IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

Cybersecurity in Automotive Vehicles

1st Sagar Dhanake, 2nd Ruturaj A. Wakchaure, 3rd Vedanti G. Shende,
4thSiddhant R. Ghungarde, 5th Linesh S. Sonawane
Assistant Professor, Department of Computer Engineering[1]
UG Students, Department of Computer Engineering [2,3,4,5]
Dr. D. Y. Patil College of Engineering and Innovation, Varale
Corresponding Author email-id: wakchaureruturaj3012@gmail.com

Abstract: *Modern vehicles are increasingly connected and intelligent; however, this connectivity introduces significant vulnerabilities, making them more susceptible to cyberattacks. This study presents the process of building a real-time vehicle security monitoring system from the ground up, employing an ESP8266 microcontroller and several affordable sensors. The proposed system simultaneously monitors five parameters sound levels, vibrations, temperature, physical touch, and movement using a three-axis accelerometer. All data are transmitted to a web-based dashboard that features a distinctive hexagonal layout, is updated in real time, and flashes red when anomalies are detected. A simulated cyber-attack trigger was also integrated to demonstrate the system's response to attempted network tampering. The entire system is cost-effective, exhibits a response time of well under one second, and operates reliably in practice. This implementation addresses a gap in the literature: while numerous research papers discuss vehicle cybersecurity theoretically, there is a lack of practical demonstrations detailing the construction of tangible and testable solutions.*

Keywords: Internet of Things (IoT), ESP8266, Vehicle Security Monitoring, Cybersecurity, Intrusion Detection, Multi- Sensor Integration, Embedded Web Server, Real-Time Dashboard

I. INTRODUCTION

The modern automotive ecosystem is undergoing a paradigm shift from purely mechanical systems to highly interconnected and software-defined platforms. Automated vehicles (AVs) leverage advanced sensors, communication modules, and artificial intelligence to provide safer and more efficient transportation. However, this heightened connectivity exposes vehicles to a broad spectrum of cybersecurity threats, including CAN bus spoofing, denial-of-service (DoS) attacks, firmware manipulation, and remote exploitation through wireless interfaces. The Controller Area Network (CAN) bus, which serves as the primary in-vehicle communication backbone, was designed for reliability and real-time performance rather than for security. It lacks built-in encryption, sender authentication, and message integrity verification, rendering it inherently vulnerable to attacks. While significant research has been devoted to intrusion detection systems (IDS) and cryptographic protocols for CAN networks, a notable gap remains between theoretical defense frameworks and tangible, deployable prototypes that researchers and educators can use for validation and demonstration. This study addresses this gap by presenting a fully implemented IoT-based vehicle monitoring system that:

1. Integrates five distinct sensor modalities (sound, vibration, temperature, touch, and accelerometer) on a single ESP8266 NodeMCU platform.
2. Provides real-time anomaly detection through hardware interrupts and threshold-based classification.
3. Hosts a lightweight embedded web server with dedicated API endpoints for sensor data retrieval and cyber-attack simulation.
4. Delivers a responsive, hexagonal dashboard interface with CSS animations and JavaScript-driven real-time updates.
5. Demonstrates a practical, reproducible architecture that can serve as a baseline for advanced vehicular cybersecurity experiments.

DOI: 10.48175/IJARSCT-37252

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

II. DESIGN AND IMPLEMENTATION

A. Real time dashboard

This image shows the actual hardware implementation of the Cyber Security Automotive Vehicle Monitoring System prototype, where all components are connected to a breadboard for real-time sensing and data processing. At the center of the setup, an NodeMCU (ESP8266) microcontroller was mounted on a breadboard, acting as the main controller responsible for collecting and processing data from multiple sensors. Several modules are connected around the controller using jumper wires, which provide power, ground, and signal connections. At the top left, a sound sensor module (with a microphone) is used to detect noise levels inside or around the vehicle, whereas at the top right, a vibration/knock sensor module is used to sense physical disturbances or impacts. A long black wired probe sensor connected at the bottom was used for temperature measurement, allowing the monitoring of environmental or engine-related temperature conditions. Additional small modules connected through ribbon cables represent sensors such as touch or motion/accelerometer units, which help detect physical interactions and movements along three axes (X, Y, Z). The breadboard serves as a base for quick and flexible circuit assembly without soldering, making it suitable for prototyping. Overall, this setup continuously collects sensor data, such as sound intensity, vibration status, temperature, and motion, which are then processed by the NodeMCU to identify abnormal conditions or potential security threats. The processed data are later transmitted to a monitoring interface or dashboard, where system parameters and security status (SAFE/ALERT/SECURE) are displayed in real time.

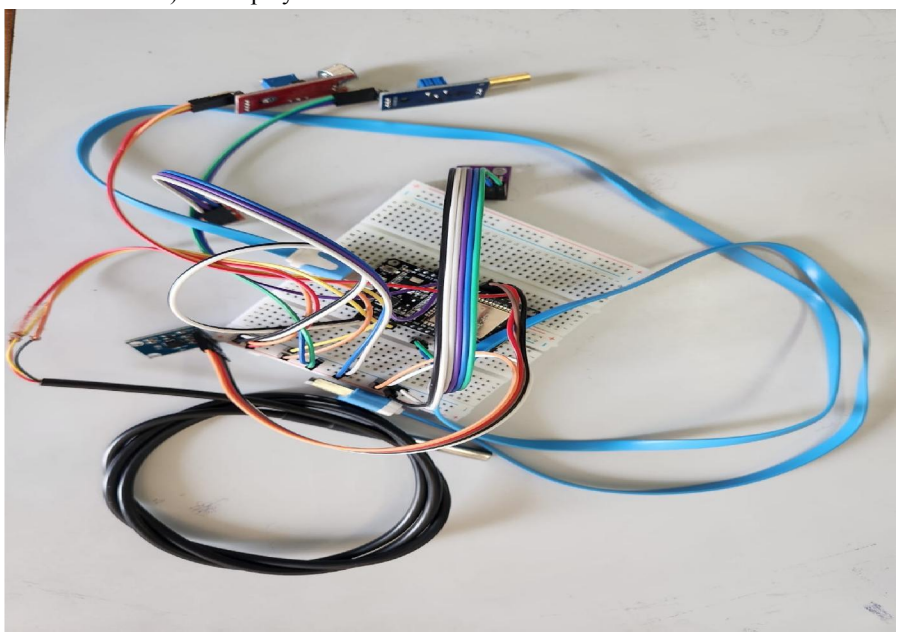

Fig. 1. Hardware

DOI: 10.48175/IJARSCT-37252

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

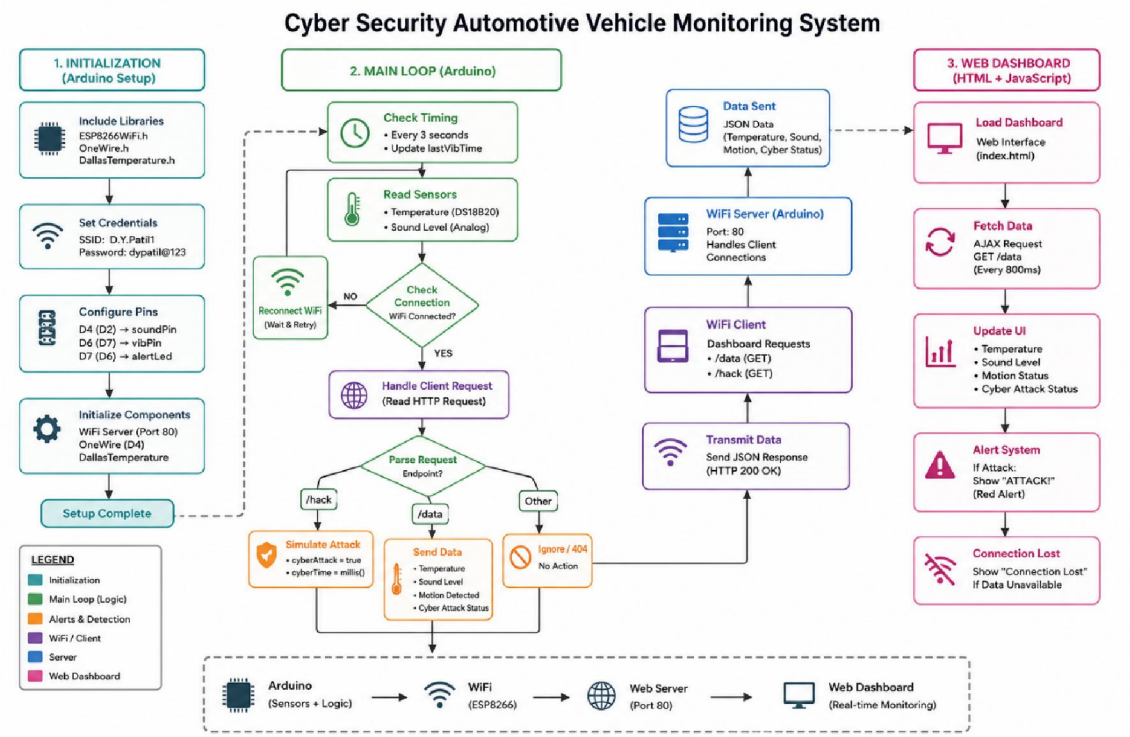

Fig.2. Project flow chart

The ESP8266 functions as a Wi-Fi server that handles requests from the dashboard. When a client accesses the /data endpoint, the system generates and sends a JSON response containing the sensor information, including the temperature, sound level, motion status, and cyber-attack status. If the /hack endpoint is triggered, the system simulates a cyber-attack by setting an attack flag, allowing users to test the alert and monitoring mechanisms of the system.

A web dashboard developed using HTML and JavaScript provides a user-friendly interface for vehicle monitoring. The dashboard periodically sent AJAX requests approximately every 800 ms to retrieve the latest data from the Arduino server. The received information is displayed in real time, allowing users to monitor the temperature, sound intensity, motion detection, and overall cybersecurity status from any connected device.

The system also includes an alert mechanism to improve the security awareness. When a cyber-attack is detected or simulated, the dashboard displays a prominent red warning message labeled “ATTACK!” to notify the user immediately. Additionally, if communication with the server is interrupted, the dashboard shows a “Connection Lost” notification, indicating that the sensor data are currently unavailable. Overall, the system combines sensor monitoring, wireless networking, cyberattack detection, and real-time visualization to enhance vehicle safety and cybersecurity.

III. OBJECTIVES

The primary objective of this study was to design, implement, and evaluate a cost-effective, real-time IoT-based cybersecurity monitoring system for automotive vehicles using the ESP8266 microcontroller platform. To achieve this, this study aims to integrate a diverse array of five distinct sensors comprising sound, vibration, temperature, touch, and a three-axis accelerometer into a unified hardware architecture capable of continuously monitoring the physical security parameters of a vehicle. Furthermore, this study focuses on developing efficient embedded firmware that employs interrupt-driven logic to ensure rapid, low-latency anomaly detection without overwhelming the microcontroller’s processing capacity. Complementing this hardware design, this study aims to develop a lightweight embedded web

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

server that can serve both a structured JSON API for seamless machine-to-machine data exchange and a responsive hexagonal web dashboard, providing users with intuitive, real-time visual alerts without the need for external software installations. Additionally, this study aims to validate a functional cyber-attack alert pipeline through simulated intrusion endpoints, demonstrating the system's capability to propagate threat states across the entire architecture and execute automatic recovery. Ultimately, this study aims to deliver a highly affordable and easily reproducible prototype that serves as a practical and extensible foundation for future advancements in automotive cybersecurity research and education.

IV. LITERATURE REVIEW

Sr. No	Author / Year	Title	Methodology	Outcome
1	Koscher et al. (2010)	Experimental Security Analysis of a Modern Automobile	Demonstrated CAN bus vulnerabilities by injecting malicious messages into vehicle systems.	Showed that attackers can gain unauthorized control over vehicle functions.
2	Miller and Valasek (2015)	Remote Exploitation of Jeep Cherokee	Performed a real-world remote attack through infotainment systems and CAN communication.	Proved remote attacks on vehicles are possible and dangerous.
3	Seo, Kim, Lee (2021)	Machine Learning-Based Intrusion Detection System for In-Vehicle Networks	Implemented ML-based IDS to monitor abnormal CAN traffic patterns.	Improved detection accuracy of cyberattacks in automotive systems.
4	Patel and Gupta (2022)	Lightweight Cryptographic Protocol for Secure CAN Communication	Used lightweight encryption and authentication methods for CAN messages.	Enhanced secure communication without affecting real-time performance.
5	Singh and Sharma (2022)	A Review of Automotive Cybersecurity: Challenges and Solutions	Reviewed cybersecurity threats, vulnerabilities, and protection methods in autonomous vehicles.	Provided awareness about automotive cyber threats and solutions.
6	Verma and Raj (2023)	Development of CAN Intrusion Detection Using Raspberry Pi	Developed IDS implementation using Raspberry Pi for CAN monitoring.	Achieved low-cost real-time threat detection system.
7	Zhang and Chen (2023)	Hybrid IDS for Vehicle Networks using Deep Learning	Applied deep learning techniques for anomaly detection in vehicle communication.	Enhanced intrusion detection performance using AI models.

Table.1. Literature survey

V. CONTROLLER AREA NETWORK (CAN) OVERVIEW

The Controller Area Network (CAN) in a vehicle function much like the human nervous system, enabling seamless communication between different electronic components known as Electronic Control Units (ECUs). These ECUs manage and coordinate various vehicle systems, including the engine, steering, braking, and infotainment systems, ensuring that all parts of the vehicle work together efficiently and in real time.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

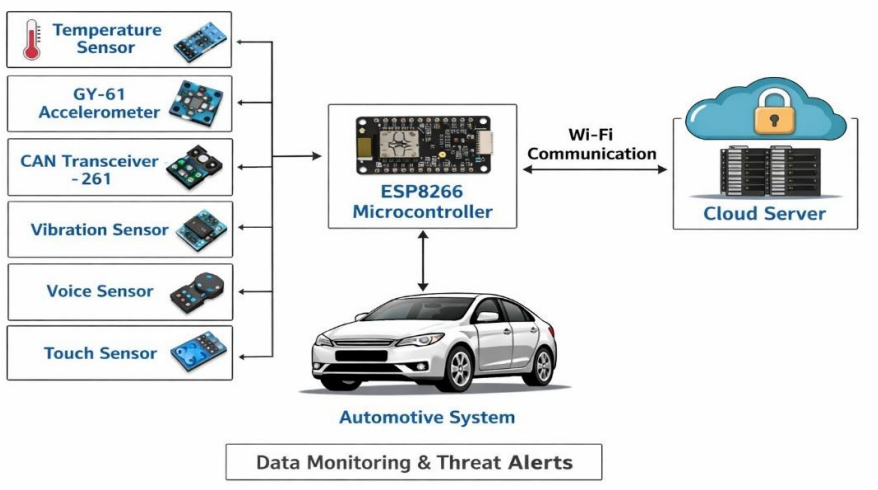

Fig. 6. CAN Architecture

VI. CYBERSECURITY THREATS IN AUTOMATED VEHICLES

Automated vehicles face several types of cybersecurity threats because they are connected and rely on electronic networks. Some of the main attack types are as follows:

1. Spoofing: Hackers send fake messages on the CAN bus, pretending to be a legitimate ECU. For example, sending a false braking message to a car can confuse the system.
2. Denial of Service (DoS): An attacker floods the CAN network with fake messages. This slows down or stops real messages from being delivered, which can prevent critical systems from functioning properly.
3. Remote Attacks: Hackers exploit wireless connections, such as Wi-Fi, Bluetooth, or cellular networks, to access a car's systems remotely. For example, controlling the infotainment system or sending commands to ECUs from a distance.
4. Firmware Hacks: Attackers modify the software inside ECUs, which can give them permanent control of certain functions. For example, the engine or braking software can be changed to behave incorrectly.
5. Sensor and Actuator Manipulation: Automated vehicles use sensors (such as radar, LiDAR, and cameras) to detect the environment and make decisions. Hackers can spoof the sensor data or send incorrect signals to the ECUs. For example, faking distance readings from a radar sensor could cause a car to accelerate into traffic or ignore obstacles. Manipulating sensors can bypass normal CAN security because the system relies on sensor inputs.
6. Physical Access Attacks: If hacker gains direct access to the CAN bus (e.g., via the OBD-II port), they can send malicious messages or reprogram ECUs. Physical access attacks are less common but extremely dangerous because they bypass all wireless security measures.

DOI: 10.48175/IJARSCT-37252

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

VII. SYSTEM ARCHITECTURE AND RESULTS

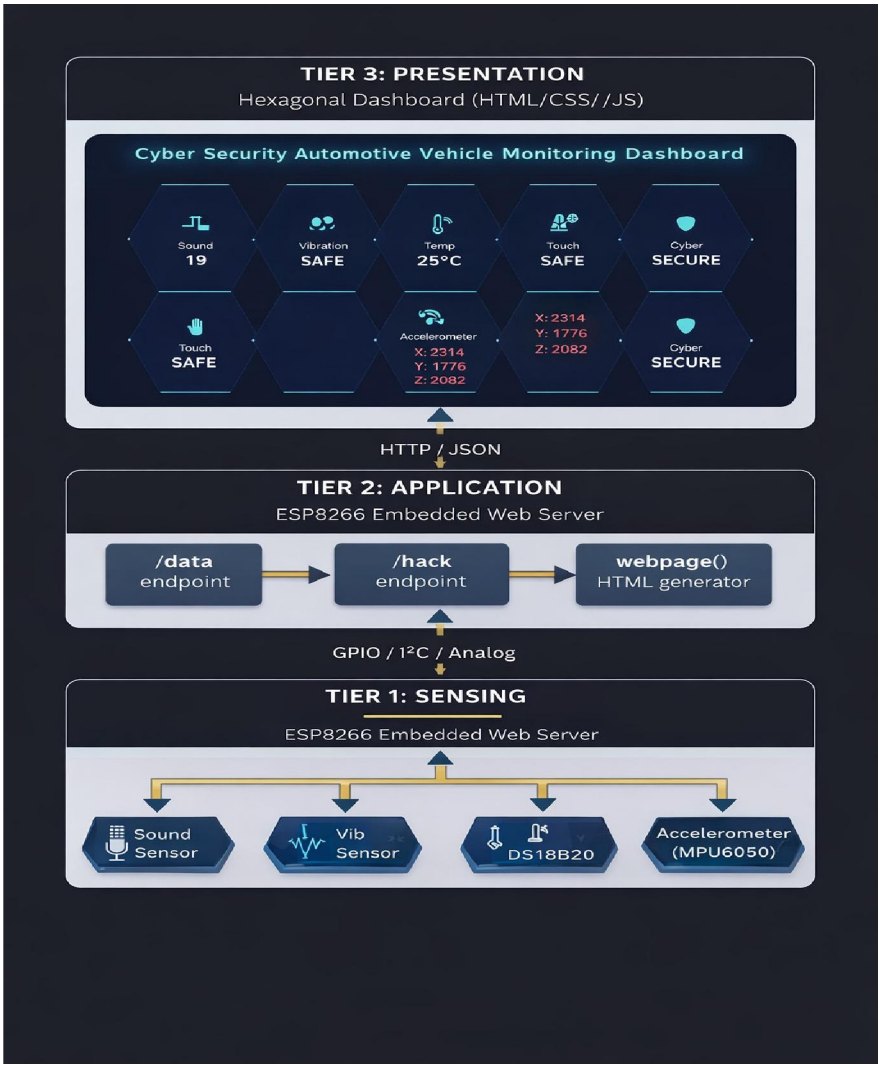

Fig. 7. System Architecture

Tier 1: Sensing Layer:

This is the hardware layer, where data are collected from the sensors.

Sensors used:

1. Sound Sensor → detects noise levels
2. Vibration Sensor → detects motion or tampering
3. DS18B20 Temperature Sensor → measures temperature
4. Accelerometer (MPU6050) → detects movement along the X, Y, and Z axes.

The communication methods include GPIO, I2C, and Analog inputs. All sensor data is sent to the ESP8266 controller.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

Tier 2: Application Layer:
This layer is handled by the ESP8266 embedded web server. It processes sensor data and provides endpoints:
1) /data → sends real-time sensor data in JSON format
2) /hack → simulates or detects cyber-attacks
3) webpage () → generates the dashboard inter
Acts as a bridge between hardware and the user interface.
Tier 3: Presentation Layer:
This is the user interface (dashboard) built using HTML, CSS, and JavaScript.
Displays real-time data in a hexagonal UI design:
1. Sound level
2. Vibration status (SAFE/DETECTED)
3. Temperature
4. Touch status
5. Cyber security status (SECURE/ATTACK)
6. Accelerometer readings (X, Y, Z)
Data is updated using HTTP/JSON communication.

A. Results:

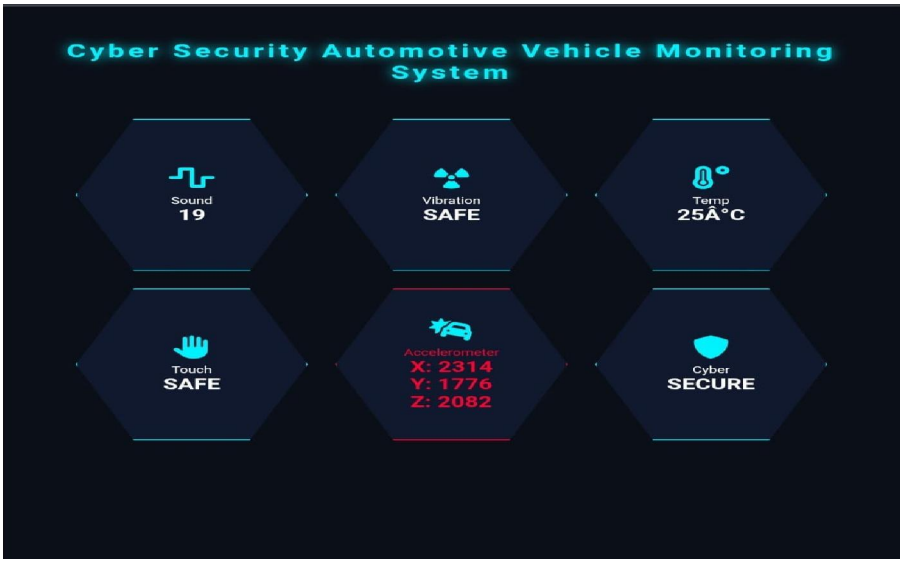

Fig. 8. Result

VIII. CHALLENGES IN SECURING CAN NETWORKS INAUTOMATED VEHICLES

A. ESP8266 ADC Limitations: The ESP8266's single 10 bit ADC (0-1023 range) is shared with the Wi-Fi module, introducing noise when reading analog sensors during active data transmission. This was mitigated by adding a 100nF decoupling capacitor across the sound sensor output and performing 10-sample averaging in software.
B. I²C Address Conflicts: The MPU6050 accelerometer uses I²C addresses 0x68 or 0x69 (depending on the AD0 pin state). Care was taken to ensure no address conflicts with other I²C devices. Future expansions adding more I²C sensors would require an I²C multiplexer (e.g., TCA9548A).

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

C. Lack of CAN Bus Integration: The current prototype monitors physical security parameters but does not interface with an actual CAN bus. Integrating an MCP2515 CAN bus transceiver module with the ESP8266 would enable direct CAN traffic monitoring and IDS deployment, which is planned as future work.

D. No Data Persistence: The system does not log sensor data to persistent storage (e.g., SD card or cloud database). All data is ephemeral, displayed only in real-time. Adding data logging would enable post-incident forensic analysis. Give me the proper and simple way.

IX. APPLICATIONS

1. Anti-Theft Monitoring: Detects unauthorized access using sound, vibration, and touch sensors.
2. Parking Surveillance: Monitors vehicle condition when parked and sends alerts.
3. Impact Detection: Identifies collisions or sudden movements using accelerometer data.
4. Environmental Monitoring: Tracks temperature to prevent overheating or fire risks.
5. Cybersecurity Learning: Useful for understanding automotive security concepts.
6. Research Platform: Supports testing of new sensors and security techniques.
7. Fleet Security: Can be expanded to monitor multiple vehicles in a network.

X. CONCLUSION AND FUTURE WORK

This paper presented the complete design, implementa2tion, and evaluation of an IoT-based cybersecurity automotive vehicle monitoring system built on the ESP8266 platform. The system integrates five heterogeneous sensors, a RESTful embedded web server, and a responsive hexagonal dashboard to deliver real-time vehicular security monitoring. Experimen`tal results confirm reliable sensor operation, sub-120ms API response times, and intuitive visual feedback through the web interface.

The principal contributions of this work are:

1) A practical, reproducible hardware-software prototype bridging the gap between theoretical CAN security research and deployable IoT systems.

2) A dual-endpoint web server architecture that serves both human operators (HTML dashboard) and automated clients (JSON API).

3) An interrupt-driven anomaly detection approach that minimizes response latency for critical security events.

4) A distinctive hexagonal dashboard design that enhances situational awareness through color-coded alert animations.

XI. FUTURE WORK

1) CAN Bus Integration: Add an MCP2515 CAN transceiver to enable direct CAN traffic monitoring and implement a lightweight IDS using message frequency and timing analysis.

2) Cloud Connectivity: Integrate MQTT or HTTP push to cloud platforms (AWS IoT, Firebase) for remote monitoring and alert notifications.

3) Machine Learning: Deploy TinyML models on the ESP8266 (or migrate to ESP32) for anomaly detection based on accelerometer and sound patterns.

4) OTA Updates: Implement secure over-the-air firmware updates to maintain cybersecurity throughout the system lifecycle.

5) Mobile Application: Develop a companion mobile app for push notifications and remote vehicle status monitoring.

REFERENCES

[1] Seo, J., Kim, H., Lee, Y. (2021). Machine Learning-Based Intrusion Detection System for In-Vehicle Networks. IEEE Access.

[2] Patel, M., Gupta, S. (2022). Lightweight Cryptographic Protocol for Secure CAN Communication. Journal of Information Security and Applications.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 10, June 2026
Impact Factor: 8.2

[3] Singh, R., Sharma, P. (2022). A Review of Automotive Cybersecurity: Challenges and Solutions. International Journal of Emerging Technologies.
[4] Verma, N., Raj, T. (2023). Development of CAN Intrusion Detection Using Raspberry Pi. International Conference on Smart Systems and IoT.
[5] Zhang, L., Chen, W. (2023). Hybrid IDS for Vehicle Networks using Deep Learning. Elsevier–Vehicular Communications Journal.
[6] Mehta, S., Rao, D. (2024). Secure Vehicle Communication Using Arduino and Raspberry Pi. IEEE International Symposium on Connected Vehicles.
[7] Khan, M. H., Javed, A. R., Iqbal, Z., Asim, M., & Awad, A. I. (2024). DivaCAN: Detecting In-Vehicle Intrusion Attacks on a Controller Area Network Using Ensemble Learning. Computers & Security, 139, 103712.
[8] Bathla, G., et al. (2023). An AI-powered security system for CAN bus attacks verification in electric vehicles. Journal of Electrical Systems, 19(1).
[9] Mokhtar, B., & Azab, M. (2020). Survey on Security Issues in Vehicular Ad Hoc Networks. IEEE Communications Surveys & Tutorials.
[10] Amato, F., et al. (2021). A machine learning approach for intrusion detection in automotive networks. Future Generation Computer Systems.
[11] Koscher, K., et al. (2010). Experimental Analysis of a Modern Automobile. IEEE Symposium on Security and Privacy. (The "foundational" study mentioned in your report).
[12] Miller, C., & Valasek, C. (2015). Remote Exploitation of an Unaltered Passenger Vehicle. Black Hat USA. (The landmark Jeep Cherokee hack mentioned in your report).
[13] Dangwal, S., & Wazid, M. (2024). ACID-CAN: A Lightweight and Efficient Intrusion Detection System for CAN Bus. IEEE Internet of Things Journal.
[14] Luo, Y., et al. (2023). In-vehicle network intrusion detection systems: A comprehensive survey and taxonomy. PeerJ Computer Science.
[15] Upstream Security (2024). Global Automotive Cybersecurity Report. (Industry standard report for recent attack statistics cited in modern research).