A Secure-by-Design Framework for Automotive On-board Network Risk Analysis

Stefano Longari*, Andrea Cannizzo†, Michele Carminati* and Stefano Zanero*
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano
Email: * {stefano.longari, michele.carminati, stefano.zanero}@polimi.it, †andrea.cannizzo@mail.polimi.it

Abstract—**Vehicles have evolved from isolated and mechanical systems, into complex ecosystems of on-board networks composed of Electronic Control Units (ECUs), sensors and actuators, which govern their functionalities. These networks have been traditionally designed as trusted, closed systems, but modern needs have opened them to remote and local connections. Researchers have shown that modern vehicles are vulnerable to multiple types of attacks leveraging remote and physical access, which allow attackers to gain control and affect safety-critical systems. Therefore, the interest of manufacturers for embedding security into the design phase of new vehicles is rising.**

In this paper, we propose a semi-automated and topology-based risk analysis framework that helps in designing and assessing the security of automotive on-board networks. The tool receives the network topology as an input and evaluates its security using state-of-the-art risk metrics. Then, it provides the analyst with security-hardened network topologies, as a countermeasure against the most dangerous attacks. We evaluate our approach on known topologies and demonstrate its effectiveness.

I. Introduction

The automotive field has witnessed, over the past 30 years, rapid adoption of electronics throughout all vehicle systems. Most of these on-board systems consist of embedded controllers known as ECUs, interconnected to on-board networks. In addition, vehicles are now extensively connected to the "outside world," through both physical access – e.g., USB, On-Board Diagnostic (OBD-II) ports – or through short and long range remote connections – e.g., Bluetooth, Wi-Fi, cellular. In the near future, the paradigm of V2X (Vehicle to Everything) communication [1] will further connect the vehicle through standards such as Dedicated Short Range Communications (DSRC) [2] and 5G. The evolution of on-board networked systems, and their interconnection with the external world has created an extensive attack surface (in a pattern that has been witnessed already in other types of cyber-physical systems, such as industrial control systems and Internet of things (IoT) devices [3]). Researchers have already shown that it is possible to gain control of vehicles from remote and be able to affect consistently the safety of people inside and around the vehicle [4]–[7]. Therefore, both automotive manufacturers and ECU vendors have started to worry about vehicle security.

In this paper we present a semi-automated topology-based risk analysis framework that receives as input the vehicle on-board network topology and evaluates its security by considering state-of-the-art risk metrics. Then, it provides the analyst with security-hardened network topology, given a set of constraints. Our solution is routed around the idea of constructing secure-by-design on-board subnetworks, by exploiting a clear understanding of on-board network topologies and a comprehensive threat modeling. To do so, we improve and adapt the threat model of the vehicle's network presented in [8]. We evaluated our approach on state-of-the-art topologies and demonstrated the effectiveness of the proposed solution. Our contributions are the following:
1. A methodology to help security analysts while designing and assessing the security of automotive on-board networks.
2. A semi-automated and topology-based tool that automatically retrieves the risk scores and subsequently proposes a security-hardened solution, as a countermeasure against the threat model under analysis.
3. An improved and generalized risk and threat model methodology based on the one proposed by Ruddle et al. [8].

II. Related Works

Since 2011 a number of security research papers have been published that demonstrate the defenselessness of the current automotive electronics environment. Checkoway et al. [5] abused of a buffer overflow in the CD reading software of a 2009 Sedan to send predetermined CAN messages on the internal network. Miller and Valasek [7] found vulnerabilities in the WPA2 key generation algorithm and lack of authentication in open services listening through the cellular connection of a Jeep Cherokee, which led to being able to reflash the infotainment system and access the internal network. A similar approach was implemented by Nie et al. [9] on a Tesla S, where a vulnerable WebKit implementation in the infotainment system browser was used to update the firmware of the vehicle. KeenLab discovered a weakness in the communication amongst the backend and infotainment system that led to obtaining code execution privileges on the unit [10]. Foster et al. [11] analyzed the vulnerabilities of aftermarket ECUs to understand how they can affect the internal networks through the OBD-II port. Each ECU can be considered an attack surface and represented as a gateway connected to a network the attacker can already access and to one or more subnets that the attacker wants to reach. Therefore, focusing on security is

This project has been partially supported by BVTech SpA under project grant UCSA, and by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement nr. 690972.

fundamental while designing both ECUs and network layouts. Considering the latter, which is the focus of this paper, the main known analyses have been done in [6] and in [8]. The latter proposes guidelines for secure design of on board networks but has not been written to assist the production of practical analyses of an existing architecture. Miller and Valasek have been known for their proof of concept attacks first through the OBD-II port of a vehicle [6] and then through external connections such as Wi-Fi and cellular networks [7]. In addition, they wrote an analysis of the on board network of multiple vehicles [12], which takes into consideration the risks involved with the layouts of the networks and the ease, for an attacker, to reach safety critical ECUs. They use the information retrieved by the analysis of the vehicle to propose a ranking that considers the architecture of the network, the amount of external attack surfaces, and the dangerousness and number of vehicle cyberphysical controls. However, they do not describe a way to systematize the process of risk analysis. This has been partially solved by Ruddle et al. [8]: they came out with a methodology to rank automotive on board networks on the base of attack trees. However, their methodology does not consider the network layout.

III. Vehicular Environment and Threat Modeling

In this section we propose a reference model of the vehicular environment from a security perspective, focusing on a comprehensive threat-model and on the importance of the on-board network topology.

A. Vehicular Environment Overview

The automotive environment can be divided into two major sections, on-board networks and external world. The core difference amongst the two is the set of communication technologies and protocols. If from one side, on-board networks are composed by devices wired together to respect real-time requirements, on the other, in the external world uses wireless technologies. The communication message is then received by one of the on-board units of the vehicle that routes it through the internal networks and gateways until it reaches the intended recipient. For simplicity, from now on, we refer to each internal devices as Electronic Control Units (ECUs), which is any automotive embedded system that controls one or more of the electrical systems or subsystems in a vehicle. The ECUs can be broadly divided according to the fact that they do, or do not, perform safety-critical features. With *safety critical* we define all the elements and systems that influence the behavior and handling of the vehicle, endangering human life inside and around it. Vehicles comprise also all sorts of non-safety-critical ECUs, such as infotainment systems, or actuators and sensors related to passenger comfort.

On-board networks of vehicles consist of multiple protocols and technologies and connect internal devices. The most common protocol is Controller Area Network (CAN) [13], [14], and its back-compatible evolution with higher bandwidth CAN-FD [15]. The reasons for CAN success are its affordability in terms of costs, as well as its real-time communication properties. In the light of these features, network segments with safety critical ECUs are almost always CAN based. Less common protocols, which are usually implemented in specific subnetworks, are: FlexRay [16], which is a more powerful but expensive substitute for CAN and is used on specific high-end vehicles, and Local Interconnect Network (LIN) [17], which is a cheaper alternative to CAN and is commonly used in sensor-to-internal-device communication lines. More recently, real-time Ethernet has been proposed as a standard to unify many alternative protocols. It is useful to point out that, CAN, FlexRay, and LIN have all been designed without strong attention to security requirements [18]. A number of works have explored different types of weaknesses of CAN, and we refer the interested reader to them [4]–[6], [19]–[21].

B. Threat and Attacker Modeling

Threat modeling is of primary importance to assess the security of cyber-physical systems [3], [22], especially in the automotive field where the safety of people is involved.

Attacker Goals. After an in-depth analysis of the state-of-the-art, and taking inspiration from [8], we divide the goals of the attacker into four non-mutually-exclusive categories:
Safety: The attacker threatens only the safety related operations of the vehicle, with the objective of harming the driver, the passengers, or people outside the vehicle itself.
Financial: The attacker is financially-motivated and seeks an economic advantage through it actions (e.g., by stealing a vehicle, by enacting a ransomware scheme).
Operational: The attacker targets all the operations of the vehicle (e.g., an attacker might want to annoy a vehicle owner by blocking ignition), including safety related ones.
Privacy: The attacker is interested in obtaining personally-identifiable information (PII) that may be stored within the vehicular environment at large (e.g., in a V2X infrastructure).
From the Original Equipment Manufacturer (OEM) point of view, white-hat hackers may be considered as a class of attacker. In fact, they may affect the automotive company brand reputation. However, we decided to not consider them since not relevant to the focus of this paper.

Attacker Capabilities. We can assume that the attacker has full knowledge of the vehicular environment they are attacking, at least for the internal segment. In fact, it is always possible for the attacker to buy a vehicle equivalent to the target one (of course, if he or she is financially motivated, costs should be considered while evaluating the actual attack risk). However, this task requires a significant effort for the attacker, since he or she has to reverse engineer the on-board network to gain sufficient understanding of the endpoints. The real discriminant in attacker's capabilities is if the attacker has *physical access* to the vehicle interior or *remote access* through short and long range communication channels (e.g., Bluetooth, LTE). In the first case, the attacker has unrestricted access to the internal interfaces since he or she is physically inside the vehicle. However, we assume that if an attacker has unfettered access to a vehicle, he or she can carry out many different

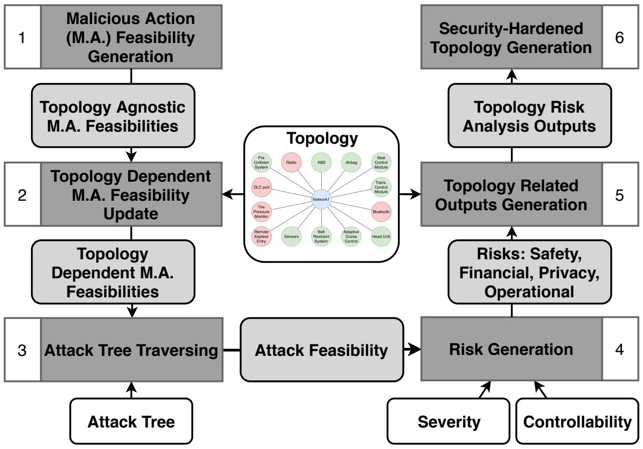

1
Malicious Action (M.A.) Feasibility Generation
Topology Agnostic M.A. Feasibilities
2
Topology Dependent M.A. Feasibility Update
Topology Dependent M.A. Feasibilities
3
Attack Tree Traversing
Attack Tree
Topology
Pre Collision System
Radio
ABS
Airbag
Seat Control Module
DLC port
Trans. Control Module
Network1
Tire Pressure Monitor
Bluetooth
Remote Keyless Entry
Sensors
Belt Restraint System
Adaptive Cruise Control
Head Unit
Attack Feasibility
Security-Hardened Topology Generation
6
Topology Risk Analysis Outputs
Topology Related Outputs Generation
5
Risks: Safety, Financial, Privacy, Operational
Risk Generation
4
Severity
Controllability

Fig. 1: Overview of the risk analysis methodology.

actions to make it unsafe and harm others. For example, if the objective of the attacker is stealing a vehicle or objects inside it, the action to gain entry to the vehicle is the main difficulty for the attacker, making attacks that require physical access not very relevant to such objectives. To implement economically viable large-scale attacks (e.g., ransomware), the attacker has to perform them remotely. In the other case (i.e., physical access), these attacks would not be feasible since they require to the attacker to compromise in a short period of time a large number of vehicles. Car sharing services (and in the future, fleets of shared self-driving vehicles) blur the line of this requirement, as they make it feasible for anyone to have unrestricted access to the vehicle interior for a given amount of time. In this situation, accessing multiple vehicles in a relatively short amount of time to perform an attack would become feasible. For this reason, we consider, although with their limitations, attack surfaces that require physical access as a viable path for the attacker.

Attack Modeling. To model the attacks and evaluate their dangerousness we apply the three definitions proposed by Ruddle et al. in [8], which are Severity, Feasibility, and Controllability. As explained later, we consider them as the three core metrics for the risk analysis methodology.
Severity (S) represents the potential damage of an attack. It is further broken down in four values depending on the attacker goal. Each of them ranges from a minimum of $S0$ ("no threat") to a maximum value of $S4$ ("very significant threat").
Feasibility (A) or "attack probability" [8] represents the attack easiness, considering all the steps required to implement it, on a scale from $A0$ (impossible) to $A5$ (very easy).
Controllability (C) represents the potential for the driver to confine the severity of the outcome. It has to be considered only when the severity vector includes a non-zero safety component. Four different levels of controllability are considered, from $C1$ (avoidance possible through human response) to $C4$ (situation impossible to influence).

Known real world attacks. The most known attacks to vehicular environments are presented in [4]–[7], [9], [10], [23]. The outcomes of these attacks space from simply turning on and off some lights or changing the speedometer value, to the retrieval by the attackers of sensitive data, to threatening the life of people. The majority of these attacks have similar endpoints, which are the CAN buses related to the target ECU. Through these buses and the use of either proprietary or diagnostic protocols such as Unified Diagnostic System (UDS) the attacker may spoof messages coming from another ECU like the Antilock Braking System (ABS) or lane assist and similar. To access these networks, depending on the vehicle internal network topology, the attacker has to make some sort of lateral movement from the ECU through which he accessed the car to the wanted CAN bus.

IV. Methodology

Our solution is routed around the idea of analyzing the risk associated with on-board vehicle networks and of constructing secure-by-design topologies, by exploiting a clear understanding of the automotive threat model, presented in §III. Our approach to the risk analysis process is composed of multiple interdependent steps, as shown in Fig. 1. First, on the basis of the threat model, we associate each attack step with its feasibility (1). Then, we update their values depending on the vehicular topology (2) and, using attack trees, we compute the feasibility (3) and the risks (4) associated with each attack. After that, exploiting the previous results and the topology, we generate an in-depth report that highlights the security-related critical points (5). Finally, on the basis of a set of empirically-generated constraints, we produce a security-hardened topology (6) that can be visualized to help the analyst in developing its design. For steps (1),(3) and (4), we extend the work of Ruddle et al. [8], [24] by automatizing the attack tree traversing and integrating it in the topology-based risk analysis process. We refer the interested reader to their work [8] for the complete description of their methodology. It is important to highlight that the core novelty of our approach lies in the topology-based risk assessment. All previous research, to the best of our knowledge, always focused on analyzing the risk either ignoring the actual architecture of the vehicle or at best assuming it to be known a priori and fixed, a sort of "black box" that influences the output but is not a parameter under assessment. To provide a systematic method to support risk analysis, we choose to model attack and threat scenarios through *Attack Trees* [25]. *Attack Trees* provide a methodical way of describing potential attacks through hierarchical diagrams that show how low-level actions interact to achieve high-level objectives. It is a simple way to describe a complex process, such as a cyberattack, dividing it into small building blocks that can be modularly assembled. If new information arises (or new methods are identified) the model can be updated easily and updates can be propagated throughout the tree chain and throughout the other steps of our methodology. We design structured trees with a precise level separation organized in this order: $Goal \longrightarrow Attack \longrightarrow Method \longrightarrow Step \longrightarrow Action$. The root of the tree is always an abstract **Goal** (level 0) which the attacker wants to obtain. The goal does not consider in any way the means through which it has to be obtained, which is represented by the **Attack** (level 1). Each attack can be implemented through different **Methods** (level 2) which represent all the ways in which a specific attack can be performed. A method, to be implemented, requires a set of **Steps** (level 3) which are all the elements that have to be used or done to accomplish that specific method. Finally, a step can be obtained through multiple **Malicious Actions** (level 4) which are the basic elements of the whole tree. The values arriving from the leaves are propagated with logic functions that act like AND (the lowest value amongst all the child nodes is propagated) or OR (the highest is propagated) up to the root node. Specifically, attack to goal and step to method propagations are done through ANDs while the others through ORs. The way Ruddle et al. [8] propose attack trees is different: there is no structured and clear division between AND and OR levels and the root-to-leaf distances are not always equal. If at first glance this could seem an advantage in terms of flexibility, it leads instead to situations in which more and less abstract elements of the tree are on the same level, hence leading to confuse and complex representations.

A. Malicious Action Feasibilities Generation

In order to map risks onto the topology, the only attack metric influenced by the topology is the attack feasibility, since controllability and severity are related only to the outcome of the attack. Therefore, the first step of our approach consists in obtaining the malicious action feasibilities. Ruddle et al. in [8] explain how to obtain them by considering the requirements that the attacker has to fulfill to implement the malicious action. These requirements comprehend elapsed time, expertise, knowledge of the system, window of opportunity and equipment of the attacker. Ruddle et al. [8] calculate a final numeric value mapped on a feasibility scale that ranges from 1 (“beyond high requirements”) to 5 (“basic requirements”). (For further details we refer the interested reader to [8]).

B. Topology Dependent Malicious Action Feasibility Update

The second step consists in updating the malicious action feasibilities by considering the topology of the assessed vehicle received in input. First, we provide a structure to model the architecture of the vehicle. Then, given the modeled architecture, we delete the malicious actions that cannot be applied and update the feasibility of the remaining ones.

Topology Modeling and Visualization. ECUs are connected to buses. Some of them are connected to more than one bus and act as gateways. Therefore, it is natural to represent such topologies using graph models and, in particular, star graphs. The attacker requires to have some sort of access to the on-board network, through a set of so-called attack surfaces, which in this case are again ECUs. We, therefore, divide ECUs in two categories, either they are normal ECUs or they are attack surfaces. Examples of topologies and their representations are shown in Fig. 2a,2b,2c. The first consequence of considering the topology while analyzing an attack is that all malicious actions that depend on a component that is not present in the vehicle are unfeasible. To handle this case all unfeasible malicious actions have their feasibility set to 0.

TABLE I: Safety risk level as a function of the attack feasibility A, the severity S and the controllability C.

Control-lability	**Severity**	**Attack Feasibility**				
		A=1	A=2	A=3	A=4	A=5
C=1	S=1	R0	R0	R1	R2	R3
	S=2	R0	R1	R2	R3	R4
	S=3	R1	R2	R3	R4	R5
	S=4	R2	R3	R4	R5	R6
C=2	S=1	R0	R1	R2	R3	R4
	S=2	R1	R2	R3	R4	R5
	S=3	R2	R3	R4	R5	R6
	S=4	R3	R4	R5	R6	R7
C=3	S=1	R1	R2	R3	R4	R5
	S=2	R2	R3	R4	R5	R6
	S=3	R3	R4	R5	R6	R7
	S=4	R4	R5	R6	R7	R8
C=4	S=1	R2	R3	R4	R5	R6
	S=2	R3	R4	R5	R6	R7
	S=3	R4	R5	R6	R7	R8
	S=4	R5	R6	R7	R8	R8

Attack Surfaces Analysis. In order to update the feasibility of the malicious actions, we need to evaluate the “dangerousness” of each attack surface, expressed in terms of a *danger parameter*. Not all attack surfaces are equally dangerous, therefore we evaluate them along three dimensions: $Cost$, $Surface$, and $Range$. The $Cost$ is a value from 1 to 3 describing the cost and the effort necessary to break into the component and take its control (1 means “high cost,” 3 means “low cost”). $Surface$ is a value from 1 to 3 related to the amount of possible new attack steps that can be done if the attacker obtains control of the component (1 means “few”, 3 means “many”). $Range$ is a value from 1 to 3 describing the necessary physical distance from the vehicle to access the surface (1 means “in car,” 3 means “remotely exploitable”). Once these values are set, they are summed and scaled to obtain a value between 1 and 3. This is defined as *danger parameter* of the attack surface.

Malicious Actions Feasibility Update. Then, we proceed in the update of the malicious action feasibilities by considering (a) the distance between each component on which the malicious action is performed and the vehicle attack surfaces and (b) their “danger parameter” values. For each malicious action a, the updated feasibility value F'_a is computed using the following function: $F'_a = F_a + (\sum_{i=1}^{n} \frac{s_i}{d_i}/w) - \delta$, where F_a is the topology agnostic feasibility of the malicious action a, n is the total number of attack surfaces, s_i is the danger parameter of surface i, d_i is the distance of the component where the malicious action takes place from surface i, δ is a normalization factor. δ is computed as the average $\sum_{i=1}^{n} \frac{s_i}{d_i}/w$ computed on all the malicious actions for all the topologies under analysis and w is a parameter used to weight the influence of the topology on the final malicious action feasibility. By doing so, each feasibility is adjusted by a value computed by lowering the strength of each danger parameter s_i depending on the distance d_i.

TABLE II: Financial, operational, and privacy risk levels as a function of the attack feasibility A amd the severity S.

Control-lability	Severity	Attack Feasibility				
		A=1	A=2	A=3	A=4	A=5
Non-Safety Severity	S=1	R0	R0	R1	R2	R3
	S=2	R0	R1	R2	R3	R4
	S=3	R1	R2	R3	R4	R5
	S=4	R2	R3	R4	R5	R6

C. Attack Tree Traversing and Risk Generation

Starting from the updated malicious action feasibilities, the goal of the attack tree traversing step is to compute the feasibility of the whole attack (A). To do so, we insert the updated malicious action feasibilities, which are dependent on the topology, as the leaves of the attack tree and propagate them up to the root, following the procedure described at the beginning of this Section. Then, we compute four risks for each attack, one for each attacker goal (i.e., safety, privacy, operational, and financial), following the methodology described in [8]. Regarding the safety-related risk, we compute its value using a function, expressed in Tab. I, which combines Controllability (C), Feasibility (A), and safety-related Severity (S) into a qualitative risk value. Instead, the financial-, operational-, and privacy-related risks, which do not depend on Controllability, are computed from a similar but simpler Tab. II, which combines Feasibility (A) and, respectively, the financial-, operational-, or privacy-related Severity (S). Each risk ranges from a minimum of $R0$ ("no risk") to a maximum of $R8$ ("unacceptable/extreme risk"). Differently from [8], we consider continuous values, instead of categorical ones. The values of the Severity (S) and Controllability (C), are obtained through the analysis of two tables proposed by Ruddle et al. (for space reasons we refer the reader to [8]) and do not require the same analysis of the attack trees required for the Feasibility(A) since they are independent of the implementation of the attack. The Severity values range from zero to four (i.e., Safety level zero is defined as "no injuries" while level four is "Life-threatening or fatal injuries for multiple vehicles"). The Controllability value ranges between 0 and 5 representing, when there is a safety component, the potential for the driver to confine the severity of the outcome.

D. Topology Related Output Generation

On the basis of the risks computed before for each attack and using the topology as input, we generate a set of outputs to help the analyst in the risk assessment of the topology:

1. A global **risk value** for each topology, that is obtained by adding together all risk values (safety, privacy, financial, operational) of each attack.

2. A ranking of the **most dangerous attacks**, by safety, privacy, financial or operational risk value.

3. A ranking of the **most targeted components**, obtained by summing, for each component, the risks of the attacks that insists on that component.

4. A ranking of the **most crossed elements**, obtained by summing, for each component and subnetwork, the risks of the attacks that traverse, in their path, that element/subnetwork.

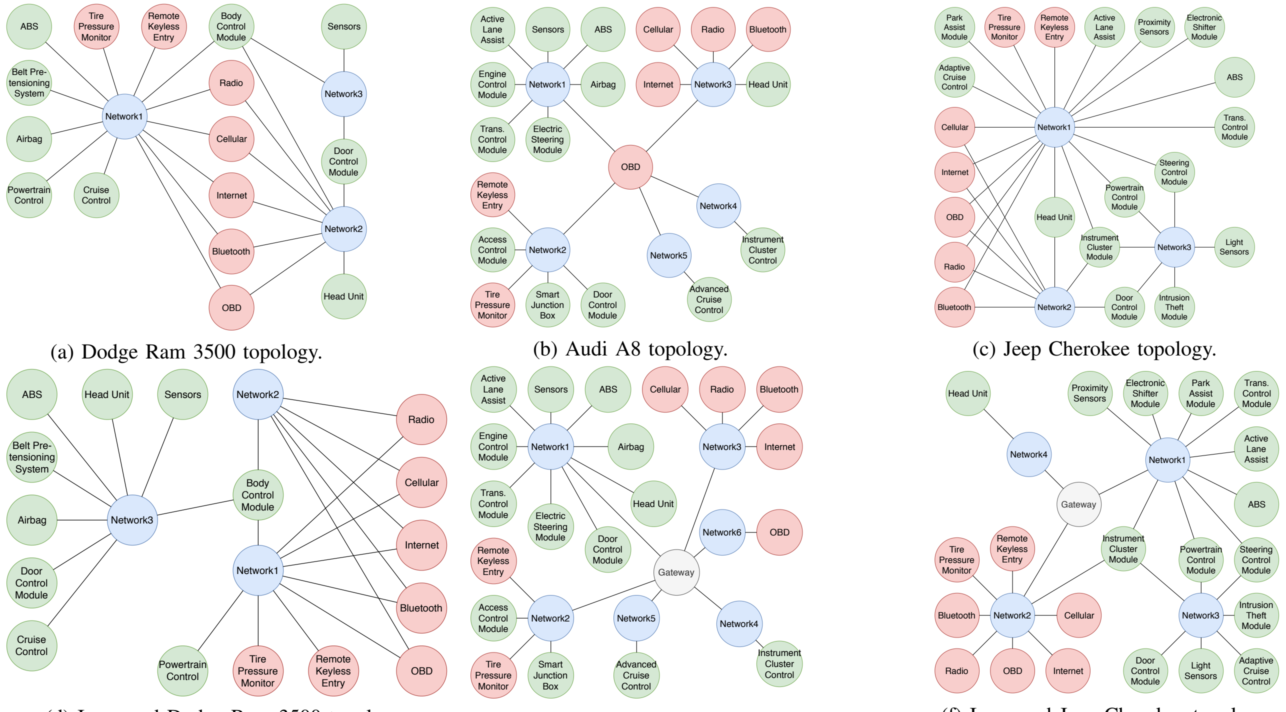
E. Security-Hardened Topology Generation

To aid in the process of developing secure-by-design topologies, we provide the analyst with a methodology that, starting from an existing topology, generates a security-hardened one, without removing existing ECUs (since we do not want to change the capabilities of the vehicle). This algorithm, given the risk analysis presented above, executes the following steps: First, it takes in input the topology and evaluates its "global risk value." Then, it iteratively changes the network topology by changing the network partitioning, inserting gateways before attack surfaces, and moving ECUs between different subnetworks, until it finds a solution that minimizes the "global risk value," given a set of empirically defined constraints. Through this process it is possible to improve the proposed topology up until the analyst requires it. However, as the knowledgeable reader might have noticed, our approach has a limitation related to the definition of constraints. In fact, the analyst may have to manually add constraints to generate topologies that are feasible in real world scenarios to meet real-time requirements (e.g., limit the distance between ECUs) or design cost ones (e.g., limit the number of gateway inserted).

To tackle this issue we add a set of constraints and allow the analyst to customize them and to insert new ones to adapt the topology under assessment. The constraints we designed, which to the best of our knowledge cover basic design constraints, are three: the first two, defined as **DisMin(A,B)** and **DisMax(A,B)**, where A,B are two ECU or gateways, require that A and B must be, respectively, at more or fewer steps away one from the other than the given value. The third, defined as **MaxGW**, restricts the number of generated gateways to be lower than the given value. To enforce these rules, the algorithm tests the generated topologies against them and skips the analysis of the topology if a rule has been violated.

V. Experimental Results

Risk Analysis Evaluation. Regarding the Safety (S) metric, the results that our tool provides are consistent with the ones presented in [12]. This is expected since Miller and Valasek [12] analyze the security of the vehicle by taking into consideration only cyberphysical components (e.g., active lane assist, brake-by-wire), which have a direct impact on the safety of people inside and outside the car. In fact, the 2014 Honda Accord and Dodge Viper, considered the least hackable ones by [12], obtain low risks, while the 2014 Jeep Cherokee, the Infiniti Q50, and the 2015 Cadillac Escalade, considered the most hackable by [12], are amongst the ones with the highest safety risk. The only outlier in our results is represented by the Audi A8, which is considered amongst the least hackable by [12] but obtains a high risk with our security assessment. The reason behind this can be inferred from its on-board network topology represented in Fig. 2b. The OBD-II port is the only element that divides the majority of the attack surfaces from all the potential attack targets. Since the authors



(a) Dodge Ram 3500 topology.

(b) Audi A8 topology.

(c) Jeep Cherokee topology.

(d) Improved Dodge Ram 3500 topology.

(e) Improved Audi A8 topology.

(f) Improved Jeep Cherokee topology.

Fig. 2: Graphical visualization of the initial on-board network topologies (top) and of the security-hardened ones generated with our methodology (bottom). In *red* the *attack surfaces*, in *green* the other *ECUs*, and in *blue* the *network elements*.

of [12] consider only remote attacks, they classify this port as not reachable from the outside and, hence, not as an attack surface. In other words, it is simply a network partitioner. In our case, instead, the OBD-II port is considered as an attack surface and due to its proximity to all ECUs, it highly increases the risk scores. In fact, if we apply the countermeasures generator to the topology under analysis, whose result is shown in Fig. 2e, we can notice that the main improvement is obtained by confining the OBD-IIs port behind a gateway, making it hardly reachable and hence closer to Miller and Valasek [12] interpretation. Regarding the Privacy metric (P), it has an overall lower risk value with respect to the others since the majority of vehicles marketed until 2014 collect a few personal data. Therefore, the severity of the attacks insisting on them is low. However, we believe that this metric must be considered since it will gain importance over the years due to the adoption of V2X technologies, which will ease the collection and sharing of personal data. The highest P values have been obtained by the newer and more advanced vehicles (e.g., the 2014 BMW X3 and the 2015 Escalade), while cars from 2006, such as the RangeRover Sport, have a P risk value of 0.0. Regarding the Operational metric (O), it follows the same trend of S. This behavior can be explained by analyzing the operational severity of the attacks. In fact, all safety related attacks have an operational implication. However, some operational related attacks do not have a safety implication (e.g., activating the wipers, locking the doors from external access), which explains why all *0* risk values are always higher than the same topology's S. Finally, the Financial metric (F) is strongly related to all the others metrics. Each attack that has a positive financial severity has at least one of the other severity metric greater than zero. This because the majority of attacks, whether privacy, safety, or operational related, can be implemented with a financial objective. Therefore, vehicles with high risk rankings in other metrics, such as the 2014 Jeep Cherokee, have a high F, while vehicles with a really low overall risks, such as the 2006 Ford Fusion and Toyota Prius, have the lowest ones. Since it is not trivial to find complete vehicle topologies annotated with a security assessment, we tested our approach on the manually analyzed topologies taken from [12]. The results of our risk analysis evaluation, along with the results presented in [12], are reported in Tab. III, where Tot is the overall risk value of the topology and S,F,P,O stand respectively for Safety, Financial, Privacy, and Operational risks. The authors of [12] evaluate the topologies through three parameters: Attack Surfaces (A.S) that measure the facility for the attacker to get inside the vehicle network from remote, Network Architectures (N.A) that measure the facility for the attacker to reach different ECUs once inside the vehicle network, and Cyberphysical (C.P) that measures the number of known cyberphysical attacks that the attacker can implement. Each of these parameters is evaluated on a scale from "Least hackable" ($--$) to "Most hackable" ($++$).

Security-hardened Topology Generation Evaluation. In Tab. III we present the risk values of the topologies generated by our methodology alongside the initial ones, followed by

TABLE III: Results of the risk analysis on both the initial topologies and the security-hardened ones, alongside with the evaluation conducted in [12]. Notation: *S*, *F*, *P*, *O* stand for *Safety*, *Financial*, *Privacy*, and *Operational* risks, while *A.S*, *N.A*, and *C.P* stand respectively for *Attack Surfaces*, *Network Architectures*, and *Cyberphysical*. Finally, the results of [12] are evaluated on a scale from "Least hackable" $(--)$ to "Most hackable" $(++)$. In **bold** the worst result for each risk category.

Topology	**Miller et al [12]**			**Risk Analysis Results**					**New Topologies Results**					**Gain**
	A.S	**N.A**	**C.P**	**Tot**	**S**	**F**	**P**	**O**	**Tot**	**S**	**F**	**P**	**O**	**%**
2014 Jeep Cherokee	$++$	$++$	$++$	**147**	**38.1**	**49.1**	11.0	**49.1**	81.5	21.0	24.7	7.0	28.7	44.6
2014 Audi A8	$++$	$--$	$+$	110	25.1	35.5	10.2	39.3	**92.5**	**23.0**	**27.7**	7.0	**34.7**	16.0
2015 Cadillac Escalade	$++$	$+$	$+$	106	24.2	34.1	**11.8**	36.3	74.7	16.5	22.1	9.66	26.4	29.8
2014 Chrysler 300	$++$	$-$	$++$	95.3	25.0	29.6	8.86	31.8	70.2	20.3	19.4	6.19	24.2	26.3
2014 Ford Fusion	$++$	$-$	$++$	86.2	19.2	28.4	11.8	26.6	77.9	19.2	24.3	9.8	24.5	9.60
2014 RangeRover Evoque	$++$	$-$	$++$	83.5	17.3	27.1	9.06	30.0	70.9	15.7	21.6	7.39	26.2	15.1
2014 Infiniti Q50	$++$	$+$	$+$	83.5	22.8	25.8	4.16	30.6	73.4	21.0	21.7	4.16	26.5	12.0
2014 BMW i12	$++$	$--$	$+$	81.1	17.1	25.4	9.2	29.2	64.7	13.8	18.9	7.46	24.4	20.1
2014 Dodge Ram 3500	$++$	$++$	$--$	79.1	19.3	26.8	10.2	22.6	45.3	11.6	13.8	7.0	12.8	42.6
2014 BMW X3	$++$	$--$	$++$	73.7	22.3	19.1	10.2	21.9	69.4	22.3	17.0	8.06	21.9	5.78
2014 BMW 3 Series	$++$	$--$	$+$	72.6	16.2	21.6	9.86	24.8	64.4	13.7	18.8	**9.86**	21.9	11.2
2010/2014 Toyota Prius	$+$	$+$	$++$	69.5	21.2	18.6	5.66	24.0	51.0	16.0	12.0	5.0	18.0	26.7
2010 Infiniti G37	$-$	$++$	$+$	67.3	20.0	17.6	6.33	23.3	44.0	14.0	9.0	5.0	16.0	34.6
2010 RangeRover Sport	$-$	$--$	$-$	48.8	10.2	15.3	6.46	16.8	35.6	8.37	9.62	5.0	12.6	27.0
2014 Dodge Viper	$++$	$-$	$--$	46.4	10.0	15.7	8.86	11.8	34.2	9.25	10.0	6.19	8.8	26.2
2014 Honda Accord LX	$-$	$+$	$+$	43.4	11.0	12.2	5.66	14.5	35.0	9.0	9.0	5.0	12.0	19.3
2006 RangeRover Sport	$-$	$--$	$-$	41.6	10.0	13.3	0.0	18.3	36.0	9.0	11.0	0.0	16.0	13.5
2006 Toyota Prius	$-$	$--$	$--$	13.2	4.0	2.63	2.63	4.0	13.0	4.0	2.5	2.5	4.0	2.01
2006 Ford Fusion	$--$	$--$	$--$	7.0	4.0	1.0	0.0	2.0	7.0	4.0	1.0	0.0	2.0	0.0

the obtained gain, computed as $Gain = 100 \cdot \frac{Tot_{old} - Tot_{new}}{Tot_{old}}$. All topologies, apart the ones with the already lowest risks, have been improved by our tool by an average of 20%, with a maximum decrease of 65 points in terms of overall risk. To do so, we implement two constraints among the ones explained in §IV: $DisMax(AtkSur, ANY) = 5$, which requires all ECUs to be at a maximum of five steps away from all attack surfaces (to avoid unfeasible, extremely spread out networks) and the $MaxGW = totECUs/5$, which means that a maximum of one gateway per 5 ECUs can be inserted (to avoid the subdivision of each network in as many as its ECUs).

Case Study: the 2014 Dodge Ram 3500. To better explain the previous results, we propose a detailed analysis of the 2014 Dodge Ram 3500 topology (see Fig. 2a). The vehicle's onboard network is composed of three different buses, the first related to safety and operational devices, the second mainly composed of infotainment and communication devices, and the last one mainly used for sensor data. The safety and operational network is, however, directly exposed. In this topology we consider safety related ECUs the airbag unit, the ABS, the cruise control, and the belt restraint system. The privacy risk *P* is derived from attacks that target the Head Unit, while the operational risk *O* comes from attacks targeting both the safety related ECUs and the door control module. As explained above, the Financial risk value *F* is related to the majority of attacks. In Tab. III we present the comparison between the risk values of the original topology and the security-hardened one: the new topology has significantly lower risk values. In fact, as it can be seen in Fig. 2a and Fig. 2d, which show a graphical comparison of the two topologies, in the security-hardened solution all the threatened ECUs are moved from directly exposed networks (i.e., attack surfaces) to less exposed ones, thereby lowering all risks of similar percentages. Tab. IV shows the risk analysis described in §IV. The most dangerous attacks are "denyBrake" and "explodeAirBag", which are caused by, respectively, the ABS and the Airbag units. It is interesting to observe that the risk metrics of these two attacks not only starts from the same values, but decrease of the same amount in the improved topology. This is due to two facts that the two units, which enable the attacks and were allocated in the same network in the initial topology, had the same severities and are moved together in the same network also in the improved topology. Therefore, they have also the same feasibilities. Descending the ranking, it is possible to see that, due to the reasons described in §V, privacy related attacks obtain low risks scores. Regarding target components, the "Head Unit" is the one with the highest risks. In fact, in our modeling of the attacks, the "Head Unit" is the ECU where all privacy related data are stored, and therefore the target of a great number OF different privacy related attacks. It is followed by the three attack surfaces – "Bluetooth", "Cellular", and "Internet Module" – whose risk values are high since they are the attack surfaces with higher danger values (they are remotely exploitable) and, therefore, they cause an increase in the topology driven feasibility of multiple attacks. Lastly, there are the "ABS module", the "Airbag module", the "Cruise Control" that are target of safety related attacks. Finally, the most traversed components of the initial topology (see Fig. 2a) are, predictably, the two networks ("Network 1" and "Network 2") on which the majority of attack surfaces and target ECUs are. The remaining traversed elements mainly comprise paths in which they are either the initial node or the target ECU. In the security-hardened topology (see Fig. 2d), instead, the two highest values are obtained by the "Body Control Module" and "Network 3". This is again predictable, since these two elements are a necessary path to reach the target ECUs. It is

TABLE IV: Comparison between the results of risk analysis on the initial topology of the Dodge Ram 3500 (left) and on the topology improved by our methodology (right).

Initial Ram topology						Improved Ram topology				
Tot	S	F	P	O		Tot	S	F	P	O
					Overall Risk values					
79.1	19.3	26.9	10.2	22.7		45.3	11.6	13.9	7.0	12.9
					Most dangerous attacks					
15.2	6.7	3.7	0.0	4.7	denyBrake/explodeAirBag	9.6	4.9	1.9	0.0	2.9
12.2	1.7	4.7	0.0	4.7	remoteAcceleration	6.6	1.9	1.9	0.0	2.9
11.2	1.7	4.7	0.0	4.7	startAlarms	5.7	0.0	2.9	0.0	2.9
6.1	0.4	3.4	0.0	2.4	lockDoors	2.75	0.0	1.9	0.0	0.9
4.6	0.0	1.3	3.3	0.0	captureCamera/trackGPS	3	0.0	0.5	2.5	0.0
3.6	0.0	1.3	2.3	0.0	captureMic	2.0	0.0	0.5	1.5	0.0
3.6	0.0	2.3	0.0	1.3	RansomLock	2.0	0.0	1.5	0.0	0.5
2.6	0.0	1.3	1.3	0.0	getAddrBook	1.0	0.0	0.5	0.5	0.0
					Most targeted components					
		19			Head Unit			11		
		15.4			Bluetooth/Cellular/Internet Module			9		
		15.2			ABS/Airbag module			9.6		
		12.2			Cruise Control			6.6		
		12.2			Belt Restraint System			5.7		
		9.7			Door Control Module			4.7		
					Most traversed elements					
		735			Network1			544		
		735			Network2			453		
		228			Head Unit			242		
		183			ABS Module			211		
		183			Airbag Module			211		
		171			Door Control Module			60		
		170			Bluetooth,Radio,Cellular,Internet Module			181		
		170			OBD-II Port			181		
		147			Cruise Control			145		
		135			Belt Restraint System			126		
		97			Remote Keyless Entry			45		
		97			Tire Pressure Monitor			0		
		87			Body Control Module			998		
		0			Network3			998		

important to notice that the traversed elements category is the only one where an increase in values do not reflect a worse topology. In fact, at least in the majority of cases, this behavior indicates longer overall paths and, therefore, lower risks.

VI. Conclusions and Future Works

In this paper, we proposed a semi-automated and topology-based risk analysis framework that helps security analysts in the definition of risks and the design of automotive on-board network topologies. Our framework assesses the security of a given topology, highlights the critical elements of the network design, and helps in the generation of security-hardened solutions on the basis of a generalized risk and threat model methodology. We evaluated our approach on twenty different topologies, demonstrating its effectiveness.

The main limitation of our work is related to the proposed hardened topologies that may not be feasible in real-world scenarios, due to non disclosed proprietary constraints and real time requirements: we were not able to test the vehicles with a set of real-world constraints, but we had to fall back to "empirical" constraints. Future works will focus on taking into consideration a broader spectrum of attacks that allow to develop a more complete and consistent set of countermeasures. In addition, we are planning to develop a metric that consider the real time requirements in terms of the overhead caused by our security-hardened topology.

References

[1] K. Abboud, H. A. Omar, and W. Zhuang, "Interworking of DSRC and cellular network technologies for V2X communications: A survey," *IEEE Transactions on Vehicular Technology*, vol. 65, December 2016.

[2] Telecommunications Standards Advisory Committee (TSAC), "Dedicated shortrange communications in intelligent transport systems."

[3] S. Zanero, "When cyber got real: Challenges in securing cyber-physical systems," in *IEEE SENSORS 2018*, (New Delhi, India), IEEE, 2018.

[4] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage, "Experimental security analysis of a modern automobile," in *2010 IEEE Symposium on Security and Privacy*, pp. 447–462, May 2010.

[5] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, "Comprehensive experimental analyses of automotive attack surfaces," in *Proceedings of the 20th USENIX Conference on Security*, (Berkeley, CA, USA), USENIX Association, 2011.

[6] C. Valasek and C. Miller, "Adventures in automotive networks and control units," August 2013.

[7] C. Miller and C. Valasek, "Remote exploitation of an unaltered passenger vehicle," August 2015.

[8] A. Ruddle, D. Ward, B. Weyl, S. Idrees, Y. Roudier, M. Friedewald, T. Leimbach, A. Fuchs, S. Gürgens, O. Henniger, *et al.*, "Security requirements for automotive on-board networks based on dark-side scenarios." EVITA Deliverable D.2.3, 2009.

[9] Y. D. Sen Nie, Ling Lie, "Free-fall: Hacking tesla from wireless to can bus," in *Black Hat USA 2017*, 2017.

[10] KeenLab Security, "Experimental security assessment of BMW cars: A summary report," 2018.

[11] I. Foster, A. Prudhomme, K. Koscher, and S. Savage, "Fast and vulnerable: A story of telematic failures," in *9th USENIX Workshop on Offensive Technologies (WOOT 15)*, (Washington, D.C.), USENIX Association, 2015.

[12] C. Miller and C. Valasek, "A survey of remote automotive attack surfaces," August 2014.

[13] National Instruments, "Controller area network (CAN) overview."

[14] T. Nolte, H. Hansson, and L. L. Bello, "Automotive communications-past, current and future," in *2005 IEEE Conference on Emerging Technologies and Factory Automation*, pp. 8 pp.–992, Sept 2005.

[15] Bosch, "Can with flexible data-rate specification version 1.0."

[16] FlexRay Consortium, "Flexray communications system protocol specification version 3.0.1."

[17] National Instruments, "Introduction to the local interconnect network (LIN) bus."

[18] D. K. Nilsson, U. E. Larson, F. Picasso, and E. Jonsson, "A first simulation of attacks in the automotive network communications protocol flexray," in *Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems CISIS08*, pp. 84–91, Springer, 2009.

[19] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive can networks — practical examples and selected short-term countermeasures," in *Proceedings of the 27th International Conference on Computer Safety, Reliability, and Security*, SAFECOMP '08, (Berlin, Heidelberg), Springer-Verlag, 2008.

[20] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive can networkspractical examples and selected short-term countermeasures," *Reliability Engineering & System Safety*, vol. 96, no. 1, p. 1125, 2011. Special Issue on Safecomp 2008.

[21] A. Palanca, E. Evenchick, F. Maggi, and S. Zanero, "A stealth, selective, link-layer denial-of-service attack against automotive networks," in *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pp. 185–206, Springer, 2017.

[22] D. Quarta, M. Pogliani, M. Polino, F. Maggi, A. M. Zanchettin, and S. Zanero, "An experimental security analysis of an industrial robot controller," in *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 268–286, May 2017.

[23] R. M. I. Roufa, H. Mustafaa, S. O. T. Taylor, W. Xu, M. Gruteser, W. Trappe, and I. Seskar, "Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study," in *19th USENIX Security Symposium, Washington DC*, pp. 11–13, 2010.

[24] "EVITA project website." https://www.evita-project.org/.

[25] C. Salter, O. S. Saydjari, B. Schneier, and J. Wallner, "Toward a secure system engineering methodolgy," in *Proceedings of the 1998 Workshop on New Security Paradigms*, NSPW '98, (New York, NY, USA), pp. 2–10, ACM, 1998.