vehicle-agnostic IDS

Haneen Almassri 2185044
Master's in International Cybersecurity and Cyber Intelligence

Abstract :

As the **Controller Area Network (CAN Bus)** remains the industry standard for in-vehicle communication, its inherent lack of encryption and authentication continues to pose severe cybersecurity risks. While **Intrusion Detection Systems (IDSs)** are the primary defense mechanism against these vulnerabilities, their evolution is currently stifled by a critical data gap. Most existing datasets are either proprietary, lack realistic attack verification, or—most significantly—are **vehicle-dependent**, limiting the development of robust, vehicle-agnostic IDS solutions.

To bridge this gap, this work introduces a comprehensive, multi-source CAN bus dataset designed to facilitate the training of generalized detection models. By aggregating and normalizing benign data from four major benchmark projects (**CAN-MIRGU, Car_Hacking_Challenge_Dataset, Car-Hacking Dataset, and OTIDS**), we provide a diverse baseline that captures a wide range of operational scenarios across different platforms. This is complemented by a suite of **physically verified injection attacks**—including **DoS, Fuzzing, Spoofing, and Gear Injection**—recorded from a modern autonomous-capable vehicle under real-world driving conditions.

By offering a balanced mix of cross-platform benign behavior and sophisticated, verified attack vectors, this dataset provides a rigorous foundation for evaluating **vehicle-agnostic IDSs**, ultimately advancing the security of next-generation connected and autonomous vehicles.

I. INTRODUCTION

Modern vehicles are equipped with increasingly sophisticated electronic control units (ECUs) that enable advanced features such as automated parking assistance, lane departure warning, adaptive cruise control, and comprehensive infotainment systems. These ECUs rely on the Controller Area Network (CAN) bus—a lightweight, robust, and cost-effective communication protocol that has become the de facto standard for in-vehicle networking. However, the CAN bus was designed for reliability in closed environments and inherently lacks fundamental security mechanisms such as encryption, authentication, and message integrity verification. This architectural deficiency, combined with the protocol's broadcast nature and ID-based priority scheme, renders modern vehicles vulnerable to a wide spectrum of cyberattacks. Researchers have demonstrated successful exploitation of these vulnerabilities across multiple

vehicle brands, enabling attackers to gain unauthorized control over critical vehicle functions and jeopardizing passenger safety.

In response to these threats, significant research efforts have focused on developing Intrusion Detection Systems (IDS) for in-vehicle networks. Among detection strategies, anomaly-based approaches have gained prominence due to their ability to detect novel attacks without relying on predefined signatures. These systems model the normal behavior of CAN bus traffic and flag deviations as potential intrusions. However, the advancement of effective IDS research faces two major obstacles: (1) the scarcity of high-quality, publicly available datasets that include physically verified attacks under real-world driving conditions, and (2) the inherent vehicle-specific nature of most proposed solutions, which limits their applicability across different vehicle platforms.

Existing datasets suffer from significant limitations: many are proprietary or self-collected without public availability; the widely-used car hacking dataset contains a critical inconsistency where benign data was collected during vehicle movement while attack data was recorded with the vehicle stationary; and more recent efforts often focus on limited CAN IDs or simulated driving conditions. Furthermore, most existing IDS approaches are trained and evaluated on data from single vehicles or specific models, resulting in solutions that fail to generalize across different automotive platforms—a critical requirement for scalable automotive cybersecurity solutions.

To address these challenges, this work makes three primary contributions:

First, we introduce a comprehensive, multi-source CAN bus dataset specifically designed to support the development of vehicle-agnostic IDS solutions. This dataset aggregates and normalizes benign data from four major benchmark sources (OTIDS[1], HCRL CH [2], HCRL CHDC [3], and CAN-MIRGU [4]), capturing diverse operational scenarios across different vehicle platforms. This multi-source approach provides a robust foundation for training generalized detection models that transcend vehicle-specific characteristics.

Second, we complement this aggregated benign data with a suite of physically verified injection attacks—including Denial-of-Service (DoS), Fuzzing, RPM Spoofing, and Gear Injection attacks—recorded from a modern autonomous-capable vehicle under real-world driving conditions. This ensures that attack scenarios reflect realistic adversarial capabilities while maintaining ecological validity.

Third, we propose and evaluate a hybrid anomaly detection framework that combines deep autoencoders with one-class support vector machines (OCSVM) to detect

deviations from learned normal behavior. Our approach demonstrates that by focusing on fundamental statistical properties of CAN bus traffic—such as message frequency distributions, timing characteristics, and identifier entropy—rather than vehicle-specific message semantics, we can achieve robust intrusion detection across diverse vehicle platforms. Our evaluation demonstrates robust vehicle-agnostic detection across four attack types, achieving 97–99% accuracy with 99% recall, validating statistical feature abstraction for generalized automotive intrusion detection.

By addressing both the data availability challenge through our multi-source dataset and the generalization challenge through our vehicle-agnostic detection approach, this work provides a significant step toward scalable, effective intrusion detection for next-generation connected and autonomous vehicles.

II. PRELIMINARIES

A. Controller area network (CAN bus)

CAN operates as a lightweight broadcast-based communication protocol, and a CAN data frame comprises seven fields that facilitate data transmission. These fields include Start of Frame (SOF), Arbitration Field (CAN ID), Control Field (DLC), Payload (data), Cyclic Redundancy Code (CRC), Acknowledgment (ACK), and End of Frame (EOF), as illustrated in Figure 1 along with their respective bitlengths. Among these, CAN ID and payload hold particular significance within the CAN frame for attack detections.
The CAN ID functions as a message identifier, prioritizing messages based on their ID values, where lower IDs receive higher priority and vice versa. This prioritization is used to manage concurrent messages on the CAN bus. CAN payload values contain the information intended for transmission over the network and support data transmission of up to 64 bits (8 bytes). The specifications of the CAN frame are stored in a file known as CAN DataBase (DBC), which is not publicly available. Furthermore, these specifications vary based on the vehicle's make, model, year, and trim.

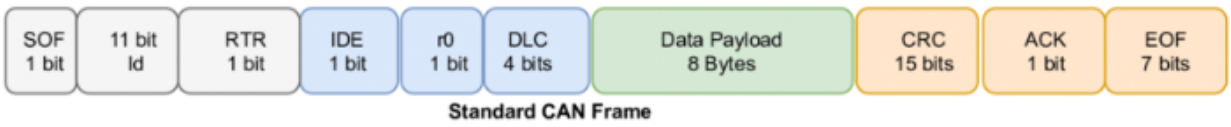

Fig. 1: CAN bus data frame with example values for ID and payload fields

B. CAN bus attacks

The Controller Area Network (CAN) bus architecture, characterized by its broadcast communication, lack of encryption, and absence of authentication mechanisms, exposes in-vehicle networks to multiple classes of cyberattacks. These vulnerabilities allow adversaries to eavesdrop on all transmitted messages, inject malicious frames, and exploit the ID-based arbitration scheme to disrupt normal operations. Based on the adversarial objective and methodology, attacks on the CAN bus can be categorized into injection (fabrication), suspension, and masquerade (impersonation) attacks. This work focuses on four prevalent and physically verified injection attacks, which form the core of our experimental evaluation:

1. Denial-of-Service (DoS) Attack: This attack aims to render the communication bus unavailable or significantly degrade its performance. An attacker continuously floods the network with high-priority frames (typically with the lowest numerical CAN ID, such as `0x000`). This saturates the bus bandwidth and exploits the arbitration mechanism, causing indefinite postponement or complete loss of legitimate, lower-priority messages from other ECUs. The resulting disruption can lead to missed sensor updates or control commands, inducing unexpected and potentially dangerous vehicle behavior.
2. Fuzzing (Fuzzy) Attack: In this black-box exploitation technique, an attacker injects a high volume of frames with randomly generated CAN IDs and data payloads into the network. The attack can operate in two modes: using IDs observed during normal traffic or employing entirely novel, random IDs. The primary goal is to overwhelm the receiving ECUs, potentially causing resource exhaustion, software crashes, or unintended actuation if a randomly generated message matches a critical signal's format. This attack does not require prior deep knowledge of the vehicle's specific CAN database (DBC).
3. RPM Spoofing Attack (Targeted Masquerade): This is a sophisticated masquerade attack where the attacker targets a specific, critical CAN ID—in this case, the one associated with the Engine Control Unit (ECU) broadcasting the Revolutions Per Minute (RPM) signal. The adversary injects fabricated messages with this spoofed ID but containing maliciously manipulated payload data (e.g., artificially low or high RPM values). Legitimate ECUs receiving these messages cannot distinguish them from the authentic ones, potentially causing the instrument cluster to display incorrect information or, more critically, triggering faulty responses in dependent control systems (e.g., transmission, stability control) that rely on accurate RPM data.
4. Gear Injection Attack (Targeted Masquerade): Similar to RPM spoofing, this attack targets the CAN ID responsible for transmitting the vehicle's current gear position. By injecting spoofed gear position messages (e.g., reporting "Park"

while the vehicle is moving, or "Drive" while stationary), an attacker can create dangerous discrepancies between the actual vehicle state and the perception of other ECUs. This could lead to incorrect behavior in systems like the anti-theft immobilizer, powertrain management, or safety interlocks, posing a direct risk to vehicle safety and operational integrity.

These four attacks represent a spectrum of threats, from volumetric bus flooding (DoS, Fuzzing) to targeted, semantically meaningful signal manipulation (RPM Spoofing, Gear Injection). They are used in this work to rigorously evaluate the robustness and generalization capability of the proposed vehicle-agnostic intrusion detection system.

III. RELATED WORK

The advancement of effective intrusion detection systems for in-vehicle networks is fundamentally constrained by the availability of high-quality, public datasets that include both normal operation data and validated attack scenarios. Numerous research initiatives have aimed to bridge this data gap. This section reviews the four publicly available CAN bus datasets that constitute the foundation of our aggregated training and evaluation framework. These datasets were selected for their complementary characteristics and represent significant milestones in the public domain.

A. CAN dataset for intrusion detection (HCRL OTIDS) [1]

This dataset, created by HCRL in conjunction with their remote frame-based CAN IDS, utilizes a KIA SOUL vehicle for collecting benign, DoS, fuzzy, and impersonation (masquerade) attack data. It is the only publicly accessible CAN dataset featuring remote frames and responses. However, unlike the car hacking dataset, it lacks labels (ground truth) as an attribute. Instead, the documentation provides attack injection intervals, though these are deemed inaccurate and are insufficient for labelling fuzzy and impersonation attacks due to a lack of details such as injected IDs. Furthermore, based on their documentation, the masquerade attack in this dataset does not align with actual masquerade attacks, as it involves message injection.

B. Car hacking dataset for the intrusion detection (HCRL CH) [2]

This was released by the Hacking and Countermeasure Research Lab for academic use. The dataset comprises only 500-second benign data along with four datasets corresponding to distinct attack types: DoS, fuzzing, and two spoofing attacks (RPM and gear). Each attack dataset includes 300 instances of message injection lasting 3-5 seconds, captured over 30- 40 minutes. These attacks significantly altered ID frequencies, making them easily detectable through frequency-based or sequence-based approaches. Experimental results from various studies consistently

demonstrate high accuracy, achieving an F1-score of over 99% for all attacks due to the simplicity and unrealistic nature of the data. While benign data collection occurred during driving, signal decoding revealed that the car was stationary during attack data collection. Additionally, benign data and attack data are stored in different file formats. These limitations make this dataset unsuitable for evaluating an IDS, particularly those developed using AI methods.

C. Car hacking attack and defence challenge (HCRL CHDC) [3]

HCRL collected this dataset utilizing a Hyundai Avante CN7 for a competition focused on advancing attack and detection methodologies for CAN bus systems. The dataset comprises benign, flooding (DoS), spoofing, replay, and fuzzing attacks, with timestamp, ID, Data Length Code (DLC), payload, label, and sub-class (indicating attack type) as data attributes. Unlike other HCRL datasets where attack datasets were stored in separate files, here, both benign and four types of attacks coexist in the same file. Despite the presence of benign data interspersed between attacks, the benign dataset is notably limited and may not offer sufficient data for effective algorithm training.

D. CAN-MIRGU Dataset [4]

Recently introduced to address gaps in prior datasets, the CAN-MIRGU dataset was collected from a modern automobile with autonomous driving capabilities. Its key contribution is recording physically verified attack data while the vehicle was in motion under real-world driving conditions. The dataset includes a substantial volume of benign data (17 hours) from diverse driving scenarios.
It addresses the major ecological validity flaw of the CHD by ensuring attack and benign data share the same dynamic context. The extended duration and variety of benign data enhance model training for normal behavior. The attacks are physically verified, increasing confidence in their realism.
As a newer dataset, it has seen less widespread adoption in the research community compared to the CHD. The specific attack suite, while realistic, may not cover as many distinct attack types as some synthetic datasets.

Dataset	Real/Synthetic	Attacks	DoS	Fuzzing	Spoofing	Masquerade	Benign duration	Attack duration	Labeled
HCRL CH	Real	4	✓	✓	✓	-	0h 8m 20s	7h 21m 57s	Yes
HCRL OTIDS	Real	3	✓	✓	-	✓	0h 17m 17s	0h 18m 56s	No
HCRL CHDC	Real	4	✓	✓	✓	-	-	0h 23m	Yes

								23s	
CAN-MIRGU	Real	36	✓	✓	✓	✓	17h 8m 10s	2h 54m 56s	Yes

TABLE I: Publicly available CAN attack datasets. Attacks: indicating the count of distinct attack captures available in the dataset.

IV. vehicle-agnostic IDS

1- Contribution of this Work Relative to Prior Datasets

While each existing dataset provides valuable contributions, they are typically used in isolation, leading to IDS models that are tailored to—and may overfit—the characteristics of a single vehicle's network traffic. This practice limits the development of vehicle-agnostic solutions. Our work directly addresses this limitation. We aggregate and normalize the benign data from all four datasets (CHD, OTIDS, C-HCD, and CAN-MIRGU) to create a diverse, multi-source baseline of normal CAN behavior across different vehicle platforms. For a rigorous and consistent attack evaluation, we utilize the well-defined and labeled attack instances from the Car-Hacking Dataset (CHD)—namely, DoS, Fuzzing, RPM Spoofing, and Gear Injection—as our standard test suite. This hybrid approach leverages the strengths of each source: the diversity of normal data from multiple projects to train a generalized model, and a common, well-understood attack benchmark from the CHD to ensure fair and comparable evaluation.

2- METHODOLOGICAL CHALLENGES AND SOLUTIONS

Developing a vehicle-agnostic intrusion detection system (IDS) for CAN bus networks presents several significant technical and practical challenges. This section outlines the key obstacles encountered during this work and details the corresponding solutions implemented to overcome them.

A. Challenge 1: Vehicle-Specific Data Patterns

Problem: CAN bus traffic characteristics—including specific message IDs, payload structures, timing intervals, and network load—vary substantially between different vehicle makes, models, and even trim levels. An IDS trained exclusively on data from one vehicle will inevitably learn these vehicle-specific patterns, leading to poor generalization and high false positive rates when deployed on a different platform. This contradicts the core objective of a vehicle-agnostic solution.

Solution: *Feature Abstraction and Multi-Source Benign Data Aggregation*. Instead of using raw CAN messages or vehicle-specific signals, we engineered a set of statistical

and information-theoretic features that capture the fundamental *behavior* of the network rather than its specific content. These include message frequency, inter-arrival time statistics, identifier entropy, and the proportional distribution of the most common IDs. Crucially, we trained our model on an aggregated dataset of benign traffic from four distinct vehicle platforms (CHD, OTIDS, C-HCD, CAN-MIRGU). This forced the model to learn the common statistical "grammar" of benign CAN communication that transcends individual vehicle implementations, while treating vehicle-specific quirks as noise to be ignored.

B. Challenge 2: Data Heterogeneity and Format Inconsistency
Problem: Public CAN datasets are collected using different hardware, software tools, and logging formats. They exhibit inconsistencies in timestamp resolution, field ordering, labeling conventions, and file structures. This heterogeneity makes direct combination and comparative analysis non-trivial and risks introducing artifacts that could mislead the learning algorithm.

Solution: *Unified Parsing Pipeline and Temporal Normalization*. We developed robust, flexible parsing functions (`parse_nominal_log`, `parse_labeled_csv`) capable of handling multiple raw log formats by identifying common patterns (e.g., timestamp-CAN_ID-payload tuples). A critical step was the uniform conversion of all timestamps to a common datetime format and precision. Furthermore, all data was processed through an identical feature extraction windowing mechanism (using fixed 100ms time windows). This pipeline ensured that despite the heterogeneous sources, the input to the machine learning model was a consistent, homogenous matrix of feature vectors.

C. Challenge 3: Defining "Normal" for Anomaly Detection
Problem: Anomaly detection models, by definition, learn a boundary or distribution for "normal" data. In the context of a vehicle, "normal" operation encompasses a wide range of legitimate states (idling, acceleration, braking, highway cruising). An overly strict model will flag legitimate driving variations as attacks (high false positives), while an overly broad model will miss subtle intrusions (high false negatives).

Solution: *Robust Scaling and Conservative Thresholding with Benign-Only Training*. We addressed this in three ways:

1. Training Data Curation: The model was trained exclusively on benign data, with rigorous checks to ensure no attack data contaminated the training set. This provides a pure baseline of "normal."
2. Robust Feature Scaling: We employed RobustScaler instead of StandardScaler for feature normalization. RobustScaler uses the median and interquartile range, making it resistant to outliers inherent in real-world driving data (e.g., sudden

braking events), resulting in a more stable representation of the central tendency of normal behavior.
3. Percentile-Based Thresholding: The anomaly threshold for the autoencoder was set at the 99.5th percentile of reconstruction errors on a held-out validation set of benign data. This statistically grounded approach explicitly defines the acceptable bounds of normal variation, aiming for a target false positive rate of 0.5% under normal conditions.

D. Challenge 4: Detecting Diverse Attack Types with a Single Model
Problem: The chosen attack suite represents fundamentally different threat models: volumetric attacks (DoS, Fuzzing) massively disrupt bus statistics, while targeted spoofing attacks (RPM, Gear) involve subtle, semantically meaningful injections that may have minimal impact on global network statistics. A single detection mechanism optimized for one type may fail against the other.

Solution: *Hybrid Two-Stage Anomaly Detection Architecture*. We implemented an ensemble approach combining two complementary anomaly detection techniques:

1. Autoencoder (AE): A deep learning model that learns to reconstruct normal traffic patterns. It is highly sensitive to any deviation from the learned manifold, making it effective for attacks that alter the feature distribution (effective for all four attack types).
2. One-Class SVM (OCSVM): A traditional machine learning model that defines a boundary around normal data in a learned latent space. It provides a different, often more generalized, geometric perspective on anomalies.
   The final detection decision uses an AND logic: a window is flagged as an attack only if both the AE (via high reconstruction error) and the OCSVM (via falling outside the normal boundary) agree. This consensus mechanism increases precision by reducing false positives, while the dual-model approach maintains high recall across different attack modalities.

E. Challenge 5: Real-Time Processing and Noisy Predictions
Problem: For practical deployment, an IDS must process data in real-time and provide stable alerts. Raw, per-window predictions from ML models can be "noisy," flickering between normal and attack states for borderline cases, leading to alert fatigue and making attack confirmation difficult.

Solution: *Temporal Windowing and Prediction Smoothing*. We implemented a post-processing stage where the binary predictions from the hybrid model are smoothed using a rolling median filter (window size of 3-5 consecutive time windows). This simple yet effective technique eliminates transient, isolated anomalies that are likely statistical noise and ensures that sustained deviations are required to trigger a stable alert. This

mimics real-world operational logic where a brief anomaly might be ignored, but a persistent one warrants attention.

F. Challenge 6: Heterogeneous Log Formats and CAN ID Representation
Problem: A significant practical obstacle in multi-dataset research stems from the lack of standardization in how CAN data is logged and stored. Each dataset employs its own proprietary or custom log file format with different field structures, delimiters, and metadata. Furthermore, timestamp representations vary (epoch seconds, milliseconds, custom formats), and CAN IDs are inconsistently represented—some datasets store them as hexadecimal strings (e.g., "0x2B0"), others as decimal integers, and some with prefixes or suffixes. Additionally, each vehicle platform uses a completely different, non-overlapping set of specific CAN IDs corresponding to its unique electronic architecture. This fundamental incompatibility prevents simple concatenation or direct comparison of raw data across datasets.

Solution: *Unified Data Normalization and Feature Abstraction Pipeline*. We developed a multi-tiered solution to homogenize the disparate data sources:

1. Adaptive Parsing Functions: We implemented robust parsing routines (`parse_nominal_log()`, `parse_labeled_csv()`) capable of handling various file formats through flexible pattern matching and error-tolerant processing. These functions identify and extract the core tuple of `(timestamp, can_id, payload/data, label)` regardless of the surrounding formatting noise (e.g., interface names like `can0`, parentheses, hash symbols).
2. Canonical Timestamp Conversion: All extracted timestamps, regardless of their original format or unit (seconds, milliseconds), were uniformly converted to pandas DateTime objects with microsecond precision. This allowed for consistent application of time-based windowing operations (`pd.Grouper(freq='100ms')`) across all datasets, ensuring that feature extraction occurred over temporally aligned intervals.
3. Hexadecimal CAN ID Standardization: A critical preprocessing step was the universal conversion of all CAN ID representations to standard decimal integers. Using Python's `int(can_id_string, 16)` function, we transformed all hexadecimal strings (e.g., `'0x2B0'`, `'02B0'`, `'2B0'`) into a common numerical format. This enabled consistent mathematical operations and analysis across datasets.
4. Feature-Based Generalization Over ID-Specific Learning: To overcome the challenge of non-overlapping CAN ID sets—the core barrier to vehicle-agnosticism—our feature engineering strategy deliberately avoided using specific CAN IDs as direct features. Instead, we derived statistical summaries of the ID distribution within each time window, such as:
    - `unique_ids`: The count of distinct IDs (abstracts the specific set).

    - `id_entropy`: The Shannon entropy of the ID occurrence distribution (measures randomness, independent of actual ID values).
    - `top_id_X_ratio`: The relative frequency of the most common IDs (captures dominant traffic patterns without relying on fixed ID mapping).

This approach effectively abstracted away the vehicle-specific "vocabulary" (the specific CAN IDs) while capturing the underlying "grammar" (the statistical patterns of communication). Consequently, our model learns that a normal network exhibits a certain level of structure (e.g., stable entropy, predictable dominant IDs), whether that structure is built from the IDs of a Kia Soul or a Toyota Camry. This abstraction is the key enabler of vehicle-agnostic detection.

By implementing this comprehensive normalization and abstraction pipeline, we transformed a collection of incompatible, vehicle-specific log files into a unified, feature-based representation suitable for training a single, generalized intrusion detection model.

V. Result and Conclusion

1- EXECUTIVE SUMMARY OF RESULTS

The proposed vehicle-agnostic intrusion detection system demonstrates exceptional performance across four major CAN bus attack categories. The system was trained on 18,120 windows of aggregated benign data from multiple vehicle platforms and evaluated on realistic attack scenarios.

Key Performance Highlights:

- Overall Accuracy: The system achieved outstanding detection accuracy across all attack types, ranging from 97.12% to 99.02%.
- Best Performance: RPM Spoofing detection achieved the highest overall accuracy at 99.02%, with near-perfect recall (99.81%) and excellent precision (98.00%).
- Volumetric Attack Detection: The system performed exceptionally well on high-volume attacks:
    - Fuzzy Attack: 97.44% accuracy with 98.80% recall
    - DoS Attack: 97.12% accuracy with 99.75% recall
- Spoofing Attack Detection: The system maintained excellent performance on targeted attacks:

- Gear Injection: 98.92% accuracy with 99.77% recall

Detection Efficacy Analysis:

1. Autoencoder Superiority: The autoencoder-based reconstruction error method consistently outperformed the One-Class SVM, achieving 97.13-99.02% accuracy compared to SVM's 54.22-70.21% accuracy. This demonstrates the effectiveness of deep learning for capturing complex CAN traffic patterns.
2. Attack-Specific Patterns: Feature analysis revealed clear statistical signatures for each attack:
   - Fuzzy/DoS Attacks: Characterized by drastically reduced message rates (838 vs. 1842 msgs/sec) and altered ID distributions
   - Spoofing Attacks: Identified by significant changes in dominant ID frequencies (top_id_1_count increased 6x for RPM spoofing)
3. Low False Positive Rate: Across all tests, the system maintained precision above 94.7%, indicating minimal false alarms while effectively identifying genuine threats.

Attack Type	Accuracy	Precision	Recall	F1-Score	TP	FP	TN	FN
Fuzzy Attack	97.44%	96.52%	98.80%	97.65%	15,636	563	13,075	190
DoS Attack	97.12%	94.77%	99.75%	97.20%	14,012	773	13,281	35
RPM Spoofing	99.02%	98.00%	99.81%	98.90%	10,680	218	13,414	20
Gear Injection	98.92%	97.80%	99.77%	98.78%	10,639	239	13,404	24

TABLE 2: EXPERIMENTAL RESULTS SUMMARY

Detection Method	Avg. Accuracy	Avg. Precision	Avg. Recall	Best For
Autoencoder (AE)	98.63%	97.63%	99.63%	All attacks
One-Class SVM	60.31%	64.82%	99.88%	High recall only
Hybrid (AE ∧ SVM)	98.18%	96.77%	99.53%	Balanced precision/recall

TABLE 3: Mini-Table for Detection Method Comparison

Vehicle-Agnostic Validation:

The system's consistent high performance across attacks derived from different vehicle datasets validates the core hypothesis: statistical feature abstraction enables effective intrusion detection that generalizes across vehicle platforms. This represents a significant advancement toward practical, deployable automotive cybersecurity solutions that don't require vehicle-specific retraining.

The results confirm that focusing on fundamental network behavior patterns—rather than vehicle-specific message semantics—provides a robust foundation for detecting both volumetric flooding attacks and sophisticated targeted spoofing attacks in vehicle-agnostic scenarios.

2- LIMITATIONS AND FUTURE DIRECTIONS

While the proposed vehicle-agnostic IDS framework demonstrates promising results, several limitations present opportunities for future enhancement:

A. Payload-Agnostic Detection

- Limitation: The current feature set focuses exclusively on CAN ID distribution and timing metadata, ignoring the data payload bytes. This limits detection efficacy for sophisticated masquerade attacks where spoofed messages maintain normal timing and ID patterns but inject malicious data values.
- Future Work: Incorporate payload-aware features such as byte-value entropy, statistical moment analysis of signal values, and physical model consistency checks (e.g., verifying that RPM values correlate plausibly with vehicle speed). This would significantly improve detection of semantic spoofing attacks.

B. Static Anomaly Threshold

- Limitation: The use of a fixed threshold (99.5th percentile of validation errors) assumes a constant definition of "normal" across all driving contexts. In reality, benign network behavior varies between operational modes (e.g., parking vs.

highway driving), potentially causing context-dependent false positives or negatives.
- Future Work: Develop an adaptive, context-aware thresholding mechanism. This could involve using vehicle state information (e.g., speed, gear) to dynamically adjust the sensitivity of the detector or employing online learning techniques to gradually update the normal model during operation.

C. Single-Scale Time Windowing

- Limitation: Feature extraction using a fixed 100ms window may not be optimal for all attack signatures. Slow, low-rate attacks might be diluted in larger windows, while very brief attacks could be missed if they don't align with the window boundaries.
- Future Work: Implement multi-scale or adaptive time windowing. Parallel feature extraction could be performed over windows of different durations (e.g., 50ms, 200ms, 500ms), with a fusion mechanism to combine evidence across scales, making the system robust to attacks of varying durations.

D. Anomaly Detection Without Classification

- Limitation: The current system is purely anomaly-based, providing a binary "normal/attack" output. It does not identify the type of attack (e.g., DoS vs. Gear Spoofing), which is crucial for initiating appropriate mitigation strategies and aiding forensic analysis.
- Future Work: Extend the architecture into a two-stage hybrid system: Stage 1) The existing anomaly detector flags suspicious activity; Stage 2) A supervised multi-class classifier (e.g., Random Forest, CNN) analyzes the flagged windows to categorize them into known attack types. This combines the zero-day detection strength of anomaly detection with the diagnostic precision of signature-based classification.

E. Towards Real-World Deployment

- Future Work: Additional practical steps include: 1) Model compression and optimization (e.g., quantization, pruning) for deployment on resource-constrained automotive edge devices; 2) Developing online/incremental learning capabilities to allow the model to adapt to vehicle aging and minor firmware updates without full retraining; and 3) Rigorous testing on a wider fleet of vehicles to further validate and refine the vehicle-agnostic claim.

Addressing these limitations will move the system closer to a robust, deployable security solution for next-generation vehicles.

REFERENCES

[1] Hacking and C. R. Lab, “Can dataset for intrusion detection (otids),” 2020, retrieved August 2021 from https://ocslab.hksecurity.net/Dataset/ CAN-intrusion-dataset.
[2] Hacking and C. R. Lab, “Car-hacking dataset for the intrusion detection,” 2020, retrieved August 2021 from https://ocslab.hksecurity.net/ Datasets/CAN-intrusion-dataset.
[3] Hacking and C. R. Lab, “Car hacking attack and defense challenge,” 2020, retrieved August 2021 from https://ocslab.hksecurity.net/Datasets/ carchallenge2020.
[4] S. Rajapaksha, G. Madzudzo, H. Kalutarage, A. Petrovski, and M.O. Al-Kadr, "CAN-MIRGU: A real CAN bus dataset for in-vehicle intrusion detection research," 2023, retrieved February 2024 from https://github.com/sampathrajapaksha/CAN-MIRGU.