A Practical Testing Framework for Car CAN Bus Assessment

Abstract---* *Over the past tens of years, we have witnessed several new car safety assessment programmes implementing car safety testings. These assessment systems provide car consumers with the vehicle safety information and push automakers to improve the level of vehicle crash safety. However, none of them has involved into car security assessment testing so far. In this paper, the author has implemented a practical CAN Bus security assessment framework which has successfully ranked tens of car models. The assessment framework includes two testing modes, the quick and deep tests, for consumers and OEMs respectively. Quick test enables consumers to spend a short period of time to make purchase decisions with the knowledge of how well the car model would protect themselves from car hackings. OEMs can leverage deep test to find ECU software security vulnerabilities. Furthermore, our results matched the ranking findings conducted by top researchers who hacked Jeep cars.

Index Terms— Security Assessment, car cybersecurity, security testing, score ranking, car hacking.

I. INTRODUCTION

For many car consumers, the safety is one of the key factors when they are shopping for new cars. In Feb 2010, Insurance Institute for Highway Safety (IIHS) conducted a survey on vehicle safety [1] and found Safety was the second most important factor for consumers to make purchase decisions on cars. 67% respondents said they would consider the safety rating when purchasing a vehicle.

A. New Car Safety Assessment

Currently there are some car assessment systems on vehicle safety testing. The National Highway Traffic Safety Administration (NHTSA)'s New Car Assessment Programme (NCAP) [2] and IIHS [3] are the most prominent. NCAP was launched in 1978 and it was the first program to provide automotive consumers the information of vehicle safety information. These information help consumers to make purchase decisions with the knowledge car crash rankings. NCAP uses five-star rating schema, and went through a revision in 2010 to enhance certain testing methods. Furthermore, NCAP has spawned various programs in other continents and areas [4], as shown in the following table.

Table 1. NCAP programmes across the continents.

Continent	Program Name	Established
Asia	China New Car Assessment Program (C-NCAP)	2006
	Japan New Car Assessment Program (JNCAP)	1991
	Korean New Car Assessment Program (KNCAP)	1999
	New Car Assessment Program for Southeast Asian Countries (ASEAN NCAP)	2011
Australia	Australasian New Car Assessment Program (ANCAP)	1992
Europe	European New Car Assessment Program (Euro-NCAP)	1997
South America	Latin American & Caribbean New Car Assessment Program (Latin NCAP)	2010

Insurance Institute for Highway Safety (IIHS) started a similar program in 1990s. IIHS is a safety-research group sponsored by auto insurers. It uses a ranked system, such as *good, acceptable, marginal* and *pool* rating categories, as well as the list of *Top Safety Picks*. Such rating schemas aim to help customers make purchase decisions based on the safety collision ratings and also encourage OEMs to design their vehicles with higher standard levels of safety protection.

J.D. Power is an independent consumer survey based on the opinions of surveyed consumers who have owned the cars being rated. J.D. Power generates a rating level of *five, four, three,* or *two* [5].

B. Car Security Assessment

However, none of the above programmes has involved into car security assessment testing so far. The vehicle cybersecurity attacks over the last two years have demonstrated that security vulnerabilities really exist.
The threat landscape is changing and the connected-car now becomes a new cyber attack target.

There is increasing awareness of cyber security requirements in the industry. For example, SAE has developed and released Cybersecurity standards and guidelines for automotive cybersecurity[6]. TEVEES18A1[7] was established as a task force with the purpose of developing appropriate SAE documentation for cybersecurity assurance testing and evaluation. Cybersecurity test focuses on finding and identifying unwanted weaknesses or vulnerabilities hiding inside the vehicle software. However, currently no method to quantify the car security testing results exists. The missing of practical quantified scoring system becomes the barrier for evaluating security performances of car models. Furthermore, the lack of suitable testing tools results in another challenge to implement a security assessment framework. The drawbacks of some available tools are that they are not stable and not working well under the heavy-traffic testing scenarios, thus not meeting the rigorous control and validation requirements of auto industry software practices.

In this paper, the author has implemented a practical CAN Bus security assessment framework. Tens of car models have been scored. The assessment framework provides two testing modes for consumers and OEMs respectively. The quick mode only takes around 30 minutes and is suitable for consumer to shop around a new car in hours. The deep scan is for OEMs to identify security issues. The researchers in [8] conducted a theoretical attack vector survey on 20+ different vehicles, and published a list of the most hackable and least hackable vehicles. Compared with their research works, our assessment framework is automated and experimental. Our top-secured results also matched their *the least hackable* pick.

The rest of the paper is organized as follows. We begin in Section 2 by describing the scoring schema of the assessment framework. In Section 3, the assessment management will be discussed. Testing results are shown in Section 4. Section 5 presents the conclusion.

II. Scoring Schema

A. Score Calculation

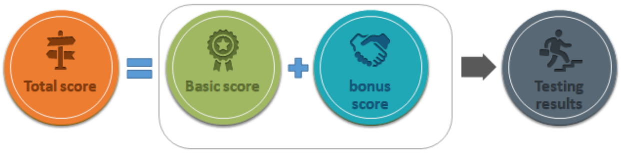

Fig. 1 Total Score

As shown in Fig. 1, the total score is composed of base and bonus scores. The max base score is 100 and the max bonus score is 10. Therefore, the max total score is 110.

$$S_{total} = \sum_{i=1}^{n} S_{base\ i} W_i V_i + \sum_{i=1}^{m} S_{bonus\ i} W_i V_i$$

Where i is the index, S_{total} is the total score, S_i is the score of the *ith* testing point, W_i is the weight of the *ith* testing point , V_i is the weight of the category including the *ith* testing point. S_{base} is the base score and S_{bonus} is the bonus score.

B. Score Categories

There are four assessment categories in the base score: information leakage, attack simulation, test damage, and driving distraction.

The bonus score is divided into two categories: anti-leakage and attack alerting. During our testing we did find some tested vehicles with security features deployed. For example, one vehicle used a whitelisting “firewall” behind the OBD II port. Another vehicle developed a mobile app which could monitor CAN Bus diagnostic information, and sent alerts to the cloud. During the testing, the app triggered warning alerts.

Table 2. Score Categories with Weight Percentages

score	**category**	**Test list**
Base（100）	**Information Leakage 25%**	CAN BUS sniff
		ECU sniff
		CAN activity
		ECU activity
		CAN Reverse engineering
	Attack Simulation 50%	ECU access attack
		ECU read/write attack
		ECU manipulation
		traffic attack
		Recoverable

	Test Damage 15%	Non recoverable
	Driving Distraction 10%	How to effect driving
Bonus （10）	**Anti-leakage 5%**	Defend against CAN information leakage
	Attack Alerting 5%	Alert when detecting attacks

The testing time for each assessment category ranges from hours to days (varies by vehicle model). We implemented a test portal. By clicking each test scenario, a testing page will pop up, where the user can configure the desired testing parameters. When testing, the test portal injects specific CAN traffic into the CAN bus via the test tool and collects related responses from the CAN Bus. The tool can flag or alert results of a series of tests designed and targeted to perform various specific tasks that should not normally be permitted by the system. The tool prepares automated reports for each test performed, including data logged when specific situations produced anomalous results. Further analysis or comments by live operators can also be amended, as desired.

The test could be a single test, or many multiples of tests as desired. The library and menu of available tests will continue to grow and evolve over time, and as new specific threats and tests are identified and confirmed.

C. Test Damage

We have defined three test damage levels classified as the high, medium and low.

Low risk – It means minor anomalies, disruption of vehicle or system operation, diagnostic trouble codes (DTCs), or unexpected behaviors noted. These consequences have minimal to no effects on safety or user driving experiences. Vehicle returns to normal operation upon shut down and restart. Any error or DTC codes reset upon restart. Examples might include illumination of a non-critical indicator, temporary interruption of a non critical system, or temporary DTC codes relating to non safety or non-emissions related system performance.

Medium risk – It means some anomalies, noted non-critical disruption of various vehicle systems, various components or displays not working, but vehicle otherwise operating. A power reset of the vehicle systems by disconnecting, and then reconnecting the battery may be required to reset and resolve these issues. Examples might include, non critical system DTC codes set, error indicators illuminated, some displays blank, Convenience or Entertainment features not working.

High risk – It means a major vehicle disruption, which means a major or critical systems not functioning or loss of system control, e.g. a power reset of the vehicle systems by disconnecting, and then reconnecting the battery may be required. Some systems may require ECU reflashing to restore operation.

D.Ranking Level

Table 3. Star-based Rank Level

Total score	**Star level**	**Certificate**	**medals**
≥85	★★★★★	Yes	Gold Silver Bronze
≥80 and <85	★★★★		
≥70 and <80	★★★	no	no
≥60 and <70	★★		
<60	★		

Our assessment framework uses the star-level ranking. Table 3 shows how to map the total score into different star level. For each group of the tested vehicles, the gold, silver and bronze medals will be selected. However, each medal candidate must be above four-star level.

IV. Assessment Management

A. Test Time and Test Mode

According to KPMG 2016 survey study [9], 82% of consumer respondents would be not willing to buy from an automaker if they had been hacked. On the other hand, J.D. Power showed the average time for consumers to spend at a dealership in making a car purchase decision was 187 minutes, more than 3 hours [10]. Is that possible to perform a quick security scan for a car in 30 minutes so the consumers would know how secured a car is when they are making purchase decisions?

30 min
Security assessment
car purchase
Other factors

Fig. 2 Quick Security Test in Car Purchasing

The answer might be yes. The normal duration for a complete security testing ranges from several days to weeks. This time is dependent on the number of CAN buses or ECUs in a vehicle. Some vehicles have as many as 80 or more ECUs. The number of CAN buses will also impact testing duration. Many vehicles have as many as four separate CAN bus architectures. To expedite the testing process, we designed the quick test mode for car purchase purpose. It means consumers are able to spend a short period of time to make purchase decisions with the knowledge how well the car model would protect themselves from car hacking or reverse engineering.

Table 4. Two Test Modes

	Name	Desc.	Target
🧍🧍🧍🧍🧍🧍🧍🧍	Quick mode	Quickly detect security issues in 30 min Normally NO damages for vehicles	car buyers
🧍🧍🧍🧍🧍🧍🧍🧍	Deep mode	Fully detect security issues in 3-7 days Might cause damages to vehicles	OEMs; testing labs

B. Test Tool Requirement

A black-box testing tool plays a key role in the assessment framework. A qualified testing tool can identify security issues with repeatable results. Even if the tester does not have prior security testing knowledge or background, he or she can still operate the test device and detect the security vulnerabilities by following automated steps with repeatable results.

The tool should be capable of testing entire vehicles, individual CAN bus, or specific components running simulations on the bench. The test could be a single test, or many multiples of tests as desired. The library and menu of available tests will continue to grow and evolve over time, and as new specific threats and tests are identified and confirmed. The tool also need have several communication modules, such as WIFI, Bluetooth, and 3G modules, for both short-range and long-range communications.

It is often desirable for cross-department collaborations that require remote testing. The testing team and the vehicle can be in different physical locations. Authorized users would conduct remote tests in varied global locations. First the tester needs to log into the cloud by inputting the login credentials. Then the tester runs the test by choosing and running the testing scenario from the cloud. Afterwards, the cloud generates the corresponding CAN traffic and forwards them to the laptop, which will forward into the vehicle via the device.

C. Test Area

The test area must be obstacle free, and the ground should be dry and clean. There are no objects left inside a car that could affect the test process. Fig 3 and 4 shows the space requirements for test area and car lift area.

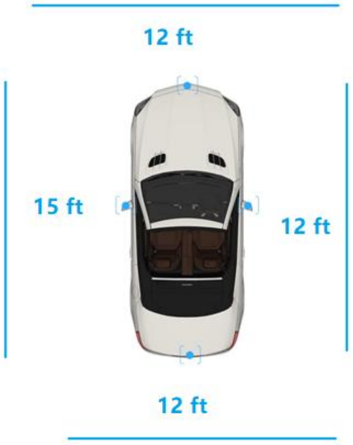

Fig. 3 Test Area Space

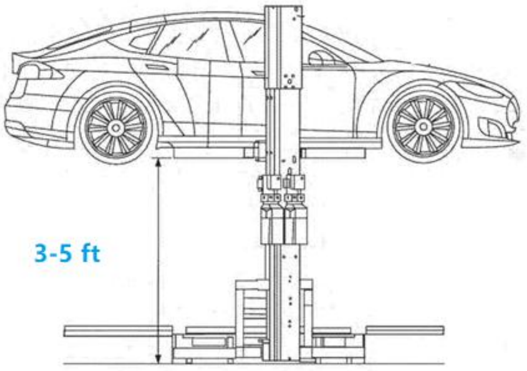

Fig. 4 Car Lift Space

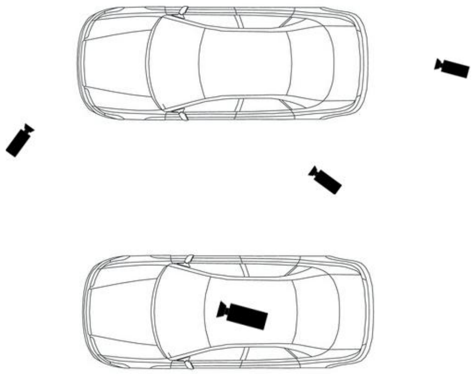

Fig. 5 Video Recorder Positions

Four video recorders are needed for logging the test results. One is put inside the car to record the cluster, and the other three will be used to record the car surroundings. The recorder resolution is at least 521x384. Fig. 6 shows the snapshot of our test working area.

Fig. 6 Testing Area

D. Trace Back Security Vulnerability

Tracing logs back and locating security risks found during the testing process is very important. Therefore, we developed the log analytics platform which can process and visualize the testing log data. Fig. 7 shows a snapshot of the platform. For example, the histogram shows the frequency of ECU IDs. By moving mouse onto each histogram bar, the portal will shows the specific CAN packet payloads for further analyses. For example, we can narrow down the specific sequences of CAN traffic trace which caused the engine misfire and then killed the engine of the vehicle.

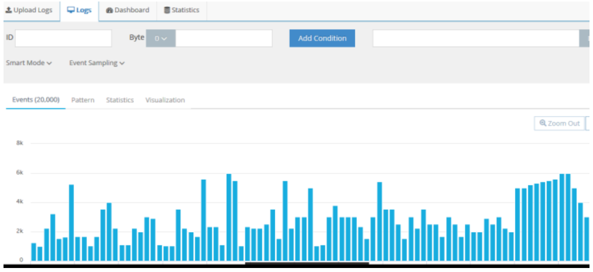

Fig. 7 Log Analytics Platform

The portal also supports the traffic replay function which is convenient for researchers to test traffic segments. By using machine learning algorithms, the classifications and anomaly detections can be applied to further automate and detect the abnormal test logs.

IV. Vehicle Testing results

A. Vehicles Tested

The framework supports various types of input feeds, e.g. real vehicles or the logs generated from supported test tools.

For the quick test, log files from tens of car models were used as the input feeds into the assessment framework. Those logs were provided by third-party who used the supported a test tool on vehicles. An online submission interface has been implemented for uploading the logs. Once receiving, the cloud portal called the scripts to analyze the testing results and calculated the scores. Afterwards, testing reports were generated for each test performed, including data logged when specific situations produced anomalous results.

For the deep scan, fifteen vehicle models have been tested. Those vehicles included both the electric vehicles and traditional ones. Some of them were purchased on the market and others were provided by OEMs. There were even a few pre-SOP (Start Of Production) vehicles which were not available on the market yet.

B. Autonomous Car Models

One of the biggest challenges facing autonomous vehicles is the cybersecurity. Before autonomous vehicles become widely deployed, car hacking is a real danger. Among the vehicles tested, five of them were popular autonomous car models, and 100+ companies around the world are building autonomous systems on their CAN platforms. It is worthwhile to compare the scores or rankings among them.

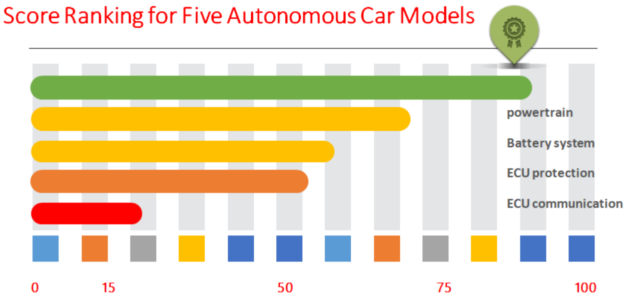

Fig. 8 Autonomous Car Test Process

We purchased five popular autonomous car models, and performed deep test on them. Fig. 8 showed the test process results of those five vehicles. The progress bar shows the time spent and percentage remaining. While the tests were running, displays would indicate various error messages or go blank, the engine misfire, or stop completely, and various system features became disabled, temporarily or permanently. Unfortunately not all vehicles tested were able to sustain the complete test. One vehicle was fully disabled and was unable to restart. Thus, problems like that prevented further test completion. It was interesting to see that each vehicle platform has different security risks, e.g. powertrain and battery management.

C. Fast Test Results

Fast test evaluated tens of car models. After importing the test logs into the assessment framework, the score schema was applied to calculate the total scores. Fig. 9 shows the snapshot and score distribution among the vehicles. The highest score was 89 and the lowest one was below 50. The average score was 68. In our understanding, it is the first time to quantify scores on so many vehicle models.

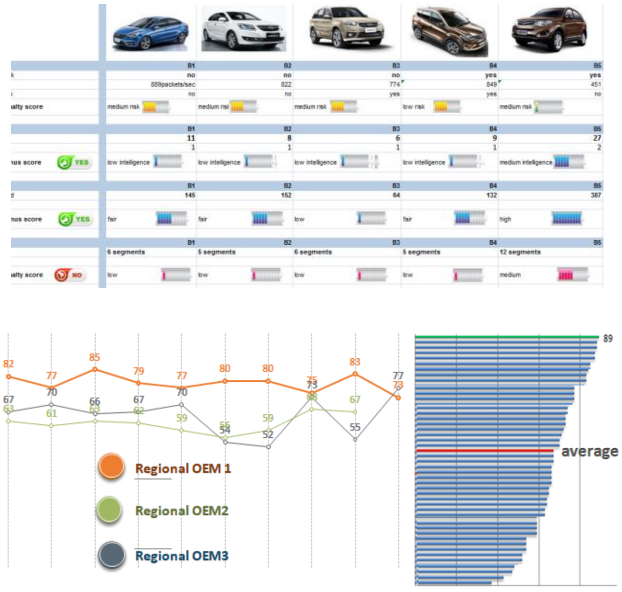

Fig. 9 Score Distribution and Lines

Three score lines were drawn on the left of Fig. 9. Each line represents a regional OEM. Different car models from the same OEM have similar scores, which means there are lots of ECU software overlapping on various vehicle CAN Buses from the same OEM. In fact, many tier-1 providers provide same ECUs to different car models. Fig. 10 shows the max, min and the average scores for two OEMs.

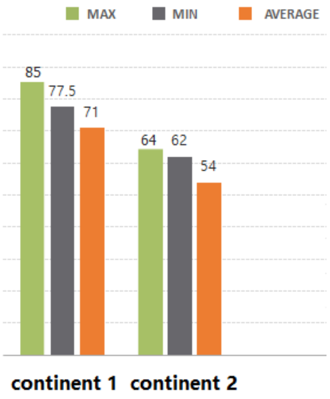

Fig. 10 Scores of Two Continent OEMs

D. Comparison to Related Work

Our results matched the works conducted by researchers who hacked the Jeep car. Miller and Valasek in [8] conducted a theoretical attack vector survey on 21 vehicle models, and published a list of the ***least hackable*** vehicles. They pointed out the 2014 Dodge Viper, and 2014 Audi A8 as the least hackable cars.

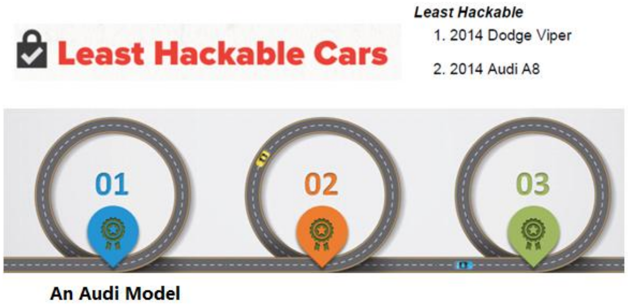

Fig. 11 Our Top-3 Rank vs. Least Hackable Cars

Compared with their theoretical works, our assessment is automatic and experimental. Besides, as shown in Fig. 11, our No. 1 ranked car, an Audi model, matched *the least hackable pick*, Audi A8. Since there was no Dodge car in our tested vehicles, the corresponding comparison is not available.

V. RESULT CORRELATIONS AND FUTURE WORK

Based on the assessment results, security correlation info can be extracted for cross-model analysis. For instance, three categories were defined for ECU fine-grained features. *ECU reuse* represents the degree of ECU re-usage cross various model platforms. *ECU manipulation* means how hard to fool around an ECU by modifying its parameters. *Gateway/access/info leakage* represents the security level applied. Fig 12 shows scores of fine-grained features among four OEMs.

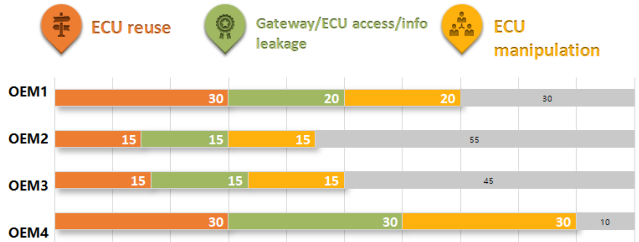

Fig. 12 ECU Fine-grained Feature Comparison

As shown in Fig. 13, our results from a study on vulnerability correlation also revealed that some joint-venture vehicle models have inherited security vulnerabilities from the original OEM model counterpart. The reason of those security vulnerabilities was caused by the irresponsible reusing ECU software codes. Even worse, some OEMs even couldn't be able to trace back which ECUs were affected by a security flaw due to poor inventory practices.

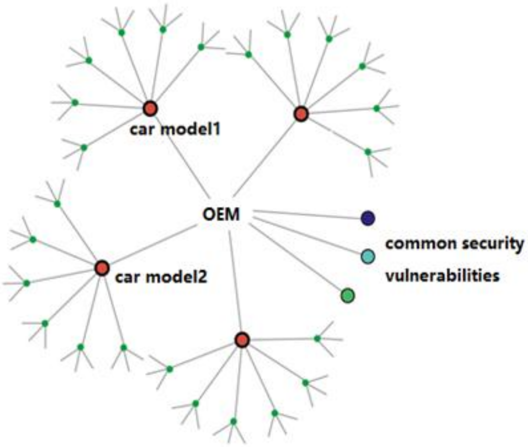

Fig. 13 Vulnerability Inherited Across Models

VI. CONCLUSION

In this paper, we have implemented a practical car security assessment framework which has successfully ranked tens of car models. In our understanding, it is the first time to quantify scores on so many vehicle models. The assessment framework includes two testing modes, the quick and deep tests, for consumers and OEMs respectively. Quick test enables consumers to spend a short period of time to make purchase decisions with the knowledge how well the car model would protect themselves from car hacking. OEMs can leverage deep test to find ECU software security vulnerabilities.

Our future work is to purchase and test more vehicles and establish more advance threat correlation algorithms and testing benchmarks for the industry.

REFERENCES

[1] http://www.iihs.org/frontend/iihs/documents/masterfiledocs.ashx?id=1661

[2] https://www.nhtsa.gov/fmvss/stars-cars-new-car-assessment-program-ncap-safety-labeling

[3] http://www.iihs.org/iihs/ratings

[4] Mohd Jawi, Zulhaidi & Hafzi, Mohd & Md Isa, Mohd Hafzi & Solah, mohd syazwan & Ariffin, Aqbal Hafeez & Abu Kassim, Khairil Anwar & Kassim, Abu & Wong, Shaw Voon. (2013). New Car Assessment Program for Southeast Asian Countries (ASEAN NCAP) – A New Paradigm Shift in the ASEAN's Automotive Ecosystem.

[5] http://www.jdpower.com/ratings-and-awards

[6] http://standards.sae.org/j3061_201601/

[7] https://www.sae.org/works/committeeHome.do?comtID=TEVEES18A1

[8] Miller, C., Valasek, C.: A survey of remote automotive attack surfaces. Black Hat USA

[9] https://info.kpmg.us/content/dam/info/consumer-loss-barometer/pdfs/CLB10-11.pdf

[10] https://www.cars.com/articles/2007/11/buying-a-car-ta/