POSTER: Intrusion Detection System for In-vehicle Networks using Sensor Correlation and Integration

Huaxin Li
Intel Labs
huaxin.li@intel.com

Li Zhao
Intel Labs
li.zhao@intel.com

Marcio Juliato
Intel Labs
marcio.juliato@intel.com

Shabbir Ahmed
Intel Labs
shabbir.ahmed@intel.com

Manoj R. Sastry
Intel Labs
manoj.r.sastry@intel.com

Lily L. Yang
Intel Labs
lily.l.yang@intel.com

ABSTRACT

The increasing utilization of Electronic Control Units (ECUs) and wireless connectivity in modern vehicles has favored the emergence of security issues. Recently, several attacks have been demonstrated against in-vehicle networks therefore drawing significant attention. This paper presents an Intrusion Detection System (IDS) based on a regression learning approach which estimates certain parameters by using correlated/redundant data. The estimated values are compared to observed ones to identify abnormal contexts that would indicate intrusion. Experiments performed with real-world vehicular data have shown that more than 90% of vehicle speed data can be precisely estimated within the error bound of 3 kph. The proposed IDS is capable of detecting and localizing attacks in real-time, which is fundamental to achieve automotive security.

CCS CONCEPTS

•**Security and privacy → Intrusion detection systems;**

KEYWORDS

vehicular security; in-vehicle intrusion detection system; cyber-physical security

1 INTRODUCTION

The increasing adoption of electronics in vehicles (e.g., electronic control unit, infotainment, V2X) has led to an expanding attack surface and even making them vulnerable to physical and remote attacks [5, 9]. The commonality among these attacks is that attackers were able to inject messages and tamper with the data in the CAN bus. Therefore, they were able to impersonate as valid electronic control units (ECUs) to subsequently perform malicious actions (e.g. braking) and deviate the system from a safe operational regime. The underlying problem that enabled the aforementioned attacks is the lack of proper message integrity in the CAN bus, which allows illegitimate messages can be consumed by the system as authentic ones. Cryptographic mechanisms for data origin authentication may not fully resolve the problem as ECUs, since attackers who have gained software execution on ECUs can still send their own authentic messages but with the attacker's malicious payload.

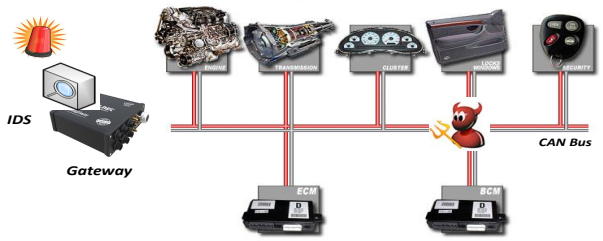

Figure 1: The intrusion detection system is deployed on gateway and monitors CAN bus messages

To ensure security and safety, it is crucial to employ intrusion detection systems (IDS) capable of deeply inspecting the bus and detecting anomalies. Meta-data based IDS focuses on each message by examining its message ID [6], frequency [7, 10], clock-skew [1], etc. Data content based IDS examines data content of messages by building machine learning models [3] and information theory models [8] for each kind of message data payload. There are limitations on what kind of attack each of the aforementioned methods can detect. For example, frequency-based IDS [7, 10] is able to detect message injection attacks by analyzing message frequencies. IDS based on physical properties [1] can identify impersonation attacks by fingerprinting each ECU. However both of them are unable to detect data alteration/falsification attacks. [3] and [8] can detect simple data spoofing attacks by examining the message content, but requires attack samples in its training phase. These limitations not only reduce the feasibility of intrusion detection in practice, but also fall short in detecting some more sophisticated types of attacks, for instance, when sensor readings are spoofed.

In this paper, we propose a machine learning based in-vehicle intrusion detection system based on correlations and redundancy among multiple sensors for detecting data spoofing attacks. Pairwise correlations between vehicular sensors have been analyzed in [2]. In our approach, a regression model is trained to integrate correlations from multiple sensors and estimate a targeted sensor value using other correlated parameters in real time. The difference between the estimated value and observed value of this sensor data is used as a signal for detecting anomalies. The integration of multiple heterogeneous sensors increases the attacker's difficulty

CCS'17, Oct. 30–Nov. 3, 2017, Dallas, TX, USA.
 ISBN 978-1-4503-4946-8/17/10.
DOI: http://dx.doi.org/10.1145/3133956.313884

of bypassing the IDS, while making the IDS more stable under different scenarios. As shown in Fig. 1, the IDS can be used in the vehicle's gateway to simultaneously monitor CAN messages containing real-time sensor signals. Experimental results on a real car show that our intrusion detection system is capable of effectively detecting sensor spoofing attacks.

2 METHODOLOGY

2.1 Problem Formulation

In order to detect anomalies, it is first necessary to estimate a targeted sensor value using other correlated parameters . It can be formulated as a prediction problem and modeled using regression models in machine learning. Regression models aim at describing statistical process for estimating the relationships among variables. So in the training phase, a set of correlated signals values are used to train a regression model M.

Given a trained regression model M and a set of parameters $S = \{s_1, s_2, ..., s_n\}$ (e.g., s_1=speed, s_2=acceleration, s_3=engine speed) extracted from CAN bus messages. Inputs of M are parameters values except one targeted value s_i that is being estimated (e.g., speed), i.e., $S - s_i$, and the output is the estimated value of s_i, denoted as s_i'. Then anomalies can be flagged by comparing s_i' with observed s_i.

2.2 Features and Model

In order to estimate the targeted sensor's values, correlated sensor values should be chosen as features of the regression model. The choice of features can be either 1) based on domain knowledge, which means people know which parameters are correlated to the targeted one based on common sense. For example, it should be expected that engine speed and longitudinal acceleration are highly correlated to the vehicle speed. 2) based on correlations, which means pairwise correlations of sensor values during some time are computed (e.g., using Pearson correlation coefficient), and high-correlation sensors are chose as the features.

Table 1 contains features we used to estimate vehicle speed as an example. They are collected from heterogeneous sensors and sent to CAN bus through multiple ECUs. For ease of presentation, we use vehicle speed and related sensor as an example throughout this paper. Without loss of generality, this model can be applied to other sensors by selecting correlated features.

Table 1: Features used for estimating vehicle speed

Features	Unit	Source ECU
Engine speed	rpm	PCM
Longitudinal acceleration	m/sec2	RCM
Lateral acceleration	m/sec2	RCM
Brake pedal	%	ABS
Yaw rate	m/sec2	ABS
Steering angle	degree	SASM
Gear	-	TCM

We choose Random Forest Regressor as the regression model in this paper. It is an ensemble of different regression trees and is used for nonlinear multiple regression [4]. A continuous estimated value will be given as the output of the model at each prediction.

2.3 Detection

At each time instance, an observed parameter's value can be interpreted from a corresponding CAN message. Meanwhile, we can estimate a value using other sensors' values extracted from other CAN messages, which is denoted as the estimated value. A simple strategy is to set a threshold for the difference between estimated values and observed values. For the examples shown in this paper, a speed threshold of 5 kilometer per hour (kph) is considered. Thus, a speed reading is considered normal if the difference between an observed value and its corresponding estimated value is less than 5 kph. Additional strategies can be applied to strengthen the confidence in the anomaly recognition. For example, if there are multiple consecutive estimations exceed the threshold, or majority of estimation exceeds the threshold over a period of time.

3 EVALUATION

To evaluate the performance of our IDS, we collected CAN data from a real car and simulated sensor spoofing attacks. The experimental settings is based on a Ubuntu 16.04 laptop connected to the car's CAN bus through ODB-II socket. Then the car was driven for collecting data. We collected three segments of driving in the city traffic (denoted as City1, City2, City3, respectively), and one segment of driving one a highway (denoted as HighWay). In the average, each segment contains more than 20 minutes of driving data.

We first evaluated the accuracy of the estimation to reflect the feasibility of our approach. For city driving data, each time we use two of the driving segments to train the regression model, and then estimate the speed in the remaining driving segment using features in Table 1. For highway driving data, we use the first half of data as training data, and the second half of data as testing data.

Table 2: Performance of estimating vehicle speed

Scenario	Error Mean (kph)	Error Median (kph)	Error $\leq$ 3 kph (%)
City1	0.996	0.147	91.25
City2	1.146	0.132	94.92
City3	0.670	0.222	91.38
HighWay	1.605	0.388	93.61

Table 2 shows the experiments results, where the (estimation) error is defined as $abs(estimated\ values - observed\ values)$. Table 2 shows that mean and median values of estimation error are closed to real observed values, and more than 90% of estimations can be done with errors smaller than 3 kph. This means that most of the estimation are accurate, and if we observe a sequence of estimated values exceed the threshold, we have high confidence to identify an attack. False positive will be caused by inaccurate estimations, and it can be reduced by: 1) increasing the training data, 2) leveraging sensor redundancy to improve estimation accuracy

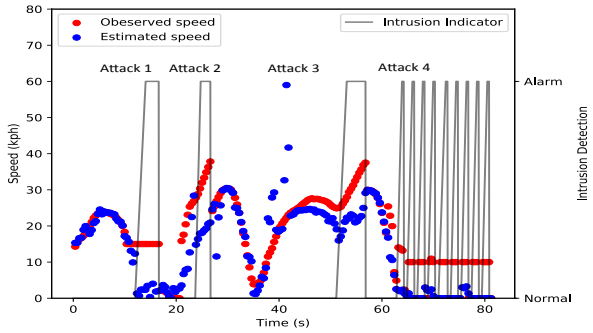

Figure 2: Illustration of intrusion detection

by taking homogeneous parameters into account (e.g., GPS speed, wheel speed v.s. vehicle speed). This will be discussed in Section 4.1.

The feasibility of detecting attacks are demonstrated by considering four types of sensor spoofing attacks:

Attack 1: spoofing sensor readings to a fixed value. For example, spoofing vehicle speed to 15 kph.

Attack 2: spoofing sensor readings to values that are deviated from real values. For example, adding an extra 15 kph to current speed readings.

Attack 3: gradually changing sensor readings. For example, adding 0.01 kph to the deviation of the current speed until reaching the attacker's goal.

Attack 4: alternately and repeatedly sending malicious and normal sensor readings. For example, the attacker can send several modified messages and then send several unaltered messages to mislead the IDS.

Fig. 2 demonstrates these four kinds of sensor spoofing attacks and the corresponding detection results respectively. Attack 1 and Attack 2 are immediately detected because the spoofed values exceed the threshold. Attack 3 is detected once the deviation exceeds the threshold of 5 kph. It means that the spoofing cannot exceed the threshold, which limits attackers' capability of spoofing sensor values and ensures safety to some extent. Attack 4 can also be detected because normal estimation should be dominated during a time period. This limits the number of messages that can spoofed by the attackers. Several wrong estimations occur at around 40th second, but they don't trigger alarms (false positives) based on our design (because they are not dominant during that time). False positives can be further minimized by increasing estimation accuracy and adopting different strategies in future work. Besides, our experiments show that these detection can be accomplished in real time, which means computations of the model will not cause delay.

4 DISCUSSION AND ANALYSIS

4.1 Sensor Redundancy

To optimize the estimation, we can further leverage redundancy among sensors. The redundancy can be found in 1) different CAN bus networks sent by different ECUs. For example, we can find vehicle speed values in both Powertrain (PT) and Chassis (CH) networks. 2) homogeneous sensors. For example, the speed computed at four wheels (i.e., wheel speed) and speed computed by GPS signals (i.e., GPS speed) are good references for vehicle speed. These redundant data/sensors can be taken as features of the model to improve accuracy. For example, after adding wheel speed as a feature, the average errors for estimating vehicle speed became only 0.101 kph.

4.2 Incremental learning

Another factor that affects estimation accuracy is under different contexts (e.g., road condition, topology, weather), the correlations among sensors would change therefore causing false positives. A solution to mitigate the influence of contexts is incrementally train the model using new data in real time, so that the model can be adjusted to learn the correlations under new contexts.

4.3 Attack Localization

The IDS introduced in this paper is capable of not only detecting intrusion/anomalies, but also localizing the parameter that was compromised. For example, when longitudinal acceleration is modified by increasing 2 m/s^2, the model targeting at longitudinal acceleration triggers an alarm immediately, while the model targeting at vehicle speed will not be sharply affected.

5 CONCLUSIONS

In this paper, we propose an approach to integrate multiple correlated/redundant parameters to detect sensor data spoofing attacks for in-vehicle networks. The proposed approach uses a Random Forest Regression as a predictor of our model. As studied in this paper, more than 90% vehicle speed parameters can be estimated by our model within the error bound of 3 kph, which provides a safety boundary. Experiments on vehicle data show that the proposed approach can effectively detect intrusion attacks and anomalies in real time, but also determine the compromised parameter during the attack.

REFERENCES

[1] Kyong-Tak Cho and Kang G Shin. 2016. Fingerprinting Electronic Control Units for Vehicle Intrusion Detection.. In *USENIX Security Symposium*. 911–927.
[2] Arun Ganesan, Jayanthi Rao, and Kang Shin. 2017. *Exploiting Consistency Among Heterogeneous Sensors for Vehicle Anomaly Detection*. Technical Report. SAE Technical Paper.
[3] Min-Joo Kang and Je-Won Kang. 2016. Intrusion detection system using deep neural network for in-vehicle network security. *PloS one* 11, 6 (2016), e0155781.
[4] Andy Liaw, Matthew Wiener, et al. 2002. Classification and regression by randomForest. *R news* 2, 3 (2002), 18–22.
[5] Charlie Miller and Chris Valasek. 2015. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA* 2015 (2015).
[6] MarchettiïijÑ Mirco and Dario Stabili. 2017. Anomaly detection of CAN bus messages through analysis of ID sequences.. In *Intelligent Vehicles Symposium (IV)*.
[7] Michael R Moore, Robert A Bridges, Frank L Combs, Michael S Starr, and Stacy J Prowell. 2017. Modeling inter-signal arrival times for accurate detection of CAN bus signal injection attacks: a data-driven approach to in-vehicle intrusion detection. In *Proceedings of the 12th Annual Conference on Cyber and Information Security Research*. ACM, 11.
[8] Michael Müter and Naim Asaj. 2011. Entropy-based anomaly detection for in-vehicle networks. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 1110–1115.
[9] Keen Security Lab of Tencent. 2016. Car Hacking Research: Remote Attack Tesla Motors. (sep 2016). http://keenlab.tencent.com/en/2016/09/19/Keen-Security-Lab-of-Tencent-Car-Hacking-Research-Remote-Attack-to-Tesla-Cars/
[10] Hyun Min Song, Ha Rang Kim, and Huy Kang Kim. 2016. Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network. In *Information Networking (ICOIN), 2016 International Conference on*. IEEE, 63–68.