Authentic Learning on DevOps Security with Labware: Git Hooks To Facilitate Automated Security Static Analysis

Md Mostafizur Rahman[1], Md Abdul Barek[2], Mst Shapna Akter[3], Abm Kamrul Islam Riad[4]
Md Abdur Rahman[5], Hossain Shahriar[6], Akond Rahman[7], Fan Wu[8]
[1]Deaprtment of Information Technology, University of West Florida, Florida, USA
[2345]Department of Intelligent Systems and Robotics, University of West Florida, Florida, USA
[6]Center of Cyber Security, University of West Florida, Florida, USA
[7]Dept. of Computer Science and Software Engineering, Auburn University, USA
[8]Tuskegee University, Alabama, USA
md.mostafizur.rn@gmail.com, mb381@students.uwf.edu, jannatul.shapna99@gmail.com
ai62@students.uwf.edu, mr252@students.uwf.edu, hshahria2012@gmail.com, akond@auburn.edu, fwu@tuskegee.edu

***Abstract*—This paper presents an innovative approach to DevOps security education, addressing the dynamic landscape of cybersecurity threats. We propose a student-centered learning methodology by developing comprehensive hands-on learning modules. Specifically, we introduce labware modules designed to automate static security analysis, empowering learners to identify known vulnerabilities efficiently. These modules offer a structured learning experience with pre-lab, hands-on, and post-lab sections, guiding students through DevOps concepts and security challenges. In this paper, we introduce hands-on learning modules that familiarize students with recognizing known security flaws through the application of Git Hooks. Through practical exercises with real-world code examples containing security flaws, students gain proficiency in detecting vulnerabilities using relevant tools. Initial evaluations conducted across educational institutions indicate that these hands-on modules foster student interest in software security and cybersecurity and equip them with practical skills to address DevOps security vulnerabilities.**

***Index Terms*—DevOps security education, vulnerabilities, Git, Git Hooks, security vulnerabilities, authentic learning.**

I. Introduction

In modern educational debate, hands-on learning has emerged as an essential instructional paradigm, incorporating learner-centric approaches and collaborative engagement to allow better comprehension and skill acquisition [1]. Based on experiential learning and real-world application, this teaching approach is crucial for cybersecurity education, as threats are constantly changing and becoming more complex, which calls for dynamic and engaging learning environments [2]. This paper endeavors to present a comprehensive framework for hands-on learning specifically tailored to address the difficulties of DevOps security. Combining software development with IT operations, or DevOps, has transformed modern software engineering methods but also brings a unique set of security risks that require expertise and specific training [3]. Although DevOps is becoming increasingly widely recognized in the technology world, there is still a need for more prominent educational resources regarding DevOps security, which prevents the effective delivery of educational material in this area. To bridge this gap, we propose a series of hands-on learning courses that are precisely intended to engage students in experiential learning about DevOps security. These modules are carefully designed to include a pre-lab orientation, hands-on tasks, and post-lab reflections, resulting in a multidimensional and iterative learning experience [4]. To have a visual reflection, follow Figure 1.

Our approach, which employs authentic learning concepts, aims to provide students with the skills to detect and mitigate security vulnerabilities inside the DevOps framework efficiently. The motivation for using hands-on learning is that it promotes deeper understanding and proficiency through direct engagement with real-world problems. By engaging students in practical tasks, hands-on learning improves active learning and fosters a sense of efficacy and mastery, ultimately increasing knowledge acquisition and retention [5]. In addition, incorporating authentic learning principles emphasizes the importance and practicality of skills learned in real-world scenarios, improving the transferability and usefulness of acquired knowledge [6].

Git is a distributed version control system and an integral part of modern software development, transforming the way teams collaborate, monitor changes, and maintain code bases [7]. Git, created by Linus Torvalds in 2005, has become widely used in the software industry, providing developers with a solid and versatile framework for version control [8]. Furthermore, Git hooks also play a crucial role in improving security within the development process [9]. Git hooks serve as powerful tools in the hands of developers, offering them the ability to automate and customize various aspects of the version control process. These hooks are scripts that Git executes before or after events, such as committing changes, merging branches, or pushing commits to a remote repository. By

defining these scripts, developers can enforce specific workflows, coding standards, or customized checks according to the requirements of their project [10]. For instance, pre-commit hooks can be utilized to run code linting, unit tests, or other validations to ensure that only clean and properly formatted code is committed. Similarly, post-commit hooks can trigger actions such as sending notifications, updating documentation, ensuring the test suite, or deploying the application to a staging environment. Check the table I to see a list of hooks for the client and server sides. Also, git hooks can trigger automated rollbacks or alerts if security checks fail, preventing potentially harmful code from being deployed or moving further along in the CI/CD pipeline until the issues are addressed. Git hooks are versatile and adaptable, enabling teams to integrate seamlessly with their existing development workflows and tools. Whether it's enforcing a consistent commit message format, integrating with continuous integration (CI) pipelines, or automating release processes, Git hooks empower developers to streamline their workflow, maintain code quality, and enhance collaboration within the team [11].

Furthermore, Git hooks play a crucial role in improving security within the development process. They can be utilized to enforce security best practices, such as scanning code for vulnerabilities before it's committed or ensuring that sensitive information like credentials for API keys is not inadvertently included in the repository. Pre-commit hooks can be configured to run security checks using static code analysis tools or vulnerability scanners, helping to identify and eliminate potential security threats early in the development cycle. Additionally, hooks can be used to enforce access controls, ensuring that only authorized users can push changes to certain branches or repositories. By integrating security checks into the Git workflow through hooks, development teams can proactively mitigate risks, strengthen code integrity, and protect against potential security breaches. This proactive approach to security not only protects the codebase but also fosters a culture of security awareness and responsibility among team members [12].

Our initiative includes the development of ten distinct learning modules covering a variety of DevOps and cybersecurity topics. Each module is thoughtfully prepared to create a scaffolded learning experience that starts with fundamental principles and ends with sophisticated applications and optimizations. Furthermore, our approach is underscored by its accessibility and inclusivity, with all learning materials openly available and adaptable to diverse educational contexts. This paper seeks to articulate a robust framework for hands-on learning in DevOps security, underpinned by authentic learning principles and academic innovation. We hope to build a team of cybersecurity experts with the necessary knowledge to navigate the complex environment of DevOps security confidently and effectively by spreading accessible and engaging instructional resources.

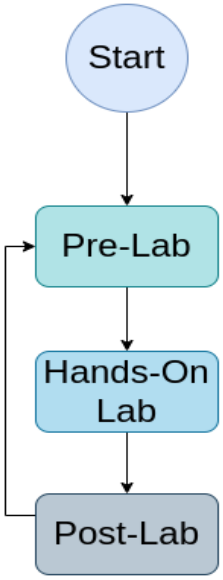


Fig. 1: Steps of the Labware.

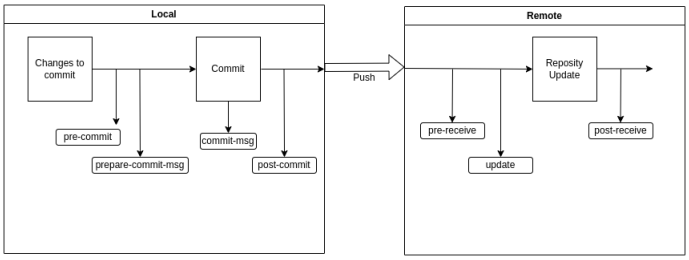


Fig. 2: Types of Git Hooks

II. Related work

According to our literature review, several studies have used case studies, project-based learning, and authentic learning methodologies in cybersecurity, software engineering, and some other engineering fields like mechanical engineering,

Bai et al. [13] present a project-based Software Engineering (SE) course framework to enhance practical training. They advocate for diversified project choices to engage students, collaborating with industry partners to design varied projects. Addressing challenges in project management and evaluation, they introduce a DevOps platform based on GitLab, integrating continuous integration and testing tools. Over four years, the platform showed promising results in improving SE education, with over 500 students and 130 project teams involved. This work highlights the importance of agile methodologies and DevOps practices in the enhancement of project-based SE courses. Frank et al. [14] investigate freshmen students' impressions of an introductory Project-Based Learning (PBL) course in mechanical engineering. The course "A creative introduction to mechanical engineering," consists of student teams working on mini-projects. The study uses qualitative

methods such as interviews, observations, and report analysis to investigate students' perspectives on course objectives, instructor responsibilities, PBL characteristics and advantages, and consequences for future engineers. Huang et al. [15] used project-based learning to integrate machine learning technologies into the engineering curriculum. In the context of software engineering research, Garg et al. [16] contrasted lecture-based versus case study-based methodologies. Deng et al. [17] designed a case-study-based learning method for machine learning-based hands-on laboratory work in cybersecurity. Blanken et al. [18] used case-study-based modules to engage learners in ethical concerns related to cybersecurity. Similarly, Lo et al. [19] carried out an authentic learning project in which students had to solve cybersecurity problems that existed in the real world. Frontera et al. [20] used a project-based learning framework to assess cyber threats in a cyber-physical system. Faruk et al. [21] concentrated on incorporating authentic learning in machine learning for cybersecurity, creating learning modules that included hands-on exercises to tackle security issues. Finally, Qian et al. [22] adopted an authentic learning approach to secure software development, providing students with hands-on lab lessons on secure mobile app development.

Although these studies offer valuable information, it is important to note that none of them explicitly focused on DevOps security education. While case study-based, project-based, and authentic learning methodologies have been applied across diverse fields, there remains a void in the literature regarding their implementation within the realm of DevOps security education. Hence, our research endeavors to bridge this gap by creating authentic learning modules designed specifically for DevOps security education.

III. Lab Design

The lab framework is divided into three sections: pre-lab, hands-on, and post-lab, all of which have been rigorously planned to ensure a productive learning trip. Comprehensive instructions are gathered and maintained on a dedicated Google site to ensure that learners can access and understand them. Utilizing Git for coding and resource management allows for smooth collaboration and precise version control, which improves the entire learning experience. The pre-lab part provides learners with foundational knowledge relevant to the topic. The hands-on lab part provides in-depth insights and practical applications, promoting greater understanding and proficiency. Finally, the post-lab portion encourages students to explore the topic more deeply, allowing for more extensive research and comprehension.

A. *Pre-lab*

The pre-lab section introduces Git Hooks in the DevOps security environment. Learners gain fundamental knowledge of Git Hooks, including their purpose, functionality, and importance in supporting secure coding practices in DevOps contexts. An overview of Git Hooks, including various types (such as pre-commit, post-commit, and pre-push), as well as their importance in automating security checks and enforcing coding standards. Learners may also investigate common security risks that Git Hooks assists in minimizing, such as preventing sensitive data exposure or enforcing code review procedures.

Hook	Command	Param	Remarks
applypatch-msg	git-am	1	Edit message file
pre-applypatch	git-am	0	After patching and before commit
pre-commit	git-commit	0	Before commit workflow starts
pre-marge	git-merge	0	After merge and before commit message
pre-commit-msg	git-commit	1-3	After preparing commit-msg and before opening editor
pre-receive	git-receive-pack	0	Before updating refs on remote.
update	git-recive-pack	3	before updating each refs on remote
proc-receive	git-receive-pack	0	Execute once for the receive operation
post-update	git-receive-pack	variable	After all refs are updated
push-to-checkout	git-receive-pack	1	Merge with any server-side edits on checked out branch.

TABLE I: List of GitHooks [23]
**Server side hooks
** Not all hooks are listed here.

B. *Hands-on lab*

In the hands-on lab part, students explore further into the practical applications of Git Hooks to improve DevOps security. Students learn how to install and configure Git Hooks in their development processes through guided exercises and real-world scenarios. This could include creating pre-commit hooks to automate code linting, performing static code analysis, or executing security scans to uncover vulnerabilities before committing code to the repository. Learners can also examine advanced subjects such as modifying Git hooks to interface with third-party security tools or enforcing secure coding practices in DevOps contexts.

C. *Post-lab*

The post-lab section encourages students to reflect on their hands-on experience with Git Hooks and explore the subject. This section may include discussions about the best practices for integrating Git Hooks into continuous integration/continuous deployment (CI/CD) pipelines, improving security checks for efficiency, and addressing typical implementation issues and limits. Learners may also look at additional resources, such as documentation, tutorials, or case studies, to gain a better understanding of Git Hooks and its role in DevOps security. Finally, the post-lab part may encourage students to brainstorm and propose ideas for incorporating Git Hooks into their DevOps projects or environments.

IV. STUDENTS SURVEY

In the spring of 2024, we deployed the module in three schools. A preliminary poll was conducted at the University of West Florida, Auburn University, and Tuskegee University. Surveys are offered in both quantitative and qualitative formats. We did a pre-lab and post-lab survey by asking numerous questions.

Prelab Survey: Among 20 students surveyed, most of the students (50%) were in the age group between 18 to 25. Some of them (25%) were between 26 and 35, 20% of them were between 36 and 45, and only one student was in the age group of 46-55.

TABLE II: Display the responses of students on age group

#	Answer	%	Count
1	$<$ 18 years	0%	0
2	Between 18 and 25 years	50%	10
3	Between 26 and 35 years	25%	5
4	Between 36 and 45 years	20%	4
5	Between 46 and 55 years	5%	1
6	$>$55 years	0%	0

We asked the students to identify their level of education in DevOps Security and according to the pre-lab survey, a substantial proportion lacked experience in DevOps Security coursework, with 8 of the total respondents indicating no prior exposure. In contrast, the majority of students (19) reported previous experience in software engineering. Furthermore, a sizable proportion (17 students) reported having received software cybersecurity training.

In terms of programming proficiency, the poll found that the majority of respondents had moderate to expert-level skills in at least one programming language. Specifically, 18 pupils declared expertise in Python, while 15 claimed experience in programming in C. In contrast, only one student reported knowledge of Ruby programming. These findings indicate that many students have experience with numerous technological domains, such as software engineering and cybersecurity. Still, there is a significant gap in DevOps Security education within the questioned series. Furthermore, the predominance of programming experience demonstrates the student body's technical proficiency. See the Figure 3

We have asked students another important question regarding their learning preferences, and there were five different choices, which are Strongly Disagree, Disagree, Neutral, Agree, and Strongly Agree.
See the table below for a better understanding:

The above table indicates that most students prefer interactive learning methodologies, particularly favoring hands-on lab exercises and working through examples. Specifically, 15 students expressed comfort with hands-on lab activities,

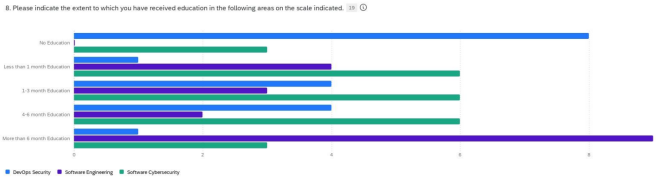
TABLE III: Display the responses of students on learning preferences

#	Question	S. Disagree	Disagree	Neutral	Agree	S. Agree
1	I learn better by engaging hands-on lab	0	0	0	5	15
2	I learn better by listening to lectures	1	5	3	8	3
3	I learn better by personally doing or working through examples	0	0	0	4	16
4	I learn better by reading the materials on my own	1	4	2	8	5
5	I learn better by having a learning/tutorial system that provides feedback	0	0	2	7	11

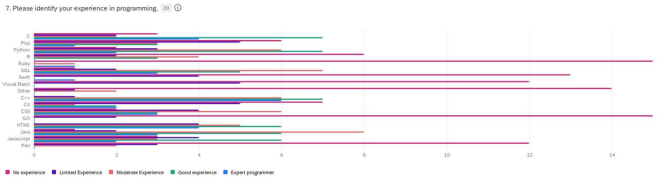
while 16 students reported a preference for engaging with examples to facilitate their learning process. Additionally, 11 students indicated a preference for tutorials or systems that offer feedback as part of their learning experience. These results underscore the importance of incorporating interactive elements and feedback mechanisms into educational materials to enhance student engagement and comprehension. See the Figure 4.

Post-Test Survey: In the post-lab survey, we found that a total of 9 students completed the post-lab survey by completing numerous questions. We asked the students about the benefits of the hands-on lab and if the prelab helped them understand the topic properly. We completed the survey after the students were involved in the hands-on module. From the survey result, we found that authentic learning procedures in the area of DevOps Security are mostly positive from student perspectives. For most of the statements had 5 options to choose from, they are: Strongly Disagree, Disagree, Neutral, Agree and, Strongly Agree.

For the question, "I like being able to work with the secure DevOps hands-on material," 78% of students strongly agreed, and 22% agreed. None of them disagreed. For the question, "The outline tutorial in pre-lab help me learn more on the topic," 56% of students strongly agreed, and 44% agreed. We also asked students about the hands-on materials and procedures. For the question, "The hands-on labs help me understand better on DevOps Security", 56% of students strongly agreed, and 44% agreed. For the question, " The hands-on labs help my learning experience on secure DevOps coding and best practices.", 67% of students strongly agreed, and 33% agreed. None of them disagreed. For the statement, " The real-world relevant applications engage my learning on cybersecurity.", we have found 67% of students are strongly agreed and 33% of them are agreed. Students' response to the secure DevOps hands-on materials and real-world applicable applications is overwhelmingly favorable, indicating that these



(a) Level Of DevOps Security Education



(b) Programming Language Experienc

Fig. 3: Figure (a), (b) displays the responses from the pre-survey questions.

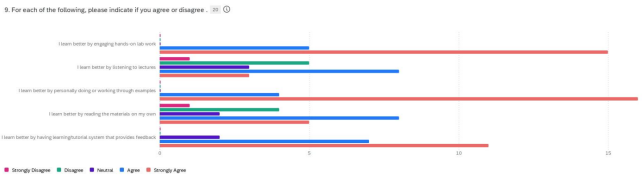


Fig. 4: Student Leaning Preferences

tools are useful and have an impact. As shown in Figure 6, a large majority of participants recognized the benefit of the secure DevOps materials, particularly in improving their understanding of the subject topic. Notably, more participants agreed or strongly agreed that the hands-on labs and tutorials were valuable to their learning journey, particularly in terms of secure DevOps coding best practices and understanding DevOps security principles. This affirmation emphasizes the effectiveness of interactive and practical learning methodologies in developing comprehension and expertise in DevOps security principles. For information follow the Figure 5 and Figure 6

V. CONCLUSION

This educational labware addresses the challenges and requirements of DevOps security, specifically facilitating automated static security analysis using Git Hooks. It uses dynamic and engaging learning methods to fill the gap in educational resources and practical learning environments. The pre-lab, hands-on, and post-lab allow students to understand a deeper concept of DevOps security, with a proper explanation of coding practices and further instructions for doing more experiments with real-life problems. Early feedback suggests that students understand the basics and actively improve their skills through interactive lab sessions.

ACKNOWLEDGEMENT

This research is funded by the National Science Foundation through grants NSF Award #2310179, #2209637, #2421324, and #1946442. The views, conclusions, and recommendations presented in this material are those of the authors and do not necessarily represent the perspectives of the National Science Foundation.

REFERENCES

[1] M. S. Akter, H. Shahriar, J. Rodriguez-Cardenas, M. M. Rahman, A. Rahman, and F. Wu, "Teaching devops security education with hands-on labware: Automated detection of security weakness in python," in *Proceedings of the ISCAP Conference ISSN*, vol. 2473, p. 4901, 2023.

[2] M. M. Rahman, A. S. Arshi, M. M. Hasan, S. F. Mishu, H. Shahriar, and F. Wu, "Security risk and attacks in ai: A survey of security and privacy," in *2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 1834–1839, IEEE, 2023.

[3] M. S. Akter, H. Shahriar, S. I. Ahamed, K. D. Gupta, M. Rahman, A. Mohamed, M. Rahman, A. Rahman, and F. Wu, "Case study-based approach of quantum machine learning in cybersecurity: Quantum support vector machine for malware classification and protection," *arXiv preprint arXiv:2306.00284*, 2023.

[4] A. Rahman, S. I. Shamim, H. Shahriar, and F. Wu, "Can we use authentic learning to educate students about secure infrastructure as code development?," in *Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 2*, pp. 631–631, 2022.

[5] S. Goldin-Meadow, "Taking a hands-on approach to learning," *Policy insights from the Behavioral and Brain Sciences*, vol. 5, no. 2, pp. 163–170, 2018.

[6] K. Roach, E. Tilley, and J. Mitchell, "How authentic does authentic learning have to be?," *Higher Education Pedagogies*, vol. 3, no. 1, pp. 495–509, 2018.

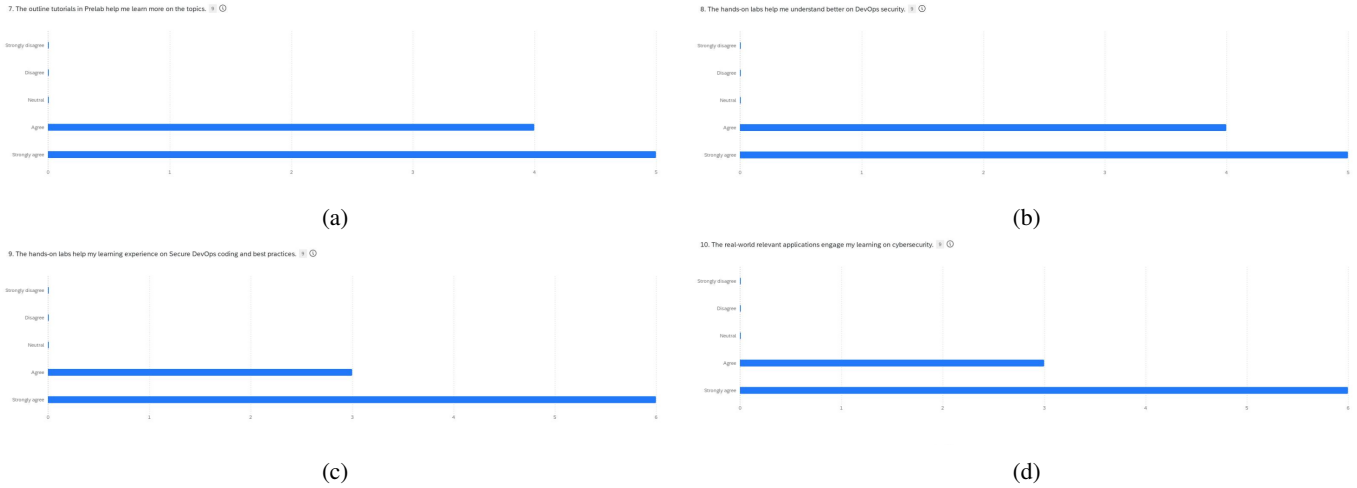
[7] D. Spinellis, "Git," *IEEE software*, vol. 29, no. 3, pp. 100–101, 2012.

[8] D. M. German, B. Adams, and A. E. Hassan, "A dataset of the activity of the git super-repository of linux in 2012," in *2015 IEEE/ACM 12th Working Conference on Mining Software Repositories*, pp. 470–473, IEEE, 2015.

[9] G. Benedetti, L. Verderame, and A. Merlo, "Automatic security assessment of github actions workflows," in *Proceedings of the 2022 ACM Workshop on Software Supply Chain Offensive Research and Ecosystem Defenses*, pp. 37–45, 2022.

[10] J. D. Blischak, E. R. Davenport, and G. Wilson, "A quick introduction to version control with git and github," *PLoS computational biology*, vol. 12, no. 1, p. e1004668, 2016.

[11] Y. Perez-Riverol, L. Gatto, R. Wang, T. Sachsenberg,



(a) (b)

(c) (d)

Fig. 5: Figure (a), (b), (c), and (d) displays the responses from the post-survey questions.

(a) (b)

Fig. 6: Figure (a), (b), and (c) display the responses from the post-survey questions.

J. Uszkoreit, F. d. V. Leprevost, C. Fufezan, T. Ternent, S. J. Eglen, D. S. Katz, *et al.*, "Ten simple rules for taking advantage of git and github," 2016.

[12] A. Rahman, H. Shahriar, and D. B. Bose, "How do students feel about automated security static analysis exercises?," in *2021 IEEE Frontiers in Education Conference (FIE)*, pp. 1–4, IEEE, 2021.

[13] X. Bai, D. Pei, M. Li, and S. Li, "The devops lab platform for managing diversified projects in educating agile software engineering," in *2018 IEEE Frontiers in Education Conference (FIE)*, pp. 1–5, 2018.

[14] M. Frank, I. Lavy, and D. Elata, "Implementing the project-based learning approach in an academic engineering course," *International Journal of Technology and Design Education*, vol. 13, pp. 273–288, Oct 2003.

[15] L. Huang, "Integrating machine learning to undergraduate engineering curricula through project-based learning," in *2019 IEEE Frontiers in Education Conference (FIE)*, pp. 1–4, IEEE, 2019.

[16] K. Garg and V. Varma, "A study of the effectiveness of case study approach in software engineering education," in *20th Conference on Software Engineering Education & Training (CSEET'07)*, pp. 309–316, IEEE, 2007.

[17] Y. Deng, D. Lu, D. Huang, C.-J. Chung, and F. Lin, "Knowledge graph based learning guidance for cybersecurity hands-on labs," in *Proceedings of the ACM conference on global computing education*, pp. 194–200, 2019.

[18] J. Blanken-Webb, I. Palmer, S.-E. Deshaies, N. C. Burbules, R. H. Campbell, and M. Bashir, "A case study-based cybersecurity ethics curriculum," in *2018 USENIX Workshop on Advances in Security Education (ASE 18)*, 2018.

[19] D. C.-T. Lo, K. Qian, W. Chen, H. Shahriar, and V. Clincy, "Authentic learning in network and security with portable labs," in *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*, pp. 1–5, IEEE, 2014.

[20] P. J. Frontera and E. J. Rodríguez-Seda, "Network attacks on cyber–physical systems project-based learning activity," *IEEE Transactions on Education*, vol. 64, no. 2, pp. 110–116, 2020.

[21] M. J. H. Faruk, M. Masum, H. Shahriar, K. Qian, and D. Lo, "Authentic learning of machine learning to ransomware detection and prevention," in *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 442–443, IEEE, 2022.

[22] K. Qian, D. Lo, R. Parizi, F. Wu, E. Agu, and B.-T. Chu, "Authentic learning secure software development (ssd) in computing education," in *2018 IEEE Frontiers in Education Conference (FIE)*, pp. 1–9, IEEE, 2018.

[23] Arvindpdmn, "Git hooks," Dec 2022.