2021 NDIA GROUND VEHICLE SYSTEMS ENGINEERING and TECHNOLOGY SYMPOSIUM
Cyber Technical Session
August 10-12, 2021 - Novi, Michigan

Cyberattack Defense Through Digital Fingerprinting, Detection Algorithms, and Bus Segmentation in Ground Vehicles

Jonathan Wolford[1], Courtney Westrick[2], Peter Moldenhauer[1]

[1]Southwest Research Institute, San Antonio, TX
[2]Ground Vehicle Systems Center, Warren, MI

ABSTRACT

This paper describes strategies to secure military ground vehicles by using digital fingerprinting, detection algorithms, and bus segmentation to identify and remove anomalous messages from the Controller Area Network (CAN) bus. Modern automotive buses were designed for reliability rather than security. This lack of security means that any node on the bus can transmit a message to any other node, and the receiver cannot verify the sender or that the message is unaltered. The intrusion defense system (IDS) protects the bus by actively monitoring traffic on all connected busses and removing messages identified as anomalies. Digital fingerprinting combined with various detection algorithms identifies these anomalies while bus segmentation simultaneously defends the CAN bus by removing anomalous messages.

Citation: J. Wolford, C. Westrick, P. Moldenhauer, "Cyberattack Defense Through Digital Fingerprinting, Detection Algorithms, and Bus Segmentation in Ground Vehicles", In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 10-12, 2021.

1. INTRODUCTION

Controller Area Network (CAN) is the reliable automotive bus protocol that enables communication among the various nodes or electronic control units (ECUs) in a vehicle. CAN was designed for reliability and flexibility to perform in a wide range of applications rather than for security. Security from cyberattacks was of little concern during the development of CAN in 1986. Other than a checksum, CAN buses do not provide integrity or authentication verification for transmitted data, meaning that any node on the bus can transmit a message to any other node and the receiver cannot verify the sender or if a given CAN packet is unaltered. Over time, as vehicle connectivity increases, the lack of security in CAN bus networks presents an increasingly serious problem that needs to be addressed. Commercial and military ground vehicles alike are vulnerable to cyber threats.

GVSC and SwRI created a solution to this problem by using digital fingerprinting to monitor physical layer characteristics, combined with application layer monitoring to identify anomalous messages on the bus. Once anomalies are identified, these messages are removed through bus segmentation. The physical and application layer detection algorithms, along with bus segmentation, make up the automotive intrusion defense system (IDS).

2. BACKGROUND

Throughout history, ground vehicle technology has focused on vehicle safety, durability, reliability, and performance over security. CAN was developed with the main purpose of allowing vehicle ECUs to reliably communicate [1]. Security features and the ability to defend against cyberattacks were not priorities when the CAN bus protocol was officially released in 1986.

One of the key elements that CAN was designed to handle was to ensure the real-time nature of the vehicle bus. However, this action is done without any verification that the message is actually originating from the specified node, meaning that a single compromised node on a vehicle can then freely impersonate any other node [2]. Another important feature of the CAN protocol is the arbitration ID (ArbID), which not only serves to identify messages but also indicates priority, allowing for arbitration between messages simultaneously coming from the different nodes on the vehicle.

Attackers can now manipulate and/or disable a vehicle network with not only physical access but through remote access as well [3]. Remote access greatly increases the risk to automotive manufactures as attacks could be carried out against multiple vehicles at once without physical contact.

2.1. IDS Basics

IDS are designed to identify anomalous traffic in networks and alert users or take other actions to protect the affected systems. Anomalies can come from compromised systems which have privileged access to the networks. They can also come from external sources which are outside of the targeted network and attempt to gain access [4]. There are various strategies an IDS can employ to identify these intrusions. Two of these techniques are signature-based and anomaly-based detections. Signature-based detection uses the characteristics of previously identified attacks to flag anomalies, and packets that do not match the recorded signatures are not flagged. Anomaly-based detection examines the behavioral characteristics of the traffic rather than the contents. Both of these approaches are used in the demonstrated solution.

2.2. Digital Fingerprinting

One method to monitor traffic is to track the physical layer characteristics and create “digital fingerprints” for CAN messages and corresponding nodes. Fingerprinting can also be used as a method of user authentication because each broadcasting ECU has identifiable characteristics. The IDS can then detect when a node is sending a message it should not, such as a masquerade attack. It focuses on a CAN transceiver’s message transmission by analyzing the low-level voltage characteristics of the transmitted CAN frame for each arbitration ID. The IDS uses digital fingerprinting to measure the voltage transition rates and minimum/maximum voltages for each CAN frame. By developing statistics based on measured, low-level voltage characteristics for each arbitration ID, the IDS can characterize the device transmitting each message. The IDS is then able to accurately identify messages sent from unauthorized nodes by detecting statistical anomalies in the physical layer measurements.

2.3. Bus Segmentation Basics

To be effective, an IDS should be armed with mitigation removal techniques as well. This way, the IDS is not limited to just detection but also has the capability to remove anomalous messages from the network. The IDS implements bus segmentation which causes a node that is broadcasting anomalous messages in the network to be segmented or quarantined from the rest of the bus. This method allows messages to be filtered before they are sent to other nodes in the network. Valid messages are retransmitted on the primary and quarantine bus, depending on which bus the message was received from, so that valid CAN traffic remains largely unaffected when the bus is segmented. Bus segmentation is further detailed in

the previous paper discussing the IDS detection methods [2].

3. SYSTEM ARCHITECTURE

SwRI and GVSC have developed an IDS that uses the combined strategies of application layer detection algorithms, digital fingerprinting, and bus segmentation to solve the security issues present on the CAN bus.

When installed on a vehicle, the IDS monitors CAN traffic among all connected nodes. When monitoring, all CAN messages are processed by the core detection and fingerprinting algorithms to identify application and physical layer characteristics. These characteristics are analyzed to identify anomalies (see Figure 1).

The application layer characteristics include signature-based (e.g., whitelisted arbitration identifiers) and anomaly-based (e.g., timing for purely periodic or event driven messages). The physical layer characteristics include voltage levels and transition rates. The IDS uses these physical layer characteristics to then identify when a message is transmitted from the wrong node.

Once an anomaly is detected by either method, the IDS identifies the responsible node and then segments the bus to move it to an isolated, secondary bus. A central gateway is inserted between the primary and secondary (isolated) buses, and the malicious messages are removed from the primary bus. Any single node can be switched from the primary onto the secondary bus.

The IDS is also transparent to the vehicle during normal operation and is designed to allow all traffic through in the event of an IDS issue.

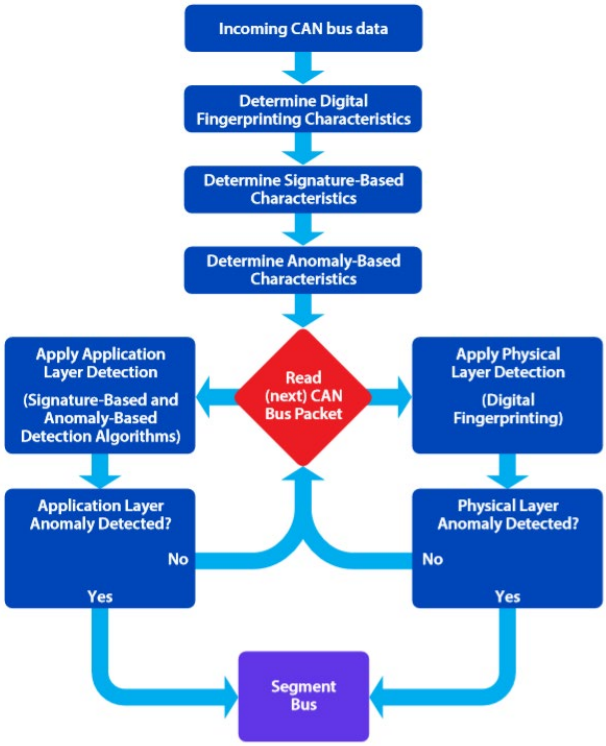

Figure 1. IDS Flow Diagram

3.1. Digital Fingerprinting (Physical Layer Detection)

To support digital fingerprinting within the IDS, the system incorporates a Field-Programmable Gate Array (FPGA) and an Analog-Digital Converter (ADC) that allow high-speed, low-level analysis of each CAN frame on the bus. The FPGA is primarily used to (1) interface to the ADC to receive samples of the signals seen in the CAN bus, (2) implement logic to detect the start and end of a CAN frame, (3) decode the arbitration ID for each CAN frame, and (4) take physical layer measurements. The FPGA calculates the positive and negative voltage transition times using the digital readings and sample rate from the ADC. The maximum and minimum voltage are also recorded for the CAN frame. The arbitration ID and recorded measurements are compiled for transmission from the FPGA to the rest of the IDS.

An amplifier and clock source are also connected to the ADC, and their primary functions are to

ensure the system can read CAN bus signals while being compatible with the electrical requirements of the CAN bus.

In parallel to the core detection algorithms, the IDS monitors data incoming from the FPGA and then analyzes the data to develop statistics for each arbitration ID in the network based on the physical layer measurements. The resulting model built from the measurements taken by the FPGA and ADC can be used to characterize the hardware that each arbitration ID originates from. The IDS then uses anomaly-based detection algorithms to flag anomalies. These algorithms look for statistical outliers for each message compared to the mean values of the positive and negative voltage transition rates, maximum voltage level, and minimum voltage level for that message's arbitration ID. These anomalies are used to identify when a message is transmitted from the wrong node.

3.2. Detection Algorithms (Application Layer Detection)

The IDS core detection algorithms utilize signature and anomaly-based detection techniques to identify malicious messages. Each individual CAN message is passed through these algorithms.

The signature-based detection algorithms only allow specified messages to be present in the vehicle network while flagging and removing unknown ones. One of the signature-based detection algorithms compares the arbitration identifiers to a whitelist file during detection. This file is based on the Database Container File (DBC) that is unique to each vehicle type. An anomaly is flagged when an unknown or invalid message is encountered. The signature-based detection algorithms also ensure that bus errors are flagged as anomalies while valid diagnostic messages are not flagged.

Another anomaly-based detection algorithm monitors timing for purely periodic and event driven messages. If the timestamp of a given packet was not consistent with the typical rate at which packets with the same arbitration identifier are transmitted, then they are flagged as anomalous. An anomaly is flagged if a given message is a statistical outlier when it is compared to the mean inter-message interval. These algorithms identify timing-based anomalies caused from messages injected into the vehicle bus in various attack scenarios such as message throttling, bus flooding, and arbitrary insertion.

3.3. Bus Segmentation

When anomalies are detected, the IDS can segment the bus by placing the anomalous node on a separate secondary bus. The IDS then becomes a gateway for the quarantined node. Anomalous messages are filtered out of the traffic on the secondary bus, and the remaining messages are retransmitted on the primary bus.

4. PERFORMANCE

This section outlines the performance of the IDS. Tests were conducted on military and commercial vehicles with the IDS running on an SELinux kernel.

CAN traffic was read from the vehicle bus and two additional nodes that were simulated using independent CAN transceivers. Each transceiver was a different model from different manufacturers. One transceiver was used for transmitting simulated anomalous CAN data. Different attacking scripts were executed, and the generated attacks were then compared with the log files from the IDS to determine the system accuracy. Figure 2 shows the hardware setup used during testing.

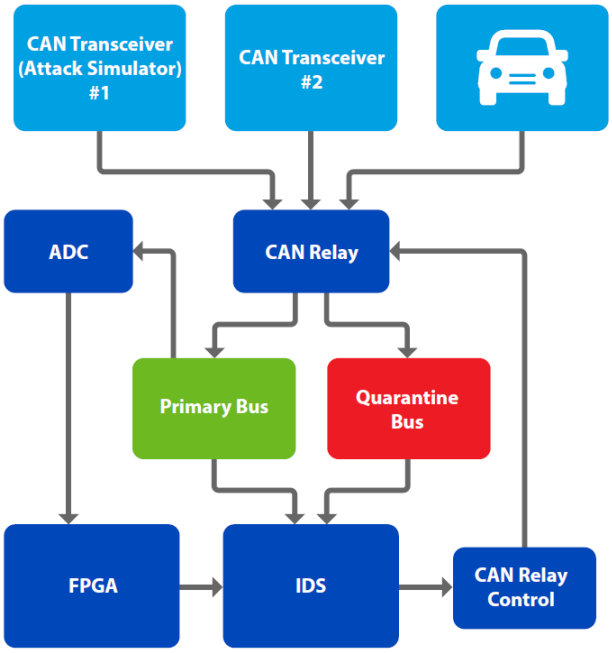

Figure 2. IDS Hardware Diagram

Upon detection, the attacking node is switched off the main bus to the secondary bus.

4.1. Digital Fingerprinting (Physical Layer Detection)

To test the digital fingerprinting methods of the IDS, two test cases were generated to ensure the system correctly identified when messages were sent from the wrong device.

1. Normal – This test case ensures the system does not flag anomalies when all nodes are sending the correct messages.
2. Masquerade Attack – One node sends messages that are normally sent by another node.

4.2. Application Layer Detection

To test the application layer detection algorithms of the IDS, the following test cases were used:

1. Arbitrary Injection – Arbitrary packets are injected into the CAN bus.
2. Bus Flood –This test injects packets at a high speed, both replayed and fuzzed, to overwrite legitimate traffic.
3. Throttling – Manipulates the speed at which packets are broadcast.
4. Whitelist – Injects packets that should not be present.
5. Normal – While not an attack, this test case is necessary to ensure that the system does not detect attacks when normal traffic is present on the bus. This test case uses real vehicle data.
6. Diagnostic –Diagnostic messages are inserted into the network traffic.

4.3. Results

Target thresholds were set for detection rates and to determine the success of each detection method. The target thresholds for detection were as follows:

- True Positive (TP) Rate: >95%
- True Negative (TN) Rate: >95%
- False Positive (FP) Rate: <5%
- False Negative (FN) Rate: <5%

True Positive (TP) rate is the percentage of detected anomalies that were generated attacks.

False Positive (FP) rate is the percentage of detected anomalies that were valid messages.

True Negative (TN) rate is the percentage of total messages that were correctly not identified as anomalies.

False Negative (FN) rate is the percentage of total messages that were incorrectly identified as valid messages.

Table 1 shows the fingerprinting detection rates and Table 2 shows the application layer detection rates.

Table 1: Fingerprinting Detection Rates

Test	TP Rate	TN Rate	FP Rate	FN Rate
Normal	Not Applicable	99.76%	0.24%	0.00%
Masquerade Attack	95.70%	99.77%	0.23%	4.30%

The fingerprinting results were excellent and was able to successfully identify when the CAN transceiver responsible for sending attacks.

Table 2: Application Layer Detection Rates

Test	TP Rate	TN Rate	FP Rate	FN Rate
Normal	Not Applicable	99.67%	0.33%	NA
Arbitrary Injection	91.80%	99.86%	0.14%	8.20%
Bus Flood	98.31%	99.46%	0.64%	1.69%
Throttling	96.80%	99.90%	0.10%	3.20%
Whitelist	100%	99.86%	0.14%	0.00%
Diagnostic	100%	99.90%	0.10%	0.00%

Target thresholds were met for each defined attack vector except for Arbitrary Injection. The timing measurements for each arbitration ID provided enough consistency for the detection algorithms to obtain great results.

Additionally, bus segmentation was also successful. When anomalies were detected, attacks were removed from the CAN bus.

For digital fingerprinting, each of the analyzed CAN transceivers had distinct voltage transition rates. These measurements were also consistent among each arbitration ID on a particular node. This demonstrates that the message contents had no significant effect on the node's transition rates. This can be seen in Table 3, which shows measurements taken from four different CAN transceivers. Nodes 1A and 1B are similar hardware from the same manufacturer. Nodes 2A and 2B are similar hardware from another manufacturer.

Table 3: CAN Transceiver Statistical Model Sample

Node	Arb ID	Rise Time (Clock Cycles)		Fall Time (Clock Cycles)	
		Mean	Std Dev	Mean	Std Dev
1A	700	10.00	0.08	11.13	0.34
	7EF	10.01	0.13	11.12	0.32
	7F0	10.01	0.08	11.12	0.32
	7FE	10.01	0.09	11.12	0.32
	7FF	10.00	0.05	11.13	0.33
1B	12A	10.31	0.46	11.04	0.20
	135	10.19	0.40	11.00	0.04
	137	10.18	0.38	11.00	0.05
	139	10.19	0.39	11.00	0.06
	160	10.18	0.39	11.00	0.00
2A	410	5.15	0.35	9.41	0.49
	415	5.19	0.39	9.46	0.50
	420	5.15	0.36	9.42	0.50
	425	5.14	0.34	9.40	0.49
	433	5.13	0.33	9.40	0.49
2B	440	5.11	0.55	10.34	0.89
	443	5.07	0.25	10.30	0.87
	444	5.10	0.30	10.27	0.88
	450	5.10	0.30	10.25	0.90
	460	5.09	0.29	10.28	0.90

Each device shows very similar means for each arbitration ID with low standard deviations. **This confirms the consistency of each device and discernability between different hardware. With this knowledge, the system can identify which hardware transmitted a packet.** However, hardware from the same manufacturer produced similar measurements making it difficult, but not impossible, to discern between two of the same devices. The team has hypothesized that a higher measurement resolution would better identify the differences between similar hardware.

As expected, measurements taken on-vehicle were different for some arbitration IDs and similar for others. In Table 4, the arbitration IDs are grouped by similar measurements for mean rise and fall times (e.g. Arb ID 172,174,176). **This infers that grouped messages are sent from the same transceiver within the vehicle.**

Table 4: Vehicle Physical Measurement Sample

Arb ID	Rise Time (Clock Cycles)		Fall Time (Clock Cycles)	
	Mean	Std Dev	Mean	Std Dev
172	22.61	0.71	19.26	0.58
174	22.84	0.54	19.19	0.50
176	22.89	0.48	19.23	0.54
1A1	19.72	0.46	19.24	0.43
1A2	19.20	0.40	20.20	0.40
1B0	18.58	0.61	19.62	0.48
224	16.57	0.52	19.13	0.34
226	16.50	0.53	19.18	0.38
228	16.48	0.57	19.16	0.37
514	12.00	0.03	22.01	0.14
52A	12.00	0.03	22.00	0.11
530	12.00	0.03	22.01	0.12

Notably, the measurements were discernable from the simulated nodes, and the fingerprinting algorithms successfully determined when arbitration IDs from the vehicle were transmitted from an incorrect node.

From the results shown, fingerprinting and application layer detection works well in identifying attacks. Furthermore, the fingerprinting solution could be used to map out a CAN bus within a vehicle through the way rise and fall time measurements fall into groups. The detection rates can be further improved, and false positive rates decreased through further development.

5. FUTURE IDS DEVELOPMENT

The results of the project have been largely positive, yet there is still room for improvement. Additional work can be conducted to prepare the IDS for fleet deployment. This includes reducing the number of false positives along with improving reliability through extensive testing. Ideally, the IDS should never generate false positives. The IDS should also be thoroughly tested on multiple vehicle types under various environment, temperature, and duration conditions to fully understand the IDS limitations and identify additional areas for improvement.

6. CONCLUSION

The IDS security solution that GVSC and SwRI are developing is unique as it leverages both physical layer and application layer detection. In both layers, the IDS detection rates are fantastic. The IDS consistently averages a true positive rate of 91-99% while also limiting the false positive rate to below 1%. The IDS, armed with these powerful detection capabilities, can be integrated onto previously deployed vehicles without redesigning the CAN bus.

7. REFERENCES

[1] W. Lawrenz, *CAN System Engineering*, 2nd ed.London: Springer, 2013, p. 3.

[2] R. Elder, C. Westrick, P. Moldenhauer, “Cyberattack Detection and Bus Segmentation in Ground Vehicles”, in *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 11-13, 2020

[3] A. Greenberg, ‘Hackers Remotely Kill a Jeep on the Highway-With Me in It’, 2015. [Online]. Available: https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/ [Accessed: 7- May- 2020].

[4] J.R. Vacca, Computer and information security handbook. Newnes (Amsterdam, 2012), pp. 47–60