A semantic-based Artificial Intelligence (AI) reasoning tool to analyse the link between cyber security and safety for Internet of Vehicle (IoV) and Autonomous Vehicles (AVs)

Cappelli, Maria Assunta

Université de Genève
Centre Universitaire d'Informatique

Mémoire de Master en Systèmes et Services Numériques:

A semantic-based Artificial Intelligence (AI) reasoning tool to analyse the link between cyber security and safety for Internet of Vehicle (IoV) and Autonomous Vehicles (AVs)

Author:
Maria Assunta Cappelli

Supervisor:
Prof. Giovanna Di Marzo Serugendo

Jury :
Prof. Anne-Francoise Cutting Decelle
Dr. Martin Strohmeier
Dr. Ashley Caselli

February, 2022

Acknowledgement

I would like to thank Prof. Mrs Giovanna Di Marzo Serugendo for giving me the opportunity to develop an innovative topic characterised by important and (still unresolved) challenges. I would also like to thank Giovanna for trusting my ideas and for her significant contribution to the development of my research curiosity. Finally, I thank her for the availability she demonstrates in frequent meetings, where she has proven to be advanced in her intelligence "to look beyond".

Next, I would like to send my greetings to Mrs Anne-Francoise Cutting Decelle and Mr Martin Strohmeier and thank them for trusting this research project to support my work.

Abstract

Current technological developments have led to a great deal of embedded sensors, connected objects, and their related networks and communication to be present in the transport area involving, Autonomous Vehicles (AVs), aircraft, trains, as well as road infrastructures. Various types of signals and connections occurring on the Internet of Vehicle (IoV) are vulnerable to security attacks, which can cause the system to fail with serious consequences on the user's safety. Research on IoV security focuses on securing communication between nodes. Only a few studies have investigated the relationships between security and safety in IoV. Our approach addresses this gap by providing semantic-based analysis to jointly explore safety and security. We propose a semantic-based Artificial Intelligence (AI) reasoning tool to analyse the causal relationships between cyber security and safety for IoV and AVs. This tool runs on the ontology, named Security-Safety Internet of Vehicles (SSIoV) ontology, which represents both security-safety knowledge about IoV domain. Our goal is to perform reasoning and inferences on security vulnerabilities and their impact on safety risks, based on actual data extracted from real-world scenarios. This research falls in the research areas of cyber-security, because: (a) it involves securing current and future vehicles and charging infrastructures; (b) it uses a semantic AI approach for enhancing cyber-defence; (c) it detects IoV and AV components, vulnerabilities, and risks. Therefore, this tool is also useful to improve preventive cyber-defence capabilities in the IoV and AVs area. Finally, this study contributes to enhance the safety of Switzerland's IoV-critical road infrastructure.

Keywords: Internet of Vehicle (IoV); Autonomous Vehicles (AVs); Semantic approach, Ontology, Cyber Security, Security vulnerability, Safety risk; AI reasoning tool; Security-Safety Internet of Vehicles (SSIoV) ontology; Semantic Rules; Semantic Causal Relationships

Summary

List of plots

Tables

Acronyms

2G Second-generation cellular network. 11

3G Third-generation cellular network. 11, 59, 137

3L Third Level. 8

4G/LTE Forth-generation cellular network. 11, 13, 59, 137

4L Forth Level. 8

5L Fifth Level. 8

5g Fifth-generation of mobile telecommunications technology. 137

ABS Anti Lock Braking System. 10

ACEA European Automobile Manufacturers' Association. 21

ADAS Advanced Driver Assistance Systems. 10, 25

AI Artificial Intelligence. 1–9, 37, 40, 41, 43, 44, 97, 98, 107–109, 111, 112, 116–118, 121, 122

APCN Novel Framework with Preservation and Repudiation for . 19, 20

ASC Smart Card Protocol. 19

ASR Anti-Slip Regulation. 10

Auto-ISAC US Automotive Information Sharing and Analysis Center. 21

AV Autonomous Vehicle. 2, 3, 5–8, 12, 21, 23, 25, 34–36, 41, 44, 46, 47, 49, 50, 52, 63, 65, 66, 72–74, 76–80, 109, 110, 116–119, 121, 122, 139

AVs Autonomous Vehicles. 1–3, 5–10, 12, 15, 21–23, 33, 36, 40–44, 47, 49, 57, 59, 60, 67, 70, 73, 75–79, 89, 111, 117, 121, 122, 124

BSM Basic Safety Messages. 14, 80, 87

C2NET Cloud Collaborative Manufacturing Networks. 28, 33

CALM Continuous Air interface for Long and Medium distance. 11, 13, 137

CAN Controller Area Network. 71, 75–80, 89, 136

CAS Cooperative authentication scheme. 18–20

CC Cloud Computing. 8, 9

CoP Code of Practice. 22

CTDO) Connected Traffic Data Ontology. 38

DARC Data Radio Channel. 9

DIKE Dynamic Privacy-Preserving Key Management Scheme. 18, 20

DMR Driving Monitoring Record. 19

DOAM Description of a Model. 24

DSRC Dedicated Short-Range Communications. 11, 13, 49, 87, 125, 137

EBD Electronic Brake-force Distribution. 10

ECU Electronic control unit. 9, 71, 77, 80

EGR Exhaust gas recirculation. 129

ENISA European Network and Information Security Agency. 21

ESC Electronic Stability Control. 10

EV Electric Vehicle. 9, 139

EVAP Evaporative Emission Control System. 125, 129, 130

FM Fact Model. 9

FMEA Failure Modes and Effects Analysis. 73

FTA Fault Tree Analysis. 73

GNNS Global Navigation Satellite System,. 9, 58, 111, 143

GPRS General Packet Radio Services. 137

GPS Global Positioning System. 4, 6, 17, 25, 48, 57–59, 64–66, 69, 72, 76, 77, 79, 80, 87–89, 111, 112

GSM Global System for Mobile Communications. 28, 137

HARA Hazard Analysis and Risk Assessments. 41, 70, 73

HAZOP Hazard and operability study. 73

IEC International Electrotechnical Commission. 16

IET Institute of Engineering and Technology. 22

IoT Internet of Things. 3–5, 9, 23, 26–30, 33–36, 38, 40–42, 46, 47, 49, 52, 124, 125

IoV Internet of Vehicle. 1–6, 9, 12–15, 20, 34–36, 38, 40–42, 44–49, 52, 121, 122, 124

ISM Ignition Switch Module. 9

ISO International Standards Organization. 16, 22

IT Intelligent Transportation. 2, 6, 9, 139

ITS Intelligent Transportation System. 13

ITSs Intelligent Transportation Systems. 6

IVI In vehicle Infotainment. 139

KUD Key Update Delay. 18

LBSs Location-based services. 18

LIDAR Laser Detection and Ranging. 9, 72, 76, 79, 139

LIN Local Interconnect Network. 136

LTE Long Term Evolution. 11, 13, 28, 137

LuST Luxembourg SUMO Traffic. 87

M2M Machine-to-Machine. 26, 28, 30

MANET Mobile ad hoc Network. 10

ML Machine Learning. 8, 9

MOST Media Oriented Systems Transport. 136

NFC Near Field Communication. 124, 137

NHTSA National Highway Traffic Safety Administration. 21

OBD On-board diagnostics. 9, 72, 77, 139

OBDA Ontology-based Data Access. 43

OBUs On-board units. 11, 49, 71, 72, 75–80, 124

OEM Original Equipment Manufacturer. 142

OWL Web Ontology Language. 5, 6, 8, 24, 26, 28, 41, 53, 57, 66, 83, 85, 97, 146, 147

PAN Personal Area Network. 9, 127, 132

PCT Protocols and Communication Technologies. 136, 137

PKC Public-Key Cryptography. 20

PPA Privacy-Preserving Authentication. 18

R2R Road-side Unit to Road-side Unit. 10

RDF Resource Description Framework. 24

RDS Radio Data System. 9

RFID Radio Frequency Identification. 124

RSUs Road Side Units. 10, 11, 18, 49, 71, 75–80, 124

RuleML Rule Markup Language. 66

SAE Automotive Engineers Society. 7–9

SAREF Smart Applications REFerence. 23, 24

SDTR Safety Data Transmission Rate. 13

SINR Signal-to-inerence-plus-noise-ratio. 14

SM/DS Safety Messages or Data Size. 13, 14

SOSA Sensor, Observation, Sample, and Actuator. 23, 38, 49

SPARQL SPARQL Protocol and RDF Query Language. 25

SQuaRE Software Engineering Standard. 29

SQWRL Semantic Query-Enhanced Web Rule Language. 7, 97, 99, 101, 103, 105

SSIoV Security-Safety Internet of Vehicles. 2, 4–9, 36, 41, 45–48, 50–59, 61–63, 67, 70, 82–97, 109, 110, 122, 123, 146, 147

STAMP Systems-Theoretic Accident Model and Processes. 4, 5, 10, 31, 38, 42, 47, 50, 52, 136–142

STPA Systems-Theoretic Process Analysis. 35, 41, 42

STPA-SafeSec Systems Theoretic Process Approach-Safety and Security. 42, 70

STPA-Sec Systems-Theoretic Accident Model and Processes for Security process. 31–33, 35, 42, 50

SWRL Semantic Web Rule Language. 6, 7, 9, 25, 29, 33, 66, 67, 69, 70, 73, 75, 77, 78, 81, 84–86, 88, 97, 109, 122

TA Trusted Authority. 18

TARA Threat Analysis and Risk Assessment. 41, 70

UNECE United Nations Economic Commission for Europe. 21

UNSPSC United Nations Standard Products and Services Code. 45

UTMS Universal Mobile Telecommunications System. 28

V2H Vehicle-to-Human. 2

V2I Vehicle-to-Infrastructure. 2, 49, 136–138

V2P Vehicle-to-Person. 49, 136

V2R Vehicle-to-Road. 2, 10, 11, 136

V2S Vehicle-to-Sensor. 2, 49, 137

V2V Vehicle-to-Vehicle. 2, 9–11, 14, 18, 49, 59, 136–138

V2X Vehicle-to-everything. 9, 136–139

VANET Vehicular ad hoc network. 2, 3, 5, 6, 9–13, 15–18, 20, 21, 34–36, 43

VeReMi Vehicular Reference Misbehavior. 42, 87, 88

VICS Vehicle Information and Communication Systems. 8–10

VSS Vehicle Signal Specification. 23, 38, 49

VSSO Vehicle Signal Specification Ontology. 4, 5, 9, 10, 23, 27, 38, 47, 49, 50, 52, 125–135

W3C World Wide Web Consortium. 24

WAVE Wireless Access in the Vehicular Environment. 11, 13, 137

Wi-Fi GSM UTMS. 28, 137, 138

WIMA Worldwide Interoperability for Microwave Access. 137

WiMAX Worldwide Interoperability for Microwave Access. 11, 13, 138

Chapter 1

Introduction

In the next few years, millions of Autonomous Vehicles (AVs) and Intelligent Transportation (IT) will be on the road. This event is the expectation of the automakers, which are working to develop high-performance AVs with the partnership of hi-tech start-ups. Driverless cars also attract the attention of automotive safety institutions because the great promise of AVs is the decrease in the percentage of accidents on the road. Human error is one of the leading causes of crashes. The use of self-driving cars should decrease or unset human error providing a reduction of road accident victims. However, safety is not the only reason for the development of AVs. There are many other reasons to support the development of self-driving cars. The AVs can have a positive impact on the efficiency and comfort of the driving experience. Additionally, the electrification of AVs has led to a focus on the environmental benefits of electric AVs.

The long-awaited AVs have a large number of embedded sensors. They are connected to other objects within the ad-hoc networks called Vehicular ad hoc network (VANET).

VANET technology is one of the fascinating applications of the principles of wireless communication, where AVs and roadside units communicate with each other objects. The connected environment is based on the IoV paradigm where multiple communication channels are implemented. Various types of signals and connections appear in the IoV, such as Vehicle-to-Vehicle (V2V); Vehicle-to-Road (V2R); Vehicle-to-Infrastructure (V2I); Vehicle-to-Human (V2H); Vehicle-to-Sensor (V2S) [2].

AVs are poised to improve current mobility and general quality of life. However, its potential social benefits cannot be exploited right now because the security degree of the information exchange in VANET is still not satisfactory. The network is vulnerable to hackers' attacks, which can deflect the information as well as manipulate the nodes. Cyber security attacks can cause catastrophic consequences in terms of safety (e.g. spoofing, tampering with electric signals, etc.), which can impact the human life of the "road users".

Test of AV prototypes confirmed that AVs cannot meet certain security standards. There is currently no affordable architecture to prevent malicious parties from accessing the vehicle network. VANET cannot reach a higher level of confidentiality, integrity, authenticity, availability, and non-repudiation. Even though some researchers have proposed many techniques to overcome the lack of security in VANET, their studies are limited to dealing with some security requirements.

These observations lead us to shift our attention from security concerns to both security and safety issues. If we only analyse security in VANET is not enough to achieve a higher level of security. We can reduce security concerns if we relate security to safety based on causality. The definition of these security issues ensures to focus on relevant safety issues that have an impact on safety.

To achieve our objectives, we use the semantic approach. The approach allows fine grained formalisation of the intersection between safety and security and sets up the relationships between these two domains. There are multiple ontologies for safety and/or security of the IoT, AVs cyber security, etc. However, only a few of them address both safety and security for IoV and AVs.

The master's thesis shows two parties. The first one is about the state-of-the-art (I), where we explain the current studies on the semantic approach of self-driving cars. We focus on research that deal with the safety and/or security issues of AVs. The second part (II) explains our research approach to develop an AI semantic tool that can analyse the causal relationships between cyber security and safety for IoV and AVs.

Part I

State-of-the-art

The next part is about the state-of-the-art, which we organise as follows.

Chapter 2 deals with the components of the AVs domain. This chapter aims to make an exhaustive technical framework. First, we discuss of AV (2.1). Second, we deal with the VANET (2.2), and finally we deal with the IoV (2.3).

Chapter 3 deals with the security information of both AVs, and VANET. We explain the requirements to evaluate the secure information system (3.1). We study the risk for threats and attacks in VANET (3.1.1). Then, we examine the conventional methods for securing the information (3.1.2). Traditional methods show their weaknesses as they focus on certain aspects of security. There is no way to ensure minimum requirements for secure information. The absence of this minimum-security degree leads us to look for other techniques. Furthermore, we focus on the limitations of the conventional methods regarding security in VANET (3.1.2.1). Then, we point out one of the issues with VANET's lack of higher-level of security, such as the lack of the standard for the AV domain regarding AV security and AV safety (3.2).

Chapter 4 analyses the studies on security and safety for AVs. Section (4.1) examines the existing and available ontologies about the risk, safety, and security of a generic system. Section (4.2) examines the semantic research about risk (4.2.1); safety (4.2.2); cyber security (4.2.3); transportation (4.2.4); IoT (4.2.5), and joint safety-security (4.2.6).

Finally, chapter 4.3 aims to analyse and summarise the state-of-the-art on security and safety ontology framework. Our analysis provides an overview of ontology and recent semantic research. We note the semantic approach are still an unexplored technique, especially for the AV domain.

Chapter 2

General Background on AVs domain

The AV domain encompasses three main components, such as AV, VANET and IoV.

2.1 Autonomous Vehicle (AV)

Over the last few years, We have been observing the development of AVs on which car manufacturers are investing. The main reason underlying the AVs development is road safety. In fact, the crash risks considerably should decrease with AV that should always be on alert, or ready to respect the rules of the road. However, many governments are concerned about this phenomenon due to the lack of a regulatory framework.

AVs are IT applications. These systems use automation, computer science, and communication technologies to improve the efficiency and safety of cars. ITSs applies information and communication technologies in the field of road transport. It also involves infrastructures, vehicles and users, traffic, and mobility management.

The AV assists the driver in several ways. This system can alert the driver when hazards occur; or for drawing the driver's attention to operate correct manoeuvres; or for replacing the driver partially. This assistance is intended to be more effective when the AV will be able to replace the driver. The AV will become the real driver by achieving full autonomy.

AVs are different from conventional vehicles. Driving systems make decisions about the guide-way. They evaluate other driving, traffic signs, pedestrians' behaviour, and viability. Also, they determine driving manoeuvres, and speeds and are responsible for alerting other driving systems or pedestrians. The importance of the driver role is inversely proportional to the AV's autonomy.

Also, AV is dynamic in that it can interact with the environment. This dynamism enables it to be involved in a network. The involvement means they have to adapt their manoeuvres by learning. The AV learn from the data. They collect every data available on the network. These data include everything, from the information about the vehicle (velocity, pneumatic conditions, position, driver data), to data of other vehicles (velocity, pneumatic conditions, position, driver data), to the environmental conditions (road traffic, weather).

The AV processes the data quickly, understanding if the data is useful or unnecessary. Since its program is able to collect data under defined codes, it can remove contradictory data. The planning is dynamic because it adapts itself to any unforeseeable situations, and it is responsive to change its plan in a few seconds.

The complexity of the planning depends on the AV's intelligence. The AV with a lower intelligence degree follow a pre-programmed plan. In contrast, a higher intelligence level allows AV to be more autonomous to enable it to carry out a progressive plan while driving. The level of intelligence makes the difference on the AV's abilities.

The Automotive Engineers Society (SAE) provided a common taxonomy that shows different degrees of the AVs' intelligence [7]. In Table 2.1 we reproduce the AVs classification proposed by the SAE according to the J3016 "Levels of Driving Automation" standard[1].

[1] See https://www.sae.org/news/2019/01/sae-updates-j3016-automated-driving-graphic.com

Level	**Description**
Level 0	Zero automation
Level 1	Driver assistant
Level 2	Partially autonomous driving
Level 3	Semi-autonomous driving
Level 4	Highly autonomous driving
Level 5	Fully autonomous driving

Table 2.1: The taxonomy of the driving automation issued by SAE

We are interested in 3L, 4L, and 5L. A 3L AV takes the driving in a defined situation evaluating other driving behaviour, traffic signs, pedestrians' behaviour, and viability. Also, it determines driving movement, velocity and it is responsible to alert other driving systems or pedestrians when a safety risk occurs. Therefore, the role of the driver changes with vehicle performance. This change is directly proportionate to the AV's autonomy.

The 4L AV can perform all driving functions under certain conditions, such as the type of road. Human driver intervention is required for all situations that come out of the defined conditions. The difference compared to 3L is the need of human intervention only for undefined conditions. When there is a particular road, human intervention is not demanded. Otherwise, the absence of a specific road implies a human driving intervention. The 4L AV is fully autonomous, even though the driver's control remains on the driver.

The 5L AV is capable of performing all driving functions under all conditions. The human driver intervention is not provided. This level does not come with a steering wheel or accelerator or brake pedals.

2.1.1 Technologies for AVs

The difference among these levels depends on the technological equipment used for AVs. The combined application of advanced technologies increases the AVs' intelligence. These technologies are heterogeneous, such as sensors and actuators; AI, Cloud Computing (CC); Machine Learning (ML), and Vehicle Information and Communication Systems (VICS).

The sensors are designed to monitor the area around AVs by detecting objects, positions, distance from other objects, vehicles, and pedestrians. The actuators are: brakes, engine, lights, speed, steering wheel; etc. that ensures AVs to act.

The AVs are equipped with several sensors and actuators, as we note from the list below[2]:

- ⋄ **Standard sensor**: (a) powertrain sensor; (b) tire pressure sensor; airbag impact sensor; (c) Global Navigation Satellite System, (GNNS) sensor; speed sensor; etc.
- ⋄ **Sensor for AVs**: (a) Laser Detection and Ranging (LIDAR); (b) laser; (c) cameras; (d) radars; (e) ultrasonic sensors
- ⋄ **Actuators**: (a) engine control; (b) suspension; (c) transmission; (d) brake system; (e) steering; etc.

AVs make decisions through the algorithms. The AI's algorithms allow ECU to process and make decisions to perform intelligent tasks. The information they process includes: sensor data; map data; keys and certificates; V2X information; devices information. Also, ML algorithms let AVs to predict events.

The CC enables to share sets of resources. These sets are database servers; map servers and 3rd party service providers' servers.

VICS are a system capable of receiving real-time traffic information on congestion and regulation. The on board VICS interface is a monitor, where the driver can see road and traffic information. Data transmission is possible thanks to Infrared, Microwaves in the Ignition Switch Module (ISM) band and Fact Model (FM), such as Radio Data System (RDS) or Data Radio Channel (DARC). Infrared is considered a Personal Area Network (PAN) technology that transfers data at a lower rate than Bluetooth. However, it has some advantages, such as its large bandwidth that enables high network traffic in V2V applications. However, their use is limited to very short distances since infrared signals are strongly affected by obstacles [9]. We can consider the two following VICS groups :

- ⋄ **Inside Vehicle Communication Components**, which includes: (a) telematics box; (b) vehicle IT station; (c) in-vehicle Gateway; (d) in-vehicle infotainment; (e) On-board diagnostics (OBD)-II port; (f) EV charging connector

[2] This distinction has been shown by ENISA [8].

- **Nearby external component** that includes: (a) Road Side Units (RSUs); (b) traffic signs and systems

Again, AVs have some Advanced Driver Assistance Systems (ADAS) that allow them to control how they perform on the road. The ADAS ensures the navigation, stabilisation, and the manoeuvring of the vehicle. These are the:

- Anti Lock Braking System (ABS);
- Electronic Brake-force Distribution (EBD);
- Anti-Slip Regulation (ASR);
- Electronic Stability Control (ESC).

ADAS influences the driver indirectly. They do not act on the vehicle, but assist him. The driving system interprets the recommendations made by ADAS, and considers other information (e.g. weather conditions, viability and traffic) that may affect the validity of data suggested by ADAS. If there is a discrepancy between the data, the driver should take control and intervene to ensure the proper use of the vehicle.

The difference between VICS and ADAS is that the latter can intervene in driving while VICS are mostly information systems. However, both these systems affect drivers increasing or decreasing safety.

2.2 Vehicular ad hoc network (VANET)

the VANET is one of the fascinating applications of the wireless communication principle. This technology applies the Mobile ad hoc Network (MANET) architecture. Zeadally et al. [10] note that VANET has a great potential with regard to road safety, traffic efficiency, convenience as well as comfort for both drivers and passengers.

The VANET is the outcome of communications among everything that is part of it through multiple communication channels. The V2V, the V2R, and the Road-side Unit to Road-side Unit (R2R) are communication systems that ensure the information exchange [11] thanks to being connected to the Internet. The systems process a massive amount of data, which is converted into message content and broadcast on the network. This information may include traffic data, location data, or alarm situations.

The V2V lets to send safety messages among vehicles to avoid collisions through the alarm systems that notify dangerous situations. The V2V system can communicate with other vehicles indirectly. Hence, the data arrive at the destination after having crossed the road-site units. The communication also can be direct via Wireless Access in the Vehicular Environment (WAVE) for high-speed data transmission, or Continuous Air interface for Long and Medium distance (CALM) communication standard. In the first case, the communication involves many hops, while the communication involves a single hop in the second case [11].

The V2R connects vehicles with buildings, traffic lights, infrastructures composed by *"several base stations that give signals over a long-range, such as cellular networks that are designed for voice data exchange or Worldwide Interoperability for Microwave Access (WiMAX) that can provide wireless data (e.g., high-speed Internet) for mobile users"* [12]. Cellular networks can provide different vehicular communications based on radio waves over long distances [9] and at high mobile speeds. It includes *"different cellular services such as 2G, 3G, and 4G/LTE/LTE technologies that differ in their bandwidth, latency, and data transfer rate"* [2].

In VANET communications take place on the basis of two tools: transponder On-board units (OBUs) and RSUs. The first one is the radios in the vehicle; they ensure to communicate with other vehicles. The RSUs are fixed units on the road that permit the communication with the infrastructure. The tools contain devices to operate on Dedicated Short-Range Communications (DSRC) [13].

VANET is a complex network in three ways. First, the structure of the network is heterogeneous due to channels of communication that provide a multi-layer architecture. Each channel is a set that includes many subsets of communication channels among different parties. These channels run in parallel or intersect each other and many nodes exchange data. Therefore, the structure is decentralised, mobile, open, and dynamic. Based on this feature, it is difficult to guarantee the security of the network. If something happens, it is more complex finding the problem source as well as understand the extent of an attack's event or the quality of the nodes. These events can have an impact on security issues.

Second, the complexity of the VANET is due to the environment where vehicles drive. Hezam et al. [14] suppose the environment is different for each road type. If the vehicle is driving on a city road, the number of obstacles will be more than obstacles on a highway, which is more organised. These differences lead nodes to follow different movements, because the environment has a great influence on the network.

Finally, the other complex aspect is the interconnection between the multi-layered structure and the environment, involving the enhancement of the ability of AVs in VANET.

AVs with higher autonomy level may face these complexities involving self-learning and improving their capabilities. The more diverse the environment, the more opportunities to learn AVs have.

We note that VANET is a network with communication channels of the AVs as a subset of the network. This configuration can cause interference. The security information of a network's subset can be affected by every crisis arising from other network's subsets. The vulnerability of the VANET is one of the obstacles to the development of AVs. The AVs' potential benefits cannot be exploited right now because of the issues related to the security of the information exchange as confirmed by the tests on the AV prototypes. The prototypes do not live up to certain security standards.

There is currently no cost-effective architecture to prevent malicious parties from accessing the in-vehicle network. The hacker attacks can deflect both the content of the information as well as to manipulate the nodes into the IoV platform.

2.3 Internet of Vehicle (IoV)

VANET is evolving into the IoV paradigm by to new technologies in cloud computing that improves the capabilities of sensors and wireless communication.

Gasmi and Aliouat [2] explain the difference between VANET and IoV, identifying some main differences that we summarise in the Table 2.2:

	VANET	**IoV**
Architecture	(a) Vehicle-to-Vehicle (b) Vehicle-to-Road (c) Vehicle-to-Infrastructure	(a) Vehicle-to-Vehicle (b) Vehicle to Personal Devices (c) Vehicle to Human (d) Vehicle-to-Road (e) Vehicle-to-Infrastructure
Network Technologies	(a) WAVE (b) CALM	(a) WAVE (b) CALM (c) Bluetooth (d) ZigBe (e) 4G/LTE/LTE technology (f) WiMAX
Cloud platform		(a) Basic Cloud Services (b) Smart ITS Application Servers (c) Information Consumer and Producer
Network Layered Architecture	(a) Access layer (b) Network and transport layer (c) Security layer (d) Management layer (e) Application layer	(a) User Interaction Layer (b) Coordination Layer (c) Processing and Analysis Layer (d) Application Layer (e) Business Layer

Table 2.2: The main differences between VANET and IoV [2]

IoV has some safety applications that improve safety and reduce accident levels [15]. Some of the safety applications are the following:

- Advanced Driving Assistance;
- Collision Avoidance Applications;
- Emergency-Braking Application;
- Warning-on Application;
- Hazardous Location Notification Application;
- Lane-changing Assist;
- Left and Right Turn Assist;
- Hazardous Location Notification; etc. [15].

The safety applications must ensure efficient data transmission with high reliability. Azzahar et al. [15] identify the three factors, such as (a) DSRC, (b) Safety Data Transmission Rate (SDTR), (c) Safety Messages or Data Size (SM/DS) to ensure the best metrics.

DSRC is used as a wireless communication technology in vehicular networks. SDTR is used for safety requirements. Researchers prefer the lowest data transmission rate (6Mbps), because the Signal-to-inerence-plus-noise-ratio (SINR) threshold (dB) is required. SM/DS is broadcast in the V2V communication and it is known as Basic Safety Messages (BSM), which consists of two main types of messages, namely periodic messages, and event-driven messages [15].

Figure 2.1 represents the core technologies of IoV.

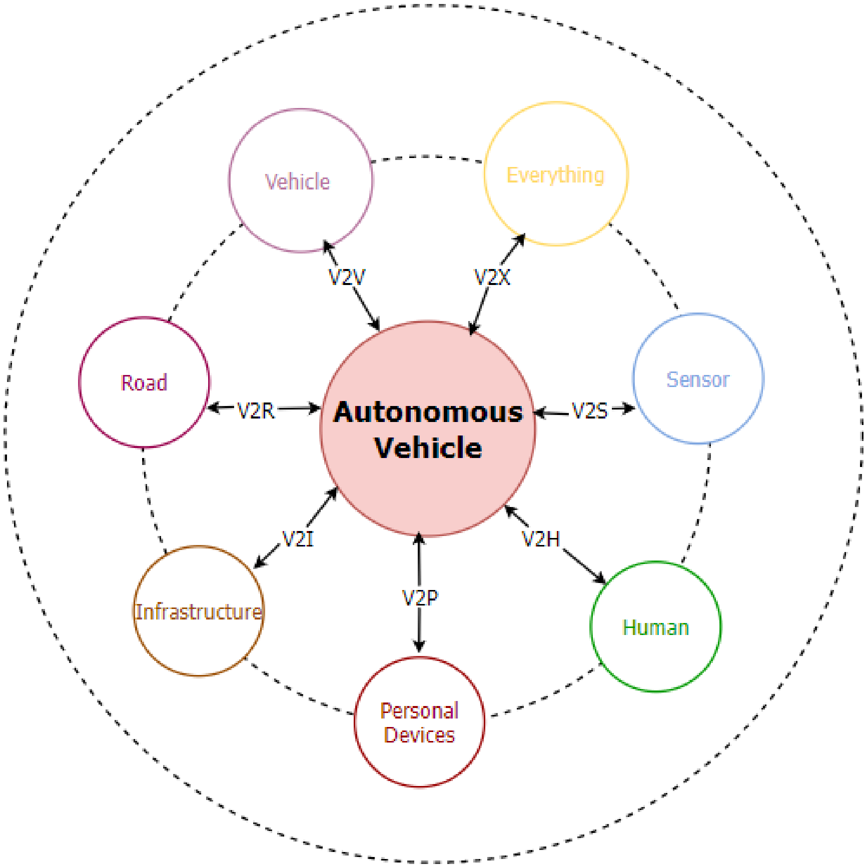


Figure 2.1: IoV representation

Chapter 3

The Security in VANET and IoV

Security in AVs represents one of the most challenging problems. Many fields are interested in developing methods to counter third-party attacks on AVs, in particular VANET and IoV.

3.1 Information Security in VANET and IoV

Information security in VANET and IoV is one of the most concerned issues because of the open nature of the network, in which information is disseminated. Vulnerabilities in VANET also affect AVs. The openness and the dynamism of VANET put it at risk from several threats, such as:

- Software Attack
- Theft Identity
- Information Theft
- Information Distortion[1]

There is a debate about the requirements for secure information. In general, the security information is based on the classic method that is based on the assumption that secure system should ensure information security under three components:

[1] See https://en.wikipedia.org/wiki/Information_security

- ◇ Confidentiality
- ◇ Integrity
- ◇ Availability

However, there are many proposals to increase the number of these criteria. The Parkerian hexad[2] adds three additional attributes (Authenticity, Possession or control, Utility) to the three classic security attributes of the CIA triad.

We consider three classic requirements, authenticity and the non-repudiation to evaluate the information security of the VANET. Hence, we have:

- ◇ Confidentiality
- ◇ Integrity
- ◇ Authenticity
- ◇ Availability
- ◇ Non-repudiation

The ISO standard defined each of these requirements. **Confidentiality** is *"the property that information is not made available or disclosed to unauthorised individuals, entities, or processes"* (ISO/IEC 27000)[3].

Integrity is the *"property of accuracy and completeness"* (ISO/IEC 27000). **Authenticity** is *"the property that an entity is what it claims to be"* (ISO/IEC 27000). **Availability** is the *"property of being accessible and usable on demand by an authorised entity"* (ISO/IEC 27000). **Non-repudiation** is the *"ability to prove the occurrence of a claimed event or action and its originating entities"* (ISO/IEC 27000)[4].

3.1.1 Threats and Attacks Risk in VANET

Before presenting the technical solution to enhance some aspects of the information security in VANET, we propose an overview on threats and attack risk for this network in Table 3.1.

[2] The Parkerian hexad is a set of six elements of information security proposed by Donn B. Parker in 1998

[3] See https://www.iso.org/obp/ui/#iso:std:iso-iec:27000:ed-5:v1:en

[4] See https://www.iso.org/obp/ui/#iso:std:iso-iec:27000:ed-5:v1:en

CONFIDENTIALITY	Eavesdropping Attack Traffic Analysis Attack Man-in-the-Middle Attack
INTEGRITY	Masquerading Attack Replay Attack Message Tampering Attack Illusion Attack
AUTHENTICITY	Sybil Attack Tunnelling Attack GPS Spoofing Node Impersonation Attack Free Riding Attack Replay Attack Key and/or Certificate Replication Attack Message tampering Masquerading Attack
AVAILABILITY	Denial-of-service attacks Jamming Attack Malware Attack Broadcast Tampering Attack Blackhole Attack Grayhole Attack Greedy Behaviour Attack Spamming Attack
NON–REPUDIATION	Repudiation Attack

Table 3.1: Threats and attacks in VANET

For each security criteria, we have specific attacks or threats that can violate them[5].

3.1.2 Conventional Security Methods in VANET

We analyse some of the techniques proposed against threats and attacks to the VANET. The goal is to understand whether the techniques can effectively

[5] Table 3.1 is not exhaustive. For more detail threats and attacks taxonomy issued by ENISA [8] that includes many other examples. Also, ENISA [8] provides some examples of security attack scenarios, by classifying the severity of potential attacks into three different levels: high, medium, and low (p.22).

ensure the integrity of one or more security requirements.

Many researchers have developed some techniques to ensure security in the VANET proposing relevant solution to strengthen information security standards. We present five of these studies, which aim to reach the authentication of the VANET. We analyse the following approaches with regard to technique, goal, description and outcomes.

Confidentiality

Technique: Dynamic Privacy-Preserving Key Management Scheme (DIKE) for Location-based services (LBSs) [16].
Goal: vehicle user's privacy and key update efficiency.
Description: Privacy-Preserving Authentication (PPA) mechanism uses a group signature for vehicle user's privacy preservation and for restricting the vehicle user's double registration. Then, PPA uses a forward-secrecy technique. The user can use it to autonomously update the new session key. That reduces the Key Update Delay (KUD) when the vehicle does not depart from the service session. Finally, *"DIKE provides a new cooperative key update alternative. It combines a dynamic threshold technique with the V2V communications"*.
Outcomes: (a) DIKE significantly reduces the KUD due to the user departure event; (b) the session key's forward secrecy and backward secrecy resist possible collusion from the departed vehicle users; (c) DIKE scheme can achieve much better efficiency about the average KUD and average KUD during each key update procedure.

Integrity

Technique: Cooperative authentication scheme [17].
Goal: increasing the authentication overhead on individual vehicles and decreasing the authentication delay.
Description: *"the Cooperative authentication scheme (CAS) maximally removes redundant authentication efforts on the same message by different vehicles"*. Then, *"the CAS uses an evidence-token approach to control the authentication workload"* and *"the CAS does not involve a Trusted Authority (TA)"*. Moreover, *"the vehicle, passing a RSUs, obtains an evidence token from the TA via the RSUs"*. Finally, *"the token reflects the contribution of the vehicle to cooperative authentication. It ensures that the vehicle can benefit from other vehicles' authentication efforts in the future"*.

Outcomes: (a) CAS reduces its own workload; (b) CAS allows saving the workload; (c) the CAS increases the ability of the vehicle to resist to free-riding attacks.

Authenticity

Technique: vehicle authentication and the validation of the exchanged messages [18].
Goal: vehicle authentication and the validation of the exchanged messages.
Description: a Smart Card Protocol (ASC) uses low-cost cryptography. Ying et al. [18] note that: *"ASC verifies the identity of each user having a smart card"*; *"ASC allows the anonymity thanks to a dynamically changing of the user identity at the access"*; *"ASC ensures a dynamically changing of the passwords without the intervention of a trusted authority"*. The authentication of the messages takes place with two chains of cryptography hashes.
Outcomes: (a) ASC is better than the other protocols ASC in terms of efficiency; (b) ASC leads to (b.1) higher computational costs and, (b.2) a strong difficult to detect the dangerous nodes because of the dynamical update both identity and passwords.

Availability

Technique: data replication method for data access applications [19].
Goal: effect decreasing the intermittent connectivity and improving data access performance in distributed systems.
Description: a data replication method for data access applications works as follows: (a) The vehicles are grouped into a platoon; (b) The vehicles contribute part of their buffers to replicate data for other in the same platoon; (c) The vehicles share data with others; (d) The vehicle can still access the data after it leaves.
Outcomes: (a) DMR provides high data availability; (b) Driving Monitoring Record (DMR) lets a low data access overhead; (c) DMR provides low false alarm rate.

Non-repudiation

Technique: the novel framework with preservation and repudiation [20]. **Goal**: authentication with privacy preservation and non-repudiation.
Description: Novel Framework with Preservation and Repudiation

for (APCN) introduces the PKC to the pseudonym generation. Then, APCN ensures legitimate third parties to achieve the non-repudiation of vehicles by obtaining vehicles' real IDs, Also, the self-generated PKC based pseudonyms are also used as identifiers instead of vehicle IDs for privacy-preserving authentication. Finally, the update of the pseudonyms depends on vehicular demands.
Outcomes: APCN is feasible and adequate to be used efficiently in the VANET environment.

3.1.2.1 The Limits of Conventional Methods relate to VANET Security

There are several studies about cyber security related to VANET (or IoV) which propose conventional methods - i.e., network segmentation and cryptography - for ensuring secure communication between nodes [21]. However, traditional methods that deal with security issues focus on individual parties by ignoring the security of the entire system [22].

Most of the research deals with the authentication steps, because they consider that the main security issue can be solved by improving the access to the platform. Many researchers have created a number of protocols that focus on enhancing access security by dynamically changing an account's identity and password. In this way, the computational consumption is higher. It requires relevant computing power, and it causes the opposite effect due to the constant change. Therefore, it is not easy to analyse the existence of malicious nodes whose activity cannot be tracked.

Hence, the above studies try to intervene on the authentication aspect of VANET dealing with some criteria.

The DIKE for location-based services comes with the vehicle user's privacy and key update efficiency[16]. The CAS increases the authentication overhead on individual vehicles and decreases the authentication delay [17]. Vehicle Authentication and Validation of Exchanged Messages concern vehicle authentication and validation of exchanged messages [18]. Data Replication methods for data access applications deal with the effect of reducing intermittent connectivity and improving data access performance in distributed systems [19]. The APCN deals with authentication with privacy preservation and non-repudiation [20].

These studies are valid for one or more security requirements, but they cannot meet all security standards.

To reach a higher security level of the VANET, we should focus on each safety requirement: confidentiality, integrity, authenticity, availability, and non-repudiation. The semantic approach can meet all safety requirements, because it is a holistic approach that includes security requirements as classes of the ontology. The classes are connected to make axioms, rules that enable to control each safety information requirement.

3.2 The lack of Standards for AV

A) **Soft Law for AVs Security**

The uncertainty about the security in VANET also depends on the lack of standards for this ad hoc network. The attention to this aspect arose from several attacks on AVs, which showed the awareness of the VANET under cyber security [23], [24].

There are a number of good practices and security measures that drive AV security for insiders. However, there is still no standard for AVs.

We list some soft law in the field of AV, which represent the policy adopted by the government in this domain.

- ◇ National Highway Traffic Safety Administration (NHTSA), from the U.S. government issued a document about the cyber security best practices for smart cars, in 2016 [25]
- ◇ US Automotive Information Sharing and Analysis Center (Auto-ISAC) issued the Automotive cyber security Best Practices, which provide guidance on the implementation of automotive cyber security principles [26];
- ◇ European Automobile Manufacturers' Association (ACEA) issued the Principles of Automobile cyber security [27]
- ◇ United Nations Economic Commission for Europe (UNECE) issued a proposal for a recommendation on cyber security. The proposal focuses on cyber threats and vulnerabilities against vehicles as well as measures to be considered to mitigate the identified threats [28]
- ◇ European Network and Information Security Agency (ENISA) issued the Code of Practice, Good Practices for security of smart cars [8]

B) **Soft Law for AVs Safety**

There is no security standard for AVs as the automotive safety standards ISO 26262[6]. This standard aims to develop the functional safety of electric or electrical systems in the automotive industry. However, it does not deal with AVs.

Now, we have the Code of Practice (CoP) on the interaction between the disciplines of functional safety and cyber security issued by the Institute of Engineering and Technology (IET) [29].

[6] ISO 26262:2011, Road Vehicles – Functional Safety, 2011.

Chapter 4

Related Work on Security and Safety for AVs

4.1 Related Work on Ontologies for Safety, Security and Risks

Many researchers use the semantic approach for integrating data that are modelled with dynamic properties. This is the case for the data collected by the AVs through their sensors. The current studies attempt to model these data from three aspects: spatial, semantics, and temporal [30].

By focusing on semantic aspect, one of the most important ontology is W3C standard, called Sensor, Observation, Sample, and Actuator (SOSA) [31], which is a light-weight ontology. An application of SOSA in the automotive domain is Vehicle Signal Specification Ontology (VSSO), which complements it with the specific concepts of the vehicle [30]. VSSO relies on the Vehicle Signal Specification (VSS) taxonomy and follows the SOSA pattern to represent observations and actuations [32]. Both sensors and (VSS or VSSO) ontologies focus on vehicles, leaving aside other sensing and actuating devices that are in the environment, such as: traffic lights, speed sensors, induction loops, variable signalisation, and other parts of digital road infrastructure. This ontology does not focus on safety or security, but it is useful to design a safety or security ontology about the AV domain.

Smart Applications REFerence (SAREF) is an IoT ontology that describes the devices in IoT and their properties. This ontology focuses on the *"concept of a device, which is a tangible object designed to accomplish a* *particular task in households, common public buildings or offices"* [33]. In SAREF, the concepts are a light switch, temperature sensor, energy metre, and washing machine. The architecture is built in a modular way to define any device from predefined building blocks. Also, each device plays an important function for realising the task. Each function has a command that can act on the state and each device provides a service [33].

SAREF present device with some properties, such as `saref:hasModel` and `saref:hasManufacturer` [33].

Description of a Model (DOAM) is a *"framework that aims at representing and categorising knowledge about risk models that codify the relationships between the various components of a risk model universe"*. DOAM is inspired by the Description of a Project (DOAP) vocabulary developed by Edd Dumbill that was used for the annotation of open-source python applications. It was described as W3C RDF Schema and the OWL [34].

OntoSafe is a chemical process safety ontology. It is a public ontology, and it contains the main concepts for the process safety community. This ontology has 513 classes, 80 object properties, 70 data types and 58 individuals that complement the classes. Ontosafe, *"pretends to cover all the aspects related to process safety from toxicology to hazardous substances handling, to human factors, to risk analysis, to emission dispersion models, etc."* [35]. The ontology has been built considering: (a) chemical process safety fundamental concepts; (b) chemical process safety system; (c) industrial hygiene; (d) safety standards, regulations, and organisations; (e) mathematical models.

4.2 Related Work on Semantic Approaches

4.2.1 Risk

Xing et al. [36] focus on knowledge for safety risk identification in metro construction and design the framework of the SRI-Onto.

Description - The SRI-Onto consists of two main parts - such as the risk context and risk - and seven classes, that are the following: *"project, construction activity, risk factor, risk, risk grade, risk consequence, and risk prevention measure"* [36]. The risk categories cover the main safety risks in the main metro construction situations, and can meet the requirements for safety risk identification of common metro construction projects. The research focuses on *"risks closely related to the construction activities in risk identification of the metro projects (such as technical risks, geological risks, and environmental risks)"* [36].

Outcomes - The SRI-Onto is applied to identify the risk through an information system for assessing its competency. The authors develop a prototype of an automated risk recognition system for construction safety risk of metro projects (MRARS). The SRI-Onto is integrated into MRARS. The SRI-Onto has been used in the *"Fact Base Management (to describe safety risk knowledge of metro projects), the Rule Base Management (to describe the reasoning rules for safety risk knowledge), and Case Base Management (to describe existing cases) of the knowledge base management subsystem"* [36].

4.2.2 Safety

Zhao et al. [37] develop an ontology to represent maps, driving paths, and knowledge of the driving environments to improve the safety of intelligent vehicles. The goal is to enable intelligent vehicles to understand the driving environment.

Description - The dataset is a machine-understandable knowledge base for smart vehicles, which is constructed using some ontologies, such as: (a) map ontology, (b) control ontology, and (c) car ontology. The map ontology describes road networks such as roads, intersections, lanes, traffic light information, etc. The Control Ontology is intended to represent the driving behaviour and path of AV. The authors represent a path through instances of `control:PathSegment` instead of a collection of GPS points of a trajectory. Finally, Car Ontology includes the concept of different types of vehicles and devices installed in the car, such as sensors and engines. This dataset is used to develop real-time ADAS that can improve the safety in autonomous driving [37].

Outcomes - Zhao et al. [37] develop an Intelligent decision-making system to improve driving safety in ADAS. The decision-making system mainly consists of (a) sensor data receiver; (b) ontology-based Knowledge Base; (c) SPARQL query engine; and (d) Semantic Web Rule Language (SWRL) rule reasoner. *"The system makes decisions such as "Stop", "Go", "ToLeft", or "Give Way" in compliance with traffic regulations when it detects other nearby vehicles"*.

4.2.3 Cyber security

Many studies have explored semantic approaches that cover the domain of the cyber security domain. Torr [38] notes that semantic models in the cyber security domain operate on top of holistic model designed *"to make understand a product's threat environment and defend against potential attacks"*. **de Franco Rosa et al.** [6] develop a secure ontology to evaluate aspects. Their ontology is SecAOnto (Security Assessment Ontology), which aims to formalise the knowledge of security assessment aspects and particularities.

Description - SecAOnto is an OWL-based that is publicly available. It describes *concepts that consider both information security domain ontology and system assessment task ontology"* (p.1) [6]. SecAOnto comes from glossaries, vocabulary, taxonomies, anthologies and market's guidelines. However, *"these concepts are defined from a new perspective because the researchers adapted concepts to countermeasures, assets and attacks"* (p.2) [6].

The core concepts of SecAOnto include: (a) Systems Assessment (Assessment, Test, Verification and Evaluation); (b) Information Security (security, defences, vulnerability; attack; risk; threat); (c) Security Assessment (design defect; development defect; operation defect).

SecAOnto is applied as a core element in the development of a coverage calculus algorithm. It is used for identifying concepts in descriptions of assessment items (p.3) [6].

Outcomes - SecAOnto is an ALCHIQ(D) ontology that contains 758 Axioms, 290 Logical Axioms, 156 Classes, 37 Object Properties, 14 Object Properties Domains, 56 Individual Axioms, and 202 Annotation Axioms (p.3) [6].

Gyrard et al. [39] adopt a semantic approach to secure the ETSI Machine-to-Machine Architecture. They propose the Security Toolbox: Attack & Countermeasure (STAC) ontology-based security knowledge. The goal is to help software developers or designers of the ETSI M2M architecture to choose security mechanisms to make secure IoT applications.

Description - STAC ontology relies on current ontologies for wireless communications (cellular, wireless, wired), devices (sensor or mobile phone) and applications (programming languages, frameworks, database).

Outcomes - The ontology proposes countermeasures that can be used against threat, but does not describe the vulnerability of the M2M technologies.

4.2.4 Transportation sector

Alvarez-Coello and Gomez [30] propose an ontology-based method for integrating vehicle-related data that come from three different applications. Their goal is to show the *"sharing concepts with a predefined graph-like schema that can enable cross-application queries in a vehicle"* (p.1) [30].

Description - The authors use the semantic model: IoT-Streams ontology to cover the outcomes of applications built from vehicle data. In IoT-Streams ontology, *"the dynamic behaviour is captured by the concepts of Stream Observation and Event"*. Then, the VSSO complements it with the specific concepts of the vehicle (p.2) [30].

They define the annotation pattern to integrate data and use two criteria for annotating the data: *"(a) the source stream from which the stream of interest derives, and (b) the type of data of the feature of interest"* (p.3) [30].

Also, the authors validate their approach *"by implementing different applications that use vehicle data and apply the proposed semantic annotations to their outcomes. The resulting semantic data was then queried for cross-application analytical questions"* (p.4) [30]. The implementation consists of three steps: (a) test data; (b) applications; (c) cross-application queries.

(a) The test data are collected from a vehicle and the route has several segments that correspond to the geometries of either a left curve, right curve, or a straight section. (b) There are the three following applications: Semantic Sensor Data Stream (which is a description of the vehicle data stream); Dangerous Driving classification; Track Location classification (which is a classifier of the current position of a vehicle). (c) Finally, cross-application querying consists of querying the resulting graph after annotating it with a semantic model (p.4) [30].

Outcomes - The researchers develop an ontology with three straight points: *"(a) straight forward implementation of analytical queries that are stable over time; (b) re-usability of specific outcomes; and (c) increased semantics. In the experimental setup, they implement three basic applications using vehicle data, the results of which were semantically annotated using the proposed approach. Then, the analytical questions are formulated and satisfied with queries that follow the semantic model's pattern"* (p.1) [30].

Also, the authors *"implement three different applications that use vehicle data and applied the proposed annotations to achieve ontology-based data integration"* (p.5) [30]. Finally, they showed that *"several possible interactions between applications are achievable with queries that follow the pattern of the* *semantic model, deriving analytic that could serve as the basis for countless use cases"* (p.5) [30].

4.2.5 IoT

Mozzaquatro et al. [40] propose an ontology (IoTSec) with M2M communication security concepts to find security solutions in IoT environments. The IoTSec reference ontology is implemented in the OWL.

Description - The authors make a reference ontology through the following three steps. The first step explores the keywords for IoT. The second step consists of collecting existing ontology and taxonomies. This collection aims to set up and identify similarities and differences. The third step aims to create a harmonisation and mapping process of existing ontology in the design of a reference ontology.

IoTSec ontology encompasses the main categories of security information, such as (a) Assets (Wi-Fi, web, GSM, UTMS, LTE, Ethernet, Bluetooth, Sensor, etc.); (b) Threats (focuses on attacks that exploit the applications' weakness); (c) Security mechanism (detective, preventive, corrective, recovery, response, etc., mechanisms); (d) Vulnerability (potential weakness of M2M technologies). In particular, Assets require security properties to be considered secure in terms of availability, confidentiality, integrity, and non-repudiation.

The ontology explores the relationships between classic components of risk analysis to provide an overview of the domain of security in the IoT. For example, the mitigate attribute represents the relationship between security mechanism and vulnerability classes.

Outcome - The authors apply the IoTSec ontology in the network of industrial companies for ensuring a secure environment in data communication between companies' smart devices and Cloud Collaborative Manufacturing Networks (C2NET) platform. IoTSec is a knowledge base to feed the ontology-based security framework, which could seek information and infer new security mechanisms for the situation according to the IoTSec ontology's information.

Alvarez-Coello and Gomez [41] propose an ontology-based cyber security framework to address security issues and strengthen the protection of IoT devices and IoT business processes. They try to improve IoT cyber security from an ontological analysis. Also, the authors use their ontology, that is IoTSec ontology.

Description - The researchers propose an ontology-based cyber security relying on two approaches. The first one is the design time, which provides a dynamic method to make security services through the application of a model-driven method considering the existing enterprise processes. The second one is run time, which consists in monitoring the IoT environment and classifying threats and vulnerabilities by ensuring the correct adaptation of the existing services.

This study is interesting from two points of view. The first one is the methodology used to evaluate the ontology. They use the Software Engineering Standard (SQuaRE) that enables to evaluate the ontology with regard to: structural, functional adequacy, adaptability, reliability, transferability, maintainability and operability features.

The second one is the definition of inference rules by the SWRL with the Protégé editor by using the reasoner Pellet to make the rule processing. *"The reasoner manipulates the ontology using inference rules to reason with individuals, user-defined data types, and debugging support for ontology. Knowledge reasoning can infer in several cases, discovering the relationships among assets, vulnerabilities, threat security properties, and security mechanisms"* [41].

Tao et al. [42] propose a novel multi-layer cloud architecture model for IoT-based smart homes. This model helps to establish interaction and/or interoperations between heterogeneous home devices and services provided by different vendors. The main core of this model is ontology, which aims to solve the heterogeneity problem in the layered cloud platform. Furthermore, the authors use ontology to support security and privacy protection during interactions or interoperations.

Description - The smart home domain ontology contains some general concepts of smart home scenarios and is organised in a hierarchical structure. Top-level structures include: Home Device, Entertainment, Environment, Data communications, and Security. The low-level structure of the smart home domain details these general concepts. The authors then define the relationship between concepts of interacting or interoperating on heterogeneous home devices and services.

The information implicit in the ontology can be inferred. SWRL is used as a selection tool for defining the inference rules necessary to achieve mutual understanding and interactions/interoperations between the heterogeneous devices and services involved.

Alam et al. [43] write security-enhanced ontologies in IoT. They propose *"a functional architecture of the IoT framework that incorporates secure access provision. They implemented several components of the functional architecture using semantic technologies"* (p.568) [43]. Their goal is to improve the security of IoT and the interoperability of the security aspects.

Description - The authors create an ontology composed of three interconnected ontologies: Sensor Ontology, Event Ontology, and Access Control Ontology. Sensor Ontology *"describes the sensors and the retrieved data by the sensors. The Event Ontology describes the fault and its characteristics. Most of the instances of these classes are derived from Sensor Ontology using certain policies. Access Control Ontology describes the actors involved in secure access provisioning"* (p.578) [43]. They implement their IoT architecture with several components. The key points of this study are the following: (a) *"the security reasoning module would be located in the semantic overlay layer of the functional architecture. Therefore, the role of semantics is to facilitate the comprehension of the information"*; (b) *"the security reasoning allows the system to take the authorisation decisions to IoT-enabled services"* (p.576) [43]. It is possible because the system contains the formal knowledge of the domain; (c) "the domain includes the sensors, sensor data, user and user attributes. Then, the semantic rules *"specifies the access authorisation constraints and the execution of rules will generate the authorisation decisions"* (p.577) [43]; (d) the semantic rule *"allows only specific Role group members in the service provider administrative domain to access an application such as monitoring"* (p.579) [43]; (e) the researchers ensure the interoperability of the system through the ontology and the Shepherd. This last is a *"M2M platform that for interoperability and integration that supports communication between connected devices and makes them accessible from anywhere at anytime"* (p.582) [43]; (f) *"the implementation occurred by the establishment of an intended two-way communication between Sun SPOT sensors and its base station, and also two-way communication between the embedded Linux system (where host application was installed) and the Shepherd Platform"* (p.582) [43].

Outcomes - The authors conclude that: (a) *"the Light weight semantics make the information machine-readable that facilitates the export of knowledge by software agents and automated machine"*; (b) *"the scalability of semantic enhancement is a real concern considering the sheer size of IoT envi-*

ronment"; (c) another concern is the reasoning about low-power sensors and devices (p.583) [43].

Qamar and Bawany [44] make an application of semantic modelling for smart cities. The author creates an architecture (ICADS) that provides smart city services to meet its security issues.

Description - Researchers design two ontologies that are two ICADS models: OntoICADS and Secure-OntoICADS to deal with the dynamics and security of smart cities. Secure-OntoICADS secure the OntoICADS formalising four security elements: vulnerability, attack, security requirement, and security mechanisms.

Outcome - The secure-OntoICADS was applied in three scenarios. They represent the smart grid, the smart traffic management; the smart parking in terms of OntoICADS.

4.2.6 Joint analysis of Safety and Security

Pereira et al. [5] provide an ontology that represent joint safety and security knowledge. They use the Systems-Theoretic Accident Model and Processes for Security process (STPA-Sec) to identify causal scenarios between safety and security. The goal is to help safety and security engineers to determine the mitigation needed for addressing hazards.

Description - Researchers make a joint safety and security ontology at the early stages of the system life cycle. Their approach is divided into three steps: the first step is a unified STAMP-based Ontology which combines existing safety and security ontologies. One of the important outcomes of STPA-Sec is security measures and safety recommendations. Safety Recommendation is *"the recommendation or mechanism to mitigate the causal factors identified in Safety Scenario; while Security Measure addresses causal factors identified in Security Scenario"* [5].

The second step is to formalise the ontology through Protégé so that the ontology can reason. The researchers create examples of scenarios for each attack mechanism. Then, they identify the relationship between attack and causal factors, security properties, and recommendations. This work *"enables us to use the reasoning service to extract distinct possibilities of an attack to damage an asset, which provides the systems engineer with the path to create scenarios"* [5].

The third step is to create a user interface that enables systems engineers to assess the safety and security of the system. The interface contains a few scenario drop-down boxes, including a list of attack mechanisms and categories to populate the combo box lists. By this interface, systems engineers are able to choose the causal factor, recommendation, and security property to create a scenario.

Outcomes - The ontology has been evaluated on an aircraft system that enables the avionic systems to update their database and software via a wireless connection. They chose this area because these features would reduce cost and time, but could introduce cyber security vulnerabilities that could compromise the safety of the aircraft. The authors do the first step of STPA-Sec according to the original guidelines. This step lets to use the outcome to perform the second step that consists in identifying system purpose and scope, assumptions and constraints associated with the analysis, unacceptable losses, hazards, and system boundaries, and modelling the mission functional control structure.

4.3 Analysis and Synthesis

This section aims to compare current ontologies on risk, safety, and security to understand the differences and similarities among them. First, we summarise the existing ontologies and studies about semantic approach under five key points: Name; Scope; Domain; Implementations; Application as shown in Table 4.1.

	NAME	SCOPE	DOMAIN	IMPLEMENTATION	APPLICATION
Xing et al. [36]	SRI-Onto	risk	metro construction	(1) define key concepts of design ontology ; (2) identifying the risk through the existing regulation	development of an automatic risk recognition system prototype for construction the safety risk of metro project (MRARS)
Risk [34]	DOAM	risk	all	(1) codify the relationships between the various components of a risk model universe	
Rodriguez and Laguia [35]	OntoSafe	safety	chemical process	the ontology relies on several concepts, as: (1) chemical process safety fundamental concepts; (2) chemical process safety system; (3) industrial hygiene; (4) safety standards, regulations and organizations; (5) mathematical models	
Daniele et al. [33]	SAREF	cyber security	IoT	(1) define the concept of devices (2) define the properties of the concepts	
de Franco Rosa et al. [6]	SecAOnto	cyber security	system assessment	(1) using the existing glossary, vocabulary, taxonomies, anthologies, and market guidelines (2) adaptation of concepts to countermeasures, assets and attacks	(1) application in the development of a coverage calculus algorithm; (2) used for identifying concepts in descriptions of assessment items
Gyrard et al. [39]	STAC	cyber security	Machine-to-Machine Architecture	(1) combining current ontology about wireless communications, devices and applications	(1) security Toolbox about attack countermeasure
Alvarez-Coello and Gomez [30]	IoT-Streams	cyber security	AVs	(1) integration of vehicle-related data which comes from three different applications	(1) application of semantic annotations to vehicle data (2) querying the semantic data
Mozzaquatro et al. [40]	IoTSec	cyber security	IoT	(1) exploring the keywords concerning IoT (2) collecting existing ontology and taxonomies; (3) harmonising and mapping process	(1) applied to the network of industrial companies concerning data communication between companies' smart devices and C2NET platform
Mozzaquatro et al. [41]	IoTSec	cyber security	(1) IoT devices (2) business processes of the IoT	(1) software engineering standard, called SQuaRE to evaluate the ontology (2) inference rules by SWRL with the Protégé	
Tao et al. [42]	IoT-based smart homes	(1) cyber security (2) privacy	IoT for smart homes	(1) design the architecture and the relations between concepts to make interaction among devices	
Alam et al. [43]		cyber security	IoT	(1) create an ontology including three interconnected ontologies: Sensor Ontology, Event Ontology and Access Control Ontology	
Qamar and Bawany [44]	(1) OntoICADS (2) Secure-OntoICADS	cyber security	smart cities	(1) create a OntoICADS for smart cities (2) create a secure OntoICADS formalising security elements	(1) applied to the smart grid, smart traffic management, and smart parking
Pereira et al. [5]		cyber security and safety joint ontology	IoT	(1) STPA-Sec to identify causal scenarios between safety and security (2) formalise ontology via Protegé (3) create interface enabling the systems engineer to assess system safety and security	(1) assess aircraft systems enabling avionic systems to update its database and software through wireless connection

Table 4.1: Summary of current ontologies and semantic studies

The second stage is understanding how many ontologies and research semantic works have some features that are interesting for our research. Our interest is to verify if the existing studies are related to the AV domain, or at least to the IoT. Then, see how many of these studies concern risk, safety, or cyber security spectrum. This classification lets to assess gaps in the current state-of-the-art on the topic. Therefore, we classify the research using seven key points: IoT; VANET; IoV; AV; Risk; Safety; Cyber Security as shown in Table 4.2:

	IoT	**VANET- IOV-AV**	**RISK**	**SAFETY**	**CYBER SECURITY**
SRI-Onto			X		
DOAM			X		
OntoSafe				X	
SAREF	X				X
SecAOnto					X
STAC		X			X
IoT-Stream		X			X
IoTSec	X				X
IoT-based smart homes	X				X
Alam et al. [43]	X				X
Secure-OntoICADS	X				X
Pereira et al. [5]	X			X	X

Table 4.2: The classification of current ontologies and semantic research based on features of interest for our study

Table 4.2 gives us an on-the-stop state-of-art about the existing research of the AV domain. We can see that:

A few numbers of studies focus on the AV domain.

The majority of the studies apply the ontology to the IoT. The studies are interesting for our research because IoV is an IoT extension, so we can use the same concepts. For example, Pereira et al. [5] formalise the ontologies for IoT, which is then applied to the aircraft system.

A few ontologies focus on the risk of the system, and only one ontology aims at enabling the security of the system. This fact is understandable because formalising risks or security risks for a specific domain needs of existing rules or regulations related to this domain. To identify risks and related safety axioms, we have to use the legal framework that regulates risks with regard to a certain domain. Xing et al. [36] use relevant regulations, case collections, related research reports, similar system platforms, and conclusion of expert seminar. All these sources represent the main knowledge sources of the SRI-Onto. The authors note that the *"SRI-Onto development process can be deemed as a process of extracting and formalising all domain knowledge from above sources"* [36]. The AV domain is characterised by the lack of standards as well as political documents that serve as guidelines or codes of conduct.

Most of the ontology focus on different aspects of cyber security. These ontologies are intended to describe the Information Security domain (more generically), or other specific security sub domains, but they do not address safety and security joint analysis within the AV domain. In contrast to risk and safety ontologies, the formalisation of cyber security is more practical because the cyber security concepts are common to all domains.

We note that only Pereira et al. [5] focus on safety and cyber security. The researcher uses the Systems-Theoretic Process Analysis (STPA) method to create a causal relationship between these two scopes. This event is useful for our research because we can create the bridge between safety and security based on the causal relationships' parameters. The authors use the STPA-Sec that is a STPA extension to identify system vulnerabilities and requirements for cyber and cyber-physical systems. Pereira et al. [5] note that: STPA-Sec helps to identify some hazardous control actions, causal scenarios, and causal factors. STPA-Sec underlines the identification of causal factors to provide an explanation of why an unacceptable loss occurs. STPA-Sec enables to generate security measures and safety recommendations to prevent unacceptable losses [5].

The analysis of current ontologies and studies shows that VANET, IoV and, in general, the AV domain is still an unexplored area from the semantic modelling perspective. Then, most of the existing ontologies do not explore safety and cyber security. Only Pereira et al. [5] designed a joint safety and cyber security ontology for the IoT environment and applied it to aircraft systems. In order to achieve a secure IoV, it needs to generate joint knowledge between safety and security concepts and axioms, since it ensures to identify the main security issues that have an impact on safety. Otherwise, we risk of putting many security concepts into the ontology, even if some of them are not important for ensuring a safe VANET or IoV. However, Pereira et al. [5] do not extend the implementation of the ontology to the AV domain.

4.4 Conclusion

This study shows a fraction of the complexity of the current state-of-the-art regarding VANET or IoV security. We examine current ontologies and semantic research works on risk, safety, and security. Our research curiosity aims to understand if security information in AV can be increased using a semantic approach.

Then, the studies concern risk or safety or security without a focus on all these aspects. This gap prevents us from reaching a higher level of safety in the AV domain. By contrast, we note that safety and security joint ontology can satisfy all safety criteria of a system. This is a result of the innovative nature of the semantic model that encompasses safety and security.

This paper presents the semantic-based approach that we propose with a preliminary ontology, which ensures to perform reasoning and inferences to analyse the link between security vulnerabilities and safety risks. The approach relies on: (a) a high-level ontology that incorporate safety and security concepts, relations, axioms, and rules. We leverage existing ontologies from the IoT, risk, safety and security areas, and design a new ontology that focuses on the safety-security link for the automotive domain, named Security-Safety Internet of Vehicles (SSIoV); (b) the instantiation of current data from the AVs area into the ontology (concepts, axioms, and rules), through a graph database that integrates both the ontology and data; (c) the analysis of the security vulnerabilities and safety risks, by exploiting the inference abilities provided by the graph database, identifying rules that demonstrate incompatibilities with both safety and security.

Part II

The Semantic-based AI Reasoning Tool

This approach has been shown at the 6th International Workshop on Critical Automotive Applications: Robustness & Safety (CARS), 13 Sep 2021, Münich, Germany. The following part is inspired by the paper accepted at the conference [45].

Conventional methods propose security solutions aimed at detecting specific network cyber security attacks, not involving safety [22]. A semantic approach provides a holistic perspective [38, 46].

Semantic approaches to cyber security. de Franco Rosa et al. [6] develop a Security Assessment Ontology (SecAOnto), which includes concepts for countermeasures, assets, and attacks.

A Semantic Approach in the transportation sector. Debbech [47] introduces an ontological approach for safety critical railways systems. Klotz et al. [4] present VSSO which utilises a VSS taxonomy for adapting Sensor, Observation, Sample, and Actuator framework to the vehicle domain. Viktorovi ́c et al. [48] propose the Connected Traffic Data Ontology (CTDO)) based on the SOSA ontology [31] to represent vehicles in the transportation ecosystem. Corsar et al. [49] make the Transport Disruption ontology to model travel and transport related events that have a disruptive impact on an agent's planned travel.

Semantic approaches to IoT. Bermudez-Edo et al. [3] develop IoT-Lite, a lightweight ontology representing IoT resources, entities, and services. Elsaleh et al. [50] propose IoT-Stream - a lightweight extension of SOSA ontology to annotate Stream Data in the IoT context.

Approaches for joint analysis of safety and security in the transportation field. [5] provide a unified Systems-Theoretic Accident Model and Processes (STAMP)-based ontology to represent safety and security knowledge to help safety and security engineers to determine the mitigation needed to address identified hazards in complex systems. Martin et al. [51] presents a schema for the joint use of safety and security analysis in the automotive domain without the use of ontologies.

The above studies provide a partial view of the issues that we take into account (IoT instead of IoV, security without safety, transport instead of cars, joint safety and security instead of ontology). Our research addresses this gap by providing semantic analysis to explore joint safety and security in IoV and apply it to real data. We combine, adapt, and extend some of the above ontologies, such as: IoT-Lite [3], VSSO [4], STAMP-based ontology [5]. Our research goes beyond the state-of-the-art as it involves an additional step including instantiating the current dataset into an ontology (concepts, axioms and rules). This instantiation would be achieved through a graph database, that integrates both the ontology and data. We verify the reasoning abilities of the graph database by querying it. The graph should enable automated analysis of cyber security impacting the safety automatically.

Chapter 5

Research Approach and Methodology

IoV is an IoT application, and it is a large-scale distributed system featuring wireless communication and information exchange on the internet among AVs, roads, and users. The connectivity in IoV is prone to hackers' attacks such as: sending commands to the vehicle for stealing data, tracking AVs, controlling cars' sensors or actuators; tampering with electric signals; diverting non-safety or safety critical functions, and so on [52].

5.1 Research Questions and Objectives

Cyber security attacks due to security breaches can have catastrophic consequences in terms of safety. Based on this assumption, we define our research question.

- ⋄ Can we design and implement a semantic-based AI reasoning tool for analysing causal security-safety issues?

  1. How can we model the knowledge of the safety and security domains to perform a semantic and automatic cyber security analysis, applied to AVs and IoV?
  2. Can we automatically identify security holes by reasoning on safety rules and vice-versa?

By following these research questions, we identify the objectives of our thesis

- ⋄ Designing and applying an ontology - jointly representing security-safety knowledge - to data (extracted from real scenarios);
- ⋄ Developing a semantic-based AI reasoning tool for automating the security-safety analysis.

We provide a novel semantic-based reasoning tool for AI, explore joint safety and security in IoV, apply it to real data, and identify safety vulnerabilities caused by security breaches [45]. Therefore, we generate a semantic reasoner, which is an application that makes logical inferences from a set of axioms, logical rules and asserted facts. Our application will understand whether a security attack against a vehicle can lead to safety issues for that vehicle and for other involved vehicles.

In the literature, there is a gap about a semantic reasoner for IoV. Now, safety engineers evaluate the causal relationships between safety and security by hand. They apply some analysis methods to assess the safety or security of the system, such as: Hazard Analysis and Risk Assessments (HARA) [53], Threat Analysis and Risk Assessment (TARA) [54], and STPA [55]. These methods require effort in terms of time and resources due to the lack of abstract knowledge.

5.2 The Semantic-based AI Reasoning Tool

This project aims to provide a semantic analysis of the link between security vulnerabilities and safety risks in the AV domain. We concentrate on security vulnerabilities involving signals and sensor networks from the automotive industry, and on safety risks involving faults, errors, up to failures linked to hardware (sensors, various communication signals, cars' actions) or human involvement (e.g., reaction time).

The first objective is contained in **Developing an ontology unifying safety and security in the AVs domain** (see ch. 6). We use Protégé-OWL 5.5.0-beta-9 software [56] to formalise the ontology. The SSIoV ontology is a complete ontology that refers to IoV and AV based on current ontologies. We extract and define parts of the ontology that are useful for this work, especially those associated with security vulnerabilities and safety risks involving signals, sensors, actors, and organisational aspects (possibly adjusting some concepts). The ontology define concepts; relationships and identifies causal relationships with a common vocabulary. For example, IoT-Lite is a lightweight ontology to represent IoT resources, entities, and services. We adapt IoT-Lite for describing IoV and AVs. We create the structure of our ontology by extracting the useful parts from the existing ontologies. It includes: *IoV organisation* (e.g. assets; object; system; service; etc.), *safety components* (e.g. near collision, deviation, safe stopping distance; emergency stopping manoeuvre; etc.), and *security components* (e.g. threats, attacks, etc.). We will also align the various ontologies (aligning concepts, verifying the underlying semantics, unifying names, and relationships).

The second objective consists in **modelling relationships between concepts; writing rules and axioms for safety risks and security vulnerabilities** (see ch. 7). We focus on events and rules, where security has an impact and causal relationships on safety. Axioms of security vulnerabilities simulates various security breaches and describe how they impact signals or sensors. The safety risk axiom model the variation of a system due to failure. We combine these two types of axioms to create security-safety rules, expressing causality from security to safety.

To identify the causal relationships, we use the STPA that is a safety analysis method on the STAMP model [55]. STPA lets the description of accident scenario to eliminate or control hazards in complex systems. STPA-Sec is a security extension of STPA extending it from safety to cyber security analysis [57], while STPA-SafeSec is a unified approach combining both safety and security analysis [58]. Some researchers applied the STPA approach to AVs [59, 60, 61] Their research outcomes will enrich our ontology as we convert these results into rules, which are then formalised and entered in Protégé.

Furthermore, to identify rules and axioms, we use existing regulations on the automotive domain. The existing sources are cyber security best practises, which provide guidance on the implementation of automotive cyber security principles [8], [27]; Recommendation on cyber security [28].

The third objective lies in **Verifying the ontology consistency through a reasoning engine** (see ch. 8). We perform analytical reasoning to verify the consistency of the developed unifying ontology, safety and security axioms, and security-safety rules. Once the consistency of our ontology and rules have been confirmed, we can move on to the next step.

The fourth objective consists in **Instantiating data into the ontology (concepts, axioms and rules)** (see ch. 9). This instantiation is achieved through a graph database, that integrates both the ontology and data. Vehicular Reference Misbehavior (VeReMi) [62] is a current dataset used to evaluate misbehaviour detection mechanisms for Vehicular ad hoc network (VANET). The dataset includes message logs of on board units and a labelled ground truth, generated from a simulation environment, as well as malicious messages designed to trigger incorrect application behaviour. This integration permits, for example, the detection of a spoofing attack that can lead to rear end-collision among AVs.

The fifth objective consists in **Querying the semantic-based AI reasoning tool** (see ch. 10). This task follows the Ontology-based Data Access (OBDA) method[1]. Hence, we can design, implement and execute specific queries and inferences on the graph database. For our example above: (a) we query the system to select spoofing attacks (the security breach); (b) we query the system to select rules involving spoofing attacks that have an impact (causal relationships) on the safety (safety rule and safety rule consequences); (c) the system answers with cases found in data and that involves this rule (safety issue).

The sixth objective consists in **analysing, evaluating and validating the outcomes** (see ch. 11). We identify and list the rules that are incompatible with security and safety, and those that do not prove incompatible. We will identify incompatibility when: (1) ontological reasoning indicates that some data violates the rules (e.g., in the above scenario, the car does not brake fast enough). This event means the rule, or its expression is not sufficient for ensuring safety in all cases; (2) ontological reasoning identifies some risks involved even if no data violate the rules; (3) manual analysis of risks and rules incompatibility for the remaining cases. Finally, we verify and validate our results (false positives and false negatives).

Figure 11.1 shows our methodology to design our semantic-based AI reasoning tool.

[1] See http://optique-project/

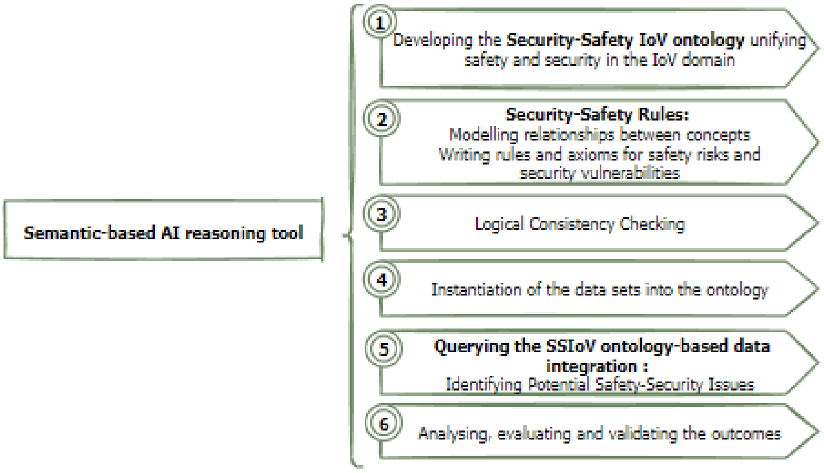

Semantic-based AI reasoning tool
1
Developing the Security-Safety IoV ontology unifying safety and security in the IoV domain
2
Security-Safety Rules:
Modelling relationships between concepts
Writing rules and axioms for safety risks and security vulnerabilities
3
Logical Consistency Checking
4
Instantiation of the data sets into the ontology
5
Querying the SSIoV ontology-based data integration :
Identifying Potential Safety-Security Issues
6
Analysing, evaluating and validating the outcomes

Figure 5.1: Methodology for our semantic-based AI reasoning tool

5.3 Conclusion

IoV has become the core paradigm for AVs. To exploit this network, we need to face the security challenges raised from the IoV connectivity and their impact on safety. We developed a methodology that uses an ontology with a semantic reasoner to investigate the link between safety and security, specifically targeting AVs.

In this paper, we present the preliminary results of our research. The study aims to provide a tool for improving preventive cyber defence capabilities in the IoV and AVs domains. Based on an integrated security and safety ontologies with rules, the tool highlights cyber security vulnerabilities that lead to safety risks. This work contributes to improve the security of IoV critical road infrastructures. Also, our research can contribute to improve security for transportation infrastructures at large, including aviation and railways in the long term. Finally, the research can have an impact of improving and influencing the current standards that are being produced in the IoV and AV domains.

Chapter 6

I. SSIoV ontology

6.1 Developing SSIoV ontology

The ontology defines a *“common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them”* [63]. Noy and McGuinness [63] gives some reasons for designing an ontology, such as:

1. “To share common understanding of the structure of information among people or software agents
2. To enable reuse of domain knowledge
3. To make domain assumptions explicit
4. To separate domain knowledge from the operational knowledge
5. To analyse domain knowledge” (p.1)

We design the SSIoV ontology for the first two reasons. About the reason n.2, Noy and McGuinness [63] note that *“if one group of researchers develops such an ontology in detail, others can simply reuse it for their domains. Also, if we need to build a large ontology, we can integrate several current ontologies describing portions of the large domain. We can also reuse a general ontology, such as the UNSPSC ontology, and extend it to describe our domain of interest”*. As we need to make a large ontology that encompasses several components of IoV, we integrate or extend some ontologies to our domain interest. This work consists in combining the existing ontologies in each of our different components to design a general structure of IoV, which includes vehicles, vehicular communication, security, and safety components.

Moreover, about the reason n.3, we do explicit specifications of domain knowledge as their ontologies' authors did. They describe the concepts that differentiate each component. This way is useful for users who do not have specific knowledge in the domain (p. 2) [63].

Finally, [63] propose six steps to develop an ontology as follows:

- Determine the domain and scope of the ontology;
- Consider reusing current ontologies;
- Enumerate important terms in the ontology;
- Define the classes and the class hierarchy;
- Define the properties of classes;
- Define the facets of the slots.

Based on this list, we explain the development of SSIoV ontology. We note that IoT-Lite is not intended to be a complete ontology of IoV. Our aim is to create a core lightweight ontology that enables to determine the causal relationships between safety and security issues.

6.1.1 The Domain and Scope of SSIoV ontology

About the domain and the scope of an ontology, Noy and McGuinness [63] raises some questions, such as:

- What is the domain that the ontology will cover?
- For what we are going to use the ontology?
- For what types of questions the information in the ontology should provide answers?
- Who will use and maintain the ontology?

The SSIoV ontology is intended to contain a large body of knowledge about IoV encompassing some of its important components. We recall that the objectives of this research consists of **in providing a semantic analysis of the link between security vulnerabilities and safety risks in the AV domain. Therefore, we focus on security vulnerabilities involving signals and sensor networks in the automotive industry, as well as safety risks involving faults, errors, up to failures related to hardware or human involvement**. Therefore: a) SSIoV ontology field is a representation of security and safety in IoV; and b) we plan it to establish a causal relationship between safety and security concerns.

Naturally, SSIoV encompasses four core parts: (a) IoV concepts and relationships; (b) vehicle's signals and sensors; (c) safety components; (d) cyber security components. At the same time, SSIoV ontology includes concepts to link security and safety ontology.

These components are useful to face the "competency questions" that enables to determine the scope of the ontology [63]:

- What are the safety consequences for vehicles when they suffer a security attack?
- When do AVs suffer a cyber-attack?
- What are the security attacks against a vehicle?
- What happens if a vehicle does not maintain the minimum safety distance?
- Which part of vehicle would be affected by a safety attack?
- What AV's sensor could be affected by an attack?
- What safety rules do the AVs have to follow?
- When do the AVs follow the safety rules?

These questions are the basis of our ontology, and they can change during our work because it is not an exhaustive list.

6.1.2 Re-use Current Ontologies

The SSIoV ontology combines, adapts and extends some of the above ontologies, namely: IoT-Lite [3], VSSO [4], STAMP-based ontology [5] as shown in Figure 6.1.

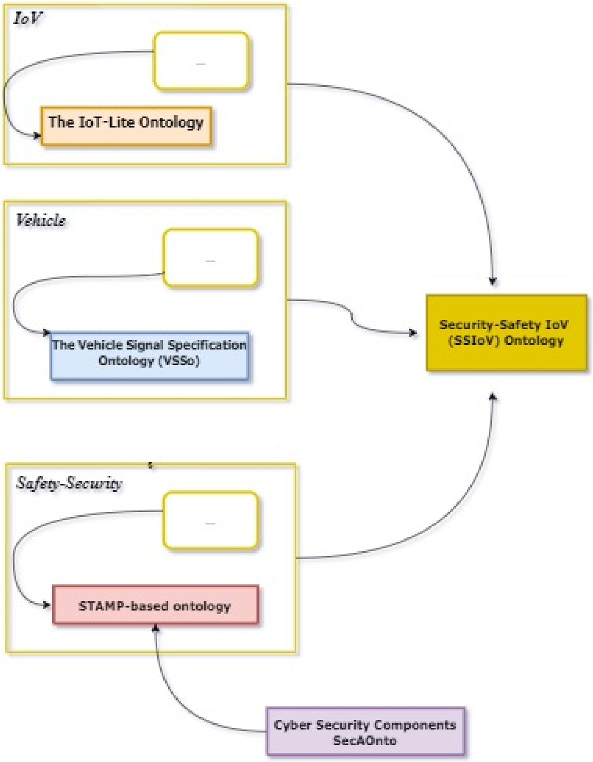

IoV
The IoT-Lite Ontology
Vehicle
The Vehicle Signal Specification Ontology (VSSo)
Security-Safety IoV (SSIoV) Ontology
Safety-Security
STAMP-based ontology
Cyber Security Components SecAOnto

Figure 6.1: Reusing, adapting and extending current ontologies

The SSIoV ontology takes up the existing ontologies that already organise the vocabulary. It is easier to reuse these ontologies because they provide specific vocabulary and taxonomies that are intricate about the IoV. In fact, each of these ontologies refer to a particular aspect of IoV. However, we avoid a simple reuse of these ontologies, because not all concepts are useful for our work. We only take the most common terms for each IoV component. For example, our inference tool tries to understand what happens in the case of GPS spoofing attack. It queries the ontology for the consequences of this cyber-attack. Hence, the ontology needs the concept of attacker, GPS sensors, the minimum safe distance, etc. Other concepts are irrelevant to this query. Moreover, these concepts should be easily accessible by avoiding complex hierarchy or relationships between concepts. Otherwise, the query will retrieve the unexpected outcomes. We try to make a simple ontology to ensure its scalability too.

[3] note that IoV and connected roadways also depend on physical devices, such as sensing or actuating devices, AVs, or RSUs, OBUs. Hence, to develop an ontology for IoV and connected roadways, we should consider other innovative ontologies, such as AVs or mobility sensor ontology. In fact, the IoV technologies are implemented through some physical RSUs or OBUs and connected to different sensors on vehicles, robots, infrastructure, or personal devices (smartphones, smartwatches, etc.).

Then, to develop IoV ontology we can also benefit from ontologies concerning the IoT domain because these ontologies share some common same concepts and technologies.Hence, we re-use IoT-Lite [3], that is an ontology developed for the IoT domain. IoT-Lite has three main classes, i.e., objects, system, and services. Objects are *"any entity in the IoT environment. A System is a unit of abstraction for all the physical entities for sensing. The system has components and subsystems. Service refers to any service provided by IoT devices"* [3]. For examples, AV, RSUs and OBUs can be defined as instances of class Object. Some of the objects, such as RSUs and OBUs, provide a service, as for instance DSRC service. Such devices have a Coverage property. The coverage of a device is of geospatial data type, which shows the area covered by a device.

Moreover, any mobility sensing device can have a connected property to a service, such as a device that provides a DSRC communication. Also, the ontology of the traffic road network should be connected to the IoV ontology. In addition, vehicle ontology should be extended to include the innovative OBUs and other sensors used for V2V, V2I, V2P, or V2S communications [64].

VSSO provides vehicle-specific concepts [30]. It relies on the VSS taxonomy and follows the SOSA pattern for observations and actuations [32]. Both sensors ontologies (VSS or VSSO) focus on vehicles, leaving aside other sensing and actuating devices that are in the environment, such as traffic lights, speed sensors, induction loops, variable signalisation, and other parts of digital road infrastructure. This ontology does not focus on safety or security, but it helps in designing safety or security ontology on AV domain.

We use VSSO because it let you understand specific signals and sensors providing a specific vocabulary about it. However, a) VSSO only focuses on vehicles. Therefore, it does not incorporate other sensing and actuating devices, such as traffic lights, speed sensors, induction loops, variable signalisation, and other parts of digital road infrastructure; b) VSSO refers to classic vehicle's signals and sensors. It does not include AV's signals and sensors. We adapt VSSO to our area of interest by using the most common terms in the field. We highlight that SSIoV ontology will integrate the AV's signals and sensors with STAMP-based ontology.

STAMP-based ontology. Pereira et al. [5] provides an ontology that represent joint safety and security knowledge. They use the STPA-Sec to identify causal scenarios between safety and security. Their goal is to help safety and security engineers to identify the mitigation needed to address the identified hazards. We use this ontology because it deals with safety-security jointly. Then, we enrich it through: (a) the AV's Assets that are in ENISA [8]; (b) description of assets written by Yazdizadeh and Farooq [64]; (c) we do not use the specific categories about STPA-Sec components.

SecAOnto (Security Assessment Ontology) aims to formalise the knowledge on security assessment aspects and particularities. It describes concepts concerning both information security domain ontology and system assessment task ontology. SecAOnto comes from glossaries, vocabulary, taxonomies, anthologies and market's guidelines. The novelty of this research lies in defining concepts in a new perspective adapting them to countermeasures, assets and attacks [6]. SecAOnto includes (a) Systems Assessment (Assessment, Test, Verification, and Evaluation); (b) Information Security (security, defences, vulnerability; attack; risk; threat); (c) Security Assessment (Design defect; Development Defect; Operation Defect). We only consider the information security party as it shows a complete taxonomy of security attacks. Additionally, Pereira et al. [5] integrate STAMP-based ontology with SecAOnto. Finally, we reuse SecAOnto, enriching it with the list of attacks shown in ENISA [8].

6.1.3 Listing Concepts and Relationships in SSIoV ontology

SSIoV ontology includes the main concepts and relations that represent the central core of our ontology, as shown in Table 6.1.

CONCEPT	**RELATIONSHIP**	**CONCEPT**
Asset	has	Vulnerabilities
Attack	threatens	Asset
Attack	exploits	Vulnerabilities
Attack	threatens	Safety Properties
Attack	causes	Hazard
Attack	threatens	Security Property
Hazard	damages	Asset

Table 6.1: Main concepts and relations of SSIoV ontology

The three columns capture concepts and relationships of a cyber-attack scenario. The *asset*'s *vulnerabilities* enable the *attack* to exploit the *asset*. This *attack* can cause *hazards*. The main concepts and relationships are also shown in Figure 6.2

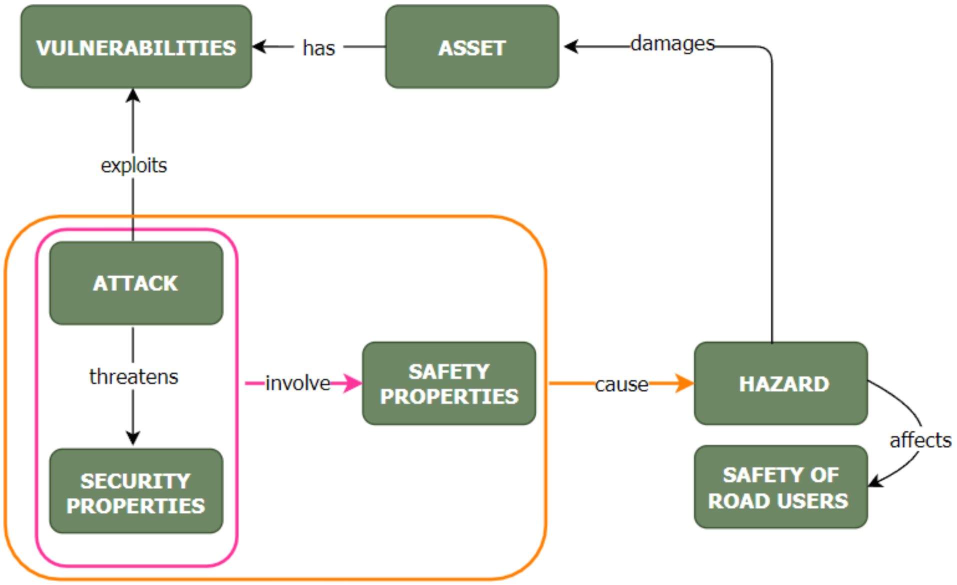


Figure 6.2: The relations between main concepts of SSIoV ontology

6.1.4 Defining Classes, and Class Hierarchy in SSIoV ontology

To develop class hierarchies, the available ontologies use a top-down approach that starts by defining the most general concepts (classes) in the domain and subsequent specialisation of the concepts (sub classes). There are several possible ways to develop class hierarchies [65].

6.1.4.1 Adaptation of IoT-Lite to IoV

We adapt the VSSO ontology as described in section 6.1.2. Classes, sub classes, instances, and annotations of safety-security components are shown in Table A.1.

6.1.4.2 Adaptation of VSSO to our interest domain

We adapt the SSIoV ontology as explained in section 6.1.2. Classes, sub classes, instances and annotations of safety-security components are listed in Table A.2.

6.1.4.3 Adaptation of STAMP-based ontology to our interest domain

We adapt the STAMP-based ontology as explained in section 6.1.2. Classes, sub classes, instances, and annotations of safety-security components are listed in Table A.3.

6.1.4.4 Adaptation of SecAOnto to our interest domain

We adapt the STAMP-based ontology as described in section 6.1.2. Classes, sub classes, properties and annotations are listed in Table A.4.

The combining procedure is shown in Figure 6.3, where we have the representation of SSIoV ontology. SSIoV covers four core parties of our interest domain: IoV (to which we use the IoT-Lite ontology); AV (for which we use the VSSO by adding the AV's sensors); and security and safety components.

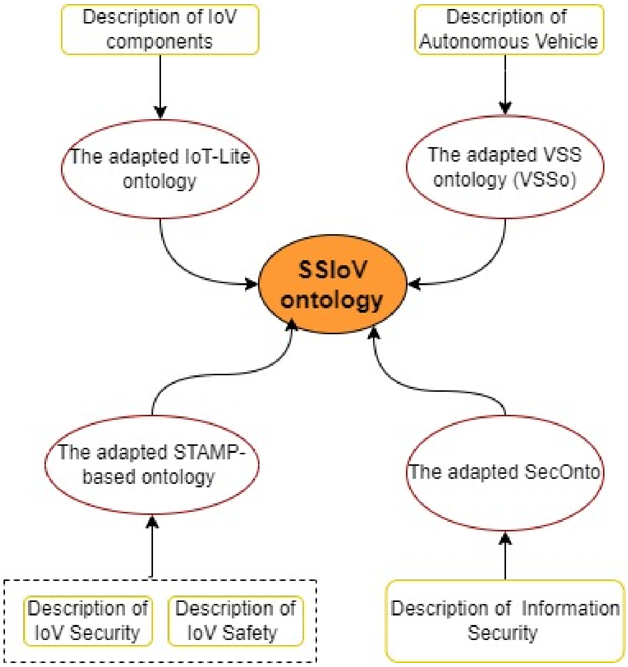
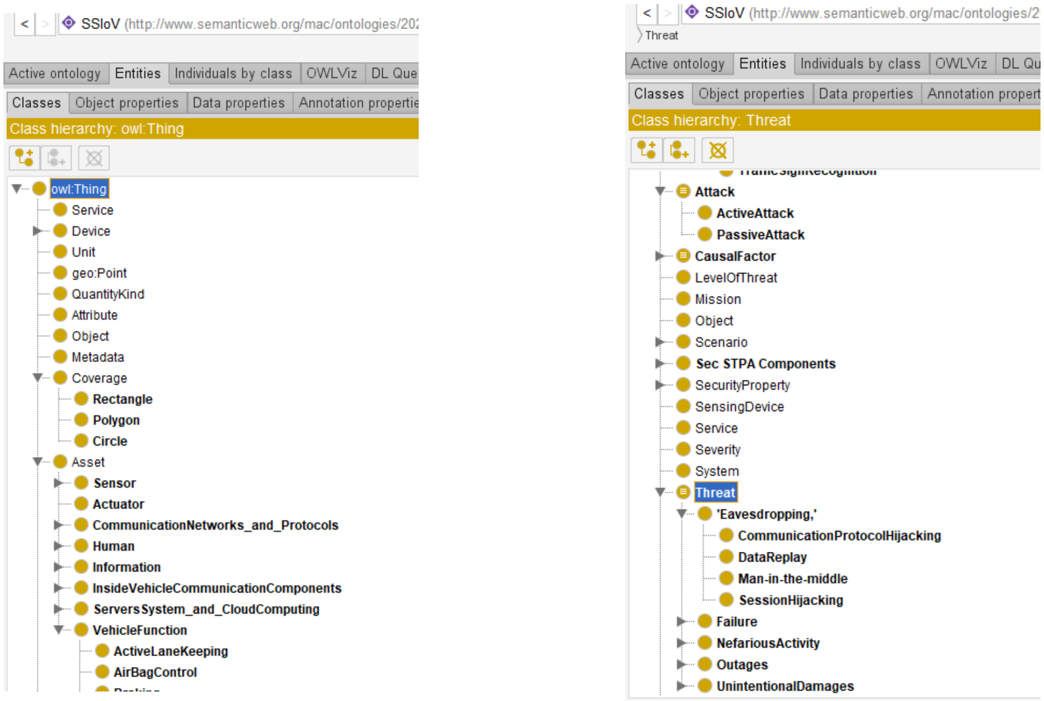

Description of IoV components
Description of Autonomous Vehicle
The adapted IoT-Lite ontology
The adapted VSS ontology (VSSo)
SSIoV ontology
The adapted STAMP-based ontology
The adapted SecOnto
Description of IoV Security
Description of IoV Safety
Description of Information Security

Figure 6.3: The combination of 4 current ontologies for SSIoV ontology

Some results of this combining process are shown below in Figure 6.4 that is extracted from the SSIoV ontology made on Protégé-OWL 5.5.0-beta-9.



< > SSIoV (http://www.semanticweb.org/mac/ontologies/202

Active ontology | Entities | Individuals by class | OWLViz | DL Que

Classes | Object properties | Data properties | Annotation propertie

Class hierarchy: owl:Thing

- owl:Thing
  - Service
  - Device
  - Unit
  - geo:Point
  - QuantityKind
  - Attribute
  - Object
  - Metadata
  - Coverage
    - Rectangle
    - Polygon
    - Circle
  - Asset
    - Sensor
    - Actuator
    - CommunicationNetworks_and_Protocols
    - Human
    - Information
    - InsideVehicleCommunicationComponents
    - ServersSystem_and_CloudComputing
    - VehicleFunction
      - ActiveLaneKeeping
      - AirBagControl

(a) Image A

< > SSIoV (http://www.semanticweb.org/mac/ontologies/2

Threat

Active ontology | Entities | Individuals by class | OWLViz | DL Qu

Classes | Object properties | Data properties | Annotation propert

Class hierarchy: Threat

- Attack
  - ActiveAttack
  - PassiveAttack
- CausalFactor
- LevelOfThreat
- Mission
- Object
- Scenario
- Sec STPA Components
- SecurityProperty
- SensingDevice
- Service
- Severity
- System
- Threat
  - 'Eavesdropping,'
    - CommunicationProtocolHijacking
    - DataReplay
    - Man-in-the-middle
    - SessionHijacking
  - Failure
  - NefariousActivity
  - Outages
  - UnintentionalDamages

(b) Image B

Figure 6.4: A portion of SSIoV ontology

6.1.5 Defining Relationships among Concepts in SSIoV ontology

We already listed the main relationships between the core concepts. Here, the goal is to extend this list by identifying more correlations between classes and subclasses. The following relationships list (6.2) is not exhaustive, but it provides an example about the semantic links among concepts. Most of these relationships are taken from the reused ontologies.

Subject	Predicate	Object
Coverage	has Point	geo: Point
Device	is subsystem of	System
Device	on platform by	Platform
Device	exposed by	Service
Device	has unit	degree
Device	has quantity Kind	temperature
Entity	has attribute	attribute
Object	has attribute	Attribute
Platform	geo:has location	geo:Point
Service	has Coverage	Coverage
Service	exposes	Device
Sensor	has quantity kind	Quantity Kind
System	has deployment	Deployment
RSUs	provide	DSRC
Vehicles	provide	DSRC
OBUs	provide	DSRC
System	aims to do	Mission
System	must do	Mission
Attack	causes	Unacceptable Loss
Causal Factor	can lead to	Hazard
Vulnerability	can lead to	Breach of security
Causal Factors	are identified into	Safety Scenario
Causal Factors	can cause	Unacceptable Loss
Causal Factors	violate	Security Properties
Safety Scenario	lead to apply	Safety measure

Continued on next page

Table 6.2: Some relationships among SSIoV ontology's concepts

Table6.2 – continued from previous page

Subject	Predicate	Object
Assurance	equivalent	Security
Dependability	equivalent	Assurance
Dependability	equivalent	Security
Human	exercise	Defect
Human	make	Mistake
Attack	disjoint with	Threat
Attack	disjoint with	Risk
Defect	is exploited by	Attack
Error	is generated by	Defect
Failure	is propagated by	Error
Mistake	insert	Defect
Vulnerability	equivalent	Weakness
Vulnerability	has risk	

The SSIoV ontology contains 282 classes; 115 object properties; 31 data property; 38 individual; 3174 axioms; 2560 logical axioms.

Figure 6.5 represents some concepts and relations of SSIoV ontology.

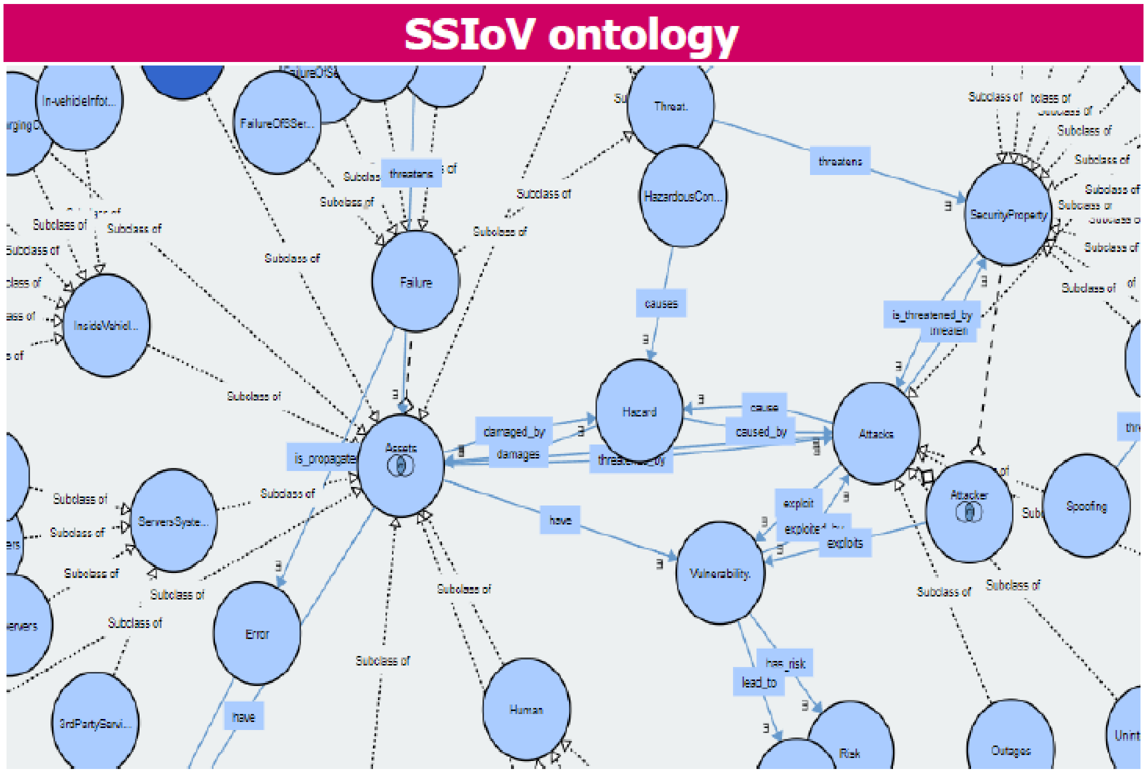


Figure 6.5: The representation of an extract of SSIoV ontology on WebVOWL [1]

6.1.6 The Representation of Two Case Studies with SSIoV ontology

We use the concepts and relationships developed in SSIoV ontology to depict two cases study. We assume that GPS spoofing attacks have safety consequences on AVs. Then, we assume a security attack that manipulates the AVs's manoeuvres that are part of a vehicle platoon. The security attack causes the consequences in terms of safety.

6.1.6.1 Use case 1: GPS Spoofing Attack against a Target Vehicle leads to a Collision

AVs broadcast beacon GPS signal messages to inform of their presence. In Figure 6.6 an *attacker* sends a falsified *GPS signal* (that is a type of

GNNS signal [8]) of its own position to the *target vehicle*. The *spoofing* attack threatens the *authenticity* of the *sensors* signal. The GPS signal (falsely) mentions that the position of the attacker is very close to that of the target vehicle. The latter then applies a safety measure (*emergency stopping manoeuvre*) for ensuring a *safety property* (*safe stopping distance*) that leads to a *rear-end collision* with the rear vehicle (*hazard*).

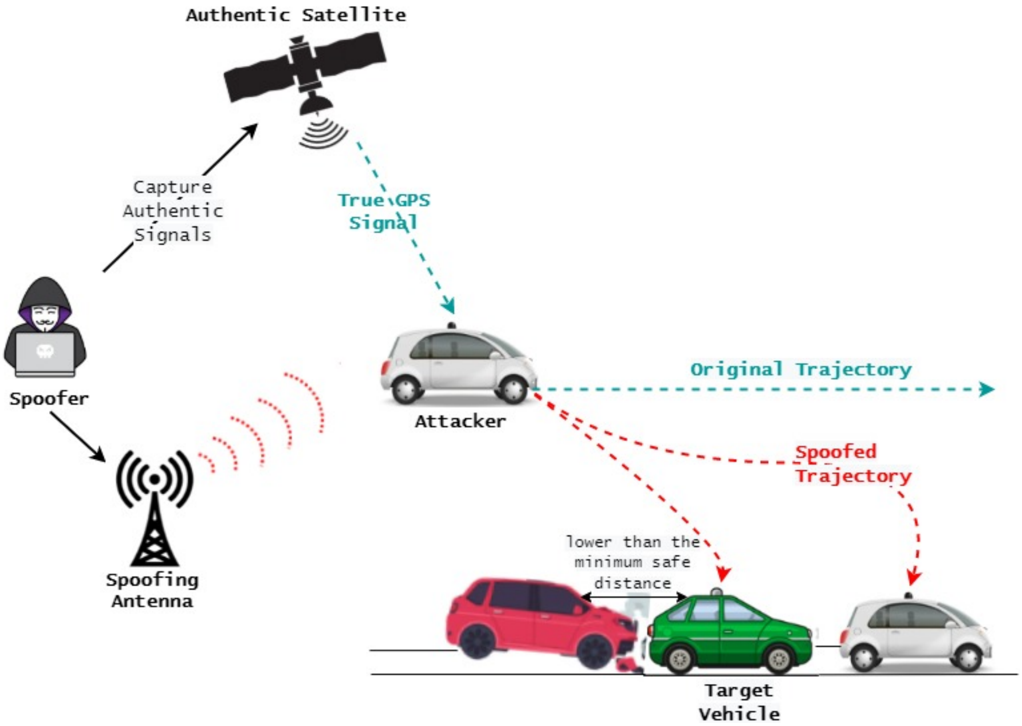


Figure 6.6: The representations of the GPS spoofing attack against a target vehicle

Starting from these simple relations among concepts, we develop the SSIoV ontology, as follows in Figure 6.7.

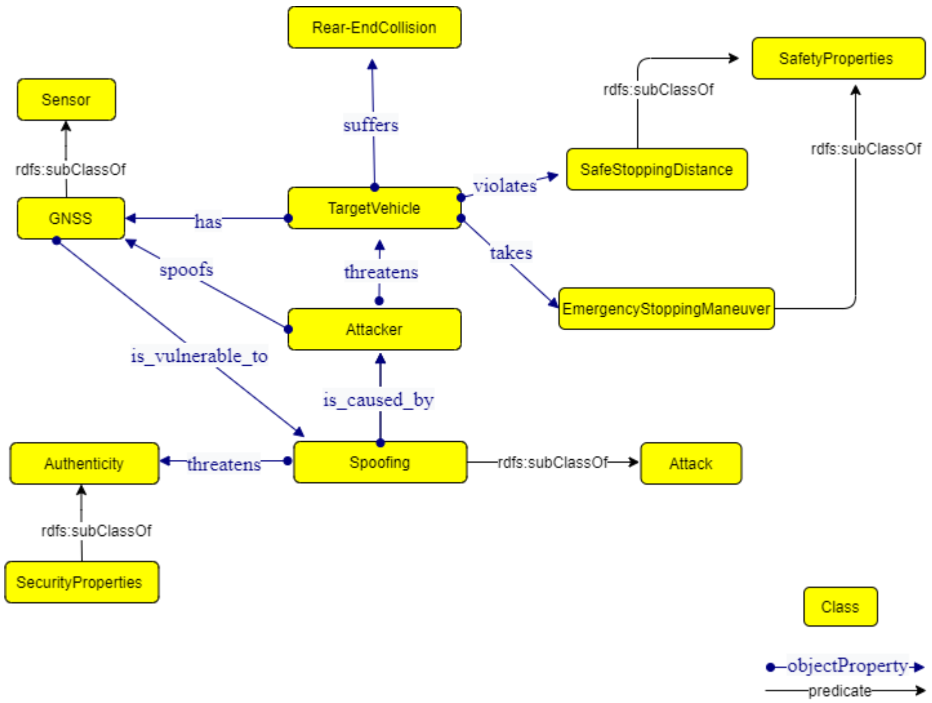


Figure 6.7: Formalisation of SSIoV ontology: concepts and relationships of the first running example with Graffoo[1]

Figure 6.7 shows a portion of the ontology that correspond to our running example. The figure represents a *spoofing attack* propagated by an *attacker* against a *target vehicle*. The figure contains the main concepts and relationships shown in the Table 6.1 : Target Vehicle and GPS are *Assets* [66]; spoofing is an *Attack*; Authenticity is a *Security Property*; Safe Stopping Distance and Emergency Stopping Manoeuvre are *Safety Properties*; Rear end-Collision is the *Hazard*.

6.1.6.2 Use case 2: The Platoon Dissolution due to a Reply Attack

The AVs of a platoon communicate with each other through sensors, cameras, V2V and 3G/4G/LTE. The attacker can tamper these sensors or can intercept the wireless communication signals. Vehicles should exchange velocity; position; acceleration to adjust their speed accordingly. The safety that impacts on the security attack can be dangerous. The platoon is vulnerable to the hacker's attack, which can detect and manipulate the communication between these vehicles for re-transmitting the information to other vehicles of the platoon.

Let's suppose an example about a reply attack against a member vehicle that leads to platoon dissolution. In the Figure 6.8 the *"lead vehicle "a" communicates to the member vehicle behind it to close the gap at X time. After a few seconds, the leader transmits a signal to the vehicle behind it to back off a little. An attacker recorded the message transmitted at time X and replayed that at time Y, which is after the leader requested member vehicle "b" to back off. Member vehicle "b" will now discount the previous message and instead, seek to close the gap. If repeatedly done, then by replaying the old message, the attacker will make the platoon oscillate as members try to position themselves into the best positions based on the information they receive. This can lead to discomfort for the passenger's and even vehicle collisions"* (p.5) [67].

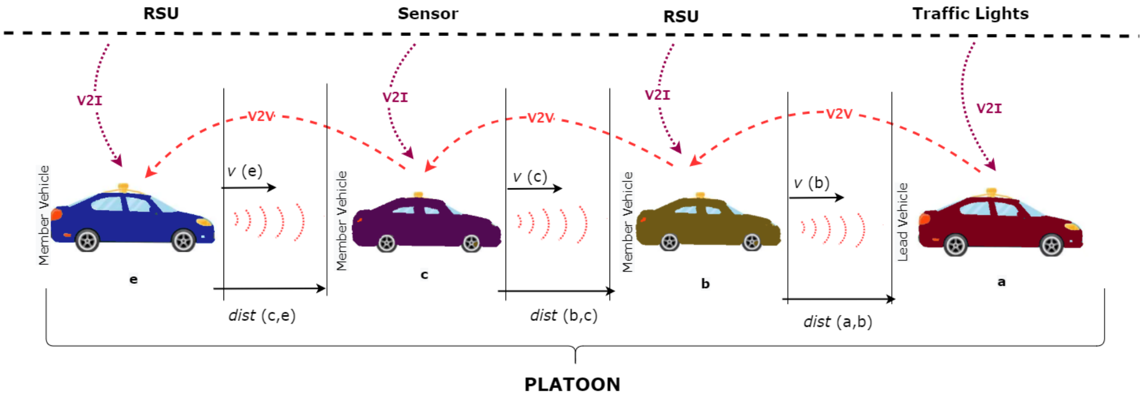


Figure 6.8: A stable vehicle platoon

In a *platoon* of AVs, maintaining homogeneous speed, constant distance is a safety requirement for platoon stability. A *platoon* of AVs travels at a speed of 25 m/s and at a distance gap of 15 m on a single lane. The *lead vehicle* "a" communicates to the *member vehicle* "b" behind it to close the gap at time "t". After a few seconds (at time "t+1"), the leader transmits a signal to the vehicle "b" behind it to back off a little. An *attacker* recorded the message transmitted at time "t" and replayed that at time "t+2", after the leader asked member vehicle "b" to back off. Member vehicle "b" will now discount the previous message and instead tries to close the gap. If repeatedly done, then by replaying the old message, the attacker will make the platoon oscillate as members try to position themselves into the best positions based on the information they receive. This event can lead to discomfort for the passenger and vehicle *collisions* [68] as shown in Figure 6.9.

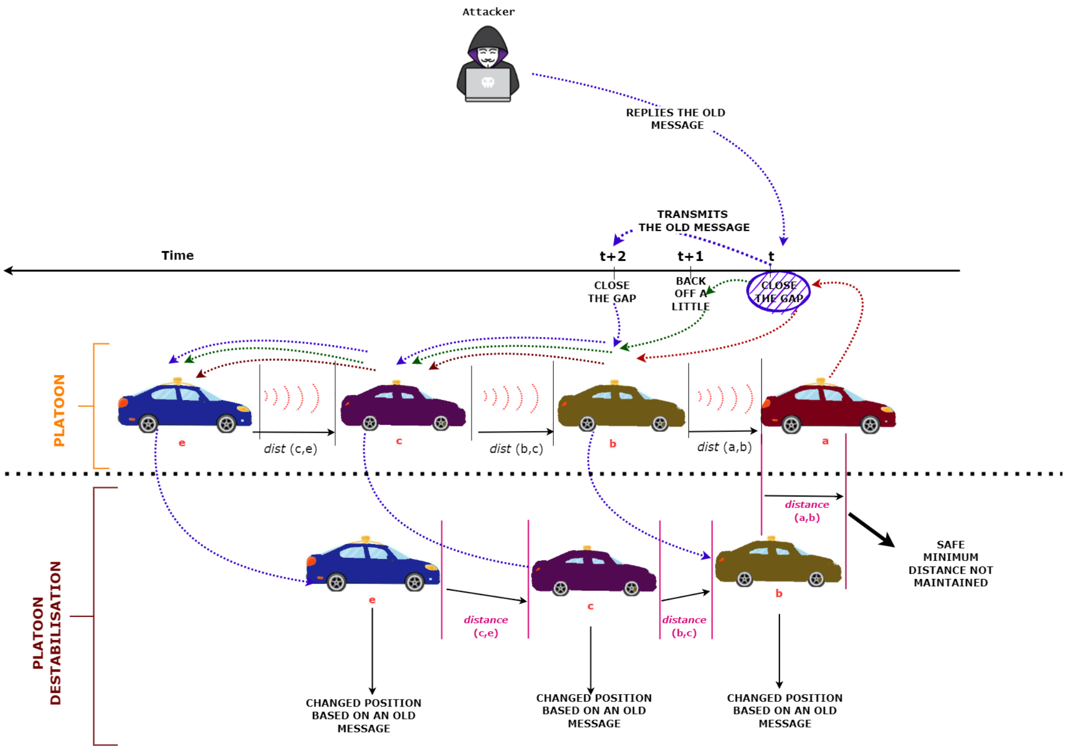


Figure 6.9: The disruption of a vehicle platoon due to a reply attack

Starting from these simple relationships between concepts, we develop the SSIoV ontology, as shown in Figure 6.10.

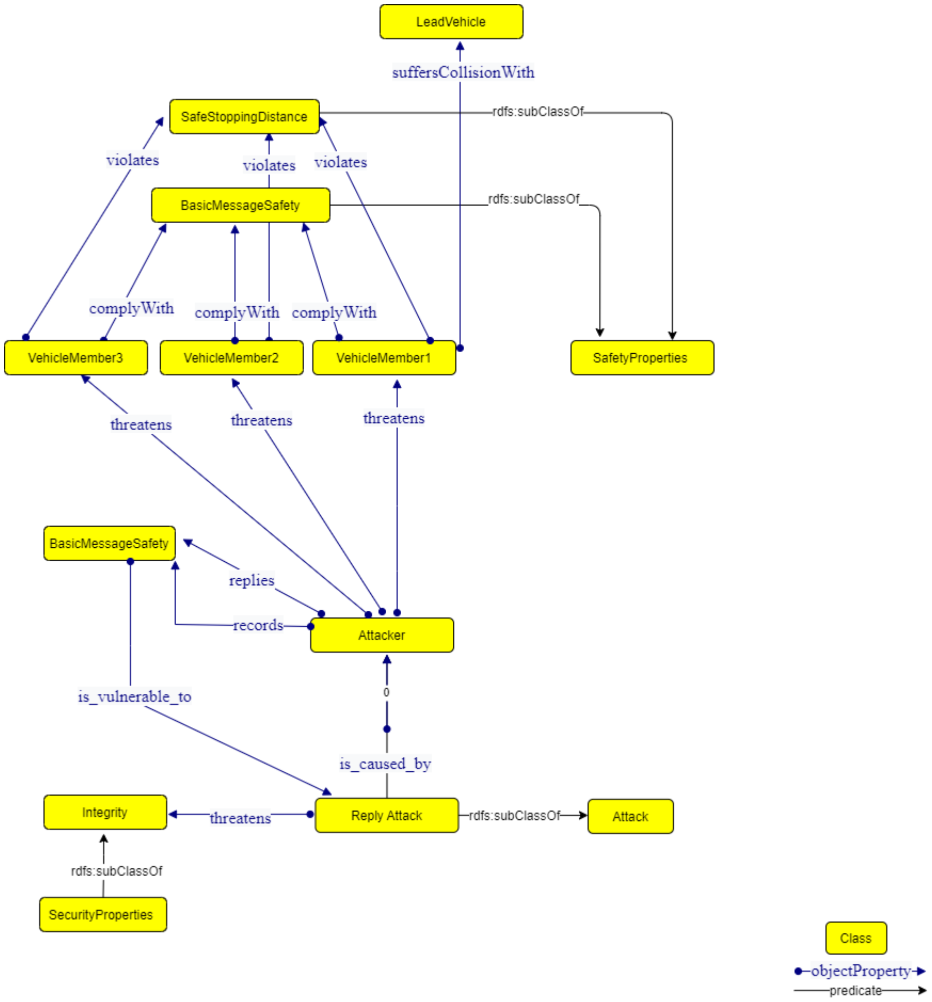


Figure 6.10: Formalisation of SSIoV ontology: concepts and relationships of the second running example with Graffoo[2]

It shows a portion of the ontology corresponding to our running example. The figure represents a *reply attack* propagated by an *attacker* against a *member vehicle*. The figure contains the main concepts and relationships shown in Table 6.1: Lead Vehicle; Member Vehicles are *Assets* [66]; Reply attack is an *Attack*; Integrity is a *Security Property*; Basic Message Safety and Safe Stopping Distance are *Safety Properties*; *Suffers Collision With* represents *Hazard*.

Chapter 7

II. Safety and Security Rules in SSIoV ontology

7.1 Drawing up Rules

The second step in our methodology is to model axioms and rules for safety risks and security vulnerabilities. The goal is to identify and formulate logical rules to connect security and safety aspects using ontology classes.

Our starting point for writing the rules is to consider the causal relationship between security and safety issues. We note that security attacks can have multiple impacts on AV safety. For example: a) an attack attempt can result in a security attack against an AV; b) this security attack can then cause that the AV complies with a safety rule. The AV complies with the minimum safe distance rule if its safe distance with the following AV is not maintained; c) then, the security attack can cause safety consequences when the AV violates the safety rule, it will cause some impacts in terms of safety (e.g. not keeping the minimum safety distance); d) finally, the security attack may lead to hazardous events due to the collision of the AV.

Each of these events can happen separately or at the same time. Moreover, each of these events has a causal relationship with the security attack as shown in Figure 7.1.

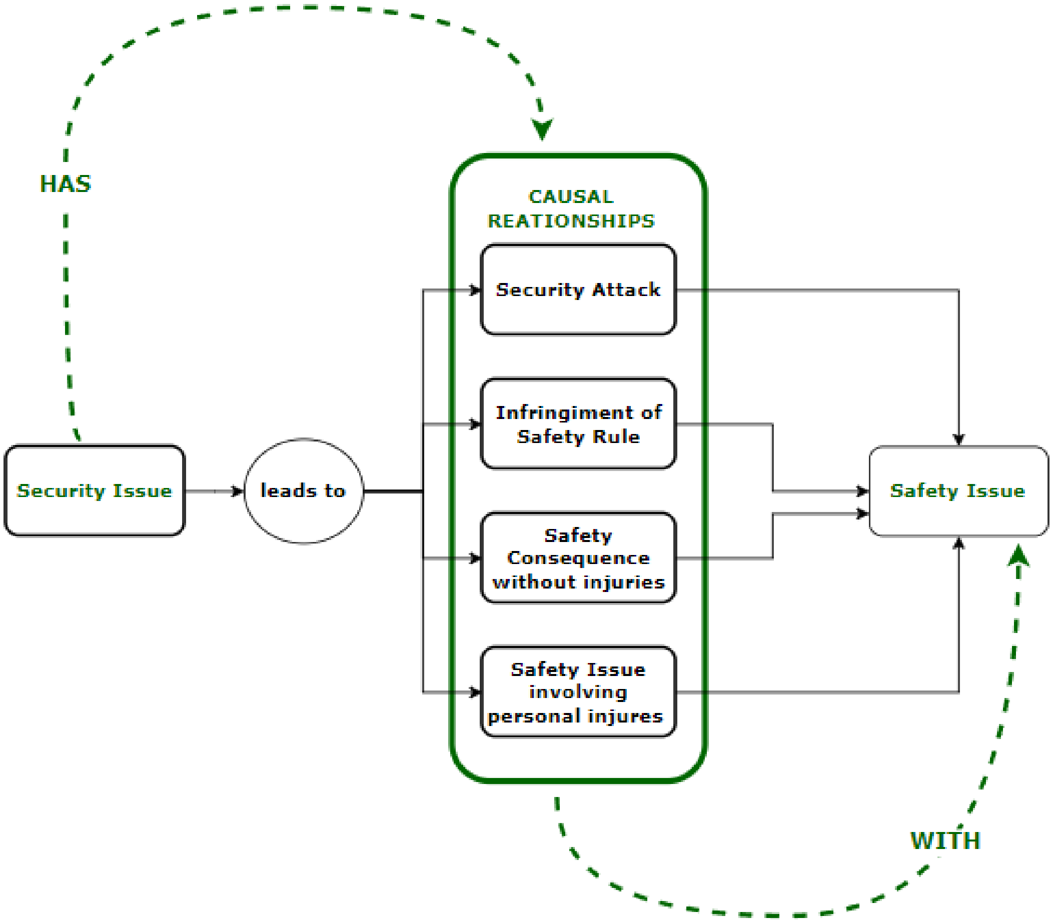

HAS
CAUSAL REATIONSHIPS
Security Attack
Infringiment of Safety Rule
Safety Consequence without injuries
Safety Issue involving personal injures
Security Issue
leads to
Safety Issue
WITH

Figure 7.1: The structure of causal relationships between safety and security issues

We consider the use case n.1 6.1.6, where: An attacker sends a falsified GPS signal [=**cyber attack event**] of its own position to the target vehicle. The spoofing attack here threatens the authenticity of the sensor's signal. The GPS (falsely) mentions that the position of the attacker is very close to the position of the target vehicle. The latter then applies a safety measure *(emergency stopping manoeuvre)* to ensure the safety property *(safe stopping distance)* [=**safety events**] that leads to a rear-end collision with the rear vehicle [=**the causal event**]".

To reason about security breach and its causal impact on safety, we model four types of inference rules:

- **Security reasoning rules**: we identify and model security rules describing security vulnerabilities *(An attacker generates a fake GPS signal and transmits it to the target vehicle)*;
- **Safety reasoning rules**: we model rules describing safety behaviour applied by vehicles when they detect safety risks (*The target vehicle takes an emergency stopping manoeuvre, slowing down suddenly, because the signal shows that the safety distance with the vehicle ahead is not maintained*);
- **Safety risks reasoning rules**: we model these rules by which safety actions lead to safety risks *(*The target vehicle makes an emergency stop, suddenly decelerates, and no longer maintain the minimum safe stopping distance from the rear vehicle),
- **Security-safety causal relationship reasoning rules**: we combine the three above types of reasoning rules to create security-safety rules that express the causal relationships from security to safety *(The target AV suffers a rear-end collision with the rear vehicle due to the GPS spoofing attack*).

Table 7.2 defines the safety and security events for our running example:

Rule Type	**Event**	**Explanation**
Security-Breach	Security Breach (attack)	An attacker generates falsified GPS signal and transmits it to the target vehicle
Safety-Rule	Safety Rule (trigger)	The target vehicle takes an emergency stopping manoeuvre - slowing down suddenly
Safety-Cons	Safety Consequence (fact)	The minimum safe stopping distance with the rear vehicle is not maintained
Safety-Issue	Safety Issue (Consequence)	The target AV suffers a rear-end collision with the vehicle behind it

Table 7.1: Causal events between security and safety - use case n.1

Based on Table 7.2, the reasoning then leads to the following conclusions for our running example:

$$\textbf{Security-Breach} \Rightarrow \textbf{Safety-Rule}$$
$$\textbf{Safety-Rule} \Rightarrow \textbf{Safety-Cons}$$
$$\textbf{Safety-Cons} \Rightarrow \textbf{Safety-Issue}$$

A spoofing attack (**Security-Breach**) uses a fake GPS and eventually leads to a rear-end collision (**Safety-Issue**). We conclude that :

$$\textbf{Security-Breach} \Rightarrow \textbf{Safety-Issue}.$$

We convert the natural language into SWRL that is a standard rule language based on a *"combination of the OWL DL and OWL Lite sub-languages of the OWL with the Unary/Binary Datalog RuleML sub-languages of the Rule Markup Language (RuleML)"* [69]. The syntax of SWRL rules is the following: **antecedent ("if")** and **consequent ("then")**. *"The rules are in the form of an implication between an antecedent (body) and consequent (head). The intended meaning can be read as: whenever the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold"* [69] .

Rule Type	Event Encoded with SWRL Rule
Security-Breach (SecBreach)	`ssiov:Attacker (?a) ∧ ssiov:TargetVehicle (?v) ∧ ssiov:falsifyGPSof (?a, ?v) ∧ ssiov:transmitFalsifiedGPSTo (?a, ?v) → ssiov:doGPSSpoofingAttackAgainst (?a, ?v) ∧ ssiov:sufferGPSSpoofingAttackBy(?v, ?a)`
Safety-Rule (SafeRule)	`ssiov:TargetVehicle (?v) ∧ ssiov:safeFollowingDistance (?v, ?sfd) ∧ swrlb:lessThan (?sfd, 30 xsd:int) ∧ ssiov:EmergencyStoppingManeuver (?esm) → ssiov:take (?v, ?esm)`
Safety-Consequences (SafeConseq)	`ssiov:Attacker(?a) ∧ ssiov:TargetVehicle(?v) ∧ ssiov:sufferGPSSpoofingAttackBy(?v, ?a) ∧ ssiov:SafeDistanceToRearVehicle(?sdr) ∧ ssiov:backSafetyDistance(?v, ?bsd) ∧ swrlb:lessThanOrEqual(?bsd, 30 xsd:int) ∧ ssiov:safeFollowingDistance(?v, ?sfd) ∧ swrlb:lessThanOrEqual(?sfd, 30 xsd:int) → ssiov:violates(?v, ?sdr)`
Safety-Issue (SafeIssue)	`ssiov:TargetVehicle(?v) ∧ ssiov:sufferGPSSpoofingAttackBy(?v, ?a) ∧ ssiov:SafeDistanceToRearVehicle(?sdr) ∧ ssiov:violates(?v, ?sdr) ∧ ssiov:RearVehicle(?rv) → ssiov:sufferRearEndCollisionWith(?v, ?rv)`

Table 7.2: A part of the SWRL security and safety rules encoded in the SSIoV ontology for the use case n.1 (6.1.6)

We focus on the use case n.2 (6.1.6). In a *platoon* of AVs, maintaining a uniform velocity and a constant distance is a safe requirement for platoon stability. A *platoon* of AVs travels at a speed of 25 m/s and at a distance gap of 15 m on a single lane. The *lead vehicle* “a” communicates to the *member vehicle* “b” behind it to close the gap at “t” time. After a few seconds (at time “t+1”), the leader transmits a signal to the vehicle “b” behind it to back off a little. An *attacker* recorded the message transmitted at time “t” and replayed it at time “t+2”, after the leader asked member vehicle “b” to back off. Member vehicle “b” will now discount the previous message and instead tries to close the gap. If this is done repeatedly, then by replaying old messages, the attacker will cause the platoon to oscillate as members try to position themselves in the best positions based on the information received. This event can lead to discomfort for the passenger and vehicle *collisions*

Table 7.3 defines the following events that are related to safety and security for our running example:

Rule Type	**Event**	**Explanation**
Security-Breach	Security breach (Attack)	The attacker records the message transmitted at time "t" and replays it at time "t+2", this is after the leader asks member vehicle to back off and the member vehicle "b" tries to close the gap
Safety-Rule	Safety rule (Trigger)	The member vehicle "b" discounts the previous message and instead, prompted by the delayed message, seeks to close the gap
Safety-Cons	Safety rule consequence (Fact)	Member vehicle "b" does not maintain a safe distance from the lead vehicle. The same safety fact occurs for the other member vehicles, which try to position themselves into the best positions based on the information they receive
Safety-Issue	Safety issue (Consequence)	The platoon dissolution causes member vehicle "b" to collide with the lead vehicle "a", because the member vehicle "b" violates the minimum safe distance from the following vehicle

Table 7.3: Causal events between security and safety - use case n.2

Based on Table 7.3, the reasoning then leads to the following conclusions for our running example:

$$\begin{aligned}\textbf{Security-Breach} &\Rightarrow \textbf{Safety-Rule}\\ \textbf{Safety-Rule} &\Rightarrow \textbf{Safety-Cons}\\ \textbf{Safety-Cons} &\Rightarrow \textbf{Safety-Issue}\end{aligned}$$

A spoofing attack (**Security-Breach**) uses a fake GPS and eventually leads to a rear-end collision (**Safety-Issue**). We conclude that:

$$\textbf{Security-Breach} \Rightarrow \textbf{Safety-Issue}.$$

We convert the natural language into SWRL as shown in Table 7.4.

Rule Type	Event Encoded with SWRL Rule
Security-Breach (SecBreach)	ssiov:Attacker (?a) ∧ ssiov:BMS (?bms) ∧ ssiov:record (?a, ?obms) ∧ ssiov:Vehicle124 (?v) ∧ transmitBMSTo (?a,?v) → doReplyAttackAgainst (?a, ?v)
Safety-Rule (SafeRule)	ssiov:Vehicle124 (?v1) ∧ ssiov:LeadVehicle (?lv) ∧ ssiov:receiveBMSBy (?v,?lv) ∧ ssiov:Vehicle125 (?v2) ∧ ssiov:transmitBMSTo (?v1, ?v2) → ssiov:complyWithOldBMSof (?v1, ?lv)
Safety-Consequences (SafeConseq)	ssiov:Vehicle124 (?v1) ∧ ssiov:ReplyAttack (?ra) ∧ ssiov:suffer (?v, ?ra) ∧ ssiov:safeFollowingDistance (?v1, ?sfd) ∧ swrlb:lessThan (?sfd, 30 xsd:int) ∧ ssiov:Vehicle125 (?v2) ∧ ssiov:Vehicle126 (?v3) ∧ ssiov:safeFollowingDistance (?v2, ?sfd) ∧ swrlb:lessThan (?sfd, 30 xsd:int) ∧ ssiov:Vehicle126 (?v3) ∧ ssiov:safeFollowingDistance (?v3, ?sfd) ∧ swrlb:lessThan (?sfd, "30" xsd:int) → ssiov:violates (?v1, ssiov:minimumSafeDistance) ∧ ssiov:violates (?v1,ssiov:minimumSafeDistance) ∧ ssiov:violates (?v1,ssiov:minimumSafeDistance)
Safety-Issue (SafeIssue)	ssiov:Vehicle124 (?v) ∧ ssiov:ReplyAttack (?sa) ∧ ssiov:suffer (?v, ?sa) ∧ ssiov:violates (?v, ssiov:minimumSafeDistance) → ssiov:sufferCollisionWith (?v, ssiov:leadVehicle)

Table 7.4: A part of the SWRL security and safety rules that are encoded in the SSIoV ontology for the use case n.1 6.1.6

.

We identify security and safety rules separately by applying research results of analytical methods used by the engineers, such as STPA-SafeSec to combine both safety and security analysis [58], [70], [71]. We also use the research that applies the TARA, HARA and other analysis methods.

7.1.1 Security-Breach Rules

Let us look at some examples from Chowdhury et al. [72] who study the attacks against AVs. The researchers provide a broad overview about potential cyber attacks, their impact on these vehicles and their vulnerabilities.

We enumerate some scenario as follows:

SecBreach1: The *malware attack* consists in manipulating the *"radio of the vehicle using a Bluetooth stack weakness and inserting the malware codes by syncing their mobile phones with the radio. The inserted code could send messages to the ECU of the vehicle that could lock the brakes"* [72].

```
ssiov:Attacker(?a) ∧ ssiov:Vehicle(?v) ∧ ssiov:manipulate(?a, ?v) ∧
ssiov:injectFalseMessage(?a, ?v) → ssiov:doMalwareAttackAgainst(?a,
                                   ?v)
```

The rule asserts that if an attacker manipulates the radio of the vehicle by sending a message to the vehicle, then the vehicle suffers a malware attack. The terms *manipulate* and *injectFalseMessage* doMalwareAttackAgainst are object properties, while *Attacker* and *Vehicle* are classes.

Table 7.5 presents some examples of security breach rules and their explanation.

Table 7.5: Security Breach Rules

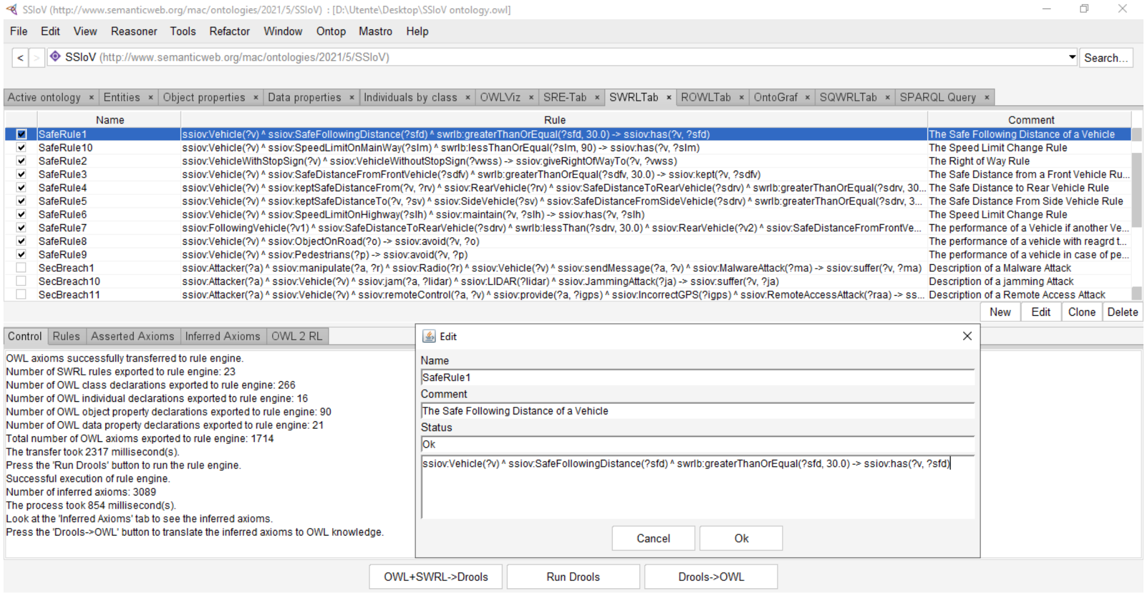
Rule Type	**Event Explanation**
SecBreach1	The malware attack consists in manipulating the radio of the vehicle using a Bluetooth stack weakness and inserting the malware codes by syncing their mobile phones with the radio. The inserted code could send messages to the ECU of the vehicle,*then* these messages could lock the brakes [72]
SecBreach2	*If* an attacker manipulates the communication messages between two entities (while both entities believe that they are in direct communication with each other), by controlling OBUs or RSUs, eavesdropping, replaying, and modifying their CAN messages (that regulate the steering, brakes, and vehicle acceleration), *then* the vehicle suffer a man-in-the-middle attack [72]
SecBreach3	*If* an attacker encrypts personal media repository, communication logs, freight monitoring logs, important control parameters in self-driving cars, *then* the vehicle suffers a ransomware attack [72]

Continued on next page

Table7.5 – continued from previous page

Rule Type	Event Explanation
SecBreach4	*If* an attacker send signals shot to the LIDAR at the nanosecond level, and the Lidar of the vehicle believes that there was an object in front of the vehicle, *then* the vehicle suffers a spoofing attack [72]
SecBreach5	*If* an attacker sends misleading location and traffic information to the vehicle, which comes with an incorrect GPS location, *then* the vehicle suffers a Sybil attack [72]
SecBreach6	*If* an attacker uses the OBD port to insert a malicious program into the AV software, *then* the vehicle will suffer an Attack on its software [72]
SecBreach7	*If* an attacker jams the GPS component of the vehicle, *then* the vehicle is attacked on its OBUs [72]
SecBreach8	*If* an attacker personifies a speed control sensor, *then* the vehicle is subject to an Attack on Speed Control Sensor [72]
SecBreach9	*If* an attacker remotely control the vehicle via remote access and gives it the wrong GPS, *then* the vehicle suffers a remote access attack [72]
SecBreach10	*If* an attacker jams the Lidar of the vehicle and this last cannot receive sensitive information and uses network services, *then* the vehicle is subject to a Jamming Attack [72]
SecBreach11	*If* an attacker injects false message to the vehicle, *then* the vehicle suffers a man-in-the-middle attack [72]
SecBreach12	*If* an attacker manipulates message and the vehicle receives incorrect traffic report, *then* the vehicle is subject to a spoofing attack [72]
SecBreach13	*If* an attacker can gain unauthorised access on key-less entry system, *then* the vehicle would be vulnerable to eavesdropping attack [72]
SecBreach14 Use case n.1	*If* an attacker generates fake GPS signal and transmits it to the target vehicle [21], *then* the vehicle suffers a GPS spoofing attack
SecBreach15 Use case n.2	*If* an attacker records the (transmitted by the lead vehicle to the first member vehicle of a platoon, at time X), and then the attacker replies the old to the first member vehicle at time Y, *then* the first member vehicle suffers a reply attack [67]

Figure 7.2 shows the Security Breach Rules in SWRLTab.



	Name	Rule	Comment
☑	SafeRule1	ssiov:Vehicle(?v) ^ ssiov:SafeFollowingDistance(?sfd) ^ swrlb:greaterThanOrEqual(?sfd, 30.0) -> ssiov:has(?v, ?sfd)	The Safe Following Distance of a Vehicle
☑	SafeRule10	ssiov:Vehicle(?v) ^ ssiov:SpeedLimitOnMainWay(?slm) ^ swrlb:lessThanOrEqual(?slm, 90) -> ssiov:has(?v, ?slm)	The Speed Limit Change Rule
☑	SafeRule2	ssiov:VehicleWithStopSign(?v) ^ ssiov:VehicleWithoutStopSign(?vwss) -> ssiov:giveRightOfWayTo(?v, ?vwss)	The Right of Way Rule
☑	SafeRule3	ssiov:Vehicle(?v) ^ ssiov:SafeDistanceFromFrontVehicle(?sdfv) ^ swrlb:greaterThanOrEqual(?sdfv, 30.0) -> ssiov:kept(?v, ?sdfv)	The Safe Distance from a Front Vehicle Ru...
☑	SafeRule4	ssiov:Vehicle(?v) ^ ssiov:keptSafeDistanceFrom(?v, ?rv) ^ ssiov:RearVehicle(?rv) ^ ssiov:SafeDistanceToRearVehicle(?sdrv) ^ swrlb:greaterThanOrEqual(?sdrv, 30...	The Safe Distance to Rear Vehicle Rule
☑	SafeRule5	ssiov:Vehicle(?v) ^ ssiov:keptSafeDistanceTo(?v, ?sv) ^ ssiov:SideVehicle(?sv) ^ ssiov:SafeDistanceFromSideVehicle(?sdrv) ^ swrlb:greaterThanOrEqual(?sdrv, 3...	The Safe Distance From Side Vehicle Rule
☑	SafeRule6	ssiov:Vehicle(?v) ^ ssiov:SpeedLimitOnHighway(?slh) ^ ssiov:maintain(?v, ?slh) -> ssiov:has(?v, ?slh)	The Speed Limit Change Rule
☑	SafeRule7	ssiov:FollowingVehicle(?v1) ^ ssiov:SafeDistanceToRearVehicle(?sdrv) ^ swrlb:lessThan(?sdrv, 30.0) ^ ssiov:RearVehicle(?v2) ^ ssiov:SafeDistanceFromFrontVe...	The performance of a Vehicle if another Ve...
☑	SafeRule8	ssiov:Vehicle(?v) ^ ssiov:ObjectOnRoad(?o) -> ssiov:avoid(?v, ?o)	The performance of a vehicle with reagrd t...
☑	SafeRule9	ssiov:Vehicle(?v) ^ ssiov:Pedestrians(?p) -> ssiov:avoid(?v, ?p)	The performance of a vehicle in case of pe...
☐	SecBreach1	ssiov:Attacker(?a) ^ ssiov:manipulate(?a, ?r) ^ ssiov:Radio(?r) ^ ssiov:Vehicle(?v) ^ ssiov:sendMessage(?a, ?v) ^ ssiov:MalwareAttack(?ma) -> ssiov:suffer(?v, ?ma)	Description of a Malware Attack
☐	SecBreach10	ssiov:Attacker(?a) ^ ssiov:Vehicle(?v) ^ ssiov:jam(?a, ?lidar) ^ ssiov:LIDAR(?lidar) ^ ssiov:JammingAttack(?ja) -> ssiov:suffer(?v, ?ja)	Description of a jamming Attack
☐	SecBreach11	ssiov:Attacker(?a) ^ ssiov:Vehicle(?v) ^ ssiov:remoteControl(?a, ?v) ^ ssiov:provide(?a, ?igps) ^ ssiov:IncorrectGPS(?igps) ^ ssiov:RemoteAccessAttack(?raa) -> ss...	Description of a Remote Access Attack

```
OWL axioms successfully transferred to rule engine.
Number of SWRL rules exported to rule engine: 23
Number of OWL class declarations exported to rule engine: 266
Number of OWL individual declarations exported to rule engine: 16
Number of OWL object property declarations exported to rule engine: 90
Number of OWL data property declarations exported to rule engine: 21
Total number of OWL axioms exported to rule engine: 1714
The transfer took 2317 millisecond(s).
Press the 'Run Drools' button to run the rule engine.
Successful execution of rule engine.
Number of inferred axioms: 3089
The process took 854 millisecond(s).
Look at the 'Inferred Axioms' tab to see the inferred axioms.
Press the 'Drools->OWL' button to translate the inferred axioms to OWL knowledge.
```

Edit

Name: SafeRule1

Comment: The Safe Following Distance of a Vehicle

Status: Ok

```
ssiov:Vehicle(?v) ^ ssiov:SafeFollowingDistance(?sfd) ^ swrlb:greaterThanOrEqual(?sfd, 30.0) -> ssiov:has(?v, ?sfd)
```

Figure 7.2: Security Breach Rules in SWRLTab

7.1.2 Safety Rules

There are various low-level risk analyses such as HARA, FMEA, FTA, and HAZOP in order to formulate safety requirements that can lead to a safer system design for AVs [73].

Based on the results of these studies, we look at some examples of safety recommendations for AVs to ensure their security en route. As we discuss in 3.2, there is no standard for AV safety measures. Therefore, we assume a safe distance of 30 m between two vehicles under normal traffic and weather conditions.

SafeRule1: AV must maintain a *Safe Distance* from a *Forward vehicle* [21]

```
ssiov:Vehicle (?v) ∧ ssiov:safeFollowingDistance (?v, ?sfd) ∧
        swrlb:greaterThanOrEqual (?sfd, "30" xsd:int) →
              ssiov:safeFollowingDistance (?v, true)
```

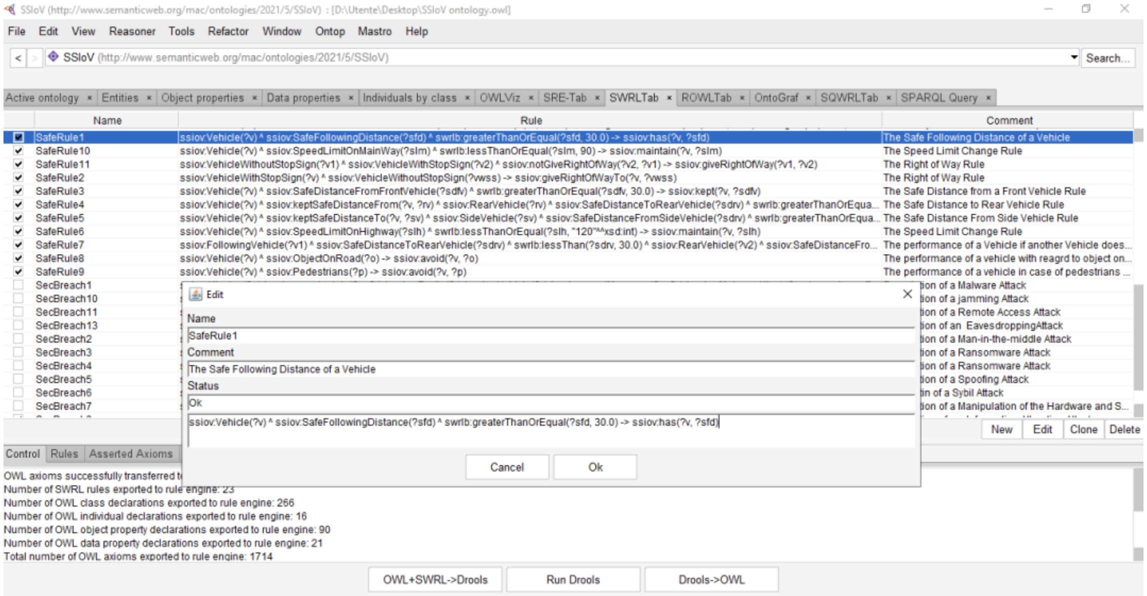
The following rule asserts that a vehicle must maintain a safe distance from the forward vehicle. The terms *hasSafeDistanceFrom* is an object property, *Vehicle* and *ForwardVehicle* are classes, *safeDistance* is a data property, *ForwardVehicle* is a class and *safeDistance* is a data property. Then, we assume the minimum value of the safe distance is about 30 m.

Table 7.6 presents some examples of the safety rules and their explanation.

Table 7.6: Safe Rules

Rule Type	**Event Explanation**
SafeRule1	AV must maintain a *Safe Distance* from the *Forward vehicle* [21]
SafeRule2	On a T-junction with a stop sign AV must give the right of way to vehicles without a stop sign
SafeRule3	AV must keep safe distance to front vehicle below the minimum value [74]
SafeRule4	AV must keep Safe distance to rear vehicle below the minimum value [74]
SafeRule5	AV must keep Safe distance to side vehicle below the minimum value [74]
SafeRule6	AV must maintain the Speed Limit Changes [73]
SafeRule7	*If* a following vehicle slows down and the safe distance with the rear vehicle decreases, *then* the rear vehicle must slow down to restore the safe distance with the following vehicle
SafeRule8	AV must detect and respond to Static Obstacles in the Vehicle's Path [73]
SafeRule9	AV must detect and respond to Pedestrians on the Road (Not Walking Through Intersection or Crosswalk) [73]
SafeRule10	AV must maintain the Speed Limit Changes [73]
SafeRule11	On a T-junction without a stop sign, *if* an AV with stop sign does not respect the right to way, *then* the AV, which has not a stop sign, must give the right of way to AV that does not respect the right to way
SafeRule12 Use case n.1	AV must take an emergency stopping manoeuvre - slowing down suddenly - *if* the following distance with the ahead vehicle is not maintained [21]
SafeRule13 Use case n.2	The first member vehicle of a platoon must discount the previous transmitted by the lead vehicle, and complies with the latest [67]

Figure 7.4 shows Safety Rules in SWRLTab.



	Name	Rule	Comment
☑	SafeRule1	ssiov:Vehicle(?v) ^ ssiov:SafeFollowingDistance(?sfd) ^ swrlb:greaterThanOrEqual(?sfd, 30.0) -> ssiov:has(?v, ?sfd)	The Safe Following Distance of a Vehicle
☑	SafeRule10	ssiov:Vehicle(?v) ^ ssiov:SpeedLimitOnMainWay(?slm) ^ swrlb:lessThanOrEqual(?slm, 90) -> ssiov:maintain(?v, ?slm)	The Speed Limit Change Rule
☑	SafeRule11	ssiov:VehicleWithoutStopSign(?v1) ^ ssiov:VehicleWithStopSign(?v2) ^ ssiov:notGiveRightOfWay(?v2, ?v1) -> ssiov:giveRightOfWay(?v1, ?v2)	The Right of Way Rule
☑	SafeRule2	ssiov:VehicleWithStopSign(?v) ^ ssiov:VehicleWithoutStopSign(?vwss) -> ssiov:giveRightOfWayTo(?v, ?vwss)	The Right of Way Rule
☑	SafeRule3	ssiov:Vehicle(?v) ^ ssiov:SafeDistanceFromFrontVehicle(?sdfv) ^ swrlb:greaterThanOrEqual(?sdfv, 30.0) -> ssiov:kept(?v, ?sdfv)	The Safe Distance from a Front Vehicle Rule
☑	SafeRule4	ssiov:Vehicle(?v) ^ ssiov:keptSafeDistanceFrom(?v, ?rv) ^ ssiov:RearVehicle(?rv) ^ ssiov:SafeDistanceToRearVehicle(?sdrv) ^ swrlb:greaterThanOrEqua...	The Safe Distance to Rear Vehicle Rule
☑	SafeRule5	ssiov:Vehicle(?v) ^ ssiov:keptSafeDistanceTo(?v, ?sv) ^ ssiov:SideVehicle(?sv) ^ ssiov:SafeDistanceFromSideVehicle(?sdrv) ^ swrlb:greaterThanOrEqua...	The Safe Distance From Side Vehicle Rule
☑	SafeRule6	ssiov:Vehicle(?v) ^ ssiov:SpeedLimitOnHighway(?slh) ^ swrlb:lessThanOrEqual(?slh, "120"^^xsd:int) -> ssiov:maintain(?v, ?slh)	The Speed Limit Change Rule
☑	SafeRule7	ssiov:FollowingVehicle(?v1) ^ ssiov:SafeDistanceToRearVehicle(?sdrv) ^ swrlb:lessThan(?sdrv, 30.0) ^ ssiov:RearVehicle(?v2) ^ ssiov:SafeDistanceFro...	The performance of a Vehicle if another Vehicle does...
☑	SafeRule8	ssiov:Vehicle(?v) ^ ssiov:ObjectOnRoad(?o) -> ssiov:avoid(?v, ?o)	The performance of a vehicle with reagrd to object on...
☑	SafeRule9	ssiov:Vehicle(?v) ^ ssiov:Pedestrians(?p) -> ssiov:avoid(?v, ?p)	The performance of a vehicle in case of pedestrians ...
☐	SecBreach1		ion of a Malware Attack
☐	SecBreach10		ion of a jamming Attack
☐	SecBreach11		ion of a Remote Access Attack
☐	SecBreach13		ion of an EavesdroppingAttack
☐	SecBreach2		ion of a Man-in-the-middle Attack
☐	SecBreach3		ion of a Ransomware Attack
☐	SecBreach4		ion of a Ransomware Attack
☐	SecBreach5		ion of a Spoofing Attack
☐	SecBreach6		in of a Sybil Attack
☐	SecBreach7		ion of a Manipulation of the Hardware and S...

Edit

Name

SafeRule1

Comment

The Safe Following Distance of a Vehicle

Status

Ok

ssiov:Vehicle(?v) ^ ssiov:SafeFollowingDistance(?sfd) ^ swrlb:greaterThanOrEqual(?sfd, 30.0) -> ssiov:has(?v, ?sfd)

```
OWL axioms successfully transferred t
Number of SWRL rules exported to rule engine: 23
Number of OWL class declarations exported to rule engine: 266
Number of OWL individual declarations exported to rule engine: 16
Number of OWL object property declarations exported to rule engine: 90
Number of OWL data property declarations exported to rule engine: 21
Total number of OWL axioms exported to rule engine: 1714
```

Figure 7.3: Safety Rules in SWRLTab

7.1.3 Safety-Consequences Rules

SafeConseq1: <u>*If*</u> an attacker manipulates the communication messages between two entities (while both entities believe that they are in direct communication with each other), by controlling OBUs or RSUs and eavesdropping, replaying, and modifying their CAN messages (that regulate the steering, brakes, and vehicle acceleration), <u>*then*</u> vehicles suffer a man-in-the-middle attack, <u>*and then*</u> AVs receive false messages [72].

```
Vehicle (?v1) ∧ Vehicle (?v2) ∧ sendCANMessage (?v1, ?v2) ∧
Attacker (?a) ∧ manipulate (?a, "CANMessage") → injectFalseMessage
(?a, ?v1) ∧ injectFalseMessage (?a, ?v2)
```

The above rule asserts that an *Attacker* manipulates CAN messages between two vehicles by injecting False Messages. The terms *manipulate*, *sendCANMessage*, and *injectFalseMessage* are object properties. The *Attacker*, *Vehicle*, and *CANMessage* are classes. We assume that the minimum value of the safe distance is about 30 m.

Table 7.7 presents some examples of Safety Consequences Rules and their explanation.

Table 7.7: Safety Consequences Rules

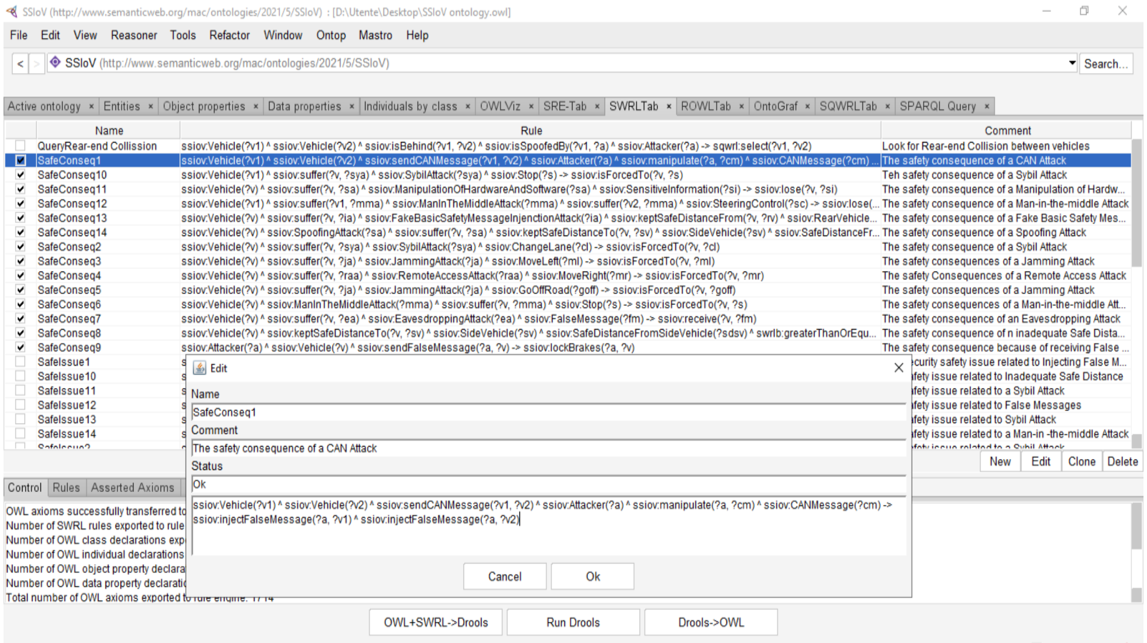
Rule Type	Event Explanation
SafeConseq1	*If* an attacker manipulates the communication messages between two entities (while both entities believe that they are in direct communication with each other), by controlling OBUs or RSUs, and eavesdropping, replaying, and modifying their CAN messages (that regulate the steering, brakes, and vehicle acceleration), *then* vehicles suffer a man-in-the-middle attack, *and then* the AVs receive false messages [72]
SafeConseq2	*If* an attacker sends misleading location and traffic information to the vehicle, the latter comes with an incorrect GPS location, *then* the vehicle suffers a sybil attack [72], *and then* the vehicle is forced to change lane; move left or right; go off-road and make a sudden stop
SafeConseq3	*If* an attacker jams the GPS component of the vehicle, *then* the vehicle suffers an Attack on its OBUs [72] that consists in injected false message, *and then* the vehicle is forced to change lane; move left or right; go off-road, and make a sudden stop
SafeConseq4	*If* an attacker remotely controls the vehicle via remote access and gives it the wrong GPS, *then* the vehicle suffers a remote access attack *and then* AV is forced to change lane; move left or right; go off-road, and make a sudden stop [72]
SafeConseq5	*If* an attacker jams the LIDAR of the vehicle and this latter cannot receive sensitive information and uses network services, *then* the vehicle suffers a jamming attack [72], *and then* AV is forced to change lane; move left or right; go off-road, and make a sudden stop
SafeConseq6	*If* an attacker injects the false message, *then* the vehicle suffers a man-in-the-middle attack [72], *and then* AV is forced to change lane; move left or right; go off-road, and make a sudden stop
SafeConseq7	*If* an attacker enables gain unauthorised access on keyless entry system, *then* the vehicle is eavesdropped, *and then* AV receives a false message [72]
SafeConseq8	*If* an AV does not maintain a safe distance to the vehicle ahead below the minimum value, *then* the distance to the ahead vehicle is inadequate [74]

Continued on next page

Table7.7 – continued from previous page

Rule Type	Event Explanation
SafeConseq9	*If* a malware attack manipulates the radio of the vehicle by using a Bluetooth stack weakness, and inserts the malware codes by syncing their mobile phones with the radio, *then* the inserted code could send messages to the ECU of the vehicle that could lock the brakes [72]
SafeConseq10	*If* an attacker sends non-encrypted messages to the AV, and fake nodes are used to send misleading location and traffic condition information to the AV, *then* the AV suffers a sybil attack, *and then* the AV stops in the middle of the road [72]
SafeConseq11	*If* an attacker uses OBD port to insert the malware program into the AV software, *then* the vehicle will suffer an Attack on its software [72], *and then* sensitive information will be leaked
SafeConseq12	*If* an attacker manipulates the communication messages between two entities (while both entities believe that they are in direct communication with each other), by controlling OBUs or RSUs and eavesdropping, replaying, and modifying their CAN messages (that regulate the steering, brakes, and vehicle acceleration), *then* the vehicle suffers a man-in-the-middle attack [72], *and then* vehicle loses the steering control
SafeConseq13	*If* an AV does not keep the Safe Distance to the rear vehicle below the minimum value due to a Fake Basic Safety Message Injection Attack, *then* [74] the distance to rear vehicle is not enough
SafeConseq14	*If* an AV does not keep the Safe distance to side vehicle below the minimum value due to a Fake Basic Safety Message Injection Attack, *then* the distance to side vehicle is insufficient [74]
SafeConseq15 Use case n.1	*If* an AV does not maintain the minimum safe stopping distance with the rear vehicle due to a GPS spoofing attack, *then* the AV violates the minimum safe distance [21]
SafeConseq16 Use case n.2	*If* the first member vehicle does not maintain the safe following distance with the lead vehicle, and the other vehicles of a platoon do not maintain their safe following distance with the others, *then* the AVs of the platoon violate the minimum safe distance [67]

Figure 7.4 shows Safety Consequences Rule in SWRLTab



SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]

	Name	Rule	Comment
□	QueryRear-end Collission	ssiov:Vehicle(?v1) ^ ssiov:Vehicle(?v2) ^ ssiov:isBehind(?v1, ?v2) ^ ssiov:isSpoofedBy(?v1, ?a) ^ ssiov:Attacker(?a) -> sqwrl:select(?v1, ?v2)	Look for Rear-end Collision between vehicles
☑	SafeConseq1	ssiov:Vehicle(?v1) ^ ssiov:Vehicle(?v2) ^ ssiov:sendCANMessage(?v1, ?v2) ^ ssiov:Attacker(?a) ^ ssiov:manipulate(?a, ?cm) ^ ssiov:CANMessage(?cm) ...	The safety consequence of a CAN Attack
☑	SafeConseq10	ssiov:Vehicle(?v1) ^ ssiov:suffer(?v, ?sya) ^ ssiov:SybilAttack(?sya) ^ ssiov:Stop(?s) -> ssiov:isForcedTo(?v, ?s)	Teh safety consequence of a Sybil Attack
☑	SafeConseq11	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?sa) ^ ssiov:ManipulationOfHardwareAndSoftware(?sa) ^ ssiov:SensitiveInformation(?si) -> ssiov:lose(?v, ?si)	The safety consequence of a Manipulation of Hardw...
☑	SafeConseq12	ssiov:Vehicle(?v1) ^ ssiov:suffer(?v1, ?mma) ^ ssiov:ManInTheMiddleAttack(?mma) ^ ssiov:suffer(?v2, ?mma) ^ ssiov:SteeringControl(?sc) -> ssiov:lose(...	The safety consequence of a Man-in-the-middle Attack
☑	SafeConseq13	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ia) ^ ssiov:FakeBasicSafetyMessageInjenctionAttack(?ia) ^ ssiov:keptSafeDistanceFrom(?v, ?rv) ^ ssiov:RearVehicle...	The safety consequence of a Fake Basic Safety Mes...
☑	SafeConseq14	ssiov:Vehicle(?v) ^ ssiov:SpoofingAttack(?sa) ^ ssiov:suffer(?v, ?sa) ^ ssiov:keptSafeDistanceTo(?v, ?sv) ^ ssiov:SideVehicle(?sv) ^ ssiov:SafeDistanceFr...	The safety consequence of a Spoofing Attack
☑	SafeConseq2	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?sya) ^ ssiov:SybilAttack(?sya) ^ ssiov:ChangeLane(?cl) -> ssiov:isForcedTo(?v, ?cl)	The safety consequence of a Sybil Attack
☑	SafeConseq3	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ja) ^ ssiov:JammingAttack(?ja) ^ ssiov:MoveLeft(?ml) -> ssiov:isForcedTo(?v, ?ml)	The safety consequences of a Jamming Attack
☑	SafeConseq4	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?raa) ^ ssiov:RemoteAccessAttack(?raa) ^ ssiov:MoveRight(?mr) -> ssiov:isForcedTo(?v, ?mr)	The safety Consequences of a Remote Access Attack
☑	SafeConseq5	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ja) ^ ssiov:JammingAttack(?ja) ^ ssiov:GoOffRoad(?goff) -> ssiov:isForcedTo(?v, ?goff)	The safety consequences of a Jamming Attack
☑	SafeConseq6	ssiov:Vehicle(?v) ^ ssiov:ManInTheMiddleAttack(?mma) ^ ssiov:suffer(?v, ?mma) ^ ssiov:Stop(?s) -> ssiov:isForcedTo(?v, ?s)	The safety consequences of a Man-in-the-middle Att...
☑	SafeConseq7	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ea) ^ ssiov:EavesdroppingAttack(?ea) ^ ssiov:FalseMessage(?fm) -> ssiov:receive(?v, ?fm)	The safety consequence of an Eavesdropping Attack
☑	SafeConseq8	ssiov:Vehicle(?v) ^ ssiov:keptSafeDistanceTo(?v, ?sv) ^ ssiov:SideVehicle(?sv) ^ ssiov:SafeDistanceFromSideVehicle(?sdsv) ^ swrlb:greaterThanOrEqu...	The safety consequence of n inadequate Safe Dista...
☑	SafeConseq9	ssiov:Attacker(?a) ^ ssiov:Vehicle(?v) ^ ssiov:sendFalseMessage(?a, ?v) -> ssiov:lockBrakes(?a, ?v)	The safety consequence because of receiving False ...
□	SafeIssue1	s	curity safety issue related to Injecting False M...
□	SafeIssue10	s	fety issue related to Inadequate Safe Distance
□	SafeIssue11	s	fety issue related to a Sybil Attack
□	SafeIssue12	s	fety issue related to False Messages
□	SafeIssue13	s	fety issue related to Sybil Attack
□	SafeIssue14	s	fety issue related to a Man-in -the-middle Attack
□	SafeIssue2	s	fety issue related to a Sybil Attack

Edit

Name: SafeConseq1

Comment: The safety consequence of a CAN Attack

Status: Ok

```
ssiov:Vehicle(?v1) ^ ssiov:Vehicle(?v2) ^ ssiov:sendCANMessage(?v1, ?v2) ^ ssiov:Attacker(?a) ^ ssiov:manipulate(?a, ?cm) ^ ssiov:CANMessage(?cm) ->
ssiov:injectFalseMessage(?a, ?v1) ^ ssiov:injectFalseMessage(?a, ?v2)
```

```
OWL axioms successfully transferred to
Number of SWRL rules exported to rule
Number of OWL class declarations exp
Number of OWL individual declarations
Number of OWL object property declara
Number of OWL data property declaratio
Total number of OWL axioms exported to rule engine: 1714
```

Figure 7.4: Safety Consequence Rules in SWRLTab

7.1.4 Safety Issues Rules

SafeIssue1: *If* an attacker manipulates communication messages between two entities (while both entities believe that they are in direct communication with each other), by controlling OBUs or RSUs and eavesdropping, replaying, and modifying their CAN messages (that regulate the steering, brakes, and vehicle acceleration), *then* vehicles suffer a man-in-the-middle attack, *and then* the AVs receive false messages [72], and this *causes* the collision with another AV on the other side.

```
ssiov:Vehicle(?v1) ∧ ssiov:Vehicle(?v2) ∧ ssiov:injectFalseMessage
(?a, ?v1) ∧ ssiov:injectFalseMessage(?a, ?v2) ∧ ssiov:Vehicle (?v3)
→ ssiov:collideWith (?v1, ?v3)
```

The above rule asserts that an *Attacker* manipulates CAN messages between two vehicles by injecting False Messages, and this manipulation causes the collision among two vehicles and another vehicle. The terms: *injectFalseMessage* and *collideWith* are object properties. The *Attacker* and *Vehicle* are classes.

Table 7.8 provides some examples of Safety Issues Rules and their explanation.

Table 7.8: Safety Issues Rules

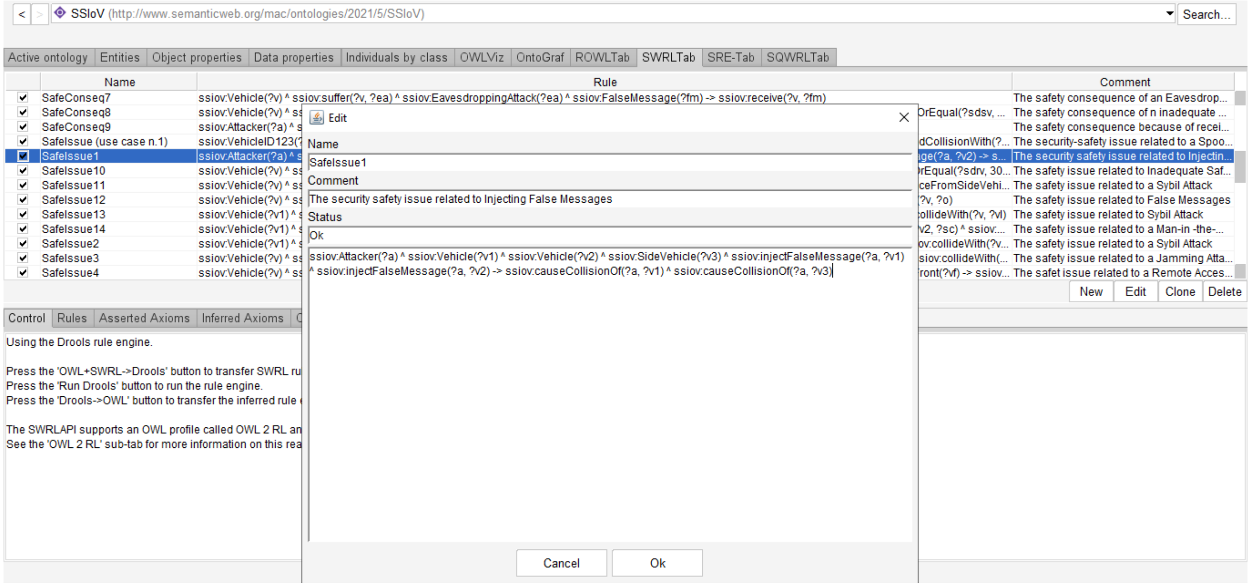
Rule Type	Event Explanation
SafeIssue1	*If* an attacker manipulates the communication messages between two entities (while both entities believe that they are in direct communication with each other), by controlling OBUs or RSUs and eavesdropping, replaying, and modifying their CAN messages (that regulate the steering, brakes, and vehicle acceleration), *then* the vehicle suffers a man-in-the-middle attack *and then* the AVs receive false messages [72] and this *causes* a collision with another vehicle
SafeIssue2	*If* an attacker sends misleading location and traffic information to the vehicle, the last comes with an incorrect GPS location, *then* the vehicle suffers a sybil attack [72], *and then* the vehicle is forced to change lane; move left or right; go off-road, and make a sudden stop. This event *causes* a crash hazard
SafeIssue3	*If* an attacker jams the GPS component of the vehicle, *then* the vehicle suffers an Attack on its OBUs [72] (that consists in injected false messages), *and then* the vehicle is forced to change lane; move left or right; go off-road, and make a sudden stop
SafeIssue4	*If* an attacker controls the vehicle via remote access and gives it the wrong GPS, *then* the vehicle suffers a remote access attack, *and then* the AV is forced to change lane; move left or right; go off-road, and make a sudden stop [72]
SafeIssue5	*If* an attacker jams the LIDAR of the vehicle and this last vehicle cannot receive sensitive information and uses network services, *then* the vehicle suffers a jamming attack [72], *and then* AV is forced to change lane; move left or right; go off-road, and make a sudden stop
SafeIssue6	*If* an attacker injects false message, *then* the vehicle suffers a man-in-the-middle attack [72] *and then* AV is forced
SafeIssue7	*If* an attacker manipulates message, and the vehicle receives the incorrect road condition report, *then* the vehicle suffers a spoofing attack [72] *and then* the vehicle is forced to change lane; move left or right; go off-road and come to an abrupt stop

Continued on next page

Table7.8 – continued from previous page

Rule Type	Event Explanation
SafeIssue8	*If* an attacker can gain unauthorised access on the key less entry system, *then* the vehicle will be subject to an eavesdropping attack, *and then* AV receives false message [72]
SafeIssue9	*If* an AV does not keep the Safe Distance to the front vehicle below the minimum value, *then* the distance to the frontal vehicle is inadequate [74]
SafeIssue10	*If* AV does not maintain a Safe distance to the rear vehicle below the minimum, *then* [74] the distance to the rear vehicle is inadequate
SafeIssue11	*If* an AV does not kept a Safe distance to the side vehicle below the minimum value, *then* the distance to the side vehicle is inadequate [74]
SafeIssue12	*If* a malware attack manipulates the radio of the vehicle using a Bluetooth stack weakness and inserted the malware codes by syncing their mobile phones with the radio, *then* the inserted code could send messages to the ECU of the vehicle that could lock the brakes [72]
SafeIssue13	*If* an attacker sends non-encrypted messages to AV and fake nodes are used to send misleading location and traffic condition information to the AV; *then* the AV suffers a sybil attack *and then*, the AV stops in the middle of the road [72]
SafeIssue14	*If* an attacker manipulates the communication messages between two entities (while both entities believe that they are in direct communication with each other), by controlling OBUs or RSUs and eavesdropping, replaying, and modifying their CAN messages (that regulate the steering, brakes, and vehicle acceleration), *then* the vehicle suffers a man-in-the-middle attack [72], *and then* the vehicle loses the steering control
SafeIssue15 Use case n.1	*If* an attacker generates a fake GPS signal and transmits it to the target vehicle, *then* the vehicle suffers a GPS spoofing attack, *and then* the AV suffers a rear-end collision with the vehicle behind it [21]
SafeIssue16 Use case n.2	*If* an attacker records the BSM (transmitted by the lead vehicle to the first member vehicle of a platoon, at time X), and then the attacker replies the old BSM to the first member vehicle at time Y, *then* the first member vehicle suffers a reply, *and then* this event causes the collision between the first member vehicle and the lead vehicle provoking the platoon dissolution [67]

Figure 7.5 shows Safety Issues Rules in SWRLTab.



	Name	Rule		Comment
☑	SafeConseq7	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ea) ^ ssiov:EavesdroppingAttack(?ea) ^ ssiov:FalseMessage(?fm) -> ssiov:receive(?v, ?fm)		The safety consequence of an Eavesdrop...
☑	SafeConseq8	ssiov:Vehicle(?v) ^ ss	OrEqual(?sdsv, ...	The safety consequence of n inadequate ...
☑	SafeConseq9	ssiov:Attacker(?a) ^ s		The safety consequence because of recei...
☑	SafeIssue (use case n.1)	ssiov:VehicleID123(?	dCollisionWith(?...	The security-safety issue related to a Spoo...
☑	SafeIssue1	ssiov:Attacker(?a) ^ s	age(?a, ?v2) -> s...	The security safety issue related to Injectin...
☑	SafeIssue10	ssiov:Vehicle(?v) ^ ss	rEqual(?sdrv, 30...	The safety issue related to Inadequate Saf...
☑	SafeIssue11	ssiov:Vehicle(?v) ^ ss	ceFromSideVehi...	The safety issue related to a Sybil Attack
☑	SafeIssue12	ssiov:Vehicle(?v) ^ ss	(?v, ?o)	The safety issue related to False Messages
☑	SafeIssue13	ssiov:Vehicle(?v1) ^ s	collideWith(?v, ?vl)	The safety issue related to Sybil Attack
☑	SafeIssue14	ssiov:Vehicle(?v1) ^ s	v2, ?sc) ^ ssiov:...	The safety issue related to a Man-in -the-...
☑	SafeIssue2	ssiov:Vehicle(?v1) ^ s	ov:collideWith(?v...	The safety issue related to a Sybil Attack
☑	SafeIssue3	ssiov:Vehicle(?v) ^ ss	siov:collideWith(...	The safety issue related to a Jamming Atta...
☑	SafeIssue4	ssiov:Vehicle(?v) ^ ss	ront(?vf) -> ssiov...	The safet issue related to a Remote Acces...

Name

SafeIssue1

Comment

The security safety issue related to Injecting False Messages

Status

Ok

```
ssiov:Attacker(?a) ^ ssiov:Vehicle(?v1) ^ ssiov:Vehicle(?v2) ^ ssiov:SideVehicle(?v3) ^ ssiov:injectFalseMessage(?a, ?v1)
^ ssiov:injectFalseMessage(?a, ?v2) -> ssiov:causeCollisionOf(?a, ?v1) ^ ssiov:causeCollisionOf(?a, ?v3)
```

Using the Drools rule engine.

Press the 'OWL+SWRL->Drools' button to transfer SWRL ru
Press the 'Run Drools' button to run the rule engine.
Press the 'Drools->OWL' button to transfer the inferred rule

The SWRLAPI supports an OWL profile called OWL 2 RL an
See the 'OWL 2 RL' sub-tab for more information on this rea

Figure 7.5: Safety Issue Rules in SWRLTab

Chapter 8

III. Validation of SSIoV ontology

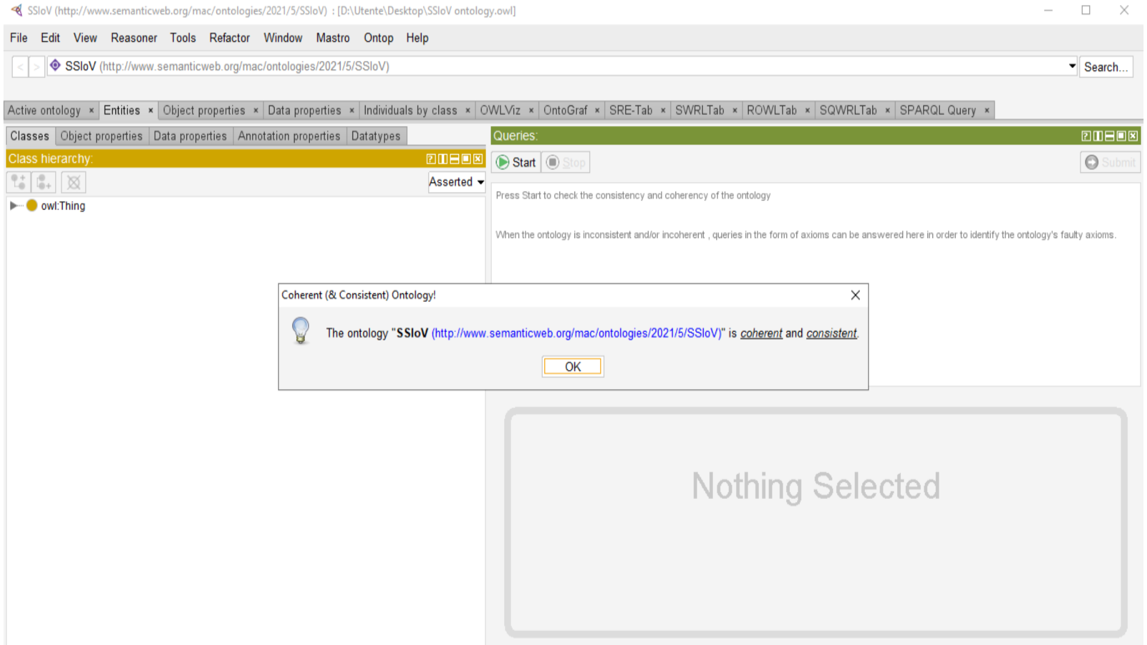
The third objective of our methodology is to verify the consistency of the ontology through the inference engine. We perform analytical reasoning to verify the consistency of SSIoV ontology, safety and security axioms, and rules.

8.1 Checking the Consistency of SSIoV ontology

First, we intend to do a syntactical validation through OntoDebug (that is a free and open-source interactive ontology debugger plugin for Protégé) to resolve and repair inconsistent and incoherent ontologies. OntoDebug *"supports in the discovery and identification of axioms that are responsible for the inconsistency or in-coherency in faulty ontologies by applying interactive ontology debugging. Interactive ontology debugging is implemented by interactively stating queries in the form of axioms the ontology engineer has to answer. This iterative process narrows down the set of possible faulty axioms until the final set of faulty axioms is identified"*[1].

We run the OntoDebug plug-in to verify the consistency of the SSIoV ontology, and the results are as follows in Figure 8.1.

[1] See https://protegewiki.stanford.edu/wiki/OntoDebug


SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × OntoGraf × SRE-Tab × SWRLTab × ROWLTab × SQWRLTab × SPARQL Query ×
Classes Object properties Data properties Annotation properties Datatypes
Class hierarchy:
Asserted
owl:Thing
Queries:
Start
Stop
Submit
OK
Nothing Selected

Press Start to check the consistency and coherency of the ontology

When the ontology is inconsistent and/or incoherent , queries in the form of axioms can be answered here in order to identify the ontology's faulty axioms.

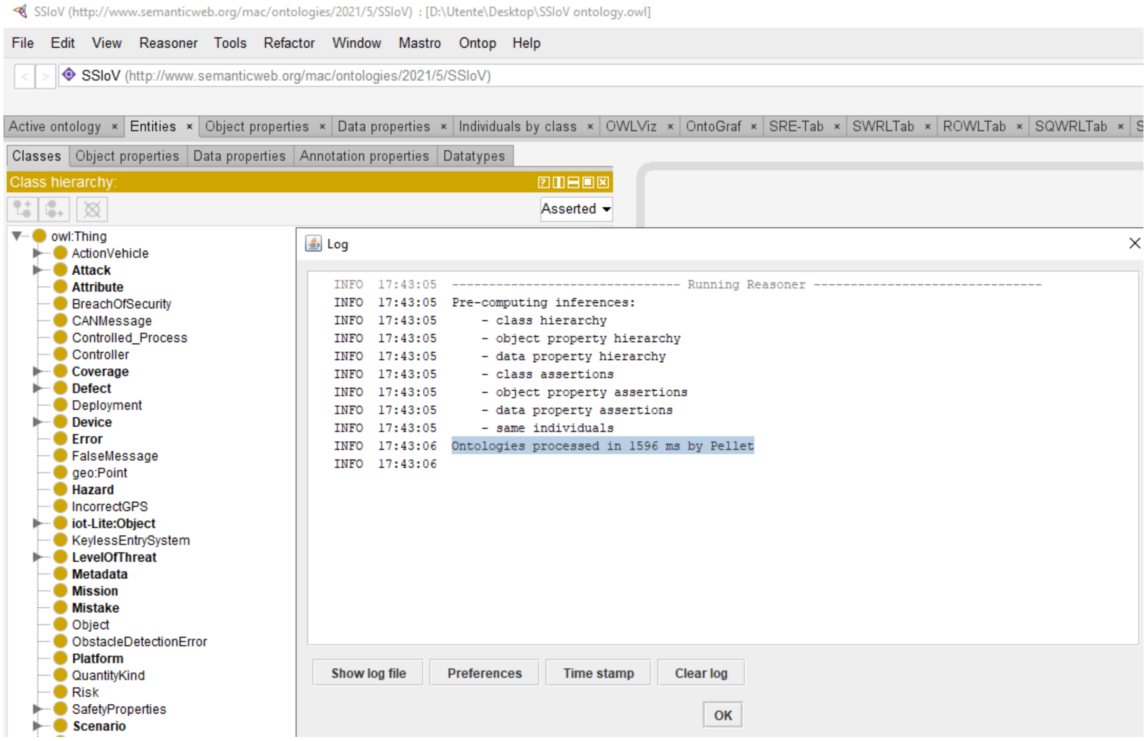
Coherent (& Consistent) Ontology!

The ontology "**SSIoV** (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)" is coherent and consistent.

Figure 8.1: The outcome of running OntoDebug plug-in for consistency checking of SSIoV ontology

We use a reasoning engine to validate the SSIoV as regards to the syntactical evaluation. SSIoV is developed using Protégé-OWL-5.5.0-beta-9. The Pellet reasoner embedded in Protégé-OWL can be used to detect syntactical errors of SSIoV. Pellet is an open-source Java based OWL DL reasoner, and it can be used with Jena and OWL API libraries. Pellet ensures to check the ontology consistency by ensuring that ontology does not contain any contradictory facts. Then, Pellet checks the conceptual satisfiability thereby verifying whether a class is likely to have any instances. Pellet also allows the classification between every class to create the complete class hierarchy [75].

Figure 8.2 shows the log of running that the Pellet plug-in uses to complete consistency check of the SSIoV ontology.


SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × OntoGraf × SRE-Tab × SWRLTab × ROWLTab × SQWRLTab ×
Classes Object properties Data properties Annotation properties Datatypes
Class hierarchy:
Asserted
owl:Thing
ActionVehicle
Attack
Attribute
BreachOfSecurity
CANMessage
Controlled_Process
Controller
Coverage
Defect
Deployment
Device
Error
FalseMessage
geo:Point
Hazard
IncorrectGPS
iot-Lite:Object
KeylessEntrySystem
LevelOfThreat
Metadata
Mission
Mistake
Object
ObstacleDetectionError
Platform
QuantityKind
Risk
SafetyProperties
Scenario
Log
Show log file
Preferences
Time stamp
Clear log
OK

```
INFO  17:43:05  ------------------------------ Running Reasoner ------------------------------
INFO  17:43:05  Pre-computing inferences:
INFO  17:43:05      - class hierarchy
INFO  17:43:05      - object property hierarchy
INFO  17:43:05      - data property hierarchy
INFO  17:43:05      - class assertions
INFO  17:43:05      - object property assertions
INFO  17:43:05      - data property assertions
INFO  17:43:05      - same individuals
INFO  17:43:06  Ontologies processed in 1596 ms by Pellet
INFO  17:43:06
```

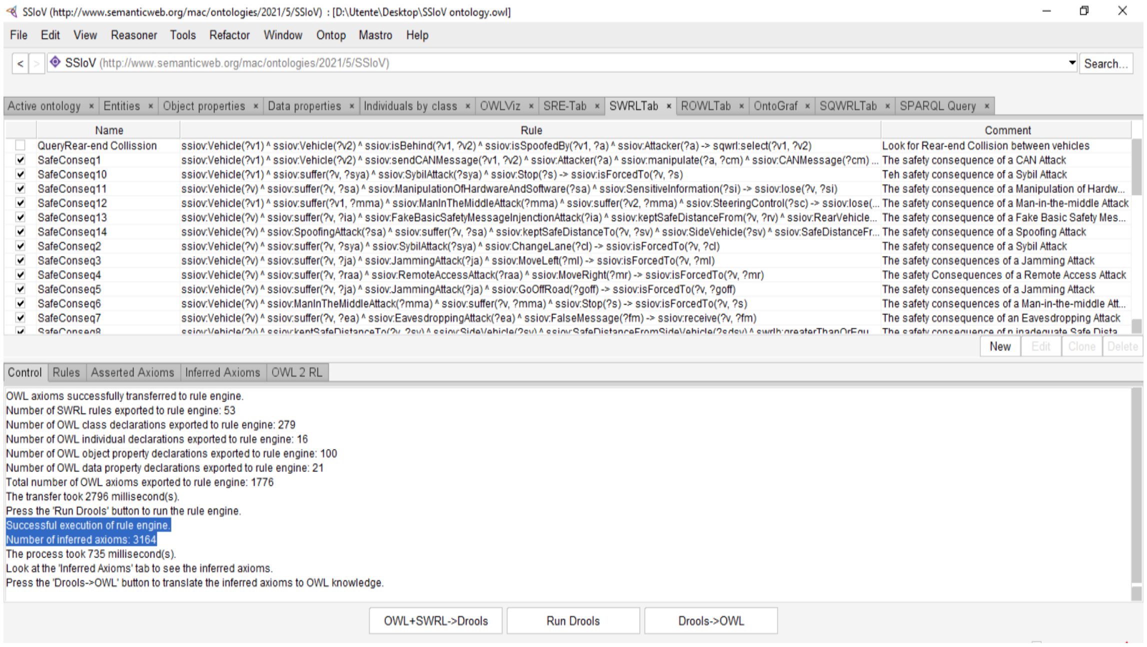
Figure 8.2: The log of running Pellet plug-in for consistency checking of SSIoV ontology

The Pellet reasoner completed the consistency check in 1596 ms.

Therefore, the SSIoV ontology is coherent and consistent as shown by the results of the two plug-ins in Protégé.

8.2 Validating SSIoV ontology's Rules

To validate the SWRL Rules, we use a plug-in called SWRLTab. It ensures a preliminary validation of rules by showing a log of the running SWRLTab plug-in as in Figure 8.3.

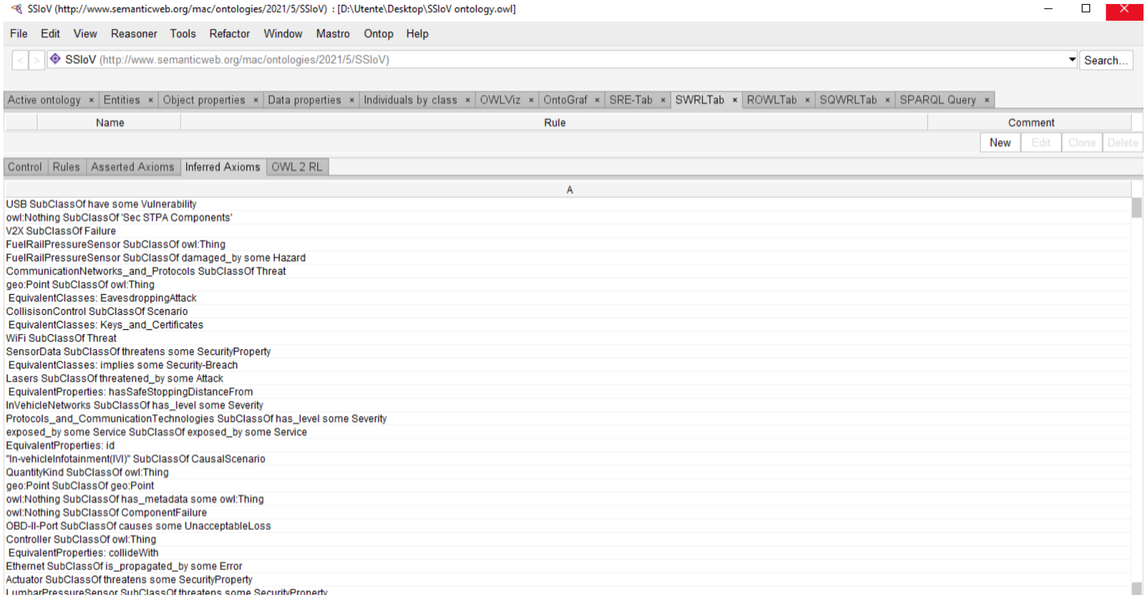


	Name	Rule	Comment
□	QueryRear-end Collission	ssiov:Vehicle(?v1) ^ ssiov:Vehicle(?v2) ^ ssiov:isBehind(?v1, ?v2) ^ ssiov:isSpoofedBy(?v1, ?a) ^ ssiov:Attacker(?a) -> sqwrl:select(?v1, ?v2)	Look for Rear-end Collision between vehicles
☑	SafeConseq1	ssiov:Vehicle(?v1) ^ ssiov:Vehicle(?v2) ^ ssiov:sendCANMessage(?v1, ?v2) ^ ssiov:Attacker(?a) ^ ssiov:manipulate(?a, ?cm) ^ ssiov:CANMessage(?cm) ...	The safety consequence of a CAN Attack
☑	SafeConseq10	ssiov:Vehicle(?v1) ^ ssiov:suffer(?v, ?sya) ^ ssiov:SybilAttack(?sya) ^ ssiov:Stop(?s) -> ssiov:isForcedTo(?v, ?s)	Teh safety consequence of a Sybil Attack
☑	SafeConseq11	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?sa) ^ ssiov:ManipulationOfHardwareAndSoftware(?sa) ^ ssiov:SensitiveInformation(?si) -> ssiov:lose(?v, ?si)	The safety consequence of a Manipulation of Hardw...
☑	SafeConseq12	ssiov:Vehicle(?v1) ^ ssiov:suffer(?v1, ?mma) ^ ssiov:ManInTheMiddleAttack(?mma) ^ ssiov:suffer(?v2, ?mma) ^ ssiov:SteeringControl(?sc) -> ssiov:lose(...	The safety consequence of a Man-in-the-middle Attack
☑	SafeConseq13	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ia) ^ ssiov:FakeBasicSafetyMessageInjenctionAttack(?ia) ^ ssiov:keptSafeDistanceFrom(?v, ?rv) ^ ssiov:RearVehicle...	The safety consequence of a Fake Basic Safety Mes...
☑	SafeConseq14	ssiov:Vehicle(?v) ^ ssiov:SpoofingAttack(?sa) ^ ssiov:suffer(?v, ?sa) ^ ssiov:keptSafeDistanceTo(?v, ?sv) ^ ssiov:SideVehicle(?sv) ^ ssiov:SafeDistanceFr...	The safety consequence of a Spoofing Attack
☑	SafeConseq2	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?sya) ^ ssiov:SybilAttack(?sya) ^ ssiov:ChangeLane(?cl) -> ssiov:isForcedTo(?v, ?cl)	The safety consequence of a Sybil Attack
☑	SafeConseq3	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ja) ^ ssiov:JammingAttack(?ja) ^ ssiov:MoveLeft(?ml) -> ssiov:isForcedTo(?v, ?ml)	The safety consequences of a Jamming Attack
☑	SafeConseq4	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?raa) ^ ssiov:RemoteAccessAttack(?raa) ^ ssiov:MoveRight(?mr) -> ssiov:isForcedTo(?v, ?mr)	The safety Consequences of a Remote Access Attack
☑	SafeConseq5	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ja) ^ ssiov:JammingAttack(?ja) ^ ssiov:GoOffRoad(?goff) -> ssiov:isForcedTo(?v, ?goff)	The safety consequences of a Jamming Attack
☑	SafeConseq6	ssiov:Vehicle(?v) ^ ssiov:ManInTheMiddleAttack(?mma) ^ ssiov:suffer(?v, ?mma) ^ ssiov:Stop(?s) -> ssiov:isForcedTo(?v, ?s)	The safety consequences of a Man-in-the-middle Att...
☑	SafeConseq7	ssiov:Vehicle(?v) ^ ssiov:suffer(?v, ?ea) ^ ssiov:EavesdroppingAttack(?ea) ^ ssiov:FalseMessage(?fm) -> ssiov:receive(?v, ?fm)	The safety consequence of an Eavesdropping Attack
☑	SafeConseq8	ssiov:Vehicle(?v) ^ ssiov:keptSafeDistanceTo(?v, ?sv) ^ ssiov:SideVehicle(?sv) ^ ssiov:SafeDistanceFromSideVehicle(?sdsv) ^ swrlb:greaterThanOrEqu	The safety consequence of n inadequate Safe Dista

```
OWL axioms successfully transferred to rule engine.
Number of SWRL rules exported to rule engine: 53
Number of OWL class declarations exported to rule engine: 279
Number of OWL individual declarations exported to rule engine: 16
Number of OWL object property declarations exported to rule engine: 100
Number of OWL data property declarations exported to rule engine: 21
Total number of OWL axioms exported to rule engine: 1776
The transfer took 2796 millisecond(s).
Press the 'Run Drools' button to run the rule engine.
Successful execution of rule engine.
Number of inferred axioms: 3164
The process took 735 millisecond(s).
Look at the 'Inferred Axioms' tab to see the inferred axioms.
Press the 'Drools->OWL' button to translate the inferred axioms to OWL knowledge.
```

Figure 8.3: The log of running SWRLTab plug-in for the SWRL rules validation of SSIoV ontology

The number of OWL axioms output by the rule engine is 1776. The 3164 inferred axioms are inferred as shown in Figure 8.4.


SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × OntoGraf × SRE-Tab × SWRLTab × ROWLTab × SQWRLTab × SPARQL Query ×
Name
Rule
Comment
New Edit Clone Delete
Control Rules Asserted Axioms Inferred Axioms OWL 2 RL
A

```
USB SubClassOf have some Vulnerability
owl:Nothing SubClassOf 'Sec STPA Components'
V2X SubClassOf Failure
FuelRailPressureSensor SubClassOf owl:Thing
FuelRailPressureSensor SubClassOf damaged_by some Hazard
CommunicationNetworks_and_Protocols SubClassOf Threat
geo:Point SubClassOf owl:Thing
 EquivalentClasses: EavesdroppingAttack
CollisisonControl SubClassOf Scenario
 EquivalentClasses: Keys_and_Certificates
WiFi SubClassOf Threat
SensorData SubClassOf threatens some SecurityProperty
 EquivalentClasses: implies some Security-Breach
Lasers SubClassOf threatened_by some Attack
 EquivalentProperties: hasSafeStoppingDistanceFrom
InVehicleNetworks SubClassOf has_level some Severity
Protocols_and_CommunicationTechnologies SubClassOf has_level some Severity
exposed_by some Service SubClassOf exposed_by some Service
EquivalentProperties: id
"In-vehicleInfotainment(IVI)" SubClassOf CausalScenario
QuantityKind SubClassOf owl:Thing
geo:Point SubClassOf geo:Point
owl:Nothing SubClassOf has_metadata some owl:Thing
owl:Nothing SubClassOf ComponentFailure
OBD-II-Port SubClassOf causes some UnacceptableLoss
Controller SubClassOf owl:Thing
 EquivalentProperties: collideWith
Ethernet SubClassOf is_propagated_by some Error
Actuator SubClassOf threatens some SecurityProperty
LumbarPressureSensor SubClassOf threatens some SecurityProperty
```

Figure 8.4: Inferred axioms of running SWRLTab plug-in for the SWRL rules validation of SSIoV ontology

Chapter 9

IV. Instantiating Dataset into SSIoV ontology

9.1 Instances for SSIoV ontology

Our idea is to use the VeReMi dataset to evaluate the SSIoVontology's rules.

VeReMi is a *"labelled simulated dataset providing a wide range of traffic behaviour and attacker implementations. The simulations were performed in LuST scenario, which aims to provide comprehensive scenarios for evaluation in VEINS simulator"* [62] The VeReMi's structure is the following:

- 225 individual simulations;
- 5 different attackers;
- 3 different attacker densities: *"The low density corresponding to a run starting at 3:00 and it has 35 to 39 vehicles, the medium density a run at 5:00 and it has between 97 and 108 vehicles, and the high density (7:00) has between 491 and 519 vehicles"* [62];
- Each receiver generates a reception log file;
- There are **message logs** for every vehicle in the simulation. Each message log contains both GPS data about the local vehicle and BSM messages received from other vehicles though DSRC. The message log include speed, claimed transmission time, reception time, position for each receiver;
- The **ground truth file** that specifies the attacker's behaviour;

- **type** of periodic messages from a GPS module in the vehicle. The attackers are the constant attacker, the constant offset attacker, the random attacker, the random offset attacker, and the eventual stop attacker [62].

However, VeReMi's dataset does not have data corresponding to our ontology. It only processes security-related data. Contrary, we need data about safety and security. Moreover, VeReMi can be used to create some instances to evaluate our ontology. For example, each vehicle has its ID, speed, acceleration, position, etc. Therefore, we re-use some of the data (that are in VeReMi) on these vehicle-related features.

Figure 9.1 shows some individuals defined in the SSIoV ontology. Here, we have: `ssiov:vehicle1`; `ssiov:vehicle2`; `ssiov:vehicle3`, and so on, with their features (`ssiov:speed`; `ssiov:position`; and so on) and their actions (`ssiov:manipulate`, and so on). These individuals are defined as considering the subsequent inclusion of SWRL rules into the SSIoV ontology.

Individual	Class	Data property	Object Property
Attacker1	Attacker		injectFalseMessage sendFalseMessage manipulate
Attacker2	Attacker		manipulateCANMessage
Attacker3	Attacker		encryptPersonalMediaRepository A59
Attacker4	Attacker		transmitFalsifiedGPSTo falsifyGPS of
Vehicle1	Vehicle	ID 145 safeFollowing Distance 10 m speed -0,075827101 position 267.24703617 acceleration 0,2383660089	
Vehicle2	Vehicle	ID 416 safeFollowingDistance 35 m speed 4.2383660089 position 118.23836600 acceleration 0,138366880089	
Vehicle3	Vehicle	ID 417 safeFollowingDistance 40 m speed 3.117997008591 position 103.117997008591 acceleration 01.117997008591	suffer a sybil attack
Vehicleid123	TargetVehicle	backSafetyDistance safeFollowingDistance 15 m	
Vehicleid122	RearVehicle		

Table 9.1: Example of some Individuals for SSIoV ontology in Protégé

9.2 Evaluating the Effectiveness of SSIoV ontology's Rules

The further step is to verify the effectiveness and feasibility of the SSIoV ontology's rules. Our ontology can generate new facts which are the consequences of some action of an attacker or AVs. To verify it, we create some instances. The forward figures 9.1, 9.2, 9.3, 9.4, 9.5, 9.6, 9.7, 9.8 show the outcomes that we have obtained after performing the reasoning with the reasoner Pellet for four SSIoV ontology's rules.

9.2.1 Evaluating the SecBreach1 Rule

SecBreach1: ssiov:Attacker (?a) ∧ ssiov:Vehicle (?v) ∧
ssiov:manipulate (?a, ?v) ∧ ssiov:injectFalseMessage (?a, ?v) ∧
ssiov:Vehicle (?v3) → ssiov:doMalwareAttackAgainst (?a, ?v)

Based on this rule, when an attacker manipulates and injects fake messages to the vehicle, then the attacker performs a malware attack against the vehicle. Therefore, if we launch the engine reasoner, it should understand that the Attacker1 (= Instance) that manipulates and injects false message to the Vehicle1 (= Instance), does a malware attack against the Vehicle1. The outcome of the engine reasoning is highlighted in pink in Figure 9.1.

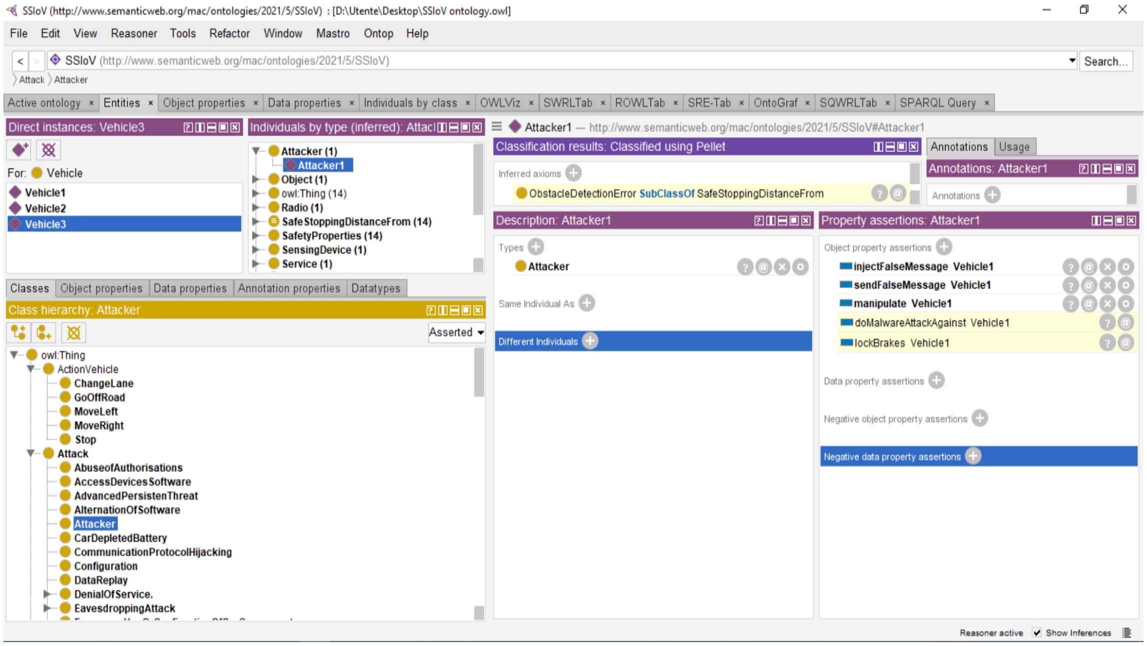
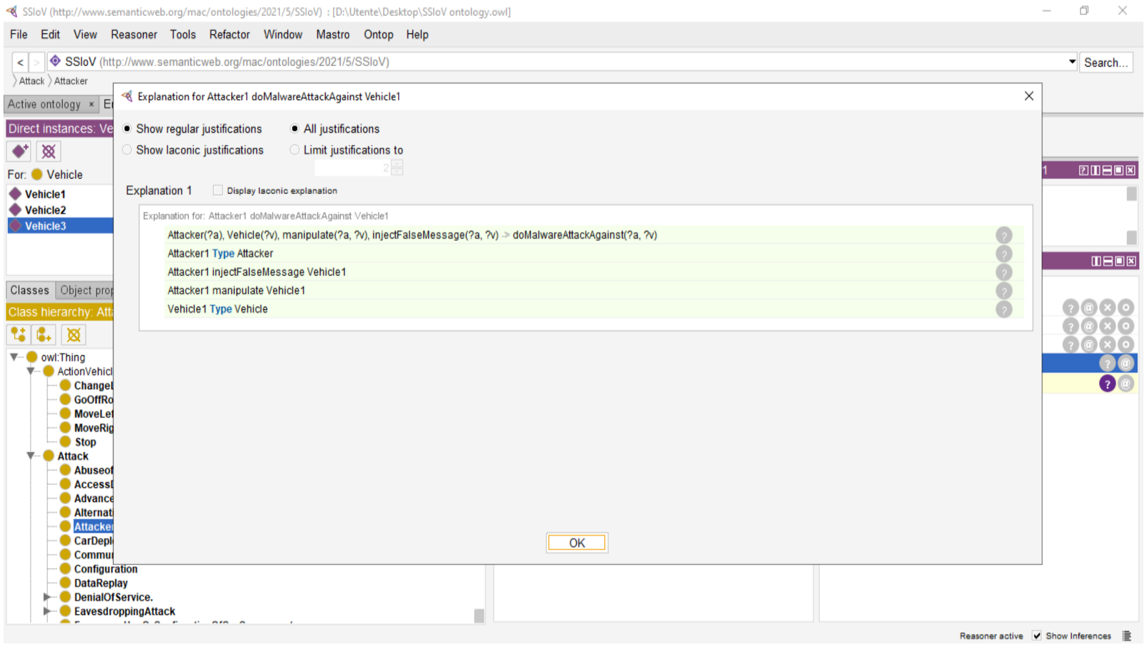

SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
Attack 〉 Attacker
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × SWRLTab × ROWLTab × SRE-Tab × OntoGraf × SQWRLTab × SPARQL Query ×
Direct instances: Vehicle3
For: Vehicle
Vehicle1
Vehicle2
Vehicle3
Individuals by type (inferred): Attacl
Attacker (1)
Attacker1
Object (1)
owl:Thing (14)
Radio (1)
SafeStoppingDistanceFrom (14)
SafetyProperties (14)
SensingDevice (1)
Service (1)
Classes Object properties Data properties Annotation properties Datatypes
Class hierarchy: Attacker
Asserted
owl:Thing
ActionVehicle
ChangeLane
GoOffRoad
MoveLeft
MoveRight
Stop
Attack
AbuseofAuthorisations
AccessDevicesSoftware
AdvancedPersistenThreat
AlternationOfSoftware
Attacker
CarDepletedBattery
CommunicationProtocolHijacking
Configuration
DataReplay
DenialOfService.
EavesdroppingAttack
Attacker1 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#Attacker1
Classification results: Classified using Pellet
Inferred axioms
ObstacleDetectionError SubClassOf SafeStoppingDistanceFrom
Annotations Usage
Annotations: Attacker1
Annotations
Description: Attacker1
Types
Attacker
Same Individual As
Different Individuals
Property assertions: Attacker1
Object property assertions
injectFalseMessage Vehicle1
sendFalseMessage Vehicle1
manipulate Vehicle1
doMalwareAttackAgainst Vehicle1
lockBrakes Vehicle1
Data property assertions
Negative object property assertions
Negative data property assertions
Reasoner active
Show Inferences

Figure 9.1: Inferred axioms of running Pellet plug-in for SSIoV ontology

Figure 9.2 shows the explanation of the outcome. We can note that the reasoner uses the **SecBreach1** Rule to infer the result.



Explanation for Attacker1 doMalwareAttackAgainst Vehicle1

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: Attacker1 doMalwareAttackAgainst Vehicle1

Attacker(?a), Vehicle(?v), manipulate(?a, ?v), injectFalseMessage(?a, ?v) -> doMalwareAttackAgainst(?a, ?v)

Attacker1 Type Attacker

Attacker1 injectFalseMessage Vehicle1

Attacker1 manipulate Vehicle1

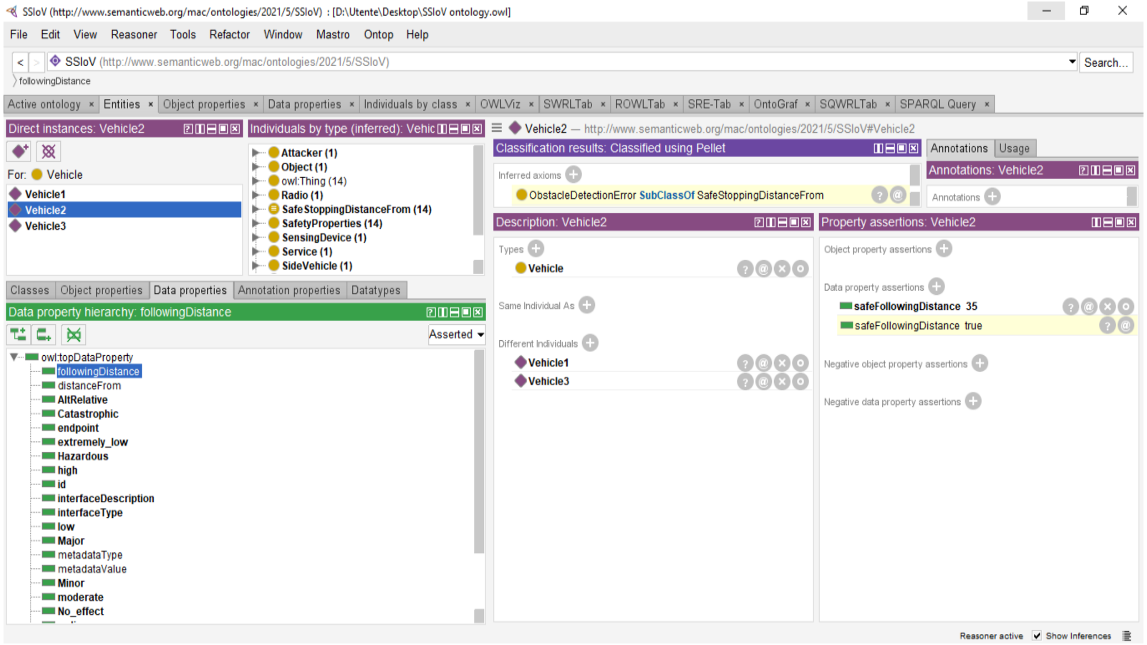
Vehicle1 Type Vehicle

Figure 9.2: Inferred axioms explanation of running Pellet plug-in for SSIoV

9.2.2 Evaluating the SafeRule1

SafeRule1: `ssiov:Vehicle (?v) ∧ ssiov:safeFollowingDistance (?v, ?sfd) ∧ swrlb:greaterThanOrEqual (?sfd, 30) → ssiov:safeFollowingDistance (?v, true)`

According to this rule, if the distance between the vehicle and the forward vehicle is greater than or equal to 30 m, then the distance between two vehicles is safe. Therefore, if we launch the engine reasoner, it should understand that the Vehicle2 (= Instance), which has a distance greater than or equal to 30 m, has a safe distance with the forward vehicle. The results of the engine reasoning are highlighted in pink in Figure 9.3 .



SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
followingDistance
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × SWRLTab × ROWLTab × SRE-Tab × OntoGraf × SQWRLTab × SPARQL Query ×
Direct instances: Vehicle2
Individuals by type (inferred): Vehic
Classes Object properties Data properties Annotation properties Datatypes
Data property hierarchy: followingDistance
Asserted
Classification results: Classified using Pellet
Annotations Usage
Annotations: Vehicle2
Annotations
Description: Vehicle2
Property assertions: Vehicle2
Reasoner active
☑ Show Inferences

For: Vehicle

- Vehicle1
- Vehicle2
- Vehicle3

- Attacker (1)
- Object (1)
- owl:Thing (14)
- Radio (1)
- SafeStoppingDistanceFrom (14)
- SafetyProperties (14)
- SensingDevice (1)
- Service (1)
- SideVehicle (1)

owl:topDataProperty
- followingDistance
- distanceFrom
- AltRelative
- Catastrophic
- endpoint
- extremely_low
- Hazardous
- high
- id
- interfaceDescription
- interfaceType
- low
- Major
- metadataType
- metadataValue
- Minor
- moderate
- No_effect

Vehicle2 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#Vehicle2

Inferred axioms

ObstacleDetectionError SubClassOf SafeStoppingDistanceFrom

Types

Vehicle

Same Individual As

Different Individuals

Vehicle1

Vehicle3

Object property assertions

Data property assertions

safeFollowingDistance 35

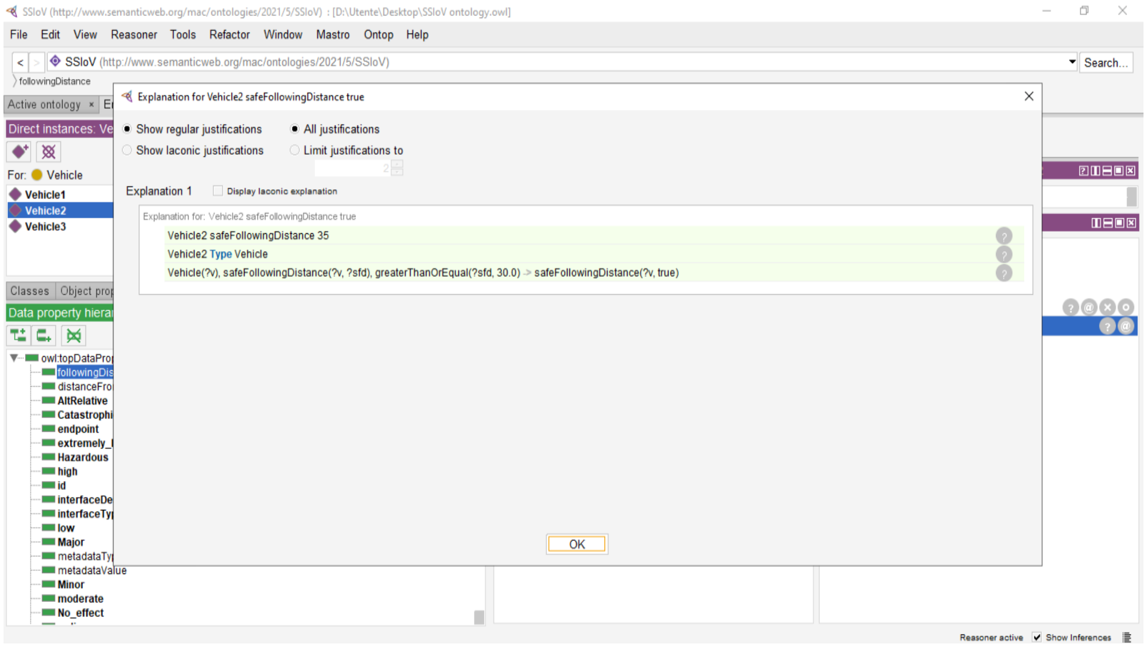
safeFollowingDistance true

Negative object property assertions

Negative data property assertions

Figure 9.3: Inferred axioms of running Pellet plug-in for SSIoV ontology

Figure 9.4 shows the explanation of the outcome. The reasoner uses the **SafeRule1** to infer the outcome.



Explanation for Vehicle2 safeFollowingDistance true

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: Vehicle2 safeFollowingDistance true

Vehicle2 safeFollowingDistance 35

Vehicle2 Type Vehicle

Vehicle(?v), safeFollowingDistance(?v, ?sfd), greaterThanOrEqual(?sfd, 30.0) -> safeFollowingDistance(?v, true)

OK

Figure 9.4: Inferred axioms explanation of running Pellet plug-in for SSIoV

9.2.3 Evaluating the SafeConseq10 Rule

SafeConseq10: `ssiov:Vehicle (?v)` ∧ `ssiov:SybilAttack (?sya)` ∧ `ssiov:suffer (?v, ?sya)` → `ssiov:isForcedTo (?v, ssiov:stop)`

According to this rule, if the vehicle suffers a sybil attack, the vehicle will be forced to stop. Therefore, if we launch the engine reasoner, it should understand that if the Vehicle 3 (= Instance) suffers a Sybil Attack, then the Vehicle 3 is forced to stop. The outcome of the engine reasoning is highlighted in pink in Figure 9.5.

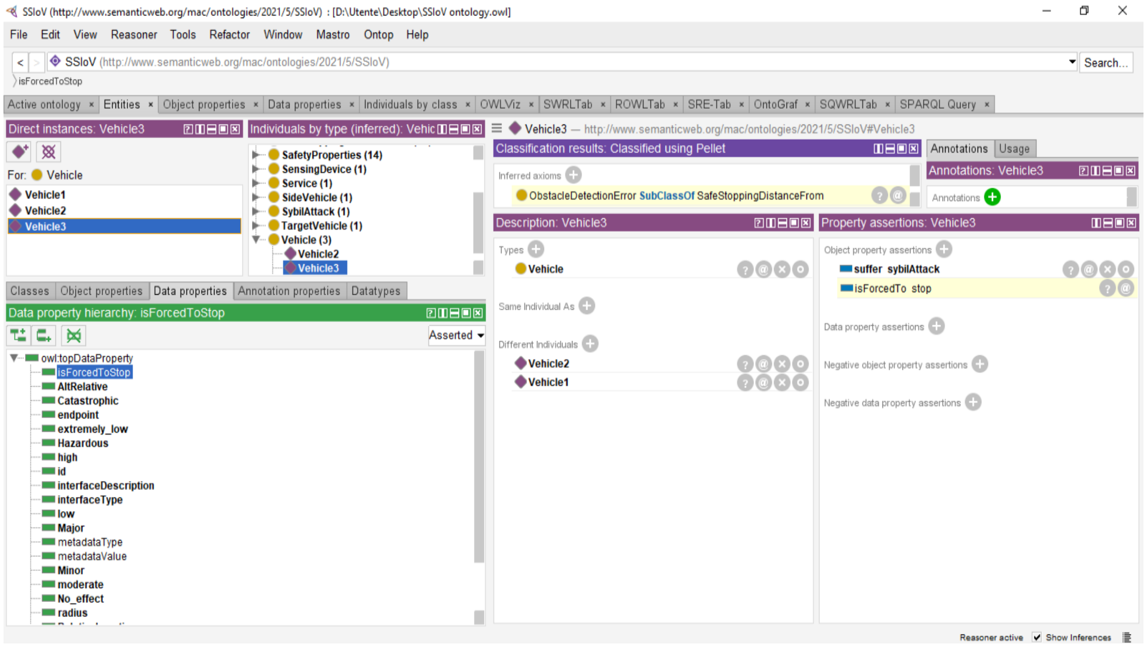
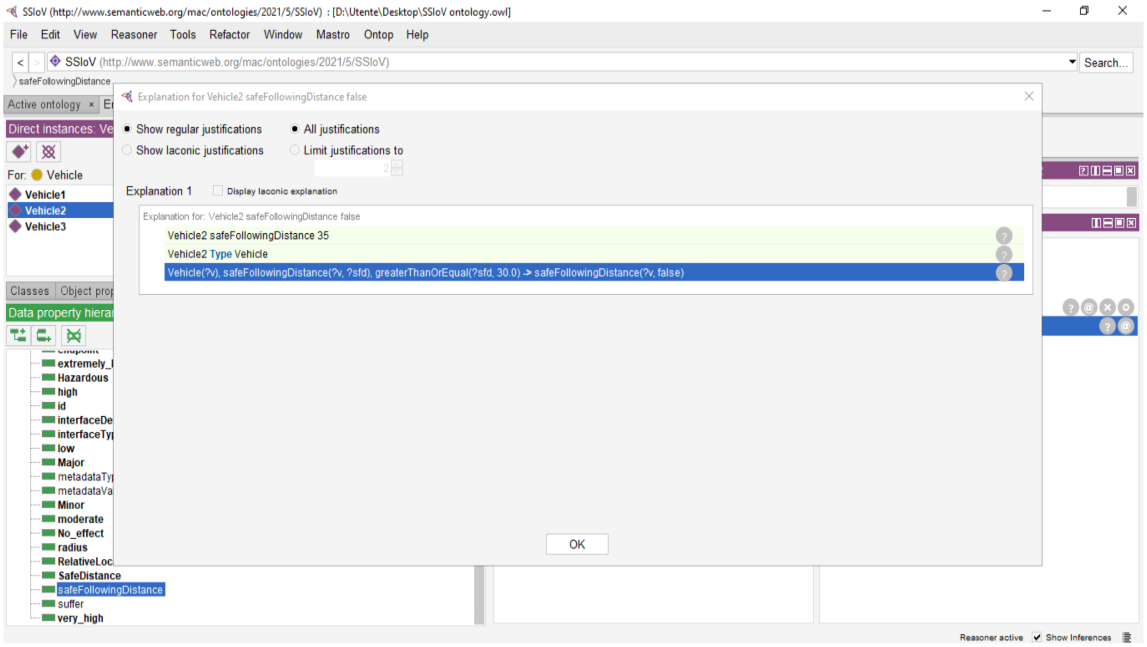

SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
isForcedToStop
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × SWRLTab × ROWLTab × SRE-Tab × OntoGraf × SQWRLTab × SPARQL Query ×
Direct instances: Vehicle3
For: Vehicle
Vehicle1
Vehicle2
Vehicle3
Individuals by type (inferred): Vehic
SafetyProperties (14)
SensingDevice (1)
Service (1)
SideVehicle (1)
SybilAttack (1)
TargetVehicle (1)
Vehicle (3)
Vehicle2
Vehicle3
Vehicle3 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#Vehicle3
Classification results: Classified using Pellet
Inferred axioms
ObstacleDetectionError SubClassOf SafeStoppingDistanceFrom
Annotations Usage
Annotations: Vehicle3
Annotations
Classes Object properties Data properties Annotation properties Datatypes
Data property hierarchy: isForcedToStop
Asserted
owl:topDataProperty
isForcedToStop
AltRelative
Catastrophic
endpoint
extremely_low
Hazardous
high
id
interfaceDescription
interfaceType
low
Major
metadataType
metadataValue
Minor
moderate
No_effect
radius
Description: Vehicle3
Types
Vehicle
Same Individual As
Different Individuals
Vehicle2
Vehicle1
Property assertions: Vehicle3
Object property assertions
suffer sybilAttack
isForcedTo stop
Data property assertions
Negative object property assertions
Negative data property assertions
Reasoner active
Show Inferences

Figure 9.5: Inferred axioms of running Pellet plug-in for SSIoV ontology

The following Figure 9.6 shows the explanation of the outcome. We can note that the reasoner uses the **SafeConseq10** to infer this result.



Explanation for Vehicle2 safeFollowingDistance false

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: Vehicle2 safeFollowingDistance false

Vehicle2 safeFollowingDistance 35

Vehicle2 Type Vehicle

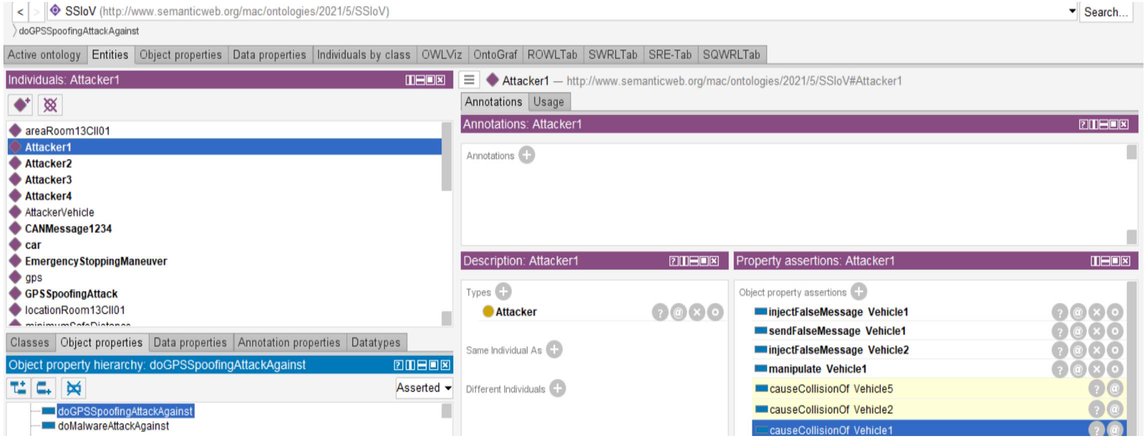
Vehicle(?v), safeFollowingDistance(?v, ?sfd), greaterThanOrEqual(?sfd, 30.0) -> safeFollowingDistance(?v, false)

Figure 9.6: Inferred axioms explanation of running Pellet plug-in for SSIoV ontology

9.2.4 Evaluating the SafeIssue1 Rule

SafeIssue1: ssiov:Vehicle (?v1) ∧ ssiov:Vehicle (?v2) ∧ ssiov:injectFalseMessage (?a, ?v1) ∧ ssiov:injectFalseMessage (?a, ?v2) ∧ ssiov:Vehicle (?v3) → ssiov:collideWith (?v1, ?v3)

According to this rule, if an attacker injects false messages into two vehicles, then one of the two vehicles collides with the other. Hence, if we launch the engine reasoner, it should understand that if the Vehicle 1 (= Instance) and Vehicle 2 (= Instances) receive false messages, the Vehicle 1 (= Instance) collides with another Vehicle 3 (= Instance). The outcome of the engine reasoning is highlighted in pink in Figure 9.7.



areaRoom13CII01
Attacker1
Attacker2
Attacker3
Attacker4
AttackerVehicle
CANMessage1234
car
EmergencyStoppingManeuver
gps
GPSSpoofingAttack
locationRoom13CII01

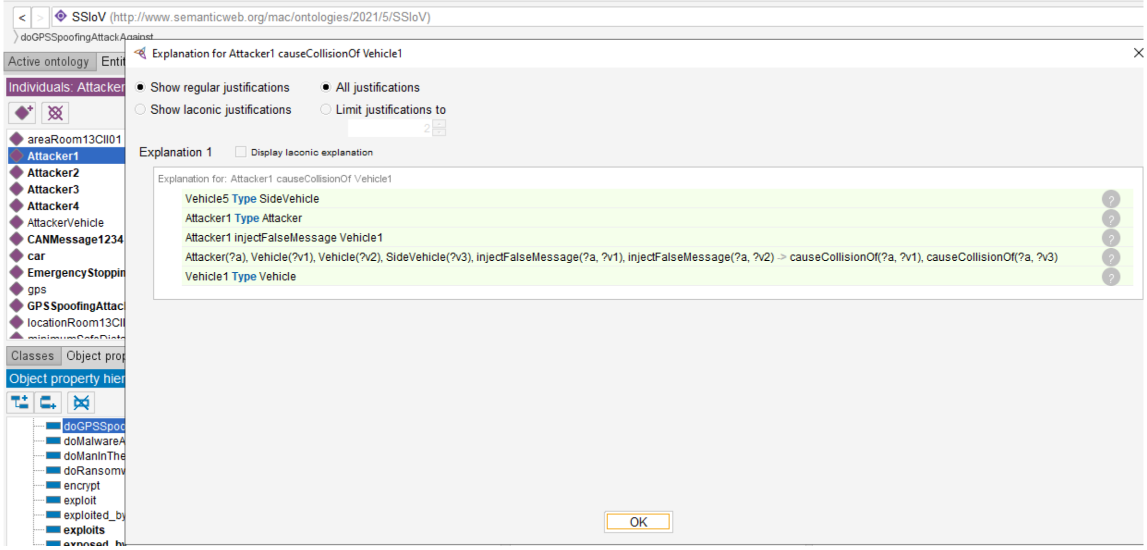
doGPSSpoofingAttackAgainst
doMalwareAttackAgainst

Types: Attacker

Object property assertions:
injectFalseMessage Vehicle1
sendFalseMessage Vehicle1
injectFalseMessage Vehicle2
manipulate Vehicle1
causeCollisionOf Vehicle5
causeCollisionOf Vehicle2
causeCollisionOf Vehicle1

Figure 9.7: Inferred axioms of running Pellet plug-in for SSIoV ontology

Figure 9.8 shows the explanation of the outcome. We can note that the reasoner uses the **SafeIssue1** to infer this result.


SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
doGPSSpoofingAttackAgainst
Active ontology
Entit
Individuals: Attacker
Classes
Object prop
Object property hier
OK

Explanation for Attacker1 causeCollisionOf Vehicle1

● Show regular justifications
○ Show laconic justifications
● All justifications
○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: Attacker1 causeCollisionOf Vehicle1

```
Vehicle5 Type SideVehicle
Attacker1 Type Attacker
Attacker1 injectFalseMessage Vehicle1
Attacker(?a), Vehicle(?v1), Vehicle(?v2), SideVehicle(?v3), injectFalseMessage(?a, ?v1), injectFalseMessage(?a, ?v2) -> causeCollisionOf(?a, ?v1), causeCollisionOf(?a, ?v3)
Vehicle1 Type Vehicle
```

areaRoom13CII01
Attacker1
Attacker2
Attacker3
Attacker4
AttackerVehicle
CANMessage1234
car
EmergencyStoppin
gps
GPSSpoofingAttac
locationRoom13CI

doGPSSpoo
doMalwareA
doManInThe
doRansomw
encrypt
exploit
exploited_by
exploits

Figure 9.8: Inferred axioms explanation of running Pellet plug-in for SSIoV

Chapter 10

Querying SSIoV ontology

We can query the SSIoV ontology by using a plug-in called SQWRLQueryTAB that helps us to inquire about the generated facts.

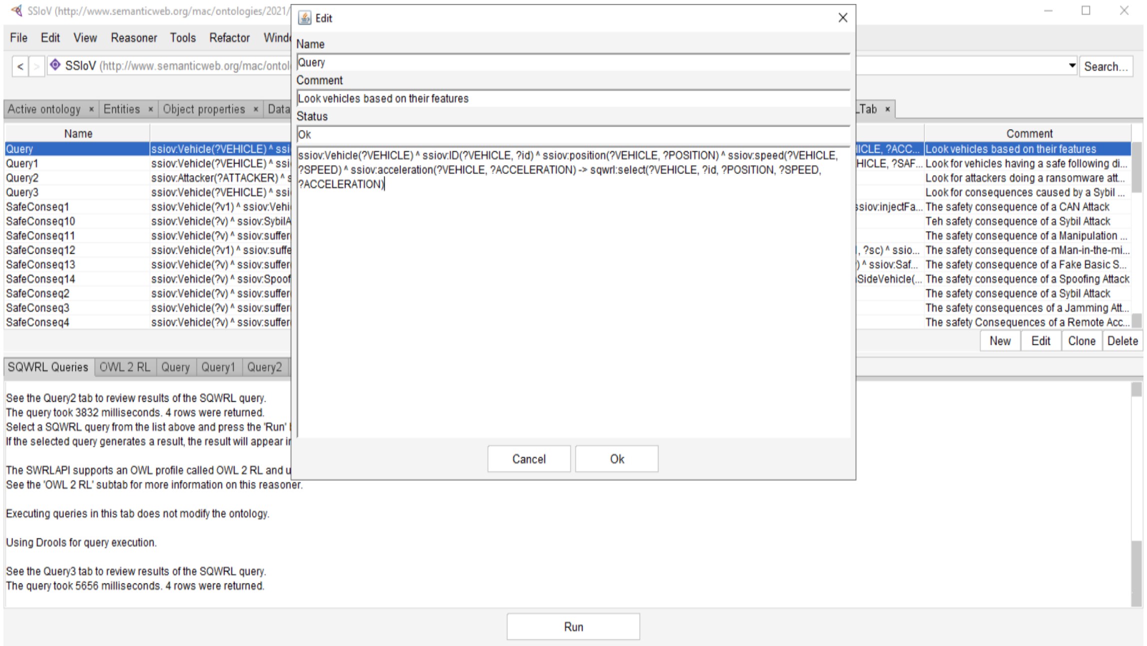
SQWRL is a *"SWRL-based query that will retrieve information from OWL. It is built on the SWRL rule language and assumes the standard SWRL rules antecedent making out a query or pattern specification for retrieving information from OWL and standard SWRL serialisation mechanisms can be used so queries can be stored in OWL ontologies"* [76].

We test four queries through our semantic-based AI reasoning tool as shown in Table 10.1.

Query	Explanation Query	Query Encoded with SQWRL	Outcome Query
Query Vehicles Features	Look for vehicles and their features in terms of ID, position, speed and acceleration	`ssiov:Vehicle(?v) ∧ ssiov:ID (?v, ?id) ssiov:position (?v, ?pos) ∧ ssiov:speed (?v, ?spe) ∧ ssiov:acceleration (?v, ?acc) → sqwrl:select (?v, ?id, ?pos, ?spe, ?acc)`	List vehicles with their corresponding features
Query Safe Following Distance	It queries the ontology to find vehicles with a safe following distance	`ssiov:Vehicle (?v) ssiov:safeFollowing Distance (?v, ?safedistance) → sqwrl:select (?v, ?safedistance)`	List vehicles and their safe distance (greater than 20 m) from the forward vehicle
Query Attack Type	It queries the ontology to look for attacks against vehicles	`ssiov:Attacker (?a) ∧ ssiov:Vehicle (?v) ssiov:do RansomwareAttack (?a, ?v) → sqwrl:select (?a, ?v)`	List the attacks, which an attacker launches against vehicles
Query Safety Consequence	It queries the ontology to look for a vehicle that has safety consequences due to a security attack	`ssiov:Vehicle(?v) ∧ ssiov:Sybil Attack(?sya) ssiov:suffer(?v, ?sya) ∧ ssiov:isForcedTo(?v, ssiov:stop) → sqwrl:select(?v)`	List vehicles that are forced to stop due to a sybil attack

Table 10.1: Four queries to test the semantic-based AI reasoning tool

```
    ssiov:Vehicle(?v) ∧ ssiov:ID (?v, ?id) ssiov:position (?v,
?pos) ∧ ssiov:speed (?v, ?spe) ∧ ssiov:acceleration (?v, ?acc)
           → sqwrl:select (?v, ?id, ?pos, ?spe, ?acc)
```



Name
Query
Comment
Look vehicles based on their features
Status
Ok

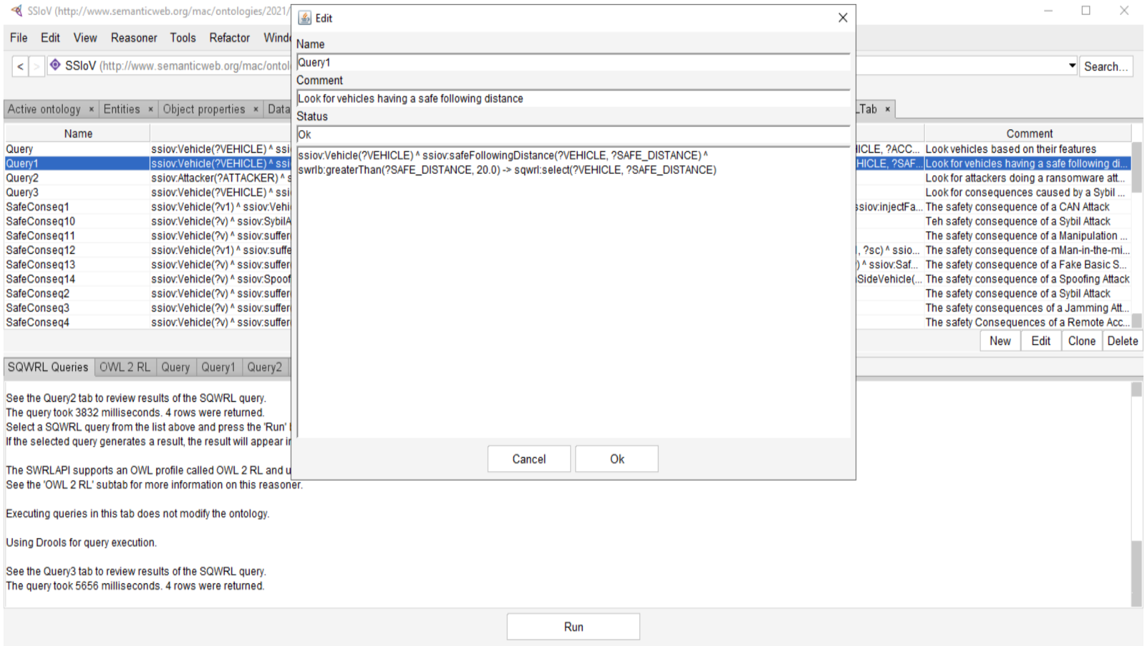
```
ssiov:Vehicle(?VEHICLE) ^ ssiov:ID(?VEHICLE, ?id) ^ ssiov:position(?VEHICLE, ?POSITION) ^ ssiov:speed(?VEHICLE,
?SPEED) ^ ssiov:acceleration(?VEHICLE, ?ACCELERATION) -> sqwrl:select(?VEHICLE, ?id, ?POSITION, ?SPEED,
?ACCELERATION)
```

Name			Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssi	ICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssi	HICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ s		Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssi		Look for consequences caused by a Sybil ...
SafeConseq1	ssiov:Vehicle(?v1) ^ ssiov:Vehi	ssiov:injectFa...	The safety consequence of a CAN Attack
SafeConseq10	ssiov:Vehicle(?v) ^ ssiov:SybilA		Teh safety consequence of a Sybil Attack
SafeConseq11	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Manipulation ...
SafeConseq12	ssiov:Vehicle(?v1) ^ ssiov:suffe	, ?sc) ^ ssio...	The safety consequence of a Man-in-the-mi...
SafeConseq13	ssiov:Vehicle(?v) ^ ssiov:suffer	) ^ ssiov:Saf...	The safety consequence of a Fake Basic S...
SafeConseq14	ssiov:Vehicle(?v) ^ ssiov:Spoof	SideVehicle(...	The safety consequence of a Spoofing Attack
SafeConseq2	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Sybil Attack
SafeConseq3	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequences of a Jamming Att...
SafeConseq4	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety Consequences of a Remote Acc...

```
See the Query2 tab to review results of the SQWRL query.
The query took 3832 milliseconds. 4 rows were returned.
Select a SQWRL query from the list above and press the 'Run' I
If the selected query generates a result, the result will appear ir

The SWRLAPI supports an OWL profile called OWL 2 RL and u
See the 'OWL 2 RL' subtab for more information on this reasoner.

Executing queries in this tab does not modify the ontology.

Using Drools for query execution.

See the Query3 tab to review results of the SQWRL query.
The query took 5656 milliseconds. 4 rows were returned.
```

Figure 10.1: SQWRL rules to query vehicles features (id, position, velocity and acceleration)

The result of this query is a list of vehicles and their features. The query retrieves all vehicles and their features as shown in Figure 10.2.



Name			Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssi	ICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssi	HICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ s		Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssi		Look for consequences caused by a Sybil ...
SafeConseq1	ssiov:Vehicle(?v1) ^ ssiov:Vehi	ssiov:injectFa...	The safety consequence of a CAN Attack
SafeConseq10	ssiov:Vehicle(?v) ^ ssiov:SybilA		Teh safety consequence of a Sybil Attack
SafeConseq11	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Manipulation ...
SafeConseq12	ssiov:Vehicle(?v1) ^ ssiov:suffe	, ?sc) ^ ssio...	The safety consequence of a Man-in-the-mi...
SafeConseq13	ssiov:Vehicle(?v) ^ ssiov:suffer	) ^ ssiov:Saf...	The safety consequence of a Fake Basic S...
SafeConseq14	ssiov:Vehicle(?v) ^ ssiov:Spoof	SideVehicle(...	The safety consequence of a Spoofing Attack
SafeConseq2	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Sybil Attack
SafeConseq3	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequences of a Jamming Att...
SafeConseq4	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety Consequences of a Remote Acc...

Edit

Name

Query1

Comment

Look for vehicles having a safe following distance

Status

Ok

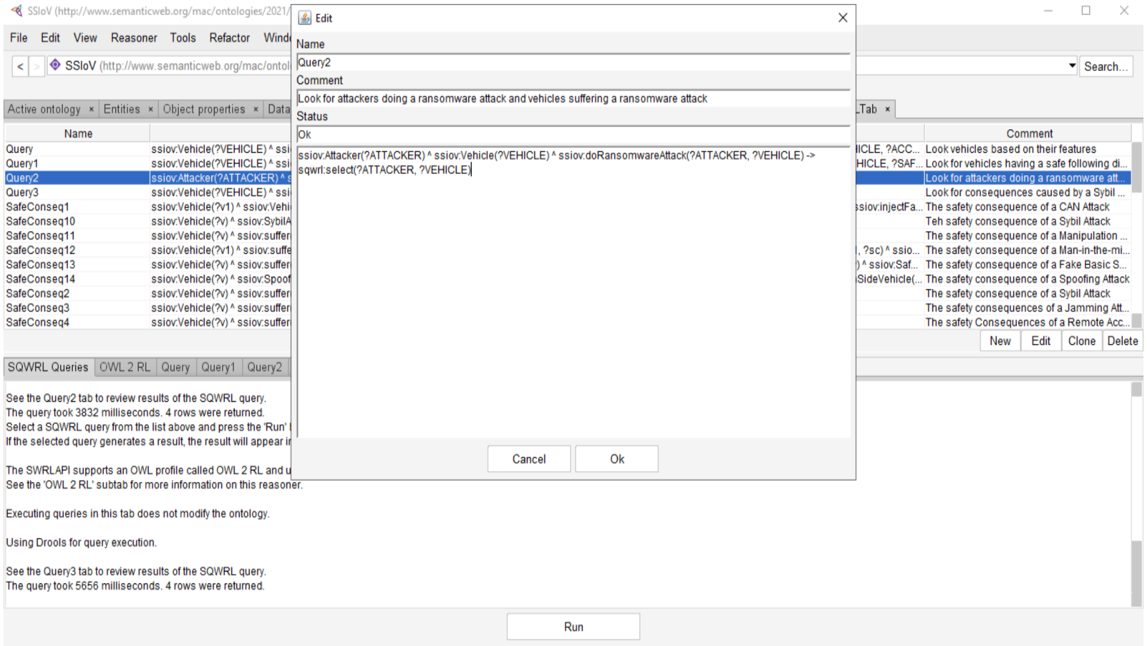
```
ssiov:Vehicle(?VEHICLE) ^ ssiov:safeFollowingDistance(?VEHICLE, ?SAFE_DISTANCE) ^
swrlb:greaterThan(?SAFE_DISTANCE, 20.0) -> sqwrl:select(?VEHICLE, ?SAFE_DISTANCE)
```

Cancel Ok

```
See the Query2 tab to review results of the SQWRL query.
The query took 3832 milliseconds. 4 rows were returned.
Select a SQWRL query from the list above and press the 'Run'
If the selected query generates a result, the result will appear i

The SWRLAPI supports an OWL profile called OWL 2 RL and u
See the 'OWL 2 RL' subtab for more information on this reasoner.

Executing queries in this tab does not modify the ontology.

Using Drools for query execution.

See the Query3 tab to review results of the SQWRL query.
The query took 5656 milliseconds. 4 rows were returned.
```

Figure 10.2: Execution and results of the query in Figure 10.1

```
ssiov:Vehicle (?v)  ssiov:safeFollowingDistance (?v,
 ?safedistance) → sqwrl:select (?v, ?safedistance)
```



Name
Query2
Comment
Look for attackers doing a ransomware attack and vehicles suffering a ransomware attack
Status
Ok

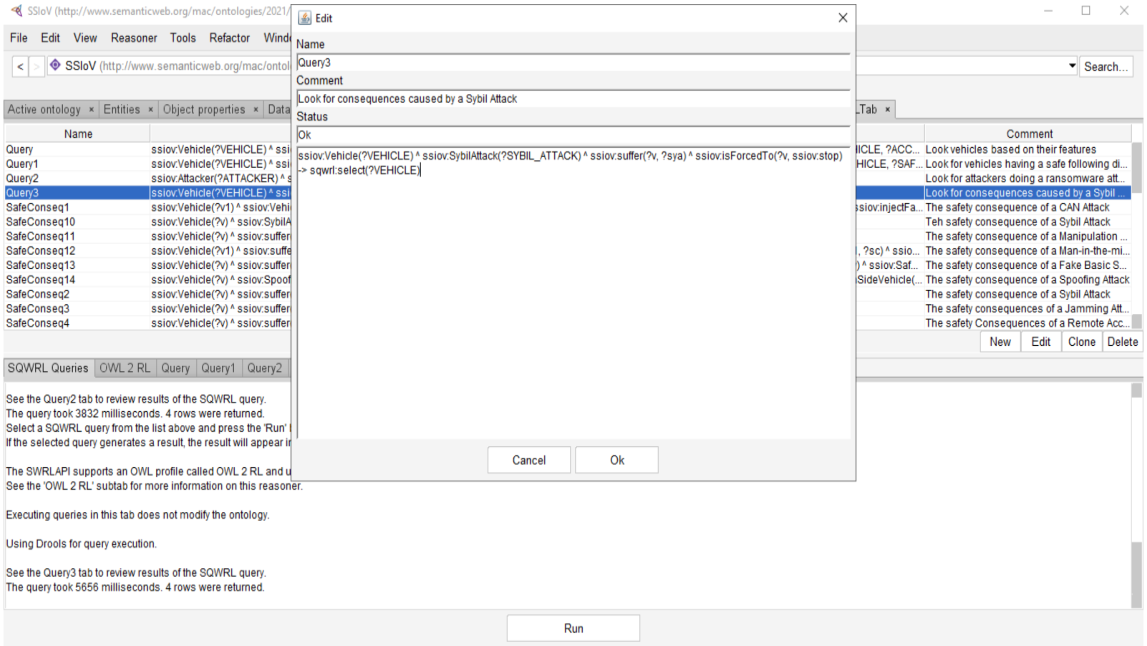
```
ssiov:Attacker(?ATTACKER) ^ ssiov:Vehicle(?VEHICLE) ^ ssiov:doRansomwareAttack(?ATTACKER, ?VEHICLE) ->
sqwrl:select(?ATTACKER, ?VEHICLE)
```

Name			Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssi	ICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssi	HICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ s		Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssi		Look for consequences caused by a Sybil ...
SafeConseq1	ssiov:Vehicle(?v1) ^ ssiov:Vehi	ssiov:injectFa...	The safety consequence of a CAN Attack
SafeConseq10	ssiov:Vehicle(?v) ^ ssiov:SybilA		Teh safety consequence of a Sybil Attack
SafeConseq11	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Manipulation ...
SafeConseq12	ssiov:Vehicle(?v1) ^ ssiov:suffe	, ?sc) ^ ssio...	The safety consequence of a Man-in-the-mi...
SafeConseq13	ssiov:Vehicle(?v) ^ ssiov:suffer	) ^ ssiov:Saf...	The safety consequence of a Fake Basic S...
SafeConseq14	ssiov:Vehicle(?v) ^ ssiov:Spoof	SideVehicle(...	The safety consequence of a Spoofing Attack
SafeConseq2	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Sybil Attack
SafeConseq3	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequences of a Jamming Att...
SafeConseq4	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety Consequences of a Remote Acc...

```
See the Query2 tab to review results of the SQWRL query.
The query took 3832 milliseconds. 4 rows were returned.
Select a SQWRL query from the list above and press the 'Run' I
If the selected query generates a result, the result will appear ir

The SWRLAPI supports an OWL profile called OWL 2 RL and u
See the 'OWL 2 RL' subtab for more information on this reasoner.

Executing queries in this tab does not modify the ontology.

Using Drools for query execution.

See the Query3 tab to review results of the SQWRL query.
The query took 5656 milliseconds. 4 rows were returned.
```

Figure 10.3: SQWRL rules to query the safe following distance of vehicles

The result of this query is a list of vehicles and their safe following distance. The query retrieves vehicles and their safe distance from the forward vehicle, whose distance must be greater than 20 m as shown in Figure 10.4.


SSIoV (http://www.semanticweb.org/mac/ontologies/2021/
File Edit View Reasoner Tools Refactor Wind
SSIoV (http://www.semanticweb.org/mac/ontol
Search...
Active ontology × Entities × Object properties × Data
LTab ×
New Edit Clone Delete
SQWRL Queries OWL 2 RL Query Query1 Query2
Run

Edit

Name

Query3

Comment

Look for consequences caused by a Sybil Attack

Status

Ok

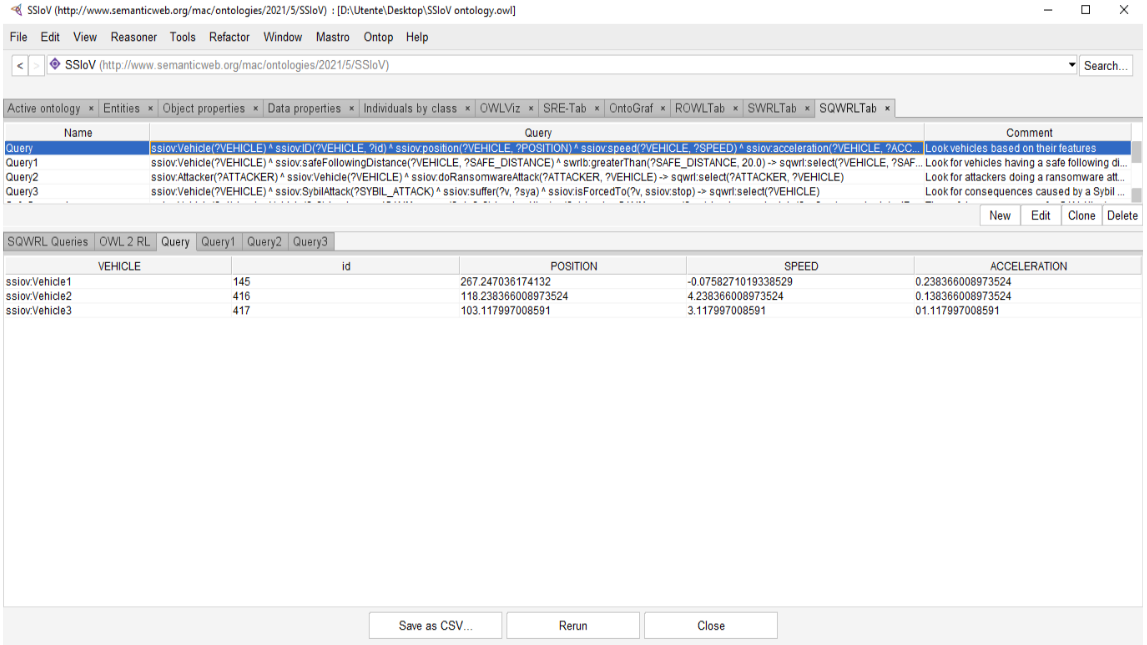
```
ssiov:Vehicle(?VEHICLE) ^ ssiov:SybilAttack(?SYBIL_ATTACK) ^ ssiov:suffer(?v, ?sya) ^ ssiov:isForcedTo(?v, ssiov:stop)
-> sqwrl:select(?VEHICLE)
```

Name			Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssi	ICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssi	HICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ s		Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssi		Look for consequences caused by a Sybil ...
SafeConseq1	ssiov:Vehicle(?v1) ^ ssiov:Vehi	ssiov:injectFa...	The safety consequence of a CAN Attack
SafeConseq10	ssiov:Vehicle(?v) ^ ssiov:SybilA		Teh safety consequence of a Sybil Attack
SafeConseq11	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Manipulation ...
SafeConseq12	ssiov:Vehicle(?v1) ^ ssiov:suffe	, ?sc) ^ ssio...	The safety consequence of a Man-in-the-mi...
SafeConseq13	ssiov:Vehicle(?v) ^ ssiov:suffer	) ^ ssiov:Saf...	The safety consequence of a Fake Basic S...
SafeConseq14	ssiov:Vehicle(?v) ^ ssiov:Spoof	SideVehicle(...	The safety consequence of a Spoofing Attack
SafeConseq2	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequence of a Sybil Attack
SafeConseq3	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety consequences of a Jamming Att...
SafeConseq4	ssiov:Vehicle(?v) ^ ssiov:suffer		The safety Consequences of a Remote Acc...

```
See the Query2 tab to review results of the SQWRL query.
The query took 3832 milliseconds. 4 rows were returned.
Select a SQWRL query from the list above and press the 'Run' I
If the selected query generates a result, the result will appear ir

The SWRLAPI supports an OWL profile called OWL 2 RL and u
See the 'OWL 2 RL' subtab for more information on this reasoner.

Executing queries in this tab does not modify the ontology.

Using Drools for query execution.

See the Query3 tab to review results of the SQWRL query.
The query took 5656 milliseconds. 4 rows were returned.
```

Figure 10.4: Execution and results of the query in Figure 10.3

```
ssiov:Attacker (?a) ∧ ssiov:Vehicle (?v)
ssiov:doRansomwareAttack (?a, ?v) → sqwrl:select (?a, ?v)
```



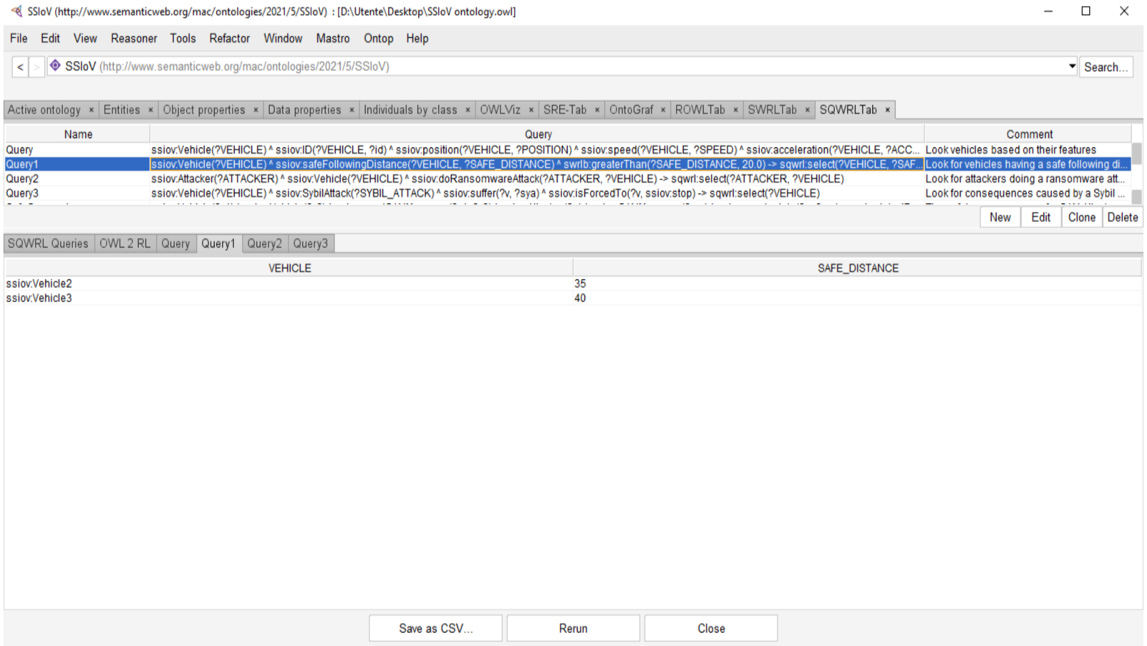
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × SRE-Tab × OntoGraf × ROWLTab × SWRLTab × SQWRLTab ×
New Edit Clone Delete
SQWRL Queries OWL 2 RL Query Query1 Query2 Query3
Save as CSV... Rerun Close

Name	Query	Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssiov:ID(?VEHICLE, ?id) ^ ssiov:position(?VEHICLE, ?POSITION) ^ ssiov:speed(?VEHICLE, ?SPEED) ^ ssiov:acceleration(?VEHICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssiov:safeFollowingDistance(?VEHICLE, ?SAFE_DISTANCE) ^ swrlb:greaterThan(?SAFE_DISTANCE, 20.0) -> sqwrl:select(?VEHICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ ssiov:Vehicle(?VEHICLE) ^ ssiov:doRansomwareAttack(?ATTACKER, ?VEHICLE) -> sqwrl:select(?ATTACKER, ?VEHICLE)	Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssiov:SybilAttack(?SYBIL_ATTACK) ^ ssiov:suffer(?v, ?sya) ^ ssiov:isForcedTo(?v, ssiov:stop) -> sqwrl:select(?VEHICLE)	Look for consequences caused by a Sybil ...

VEHICLE	id	POSITION	SPEED	ACCELERATION
ssiov:Vehicle1	145	267.247036174132	-0.0758271019338529	0.238366008973524
ssiov:Vehicle2	416	118.238366008973524	4.238366008973524	0.138366008973524
ssiov:Vehicle3	417	103.117997008591	3.117997008591	01.117997008591

Figure 10.5: SQWRL rules to query types of attacks against vehicles

The result of this query is a list of specific types of attacks launched by the attacker against vehicles, as shown in Figure 10.6.

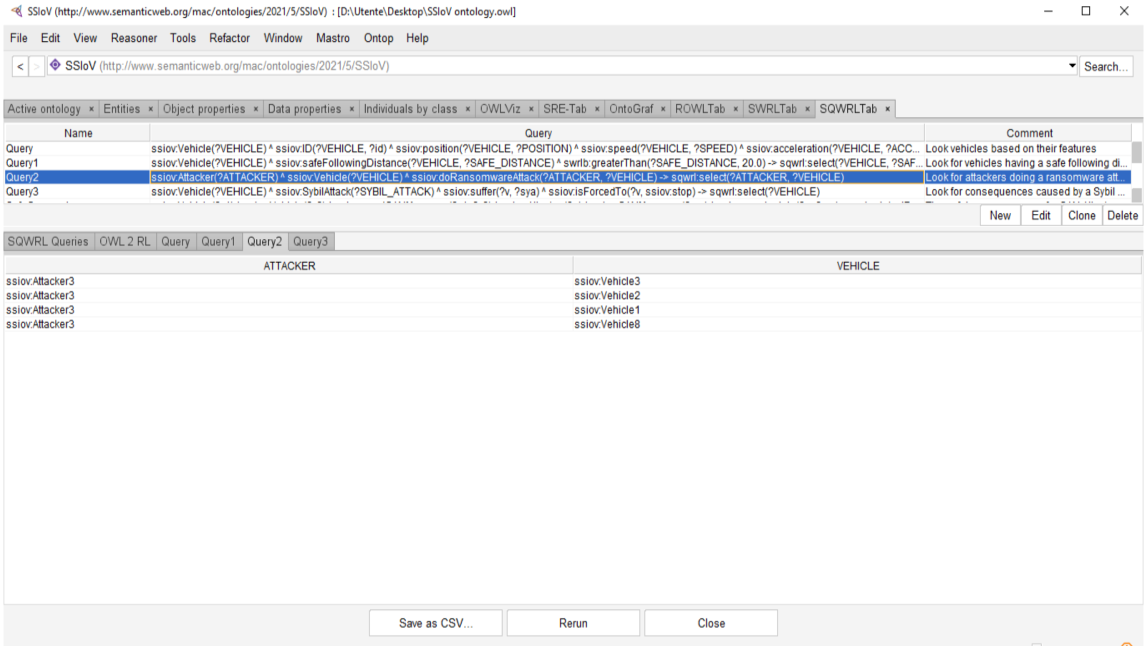

SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × SRE-Tab × OntoGraf × ROWLTab × SWRLTab × SQWRLTab ×
New Edit Clone Delete
SQWRL Queries OWL 2 RL Query Query1 Query2 Query3
Save as CSV... Rerun Close

Name	Query	Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssiov:ID(?VEHICLE, ?id) ^ ssiov:position(?VEHICLE, ?POSITION) ^ ssiov:speed(?VEHICLE, ?SPEED) ^ ssiov:acceleration(?VEHICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssiov:safeFollowingDistance(?VEHICLE, ?SAFE_DISTANCE) ^ swrlb:greaterThan(?SAFE_DISTANCE, 20.0) -> sqwrl:select(?VEHICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ ssiov:Vehicle(?VEHICLE) ^ ssiov:doRansomwareAttack(?ATTACKER, ?VEHICLE) -> sqwrl:select(?ATTACKER, ?VEHICLE)	Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssiov:SybilAttack(?SYBIL_ATTACK) ^ ssiov:suffer(?v, ?sya) ^ ssiov:isForcedTo(?v, ssiov:stop) -> sqwrl:select(?VEHICLE)	Look for consequences caused by a Sybil ...

VEHICLE	SAFE_DISTANCE
ssiov:Vehicle2	35
ssiov:Vehicle3	40

Figure 10.6: Execution and results of the query in Figure 10.5

ssiov:Vehicle(?v) ∧ ssiov:SybilAttack(?sya) ssiov:suffer(?v, ?sya) ∧ ssiov:isForcedTo(?v, ssiov:stop) → sqwrl:select(?v)

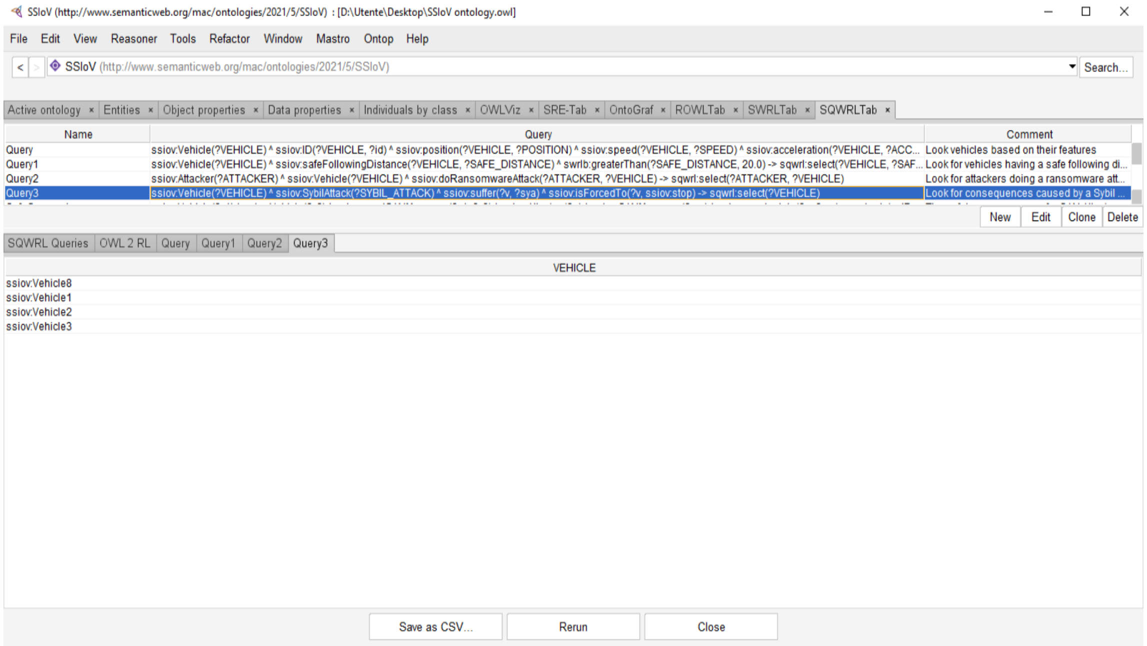

SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × SRE-Tab × OntoGraf × ROWLTab × SWRLTab × SQWRLTab ×
New Edit Clone Delete
SQWRL Queries OWL 2 RL Query Query1 Query2 Query3
Save as CSV... Rerun Close

Name	Query	Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssiov:ID(?VEHICLE, ?id) ^ ssiov:position(?VEHICLE, ?POSITION) ^ ssiov:speed(?VEHICLE, ?SPEED) ^ ssiov:acceleration(?VEHICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssiov:safeFollowingDistance(?VEHICLE, ?SAFE_DISTANCE) ^ swrlb:greaterThan(?SAFE_DISTANCE, 20.0) -> sqwrl:select(?VEHICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ ssiov:Vehicle(?VEHICLE) ^ ssiov:doRansomwareAttack(?ATTACKER, ?VEHICLE) -> sqwrl:select(?ATTACKER, ?VEHICLE)	Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssiov:SybilAttack(?SYBIL_ATTACK) ^ ssiov:suffer(?v, ?sya) ^ ssiov:isForcedTo(?v, ssiov:stop) -> sqwrl:select(?VEHICLE)	Look for consequences caused by a Sybil ...

ATTACKER	VEHICLE
ssiov:Attacker3	ssiov:Vehicle3
ssiov:Attacker3	ssiov:Vehicle2
ssiov:Attacker3	ssiov:Vehicle1
ssiov:Attacker3	ssiov:Vehicle8

Figure 10.7: SQWRL rules to query the vehicle that suffers safety consequences due to a security attack

The result of this query is a list of vehicles that are forced to stop due to the sybil attack, as shown in Figure 10.8.



SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV) : [D:\Utente\Desktop\SSIoV ontology.owl]
File Edit View Reasoner Tools Refactor Window Mastro Ontop Help
SSIoV (http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV)
Search...
Active ontology × Entities × Object properties × Data properties × Individuals by class × OWLViz × SRE-Tab × OntoGraf × ROWLTab × SWRLTab × SQWRLTab ×
New
Edit
Clone
Delete
SQWRL Queries OWL 2 RL Query Query1 Query2 Query3
Save as CSV...
Rerun
Close

Name	Query	Comment
Query	ssiov:Vehicle(?VEHICLE) ^ ssiov:ID(?VEHICLE, ?id) ^ ssiov:position(?VEHICLE, ?POSITION) ^ ssiov:speed(?VEHICLE, ?SPEED) ^ ssiov:acceleration(?VEHICLE, ?ACC...	Look vehicles based on their features
Query1	ssiov:Vehicle(?VEHICLE) ^ ssiov:safeFollowingDistance(?VEHICLE, ?SAFE_DISTANCE) ^ swrlb:greaterThan(?SAFE_DISTANCE, 20.0) -> sqwrl:select(?VEHICLE, ?SAF...	Look for vehicles having a safe following di...
Query2	ssiov:Attacker(?ATTACKER) ^ ssiov:Vehicle(?VEHICLE) ^ ssiov:doRansomwareAttack(?ATTACKER, ?VEHICLE) -> sqwrl:select(?ATTACKER, ?VEHICLE)	Look for attackers doing a ransomware att...
Query3	ssiov:Vehicle(?VEHICLE) ^ ssiov:SybilAttack(?SYBIL_ATTACK) ^ ssiov:suffer(?v, ?sya) ^ ssiov:isForcedTo(?v, ssiov:stop) -> sqwrl:select(?VEHICLE)	Look for consequences caused by a Sybil ...

VEHICLE
ssiov:Vehicle8
ssiov:Vehicle1
ssiov:Vehicle2
ssiov:Vehicle3

Figure 10.8: Execution and results of the query of the Figure 10.7

Chapter 11

The Semantic-based AI Reasoning Tool

This chapter aims to present the outcome of our work. We work about a reasoner engine that is able to infer new facts from ontology and data. Our reasoning tool consists of six steps to make our semantic-based AI reasoning tool as shown in Figure 11.1.

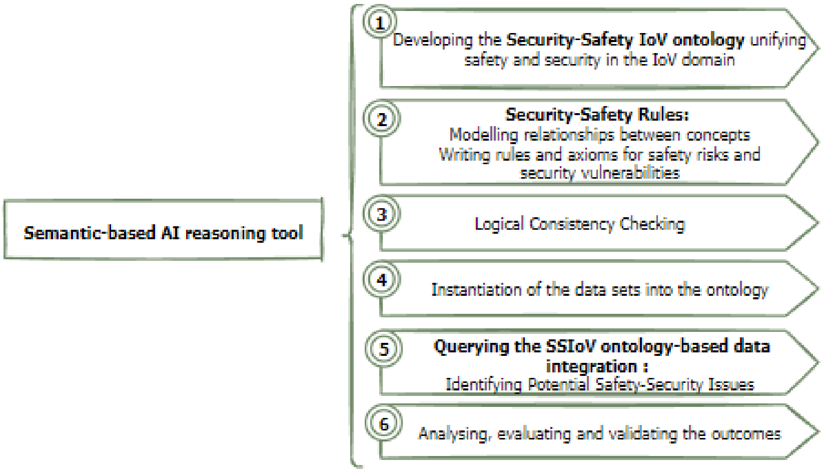

Semantic-based AI reasoning tool
1
Developing the Security-Safety IoV ontology unifying safety and security in the IoV domain
2
Security-Safety Rules:
Modelling relationships between concepts
Writing rules and axioms for safety risks and security vulnerabilities
3
Logical Consistency Checking
4
Instantiation of the data sets into the ontology
5
Querying the SSIoV ontology-based data integration :
Identifying Potential Safety-Security Issues
6
Analysing, evaluating and validating the outcomes

Figure 11.1: Methodology for our semantic-based AI reasoning tool

11.1 The Semantic-based AI Reasoning Tool Architecture

Figure 11.2 shows the architecture of the AI reasoning engine implementation.

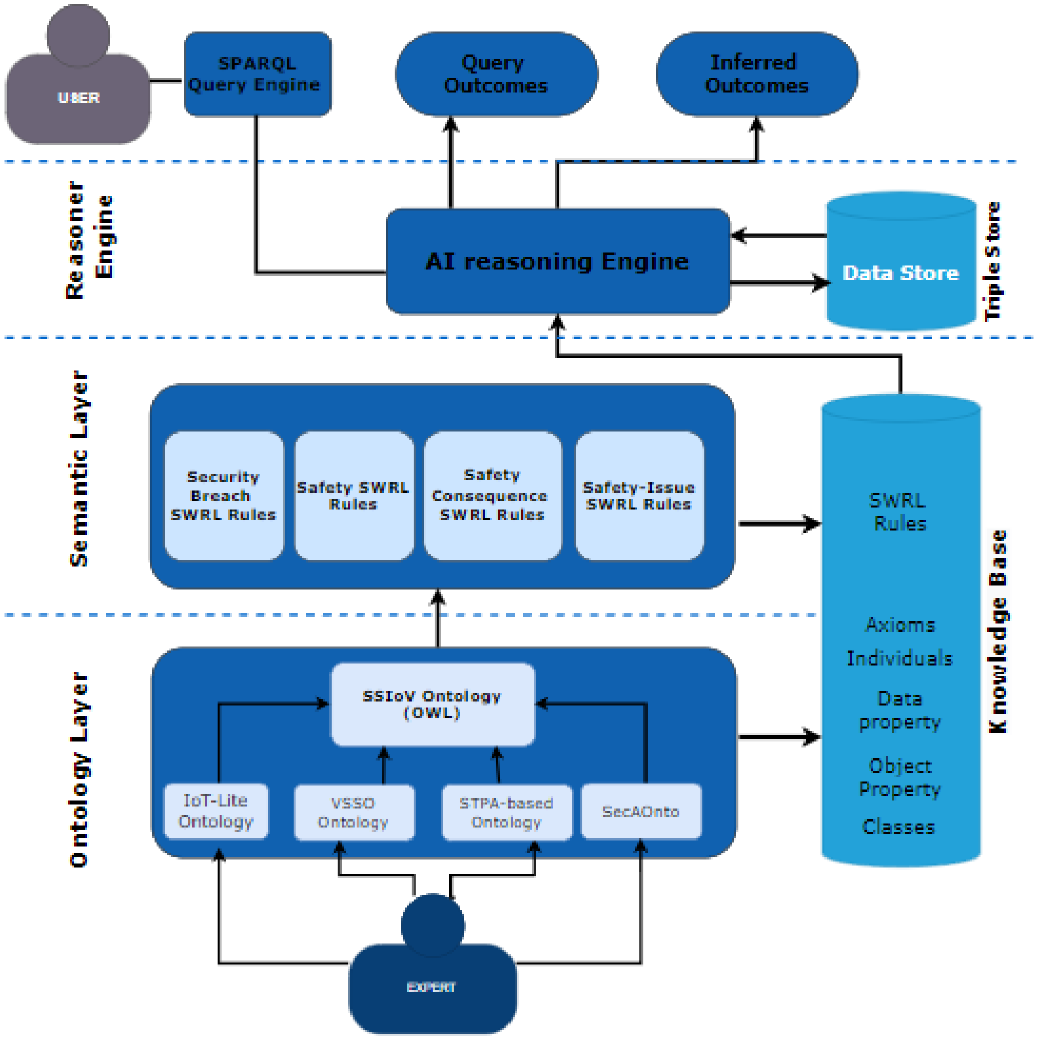


Figure 11.2: The architecture of the AI reasoning tool

Here, we present the AI reasoner engine architecture, which is divided into three layers: a) Ontology Layer; b) Semantic Layer; c) Reasoner Layer.

The first layer is the result of the first step of our methodology (see Figure 11.1) where we reuse and combine different ontologies. The second layer is about the SWRL rules, which we develop under 4 types: a) Security Breach Rules; b) Safety Rules; c) Safety Consequences Rules; d) Safety Issue Rules. The formalised data into the SSIoV ontology, and the SWRL Rules make up the knowledge base.

Then, the AI reasoning engine runs on new data using the knowledge base. The reasoner can detect the consequences of the AV's actions. For example, if the AV does not maintain the minimum safe distance, the reasoning engine recommends the vehicle to stop reasoning based on the safety rules. Therefore, it applies rules to data, inferring new facts.

Also, the user can use the system by querying it. For example, users can look for vehicles that have been attacked by ransomware, or search for vehicles that have been forced to comply with some safety rules.

The flow of the AI reasoning engine is shown in Figure 11.3.

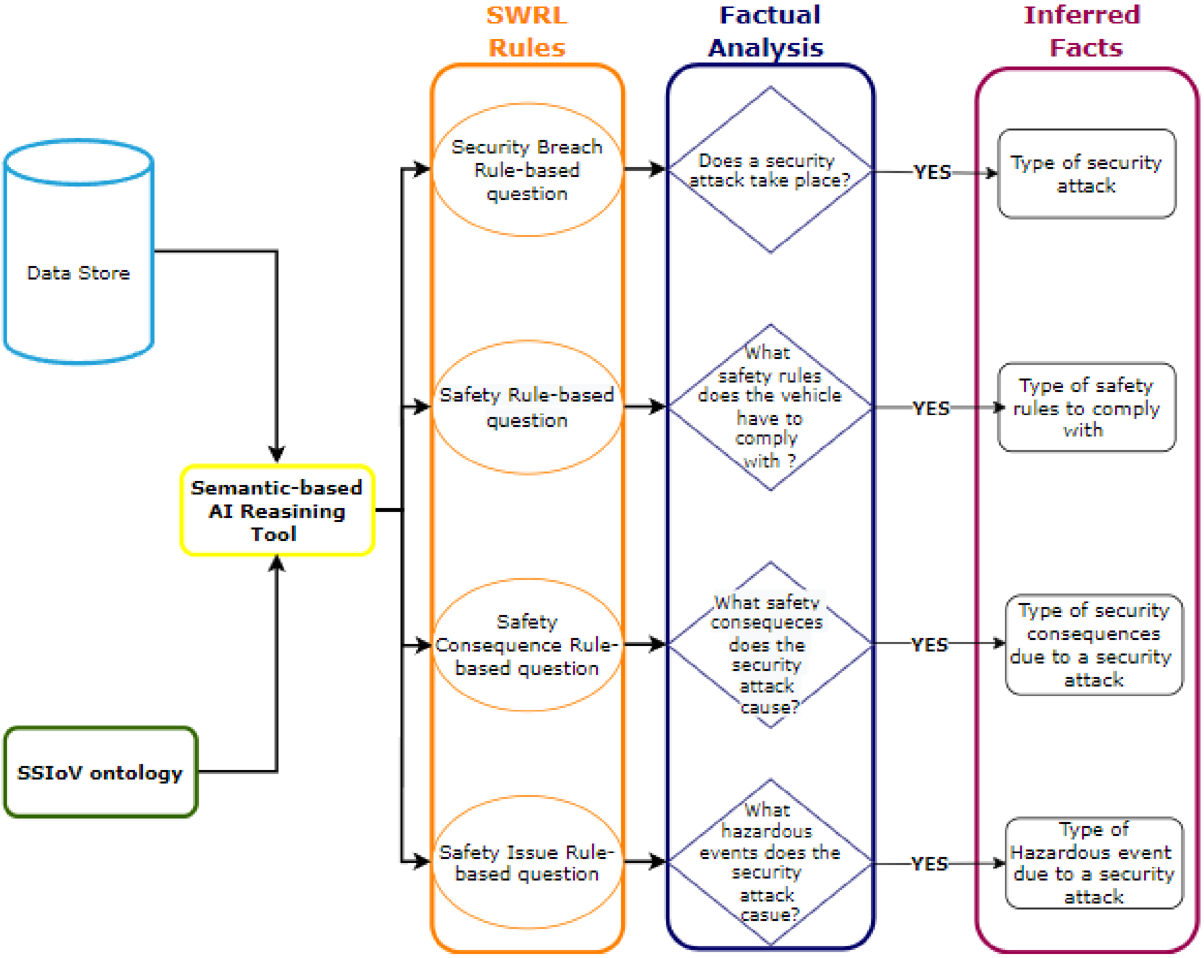

SWRL Rules
Factual Analysis
Inferred Facts
Data Store
Semantic-based AI Reasining Tool
SSIoV ontology
Security Breach Rule-based question
Safety Rule-based question
Safety Consequence Rule-based question
Safety Issue Rule-based question
Does a security attack take place?
What safety rules does the vehicle have to comply with ?
What safety conseqeces does the security attack cause?
What hazardous events does the security attack casue?
YES
YES
YES
YES
Type of security attack
Type of safety rules to comply with
Type of security consequences due to a security attack
Type of Hazardous event due to a security attack

Figure 11.3: The reasoning engine flow

The reasoner operates on the data using the SSIoV ontology. Then, it analyses the data according to 4 types of rules. If the reasoner satisfies certain conditions, it analyses the data as follows.

The **Security Breach Rules** leads the reasoner to look for events from which can infer what type of security attack occurred. Then, the **Safety Rules** lead the reasoner to look for some events from which it can infer the type of safety rules that AV must comply with. Also, the **Safety Consequence Rule** leads the reasoner to look for the type of the security consequence due to a security attack. The **Safety Issue Rules** leads to looking for the types of hazardous events due to a security attack.

11.2 The Implementation of AI Reasoning Tool to Use Case n.1

We use our first use case for the AI reasoner implementation. AVs broadcast beacon GPS signal messages to inform of their presence. In Figure 6.6, an *attacker* sends a falsified *GPS signal* (that is a type of *GNNS* signal [8]) of its own position to the *target vehicle*. The *spoofing* attack here threatens the *authenticity* of the *sensors* signal. The GPS signal (falsely) mentions that the position of the attacker is very close to that of the target vehicle. The latter then applies a safety measure (*emergency stopping manoeuvre*) to ensure a *safety property* (*safe stopping distance*) that leads to a *rear-end collision* with the rear vehicle (*hazard*).

The use case consists of 4 Rules, namely:

1. SecBreach Rule
2. SafeRule
3. SafeConsequ Rule
4. SafeIssue Rule

The AI reasoning tool can identify these 4 rules on data. We add the data in Protégé (in the form of Instances), and we launch the reasoner to establish if it can understand the following relationships:

1. SecBreach Rule ⇒ the **type of security attack**;
2. SafeRule ⇒ the **safety action** that must be respected by the AVs;
3. SafeConseq Rule ⇒ the **causal relationships between a security attack and security consequences** (without considering the people injures);
4. SafeIssue Rule ⇒ the **causal relationship between a security attack and damage events**.

We formalise the rules in a separate way. Each of these is independent of each other. Therefore, our reasoner can understand different events - related to the 4 rules - separately. The working principles of the reasoner are shown in Figure 11.4.

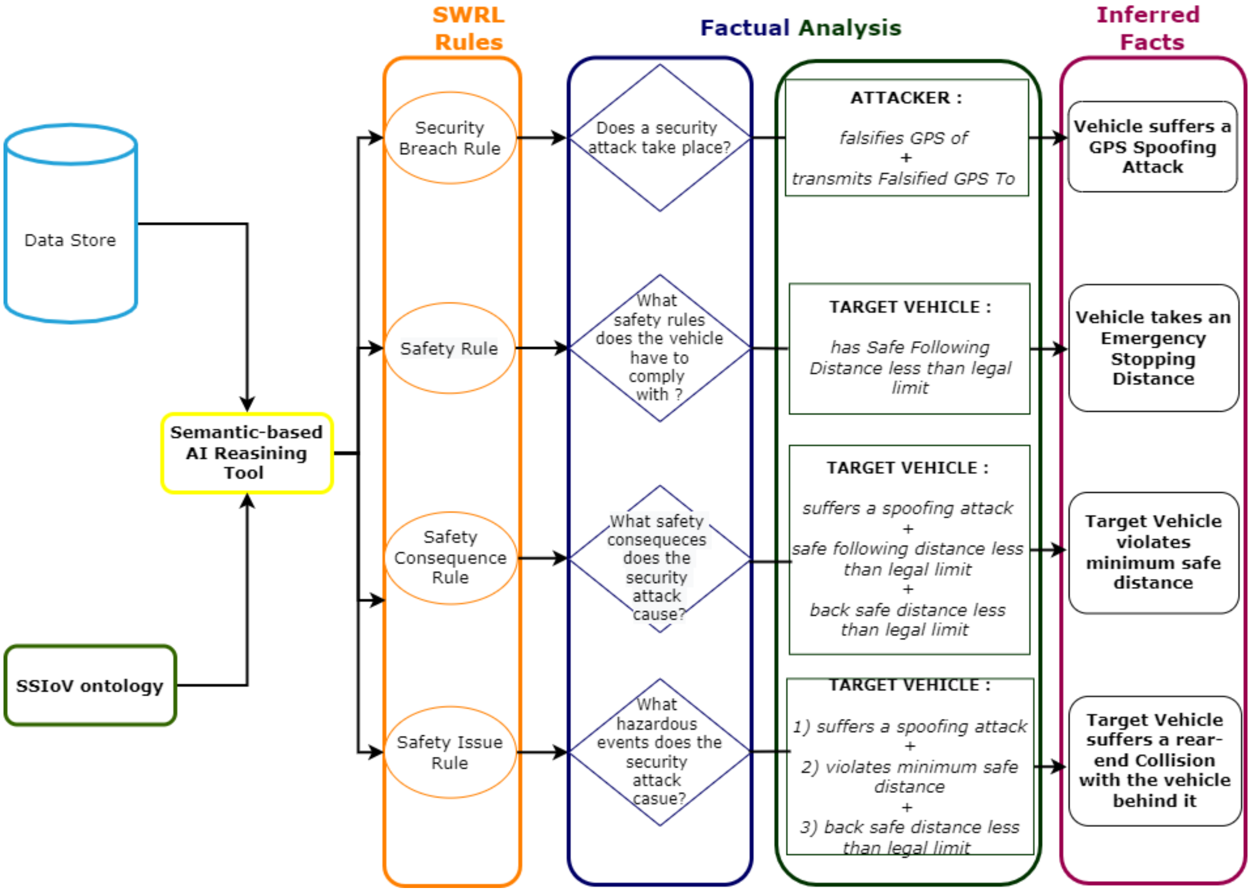
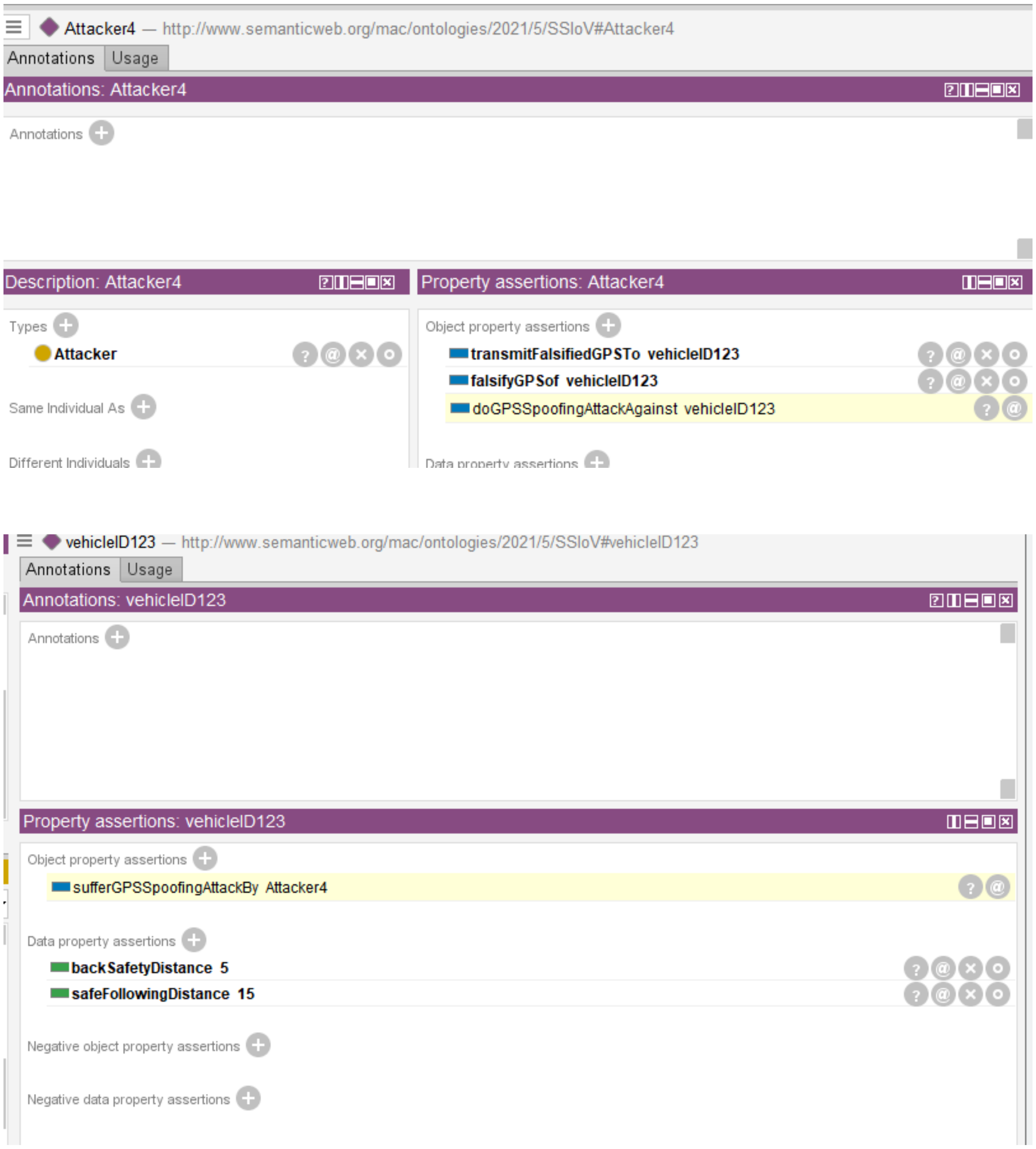

SWRL Rules
Factual Analysis
Inferred Facts
Data Store
SSIoV ontology
Semantic-based AI Reasining Tool
Security Breach Rule
Safety Rule
Safety Consequence Rule
Safety Issue Rule
Does a security attack take place?
What safety rules does the vehicle have to comply with ?
What safety conseqences does the security attack cause?
What hazardous events does the security attack casue?
ATTACKER : falsifies GPS of + transmits Falsified GPS To
TARGET VEHICLE : has Safe Following Distance less than legal limit
TARGET VEHICLE : suffers a spoofing attack + safe following distance less than legal limit + back safe distance less than legal limit
TARGET VEHICLE : 1) suffers a spoofing attack + 2) violates minimum safe distance + 3) back safe distance less than legal limit
Vehicle suffers a GPS Spoofing Attack
Vehicle takes an Emergency Stopping Distance
Target Vehicle violates minimum safe distance
Target Vehicle suffers a rear-end Collision with the vehicle behind it

Figure 11.4: The inferred fact of running the reasoner - use case n.1

- **SecBreach Rule-based AI reasoning tool**

  1. We know that an Attacker(=Attacker4) falsifies the GPS signal and transmits it to the target vehicle (=vehicleID123);
  2. The reasoner knows that if an Attacker falsifies a GPS signal and transmits it to the target vehicle (=SecBreach Rule), then the Target Vehicle suffers a GPS spoofing attack;

3. Therefore, the reasoner **can detect the security attack**, which took place as shown in Figure 11.5 (where the inferred fact is highlighted in light pink) and in Figure 11.6 (where the reasoner explains the inferred outcome).



Attacker4 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#Attacker4

Annotations: Attacker4

Description: Attacker4

Types
- Attacker

Property assertions: Attacker4

Object property assertions
- transmitFalsifiedGPSTo vehicleID123
- falsifyGPSof vehicleID123
- doGPSSpoofingAttackAgainst vehicleID123

vehicleID123 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#vehicleID123

Annotations: vehicleID123

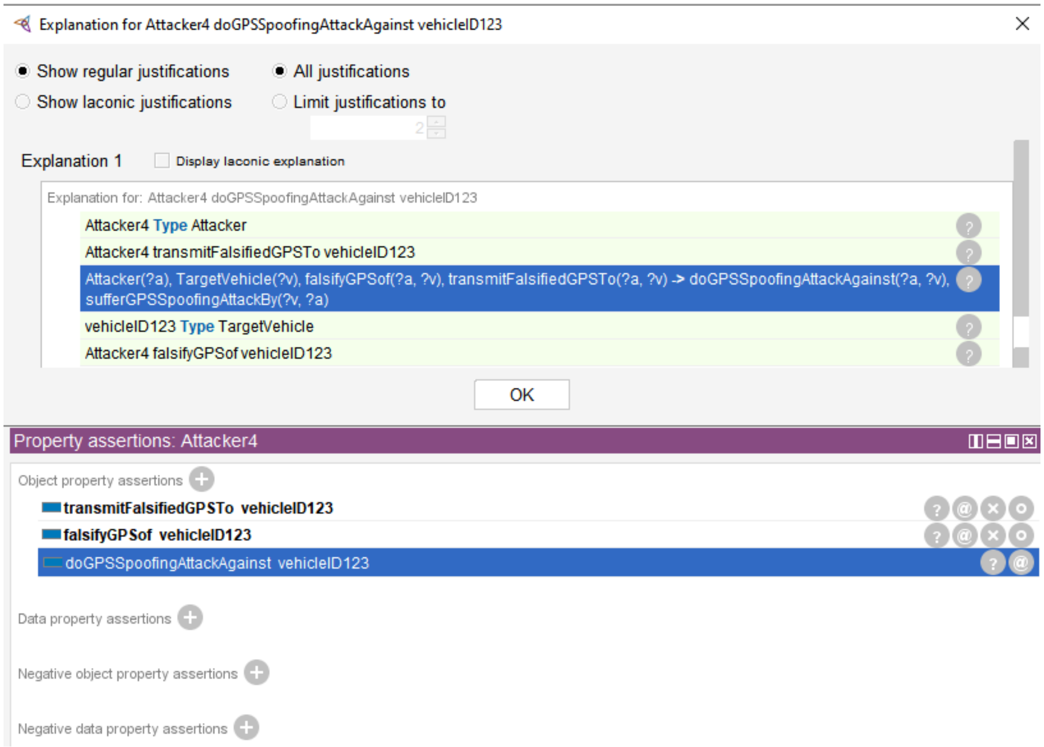
Property assertions: vehicleID123

Object property assertions
- sufferGPSSpoofingAttackBy Attacker4

Data property assertions
- backSafetyDistance 5
- safeFollowingDistance 15

Negative object property assertions
Negative data property assertions

Figure 11.5: Inferred axioms of running Pellet plug-in - use case n.1



Explanation for Attacker4 doGPSSpoofingAttackAgainst vehicleID123

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: Attacker4 doGPSSpoofingAttackAgainst vehicleID123

Attacker4 Type Attacker

Attacker4 transmitFalsifiedGPSTo vehicleID123

Attacker(?a), TargetVehicle(?v), falsifyGPSof(?a, ?v), transmitFalsifiedGPSTo(?a, ?v) -> doGPSSpoofingAttackAgainst(?a, ?v), sufferGPSSpoofingAttackBy(?v, ?a)

vehicleID123 Type TargetVehicle

Attacker4 falsifyGPSof vehicleID123

Property assertions: Attacker4

Object property assertions

transmitFalsifiedGPSTo vehicleID123

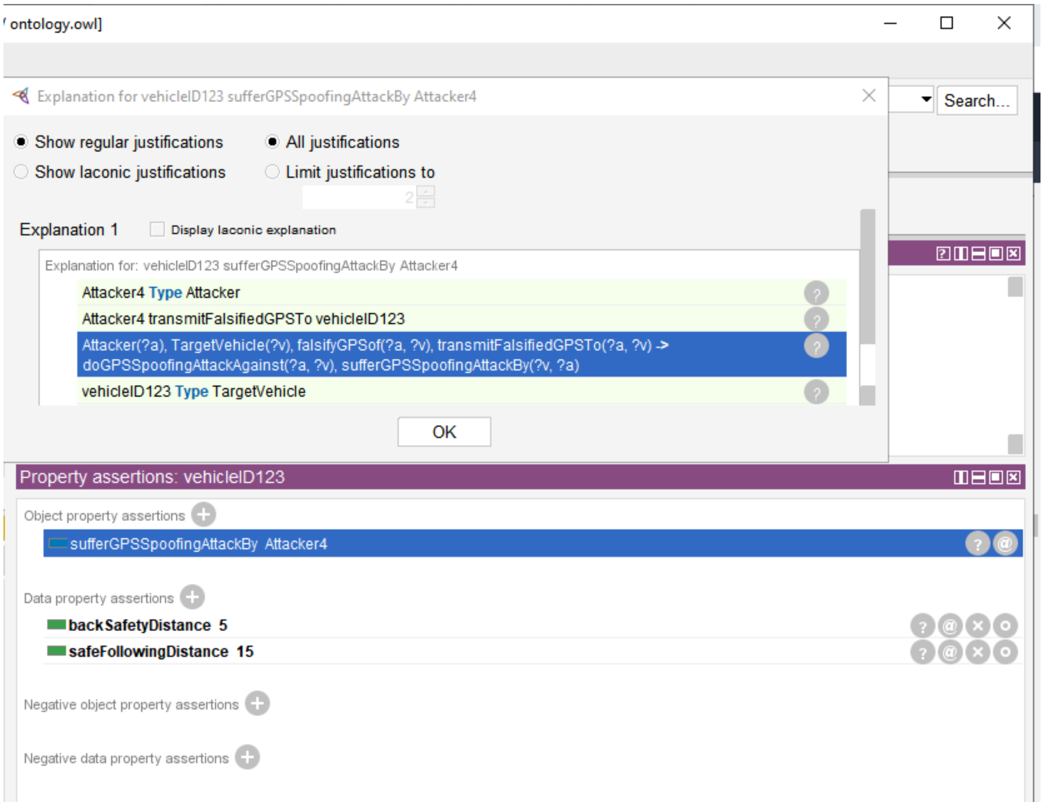
falsifyGPSof vehicleID123

doGPSSpoofingAttackAgainst vehicleID123

Data property assertions

Negative object property assertions

Negative data property assertions



Explanation for vehicleID123 sufferGPSSpoofingAttackBy Attacker4

● Show regular justifications
○ Show laconic justifications

● All justifications
○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: vehicleID123 sufferGPSSpoofingAttackBy Attacker4

- Attacker4 Type Attacker
- Attacker4 transmitFalsifiedGPSTo vehicleID123
- Attacker(?a), TargetVehicle(?v), falsifyGPSof(?a, ?v), transmitFalsifiedGPSTo(?a, ?v) -> doGPSSpoofingAttackAgainst(?a, ?v), sufferGPSSpoofingAttackBy(?v, ?a)
- vehicleID123 Type TargetVehicle

Property assertions: vehicleID123

Object property assertions

- sufferGPSSpoofingAttackBy Attacker4

Data property assertions

- backSafetyDistance 5
- safeFollowingDistance 15

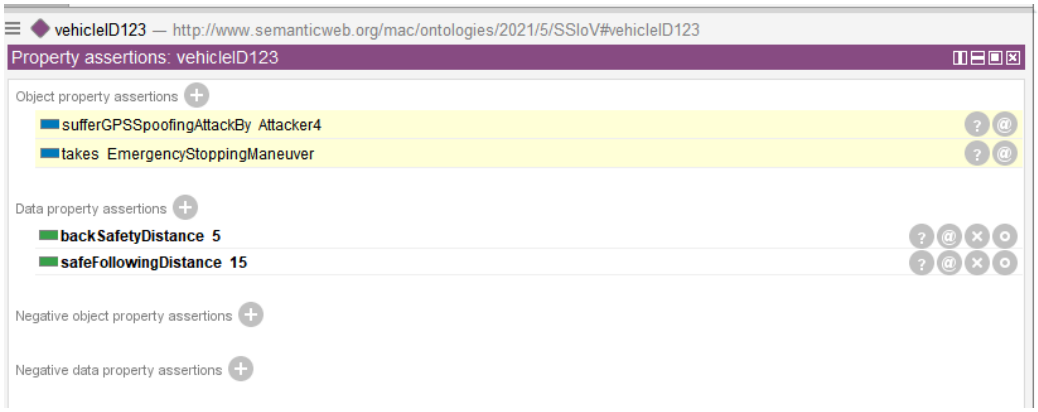
Negative object property assertions

Negative data property assertions

Figure 11.6: Inferred axioms explanation of running Pellet plug-in - use case n.1

- **SafeRule-based AI reasoning tool**

  1. We know that the AV takes an emergency stopping manoeuvre - slowing down suddenly, if it detects that the minimum safe distance with the vehicle in front of it is not complied;
  2. The reasoner knows that if an AV do not maintain the minimum safe following distance, then it must take an emergency stopping manoeuvre for slowing down (=SafeRule);
  3. Therefore, the reasoner **can advise the AV about the type of security actions** it must obey.



vehicleID123 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#vehicleID123

Property assertions: vehicleID123

Object property assertions

sufferGPSSpoofingAttackBy Attacker4

takes EmergencyStoppingManeuver

Data property assertions

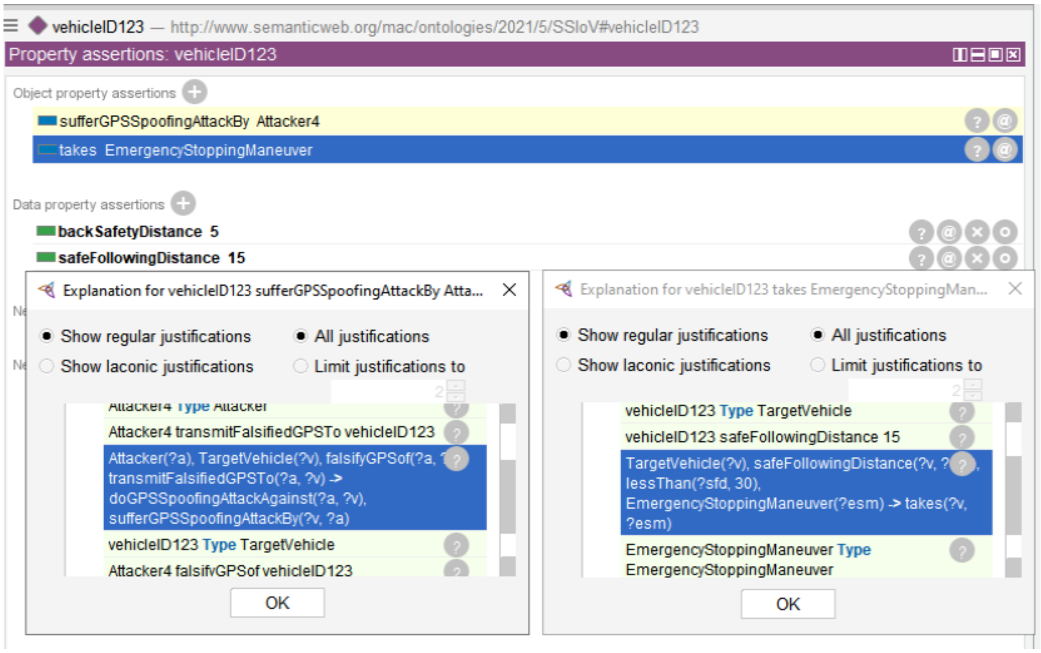
backSafetyDistance 5

safeFollowingDistance 15

Negative object property assertions

Negative data property assertions

Figure 11.7: Inferred axioms of running Pellet plug-in - use case n.1



vehicleID123 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#vehicleID123

Property assertions: vehicleID123

Object property assertions

sufferGPSSpoofingAttackBy Attacker4

takes EmergencyStoppingManeuver

Data property assertions

backSafetyDistance 5

safeFollowingDistance 15

Explanation for vehicleID123 sufferGPSSpoofingAttackBy Atta...

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

Attacker4 Type Attacker

Attacker4 transmitFalsifiedGPSTo vehicleID123

Attacker(?a), TargetVehicle(?v), falsifyGPSof(?a, ?
transmitFalsifiedGPSTo(?a, ?v) ->
doGPSSpoofingAttackAgainst(?a, ?v),
sufferGPSSpoofingAttackBy(?v, ?a)

vehicleID123 Type TargetVehicle

Attacker4 falsifvGPSof vehicleID123

Explanation for vehicleID123 takes EmergencyStoppingMan...

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

vehicleID123 Type TargetVehicle

vehicleID123 safeFollowingDistance 15

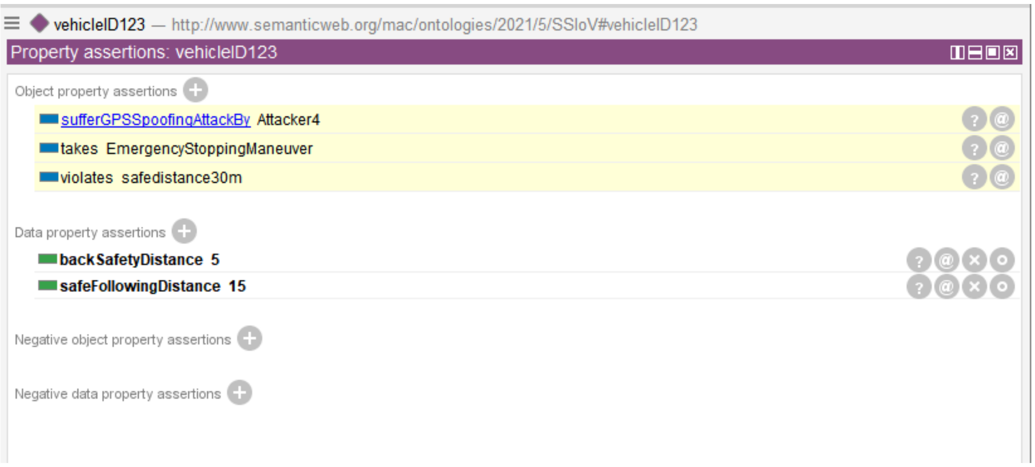
TargetVehicle(?v), safeFollowingDistance(?v, ?
lessThan(?sfd, 30),
EmergencyStoppingManeuver(?esm) -> takes(?v,
?esm)

EmergencyStoppingManeuver Type
EmergencyStoppingManeuver

Figure 11.8: Inferred axioms explanation of running Pellet plug-in - use case n.1

- **SafeConseq Rule-based AI reasoning tool**

  1. We know that the AV is subject to spoofing attack, does not comply with safe distance restrictions from other AVs;
  2. The reasoner knows that if an AV suffers a spoofing attack and it does not comply with the safe distance limit, then it violates the minimum safe distance (=SafeConseq Rule);
  3. Hence, the reasoner **can detect the causal relationships** between a spoofing attack and the failure to meet the minimum safe distance as shown in Figure 11.9 (where the inferred fact is highlighted in light pink) and Figure 11.10 (where the reasoner explains the inferred outcome).



vehicleID123 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#vehicleID123

Property assertions: vehicleID123

Object property assertions

- sufferGPSSpoofingAttackBy Attacker4
- takes EmergencyStoppingManeuver
- violates safedistance30m

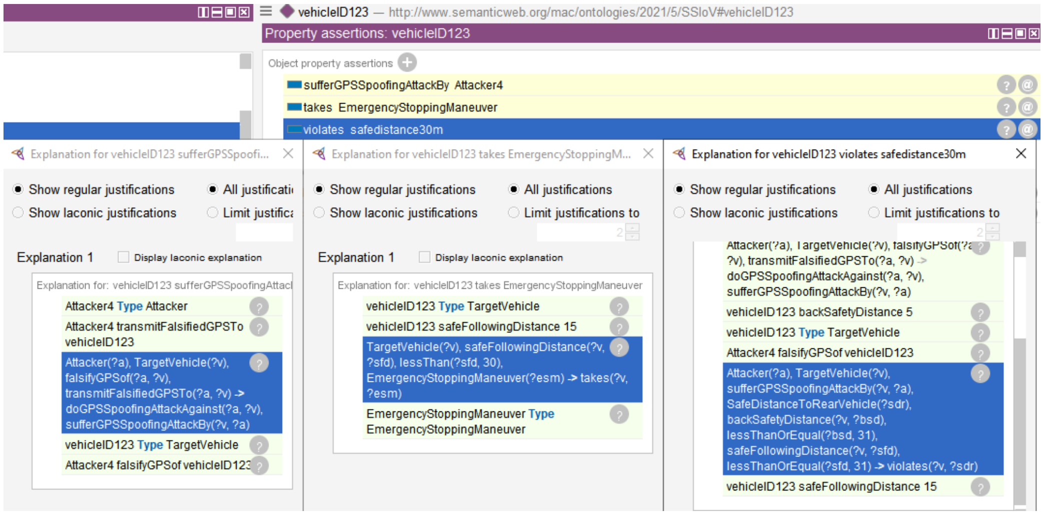
Data property assertions

- backSafetyDistance 5
- safeFollowingDistance 15

Negative object property assertions

Negative data property assertions

Figure 11.9: Inferred axioms of running Pellet plug-in - use case n.1



vehicleID123 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#vehicleID123

Property assertions: vehicleID123

Object property assertions

- sufferGPSSpoofingAttackBy Attacker4
- takes EmergencyStoppingManeuver
- violates safedistance30m

Explanation for vehicleID123 sufferGPSSpoofi...

● Show regular justifications ● All justificati

○ Show laconic justifications ○ Limit justifica

Explanation 1 □ Display laconic explanation

Explanation for: vehicleID123 sufferGPSSpoofingAttacl

- Attacker4 Type Attacker
- Attacker4 transmitFalsifiedGPSTo vehicleID123
- Attacker(?a), TargetVehicle(?v), falsifyGPSof(?a, ?v), transmitFalsifiedGPSTo(?a, ?v) -> doGPSSpoofingAttackAgainst(?a, ?v), sufferGPSSpoofingAttackBy(?v, ?a)
- vehicleID123 Type TargetVehicle
- Attacker4 falsifyGPSof vehicleID123

Explanation for vehicleID123 takes EmergencyStoppingM...

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: vehicleID123 takes EmergencyStoppingManeuver

- vehicleID123 Type TargetVehicle
- vehicleID123 safeFollowingDistance 15
- TargetVehicle(?v), safeFollowingDistance(?v, ?sfd), lessThan(?sfd, 30), EmergencyStoppingManeuver(?esm) -> takes(?v, ?esm)
- EmergencyStoppingManeuver Type EmergencyStoppingManeuver

Explanation for vehicleID123 violates safedistance30m

● Show regular justifications ● All justifications

○ Show laconic justifications ○ Limit justifications to 2

- Attacker(?a), TargetVehicle(?v), falsifyGPSof(?a, ?v), transmitFalsifiedGPSTo(?a, ?v) -> doGPSSpoofingAttackAgainst(?a, ?v), sufferGPSSpoofingAttackBy(?v, ?a)
- vehicleID123 backSafetyDistance 5
- vehicleID123 Type TargetVehicle
- Attacker4 falsifyGPSof vehicleID123
- Attacker(?a), TargetVehicle(?v), sufferGPSSpoofingAttackBy(?v, ?a), SafeDistanceToRearVehicle(?sdr), backSafetyDistance(?v, ?bsd), lessThanOrEqual(?bsd, 31), safeFollowingDistance(?v, ?sfd), lessThanOrEqual(?sfd, 31) -> violates(?v, ?sdr)
- vehicleID123 safeFollowingDistance 15

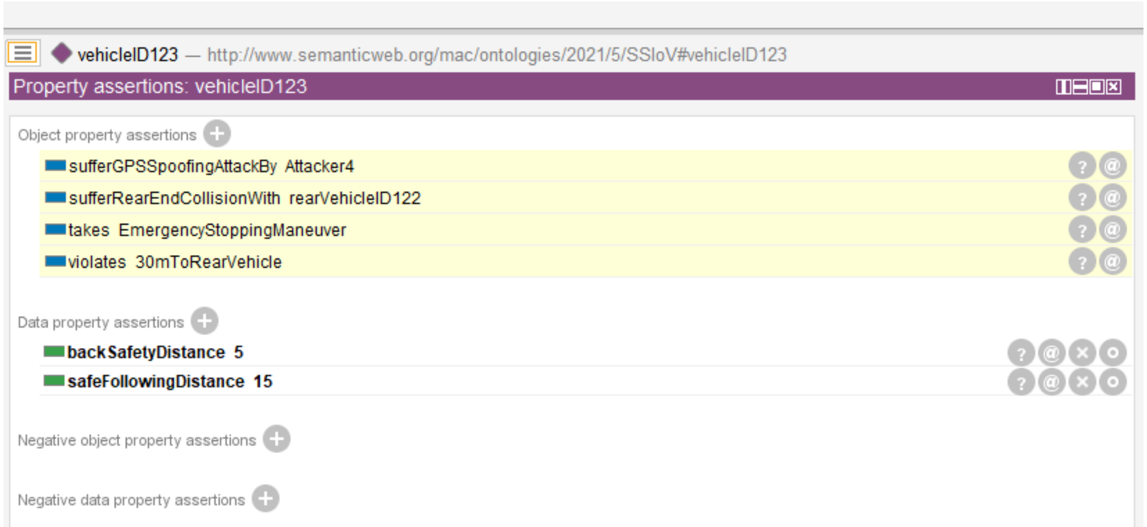
Figure 11.10: Inferred axioms explanation of running Pellet plug-in - use case n.1

- **SafeIssue-based AI reasoning tool**

  1. We know that the AV is subject to a spoofing attack, violates the minimum safe distance.
  2. The reasoner knows that if an AV suffers a spoofing attack, and

it does not comply with the minimum safe distance, then the AV is involved in a rear-end collision (=SafeIssue Rule).

3. Therefore, the reasoner **can detect the causal relationships** between a spoofing attack, the failure to meet the minimum safe distance and the rear-end Collision event as shown in Figure 11.11 (where the inferred fact is highlighted in light pink) and Figure 11.12 (where the reasoner explains the inferred outcome).



vehicleID123 — http://www.semanticweb.org/mac/ontologies/2021/5/SSIoV#vehicleID123

Property assertions: vehicleID123

Object property assertions

- sufferGPSSpoofingAttackBy Attacker4
- sufferRearEndCollisionWith rearVehicleID122
- takes EmergencyStoppingManeuver
- violates 30mToRearVehicle

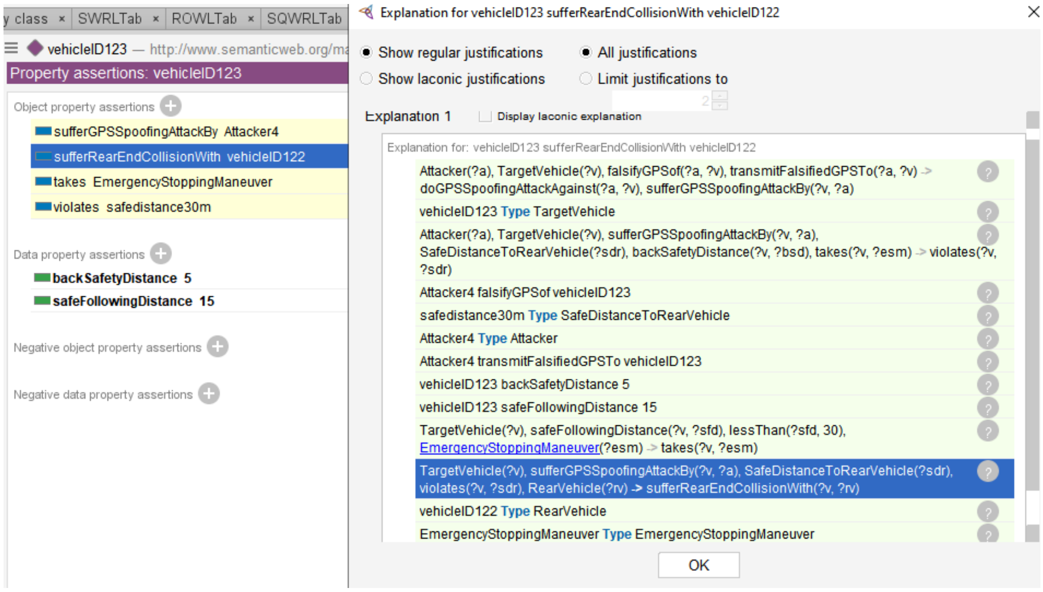
Data property assertions

- **backSafetyDistance 5**
- **safeFollowingDistance 15**

Negative object property assertions

Negative data property assertions

Figure 11.11: Inferred axioms of running Pellet plug-in - use case n.1



vehicleID123 — http://www.semanticweb.org/ma

Property assertions: vehicleID123

Object property assertions

- sufferGPSSpoofingAttackBy Attacker4
- sufferRearEndCollisionWith vehicleID122
- takes EmergencyStoppingManeuver
- violates safedistance30m

Data property assertions

- **backSafetyDistance 5**
- **safeFollowingDistance 15**

Negative object property assertions

Negative data property assertions

Explanation for vehicleID123 sufferRearEndCollisionWith vehicleID122

● Show regular justifications ● All justifications
○ Show laconic justifications ○ Limit justifications to 2

Explanation 1 □ Display laconic explanation

Explanation for: vehicleID123 sufferRearEndCollisionWith vehicleID122

- Attacker(?a), TargetVehicle(?v), falsifyGPSof(?a, ?v), transmitFalsifiedGPSTo(?a, ?v) -> doGPSSpoofingAttackAgainst(?a, ?v), sufferGPSSpoofingAttackBy(?v, ?a)
- vehicleID123 Type TargetVehicle
- Attacker(?a), TargetVehicle(?v), sufferGPSSpoofingAttackBy(?v, ?a), SafeDistanceToRearVehicle(?sdr), backSafetyDistance(?v, ?bsd), takes(?v, ?esm) -> violates(?v, ?sdr)
- Attacker4 falsifyGPSof vehicleID123
- safedistance30m Type SafeDistanceToRearVehicle
- Attacker4 Type Attacker
- Attacker4 transmitFalsifiedGPSTo vehicleID123
- vehicleID123 backSafetyDistance 5
- vehicleID123 safeFollowingDistance 15
- TargetVehicle(?v), safeFollowingDistance(?v, ?sfd), lessThan(?sfd, 30), EmergencyStoppingManeuver(?esm) -> takes(?v, ?esm)
- TargetVehicle(?v), sufferGPSSpoofingAttackBy(?v, ?a), SafeDistanceToRearVehicle(?sdr), violates(?v, ?sdr), RearVehicle(?rv) -> sufferRearEndCollisionWith(?v, ?rv)
- vehicleID122 Type RearVehicle
- EmergencyStoppingManeuver Type EmergencyStoppingManeuver

Figure 11.12: Inferred axioms explanation of running Pellet plug-in - use case n.1

Chapter 12

Conclusion

IoV has become the core network for AV scenarios. However, in order to exploit this network, we need to face the security challenges (raised from the IoV connectivity) and their impact on safety. We develop a methodology using ontology and reasoning rules to investigate the link between safety and security, specifically for AVs.

In this paper, we present our findings aimed at providing a semantic approach to enhance cyber security in the automotive domain. This work aims to provide a tool for improving preventive cyber defence capabilities in the IoV and AVs domains. Based on integrated security-safety ontology and corresponding rules, the tool highlights cyber security vulnerabilities that lead to safety risks. This work contributes to improve security of critical road infrastructures for IoV.

12.1 Research questions and contributions

We have set some research questions as follow:

- ⋄ Can we design and implement a semantic-based AI reasoning tool to analyse causal security-safety issues?

    1. How can we model the knowledge of the safety and security domains to perform a semantic and automatic cyber security analysis, applied to AVs and IoV?
    2. Can we automatically identify security holes by reasoning on safety rules and vice-versa?

Our contributions answer these questions as we explain below.

Sub question 1: *How can we model the knowledge of the safety and security domains to perform a semantic and automatic cyber security analysis, applied to AVs and IoV?*

Contribution to sub question 1: We develop the SSIoV ontology, which combines four current ontologies and lets to model IoV, AV, safety and security domains knowledge (see ch. 6).

Contribution to sub question 1: We develop four sets of SWRL rules to establish relationships between concepts. These rules concern both security and safety domain. Therefore, to reason on security breach and its causal impact on safety, we model four types of reasoning rules: a) **Security reasoning rules**: we identify and model security rules describing security vulnerabilities; b) **Safety reasoning rules**: we model rules that describe safety behaviour that is applied by vehicles when they detect safety risks; c) **Safety risks reasoning rules**: we model rules where those safety behaviour lead to safety risks; d) **Security-safety causal relationship reasoning rules**: we combine three above types of reasoning rules to create security-safety rules expressing the causal relationships from security to safety (see ch. 7, 8, 9).

Sub question 2: *Can we automatically identify security holes by reasoning on safety rules and vice-versa?*

Contribution to sub question 2: we develop the semantic reasoning tool that can identify causal relationships between security and safety events. For example, if AV does not maintain the minimum safe distance, then the AI tool suggests AV to take an emergency stop (see ch. 11)

Contribution to sub question 2: we can query the semantic reasoning tool to find safety consequences of security attacks. For example, if AV suffers a security attack, we can know what will happen in terms of safety (e.g. forced to Stop) (see ch. 10). This function enables that new facts can be inferred from traffic data.

12.2 Future works

We encountered several research limitations, including lack of data to integrate into the SSIoV ontology. Therefore, our future perspective is the formalisation of a security-safety domain through the development of an ontology to apply to data, by instantiating the dataset into the ontology (concepts, axioms, and rules) through a graph database that integrates both the ontology and actual data.

Our future work will involve the definition of safety and security rules, and the evaluation of data based on actual security-safety scenarios, investigating reverse resilience cases where safety rules can lead to security issues.

Appendix A

Appendix Title

A.1 Adaptation of IoT-Lite to IoV

Table A.1: Adaptation of IoT-Lite to IoV [3]

Class	Subclass	Properties	Annotations
Device	Tag Device		An IoT element that have sensing or actuating capabilities including redirection to information such as Tags
Device	Actuating Device		
Tag Device			Device that can redirect to a resource
Actuating Device			An IoT device that provides actuation (i.e. a device that can open and close a window) information (i.e. RFID, NFC, QR-codes, bar-codes)
Attribute			An attribute of an IoT object that can be exposed by an IoT service (i.e. a room (IoT Object) has a temperature (attribute), that can be exposed by a temperature sensor (IoT device)
Metadata			Any metadata that a sensor can provide not include in the classes qu:Units or qu:QuantityKind.
Object		AVs RSUs OBUs	An Object or IoT entity (i.e. room, car, table)

Continued on next page

TableA.1 – continued from previous page

Class	Subclass	Properties	Annotations
Coverage	Polygon		
Coverage	Rectangle		
Coverage	Circle		
Circle		radius	
Device		id	
Geo:Point		Relative Location	
Geo:Point		alt Relative	
Metadata		has Metadata	
Object		Interface Description	
Service		DSRC Service	IoT service provided by an IoT device.
Service		Interface Description	
Service		Endpoint	

A.2 Adaptation of VSSO

Class	Subclass	Annotations
Actuable Property	Actual signal	
Actuable Signal		All actuable signals that can dynamically be updated by the vehicle
Actuable Signal	Action	
Actuable Signal	Air Distribution	
Actuable Signal	Air Status	
Actuable Signal	Aux Input Status	
Actuable Signal	Backward	
Actuable Signal	Backward	
Actuable Signal	Backward	
Actuable Signal	Commande EVAP	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Actuable Signal	Cooler	
Actuable Signal	Decrease	
Actuable Signal	Deflate	
Actuable Signal	Deflate	
Actuable Signal	Dimming Level	
Actuable Signal	Down	
Actuable Signal	Down	
Actuable Signal	Down	
Actuable Signal	Down	
Actuable Signal	Extension	
Actuable Signal	Fan Speed	
Actuable Signal	Forward	
Actuable Signal	Forward	
Actuable Signal	Forward	
Actuable Signal	Gear	
Actuable Signal	Gear Change Mode	
Actuable Signal	Increase	
Actuable Signal	Inflate	
Actuable Signal	Inflate	
Actuable Signal	is Active	
Actuable Signal	is Active	
Actuable Signal	is Active	
Actuable Signal	is Active	
Actuable Signal	is Active	
Actuable Signal	is Active	
Actuable Signal	is Active	
Actuable Signal	is Backup on	
Actuable Signal	is Brake on	
Actuable Signal	is dome on	
Actuable Signal	is engaged	
Actuable Signal	is front defroster active	
Actuable Signal	is front fog on	
Actuable Signal	is glove box on	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Actuable Signal	is high beam on on	
Actuable Signal	is left indicator on	
Actuable Signal	is locked	
Actuable Signal	is locked on	
Actuable Signal	is low beam on	
Actuable Signal	is open	
Actuable Signal	is open	
Actuable Signal	is open	
Actuable Signal	is parking on	
Actuable Signal	is passenger on	
Actuable Signal	is rear defroster active	
Actuable Signal	is rear fog on	
Actuable Signal	is recirculation active	
Actuable Signal	is right indicator on	
Actuable Signal	is running on	
Actuable Signal	is trunk on	
Actuable Signal	latitude	
Actuable Signal	longitude	
Actuable Signal	PAN	
Actuable Signal	performance mode	
Actuable Signal	position	
Actuable Signal	position	
Actuable Signal	position	
Actuable Signal	selected URI	
Actuable Signal	source	
Actuable Signal	speed set	
Actuable Signal	status	
Actuable Signal	status	
Actuable Signal	status	
Actuable Signal	switch	
Actuable Signal	switch	
Actuable Signal	switch	
Actuable Signal	switch	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Actuable Signal	switch	
Actuable Signal	temperature	
Actuable Signal	throttle actuator	
Actuable Signal	tilt	
Actuable Signal	tilt	
Actuable Signal	up	
Actuable Signal	up	
Actuable Signal	up	
Actuable Signal	up	
Actuable Signal	volume	
Actuable Signal	warmer	
Actuation		An Actuation carries out an (Actuation) Procedure to change the state of the world using an Actuator
Actuator		A device that is used by, or implements, an (Actuation) Procedure that changes the state of the world
Observable Property		An observable quality (property, characteristic) of a FeatureOfInterest.
Observable Property	Observable Signal	
Observable Signal	Absolute Load	
Observable Signal	Accelerate Position	
Observable Signal	Accuracy	
Observable Signal	Action	
Observable Signal	Air Distribution	
Observable Signal	Air Status	
Observable Signal	Album	
Observable Signal	Ambient Air Temperature	
Observable Signal	Ambient Light	
Observable Signal	Angle	
Observable Signal	Artist	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Observable Signal	Aux Input Status	
Observable Signal	Average Consumption	
Observable Signal	Barometric Pressure	
Observable Signal	Battery Capacity	
Observable Signal	Battery Temperature	
Observable Signal	Brakes Worn	
Observable Signal	Clutch Wear	
Observable Signal	Commanded EGR	
Observable Signal	Commanded Equivalence Ratio	
Observable Signal	Commanded EVAP	
Observable Signal	Consumption since start	
Observable Signal	Control Module Voltage	
Observable Signal	Coolant Temperature	
Observable Signal	Current	
Observable Signal	Declined URI	
Observable Signal	Dimming Level	
Observable Signal	Distance since DTC Clear	
Observable Signal	Distance with MIL	
Observable Signal	Dive Time	
Observable Signal	DTC Count	
Observable Signal	ECT	
Observable Signal	EGR Error	
Observable Signal	Engine Load	
Observable Signal	EOP	
Observable Signal	EOT	
Observable Signal	Error	
Observable Signal	Error	
Observable Signal	Error	
Observable Signal	Error	
Observable Signal	Error	
Observable Signal	Error	
Observable Signal	Ethanol Percent	
Observable Signal	EVAP Vapor Pressure	
		Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Observable Signal	EVAP Vapor Pressure Alternate	
Observable Signal	Extension	
Observable Signal	Fan Speed	
Observable Signal	Fluid Level	
Observable Signal	Fluid Level Low	
Observable Signal	Freeze DTC	
Observable Signal	Fuel Injection Timing	
Observable Signal	Fuel Pressure	
Observable Signal	Fuel Rail Pressure Absolute	
Observable Signal	Fuel Rail Pressure Direct	
Observable Signal	Fuel Rail Pressure Vac	
Observable Signal	Fuel Rate	
Observable Signal	Fuel Status	
Observable Signal	Fuel Type	
Observable Signal	Gear	
Observable Signal	Gear Box Temperature	
Observable Signal	Gear Change Mode	
Observable Signal	Has Passenger	
Observable Signal	Heading	
Observable Signal	Heating	
Observable Signal	Height	
Observable Signal	Height	
Observable Signal	Height	
Observable Signal	Hybrid Battery Remaining	
Observable Signal	Idle Time	
Observable Signal	Ignition off time	
Observable Signal	Ignition on time	
Observable Signal	Inflation	
Observable Signal	Inflation	
Observable Signal	Instant Consumption	
Observable Signal	Intake temperature	
Observable Signal	Intensity	
Observable Signal	is Active	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Observable Signal	is Active	
Observable Signal	is Active	
Observable Signal	is Active	
Observable Signal	is Active	
Observable Signal	is Active	
Observable Signal	is Active	
Observable Signal	is Air Conditioning Active	
Observable Signal	is Backup on	
Observable Signal	is Belted	
Observable Signal	is Brake on	
Observable Signal	is Child lock active	
Observable Signal	is Deployed	
Observable Signal	is Dome on	
Observable Signal	is Engaged	
Observable Signal	is Engaged	
Observable Signal	is Engaged	
Observable Signal	is Engaged	
Observable Signal	is front defroster active	
Observable Signal	is front fog on	
Observable Signal	is glove box on	
Observable Signal	is hazard on	
Observable Signal	is high beam on on	
Observable Signal	is left indicator on	
Observable Signal	is locked	
Observable Signal	is locked on	
Observable Signal	is low beam on	
Observable Signal	is open	
Observable Signal	is open	
Observable Signal	is open	
Observable Signal	is parking on	
Observable Signal	is passenger on	
Observable Signal	is rear defroster active	
Observable Signal	is rear fog on	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Observable Signal	is recirculation active	
Observable Signal	is right indicator on	
Observable Signal	is running on	
Observable Signal	is trunk on	
Observable Signal	lateral	
Observable Signal	latitude	
Observable Signal	latitude	
Observable Signal	length	
Observable Signal	level	
Observable Signal	level	
Observable Signal	level low	
Observable Signal	light intensity	
Observable Signal	Longitude	
Observable Signal	Longitude	
Observable Signal	Longitude	
Observable Signal	Longitudinal	
Observable Signal	Long Term Fuel Trim1	
Observable Signal	Log Term O2 Trim	
Observable Signal	MAF	
Observable Signal	MAP	
Observable Signal	Massage	
Observable Signal	MaxMAF	
Observable Signal	MIL	
Observable Signal	Pad Wear	
Observable Signal	PAN	PAN services support low bandwidth and energy consumption communications
Observable Signal	Pedal Position	
Observable Signal	Pedal Position	
Observable Signal	Performance Mode	
Observable Signal	PidsA	
Observable Signal	PidsB	
Observable Signal	PidsC	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Observable Signal	Pitch	
Observable Signal	Position	
Observable Signal	Position	
Observable Signal	Position	
Observable Signal	Power	
Observable Signal	Pressure	
Observable Signal	Pressure Low	
Observable Signal	Range	
Observable Signal	Rear Left	
Observable Signal	Recline	
Observable Signal	Relative Accelerator Position	
Observable Signal	Relative Throttle Position	
Observable Signal	Roll	
Observable Signal	Run Time	
Observable Signal	Run Time MIL	
Observable Signal	Selected URI	
Observable Signal	Short Term Fuel Trim1	
Observable Signal	Short Term Fuel o2 Trim	
Observable Signal	Source	
Observable Signal	Speed	
Observable Signal	Speed	
Observable Signal	Speed Set	
Observable Signal	Status	
Observable Signal	Status	
Observable Signal	Status	
Observable Signal	Temperature	
Observable Signal	Temperature	
Observable Signal	Temperature1	
Observable Signal	Throttle Actuator	
Observable Signal	Throttle Position	
Observable Signal	Throttle Position B	
Observable Signal	Tilt	
Observable Signal	Tilt	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Observable Signal	Time since DTC Cleared	
Observable Signal	Time Since Start	
Observable Signal	Timing Advance	
Observable Signal	Torque	
Observable Signal	TPS	
Observable Signal	Track	
Observable Signal	Travelled Distance	
Signal	Travelled Distance	
Observable Signal	Trip Meter Reading	
Observable Signal	URI	
Observable Signal	Vertical B	
Observable Signal	Voltage	
Observable Signal	Voltage	
Observable Signal	Volume	
Warning	Collision	
Observation		Act of carrying out an (Observation) Procedure to estimate or calculate a value of a property of a FeatureOfInterest. Links to a Sensor to describe what made the Observation and how; links to an ObservableProperty to describe what the result is an estimate of, and to a FeatureOfInterest to detail what that property was associated with
Platform		A Platform is an entity that hosts other entities, particularly Sensors, Actuators, Samplers, and other Platforms
		Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Procedure		A workflow, protocol, plan, algorithm, or computational method specifying how to make an Observation, create a Sample, or make a change to the state of the world (via an Actuator). A Procedure is reusable, and might be involved in many Observations, Samplings, or Actuations. It explains the steps to be carried out to arrive at reproducible results
Sensors	Battery Sensor	
Sensors	Catalyst Temperature Sensor	
Sensors	Coolant Temperature Sensor	
Sensors	Crankshaft Position Sensor	
Sensors	Cushion Position Sensor	
Sensors	Fluid Sensor	
Sensors	Fuel Pressure Sensor	
Sensors	Fuel Rail Pressure Sensor	
Sensors	Intake-Air Temperature Sensor	
Sensors	Lumbar Pressure Sensor	
Sensors	MAF Sensor	
Sensors	Voltage Sensor	
Standard Sensors	Accelerometer	
Standard Sensors	Air Conditioning System	
Standard Sensors	Airbag System	
Standard Sensors	Antilock Braking System	
Standard Sensors	Backup Light Switch	
Standard Sensors	Battery Monitor	
Standard Sensors	Belt Sensor	
Standard Sensors	Brake Fluid Level Sensor	
Standard Sensors	Brake Light Switch	
Standard Sensors	Brake Padwear sensor	
Standard Sensors	Child Lock	
Standard Sensors	Clutch Wear Indicator	

Continued on next page

Table A.2: Adaptation of VSSO [4]

TableA.2 – continued from previous page

Class	Subclass	Annotations
Standard Sensors	Coolant thermometer	
Standard Sensors	Cruise Control System	
Standard Sensors	Defroster	
Standard Sensors	Dimming System	
Standard Sensors	Dome Light	
Standard Sensors	Door Contact Sensor	
Standard Sensors	Door Lock	
Standard Sensors	Steering Wheel Position Sensor	
Standard Sensors	Wiper Switch	

A.3 Adaptation of STAMP-based ontology

Class	Subclass	Properties	Annotations
Asset	Actuator		
Asset	Communication Networks and Protocols		
Communication Networks and Protocols	In vehicle Networks V2I V2P V2R V2V V2X PCT	Bluetooth	Bluetooth is also a short-range communication network that mainly supports or communications in many of today's vehicles [64]
In vehicle Networks	CAN Ethernet FlexRay LIN MOST		

Continued on next page

Table A.3: Adaptation of STAMP-based ontology [5]

TableA.3 – continued from previous page

Class	Subclass	Properties	Annotations
PCT	CALM DSRC C-V2X GSM GPRS 3G 4G/LTE LTE 5g NFC USB WAVE Wi-Fi WIMA ZigBee		
ZigBee			ZigBee is a low-cost communication technology that supports short-range information exchange between a vehicle and its internal sensors V2S [64]. A WAVE system, also known as DSRC, refers to a system designed for efficient and reliable radio communications for V2V, V2V, or V2I direct connections.
Continued on next page			

Table A.3: Adaptation of STAMP-based ontology [5]

TableA.3 – continued from previous page

Class	Subclass	Properties	Annotations
Wi-Fi			Wi-Fi technology for vehicular communication consists of roadside units, as wireless access points, to support vehicular communications inside their coverage area. Wi-Fi services provide V2I and ad-hoc V2V communication [77]. Wi-Fi technology coverage range is up to 100 m. However, it does not support vehicles moving at high speed.
WiMAX			WiMAX supports vehicle communication to the Internet at a maximum distance of 50 km. It is considered as a fast and high bandwidth connection providing V2X communication.
Asset	Human		
Human	Drivers Passengers Pedestrians		
Asset	Information		

Continued on next page

Table A.3: Adaptation of STAMP-based ontology [5]

TableA.3 – continued from previous page

Class	Subclass	Properties	Annotations
Information	Device Information Keys and Certificate Map Data Sensor Data User Information V2X Information		
Asset	Inside Vehicle Communication Components		
Inside Vehicle Communication Components	IVI EV Charging Connector In-vehicle Gateway OBD-II-Port Telematics Box Vehicle IT Station		
Asset	Sensors		
Sensor	Sensor for AV		
Sensor for AV	Cameras Lasers LIDAR Radars Ultrasonic		
Asset	Servers System and Cloud Computing		
Servers System and Cloud Computing	Service Providers Servers (3rd) Database Servers Map Servers Systems		
Asset	Vehicle function		
Vehicle function	Active Lane Keeping Air Bag Control Braking Climate Control Collision Control Door Lacking Navigation/Route Planning Steering Traffic Sign Recognition		

Continued on next page

Table A.3: Adaptation of STAMP-based ontology [5]

TableA.3 – continued from previous page

Class	Subclass	Properties	Annotations
Vehicle action	Go Go Backward Go Forward Stop To Left To Right Turn Left Turn Right Wait Then Go		
Severity			It is a qualitative indication of the magnitude of the adverse effect of a Causal Scenario
Severity	Catastrophic Hazardous Major Minor No Effect		
Causal Scenario	Safety Scenario Security Scenario		
Safety Scenario			It covers the unintentional actions that describe how incorrect feedback, design errors, component failures, and other factors can lead to a Hazardous Control Action and Unacceptable Loss
Safety Scenario	Causal Factor		
Security Scenario			It covers intentional actions, explaining how a control flaw can be introduced by an adversary
Security Scenario	Threat		

Continued on next page

Table A.3: Adaptation of STAMP-based ontology [5]

TableA.3 – continued from previous page

Class	Subclass	Properties	Annotations
Causal Factor			It consists of generic factors
Causal Factor	Component Failure Control Action Issue Control Input External Information Feedback Issue Inadequate Control Algorithm Inconsistent Process Model Incorrect Process Model Measurement Inaccuracy Process Model Issues Wrong External Information Wrong Input		
Level of threat			It is a qualitative evaluation of the possibility of the Security Scenario taking place
Level of threat	Automation Level		The entity Automation Level identifies the degree to which the attack is automated
Automation Level		extremely low low moderate high very high	
Level of threat	Attacker Location		The entity Attacker Location refers to where the attack is located. The attack can be launched from inside or outside or both of security perimeter

Continued on next page

Table A.3: Adaptation of STAMP-based ontology [5]

TableA.3 – continued from previous page

Class	Subclass	Properties	Annotations
Attacker Location		extremely low low moderate high very high	
Level of threat	Mission Phase Attack		The entity Mission Phase Attack denotes in which mission phase the attack can be launched such as operation, manufacturing, or maintenance
Mission Phase Attack		extremely low low moderate high very high	
Threat	Eavesdropping		
Eavesdropping	Protocol Hijacking Data Reply Man-in-the-middle Session Hijacking		
Threat	Failure		
Failure	Failure of Actuator Failure of Sensors Failure of Services Failure of Software Vulnerabilities Exploitation		
Threat	Nefarious Activity		
Nefarious Activity	Abuse of Authorisation Denial of Service Identity Theft Manipulation of hardware Manipulation of software OEM Target Attacks Unauthorised Activities		

Continued on next page

Table A.3: Adaptation of STAMP-based ontology [5]

TableA.3 – continued from previous page

Class	Subclass	Properties	Annotations
Denial of Service			It consists of prevent authorised access to resources or the delaying of time-critical operations. The exploited causal factors are Missing Communication, Missing Feedback, Missing Input, Missing Control Action, and Missing External Information
Threat	Outages Car Depleted Battery Loss of GNNS-Signal Network Outage		
Threat	Unintentional Damages		
Unintentional Damages	Erroneous Use or Configuration of car components Information Leakage Unintentional Change of data Unintentional Change of car components configuration Using Information Devices from unreliable source		

A.4 Adaptation of SecAOnto

Class	Subclass	Instance	Annotations
Asset	Assurance Dependability Human Security		
Attack	Active Attack Passive Attack		
Active Attack	Brute force Denial of service Disruption Spoofing Eavesdropping Malformed Input Network Infrastructure		
Passive Attack	Man in the middle Phishing Side Channel System Mapping Spyware		
Failure			It results from error propagation. A failure is noticed when the produced result is different from the expected result
Mistake			It is human action that can produce a defect. Programmers (development phase), Engineers (project phase) or Operators (deployment phase) make mistakes for various reasons (forgetfulness, lack of knowledge, etc.
Security Property	Auditability Authenticity Availability Confidentiality Integrity Legality Resilience Non-repudiation Non-Retroactivity Privacy		

Continued on next page

Table A.4: Adaptation of SecAOnto [6]

TableA.4 – continued from previous page

Class	Subclass	Instance	Annotations
Auditability			System has capability to generate and provide evidences that security requirements have been achieved
Authenticity			System allows prove the veracity of a particular act or document. This property is regarding whether information or documents are true (authentic) or false
Confidentiality			Information is accessible and usable only for authorised users or systems. Usually, profiles, levels or degrees of secrecy are defined
Integrity			Information or system have not been modified or destroyed in an unauthorised or accidental way. This property is regarding whether the information is correct or whether the system provides correct data.
Legality			System and process are in accordance with applicable law or regulation
Non-repudiation			System records corroborative evidences of important acts, so as not to let users or other systems refuse the authorship of performed actions
Non-Retroactivity			System does not allow perform actions or generate documents retroactively in time
Privacy			System does not disclose indiscriminately, or without specific permission, information about personal intimacy (personal information). This intimacy has several levels of perception

Continued on next page

Table A.4: Adaptation of SecAOnto [6]

TableA.4 – continued from previous page

Class	Subclass	Instance	Annotations
Resilience			System can continue operating even though in adverse conditions, such as operating environment problems, or failures caused by cyber attacks
Traceability			System records information about critical actions to enable reassembly of the history of actions, when it is necessary
Vulnerability			It is a weakness that can lead to a breach of security in presence of a threat

A.5 Representation of Main Concepts of SSIoV ontology through OWLViz

Figure A.1 represents some concepts and relationships of SSIoV ontology.

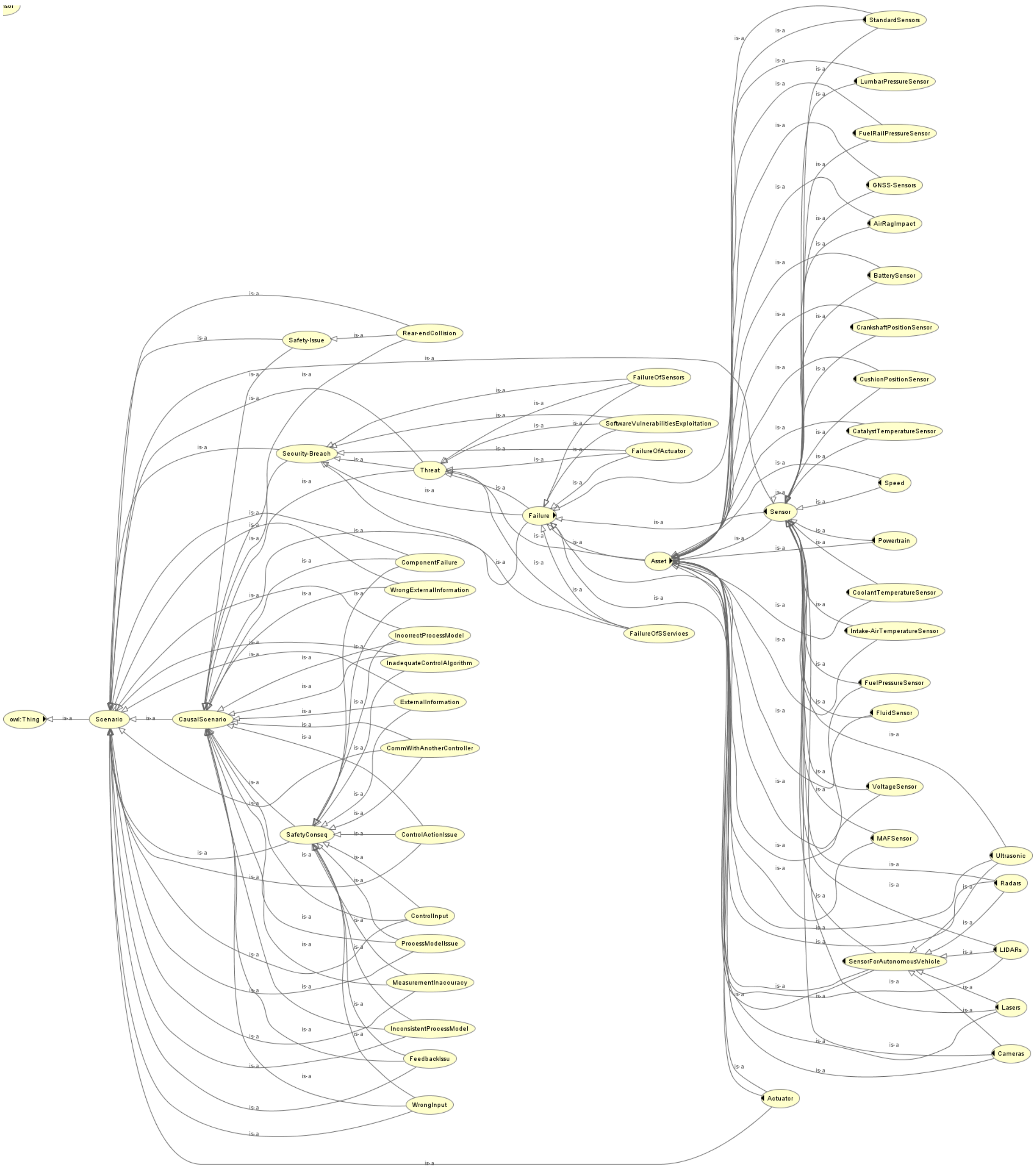


Figure A.1: A screenshot of some concepts and relations of the SSIoV ontology through OWLViz that is a Protégé plugin

Bibliography

[1] V. Ivanova, T. Kauppinen, S. Lohmann, S. Mazumdar, C. Pesquita, and K. Xu, "Workshop on visualizations and user interfaces for knowledge engineering and linked data analytics (ekaw 2014 satellite events)," 2015.

[2] R. Gasmi and M. Aliouat, "Vehicular ad hoc networks versus internet of vehicles-a comparative view," in *2019 International Conference on Networking and Advanced Systems (ICNAS)*. IEEE, 2019, pp. 1–6.

[3] M. Bermudez-Edo, T. Elsaleh, P. Barnaghi, and K. Taylor, "Iotlite ontology. w3c member submission," *World Wide Web Consortium*, 2015.

[4] B. Klotz, R. Troncy, D. Wilms, and C. Bonnet, "Vsso-a vehicle signal and attribute ontology (short paper)," in *SSN Workshop at ISWC. CEUR Workshop Proceedings*, 2018.

[5] D. P. Pereira, C. Hirata, and S. Nadjm-Tehrani, "A stamp-based ontology approach to support safety and security analyses," *Journal of Information Security and Applications*, vol. 47, pp. 302–319, 2019.

[6] F. de Franco Rosa, M. Jino, and R. Bonacin, "Towards an ontology of security assessment: A core model proposal," in *Information Technology-New Generations*. Springer, 2018, pp. 75–80.

[7] J. Shuttleworth, "Sae standards news: J3016 automated-driving graphic update," *SAE International*, 2019.

[8] ENISA. (2019) Code of practice, good practices for security of smart cars. [Online]. Available: https://www.enisa.europa.eu/publications/smart-cars

[9] M. S. Anwer and C. Guy, “A survey of vanet technologies,” *Journal of Emerging Trends in Computing and Information Sciences*, vol. 5, no. 9, pp. 661–671, 2014.

[10] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, “Vehicular ad hoc networks (vanets): status, results, and challenges,” *Telecommunication Systems*, vol. 50, no. 4, pp. 217–241, 2012.

[11] N. J. Patel and R. H. Jhaveri, “Trust based approaches for secure routing in vanet: A survey,” *Procedia Computer Science*, vol. 45, pp. 592–601, 2015.

[12] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, “A comprehensive survey on vehicular ad hoc network,” *Journal of network and computer applications*, vol. 37, pp. 380–392, 2014.

[13] M. Smita and N. Pathak, “Secured communication in real time vanet,” in *Proceedings of the International Conference on Emerging Trends in Engineering and Technology (ICETET)*, 2009, pp. 1151–1155.

[14] M. A. Hezam, A. Junaid, A. Syed, M. Nazri, M. Warip, K. N. Fazira, K. Azir, and R. Nurul Hidayah, “Classification of security attacks in vanet: A review of requirements and perspectives,” 2018.

[15] D. M. M. Azzahar, M. Y. Darus, S. J. Elias, J. Jasmis, M. Z. Zakaria, and S. R. M. Dawam, “A review: Standard requirements for internet of vehicles (iov) safety applications,” in *2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*. IEEE, 2020, pp. 1–5.

[16] R. Lu, X. Lin, X. Liang, and X. Shen, “A dynamic privacy-preserving key management scheme for location-based services in vanets,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 127–139, 2011.

[17] X. Lin and X. Li, “Achieving efficient cooperative message authentication in vehicular ad hoc networks,” *IEEE Transactions on Vehicular Technology*, vol. 62, no. 7, pp. 3339–3348, 2013.

[18] B. Ying and A. Nayak, "Anonymous and lightweight authentication for secure vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, pp. 10 626–10 636, 2017.

[19] M. Akila and T. Iswarya, "An efficient data replication method for data access applications in vehicular ad-hoc networks," in *2011 International Conference on Electronics, Communication and Computing Technologies*. IEEE, 2011, pp. 17–22.

[20] J. Li, H. Lu, and M. Guizani, "Acpn: A novel authentication framework with conditional privacy-preservation and non-repudiation for vanets," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 938–948, 2014.

[21] N. Sharma, N. Chauhan, and N. Chand, "Security challenges in internet of vehicles (iov) environment," in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*. IEEE, 2018, pp. 203–207.

[22] S. Sharma and B. Kaushik, "A survey on internet of vehicles: Applications, security issues et solutions," *Vehicular Communications*, vol. 20, p. 100182, 2019.

[23] K. C. Zeng, S. Liu, Y. Shu, D. Wang, H. Li, Y. Dou, G. Wang, and Y. Yang, "All your {GPS} are belong to us: Towards stealthy manipulation of road navigation systems," in *27th {USENIX} Security Symposium ({USENIX} Security 18)*, 2018, pp. 1527–1544.

[24] B. Mufson, "Meet the artist using ritual magic to trap self-driving cars," *Vice Creators*, 2017.

[25] N. H. T. S. A. NHTSA, "Cybersecurity best practices for modern vehicles," *Report No. DOT HS*, vol. 812, no. 333, pp. 17–20, 2016.

[26] S. I.-N. Delhi, "Automotive cyber security best practices," *Auto Tech Review*, vol. 8, no. 5, pp. 20–25, 2016.

[27] E. A. M. A. ACEA. (2017) Acea principles of automobile cybersecurity. [Online]. Available: https://www.acea.be/uploads/publications/ACEA_Principles_of_Automobile_Cybersecurity.pdf

[28] N. UNECE, "Proposal for recommendation on cyber security," 2019. [Online]. Available: https://www.unece.org/fileadmin/DAM/trans/doc/2019/wp29grva/ECE-TRANS-WP29-GRVA-2019-02e.pdf

[29] IoT. (2021) Code of practice on the interaction between the disciplines of functional safety and cyber security. [Online]. Available: https://electrical.theiet.org/guidance-codes-ofpractice/publications-by-category/cyber-security/code-of-practice-cyber-security-and-safety/

[30] D. Alvarez-Coello and J. M. Gomez, "Ontology-based integration of vehicle-related data," in *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*. IEEE, 2021, pp. 437–442.

[31] A. Haller, K. Janowicz, S. Cox, M. Lefrançois, K. Taylor, D. Le Phuoc, J. Lieberman, R. García-Castro, R. Atkinson, and C. Stadler, "Sosa: A lightweight ontology for sensors, observations, samples, and actuators," *Semantic Web Journal*, 2018.

[32] B. Klotz, S. K. Datta, D. Wilms, R. Troncy, and C. Bonnet, "A car as a semantic web thing: motivation and demonstration," in *2018 Global Internet of Things Summit (GIoTS)*. IEEE, 2018, pp. 1–6.

[33] L. Daniele, R. Garcia-Castro, M. Lefrançois, and M. Poveda-Villalon, "The smart appliances reference (saref) ontology," 2020.

[34] O. Risk, "Description of a model (doam). a risk model ontology," 2016. [Online]. Available: https://www.openriskmanual.org/ns/doam/index-en.html

[35] M. Rodriguez and J. Laguia, "An ontology for process safety," *Chemical Engineering Transactions*, vol. 77, pp. 67–72, 2019.

[36] X. Xing, B. Zhong, H. Luo, H. Li, and H. Wu, "Ontology for safety risk identification in metro construction," *Computers in Industry*, vol. 109, pp. 14–30, 2019.

[37] L. Zhao, R. Ichise, S. Mita, and Y. Sasaki, "Core ontologies for safe autonomous driving." in *International Semantic Web Conference (Posters and Demos)*, 2015.

[38] P. Torr, “Demystifying the threat modeling process,” *IEEE Security & Privacy*, vol. 3, no. 5, pp. 66–70, 2005.

[39] A. Gyrard, C. Bonnet, and K. Boudaoud, “An ontology-based approach for helping to secure the etsi machine-to-machine architecture,” in *2014 IEEE International Conference on Internet of Things (iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom)*. IEEE, 2014, pp. 109–116.

[40] B. A. Mozzaquatro, R. Jardim-Goncalves, and C. Agostinho, “Towards a reference ontology for security in the internet of things,” in *2015 IEEE International Workshop on Measurements & Networking (M&N)*. IEEE, 2015, pp. 1–6.

[41] B. A. Mozzaquatro, C. Agostinho, D. Goncalves, J. Martins, and R. Jardim-Goncalves, “An ontology-based cybersecurity framework for the internet of things,” *Sensors*, vol. 18, no. 9, p. 3053, 2018.

[42] M. Tao, J. Zuo, Z. Liu, A. Castiglione, and F. Palmieri, “Multi-layer cloud architectural model and ontology-based security service framework for iot-based smart homes,” *Future Generation Computer Systems*, vol. 78, pp. 1040–1051, 2018.

[43] S. Alam, M. M. Chowdhury, and J. Noll, “Interoperability of security-enabled internet of things,” *Wireless Personal Communications*, vol. 61, no. 3, pp. 567–586, 2011.

[44] T. Qamar and N. Z. Bawany, “A cyber security ontology for smart city,” *International Journal on Information Technologies & Security*, vol. 12, no. 3, 2020.

[45] M. A. Cappelli, G. Di Marzo Serugendo, A.-F. Cutting-Decelle, and M. Strohmeier, “A semantic-based approach to analyze the link between security and safety for Internet of Vehicle (IoV) and Autonomous Vehicles (AVs),” in *CARS 2021 6th International Workshop on Critical Automotive Applications: Robustness
Safety*, Münich, Germany, Sep. 2021. [Online]. Available: https://hal.archives-ouvertes.fr/hal-03366378

[46] T. Takahashi, Y. Kadobayashi, and H. Fujiwara, “Ontological approach toward cybersecurity in cloud computing,” in *Proceedings of the 3rd international conference on Security of information and networks*, 2010, pp. 100–109.

[47] S. Debbech, “Ontologies pour la gestion de sécurité ferroviaire: intégration de l’analyse dysfonctionnelle dans la conception,” Ph.D. dissertation, Ecole centrale de Lille, 2019.

[48] M. Viktorović, D. Yang, and B. d. Vries, “Connected traffic data ontology (ctdo) for intelligent urban traffic systems focused on connected (semi) autonomous vehicles,” *Sensors*, vol. 20, no. 10, p. 2961, 2020.

[49] D. Corsar, M. Markovic, P. Edwards, and J. D. Nelson, “The transport disruption ontology,” in *International Semantic Web Conference.* Springer, 2015, pp. 329–336.

[50] T. Elsaleh, M. Bermudez-Edo, S. Enshaeifar, S. T. Acton, R. Rezvani, and P. Barnaghi, “Iot-stream: a lightweight ontology for internet of things data streams,” in *2019 Global IoT Summit (GIoTS).* IEEE, 2019, pp. 1–6.

[51] H. Martin, Z. Ma, C. Schmittner, B. Winkler, M. Krammer, D. Schneider, T. Amorim, G. Macher, and C. Kreiner, “Combined automotive safety and security pattern engineering approach,” *Reliability Engineering & System Safety*, vol. 198, p. 106773, 2020.

[52] Y. Sun, L. Wu, S. Wu, S. Li, T. Zhang, L. Zhang, J. Xu, Y. Xiong, and X. Cui, “Attacks and countermeasures in the internet of vehicles,” *Annals of Telecommunications*, vol. 72, no. 5-6, pp. 283–295, 2017.

[53] J. Dürrwang, M. Braun, R. Kriesten, and A. Pretschner, “Enhancement of automotive penetration testing with threat analyses results,” *To appear in SAE International Journal of Transportation Cybersecurity and Privacy*, 2018.

[54] J. Cui and G. Sabaliauskaite, “On the alignment of safety and security for autonomous vehicles,” *Proc. IARIA CYBER*, pp. 1–6, 2017.

[55] N. G. Leveson, *Engineering a safer world: Systems thinking applied to safety.* The MIT Press, 2016.

[56] M. A. Musen, “The protégé project: a look back and a look forward,” *AI matters*, vol. 1, no. 4, pp. 4–12, 2015.

[57] W. Young and N. G. Leveson, “An integrated approach to safety and security based on systems theory,” *Communications of the ACM*, vol. 57, no. 2, pp. 31–35, 2014.

[58] I. Friedberg, K. McLaughlin, P. Smith, D. Laverty, and S. Sezer, “Stpa-safesec: Safety and security analysis for cyber-physical systems,” *Journal of information security and applications*, vol. 34, pp. 183–196, 2017.

[59] A. Abdulkhaleq, S. Wagner, D. Lammering, H. Boehmert, and P. Blueher, “Using stpa in compliance with iso 26262 for developing a safe architecture for fully automated vehicles,” *arXiv preprint arXiv:1703.03657*, 2017.

[60] L. Shan, C. Loiseaux, N. Marko, and J. C. Triginer, “Safety-security co-analysis with stpa: A case study on connected cars.”

[61] S. Placke, J. Thomas, and D. Suo, “Integration of multiple active safety systems using stpa,” SAE Technical Paper, Tech. Rep., 2015.

[62] R. W. van der Heijden, T. Lukaseder, and F. Kargl, “Veremi: A dataset for comparable evaluation of misbehavior detection in vanets,” in *International Conference on Security and Privacy in Communication Systems.* Springer, 2018, pp. 318–337.

[63] N. F. Noy, D. L. McGuinness *et al.*, “Ontology development 101: A guide to creating your first ontology,” 2001.

[64] A. Yazdizadeh and B. Farooq, “Smart mobility ontology: Current trends and future directions,” *arXiv preprint arXiv:2012.08622*, 2020.

[65] M. Uschold and M. Gruninger, “Ontologies: Principles, methods and applications,” *The knowledge engineering review*, vol. 11, no. 2, pp. 93–136, 1996.

[66] R. Ross, M. McEvilley, and J. Oren, “Systems security engineering: Considerations for a multidisciplinary approach in the engineering of trustworthy secure systems,” National Institute of Standards and Technology, Tech. Rep., 2016.

[67] S. J. Taylor, F. Ahmad, H. N. Nguyen, S. A. Shaikh, D. Evans, and D. Price, "Vehicular platoon communication: Cybersecurity threats and open challenges," in *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. IEEE, 2021, pp. 19–26.

[68] M. Amoozadeh, A. Raghuramu, C.-N. Chuah, D. Ghosal, H. M. Zhang, J. Rowe, and K. Levitt, "Security vulnerabilities of connected vehicle streams and their impact on cooperative driving," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 126–132, 2015.

[69] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean *et al.*, "Swrl: A semantic web rule language combining owl and ruleml," *W3C Member submission*, vol. 21, no. 79, pp. 1–31, 2004.

[70] A. Abdulkhaleq, D. Lammering, S. Wagner, J. Röder, N. Balbierer, L. Ramsauer, T. Raste, and H. Boehmert, "A systematic approach based on stpa for developing a dependable architecture for fully automated driving vehicles," *Procedia Engineering*, vol. 179, pp. 41–51, 2017.

[71] L. C. Wei and S. Madnick, "A system theoretic approach to cybersecurity risk analysis and mitigation for autonomous passenger vehicles," 2018.

[72] A. Chowdhury, G. Karmakar, J. Kamruzzaman, A. Jolfaei, and R. Das, "Attacks on self-driving cars and their countermeasures: A survey," *IEEE Access*, vol. 8, pp. 207 308–207 342, 2020.

[73] M. Carre, "Autonomic framework for safety management in the autonomous vehicle," Ph.D. dissertation, Université de Pau et des Pays de l'Adour, 2019.

[74] M. Stoltz-Sundnes, "Stpa-inspired safety analysis of driver-vehicle interaction in cooperative driving automation," 2019.

[75] M. Dutra, C. F. da Silva, P. Ghodous, and R. Gonçalves, "Using an inference engine to detect conflicts in collaborative design," in *2008 IEEE International Technology Management Conference (ICE)*. IEEE, 2008, pp. 1–8.

[76] M. J. O'Connor and A. K. Das, "Sqwrl: a query language for owl." in *OWLED*, vol. 529, no. 2009, 2009.

[77] A. Zekri and W. Jia, "Heterogeneous vehicular communications: A comprehensive study," *Ad Hoc Networks*, vol. 75, pp. 52–79, 2018.