IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 7, 2021, accepted October 26, 2021, date of publication November 1, 2021, date of current version November 11, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3124565

A Cyber Security Evaluation Framework for In-Vehicle Electrical Control Units

HAICHUN ZHANG[1], YUQIAN PAN[2], ZHAOJUN LU[2], JIE WANG[3], AND ZHENGLIN LIU[1]

[1]School of Optical and Electronic Information, Huazhong University of Science and Technology, Wuhan 430074, China
[2]School of Cyber Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China
[3]Shenzhen Kaiyuan Internet Security Company Ltd., Shenzhen 518110, China

Corresponding author: Zhaojun Lu (lzj_cse@hust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61874047) and a grant of key technologies R&D general program of Shenzhen, No.JSGG20201102170601003.

ABSTRACT Modern vehicles are equipped with more than 100 Electrical Control Units (ECUs) with over 2500 signals to transmit internally. The application of advanced electronics and communication techniques helps a vehicle transform from an information island into a powerful distribution center. However, a large number of ECUs have introduced a wider range of security threats for vehicles. The attackers can compromise a vehicle remotely through a vulnerable ECU. How to evaluate the cyber security of in-vehicle ECUs has become an important issue. Current Threat Analysis and Risk Assessment (TARA) only carries out theoretical analysis on the potential threats and risks faced by the vehicle in the conceptual design phase of the lifecycle, but lacks the details of actual security evaluation. In this paper, we proposed a Cyber Security Evaluation Framework (CSEF) to independently evaluate the security of the in-vehicle ECUs, which is composed of the asset identification, the threat analysis, the risk assessment, and the security test. The proposed CSEF is applied to a pre-installed On-Bord Unit (OBU) to provide a use case. The use case show that the proposed CSEF is able to figure out assets, threats, risks behind threats, and vulnerabilities of OBU, playing an important role in guiding others to conduct security evaluation. Moreover, CSEF can be extended to evaluate the cyber security of other critical ECUs, such as the Telematic Box, the infotainment units, and the gateway.

INDEX TERMS In-vehicle electrical control units, cyber security evaluation framework, threat analysis, risk assessment.

I. INTRODUCTION

Modern vehicles are not only mechanical tools for transportation, but also mobile smart devices for autonomous driving, audio-visual entertainment, and information sharing, etc [1]. The advancement of network communication and electronics techniques has improved the level of automotive intelligence, informatization, and automation. Automobiles, sensors, mobile terminals, road traffic infrastructures, internet and other smart devices form an information sharing network, which is called Internet of Vehicles (IoV) [2]. ECUs with computing and storage capabilities are responsible for processing massive information received, instructing the actuator to control the vehicle based on the intelligent driving instruction [3].

Modern vehicles have more than 100 ECUs connecting via the in-vehicle buses [4]. Some critical in-vehicle ECUs, such as On-Board Unit (OBU)[5], Telematic Box(T-Box)[6], and vehicle central gateway, have more powerful computing capabilities and data storage capabilities to meet the requirements in complex scenarios. These critical ECUs are equipped with several communication modules such as Bluetooth, WiFi, and cellular communication [7]. For the complex resource management and task management, some ECUs have complex operating system and applications with more vulnerabilities, which provides attackers opportunities to access the resource, obtain the privilege, and control the system. ECUs may also reserve hardware communication interfaces and debug interfaces when the vehicle is released, which introduces serious security risk [8], [9]. In addition, the openness of the wireless channel allows the attackers to intercept the radio waves carrying communication data and parse the radio waves to obtain communication secrets. Unfortunately, since

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Quan.

the in-vehicle ECUs are all connected to the bus network, if an attacker compromises an ECU, they may use it as a springboard to control the vehicle, which will cause privacy disclosure, financial damage, functionality failure, and personal injury. In addition, privacy in IoV services has become an increasingly important security hazard [10], [11]. Attackers can analyze the behaviors and habits of target vehicle users based on private data such as identity and location to lay the foundation for subsequent attacks.

In order to evaluate cyber security of the in-vehicle ECU to fully grasp the security status of the ECU in the vehicle, we propose a Cyber Security Evaluation Framework (CSEF) to comply with the ISO/SAE 21434 standard [12]. CSEF consists of the asset identification [13], the threat analysis [14], the risk assessment [15], and the security test. The asset identification gives the asset that attackers are interested in.The threat analysis systematically exposes the potential threats that the assets may face. The risk assessment rates the level of risk according to the potential consequences of threats. And the security test case set is designed and conducted to verify that if the target object meets the security objectives. The major contributions of this paper are as follows:

First, unlike the existing Threat Analysis and Risk Assessment(TARA) frameworks that only conduct theoretical threat analysis in the conceptual design phase, CSEF combines the TARA and the security test, which makes it can be applied to both the development phase and the security evaluation phase when the vehicle is released.

Second, the framework can be applied to critical in-vehicle ECUs, such as the Telematics Box, the in-vehicle gateways, the infotainment systems, the navigation systems, the domain controllers, and the On-Board Unit (OBU), etc. It has a wide range of applications and good evaluation effect. We applied the proposed cyber security framework to an actual preinstalled in-vehicle OBU to show how to use it. According to the use case, CSEF is able to expose assets, threats, risks, and several actual vulnerabilities of ECU, such as Serial Wire Debug (SWD)[16] information leakage, SWD debugging privilege, and firmware logic leakage, which proves the values of CSEF.

Third, CSEF complies with the ISO/SAE 21434 standard and has been deeply optimized on to be more suitable for the IoVs. Compared with the existing TARA framework, CSEF has more evaluation details to help the evaluator fully understand the security status of the target ECUs.

The rest of the paper is organized as follows. Section II gives the related work. Section III provides the basic knowledge related to modern vehicles and In-Vehicle Networks. Section IV describes the proposed cyber security evaluation framework. An usage case and the effectiveness of the framework is provided in Section V. Finally, section VI draws conclusions.

II. RELATED WORK

The vehicle has a very complex supply chain, both horizontally and vertically, which make it is a difficult tack to manage the supply chain. From the cyber security perspective for the lifetime of a modern vehicle, there will be new threats and attack vectors introduced constantly.To address the security issue of the in-vehicle ECUs, researchers mainly focuses on three directions: threat modeling, risk assessment, and security testing. In 2016, Zhendong Ma *et al.* [17] proposed a practical approach to conduct threat modeling for automotive security analysis during the development lifecycle, but it did not explain how to model and analyze the threat items of in-vehicle systems. Mohammad *et al.* [18] tried to revise the existing threat modeling efforts in the vehicular domain with only four categories of attack objective, which lacked fine-granularity for complex in-vehicle threat analysis. In 2017, Karahasanovic *et al.* [19] demonstrated that the threat modeling process in the computer industry can be adapted and applied in the automotive industry. They did not propose a new approach or optimization to make existing approaches more efficient. Recently, Mohammad *et al.* [20] proposed an approach that combines different existing threat modeling approaches to create a comprehensive and hybrid threat model to support security analysis for in-vehicle systems. However, the approach had no in-depth analysis of vehicular security, and the optimization for vehicular system was till not enough. Since the security evaluation of the vehicle is performed after product development, it requires additional testing on the basis of threat analysis and risk assessment for the security evaluation. The paper [21]–[23] proposed the security evaluation framework for in-vehicle CAN bus. The paper [24]–[27] proposed the test framework for the subsystem in the vehicle. Unfortunately, existing test frameworks only partially examined a certain aspect or a particular sub-system independently. On one hand, there is no universal test framework that can be used for various in-vehicle ECUs. On the other hand, the threat modeling has not been deep optimized to improve the efficiency for the modern vehicles.

In addition, different approaches have been proposed to manage massive threats and risks of complex vehicle systems. Typically, the existing TARA methods recommended in SAE J3061 [28] just focus on the concept design phase in the lifetime of vehicles. In 2020, the ISO/SAE 21434 standard proposed a security framework to guide security practices in the lifecycle of the modern vehicles and the in-vehicle ECUs without the actual implementation process. The two most commonly used tools, EVITA [29] and HEAVENS [30], provide the TARA framework, but lack the details of actual threat analysis. EVITA only describes the threat from the attack probability and HEAVENS considers too few factors in the threat modeling process.

A. EVITA

The EVITA method considers four cyber objectives, "Operational", "Safety", "Privacy", and "Financial". For each of the cyber objectives, the EVITA method identify potential threats with the help of the attack tree approach and classify threat risk level based on severity of the threat outcome and probability of a successful attack. The probability of a successful attack in EVITA is based on the concept of ‘‘attack potential’’ used in IT security evaluation, and considers both the attacker and the system. The severity and attack probability are then combined using a ‘‘risk graph’’ approach to identify the risk associated with each threat.

However, in the EVITA model, assets are not strictly defined, which makes it difficult for security assessors to accurately and comprehensively determine threatened target assets when conducting TARA. The indicators for evaluating the probability of an attack are not comprehensive. Although the evaluation indicators in the IT security field have been introduced, they have not been optimized for the automotive field to fully describe the probability of attacks in the automotive field. automotive field.

B. HEAVENS

The HEAVENS security model adopt Microsoft’s STRIDE [31] approach for identification of the threats associated with the assets of the evaluated objects. The STRIDE extends the original CIA model by correlating threats with security attributes, which can be used to discover and enumerate threats present in a software system. The threat level based on the attack complexity and the impact level based on the severity of the attack jointly determine the risk level.

The threat level based on the attack complexity and the impact level based on the severity of the attack jointly determine the risk level. Similar to the EVITA model, when evaluating the impact of an attack, HEAVENS considers four parameters,‘‘Safety’’, ‘‘Financial’’, ‘‘Operational’’, ‘‘Privacy and legislation’’. But HEAVENS has more detailed attack impact severity scoring rules, which can guide security assessors to conduct more detailed security assessments. However, HEAVENS oversimplifies the evaluation criteria while evaluating the complexity of the attack. It only considers the knowledge level of the attacker, the knowledge of the attack target, the window of opportunity for attack, and the complexity of the equipment used for the attack.

C. OCTAVE

The OCTAVE [32] is a process-driven threat assessment and risk assessment methodology and is best suited for enterprise information security risk assessments. OCTAVE is especially good at bringing together stakeholders with system experience and subject matter experts with security experience through a progressive series of workshops to develop a thorough organizational and technological view of the problem domain. A series of detailed worksheets are completed in the workshops to identify assets, current practices, Cyber security requirements, threats, and vulnerabilities and then to develop a strategy and plan for mitigating risks and protecting assets.

The OCTAVE method focuses on bringing together stake holders of security through a progressive series of workshops, which is best suited for enterprise information security risk assessments but not readily applicable for embedded automotive systems [33].

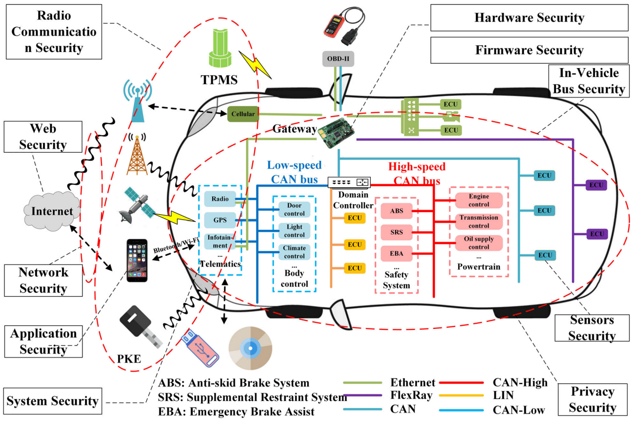


FIGURE 1. The architecture and security concerns of vehicles.

III. SECURITY CONCERNS OF MODERN VEHICLES

The advances in automotive electronics lead to the rapid increase in complexity and diversity of the in-vehicle ECUs. Each ECU relies on a set of sensors and actuators to serve one or more of the Electrical and electronic(E/E)[34] systems or subsystems in the vehicle. These ECUs are interconnected through various bus protocols, forming complex In -Vehicle Networks(IVNs), which play a critical role to improve road safety and driving conditions in the Intelligent Transportation Systems(ITSs) [16], [35].With numerous sensors, actors, and processors being connected on IVNs, a vehicle provides drivers and manufacturers with the data-processing services, including infotainment, vehicle diagnostics, Firmware Over-The-Air [36], and automatic driving [37].

Fig.1 is the typical IVNs based on CAN bus. The original CAN protocol is designed under the assumption that all ECUs are legitimate, trustworthy, and operating according to their specifications [38]. Without considering security, this results in several intrinsic vulnerabilities for the CAN protocol. Moreover, external interfaces such as the Second On-Board Diagnostic (OBD-II)[39], Bluetooth, Wi-Fi, and the Global Positioning System (GPS) provide opportunities for adversaries to break into the open in-vehicle systems through the unprotected CAN bus [40]. Once attackers compromise the CAN bus through numerous external interfaces, they can indirectly control the ECU, thereby realizing vehicle control, leading to serious hazards[41]. According to the automotive electronic and electrical architecture, the universal security evaluation framework for various in-vehicle ECUs involves ten security fields in four levels.

A. PHYSICAL LEVEL

1) HARDWARE SECURITY

The hardware security includes cyber security problems that may be caused by ECU hardware, including Printed Circuit Board security, bus security, hardware interface security, and chip security. Malicious attackers can obtain the information about the chip, hardware interface and bus protocol used in the evaluated object through PCB. Through the hardware interface, the attacker can obtain the startup information about target ECU, debugging privilege, interactive data, firmware, internal storage data, and other information. Through the bus, the attacker can obtain communication data, internal storage data and other sensitive data. By analyzing the chip, the attacker can obtain the chip power consumption, running time, electromagnetic information, which is helpful to deduce key, random number and other critical sensitive data.

2) IN-VEHICLE BUS SECURITY

The ECUs in the vehicle are connected by the buses. And the data is exchanged among ECUs with the help of various bus protocols. There are five bus protocols in modern vehicle for communication, CAN bus, Media Oriented System Transport(MOST) bus[42], FlexRay bus[43], Local Interconnection Network(LIN) bus[44] and vehicle-mounted Ethernet bus[45]. The bus protocols have various performance and are used in different communication scenarios. And the bus protocols were primarily designed for reliable communication without considering cyber security.The lack of encryption, authentication, and integrity checking introduces vulnerabilities for bus protocol making the vehicle vulnerable to cyber-attacks. So, the evaluation framework needs focus on the cyber security of in-vehicle bus to figure out potential cyber security threats.

3) SENSORS SECURITY

For achieving self-driving, the modern automobile is equipped various sensors to perceive surroundings. There are several kinds of sensors embedded in the modern intelligent vehicle, camera, Lidar, ultrasonic radar, millimeter-wave radar, GPS, Beidou and so on[46]. Sensors offer the possibility of autonomous driving, while also introducing the serious cyber security threat to the automobile that has self-driving capability. The malicious attacker can blind or disturb the optical camera to launch the deny of service attack. Besides, the attacker can forge the surrounding information that will be caught by the camera, Lidar and radar to deceive the sensor data processing algorithms, resulting in dangerous autopilot decisions.

B. SYSTEM LEVEL

1) FIRMWARE SECURITY

It is very important for malicious attackers to obtain firmware of ECUs. The attacker can expose the operating logic of the target ECU through reverse engineering technology. The negligence on the part of developers can lead to the disclosure of sensitive data such as passwords, web addresses, accounts, and email addresses that are hard coded in the firmware.

2) SYSTEM SECURITY

The modern vehicle has more and more ECUs that are composed of several embedded systems, controlling the actuators in the vehicle that provide the vehicular functionalities. As for the infotainment system, the ECU is very complex, consisting of an operation system and various applications, which introducing a number of cyber security concerns into the modern automobiles. The system security requires the evaluator to conduct the cyber security evaluation of the vehicle ECU operating system, the system services and applications running on it to identify potential cyber security threats.

C. NETWORK LEVEL

1) RADIO SECURITY

The modern vehicle is no longer an information island. With the development of the IoV technology, vehicles need to communicate with surrounding vehicles, cloud devices, mobile devices, sensors and road infrastructures. And the network architecture of the vehicle becomes more and more complex. The modern vehicle uses Bluetooth, WiFi, NFC/RFID, cellular and other wireless communication technology that occupy various radio frequencies. Because the radio channel is open, any attackers can eavesdrop and manipulate the wireless communication traffic. If the traffic is not encrypted and no integrity check mechanism is adopted to protect the traffic, the malicious attackers can obtain the secrets in the traffic and temper the traffic at will.

2) NETWORK SECURITY

The radio communication security is more concerned about the physical layer and the evaluator is more concerned about the security of the wireless protocol itself. The network security focus on the security of network traffic between traffic the vehicle and the cloud. Compared with the radio communication security, the network security involves a higher level of communication protocol, such as network layer, transport layer and application layer. The malicious attacker may exploit the vulnerabilities of network communication procedure to invade the target evaluated object.

D. APPLICATION LEVEL

1) WEB SECURITY

For more efficient management and more convenient service, there are some automotive management and platform based web applications. The web application likely introduces vulnerabilities that are exploited by the malicious attackers to access the private data, leading to serious information disclosure.

2) APPLICATION SECURITY

For convenience, the vehicular manufacturers provide the functionality that users can control specific automotive behavior through applications of mobile phone. And the applications have potential cyber security threat against the vehicle. The hackers may use the application as a springboard to launch attacks against the target vehicle.

3) PRIVACY SECURITY

In the IoV service, vehicles need to constantly broadcast their real-time information to surrounding vehicles, including

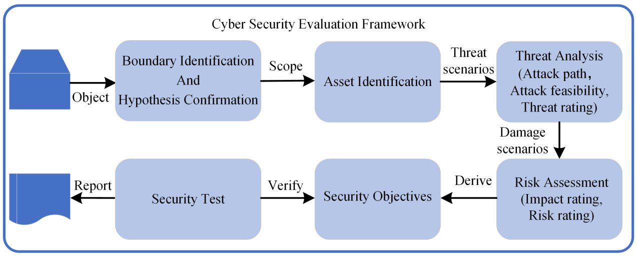

Cyber Security Evaluation Framework
Object
Boundary Identification And Hypothesis Confirmation
Scope
Asset Identification
Threat scenarios
Threat Analysis (Attack path, Attack feasibility, Threat rating)
Damage scenarios
Risk Assessment (Impact rating, Risk rating)
Derive
Security Objectives
Verify
Security Test
Report

FIGURE 2. **The proposed cyber security evaluation framework.**

speed and position, to improve traffic efficiency. The speed, position and other personal information are the privacy that need to be protected. Once the malicious attackers obtain the critical private data, they may pose threats to the drivers' life and property.

IV. CYBER SECURITY EVALUATION FRAMEWORK

The proposed Cyber Security Evaluation Framework (CSEF) is described in Fig.2. The "Object" means the evaluated object in detail, including hardware and software composition, logic functionalities, security features. The "Hypothesis" specifies the supporting environment that the evaluated object needs to run properly, avoiding system errors introduced by abnormal operations. The "Object" and "Hypothesis" are the premise of CSEF that restricts the scope of cyber security evaluation. Based on the "Object" and "Hypothesis", the framework conduct the asset identification, threat analysis, and the risk assessment, which derives the "Asset", "Threat", and "Risk". The "Asset" indicates the data, service, and privilege assets the evaluated should protect. The "Threat" comprehensively describes the potential attack scenarios from the target assets, attack path, and attack impact on the assets. The "Risk" considers financial damage, personal injury, operational failure, and privacy disclosure that a certain threat may cause. Since the "Threat" and "Risk" are only theoretical. We should conduct the practical "Test" case to verify if the threat will actually causes the corresponding risk. And the "Security Objective", derived from the "Threat" and "Risk", is the practical test criteria. The "Security Objective" describes the requirements that the evaluated object needs to protect the target "Asset". When the test result show that the evaluated object does meet the security requirements, the evaluated does have the ability to protect the target assets. And the test passes.

A. OBJECT

The "Object" describes the target object that will be evaluated in details. The object defines the physical and logical boundaries of the evaluated target, distinguishing it from the external entities that do not need to be evaluated. With the help of the "Object", the evaluator can have a detailed knowledge about the hardware, software, logical functionalities, and security capabilities of the evaluated target object, so as to a deeper understanding of the evaluated object and the more sophisticated cyber security evaluation scheme.

B. HYPOTHESIS

The "Hypothesis" explains the assumption we should make about the environment such that the evaluated object is able to provide functionality properly. If the object under evaluation is placed in an operating environment that does not meet these assumptions, the evaluated object may no longer provide all of its functionalities. The hypothesis can be about the physical, human, and runtime aspects of the environment.

With respect to the physical aspects of the physical operating environment, we may assume that the evaluated object is placed in a room designed to minimize electromagnetic radiation, or the admin console of the evaluated object is placed in a restricted access area. In terms of operator, we assume that users are sufficiently trained to operate the evaluated object, or users will not write down their passwords. On the assumption of runtime environment, we may assume that a PC workstation has at least 10 GB of available disk space to run the evaluated object, or the evaluated object will not connect to an untrusted network.

We should note that these assumptions are considered to be true during the assessment: they will not be tested in any way. For these reasons, we can only make assumptions about the running environment and the behavior of the evaluated object must never be assumed.

C. ASSETS

The cyber security is related to the assets that need to be protected. The malicious attackers will try to acquire assets to obtain financial interests or to destroy the target system. Assets come in a variety of forms, from the contents of files or servers, to the availability of instant messaging programs, to the operating privileges of confidential facilities, and so on. Many assets are stored, processed, and transmitted by IT products in the form of information to meet the requirements of the information owner. To avoid the subjective assessment leading to the subjective assessment to situation where almost anything can become an asset, we classify assets into three categories: data, service, and privilege. All attacks against the evaluated object must be aimed to manipulate the three kinds of assets.

1) DATA:

The data is the asset that is stored, processed and transmitted in the evaluated object. The attacker can obtain and tamper the data asset, and can also manipulate the data asset by impersonating the authorized user or the communication entity of the target data. Attributes of the data asset consist of confidentiality, integrity, and availability.

2) SERVICE:

A service is a function provided externally by the evaluated object. Attackers can take corresponding actions to force the target to lose the ability of external service delivery, which will affect the availability of the target service. The attribute of the service asset is availability.

3) PRIVILEGE:

The evaluated object usually sets different levels of permissions for system resources based on different roles.The privilege of different level can access different resource. Malicious attackers often want to gain higher privileges to access more important resources. The attribute of the privilege asset is availability.

D. THREATS

A threat consists of a hostile behavior to the asset, which will affect one or more attributes of the asset, and the asset reflects its value through these attributes. A threat agent can be described as a single entity, but in some cases it may be better described as an entity class or group of entities. Threat agents can be attackers, users, computer processes, unexpected events, and so on. The threat can be further described in terms of expertise, resources, opportunities, and motivations.

In the proposed framework, we represent the threat with several attributes, including name, description, type, entry, path, connectivity, threatened asset, risk, vulnerability and countermeasure. According to the STRIDE threat model, there are six types of threats: spooling, tampering, repudiation, information disclosure, Deny of Service (DoS), and elevation of privilege. The entry, the path, and the threatened asset describe in detail how potential attacks pose a threat to targeted assets. The potential attack behind the threat may access the evaluated object through physical connectivity, near-field wireless connectivity and remote wireless connectivity. The risk describes the consequences that a potential attack might have on the system and the likelihood of a successful execution of the potential attack. The vulnerability is the attack event that is actually occurred. And the countermeasure is the security mechanism we should take to protect the target system against known threats.

The proposed framework adopts the attack tree model to model the threat.The framework will model the threat with ten aspects based on the hardware architecture, software architecture, and the external interactive entity of the evaluated object. As for the evaluated object that is related with the vehicle, the threat model should consider hardware security, firmware security, system security, In-Vehicle bus security, radio communication security, network security, web security, sensors security, and privacy security.

E. RISKS

The risk describes the serious consequences that a potential attack might have on the target object and the external world, the threat severity, and the likelihood of a successful execution of the potential attack. The level of the risk is determined by the matrix of threat severity and impact severity.

1) THREAT SEVERITY (TS)

The automotive industry is similar to the traditional computer industry, but there are some differences. The automotive industry has stringent security requirements. The harm caused by security accidents in the computer industry may only be financial loss or privacy leakage, but security accidents in the automotive industry may cause personal injury. Secondly, in the long lifecycle of automobiles, the architecture of in-vehicle ECU changes slowly, making the security evaluation more valuable. Based on the above characteristics, the framework optimized the Common Vulnerability Scoring System (CVSS) [47] to propose a threat severity assessment mechanism.

In general, the threat severity takes into account a number of different factors, including attack vector, attack scope, attack method, time window, expert knowledge, target information, attack equipment, privileges required, user interface, confidentiality impact, integrity impact, availability impact, and weight coefficient. The 13 metrics are divided into three groups: exploitability metrics, knowledge metrics, and impact metrics. The exploitability metrics represent the specific details of the actual attack of the threat. The knowledge metrics represent the information, knowledge, and authorization before the attack of the threat. And the impact metrics describe the impact on the evaluated object introduced by the threat. The parameters and scores are shown in TABLE 1.

a: EXPLOITABILITY METRICS(EM)

a.1) Attack Vector(AV): The attack vector measures how an attacker connects to a target system. There are three kinds of attack vectors. The remote wireless connectivity means that an attacker can launch a remote attack on a target via either LTE or 5G cellular networks. The near-field wireless connectivity means that an attacker can launch a certain degree of remote attack to the target only through WiFi, Bluetooth and other short-range wireless networks. The physical attack indicates that an attacker must physically access the target in order to launch an attack.

a.2) Attack Scope(AS): The attack scope refers to the range that the threat affects. Some threats affect only one target, some affect multiple targets, and some affect all targets.

a.3) Attack Method(AM): The attack method identifies how difficult an attack is. The attack can be a single operation or a combination of operations.

a.4) Attack Equipment(AE): The difficulty of acquiring an attack equipment varies with the threat. The attack equipment may be an open source device or a complex device that needs to be customized.

*a.5) Time Window(TW):*The time window indicates when an attacker can launch an attack on a target. The attacker can launch an attack at any time, or when the target is in a specific state, or only when the target triggers a specific behavior.

TABLE 1. Assets, threats, and risks.

Metric	Parameter	Score
Attack Vector(AV)	Remote wireless attack	1.0
	Short range wireless attack	0.7
	Physical contact attack	0.3
Attack Scope(AS)	Signal device	1.0
	Multiple devices	0.7
	All devices	0.3
Attack Method(AM)	Single aggressive behavior	1.0
	Multiple aggressive behaviors	0.8
Attack Equipment(AE)	Open common hardware and software	1.0
	Open dedicated hardware and software	0.9
	Customized or proprietary hardware or software	0.7
	Multiple customized or proprietary hardware or software	0.6
Time Window(TW)	Attacks can be launched at any time	1.0
	Attacks can be launched frequently	0.8
	Attacks can be launched under certain conditions	0.6
Expert Knowledge(EK)	Amateur	1.0
	Technician	0.8
	Expert/hacker	0.6
	Multi-field Security Expert Group	0.3
Target Information(TI)	Target information is publicly available	1.0
	Access to target information is restrictedg	0.8
Privileges Required(PR)	Authorization required	1.0
	Authorization not required	0.7
Visible Interface(VI)	Visible interface required	1.0
	Visible interface not required	0.7
Confidentiality Impact(CI)	No impact	0
	Low impact	0.6
	High impact	1.0
Integrity Impact(CI)	No impact	0
	Low impact	0.6
	High impact	1.0
Availability Impact(CI)	No impact	0
	Low impact	0.6
	High impact	1.0

Score is the score value of the 13 metrics in the "Threat Severity", which indicates the difficulty of launching an attack corresponding to a threat. For example, if the attack vector of a threat is remote wireless attack, according to the TABLE 1, the score of the metric is 1.0, which indicates that long-range attacks are easier to launch than physical contact attacks.

b: KNOWLEDGE METRICS(KM)

b.1) Expert Knowledge(EK): The expert knowledge indicates whether the attacker is an amateur, skilled person, expert, or expert in a variety of fields.

b.2) Target Information(TI): The malicious attacker collects information about the target before launching the specific attack. And the metric of target information measures how easy it is to get information.

b.3) Privileges Required(PR): The metric shows whether the attack was authorized by the target user.

b.4) Visible Interface(VI): The metric indicates whether the attack needs the visible interface.

c: IMPACT METRICS(IM)

c.1) Confidentiality Impact(CI): The metric indicates the impact on the confidentiality of data. The attacker may obtain critical data that has serious and direct impact on the target. For example, if the hackers obtain the key material of the system manage, they can access most resource of the target system.

TABLE 2. The Threat Severity Levels.

$Score_{TS}$	Threat Severity Value(TSV)	Threat Severity Level(TSL)
$[0, 2)$	0	Light
$[2, 4)$	1	Low
$[4, 6)$	2	Medium
$[6, 8)$	3	High
$[8, 10]$	4	Critical

$Score_{TS}$ represents the complexity of a certain threat, indicating how easy it is to actually transform the threat into an attack. The smaller the score, the more complex the attack, the harder it is to actually happen, and the lower the threat severity level.

c.2) Integrity Impact(II): The metric indicates the impact on the integrity of data. An attacker can modify the data as he wishes to make it completely incomplete or completely unprotected.

c.3) Availability Impact(AI): The metric indicates the impact on the availability of data. An attacker may cause a complete denial of service or degrade the performance of the target data system.

We define the $Score_{TS}$ as the score of the threat severity, the $Score_{EM}$ as the score of the exploitability metrics, the $Score_{KM}$ as the score of knowledge metrics, and the $Score_{IM}$ as the score of impact metrics.The setting of the score can qualitatively indicate the importance of a certain metric, and the value refers to CVSS. If necessary, other values can also be used completely, as long as it can reflect the difference between different parameters of a certain metric. And then, the scores of metrics are calculated as follows.

$$Score_{EM} = \alpha_1 S_{AV} + \alpha_2 S_{AS} + \alpha_3 S_{AM} + \alpha_4 S_{AE} + \alpha_5 S_{TW} \tag{1}$$

$$Score_{KM} = \beta_1 S_{EK} + \beta_2 S_{TI} + \beta_3 S_{PR} + \beta_4 S_{VI} \tag{2}$$

$$Score_{IM} = \gamma_1 S_{CI} + \gamma_2 S_{II} + \gamma_3 S_{AI} \tag{3}$$

$$Score_{TS} = 10 * Score_{EM} * Score_{KM} * Score_{IM} \tag{4}$$

The $\alpha_1, \ldots, \alpha_5$ are weight coefficients of the metrics in the exploitability metrics group. The $\beta_1, \ldots, \beta_4$ are weight coefficients of the metrics in the knowledge metrics group. And the $\gamma_1, \gamma_2, \gamma_3$ are weight coefficients of the metrics in the impact metrics group. The S with various subscripts represent the value of metric parameter. To restrict the score within 10, we use the coefficient 10 in the formula for calculating the $Score_{TS}$. According to the score of the threat severity, the threat can be classified into several levels by the proposed evaluation framework, as depicted in TABLE 2.

2) ATTACK PROBABILITY(AP)

The probability that a potential attack will succeed is called attack probability. The attack probability varies according to the threat analysis and risk assessment methods used in the evaluation framework. The framework measures the attack probability of a special risk item from five impact factors, namely professional knowledge, auxiliary tools, target knowledge, target environment and time cost, as shown in TABLE 3.

TABLE 3. The Attack Probability Metrics.

Metric	Parameter	Score
Auxiliary Tools(AT)	Public hardware and software	1.0
	Customized or proprietary hardware or software	0.6
Professional Knowledge(PK)	Amateur	1.0
	Technician	0.8
	Expert/hacker	0.4
	Multi-field Security Expert Group	0.2
Target Knowledge(TK)	Target information is publicly available	1.0
	Access to target information is restricted	0.6
Target Environment(TE)	The attack environment is simple	1.0
	The attack environment is complex	0.6
Time Cost(TC)	In a day	1.0
	In one week	0.6
	In one month	0.4
	More one month	0.2

$Score$ is the score value of the metrics in the "Attack Probability", which indicates the probability of launching an attack corresponding to a threat. For example, if the attack vector of a threat is remote wireless attack, according to the TABLE 1, the score of the metric is 1.0, which indicates that the probability of a long-range attack is higher.

a: PROFESSIONAL KNOWLEDGE(PK)

Professional Knowledge(PK): The professional knowledge indicates the level of expertise required by the attacker to carry out the attack. The attacker may be an amateur, skilled person, expert, or expert in a variety of fields.

b: AUXILIARY TOOLS(AT)

The auxiliary tool indicates whether the tool is special tool or the tool that can be obtained publicly.

c: TARGET KNOWLEDGE(TK)

The target knowledge indicates whether the information about the target object can be obtained publicly. In some case, we cannot obtain any information about the target object because of the private information.

d: TARGET ENVIRONMENT(TE)

The metric tells us whether the attack environment is simple or complex. In the simple attack environment, it is easier for an attacker to access the target and carry out an attack.

e: TIME COST(TC)

The time cost indicates how long the attack will take. The factors affecting the time cost include the construction of attack environment, the collection of target object information, the development of attack tool, vulnerability mining, vulnerability verification and so on.

We define δ as the vulnerability conversion factor that indicates the probability of a successful potential attack in the proposed evaluation framework. The S with various subscripts represent the value of metric parameter. And α_1, α_2 are weight coefficients of the metrics.

$$\delta = S_{AT} * (\alpha_1 S_{PK} + \alpha_2 S_{TK}) * S_{TE} * S_{TC} \tag{5}$$

3) IMPACT SEVERITY(IS)

The impact severity indicates the potential harm of the threat. The proposed evaluation framework takes financial damage, personal injury, operational failure and privacy disclosure as the reference indexes to evaluate the potential impact level of a certain threat.

a: FINANCIAL DAMAGE(FD)

The financial damage considers all financial losses that can be either direct or indirect. Direct financial damages may include product liability issues, legislation issues and product features. For example, the attack may lead to product recalls and significant losses. And the attack may result in loss of sales due to product defects. On the other hand, indirect financial damages include damage to reputation, loss of market share, intellectual property infringement and etc. To summary, the financial damage is the sum of direct and indirect costs for the manufacturer and the root cause may originate from any of the stakeholders.

b: PERSONAL INJURY(PI)

The personal injury indicates the damage to the person caused by the attack against the evaluated object. And the safety to the passengers and pedestrian is the highest priority. According to ISO 26262 [48], the injury can be classified as no injury, light and moderate injuries, severe injuries, and life-threating injuries and fatal injuries.

c: OPERATIONAL FAILURE (OF)

The operational failure includes operational damages caused by unexpected loss or control of a vehicular function. The attacker may control the vehicular function or make the vehicular function deny of service. Examples of such operational damages include critical and secondary functionalities loss. However, in certain situations, operational damages may cause safety and financial damages. If the safety-related vehicle functionalities are controlled by malicious attackers, the personal safety of passengers and road users will not be guaranteed.

d: PRIVACY DISCLOSURE(PD)

The privacy disclosure considers damages caused by privacy violation of stakeholders, such as vehicle owner, driver and passengers. Usually, the privacy disclosure do not have direct injury, financial and operational dimensions. However, in certain situations, privacy violations may lead to the loss of access to certain market and operational damages to the stakeholders.

The proposed evaluation framework refers to the HEAVENS Security Model putting forward the impact severity assessment method. The parameters and scores are shown in TABLE 4. We define the $Score_{IS}$ as the score of the impact severity. The S with various subscripts represent the value of metric parameter. And then, the score of impact severity is calculated as follows. According to the score of the

TABLE 4. The Impact Severity Metrics.

Metric	Parameter	Score
Personal Injury(PI)	No injury	0
	Light and moderate injuries	10
	Severe injuries	100
	Life-threating injuries and fatal injuries	1000
Financial Damage(FD)	No damage	0
	Light damage	10
	Moderate damages	100
	Severe damages	1000
Operational Failure(OF)	No damage	0
	Light damages	1
	Moderate damages	10
	Severe damages	100
Privacy Disclosure(PD)	No damage	0
	Light damages	1
	Moderate damages	10
	Severe damages	100

Score is the score value of the metrics in the "Impact Severity", which indicates the degree of potential harm caused by a threat. The score is higher, the harm is more serious.

TABLE 5. The Impact Severity Level.

$Score_{IS}$	Impact Severity Value(TSV)	Impact Severity Level(TSL)
$[0, 1)$	0	Light
$[1, 20)$	1	Low
$[20, 100)$	2	Medium
$[100, 1000)$	3	High
$[1000, +\infty)$	4	Critical

$Score_{IS}$ represents the severity of the harm caused by a threat. The higher the score, the greater the harm caused by threat and the higher the risk. Based on different scores, the potential impact severity caused by threats are divided into 5 levels.

TABLE 6. Risk Matrix.

Risk Level(RL)		Impact Severity Level (ISL)				
		0	1	2	3	4
Threat Severity Level (TSL)	0	Light	Light	Light	Light	Low
	1	Light	Low	Low	Low	Medium
	2	Light	Low	Medium	Medium	High
	3	Light	Low	Medium	High	High
	4	Low	Medium	High	High	Critical

impact severity, the impact of a certain threat can be classified into several levels by the proposed evaluation framework, as depicted in TABLE 5.

$$Score_{IS} = \delta * (S_{PI} + S_{FD}) + S_{OF} + S_{PD} \qquad (6)$$

4) RISK MATRIX

As depicted in TABLE 6, the proposed evaluation framework combines the threat severity and the impact severity level to derive the risk security level. The higher the threat severity level and the higher the impact severity level, the higher the possibility that the threat will be transformed into an actual attack and the greater the potential harm caused, the higher the risk level.

F. SECURITY OBJECTIVES AND TESTS

Threat analysis and risk assessment are usually conducted in the concept design phase of the life cycle to derive the security objectives that will be achieved through technology in subsequent development phase. However, the products in the markets are in the operation phase. After the threat analysis and risk assessment, the evaluation framework just presents the theoretical cyber security threats and the corresponding risk, which is not enough for the products that have been finalized and marketed. Given that, we need to design the conduct the test to verify whether the product meet the security objectives derived from the threat analysis and risk assessment. The security objectives are the test criteria of the evaluated object. If the evaluated object does not meet the security objectives, there may be vulnerabilities. The test case is designed based on the threat and risk and is conducted based on the security objectives. According to the cyber security framework of the vehicle, the test case in our evaluation framework is designed based on following ten aspects.

1) Hardware security test case set. The test case set includes PCB printed word test, hardware bus protocol test, hardware interface test, and chip security test.
2) Firmware security test case set. The test case set includes sensitive data test, sensitive logic test, and firmware tampering test.
3) System security test case set. The test case set includes port scanning test, vulnerability test, security audit test.
4) In-Vehicle Bus security test case set. The test case set includes bus data interception test, bus data tampering test, bus node spoofing test, and bus DoS test.
5) Radio security test case set. The test case set includes WiFi security test, Bluetooth security test, TPMS security test, GPS security test, DSRC security test, C-V2X security test, and other radio communication protocols used in the vehicle.
6) Network security test case set. The test case set includes data encryption test, data integrity check test, entity spoofing test, data replay test, and man-in-the-middle attack test.
7) Web security test case set. The test case set includes common web security test items, such as OWASP top ten web security vulnerabilities.
8) Application security test case set. The test case set includes app environment security test, app code security test, app service interface test, local data security test, network communication security test, and authentication certification security test.
9) Sensors security test case set. The test case set includes Lidar security test, camera security test, millimeter wave radar security test, and ultrasonic radar security test.
10) Privacy security test case set. The test case set includes position privacy security test and identity privacy security test.

G. COMPARISONS

The comparisons between CSEF and other TARA methods are shown in Table 7. Compared with other methods that can

TABLE 7. Comparisons between CSEF and other methods.

Method	Asset Identification	Threat Analysis	Risk Assessment	Phase
CSEF	Attack Tree	comprehensive	comprehensive	Design/Test
HEAVENS	brainstorming	simple	simple	Design
EVITA	Attack Tree	simple	simple	Design
OCTAVE	brainstorming	simple	simple	Design
CVSS	N/A	N/A	comprehensive	Operation
Attack Tree	Attack Tree	simple	simple	Design

only be used in the conceptual design stage, CSEF can not only reduce the cyber security risk of the in-vehicle ECUs in the conceptual design stage, but also guide testers to conduct cyber security testing after the ECUs are released. In addition, CSEF uses the attack tree method to identify important assets, which is more comprehensive than brainstorming.Compared with the traditional TARA method, threat analysis and risk assessment method in CSEF are deeply optimized for the automotive field.

V. USE CASE

A. OBJECT AND HYPOTHESIS

The OBU under the experiment in our work is an active dual-chip pre-installed OBU product, which is an important in-vehicle ECUs in the ETC system. The ETC system is an internationally recognized effective technology to solve the problem of automatic charging at road, bridge, parking lot and other toll station. In the ETC system, the OBU communication with the RSU adopting the DSRC protocol. And the wireless communication technology effectively improve traffic efficiency and alleviate traffic congestion in Bridges and tunnels, expressway entrances and exits, urban trunk roads and other places with large traffic flow.

As shown in Fig.3, the ETC system mainly includes the lane control system, background database system, RSU, OBU and Integrated Circuit(IC) card. The lane control system serves as road gate control, passage light control, vehicle capture, etc. The background database system assists the completion of registration, settlement, and related operations. And the OBU, RSU, IC card are used to automatic vehicle identification, automatic cost collection and other functionalities. The OBU stores the vehicle identification information such as the prepaid amount, vehicle model, vehicle color, license plate number and owner information. The RSU installed in the gantry frame at the road edge of toll station or above the lane use RF antenna and microwave technology to read the relevant information in the on-board OBU, identifying the vehicle and calculating the charge amount, thus completing the automatic non-parking charge.

The ETC system adopts 5.8GHz frequency band as communication frequency band, mainly including physical layer, data link layer and application layer. The physical layer specification provides data transmission, synchronization and timing functions to realize the physical connection of data transmission. The ETC technical national standard specifies the physical layer parameters and performance standards for OBU devices. The data link layer specifies the key parameters, data frame format, communication link establishment,Media Access Control(MAC) and Logical Link Control(LLC) sub-layers in ETC communication process, which are used to regulate the communication between RSU and OBU. The application layer describes the core framework of ETC system and the basic services provided by the kernel.

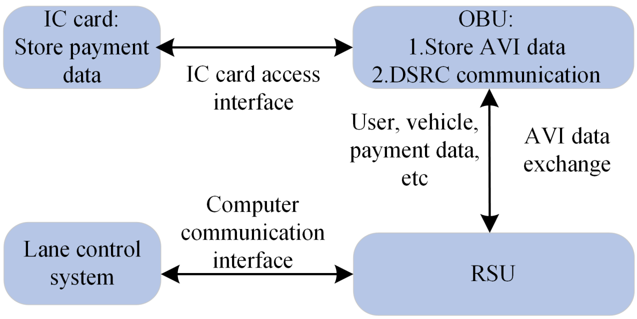


FIGURE 3. The architecture of the ETC system.

The OBU plays a role of both microwave communication and information storage in ETC system. On the one hand, the OBU stores vehicle information, road information, owner information and other key information used for road and bridge cost settlement. On the other hand, the 5.8GHz microwave frequency band is used for DSRC communication between OBU and RSU. According to the power supply mode, the OBU can be divided into active OBU and passive OBU. After awaken, the active OBU will actively send the vehicle information stored in the OBU. The passive OBU transmits relevant information through induced current. According to the presence or absence of IC card, the OBU can be divided into single-chip type and dual-chip type. Single-chip OBU has no IC interface, which is of high risk. Dual-chip OBU has IC card interface, which can fuse the functions of OBU and IC card with higher security and relatively high cost.

The chip and hardware interface of the OBU are shown in Fig.4. From the printed word in the printed circuit board, there are SWD and Universal Asynchronous Receiver/Transmitter(UART) interfaces for debug in the development phase. And there are BLE and CAN interfaces for communication with BLE devices and CAN bus network. The main chips used in the OBU are SE chip, BLE chip, ETC chip, CAN controller, and s32k118 Micro Controller Unit(MCU). The interfaces and chips in the OBU can help to infer the functionalities provided form the OBU. In summary, the information obtained from the OBU hardware board is as follows.

1) The Bluetooh Low Energy(BLE) chip and BLE communication ability.
2) The security element chip with secure storage and the data encryption/decryption ability.

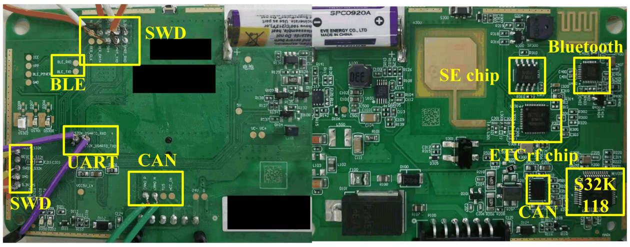


FIGURE 4. The hardware board of OBU.

3) The CAN controller and communication functionality with vehicle-mounted CAN network.
4) The MCU to coordinate and control the operation of each chip module of the OBU.
5) The ETC Radio Frequency(RF) chip to process DSRC communication data and the capability to communicate wirelessly with RSUs.
6) The UART, SWD, CAN, BLE, ISO7816 and other external interfaces.
7) The Inter Integrated-Circuit(I2C) and Serial Peripheral Interface(SPI) bus protocols for data transmission between chip modules.

Based on the proposed security evaluation framework and the information of the OBU to be evaluated, we conducted the following experiment. The experiment analyzed the cyber security risks of the OBU products from five perspectives: asset, hypothesis, threat, risk and test. The asset is the valuable data, privilege, and functionality service in the target OBU. The hypothesis is a prerequisite for the cyber security evaluation of the target OBU, which ensures that the OBU to be evaluated can operate normally in the environment constrained in the hypothesis. The threat is a potential cyber security threat to target OBU. The risk is the potential harm that cyber security threat may bring. And the test assesses the probability that the risk will translate into an actual attack.

The hypothesis specifies the preconditions for stable and normal operation of the target OBU. The use of hypothesis in the cyber security evaluation framework can eliminate cyber security threats caused by manufacturing, improper operation, improper management, etc. The hypothesis of the OBU is as follows.

1) SECURE MANUFACTURING

During the manufacturing integration phase of the OBU life cycle, it is assumed that appropriate technologies and measures have been taken to ensure the security of the OBU assets, including the generation, installation and import of materials such as the initial key.

2) STANDARDIZED OPERATION

At each stage of the OBU life cycle, it is assumed that the operation of the target OBU is performed in accordance with the standard procedure and will not infringe the OBU asset due to improper operation.

3) STANDARDIZED MANAGEMENT

It is assumed that all documents, data, keys and other materials related to the target OBU are under the control of the management, and no material leakage event will occur.

4) TRUSTED STAFF

It is assumed that all the technicians and managers who come into contact with the OBU before it is put on the market are credible and will not pose a security threat to the OBU assets.

5) PHYSICAL PROTECTION

Suppose that the OBU product has a corresponding physical protection mechanism.

6) RUNNING NORMALLY

It is assumed that the OBU is running normally during the running phase of the life cycle and will not run incorrectly with no reason.

B. ASSETS OF OBU

According to the ‘‘Object’’ and ‘‘ Hypothesis’’, the framework conduct the asset identification, threat analysis, and the risk assessment. Based on the knowledge of the OBU in section ‘‘Object and Hypothesis’’, the firmware in the OBU is the binary codes without operating system. And the OBU does not communicate with the cloud server. So, according to the security concerns in section II and the characteristics of the OBU, the proposed evaluation framework identifies the valuable assets in the OBU from four dimensions of hardware security, firmware security, in-vehicle bus security, and radio communication security.

The assets of the target OBU are listed in TABLE 8. The CIA (Confidentiality, Integrity, and Availability) security model is adopted in the evaluation framework to indicate that we should protect which security characteristic of the target asset.

According to the information obtained from the hardware board and the functionalities inferred, we can identify the assets of OBU. The data assets include the data that needs to be transmitted during the communication process and the data stored by the device. The Privilege assets include system code execution permissions that can be leaked through device hardware and wireless interfaces. The service assets include the service provided form the OBU.

C. THREATS AND RISKS

In order to apply the security evaluation framework to the automotive field, we uses the attack tree method to conduct threat analysis under the automotive security framework. The threats described in TABLE 9 are classified as follows. All assets in the TABLE 8 have threats and potential risks. All threats will be analyzed in order to grasp the potential attack point of the target OBU. In addition, the risk level will be provided to indicate the severity of the impact to person, environment, and vehicle.

TABLE 8. Assets.

Asset	C	I	A	Type
PCB data	✓			Data
UART data	✓		✓	Data
Privilege from UART			✓	Privilege
SWD data	✓	✓		Data
Privilege from SWD			✓	Privilege
Firmware logic	✓	✓		Data
Firmware data	✓	✓		Data
I2C data	✓	✓		Data
SPI data	✓	✓		Data
DSRC data	✓	✓	✓	Data
BLE data	✓	✓	✓	Data
BLE service			✓	Service
CAN data	✓	✓	✓	Data
CAN service			✓	Service
ISO7816 data	✓	✓	✓	Data
SE chip data	✓	✓		Data

All assets are divided into three categories, data, services, and privileges.
C = Confidentiality; I = Integrity; A = Availability.

1) HARDWARE SECURITY THREATS

a: PCB DATA LEAKAGE

The printing word on the PCB board of OBU and other in-vehicle ECUs will reveal information such as hardware debugging interface, chip, communication protocol, etc., which is of great significance for information collection before malicious attacks.

b: UART DATA LEAKAGE

In addition to being used for serial communication, the UART interface is often used as a debugging interface. A malicious attacker can physically contact the UART interface to obtain system startup information, including key information such as u-boot version, chip name used by the system, memory layout, and so on.

c: UART PRIVILEGE

Malicious attackers can not only obtain sensitive data of the ECU and system through the UART interface, but also obtain system privileges. If the developer reserves a command line debugging interface with super authority in the UART interface, the attacker can crack the login password to obtain the system super authority, which will bring great harm.

d: SWD DATA LEAKAGE

If the debugging function is not turned off before the ECU is re-released, the attacker can start the hardware debugging function of the target ECU through the Joint Test Action Group(JTAG) or SWD debugging protocol. This functionality allows the attacker to read the data inside the CPU and the firmware of the ECU.

e: SWD PRIVILEGE

Besides the firmware and the sensitive data inside the CPU, the SWD interface may allow the attacker to load the firmware into the ECU to achieve the purpose of controlling the ECU.

f: I2C/SPI/ISO7816 DATA INTERCEPTION

With the help of an oscilloscope and digital logic analyzer, the attacker can monitor and parse protocol data to obtain communication content. If the time is right, the attacker can even execute the attack within the firmware transfer window to obtain the complete firmware.

g: I2C/SPI/ISO7816 DATA TAMPERING

On the basis of monitoring and parsing the protocol data, the attacker can change the protocol data transmitted on the bus within a specific time window to interfere with normal data transmission.

h: SE CHIP DATA LEAKAGE

Although it is a difficult task to compromise a security chip, it is still a potential security hazard to attack the security chip through side channel analysis, fault injection attacks to access the data inside the chip. Attackers may obtain sensitive information such as keys stored in the security chip.

i: SE CHIP DATA TAMPERING

Furthermore, an attacker can tamper with the data stored inside the chip if the security chip have been compromised.

2) FIRMWARE SECURITY THREATS

a: FIRMWARE LOGIC LEAKAGE

After obtaining the firmware, the attacker can reveal the functional logic of the firmware through reverse engineering which helps understand the operation of the target ECU and discover potential security vulnerabilities that can be exploited.

b: FIRMWARE DATA LEAKAGE

In addition to the function logic, due to the negligence of the developer, there may also be plaintext data in the firmware, which may expose sensitive information such as key IP addresses, email addresses, and security keys.

c: FIRMWARE LOGIC TAMPERING

It is obvious that the attacker can tamper the content of the firmware to precisely control the target ECU.

3) IN-VEHICLE BUS SECURITY THREATS

a: CAN DATA INTERCEPTION

The CAN bus protocol lacks identity authentication, and attackers can forge malicious nodes to connect to the vehicular CAN bus network. Based on the broadcast transmission mechanism of the CAN bus, an attacker can receive all the data transmitted by the CAN bus. Due to the lack of security encryption mechanism, an attacker can obtain the content of the data transmitted by the CAN bus.

b: CAN DATA TAMPERING

Not only the interception, the CAN bus protocol does not have a data integrity check mechanism, and malicious attackers can tamper with the data content without the receiver's awareness.

c: CAN ENTITY SPOOFING

As mentioned above, without the protection of an identity authentication mechanism, a malicious attacker can fake a CAN controller as a legitimate node to connect to the vehicular CAN bus network.

d: CAN BUS DoS

In addition, the CAN bus protocol uses a priority-based blanking mechanism to solve the problem of bus competition. The attacker can construct CAN bus data frames with higher priority and send them to the CAN bus to occupy CAN bus resources for a long time, causing denial of service effect.

4) RADIO SECURITY THREATS

a: DSRC/BLE DATA INTERCEPTION

The openness of wireless communication channels allows any attacker to capture electromagnetic waves transmitted in the air through software define radio equipment and parse them into data bit streams according to protocol specifications. Without an encryption algorithm to protect the transmission content, an attacker can obtain the transmitted message content.

b: DSRC/BLE DATA TEMPERING

In the absence of a data integrity check mechanism, an attacker can tamper with the data content transmitted in the wireless channel.

c: DSRC/BLE ENTITY SPOOFING

If the wireless communication protocol does not use an identity authentication mechanism or uses a weak identity authentication mechanism to verify the identity of the communicating entity, an attacker can break through the identity authentication and forge an arbitrary communicating entity.

d: BLE DoS

Like other wireless technologies, Bluetooth is also vulnerable to DoS attacks, making the ECU's Bluetooth interface unusable and draining the ECU's battery.

In summary, the attacker may launch an attack on the target device through any potential attack path, in order to steal the target device's data, destroy its service, or obtain the execution privilege of the target device. And in terms of the impact of attacks on target assets, threats can be classified using the stride model.

According the threat analysis, we assign the attributes of the threat to calculate the threat severity level. And the impact severity level can be calculated based on the risk assessment. Furthermore, the risk level can be determined by the risk matrix that comprehensively considers the effects of threat severity level and impact severity level on risk level. The specific risk level values are shown in TABLE 10.

TABLE 9. Threats.

Name	STRIDE	Entry	Connectivity
PCB data leakage	I	PCB	Physical
UART data leakage	I	UART	Physical
UART privilege	E	UART	Physical
SWD data leakage	I	SWD	Physical
SWD privilege	E	SWD	Physical
Firmware logic leaks	I	SWD/UART	Physical
Firmwar data leaks	I	SWD/UART	Physical
Firmware logic tampering	T/E	SWD/UART	Physical
I2C data interception	I	I2C	Physical
I2C data tampering	T	I2C	Physical
SPI data interception	I	SPI	Physical
SPI data tampering	T	SPI	Physical
DSRC data interception	I	DSRC	Short-range wireless
DSRC data tampering	T	DSRC	Short-range wireless
DSRC entity spoofing	S	DSRC	Short-range wireless
BLE data interception	I	BLE	Short-range wireless
BLE data tampering	T	BLE	Short-range wireless
BLE entity spoofing	S	BLE	Short-range wireless
BLE DoS	D	BLE	Short-range wireless
CAN data interception	I	CAN	Physical
CAN data tampering	T	CAN	Physical
CAN entity spoofing	S	CAN	Physical
CAN bus DoS	D	CAN	Physical
ISO7816 data interception	I	ISO7816	Physical
ISO7816 data tampering	T	ISO7816	Physical
SE chip data leakage	I	SE chip	Physical
SE chip data tampering	T	SE chip	Physical

Entry indicates the attack entry, and connectivity indicates whether the attack is initiated remotely or via a physical connection..
S = Spoofing Identity;T = Tampering;R = Repudiation;I = Information Disclosure; D = Denial of Service; E = Elevation of Privilege.

D. SECURITY OBJECTIVES AND TESTS

After TARA, we have systematically grasped the threats faced by the target OBU and the potential impact that the threats may cause. In order to reduce the security risk of the target OBU, theoretically, corresponding security measures need to be taken to protect the OBU from threats. These security measures are the security goals. However, whether the target OBU has actually taken security measures requires security testing to verify. Only by passing a security test with clear inspection standards, does it show that the target OBU has a certain degree of security protection capability, which can be regarded as achieving the security goal.

The security objectives against threats are shown in TABLE 11. Every security threat has a corresponding security goal as a security protection measure. We need to design test sets to verify whether the OBU actually meets the security goal. If a certain security goal is not met, the OBU faces security threats, and there are corresponding security risks introduced by the threat. The test cases are following.

1) PCB TEST

In the PCB test, the security assessor needs to carefully examine the printed information on the PCB with the help of tools such as a microscope and a magnifying glass to see if

TABLE 10. Risks.

Threat	TSL	ISL	RL
PCB data leakage	Medium	Low	Low
UART data leakage	Medium	Low	Low
UART privilege	Low	High	Medium
SWD data leakage	Medium	Low	Low
SWD privilege	Low	High	Medium
Firmware logic leaks	Medium	High	Medium
Firmware data leaks	Medium	High	Medium
Firmware logic tampering	Medium	Critical	High
I2C data interception	Low	Medium	Low
I2C data tampering	Low	Medium	Low
SPI data interception	Low	Medium	Low
SPI data tampering	Low	Medium	Low
DSRC data interception	Medium	Medium	Medium
DSRC data tampering	Low	High	Low
DSRC entity spoofing	Low	High	Low
BLE data interception	Medium	Medium	Medium
BLE data tampering	Low	High	Low
BLE entity spoofing	Low	High	Low
BLE DoS	Low	High	Low
CAN data interception	Medium	Medium	Medium
CAN data tampering	Medium	High	Medium
CAN entity spoofing	Medium	High	Medium
CAN bus DoS	Medium	High	Medium
ISO7816 data interception	Low	Medium	Low
ISO7816 data tampering	Low	Medium	Low
SE chip data leakage	Low	Medium	Low
SE chip data tampering	Low	High	Low

TSL = Threat Severity Level;ISL = Impact Severity Level;RL = Risk Level.

TABLE 11. Security objectives and test.

Security Objective	Test case	Pass
No PCB data leaks	PCB data leaks test	No
No UART information leakage	UART data leakage test	Yes
No UART interface privilege	UART privilege test	Yes
No SWD information leakage	SWD data leakage test	No
No SWD interface privilege	SWD privilege test	No
No Firmware logic leaks	Firmware logic leakage test	No
No Firmware data leakage	Firmware sensitive data test	No
No Firmware logic tampering	Firmware tampering test	No
I2C data encryption	I2C data interception test	No
I2C data integrity check	I2C data tampering test	No
SPI data encryption	SPI data interception test	No
SPI data integrity check	SPI data tampering test	No
DSRC data encryption	DSRC data interception test	Yes
DSRC data integrity check	DSRC data tampering test	Yes
DSRC entity authentication	DSRC spoofing test	Yes
BLE data encryption	BLE data interception test	Yes
BLE data integrity check	BLE data tampering test	Yes
BLE entity authentication	BLE spoofing test	Yes
No BLE DoS	BLE DoS test	Yes
CAN data encryption	CAN data interception test	No
CAN data integrity check	CAN data tampering test	No
CAN entity authentication	CAN spoofing test	No
No CAN bus DoS	CAN DoS test	Yes
ISO7816 data encryption	ISO7816 data disclosure test	No
ISO7816 data integrity check	ISO7816 data tampering test	No
No SE chip data leakage	SE chip data leakage test	Yes
SE chip data integrity check	SE chip data tampering test	Yes

"Pass" indicates whether the test passed. If the test passes, the OBU has met the corresponding security goals. Otherwise, OBU does not meet the corresponding security goals, and there are security vulnerabilities that can be actually exploited by malicious attackers

it has leaked information such as UART interface information and chip model information.

2) UART TEST

In uart test, we connect the test machine to the uart interface in the OBU with the USB to Transistor Transistor Logic (TTL) tool that supports uart communication protocol. The test machine runs a Serial debugging software tool such as minicom[49]. With the appropriate baud rate, we observe whether the OBU outputs startup information when power up. If there is startup information from uart, we test whether attacker can enter the bootloader shell by pressing any key. And we need to confirm whether it will automatically enter the system shell after the startup process is completed.

3) SWD TEST

The SWD test is similar to the uart test, but the tools used are different. In the SWD test, we connect the test machine to the swd interface with the JTAG debuger such as J-Link. With the help of openOCD [50], a debug software in the Linux platform, we can read data from the MCU and write data to the MCU, which can help to test whether the SWD interface introduce risk to the OBU. If the OBU permit hardware debug through SWD interface, there is security risk.

4) FIRMWARE TEST

After obtaining the firmware, we should analysis the firmware to confirm that the firmware will not disclose sensitive information such as passwords and URLs. With the help of reverse engineering, we also can ensure whether the attacker can easily figure out and tamper the functionality logic of the OBU.

5) I2C/SPI/ISO 7816 TEST

In the I2C, SPI, and ISO 7816 test, We use a digital logic analyzer to capture and analyze the electrical signals on the chip's I2C pins, SPI pins, and ISO 7816 pins to verify whether the data can disclose critical information.

6) DSRC TEST

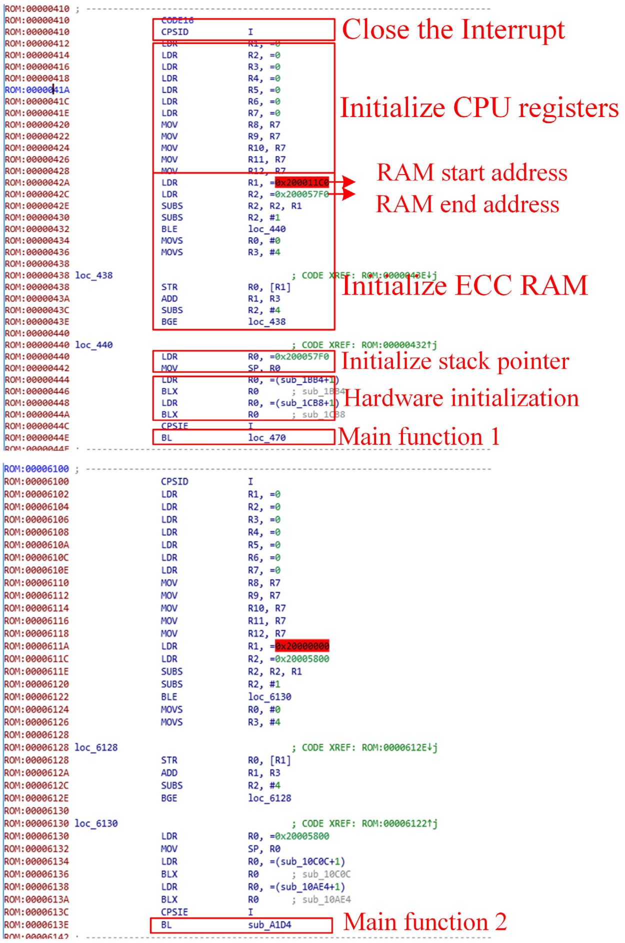
We capture the radio signal between OBU and RSU with the help of USRP B210[51] from Ettus, an soft defined radio(SDR) tool. After analyzing the signal based on the DSRC protocol specification, we can obtain the transmitted data. And we can simulate RSU and OBU by replaying data or sending fake data.

7) BLE TEST

With the Ubertooth one[52], a 2.4G Bluetooth data sniffer, we can capture the data between Bluetooth devices to verify whether the data is encrypted. And we can simulate Bluetooth data packets to test whether we can fake a Bluetooth device. Send a large number of junk data messages on the Bluetooth communication channel to test whether it can block the Bluetooth communication and cause a denial of service.

8) CAN TEST

The CAN bus protocol lacks security mechanisms such as encryption, integrity verification, and identity authentication.



```
ROM:00000410 ; ---------------------------------------------------------------
ROM:00000410                 CODE16
ROM:00000410                 CPSID           I
ROM:00000412                 LDR             R1, =0
ROM:00000414                 LDR             R2, =0
ROM:00000416                 LDR             R3, =0
ROM:00000418                 LDR             R4, =0
ROM:0000041A                 LDR             R5, =0
ROM:0000041C                 LDR             R6, =0
ROM:0000041E                 LDR             R7, =0
ROM:00000420                 MOV             R8, R7
ROM:00000422                 MOV             R9, R7
ROM:00000424                 MOV             R10, R7
ROM:00000426                 MOV             R11, R7
ROM:00000428                 MOV             R12, R7
ROM:0000042A                 LDR             R1, =0x200011C0
ROM:0000042C                 LDR             R2, =0x200057F0
ROM:0000042E                 SUBS            R2, R2, R1
ROM:00000430                 SUBS            R2, #1
ROM:00000432                 BLE             loc_440
ROM:00000434                 MOVS            R0, #0
ROM:00000436                 MOVS            R3, #4
ROM:00000438
ROM:00000438 loc_438                                 ; CODE XREF: ROM:0000043E↓j
ROM:00000438                 STR             R0, [R1]
ROM:0000043A                 ADD             R1, R3
ROM:0000043C                 SUBS            R2, #4
ROM:0000043E                 BGE             loc_438
ROM:00000440
ROM:00000440 loc_440                                 ; CODE XREF: ROM:00000432↑j
ROM:00000440                 LDR             R0, =0x200057F0
ROM:00000442                 MOV             SP, R0
ROM:00000444                 LDR             R0, =(sub_1BB4+1)
ROM:00000446                 BLX             R0      ; sub_1BB4
ROM:00000448                 LDR             R0, =(sub_1CB8+1)
ROM:0000044A                 BLX             R0      ; sub_1CB8
ROM:0000044C                 CPSIE           I
ROM:0000044E                 BL              loc_470
ROM:0000044F ; ---------------------------------------------------------------

ROM:00006100 ; ---------------------------------------------------------------
ROM:00006100                 CPSID           I
ROM:00006102                 LDR             R1, =0
ROM:00006104                 LDR             R2, =0
ROM:00006106                 LDR             R3, =0
ROM:00006108                 LDR             R4, =0
ROM:0000610A                 LDR             R5, =0
ROM:0000610C                 LDR             R6, =0
ROM:0000610E                 LDR             R7, =0
ROM:00006110                 MOV             R8, R7
ROM:00006112                 MOV             R9, R7
ROM:00006114                 MOV             R10, R7
ROM:00006116                 MOV             R11, R7
ROM:00006118                 MOV             R12, R7
ROM:0000611A                 LDR             R1, =0x20000000
ROM:0000611C                 LDR             R2, =0x20005800
ROM:0000611E                 SUBS            R2, R2, R1
ROM:00006120                 SUBS            R2, #1
ROM:00006122                 BLE             loc_6130
ROM:00006124                 MOVS            R0, #0
ROM:00006126                 MOVS            R3, #4
ROM:00006128
ROM:00006128 loc_6128                                ; CODE XREF: ROM:0000612E↓j
ROM:00006128                 STR             R0, [R1]
ROM:0000612A                 ADD             R1, R3
ROM:0000612C                 SUBS            R2, #4
ROM:0000612E                 BGE             loc_6128
ROM:00006130
ROM:00006130 loc_6130                                ; CODE XREF: ROM:00006122↑j
ROM:00006130                 LDR             R0, =0x20005800
ROM:00006132                 MOV             SP, R0
ROM:00006134                 LDR             R0, =(sub_10C0C+1)
ROM:00006136                 BLX             R0      ; sub_10C0C
ROM:00006138                 LDR             R0, =(sub_10AE4+1)
ROM:0000613A                 BLX             R0      ; sub_10AE4
ROM:0000613C                 CPSIE           I
ROM:0000613E                 BL              sub_A1D4
ROM:00006142 ; ---------------------------------------------------------------
```

FIGURE 5. The firmware extracted through the swd interface.

We can use the CAN transceiver to intervene in the CAN network to monitor and tamper with the bus transmission data and counterfeit the transmission node. Based on the arbitration mechanism of the CAN bus protocol, a higher priority data message can be sent to the CAN bus to preempt bus resources, resulting in a denial of service.

9) SE CHIP TEST

The SE chip provides cryptographic calculation and stores key keys. The side channel attack can be used to analyze whether the SE chip has leaked the key during operation.

In summary, as depicted in TABLE 11, the OBU under our evaluation has three kinds of security threats: information disclosure, spoofing, and elevation of privilege. Although the data leakage of I2C and SPI bus allows the attacker to obtain the corresponding data, it is very difficult to parse and utilize the data due to the high coupling between bus communication and running time window. In addition, although the BLE module will expose the subservices provided by the OBU, they are all public services and cannot be effectively utilized. For the DSRC communication, the communication data between the OBU and RSU can be intercepted by software radio tools. However, since the DSRC communication data is encrypted, the RSU will verify the OBU's validity, and it becomes very difficult to effectively utilize the intercepted data. Although the CAN bus have no security mechanism, it is difficult to attack the CAN bus because it requires physical access.

Different from information disclosure, the elevation of privilege in this OBU evaluation has a high level security risk.The target OBU exposed the SWD debugging interface. Unfortunately, the vendor did not turn off SWD debugging functionality, but instead opened debugging permissions. Attackers can use OpenOCD, ST-Link, and other tools to debug the target OBU obtaining the internal stored data, firmware, and malicious code execution privilege of the target object.

The target firmware data obtained in the experiment is shown in Fig.5. The firmware contains two pieces of main logic, which are used in different phases of the OBU's life cycle. The SWD debugging permission is very dangerous and can cause significant property damage. In addition, in this experiment, the OBU opens debugging permission by default, so attackers do not need to bypass SWD read and write protection mechanism, which greatly reduces the difficulty of attack.

VI. CONCLUSION

In order to better apply the security assessment to the IoV, we analyzed the security architecture of the IoV in detail, and proposed a security framework for the IoV. The security framework focuses on ten security aspects of smart vehicles and in-vehicle ECUs at four levels, which regulates the scope of security evaluation. In addition, we proposed CSEF that can be applied to in-vehicle ECUs to evaluate the cyber security of in-vehicle ECUs.The CSEF is designed based on the ISO/SAE 21434 standard and is optimized to have richer security assessment details, which can be better applied to the field of automotive security. The framework aims to solve five main problems:(1) identifies the assets of the evaluated objectives from ten aspects. (2) Identifies the cyber security threats faced by in-vehicle ECU through threat analysis that uses the attack tree method. (3) Rates the security risks faced by ECUs in the vehicle based on financial damage, personal injury, operational failure, and privacy disclosure. (4) Defines the security requirements of the identified asset (5) Confirms whether the evaluated target has security vulnerabilities that does not meet the cyber security target through the security test set. To show how to apply CSEF in the security evaluation of in-vehicle ECUs, we provide a use case of on-board OBU. The use case showed that the evaluation framework can be used to expose most threats and the potential vulnerabilities introduced by inappropriate design or coding. With the help of CSEF, OBU developers and other roles can have a deep grasp of the security status and potential security risks of the designed products. Security goals can be proposed to effectively protect the target products according to the threats and risks, and test cases can be used to verify whether the evaluated object meet the security goals. CSEF can plays an important role in guiding relevant personnel to conduct security evaluation activities.Based on the universal security framework proposed in section II, CSEF can be extended to other ECUs in the vehicle.

REFERENCES

[1] K. B. Kelarestaghi, M. Foruhandeh, K. Heaslip, and R. Gerdes, "Intelligent transportation system security: Impact-oriented risk assessment of in-vehicle networks," *IEEE Intell. Transp. Syst. Mag.*, vol. 13, no. 2, pp. 91–104, May 2021.

[2] T. T. Dandala, V. Krishnamurthy, and R. Alwan, "Internet of vehicles (IoV) for traffic management," in *Proc. Int. Conf. Comput., Commun. Signal Process. (ICCCSP)*, Jan. 2017, pp. 1–4.

[3] W. Yanbang, Y. Jing, and Y. Zhilou, "Auto-driving vehicle testing method, ECU and system," Inspur, China, Tech. Rep. CN108762226A, 2018.

[4] H. Pingguo, Y. Jingjing, and C. Xiao, "Security access control method for vehicle diagnosis system," BeiBen Trucks Group, China, Tech. Rep. CN103529823A, Jun. 2014.

[5] R. Q. Malik, K. N. Ramli, Z. H. Kareem, M. I. Habelalmatee, A. H. Abbas, and A. Alamoody, "An overview on V2P communication system: Architecture and application," in *Proc. 3rd Int. Conf. Eng. Technol. Appl. (IICETA)*, Sep. 2020, pp. 174–178.

[6] Y. Zhao, "Telematics: Safe and fun driving," *IEEE Intell. Syst.*, vol. 17, no. 1, pp. 10–14, Jan. 2002.

[7] Y. Li, J. Lin, and K. Xiong, "An adversarial attack defending system for securing in-vehicle networks," in *Proc. IEEE 18th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2021, pp. 1–6.

[8] J. Zhang, C. Shen, H. Su, M. T. Arafin, and G. Qu, "Voltage Over-scaling-based lightweight authentication for IoT security," *IEEE Trans. Comput.*, early access, Jan. 6, 2021, doi: 10.1109/TC.2021.3049543.

[9] J. Zhang and G. Qu, "Recent attacks and defenses on FPGA-based systems," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 12, no. 3, pp. 1–24, Sep. 2019.

[10] Y. Liu, H. Wang, M. Peng, J. Guan, and Y. Wang, "An incentive mechanism for privacy-preserving crowdsensing via deep reinforcement learning," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 8616–8631, May 2021.

[11] Y. Liu, T. Feng, M. Peng, J. Guan, and Y. Wang, "DREAM: Online control mechanisms for data aggregation error minimization in privacy-preserving crowdsensing," *IEEE Trans. Depend. Sec. Comput.*, early access, Jul. 24, 2020, doi: 10.1109/TDSC.2020.3011679.

[12] *Road Vehicles Cybersecurity Engineering*, document ISO/SAE DIS 21434, ISO, SAE, Draft International Standard, 2020.

[13] P. Shedden, A. Ahmad, W. Smith, H. Tscherning, and R. Scheepers, "Asset identification in information security risk assessment: A business practice approach," *Commun. Assoc. Inf. Syst.*, vol. 39, no. 1, p. 15, 2016.

[14] A. Y. Javaid, W. Sun, V. K. Devabhaktuni, and M. Alam, "Cyber security threat analysis and modeling of an unmanned aerial vehicle system," in *Proc. IEEE Conf. Technol. Homeland Secur. (HST)*, Nov. 2012, pp. 585–590.

[15] D. J. Landoll and D. Landoll, *The Security Risk Assessment Handbook: A Complete Guide for Performing Security Risk Assessments*. Boca Raton, FL, USA: CRC Press, 2005.

[16] B. Singh and S. Patil, "Single wire debug interface," in *Proc. IEEE 63rd Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2020, pp. 814–817.

[17] Z. Ma and C. Schmittner, "Threat modeling for automotive security analysis," *Adv. Sci. Technol. Lett.*, vol. 139, pp. 333–339, Nov. 2016.

[18] M. Hamad, M. Nolte, and V. Prevelakis, "Towards comprehensive threat modeling for vehicles," in *Proc. 1st Workshop Secur. Dependability Crit. Embedded Real-Time Syst.*, Nov. 2016, p. 31.

[19] A. Karahasanovic, P. Kleberger, and M. Almgren, "Adapting threat modeling methods for the automotive industry," in *Proc. 15th ESCAR Conf.*, 2017, pp. 1–10.

[20] M. Hamad and V. Prevelakis, "SAVTA: A hybrid vehicular threat model: Overview and case study," *Information*, vol. 11, no. 5, p. 273, May 2020.

[21] H.-B. Park, Y. Kim, J. Jeon, H. Moon, and S. Woo, "Practical methodology for in-vehicle CAN security evaluation," *J. Internet Serv. Inf. Secur.*, vol. 9, pp. 42–56, May 2019.

[22] H. Zhang, X. Meng, X. Zhang, and Z. Liu, "CANsec: A practical in-vehicle controller area network security evaluation tool," *Sensors*, vol. 20, no. 17, p. 4900, Aug. 2020.

[23] T. Huang, J. Zhou, and A. Bytes, "ATG: An attack traffic generation tool for security testing of in-vehicle CAN bus," in *Proc. 13th Int. Conf. Availability, Rel. Secur.*, Aug. 2018, pp. 1–6.

[24] R. Marinescu, M. Saadatmand, A. Bucaioni, C. Seceleanu, and P. Pettersson, "A model-based testing framework for automotive embedded systems," in *Proc. 40th EUROMICRO Conf. Softw. Eng. Adv. Appl.*, Aug. 2014, pp. 38–47.

[25] M. Ring, T. Rensen, and R. Kriesten, "Evaluation of vehicle diagnostics security–implementation of a reproducible security access," in *Proc. SECURWARE*, Nov. 2014, p. 213.

[26] M. Cheah, S. A. Shaikh, J. Bryans, and P. Wooderson, "Building an automotive security assurance case using systematic security evaluations," *Comput. Secur.*, vol. 77, pp. 360–379, Aug. 2018.

[27] M. N. Aladwan, F. M. Awaysheh, S. Alawadi, M. Alazab, T. F. Pena, and J. C. Cabaleiro, "TrustE-VC: Trustworthy evaluation framework for industrial connected vehicles in the cloud," *IEEE Trans. Ind. Informat.*, vol. 16, no. 9, pp. 6203–6213, Sep. 2020.

[28] *Vehicle Electrical System Security Committee*, document SAE J3061, Cybersecurity Guidebook for Cyber-Physical Automotive Systems, 2016.

[29] O. Henniger, A. Ruddle, H. Seudié, B. Weyl, M. Wolf, and T. Wollinger, "Securing vehicular on-board it systems: The evita project," in *Proc. VDI/VW Automot. Secur. Conf.*, 2009, p. 41.

[30] A. Lautenbach and M. Islam, "HEAVENS–HEAling Vulnerabilities to EN-hance software security and safety," The HEAVENS Consortium (Borås SE), Vinnova, Sweden, Tech. Rep. 2012-04625, 2016.

[31] Z. Yang and Z. Zhang, "The study on resolutions of STRIDE threat model," in *Proc. 1st IEEE Int. Symp. Inf. Technol. Appl. Educ.*, Nov. 2007, pp. 271–273.

[32] C. Alberts, A. Dorofee, J. Stevens, and C. Woody, *Introduction to the OCTAVE Approach*. Pittsburgh, PA, USA: Carnegie-Mellon Univ. Software Engineering Institute, 2003.

[33] G. Macher, E. Armengaud, E. Brenner, and C. Kreiner, "A review of threat analysis and risk assessment methods in the automotive context," in *Proc. Int. Conf. Comput. Saf., Rel., Secur.* Cham, Switzerland: Springer, 2016, pp. 130–141.

[34] P. Waszecki, P. Mundhenk, S. Steinhorst, M. Lukasiewycz, R. Karri, and S. Chakraborty, "Automotive electrical and electronic architecture security via distributed in-vehicle traffic monitoring," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 36, no. 11, pp. 1790–1803, Nov. 2017.

[35] Z. Lu, G. Qu, and Z. Liu, "A survey on recent advances in vehicular network security, trust, and privacy," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 2, pp. 760–776, Feb. 2019.

[36] M. Steger, A. Dorri, S. Kanhere, K. Romer, R. Jurdak, and M. Karner, "Secure wireless automotive software updates using blockchains: A proof of concept," in *Advanced Microsystems for Automotive Applications 2017: Smart Systems Transforming the Automobile* (Lecture Notes in Mobility), G. Meyer, B. Müller, and C. Zachaus, Eds. Cham, Switzerland: Springer, 2018, pp. 137–149.

[37] Y. Zheng, N. Shokouhi, and N. Thomsen, "Towards developing a distraction-reduced hands-off interactive driving experience using portable smart devices," in *Proc. SAE Tech. Paper*, Apr. 2016, pp. 1–7.

[38] S. Ray, W. Chen, J. Bhadra, and M. A. Al Faruque, "Extensibility in automotive security: Current practice and challenges: Invited," in *Proc. 54th Annu. Design Autom. Conf.*, Jun. 2017, pp. 1–6.

[39] L. Pu, Z. Liu, Z. Meng, X. Yang, K. Zhu, and L. Zhang, "Implementing on-board diagnostic and GPS on VANET to safe the vehicle," in *Proc. Int. Conf. Connected Vehicles Expo (ICCVE)*, Oct. 2015, pp. 13–18.

[40] 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 *Proc. USENIX Secur. Symp.*, vol. 4. San Francisco, CA, USA, 2011, pp. 447–462.

[41] Z. Lu, Q. Wang, X. Chen, G. Qu, Y. Lyu, and Z. Liu, "LEAP: A lightweight encryption and authentication protocol for in-vehicle communications," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 1158–1164.

[42] M. Wolf, A. Weimerskirch, and C. Paar, "Security in automotive bus systems," in *Proc. Workshop Embedded Secur. Cars.*, 2004, pp. 1–13.

[43] P. S. Murvay and B. Groza, "Practical security exploits of the FlexRay in-vehicle communication protocol," in *Proc. Int. Conf. Risks Secur. Internet Syst.* Cham, Switzerland: Springer, 2018, pp. 172–187.

[44] J. M. Ernst and A. J. Michaels, "LIN bus security analysis," in *Proc. IECON 44th Annu. Conf. IEEE Ind. Electron. Soc.*, Oct. 2018, pp. 2085–2090.

[45] T. Huang, J. Zhou, Y. Wang, and A. Cheng, "On the security of in-vehicle hybrid network: Status and challenges," in *Proc. Int. Conf. Inf. Secur. Pract. Exper.* Cham, Switzerland: Springer, 2017, pp. 621–637.

[46] K. Ren, Q. Wang, C. Wang, Z. Qin, and X. Lin, "The security of autonomous driving: Threats, defenses, and future directions," *Proc. IEEE*, vol. 108, no. 2, pp. 357–372, Feb. 2020.

[47] K. Scarfone and P. Mell, "An analysis of CVSS version 2 vulnerability scoring," in *Proc. 3rd Int. Symp. Empirical Softw. Eng. Meas.*, Oct. 2009, pp. 516–525.

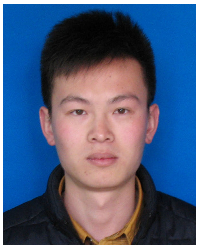
[48] *Road Vehicles Functional Safety*, document ISO 26262-2011, International Organization for Standardization, 2011.

[49] S. H. Jayantilal, "Interfacing of AT command based HC-05 serial Bluetooth module with minicom in Linux," *Int. J. Sci. Res. Develop.*, vol. 2, no. 3, pp. 329–332, 2014.

[50] H. Högl and D. Rath, "Open on-chip debugger–openocd," Fakultat fur Informatik, Augsburg, Germany, Tech. Rep. 20070115, 2006.

[51] L. Liu, B. Wu, and W. Shi, "A comparison of communication mechanisms in vehicular edge computing," in *Proc. 3rd USENIX Workshop Hot Topics Edge Comput. (HotEdge)*, 2020, pp. 1–12.

[52] S. Sarkar, J. Liu, and E. Jovanov, "A robust algorithm for sniffing BLE long-lived connections in real-time," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.



HAICHUN ZHANG received the B.E. degree in electronic science and technology from the Huazhong University of Science and Technology, Wuhan, China, in 2016, where he is currently pursuing the Ph.D. degree with the School of Optical and Electronic Information. His research interest includes security of VANET.

YUQIAN PAN received the M.E. degree in software engineering from Xidian University, Xian, China, in 2017. She is currently pursuing the Ph.D. degree with the School of Cyber Science and Engineering, Huazhong University of Science and Technology, Wuhan, China. Her research interests include NAND flash memory and embedded system security.

ZHAOJUN LU received the B.S. degree in electronic science and technology and the Ph.D. degree in microelectronics and solid state electronics from the Huazhong University of Science and Technology, Wuhan, China, in 2013 and 2018, respectively. He was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Maryland, College Park, in 2018. He is currently a Lecturer with the School of Cyber Science and Engineering, Huazhong University of Science and Technology. His research interests include embedded system security, very large-scale integration design, and vehicular *ad-hoc* network security.

JIE WANG received the Ph.D. degree from Loughborough University, U.K., in 2012. In 2021, he was appointed as the Director of the Shenzhen Longhua District Network Security Research Centre. He is currently a Co-Founder of Shenzhen Kaiyuan Internet Security Company Ltd., China, specializing in secure software development lifecycle (S-SDLC), software supply chain security, application security testing, network security assurance, and IoV security. He has significant experience of using software security techniques and software assurance method in a number of research projects. He has published over 20 papers in journals and conference papers, and three software security books. He has been serving as a reviewer for several international journals and conferences.

ZHENGLIN LIU received the Ph.D. degree from the Department of Electronic Science and Technology, Huazhong University of Science and Technology, Wuhan, China, in 2001. He is currently a Professor with the School of Optical and Electronic Information, Huazhong University of Science and Technology. His research interests include embedded system security and very large-scale integration design.