Hands-on Exploration and Learning Platform for Automotive Wheel Speed Sensor Attacks

Bhagawat Baanav Yedla Ravi and Sandip Ray
Department of ECE, University of Florida, Gainesville, FL 32611. USA.
b.yedlaravi@ufl.edu, sandip@ece.ufl.edu.

2025 IEEE International Conference on Consumer Electronics (ICCE) | 979-8-3315-2116-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICCE63647.2025.10929833

***Abstract*—Security is a critical concern in the rapidly advancing era of autonomous vehicles. However, the security challenges specific to automotive systems are often not well-understood beyond a small group of specialized experts. In this paper, we address this gap by introducing the first exploration platform designed to provide a hands-on understanding of security vulnerabilities related to wheel speed sensors. Unlike previous research that focused on identifying or creating new attacks, the platform aims to educate and engage users ranging from students to industry experts by allowing them to explore and understand the implications of security breaches in automotive systems. We detail the unique challenges and considerations in the design of such a novel exploration platform and illustrate its use in examining various attack vectors targeting wheel speed sensors. This work serves as both an educational tool and a practical framework for enhancing awareness and knowledge in the field of automotive security.**

I. Introduction

Automotive systems have undergone rapid transformation, especially with the rise of autonomous and driver assistance features. Sensors and actuators play a crucial role in this transition. However, this increased reliance on digital systems has also led to a higher risk of cyber-attack exposure. For example, a compromised wheel speed sensor can cause serious damage to the transportation infrastructure through catastrophic accidents, and research has shown that it can be alarmingly easy for a malicious actor to compromise a vehicle's systems such as a wheel speed sensor. Given these risks, there is a critical need for greater awareness of cybersecurity in the automotive industry across a wide range of stakeholders, including designers, manufacturers, users, policymakers, and the cybersecurity community. While several papers [1] [2] [3] [4] have addressed attacks on automotive systems, the knowledge and understanding of these issues among the broader community remain limited.

Our goal is to address this gap by developing an exploration platform for wheel speed sensor attacks that equips users with limited expertise in automotive cybersecurity with the knowledge and experience needed to navigate this complex field. The primary objective of the platform is to provide an interactive environment that allows users to intuitively and iteratively explore vulnerabilities firsthand. Our platform HAWA (**H**ands on Exploration Platform for **A**utomotive **W**heel Speed Sensor **A**ttacks), focuses on a critical component of driver assistance systems: wheel speed sensors. This platform is part of the HELP (**H**ands-on **E**xploration and **L**earning **P**latforms) series which aims to advance research, foster innovation, and enhance education in the fields of automotive safety and security. The HAWA setup is unique in its interactivity, providing a hands-on educational and exploration platform that teaches cybersecurity compromises in wheel speed sensors. It also features scalability and reconfigurability, allowing expert users to explore and develop detection and mitigation strategies. To our knowledge, there exists no educational and exploration platform on wheel speed sensor attacks that interactively guides and lets the user explore security vulnerabilities and see their effects. These sensors supply essential data that vehicles use to make decisions related to driving dynamics. Any errors or misleading information from these sensors can cause the vehicle to become unstable and lead to accidents. We also discuss some of the unique requirements involved in developing this platform.

The remainder of the paper is structured as follows: Section II discusses the overview of wheel speed sensors, relevant attacks, and exploration platforms. Section III elaborates the HAWA approach, its design, and constituent entities. Section IV explains the use cases for the HAWA platform. Section V concludes the paper. An interested user is encouraged to watch a video [5] that features the working of the HAWA platform.

II. Background and Related Work

A. Overview of wheel speed sensors and related attacks

Wheel speed sensors have been integral components in automobiles for a long time. Essentially, these sensors gather wheel speed data, which is critical for various systems, particularly those related to safety. Beyond the speedometer, the most common safety function is noticeable in most cars, such as doors locking automatically once the vehicle reaches a particular speed. Wheel speed sensors play a crucial role in systems like Anti-lock Braking Systems (ABS), Traction Control Systems (TCS), and Electronic Stability Programs (ESP). These sensors are typically connected to the ABS ECU, which measures abrupt changes in speed and correlates them with traction loss. There are two types of wheel speed sensors used in vehicles: passive and active. Passive sensors consist of a solenoid wound around a permanent magnet that generates an analog signal (sine wave) when a toothed tone ring or a reluctor rotates by the sensor. On the other hand, active sensors are more accurate at lower speeds by providing a

*This research has been supported in part by the National Science Foundation under Grant No. SATC-2221900 and REU-2150136.

DC square wave. Active sensors also use a magnetic ring and a toothed ring made of ferromagnetic material placed on the wheel shaft. Both types of sensors function using a magnetic field. Adversarial actions can involve generating a stronger magnetic field that disrupts or interacts with the wheel speed sensor, causing it to read erroneous values. The HAWA platform allows users to perform similar actions to create scenarios for spoofing and jamming attacks. A jamming attack, a form of denial-of-service attack, renders the sensor unable to read values. Meanwhile, a spoofing attack leads to the sensor generating false readings.

B. Related research on wheel speed sensors and exploration platforms

Numerous research papers have explored wheel speed sensors, focusing on topics ranging from performance improvement to energy harvesting. Several of these studies have also exploited the physics behind these functions to attack the sensors. For example, Pollny *et al.* [1] used an audio amplifier and Helmholtz coil to emit electromagnetic fields at varying frequencies, analyzing their effect on electronic stability control using a Hardware-in-the-Loop (HiL) setup. Shoukry *et al.* [2] developed a prototype ABS spoofer to enable such attacks and discussed the potential consequences of these vulnerabilities. Other papers have described methods to detect and mitigate wheel speed sensor attacks. For instance, Shin *et al.* [3] the authors used deep neural networks (DNNs) to detect attacks on wheel speed sensors. In another study, Kang *et al.* [4] introduced a technique to predict vehicle states based on previous data and developed a mitigation strategy where attack-induced changes are subtracted from the measured vehicle states. These papers provide methodologies to perform attacks on wheel speed sensors and detect and mitigate them. However, they do not teach individuals new to vehicular security. Additionally, Wang *et al.* [6] presented a benchtop teaching platform to help students understand ABS functionality, though it was not designed with cybersecurity exploration and education in mind. Automotive cybersecurity platforms were developed for ultrasonic sensors [7] by Ravi *et al.* and [8] by Carter *et al.*, V2X systems by Prakash *et al.* [9] and computer vision modules [10] by Chamarthi *et al.* but there has not been a learning platform targeting wheel speed sensor attacks. Kabir *et al.* [11] presented ViSE, a digital twin platform that facilitates the exploration of automotive functional safety and cyber security.

III. HAWA Approach

A. Requirements for exploration and learning platform

Hands-on experimentation is a primary requirement for any exploration platform to provide a learning experience. Instead of merely watching an expert demonstrate an attack, the platform should allow users to interact and engage actively. Learning from mistakes and progressing towards a successful attack requires feedback. An exploration or learning platform must be capable of comparing the deviation from ground truth using data points thus providing feedback to the user.

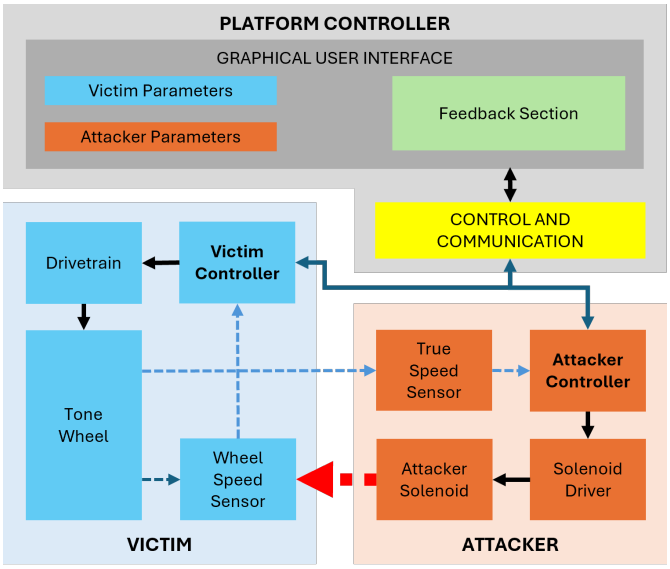

Figure 1. HAWA system architecture showing three entities: the victim, the attacker and the platform controller

This lets the user gauge the success of his/her trial. This lets the user continue to use the platform until novel attacks are explored. To provide a realistic experience similar to working on an actual vehicle, the platform must accurately replicate the behavior of a real vehicle. On the other hand, working on a real vehicle might solve the issue of realism, it is not economically feasible hence platform should be within reach of users, especially students. The platform should be configurable, extensible, and interoperable with other platforms to perform advanced attacks. To support beginners in becoming experts, the platform should offer foundational guidance and instruction. It begins by educating users on how the system functions and how it integrates with real vehicles to achieve its intended purpose. After establishing this fundamental knowledge, users advance to an intermediate level where they experiment with attack parameters and observe the system's behavior. Finally, at the expert level, users intuitively adjust various parameters to control outcomes, leveraging the insights gained from the intermediate stage.

B. HAWA *Design and Implementation*

The platform consists of three entities: (1) *The Victim* entity that carries components subjected to attack. (2) *The Attacker* the entity that performs malicious actions and (3) *The Platform Controller* the entity that transmits control commands and receives feedback data from both the victim and the attacker. Fig.1 shows the high-level architecture of HAWA. The victim consists of a microcontroller, drivetrain subassembly, tonewheel, and a Hall effect-based wheel speed sensor that provides control feedback to the victim controller. Fig.2 and 3 show the HAWA's drivetrain which is derived from one of a typical vehicle. It includes a motor that acts as a source of rotational motion, a gearbox consisting of a reduction gearset an electronic speed controller (ESC) that

controls the motor, and an output shaft that drives the wheels. As mentioned for the requirements of an exploration platform in III-A, the HAWA platform allows users to understand the workings of a drivetrain unit, including its modes, settings, and outputs. It also progressively introduces basic attacks with guided assistance. A set of magnets is placed on the perimeter of the output shaft that acts as a tone wheel to the wheel speed sensor. The setup lets the user choose between the type of tone wheel (magnetic ring or toothed ring) and the type of sensor (active or passive). For a tone wheel that has 10 magnets, the passage of these 10 magnets at the wheel speed sensor represents one rotation. If these 10 triggers happen in 1 second, the wheel speed sensor reads this as 1 rotation per second or 60 rotations per minute (60 rpm). The attacker consists of an electromagnet/solenoid that is placed close to the wheel speed sensor and a true speed sensor. This solenoid is connected to a driver and is controlled by the attacker's microcontroller. The HAWA victim satisfies the requirements of an exploration platform highlighted in III-A by including a true speed sensor that compares the ground truth information to provide feedback. The true speed sensor (Hall effect-based) on the attacker module provides ground truth information about the actual real-time wheel speed. Unlike the wheel speed sensor on the victim assembly, this sensor is placed in such a way that it is unaffected by the attacker when it is in operation. The true speed sensor values are essential to provide a comparison with respect to the wheel speed sensor values during an attack. The attacker generates electromagnetic fields strong enough to overshadow the field from the tone wheel. The victim microcontroller and the attacker microcontroller are connected to the platform controller such as a Raspberry Pi or a PC. The platform controller is a hardware/software interface consisting of a GUI that lets the user control various parameters of the victim and the attacker entities discussed in the next subsection.

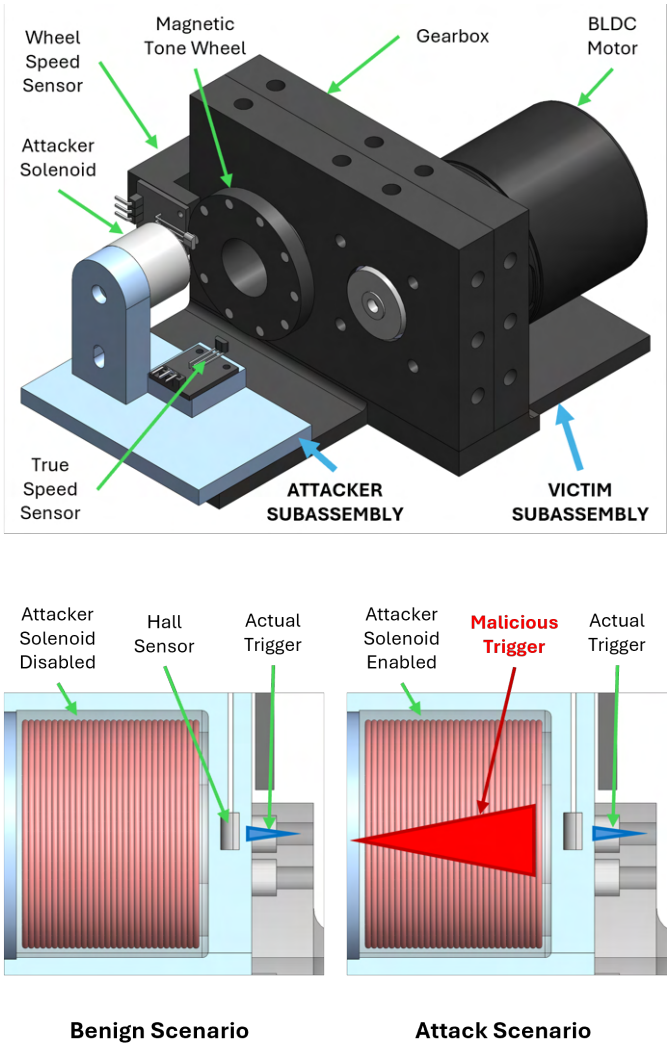

Figure 2. *(Above)* HAWA Assembly and its components. *(Below)* Representation of benign and attack scenarios: Malicious trigger from the attacker solenoid produces false speed readings

C. GUI Parameters

The HAWA platform's GUI is shown in Fig.4. It consists of three frames: the victim frame, the attacker frame, and a feedback section frame. The victim control section includes parameters namely: *Wheel Speed* to control the speed of the wheel. Intermediate and Advanced modes contain parameters such as *Direction* to choose a rotational direction, *Acceleration*, *Deceleration*, *Accelerator Position* if the user decides to control a range of speeds from 0 to 100% *Brake* and *Speed Hold* button to hold the rotational speed similar to the cruise control setting. The Beginner Mode of attacker section consists of a *Induced Speed* that controls the trigger of the electromagnetic field to mimic the rotation of the tone wheel. The frequency of triggers can falsify speed data sensed by the wheel speed sensor. A high trigger rate can saturate the magnetic field leading to blinding the sensor. The attack and benign scenarios are shown in Fig.2. The feedback section provides the current state of the victim and the attacker. The difference in the desired speed value and the true speed is called *Deviation*, which provides the strength, success, and the kind of attack. For example, if the deviation value is zero it corresponds to a benign scenario. A positive value means the wheel leads the motor. This corresponds to the vehicle falsely interpreting that the loss of traction occurred during acceleration. The GUI is modular, which means it allows the user to add and remove parameters for Intermediate and Expert Modes. An advanced user can program the GUI to display a graph apart from numerical values to obtain the trend of the dynamic behavior of the drivetrain unit. In the upcoming sections, we explain the GUI for specific case studies shown in Figs.5, 6, 7 and 8. If a sensor is attacked, it reads false wheel speed values and inject them in the feedback loop of the control system causing the motor to not run at its intended speed. The user is given the provision to play with several such combinations of parameters to explore a variety of use cases. In the backend, these parameters are processed as shown below to provide feedback to the user.

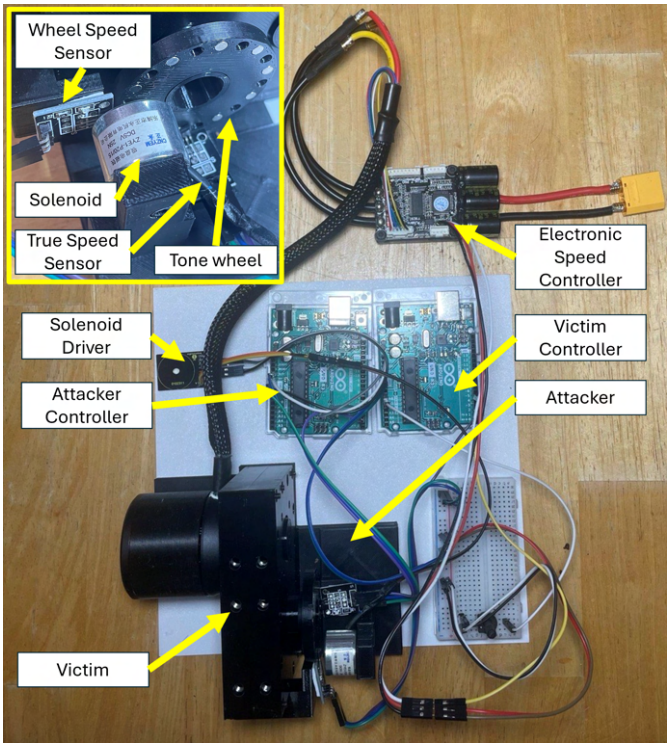

Figure 3. HAWA apparatus components. *(Above right)* Close-up view of the victim components: Wheel Speed Sensor (WSS) and magnetic tonewheel, and the Attacker Components: Attacker Solenoid and True Speed Sensor.

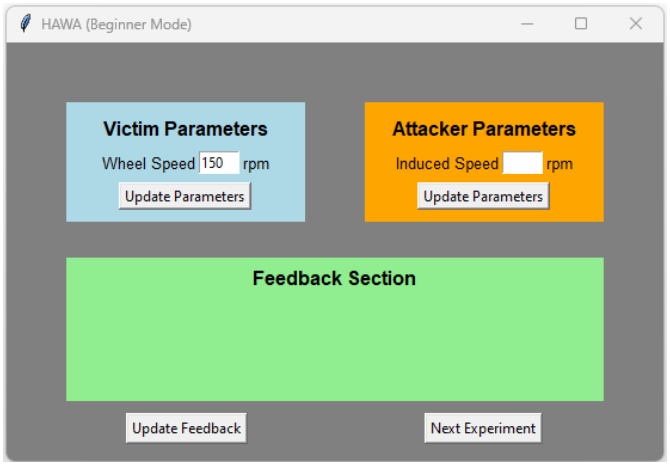

Figure 4. HAWA GUI with its parameters.

- N_{ds} = Desired Speed (GUI Wheel Speed Input)
- N_{ts} = True Speed (Ground Truth Value)
- $Deviation = N_{ds} - N_{ts}$
- If Deviation = 0 in benign scenario
- If Deviation > 0 Attack scenario with a leading wheel
- If Deviation < 0 Attack scenario with a lagging wheel

D. HAWA *Workflow*

The HAWA platform features three modes: The beginner mode for a user new to automotive systems and cybersecurity, an intermediate mode for a user with familiarity with automotive systems but limited knowledge and experience in cybersecurity, and the expert mode for users who possess a fair amount of exposure in both automotive systems and cybersecurity.

1) Beginner Mode: In the beginner mode, the user starts with a benign scenario. The user gets to change the wheel speed values to understand the basic functioning of the drivetrain. In the benign scenario, an induced speed value of "0" rpm corresponds to a benign scenario. The user proceeds to the next experiment by clicking *Next Experiment* button. Now the user is given a wheel speed value by the GUI and the user gets to change Attacker Induced Speed. The user tries different induced speed values and observes the behavior. The feedback section provides enough insight in terms of deviation from benign behavior. Once the user is ready for the next experiment, the next experiment button is clicked. A new wheel speed value is given to the user and is allowed to repeat trials and observe the behavior with the new wheel speed values. The same process is expected to repeat with more wheel speeds automatically given by the GUI. After this, the user is allowed to change the wheel speed values and the induced speed value at will. This allows the user to try out wheel speeds other than the speeds previously given automatically by the GUI. An interested user tries to figure out the trend in the behavior of the wheel speeds and corresponding attacker parameters.

2) Intermediate Mode: In the intermediate mode, the user is provided with additional parameters both the victim Section and the Attacker Section. Under the victim section, additional parameters such as *Direction* of rotation, *Acceleration*, *Deceleration*, *Accelerator position*, and *Brake Position* values are provided. The new attacker parameters section consists of *Chirp Rise Rate* and *Chirp Fall Rate* to mimic acceleration and deceleration behavior using the attacker solenoid.

3) Expert Mode: In the Advanced or Expert Mode, the user is given a speed cycle by the GUI. This consists of a dynamic plot of varying speeds, acceleration, and deceleration values with respect to time. The user is expected to create a corresponding attacker cycle to subvert the drivetrain's behavior in a controlled manner.

IV. Case Study

A. Benign Scenario

This benign scenario is illustrated in Fig.5, which displays the GUI inputs used. The HAWA platform initially enables users to experiment with various wheel speed values to understand how specific parameters influence the drivetrain control. The induced speed is set to 0 rpm, which corresponds to a non-attack state because the attacker solenoid is deactivated. To establish a baseline understanding of the wheel speed sensor's functionality, we consider a benign scenario where the wheel speed is maintained at 120 rpm and the induced speed remains at 0 rpm, keeping the solenoid inactive. Under these conditions, the user observes that the speed consistently stays at 120 rpm, confirming that no attack is occurring. Users are encouraged to try different wheel speed values and observe

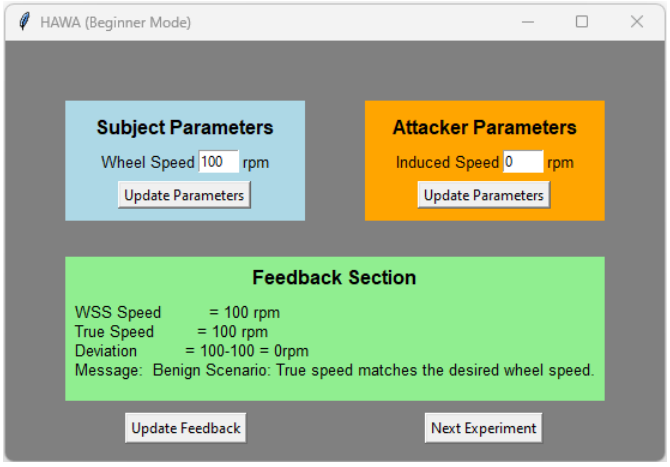

Figure 5. HAWA GUI showing a benign scenario experiment

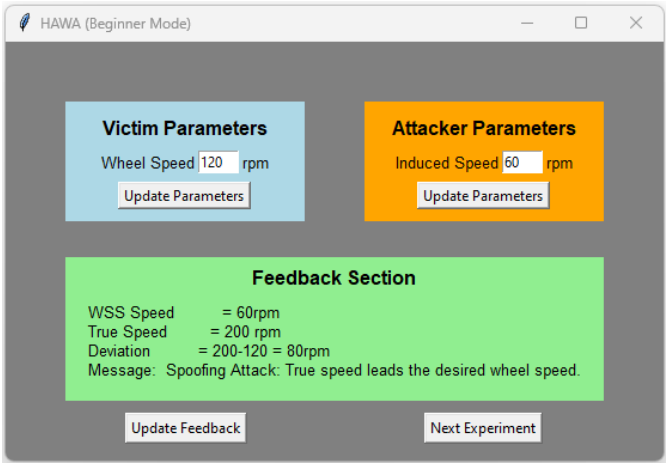

Figure 6. HAWA GUI showing a spoofing attack scenario experiment with wheel leading the desired speed.

the corresponding outputs in the feedback section; notably, the deviation remains zero as long as the attacker is not engaged.

B. Spoofing Attack Scenario

This spoofing attack scenario is illustrated in Figs.6 and 7. Spoofing attacks are a form of sensor manipulation attacks that lead to false perception of the ground truth data. In this situation, the speed data is misread by the wheel speed sensor. An attacker can cause the sensor to read either random data are controlled constant false data. We can observe two cases within this form of attack using the HAWA platform. The user can induce a speed lower than the desired speed which leads to a rise in actual speed as shown in Fig.6 or can induce a speed higher than the desired speed which slows down the vehicle shown in Fig.7. As an example let us create an attack scenario. The wheel speed value under victim parameters is kept at 120 rpm. The induced speed under the attacker parameters is set to 60 rpm. The attacker solenoid triggers at a frequency that mimics the tone wheels as if it were rotating at 60 rpm. The wheel speed sensor misreads the reading to be 60 rpm causing the victim microcontroller to raise the speed to reach 120 rpm. However, this rise in throttle causes the speed to overshoot the desired speed causing the wheel to rotate at 200rpm. This corresponds to a spoofing attack with the true wheel speed leading to the desired wheel speed. Similarly, if the induced speed is set to 240 rpm the opposite happens. The wheel speed sensor perceives an overshoot in the rotational speed of the wheel and decreases the throttle reducing the speed to 90rpm. This negative deviation from the desired speed indicates a spoofing attack with the true speed lagging the desired wheel speed of 120 rpm.

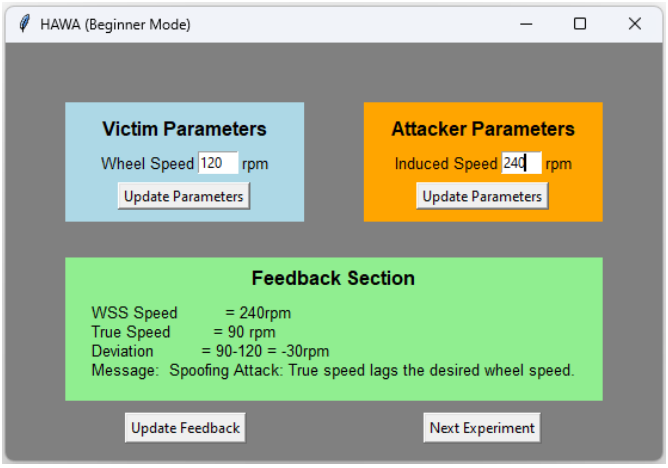

Figure 7. HAWA GUI showing a spoofing attack scenario experiment with wheel lagging the desired speed.

C. Jamming Attack Scenario

Fig.8 illustrates the GUI for the jamming attack scenario. A jamming attack is a form of denial of service attack that essentially blinds the sensor. Blinding a sensor on the HAWA platform leads to the sensor reading 0rpm which means the sensor is incapable of sensing any rotation. This can be

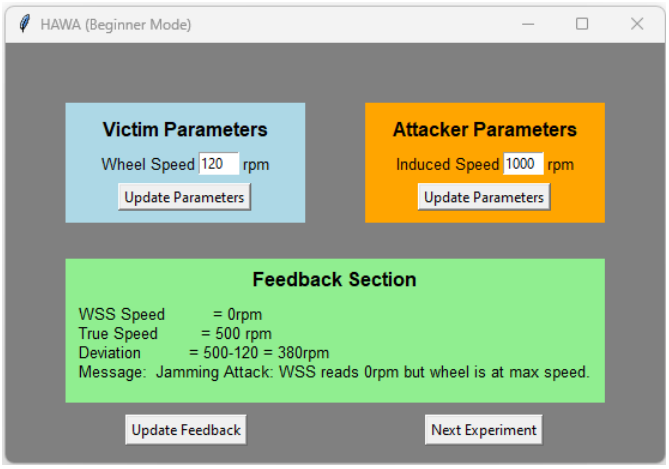

Figure 8. HAWA GUI showing a jamming attack scenario in which the sensor reads zero causing the microcontroller to increase the throttle to reach the desired speed.

achieved by exciting the solenoid at a very high frequency. This means the user has to input a very high induced speed value say 1000 rpm. Now the attacker behaves as though the solenoid is always on. With this attacker's magnetic field overshadowing the tone wheel magnetic field, the sensor senses no change in the overall magnetic field causing the sensor to to believe that the wheel is not rotating. This raises the throttle causing the wheel to run at the maximum speed the platform is safely limited to, 500 rpm.

The users can play with chirp rise and chirp fall values to mimic acceleration and deceleration scenarios. These particularly are useful in exploring ABS and TCS attacks to represent wheel slip rates. The users are provided with several such combinations to explore a wide variety of attacks.

V. Conclusion

Wheel speed sensors are crucial for measuring a vehicle's wheel speed, enabling essential active safety features such as ABS, ESP, and TCS. Any compromise to these sensors can lead to a loss of vehicle control, posing significant safety risks. Apart from security attacks on wheel speed sensors causing serious disruption, they are easy to fabricate which makes it a critical aspect to address. Many aspects of automotive security especially the well speed sensors are not very well understood. In this paper, we have presented a platform HAWA designed to allow users to explore and understand the vulnerabilities associated with wheel speed sensors, including spoofing and jamming attacks through an exploration platform called HAWA. HAWA offers an interactive learning environment suitable for users ranging from students to industry professionals who possess knowledge of automotive system functionality but may lack expertise in cybersecurity. In future work, we plan to integrate this platform into a broader learning environment that includes a scaled-down version of a real vehicle. We plan to do further evaluation of the platform to explore interesting attacks and their outcomes. We also plan to extend this form of attack to similar systems that function on the same physics.

References

[1] O. Pöllny, A. Held, and F. Kargl, "On the manipulation of wheel speed sensors and their impact in autonomous vehicles," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020, pp. 1530–1537.

[2] Y. Shoukry, P. Martin, P. Tabuada, and M. Srivastava, "Non-invasive spoofing attacks for anti-lock braking systems," in *Cryptographic Hardware and Embedded Systems-CHES 2013: 15th International Workshop, Santa Barbara, CA, USA, August 20-23, 2013. Proceedings 15*. Springer, 2013, pp. 55–72.

[3] J. Shin, Y. Baek, Y. Eun, and S. H. Son, "Intelligent sensor attack detection and identification for automotive cyber-physical systems," in *2017 IEEE Symposium series on computational intelligence (SSCI)*. IEEE, 2017, pp. 1–8.

[4] L. Kang and H. Shen, "Attack detection and mitigation for sensor and can bus attacks in vehicle anti-lock braking systems," in *2020 29th International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2020, pp. 1–9.

[5] B. B. Y. Ravi. (2024) Hands-on exploration and learning platform for automotive wheel speed sensor attacks. [Online]. Available: https://www.youtube.com/watch?v=iH8v8uam_4w

[6] B. Wang, "Design of teaching platform for abs wheel speed sensor," in *Journal of Physics: Conference Series*, vol. 2187, no. 1. IOP Publishing, 2022, p. 012004.

[7] B. B. Y. Ravi, M. R. Kabir, N. Mishra, S. Boddupalli, and S. Ray, "Autohal: An exploration platform for ranging sensor attacks on automotive systems," in *2022 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 2022, pp. 1–2.

[8] J. Carter, B. B. Yedla Ravi, M. R. Kabir, and S. Ray, "Poster: Efficient exploration of automotive ranging sensor attacks," in *Proceedings of the Twenty-Fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, ser. MobiHoc '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 589–591. [Online]. Available: https://doi.org/10.1145/3565287.3617986

[9] D. M. Prakash, B. B. Y. Ravi, S. Boddupalli, and S. Ray, "Vecaep: A hands-on exploration platform for vehicular communication attacks," in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*. IEEE, 2023, pp. 1–5.

[10] V. S. Gireesh Chamarthi, X. Chen, B. B. Yedla Ravi, and S. Ray, "Exploration of machine learning attacks in automotive systems using physical and mixed reality platforms," in *2023 IEEE International Conference on Consumer Electronics (ICCE)*, 2023, pp. 1–4.

[11] M. R. Kabir and S. Ray, "Vise: Digital twin exploration for automotive functional safety and cybersecurity," *Journal of Hardware and Systems Security*, pp. 1–12, 2024.