PRISSMA Project
Plateforme de Recherche et d'Investissement pour la Sûreté
et la Sécurité de la Mobilité Autonome
04/2021 - 04/2024

[L5.3] CYBERSECURITY OBJECTIVES FOR AUTONOMOUS VEHICLE SYSTEMS

DEFINITION DES OBJECTIFS DE CYBERSECURITE DANS LES ECOSYSTEMES VEHICULAIRES AUTONOMES

Main authors: Boutheina BANNOUR, Jean CASSOU-MOUNAT, Virginie DENIAU, Richard DENIS, Christophe GRANSART, Sammy HADDAD, Reda YAICH

Keywords: Security Objectives, Security assurance

Abstract: In this deliverable we define the security objectives for the Autonomous Road Transports Systems (ARTS) using AI. Those security objectives consist in:

- identification of the threats to be mitigated (based on the state of the art defined in L5.2),
- definition of security requirements to be implemented to mitigate the chosen threats,
- and definition of assurance (evaluation) activities to be performed to evaluate that the system and its components fulfil those requirements.

We also discuss the associated framework to be used in term of responsibilities and competences to run and manage the evaluations, their results and validity over time.

Résumé : Dans ce document nous présentons l'ensemble des objectifs de sécurité retenus pour valider la sécurité des Systèmes de Transport Autonomes Routiers à base d'IA. Pour cela, nous avons défini l'ensemble des éléments suivants :

- les risques à prendre en compte pour l'évaluation de la sécurité,
- l'ensemble des exigences de sécurité permettant de contrer ces risques,
- et enfin l'ensemble des tâches d'évaluation permettant de valider que le système vérifie ces exigences et contre bien les risques retenus.

Dans ce document, nous définissons aussi le schéma d'évaluation associé à nos objectifs à savoir l'ensemble des rôles, droits et compétences nécessaires pour l'évaluation de la conformité du système et de ses composant à l'ensemble de ces exigences.

Table of content

1 Introduction 4
2 Threats and security objectives identification 6
2.1 Systems 7
2.1.1 Central ITS stations 7
2.1.2 Developers' premises 10
2.1.3 The Autonomous Road Transport System (ARTS) 11
2.2 Technical components 13
2.2.1 Main ARTS technical components 13
2.2.2 AI sub systems 22
3 Security requirements 27
3.1 Systems requirements 27
3.1.1 ISO 27000 series and ISO 21 434 28
3.1.2 ARTS requirements 30
3.1.3 AI developments site requirements 31
3.1.4 PRISSMA requirements 31
3.2 Technical components requirements 31
3.2.1 C-ITS station (OBU/RSU/VRU/Central station) 31
3.2.2 AI requirements 34
4 Assurance requirements 36
4.1 Security evaluation State of the art 36
4.1.1 ISO 15408 - The Common Criteria for Information Technology Security Evaluation (CC) 37
4.1.2 CSPN 41
4.1.3 Automated Road Transport System (ARTS) regulation and existing assurance requirements 41
4.1.4 System audits 43
4.2 PRISSMA assurance requirements 44
4.2.1 Site assurance requirements 45
4.2.2 Components 45
5 PRISSMA assurance scheme 51
5.1 Evaluation output 52
5.1.1 Evaluation output format 53
5.1.2 Evaluation output validity period 53
5.2 Scheme 54
5.2.1 Actors and responsibilities 54

5.2.2 Certification 57
5.2.3 Scheme maintenance 57
5.2.4 Assurance continuity 58
5.2.5 Scheme implementation 58
6 Annex: Acronyms 60
7 Bibliography 62

1 Introduction

The goal of task 5.3 is to define security objectives for AI based autonomous systems (later referred to as Autonomous Road Transport Systems or ARTS in this document)[1][2].The first activity of the task has been to clarify and define more precisely the task purpose.
In fact, "Security Objectives" can have different meaning in different context or for different actors (managers, administrators, etc.), e.g.: security measure implementation requirements, list of threats or risks to be mitigated, regulation compliance, standard conformity, etc. Thus, we decided to define the objectives we though would best match the project needs.
Since the two main focuses of WP5 are (i) the definition of communication performances objectives associated to the means to measure them and (ii) the definition of a dedicated security assurance framework for ARTS. In order to define an assurance framework, we need to tackle the three following challenges:

- Defining proper security targets, i. e. identifying product or system parts to be evaluated, the threats to be mitigated, the assumptions used and the definition of the target environment.
- Defining the evaluation activities adapted to evaluate the security targets.
- Identify requirement for evaluation supervision, evaluators qualification, evaluation results maintenance, framework ownership, etc.

For instance, those elements have been defined by the Common Criteria [1]. Common Criteria is the only standardized and internationally recognized evaluation scheme designed for cybersecurity products. It benefits from a high level of maturity. It has been maintained by a large community since 1998. That's why we chose to refer here to their definition of security objectives and assurance security concepts (e.g. security target, target of evaluation, security problem definition, etc.). So, CC defines security objectives as: "the intended solution to the security problem", where the security problem is the set of threats to the Target Of Evaluation's (TOE) assets, the set of security policies to be enforced (regulation requirements, enforced best practices) and the assumptions made for the evaluation.
That's why, based on this definition in this deliverable we define in section 2 (i) threat agents, (ii) threats, (iii) security policies if required and (iv) security objectives, for the global system and its most critical components.
However, security objectives only provide high level objectives for a cybersecurity evaluation. It does not specify any guidelines on how to implement nor verify them. Since PRISSMA objective is to define all the necessary components of an evaluation framework, only defining security objectives would not be sufficient to reach that goal. It is important not only to define the security objectives to be validated, but then also to define the security requirement that will enforce those objectives, which we do in section 3. Objectives can be fulfilled in many ways, and the more possibilities to fulfil them the less optimized the evaluation scheme will be. That's why it is important to fix them as much as possible. In fact, if all possible implementations are allowed, then evaluation tasks will have to be adapted to each of those solutions. This implies to perform almost exclusively ad hoc evaluations for every solution, limiting by consequence the comparability of their results. This would impact the

[1] *In this deliverable even if we cover assurance challenges of various type of autonomous driving systems not all corresponding to [2] definition, for the sake of vocabulary consistency and since it includes ARTS, we will use the same vocabulary and use the term ARTS as well as other definition presented in section 4.1.3.*

obtained confidence in the evaluation results. It also wouldn't help to enforce security interoperability yet required when independently evaluated solutions are combined in one single ARTS.
Afterwards in section 4, we define also the assurance activities to evaluate those requirements to provide a complete security assurance framework for AI bases autonomous transport systems.
This security assurance framework is one component of the global PRISSMA evaluation framework. It will be used in addition to all the other functional and safety validation process defined by the project.
As defined by the project description, PRISSMA targets the validation of ARTS with autonomous level 4 ([3]). However the present security assurance framework does not limit its scope to such ARTS as defined by French decree n° 2021-873 [2]. Our generic assurance framework is applicable to any autonomous systems without restriction (e. g. used on limited or predefined roads) or any autonomous level 5 SAE ([3]).This stems from the facts that (i) our assurance assessment is based on ad hoc risk analysis for a specific targets, associated to (ii) predefined security requirements that apply to most representative components to be completed by ad hoc once derived from the risk analysis and (iii) adapted assurance assessment technics applicable to any system components based on the level of risk they are associated no matter what they are (from low to critical) and their level of complexity.
The scope of the assurance framework is the complete ARTS and not only its components. We provide specific security and assurance requirements for the whole systems and its different components. Knowing that the full system assurance (presented in section 4.2.1) relies among other things on its critical components assurance (presented in sections 4.2.1 and 4.2.2). So, this framework is not limited to the security assurance of AI components. However, we present dedicated requirements for AI (presented in sections 3.1.3 and 3.2.2).
Our choice to define an adaptable assurance framework comes from the fact that as soon as driving systems are connected to Information Technology (IT) systems, the risk of critical events to be triggered by attackers to influence a vehicle behaviour and potentially be the cause of an accidents arise. And this, no matter their level of automation or the environment they are in. Thus, the need of (potentially) high assurance for the most sensitive system components is shared by any such system.
Our current proposition is generic enough to be agnostic of a specific ARTS architecture, specific risks, or automation level. At the same time, we provide predefined catalogues of (i) security requirements to be applied to standardized components and (ii) assurance requirements for the different type of components adapted to all risks.
It is not meant to replace any existing regulation requirements or assurance propositions. To our knowledge, its flexible designed makes it possible to integrate it to most existing homologation or type approval process. Those catalogues are made to match most cybersecurity evaluation needs identified so far. But if they appear to be insufficient or have to be updated due to state of the arts updates, they would be adequately adapted we also discuss the specifications of the framework management process in section 5.2.1.4.

2 Threats and security objectives identification

In task 5.2 of the PRISSMA project, we have produced a state-of-the-art of attacks and threats related to autonomous driving systems components as well as AIs in general. To perform this state-of-the-art we have first defined a generic architecture presenting the recurring component of such systems (Cf Figure 1).

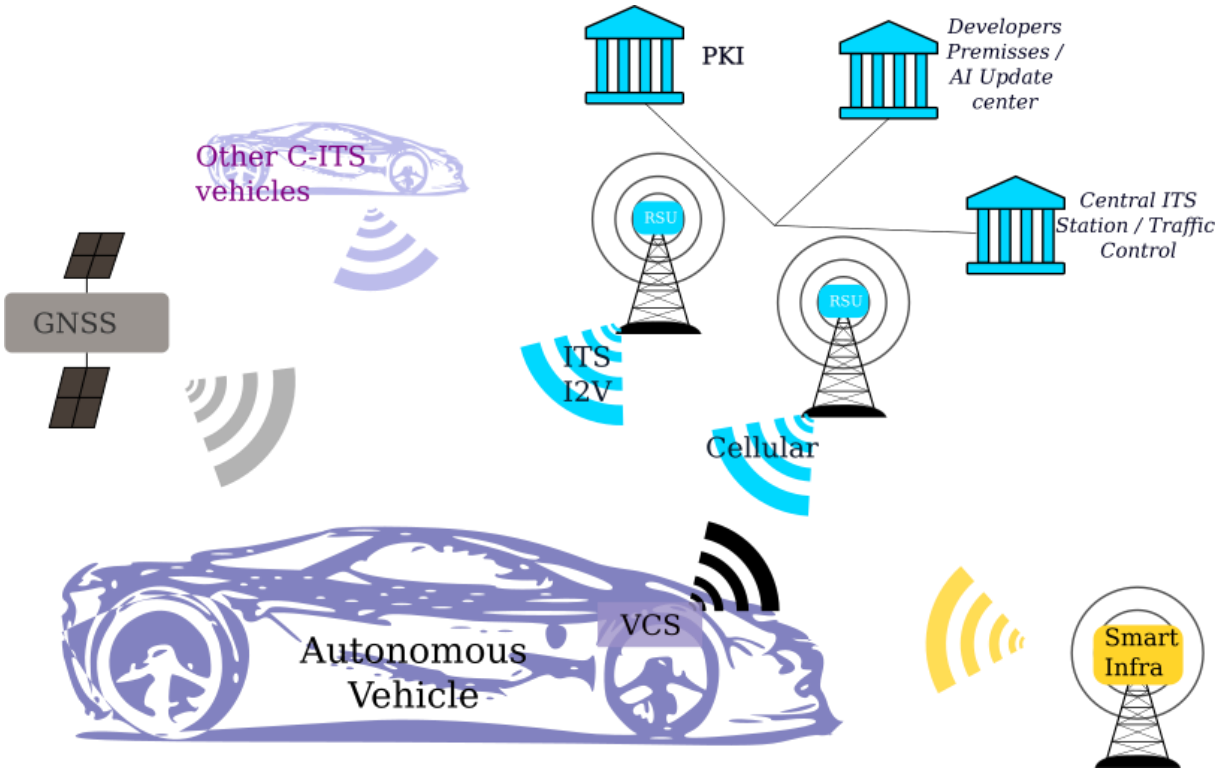


Figure 1ARTS generic architecture

In this architecture we have identify the following main components:

- **Autonomous Vehicle (AV)** equipped with sensitive communication (cellular and V2X) and AI
- **RoadSide Units (RSU)** providing communication gateway functions to the Autonomous vehicle toward the Internet or cellular network thus providing connectivity means to any central ITS station, PKI, Developers premises or the Internet.
- **GNSS** providing time, positioning services to the autonomous vehicle.
- **Central Stations** which gather and provides ITS data to the vehicle and the rest of the infrastructure or potential remote management of the AV.

As we can see the nature of those components is very different in terms of size, technologies, connectivity properties, importance for the final service, etc. The threats associated to those elements are thus as different as the nature of those components. A first level of analysis made in this task identified 4 different sets of components based on the nature of the threat they face and the expected complexity of assessing their resiliency to those threat. In this document we will follow this classification:

- **Systems**
  - The **complete ARTS infrastructure** itself

    - Any **ARTS IT systems** (or **Systems**) components characterized by the fact that they are large (tens to hundreds of PCs), complex (using various sets of software), and possibly multi-tenant system systems of systems.
- **Technical components**
    - **Main ARTS technical components** (vehicle, road infrastructure components including RSUs and infrastructure sensors and other equipped road users) communicating with each other for more efficient and safer transport service.
    - **AI subsystems** (gathered in the system: vehicle, central station, road infrastructure sensors, etc.) which is the core software enabling autonomous capacities (decision making, environment objects identification) and for which PRISSMA has a specific focus.

We will define dedicated objectives for each important component of those four sets, requirements, and assurance approaches to handle their specificity.
In this section we start by defining the security objective associated to those components. We present for each component the following elements:

- Component name
- Description of the component
- Identification and description of its external interfaces
- Threats and associated threat agents applicable to the component and its interfaces
- The security objectives to counter and mitigate the identified threats
- Additional comments are provided when required

Specific approaches should be applied to larger components of the system or the global system itself due to their size and complexity. If all interfaces of a single product (communication unit, sensor, AI, etc) can be tested in a reasonable delay, the same is not true for a complete developer IT system or traffic control system containing tens or hundreds of PC.

2.1 Systems

If important system components will be studied individually later in this document. We first discuss and identify threats to be considered for the global system and its largest components.
ARTS are system of system that will gather several types of elements of different size, complexity, offering different number of interfaces. They will also be owned or managed by different entities alongside the ARTS lifecycle. So, even if separately the system elements can be secured (demonstrated by security assurance method), threats will still apply to their composition (the complete system).
This is the case for (at least) the following large and complex components:

- Central ITS stations
- Developers' premises
- The complete ARTS

Those elements are recurring and strategical components of autonomous ITS architecture based on AI systems. For them, generic security audit approaches should be followed rather than extensive technical audit or vulnerability tests. Such approaches are not adapted to systems with high level of complexity that are updated, evolving, or patched on a daily basis.
We present those specific large components here; more detailed descriptions can be found in [4].

2.1.1 Central ITS stations

COMPONENT	Central ITS Station

DESCRIPTION

Central ITS station gathers and provides ITS data from and to vehicles and the infrastructure. It covers:

1. **Traffic Control Centre (TCC)** sending and receiving "traffic management" data and maintaining a global model of the current traffic status that can be transferred to the ARTS for information. Traffic Management is a well-established activity that do not yet participate nor belong to ARTS. However, in the context of our study different types of TCC can be supporting or integrated in ARTS. We envision that some can be fully part of the ARTS when they only manage the ARTS, or they can be current TCC connected to the ARTS to provide traffic status inputs. In both cases as they provide inputs to ARTS they become part of the assurance framework. The associated requirement and evaluation tasks will depend on their specific interactions with the ARTS to be studied via a risk analysis.
2. **Remote Control Centre** from which Autonomous Vehicle (AV) can be remotely controlled
   - To be considered a central ITS station an IT system shall at least possess a V2X interface or provide a direct communication mean with ITS vehicles.
   - To guarantee the security of V2X communication it must then possess all the elements necessary to be defined for it (Certificates, trust lists, security format conformity, etc.).

INTERNAL INTERFACES AND ASSETS

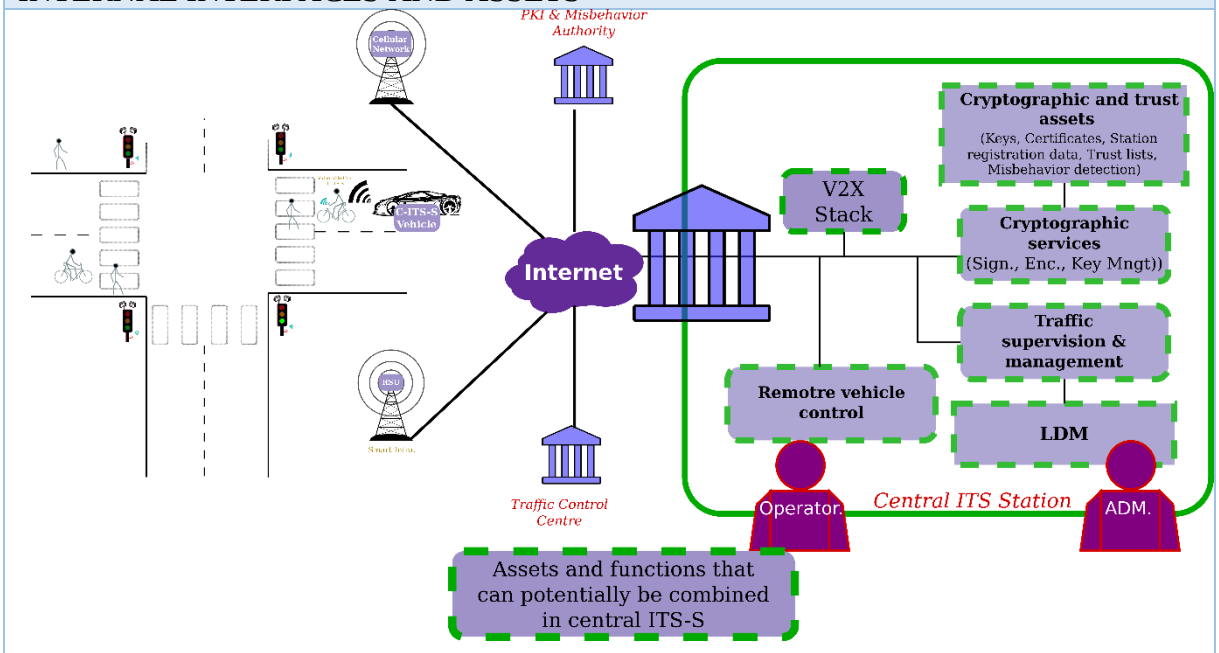


Figure 2 Central ITS station internal architecture

The following assets have been identified for the central stations:

- **DATA ASSETS**
  - ITS cryptographic and trust elements necessary for secure V2X communication
    - Keys
      - Canonical Public Key, Data encryption key, CA private keys
    - Certificates

- CA Certificates, Enrolment Credential (EC), Authorization Ticket (AT), TLM certificate
- Station registration data
  - Canonical ID, ITS-S Profile, Tag, HMAC key, CA Network addresses, DC network address
- Trust lists
  - CPOC Network address, CRL, CTL, ECTL
- Misbehaviour detection report to be sent to misbehaviour authorities if available.
- ITS Data
  - X2V Safety sensitive ITS application data, X2V Sensitive ITS application data, X2V Informative ITS application data, LDM, ITS software

- **FUNCTION ASSETS (depending on station type)**
  - Traffic management for TCC
  - Vehicle remote control for RCC

Thus, as presented in the figure above the Central ITS station will provide with respect to autonomous transport services critical functions two main functional blocks (the asset functions).
The traffic management function will need to interact with the Local Dynamic Map (LDM) and the V2X stack to transmit and receive the ITS data through its v2X interface.
While the remote-control vehicle function will directly interact with the internet interface to communicate with the vehicle.
The V2X communication stack will need to access the cryptography elements and trust list to secure the V2X communications as defined by the ETSI standards (ETSI 107 097 and 103 941).
Some stations can have both roles and thus provide the two functions.

AGENTS AND THREATS

Central station should most probably be connected to the internet. Even if they won't publicly expose their interfaces, they will be prone to face internet attacks (Malware, fishing, privilege escalation, brute force, DoS, etc.). So those components will be subject to classical internet threat and threat agents.

AGENTS:

- Remote internet attacker
  - Attacker sending or intercepting messages between the infrastructure (PKI, central ITS, developer premises) and the vehicles ARTS and other trusted entities (e.g. external TCC) or trying to get unauthorized access to the vehicle or the infrastructure components (data or functionality).
- Local attackers
  - Rogue User

THREATS:

- All possible IT threats: Replay, Man in the middle, Gain of unauthorized access, arbitrary/remote code execution, etc.

SECURITY OBJECTIVES

The Central ITS-Station shall resist to the various existing internet attacks to protect its numerous assets.

COMMENTS

It is clearly not possible to describe here all internet attacks, especially since Central ITS station can take any IT system form (web interfaces, linux/windows/... environments,

virtualized systems (vmware, virtualbox, openstack…) etc.). This is why we separate those large elements for the other more specific ones identified in the next section and for which we have more detailed objectives.
This is also why for such systems we will not require specific assurance evaluations but regular audit process, which have already demonstrated their benefits and suitability for such exercises. This study will have to be combined with requirements identified in [5] when the target is an ARTS as identified by decree n° 2021-873 of the 29th of June 2021

Table 1 Central ITS Station description and objectives

2.1.2 Developers' premises

COMPONENT	Developers premises / AI update repository
DESCRIPTION	
Online and off-line IT system aiming at gathering AI's training data, develop, and update AI models, provides AI update repository.	
INTERNAL INTERFACES AND ASSETS	

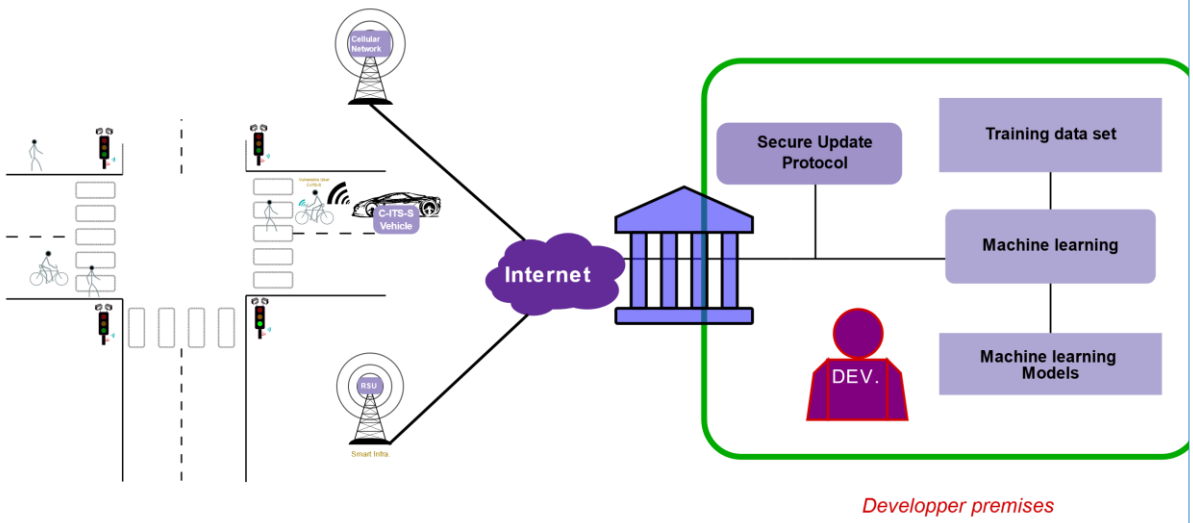


Figure 3 Developers' premises architecture

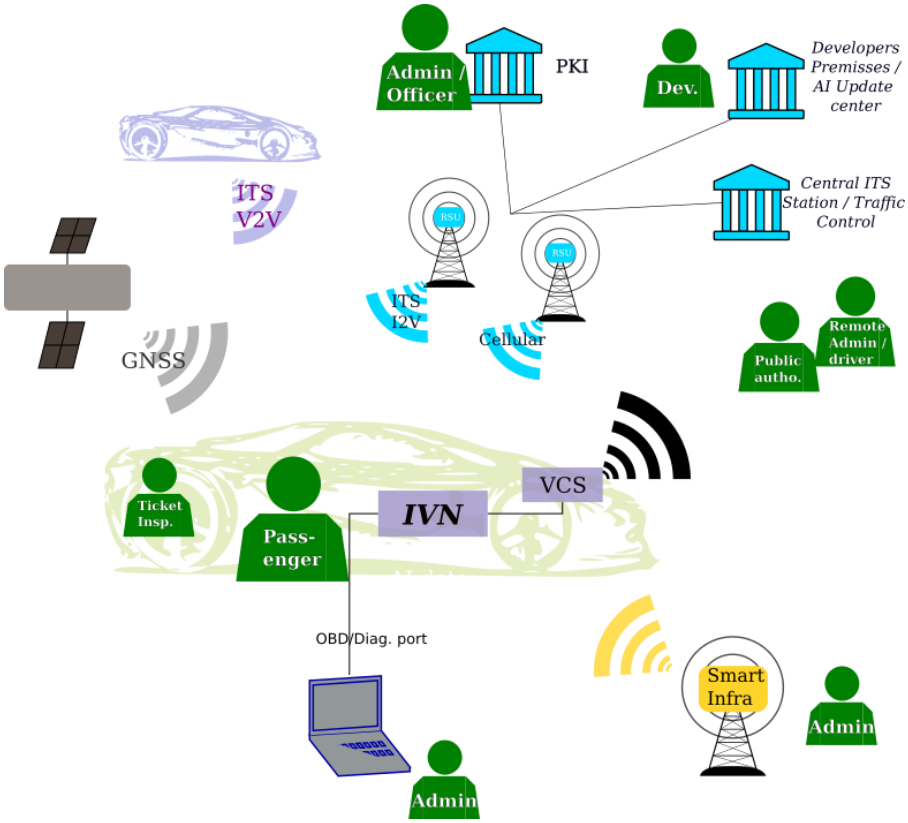
AGENTS AND THREATS
As for central stations, AIs Developers premises should most probably be connected to the internet. Even if they won't publicly expose their interfaces, they will be prone to face internet attacks (malware, fishing, privilege escalation, brute force, DoS, etc.). So those components will be subject to classical internet threat and threat agents. **AGENTS**: • Remote internet attacker ◦ Attacker sending or intercepting messages between the infrastructure (PKI, central ITS, developer premises) and the vehicles ARTS and other trusted entities (e.g. external TCC) or trying to get unauthorized access to the vehicle or the infrastructure components (data or functionality). • Local attackers ◦ Rogue User

THREATS: • All possible IT threats: Replay, Man in the middle, Gain of unauthorized access, arbitrary code execution, etc. • Specific AI developments servers threats: ◦ **Poisoning attacks** can be achieved by rogue user or internet attackers before the training phase by introducing perturbations among the training data to generate a corrupted model. ◦ **Extraction attacks** try to steal the parameters of a remote model in order to reproduce its behaviour or rob confidential information, that could be performed by internet attackers or rogue ITS-S administrators.
SECURITY OBJECTIVES
Developer premises shall resist to the various existing internet attacks to protect its numerous assets.
COMMENTS
Same comment as for central ITS stations.

Table 2 AI developers' premises description and objectives

2.1.3 The Autonomous Road Transport System (ARTS)

COMPONENT	ARTS
DESCRIPTION	
The global autonomous transports service is the complete ARTS required to perform and supervise autonomous transport systems. In this security assurance study, contrary to other PRISSMA evaluation requirements, the present framework does not limit its scope to ARTS as defined by decree n° 2021-873 [2]. This assurance framework can be applied to any autonomous transport systems (up to autonomous level 5 as defined by SAE in [3]) where attackers can impact vehicle driving and cause accidents, thus where cyber threats are critical. The main recurring components of those systems are: • The Autonomous Vehicle (AV) providing transport services to the final users (passenger) • The Global Navigation Satellite Systems (GNSS) providing time and allowing positioning to the autonomous vehicle. This system even if outside of the transport system scope is however one critical input provider for autonomous transport services. • Roadside equipment providing communication gateway functions to the Autonomous vehicles toward the Internet or cellular network and the other system component. They can also include sensor (camera, lidar, radar, connected traffic lights, remote sensors, etc.). • Central ITS Station as seen before in this section 2.1.1; • Developers premisses / AI update repository also seen before in this section.	
INTERNAL INTERFACES AND ASSETS	



We do not provide here the complete list of assets that has already been described in [4], but the complete system contains and manages all the necessary ITS assets such as: ITS Data (V2X, sensor, etc.) cryptographic keys, certificates, registration data, policies, etc.

INTERFACES

The main internal interfaces of the ARTS are the one of its components describe in section 2.2.

AGENTS AND THREATS

Here we do not consider internal threat for each component but rather all external threats agent that can interact with the system without any prior specific physical or technical access.

AGENTS:

- Remote radio attackers
  - An attacker able to emit or receive GNSS, G5, cellular radio signals to intercept, jam, replay, fake messages from or to the vehicle or the infrastructure.
- Remote internet attacker
  - Attacker sending or intercepting messages between the infrastructure elements themselves (RSU, PKI, central ITS, developer premises) or the infrastructures and the vehicles or trying to get unauthorized access to the vehicle or the infrastructure components (data or functionality).

THREATS:

- All possible IT threats: Replay, Man in the middle, Gain of unauthorized access, arbitrary code execution, etc.

SECURITY OBJECTIVES

Developer premises shall resist to the various existing internet attacks to protect its numerous assets. Radio attacks are not considered here since they are already included other components scopes (vehicles and RSU).
COMMENTS
Same as central ITS stations. Also, here we identify the ARTS as the system of system composed of all the elements described in this section, target of the final assurance assessment. So, each of its component will have to be evaluated independently depending on the requirements identified later on (cf. section 3.2) or identified in a dedicated risk analysis of the complete system (cf. section 3.1) depending on their type. Specific technical components are presented in the following section 2.2, that's why we don't provide further details (interfaces, data exchanged between sub-components, etc.).

Table 3 ARTS description and objectives

2.2 Technical components

2.2.1 Main ARTS technical components

COMPONENT	Public Key Infrastructure (PKI)
DESCRIPTION	
The PKI here is the technical component providing the services associated to the Root Certificate Authorities (RCA), Enrolment Authorities (EA), and Authorization Authority (AA) (as defined in [6] and [7]). Each ITS station holds an asymmetric key pair where the public key is part of a digital certificate provided by the PKI. It is the technical root of trust of the ITS communications. Those technical components have to be managed by competent trustworthy personnel, include very specific roles such as Officers responsible for any required key ceremonies.	
EXTERNAL INTERFACES	

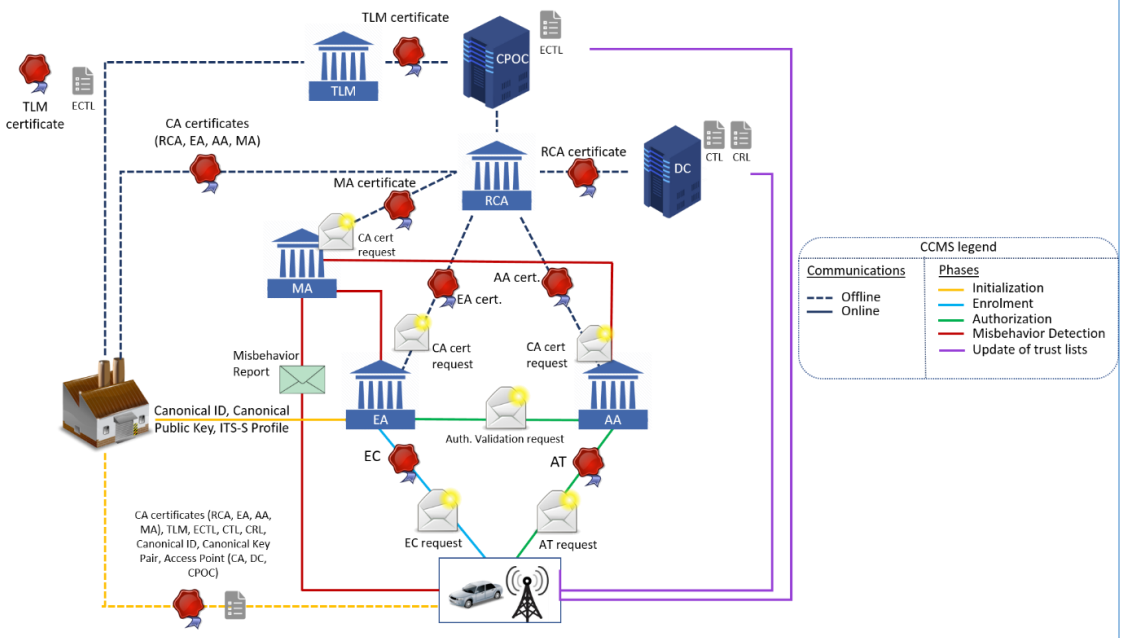


Figure 4 PKI External Interfaces

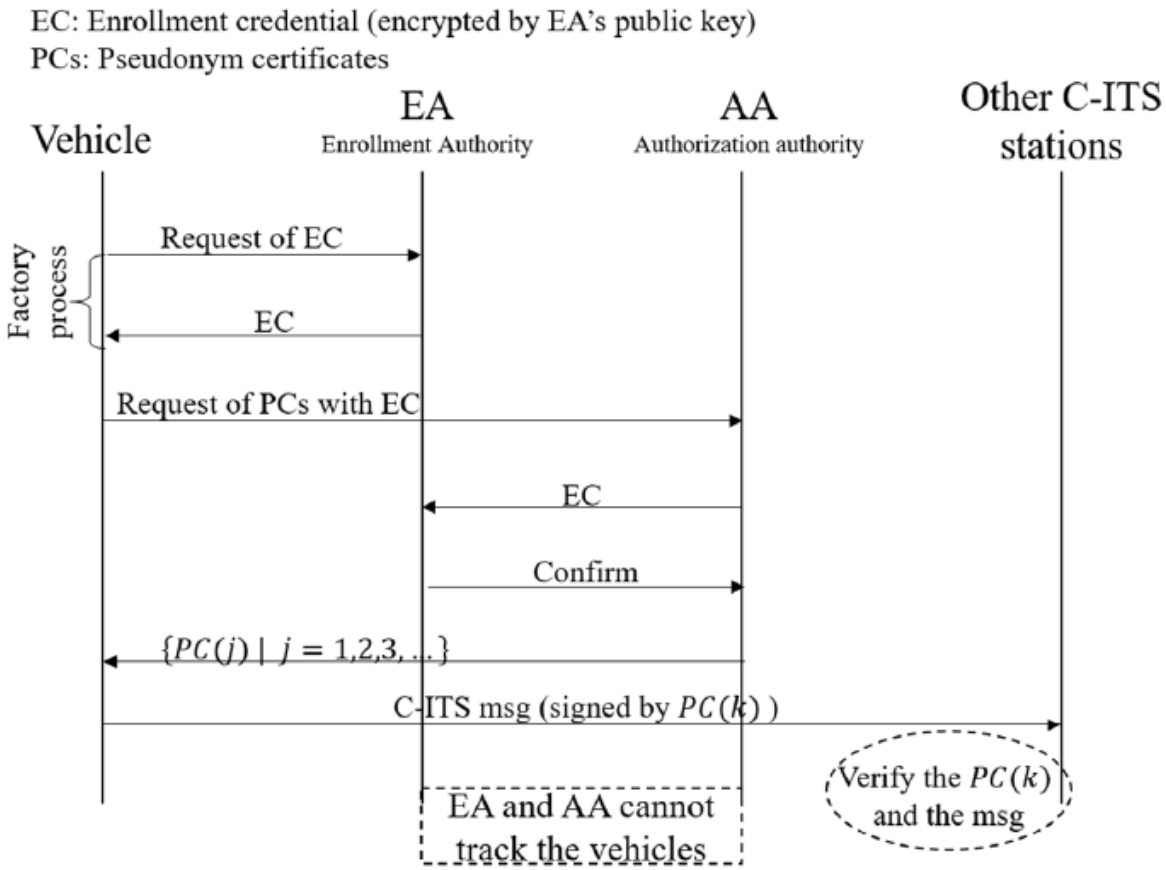


Figure 5 Message exchange process with the PKI

PKI external interfaces are mainly communication channels to vehicles and connected units (such as RSU, connected charging station, toll gates, etc.). This communication channels allow these units to download their certificates (LTC – Long Term Certificate, PC –

Pseudonym Certificate, TSL – Trust-service Status List, CRL - Certificate Revocation List). Messages are sent as HTTP GET or POST requests with ASN.1 DER encoding rule [8].

INTERNAL INTERFACES AND ASSETS

As shown in Fig. 1, internal interfaces are:

- Link between different authorities (ex: RCA-DC or PCA-LTCA)
- Link between PKIs
- Administration access
- PKI administrators
  - Administrator of the PKI software and hardware, configuring and managing PKI elements (HSM, servers, etc.). Including following PKI configuration, e.g.:
    - Set cryptographic algorithms
    - Certificate revocation
    - Addition new CA certificate
    - Downloading new CTL or CRLs
- PKI officers
  - Configures CA's policies, e.g., for ETSI standardized PKIs
    - 'region' of type Geographic Region
    - 'appPermissions' indicate message signing permissions, i.e., permissions to sign certificate response messages contained in a ETSI_TS_103097_Data
    - 'certIssuePermissions': this component shall be used to indicate issuing permissions, i.e., permissions to sign an enrolment credential / authorization ticket with certain permissions

ASSETS:

- CA Certificates
- Enrolment Credential (EC)
- Authorization Ticket (AT)
- TLM certificate

AGENTS AND THREATS

AGENTS:

- PKI administration malicious access
- Attack on external interfaces

THREATS:

- Fake nodes (that compromise privacy or provide false information or data/certificates)
- DoS by sending large amount of requests
- Requests replay
- Requests and private keys disclosure
- Trust list replay
- Misbehaviour reporting tampering
- Private keys disclosure
- Certificates tampering

SECURITY OBJECTIVES

Integrity and confidentiality for all keys used in the process (Canonical Public Key, Data encryption key, CA private keys).

Integrity of certificates. Availability of PKI authorities (RCA, PCA, LTCA, DC).

Table 4 PKI component description and objectives

COMPONENT	VEHICLE C-ITS station / OnBOARD UNIT (VCS/OBU)
DESCRIPTION	
The on-board unit is a Unit of Technologies of Information and Communication (UTIC). It is used to get both information from vehicle (through bus communication interface) and ADS (through digital interface). It is used as a gateway to exchange information with road-side unit or other OBUs.	
EXTERNAL INTERFACES	

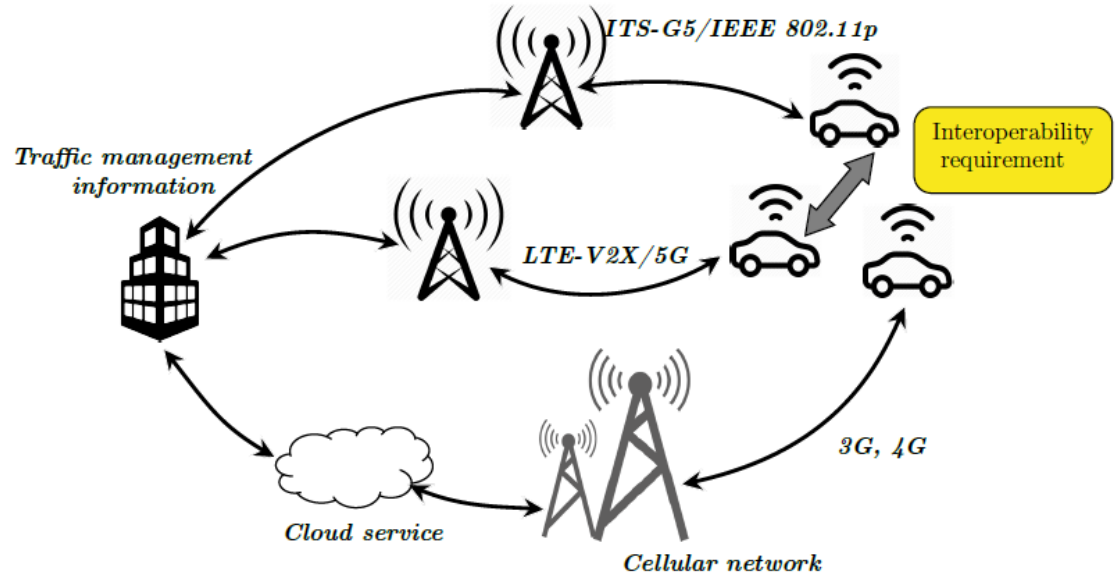


Figure 6 C-ITS Architecture

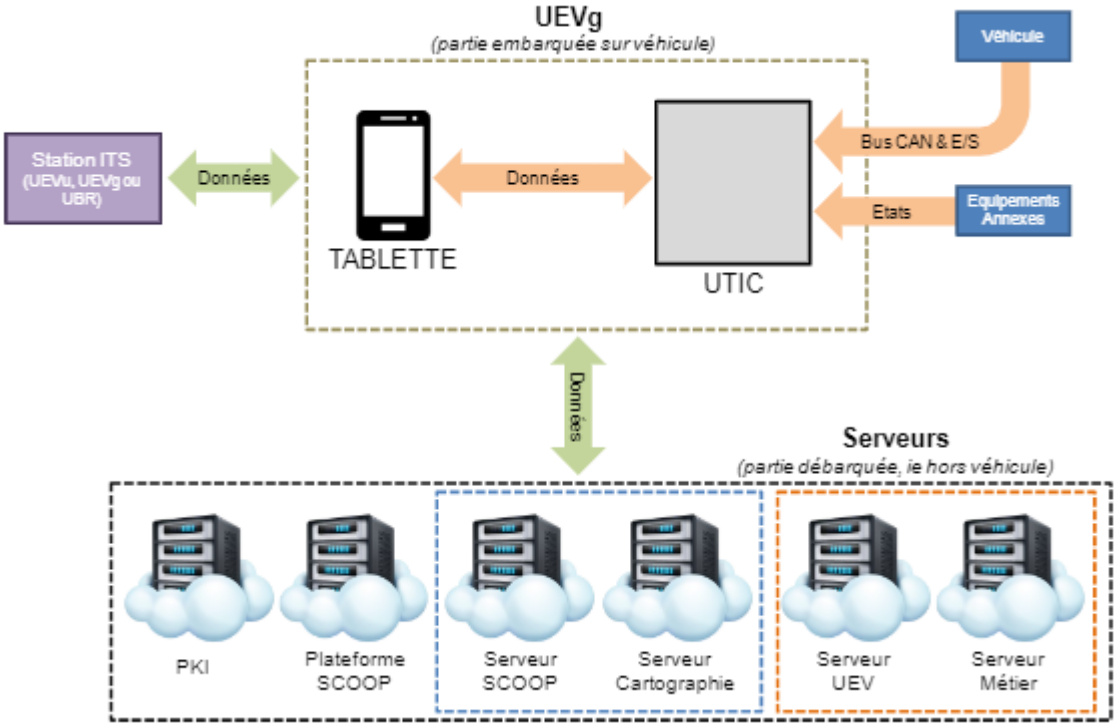


Figure 7 OBU interfaces from SCOOP@F

As shown in Fig.1, OBU external interfaces are mainly for V2I and V2V processes. For this purpose, the OBU is composed of several communication technologies:

- ITS-G5 and C-V2X for direct communication between nodes
- Cellular (like 4G or 5G) for administration purpose and future C-ITS communication using 5G slicing

Some OBUs are also equipped with Wi-Fi interfaces for administration purposes and Bluetooth interfaces for traffic detection.

INTERNAL INTERFACES AND ASSETS

INTERFACES:

As shown in Fig.2, internal interfaces are mainly:

- CAN bus for the exchange of data between OBU and car sensors
- Ethernet for the link between UTIC and remote interface
- RF cable for GNSS and V2X antennas

ASSETS:

The assets of the OBU are the following:

- Canonical Public Key
- Data encryption key
- Certificates (CA certificates, EC, AT, TLM certificates)
- Station registration data
- Policies
- Trust lists
- Misbehaviour detection
- ITS Data
- Configuration and calibration data
- Journey management
- LDM
- V2X communication
- ADS
- Environment perception
- Audit and diagnostic
- Remote control and management

AGENTS AND THREATS

AGENTS:

- User malicious access
- Attack on internal and external interfaces

THREATS:

- Fake nodes (that compromise privacy or provide false information or da-ta/certifi-cates)
- Message replay
- Message congestion
- Traffic capture
- Jamming and DDoS
- Privilege escalation
- Falsification attack
- Illegitimate access to the car components
- Sybil attack

SECURITY OBJECTIVES

- Secure communication channel with external ITS entities protected in integrity and authenticity
- Replay protection
- Plausibility and consistency verification of ITS data
- Preserve privacy of station
- Protection against unauthorized modification of assets
- Secure initialization mechanism protecting confidentiality and integrity of initial station registration data
- PKI request protection in integrity and confidentiality

- Providing audit mechanisms for functional audit and forensic mechanisms
- Provide confidentiality and integrity communication mechanisms for remote administration
- Secure update

Table 5 CVS/OBU component description and objectives

COMPONENT	C-ITS ROADSIDE Station / Roadside UNIT (CRS, RSU, 5G GATEWAY…)
DESCRIPTION	
The road-side unit is a unit of technologies of information and communication (UTIC). It is used to get both information from infrastructure (through road operator interface, traffic light controllers, etc.) and vehicles (through V2X communication channels). The unit is then used as a gateway to exchange information with OBUs, other RSUs or PKI system.	
EXTERNAL INTERFACES	

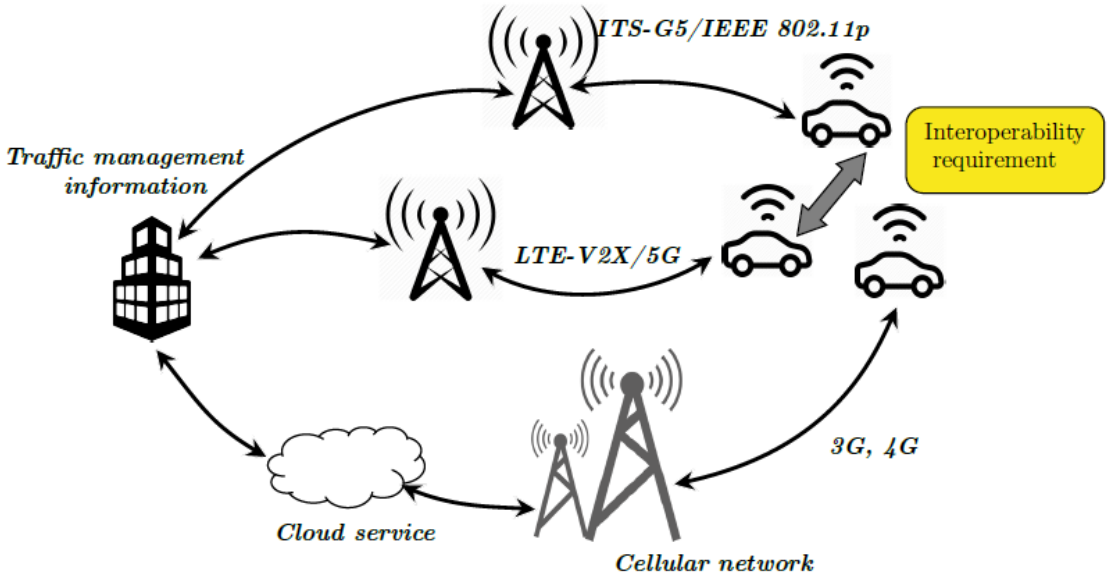


Figure 8 C-ITS Architecture

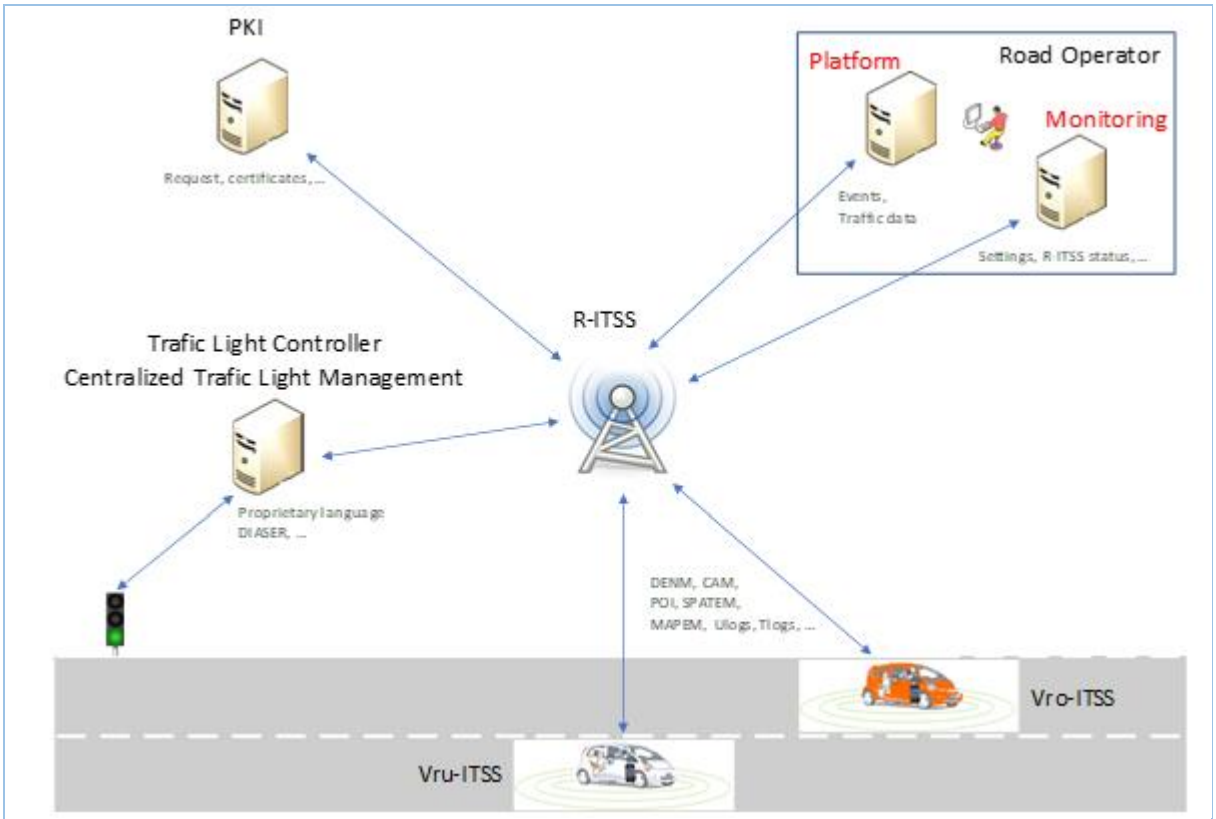


Figure 9 RSU interfaces from SCOOP@F

As shown in Figure 8 and Figure 9, CRS/RSU external interfaces are:

- PKI (through internet access)
- Traffic light controllers or CTLM (through internet access)
- Road operator systems (through internet access)
- Vehicles (through V2X communication channels)

Some RSUs are also equipped with Wi-Fi interfaces for administration purposes and Bluetooth interfaces for traffic detection.

INTERNAL INTERFACES AND ASSETS

Internal interfaces are not standardized and can vary a lot, so it is not relevant to try to list them.

ASSETS:

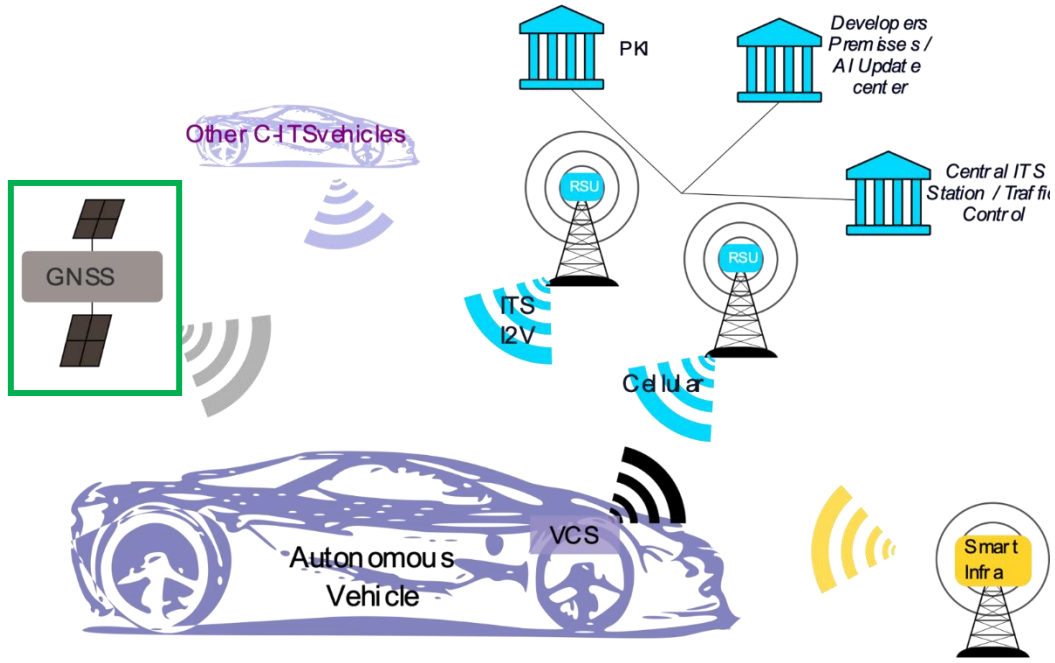
The assets of the RSU are the following:

- Canonical Public Key
- Data encryption key
- Certificates (CA certificates, EC, AT, TLM certificates)
- Station registration data
- Policies
- Trust lists
- Misbehaviour detection
- ITS Data
- Configuration and calibration data
- Journey management
- LDM
- V2X communication
- ADS

• Environment perception • Audit and diagnostic • Remote control and management
AGENTS AND THREATS
AGENTS: • Attack on internal and external interfaces **THREATS:** • Fake nodes (that compromise privacy or provide false information or da-ta/certifi-cates) • Message replay • Message congestion • Traffic capture • Jamming and DDoS • Privilege escalation • Falsification attack • Illegitimate access to the car components • Sybil attack
SECURITY objectives
• Only authorized users can access the • When possible, critical safety signals should be transported in a manner inaccessible through external vehicle interfaces • All networks and systems external to a vehicle's wireless interfaces shall be pro-tected in terms of confidentiality, integrity and authenticity (using appropriate tech-niques to mitigate potential threats). • Network segmentation • Gateways with strong boundary controls

Table 6 CRS/OBU description and objectives

COMPONENT	GNSS
DESCRIPTION	
GNSS (Global Navigation Satellite Systems) is a general term describing any satellite con-stellation that provides positioning, navigation, and timing (PNT) services on a global or regional basis such as GPS, Galileo, Beidou, Glonas, ... A GNSS system is composed of satellites and receivers located into the vehicles. Inside ve-hicles, a navigation system (with maps) is also included.	
EXTERNAL INTERFACES	



External interface is the GNSS receiver onboard received signals from the various satellites.

INTERNAL INTERFACES AND ASSETS
ASSETS: • Time and position
AGENTS AND THREATS
AGENTS: • Attack external interfaces **THREATS:** • GNSS spoofing • Radio jamming
SECURITY OBJECTIVES
Ensure service availability and integrity.

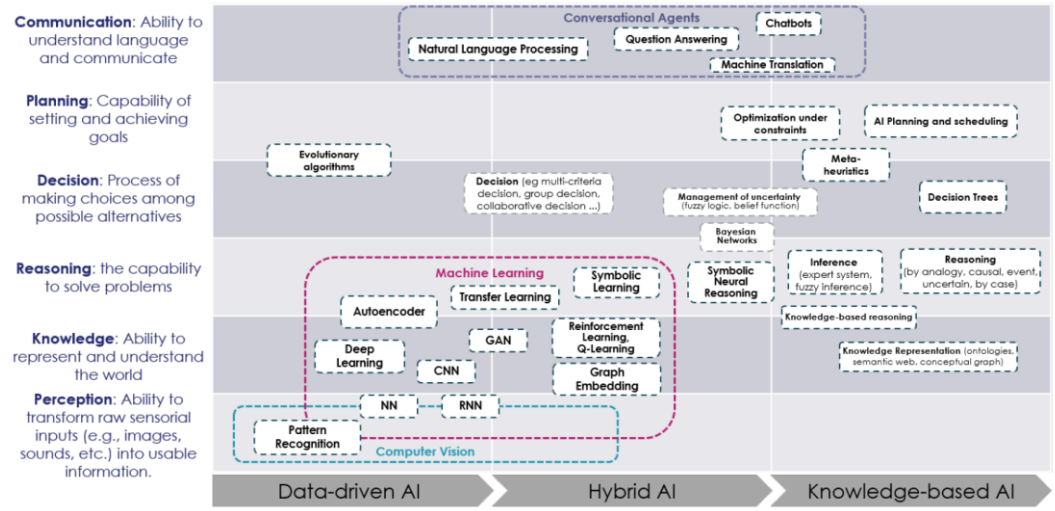
Table 7 GNSS

2.2.2 AI sub systems

2.2.2.1 Sensors

COMPONENT	Sensors
DESCRIPTION	
Exteroceptive Sensors (eg radar, camera, lidar, ultrasonic, etc.) & Data returned by these sensors. Basically, exteroceptive sensors aim at : • detecting all surrounding objects, road, miscellaneous scene elements and environmental conditions that may influence the behavior of the automated vehicle, e.g. pedestrians, vehicles (passenger car, trucks, etc), road components (eg lane marking, traffic signs, etc), rain conditions, etc. • characterizing such scene elements & environmental conditions (e.g. type of object, dimensions, speed, relative distance, etc.). • Such sensors (and their further processing, by e.g. environment perception & data fusion algorithms) may rely on the 3 different types of IA (according to taxonomy of IA defined by Confiance AI project : data-driven, hybrid or knowledge-based).	

[L5.3] Cybersecurity Objectives for Autonomous Vehicle Systems



(source : L1.1.1 of "Confiance AI" project, Oct 2021)

NOTE : Interoceptive sensors, describing Autonomous Vehicle state, are not addressed here (e.g. GNSS, IMU, etc.).

EXTERNAL INTERFACES

The sensor physical interface that analysis the sensor environment.

INTERNAL INTERFACES AND ASSETS

- Environment description

AGENTS AND THREATS

AGENTS:

- Rode side attacker: An attacker trying to modify AV surrounding to force AV wrong or potentially dangerous decisions by impacting/modifying AV sensor observations (light perturbation, objects modification or introduction e.g. painting signs, using sensor blinders, etc.).

THREATS:

- Camera spoofing ("phantom/illusion attack", "adversarial attack")
- Camera denial of device (Camera blinding attack)
- Lidar spoofing (Lidar Replay/Relay attack, "Lidar adversarial attack")
- Lidar denial of service/jamming (sensor saturation/blinding attack)
- Ultrasonic sensors Spoofing/Denial of service (Jamming/Acoustic Quieting)
- Radar Spoofing (Relay)/ Denial of service (Jamming).

SECURITY OBJECTIVES

SEC_OBJ_EXT_SENSORS_1: Exteroceptive sensors used by AV shall be protected against attacks targeting the integrity & availability of the data they return (cf Threats above)

NOTE: This security objective is in line with the security measures recommended by ENISA ("Good practices for security of smart cars", Nov. 2019)

> **TM-43**: Protect critical autonomous sensors to prevent the different attacks aiming to alter smart cars environment perception.
>
> **TM-44**: Harden against Adversarial attacks, to prevent AI and ML components from being tricked.

SEC_OBJ_EXT_SENSORS_2: AV shall be protected against the replacement of its exteroceptive sensors by unauthorized electronic hardware.

This objective addresses "physical manipulation of systems" attack identified by UN R155 (see below extract)

32	Physical manipulation of systems can enable an attack	32.1	**Manipulation of electronic hardware**, e.g. unauthorized electronic hardware added to a vehicle to enable "man-in-the-middle" attack **Replacement of authorized electronic hardware** (e.g., sensors) with unauthorized electronic hardware **Manipulation of the information** collected by a sensor (for example, using a magnet to tamper with the Hall effect sensor connected to the gearbox)

2.2.2.2 *Other AI Components*

COMPONENT	AI components
DESCRIPTION	

Artificial Intelligence is used nowadays in a wide variety of applications and services of modern vehicles to enhance performance, comfort and safety and the overall driving experience. Among the most important aspects we can mention, Advanced Driver Assistance Systems, Autonomous Driving Systems, Traffic prediction, Driving personalisation, Voice command, Predictive Cars Maintenance.

Unlike traditional software development, the development of these applications and services complies with the development cycle of machine learning models. This cycle involves several phases as illustrated below.

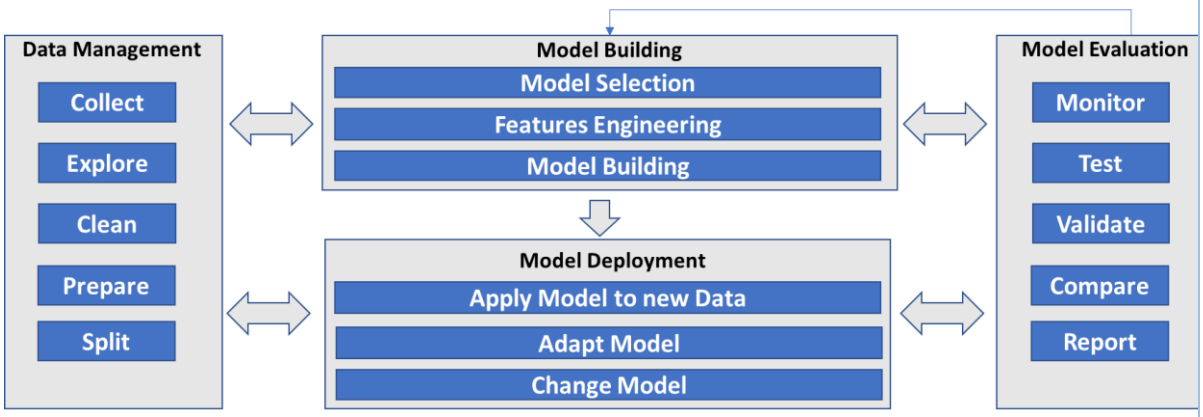


Figure 10 Development cycle of machine learning models

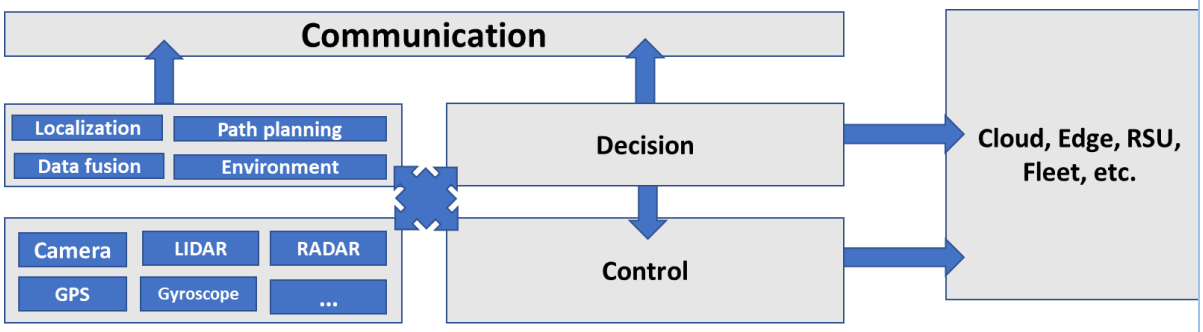
Once the model build, it's deployed and here again, in the current and future vehicle architecture, the deployment of the Machine Learning Model can vary depending on the application or the services it's used for. The table below presents an overview of where the ML models are/will most likely be deployed for each one of the applications listed above.

	Sensing	Perception	Decision	Control	Collaboration
Advanced Driver Assistance Systems	•	•	•		

Autonomous Driving Systems	•	•	•		
Driving Personalisation			•	•	
Voice Command				•	
Traffic prediction			•	•	•
Predictive Maintenance				•	•

EXTERNAL INTERFACES

One can observe that for an autonomy level of 3 or 4. AI modules need to be deployed at least in the sensing, perception, decision, and control modules. Collaboration refers to intelligence built based on models deployed in the edge, the cloud or external actors (road side units, other vehicles, etc).



INTERNAL INTERFACES AND ASSETS

In our context, no specific internal interfaces are identified. AI are usually one functional block taking one information as input and providing the associated output as a result.

AGENTS AND THREATS

AGENTS:

- Rogue user in developers' facilities
- Internet attacker
- Rogue ITS-S administrator
- Roadside attacker

THREATS:

For each of those components, threats have been identified in [4].

1. **Poisoning attacks** can be achieved by rogue user or internet attackers before the training phase by introducing perturbations among the training data to generate a corrupted model.
2. **Evasion attacks** happen after the model is trained. They are used to manipulate the input data of a model to trigger erroneous predictions by roadside attacker to force AI to take wrong decisions.
3. **Extraction attacks** try to steal the parameters of a remote model in order to reproduce its behaviour or rob confidential information, that could be performed by internet attackers or rogue ITS-S administrators.
4. **Inference (Inversion) attacks** abuse a model to extort sensitive information learned from the training data that could be performed by internet attackers or rogue ITS-S administrators.

All those must be considered in the context of PRISSMA, however extraction and inference should have lesser impact on the system security and as for poisoning should be covered by security countermeasures in developers premisses and deployment phases.
SECURITY OBJECTIVES
• Confidentiality and integrity protection of all development life cycle (cf. Figure 10). • Resilience to evasion attacks.

3 Security requirements

In this section we present the different security requirements we define to fulfil the security objectives identified in the section 2. In section 2 we have categorized security objectives for 2 types of components:

- Objectives for large system components as well as the global system itself
- Objectives for specific and critical C-ITS components

For the first category, the size, complexity and variety of possible technologies and architecture do not allow to predefine precise objectives or requirements applicable for any such element. Those components can theoretically face any possible cyber threats: web, system, network attacks, social engineering attacks, etc. Thus, it would be vain to try to enumerate in this deliverable all possible attacks or all possible requirements for counter measures deployment. More generic approaches are required. The state-of-the-art already cover those cases, it provides mature and well recognized framework of general security audit and requirements. We will thus for the sake of pragmatism re-use well established audit method as well as automotive industry initiative for cyber-security as presented in section 3.1.
However, there is one specific point we address here for one component identified in section 2.1, AI development sites for which specific requirement concerning the different AI life-cycle stages (training data sets, generated models and field data used as input to study potential update of training data sets).
For more specific C-ITS equipment we define more precise requirements, thanks to the reference specifications, threats and their associated security objectives provide in section 2.2.2. We present those requirements for each main recurring architecture components identified previously, that is: C-ITS stations, AI components and sensor components.

3.1 Systems requirements

As previously stated, the generic ARTS architecture targeted by the PRISSMA project (cf Table 3) presents recurring “large” components (complete and independent IT systems):

- central C-ITS stations and developers’ premises.

For those components, potentially facing all possible IT threats, providing a predefined set of objectives or requirements is not relevant since there are too many possibilities, and the cyber-security state of the art is evolving too fast. At least not for the moment, regarding current ARTS level of standardization of those components.
In our proposition, specific ARTS risk management are defined (for AI development sites) that will complement ISO 27000 series requirements as presented in section 3.1.1, but the core requirements we define for those components will in fact be those of ISO 27001 and 27002. We recommend the application and reuse of these already well established and recognized approach.
The ISO 27000 series contains the following document of interest to us:

- ISO/IEC 27000 [9]
- ISO/IEC 27001 [10]
- ISO/IEC 27002 [11]

The first document serves as a dictionary and present in general the ISO 27000 series. This document is of interest for anyone implied in any cyber security activity governed by ISO 27000 series requirements. Which is our case since we require the owner of the targeted sites and ARTS owner to be audited following ISO 27001 and 27002 requirements.
The ISO 27001 (cf. section 3.1.1.1) defines the requirements to deploy an Information Security Management System (ISMS), when the ISO 27002 (cf. section 3.1.1.2) defines best practices or more detailed technical requirement to be deployed by the ISMS.

The ISO 27001 requirement can be certified, several companies in France can perform those certifications.
We add to this list of input standards an automotive industry-oriented standard, ISO/SAE 21434 [12]. This standard specifies requirements for cybersecurity risk management for road vehicles, their components and interfaces, throughout engineering (e.g. concept, design, development), production, operation, maintenance, and decommissioning.
In this deliverable we don't present the full content of those standards. We only provide an overview of their content. However, for the PRISSMA assurance framework we assume that those standards will be accessible to the different actors involved the evaluation, and full conformance can be required.
Also, for the complete ARTS itself, due again to its complexity and diversity of potential implementations we require to validate its final security (the composition of the security of its sub-components), in the same way, i. e. an ISO 27001 audit. However, since it is the composition of all the sub-components mentioned before its security will depend on (i) the security of those sub-component and (ii) the composition of their local security properties. So, the ARTS audit will have to verify the audit and certification obtained by its sub-component and to verify and then validate that their composition is secured.

3.1.1 ISO 27000 series and ISO 21 434

3.1.1.1 ISO/IEC 27001:2013 Information security, cybersecurity and privacy protection — Information security controls.

The goal of the ISO/IEC 27001 is to help organization to define an ISMS in order to protect their informational assets. The standard provides organizational and cybersecurity requirements that help to protect the organization IT systems both functions and data.
The standard defines several steps to be followed to define, manage, evaluate, and update an ISMS:

- Identify and establish leadership needs to manage the ISMS (define policy, objectives and strategy, identify and support necessary resources, etc.).
- Plan actions starting with a risk analysis to identify threats, evaluate the associated risks and define the objectives of the ISMS to mitigate them.
- Define and provide the required support to fulfil the objectives (technical and human resources, required competences, user awareness management, communication needs linked to security and documentation to support ISMS).
- Operational management of the ISMS (define and provide means to manage and control the ISMS deployment and risk analysis update).
- Performance evaluation of the ISMS (definition and application of performance evaluation means, internal audits and management review).
- ISMS improvement based on evaluation and risk updates.

The requirements presented by the ISO/IEC 27001 are very generic. More technical requirements are provided in annex of the standard, which are a summary of the ISO/IEC 27002.

3.1.1.2 ISO/IEC 27002:2020 Information technology — Security techniques — Code of practice for information security controls

ISO/IEC provides more specific best practices for ISMS deployment. The document present 14 requirements categories such as: Information Security Policies, Organization of Information Security, Human Resource Security, Asset Management, Access Control,

Cryptography, Physical and environmental security, Operational Security (actual IT security solutions), Incident management, etc.
The document provides over a hundred of requirements classified in those 14 categories. Those requirements, if more precise than ISO/IEC 27001 are still very generic to be applicable in any IT environment, which implies an almost infinite possibilities of technologies, size, complexity of systems. provides some examples of requirements extracted from the standard. And the complete list can of course be seen in the document itself.
We also provide some representative ones here:

- Organizational controls
  - Information security roles and responsibilities should be defined and allocated according to the organization needs to establish a defined, approved and understood structure for the implementation, operation and management of information security within the organization.
- Physical and Environmental security
  - Security perimeters should be defined and used to protect areas that contain information and other associated assets to prevent unauthorized physical access, damage and interference to the organization's information and other associated assets.
  - Secure areas should be protected by appropriate entry controls and access points to ensure only authorized physical access to the organization's information and other associated assets occurs.
- Technological controls
  - Information stored on, processed by or accessible via user endpoint devices should be protected to protect information against the risks introduced by using user endpoint devices.
  - The allocation and use of privileged access rights should be restricted and managed to ensure only authorized users, software components and services are provided with privileged.
  - Information about technical vulnerabilities of information systems in use should be obtained, the organization's exposure to such vulnerabilities should be evaluated and appropriate measures should be taken to prevent exploitation of technical vulnerabilities.
  - Backup copies of information, software and systems should be maintained and regularly tested in accordance with the agreed topic-specific policy on backup to enable recovery from loss of data or systems.

Each of those requirements are completed with guidance and other information to further detail requirements potential implementation but providing an equivalent level of precision to again be applicable in most conditions.

3.1.1.3 ISO/SAE 21434:2021 Road vehicles — Cybersecurity engineering

The ISO/SAE 21 434 standard is an automotive industry oriented standard presenting requirement for cybersecurity engineering of road vehicle and their equipment. The standard covers the whole life-cycles of the developments of vehicles and their equipment.
Only parts of this norm apply to our requirements scope and the C-ITS sites we have identified in our reference ARTS architecture. For instance, the AI developers' sites, since it includes vehicle component developments, fail under the scope of the norm and to fulfil some of the requirements it defines. The rest of the ARTS architecture is out of scope of the ISO/SAE 21434. The complete architecture provides a service when the ISO/SAE 21 434

targets vehicles and their components developments life-cycle not including requirements on their operational environment.
ISO/SAE 21 434 covers all the different stages of the development life-cycle:

- Overall cybersecurity management (included in ISO 27001 scope)
- Project dependent cybersecurity management
- Risk assessment methods (included in ISO 27001 scope)
- Concept phase
- Product developments
- Production
- Operations (included in ISO 27001 scope)
- Maintenance (included in ISO 27001 scope)
- Decommissioning
- Supporting processes (included in ISO 27001 scope)

As identified, several of those requirement's topics overlap or correspond to ISO/IEC 27001 and 27002 requirements. ISO/SAE 21 434 even explicitly requires defining and maintain an ISMS which is the exact ISO 27001 scope. Also, others are out of scope since they are only relevant to product development while here we focus on a system composing already available products, making those two approaches complementary.
The audit elements to be included in our requirement definition scope are the following:

- For central ITS stations
    - Overall cybersecurity management
    - Project dependent cybersecurity management
    - Risk assessment methods
- For the developers' premises
    - All parts
- For the complete ARTS (all components identified as included in ISO 27001 scope above, the others being excluded)
    - Overall cybersecurity management
    - Project dependent cybersecurity management
    - Risk assessment methods
    - Operations
    - Maintenance
    - Supporting processes

By default, the identified elements are to be included in the site audits, however for each specific site it must be evaluated if they all are relevant to the site activities scope.

3.1.2 ARTS requirements

ARTS (Automated Road Transport System) are regulated by the decree n° 2021-873 of the 29th of June 2021. An application guide has been produced by the STRMTG to define cybersecurity requirements of the system under consideration[3] which shall be applied, if applicable, by the responsible entities defined in the decree ([5], cf. section 4.1.3).
When applicable, depending on the system under consideration, those requirements have to be taken into account and added to PRISSMA requirements.

[3] System under consideration can have different shapes: the ARTS, the technical system of the ARTS, a component, the SMS of the operator…

A specific study must be conducted by the ARTS responsible entity to evaluate if the PRISSMA requirements are applicable or not for the system under consideration. This evaluation shall be audited by an independent qualified organism (AQO) approved by the STRMTG.

3.1.3 AI developments site requirements

As identified in section 2.2.2, specific threats apply to AI components. Particularly, attacks on AI can target all the AI model generation life cycle (cf. data management or model building phases illustrated in Figure 10).
Therefore, the physical and IT security of all AI developments site must be assessed to demonstrate that only authorized personnel, can physically and logically access or modify and so that:

- Training data set collection and storage mechanisms must demonstrate confidentiality and integrity protection
- Model building mechanisms must demonstrate confidentiality and integrity protection
- Model deployment must demonstrate confidentiality and integrity protection

In fact, during the audit we require that specific elements of proof are provided to demonstrate that countermeasures have been implemented to specifically counter poisoning or extraction attacks.
Again, specific requirements are not provided by us since many different security measures can be implemented to fulfil these requirements. But, these specific threat have to be explicitly managed (e.g. integrity protection mechanisms, data sets reviews, regression tests, etc.).

3.1.4 PRISSMA requirements

To summarize, PRISSMA assurance scheme combines the requirement of conformity to the ISO 27001 and 27002 together with the French ARTS requirements [5] and AI requirements as identified in this section 3.1.3.
Whenever applicable some parts of the ISO 21434 requirements are also to be fulfilled. This must be defined by the auditor together with the audited entity at the audit start.
This approach is complementary and does not replace any type approval requirements of vehicles or components, as well as ARTS regulation requirements.

3.2 Technical components requirements

3.2.1 C-ITS station (OBU/RSU/VRU/Central station)

The following table presents security requirements for all C-ITS stations implied in the deployment of autonomous driving services. Not all are made mandatory,

Name	Description
Secure association	The C-ITS station ensure communication security with other C-ITS station by sending and verifying C-ITS certificates conformant to ETSI 103 097. The verification of the certificate format and validity is performed following IEEE 1609.2 section 5.1 This implies that the exchanged certificates shall not be included in the system PKI CRL and the certificate signatures shall be verified against the PKI chain of trust.
Message protection	All C-ITS application data used by ADS systems must be signed for message proof of origin and integrity validation. PKI requests must be encrypted and follow protocol and format as defined by ETSI 102 940 [6] and 102 941 [7].

Replay protection	Mechanisms shall be implemented to forbid attack based on message replay, e.g.: • verification if the exact same message has already been received • timestamp validation • Message signature validation • etc.
Privacy	No long-term data shall be sent that can allow AT linkage. This includes any information that can be used to identify the Personal Identifiable Information (PII) principal to whom such information relates or is or might be directly or indirectly linked to a PII principal. This includes any IDs different from the certificates. IDs of vehicle components (different form the Canonical ID) to which no AT should be linkable to. This includes at least EC, EC CertificateID, Vehicle ID (e.g. VIN). But, depending on the TOE applications and interfaces implementation, it could also cover such IDs as Licence plate number, Software Identifier/licence number, user IDs, etc. It also includes IDs used by the communication stack like IP or MAC addresses, Mobile Station ISDN Number (MSISDN). Including at least all G5 IDs. Public IDs that can be linked to at most one AT. Also all PII shall be erased as soon as they are no longer used (ITS data time storage shall be limited, all private keys and certificates must be erased as soon as their associated are no longer valid or no longer used).

Plausibility and consistency checks	Every received data from other ITS-S must be verified for plausibility and consistency to limit attacks or error impact. At least the following elements should be considered depending on station type ([13], [14]), in fact an OBU speed of 100 km/h, might not be interpreted the same as a vehicle going at that speed:
	• Position
	  ○ Incompatible with speed or heading
	  ○ Position not on the road
	  ○ Position overlaps with other vehicles
	  ○ Too far from the receiver
	• Heading for mobile ITS-S
	  ○ Heading direction not compatible with speed (U-turn at 100km/h)
	  ○ With road heading
	• Speed
	  ○ Identify threshold for speed acceptance
	  ○ Speed data incompatible with acceleration
	• Vehicle length and width
	  ○ Identify thresholds for vehicle length and width
	• Curvature for mobile ITS-S
	  ○ Incompatible with speed or heading changes
	• Received signal strength
	  ○ Incompatible with sender position

Access control	ITS station shall maintain the following roles:

	• Station administrator responsible for the main management and configuration functions of the ITS station including:
	  ○ Management of the Position and Timing parameters
	  ○ Management of the SW update procedure
	  ○ Configuration of the access to the vehicle interface
	  ○ Configuration of the diagnostic level and of the collection of the diagnostic results if no auditor is defined
	• V2X administrator responsible for the main management and configuration of the ITS communications functions of the station including:
	  ○ Configuration of the filter of the V2X objects
	  ○ Configuration of the access layer parameters of the different V2X communication parameters
	• Auditor (optional) responsible for accessing and managing the station audit traces (configure, read, modify audit traces)
	All users must be identified and authenticated before any actions.
	If login/password mechanisms are used no login feedback shall leak login information.
	Each login failures shall induce time increase before next possible attempt to avoid brute force.
	Automatic session locking should be enforced after a specified time interval of user inactivity by clearing or overwriting display devices, making the current contents unreadable and disabling any activity of the user's data access/display devices other than unlocking the session.

Initialization	The ITS station must generate or provide secure import of canonical key pair to generate the first EC request all conformant to ETSI 102 942. The station must download up to date trust elements before starting to send and receiving ITS messages.
Trust elements update	The station must download up to date CRL, CTL, CA certs whenever available. To do that it must perform regular verification based on administrator define *threshold time* or whenever expired.
Enrolment	The ITS station must generate enrolment request conformant to ETSI 102 941 when valid certificates are missing, or when time left before expiration reaches a define threshold. Every signature of request shall be performed with former enrolment certificates. Old key and certificate (invalid or not used anymore) shall be erased whenever possible.
Authorization	The ITS station must generate authorization request conformant to ETSI 102 941 when certificates are missing, or when time left before expiration reaches a define threshold. Every signature of request shall be performed with currently valid enrolment certificates. Old key and certificate (invalid or not used anymore) shall be erased whenever possible.

Check operation	The station shall run a suite of self-tests during initial start-up to demonstrate the correct operation of the TSF. This should include at least software integrity verification at start up.
Audit	Log generation of security related events shall be provided, this should cover events like signature verification failure, certificate verification failure, user authentication attempts, configuration modifications, software updates, etc.
Administration flow protection (optional)	When remote administration is allowed by the station (which is not mandatory), encryption and integrity mechanisms shall be provided. Those mechanism shall follow recommendations provided by [15] and [16].
Software update (optional)	When software update mechanisms are provided, which might not be the case for every station (potentially not for certified products), signature verification shall be performed for every software update. Also, version verification shall be performed to provide anti-rollback mechanisms.

3.2.2 AI requirements

One of the main challenges when it comes to ensuring security for AI is that AIs often don't have explicit specifications, while assurance is all about validating requirements (i.e., specifications). AI properties are indirectly defined by their training datasets.
For example, an AI designed to classify an object as a person in an image or video stream is usually not specified with any formal constraints such as:

- heigh in [0.8m,2.2m],
- width in [0.3 m,1m],
- number of limbs [0-4],
- posture in [standing|lying|bending|walking|running|jumping|…],
- etc.

The AI will be able to classify objects in a picture as being a human because it has been trained with pictures including humans labeled as such. If a human can easily identify an issue (or potential attack) when an AI classifies the picture of a cat as a human, some other examples might not be so trivial to classify as mistakes or results of an attack. For instance, a real person in an image might not be recognized due to the image being too blurred, the person being half-hidden by an object, or over/under light exposure, etc. Therefore, what would be a vulnerability in that context where even "normal" behavior is not specified nor standardized?
To the best of our knowledge, the state of the art only provides high-level requirements and recommendations on how an AI should be validated. This currently aligns with our high-level assurance methodology.
As identified in WP2 [17], several properties associated with AI have been defined by the state of the art. In our assurance framework, we require that developers provide evidence for the following properties of their AI:

- **Accountability:** the capability of the system to be answerable for its decisions, actions and performance.
- **Explainability:** the property of an AI system to express important factors influencing the AI system results or to provide details/reasons behind its functioning so that humans can understand.
- **Interpretability:** refers to the degree to which a human can understand the cause of a decision.

- **Reliability:** refers to the property of consistent intended behaviour and results.
- **Understandability** (equivalent to intelligibility): denotes the characteristic of a model to make a human understand its function –how the model works –without any need for explaining its internal structure or the algorithmic means by which the model processes data internally.

Additionally, since an AI's capability to correctly perform its task and limit its errors (and thus likely resist basic attacks) is directly dependent on its training dataset, the developer should also provide proof that their training datasets fulfil the following properties:

- **Fairness**
  - **Bias Mitigation**: The dataset should minimize bias to prevent the model from making unfair or discriminatory decisions.
  - **Equality of Representation**: Different groups should be represented proportionally to avoid skewed outcomes.
- **Representativeness**
  - **Diverse Samples**: The dataset should include a wide range of examples that reflect the variability of the real-world scenarios the model will encounter.
  - **Population Coverage**: It should cover all relevant subpopulations adequately to ensure generalization across different segments.
- **Accuracy**
  - **Correct Labels**: Data should be accurately labelled, ensuring that each example is correctly annotated.
  - **Data Quality**: High-quality, error-free data is essential for training reliable models.
- **Completeness**
  - **Comprehensive Data:** The dataset should cover all necessary aspects of the domain to avoid gaps that could lead to incomplete learning.
  - **Balanced Classes**: For classification tasks, classes should be balanced to prevent models from being biased towards the majority class.
- **Relevance**
  - **Domain Appropriateness:** The data should be pertinent to the problem domain and context in which the model will be used.
  - **Up-to-date Information:** Data should be current to ensure the model's applicability to recent and future scenarios.

Thus, for the evaluation of AI-based components, developers will have to provide explicit evidence to demonstrate the fulfilment of all the aforementioned properties. The validation of this evidence will mostly rely on the evaluator's interpretation. However, our main goal in stating this requirement is for the evaluator to verify that developers did not make identifiable mistakes based on the state-of-the-art knowledge, rather than to try to validate exhaustive proof.

Also, in the PRISSMA project, work has been done to enhance Operational Domain Design (ODD) specifications using a scenario-based approach. These scenarios indirectly define the expected behaviour of the ARTS AI in the different situations it can encounter. Thus, we require that developers, as part of this evaluation, provide a specification of the ODD their AI will be integrated into, to be used as an indirect specification of their expected behaviour. All aforementioned properties will then be evaluated with respect to these scenario definitions.

4 Assurance requirements

4.1 Security evaluation State of the art

Over the last three decades many researchers and practitioners have addressed the general problem of IT products validation, to try to find more specific, efficient, or formalized approaches. So far, not fully satisfying (i.e. universal recognition and no known drawbacks) solution has been found and it will probably never be. Several comprehensive overviews of the various efforts made on the evaluation and measurement of IT security domain can be found in [17], [18], [19], [20], [21]. Those surveys present hundreds of different attempts and approaches to enhance current assurance practices.

Before discussing any evaluation schemes and methodologies pros and cons, we first start by presenting important aspects on which existing security evaluations schemes might differ.

In fact, all existing IT security evaluation methods have to address the following three topics either doing it directly or not:

- Defining what must be evaluated
    - Which product and which version of the product?
    - Which function of the product?
    - In which environment and for which type of threat?
- Defining which evaluation activities to use
    - Evaluate the development
    - Evaluate the product architecture
    - Test the external/internal interfaces
    - Analyse the code, the guides, etc.
- Defining who is in charge or competent to define, manage or perform evaluation activities
    - Who will pay and be the sponsor of the evaluation?
    - Who has the expertise and required test environment?
    - What does the developer have and what information must he provide for the evaluation of its product?
    - What is the end user's point of view?

The above three dimensions correspond to what is generally called:

- Definition of the Security Target (ST).
- Definition of the assurance components.
- Definition of the evaluation scheme.

All IT security evaluation schemes have their own definition of the important elements to address and how for each of these three dimensions. Those are in fact the three pillars of any assurance methodologies, where security assurance is the process of demonstrating that a system meets its security requirements.

One has to understand that there is no universal solution for the problem of IT security evaluation since all known solutions have pros and cons. Those pros and cons will always reflect the necessary trade-off between the obtained confidence level and its evaluation cost.

Security evaluation is a tedious problem and would probably remain so because IT systems are complex, evolve rapidly and security evaluation have to study as exhaustively as possible their security properties since security issues lies in the implementation details.

Whether it will be feasible or not to obtain one day a fully automated process based on formal proof for any security requirement validation, the current state of the art is still making efforts to enhance existing approaches. It tries to lower evaluation costs while providing higher assurance levels for ever more complex systems.

AI based systems are part of those complex systems for which current evaluation methods can be argued not to be optimal and for which expert still can't demonstrate the exact confidence provided by current evaluation processes.
The general problem of IT products validation has been addressed by many researchers and practitioners, in search of finding more specific and formalized approaches. So far, no fully satisfying solution has been found, i.e. a solution reaching universal recognition with no known drawbacks.
The main identified challenges and gaps identified by the state of the art for current security evaluation methodologies are [20]:

1) Elucidation, Modelling, and Validation of Security Requirements.
2) Security Assurance in Component-based Development.
3) Operational Security Assurance.
4) Security Assurance in Service Selection (assurance for systems providing configurable service-oriented architectures).
5) Security Assurance Aggregation (combination of assurance of different system components).
6) Security assurance Tool.
7) CC Protection Profile for Trusted Computing Features.
8) Automation of Security Assurance.
9) Identification and Prediction of Security vulnerability

In the PRISSMA project we do not aim at tackling all those aspects. We focus on the core challenge of assessing AI security properties focusing mainly on points 1, 5, 6 and 7 (partially). AI security evaluation is a recent problem. Basic assurance components still need to be addressed. Confidence in that field is still low and PRISSMA will work on gathering and developing state of the art best practices to provide good confidence in ARTS security.
In this deliverable section, we present the most relevant IT evaluation processes that could be used as inputs for PRISSMA assurance scheme proposition that we detail in section 5.

4.1.1 ISO 15408 - The Common Criteria for Information Technology Security Evaluation (CC)

The Assurance framework approach is the most complete and exhaustive approach in the cyber security evaluation domain. It provides the highest assurance levels (i.e., level of confidence in the product security), at the cost of being more expensive and time consuming than the other approaches. It also requires the involvement of (rare and) expensive accredited evaluators expertise. So, this type of approach is adapted to product used in critical context where cybersecurity is paramount. It's only adapted to mature and stable solutions since the evaluation is long and its valid limiter to one version. It wouldn't make sense to certify product with version life-time shorter than the certification process (several month).
There are two main types of security validation: (i) evaluation or (ii) accreditation processes. The first one used for products and the second one for systems, where the differences between system and products only lies in their respective size and complexity (the second one being a composition of the first one). The more formal and structured ones are for products: CC, ITSEC, TCSEC; when system security assessment includes more generic definitions of procedures, it is also called Information Security Management System (ISMS) such as ISO/IEC 27001 [10] presented earlier.
In fact, the main problem in security assurance frameworks comes from the fact that assessing security properties of an IT product fully depends on the product itself: its purposes, the technologies used to implement it, its functional and security architecture, its operational environment (e.g. users, interconnections), etc; and finally, the current state of the art of attacks.

All these parameters cannot be constantly standardized for every possible IT product in an up-to-date manner.
Clearly, we cannot evaluate in the same way products such as: firewalls, data bases, web sites or operating systems. It is also very difficult to compare the results of any evaluation of this product, since even if they would belong to the same categories, they would still be different and not subject to the same sets of attacks and threats; that is because of the technologies used or their operational environment.
Thus, every pragmatic evaluation framework takes this empirical fact as an axiom and does not try to assess an overall security rating comparable for any product, since there is no such thing. All known methodologies adopt the same general structure:

1. Identify the product to be evaluated.
2. Define the security problem
    1. Identify the assets to be protected.
    2. Identify the threats for the assets to be mitigated.
3. Defining the security functions to be validated for the product to mitigate the identified threats.
4. Defining a set of evaluation task to apply for the validation of the product's security functions (possibly set of tasks dedicated to the specific product type or category).
5. Defining specific tests for the product to be evaluated.

A main difference between the methodologies lies in:

- either each of these points are directly defined by the methodology and thus directly constrained by it (limiting the possible application of the methodology);
- or the methodology asks for these points to be defined ad hoc for each evaluation, being more flexible but less precise.

Another main difference in the referenced approaches is the fact that the scope of evaluation (functionality evaluated) and the assurance level (evaluation tasks to validate the functions to be evaluated) may be independent from each other or not.
Concerning security assurance for products, there is one main reference. In fact, the state of the art only knows one evaluation process that is recognized internationally. This reference are that is of Common Criteria for Information Technology Security Evaluation, known as Common Criteria (CC). [1] The CC is inspired from two important assurance schemes appeared in United States and Europe: [22] and [23].
The first version of the Common Criteria for Information Technology Security Evaluation, known as Common Criteria (CC) dates to 1994 and the last version to be standardized [1] has released in 2020. Since then, regular revisions have been done but the global approach has not change. The current version accessible on the common criteria portal and used for evaluations is the 3.1 Release 5, to be replaced by CC:2022 release 1 for evaluation starting in 2027.
It keeps the main concepts of ITSEC: (i) the need of a proper ST, (ii) the decomposition of the evaluation in generic evaluation tasks independent of any product or security requirements, (iii) the definition of several evaluation assurance levels, each providing a set of more stringent evaluation tasks and evidence requirements.
Eventually the CC provides a complete description and a reference set of security requirements to write formalized STs and the most extensive list of evaluation activities including any activities empirically recognized as having a potential impact on the final product security.

4.1.1.1 Evaluation Activities

The CC global approach consists of the evaluation of every life cycle element that helps to demonstrate that security requirements identified in the ST can be traced to the real product implementation delivered to the end user. It proposes to evaluate:

- the product life cycle management,
- the product architecture and full specification,
- the user and administrator guides provided with the product to demonstrate that it can be easily used with the proper security configuration,
- functional test performed by the developer and the independent evaluator run on the product
- and finally the vulnerability test

Also those activities complete the whole assessment that the product fulfils the requirements stated in the ST and that those requirements cannot be bypassed. For each of those assurance classes, the CC defines tenth of requirements for the developers and the evaluator. The developer has to provide specific documents including specific information requested by the CC. Typically elements demonstrating the tracing between the document content and the ST elements (requirements, functional description, environment requirements, etc.). some of this evaluation tasks are further described in section 4.2.

No other methodology covers as many aspects or is as well structured. That is why it is the best approach to get the highest security trust and accordingly the most expensive one. Also, it is the only one to benefit from an official international recognition agreement.

The main drawback of this approach is that it provides a static result (certificate valid for one version of the product and for a limited period), time-consuming, and do not scale well to the extensive, networked, IT-driven system. It also does not offer continuous security assurance. Many researchers have made efforts to resolve these challenges. However, it is still an open issue [20], mostly dependant of the nature of the tasks which consists in demonstrating that every implementation details are correct regarding their security requirements.

4.1.1.2 Evaluation scheme

The methodology involves 3 types of actors:

- The consumers
- The developers
- The evaluators

The first audience, the consumers is only targeted to explain them what the real purpose and the real meaning of a certification is. Mainly, it explains to the consumers what the meaning of a TOE and an ST is. The consumer or final user (for PRISSMA it is not the passenger but the entities deploying the service and its components) is in fact never directly involved in a certification process. It has in general an indirect impact on the certification scheme regarding the ST quality and relevance. The whole framework must be adapted so that the certification fits real customers need.

The standard explains to the developer what they must provide in terms of documents and elements of proofs for the different evaluation tasks and so how they participate in the different certification process steps.

For the evaluator the standard describes what evaluation tasks they have to perform.

The above audience targeted by the documents forming the standard is not exactly the real set of actors involved in the known CC certification scheme. Four main roles are identified in the certification process:

- The **evaluation authorities** responsible of the "regulatory framework". They supervise the whole certification process, and they deliver the final certificate. They are responsible of the certification process quality. For example, they are the one to be advised by the standard to carefully check the products, properties, and methods to determine that an evaluation will provide meaningful results.
- The **sponsors** of evaluation who are responsible for requesting and supporting an evaluation. Usually, the ones who pay for the evaluation.
- The **developer** which is the organization responsible for the development of the TOE.
- The **evaluator** which is the organization responsible for the evaluation tasks execution, usually this role is taken by accredited laboratories named IT Security Evaluation Facility (ITSEF)

Regarding the different schemes or certification cases, some of these actors can be the same entities. Most of the time, the developer of the TOE is the sponsor of the certification.
In some rare cases the sponsor can be a specific end-user.
In some evaluation schemes the evaluation authorities is also an ITSEF and thus is both the evaluator and the evaluation authorities.
In France, the IT security certification scheme is defined by the law [2]. This decree identifies the different actors involved in the certification process and their roles:

- Sponsor
    - Sends a certification request to the ANSSI
    - Chooses the evaluation lab
    - Pays the lab for the tests
    - Provides the product and the ST
    - Receives the report produced by the lab
- "Agence Nationale de la Sécurité des Systèmes d'Information" (ANSSI).
    - That validates the security objectives defined in the ST
    - They have to be coherent and relevant regarding the technology and the current state of the art
    - Review the evaluation report produced by the evaluation labs
- Prime Minister
    - Who deliver the evaluation certificate
    - The certificates states that the product tested during the evaluation has the specified security properties since it has been tested in conformity with the current rules and norms in effect
- Evaluation lab
    - It has to be qualified ("agréé") in conformance with the decree
    - This qualification is signed by the prime minister and is:
    - Valid for 2 years renewable
    - In conformance with the ANSSI validation of its technical competences
    - Following an accreditation ISO/IEC 17025 : 2005 by the French national accreditation body (COFRAC) for its ability to follow the appropriate evaluation procedure
- Certification steering comity
    - Composed of representatives from 13 ministry
    - Provides guidelines for certification standards and procedure to be used

In France the certification steering comity has delegated to the ANSSI the definition of guidelines and the choice of standards to be applied for IT security evaluation.

4.1.2 CSPN

France currently proposes a second product certification scheme named “Certification de Sécurité de Premier Niveau” (CSPN, first level security certification) [24]. This scheme only exists in France. This certification process aims at addressing a specific need not covered by other schemes. It aims at providing a proof that a product resists vulnerability testing done by accredited experts in limited time. It provides evidence that the product resists enhanced-basic attack potential (attacker with good competences but restricted time and resources as defined by the CC).
The idea is to provide a certificate which states that the product has been tested during 25 days by security experts and no vulnerability has been found. The CSPN evaluation activities as presented in section 4.1.2.1 are conducted regarding the ST specifications and so aims at declaring that it is conform to this ST security specification (or not).

4.1.2.1 Evaluation Activities

There are four evaluation activities classes defined by the CSPN [24]:

- A conformity analysis to the security target
  - Including documentation review
- A vulnerability survey
  - Survey of security related information that could be available in the public domain for the product and its components
- Penetration testing
  - Where for each security function defined in the ST the experts try to bypass it
- Cryptographic analysis if required

In specific cases e.g., if the product is open source, a minor code review is also added.
The vulnerability tests are chosen by the security experts and reviewed by the ANSSI.
The confidence in the appropriate testing in the product is provided by the fact that only accredited laboratories are allowed to conduct the tests, and by the review of the report produced by the expert by the ANSSI. This report should contain the details of the tests conducted together with their results. If the report is not precise enough, if the tests are not sufficient or if the results are too doubtful, more tests can be required by the ANSSI.

4.1.2.2 Evaluation scheme

In this section we present existing evaluation and assurance schemes that could relate to our PRISSMA security assurance scheme. However, since the CSPN evaluation scheme is the same as the CC one in France we present them both in section 4.1.1.2.

4.1.3 Automated Road Transport System (ARTS) regulation and existing assurance requirements

ARTS (Automated Road Transport System) are regulated by the decree n° 2021-873 of the 29th of June 2021. The decree states that the vehicles integrated in the ARTS have to be type-approved as a prerequisite. Thus, the decree applies after the type-approval phase from the conception of the technical system (vehicles and their technical installations) to the commissioning and then the safe operation of ARTS. This is a main difference with PRISSMA project that includes the automated vehicles lifecycle in its scope.

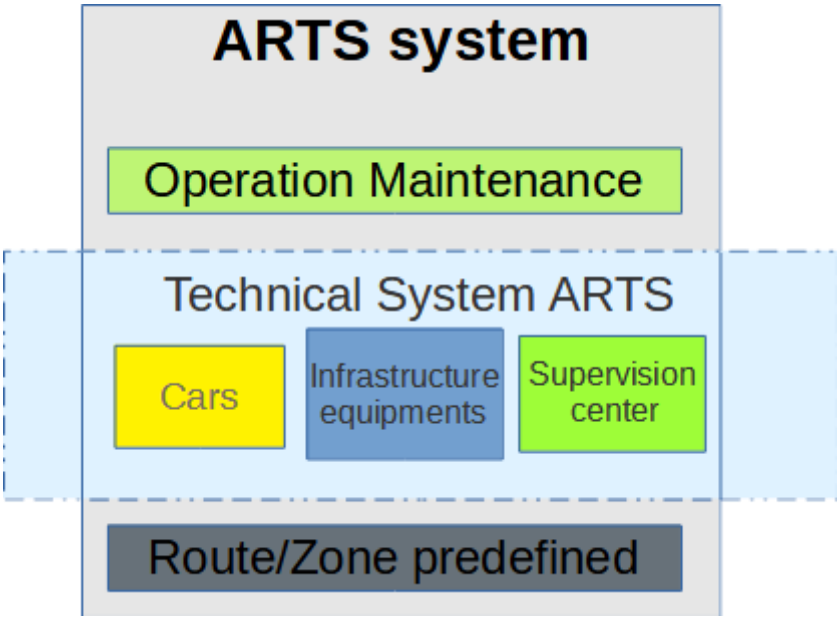
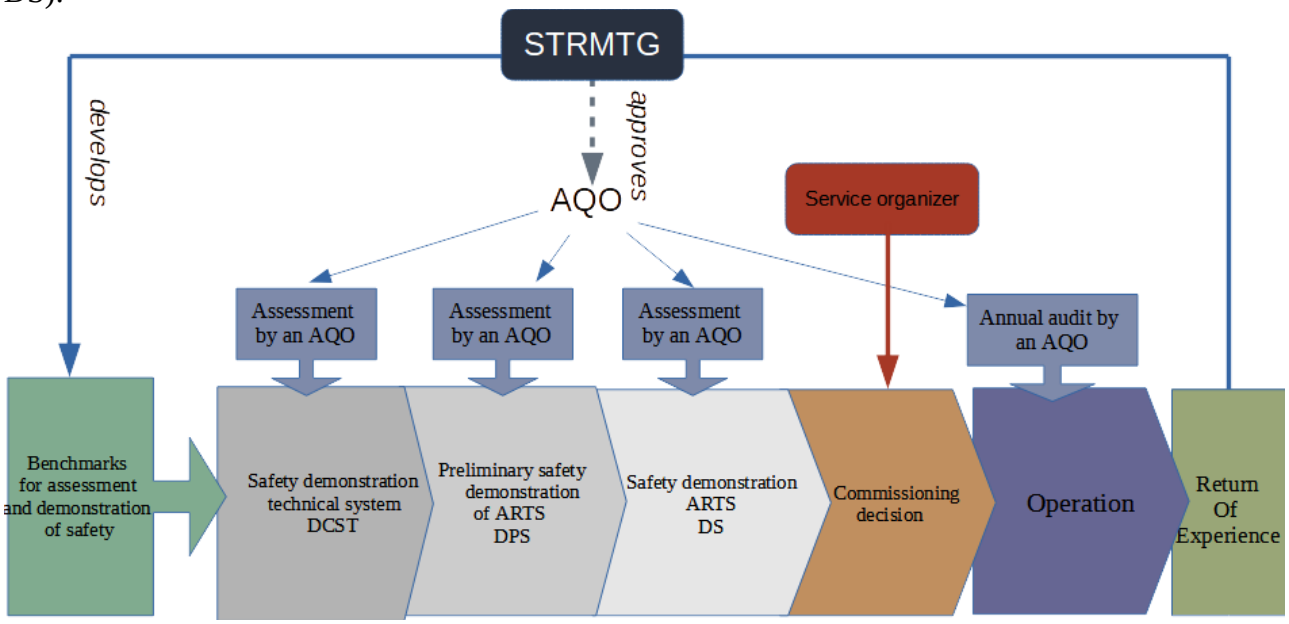


Figure 11 Composition of the bricks of an ARTS

During the ARTS lifecycle, Approved Qualified Organism (AQO) has the mission to evaluate the safety demonstration of the system at some stages (DCST, DPS, DS) before the ARTS commissioning and also intervenes after the commissioning (analyse accidents, diagnose the ARTS, audit the SGS). AQO assesses the compatibility between what has been demonstrated for the vehicle reception and what is demonstrated by the ARTS systemic approach. Additionally, a substantial modification of the system, that consists in the modification of the safety evaluation established in the set of regulatory files (DCST, DPS and DS) subject to an AQO opinion, implies a return to the regulatory procedure of the required safety files (DCST, DPS, DS).



Regulatory process for safety assessment and safety monitoring for ARTS

A particular AQO is assigned to the technical domain “cybersecurity”. The STRMTG has described his missions in the following guidebook yet to be approved by the administration :

"Guide d'application relatif à la mission de l'organisme qualifié agréé pour l'évaluation de la sécurité et pour l'audit de sécurité en exploitation des STRA".
The perimeter of intervention of the AQO is set to the safety of the people carried by the vehicles of the ARTS and the safety of third parties (encountered along the route).
ARTS are not subject to approval as such, but rather to a commissioning decision taken by the service organiser on the basis of:

- The DCST, established by the technical system designer under its responsibility and associated with the positive notice from the AQO,
- The DPS and the DS, established under service organiser's responsibility associated with positive notices from the AQOs.

The system under consideration is a generic notion. It may refer to the ARTS, the technical system of road automated transport, a subsystem, a component or an equipment depending of the object under study.
Depending on the system under consideration, the actors and the responsibilities are changing:

- For the DCST, the system under consideration is the technical system and the responsible entity is the technical system designer.
- For the DPS and DS, the system under consideration is the ARTS and the responsible entity is the service organiser.

After commissioning, the system under consideration is still the ARTS and the responsible entity is the operator of the ARTS.
That is to say that the ecosystems of actors varies from stage to stage.
Each responsible entity is responsible for ensuring compliance with the applicable cybersecurity requirements on the systems for which they are responsible.
A responsible entity may choose to impose all or part of the cybersecurity requirements on another stakeholder (supplier, subcontractor, etc.), it excludes the car manufacturer who is subject to requirements for the vehicle type-approval.
The STRMTG has no direct role in this process. However, it is the authority responsible for delivering agreements, developing and maintaining safety assessment and demonstration standards, and using the analysis of incidents and accidents.

4.1.4 System audits

Several audit frameworks exist in France an in Europe. We have already presented the ISO/IEC 27001 and 27002, ISO/SAE 21434 (cf. section 3.1.1) audit requirements but we could also mention in France the PASSI LPM audits (audit performed by qualified auditors "Prestataires d'audit de la sécurité des systèmes d'information" to assess conformity to the requirements defined by the French "Loi de Programmation Militaire"). In all three cases, the audit approach is the same and follows roughly the same steps:

1. An authority qualifies auditors as competent to assess a system compliance with the requirements defined by the reference document (the standards in the firsts cases and the LPM in the second).
2. A system\project managers who wants or has to have its systems certified (for ISO certifications), qualified (for PASSI LPM audits), or homologated (for author audit framework not mentioned here) contact a qualified auditor either referenced by the authority qualifying them or simply looking for their own advertisement. For instance ISO 27001 can be found searching on the internet when PASSI can be found on the ANSSI web site [25].

3. The system\project managers together with the auditor identify the system or part of the system target of the audit and the requirements defined by the audit framework that apply to the target.
4. The system\project managers gather or has elements produced to fulfil the audit proof production requirements (each audit requirements usually requires document to describe and justify how the system target of audit fulfils the audit requirements, e.g. HR documents demonstrating that employees with required competences are in charge of the system, network documentation demonstrating the security architecture of the system, documents defining security policies or best practices to be applied by the system managers and users, etc.
5. The auditor reviews the elements of proof provided by the system\project managers and provides reports until reaching a satisfying level of confidence that the evidences demonstrate the coverage and fulfilment of the audit framework requirements.
6. The auditor visits the sites included in the audit perimeter and presented in the audit documentation provided at the previous steps. They validate on site that all documented elements fulfilling requirements are conform to what has been described and in fact fulfil the requirements.
7. The physical audit usually ends by a meeting with the audited personnel to present the first audit results and required updates if any, in case of none-conformity.
8. If all requirements are deemed fulfilled by the auditor, they provide a positive audit report explaining how the provided inputs and the audit demonstrated the system conformity. Or they provide a report identifying the requirements not fulfilled and the necessary updated required to re-assess the conformity, going back to step 4.

4.2 PRISSMA assurance requirements

The PRISSMA evaluation framework must assess several aspects of the safety and security of ARTS. We must define functional, safety and security validation aspects to guarantee that ARTS provide the best service in the safest way to the passenger. As presented in section 2, those systems are the composition of several complex components which can already be subjects to their own regulation or best practices requirements. PRISSMA focus on the validation of the final security and safety of the complete ARTS.

We design the PRISSMA approach to be complementary to existing requirements. For instance, PRISSMA is not meant to replace or add new type approval requirements or replace current work on ARTS requirements (cf. section 3.1.2). In fact, we suppose here that the PRISSMA evaluation framework assume that vehicles and their components are already type approved, and IT components meet all other regulation requirement (e.g., LPM requirements for specific transport systems, UNECE R155, RED, NIST, etc.). We provide additional evaluation objectives trying to avoid redundances. However, even if PRISSMA partners have a good knowledge of those pre-existing requirements, some redundancy might still exist. In that case previous evaluation reports or produced evidence can and shall be reused. Also PRISSMA comes in addition to existing ARTS cybersecurity requirements [5]. As already discussed previously PRISSMA scope is wider in terms of targeted systems and it also aims at providing more specific requirements and higher assurance. In the case were [5] is applicable, both sets of requirements should be merged where PRISSMA should provide more precise requirements to be integrated in [5], if not the most restrictive requirements should be kept.

In this section we present the assurance requirements we define for the PRISSMA framework to validate ARTS security. As define previously, assurance requirements are evaluation tasks to be performed to assess that the target of evaluation fulfils its security requirements. Thus,

we present here evaluation tasks required by the PRISSMA evaluation framework to assess the security requirements that we have define in section 3.
So far, we have identified two categories of architecture components (cf. section 2.1 and 2.2) for which we have define two different level of requirements, i.e. requirements with different technical details levels. The first one is the systems components including the ARTS itself or any of its sub-systems formed by systems of systems, and the more ITS specific and smaller technical components. In accordance with these two levels, we define two different assurance approaches with two different sets of assurance components: (i) site assurance requirements (cf. section 4.2.1), .(ii) equipment requirements (cf. section 4.2.2).

4.2.1 Site assurance requirements

PRISSMA evaluation framework requires to have audit performed by independent authorized entities for:

- Central ITS stations
- Developers' premises
- The complete ARTS

Those audits should follow ISO 27001 approaches for requirements identified in section 3.1.1.1.
We add **specific requirement to regular ISO 27001** for the **complete ARTS**. As for [5] but with a larger scope (not only identifying transport related threats), the ARTS architecture must be defined and a **risk analysis** shall be perform on it in order to identify all **critical system site and components** that either provide communication means supporting traffic management or autonomous driving function. This should include all components types for which we have define requirements in section 3.
All such identified elements must be **certified as defined by this PRISSMA framework**, as well as the complete system (which will have to justify the identification and certification of those sub-components to be certified by the PRISSMA approach).

4.2.2 Components

For the critical components of the system including at least the one identified in section 3.2, we propose to use an assurance evaluation strategy that is a trade-off between the CSPN and the CC approaches. This can also be applied to other critical components as identified by the risk analysis (cf. section 4.2.1). Then either the evaluation is based on the specific security requirements defined in section 3.2 or for products not identified by us and for which assurance needs are identified during the ARTS risk analysis specific security requirements are to be defined for the evaluation.
As presented in section 4.1, CC is the main and only internationally recognized assurance approach which can provide the highest assurance levels (confidences in a product security properties). However, the cost of such evaluation is too high for now for the automotive industry and the autonomous transport services. The size and complexity (and current level of readiness) of those systems does not allow to require evaluation of all critical component that last month's and sometimes up to years. The CC formalism, their strong structure of evaluation activities and their high input requirements are currently a limitation for its adoption in the domain of autonomous ITS.
On the other side, the CSPN approach, which provides faster and cheaper evaluations, might not provide sufficient assurance for cyber-physical system for which security breaches can lead to passengers' injuries or death. That's why we propose an in between assurance approach that would provide more assurance than a CSPN (for a better protection), but easier to

pass than full CC approach that is not industrially acceptable. However, this approach should still correspond to high assurance level evaluation as defined by the most common evaluation schemes (CAL 4 for [5] or high for [26]).
We have identified evaluation tasks defined by the CC and not applied in CSPN that provide from our point of view efficient and complementary assurance to be added to the CSPN approach. We propose thus to combine those to the current CSPN evaluation tasks. We also identify in section 5 how to decrease CC certification schemes burden which require several administrative steps and evaluator requirements for all evaluation tasks that might not be necessary in PRISSMA context.
In the end the list of evaluation tasks we proposed the PRISSMA security assurance framework are the following ones, that we present in more details in the following sections:

- Security target evaluation
    - Reusing CSPN format and evaluation task definition
- Architecture (ARC)
    - Re-using CC objectives for the task architecture evaluation task ADV_ARC.1
- Functional specification evaluation (FSP)
    - Re-using CC objectives for the task functional specification task ADV_FSP.3 (Functional specification with complete summary)
- Guides (AGD)
    - Re-using CC objectives for the operational user guides evaluation task AGD_OPE.1 and the installation guides evaluation AGD_PRE.1
- Tests (ATE)
    - Re-using CC objectives for the functional tests and coverage evaluation task ATE_FUN.1 and ATE_COV
- Vulnerability analysis
    - Reusing CSPN approach but introducing the iterative aspects of the CC evaluations
- Cryptographic analysis
    - Reusing CSPN approach but introducing the iterative aspects of the CC evaluations

In addition to this new set of assurance evaluation task, the other important differences between CSPN and CC, for which we propose a trade-off are:

- The normalisation and formalism of the input description
- The iterative aspects of the evaluation

For the CC evaluation, a lot of specific vocabulary and notions are defined. Those are very useful in a context where the objective is to have the most thorough and exhaustive assurance process where every security details need to be formalized and reviewed. Without a high level of structure and formalism of the work it is unlikely to be able to get the highest assurance possible. Since again our goal here is not to get the highest but rather the most efficient assurance we do not require to reuse the complete CC formalism but only a sub-part of it. The main notion of interest for us defined by the CC are:

- Target Of Evaluation (TOE)
    - The product to be evaluated in its exact version
- Security Functional requirement
    - The security functions of the product to be evaluated
- TOE Security Functionality (TSF)
    - The subset of the TOE that implements the SFRs
- TSFI

    - The interfaces giving access to the TSF
- However, in the PRISSMA security assurance framework context we do not need to use the full CC formalism. We will on us the aforementioned definition which are useful in any security assurance context so we will adopt them here, but we won't use any other formalism.

4.2.2.1 Security target evaluation

The first element to evaluate before starting any other evaluation task is the security target (ST). The ST is the document that will identify what has to be evaluated and under which condition.
The evaluation of the ST consists for the evaluator in verifying that all the required elements are correctly defined, that is fully understandable for the evaluator. The required content for the PRISSMA assurance framework is the content of a CSPN ST [24]:

- An unambiguous identification of the TOE to be evaluated
- An identification of the TOE developer(s)
- A TOE rationale describing the use of the TOE, and the context in which it is supposed to be used
- The technical environment in which the TOE works (computer model, operating system, etc.)
- The list of sensitive assets that the TOE must protect;
- Environment-specific measures required for the proper execution of the TOE
- The threats against which the TOE offers protection;
- The description of security functions implemented by the TOE that counter identified threats. These functions are the one to be evaluated.

4.2.2.2 Architecture

Objectives of the task
The study of the TSF security architecture allows to analysis if it is properly designed to achieves the desired properties. Without a sound architecture, the entire TOE functionality would have to be examined. Thus this task allows to verify that the developer is able to justify that their product's TSF fulfill the SFRs in the ST.
Task input
The developer shall provide a security architecture description of the TSF to demonstrate that they have design and implement the TOE so that the security features of the TSF cannot be bypassed and so that it is able to protect itself from tampering by untrusted active entities.
The developer must provide documentation to justify their TOE security architecture and how it satisfies the SFR defined in the ST. Thus they must provide information on such elements as:

- Arguments on the security of the boot sequence
- Arguments on the security domains used for segregation
- Integrity of the TOE in operation (how measures assure TOE integrity)
- How the TOE design does not allow to by-pass security functions

The idea is for the developer to provide evidences that they understands the security design of their products and knows how it counters the threat and possible attempt to bypass the security measures.
Output

The output of this evaluation task is the validation of the security architecture and choices of the developer for the TOE regarding the internal TOE conception and specification provided for ADV_FSP and ADV_TDS.

4.2.2.3 Functional specification evaluation

Objectives

The objective of this task is to determine if the conception of the TSF implements correctly the SFR it is linked to. For that the evaluator has to validate the conception of the TOE by verifying how the TOE is composed in terms of sub-systems and for higher level of assurance how these sub-systems are in their turn decomposed in lower sub-systems or modules and how each of this components interact with each other and participate in TSFs.

Input

The developer shall provide the description of the TOE's design and a mapping from the TSFI of the functional specification to the TOE sub-systems.

The developer has to provide the decomposition of the TOE into smaller sub-systems and module (smallest functional entities in terms of design) in order to provide more details on how the TOE works and how security functions are implemented and decomposed in the TOE. Thus the documentation must describe the decomposition of the TOE and for each sub-system or module:

- Its purpose
- Its general behavior
- Its interfaces
  - Their specification (details of the operations executed)
  - The Format of data of the input and output
- Its interaction with other sub-systems/modules
- How it supports/implements TSFs

Output

The output is the validation of architecture choices for the implementation of the security. The idea is to verify that the TSF really implements SFR, i.e.: correct interpretation of what the SFR are, no cases identified in the SFR have been forgotten, correct interactions of the sub-systems, etc.

For complex systems managing to completely control all the aspect of one design is very challenging: mistakes can be done, things can be forgotten or miss-interpreted, etc. The verification of the architecture helps to detect problems that can lead to security breaches.

The required information might imply some sensitive industrial knowledge or implementation choices. Thus they have to be performed by a trusted independent entity.

4.2.2.4 Guides (installation and operation)

Objectives

The objective of this task is twofold. First it aims at validating that the documentation or supports provided to the TOE user allows him to transform the delivered object (cf. delivery procedures) into an operational TOE such as identified in the ST. If a security product is not correctly installed and configured, most of the time it does not provide the expected security features. It can even add vulnerabilities to a system or dangerously provide the illusion of security . Many default configuration of security product deactivate or do not configure security function. They have to be correctly activated and configured for the operational system to be secured. Thus it has to be evaluated that the TOE guides allow the TOE installation and configuration by the user to verify every recommendation or constraints made on in the ST.

Second, this evaluation task also validates the operational guidance allows the user to operate the TOE in use cases stated in the security target.
Input
For this task the inputs are the TOE, the installation and operational guidance.
When the installation is too complicated, or the product always integrated by the developer themselves in the user operational environment then the regular installation procedure is the subject of the evaluation.
Operational user guidance refers to written material that is intended to be used by all types of users of the TOE in its evaluated configuration: end-users, persons responsible for maintaining and administering the TOE in a correct manner for maximum security, and by others (e.g. programmers) using the TOE's external interfaces.
The objective is to minimize the risk of human or other errors in operation that may deactivate, disable, or fail to activate security functionality, resulting in an undetected insecure state.
Output
Validation of the guidance or installation means to guarantee that the user can in fact get an operational TOE that correspond to the one described in the ST, including the conformity to all the hypotheses done in the ST. The verification includes the evaluation of the consistency with the reception procedure. It validates that options that correspond to the use case stated in the ST must be present and configurable as such. The validation is verified by the fact that the evaluator must be able to install correctly the TOE. The validation does not require specific knowledge nor access to sensitive data. Thus, it can be done by any independent body. It could be done by example by the end user itself, here the car manufacturer.
Concerning the validation of the operational guidance it should be demonstrated that it provides enough information to the user to use the TOE in a secure way, as described in the ST and that the guidance are consistent with the ST and functional specification provided in ADV_FSP.
The validation does not require specific knowledge nor access to sensitive data. Thus, it can be done by any independent body. It could be done by example by the end user itself, here the car manufacturer.

4.2.2.5 Cryptographic analysis

Specific cryptography analysis requirements are defined by the ANSSI. We will also require the same analysis.
In fact, in the French certification schemes, the cryptography functions have to be evaluated when they are used by the functions identified as to be evaluated in the ST. In that case, the evaluator must review technical documents made available by the developer to validate the following elements.
Input
The developer shall provide documents providing the following information:

- Information on used algorithms including:
  - The description of the cryptographic functions provided by The product (encryption, signature, key management, etc.)
  - The reference of algorithms to recognised, unambiguous standards, whose technical details are easily accessible and without conditions, with the settings and operating modes of their implementation
- Information on key management including:
  - The size of the keys
  - The key distribution method

    - The key generation method
    - The key storage format
    - The key transmission format
- Information relating to data processing including:
    - The description of pre-processing experienced by plain text data before encryption (compression, formatting, adding a header, etc.)
    - The description of post-processing of encrypted data, after its encryption
    - A reference product outputs set produced from a randomly chosen plain text and key

Based on those elements which real implementation has to be validated thanks to code review activities, the evaluator has to assess their compliance of with ANSSI's requirements defined in the "Guide des méchanismes cryptographiques" [15].
Compliance of the implementation of these mechanisms by the product is verified in different ways:

- By comparing the results of cryptographic processing performed by the product with respect to a reference implementation. This implies that several reference inputs/outputs (key, plain text, encrypted) are made available to the evaluator.
- By analysing the source code with unit tests of certain functions (for example, check that an AES function does in fact perform an AES;
- By checking that the TOE communicates in fact in encrypted mode with a reference equipment.

4.2.2.6 Vulnerability analysis

General IT vulnerability analysis
Even if security functions comply with their its specification, their implementation can still introduce the possibility to bypass, disable, alter, or make them ineffective by an attack. Such attacks usually take advantage of shortcomings in the implementation of the product, its design or in its underlying security principles.
The evaluator's role is to:

- Identify TOE functions and APIs that implement security functions;
- Analyses the conformity of the identified functions and API with the required SFRs identified in the ST
- Identify potential vulnerabilities (bugs, potential weaknesses, known vulnerabilities in similar products, protocol or library used by the TOE, etc.)
- Use any means using resources limits defined for the evaluation to exploit those vulnerabilities (the ones identified by the CSPN [24])

Such an analysis must consider the level of resources required by the attacker to succeed. Not all attack can be tested in a limited amount of time, thus vulnerability analysis shall be limited by the required resources to test them: time, required expertise, equipment capabilities, etc. Those limitation must be proportional to the identified threat or the desired assurance level to be obtained. We re-use the ones identified by the CSPN.
AI vulnerability analysis
Specific vulnerability analysis shall be performed on AI components. For each of those components a dedicated vulnerability identification based on existing state of the art attacks for comparable AI functionalities.
A survey shall be provided by the evaluator on existing related evasion attacks, and a minimal exploitability assessment shall be run also based on resources limits defined for the evaluation to exploit those vulnerabilities (the ones identified by the CSPN [24]).

Exploiting a potential AI evasion vulnerability can be very costly and it cannot be expected to spend too high efforts. For that reason, the AI vulnerability assessment shall be performed with the support of the developer, who should provide arguments on whether state of the art evasion techniques could be applicable or not.

5 PRISSMA assurance scheme

ARTS systems are cyber-physical systems whose failure can have dramatic consequences or costly physical damages impacting private or public infrastructure, implying potential injuries or even death. While on the other hand they are meant to provide highly beneficial impact on our societies: increasing travel safety, minimising environmental impact, improving traffic management, and maximising the benefits of transportation to both commercial users and the public. To ensure that those systems are more beneficial than threats, adequate security assurance must be provided to guarantee that they cannot be easily subject to too attacks. This security assurance process is to be performed together with all other functional and safety analysis required by the complete PRISSMA framework which much more activities than just cybersecurity assessment.

For that we have already define both, security requirement (section 3) and assurance evaluation requirements (section 4) to assess that the systems and its critical components meet those requirements.

However, adequate final assurance is still a trade-off between the risks, the evaluation efforts, and their cost to assess that the system and its components meets the appropriate requirements. For the same set of security requirements to be validated by the same set of assurance requirements, both the cost and the confidence in the results can greatly vary depending on the assurance scheme used to perform the evaluation. To this respect, if the evaluation cost is too high ARTS systems might not be deployed, but if efforts or confidence in the results are too low, basic attackers might be able to do important damages. Thus, the quality of an adapted assurance framework depends on this trade-off, which is highly impacted by the choice of its framework of execution.

It is to be understood that the **cost** of an evaluation depends on several factors and not just direct cost associated to tests. Costs are:

- Efforts related to production of evaluation inputs and their required correction (cf; section 4). Most of the time the main cost is the production of a secured product or system, i.e. a product or systems that passes the evaluation
- Efforts to run the evaluation tasks
- Duration of the whole evaluation process

All those different costs must be considered when designing an assurance framework. For instance, one main drawback of CC evaluations is the time required for the whole certification process to end. It takes month up to years in worst case scenarios.

The cost greatly impacts the trust in the final evaluation result. Obviously, spending 10 days for vulnerability is not the same as spending 40 days result wise, but in the same way for the same duration having the tests performed by a senior independent expert or a junior tester from the developer's company has different results for different costs.

To optimize this cost/assurance trade-off it is very important to define appropriate **targets** in terms of input and output. It directly impacts the cost of the evaluation since it directly impacts the assurance evaluation process feasibility and ease of execution.

If we have already identified inputs for PRISSMA evaluation tasks in section 4, the list of expected **outputs** of the evaluation is **still to be defined.** Output requirements can directly impact the cost and benefits of the final assurance. Their definition will provide constraints on

the assurance tasks and assurance scheme that can produce them. Output can vary in form: reglementary certification, private certification, audit report, system qualification, etc.; in confidentiality: it can be a confidential report produce for the sole developer or service provider, it can be an official certificate signed by the prime minister with a time limited time recognition, etc.; but also, its validity period can vary: month, years, always valid, valid until regulation/reference standard modification, etc.; and it can have a level of recognition depending on the entity endorsing the final assessment: certification provide by a private entity, by the administration, by an accredited body, etc; and finally this output can cover only part of the target, its totality, etc.
Once the targets are properly defined then we can consequently define:

- The **assurance tasks** applicable to the input that allow to provide the expected evaluation output (already defined in section 4.2)
- The **assurance scheme** that will enforce correct execution of the assurance tasks and output production and help to keep an UpToDate assurance scheme

Evaluation tasks have already been defined in the section 4.2. Here we discuss the assurance scheme necessary to manage the proper execution of those task and the proper validation of the requirements identified in section 3.
Those output production will be more or less difficult to obtain depending on the evaluation input (the target). In fact, certifying a complete system is much longer and more complicated than certifying one single product since their size and complexity differs by several order of magnitude.
In this section we will first define the expected output of our assurance framework (system wide, specific component, duration, level of recognition, etc.). Then we will define the required competence and responsibilities to perform, manage and maintain evaluation quality and relevance over time.
It is worse mentioning that our approach is complementary to ARTS regulation requirements (cf section 3.1.2) and French requirement on OQA [26]. Our evaluation framework is built to tackle more use cases (e.g., open roads without predefined path, systems with full automation capabilities and no human driver in the loop, etc.). It is also meant to provide more precise and higher assurance requirements including certification of site and product that is out of scope of [26].

5.1 Evaluation output

Evaluation output might vary in:

- Form (report, certificate, accreditation, etc.)
- Perimeter (complete system, sub-parts, specific components)
- Level of recognition (European, national, international, industrial, etc.)
- Validity period (unlimited, limited, renewable, etc.)

In the PRISSMA context we aim at guaranteeing the security of any operational AI based ARTS. To do that we have to define an assurance process that covers both the whole system and its entire operational life, from the time it starts to provide service until its decommissioning.
We must provide means for assurance evaluation that cover the entire system and its entire life. However, we have already discussed that evaluating an entire system usually leads to audits which cannot go into deep technical evaluation due to the size, complexity and implementation freedom of their targets. If audits are beneficial, they are not sufficient for the level of assurance we are looking for. Thus, we propose a twostep composition approach which requires to **evaluate first** both type of ARTS main **components** type: IT sites (ITS central

stations, vehicle remote control sites, AI maintenance servers, etc.) as well as evaluate more specifically and thoroughly the most critical technical components of the ARTS. Then, all those local evaluations are used and integrated in the **final global ARTS accreditation audit**.

5.1.1 Evaluation output format

So, in terms of form, the PRISSMA evaluation results will provide three types of results:

- **Product certification** produced based on evaluation results presented in the form of an evaluation report.
- **Site certification** produced based on site audit reports.
- **ARTS certification** based on complete system audit including product accreditation and site accreditation.

5.1.2 Evaluation output validity period

5.1.2.1 Product certification

Current CSPN and CC product certificate are valid for a period of 5 years. This is a relatively short period regarding ARTS lifetime which should be potentially over decades. This corresponds to the rapid evolution of the state of the art. For most Commercial Of The Shelves (COTS) widely used in common IT components such as: OSs, java libraries and JVM, web-servers, etc.; vulnerabilities are found on a weekly basis [27]. Thus, it is not possible to trust evaluation results without any reassessment for on a too long period. Study must be made to validate that the state-of-the-art evolution does not impact it.

Both in CC and CSPN certification, the recognition period of the certificate is limited. Also, the developer must continuously do vulnerability analysis of its certified product to verify that no known vulnerabilities exist in their components. If one is identified, they have to inform the certification body. A specific study is then made to assess if the identified vulnerability has in fact an impact or not (is exploitable or not in the context of the certification). If yes, the product must be recertified. If no, then the certificate validity can be maintained. The way the developer performs this surveillance is not formalized and left to the developer choice.

In our case, we recommend following a similar approach, that we slightly adapt. We propose to have the same validity duration, but slightly change the continuous surveillance process requirements and propose a validity extension mechanism of the certificate when the product did not evolve.

For the continuous surveillance, we formalize that the developer must send a monthly report to the certificatory to present its surveillance results. This report must be validated for the certificate recognition to go on. When the certificate recognition expires, then if the developer wishes to maintain the certificate, a vulnerability analysis shall be performed by the evaluator (cf. section 4.2.2.6). If no vulnerability is found regarding the state-of-the-art evolution, then the validity of the certificate is extended for 2 years.

A grace period of 2 month can be granted to change product if at the end of the validity period the certificate recognition prolongation fails. However, it is the ARTS manager duty together with the developer to make sure that all components required to be certified have a valid certificate and change adequately those who don’t.

5.1.2.2 Site certification

Concerning the identified site for which threats impacting the security and the safety of the ARTS, we have defined audit requirements (Cf. section 3.1).

Those audits must generate an audit report and when this report presents a successful audit result (audit demonstrate that all requirements are met) then the considered site can be certified.

We recommend combining the ISO 27001 and ARTS cybersecurity guides [5] best practices regarding certification validity and audit periods. For ISO 27001 a certificate is valid for 3 years during which an annual surveillance audit must be performed to evaluate that the ISMS is still correctly managed. As for ARTS guidelines it is recommended to have a yearly audit. So having a 3 year certification of the ARTS site with annual audit as for ISO 27001 certification also matches [5].
If this surveillance audit fails, the site owner can perform corrective actions to maintain it certification. If those correction action are validated by the auditor, the certificate is maintained otherwise if they fail to do so the certificate is revoked.
For developers' site, this mean that there are not allowed to provide new product or further updates to the ARTS before being certified again. For the other site this will imply their exclusion from the operational service.

5.1.2.3 ARTS certification

For the complete system a similar certification is performed as for its main site certification. This will be based on and will reuse all product and site certification, as identified in section 4.2.1.
Thus, we recommend to have the same validity management as the one presented before for sites, i.e. 3 years validity with annual surveillance audit. However, with the extra requirements that all critical components are currently certified and if some elements are in a grace period for certification renewing then the final surveillance results shall be delivered after validation of the certificate renewing or correction of the situation (changing of the elements for a new certified one, adaptation of the architecture to exclude the uncertified component for the scope of it critical elements, etc.).
This certification is a PRISSMA proposition compatible to the ARTS cybersecurity application guide [5] as longue as all PRISSMA requirements are met using the guide approach. Also, the PRISSMA approach can be applied to a wider range of ARTS and provides more precise security and assurance requirements for the most critical recuring components.

5.2 Scheme

An evaluation scheme is the set of entities involved in the evaluation process and their different responsibilities and competences. When defining a scheme, not only we have to identify entities responsible to provide elements or perform tasks but also we have to define if and how those entities are to be qualified to fulfil the scheme requirements.
The definition of the this set of involved entities and their associated duties can have a tremendous impact on the evaluations' different parameters (recognition, costs,
In this section we identify the required entities to be involved in the different scheme task: input provider, input evaluator, certificator, site auditors, scheme manager, entity qualifying required qualified evaluators and/or auditors.

5.2.1 Actors and responsibilities

5.2.1.1 Evaluation sponsor

The evaluation sponsor is the responsible for requesting and supporting the evaluation. For the complete ARTS validation, several sponsor might be necessary: (i) sponsors for site and components audits and evaluations, (ii) sponsors for the complete ARTS audit.

5.2.1.2 Develope

The developer is a generic term covering any entity or technical team responsible and capable to provide required inputs for product certification (as defined in section 4.2.2). In the field of

automotive industry, the corresponding personnel might belong to different societies. This is one of the difficulties that can arise during evaluation. This is for the evaluation sponsor to overcome that difficulty.
The developer is responsible for providing inputs that fully answer evaluation requirements. This includes the responsibility to develop a secure product, meeting the PRISSMA security requirements defined in section 3. As well as producing adequate elements to demonstrate that fulfilment, adequate in terms of quality and coverage to convince evaluator that it is the case.
If precise or specific requirements cannot be made in term of documentation structures, content, format, models used, developers' tests, quality and development process, etc. Developers have to be free to use any means that allow them to answer PRISSMA requirements. However, we still recommend using integrated security approach in product life-cycles. In fact, if a developer uses low quality approaches, unprecise specification tools, weak management process, etc. The quality of their product will be more likely to be lower than if they use more precise or advanced technique. One of the main costs of an evaluation is to develop a product that can pass the evaluation. Starting an evaluation and failing it because the evaluator finds vulnerabilities, requirements not implemented, etc.; adds delays in the certification process, developments cost that could have been lowered in anticipated, evaluators payment, etc.
To reduce those risks, we recommend for entities that do not already have very strong competences in security developments, testing, development managed, integrated security by design process, etc.; to use available tools whenever possible. Plenty such tools exist. Not all are pertinent for every development and still few are available for AI but for general IT several interesting solutions exists. It is for the developers to review existing tools, but we strongly suggest them to do that review and adopt as many as possible.
In the PRISSMA project since we recommend to the developers to use best quality and assurance approaches for their developments, which implies to start as early as the design phase to formulate the proper Requirements Engineering (RE) matching assurance requirement. PRISSMA partners develop tools than can be used by developers targeting our certification. Thus we can at least recommend those: MAAT REQ and DIVERSITY.
Indeed, formulating requirements that accurately describe intended system behaviours, and eliminating misunderstandings, is a crucial but difficult task for designers targeting security assurance certification. The complexity of requirements has made the use of formal methods widespread to check their desired properties and clarify their formulation. The CEA's tool MAAT REQ implements a process algebra-based framework for real-time requirements analysis [28] [29]. MAAT REQ provides an automatic transformation of structured textual requirements into a timed Process Algebra (PA) and exploits it in the exploration of their intended real-time behaviours and concurrency. The tool can test whether certain requirements can be covered at some point in the PA execution with traceability feedback. Moreover, it can detect various requirement inconsistencies that the tool can highlight as synchronization lacks or deadlocks in the PA execution.
Then we can also recommend Model Based Testing (MBT) at operation-time. Using formal methods to generate test cases and to compute verdicts has been widely studied in the frame of Model Based Testing (MBT). On the other hand, Runtime Verification (RV) approaches are often found for solving the Oracle problem in MBT. Algorithms to solve the oracle problem in MBT are technically very similar to offline RV algorithms: they both consist in analysing an execution (typically a log) to detect non-conformance to a specification. The CEA's tool DIVERSITY implements algorithms to generate test cases and solve the oracle problem against specification as (product) of symbolic automata using constraints solving techniques [30] [31]. The CEA's tool IAT provides algorithms of offline RV against specification as

interaction models akin to UML Sequence Diagrams, or Message Sequence Charts (MSC) which are more focused on specifying communication flows in the system [32] [33]. Both tools can detect non-conformities to the soonest. Thus, they allow for early action by applying adequate batches. In particular, in case of offline RV, non-conformities can reveal weaknesses in communication flows that are potential opportunities for malicious actors.

5.2.1.3 Evaluators

5.2.1.3.1 Site

To perform the audit of the different site and the complete ARTS we require to have them performed by accredited and independent auditors. Since we advocated for the use of ISO 27000 suits to perform site audits, we also advocate that accredited auditors for those audits intervene in our evaluation scheme.
In fact, self-audit even if correctly performed, do not provide the same level of confidence as independents audit performed by personnel with the appropriate and recognized competences. When it comes to self-assess efforts provided by our own company, two common biases can erase: (i) difficulty to be able to see its own mistakes, (ii) being tempted to tempering results for its own benefit.
The use of independent accredited auditors helps to overcome those biases and provide more trustful results.

5.2.1.3.2 Product

Evaluating cyber security is a challenging activity that requires specific expertise to be constantly maintain and regularly challenge to see if it matches the state of the art. The trust in evaluation results directly depends on the trust in the evaluator competences. Here again, in case of self-assessment or assessment by a private company not officially accredited can biased the results in the same way as for the audits.
In our framework we define that it is the responsibility of the scheme owner to determine which evaluator should by accredited for product evaluation. In France, this could be an "Organisme Qualifiée et Agrée" (OQA) as defined by the decree n° 2021-873 of the 29th of June 2021 or a "Centre d'Evaluation de la Sécurité Des Technologie de l'Information" (CESTI) accredited by the ANSSI. It could also be PRISSMA Qualified Evaluation Entities (PQEE) qualified by the PRISSMA scheme owner, if none of the two other possibilities are deemed suitable for that role.
As for ISO 27000 series auditors, or any such accredited activities, PRISSMA evaluation entities shall be accredited for a limited period. It is the responsibility of the scheme owner to audit et accredit periodically the PQEE. In the PRISSMA framework, we require a 2-year qualification of PQEE responsible of product evaluation.
Again, such accredited entities for cybersecurity evaluation already exist in France: CESTIs. We recommend using CESTI, in our framework. However, it is up to the scheme owner to decide if so or to define alternative process to qualify the necessary entities and maintain their qualification.

5.2.1.4 Scheme owner and consortium maintenance

Everything presented in this document must be validated and maintained overtime. **An entity must be responsible for that**. This entity does not have to perform all tasks associated to the maintenance of the framework, they can delegate part of all of it, but they are responsible of its good management.
Several organisms could be owner or share ownership of this scheme, such as:

- The "Agence Nationnale de le Sécurité des Systèmes de l'Information"(ANSSI) which is the French national agency already responsible of existing cybersecurity

certification schemes, since it has the expertise to deploy and supervise such cybersecurity related evaluation schemes.
- The “Union Technique de Automobile, du motocyle et du Cycle” (UTAC) who is a well know actor in the field of automotive testing and certification since 1945.
- The “Centre National de Réception des Véhicules” the French national entity responsible for type approval certification since it has the expertise of qualifying automotive products
- 5]Any other public or private entity

This entity or consortium can pass on this responsibility to other later own if need be.
We strongly recommend that one entity or consortium should be identified by the PRISSMA partners before the end of the project to guarantee the PRISSMA framework maintenance and continuity even if the PRISSMA consortium is not mandated to or does not have the power to enforce it.
When outside the scope of the decree n° 2021-873 of the 29th of June 2021 on STRA, it is not expected for the scheme owner to have all the required competences to perform all scheme maintenance activities. It can be foreseen that the scheme owner can setup consortia to manage the different scheme components. For instance, one such consortia could be responsible for maintaining components requirements update, another maintain a reference architecture and its associated threats, another one maintains PQEE qualification process, etc. Those consortia can be any composition of private and public entities as seen feet by the scheme owner.

5.2.2 Certification

In our framework we have define two level of certification, site and products and complete system. We do not require to have a specific type of certification. This can be provided either by a fully private scheme where all evaluators and certificators are private entities, or it can be a public certification provide by the responsible administration. This is up to the scheme owner to define it.
Our framework is not meant to supersede any existing automotive certification, homologation or any other regulatory requirement. For instance, this is not part of the type approval process and vehicle running ARTS services will have to be approved.
Our framework is meant to provide to any authorities in need of such framework to define ARTS cybersecurity certification requirement before any deployment. Thus it up to this authority to become the scheme owner and to set its own certification requirements based on our framework.

5.2.3 Scheme maintenance

As define previously, the scheme must be maintained over time by its owner or entity to which they delegated this task. Maintenance must be performed and shall define periodically its own goals based on scheme execution feedbacks. Such maintenance should cover such activities as:
- Reference architecture and functional requirements maintenance
- Threat identification
- Components security and site audit requirements
- Scheme description maintenance
- PQEE qualification and overview
- Identify and resolve potential conflicts, issues, discrepancy between evaluations, actors, reference documents, etc.
- Compatibility with other existing frameworks or regulation

This list is clearly not exhaustive, and it is actually for the scheme owner to make sur that any other maintenance issues are identified and handled in a manner allowing to guarantee the proper assurance assessment to ARTS.

5.2.4 Assurance continuity

Security assurance can only validate what has been studied. That's why when version X of a product is certified, the certificate does not apply to version X+1, X', X next gen, etc; or whatever other version as long as at least one byte of code has change.
One of the well-known challenges of any security assurance scheme is to validate the next version from the one evaluated. This cannot be done blindly, but also it should not take to re-run the complete evaluation process neither, as if no previous work had been done.
Assurance continuity is all about efficient evaluation results reuse. This is dependent of the actors involved in each evaluation and the capacity to exchange or being able to access previous evaluation data.
For that reason, we recommend allowing full reuse of previous evaluation results. When evaluation is performed by the same evaluators this should be straight forward, but when a different evaluator must be involved the necessary Non-Disclosure Agreement shall be defined.
We cannot yet define how exactly this should be done. But the scheme shall allow the evaluator sponsor to ask for assurance continuity, when a new version of an already evaluated product is to be certified. In that case, a revaluation assessment has to be performed between the evaluator, the developer and the scheme owner or its delegated entity. Procut modifications shall be presented by the developer to the evaluator and an impact assessment shall be performed on the previous evaluation results to identify necessary and only necessary revaluation task.

5.2.5 Scheme implementation

Assurance frameworks are challenging to run, and for developers or entities to be audited successfully pass their evaluation. It is even more difficult when the associated services and systems are as innovative as ARTS.
It can easily be foreseen that the first evaluation will fail, and many issues will be raised on both sides evaluators and evaluated targets (based on empirical observation of other assurance framework). It cannot be expected that the first ARTS will succeed their evaluation immediately. Several assurance requirements phases should be instantiated before full assurance requirement as defined in this document. We recommend a three-step approach as for current EU C-ITS deployment, defined by the security, certificate policy and CPOC protocol defined. We adapted them here to our needs:

- **Phase I - Deployment and testing phase**, where the entire framework will be refined and executed based on the current proposition and experimental execution until the scheme owner can assess that the framework and ARTS participating to the trials are both mature enough to perform a real assessment. All parts of the framework shall be executed at least once on an experimental ARTS.
- **Phase II - Trial phase**, where the process is run completely on real ARTS candidates to certification. In this trial phase, ARTS will be allowed to provide services under limited constraints to limit risks (limited parkours, low speed, etc.). During this trial phase if evaluation fails ARTS owner must define remediation plans to be able to keep providing their service and go to full deployment phase. The scheme owner together with the automotive actors shall decide when sufficient maturity has been observed on both the evaluation framework and the evaluated ARTS to go to the next step where

full assurance will be mandatory. If no such maturity is observed either the scheme owner can decide to stop current services and restart deployment and testing phase with new objectives or rerun the assurance trial process to get the sufficient maturity. If sufficient maturity is observed for all actors and systems, then the framework should move to the next step.

- **Phase III - Full deployment**, where all identified corrective actions are performed and running ARTS are fully certified.

6 Annex: Acronyms

Abbreviations	Meaning
AA	Authorization Authority (synonym to PCA)
AP	Access Point
AQO	Approved Qualified Organism
AT	Authorization Ticket (synonym to PC)
AV	Autonomous Vehicle
CAM	Co-operative Awareness Message
CC	ISO 15408 Common Criteria
CSMS	Cooperative-ITS Security Management System
C-ITS	Cooperative ITS
CP	Certificate Policy
CPOC	C-ITS Point of Contact
COTS	Commercial Of The Shelves
CRL	Certificate Revocation List
DC	Distribution Centre
DENM	Decentralized Environmental Notification Message
EA	Enrolment Authority (similar to LTCA)
EC	Enrolment Credential (similar to LTC)
ECU	Electronic Control Unit
ETSI	European Telecommunications Standards Institute
GNSS	Global Navigation Satellite System
HSM	Hardware Security Module
IEEE	Institute of Electrical and Electronics Engineers
ISMS	Information Security Management System
IT	Information Technology
ITS	Intelligent Transport System
ITS-S	ITS-Station
LTC	Long Term Certificate (similar to EC)
LTCA	Long Term Certificate Authority (similar to EA)
MA	Misbehaviour Authority
OBU	On-Board Unit
OSP	Organisational Security Policy
PC	Pseudonym Certificate (similar to AT)
PCA	Pseudonym Certificate Authority (similar to AA)
PKI	Public Key Infrastructure
RCA	Root Certificate Authority
RSU	Road Side Unit
SAM	Service Announcement Message
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SFP	Security Function Policy
SOC	Security Operational Center
SPD	Security Problem Definition
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality

TLM	Trust List Manager
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
V2X	V2I or V2V

7 Bibliography

[1] c. a. p. p. ISO/IEC JTC 1/SC 27 Information security, “ISO/IEC 15408:2022 Evaluation criteria for IT security — Part 1: Introduction and general model”.

[2] F. d. n. 2002-535, « Décret modifié relatif à l'évaluation et à la certification de la sécurité offerte par les produits et les systèmes des technologies de l'information », April the 18th 2002..

[3] J3016 - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles, SAE international, 2021.

[4] PRISSMA, [L5.2] REPPORT ON CYBER-THREAT ANALYSIS IN AUTONO-MOUS VEHICULAR ECOSYSTEMS, 2022.

[5] STRMTG, Guide d’application relatif à la cybersécurité pour les STRA, Version 1 du 19 décembre 2022.

[6] ETSI, “TS 102 940 - Intelligent Transport Systems (ITS); Security; ITS communications security architecture and security management,” 2016.

[7] ETSI, “TS 102 941 Intelligent Transport Systems (ITS); Security; Trust and Privacy Management”.

[8] IETF, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, RFC 52801.

[9] ISO/IEC, 27000:2018 Information technology — Security techniques — Information security management systems — Overview and vocabulary.

[10] c. a. p. p. —. I. s. m. s. —. R. ISO/IEC 27001:2022 Information security.

[11] ISO/IEC, 27002:2013 Information technology — Security techniques — Code of practice for information security controls.

[12] ISO/SAE, 21434:2021 Road vehicles — Cybersecurity engineering.

[13] ETSI, *TR 103 460 - Intelligent Transport Systems (ITS); Security; Pre-standardisation study on Misbehavior Detection; Release 2.*

[14] C. 2. C. C. Consortium, White Paper on Misbehaviour Detection and Reporting to Misbehaviour Authority, https://www.car-2-car.org/fileadmin/documents/General_Documents/C2CCC_WP_2092_MisbehaviourDetection_and_Reporting_V1.0.pdf (last accessed 02/12/2022), 2021-12-17.

[15] ANSSI, Guide De Sélection D'Algorithmes Cryptographiques, ANSSI-PA-079, 8/3/2021.

[16] ANSSI, Recommandations de sécurité relatives à TLS, N°SDE-NT-35/ANSSI/SDE/NP, 23/05/2022.

[17] I. Freiling, Dependability Metrics, vol. 4909: Springer, 2008.

[18] B. B. K. M. G. T. W. Nadya Bartol, Measuring Cyber Security and Information Assurance, Information Assurance Technology Analysis Center (IATAC) , May 8, 2009.

[L5.3] Cybersecurity Objectives for Autonomous Vehicle Systems

[19] L. r. Stéphane Paul, “Over 20 years of research into cybersecurity and safety engineering: a short bibliography,” WIT Transactions on the Built Environment Vol 151., May 2015.

[20] B. K. L. O. N. P. K. Y. G. K. W. Ankur Shukla, “System Security Assurance: A Systematic Literature Review,” arXiv:2110.01904, 2021.

[21] J.-P. S. R. S. Mazen Mohamad, “Security assurance cases—state of the art of an emerging approach,” Empirical Software Engineering, 2021.

[22] U. D. o. defense, *Trusted Computer System Evaluation Criteria (TCSEC), DoD 5200.28-STD,* l985.

[23] SOG-IS, *ITSEC: Information Technology Security Evaluation Criteria.*

[24] ANSSI, *First Level Security Certification For Information,* April 7th, 2014, 2014.

[25] https://www.ssi.gouv.fr/entreprise/qualifications/prestataires-de-services-de-confiance-qualifies/organismes-habilites-a-proceder-a-la-qualification/.

[26] STRMTG, Guide d’application relatif à la mission de l’organisme qualifié agréé pour l’évaluation de la sécurité et pour l’audit de sécurité en exploitation des STRA, https://www.strmtg.developpement-durable.gouv.fr/guide-d-application-relatif-a-la-mission-de-l-a800.html, 20/12/2022.

[27] MITRE, CVE, https://cve.mitre.org.

[28] B. B. A. L. G. G. M. Arnaud, Investigating Process Algebra Models to Represent Structured Requirements for Time-sensitive, CPS. SEKE, 2021.

[29] A. L. P. T. G. G. B. Bannour, Demonstrating The MAAT REQ Tool: Using Algebraic Process Models To Support Time-Sensitive Requirements Design., RTSS@Work , 2022.

[30] J. P. E. C. G. P. L. G. B. Bannour, Off-Line Test Case Generation for Timed Symbolic Model-Based Conformance Testing., ICTSS , 2012.

[31] J. N. P. C. B. Bannour, Symbolic Model-based Design and Generation of Logical Scenarios for Autonomous Vehicles Validation., IV, 2021.

[32] B. B. C. G. A. L. P. L. G. E. Mahe, A small-step approach to multi-trace checking against interactions., SAC , 2021.

[33] B. B. C. G. A. L. P. L. G. E. Mahe, Interaction-based Offline Runtime Verification of Distributed Systems., FSEN , 2023.

[34] S. F. G. G. a. E. W. Andreas Ekclhart, “Ontological mapping of common criteria’s security assurance requirements.,” In IFIP International Information Security Conference. Springer, 85–95., 2007.

[35] w. https://en.wikipedia.org/wiki/ISO/IEC_27002.

[36] ANSSI, Référentiel Général de Sécurité - Annexe B1 - Mécanismes cryptographiques - Règles et recommandations concernant le choix et le dimensionnement des mécanismes cryptographiques, 2014: version 2.0.