Sources Related to Applications

Concept	Description
Run-time operational assurance	Based on run-time operational assurance, the modification of operations of an application by an attacker can be detected. In this way, it can be determined whether the application was compromised during run-time. This makes the realisation of a broad range of attacks more difficult, which can be reflected in the trustworthiness level of the application that uses the input data. An example of run-time operational assurance is Control-Flow Attestation (CFA). This is a set of mechanisms that detect alterations in the flow of executions of an application and attests to another party that the control flow was not altered. CFAs could, for example, capture and modify the flow/execution of a program by changing its Link Register.
Known application-vulnerabilities	The application version and a vulnerability database can be used to determine whether there are any known vulnerabilities in the corresponding application. The vulnerabilities usually also contain a risk value, which can be used to determine how critical the vulnerability is for the overall system. Based on that, the trustworthiness of the system can be adjusted. With the vulnerability database not only the application itself, but also libraries used in the application can be checked for vulnerabilities, since they can also contain vulnerabilities which would affect the trustworthiness of the entire application. An example of such a vulnerability database is the Common Vulnerabilities and Exposures database, which describes vulnerabilities and their risk. Based on this risk value, the trustworthiness of the node can be adjusted.

Trusted execution environment	The Trusted Execution Environment (TEE) provides a trusted environment in which data and assets can be stored and code can be executed. The code is protected in that it cannot be viewed or modified by entities outside the TEE. In addition, a TEE allows verification that the code running in the TEE is valid. Also, access to the data and assets in the TEE can be controlled to protect the data and assets from attacks outside the TEE. Thus, integrity and confidentiality of program code and data are provided by the TEE. Such TEEs can be used, for example, inside the vehicle so applications can run on the ECUs in a protected environment. But TEEs could also be deployed outside in-vehicle networks, such as in a MEC server, to protect the applications running there.

4.1.4 Trust Sources Related to Entity Behaviour

Concept	Description
Misbehaviour detection	Based on misbehaviour detection, different types of misbehaving nodes can be detected [30]. These nodes can be vehicles, but also MEC servers. To determine whether a node is misbehaving, various detectors can be used to analyse the behaviour of the node or the data it sends. Misbehaviour in this context refers to a node sending incorrect data, such as position data, so we focus on the veracity of the data. Based on the results of these detectors, the trustworthiness of the node can be increased or decreased. In the following, possible detectors are described. Each detector can either be used as a separate trust source, or all detectors can be used together to determine if the corresponding node is misbehaving, resulting in one trust source. *Plausibility check* Depending on the type of data provided by a node, different approaches are possible to check the plausibility of the data. For example, a position value could be compared with other inputs, such as a map, to check whether the position is on a road or not. *Consistency check* Depending on the type of data provided by a node, different approaches are possible to check the consistency of the data. For example, the data could be compared with other inputs from the past; a position value could be compared to values received a few milliseconds ago to verify that the provided position is consistent with those provided in the past. *Redundancy check* Redundancy checks can be used when information about another vehicle is received from several nodes. Depending on the type of data, the inputs provided by the nodes can be compared to determine if one of them is providing wrong information and thus misbehaving. *Misbehaviour reports* Misbehaviour Reports (MR) are used in the context of V2X communication. When a vehicle receives a message from another vehicle, the misbehaviour detection system checks if the data in this message is valid. If the data is not valid, a MR is created by the vehicle running the misbehaviour detection system. The MR contains the message received from another vehicle that has activated a misbehaviour detector of the misbehaviour detection system. The MR can be sent to another node to provide evidence about the misbehaving vehicle. Based on this report, the trustworthiness of the misbehaving node can be adjusted.
Reputation based system	Based on observations of a node's behaviour, a reputation is established. This reputation can build on referrals or ratings of other nodes in the network, which are created, for example, based on the results of a misbehaviour detection system. In addition, the reputation can be built from personal experience with that node. In this way, a rating of a node's past behaviour is generated. Based on this reputation, the trustworthiness of the node is adjusted.

Spoofing detection	Depending on the sensors used in the nodes, various spoofing attacks are possible that can cause the sensor to produce false sensor output. For these attacks, detection mechanisms exist that can determine the presence of spoofing. Based on the result of the spoofing detection, the trustworthiness of the values provided by the sensor should be adjusted. For example, in the context of GNSS, there are several works that use machine-learning algorithms to detect spoofing attacks on GNSS sensors.
Intrusion detection system (IDS)	A network-based IDS monitors a network of systems for malicious activities or suspicious behaviour. All malicious activity or behaviour is collected and combined to determine if a malicious activity has truly occurred or if it is a false alarm. In this way, the IDS can detect malicious entities within the network, which would make the corresponding node or system less trustworthy. Such entities could be, for example, ECUs within an in-vehicular network.

4.1.5 Sources of Trust from a Safety Point of View

The safety of systems generally and especially in the automotive sector is meant to apply measures and methods to make sure that no severe harm is generated or caused by the system. "Safety" is based on two general considerations. The first part is called functional safety, which tries to avoid internal system failures – in contrast to "security" problems largely coming from outside the system. The second conceptual way is called Safety of Intended Function (SOTIF), which makes sure the principal constraints and boundaries of the system are taken into account in the system design, development, and validation. Here, the automotive industry has two relevant standards/norms: ISO26262 [23] for functional safety and ISO21448 for SOTIF [25]. Additionally, for addressing infrastructure elements, such as RSUs, IEC 61508 [26] provides a general framework that covers the safety aspects of these components comprehensively. Overall, the measures and methods defined in those norms can also be seen as a kind of trust source and are summarised in the following table, not all detailed measures and methods are listed as this would be out of context of this document, but they can be found in the respective norms. Instead, some general concepts are listed.

Concept	Description
Item definition and decomposition	To be able to avoid failures it needs to be clear in what parts of the system certain kinds of errors might occur. To detect this, the overall system is decomposed into its major components, and the parts contributing to a certain function that might generate errors are identified.
Hazard analysis and risk assessment	As functional safety has the goal to avoid harm to persons, Hazard Analysis and Risk Assessment (HARA) analyses which errors might occur in the system and how severe they are. This is done in a systematic way using standardised objective values like the probability of an error (exposure), the severity, and the controllability. Knowing this makes it possible to concentrate avoiding hazards systematically. HARA comes up with a certain ASIL which determines the measures to be taken in the following steps.
System-level safety concept	At system level, the norm mandates the generation of a functional safety concept – a system architectural design based on a detailed requirement analysis to avoid systemic failures, measures to control random hardware failures, hardware and software specifications/interfaces for production and operation, and verification concepts. Examples for such concepts are safety mechanisms at hardware and software level, concepts for detection of faults external elements, the definition of safe states, and degradation concepts when errors are detected.

Hardware-level safety concept	The hardware (HW)-level safety concept mainly aims to make sure that no random hardware failures occur. This includes the hardware implementation of the system safety concepts, analysis of potential hardware faults and their effects, and coordination with software development. Examples for hardware design objectives derived here are hierarchical designs, precisely defined hardware interfaces, avoidance of complexity, maintainability, and testability.
Software-level safety concept	The software (SW)-level safety concept mainly aims to avoid systematic errors in the generation and maintenance of the software in the system. Examples of requirements to the software development are comprehensibility, consistency, simplicity, verifiability, modularity, abstraction, encapsulation, and maintainability. Stated principles for this include the hierarchical structure of the software components, restricted size of interfaces and components, restricted use of interrupts, etc. Furthermore, verification methods are mentioned such as the design walkthrough, inspection of design and coding, prototype generation, data and control flow analysis, etc.
Production and operation concepts	To make sure that the system components (HW and SW) are maintained at the highest level, the norm covers concepts and requirements during production and operation.
Validation and Verification	To make sure that the quality of the overall system is as specified and as wanted, a detailed validation and verification is foreseen and mandated. This includes audits for design, development and operation as well as structured and systematic testing at different levels of the system.
Identification and evaluation of hazards caused by the intended function	The potential hazards related to the SOTIF are to be systematically identified and evaluated. This takes into account specification of acceptance criteria (e.g., a validation target) to evaluate the design in the validation phase, and that possible hazardous events caused by reasonably foreseeable misuse of the function (by the user) are identified and evaluated.
Identification and evaluation of triggering events	This contains the dentification of events that can trigger potentially hazardous behaviour, and evaluation of their acceptability with respect to SOTIF.
Functional modification to reduce SOTIF-related risks	This part of the SOTIF contains the development activities of the functional modifications to reduce the SOTIF-related risks, and includes the identification and allocation of measures to avoid, reduce, or mitigate them, the estimation of the effect of the SOTIF-related measures on the intended function, and the improvement of the information required.
Definition of SOTIF-related verification and validation strategy	In this part of the norm a verification and validation strategy is to be defined to support SOTIF; ensuring that the necessary evidence is generated and procedures to provide that are developed, and it includes validation of system- and functional robustness – how well the sensors and related algorithms work in the environment.
Validation and verification of the SOTIF	The system and components (sensors, algorithms and actuators) are to be verified using sufficient testing to show that they behave as expected for known hazardous scenarios and reasonably foreseeable misuse (derived from previous analyses and knowledge). The functions of the system and the components (sensors, decision-algorithms and actuators) are to be validated to show that they do not cause an unreasonable level of risk in real-life use cases. This requires evidence that the validation targets are met.

4.1.6 Trust Sources Related to Sensor Data Integrity

The integrity concept is based on two standards, FUSA [23] and SOTIF [25], and is not to be confused with data integrity from a security point of view. This helps to establish a methodology to minimise the risks associated with system failures and environmental conditions. Some of the key parameters to consider in the context of V2X are the position, speed, etc. contained in the vehicle status data. This information exchanged through the V2X channel should also be trusted by the receiving entities. Typical properties related to sensor data integrity concepts are identified as the protection level, alert limits, integrity risks, and output latency. This concept has been elaborated

in more detail for the position data in [31].

Concept	Description
Protection level	A statistical upper-bound of the estimated error that ensures that the probability per unit of time of the hazardous misleading event (defined as the true error being greater than an Alert Limit (AL), and the Protection Level (PL) being less than or equal to the AL for longer than the time to alert (TTA) is less than a specific threshold. AL being the maximum PL tolerable by the application. The PL is a real-time and dynamic quantity which may vary from one output epoch to the next [32].
Integrity risk	The Integrity Risk (IR) is the rate at which a hazardous misleading event happens [32].
Time to alert	Time to Alert (TTA) is the maximum allowable elapsed time from when the error exceeds the bound until an alarm flag must be issued [29].
Output latency	Time of Output is described by the timestamp at which the positioning terminal provides its output; the difference between this parameter and the action timestamp is called the Output Latency [32].

4.2 Verifiability of Evidence for Evaluation of Trustworthiness

The process of evaluating trust and trustworthiness involves the ability to assess various properties applicable to a trust relationship. This evaluation relies on evidence from sources of trust, providing verifiable information about the corresponding property. So, the next important step is to verify this evidence. Verifiability involves the trustee providing evidence justifying the trustor's decision to trust them. The element of verifiability is a key part of the approach that aims to define precise conceptions of applicable and relevant properties for evaluating a trustee, and how they can demonstrate these properties. Additionally, it specifies the evidence required by trustors leading to a positive evaluation of trustworthiness, and ensuring a robust trustworthiness assessment process.

Verifiability, thus, is crucial for addressing key questions such as how a given trustee in a given trust relationship can, for example, exhibit the property of integrity. What evidence is needed to demonstrate that this trustee indeed delivers data with integrity? How can this evidence be made available so it can be assessed by the trustor?

Evaluating trustworthiness, therefore, involves verifying trust relationships through a comprehensive assessment of relevant properties and trust sources. The methodology emphasises verifiability, enabling trustors to make informed decisions about the trustworthiness of trustees based on concrete evidence. Verification of evidence is related to the use of "trust anchors", since they are the starting point for verifying the chain of trust. The National Institute of Standards and Technology (NIST) provides several definitions of a trust anchor, reflecting its multifaceted nature in different contexts. However, broadly speaking, the following different types of trust anchor can be met within any governance model, even if they are not obvious:

- Institutional/Legal Trust Anchors: Legally binding agreements and regulations that is mandatory across the nation or jurisdiction under the rule of law, e.g., Identity.
- Data (Credential) Trust Anchors: Authoritative data sources can be trust anchors, upon which the overall trust framework and operational system depend. The term "authoritative" means that the data is legally admissible in a court of law, e.g., Certificate Authorities (CA) for PKI certificates.
- Technical Trust Anchors: These anchors provide the root of technological (e.g., cryptographic) trust, bind entities and attributes to data subjects and data principals, as well as to actors within the systems that operate the trust framework.

The level of assurance provided by a given trust anchor is directly related to the confidence in the verification of evidence. For example, a robust trust anchor, such as a well-secured CA with rigorous issuance policies, contributes to a high level of assurance, thereby offering greater confidence in the verified evidence. Traditional hierarchical trust chains and emerging trust frameworks take it for granted that trust anchors are reliable and attestation/validation are objective and with absolute certainty.

However, in emerging C-ITS scenarios, verifiability of evidence is not enough to create

certainty. In such environments, trustworthiness of a source depends on evidence that holds an inherent level of uncertainty. Consider for example evidence related to the behaviour of entries listed in Section 4.1.4. In particular, the MBD system might provide verifiable evidence on the behaviour of another system, but the misbehaviour report exhibits an uncertainty by nature. Another source of uncertainty stems from the calculation of evidence (e.g., reputation) by fusing indirect evidence obtained via referral paths (source A provided a reputation score about source B to source C). Therefore, there is a need to be able to measure the correctness of attributes and the trustworthiness of the source in the presence of measurable confidence and uncertainty. That means, being able to reason with uncertainty based on evidence becomes a fundamental approach in trust assessment. This method involved collecting, analysing, and making decisions based on data and information, which might not always be complete or may carry some level of ambiguity.

There are various approaches and methods that have been proposed in the literature to ascertain information in uncertain and unpredictable conditions that could be potentially used for assessing trust, e.g., Fuzzy Logic [33], Bayesian Probability [34], Dempster-Shafer Theory [35], and Subjective Logic [36]. Bayesian reasoning plays a critical role in managing the inherent uncertainty in the evidence. This probabilistic approach allows for the integration and updating of trust assessments as new information becomes available. Bayesian methods are exceptionally suited to this task because they provide a structured way to update the probability of a hypothesis in light of new evidence. This is particularly pertinent in dynamic systems like C-ITS, where conditions and contexts can change rapidly, and decisions need to be made with the best available information. Bayesian reasoning helps in adapting to these changes by continuously recalibrating the trust assessments based on the latest available evidence. In Bayesian statistics, probability values are used as the fundamental measure of uncertainty. However, this type of probabilistic logic does not allow for seamless model situations where different agents express their beliefs about the same proposition. Dempster-Shafer Theory and Subjective Logic explicitly integrate the subjective nature and ownership of beliefs in its formalism, allowing the combination of different beliefs about the same proposition. Recently, CONNECT project has argued specifically for the advantages of Subjective Logic in assessing trust in more complex trust networks and showed how it can be used to build a Trustworthiness Level Expression Engine (TLEE), and how it can be applied to the automotive domain [37].

4.3 Trust Assessment

The quantification of trust involves a complex and multi-dimensional approach to assessing trustworthiness, which is critical for ensuring safety and security in automated and connected vehicles.

Trust is assessed for a trust relationship in a given context. As previously explained, the trust relationship is a directional relationship between two trust objects, the trustor and a trustee. The trust relationship is always defined in relation to a concrete property. For example, even if we have the same trustor and trustee, the trust relationships would be different, depending on the properties based on which we want to assess trustworthiness. In Section 3.4 we gave a long list of such properties . Of course, we

cannot create trust relationships based on all the properties that we have listed, and this might imply a hierarchy between the trust properties. For example, we can assess "integrity" as a distinct trust property and respectively build a trust relationship for that property. However, there are properties like functional "safety" or "reliability" that can be measured and are causally related to (increased or reduced) trustworthiness as well. Increased trustworthiness in the system can ultimately increase the property of safety. Lastly, depending on the trust properties of interest, different trust sources are selected to do the trustworthiness assessment and quantify the trust opinion of the trust relationship.

The output of the trust assessment is an opinion, which is calculated dynamically, since evidence is constantly changing. We call this the Actual Trust Level (ATL) compared to the Required Trust Level (RTL), which quantifies the level of trustworthiness that is desired or indeed required in order to proceed with the act of trust. So, we can define ATL and RTL as follows:

- The ATL reflects the result of an evaluation of a specific (atomic or complex) proposition for a specific scope. It quantifies the extent to which a certain node or data can be considered trustworthy based on the available evidence.
- The RTL reflects the amount of trustworthiness of a node or data that an application considers required in order to characterize this object as trusted and rely on its output during its execution.

More specifically, the risk assessment serves as a foundation for calculating the RTL [38]. The RTL can be dynamically updated during run-time, if new vulnerabilities are identified. This RTL, in essence, represents a baseline for the minimum required trustworthiness level, while it further identifies the attributes that need to be attested during run-time for the ATL. Assessing the trustworthiness of some data boils down to computing and comparing the ATL with RTL. If ATL is bigger than RTL, we can proceed with the act of trust for this data, since the corresponding data source meets the required level of trustworthiness for the intended function.

5 Conclusions

The quantification of trust involves a complex and multi-dimensional approach towards assessing trustworthiness, which is critical for ensuring safety and security in automated and connected vehicles. In order to address this challenge, we sought to establish a common definition of the related concepts in this document. The following summarises the main discussion points elaborated in the White Paper:

- First and foremost, the paper defines the concept of trust and trustworthiness in the connected and automated vehicle domain.
- Trust is assessed for a trust relationship in a given context. Trust is a directional relationship between two trust objects – the trustor and trustee. Notions of a trust network and different types of trust relationships (direct, derived, functional, and referred) are developed.
- The trust relationship is always defined in relation to a concrete property. In this document we list and define several properties that can be used in the context of connected and automated vehicles.
- The evaluation of these properties relies on evidence from sources of trust, which provide verifiable information about the corresponding property. So, in order to assess a specific trust relationship, it is important to include the evidence that is necessary to assess the particular property. This document presents a list of potential trust sources from several categories, such as security, safety, etc.
- The next important step is to verify this evidence, but this is not enough. In C-ITS and CAV applications, trustworthiness of a source depends on evidence that holds an inherent level of uncertainty. Therefore, being able to reason with uncertainty based on evidence becomes a fundamental approach in trust assessment.
- Evidence-gathering thus involves collecting, analysing, and making decisions based on data and information, which might not always be complete and may carry some level of ambiguity. Here, we describe several valuable tools for achieving this, but presenting concrete solutions is out of scope for this document, and remains the focus of work for future 5GAA Work Items.

6 References

[1] 5GAA, "C-V2X Use Cases Volume II: Examples and Service Level Requirements," Oct. 2020.

[2] ETSI GS MEC 003, "Multi-access Edge Computing (MEC); Framework and Reference Architecture," Jan. 2019.

[3] 5GAA, "MEC for Automotive in Multi-Operator Scenarios," March 2021.

[4] 5GAA, "Cybersecurity for Edge Computing," 2023.

[5] 5GAA, "C-V2X Use Cases and Service Level Requirements Volume I," Dec. 2020.

[6] ETSI TS 101 539-2 V1.1.1, "Intelligent Transport Systems (ITS); V2X Applications; Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification," 2018.

[7] CAR 2 CAR Communication Consortium, "Guidance for day 2 and beyond roadmap," Jul. 2021.

[8] CONNECT Deliverable D2.1, "Operational Landscape, Requirements and Reference Architecture - Initial Version," 2024.

[9] Z. Zhong, M. Nejad and E. E. Lee, "Autonomous and Semi Autonomous Intersection Management: A Survey," *IEEE Intelligent Transportation Systems Magazine,* vol. 13, no. 2, pp. 53 - 70, 2020.

[10] P. Lytrivis , V. Sourlas , V. Filippo and A. Danilo, "Specification of Use Cases," ICT4CART, 2020.

[11] M. Cheng, C. Yin, J. Zhang, S. Nazarian, J. Deshmukh and P. Bogdan, "A general trust framework for multi-agent systems," in *AAMAS '21: Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, 2021.

[12] K. Kim, J. S. Kim, S. Jeong, J.-H. Park and H. K. Kim, "Cybersecurity for autonomous vehicles: Review of attacks and defense," *Computers Security,* vol. 103, Apr. 2021.

[13] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner and T. Kohno, "Comprehensive Experimental Analyses of Automotive Attack Surfaces," in *USENIX Security Symposium*, 2011.

[14] L. Wouters, B. Gierlichs and B. Preneel , "My other car is your car: compromising the Tesla Model X keyless entry system," *IACR Transactions on Cryptographic Hardware and Embedded Systems,* vol. 201, no. 4, pp. 149-172, 2021.

[15] H. Li, R. Lu, J. Misic and M. Mahmoud, "Security and privacy of connected vehicular cloud computing," *IEEE Network,* vol. 32, no. 3, p. 4–6, 2018.

[16] A. Di Maio, M. R. Palattella, R. Soua, L. Lamorte, X. Vilajosana, J. Alonso-Zarate and T. Engel, "Enabling SDN in VANETS: What is the impact on security?," *Sensors* , vol. 16, no. 12, 2016.

[17] J. Audun, R. Hayward and S. Pope, "Trust network analysis with subjective logic," in *Conference Proceedings of the Twenty-Ninth Australasian Computer Science Conference (ACSW 2006)*, 2006.

[18] ISO/IEC TS 5723:2022, "Trustworthiness – Vocabulary," 2022.

[19] ISO/IEC 22624:2020, "Information technology – Cloud computing – Taxonomy based data handling for cloud services," 2020.

[20] ITU-T Y.3057, "A trust index model for information and communication technology infrastructures and services," 2023.

[21] D. Fernandez Llorca and E. Gomez, "Trustworthy Autonomous Vehicles," 2021.

[22] European Commission, "Cooperative, connected and automated mobility (CCAM), Final report of the single platform for open road testing and pre-deployment of cooperative, connected and automated and autonomous mobility platform (CCAM Platform)," 2021.

[23] 5GAA, "Safety Treatment in Connected and Automated Driving Functions Report," Mar. 2021.

[24] ISO 26262, "Road vehicles – Functional safety," 2018.

[25] ISO 21448, "Road vehicles – Safety of the intended functionality," 2022.

[26] IEC 61508-1, “Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 1: General requirements,” 2010.

[27] ISO/IEC 29167-19, “Information technology – Automatic identification and data capture techniques – Part 19: Crypto suite RAMON security services for air interface communications,” 2019.

[28] ISO/IEC 27000, “Information technology – Security techniques – Information security management systems,” 2018.

[29] 3GPP TS 38.305, v 17.1.0, “NG Radio Access Network (NG-RAN); Stage 2 functional specification of User Equipment (UE) positioning in NG-RAN”.

[30] 5GAA, “Misbehaviour Detection White Paper,” May 2022.

[31] 5GAA, “Trustable Position Metrics for V2X Applications,” Sept. 2023.

[32] CEN-CENELEC EN 16803, “Use of GNSS-based positioning for road Intelligent Transport Systems (ITS)”.

[33] A. Alnasser and H. Sun, “A fuzzy logic trust model for secure routing in smart grid networks,” *IEEE Access,* vol. 5, p. 17896–17903, 2017.

[34] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari and D. B. Rubin, “Part I Fundamentals of Bayesian Inference,” in *Bayesian Data Analysis*, CRC Press, 2014, p. 4–29.

[35] G. Shafer, A mathematical theory of evidence, Princeton University Press, 1976.

[36] A. Jøsang, Subjective Logic: A Formalism for Reasoning Under Uncertainty, Springer Publishing Company, 2016.

[37] CONNECT Deliverable D3.1, “Architectural Specification of CONNECT Trust Assessment Framework, Operation and Interaction,” 2024.

[38] CONNECT Deliverabe D3.2, “Trust & Risk Assessment and CAD Twinning Framework (Initial Version),” 2024.

Annex A Abbreviations

For the purposes of the present document, the following symbols apply:

3GPP	3rd Generation Partnership Project
5GAA	5G Automotive Association
AD	Autonomous Driving
ADAS	Advanced Driver-Assistance System
AIM	Autonomous Intersection Management
AL	Alert Limit
ASIL	Automotive Safety Integrity Level
ATL	Actual Trust Level
CA	Certification Authority
C-ACC	Cooperative Adaptive Cruise Control
CAM	Cooperative Awareness Message (EN 302 637-2)
CAN	Controller Area Network
CAVs	Connected Automated Vehicles
CCAM	Cooperative, Connected and Automated Mobility
CFA	Common Flow Attestation
C-ITS	Cooperative Intelligent Transport Systems and Services
CPM	Collective Perception Message (TS 103 324)
CVE	Common Vulnerabilities and Exposures
C-V2X	Cellular Vehicle-to-Everything
ECU	Electronic Control Unit
ETSI	European Telecommunication Standards Institute
FUSA	Functional Safety
GNSS	Global Navigation Satellite System
HARA	Hazard Analysis and Risk Assessment
ICRW	Intersection Collision Risk Warning
ICW	Intersection Collision Warning
IDS	Intrusion Detection System
IMA	Intersection Management Assist
ISO	International Organisation for Standardisation
ITS	Intelligent Transport Systems and Services
ITU	International Telecommunications Union
MEC	Multi-access Edge Computing
MNO	Mobile Network Operator
MR	Misbehaviour Report
NIST	National Institute of Standards and Technology
OBU	On-Board Unit
OEM	Original Equipment Manufacturer
OS	Operating System
OTA	Over-the-Air
PKI	Public Key Infrastructure
PL	Protection level
PLMN	Public Land Mobile Network
RSU	Road Side Unit

RTL	Required Trust Level
SOTIF	Safety Of The Intended Functionality
TEE	Trusted Execution Environment
TTA	Time-to-Alert
V2X	Vehicle-to-Everything communication

5GAA is a multi-industry association to develop, test and promote communications solutions, initiate their standardisation and accelerate their commercial availability and global market penetration to address societal need. For more information such as a complete mission statement and a list of members please see https://5gaa.org


5GAA
Automotive Association