Scission: Signal Characteristic-Based Sender Identification and Intrusion Detection in Automotive Networks

Marcel Kneib
Bosch Engineering GmbH
Abstatt, Germany
Marcel.Kneib@de.bosch.com

Christopher Huth
Robert Bosch GmbH
Renningen, Germany
Christopher.Huth@de.bosch.com

ABSTRACT

Increased connectivity increases the attack vector. This also applies to connected vehicles in which vulnerabilities not only threaten digital values but also humans and the environment. Typically, attackers try to exploit the Controller Area Network (CAN) bus, which is the most widely used standard for internal vehicle communication. Once an Electronic Control Unit (ECU) connected to the CAN bus is compromised, attackers can manipulate messages at will. The missing sender authentication by design of the CAN bus enables adversarial access to vehicle functions with severe consequences. In order to address this problem, we propose Scission, an Intrusion Detection System (IDS) which uses fingerprints extracted from CAN frames, enabling the identification of sending ECUs. Scission utilizes physical characteristics from analog values of CAN frames to assess whether it was sent by the legitimate ECU. In addition, to detect comprised ECUs, the proposed system is able to recognize attacks from unmonitored and additional devices. We show that Scission is able to identify the sender with an average probability of 99.85 %, during the evaluation on two series production cars and a prototype setup. Due to the robust design of the system, the evaluation shows that all false positives were prevented. Compared to previous approaches, we have significantly reduced hardware costs and increased identification rates, which enables a broad application of this technology.

CCS CONCEPTS

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

KEYWORDS

Automotive Security; Controller Area Network; Sender Identification; Intrusion Detection

ACM Reference Format:
Marcel Kneib and Christopher Huth. 2018. Scission: Signal Characteristic-Based Sender Identification and Intrusion Detection in Automotive Networks. In *2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18), October 15–19, 2018, Toronto, ON, Canada.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3243734.3243751

CCS '18, October 15–19, 2018, Toronto, ON, Canada

ACM ISBN 978-1-4503-5693-0/18/10…$15.00
https://doi.org/10.1145/3243734.3243751

1 INTRODUCTION

In-vehicle security has become a major issue due to the openness of modern systems and high connectivity. This is especially true if an attacker is able to infiltrate an Electronic Control Unit (ECU) by physical access. This allows an adversary to control a wide range of high safety critical functions, completely ignoring driver input, like stopping the engine or disabling the brakes [27].

Since requiring prior physical access has been viewed as unrealistic [3], the authors of [3, 30] have systematically analyzed and exploited additional attack surfaces including remote connections, e.g. Bluetooth and cellular radio. Using these external vectors, attacks on ECUs become more realistic and allow them to be executed on a variety of vehicles. The disadvantages in security and consequently also in safety through these connections were demonstrated by Miller and Valasek [31] by their work on a Jeep Cherokee, which led to a recall of 1.4 million vehicles. They were able to gain access to the in-vehicle network via the cellularly connected head unit, enabling them to control, among other functions, the engine, brakes and steering.

However, the risks resulting from compromised ECUs exist since there are no security features in most in-vehicle networks. Without adequate security measures an attacker can send forged messages from infiltrated ECUs or additional devices. This applies in particular to the Controller Area Network (CAN) [10], a serial broadcast bus, designed in 1983 at Robert Bosch GmbH, without considering any security features [16]. CAN is still the most widely used bus system for in-vehicle networks and can be found in every car today. This absence of security measures enables each bus participant to transmit every possible message. Participants on CAN receiving such messages will not be able to identify the sender and thus to verify their authenticity.

1.1 Shortcomings of Security Solutions at a Digital Level

Unfortunately, it is not practicable to apply classical cryptographic algorithms such as a Message Authentication Code (MAC), due to the extremely limited bandwidth and the short payload of a maximum of 8 bytes per message. In addition, more computing power would be required for secure communication to meet real-time requirements. These limitations make it difficult to apply classical cryptographic measures with the recommended settings in today's cars without making any profound changes. In order to keep security risks as low as possible, security concepts, including hardware security modules (HSMs) or network partitioning [25], are already taken into account during the development process.

A large number of attacks on vehicle functions involve communication via the vehicle network and thus change the actual traffic on

CAN. Systems to detect these changes, by observing network traffic at the message level, called Intrusion Detection Systems (IDS), have been well-studied [28]. There are two well defined approaches for IDS, the signature- and anomaly-based detection. The former, more common in production systems, detects known attacks based on their message pattern and content. However, the deployment is difficult as information of actual attacks are highly platform- or vehicle-specific and rarely available in practice. The latter, the anomaly detection, characterizes the expected message traffic, whereby anomalies and deviations can be detected. The expected characteristics can be trained or explicitly specified [1], whereby unknown attacks can be recognized. These systems typically suffer from a high number of false positives [34].

1.2 On the Need for Physics-Based Security

Physical characteristics can be used to enhance security in the in-vehicle networks [5, 6, 33]. Differences in the characteristics of CAN signals are used to extract fingerprints which are suitable to identify the sender. Security based on physical properties has the advantage that no additional computation is required to add security features to the communication. The characteristics of the signals are intrinsically defined by the used hardware and the structure of the bus topology. This enables the implementation of security even for lightweight communication participants with low computational capacity, such as smart sensors. Since these characteristics are hardware-defined, it is also difficult for an attacker without physical access to circumvent such systems.

Such approaches are useful additions to classic IDSs [16, 34] to detect additional attacks and increase their robustness. Even if a detected anomaly does not trigger a direct reaction, such as a warning to the driver [15, 34], these findings can be used to quickly adapt signature-based systems. The anomaly detection can be realized, e. g. in the cloud to update the rule set over the air. This allows affected vehicle models to be immunized against new attacks before the weakness in the implementation is found and eliminated by the manufacturer. Additionally, the IDS identifies the attack source faster and thereby quickens the forensic work of vulnerability removal. This allows a much faster response compared to software updates, which are usually carried out during workshop visits. Such a system can also save considerable costs if otherwise a recall by the manufacturer would be necessary.

1.3 Contributions

In this work, we present Scission (Signal characteristic-based sender identification), a system for identifying the origin of CAN messages, using immutable physical characteristics of CAN signals. This enables the system to recognize whether the sending ECU is authorized to send the evaluated message. Thus, anomalies can be detected and the compromised devices can be identified, which enables manufacturers to react appropriately to detected anomalies and prevent possible major damage. Compared to existing approaches which take advantage of several characteristics, we have been able to drastically reduce the sampling rate. Together with the use of lightweight algorithms and the fact that the bandwidth remains completely available, Scission ensures a cost-effective feasibility. In addition, the preprocessing steps and the robust design made it possible to achieve a high detection rate without the occurrence of false positives during the evaluation. Besides compromised ECUs, Scission is also able to detect unmonitored ECUs, as used in the Jeep Hack [31], and simple additional devices, which are often used for manipulations [14, 20]. In addition, we supplement the causes of the different signal characteristics used for identification. In short, our contributions are:

- We propose an IDS based on the measurement of physical characteristics, which allow us to determine the source ECU of each transmitted message.
- Our proposed IDS can be deployed to in-vehicle networks since it does not reduce the bandwidth and has low resource requirements.
- We show the applicability and usefulness of our IDS by extensive measurements in an automotive prototype setup and in two series vehicles.

1.4 Organization of the Paper

After the introduction and overview in Section 1, Section 2 provides necessary background information about the Controller Area Network, the causes of the signal characteristics and the differences between ECUs. An overview of the related work and a comparison with Scission is given in Section 3. This follows in Section 4 the description of the considered security model, the fingerprinting approach and how it is used to detect anomalies. In Section 5 the evaluation of our approach on a prototype and two vehicles follows. There, the different methods for detecting the attack models are analyzed. Section 6 wraps up the paper by describing the applicability and limitations of Scission. In Section 7 we close our paper with a final conclusion.

2 BACKGROUND

2.1 Controller Area Network

CAN is a broadcast vehicle bus, designed 1983 at Robert Robert GmbH to reduce the cost of the cable harness in vehicles. It is used to interconnect several ECUs with two twisted wires (high, low), terminated at each end with 120 Ω resistors. If a recessive bit (1) is present on the bus, the voltage of 2.5 V is applied to both lines. In case a dominant bit (0) is transmitted the high wire is driven towards 5 V and the low wire towards 0 V. Whenever five similar bits are transmitted, an contrary bit is inserted for synchronization purposes, called a *stuff bit*. The signal of a CAN frame is shown in Fig. 1, where CAN high is colored in blue and CAN low in red.

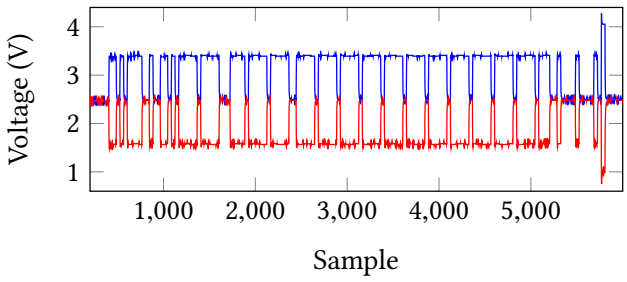


Figure 1: An exemplary CAN signal of the Fiat 500.

The data is transmitted via CAN frames, capable to carry 8 bytes of payload. The frames contain an unique identifier, representing the priority and meaning of the following data. No node addresses are present. Additionally it is possible to use an extended identifier format, which allows to increase the identifier from 11 to 29 bits. This will result in a slight change of the format, as besides the extended identifier two additional bits will be inserted.

As CAN is a broadcast bus, several bus participants may access the bus simultaneously. In this case, the dominant bits prevail and the participants whose bits have been overwritten cancel the transmission. Since the identifiers are only used by one ECU, it is ensured that only one device accesses the bus after the identifier has been completely transmitted. As soon as the bus is free again the interrupted ECUs restart the transmission. Thus, the bits of this arbitration phase may be influenced by several devices, which must be taken into account when using the signals for identification.

2.2 Cause of the Signal Characteristics

There are several causes that the analog signal deviates from a perfect theoretical square signal. As described in [41], some of these causes are bus termination, DC and AC voltages and grounding.

Our experiments show that we can extend that list by including the voltage sources for different ECUs. Variations in voltage output has been described in the work by Cho *et al.* [5]. There, variations in supply and ground voltages are used, due to in-vehicular variations of ECU's voltage regulators, to identify the source of a message. They furthermore describe how these variations in CAN transceivers nominal supply voltage result in different CAN high and low dominant voltages, as well as how transient changes in on-state resistances are reflected to the CAN bus.

Additionally, natural variations occur in all built-in parts of the CAN bus. For instance resistors come with an error tolerance of industry typical 5 %, which also cause variations in the CAN bus fingerprint. Besides the previous work for explaining variations on the analog signals on the CAN bus, we point to the direction of signal reflection. Signal reflections occur when a signal is transmitted e. g. on a cable, where some of the signal power is reflected due to imperfections causing impedance mismatches and non-linear changes in cable characteristics. For example one cause of these imperfections, ringing, is well known in the automotive industry and special transceivers try to minimize ringing [23]. But, despite all efforts signal reflections can only be minimized, because of variations in hybrid bus topologies, terminations and length of stub-lines [32]. Summarized, we identified the following sources of signal characteristics that can be used for the extraction of CAN fingerprints:

i) Variations in supply voltages,
ii) Variations in grounding,
iii) Variations in resistors, termination and cables, and
iv) Imperfections in bus topology causing reflections.

3 RELATED WORK

The known approaches [12, 29, 40] applying message authentication on CAN, cannot be easily applied due to the additional resources required or sophisticated hardware modifications needed.

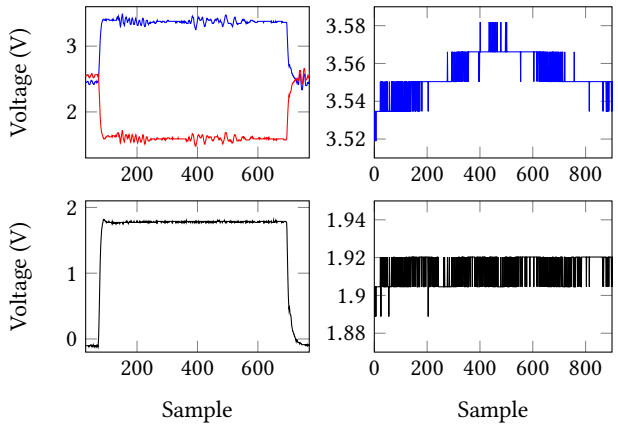
Herrewege *et al.* [40] proposed an approach to provide message authentication by using the CAN+ protocol [43]. Due to the proposed protocol changes, it is not feasible to apply such a system without making profound modifications, requiring soft- and hardware adaptations on each ECU.

Groza *et al.* [12] and Lin *et al.* [29] provide message authentication by sending additional messages. This leads to a higher bus load and thus to a reduction of available bandwidth for application specific data. The reduction of available bandwidth in CAN, especially for existing systems, has potentially wide-ranging consequences. This is due to the circumstance that such communication systems are well planned, including defined bandwidth reserves, to meet real-time and thus safety requirements. A reduction of the bandwidth has to be considered at design time of the communication, thus making a redesign necessary for existing systems. This may leads to additional separated buses to provide enough bandwidth reserves to fulfill safety requirements.

Another method is the usage of truncated MACs [38], placed in the data field. The general recommendation is to use at least 64 bit [8] for the MAC tag, to ensure security and to avoid collisions, which corresponds to the maximum payload of a CAN frame. In addition, there is an increased overhead for key management and freshness counters, which are necessary to avoid replay attacks. Even if just 32 bits were used, the available payload and thereby the bandwidth will be reduced heavily. Besides the disadvantage of reducing the available bandwidth, all relevant ECUs need more computing resources, to calculate and verify these MACs, to satisfy the real-time properties [42]. A standard using truncated MACs for secure in-vehicle communication is the AUTOSAR module Secure Onboard Communication (SecOC) [35]. Even if MACs are applied, they do not provide non-repudiation and thus cannot protect against counterfeit messages in general. As a result, ECUs capable of verifying a message are also capable of tampering them. To provide non-repudiation using cryptographic algorithms, digital signatures are unavoidable, which use much more resources and bandwidth and thus are unusable for CAN communication.

The authors of [4] introduce a Clock-based Intrusion Detection System (CIDS) which uses the clock skews of ECUs to detect malicious behavior on the CAN bus. Variations of the skew influences the difference of the expected and measured arrival time of frames. The skews are then exploited from these resulting offsets which are used for detecting intrusions. Due to the methodology the system is not able to observe aperiodic messages, it is outstanding how changes of the skew due to temperature shifts should be handled and it is not possible to reliably determine the ECU on which an attack is mounted. Furthermore, the vulnerability of that approach, demonstrated by Sagong *et al.* [36], is that an attacker can also observe the clock skew and adjust his transmission accordingly a target ECU to circumvent the CIDS.

Murvay and Groza [33] have shown that the characteristics of CAN signals are suitable for identifying the origin of frames. The signals of different ECUs are showing minor variations in e. g. how fast a rising edge is set up or how stable a signal is. They also observed that these characteristics remained unchanged over a period of several months. Using this observation, Choi *et al.* [6] introduces a system able to identify the origin of a message using an additional fixed 18-bit value in the extended identifier field. To



(a) Porsche Panamera **(b) Fiat 500**

Figure 2: Electromagnetic interference and voltage variations.

apply their method it is necessary to reprogram each device to send the fixed value in the optional extended identifier field, reducing in consequence the available bandwidth. They achieved the best results using a bit string consisting of 18 dominant bits, resulting in a total transmission of 24 additional bits, due to bit stuffing and the extended frame format. Their approach thus increases the total number of bits transmitted per message by over 20 % and can only be implemented at all if the extended identifier is not used for other purposes. In addition, the method is strongly based on the voltage level due to the use of mainly dominant bits, which means that other characteristics, such as the slope of the rising edges, are less strongly considered. In contrast, Scission uses characteristics of the messages' differential signal independently of the content transmitted, allowing the available bandwidth to be fully maintained. By this method, the extracted features can be used more specifically for identification, resulting in an average identification rate of 99.85 %. Compared to the work of Choi *et al.* [6], we greatly enhanced the feasibility of our approach by significantly lowering the resource requirements. This is due to the 125 times lower sampling rate the use of lighter algorithm for identification, as we use Logistic Regression instead of a neural network. Scission also analyzes the calculated sender probabilities in order to reduce the false positive rate, thus completely preventing the occurrence of false positives in the evaluation. Additionally, we emphasize detection of different types of attackers, such as the detection of an unmonitored ECU used in the attack on the Jeep Cherokee [31]. Lastly, we evaluated the approach in two production vehicles in addition to a prototype assembly, thus demonstrating that the approach also works in non-laboratory conditions.

Cho *et al.* [5] have introduced Viden (Voltage-based attacker identification), an approach to identify on which ECU an attack is mounted. Viden itself is an extension for IDSs, like the already mentioned CIDS [4]. The system creates a model from an average voltage level generated by a few observed dominant bits of several CAN signals, which is suitable to identify the manipulated ECU. As only a few measurements are necessary, the system has low hardware requirements. Viden uses only one signal characteristic, which is why we consider Scission to be more reliable in terms of changing conditions such as battery charge or electromagnetic compatibility, especially since Viden uses high and low signals instead of the differential signal. In contrast to the differential signals, the voltage levels of CAN high and low are affected from electromagnetic interference, for instance induced by electric consumers, as shown in Fig. 2a. Furthermore some ECUs have variations of their voltage level during the transmission of a dominant bit, like shown in Fig. 2b.

ECUs can be exposed to different conditions (hot engine compartment vs. cooled interior), which leads to different temperature changes and thus to different voltage changes. This is in contrast to the assumptions of Viden's update mechanism, as the authors assume that the voltage levels of the ECUs change evenly under changing conditions.

A comparison of the related fingerprinting approaches is shown in Table 1.

Table 1: Comparison of the fingerprinting approaches.

Approach	Choi *et al.* [6]	Viden [5]	Scission
Identification	96.48 %	99.8 %	99.85 %
False positives	3.52 %	0.2 %	0 %
Sampling rate	2.5 GS/s	50 kS/s	20 MS/s
Signal	differential	high, low	differential
Add. ECU	yes	no	yes
Unknown ECU	no	no	yes
Type	IDS, forensic	forensic	IDS, forensic

4 SCISSION

In the following section we describe the phases of our system for identifying the sender of CAN frames and how this information is used for intrusion detection. Before taking a closer look at Scission, we describe the network architecture, the security goal and the threat models we are considering.

4.1 Security and Threat Models

4.1.1 System Model. We assume a network that may consist of several separate buses, to which several ECUs are connected. The structure of a vehicle network varies between different models. There are complex networks in which the ECUs are separated according to functionality, e. g. for powertrain, comfort and body. The individual buses are connected via gateways, which can offer additional security mechanisms. In simpler networks with fewer buses, the effort for an attacker is correspondingly lower, as more ECUs can be used to send forged messages. Fig. 3 shows an exemplary architecture for both variants.

We consider Scission to be integrated into the network via an additional ECU, which is physically connected to the bus to be monitored. In order to prevent the attacker from simply bypassing the system, we assume that the ECU on which the system is implemented is secured by classic security mechanisms, such as HSM,

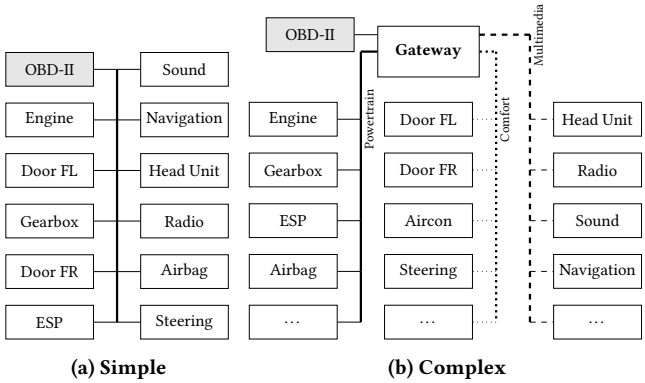


Figure 3: Example in-vehicle network architectures.

and is therefore considered trustworthy. For the integration of the system into an existing ECU, for instance gateways can be used to determine whether received messages have been sent from valid ECUs.

4.1.2 Security Goal. CAN provides no mechanisms to verify the authenticity of messages as described in Section 1. As a consequence, every bus participant is able to use all available identifiers and their receivers cannot verify that the messages originate from a valid sender. To compensate for this drawback, Scission was integrated into the system in order to determine the transmitter of a received CAN frame on the basis of the physical characteristics of its signal. The system monitors network traffic and thus detects unauthorized messages that have been sent via a compromised, unknown or additional ECU.

Since Scission determines the characteristics of the signal and the ECU's permissions based on the identifier, an attack can only be detected if a compromised or unknown ECU sends counterfeit CAN frames. For remotely compromised ECUs, whose detection is the main target, the system is also able to determine the control unit on which the attack is mounted.

4.1.3 Compromised ECU. The first attack vector we consider is to get access to the monitored CAN through an exploited vulnerability of an existing ECU. These may have additional connectivity interfaces such as cellular, WiFi or Bluetooth, which can be exploited by attackers [3, 30]. Using such an intrusion, it is possible to remotely send a variety of CAN frames, possibly without prior physical access and hidden from the vehicle or passengers. An attacker can use all possible identifiers and any message content.

4.1.4 Unmonitored ECU. The second attack, which got attention through the attack on the Jeep Cherokee [31], is the malicious usage of a passive or unmonitored device. The researchers have exploited an ECU's update mechanism to insert malicious code to turn a passive, listening-only device into a sending device. Using this, they were able to send messages to the network.

4.1.5 Additional ECU. The third attack vector is to attach an additional bus participant to the guarded network. Once an attacker has physical access to a vehicle, he can connect devices with little effort to, for example, steal vehicles [20], disable AdBlue systems or to perform engine tuning [14, 20].

The attacker can connect an additional device directly to the network or use the easy-to-reach On-board diagnostics (OBD)-II port of the vehicle. These ports are standardized for diagnostic purposes, allowing to get various information about the status of the vehicle. The OBD-II port is typically located beneath the dashboard and could be used to attach an additional device (such as a laptop or smartphone) to launch an attack.

This strategy is very similar to the attack by using an unmonitored ECU, with one difference. As already mentioned in Section 2.2, the exploited signal characteristics are defined among other factors by the bus topology. A change in the topology, as in this case, leads to an alteration in the characteristic of all connected ECUs.

4.1.6 Scission-aware Attacker. Since remote-based attack detection is the major goal, we also consider what a Scission-aware attacker may try to mislead the IDS. Since Scission uses physical characteristics of received frames, a remote attacker might attempt to mislead the IDS by influencing its signal. The characteristics which define the shape of the signal, especially of rising and falling edges, are mainly defined by causes which are not influenceable without physical access. A remote attacker could influence the supply voltage by draining the battery or heating or cooling the ECU, which primarily affects the absolute voltage level of the signals and not their general shape. Discharging the battery greatly affects the voltage level of all ECUs, as they are supplied from the same energy source, whereby a change in temperature mainly affecting the signal of the respective ECU.

4.2 Fingerprinting ECUs

In this section we describe our method to identify the origin of a received frame. An overview of the phases is shown in Figure 4. We start by sampling the signal, followed by the preprocessing, which allows us to estimate the transmitter independently of the transmitted data. There, the received frame is divided into individual bits and sorted according to the characteristic. Subsequently, different features of time and frequency domain are extracted from these characteristics, representing the actual fingerprint. To the first start of the system a model from these fingerprints is trained, which will then be used for classification respectively calculation of the probabilities. This enables the system to assign a sending ECU to a received frame, what is then evaluated by the intrusion detection. According to the available data, an alarm is triggered when an attack is detected.

4.2.1 Sampling. In the first phase, the analog signals of the received frames are recorded. For this purpose, either the differential signal or the two signals CAN high and low can be used separately. The direct use of the difference signal requires an additional circuit, but has the advantage that the system requires fewer resources because less data is stored temporarily and the two signals do not have to be combined. It is important to use the differential signal, as the separate signals can be influenced by electromagnetic interference or other variations. Otherwise, these influences could lead to incorrect predictions. The advantage of the differential signal V is that such signal noise can be compensated by each other, as

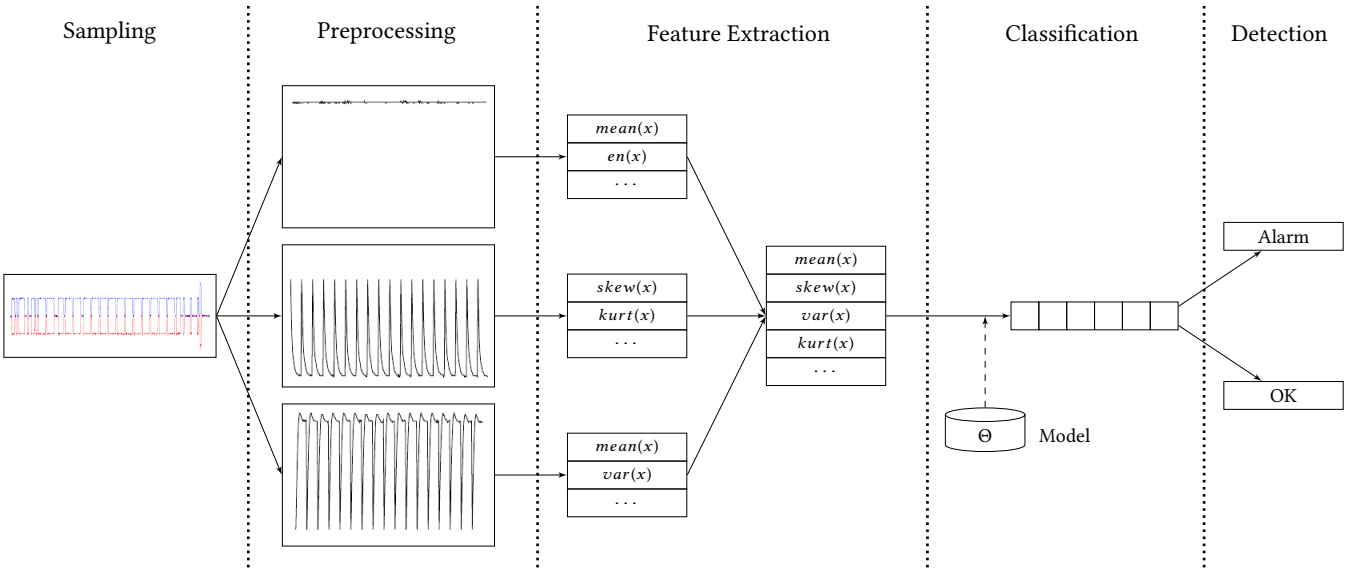


Figure 4: Overview of the ECU fingerprinting phases.

shown in Figure 2 in Section 3. The number of measured values per bit depends on the sampling and baudrate. We sample at a rate of 20MS/s, resulting in 40 measurements per symbol on a 500kb/s CAN. For a discussion on choice of sampling rate, see Section 6.3.

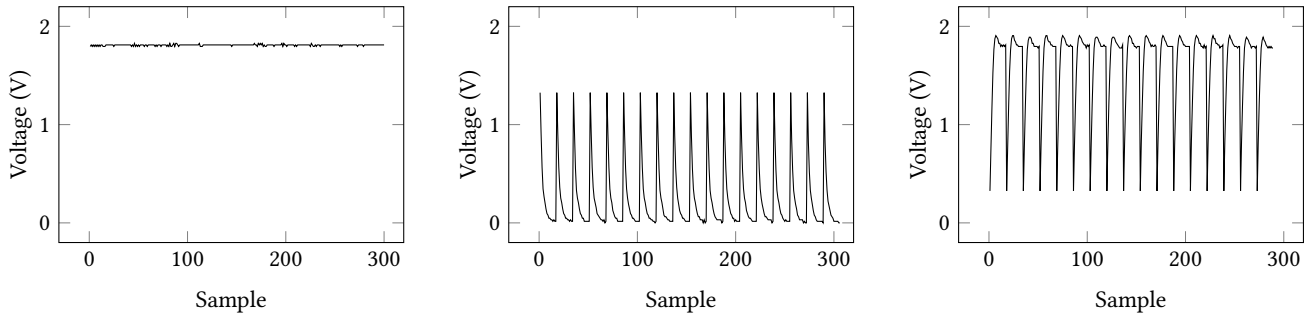
4.2.2 Preprocessing. In this step, the signal of each bit of the message recorded after the arbitration is processed individually. This results in an amount of sets containing several analog values. Due to drifts of the oscillators of individual ECUs, there can be multiple shifts of the edges in one single message. To compensate these displacements and to reduce the amount of data, only a segment of each bit is used. For example if a rising edge occurs, the window starts with the first value that exceeds 1 V.

The resulting sets, representing one single bit, will then be subsequently distributed into one of three groups, dependent on its shape. If a set, representing a dominant bit (0), contains a rising edge it is sorted into the group G_{10}. If it does not contain a rising edge, in group G_{00}. A set of a dominant bit contains a rising edge, if the previous bit was recessive. The sets, representing a recessive bit (1), containing a falling edge are sorted into the group G_{01}. A set of a recessive bit contains a falling edge, if the previous was dominant. The dominant bits, whose previous bits were also dominant, are discarded since these bits are unsuitable for classification. This is the case since all values of a group G_{11} would be almost zero, regardless of the sending ECU. After the preprocessing of the whole received CAN frame, all bits, represented by several measured voltage values, are distributed into the three groups, as shown in Figure 5.

This procedure makes it possible to use all bits after arbitration for identification and to be independent of the transmitted data. In addition, by considering the individual groups separately, it is possible that important characteristics are more observable. If the groups are not considered separately, distinguishable characteristics of the different groups may counterbalance each other. This increases both robustness and accuracy of the system.

4.2.3 Feature Extraction. After the preprocessing of the signal, the system extracts statistical features for each of the previous prepared groups. Since different features are relevant for the groups, we have analyzed these individually. Several statistical characteristics (see Table 2) of the training material from each setup were evaluated. In addition to the time domain, the magnitude values from the frequency domain were also considered. The general usability of these features for sender identification has already been shown [6]. In this work, the significant difference lies in the consideration of these features in relation to the individual groups. If features are calculated for a longer sequence of different symbols, significant characteristics could get hidden. This can be illustrated very well by the example of the mean. For this purpose, the characteristic for a sequence consisting of three symbols was calculated for two different ECUs of the Fiat 500. Each sequence is composed by a symbol from each group and are shown in Figure 6. The differences between the two curves, like the lower voltage level and the overshoot of ECU 1, are clearly visible. But this is not significantly reflected in the mean values of the curves with 1.286 V for ECU 0 and 1.285 V for ECU 1. However, if the mean values for the three symbols are calculated individually, a larger difference is visible. For ECU 0 these are $\mu(G_{10}) = 1.623\,V$, $\mu(G_{00}) = 1.947\,V$ and $\mu(G_{01}) = 0.289\,V$ and for ECU 1 $\mu(G_{10}) = 1.691\,V$, $\mu(G_{00}) = 1.89\,V$ and $\mu(G_{01}) = 0.275\,V$. By analyzing the symbols respectively the groups separately, the voltage level ($\mu(G_{00})$) and a potential overshoot ($\mu(G_{10})$) can be considered separately. This might not be the case in a sequence and would prevent the overshoot from being included in the classification, which is a particularly important characteristic.

For the selection of the features we have used the Relief-F [26] algorithm from the Weka 3 Toolkit [39]. The Relief-F algorithm is a filter method to calculate a score for each feature, which can be used to rank and select the most significant features. We have done a feature selection for each test setup. Due to the strong dependencies on the whole network, there are slightly different results in the



(a) Group 00 containing 17 symbols (b) Group 01 containing 18 symbols (c) Group 10 containing 17 symbols

Figure 5: The three groups after preprocessing the frame of the Fiat shown in Fig. 1.

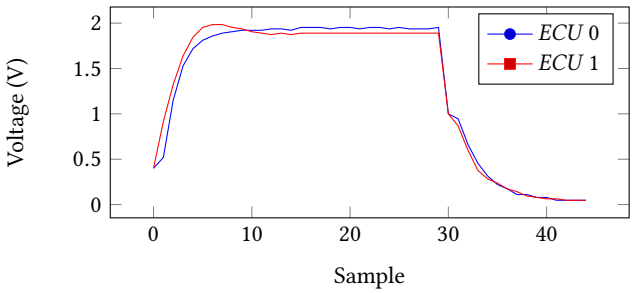


Figure 6: Sequence of different symbols from ECU 0 and ECU 1 from the Fiat 500.

selection of the best features. Therefore we have combined the best features of the test setups to get a general feature set, shown in Table 3 with their respective rank. However, the evaluation clearly showed that the most important characteristics can be found in G_{10}, which contain the rising edges. Altogether, this results in a feature vector $F(V)$, which represents the fingerprint extracted from the received CAN signal.

4.2.4 Model Generation and Classification. The identification from which ECU a received frame has been sent is a classification problem. Therefore several machine learning algorithms can be used, which are designed to identify to which class a new observation belongs. In our system we use Logistic Regression, to predict the sender of a frame. The main advantage is its simplicity, which plays a significant role in implementation on ECUs with limited resources. This applies to both training and classification.

Before it is possible to predict the origin of a received CAN frame, it is necessary to train the classification algorithm. To achieve this, we generate fingerprints of multiple CAN frames for each of the I ECUs considered. The resulting fingerprints are then used together to train the classifiers Θ^i for each ECU $i \in 1, \ldots, I$. After this supervised learning phase, the classifiers can be used to compare the features of the newly received frames with the features collected for model generation.

4.2.5 Deployment and Lifecycle. It is important to avoid forged frames during the training phase as otherwise the system will not

Table 2: Features considered in the selection, where x are the measured values in the time domain respectively the magnitude values in the frequency domain and N is the number of elements.

Feature	Description
Mean	$\mu = \frac{1}{N}\sum_{i=1}^{N} x(i)$
Standard Deviation	$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x(i)-\mu)^2}$
Variance	$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x(i)-\mu)^2$
Skewness	$skew = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x(i)-\mu}{\sigma}\right)^3$
Kurtosis	$kurt = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x(i)-\mu}{\sigma}\right)^4$
Root Mean Square	$rms = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x(i)^2}$
Maximum	$max = max(x(i))_{i=1\ldots N}$
Energy	$en = \frac{1}{N}\sum_{i=1}^{N} x(i)^2$

Table 3: Selected features for classification ordered by their rank.

1 $max(G_{10})$	**2** $en(G_{10}^{FFT})$	**3** $en(G_{00})$	**4** $max(G_{00})$	**5** $\mu(G_{10})$	**6** $\mu(G_{00})$
7 $max(G_{10}^{FFT})$	**8** $\mu(G_{10}^{FFT})$	**9** $skew(G_{10})$	**10** $kurt(G_{10}^{FFT})$	**11** $kurt(G_{10})$	**12** $\sigma^2(G_{10}^{FFT})$
13 $skew(G_{10}^{FFT})$	**14** $skew(G_{01})$	**15** $kurt(G_{01})$	**16** $skew(G_{01}^{FFT})$	**17** $kurt(G_{01}^{FFT})$	**18** $\sigma^2(G_{10})$

be able to reliably detect intrusions. Therefore we consider the vehicle to be in a safe environment, e.g. in the factory or in an authorized workshop, during the initial deployment phase. In this process, which is triggered by a protected diagnostic access, a key is assigned to each ECU to enable secure communication with the IDS. Afterwards, a safe training phase is carried out, in which only authenticated frames are used. The cryptographic algorithms required for this are available within the AUTOSAR module SecOC [35].

As fluctuations of the signal characteristics are to be expected, e. g. due to ageing of the components or corrosion, the learned model has to be adapted to this concept drift [9]. For this purpose,

we propose a performance monitor which continuously evaluates the quality of the classifiers in order to initiate a secure training if the frames can not longer be distinguished with high probability. This allows the model to be constantly adapted to changes, ensuring consistently high accuracy. In order to update the existing model, stochastic [2] or mini-batch algorithms can be used, which require less computational effort than a complete training. It would also be possible to use online machine learning methods [19], which are continuously adapted to the current characteristics.

Using classical cryptography during the lifecycle would basically have the same problem of limiting bandwidth as described in Section 3, but these messages would not be used continuously and without hard real-time requirements. This makes the calculations realistic even for ECUs with less resources and no additional hardware accelerators.

If cryptographic algorithms cannot be used, additional countermeasures should be used to prevent poisoning attacks [7, 18, 24]. These procedures reduce the influence of potential malicious data during the training phase.

4.3 Intrusion Detection using Fingerprints

In this section we describe how the probabilities of the classification can be used to detect the threat models described in Section 4.1.

4.3.1 Detecting Compromised ECUs. Based on the trained classifiers, the system estimates a probability for each ECU when a new frame is received. The easiest and most intuitive way is to select the ECU with the highest probability as the source of the frame. If the ECU selected as source is not allowed to send frames with the identifier of the received frame, an attack will be assumed. The system is aware of the information about which identifiers are used and which ECUs are allowed to use them, since the communication flow of a vehicle network is predefined.

Choosing the most likely ECU has the disadvantage that if the system receives a single frame whose origin cannot be determined with high probability, a false positive is more likely to occur. To avoid this disadvantage, we have set a threshold t_{max} which must be exceeded by the highest probability. Only if the probability of the selected ECU is higher than this threshold and the controller is not allowed to use the given identifier, an attack is assumed. Otherwise, the frame is marked as trustworthy.

To increase the robustness against outliers and electromagnetic interferences and to reduce computational effort, we introduce another threshold t_{min}. The first step is now that the system calculates the probability of the ECU being allowed to send frames with the specified identifier. Here, only a single ECU is possible as CAN does not allow multiple transmitters to use the same identifier. If the estimated probability of the selected controller is below the threshold t_{min}, the frame is marked as suspicious. In this case, the probabilities of the remaining ECUs are calculated by selecting the ECU with the highest probability as the expected source of the frame. The frame marked as suspicious is classified as malicious if the probability of the suspect device exceeds the threshold t_{max}, causing an alarm. The selected ECU also represents the ECU from which the attack is executed. If the probability does not exceed the threshold t_{max}, the frame is considered trustworthy to reduce false positives. It should be noted, that the frames which are suspicious but do not trigger an alarm are used for further detection methods.

It is necessary to keep false positives as low as possible, since these false alarms can trigger unnecessary interventions or, depending on the reaction, even endanger the safety of the passengers [16]. For this reason, and since we assume that the number of trustworthy frames is much higher than the number of malicious frames, we use both thresholds, even if they can lead to a higher false negative rate. Furthermore, the execution of attacks often requires several forged messages, which increases the likelihood of detection. Another advantage is that for most frames fewer calculations are necessary, since the probabilities of all ECUs only have to be calculated if the frame is marked as suspicious.

4.3.2 Detecting Unmonitored ECUs. If the fingerprint of the unmonitored ECU from which the attack is executed is similar to one of the other ECUs that are not allowed to use the received identifier, the attack is detected as in the previous section. The exception is that it is not possible to make a prediction about the source of the attack. In the unlikely situation that the unmonitored ECU has very similar characteristics to the ECU that the attacker wants to imitate, Scission cannot detect the attack. In the third case, if no ECU could be assigned, the frame is marked as suspicious.

In order to detect unmonitored and additional ECUs, Scission monitors the number of frames that are labeled as suspicious. For the detection of these attacks, we use a counter $c_{suf,i}$ for each ECU i, initialized with zero, which becomes incremented when a frame is marked as suspicious. Each time the system receives a frame that can be directly predicted as normal behavior, the counter of the corresponding ECU is decreased. If the counter $c_{suf,i}$ exceeds the threshold t_{suf}, an attack is assumed.

4.3.3 Detecting Additional ECUs. The detection of additional ECUs is very similar to the detection of unmonitored devices. The big difference, however, lies in the fact that connecting an additional device leads to a significant change in the topology. Since the bus structure has much more influence on the fingerprint chanracteristics than the ECU itself, as mentioned in Section 2.2, the calculated probabilities are not very likely to exceed the thresholds. The changes thus lead to a direct and massive increase of all counters $c_{suf,i}$, since all ECUs are affected.

4.3.4 Detecting Scission-aware Attacker. If a Scission-aware remote attacker can influence the voltage level of all ECUs, this leads to a measurable loss in performance of the overall identification accuracy. If the accuracy could be influenced quickly and significantly the system may not be able to identify intrusions until the model is adapted, which could be exploited to perform an attack. In order to impersonate a specific ECU, an attacker may influence just its own voltage level, by heating or cooling up the compromised ECU. Since both changes do not appear directly but steadily, especially the draining of the battery, the system is continuously able to adapt to the slightly changing conditions. In addition, a change of the absolute voltage level will not affect the general shape of a signal, which makes it even more difficult to mislead the identification. Since Scission uses several characteristics, it is unlikely for an attacker to impersonate a specific ECU. Even if the general shape is very similar, an attacker is not able to precisely adapt its signal,

due to the absence of general information about the characteristics. Altogether including the secure learning phase, we consider our system to be able to detect even a Scission-aware attacker.

5 EVALUATION

In this section, we evaluate Scission on a prototypical setup, a Fiat 500 and a Porsche Panamera S E-Hybrid. First of all, we show that our fingerprinting approach is able to identify the senders of received CAN frames with a high probability. Afterwards, we evaluate the ability of our Intrusion Detection System to identify compromised, unmonitored and additional ECUs based on fingerprints.

For the recording of the signals we used the digital storage oscilloscope PicoScope 5204 with a sampling rate of 500 MS/s and a resolution of 8 bits. Two measurement series were created per frame, one for CAN low and one for CAN high, which were then combined to obtain the differential signal. Since the use of suitable hardware for a sampling rate of 500 MS/s is very complex and therefore expensive, we have reduced the measurements to a sampling rate of 20 MS/s. This makes the applicability of this approach for use in the automotive field much more realistic.

The analysis and evaluation software is completely written in Python and uses the Scikit Machine Learning and SciPy library. This also made it possible to use the recorded frames directly in Python. For each of the following tests we have trained Scission with the first measured 200 frames per ECU. The training size was determined on the basis of the learning curve of the validation set.

5.1 Fingerprinting ECUs

5.1.1 Prototype. The first setup on which we evaluated our approach was a CAN prototype consisting of five Arduino Unos. These were each equipped with two CAN shields, on which an MCP2515 [21] controller and an MCP2551 [22] transceiver were installed. The wiring of the prototype is shown in Figure 7.

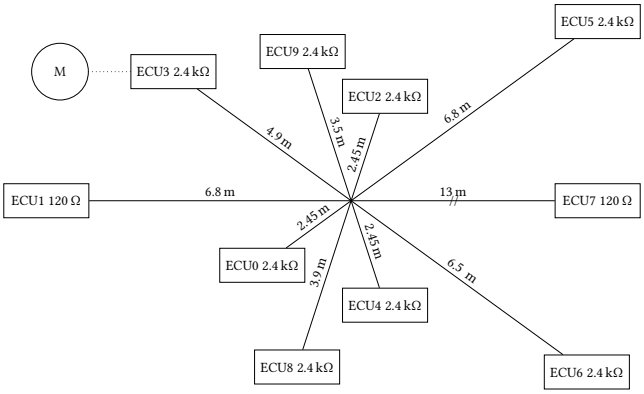


Figure 7: CAN topology of the prototype setup.

For the prototype we have used twisted wires as used in real vehicles. The bus was terminated at each end with 120 Ω resistors and at the stubs with 2400 Ω resistors, as it is often done to reduce reflections. In many vehicles, the bus is not realized exactly according to the specification, as short stub lengths would lead to higher costs. Longer stubs can lead to more and stronger reflections, which can be minimized by additional resistors at the stubs. The ECUs are all supplied by the same power source, which leads to less distinct characteristics. The oscilloscope has been connected directly in front of ECU 3.

We have recorded 56560 frames with normal and extended identifiers. For the prototype, we took random payloads to show the possibility to generate the fingerprint independently of the transferred data.

The confusion matrix is shown in Table 4. Each column shows which percentage of the received frames could be correctly identified using the fingerprint as described in Section 4.2. It can be seen that the messages from the prototype are detected with a mean identification rate of 99.9 % and a minimal identification rate of 99.58 %. It is particularly noticeable that the structural symmetries between ECU 0, ECU 2 and ECU 4 do not have any effects.

Table 4: Confusion matrix for the identification of ECUs of the prototype setup.

	ECU 0	ECU 1	ECU 2	ECU 3	ECU 4	ECU 5	ECU 6	ECU 7	ECU 8	ECU 9
ECU 0	**100**	0	0	0	0	0	0	0	0	0.42
ECU 1	0	**100**	0	0.29	0	0	0	0	0	0
ECU 2	0	0	**100**	0	0	0	0	0	0	0
ECU 3	0	0	0	**99.71**	0	0	0	0	0	0
ECU 4	0	0	0	0	**100**	0.18	0	0	0	0
ECU 5	0	0	0	0	0	**99.82**	0	0	0	0
ECU 6	0	0	0	0	0	0	**100**	0	0	0
ECU 7	0	0	0	0	0	0	0	**100**	0	0
ECU 8	0	0	0	0	0	0	0	0	**100**	0
ECU 9	0	0	0	0	0	0	0	0	0	**99.58**

5.1.2 Fiat 500. The first real car on which we have evaluated our approach was a Fiat 500, its CAN bus has six internal ECUs with up to seven identifiers each. In order to send counterfeit messages for the evaluation of the system, we have connected two Raspberry Pi 3 equipped with CAN shields to the bus. One Raspberry Pi, referred to as ECU 6, has been connected to the OBD-II port and the second Raspberry Pi, referred to as ECU 7, has been connected directly to the CAN bus in the trunk of the car. The oscilloscope was also connected to the ODB-II port, over which we recorded 25979 frames. For the Fiat 500, we achieved a mean identification rate of 99.6 % and a minimal identification rate of 98.56 %, as shown in Table 5.

5.1.3 Porsche Panamera S E-Hybrid. The second real car on which we have evaluated our fingerprinting approach was a Porsche Panamera S E-Hybrid. The vehicle has more than five separate CAN buses with several ECUs connected. For our evaluation, we have observed the CAN bus which is used for powertrain with six internally connected ECUs using up to 43 identifiers each. Since this car is much more complex in terms of hybrid system and its various functions, the bus is very highly loaded. An IDS increasing the packet size or reducing bandwidth could be difficult to deploy in such a complex network. As with the Fiat 500, we have connected two Raspberry Pi 3 (ECU 6, ECU 7) in order to increase the number of ECUs. Since the OBD-II port is not directly connected to the CAN bus, we had to connect both Pis as well as the oscilloscope directly

Table 5: Confusion matrix for the identification of ECUs of the Fiat 500

	ECU 0	ECU 1	ECU 2	ECU 3	ECU 4	ECU 5	ECU 6	ECU 7
ECU 0	**99.90**	0	0.10	0	0	0	0	0
ECU 1	0	**99.89**	0	0.04	0	0.97	0	1.44
ECU 2	0.10	0	**99.72**	0	0	0.03	0	0
ECU 3	0	0	0	**99.96**	0	0	0	0
ECU 4	0	0	0	0	**100**	0.21	0	0
ECU 5	0	0	0.18	0	0	**98.75**	0	0
ECU 6	0	0	0	0	0	0	**100**	0
ECU 7	0	0.11	0	0	0	0.03	0	**98.56**

to the CAN bus in the near of the armrest in the front of the car. We have recorded 6389 frames and the confusion matrix is shown in Table 6. As in the previous tests, we achieved a probability of 99.88 % and a minimal identification rate of 99.58 % for the Porsche.

Table 6: Confusion matrix for the identification of ECUs of the Porsche Panamera

	ECU 0	ECU 1	ECU 2	ECU 3	ECU 4	ECU 5	ECU 6	ECU 7
ECU 0	**100**	0	0	0	0	0	0	0.42
ECU 1	0.00	**100**	0	0.29	0	0	0	0
ECU 2	0.00	0	**100**	0	0	0	0	0
ECU 3	0.00	0	0	**99.71**	0	0	0	0
ECU 4	0.00	0	0	0	**100**	0.18	0	0
ECU 5	0.00	0	0	0	0	**99.82**	0	0
ECU 6	0.00	0	0	0	0	0	**100**	0
ECU 7	0.00	0	0	0	0	0	0	**99.58**

5.1.4 Summary. The evaluations of the three setups have shown that our fingerprinting approach is able to identify the sender of a frame with high probability. The results shown have been originated by selecting the ECU with the highest probability. Compared to the work of Choi *et al.* [6], with an accuracy of 96.48 %, we have achieved a higher mean success rate of 99.85 %. In addition, we used a 125 times lower sampling rate and needed less computational effort by using a simpler machine learning algorithm. We have also shown that our fingerprinting approach is able to identify the sender of a frame without using a specific CAN field that would reduce the available bandwidth.

5.2 Detecting Compromised ECUs

In the previous section we have examined the ability of the fingerprinting approach to identify the sender of received frames. The achieved success rate of 99.85 % when using the fingerprints directly would imply a false positive rate of 0.15 %. This would lead to a high number of false alarms if the bus is observed for a longer period of time.

In the following, we assess the ability of the proposed threshold method described in Section 4.3 to reduce false positives. The configuration of the parameters t_{max} and t_{min} was determined using the respective validation set.

Table 7: Confusion matrix of the IDS.

		Predicted		Suspicious
		No attack	Attack	Frames
Prototype	**No attack**	100	0	0
	Attack	1.5	98.5	0.2
Fiat	**No attack**	100	0	0.01
	Attack	0	100	0
Porsche	**No attack**	100	0	0.01
	Attack	3.18	96.82	3.18

5.2.1 Prototype. For the prototypical setup, two Arduinos have been programmed to send forged messages using the 10 normal and 10 extended identifiers. This way, we have captured additional 51 valid and 467 counterfeit frames. As shown in Table 7 we have achieved a detection rate of 98.5 % for the prototype setup with $t_{max} = 0.6$ and $t_{min} = 0.2$. Compared to the simple fingerprinting, the threshold approach reduces the false positives to 0.

5.2.2 Fiat 500. For the Fiat 500 we have sent 1230 counterfeit frames from the two Raspberry Pis (ECU 6 and 7). Unfortunately, we had no possibility to manipulate one of the existing ECUs (0-5) to send counterfeit frames from these. The results for the Fiat 500 are shown in Table 7. As observed in the former test, the ECUs of the Fiat are highly distinguishable what leads here to a high detection rate of the forged messages by having a false positive rate of 0 with $t_{max} = 0.7$ and $t_{min} = 0.2$.

5.2.3 Porsche Panamera S E-Hybrid. Since we could not estimate the consequences of faking highly critical messages of the powertrain CAN, we decided to send only counterfeit messages for ECU 6 and 7. Here, 157 counterfeit frames have been recorded. As these two ECUs are identical in construction and have been connected to the bus at the same point, their signals are very similar. This situation shows that the threshold approach can lead to an increased false negative rate. On the other hand, the false positives could also be reduced to 0. Scission has been configured with $t_{max} = 0.6$ and $t_{min} = 0.2$.

5.2.4 Summary. The results obtained show that the proposed IDS is still able to detect counterfeit frames by having a false positive rate of 0 %. Even if we did not observe false positives during our evaluation, this does not mean that no false positives can occur. The life of a vehicle is much longer than the hours in which we took our measurements.

Regarding the frames marked as suspicious, we see further potential to increase detection rates, by fine-tuning the thresholds according to the actual vehicle. We also expect better results in real conditions, as there is less interference with professional connectors. By setting up and removing the equipment for sending the counterfeit frames, we could not ensure that the measuring point was not slightly changed, which leads to lower identification rates.

5.3 Detecting Unmonitored ECUs

In this section, we evaluate the ability to detect unmonitored ECUs on the Fiat 500. To do so, we trained the system without knowledge about ECU 7. After monitoring 200 trustworthy frames, we inserted messages from ECU 7 faking messages from ECU 1. If Scission cannot classify a frame with high probability, the counter $c_{suf,1}$ is increased by 4 and otherwise decreased by 1 to a minimum value of 0. If this counter exceeds the threshold t_{suf} of 200, Scission will trigger an alarm.

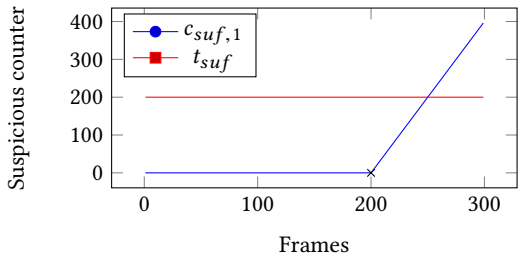


Figure 8: Suspicious counter of ECU 1 from the Fiat 500 during unmonitored ECU attack.

The counter of ECU 1 is shown in Figure 8, which shows that the system is able to detect the attack. If the attack is to be detected more quickly, it is possible to reduce the threshold value or increment the counter with a higher value.

5.4 Detecting Additional ECUs

In the following, we show the ability of Scission to detect additional connected devices. To do so, we initially let the system monitor the unchanged bus, which means that the participants are detected with a high probability. After a few frames, the system switches to frames that were recorded on a modified bus. As a result, the number of frames that are marked as suspicious increases, indicating a significant change on the CAN bus.

Whenever Scission cannot assign a frame with high probability, which means that the frame is considered suspicious, the counter $c_{suf,i}$ is incremented by 4. If a frame can be assigned to an ECU with high probability, the counter is decremented by 1 to a minimum value of 0. As soon as the probabilities decrease sharply and many frames are mistakenly classified as suspicious, the counter will increase rapidly. In order to improve the readability, we have counted all $c_{suf,i}$ values together to the counter c_{suf}. If this counter exceeds the threshold t_{suf} of 200, Scission will trigger an alarm.

5.4.1 Prototype. In order to evaluate the prototypical setup we have recorded normal frames without having ECU 9 connected to the bus. As in the previous scenarios, the oscilloscope was connected directly in front of the CAN controller of ECU 3. For the malicious test set we have used the frames recorded for the previous evaluations with having all ECUs connected to the bus. First of all we have trained the IDS using the normal frames. To simulate an additional connected device, we switched to the malicious test set after the IDS has analyzed 420 normal frames. The course of the counter c_{suf} is shown in Figure 9a. Once the data is changed from the normal test quantity to the malicious one, the curve increases sharply. The threshold has been reached after 53 frames of the malicious bus configuration were observed from the IDS, which resulted in an alarm.

5.4.2 Fiat 500. For the Fiat 500 we have recorded the normal frames without ECU 6 and 7 being connected to the bus. Only the oscilloscope was connected to the bus via the OBD-II port. The recorded data from previous tests having all ECUs connected has been used as a malicious test set. After 390 frames of the normal test set, we have switched to the malicious frames. The course of the counter c_{suf} for the Fiat 500 can be seen in Figure 9b. The additional devices, ECU 6 and 7, have been detected after 77 frames were processed by the system.

5.4.3 Porsche Panamera S E-Hybrid. For the Porsche Panamera we have used a similar setup as for the Fiat 500. In order to evaluate the system on the Porsche, we have taken several frames without the Raspberry Pis (ECU 6, ECU 7) being connected to the bus. These frames have been used as normal test set, which were used for the learning phase and the first step of the evaluation. After processing 370 normal frames we switched to the malicious frame set. It took 79 frames to be processed by the IDS before the attack was detected, as shown in Figure 9c.

5.4.4 Summary. The results of the evaluation show the ability of our system to detect additional connected devices. The counter c_{suf} increases directly after switching to the malicious test set and exceeds the threshold after 70 frames on average have been processed.

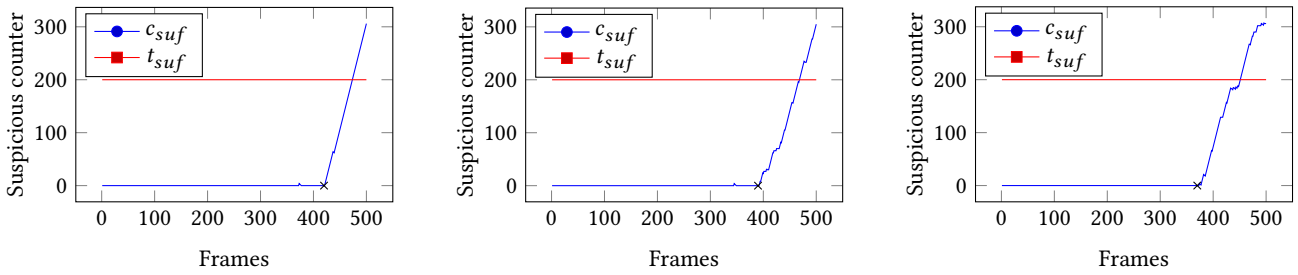
6 DISCUSSION

6.1 Stability

Murvay and Groza [33] have already shown that characteristics remain unchanged over several months. This statement is consistent with our observations. We set up another prototype and observed its characteristics over half a year, which remained stable during the period at laboratory conditions. The hardware and the bus structure is comparable to the prototype setup as used in the evaluation in Section 5.1.1 and consists of 5 ECUs. Here, a BitScope Micro BS05[1] was used for the recording of the signals and a Raspberry Pi 3 for the calculations.

However, since laboratory setups are less complex than real vehicles and thus do not have the same expressiveness, we have also analyzed the behavior of our approach under changing conditions in the Fiat. For this, we recorded additional signals of the Fiat in its original state, as described in Section 5.1.2, via the OBD-II port. The first data set was recorded in a hall at an ambient temperature of about 25°C (77°F), while the vehicle was switched off and cold as it was not moved for at least one day. From this set, the first 200 frames of each of the 6 internal ECUs were used for the training of the classifiers. From the same state, another 3369 frames were recorded and classified correctly without exception. The recording of the second data set was started after the cold start of the vehicle and contains a trip of approximately 30 minutes at an average ambient temperature of 32°C (89.6°F). After the vehicle was completely heated, the recording was finished and the vehicle was parked

[1] http://bitscope.com/product/BS05/



(a) Prototype (b) Fiat 500 (c) Porsche Panamera

Figure 9: Suspicious counters during additional ECU attack.

in an underground garage at ca. 23°C (73.4°F). Also this data set, consisting of 6672 frames, was correctly classified by the already trained classifiers, without requiring a relearning. After 3 hours of cooling, the recording of the third set of data was started while the vehicle's engine was still turned off. After a short time, the vehicle was started and driven outside the garage at approximately 36°C (96.8°F) for another 20 minutes. Also the third data set, consisting of 4863 frames, was classified completely correctly, without requiring any relearning.

The biggest change in terms of characteristics was observed in the voltage level, which dropped from a turned off and cooled down engine to a warmed up engine by ca. 0.012 V to 0.026 V. Our observations also confirm the assumption made in Section 3 that different voltage changes of the ECUs can occur under changing conditions.

6.2 Limitations

Scission can detect compromised ECUs by monitoring CAN communication. It checks whether frames have been sent from ECUs that are allowed to use the corresponding identifier. An attack is detected when an malicious ECU uses an identifier of another ECU. However, if an attacker works with the identifiers that the ECU is allowed to use under normal conditions, Scission cannot detect them.

As shown in the evaluation, Scission can detect the connection of additional ECUs. Since the system uses the characteristics of the CAN signals, which are mainly influenced by the bus structure, the system can only detect modifications if they lead to bus changes. If the bus is modified without influencing the characteristics, the system will not longer be able to reliably recognize the change.

6.3 On the Sample Rate

As mentioned in Section 4.2.3, the most important characteristics for sender distinguishability are present in the rising edges. Thus, we reason the sample rate on G_{10}. In the rising edge, the overshoot is the most noticeable characteristic and clearly recognizable in Fig. 5c. At the same time, this property represents the shortest part of the signal and is therefore significant for the choice of the sample rate. The minimum duration of the overshoot in our analyzed data is approximately 200 ns, which corresponds to a frequency of 5 MHz. According to the Nyquist-Shannon sampling theorem, a sampling with at least 10 MHz is necessary to reliably record the overshoot. Due to a deliberate oversampling with 20 MHz, the course of the curve can be recorded in more detail, providing a more precise calculation of the features. In addition, this offers a margin in case the duration of the overshoot is shorter. In Table 8, the average identification and false positive rates for different sample rates are shown. The values were determined using 1000 frames per ECU of the Fiat data set.

Table 8: Performance for different sampling rates.

Samplerate (MS/s)	1	2	5	10	15	20
Identification rate	88.23	99.57	99.71	99.72	99.85	100
False positive rate	2.94	0.35	0.26	0.17	0.14	0

6.4 Masking unmonitored ECUs

If an unmonitored ECU can send messages to the bus without triggering an alarm based on its fingerprint, the suspicious counter is used for the detection. In this case, an attacker can circumvent the system by infiltrating the counterfeit messages only irregularly, keeping the counter below the threshold. However, this danger exists mainly for aperiodic messages. For periodically sent messages, the attacker must deactivate the actual sender and then take over the sending of the messages. The frequency of the messages must be kept, as additional or missing messages can be recognized with less effort [4, 34].

6.5 Preventing Attacks

Scission can be customized to not only detect attacks, but actively prevent them. This is possible if only a certain number of the first bits are used for the analysis instead of the whole frame. In this way, a decision can be made at the appropriate calculation speed and the transmission can be disturbed, e. g. by invalidating the CRC checksum or sending an error frame. Autonomously acting systems must be considered with extreme care, as wrong decisions can lead to high risks [16, 17]. In the case of safety-critical messages, extensive investigations must be carried out in order to meet the safety requirements.

6.6 Field of Application

If the costs for implementing Scission could be reduced accordingly, it would be possible to place Scission in several ECUs. This allows

ECUs to decide more quickly and independently whether received frames are trustworthy and how to handle them. In addition, the use of several systems offers the possibility of mutual validation at different measuring points, which can further increase the reliability. Since CAN is not only used in vehicles but also in many other areas of application, the use of Scission is also suitable for these domains. These include, for instance, aerospace, automation, medicine and rail systems. In addition, many higher-level protocols such as CANopen or SafetyBUS are based on CAN.

Since CAN does not provide enough bandwidth for many future applications, CAN with a flexible data rate (CAN-FD) [11] was introduced in 2012. This enables an increased user data length from 8 to 64 bytes as well as an increased data transfer rate. In the automotive industry, an average transmission rate of 2.5 Mbit/s can be assumed [13]. Since the functional approach of CAN-FD remains basically the same, Scission can also be used for future buses with this upcoming improvement.

7 CONCLUSION

The usage of IDSs in in-vehicle networks is a promising technology for improving their security. Since CAN does not provide a reliable way to identify the sender of received messages, we have proposed an approach to extract fingerprints from the CAN signal. The evaluations show, that Scission is able to identify the correct sender with a probability of 99.85 %.

In addition to a significant increase in the identification rate, we were also able to reduce the requirements compared to the approach of Choi *et al.* [6], by using Logistic Regression and a 125 times lower sampling rate. Our sampling approach further enables us to deploy Scission without an impact on the available bandwidth. In addition, we have introduced an IDS based on the fingerprints derived from the aforementioned sampling approach which enable us to detect forged messages sent from compromised ECUs with zero false positives during the evaluation.

We have shown that the proposed IDS is also capable of detecting attacks from unmonitored and additional devices. By evaluating Scission on a prototype, a Fiat 500 and a Porsche Panamera, we evaluated and established capability in production vehicles.

Considering that most attacks on cars are successful, as it is not possible to determine whether a received frame is sent from a valid sender, our system can improve the security and thus system safety. Fingerprinting technology can enhance classical IDS approaches, it can further be used as a basis for stand-alone system or improve the security of gateways connecting different buses.

REFERENCES

[1] Stefan Axelsson. 2000. Intrusion Detection Systems: A Survey and Taxonomy.

[2] Léon Bottou. 2010. Large-Scale Machine Learning with Stochastic Gradient Descent. In *Proceedings of COMPSTAT'2010*, Yves Lechevallier and Gilbert Saporta (Eds.). Physica-Verlag HD, Heidelberg, 177–186.

[3] Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno. 2011. Comprehensive Experimental Analyses of Automotive Attack Surfaces. In *Proceedings of the 20th USENIX Conference on Security (SEC'11)*. USENIX Association, Berkeley, CA, USA, 6–6.

[4] Kyong-Tak Cho and Kang G. Shin. 2016. Fingerprinting Electronic Control Units for Vehicle Intrusion Detection. In *25th USENIX Security Symposium (USENIX Security 16)*. USENIX Association, Austin, TX, 911–927. https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/cho

[5] Kyong-Tak Cho and Kang G. Shin. 2017. Viden: Attacker Identification on In-Vehicle Networks. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17)*. ACM, New York, NY, USA, 1109–1123. https://doi.org/10.1145/3133956.3134001

[6] W. Choi, H. J. Jo, S. Woo, J. Y. Chun, J. Park, and D. H. Lee. 2018. Identifying ECUs Using Inimitable Characteristics of Signals in Controller Area Networks. *IEEE Transactions on Vehicular Technology* 67, 6 (2018), 4757–4770.

[7] Martin A. Fischler and Robert C. Bolles. 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* 24, 6 (June 1981), 381–395. https://doi.org/10.1145/358669.358692

[8] Federal Office for Information Security. 2018. TR-02102-1 Cryptographic Mechanisms: Recommendations and Key Lengths.

[9] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)* 46, 4 (2014), 44.

[10] Robert Bosch GmbH. 1991. CAN Specification v2.0.

[11] Robert Bosch GmbH. 2012. CAN with Flexible Data-Rate Specification Version 1.0.

[12] B. Groza and S. Murvay. 2013. Efficient Protocols for Secure Broadcast in Controller Area Networks. *IEEE Transactions on Industrial Informatics* 9, 4 (Nov 2013), 2034–2042. https://doi.org/10.1109/TII.2013.2239301

[13] Florian Hartwich. 2012. CAN with flexible data-rate.

[14] Aaron Higbee. 2007. Hack Your Car for Boost and Power! DEF CON 15 Hacking Conference.

[15] T. Hoppe, S. Kiltz, and J. Dittmann. 2008. Adaptive Dynamic Reaction to Automotive IT Security Incidents Using Multimedia Car Environment. In *2008 The Fourth International Conference on Information Assurance and Security*. ACM, New York, NY, USA, 295–298.

[16] Tobias Hoppe, Stefan Kiltz, and Jana Dittmann. 2008. Security Threats to Automotive CAN Networks – Practical Examples and Selected Short-Term Countermeasures. In *Computer Safety, Reliability, and Security*, Michael D. Harrison and Mark-Alexander Sujan (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 235–248.

[17] Tobias Hoppe, Stefan Kiltz, and Jana Dittmann. 2009. Applying intrusion detection to automotive it-early insights and remaining challenges. *Journal of Information Assurance and Security (JIAS)* 4, 6 (2009), 226–235.

[18] Peter J. Huber. 1992. *Robust Estimation of a Location Parameter*. Springer New York, New York, NY, 492–518. https://doi.org/10.1007/978-1-4612-4380-9_35

[19] Boris Igelnik, Boris Igelnik, and Jacek M. Zurada. 2013. *Efficiency and Scalability Methods for Computational Intellect* (1st ed.). IGI Global, Hershey, PA, USA.

[20] Alberto Garcia Illera. 2013. Dude, WTF in my car? DEF CON 21 Hacking Conference.

[21] Microchip Technology Inc. 2005. MCP2515 Stand-Alone CAN Controller With SPI Interface. Revision D.

[22] Microchip Technology Inc. 2007. MCP2551 High-Speed CAN Transceiver. Revision E.

[23] Tobias Islinger and Yasuhiro Mori. 2016. Ringing suppression in CAN FD networks. CAN Newsletter.

[24] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li. 2018. Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning. In *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, New York, NY, 19–35. https://doi.org/10.1109/SP.2018.00057

[25] B. Jungk. 2016. Automotive security state of the art and future challenges. In *2016 International Symposium on Integrated Circuits (ISIC)*. IEEE, New York, NY, 1–4.

[26] Igor Kononenko. 1994. Estimating attributes: Analysis and extensions of RELIEF. In *Machine Learning: ECML-94*, Francesco Bergadano and Luc De Raedt (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 171–182.

[27] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage. 2010. Experimental Security Analysis of a Modern Automobile. In *2010 IEEE Symposium on Security and Privacy*. IEEE, New York, NY, 447–462. https://doi.org/10.1109/SP.2010.34

[28] Hung-Jen Liao, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-Yuan Tung. 2013. Review: Intrusion Detection System: A Comprehensive Review. *J. Netw. Comput. Appl.* 36, 1 (Jan. 2013), 16–24. https://doi.org/10.1016/j.jnca.2012.09.004

[29] C. W. Lin and A. Sangiovanni-Vincentelli. 2012. Cyber-Security for the Controller Area Network (CAN) Communication Protocol. In *2012 International Conference on Cyber Security*. IEEE, New York, NY, 1–7. https://doi.org/10.1109/CyberSecurity.2012.7

[30] Charlie Miller and Chris Valasek. 2013. Adventures in automotive networks and control units. , 260–264 pages.

[31] Charlie Miller and Chris Valasek. 2015. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA* 2015 (2015), 91.

[32] H. Mori, Y. Suzuki, N. Maeda, H. Obata, and T. Kishigami. 2012. Novel ringing suppression circuit to increase the number of connectable ECUs in a linear passive star CAN. In *International Symposium on Electromagnetic Compatibility - EMC*

EUROPE. IEEE, New York, NY, 1–6. https://doi.org/10.1109/EMCEurope.2012.6396876

[33] P. S. Murvay and B. Groza. 2014. Source Identification Using Signal Characteristics in Controller Area Networks. *IEEE Signal Processing Letters* 21, 4 (April 2014), 395–399. https://doi.org/10.1109/LSP.2014.2304139

[34] Michael Müter, André Groll, and Felix C. Freiling. 2010. A structured approach to anomaly detection for in-vehicle networks. In *2010 Sixth International Conference on Information Assurance and Security*. IEEE, New York, NY, 92–98. https://doi.org/10.1109/ISIAS.2010.5604050

[35] AUTOSAR Development Partnership. 2016. Specification of Module Secure Onboard Communication.

[36] Sang Uk Sagong, Xuhang Ying, Andrew Clark, Linda Bushnell, and Radha Poovendran. 2018. Cloaking the Clock: Emulating Clock Skew in Controller Area Networks. In *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS '18)*. IEEE Press, Piscataway, NJ, USA, 32–42. https://doi.org/10.1109/ICCPS.2018.00012

[37] Florian Sagstetter, Martin Lukasiewycz, Sebastian Steinhorst, Marko Wolf, Alexandre Bouard, William R. Harris, Somesh Jha, Thomas Peyrin, Axel Poschmann, and Samarjit Chakraborty. 2013. Security Challenges in Automotive Hardware/Software Architecture Design. In *Proceedings of the Conference on Design, Automation and Test in Europe (DATE '13)*. EDA Consortium, San Jose, CA, USA, 458–463. http://dl.acm.org/citation.cfm?id=2485288.2485398

[38] H. Schweppe, Y. Roudier, B. Weyl, L. Apvrille, and D. Scheuermann. 2011. Car2X Communication: Securing the Last Meter - A Cost-Effective Approach for Ensuring Trust in Car2X Applications Using In-Vehicle Symmetric Cryptography. In *2011 IEEE Vehicular Technology Conference (VTC Fall)*. IEEE, New York, NY, 1–5. https://doi.org/10.1109/VETECF.2011.6093081

[39] Tony C. Smith and Eibe Frank. 2016. *Statistical Genomics: Methods and Protocols*. Springer, New York, NY, Chapter Introducing Machine Learning Concepts with WEKA, 353–378. http://dx.doi.org/10.1007/978-1-4939-3578-9_17

[40] Anthony Van Herrewege, Dave Singelee, and Ingrid Verbauwhede. 2011. CANAuth - A Simple, Backward Compatible Broadcast Authentication Protocol for CAN bus.

[41] Inc. Vector CANtech. 2003. Common High Speed Physical Layer Problems.

[42] Marko Wolf, André Weimerskirch, and Thomas Wollinger. 2007. State of the Art: Embedding Security in Vehicles. *EURASIP Journal on Embedded Systems* 2007, 1 (19 Jun 2007), 074706. https://doi.org/10.1155/2007/74706

[43] T. Ziermann, S. Wildermann, and J. Teich. 2009. CAN+: A new backward-compatible Controller Area Network (CAN) protocol with up to 16ĂŰ higher data rates.. In *2009 Design, Automation Test in Europe Conference Exhibition*. IEEE, New York, NY, 1088–1093. https://doi.org/10.1109/DATE.2009.5090826