Advance CAN Bus Car Diagnostics Tool

Project Summary:
I'm building a compact, ESP32-based OBD-II tool that plugs directly into a car's diagnostic port to analyze and reverse-engineer CAN bus communication. Unlike standard diagnostic tools, it will not only read engine codes and sensor data but also sniff, log, and inject CAN messages to identify cybersecurity vulnerabilities (e.g., spoofing speed or unlocking doors). The custom hardware will include safety features like read-only mode by default, while the software will help map unknown CAN signals to real-world actions. The goal is to create an open-source platform for automotive security research and ethical hacking.

FUNCTIONAL REQUIREMENT

HARDWARE

Component	Function
Microcontroller (ESP32-Wroom)	Processing, Wi-Fi, and BLE connectivity
MCP2515 CAN Controller	CAN bus communication
TJA1050 CAN Transceiver	Converting SPI to CAN signals
Logic Level Shifter	Ensuring safe signal conversion between ESP32 and MCP2515
Switch (Read/Read-Write Mode)	Allows toggling between passive and active CAN message interaction
OBD-II Connector	Connecting to the vehicle CAN bus
Laptop	Running software for logging, debugging, and analysis

SOFTWARE

Component	Function
ESP32 Firmware (Arduino/PlatformIO)	Handles low-level CAN communication
CAN Libraries	Provides CAN bus support and parsing tools
OBD-II Diagnostic PIDs	Stores and references vehicle-specific OBD-II data
DBC Files	Database files containing CAN signal descriptions for decoding messages.
Wi-Fi/Bluetooth Serial Terminal	Interfaces for communication between the microcontroller and external devices.
SavvyCAN	GUI tool for analyzing, logging, replaying, and filtering CAN data. Supports live monitoring and reverse engineering.
Mobile Application (not sure)	Remote interface for monitoring, sending, and analyzing CAN data via Bluetooth/WiFi. Supports real-time diagnostics and message injection.

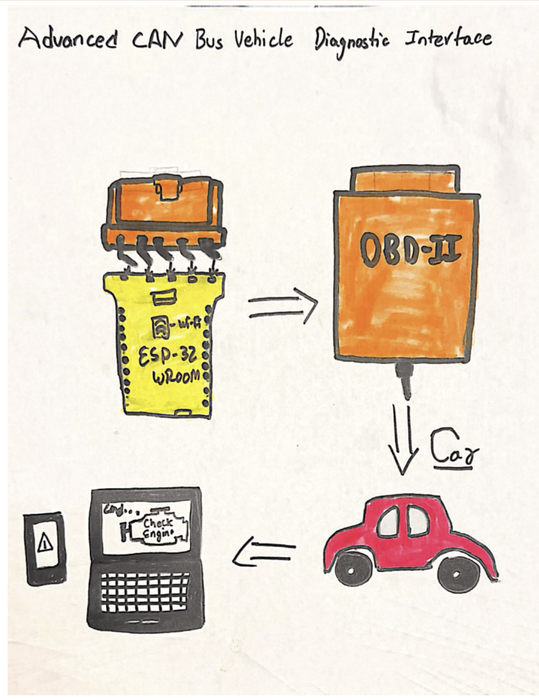
Capstone - TEAM

SK

SIMANT KHADKA
Bachelor of Science in Computer Science
NU'25
Norwich University
skhadka2@norwich.edu

Jeremy A. Hansen, PhD
Associate Professor & Esports Head Coach
Cybersecurity & Advanced Computing
Norwich University
+1 802-485-2221

Charles Snow, PhD
Associate Professor of Computer Science
Patrick Leahy School of Cybersecurity and Advanced Computing
Norwich University
csnow@norwich.edu



FUNCTIONAL CAPABILITIES & SYSTEM FEATURES

01 VEHICLE DIAGNOSTICS & OBD-II SUPPORT

- Read and clear Diagnostic Trouble Codes (DTCs).
- Monitor live vehicle parameters (RPM, speed, fuel level, etc.).
- Request custom OBD-II PIDs for additional data.

02 REVERSE ENGINEERING & SECURITY TESTING

- Identify unknown CAN messages and their functions.
- Track changes in data to map vehicle actions.
- Simulate cyber-attacks like spoofing and denial-of-service (DoS).

03 CAN BUS MONITORING & DATA ANALYSIS

- Capture and log real-time CAN bus messages.
- Filter and analyze specific CAN IDs.
- Display message timestamps and frequency.
- Highlight anomalous traffic patterns that may indicate tampering or faults.

04 CAN MESSAGE INJECTION & MANIPULATION

- Send custom CAN messages to interact with vehicle systems.
- Modify or replay captured messages for testing.
- Perform man-in-the-middle attacks to analyze security vulnerabilities.

05 WIRELESS COMMUNICATION & REMOTE ACCESS

- Enable Bluetooth or WiFi connectivity for mobile/PC control.
- Integrate with an app for remote diagnostics.
- Use a lightweight mobile dashboard to view diagnostics and send test commands remotely.