2024 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
MODULAR OPEN SYSTEMS APPROACH (MOSA) TECHNICAL SESSION
AUG. 13-15, 2024 - NOVI, MICHIGAN

A SIMULATION FRAMEWORK FOR EVALUATING THE CYBERSECURITY OF AUTONOMOUS GROUND VEHICLES

Christopher Goodin[1], Sara C. Fuller[1], Daniel W. Carruth[1], Kaneesha K. Moore[1], Benjamin T. Skinner[1], and Carl L. Mueller[2]

[1]Center for Advanced Vehicular Systems, Mississippi State University, Starkville, MS
[2]Circadence Corp., Boulder, CO

ABSTRACT

Autonomous ground vehicles (AGV) are comprised of a network of interconnected components including sensors, drive-by-wire actuators, and on-board computing. This on-vehicle network is often connected to a larger network which may include a ground station, other autonomous systems, or remote servers. While AGV share many features with other mobile networked devices like cell phones, the AGV computing and networking architecture may be vulnerable in ways that other systems are not, and the consequences of an attack may result in more severe physical consequences. In this paper, we present a systematic study of the network architecture of an AGV system, a cross-domain evaluation of possible attack vectors for AGV, and an implementation of a simulated cyberphysical test range that reveals the real-world consequences of cyberphysical attacks on AGV.

Citation: Authors, "A Simulation Framework for Evaluating the Cybersecurity of Autonomous Ground Vehicles," In *Proceedings of the 2024 Ground Vehicle Systems Engineering and Technology Symposium (GVSETS)*, NDIA, Novi, MI, Aug. 13-15, 2024 .

1. INTRODUCTION

Autonomous ground vehicles (AGV) as a Department of Defense weapons system exist simultaneously in the land and cyberspace domains. Protecting AGV against adversaries therefore requires threat assessments in both the physical and cyberphysical domains. However, AGV do not fit neatly into existing categorizations of cybersecurity threat analysis. For example, the MITRE ATT&CK framework [1] defines three main domains of cybersecurity: enterprise, mobile, and integrated control systems (ICS). While AGV could be considered complex instances of wireless mobile compute devices, they also share many properties of ICS, namely, a cyber exploitation of an AGV may have immediate and catastrophic physical

consequences. Finally, AGV share some features of enterprise systems because they often make extensive use of third-party software that may be deployed across a range of distributed systems.

The integration of many different types hardware, software, and interfaces in AGV makes them vulnerable in ways that many other cyber systems are not. For example, a recent analysis of the cybersecurity of Kia vehicles found that a vulnerability in the integrated entertainment system allowed attackers to access the CAN bus and alter the physical behavior of the vehicle [2]. Furthermore, a recent review of the cybersecurity of AGV found that their reliance on deep learning algorithms makes them especially vulnerable to artificial intelligence (AI) based attacks [3]. Predicting and protecting against these novel threats requires a systematic approach to threat assessment and modeling. In this work, we present a systematic approach to cyber threat environment modeling and simulation that combines 1) a system decomposition of the AGV cyberphysical system, 2) a dedicated threat assessment for AGV, and 3) an AGV-focused simulation environment. The integration of these analyses and capabilities will allow vulnerability assessment for AGV to keep pace with the rapid technological advances in AGV development. More detail on each of these areas is presented in the sections below.

2. AGV SYSTEM DECOMPOSITION AND ATTACK VECTORS

Most AGV developed by and for the US Department of Defense (DoD) are implemented as a package of networking, hardware, and software that is added to an existing vehicle platform that may not have been originally designed for autonomous operation [4]. AGV are therefore comprised of a complex cyber-physical system that can be broken down into many different external and internal subsystems. External subsystems are any components integrated into the vehicle by the AGV development team team, while internal subsystems are subsystems that are integrated into the vehicle by the original manufacturer. In more commercial language, *internal* subsystems come with the vehicle "out-of-the-box", while *external* subsystems are aftermarket additions and alterations made by the development team. Therefore, a steering actuator added by the development team to enable drive-by-wire for a mechanically controlled vehicle would be considered external, whereas for an electronic vehicle with a built-in drive-by-wire kit, the actuator would be considered internal.

2.1. External Components

Figure 1 shows an AGV's external components and their connections. External components addressed in the following system decomposition include perception sensors, drive-by-wire systems, onboard computing, positioning sensors, and auxiliary systems.

Perception sensors like cameras and lidar are used for environmental perception. Lidar sensors emit laser beams to measure distances and create detailed 3D maps of the surroundings. This information is essential for accurate navigation, helping the AGV to precisely identify its position within its environment. Cameras capture visual information that is crucial for lane detection, traffic sign recognition, and overall path following. The imagery helps the AGV understand the road layout and make informed navigation decisions. Lidar sensors are connected to the vehicles cyber-physical system through a wired ethernet connection to the vehicle's ethernet hub. The cameras are also connected to a GMSL interface for transmitting their data.

Drive-by-wire actuator subsystems are a critical component of AGV cyber-physical systems, responsible for translating electronic commands into physical actions. The drive-by-wire system controls various actuators that interface with the vehicle's mechanical components, allowing for precise control and maneuverability.

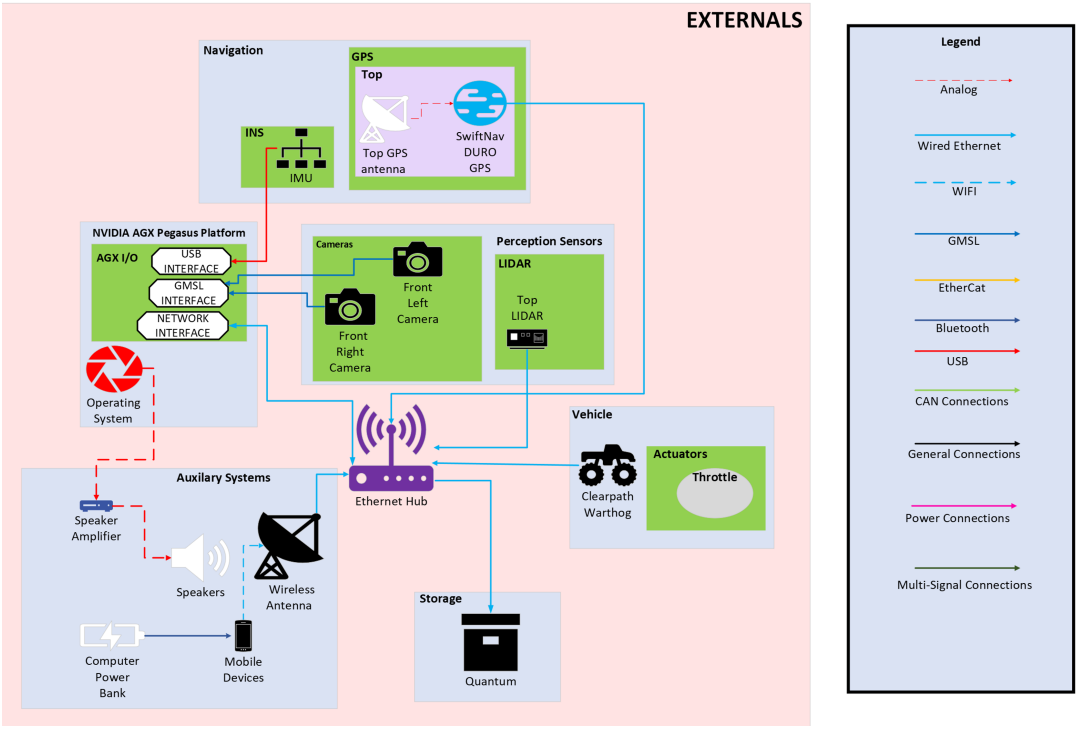

Figure 1: External components of a typical AGV and their connectivity

Onboard computing encapsulates the core computational components responsible for processing data, running algorithms, and making decisions. This includes central processing units (CPUs), graphics processing units (GPUs), and specialized computing platforms tailored for AGV applications. It manages data flow, task allocation, and communication between different subsystems, ensuring seamless coordination for efficient and safe vehicle operation. The AGV computing architecture is designed to handle real-time data processing, crucial for navigation, control, and swift responses to dynamic environments. The onboard computer's network interface has a wired connection to the vehicle's ethernet hub. The Gigabit Multimedia Serial Link (GMSL) interface is also here, where it receives information from the camera sensors. The navigation subsystem also sends information to the computer through the computer's USB interface. Lastly, any auxiliary systems like speaker amplifiers have an analog connection to the computer's operating system.

Navigation and positioning are critical for AGV operation. At the heart of the positioning system lies the Inertial Measurement Unit (IMU) and the Global Positioning System (GPS), working in tandem to provide real-time data that orchestrates the AGV path through the physical world. The IMU is a critical component of the AGV navigation subsystem, providing real-time information about the vehicle's acceleration and orientation. It consists of accelerometers, gyroscopes, and magnetometers that work together to capture and quantify the AGV movements in three-dimensional space. GPS is a satellite-based navigation system that allows the AGV to determine its precise location, velocity, and time. By receiving signals from multiple satellites, the AGV GPS receiver triangulates its position, providing accurate geospatial information.

Auxiliary systems introduce functions designed to amplify communication, power management, and user interaction. Including components such as the Speaker Amplifier, Power Bank, and features like mobile device connectivity, these subsystems not only enhances operational efficiency but also facilitates seamless integration with the AGV broader network. The Speaker Amplifier within the Auxiliary System enhances the AGV communication capabilities, facilitating audible alerts or messages. Its analog connection to the Operating System (OS) ensures seamless integration with the vehicle's control mechanisms. The Power Bank is what powers the electronics on the vehicle. Its accessibility by mobile devices allows for convenient monitoring of battery levels, ensuring optimal power management. Mobile Devices within the Auxiliary System serve as versatile interfaces for users or operators to interact with the AGV. These devices are equipped with functions such as battery monitoring and connectivity to the AGV's communication infrastructure. They connect to the vehicle through a Bluetooth and/or a Wi-Fi antenna that is connected to the ethernet hub.

2.2. Internal Components

Internal components are components that come built in to the vehicle by the manufacturer. Figure 2 shows an example network layout of AGV internal subsystems and their connectivity.

Emergency stops (e-stops) on an AGV are physical mechanisms or buttons designed for immediate and forceful intervention in case of an emergency.

The *Printed Circuit Board* (PCB) serves as a critical electronic platform that connects and integrates various components, such as microcontrollers, sensors, and communication modules. The PCB facilitates the flow of electrical signals, enabling seamless communication and coordination among different systems within the AGV.

The *Printed Circuit Board Microcontroller Unit* (PCB MCU) plays a pivotal role as the brain of the electronic system. The MCU on the PCB functions as the central processing unit, executing programmed instructions, managing data flow, and controlling various vehicle functions. This integrated platform facilitates communication between sensors, actuators, and other components.

The *Remote Control antenna* on an AGV serves as a pivotal component responsible for enabling the seamless transition between autonomous and manual control modes. This antenna establishes a reliable communication link between the AGV and the remote-control system, allowing operators to take command when necessary.

Lights on an AGV are visual signaling devices that convey real-time information about the vehicle's operational status in addition to providing illumination during dark or nighttime operation. These lights typically display key information such as power status, system health, or specific operational modes. The color, blinking pattern, or combination of lights provides operators and observers with quick and intuitive insights into the AGV's condition, allowing for prompt assessment and response.

Bilge Pumps are an essential component on amphibious AGV designed to remove accumulated water from the bilge, the lowest compartment of the vehicle. These pumps prevent water ingress, ensuring the AGV remains buoyant, stable, and operational.

The *Onboard Internal Computer* is already connected to the other internal, readily available components that come with the vehicle. This built-in system connects with the external Onboard Computer Platform to implement a seamless network between the internal and external components.

2.3. Analysis of Attack Vectors

As described in the previous section, detailing potential attack vectors, which can be exploitable vulnerabilities, depends on the cyberphysical

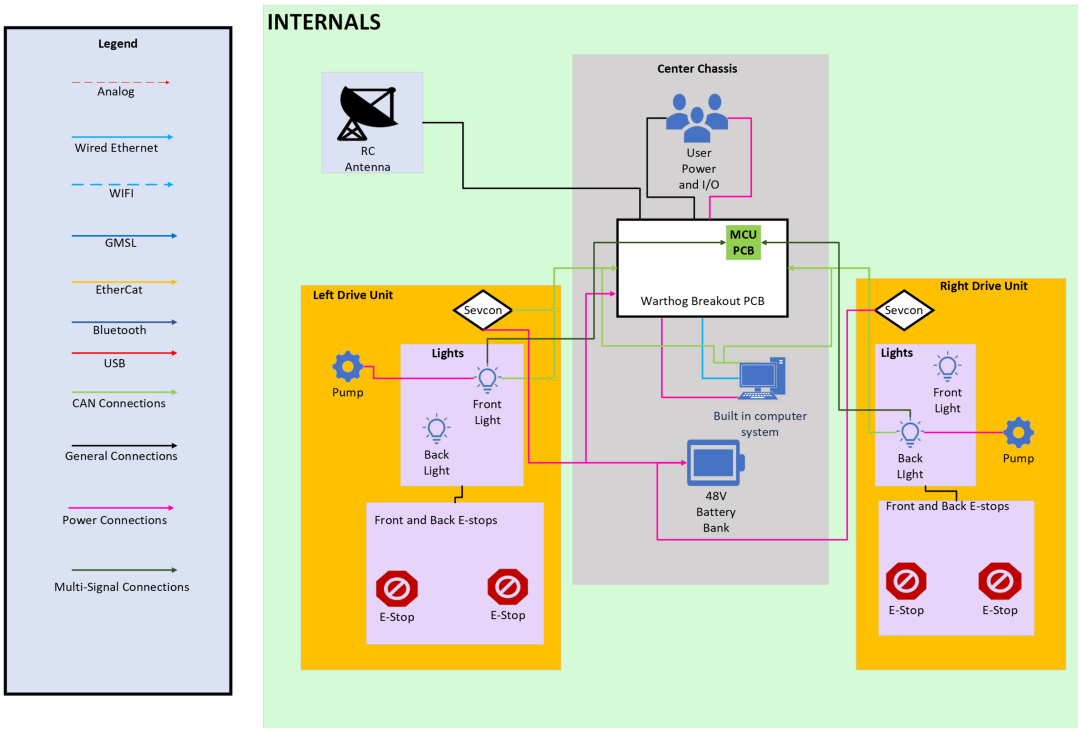

Figure 2: Internal components of a typical AGV and their connectivity

structure of an AGV. There are several standards and approaches (MITRE [1], NIST [5], IEEE [6]) that are a strong starting point for vulnerability analysis, but due to the nature of AGV, additional resources are required to account for the unique structure and deployment of autonomous vehicles. As AGV adoption becomes more widespread, an organized, framework is needed to ensure the prevention of vulnerabilities.

The process for developing such a procedural framework can be organized in a linear fashion - from decomposition to existing TTP analysis and adaptation, novel TTP identification, and standardization. Each component of the system must be viewed both independently and as a part of the overall vehicle. For example, the battery is a crucial component to any vehicle, but depending on the type of battery used and the features it offers (*e.g.* Bluetooth connectivity), it can introduce potential exploitable vectors. Examining the battery independently allows for the investigation of any potential weaknesses within the component while studying the component as it relates to the overall vehicle, such as communication protocols and physical connections, and can reveal avenues for cyber attack.

3. PHYSICAL SIMULATION FRAMEWORK

In the last decade, researchers at the Mississippi State University Center for Cyber Innovation (MSU-CCI) and the Circadence corporation have developed the Netmapper/Cyber Range Automation Framework (N/CRAF), a tool to "develop next-generation cyber learning and training software that can scan and map the military's complex computer network infrastructures." [7]. The N/CRAF tool allows the United States Department of Defense (DoD) to create a virtual cyber test-range for training the information technology personnel responsible for protecting the DoD networks. This

concept of a virtual cyber range must be extended to include a virtual *physical* range in order to test the combined cyberphysical system of an AGV. This requires a simulator capable of recreating the physical test environment of the AGV, the motion dynamics and control of the vehicle, and the data generated by the networked sensors on board the AGV. Additionally, the simulator must connect to the on-vehicle network in a realistic way to enable analysis of the cyberphysical system. The MSU Autonomous Vehicle Simulator (MAVS) [8] was chosen for this project because it meets these requirements for simulating the cyberphysical AGV system.

The MAVS is a C++ simulation software library that features realistic simulations of sensors in complex terrains [9] and weather [10] and of vehicle motion through complex terrain [11]. Because MAVS has been used to simulate physical test ranges in past experiments [8], the extension to cyber-range simulation was straightforward. Recent work has shown that MAVS can be used for direct comparison to physical tests by using the Robotic Operating System (ROS) [12] as middleware between the vehicle and sensor hardware (real or simulated) and the autonomous software [13]. Therefore we used ROS in our implementation of the cyberphysical test range as a network-enabled middleware between the simulated hardware and the larger network architecture being simulated by the N/CRAF tool.

The autonomy software used to drive the vehicle in the virtual test range is the NATURE (Navigating All Terrains Using Robotic Exploration) stack, an open-source, end-to-end stack for off-road navigation. NATURE has been used in recent work comparing real and virtual testing [14] and is therefore ideal for our purposes of creating a virtual test range. In addition, NATURE is compatible with ROS and ROS2, making it adaptable for our network simulation purposes. In the following subsections we discuss the simulated scenario and show how the NATURE/MAVS combination was implemented using the ROS architecture.

3.1. Simulated Scenario

We chose a mission scenario that would highlight the interconnected nature of many autonomous operations. In the selected scenario, an unmanned aerial vehicle (UAV) performing overwatch and reconnaissance transmits images to an operator at a ground station. The operator identifies a suspicious target (in this case, people moving on foot through mountainous terrain) and sends a set of mission waypoints from the ground station to a waiting AGV to go investigate the target. The AGV then navigates autonomously through the mountainous terrain using the NATURE autonomy software and transmits images back to the ground station. The scenario therefore requires bidirectional communication between the ground station and UAV and the ground station and UGV. The scenario features rugged off-road terrain. In addition, we can add virtual rain to the simulation, which affects both the vehicles' communication with the ground station and the functioning of the UGV autonomy system, via disruption to the lidar sensor.

Figure 3 shows some real-time output of the simulation as it runs during a demonstrated live test. The top right image is the live feed of the UAV-mounted camera, while the top left image is a diagnostic map for monitoring the simulation showing the trajectories of the UAV and UGV. The bottom left image shows a top-down view of the point cloud being generated from the UGV-mounted lidar sensor. The remaining two images show a third-person view of the UGV and UAV for debugging and playback purposes.

The communications link between the vehicles and the ground station simulates the possible degradation of the signal due to range losses and rain interference. The range-based communication degradation is simulated using the same empirical approach as our previous work investigating robotic swarming [15]. For a specified maximum radio

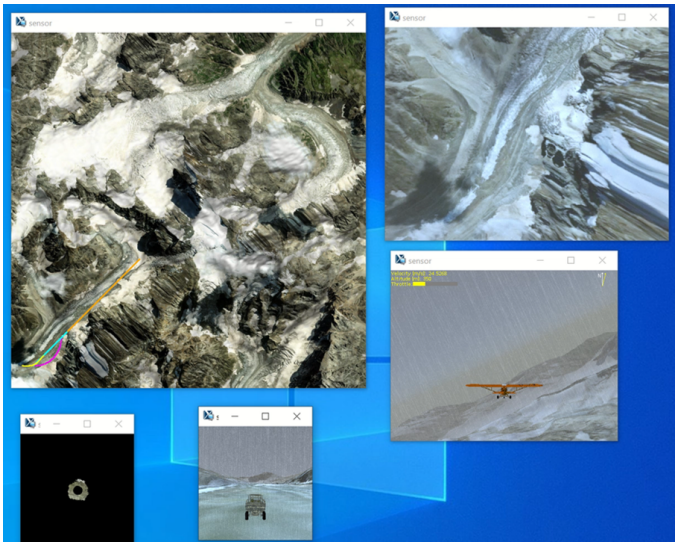

Figure 3: Real-time output from the virtual proving ground during the simulated scenario

range (r_n), we may expect a connection failure rate (f_r). When the range increases to $2r_n$, we can expect a more drastic failure rate (f_{r2}). For radios separated by a range r, we then model the connection probability $\sigma(r)$

$$\sigma(r) = 1 - \frac{1}{1 + e^{-\sigma_g r - \sigma_b}} \tag{1}$$

where the signal gain (σ_g) and signal bias (σ_b) are given by

$$\sigma_g = \frac{1}{r_{max}} \left[\ln\left(\frac{1}{f_r} - 1\right) - \ln\left(\frac{1}{f_{r2}} - 1\right) \right] \tag{2}$$

and

$$\sigma_b = -2\ln\left(\frac{1}{f_r} - 1\right) + \ln\left(\frac{1}{f_{r2}} - 1\right) \tag{3}$$

In the simulated scenarios presented here, the maximum range was modeled as $r_n = 2$km and the failure rate at that range was $f_r = 0.01$. The failure rate at $2r_n = 4$km was taken to be $f_{r2} = 0.075$.

We also simulated the influence of rain on the radio communication following the work of [16], who presented an equation for the influence of rain as a function of radio frequency and rain rate, and [17] who further studied this phenomenon for radio frequencies typically used in UAV - ground station communication, which typically operate at 2.4-5.8 GHz. Following the model presented by [16], we model the signal attenuation coefficient due to rain (μ_r) as

$$\mu_r = \alpha(\nu) R^{b(\nu)} \tag{4}$$

where R is the rain rate in centimeters/hour, ν is the radio frequency, and α and b are parameters that depend on the radio frequency according the following curves show in Figure 4.

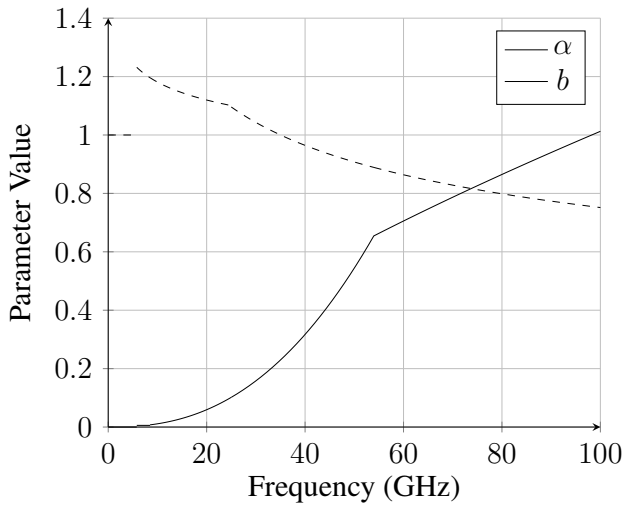

Figure 4: α and b parameters of the radio rain attenuation model as a function of frequency in GHz

The radio simulation is implemented as an intermediate network node between the simulation of the physical UAV/UGV and ground station. The simulation architecture is discussed more in the following section.

3.2. Simulation Architecture

The purpose of the simulation architecture is twofold. First, we would like to expose the real communication network of the physical

UGV to a virtual cyber range. Second, we need to enable modular simulation components to communicate with each other during run time. The ROS-2 middleware was chosen for this purpose because it allows networked nodes to publish and subscribe to messages on named topics. Topics that are "off-limits" for cybersecurity vulnerability assessment can be separated from those that represent real communications processes.

The resulting ROS-2 network is shown in Figure 5. In this figure, message topics are shown as rectangles and computational nodes are shown as ovals. Ovals circled in blue represent physical simulation of the communication between the ground station (circled in brown) and UGV (green) and UAV (orange). The message topics related to the functioning of the inter-vehicle UGV autonomy system are shown in the "nature" namespace at the bottom right of the figure, while the topics related to the vehicle-to-ground station simulation are shown in the "uav_sim" namespace at the top left. Additionally, a simulation monitor node can be seen in the top right of the figure. The purpose of this node is to monitor simulation execution and present status and debug information to the simulation operator that does not exist in the real system.

We were able to simulate the proposed reconnaissance scenario using the architecture shown in Figure 5 The human interaction with the ground station node was scripted to enable a totally closed loop simulation process. This initial simulation development, combined with the analysis of the UGV components and attack vulnerabilities, will allow DoD users to conduct cybersecurity vulnerability assessments in a virtual cyberphysical test range. The purpose of the simulation was to demonstrate the integration of all the components of the cyberphysical range into a single framework. Our ongoing efforts to complete this virtual test range are discussed in the next section.

4. SIMULATED ATTACK CHAIN

In this section we present an example scenario derived from analysis using the MITRE ATT&CK framework and demonstrate the physical result of this scenario in the simulation environment. Example attack techniques used in the scenario are shown in Table 1

The example scenario begins with initial access to the ground station using *stolen administrative credentials* (T0859) entered through *remote services* (T08886). Once access is achieved, the intruder enters a discovery phase using some combination of network sniffing (T0842), remote system discovery (T0846), and wireless sniffing (T0887).

Using information gathered in the discovery process, the intruder uses IP spoofing to disguise itself as the vehicle to the ground station and as the ground station to vehicle through *connection proxy* (T0884), *adversary in the middle* (T0830), using the ground station as a means to connect to vehicle.

Once the adversary is in the middle of the communication, they can alter messages sent to and from the ground station. They do not necessarily have to see what is in the messages but rather, all they need to do is alter it slightly to compromise the information. At this point, if the adversary wants to connect to the vehicle itself, it can scan ports since it knows all the information pertaining to connecting to the vehicle due to infiltrating the ground station. By connecting to the vehicle, they could establish persistence (*e.g.* a backdoor) using hardcoded credentials onto the vehicle, removing the need to connect through a proxy (the ground station). In the resulting simulation (Figure 3), the ground vehicle is routed *away* from the desired target instead of towards it, resulting in mission failure.

5. CONCLUSIONS AND FUTURE WORK

The goal of this work is to enable analysis of the physical consequences of the cybersecurity vulnerabilities of UGV. The role of UGV systems as a potential access point to larger networks will

Figure 5: The ROS-2 network used to simulate the scenario shown in Figure 3

Table 1: Example Attack Techniques for AGV

ID	**Technique**	**Impacts**
T0859	Valid Accounts	Integrity; allows for discovery evasion
T0886	Remote Services	Integrity; initial access to vehicle
T0842	Network Sniffing	Confidentiality
T0846	Remote System Discovery	Confidentiality
T0887	Wireless Sniffing	Confidentiality
T0884	Connection Proxy	Confidentiality, Integrity; enables lateral movement
T0830	Adversary-in-the-Middle	Confidentiality, Integrity; enables message manipulation
T0885	Commonly Used Ports	Confidentiality Integrity, enables attacker to evade detection

be explored through the N/CRAF tool. In the final months of this project we will integrate the simulation capabilities shown in Figures 3-5 into the N/CRAF environment to enable systematic study.

Future scenarios may be more complex and include connections to larger network services that provide information like real time GPS correction that represent potential access points for the AGV software. Additionally, the potential to disrupt AGV using novel techniques such as environment modification for the purpose of corrupting sensor signatures are create failures in the deep-learning algorithms that control the system will be explored.

ACKNOWLEDGEMENTS

DISTRIBUTION A. Approved for public release; distribution unlimited. OPSEC #: TSMO-24-0001

References

[1] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "Mitre att&ck: Design and philosophy," in *Technical report*. The MITRE Corporation, 2018.

[2] G. Costantino and I. Matteucci, "Reversing kia motors head unit to discover and exploit software vulnerabilities," *Journal of Computer Virology and Hacking Techniques*, vol. 19, no. 1, pp. 33–49, 2023.

[3] M. Girdhar, J. Hong, and J. Moore, "Cybersecurity of autonomous vehicles: A systematic literature review of adversarial attacks and defense models," *IEEE Open Journal of Vehicular Technology*, 2023.

[4] B. Theisen, E. Schoenherr, D. Simon, and T. Schulteis, "Amas jctd operational demonstration lessons learned," in *2015 NDIA Ground Vehicle Systems Engineering and Technology Symposium*, 2015.

[5] A. Calder, *NIST Cybersecurity Framework: A pocket guide*. IT Governance Publishing Ltd, 2018.

[6] G. Adamson, "Ieee and the protection of cyberspace: Increasing ieee's cybersecurity role," *IEEE and the Protection of Cyberspace: Increasing IEEE's Cybersecurity Role*, pp. 1–49, 2023.

[7] D. Godwin, "Msu, circadence partner to create virtual cyber defense tool," Apr 2020. [Online]. Available: https://www.msstate.edu/newsroom/article/2020/04/msu-circadence-partner-create-virtual...

[8] C. Goodin, D. W. Carruth, L. Dabbiru, C. H. Hudson, L. D. Cagle, N. Scherrer, M. N. Moore, and P. Jayakumar, "Simulation-based testing of autonomous ground vehicles," in *Autonomous Systems: Sensors, Processing and Security for Ground, Air, Sea and Space Vehicles and Infrastructure 2022*, vol. 12115. SPIE, 2022, pp. 167–174.

[9] C. Goodin, M. Doude, C. R. Hudson, and D. W. Carruth, "Enabling off-road autonomous navigation-simulation of lidar in dense vegetation," *Electronics*, vol. 7, no. 9, p. 154, 2018.

[10] C. Goodin, D. Carruth, M. Doude, and C. Hudson, "Predicting the influence of rain on lidar in adas," *Electronics*, vol. 8, no. 1, p. 89, 2019.

[11] C. Hudson, C. Goodin, Z. Miller, W. Wheeler, and D. Carruth, "Mississippi state university autonomous vehicle simulation library," in *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium*, 2020, pp. 11–13.

[12] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng *et al.*,

"Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[13] D. W. Carruth, C. Goodin, L. Dabbiru, N. Scherrer, M. N. Moore, C. H. Hudson, L. D. Cagle, and P. Jayakumar, "Comparing real and simulated performance for an off-road autonomous ground vehicle in obstacle avoidance," *Journal of Field Robotics*, 2024.

[14] W. P. Johnson, *Assessment of Simulated and Real-World Autonomy Performance with Small-Scale Unmanned Ground Vehicles*. Mississippi State University, 2022.

[15] C. Goodin, L. Cagle, G. Henley, R. Fereday, J. Carrillo, P. Song, and D. McInnis, "Evaluating tradeoffs for swarm reconnaissance with autonomous ground vehicles," *Journal of Autonomous Vehicles and Systems*, vol. 2, no. 1, p. 011002, 2022.

[16] G. Hyde and P. L. Bargellini, "Satellite and space communications," in *Reference Data for Engineers*. Elsevier, 2002, pp. 27–1.

[17] U. Kesavan, A. Tharek, S. Rahim, and I. M. Rafiqul, "Propagation studies on rain for 5.8 ghz and 23 ghz point to point terrestrial link," in *2012 International Conference on Computer and Communication Engineering (ICCCE)*. IEEE, 2012, pp. 515–519.