Annotated Bibliography

Phineas Giegengack

F. Amato, L. Coppolino, F. Mercaldo, F. Moscato, R. Nardone and A. Santone, "CAN-Bus Attack Detection With Deep Learning," in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 8, pp. 5081-5090, Aug. 2021, doi: 10.1109/TITS.2020.3046974.

- High-level description of CAN protocol, CAN packet content, deep learning and statistical methods (Neural Network, Multi-layer perceptron).
- Considers 4 types of CAN bus attacks: Denial of service (DOS), fuzzing, drive gear spoofing, and RPM gauge spoofing.
- Machine learning used to classify CAN messages which are labeled as either 'normal' or one of the four attack types
    - Neural net achieved high precision and recall for all types of attacks
    - MLP was challenged by DOS and fuzzing attacks but adding additional layers allowed it to outperform neural net
- Shows promise for using machine learning to detect CAN bus intrusion, but I suspect their results would not apply to a real-time/real world application.
    - Model does not consider context of each CAN bus message (what message precedes a given message could reveal an attack)
    - Overfitting for spoofing attacks in this dataset (achieved precision and recall of 1.0 for both spoofing attacks which doesn't make sense)

Ma, Haoyu, Cao, Jianqiu, Mi, Bo, Huang, Darong, Liu, Yang, Li, Shaoqian, A GRU-Based Lightweight System for CAN Intrusion Detection in Real Time, Security and Communication Networks, 2022, 5827056, 11 pages, 2022. https://doi.org/10.1155/2022/5827056

- Proposes a system architecture for deploying, training, and updating CAN intrusion detection models over-the-air
- Develops a feature-extraction algorithm for CAN bus messages
- Develops a GRU-based neural network that is lightweight enough to be deployed on embedded hardware
- Feature extraction: uses CAN ID, Data length code, data payload and timestamp
- Trained on a server, ran on Nvidia Jetson
- Hyperparameters are given for each model they tried (could be handy)

- Nvidia Jetson is a fairly low-cost hardware platform (~150$) but I may seek something cheaper/less computational power which will likely yield worse model performance.

J. Guidry, F. Sohrab, R. Gottumukkala, S. Katragadda and M. Gabbouj, "One-Class Classification for Intrusion Detection on Vehicular Networks," 2023 IEEE Symposium Series on Computational Intelligence (SSCI), Mexico City, Mexico, 2023, pp. 1176-1182, doi: 10.1109/SSCI52147.2023.10371899. keywords: {Support vector machine classification;Intrusion detection;Machine learning;Data models;Computer crime;Computational intelligence;Cyber Security;Vehicular Security;One-Class Classification},

- Support Vector Machine (SVM) used to be able to train only on nominal CAN bus data and detect scenarios that fall outside of expected behavior
- Researchers tried numerous state-of-the-art SVM methods to detect CAN bus intrusions
- Collected real CAN bus data from Nissan Leaf and Chevy Volt
- Did not run on embedded hardware
- Researchers acknowledged that the features used were trivial, and better performance may have been achieved with more consideration as to which features of the messages were most descriptive

C. Chupong, N. Junhuathon, K. Kitwattana, T. Muankhaw, N. Ha-Upala and M. Nawong, "Intrusion Detection in CAN Bus using the Entropy of Data and One-class Classification," 2024 International Conference on Power, Energy and Innovations (ICPEI), Nakhon Ratchasima, Thailand, 2024, pp. 157-160, doi: 10.1109/ICPEI61831.2024.10748816. keywords: {Support vector machines;Technological innovation;Accuracy;Intrusion detection;Entropy;Anomaly detection;Testing;CAN bus;intrusion detection;anomaly detection;entropy of data;One-Class SVM},

- Uses a one-class SVM to detect anomalies in CAN bus behavior
- Features include timestamp and entropy, a measure of the randomness of each message
- Feature vector size is reduced by this method, and performance is still generally good, although on a limited data set and not run on real embedded hardware/real time

L. Liang *et al.*, "Intrusion Detection Model for In-vehicle CAN Bus Based on TPE-LightGBM Algorithm," *2025 IEEE 34th Wireless and Optical Communications Conference (WOCC)*, Taipa, Macao, 2025, pp. 419-423, doi: 10.1109/WOCC63563.2025.11082193.

- Most recent (2025) paper on this subject
- Uses Gradient boost model to develop a lightweight detection of CAN bus intrusion
- Includes a nice mathematical description of LightGBM
- Feature extraction focuses on CAN IDs, IDs of previous CAN messages, and time intervals between previous message, and time interval between previous message with same ID

J. N. Brewer and G. Dimitoglou, "Evaluation of Attack Vectors and Risks in Automobiles and Road Infrastructure," *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, 2019, pp. 84-89, doi: 10.1109/CSCI49370.2019.00021.
keywords: {5G mobile communication;Engines;Scientific computing;Computational intelligence;Telematics;Fuzzing;Relays;Automotive cybersecurity, IoT, Autonomous cars, attack vectors, vulnerabilities},

- Seems to be a much-referenced paper that outlines the attack vector on the CAN bus, that is, a number of ways intruders could affect the behavior of the vehicle if they were able to sniff and insert packets onto the CAN bus
- Proves capability of someone with CAN bus access to perform packet injection, fuzzing, denial of service, replay attacks, which have a variety of alarming effects on the vehicle.

Y. Liao and B. Yang, "To Generalize or Not to Generalize: Towards Autoencoders in One-Class Classification," *2022 International Joint Conference on Neural Networks (IJCNN)*, Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892812.
keywords: {Degradation;Training;Neural networks;Training data;Benchmark testing;Feature extraction;Behavioral sciences;autoencoder;one-class classification;batch-wise feature weighting;neural network;anomaly detection},

- Math behind Autoencoders for 1-class classification
- Must cite this in my explanation of Autoencoders since I'm using their math

J. Lee, S. Park, S. Shin, H. Im, J. Lee and S. Lee, "ASIC Design for Real-Time CAN-Bus Intrusion Detection and Prevention System Using Random Forest," in *IEEE Access*, vol. 13,

pp. 129856-129869, 2025, doi: 10.1109/ACCESS.2025.3585956. keywords: {Controller area networks;Real-time systems;Security;Protocols;Prevention and mitigation;Intrusion detection;Random forests;Standards;Computational modeling;Image edge detection;Application-specific integrated circuit;controller area network;intrusion detection system;intrusion prevention system;machine learning;random forest},

- Design and tape-out of an ASIC to perform real-time execution of isolation forest ML model for detecting CAN bus intrusions.
- Really interesting and unique assessment of real-time potential for ML detection of CAN bus intrusions that I haven't seen in any other studies

Bi, Zixiang, Xu, Guoai, Xu, Guosheng, Tian, Miaoqing, Jiang, Ruobing, Zhang, Sutao, Intrusion Detection Method for In-Vehicle CAN Bus Based on Message and Time Transfer Matrix, *Security and Communication Networks*, 2022, 2554280, 19 pages, 2022. https://doi.org/10.1155/2022/2554280

- still need to read this one but could be relevant

B. Lampe and W. Meng, "can-train-and-test: A New CAN Intrusion Detection Dataset," *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*, Hong Kong, Hong Kong, 2023, pp. 1-7, doi: 10.1109/VTC2023-Fall60731.2023.10333756. keywords: {Authorization;Training;Intrusion detection;Authentication;Machine learning;Traction motors;Data models;Controller area network (CAN);in-vehicle network (IVN);automotive network;automotive security;intrusion detection system (IDS);logging;dataset},

- dataset I have chosen to use
- can-train-and-test
- includes a few datasets of CAN bus activity for different vehicles and attack-free data and examples of a few attacks

B. M. Tóth and A. Bánáti, "Autoencoder Based CAN BUS IDS System Architecture and Performance Evaluation," *2025 IEEE 19th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, Timisoara, Romania, 2025, pp. 000099-

000104, doi: 10.1109/SACI66288.2025.11030168.
keywords: {Systematics;Microcontrollers;Tiny machine learning;Computational modeling;Autoencoders;Intrusion detection;Real-time systems;Computational efficiency;Computer security;Tuning;autoencoder;IDS;embedded;TinyML;CAN BUS;automotive cybersecurity},

- most similar existing work to what I am trying to do, although their results were poor (F1 score of ~0.3)
- quantized their model and ran it on a tiny microcontroller
- obviously did not do real-time training

Michael Kemmler, Erik Rodner, Esther-Sabrina Wacker, Joachim Denzler, One-class classification with Gaussian processes, Pattern Recognition, Volume 46, Issue 12, 2013, Pages 3507-3518, ISSN 0031-3203, https://doi.org/10.1016/j.patcog.2013.06.005. (https://www.sciencedirect.com/science/article/pii/S0031320313002574)

- Gaussian processes for 1-class classification
- Ultimately infeasible for my use case
- Shows that I considered other options

F. Sohrab, J. Raitoharju, M. Gabbouj and A. Iosifidis, "Subspace Support Vector Data Description," *2018 24th International Conference on Pattern Recognition (ICPR)*, Beijing, China, 2018, pp. 722-727, doi: 10.1109/ICPR.2018.8545819.
keywords: {Optimization;Training;Kernel;Data models;Training data;Support vector machine classification;One-class Classification;Support Vector Data Description;Subspace Learning},

- S-SVDD for OCC (one-class classification)
- Can’t do retraining easily with this architecture, but may return to it if I run into trouble with autoencoder