Design Hands-on Lab Exercises for Cyber-physical Systems Security Education

Hongmei Chi
Department of Computer and Information Sciences, Florida A&M University
Tallahassee, FL 32307, USA
hongmei.chi@famu.edu

Jinwei Liu
Department of Computer and Information Sciences, Florida A&M University
Tallahassee, FL 32310, USA
jinwei.liu@famu.edu

Weifeng Xu
School of Criminal Justice, University of Baltimore
Baltimore, MD 21202, USA
wxu@ubalt.edu

Mingming Peng
College of Pharmacy, Florida A&M University
Tallahassee, FL 32310, USA
mingming1.peng@famu.edu

Jon deGoicoechea
Department of Computer and Information Sciences, Florida A&M University
Tallahassee, FL 32307, USA
jon.degoicoechea@famu.edu

Abstract—The integration of cyber-physical systems (CPS) has been extremely advantageous to society, it merges the attention of cybersecurity for vehicles as a timely concern as a matter of public and individual. The failure of any vehicle system could have a serious impact on vehicle control and cause undesired consequences. With the growing demand for security in CPS, there are few hands-on labs/modules available for training current students, future engineers, or IT professionals to understand cybersecurity in CPS. This study describes the execution of a free security testbed to replicate a vehicle's network system and the implementation of this testbed via hands-on lab designed to introduce concepts of vehicle control systems. The hands-on lab simulates insider threat scenarios where students had to use can-utils toolkits and SavvyCAN to send, modify, and capture the network packet and exploit the system vulnerability threats such as replay attacks and fuzzing attacks on the vehicle system. We conducted a case study with 21 university-level students, and all students completed the hands-on lab, pretest, posttest, and a satisfaction survey as part of a non-graded class assignment. The experimental results show that most students were not familiar with cyber-physical systems and vehicle control systems and never had the chance to do any hands-on lab in this field before. Furthermore, students reported that the hands-on lab helped them learn about CAN-bus and rated high scores for enjoyment. We discussed the design of an affordable tool to teach about vehicle control systems and proposed directions for future work.

Keywords—hands-on lab, active learning, cybersecurity, threats, cyber-physical systems, vehicle system

I. INTRODUCTION

Cyber-physical systems (CPS) are physical and engineered systems that integrate computation, control networking, and physical components. A cyber-physical system comprises the interaction with the physical world, and it is composed of networked agents such as sensors, actuators, control processing units, and communication [1]. The integration of CPS has benefited several technologies, such as enabling power grids to generate economically and zero carbon foot-print electricity, deliver sustainable energy to communities, and enhance grid resilience [2], and improving home automation.

With the emergence of CPS, this system has become a viable solution for governments and industry [3]. One of the most common applications of CPS, is the Cyber-physical vehicle systems (CPVSs). A typical example of a CPVS is an intelligent electric vehicle which involves the controller, physical vehicle, the driver and the cyber environment subsystems. A wide variety of CPS applications in CPVSs ranging from automobile to aircraft and marine craft promote studying and mitigating of potential cybersecurity threats to increase autonomy, reconfigurability, reliability, system capacity, safety, energy efficiency and robustness in such systems [4].

Even though emergence of in-vehicle networks reduced the difficulty of vehicle designing, vulnerability of in-vehicle networks has increased due to the ease of implementation of adversarial attacks on them. Liu et al [5] in their work were able to perform a set of attacks on the in-vehicle network such as Denial of Service (DoS), Distributed Denial-of-service (DDoS), Black-Hole, Replay, and Sybil Attacks . The work [6] has shown that the attacker can intercept the radio waves from a remote keyless system (i.e., key fob) and attempt to replace the signal using buffered replay attacks to unlock the vehicle doors. In 2015, researchers found that the Jeep Cherokee could be remotely 'hijacked' through its dashboard computer, which forced the Chrysler manufacturer to recall 1.4 million vehicles to update the vehicle's software [7].

Nonetheless the increased usage of vulnerability testing software for in-vehicle networks by the automotive industry to identify potential threats, more automotive vulnerabilities are emerging. One of the most recent published vulnerabilities related to connected cars is founded by The Common Vulnerabilities and Exposures (CVE) system and published by the National Vulnerability Database. The CVE-2019-13582 CVE-2019-13582 was registered due to an exposure found in the Marvell Wi-Fi Chip (88W8688) used on Tesla Model S/X vehicles manufactured before March

2016 which the attacker could modify the firmware and overflow the system with a denial of service or arbitrary code execution via malformed Wi-Fi packets [8].

Although there exist some training programs offered by institutions such as Idaho National Laboratory [9], InfoSec [10] and Washington Community College [11] but they either often focus on advanced programs for experts in the field and cover different CPS applications such as industry 4.0 [12] or they require basic knowledge on electrical theory, use and interpretation of automotive wiring diagrams, and use of electrical testing equipment. Thus, with the ever-increasing demand for CPS professionals and developers, there is a lack of material with the focus of cybersecurity to train students on vehicle control systems and deliver the future developer professionals [13].

In this study, we describe the execution of a free, open-source car-hacking virtual environment to replicate a vehicle's network system. This environment will be implemented in a series of hands-on labs to introduce concepts of network packets such as sending/receiving packets, encoding and/or decoding packets to the students with the focus of Controller Area Network (CANBus) vulnerabilities. In view of the fact that CAN-bus is the central system of a vehicle, and it is responsible for the communication between sensors and several electronic units (ECUs) this hands-on-lab series will engage students in a real-world scenario and prepare them for security issues in vehicle systems that may occur in real-life.

The rest of this paper is organized as follows. Section II reviews the related work. Section III presents the design of hands-on labs with real-world scenarios. Section IV describes the challenges faced in students' learning curve and principles for adopting active learning. Section V presents a background to CPS hands-on lab adoption via ICsim. Section VI demonstrates the feedback received from students. Section VII provides the conclusions and future works.

II. Related Work

There exist many hands-on lab instruction for mainstream concepts and vulnerabilities related to cybersecurity such as SEED labs [14]. But there are only a few hands-on labs developed regarding vulnerability threats to CPS. There are many published research papers related to cyber-physical systems applications and vehicle system vulnerabilities.

The demands for more efficient systems, better interaction with the physical world has turned CPS a part of our daily life. The need for new approaches and test beds to introduce CPS concepts into classrooms and research also becomes evident. The success to fill the gap between students' current knowledge and the market demands for security in vehicles relies on training and education in this field [15].

Konstantinou [15] created a security education course composed of six hands-on exercises to encourage CPS security students. The goal of their hands-on labs is to educate students in the unique aspects of CPS security, expose them to fundamental security primitives, and help them understand the challenges in designing and securing CPS but they do not address vehicle technology and the implementation of a workstation for future research.

AeroTech Digital Summit [16] offers a course "Introduction to Car Hacking with CANbus C1857" which introduces the modern automotive in-vehicle communication networks, the CAN communications protocol and the OBD-II interface threat models, hacking into the OBD-II diagnostics interface, ECU cracking, and vehicle network cyber penetration testing to students. However, it is preferred for the participants of their course to have a background or some experience with automotive electronics and vehicle systems and the course material is not open source meaning participants would have to pay for it.

Quarkslab [17] also offers a course "Training — Practical car hacking" which aims to introduce the basic theory about the CAN bus and its communication so that the attendees can try their hands on a CAN bus. The course provides the necessary CAN tools and performs attacks on simulated systems as well as real cars. It will teach the audience to reproduce attacks like breaking a security session, fuzzing an ECU, and spoofing messages. However, their provided course is not designed for undergraduate students and the audience of the course are security researchers, automotive manufacturers and suppliers and hackers interested in cars. In addition, This course is not open source and it requires the attendees to pay for the material.

The Car Hacker's Handbook (2016) provides a deep understanding of computer systems integrated in modern vehicles [18]. The authors developed the Instrument Cluster Simulator for researchers and educators to build necessary skills for security practice. This tool contains a dashboard with a speedometer, door lock indicators, and turn signal indicators turning practical learning realistic and enjoyable. Also, it includes a control panel used to interact with the simulated dashboard through the network where users can accelerate, control the door locks, and turn signals in our vehicle. The book serves as a guide for penetration testing, and was used by our work as the foundation of our hands-on lab simulations. In Section II A and B some of the vulnerabilities that have been detected in automotive in-vehicle communication networks will be discussed.

A. Common Vulnerabilities

CPS devices are frequently collecting data from a diverse set of human activities such as location information, driving habits, and biosensor data at unprecedented levels of granularity and electricity consumption of the ride by the driver [19]. Such data, if extracted, could contain several information related to the control systems, such as odometer information, cabin temperature, and battery status. Any of these gathered information, in attackers' hands, can easily bring threats to the safety and security of the control system.

In 2016, [20] a computer security services firm known for reporting high severity security vulnerabilities had released a metadata analysis of private vehicle security assessments. There are several methodologies for an attacker to harm the

CPS, such as targeting the entire vehicle or just a component in the system. Fig. 1 represents the vulnerability attack vectors published by [20] which is useful to compare against the threat model for a given element or system.

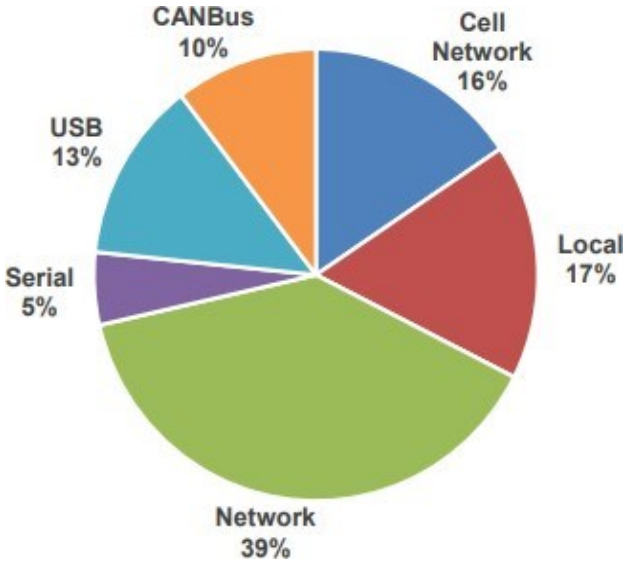

Fig. 1. Vulnerabilities Attack Location (IOActive, 2016).

According to this data 65% of the vulnerabilities are related to network traffic which appear in the following categories: 39% Network (general category that includes all network traffic, such as Ethernet or web), 10% CAN bus and 16% cellular network. Although realistic laboratory exercises are challenging to create or assess, simulated hands-on lab exercises which help students learn fundamental concepts through exploration are essential to prepare them to overcome a possible threat in vehicles' security systems.

B. CPS Vulnerabilities

CPS are the core of vehicle control systems and performs a vital function in critical infrastructure such as adaptive cruise control, vision sensors, battery and gas consumption, and anti-lock braking system. As CPS applications continue to increase, potential cyber-attacks that can cause system failures must be taken into high consideration. Attacks on CPS can happen at any point in the system architecture and the most common types of targets in vehicle control systems that are compromised under such attacks are sensors, delay time response, compromised controller, and physical process.

In particular, a control logic injection attack can happen on a compromised sensor. In this attack, the attacker targets the sensor data blocks to inject false signals and transfer tampered data to the programmable logic controllers (PLC) via control logic system and modify the PLC's system control flow to execute the logic located in data blocks [21]. An example of this attack is covered in the hands-on lab using the door lock sensors. Delay time response attacks are usually implemented by targeting the communication path between the Electronic Controller Units (ECU) and delaying the communication or blocking the signals sent in the network affecting the controller response Using a denial-of-service (DDos) or stale data attack. Compromised controller attacks can happen using reverse engineering attacks by taking control of the system and sending incorrect signals to the ECUs forcing undesired action. Physical process attacks are intended to cause physical damage to the system (blinding or confusing camera auto controls) to create an opportunity for a cyber-attack [22].

Even though many businesses are interested in CPS security training, its implementation is not given much attention in education programs due its high costs and the lack of test-beds capable of representing actual applications. In order to cover these challenges, this project proposes a workstation which includes open-source tools to adopt the parameters for a virtual training and a lab-based teaching environment.

III. Hands-on Lab Structure

The objective of this work was to provide students with a learning modality which helps them understand the key concepts of cybersecurity in CPVSs, demonstrate the potential vulnerabilities to them and also implement some of the attacks to understand how these threats could manipulate and jeopardize the CPVSs. In order to accomplish our goal, the lab environment was installed in a local machine to simulate a vehicle control (CAN bus), which could demonstrate the communication from the CPS and the Electronic Controller Units (ECU). The environment was set up using a simulator to allow easy transfer, installation and demonstration of the environment on all students personal computers with any processing capabilities to allow them to learn these materials during the COVID-19 pandemic with online learning. Our workstation was implemented using open-source tools to monitor malware injection or malicious code behavior in vehicle control. The network traffic monitoring using the workstation tools enables the students to perform the following tasks: observe the communication in the vehicle control in real-time, extract the data file and analyze the content for any abnormalities. Fig. 2 demonstrates the methodology and implementation of the work station along with the hands-on lab.

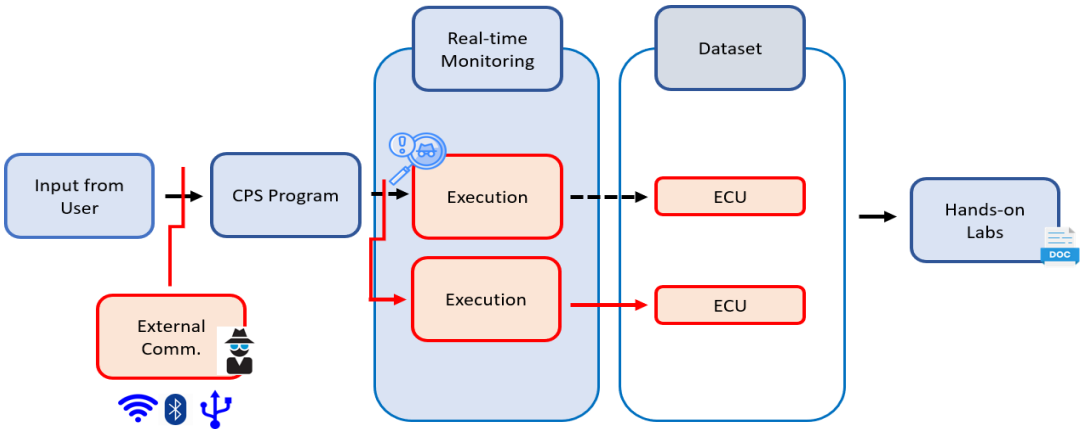

Fig. 2. Hands-on Lab Design

As shown in Fig. 2, an internet-connected device or vehicle can communicate to external agents or networks via several technologies' interfaces (e.g., Bluetooth, Wi-Fi, USB Flash drive, etc.). Although many vehicles are equipped with security systems focused on the prevention of unauthorized access, attackers can exploit the vehicle security intercepting the communication or take control from the cyber-physical systems to modify the ECU actions.

Our hands-on lab intended to provide students with practical experience and reflect their feedback on this experimental learning. The lab exercises introduced the concept of Controller Area Network(CAN bus) and network sniffing techniques which could be used by an attacker to apply threats on the most common targets of a vehicle control system. To be more precise, the focus of this work was to familiarize students to vulnerabilities related to the network and CAN packets concept as well as demonstrating some common attacks on the CAN bus system such as replay and fuzzing attack.

IV. Methodology

In the following section we will present methods that have been used to design and implement the lab series as well as the workstation and the performed case study.

A. Active Learning

The "learning process" shown in Fig. 3 begins with the initial exposure by doing hands-on labs, where the student witnesses the use of CPS security vulnerabilities in simulation of the CPS environment to explain or explore a problem. The next level of learning is creating a new hands-on lab by students given a specific problem, where one has become familiar enough with a CPS concept or model to alter the parameters of the CPS security. The stages of creating new hands-on labs mark the onset of research. Advanced research involves creating new solutions for specific problems such as use and employment of new models or threats. Interested students were offered with additional opportunities to learn to apply their skills to first perform an experiment and next solve a related real-world problem.

Fig. 3. Learning process from education to research

B. Environment Open Source Tools

We created a free, open-source car-hacking workstation using Kali Linux [23], Instrument Cluster Simulator [18], Savvy CAN [24], WireShark [25], and SocketCAN utilities [26]. Below is a brief description of the tools and their employment in the lab.

- Kali Linux which is a Debian-based Linux distribution geared towards various information security tasks, such as Penetration Testing, Security Research, Computer Forensics and Reverse Engineering was chosen as the main operating system of the environment because it provides many applications from password crackers to digital forensics software and is completely customizable.
- ICSim is an open source software which simulates a vehicle dashboard panel that includes a speedometer, door lock indicators, and turn signal indicators. Furthermore, the software contains a control panel responsible to interact with the simulated vehicle network. The control panel allows the user to control the virtual vehicle and perform a set of tasks such as pressing the accelerator and brake pedal, controlling the vehicle door system (lock/unlock) and turning signals. This software enabled us to familiarize the students with the CAN bus system without having to access a real vehicle.
- SavvyCAN is a cross platform QT based C++ program which is a CAN bus reverse engineering and capture tool. Using SavvyCAN we were able to

Scan captured traffic for data that looked coherent, Load and Save DBC files which are used to store definitions for how data is formatted on the bus.

- Wireshark is a free and open-source packet analyzer which is used for network troubleshooting, analysis, software and communications protocol development. We have used Wireshark to sniff packets from the CANbus.
- SocketCAN utilities is a Linux specific set of utilities that enables Linux to communicate with the CAN network on the vehicle. This set of utilities enabled us to display, filter and log CAN data to files, display the content differences and also replay the log files.

C. Design Hands-on Lab

Kali Linux was used to serve as a virtual machine to ensure the safety of student's local computers and implemented the Instrument Cluster Simulator to simulate a vehicle's controller area network; in other words, the simulator provides a real-time interactive vehicle dashboard. Also, we combined Savvy-CAN with can-utils to replicate threats such as replay and fuzzing attacks. We have monitored and logged data packets that run through the computer network using packet sniffers. The purpose of the hands-on lab was to offer an affordable tool (i.e., that does not require the use of an actual vehicle) to provide students with hands-on practice on the use of a Controller Area Network bus (CAN-bus) and familiarize students with CAN packets concepts (e.g., sending, receiving, encoding, and decoding packets).

All of the mentioned tools that are used for the hands-on lab series are open source, which enabled us to create a completely free environment for the students to use and also introduced them to the concept of open source, community collaboration and contribution of developers from different fields for problem solving. Along with the lab, there were detailed step by step instructions handed to the students to install each software. Students were first asked to set up the CAN interface (i.e., vehicle control and dashboard) and test traffic simulation commands on the CAN-bus. Students were then asked to use Wireshark and can-utils toolkit to sniff packets from the CAN-bus to understand the communication between the CAN-bus and the vehicle's sensors (e.g., signal lights, doors, and speedometer). Finally, students were asked to use the can-utils toolkit and the SavvyCAN to send, modify, and capture the network packet and exploit the system vulnerability threats such as replay attacks and fuzzing attacks on the vehicle system.

D. Case Study

The CAN Bus – ICSim hands on lab was created to practice CAN-Bus exploitation using the mentioned open-source tools to mimic a real case scenario where the attacker has access to the in-vehicle network. The students were familiarized with automotive security by understanding the communication between the ECUs, sniffing (capturing, inspecting, decoding, and interpreting) the CANbus network packets, analyzing for vulnerabilities, reversing engineering the CAN bus messages and perform a replay attack. Our work station requires students to simulate the vehicle system using the ICSim software. First, students were familiarized with the environment by playing the driver's role, controlling the vehicle using their keyboard. Second, students were introduced to the network packets and its related concepts by using the can-utils toolkit which allows them to view the packets sent by the control panel across the network to the dashboard simulator. This step allowed students to play an external agents role by reading and injecting packets of information into the CAN bus, and forcing the system to do undesired actions such as opening the vehicle door or accelerating the vehicle.

Once familiar with the software and concepts addressed in the hands-on lab, students were asked to simulate two potential threats to the CAN bus network, replay attack and fuzzing testing. To perform the replay attack simulation, students captured the packets in the CAN bus network, saved it into a Log file and re-injected the captured data into the system forcing the CAN bus to repeat the functionality. To perform the fuzzing testing simulation, students injected massive amounts of data into the network causing undesired effects to the simulated vehicle using the following method. One, injecting random packets into the CAN bus that corresponds to the signal lights forcing the vehicle to turn the lights on/off randomly. Two, injecting sequential packets into the CAN bus corresponding to the speedometer forcing the vehicle to accelerate until it hits the maximum speed and crashing it (returning the speedometer to 0 mph and starting the process over and over again).

V. Approach

Based on the principles of active learning, we have created a series of modules for students to learn CPS concepts using open-source tools. Students would learn by doing in a virtual training and a lab-based teaching environment. The hands-on lab environment creates an accessible and affordable workstation to engage students in a real-world scenario and increases their ability to understand CPS and the vulnerabilities related to it, specifically on vehicle control field as shown in Fig. 4. This work not only encouraged the students to learn about CPS, CPVS, their vulnerabilities, use of open source, and penetration testing techniques but also engaged them in combining all the gained knowledge to solve a real-world problem.

Active learning will help our students to overcome the learning curve in mastering concepts for detecting CPS vulnerabilities and cybersecurity. The framework of the CPS Security Education Labs shown in Fig. 4 consists of pre and post assessment surveys to identify students' current knowledge in the field and establish an achievable goal. Meanwhile, those feedbacks from students will help us to design more hands-on labs for training our students.

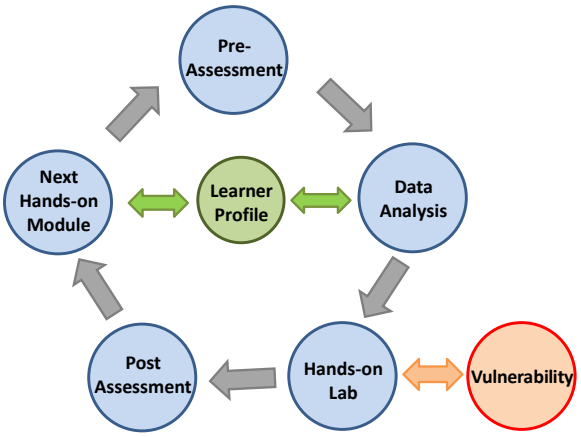

Fig. 4. CPS Security Education Labs (CPSlab)

A data analysis was applied using the preliminary results to evaluate the students' current knowledge and determine their profile to better assist them in solving the hands-on lab. The information gathered from each participant included their current academic level, their skill set on the concept of lab and their interest level in learning the provided material in the lab. As an example, students who are not familiar with virtual machines were offered with a set of additional instructions and introductions about it and if further help was required, they were assigned with a teaching assistant to help them install the environment.

After finishing the pre-survey, students were provided with the proper lab instruction, environment and required toolkit/materials to achieve the desired result using practical active learning.

Upon finishing the lab activity, students were requested to complete a post-assessment and satisfaction survey as part of a non-graded class assignment to evaluate their accomplishment and determine if their experience was satisfactory or if a closer follow up is needed before proceeding to the next hands-on module.

VI. Feedback from Students

Student feedback on the hands-on lab series was collected by a survey, which was conducted among 21 students of the computer science department at the Florida Agricultural and Mechanical University (12 undergraduate and 9 graduate students). The presented statistical population includes only those students who completed the pre-assessment, the hands-on lab series, the post-assessment and the satisfaction survey. Pre and post assessment surveys were comprised of 8 multiple-choice isomorphic questions.

As shown in Fig. 5, the results from the survey presents that only less than 50% of the students have had a prior knowledge of CPS and vehicle control systems (e.g. CAN-bus). The lack of affordable tools, which are designed for cybersecurity training in vehicle control systems might contribute to the low familiarization and practice in the area. Hence, we aim to design and develop more hands-on labs to further enhance their knowledge on the field and transfer it by offering them practical activities on CPS and CPVS.

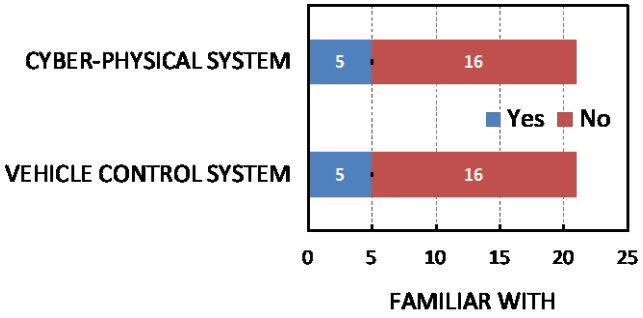

Fig. 5. Feedback from students prior to the hands-on lab

Fig. 6 represents the correlation between students response in the pre-assessment and post-assessment when asked about their level of prior knowledge on network traffic and common existing vulnerabilities to it. The results show that a majority of them had an improvement after completing the hands-on lab. We conducted item statistics analysis to identify the difficulty index of each item. The item statistics mean for pre-assessment was .774 (*Min* = .38, *Max* = .95), and for post-assessment was .72 (*Min* = .47, *Max* = .95). Further, In order to increase the forms' reliability, we revised the items with index difficulty above .86, and no item below .30 were found. The satisfaction survey was used for a deep evaluation of the collected data.

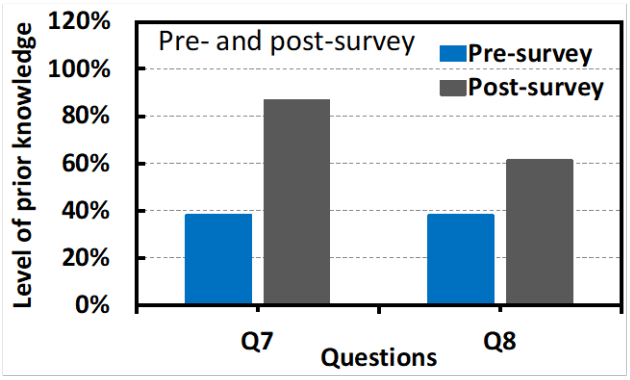

Fig. 6. Comparison between pre- and post-survey

Students were asked to evaluate their experience solving the hands-on lab on the satisfaction survey. The satisfaction survey included 5-point Likert scale items related to perceived competence (e.g., "I thought I performed well on the hands-on lab") and enjoyment (e.g. "I enjoyed the hands-on lab very much"). Fig. 7 illustrates students' satisfaction.

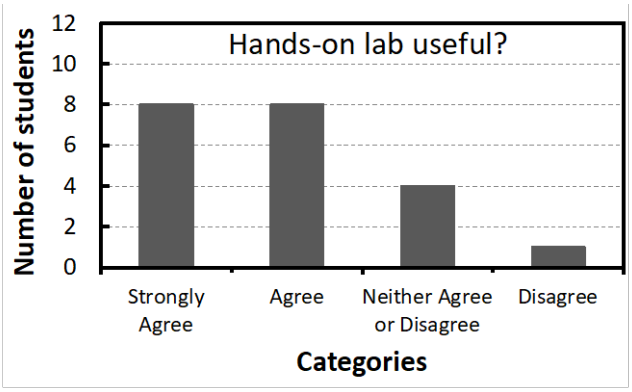

Fig. 7. Did you consider this hands-on lab useful?

Moreover, results revealed a positive correlation between students' reported effort and learning gains (r = 0.67) and a significant correlation between students' perceived competence and enjoyment (r = .832, p < .01). Fig. 8 demonstrates students' motivations in learning more about CPS including the areas of IoT and industry 4.0. Students also demonstrated a high interest in learning more about CAN-bus (*Mean* = 3.9, *SD* = 1.3, *Min* = 1, *Max* = 5). These results suggest that cybersecurity training in CPS should consider planning labs that focus on application areas based on students' interests.

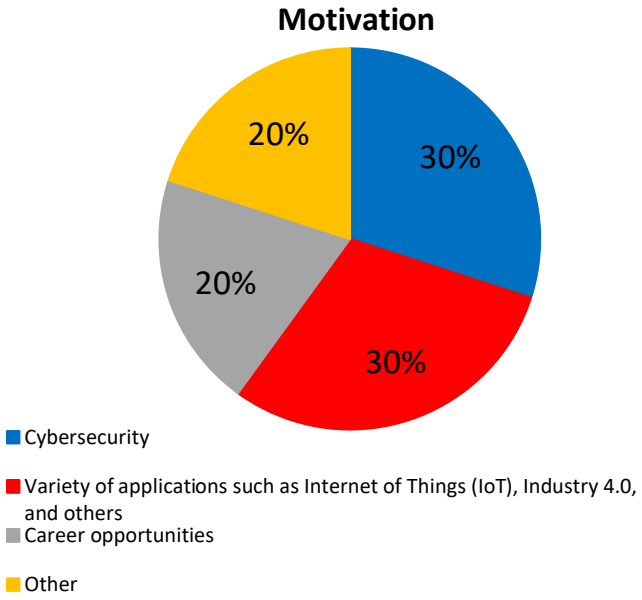

Fig. 8. Motivation to learn CPS concepts

Further, students reported that the hands-on lab helped them learn about CAN-bus (*Mean* = 4, SD = 1.0, *Min* = 2, *Max* = 3). Fig. 9 indicates the high level of enjoyment on the CPS/CPVS hands-on lab series (*Mean* = 4.26, *SD* = 1.2, *Min* = 1, *Max* = 5).

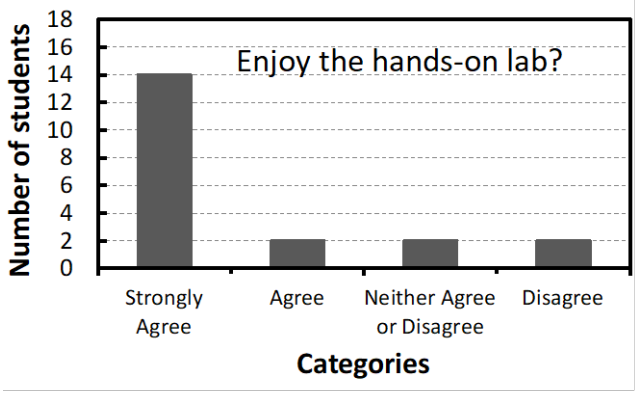

Fig. 9. Did you enjoy solving the hands-on lab?

In general, students' feedback was positive and the hands-on lab exercises helped them understand concepts of CPS and CPVS security. The practical learning method showed an effective way to help students catch current security trends.

VII. Conclusion and Future Works

Hands-on labs provided an effective learning method for students to study cyber-physical security concepts and tackle real-world technical issues via active learning. Students demonstrated positive feedback, and the majority of them like to learn more about security vulnerabilities of cyber-physical system applications and vehicle control. In addition, they gained practical skills through actively participating in hands-on exercises and familiarizing themselves with tools and equipment. Feedbacks from our preliminary surveys are promising.

Based on the results of this study, we will revise two items in the pretest and one item in the post-assessment form that had a difficult index above .86. Also, we plan to design an additional hands-on lab about cybersecurity challenges in automobile systems. In order to maximize student learning outcomes, we are developing an online course to provide additional resources on how to use tools and techniques to exploit automobile vulnerabilities and security issues in the CPS and CPVS field. The modules will cover a diverse set of applications of CPS, videos, and practical activities such as quizzes. Future research will integrate usability testing and validation of the effectiveness of the hands-on labs on learning gains and students' interests. Using our hands-on labs for the projects allows the students to use their creativity to create solutions for real-world problems including via blockchain technology and other disruptive technologies [27], [28]. In addition, the approach securing CPS by integrating Artificial Intelligence (AI) and blockchain [29] will be a practical approach. We will consider to develop some hands-on labs in those fields as well.

ACKNOWLEDGEMENT

This research is based upon work supported in part by the National Science Foundation under Grant CNS-2101161 and CNS-2131164; and Department of Justice DOJ 2019-DF-BX-K001. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation and Department of Justice.

REFERENCES

[1] Christopher Greer, Martin Burns, David Wollman, and Edward Griffor. Cyber-physical systems and internet of things, 2019.

[2] Abraham Peedikayil Kuruvila, Ioannis Zografopoulos, Kanad Basu, and Charalambos Konstantinou. Hardware-assisted detection of firmware attacks in inverter-based cyberphysical microgrids. *arXiv preprint arXiv:2009.07691*, 2020.

[3] Daniel Alejandro Rossit, Fernando Tohme´, and Mariano Frutos. Production planning and scheduling in cyber-physical production systems: a review. *International journal of computer integrated manufa*cturing, 32(4-5):385–395, 2019.

[4] Chen Lv, Yang Xing, Junzhi Zhang, and Dongpu Cao. Cyber-physical vehicle systems: Methodology and applications. *Synthesis Lectures on Advances in Automotive Technologies*, 4(1):1–85, 2020.

[5] Jiajia Liu, Shubin Zhang, Wen Sun, and Yongpeng Shi. In-vehicle network attacks and countermeasures: Challenges and future directions. *IEEE Network*, 31(5):50–58, 2017.

[6] Aurelien Francillon, Boris Danev, and Srdjan Capkun. Relay attacks on passive keyless entry and start systems in modern cars. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*. Eidgenossische Technische Hochschule Zurich, Department of Computer Science, 2011.

[7] Charlie Miller and Chris Valasek. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA*, 2015:91, 2015.

[8] National Vulnerability Database. Common vulnerabilities and exposures. cve-2019-13582, 2019.

[9] Daniel Noyes. Cyber security testing and training programs for industrial control systems. Technical report, Idaho National Laboratory (INL), 2012.

[10] InfoSec. Scada/ics security training boot camp, 2020.

[11] Washington Community College. Css 285: Essentials of automotive penetration testing, 2021.

[12] Heiner Lasi, Peter Fettke, Hans-Georg Kemper, Thomas Feld, and Michael Hoffmann. Industry 4.0. *Business & information systems engineering*, 6(4):239–242, 2014.

[13] Lihui Wang and Xi Vincent Wang. Latest advancement in cps and iot applications. *In Cloud-Based Cyber-Physical Systems in Manufacturing*, pages 33–61. Springer, 2018.

[14] Wenliang Du. Seed labs: Using hands-on lab exercises for computer security education. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, pages 704–704, 2015.

[15] Charalambos Konstantinou. Cyber-physical systems security education through hands-on lab exercises. *IEEE Design & Test*, 37(6):47–55, 2020.

[16] Society of Automotive Engineers. Introduction to car hacking with canbus c1857, 2020.

[17] Quarkslab. Training — practical car hacking, 2020.

[18] OpenGarages. Icsim: Instrument cluster simulator for socketcan, 2017.

[19] Alvaro Cardenas and Santa Cruz. Cyber-physical systems security knowledge area. *The Cyber Security Body Of Knowledge (cybok)*, 2019.

[20] Corey Thuen. Commonalities in vehicle vulnerabilities, 2016.

[21] Hyunguk Yoo and Irfan Ahmed. Control logic injection attacks on industrial control systems. In *IFIP International Conference on ICT Systems Security and Privacy Protection*, pages 33–48. Springer, 2019.

[22] Charles Barron Kirby and Bryson Payne. Automated reverse engineering of automotive can bus controls. 2019.

[23] Offensive Security. Penetration testing and ethical hacking linux distribution, Mar 2013.

[24] Collin Kidder.

[25] Wireshark. Wireshark, 1998.

[26] Linux CAN. Can-utils, Jun 2020.