Survey on Testing of Autonomous Driving Systems

Wenjie Chen
Software Engineering Institute
Shanghai Key Laboratory of Computer Software Testing & Evaluation
Shanghai Development Center of Computer Software Technology
Shanghai, China
cwj@sscenter.sh.cn

Zeyu Ma
Software Engineering Institute
Shanghai Key Laboratory of Computer Software Testing & Evaluation
Shanghai Development Center of Computer Software Technology
Shanghai, China
mzy@sscenter.sh.cn

Chao Wang
Software Engineering Institute
Shanghai Key Laboratory of Computer Software Testing & Evaluation
Shanghai Development Center of Computer Software Technology
Shanghai, China
wc@sscenter.sh.cn

Mingang Chen
Software Engineering Institute
Shanghai Key Laboratory of Computer Software Testing & Evaluation
Shanghai Development Center of Computer Software Technology
Shanghai, China
cmg@sscenter.sh.cn

Lizhi Cai
Software Engineering Institute
Shanghai Key Laboratory of Computer Software Testing & Evaluation
Shanghai Development Center of Computer Software Technology
Shanghai, China
clz@sscenter.sh.cn

***Abstract*—Before autonomous driving vehicles are commercialized, they need to undergo a series of rigorous tests. This paper first proposes the general process of autonomous driving system testing, and then summarizes the research progress of autonomous driving vehicle testing from the model, simulation, network, and vehicle levels, and analyzes the characteristics of various testing technologies. Finally, this paper gives suggestions on the development of autonomous driving testing technology.**

***Keywords*—autonomous driving, adversarial examples, simulation testing, cybersecurity**

I. Introduction

In recent years, the pervasive and tremendous breakthroughs of deep neural networks (DNNs) promote the development of autonomous driving technology. However, sometimes it is difficult to guarantee the reliability of the autonomous driving system. Banerjee et al. investigated the causes of 5,328 failures from autonomous driving systems of 12 AV manufacturers[1]. As high as 64% of the failures were found to be caused by the bugs in the machine learning system. The industry and academia have strived to improve the safety of the autonomous driving system, they have done a lot of research on testing autonomous driving system.

This paper summarizes the current research of autonomous driving testing technology from four aspects: model testing, simulation testing, cybersecurity testing, and field testing, combined with the system and software quality model defined in the ISO/IEC 25010 standard. Research on model testing is mainly focused on the adversarial attack. At present, there are already some works that can generate adversarial examples for street signs and billboards, leading to misclassification of the autonomous driving system. In terms of simulation testing, many companies have developed simulation platforms and scenario databases. The research of cybersecurity testing focuses on the vulnerability detection and data protection of the terminals and interfaces in the vehicle network. Field testing mainly focuses on the construction of a proving ground for autonomous driving.

II. Testing process of autonomous driving systems

Autonomous driving system testing is an important procedure in the development process of autonomous vehicles, it mainly focuses on testing the functional suitability, reliability and security defined in the ISO/IEC 25010 standard. The testing process of the autonomous driving system includes model testing, simulation testing, cybersecurity testing, and field testing, as shown in Fig. 1. Model testing, simulation testing and field testing mainly test functional suitability and reliability of the autonomous driving system, and cybersecurity testing mainly tests security of the autonomous driving system.

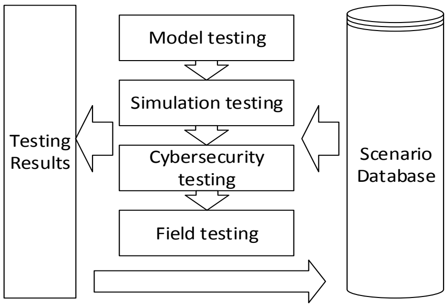

Fig. 1. Testing process of autonomous driving systems

The testing process of the autonomous driving system is a continuous cycle. During model testing, the perception model is tested using test data sets and adversarial examples. Then the simulation platform is used to test the autonomous driving system and each control unit. Cybersecurity testing mainly focuses on the network security of the autonomous driving system and its interface. Field testing is to test the autonomous driving vehicle in a proving-ground. During the test, a scenario database composed of environment, traffic participants, sensors, and other data is generated. The results of tests will also be used to update the scenario database, such as adding some newly discovered dangerous scenes.

This work was funded by Ministry of Science and Technology of the People's Republic of China Program (2018YFB1403405) and Science and Technology Commission of Shanghai Municipality Program (21511101204).

III. TESTING TECHNOLOGY FOR AUTONOMOUS DRIVING SYSTEMS

A. *Testing for autonomous driving models*

Testing of autonomous driving system models is performed by testing datasets, adversarial examples, and other methods against perception models. Most current perception models are implemented by DNNs, but due to the large dimensions of input data for models, DNNs are vulnerable to adversarial examples, which is a great concern for the prospect of autonomous driving. Many studies have shown that the object detection model used for autonomous driving can be easily deceived by the existing adversarial attack methods, thereby causing errors in autonomous driving decision-making.Current adversarial attack methods in the field of autonomous driving are mainly divided into attacks on the object detection model and attacks on LIDAR.

In 2017, a team of researchers at Columbia University proposed DeepXplore[2], an automated DNN testing system that uses several existing test inputs as seeds, then modifies the test inputs in a continuous loop iteratively through a gradient ascent method to maximize the difference between the output of the model under test and the output of other similar models. Eventually, this system is able to generate a new set of test data, which can trigger errors in the judgment of the model under test. Experiments on the Udacity challenge dataset showed that the system is capable of making the DNN-based autonomous driving systems' prediction of the car's steering angle incorrect by changing the lightness and darkness of the input images, adding noise, and so on. The same group followed up with DeepTest[3], removing the requirement that DeepXplore must provide multiple DNNs with similar functions. DeepTest also makes some changes for the generation of test data for autonomous driving systems, resulting in a more efficient system for generating data for extreme scenarios.

DeepXplore and DeepTest can generate a large number of adversarial examples, but these are very different from real scenes, and many extreme scenes such as rainy and foggy days are tough to generate by simple image transformation. DeepRoad[4] uses Generative Adversarial Networks (GAN) to generate more realistic transformed images of rainy and snowy days, compare the steering angle prediction results of the generated images with results of the original images, which can find the scenes that cause errors of DNN in the autonomous driving.

Zhou et al. proposed DeepBillboard, which generates real-world adversarial examples of billboards that could trigger steering errors in autonomous driving systems[5]. It demonstrates the possibility of generating real physical-world adversarial examples for actual autonomous driving systems. Kevin et al. perform a physical-world robustness attack on parking road signs (by sticking adversarial patches at specified locations on the road signs) to misclassify stop signs as other signs of the specified category with 100% probability, as in Fig. 2[6]. The authors proposed the RP2 algorithm: Firstly, build a model to quantify the target object's physical changes (including changes in distance, angle, illumination, etc., and transformations such as random cropping of images and changes in luminance), then construct a mask of the target object to discover "vulnerable regions" on which the attack is achieved by masking.

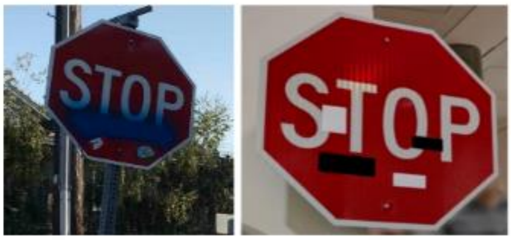

Fig. 2. Attacking stop signs with stickers[6]

Aishan Liu et al. proposed PS-GAN, which can generate adversarial patches, innovatively combining GAN network and attention mechanism[7]. PS-GAN can capture the sensitivity of spatial distribution to obtain the optimal attack locations in order to enhance the attack capability of the patches and also ensure a reasonable appearance, as in Fig. 3. However, this method does not guarantee that the patch is always on the target object.

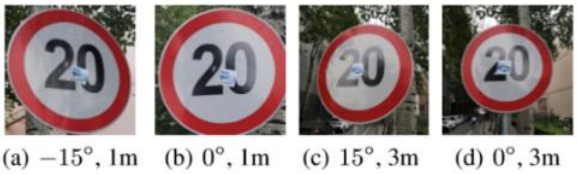
(a) −15°, 1m (b) 0°, 1m (c) 15°, 3m (d) 0°, 3m

Fig. 3. The adversarial patches generated by PS-GAN[7]

Zelun Konget al. proposed PhysGAN, an algorithm that generates adversarial examples with physical world resilience for autonomous driving systems in a continuous manner[8]. Unlike PS-GAN, the input to PhysGAN is a given scene (e.g. the content of a billboard, as in Fig. 4), so the generated adversarial examples are more realistic. What differs from other adversarial attack methods which aim at detection classifiers is that PhysGAN attacks autonomous driving navigation systems, which are regression models. So the authors use the mean squared error and the maximum error of the angle of autonomous driving navigation evaluate the result.

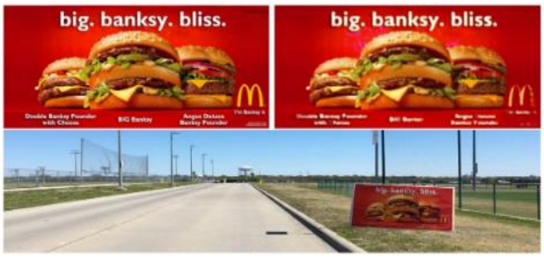

Fig. 4. Attacking autonomous driving navigation systems via PhysGAN on the McDonald's ads[8]

The above studies aim at vision-based autonomous driving systems, but LiDAR-Adv[9], a joint study of University of Michigan, UIUC, and Baidu, breaks through LiDAR systems. The researchers connected perturbations of a 3D target to a LiDAR scan (or point cloud) by modeling a differentiable LiDAR renderer. They then used the differentiable proxy function to produce 3D feature aggregations and designed different losses to ensure that 3D adversarial examples were smooth. In a specific experiment, the researchers compared a normal box with a 3D printed adversarial example on the Apollo Autopilot system. They found that the LIDAR-equipped car did not detect the target until it approached the adversarial example. In contrast, the car detected the normal box at a long distance.

The University of Toronto, Princeton University in conjunction with Uber also proposed a generic 3D adversarial object generator to fool LIDAR detectors[10]. In particular, the authors placed a generated pseudo-object on top of any target vehicle to completely hide the vehicle, resulting in an 80% success rate of not being detected by LIDAR detectors, as shown in Fig. 5.

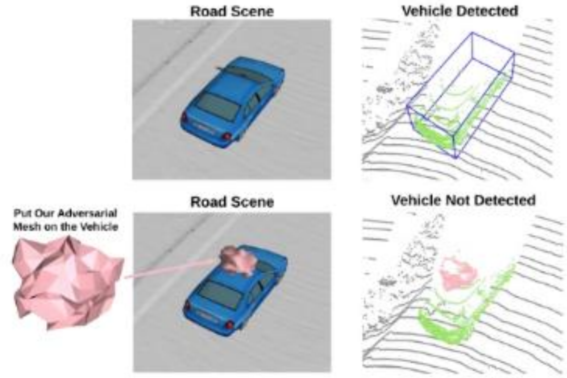

Fig. 5. Placing an antagonistic object on the target vehicle made the vehicle "invisible" to LIDAR[10]

B. *Simulation testing for autonomous driving systems*

The simulation test is to test the autonomous driving system and each control unit through the simulation platform. Currently, about 90% of autonomous driving tests are completed through simulation platforms. Both Scenario database and simulation platform are required for simulation testing.

1) Creating Scenario Database

A scenario is the overall description of the autonomous vehicle and environment components over a period, which is abundant and complex. The scenario database is a database composed of a series of test scenarios that meet certain test requirements. The construction of the scenario database is generally divided into four steps[11]: data collection, data cleaning, information labeling, and scenario clustering. The source of scenario data mainly includes real-world driving data and simulation data synthesized from real-world driving data. The test results of the simulation test will also be used to update the scenario database as required.

At present, many auto companies and autonomous driving solution providers have established their own scenario databases. Waymo collects simulation scenarios based on field tests. After testing autonomous vehicles on public roads and proving ground, Waymo accumulates thousands of scenario data, creates virtual scenarios based on these data, and produces more scenarios by modifying scenario parameters. Waymo has released a database of more than one thousand scenarios. Automotive Data of China Co., Ltd. has initially built a simulation testing scenario database that includes nearly 500,000 kilometers of driving data and traffic rules, covering important cities such as Beijing, Tianjin, and Shanghai. China Automotive Engineering Research Institute Co., Ltd. released the "China Typical Scenario Database V2.0" in 2019, including hundreds of standard traffic rules scenarios, 3,000 empirical scenarios, 50,000 functional scenarios, and 150 accident scenarios. Companies such as Tencent and Baidu have also released their own databases of autonomous driving scenarios.

2) Simulation platform

The autonomous driving simulation platform is a system that tests autonomous driving functions by simulating traffic scenes, vehicle movements, and sensor signals. Its main functions include restoring static scenes and dynamic scenes, camera and radar simulation, and vehicle dynamics simulation. At present, many companies and institutions have developed their autonomous driving simulation platforms.

Carla[12] is an open source free autopilot simulator based on unreal engine. It supports flexible configuration of sensors, environmental states, dynamic and static traffic participants and maps, and can control simulated vehicles through Python or C language API. Autoware[13] is an open source software for automatic driving technology research. It includes four modules: localization, detection, prediction and planning, and control. It supports path planning, traffic signal detection, lane detection, virtual reality and other functions. PreScan[14] is a widely used vehicle driving simulation software product of Siemens. It supports the simulation of multiple functions such as camera, radar, LiDAR, GPS, and vehicle-to-vehicle communication, and can simulate simple traffic scenarios. SiVIC[15] is similar to PreScan, but it can provide more realistic and complete sensor models. Google developed the simulation platform Carcraft, based on the scenario data collected by Waymo, combined with high-precision map information, to realistically simulate the real traffic environment. NVIDIA released the cloud-based NVIDIA Drive Constellation simulation system in 2018, which can generate realistic data, create various test environments, simulate various weather conditions such as rain and snow, simulate different roads and terrains, and simulate dazzling light during the day and limited vision at night. Microsoft open-sourced the cross-platform Unreal Engine simulator AirSim in 2017, which supports simulations of drones and autonomous driving. It can create a highly realistic traffic environment and simulate vehicles and sensors. LG Silicon Valley Lab released the open-source autonomous driving simulator LGSVL Simulator in early 2019, which supports sensor simulation and editable maps, vehicles, weather, traffic flow, pedestrians, etc.

Tencent released the autonomous driving simulation platform TAD Sim in 2018, which combines professional game engines, industrial level vehicle dynamics models, integrated virtual and real traffic flow and other technologies. Baidu's self-developed autonomous driving simulation system AADS includes a data-driven traffic flow simulation framework and a scene picture synthesis framework based on image rendering. Researchers of Jilin University independently developed the PanoSim, a virtual autonomous driving test platform[16]. They analyzed the driving habits of drivers based on the platform and proposed ADAS control strategies that consider different driving habits.

C. *Cybersecurity testing for autonomous driving systems*

The autonomous driving system is deployed in the Intelligent Connected Vehicle(ICV), which can not only assist or replace drivers to control a vehicle through advanced onboard sensors, controllers, actuators, and other devices but also integrate modern communication and network technologies to realize Vehicle-to-everything(V2X), complex environmental perception, decision making and cooperative control of multi-vehicle, etc. The vehicle network technology provides the possibility that crackers hack into the autonomous driving system or ICV system. Crackers can slow down, stop the engine, brake, or do other malicious operations to the vehicles by hacking into the cloud account. They may also hack into the mobile application to take over the control of vehicle such as unlocking and starting engine remotely; implant malicious files into the in-vehicle networking (IVN) through USB storage media to control the vehicle, etc. Yoshiyasu Takefuji listed several cybersecurity incidents, such as white hat hackers stopped the engine of a car on the highway through a remote man-in-the-middle attack in 2015; Ford car's parking assistance module could be forced to intervene the control of steering wheel through the CAN command named "0x0081"[17].

The frequent occurrence of security accidents has accelerated the in-depth examination of cybersecurity issues of ICV. The security of autonomous driving systems requires the support of layered distributed technology, including the security of in-vehicle module systems, network-side interaction and cloud information processing. The security of each layer guarantees the security of ICV and connected vehicle system, and ensures the safety of the autonomous driving system ultimately.

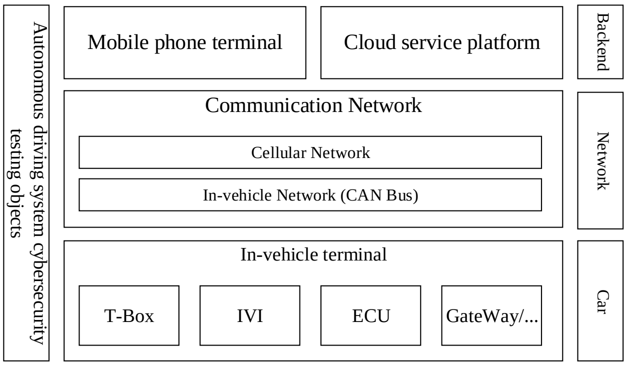

Fig. 6. Objects of autonomous driving system cybersecurity testing

According to the structure of vehicle security operation center (VSOC) defined in the white book, "Setting the Standard for Connected Cars' Cybersecurity", the objects of testing autonomous driving system cybersecurity can be classified into cloud service platform, mobile phone terminal, in-vehicle terminal, and communication network (Fig. 6). Specific testing points for these objects include the following:

a) Cloud service platform testing concerns more about the traditional Web vulnerability and the transmission security between the cloud and the other two terminals;

b) Mobile phone terminal has become the standard configuration of ICV so that testers should evaluate whether the communication key and communication protocol between the mobile and vehicle terminal can be cracked by technologies, analyze the communication protocol, and use it to forge malicious requests for vehicle control;

c) In-vehicle terminal testing objects include In-Vehicle Infotainment (IVI), Telematics-Box (T-Box), sensors, external interfaces, and other components. Generally, IVI, T-Box and other components contain operating systems, in-vehicle APPs, and a large number of third-party libraries;

d) Communication networks are tested for authentication, transmission encryption, and protocol security.

To test the four types of cybersecurity testing objects for autonomous driving systems, several researchers not only applied existing traditional effective and valuable cybersecurity testing methods to cloud service platforms and mobile phone terminals, but also proposed testing methods for the items of autonomous driving systems that require special attention. Wu Lingyun et al. proposed a random forest-based CAN bus message anomaly detection method model, which effectively detects anomalous data on ICV and improves vehicle operation security[18]. Mazloom S et al. created a malicious demo application, loaded on a mobile terminal. It uses the open MirrorLink interface on an IVI to connect to a mobile phone and discovers a heap overflow vulnerability that allows an attacker to obtain the control flow of a privileged process executing on the IVI[19]. It will further allow malicious attacks on the controller of the autonomous driving system. Testers can use the same principle to test whether the dangerous interface of IVI has been closed using the relevant Payload.

D. Field testing for autonomous driving systems

Field testing is to test the autonomous vehicle on the real-world road, typically in a proving ground. The autonomous vehicle must be tested in many scenarios and environments in a limited field.

The United States and the European Union have built some proving ground for autonomous driving testing. The Smart Road was built in Virginia by renovating part of the highway, which is 2.2 miles long and can simulate rainy and foggy weather by spraying water mist. The Mcity proving ground in Michigan contains pavements of different materials, and is equipped with abundant traffic signs, signal lights, tunnels, and other traffic elements. Google rents the Castle Air Force Base in California to test its autonomous vehicles. There are various streets, highways, traffic lights, traffic roundabouts, etc. inside the proving ground, as well as rainy weather simulators. The AstaZero Proving Ground in Sweden includes urban roads, highways, multi-lane parallel road, roundabouts and intersections, and has become a research and development platform for autonomous driving safety technology. Shanghai, Hangzhou, Wuhan, Shenzhen and some other cities in China also plan to build proving ground for autonomous driving testing.

IV. Conclusion

Testing and verifying the safety of autonomous driving systems is an important prerequisite for running autonomous vehicles on the road. The difficulty of testing is increasing as the level of autonomous driving increases. Currently, the industry and academia have carried out a lot of research on testing autonomous driving systems and developed corresponding tools and technologies to test autonomous driving models, systems, networks, and vehicles. A lot of achievements have been made in adversarial example generation, simulation platform development, network vulnerability analysis and proving ground construction. However, there are still some unresolved problems in autonomous driving system testing. For example, the cooperation mechanism of developing the scenario database is still inefficient, and the standard of autonomous driving system evaluation is not established yet. In the future, it is necessary to establish a set of testing standards and tool chains for autonomous driving systems to provide forceful supports for the development and implementation of autonomous driving technology.

REFERENCES

[1] S. S. Banerjee, S. Jha, J. Cyriac, Z. T. Kalbarczyk, and R. K.Iyer. "Hands off the wheel in autonomous vehicles?: A systems perspective on over a million miles of field data," In 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2018, pp. 586–597.

[2] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. "DeepXplore: Automated whitebox testing of deep learning systems," In Proceedings of the 26th Symposium on Operating Systems Principles,ACM, 2017, pp. 1–18.

[3] Tian, Y., Pei, K., Jana, S., and Ray, B. "Deeptest: Automated testing of deep-neural-network-driven autonomous cars," Proceedings of the 40th international conference on software engineering. 2018, pp. 303-314.

[4] Zhang, M., Zhang, Y., Zhang, L., Liu, C., and Khurshid, S. "DeepRoad: GAN-based metamorphic testing and input validation framework for autonomous driving systems," 2018 33rd IEEE/ACM International Conference on Automated Software Engineering (ASE). IEEE, 2018, pp. 132-142.

[5] Zhou H, Li W, Kong Z, et al. "Deepbillboard: Systematic physical-world testing of autonomous driving systems," 2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE). IEEE, 2020, pp. 347-358.

[6] K. Eykholt et al. "Robust physical-world attacks on deep learning visual classification," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 1625-1634.

[7] Liu, A., Liu, X., Fan, J., Ma, Y., Zhang, A., Xie, H., and Tao, D. "Perceptual-Sensitive GAN for generating adversarial patches," Proceedings of the AAAI Conference on Artificial Intelligence, 2019, Vol. 33, No. 01, pp. 1028-1035.

[8] Z. Kong, J. Guo, A. Li and C. Liu. "PhysGAN: Generating physical-world-resilient adversarial examples for autonomous driving," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 14242-14251.

[9] Yulong Cao, Chaowei Xiao, Dawei Yang, Jing Fang, Ruigang Yang, Mingyan Liu, and Bo Li. "Adversarial objects against LiDAR-Based autonomous driving systems," arXiv:1907.05418, 2019.

[10] Tu J, Ren M, Manivasagam S, et al. "Physically realizable adversarial examples for lidar object detection," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020, pp. 13716-13725.

[11] Zhang, Z., Shi, J., Guo, K., and Wang, J. "Research on construction method and application of autonomous driving test scenario database," CICTP 2020. 2020, pp. 311-323.

[12] Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A. and Koltun, V.. "CARLA: An open urban driving simulator," Proceedings of the 1st Annual Conference on Robot Learning, in PMLR, 2017, pp. 1-16.

[13] S. Kato, S. Tokunaga, Y. Maruyama, S. Maeda, M. Hirabayashi, Y. Kitsukawa, A. Monrroy, T. Ando, Y. Fujii, and T. Azumi, "Autoware on board: Enabling autonomous vehicles with embedded systems," In Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS2018), 2018, pp. 287-296.

[14] Gietelink O J. "Design and validation of advanced driver assistance systems," 2007.

[15] Gruyer D, Pechberti S, Glaser S. "Development of full speed range ACC with SiVIC, a virtual platform for ADAS prototyping, test and evaluation," 2013 IEEE Intelligent Vehicles Symposium (IV). 2013, pp. 100-105.

[16] Shanshan Wang. "PanoSim: A new generation of advanced automobile intelligent driving simulation system," Review of Science and Technology, 2015, no. 10, pp. 68-71.

[17] Y. Takefuji, "Connected vehicle security vulnerabilities [Commentary]," in IEEE Technology and Society Magazine, 2018, vol. 37, no. 1, pp. 15-18.

[18] Lingyun W U , Guihe Q , He Y U. "Anomaly detection method for in-vehicle CAN bus based on random forest," Journal of Jilin University(Science Edition), 2018, vol. 56, no. 3, pp. 663-668.

[19] Mazloom, S., Rezaeirad, M., Hunter, A., and McCoy, D. "A security analysis of an in-vehicle infotainment and app platform," In 10th {USENIX} Workshop on Offensive Technologies ({WOOT} 16), 2016.