2018 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
MODELING & SIMULATION, TESTING AND VALIDATION (MSTV) TECHNICAL SESSION
AUGUST 7-9, 2018 - NOVI, MICHIGAN

Spoofing Attack on Clock Based Intrusion Detection System in Controller Area Networks

Muhammad Tayyab
University of Michigan
Dearborn, MI

Azeem Hafeez
University of Michigan
Dearborn, MI

Hafiz Malik
University of Michigan
Dearborn, MI

ABSTRACT

The Controller Area Network (CAN) protocol is still a de-facto standard for in-vehicle communication between Electronic Control Units (ECUs). The CAN protocol lacks basic security features such as absence of sender node information, absence of authentications mechanism and the plug and play nature of the network. The payload in a CAN data packet is very small i.e. 8 bytes, therefore, implementation of cryptographic solutions for data integrity verification is not feasible. Various methods have been proposed for ECU identification, one of the methods is clock intrusion detection system (CIDS) [14]. The proposed method is based on authenticating the message sender by estimating the unique characteristics of the clock crystal. In an asynchronous network, the clocking information in a transmitted payload is entirely dependent upon the crystal which invokes the clock. These unique characteristics exists because of the asymmetry in the microstructure of the material. The challenge is to correctly estimate these unique characteristics. Authors proposed the technique to estimate the parameters sufficient enough to prove the clock uniqueness. This method is efficient in detecting the advanced variants of source spoofing attacks. In this paper, we have analyzed the design and architecture of the proposed method and found different weaknesses in the gist of the technique. These weaknesses are present in the mathematical model and can easily be exploited under the same threat models which have been used to evaluate CIDS previously. These vulnerabilities attack the assumption that constructed mathematical model of the sender node at monitoring or receiver is unique. We have proved it that exploiting the proposed vulnerabilities is sufficient to construct the same model for the traffic stream generated from the compromised node. As a proof of concept, we have proposed an attack, clock-spoofing attack, which can be used easily to bypass the CIDS by replicating the clock parameters, hence challenging the assumed uniqueness of clock parameters. Under our observations and the analysis of the found weaknesses, we have concluded that current approach to realize CIDS or similar solutions are defective and are prone to the advanced spoofing attacks. To mitigate against such advanced attacks, we have proposed an idea of authenticating the ECU's based on the immutable unique fingerprints in the electrical signals at physical layer. Electrical devices leave unique fingerprints in the transmitted signal due to natural asymmetry of the material, which can be used to fingerprint the source. As per the CAN transmission at physical level, these are the dominant bits which are transmitted instead of the recessive bits. So, a statistical analysis of the dominant bits in the *received physical signal will contain the unique characteristics of the transmitter which can be used for designing an efficient and highly accurate Intrusion Detection System for CAN Bus. We have discussed this idea briefly.*

INTRODUCTION

The Controller Area Network (CAN) is commonly used networking protocol in the embedded systems' world and can be found in various industries e.g. automotive, aerospace applications and the industrial plants [1]. It is the primary backbone for the in-vehicle communication. A modern vehicle consists of many embedded devices, also known as Electronic Control Units (ECUs), which are responsible for carrying a broad range of functionalities and implementing the advanced features. These functions range from collecting the data from numerous sensors, like Tire Pressure Monitoring System (TPMS), to the controlling different mechanical parts like Engine Control Module (ECM) [2,3]. Many functionalities and the features are possible due to the communication between different ECUs, which is made possible due to the robust and the reliable CAN bus.

CAN is a legacy protocol for embedded networking which was developed in 1980s, hence, lacks the basic security features by the design. The most important of which is the absence of the sender information in the message packet. Moreover, the concise design of packet forbids the implementation of modern source authentication techniques as well. Due to the bus-based topology of the network, every node has the capability to receive and transmit the traffic in network. All these weaknesses combined, lead to a variety of the spoofing attacks [4]. If an ECU on the bus is compromised it can be used to transmit the malicious packets on the bus. Charlie Miller and Chris Valasek could take down the FCA Jeep on the highway by remotely injecting the malicious packets from a compromised node on the CAN bus [2,5,6].

The broadcast nature of the CAN protocol makes it possible for any node to sniff the traffic and then replaying the traffic on the bus [7,8]. Such kind of attacks are still feasible even in the most of modern vehicle if the CAN bus is exposed to the attacker by any means. The small size of the data payload in a CAN packet makes it entirely infeasible for the cryptographic solutions of the message authentication. E.g. the data field in the CAN data packets is 8 bytes but the smallest message authentication code MAC requires at least 16 bytes field which twice as big as the actual data. Implementation of any such solution deteriorates the performance and robustness of the CAN protocol.

The connectivity features like WiFi, Bluetooth, In-vehicle Infotainment Systems etc. are increasing the attack surfaces for the vehicles as well. Keeping in view the CAN compromises, the researchers have proposed many solutions for in-vehicle networks security at different layers e.g. physical layer [14,23] and data link layer by using various types of message authentication methods [7,15,16,22]. Different methods have been proposed by researchers for security of vehicle.

One of the proposed method is Clock based Intrusion Detection System (CIDS), this method is efficient in detecting a broad range of spoofing attacks. In this paper, we have analyzed the architecture of the proposed method and have found the potential vulnerabilities inherent to the design and basic mathematical assumptions. If exploited, these vulnerabilities can be used to fabricate a spoofing attack which can easily bypass the CIDS. Based on our analysis and vulnerabilities, we have concluded that the current modeling of clock characteristics is infeasible and will have the limitations always. Instead, we have briefly overviewed another physical fingerprinting method based on the electrical signals in order to authenticate the message transmitter.

This paper is organized as follows, section II provides an overview of CAN-Bus protocol, section III is an overview of CIDS. In section IV we have discussed the potential weaknesses and proposed attack to bypass these vulnerabilities.

Section V provides our method which is used for identifying the ECUs based upon statistical characteristics.

CAN-BUS PROTOCOL: AN OVERVIEW

The CAN bus consists of the twisted pair cable, as a medium of the signal propagation, known as CAN High and CAN Low. Variants of CAN signaling is used to achieve different data rates. For automotive applications, generally, the CAN bus data rate is set to 500 kbps. At this speed, to transmit bit ’1’, also known as the recessive bit, both the CAN Low and CAN High wires operate at approximately 2.5 volts; while, for the transmission of the ’0’, also known as the dominant bit, CAN High wire is drawn towards the 5V and the CAN Low is drawn towards the 0 volts Figure 1.

The CAN bus is a broadcast-based networking topology where all of the nodes on the bus receive the message. The message packets do not contain the information regarding the sender and the receiver. The message packets are identified by the Arbitration Field also known as CAN ID. This field is used for prioritizing the messages, the lower the ID, the higher is the priority. When different nodes happen to send the message at the same time, the bus arbitration kicks in. In that case, the message with the lower ID is transmitted. Based on the type of the messages, the CAN packets can be classified as the data frame, control frame and error frame. The data frame is the focus of this paper. Shown in Fig. 2 is the structure of the CAN data frame.

The Arbitration Field, also known as ID, is an identifier which used for the identification of the message. Its length is 11-bit (standard CAN protocol) or 29-bit (CAN-extended protocol). The DLC field indicate the length of the data field (in bytes). The DLC field is 3 bit long, therefore, the standard CAN can support a maximum of 8 bytes of the data field. The CAN data packet also includes the CRC field which is used for detecting the transmission errors, e.g., bit flipping.

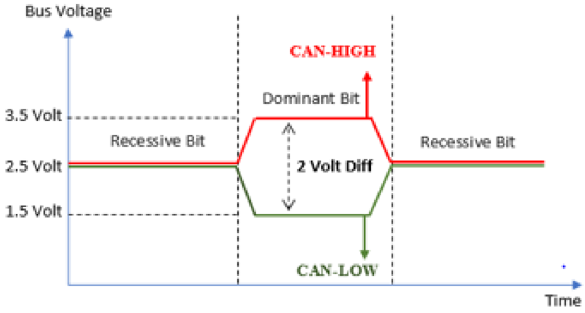


Figure 1: CAN-bus differential signal representations

In in-vehicle CAN network, the messages of a particular ID are transmitted by a certain Electronic Control Unit (ECU). By observing the characteristics of the messages of a specific ID, we can fingerprint the sender ECU of these messages. Researchers have proposed different characteristics and techniques to fingerprint each node in the network and use it to link the CAN packet to the sender node.

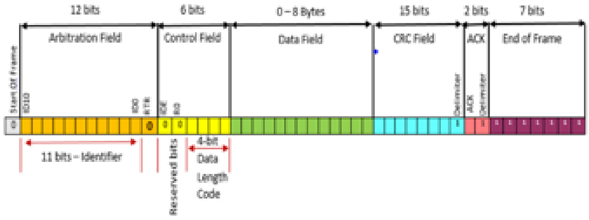


Figure 2: CAN protocol standard format

Researchers have proposed the techniques based on the cryptographic applications as well for securing the traffic in CAN protocol [7,15,16 &23]. But as mentioned, the limited bandwidth and performance overheads in the embedded world of limited resources, are the primary hurdles in adopting these solutions. So, our discussion will be primarily based on the physical fingerprinting-based solutions.

In this paper, we have analyzed the security of one of such proposed methods, known as Clock Based Intrusion Detection System which leverages the uniqueness of clock parameters of the sender node and authenticates the received packet based on the constructed clock model at receiver side using the estimated parameters. Before analyzing the weaknesses of this method, we will be giving a brief overview of CIDS in next section:

Clock Based Intrusion Detection System

Authenticating the sender of the message based on the unique and immutable physical characteristics of the clock crystal of the sender, has been a popular concept in research community. The basis of the idea is that, the clock crystals possess the inimitable physical characteristics due to the micro-structure of the material. These characteristics if carefully parameterized can be used to authenticate the message packets in an asynchronous network. The implementation of such Intrusion detection systems in the complex network architectures like TCP/IP is quite logical since the timestamps are embedded in the message packets and receiver can extract the information about the source clock based on this information.

The real challenge is the embedded networks which are quite simpler and often do not carry any timing information in the message packets. For example, in CAN packet, by design there is no field which carries or can be modified to carry the timing information. This had been a primary hurdle in the implementation of the authentication techniques based on clock characteristics for CAN networks and the embedded networks in general. Recently authors in [Shin] proposed the technique to measure the uniqueness of the clock crystals in a monitoring node in CAN networks to authenticate a packet's sender. The periodicity of the messages is used primarily to extract the parameters which are then used to model the normal behavior of clock for a message of given ID. In case of a spoofing attack situation, the anomaly in the clock behavior is detected to identify the malicious activity in the network.

In this paper, we have analyzed the weaknesses of their proposed method in detail and were able to successfully launch the attack, called clock spoofing. We have demonstrated that the extracted characteristics as proposed are not inimitable in fact and an attacker with the sufficient information who have access to transmit the messages in the network is able to bypass CIDS. Before digging into the details of the attack, we are providing a brief overview of the proposed method in this section followed by the discussion of inherent weaknesses in the design and the attack modelling. We start the overview of CIDS by discussing the clock parameters first.

Parameters of clock

The clock offset is defined as the difference in the reported time by the given clock C_i and an ideal clock C_o at a given time. In real world, the reference clock is taken as the local clock of the receiving node or monitoring node, which is also called as relative offset. Clock frequency is defined as the rate at which the subjected clock advances. Clock skew is the rate of change of the clock offset i.e. the first derivative of clock offset with notation C_i'. In any asynchronous network, the clock offset, and the clock skew is dependent upon the local clock crystal of the sender. This is the core assumption of the proposed Clock based Intrusion Detection System (CIDS).

Given the ability to estimate these parameters, a mathematical model for the clock behavior can be constructed for the messages of given ID and tracked the arrival of the messages. The anomaly in the behavior will be indicative of the suspicious network activity. Apparently, the working of CIDS can easily be dissected into two activities:

1. Clock behavior modelling
2. Anomaly detection

Clock behavior modelling

To model the clock behavior, the first parameter to be estimated is clock offset. Clock offset is the difference in reported time and the local time. But in the absence of the timestamp, a different methodology is adopted which is just applicable for the periodic messages only. From transmitter to the receiver, the total time which is taken by the packet to arrive at the destination is easily be decomposed into below components.

- Propagation delay: d
- Measurement white noise: n
- Offset between the sender and receiver: O_i

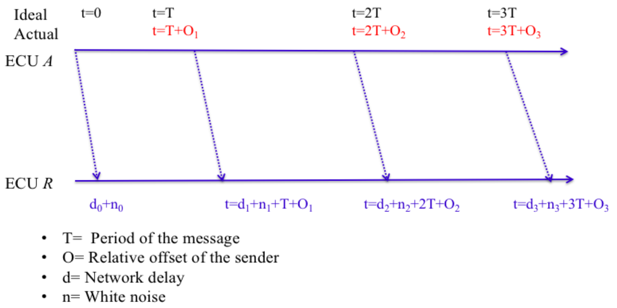


Figure 3: Break Down of the Time from Sender to Receiver

If first message leaves the transmitter at t=0, it will arrive at the receiver on t=d0+n0. Ideally the next message should be transmitted at t=T, T being the periodicity of the message. But, it will include an offset O_1 as well along with the delay and the noise. So, mathematically we can write the arrival sequences in real-world, at times as:

$t_i = O_i + n_i + d + i * T$ …………….. Eq 3.1

Where i denotes the ith message hence t_i being respective arrival timestamp at the receiver. In the ideal scenario, where there is no clock offset and the white measurement noise, we can simplify the equation to be:

$t_{i\ ideal} = d + i * T$ ………………….. Eq. 3.2

Now propagation delay, d, is constant for all messages and we can ignore this factor in both above equations as well.

$O_i + n_i = t_i - t_{ideal}$ ………………… Eq. 3.3

As the n_i is the white noise and its average is going to be 0. We take average of the above value for N messages, which will yield.

$E[O_i] = E[t_i - t_{ideal}]$ ………………..…. Eq. 3.4

Where E[] denotes the average for N messages.

Clock skew is the rate of change of clock offset. But the change between two successive offset values is negligible to compute. To overcome it, authors proposed the idea of accumulation of the absolute values of the offsets and yield a new parameter cumulative offset. This parameter is assumed to increase linearly with time. The slope of this parameter will be an estimation of the skewness. As the computations are performed after every N messages to rule out the estimation and instrumentation noise, let's for k steps the accumulated offset (O_{acc}) can be represented as:

$O_{acc}[k]=S[k]*t+e[k]$ ……………………Eq. 3.5

Where S is the slope which is skewness and e represents the deviation from the constructed parameters. These parameters can be computed using Recursive Least Square Algorithm (RLS).

Anomaly Detection

From Eq 3.5, residual parameter e, computed by RLS tracks the behavior. In case of normal activity, the parameter "e" is relatively small and close to zero. Any large value will be indicating the abnormal network activity. i.e. the subjected messages are being transmitted by the unusual transmitter hence the detection of the spoofing attack scenario. The range of "e" can be set based on the statistical distribution and can be used to set the sensitivity of the detection. Visually, this is equivalent to a hump or a sudden big change in the plotted curve of O_{acc}. This proposed technique has been relatively successful in detecting most advanced variants of the spoofing attacks. The details can be found in the original paper [14]. Here we will be discussing the weaknesses in the approach and then will provide a proof of concept working of the attack. The working of the CIDS can easily be summarized in the following flowchart

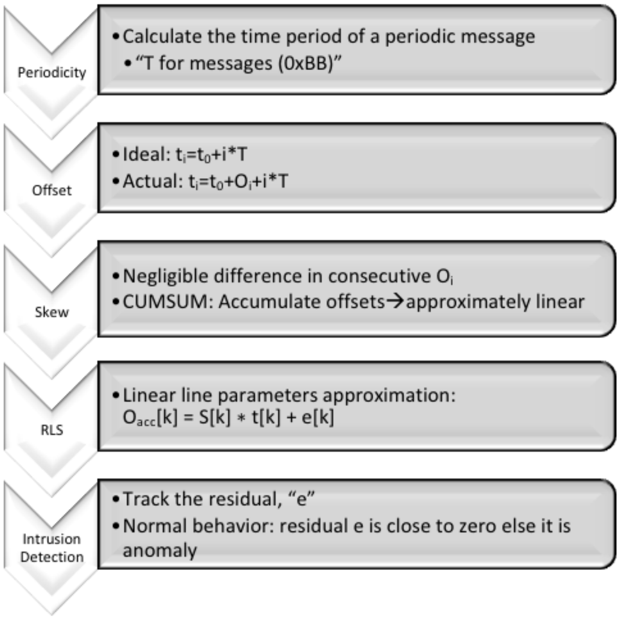

Periodicity
• Calculate the time period of a periodic message
• "T for messages (0xBB)"
Offset
• Ideal: $t_i = t_0 + i*T$
• Actual: $t_i = t_0 + O_i + i*T$
Skew
• Negligible difference in consecutive O_i
• CUMSUM: Accumulate offsets→approximately linear
RLS
• Linear line parameters approximation: $O_{acc}[k] = S[k] * t[k] + e[k]$
Intrusion Detection
• Track the residual, "e"
• Normal behavior: residual e is close to zero else it is anomaly

Figure 4: Working of CIDS

Weaknesses in CIDS

After thorough experimentation and the in-depth analysis of the design of CIDS, we have found out that there are two potential vulnerabilities present in the CIDS by design:

- Parameter dependence on message periodicity
- Non-linearity of the clock skewness

Parameters' Dependence on Message Periodicity:

The essence of the CIDS is that the estimated parameters preserve the sender clock crystal's uniqueness in them. Careful examination of the working of CIDS reveals that the computation structure of the proposed technique is built on the offsets calculation. The offset is computed based on the difference of the actual and expected arrival timestamps on receiver side. It's the actual arrival timestamp which is dependent on the senders' clock vibration. If an attacker can forge this parameter, the parameter computation, hence model can easily be adulterated. Or in another perspective we can say that if the periodicity of the messages is the actual factor which is fingerprinted. This lead us to believe that different messages with distinct periods, even when transmitted from the same sender will have different computed clock behaviors which is directly opposes the theoretical assumption.

As per actual theoretical assumption, that the estimated parameters indicate the uniqueness of the sender's clock, hence, computed models over different periodicities transmitted from same node should have the same or at least similar clock behavior. Our in-depth and careful experiments' results are in contrast with this assumption and follow our intuition i.e. different periodicities will have different clock behaviors even for the same node Figure 5.

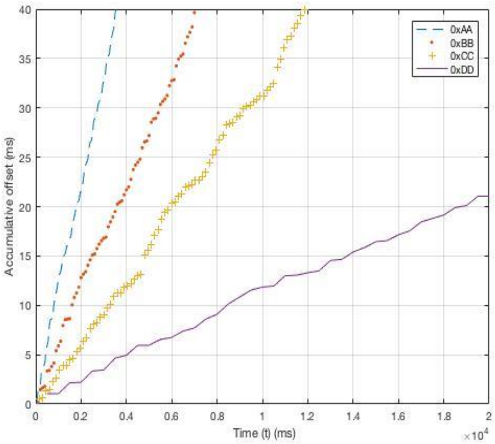


Figure 5: Dependence of clock behavior on period of message. (0xAA: 50ms, 0xBB:100ms, 0xCC:150ms and 0xDD:500ms)

Our observation can be extended to the fact that, if carefully controlled the periodicity, by fraction of the actual period, the clock behavior of the victim node can easily be regenerated. If exploited, this will enable an attacker to launch a spoofing attack against the target messages. Now the question lies, how much this period needs to be adjusted to go un-noticed on the monitoring side. This conundrum is resolved by the next vulnerability.

Non-linearity of the clock skewness

Authors have defined the accumulated offset as the clock behavior and assumed it to be linear. And this linearity is being estimated and observed by constructing a linear model and then estimating the parameters for the linear curve using RLS. The residual parameter yields some smaller values which are close to zero but not absolute zero. The sensitivity of the detection is controlled by setting the limits of residue, statistically. The smaller non-zero values are the proof that the clock behavior is not absolute linear and there is an adulteration of the non-linearity in model which is tolerated by the residual normal operation limits. Visual inspection of the curves also verifies this fact.

If attacker can launch the spoofing attack by setting the period of the transmitted messages such that the resulting residual is in the normal range, the attack is successful. In the next sections we will be discussing the implementation and the calculations required to launch the clock-spoofing attack successfully by exploiting these discussed weaknesses.

Clock Spoofing Attack

Attack Settings

To simplify the discussion and as proof of concept, we are borrowing the attack settings from the original paper [14]. We will be assuming the same working of the test-bed as of Masquerade's attack [14]. Our attack is an extended and improved version of Masquerade's attack. For simplicity, our test-bed consists of three nodes only. One is monitoring node: M, attacker node: A and the legitimate sender (S) which will be our victim. Our testbed is essentially running on Arduino- CAN Shields connected via CAN bus on speed 500 Kbps.

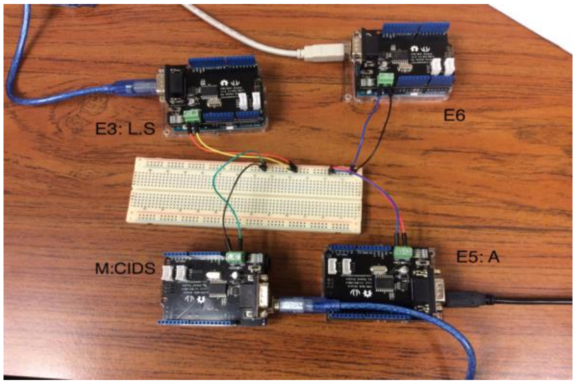


Figure 6: Attack Setting

Monitoring node M fingerprints the legitimate sender L.S. using a CIDS w.r.t to the local clock of M. The computed parameters clock offset and skew O_{LM} and S_{LM} indicates the difference of clock L.S. w.r.t to clock M. Our assumption here is that attacker is also listening to the traffic in initial phase and is running another local copy of the CIDS to observe the differences of Legitimate Sender's clock w.r.t. the clock of Attacker (A). Let's say attacker computes the parameters locally O_{LA} and S_{LA} which are the difference of Legitimate Sender's clock w.r.t. the attacker's clock. The attacker has now the information about how much different his clock from the victim's clock is. Using these differences, if the attacker slightly adjusts the periodicity of his traffic, he will be generating the clock parameters same as the victim's clock. Which will not be distinguishable by monitoring node M. The major architecture of the clock spoofing attack is like the Masquerade's attack [14]. Our attack model assumption is that the attacker has compromised an arbitrary node in the network with the ability to transmit the target messages. While the attacker also can stop the victim messages from the legitimate sender's node as well. Attacker synchronizes the two attacks in such a way that when the attacker stops the legitimate transmission of messages, he starts injecting the malicious traffic with the similar but adjusted period as per above calculations, he will be successfully bypassing the CIDS.

Attacking CIDS

At the attacker side, before transmitting the malicious traffic into the network, the attacker observes the traffic by legitimate sender and then tries to learn the relative parameters of the legitimate sender's node. The goal of the attacker is to adjust the periodicity of the malicious traffic such that the computed clock parameters on the monitoring node should lie in the normal behavior range. As mentioned before the relative parameters O_{LA} and S_{LA} indicates the relative difference between the legitimate sender node and the attacker node. The offset O_{LA} is a mere time difference between these two nodes i.e. attacker node and the legitimate sender node. If the attacker adds O_{LA} to the observed period T_{LA}, it will be just same as the legitimate sender's clock.

$$O_{LA} = t_L - t_A$$
$$t_L = t_A + O_{LA}$$

However, due to clock skew S_{LA} we have a changing O_{LA}, which can be adjusted to,

$$O_{LA}*(1+S_{LA})^{i.}$$

For transmitting the messages of time TLA, we can make use of the above equation and incorporate the clock skew as well. We can have an estimate of the actual timestamps which are expected to be arrived. Hence attacker can transmit on exactly those timestamps to yield the same impact on the mathematical model on monitoring node M. Or we can have the equivalent time periods for subsequent i messages as well for the attacker to transmit:

$$T_{iA} = T_{LA} + O_{LA} * (1 + S_{LA})^i$$

Once the attacker has learned the sufficiently accurate measurements, he can launch a suspension attack on the legitimate sender L.S. and then start transmitting the traffic according to the updated time periods as shown.

In the monitoring side, we observe the impact of the malicious traffic received because of the compromised period T_{iA}. Let's say the updated parameters on the malicious traffic are O_{AM}' and S_{AM}'. It is mathematically straightforward to show that these are in fact equal to the parameters computed on the legitimate sender's node.

$$O_{AM}' = O_{AM} + O_{LA} = t_A - t_M + t_L - t_A = t_L - t_M = O_{LM}$$

As we have shown the offset for the malicious traffic will be same as the legitimate sender. But with time, the offset will change, and the monitoring node still can observe the deviation. To counter it, attacker is already updating the offset based on the L.S. relative skew, hence, all the subsequent offsets will also be like legitimate clock. Which in turn will yield the same skew as the legitimate clock. We have mathematically shown that, the attacker will be able to generate the clock behavior of the target node exactly same as the legitimate sender's node.

Experimental results of bypassing CIDS

On our test-bed, explained in previous sections, we successfully tested the implementation of CIDS and then launched our clock spoofing attack. We compared the performance of our attack with the discussed attack “Masquerades Attack” in [14]. Masquerades attack is a simple attack where the attacker can suspend the transmission of the legitimate sender and starts injecting the traffic at the same period. The attack was tested on message with ID 0xAA with time period 50ms. We can see at t=1000ms, we launched the Masquerades attack which resulted in a sudden change of the clock behavior and is visible in the plot as well. However, when the clock phisher, (may be called as Advanced

Masq.), is launched at t=1000ms, there is no change in the clock behavior and CIDS is bypassed successfully Figure 7.

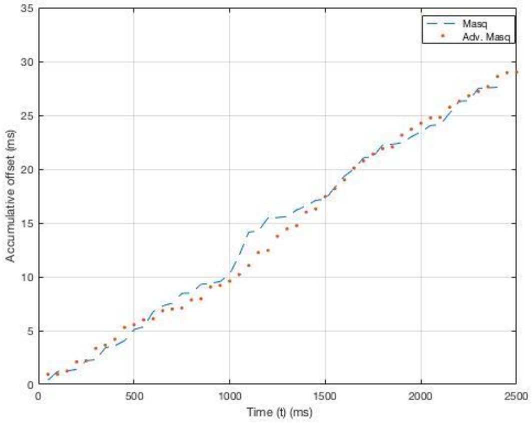


Figure 7: Clock Spoofing Attack at t=1000ms on message 0xAA (Period: 50ms)

Proposed Method: Sender Authentication Based on Physical Fingerprinting

As discussed in the previous section that the CIDS can be compromised by generating the same clock as the sender's clock. This section proposes another method which can be used along with CIDS to enhance security of vehicle. The proposed transmitter identification method relies on the fact that each electronic device (e.g. ECU) exhibit unique artifacts which can be used for linking received signal to the sending ECU. More specifically, by extracting the distinguishable statistical features of transmitting signals to identify the source of the coming message. Let $S_i(t)$ be the output of i^{th} ECU and $h_j(t)$ be the impulse response of the physical channel between i^{th} ECU and the physical fingerprinting (PhyFin) unit. The physical signal at the input of the PhyFin unit, $y_{ij}(t)$, can be expressed as Equation 1 and figure 3, respectively.

$y_{ij}(t)= h_j(t)* S_i(t)$ ……………….Eq 4.1

where, * denotes convolution operator.

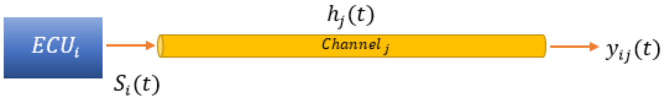


Figure 8: Physical input signal and channel response

In this study, channel is same so the effect of $h_j(t)$, is assumed to be constant. The effect of Physical signal at the input of PhyFin unit, $y_{ij}(t)$ is used for linking $y_{ij}(t)$ to its source. Shown in figure 4 are plots of four waveforms as output of four different ECUs when identical message is applied at the input.

It can be observed from figure 4 that artefacts introduced by each ECU is different for all four ECUs even when the input is kept same, which validates our claim of channel specific uniqueness. In CAN Protocol, only the dominant bits are transmitted actually, so, it is safe to assume that if only the dominant bits are observed, we can fingerprint the sender, as the dominant bits are the one where the fingerprints of senders are imparted only.

Various feature extraction methods in time. To validate effectiveness of the proposed method here, feature extraction method presented in [17] is considered. Details of the selected time-domain features are shown in Table I. The feature selection process resulted in an 8-D feature vector for channel and ECU identification.

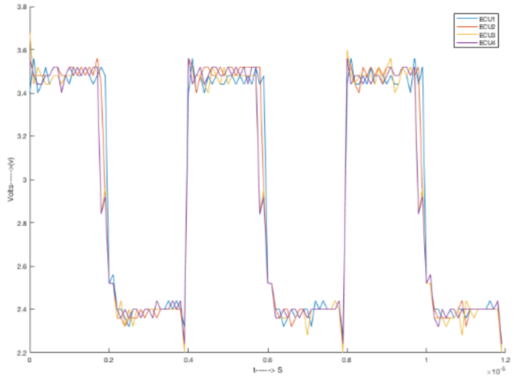


Figure 9: Waveforms of the received signals from four different ECUs with identical channel input message

TABLE I. TIME DOMAIN FEATURE SET

Feature name	Equation
Maximum	$m_{ij} = (\mathrm{Min}(y_{ij}(i)) \mid i=1\ldots N)$
Minimum	$M_{ij}=(\mathrm{Max}(y_{ij}(i)) \mid i=1\ldots N)$
Mean	$\mu_{ij}= \frac{1}{N} \sum_{i=1}^{N} y_{ij}(i)$
Variance	$\sigma_{ij}^2=\sqrt{\frac{1}{N-1} \, \Sigma_{i=1}^{N} y_{ij}(i)-\mu_{ij}}$
Skewness	$\rho_{ij}= \frac{1}{N} \sum_{i=1}^{N} \left(\frac{y_{ij}(i)-\mu_{ij}}{\sigma_{ij}}\right)^3$
Kurtosis	$\kappa_{ij}= \frac{1}{N} \sum_{i=1}^{N} \left(\frac{y_{ij}(i)-\mu_{ij}}{\sigma_{ij}}\right)^4 -3$
5th moment	$\frac{1}{N} \sum_{i=1}^{N} \left(\frac{y_{ij}(i)-\mu_{ij}}{\sigma_{ij}}\right)^5$
6th moment	$\frac{1}{N} \sum_{i=1}^{N} \left(\frac{y_{ij}(i)-\mu_{ij}}{\sigma_{ij}}\right)^6$

EXPERIMENTAL SETUP AND DATASET DESCRIPTION

Four ECUs of same make and model are used in this study. Overall, the experimental setup contains following hardware and software components:

- Four (4) Arduino Uno R2 microcontroller kits
- Four (4) CAN-Bus shield board with MCP2515 CAN-bus controller and MPC2551 CAN transceiver.
- Oscilloscope DSO1012A for the voltage samples recording with Sampling Rate of 2GSa/s, 100MHz bandwidth, and 8-bit vertical resolution.
- Script for sending an identical message continuously from different channels and ECUs to observe the unique patterns of signals from each channel and ECU.
- MATLAB R2016a software for statistical data analysis of sampled signals.

Performance of the proposed algorithm is evaluated for ECU classification. To this end, physical signal is captured at the output same cable with 1 meter and four identical ECUs with same input CAN-bus message. To this end, a dataset for four identical ECUs are collected. For each data collection setting, 144000 (3600*40) samples are collected. For performance evaluation, random partitioning is performed to divide the dataset into the training and test set (Training set: 65%, Test set: 35%). The dataset used here is collected in the same environment i.e. under the same temperature and using an identical message to observe the minute and unique variation of the digital signals.

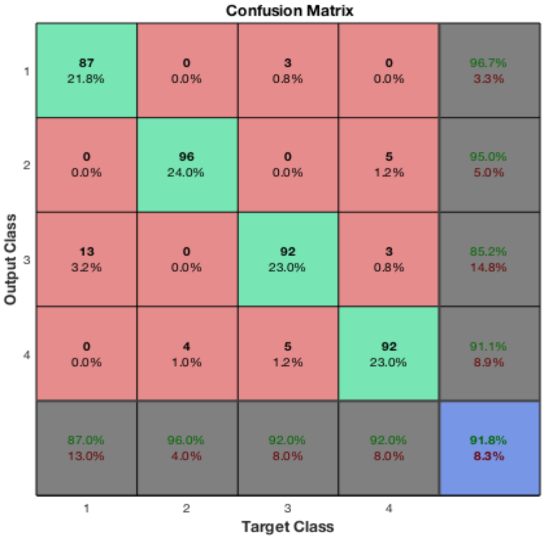
EXPERIMENTAL RESULTS AND ANALYSIS

A multilayer neural network is trained with “scaled conjugate gradient back propagation” training algorithm, 8 inputs variables (time domain), 4 outputs which corresponds to different ECUs, stopping criteria of Epochs = 2000, gradient = 1e-7, and three hidden layers with 40 and 40 hidden nodes respectively. Shown in figure 10 is the architecture of the multilayer neural network trained for channel classification.

Figure 10: Neural Network architecture of ECU classifier

We tested the proposed method on a CAN network with 4 nodes. Each node comprises of the Arduino mounted CAN Shield. ECU’s are programmed to transmit 01010101010 patterns and then using Agilent 100 oscilloscope the physical signal was acquired in MATLAB. For each ECU 25*12= 300 cycles were recorded. On the recorded data, the feature set as explained in Table-1 is computed.

TABLE II. CONFUSION MATRIX OF NEURAL NETWORK



Confusion Matrix

Output Class \ Target Class	1	2	3	4	
1	87 21.8%	0 0.0%	3 0.8%	0 0.0%	96.7% 3.3%
2	0 0.0%	96 24.0%	0 0.0%	5 1.2%	95.0% 5.0%
3	13 3.2%	0 0.0%	92 23.0%	3 0.8%	85.2% 14.8%
4	0 0.0%	4 1.0%	5 1.2%	92 23.0%	91.1% 8.9%
	87.0% 13.0%	96.0% 4.0%	92.0% 8.0%	92.0% 8.0%	91.8% 8.3%

Experimental results show that ECU's are recognized with 91.8% of accuracy. To improve the efficiency of the system, voting majority logic is implemented. Data from the same ECU is acquired 5 times and at each time ECU is classified as ECU 'N' if the ECU is recognized correctly in 4 out of 5 data acquisitions, the ECU is marked as ECU 'N'. In this way efficiency of the ECE identification is improved to 99 % accuracy.

This proposed method relies on the physical signals and independent of the timing information. It is a challenge for attacker to replicate the electrical properties of both node and the channel, which gives this proposed idea an edge.

CONCLUSION

In this paper, we have analyzed the Clock-Based Intrusion Detection System in detail and have found the vulnerabilities. These weaknesses allow the reconstruction of the assumed unique behavior of a clock crystal and hence, the security notion is broken. We have also proposed a proof of concept attack, clock-spoofing, which is an advanced variant of Masquerades Attack and can bypass the CIDS successfully.

Because the weaknesses are present in the core architecture and working principle of the CIDS, after the in-depth analysis and contemplation we have concluded that in current forms of realization of CIDS, these are in-efficient and can be bypassed.

Instead, as a remedy, we have discussed a brief solution, which is based on the physical fingerprinting of the nodes. The received signal also incorporates the channel properties, which introduces another layer of security. Now the attacker has to replicate the physical characteristics of the channel as well as transmitter. Hence, efficient and higher degree of results are achieved in proof-of-concept test beds.

REFERENCES

[1]. ["CAN-Bus Specifications," Rep. Robert Bosch GmbH. Postfach 50, D-7000. Stuttgart 1Print.

[2]. S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, "Comprehensive experimental analyses of automotive attack surfaces," in Proc. of the 20th USENIX Conference on Security, ser. SEC'11. Berkeley, CA, USA: USENIX Association, pp. 6–6, 2011.

[3]. J.M. Flores-Arias, M. Ortiz-Lopez, F.J. Quiles-Latorre, V. Pallares and A. Chen, "Complete hardware and software bench for the CAN bus," in Proc. of IEEE International Conference on Consumer Electronics (ICCE'2016), Las Vegas, NV,

pp. 211-212, 2016. doi: 10.1109/ICCE.2016.7430584.

[4]. R. Belloq, "The Golden Idol of the Hovitos", International journal of Rare Artifacts, September, 1936.

[5]. I. Studnia, V. Nicomette, E. Alata, Y. Deswarte, M. Kaaniche, and Y. Laarouchi, "Survey on security threats and protection mechanisms in embedded automotive networks," in Proc. of 2nd Workshop on Open Resilient Human-aware Cyber-Physical Systems (WORCS-2013), 2013.

[6]. A. Greenberg, "Hackers Remotely Kill a Jeep on the Highway {with me in it}" 2015. Available: http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

[7]. K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage, "Experimental security analysis of a modern automobile", In D. Evans and G. Vigna, editors, IEEE Symposium on Security and Privacy. IEEE Computer Society, May 2010.

[8]. Q. Wang and S. Sawhney, "VeCure: A practical security framework to protect the CAN bus of vehicles," in Proc. of International Conference on the Internet of Things (IOT), Cambridge, MA, pp. 13-18, 2014.

[9]. P. Kleberger, T. Olovsson, and E. Jonsson, "Security Aspects of the In-Vehicle Network in the Connected Car," in Proc. of IEEE Intelligent Vehicles Symposium (2011). Web. 5 June 2011.

[10]. B.V. Kumar and J. Ramesh, J. (2014) "Automotive in Vehicle Network Protocols," in Proc. International Conference on Computer Communication and Informatics, Coimbatore, 3-5, pp. 1-5, 2014.

[11]. T. Hoppe, S. Kiltz, and J. Dittmann, "Security Threats to Automotive CAN Networks—Practical Examples and Selected Short-term Countermeasures," in Proc. of Reliability Engineering & System Safety, Vol. 96(1), pp. 11-25. "Web", 2011.

[12]. A. Hafeez, H. Malik, O. Avatefipour, P.R. Rongali, and S. Zehra, "Comparative Study of CAN-Bus and FlexRay Protocols for In-Vehicle Communication" (No. 2017-01-0017). SAE Technical Paper, 2017.

[13]. S. Corrigan, "Introduction to the Controller Area Network (CAN)," Rep. Texas Instrument. Application Report SLOA101B-August2002-Revised May 2016.

[14]. K. Zdeněk and S. Jiří, "Simulation of CAN bus physical layer using SPICE," in Proc. of International Conference on Applied Electronics, Pilsen, pp. 1-4, 2013.

[15]. K.-T. Cho, and K.G. Shin, "Fingerprinting electronic control units for vehicle intrusion detection," In Proc. of 25th USENIX Security Symposium (USENIX Security'2016), USENIX Association, 2016.

[16]. U. Hiroshi, R. Kurachi, H. Takada, T. Mizutani, M. Inoue, and S. Horihata, "Security Authentication System for In-Vehicle Network," SEI Technical Review 81, 2015.

[17]. A. Hazem, and H.A. Fahmy, "LCAP – A Lightweight CAN Authentication Protocol for securing in-vehicle networks," in Proc. of 10th Int. Conf. on Embedded Security in Cars (ESCAR'2012), Berlin, Germany. Vol. 6, 2012.

[18]. S. Dey, N. Roy, W. Xu, P.R. Choudhury, and S. Nelakuditi, "Accelprint: Imperfections of accelerometers make

smartphones trackable," in Proc. of 21st Annual Network and Distributed System Security Symposium (NDSS'2014), San Diego, California, USA, February 23-26, 2014. [Online]. Available: http://www.internetsociety.org/doc/accelprint-imperfections-accelerometers-make-smartphones-trackable

[19]. "LibXtract: Feature Extraction Library Documentation,"

[20]. Available: http://jamiebullock.github.io/LibXtract/documentation/

[21]. G. Brown, A. Pocock, M.J. Zhao, and M. Luján, "Conditional likelihood maximisation: a unifying framework for information theoretic feature selection." Journal of Machine Learning Research, V. 13(Jan), pp. 27-66, 2013.

[22]. "SAE J1128Standard - Low Voltage Primary Cable," Sep. 2013, SAE International: http://standards.sae.org/j1128_201310/

[23]. A. Perrig, R. Canetti, J. D. Tygar, and D.X. Song, "Efficient Authentication and Signing of Multicast Streams over Lossy Channels," in Proc. of IEEE Symposium on Security and Privacy, pp. 56-73, 2000.

[24]. S. Nürnberger S., C. Rossow C. " vatiCAN –Vetted, Authenticated CAN Bus," In B. Gierlichs, A. Poschmann (eds) Cryptographic Hardware and Embedded Systems (CHES'2016), Lecture Notes in Computer Science, Vol. 9813. Springer, Berlin, Heidelberg, 2016.

[25]. P.-S. Murvay, and B. Groza, "Source identification using signal characteristics in controller area networks," IEEE Signal Processing Letters, Vol. 21(4) pp. 395-399, 2014.

[26]. O. Avatefipour, A. Hafeez, M. Tayyab, H. Malik, "Linking received packet to the transmitter through physical-fingerprinting of controller area network," in Proc. of IEEE Workshop on Information Forensics and Security (WIFS'2017), 2017. Electronic ISSN: 2157-4774.